Numéro d'ordre :

THÈSE

présentée à

L'UNIVERSITÉ DU LITTORAL DE CALAIS

pour obtenir le titre de

DOCTEUR EN MATHEMATIQUES

par

Laurent SMOCH

Application des méthodes de Krylov à la résolution de systèmes singuliers

le 19 Novembre 1999, devant la commission d'examen :

Président :	Claude BREZINSKI,	Professeur, USTL, Lille
Rapporteurs :	Bernd FISCHER,	Professeur, Université de Lübeck, Lübeck
	Gérard MEURANT,	Adjoint Chef de département CEA
Examinateurs :	Marc PREVOST,	Professeur, Université du Littoral, Calais
	Hassane SADOK,	Professeur, Université du littoral, Calais et directeur de thèse

Laboratoire de Mathématiques Pures et Appliquées Joseph Liouville
Université du Littoral, zone universitaire de la Mi-Voix, bâtiment H. Poincarré
50, rue F. Buisson, BP 699, F-62228 Calais cedex

Numéro d'ordre :

THÈSE

présentée à

L'UNIVERSITÉ DU LITTORAL DE CALAIS

pour obtenir le titre de

DOCTEUR EN MATHEMATIQUES

par

Laurent SMOCH

Application des méthodes de Krylov à la résolution de systèmes singuliers

le 19 Novembre 1999, devant la commission d'examen :

Président :	Claude BREZINSKI,	Professeur, USTL, Lille
Rapporteurs :	Bernd FISCHER,	Professeur, Université de Lübeck, Lübeck
	Gérard MEURANT,	Adjoint Chef de département CEA
Examinateurs :	Marc PREVOST,	Professeur, Université du Littoral, Calais
	Hassane SADOK,	Professeur, Université du littoral, Calais et directeur de thèse

Laboratoire de Mathématiques Pures et Appliquées Joseph Liouville
Université du Littoral, zone universitaire de la Mi-Voix, bâtiment H. Poincarré
50, rue F. Buisson, BP 699, F-62228 Calais cedex

Remerciements

Ce travail a été réalisé au sein du laboratoire de Mathématiques Pures et Appliquées à l'université du Littoral sous la direction du professeur Hassane Sadok. Je le remercie d'avoir accepté d'être mon directeur de thèse pendant ces trois années. Qu'il trouve également toute l'expression de ma reconnaissance pour avoir accepté de diriger ma recherche, pour m'avoir guidé et soutenu mais surtout pour m'avoir consacré son temps durant ces trois années.
Je tiens à exprimer ma gratitude au professeur Claude Brezinski de l'Université des Sciences et Techniques de Lille qui me fait l'honneur de présider ce jury.
Je remercie le professeur Bernd Fischer de l'Université de Lübeck ainsi que Monsieur Gérard Meurant, adjoint chef de département CEA à Bruyères-le-Chatel qui ont accepté d'être rapporteurs. Je suis également honoré de la présence de Marc Prévost, professeur à l'Université du Littoral en tant qu'examinateur.

Je remercie les membres du Laboratoire de Mathématiques Pures et Appliquées de l'Université du Littoral, ainsi que ceux du Laboratoire d'Analyse Numérique et d'Optimisation de Lille. Qu'ils soient assurés de ma gratitude pour leurs conseils et leur gentillesse de tous les jours.
Je souhaite exprimer ma reconnaissance à Mohammed Heyouni pour son soutien moral, ses conseils avisés et son éternelle gentillesse pendant ces trois années. Je remercie également Azeddine Essai, Jean-Paul Chehab, Caroline Le Calvez et Denis Vekemans pour leurs encouragements et leur précieuse compagnie.

Je remercie Teddy Zoonekynd et Eric Massé pour leur soutien de tous les jours à l'Université Lamartine où j'ai eu le plaisir d'enseigner pendant deux années.

Que Philippe Marion et Olivier Pichon trouvent l'expression de ma reconnaissance pour la disponibilité et la serviabilité dont ils ont fait preuve à mon égard tout au long de cette thèse.

J'adresse également un grand merci à François, Gabriel, Olivier, Catherine et Sophie pour leur soutien durant ma thèse ainsi que les nombreuses années qui l'ont précédée.

Enfin, je voudrais remercier tout particulièrement ma mère, ma soeur et Audrey pour leur confiance et leur dévouement à mon égard et qui ont contribué pour beaucoup à la réalisation et à l'aboutissement de ce travail.

Résumé

Nous étudions dans cette thèse les méthodes de sous-espaces de Krylov appliquées à la résolution des systèmes singuliers consistants. Les méthodes utilisées, qu'elles soient de Galerkin ou de semi-minimisation du résidu, construites à partir du processus de Hessenberg généralisé, font intervenir de nouveaux résultats quant à la détermination d'une solution de Krylov.

Dans un premier temps, on montre dans le cadre de l'arithmétique exacte et de la précision finie que la détermination d'une solution de Krylov dépend du vecteur initial utilisé dans la méthode et, plus précisément, de sa décomposition sur les sous-espaces caractéristiques associés à la matrice du système étudié.

Des précisions sont également apportées au niveau des dégénérescences rencontrées par la méthode GMRES ainsi que sur sa version redémarrée.

De nouvelles considérations sont introduites quant au traitement numérique de quelques équations aux dérivées partielles non linéaires dépendant d'un paramètre. Dans certains cas, elles font intervenir des systèmes singuliers dont la matrice admet des blocs de Jordan associés à la valeur propre nulle de dimension maximale 1, condition suffisante à l'obtention d'une solution de Krylov.
Tous ces exemples pratiques et leur traitement numérique montrent par conséquent qu'il existe des alternatives aux dégénérescences rencontrées par les algorithmes utilisés.

Mots clés

Méthodes de sous-espaces de Krylov, processus de Hessenberg généralisé, matrice de Hessenberg, systèmes linéaires singuliers, dégénérescences, équations aux dérivées partielles, systèmes non linéaires, méthode de Newton, méthodes de continuation.

Abstract

In this thesis we study the Krylov subspace methods applied to the solution of consistent singular systems. The methods in use today which are Galerkin methods or minimal residual seminorm methods, built help to the Hessenberg generalized process, call in some new results about the determination of a Krylov solution.

We show first, that in exact arithmetic or in finite precision, the determination of a Krylov solution depends on the initial vector used in the method and more precisely, on its decomposition on the characteristic subspaces associated to the matrix of the studied system.

Some precisions are also given about the breakdowns occuring in the GMRES method and on its restarted version.

Some new considerations are brought about the numerical treatment of some nonlinear partial differential equations depending on a parameter. In some cases, they call in singular systems whose matrix admits some Jordan blocks associated to the zero eigenvalue of maximal size 1, which is a sufficient condition to obtain a Krylov solution.
All these practical examples and their numerical treatment prove therefore that there exists alternatives to the breakdowns occuring in the algorithms.

Key words

Krylov subspace methods, generalized Hessenberg process, Hessenberg matrix, singular linear systems, breakdowns, partial differential equations, nonlinear systems, Newton method, continuation methods.

Table des matières

Introduction générale

La plupart des techniques itératives pratiques existantes pour résoudre des grands systèmes linéaires du type

$$Ax = b$$

avec $A \in I\!R^{n\times n}$, x, $b \in I\!R^n$ utilisent un processus de projection. Ce processus représente un moyen simple d'extraire une approximation de la solution du système à partir d'un sous-espace. Les meilleures techniques disponibles sont fondées sur des processus de projection (obliques ou orthogonaux) sur des sous-espaces de Krylov. Ces sous-espaces particuliers sont engendrés par des vecteurs de la forme $p(A)v$ où p est un polynôme et v est un vecteur particulier de $I\!R^n$. Ces techniques approchent $A^{-1}b$ par $p(A)b$ où p sera considéré comme un polynôme convenable. Les méthodes de sous-espaces de Krylov sont des méthodes itératives et elles représentent une alternative aux méthodes directes. Ces dernières permettent de factoriser A en un produit de matrices généralement plus simples à utiliser, les factorisations les plus fréquemment employées étant de la forme LU ou QR. Malheureusement, elles posent un problème majeur : lorsque A est creuse, les matrices intervenant dans les produits peuvent nécessiter un stockage nettement plus important que celui de la matrice initiale A. Certaines techniques pallient à cette difficulté et diminuent le stockage et le coût algorithmique de la méthode. On préférera donc à ces méthodes les méthodes de sous-espaces de Krylov. Elles consistent plus précisément à projeter le système $Ax = b$ sur un sous-espace $\mathcal{K}_k$ de dimension k inférieure à la dimension n du système, orthogonalement à un second sous-espace $\mathcal{L}_k$ également de dimension k et appelé sous-espace des contraintes. La taille de ces deux sous-espaces augmente avec le nombre d'itérations. Outre la condition de projection, les méthodes vont se différencier par le choix de la base de $\mathcal{K}_k$. Cette base est élaborée à l'aide d'un processus général appelé processus de Hessenberg généralisé. Il généralise les processus d'Arnoldi, de Lanczos ainsi que le processus de Hessenberg. Les méthodes dépendant de ce processus sont classées selon la condition d'orthogonalité utilisée. La première classe est celle des méthodes **RO** (Résidus Orthogonaux) obtenue par imposition de la condition de Petrov-Galerkin. Elle englobe les méthodes FOM (Full Orthogonalization Method), BCG (Bi Conjugate Gradient) ainsi que la méthode de Hessenberg. La seconde classe est obtenue par imposition d'une condition de semi-minimisation et est désignée par les méthodes **RM** (Résidus Minimaux). On se consacrera dans cette classe à l'étude des méthodes GMRES (Generalized Minimal RESidual mehod), CMRH (Changing Minimal Residual method based on the Hessenberg reduction process) et QMR (Quasi Minimal Residual method). Le processus de Hessenberg généralisé combiné à la condition d'orthogonalité donne naissance à la méthode de Hessenberg généralisée. On s'intéressera dans cette thèse à l'étude de ces méthodes dans le cadre des systèmes singuliers. On précise également que l'on ne traitera pas, dans les parties théoriques de cette thèse, les systèmes singuliers ou quasi singuliers en arithmétique flottante.

Certaines équations aux dérivées partielles obtenues par discrétisation à l'aide de la méthode des différences finies ou la méthode des éléments finis peuvent faire intervenir des systèmes linéaires singuliers. Ils admettent selon leur consistance une infinité de solutions ou aucune solution. Les méthodes de sous-espaces de Krylov appliquées à des systèmes singuliers voient leur performance affectée dans le sens où elles peuvent échouer dans la détermination d'une solution. On verra par la suite que le vecteur initial utilisé lors du démarrage de la méthode utilisée revêt une importance capitale.
On s'intéressera particulièrement aux équations aux dérivées partielles non linéaires dépendant d'un paramètre. Leur discrétisation fait intervenir des branches de solutions de systèmes non linéaires avec paramètre régulières et singulières. Le calcul de ces branches en des points singuliers, qu'ils soient de retournement ou de bifurcation, pose problème. En effet, en de tels points, la matrice jacobienne du système intervenant lors de l'utilisation de la méthode de Newton est singulière dans le cas de la dimension finie et l'opérateur linéarisé présente une singularité dans le cas général. C'est pourquoi, dans le traitement numérique des problèmes de bifurcation, des méthodes spécifiques à la théorie de la bifurcation sont utilisées. La méthode numérique permettant d'obtenir la branche de solutions complète est appelée méthode de continuation ou de cheminement et fait partie des méthodes de type prédicteur-correcteur. Cette technique utilise en fait des informations concernant un segment de la courbe déja calculé dans le but d'obtenir une extrapolation convenable permettant d'approcher le segment suivant de la courbe. Par conséquent, un certain point de la courbe extrapolée est choisi comme point initial d'une itération de correction qui, on l'espère, convergera vers un point suivant de la courbe. On développe cette méthode au quatrième chapitre dans le cadre de problèmes d'équations aux dérivées partielles non linéaires dépendant d'un paramètre, comme c'est le cas, par exemple, pour le problème de Bratu.

Cette étude se divise en quatre chapitres.
Dans le premier chapitre, on rappelle le processus de Hessenberg généralisé ainsi que certaines de ses propriétés. On s'intéresse bien évidemment aux différents processus auxquels il donne lieu, en l'occurrence ceux d'Arnoldi, de Lanczos et de Hessenberg. Le

processus de Hessenberg généralisé est ensuite utilisé dans le cadre de systèmes singuliers et des résultats concernant l'efficacité de ce dernier sont présentés.

Le chapitre 2 présente une classification des méthodes de Krylov à l'aide des critères du produit scalaire et du sous-espace de projection utilisé. Les conditions de Petrov-Galerkin et de semi-minimisation du résidu sont détaillées. Elles permettent d'introduire les méthodes **RO** ET **RM** dans le cadre des systèmes singuliers. Leur efficacité est comparée à l'aide d'exemples numériques dans le cadre de l'arithmétique exacte et de la précision finie, tous les systèmes linéaires à résoudre étant consistants.

Dans le chapitre 3, on se consacre de manière exclusive à la méthode GMRES. Les résultats théoriques présentés font intervenir essentiellement les notions de rotations de Givens ainsi que de matrice pseudo-inverse. Des résultats de convergence sont également présentés, montrant l'importance capitale du choix du vecteur initial x_0 ainsi que les répercussions qu'il provoque dans l'algorithme. De plus, certains nouveaux résultats concernant le GMRES redémarré sont introduits.

Le dernier chapitre est consacré à la présentation de cas pratiques où la résolution de systèmes singuliers intervient. Il s'agit de certains problèmes d'équations aux dérivées partielles non linéaires dépendant d'un paramètre. Tous ces exemples sont analysés théoriquement et illustrés numériquement à l'aide de la méthode GMRES ainsi que de sa version redémarrée.

Chapitre 1

Le processus de Hessenberg généralisé dans le cadre des systèmes singuliers

La plupart des méthodes utilisées pour résoudre de manière efficace des systèmes d'équations linéaires sont des méthodes itératives. Elles se répartissent en deux classes principales : les méthodes stationnaires et les méthodes de sous-espaces de Krylov. Ces dernières construisent, à l'aide d'une approximation de départ, une suite de solutions approchées de la solution exacte du système considéré, le procédé utilisé étant le même à chaque itération.

La méthode de sous-espaces de Krylov la plus connue est la méthode du gradient conjugué (méthode CG) de Hestenes et Stiefel [33] créée en 1952. Cette méthode offre plusieurs avantages, en l'occurrence un coût de stockage peu élevé et un coût algorithmique par itération faible. La recherche de ces qualités par les scientifiques a donné le jour à de nombreuses méthodes qui présentent l'avantage de s'utiliser dans le cadre des systèmes linéaires non-symétriques. On peut citer de manière chronologique les méthodes BCG (Bi-Conjugate Gradient) [42, 24], FOM (Full Orthogonalization Method) [53], GMRES (Generalized Minimal RESidual method) [57], CGS (Conjugate Gradients Squared algorithm) [61], QMR (Quasi Minimal Residual method) [28], BiCGSTAB (Bi-Conjugate Gradient **Stab**ilized) [63] et enfin CMRH (Changing Minimal Residual method based on the Hessenberg reduction process) [58].

Ces méthodes sont toutes fondées sur un même processus, celui de Hessenberg généralisé, largement développé par Sadok [60]. Ce processus est utilisé pour la réduction d'une matrice de Krylov ([69] pp.32) dans le but de construire une base de Krylov et

une matrice de Hessenberg supérieure.
On présente tout d'abord dans ce chapitre la description générale du processus de Hessenberg généralisé, suivie des différents processus auxquels il donne lieu. Des propriétés sont présentées dans le but de comprendre les phénomènes qui se produisent lors de l'utilisation de ce processus dans le cadre de la résolution de systèmes singuliers. Des conditions nécessaires et suffisantes pour l'obtention de la non-singularité de la matrice de Hessenberg sont précisées, ceci en arithmétique exacte.

1.1 Le processus de Hessenberg généralisé

Une méthode de projection générale permettant de résoudre le système linéaire

$$Ax = b \tag{1.1}$$

avec $A \in I\!R^{n\times n}$ et $x, b \in I\!R^n$ est une méthode qui construit une solution approchée x_k de x à partir du sous-espace affine $x_0 + \mathcal{K}_k$ de dimension k, en imposant la condition de Petrov-Galerkin

$$b - Ax_k \perp \mathcal{L}_k \tag{1.2}$$

où $\mathcal{L}_k$ est un second sous-espace de dimension k appelé sous-espace des contraintes. Soit $x_0 \in I\!R^n$ un vecteur initial, on appellera résidu initial de (1.1), le vecteur $r_0 = b - Ax_0 \in I\!R^n$. Dans le cas où $\mathcal{K}_k = \mathcal{K}_k(A, r_0) \equiv vect\{r_0, Ar_0, \ldots, A^{k-1}r_0\}$ où $\mathcal{K}_k(A, r_0)$ sera appelé le sous-espace de Krylov associé à A et r_0, on parlera de méthode de sous-espaces de Krylov.
Soit x^* une solution de (1.1). Les méthodes de sous-espaces de Krylov consistent à corriger le vecteur x_0 à l'aide d'un vecteur z_k de $\mathcal{K}_k(A, r_0)$ tel que l'équation

$$A(x_0 + z_k) = b \tag{1.3}$$

soit vraie dans le sous-espace $\mathcal{L}_k^\perp$. Le vecteur $x_0 + z_k$ sera alors noté x_k.
Les diverses versions des méthodes de sous-espaces de Krylov proviennent des différents choix de $\mathcal{L}_k$. Lorsque $\mathcal{K}_k = \mathcal{L}_k$, on se trouve dans le cas d'une projection orthogonale. Dans le cas contraire, quand $\mathcal{K}_k \neq \mathcal{L}_k$, la projection est oblique.

Il est utile également d'introduire la notion de polynôme minimal associé à un vecteur.

Définition 1.1.1
On appelle polynôme minimal Π_A *de la matrice* A *associé au vecteur* r_0, *le polynôme de degré minimal vérifiant* $\Pi_A(A)r_0 = 0$.

D'un point de vue polynomial, $x_k = x_0 + p_k(A)r_0$ où p_k est un polynôme de degré strictement inférieur à k à coefficients réels. De ce fait, $r_k = (I_{n,n} - Ap_k(A))r_0$. Comme la convergence a lieu lorsque $x = x^*$, c'est-à-dire lorsque $r = b - Ax = 0$, le polynôme résiduel (le polynôme associé au résidu $1 - xp_k(x)$) est égal au polynôme minimal de A associé à r_0, à une constante multiplicative près.

On supposera par la suite que le polynôme minimal de A par rapport au vecteur r_0 est de degré m.

On dispose, à l'aide des considérations précédentes, du

Théorème 1.1.1
Le sous-espace de Krylov $\mathcal{K}_k(A, r_0)$ est le sous-espace formé par tous les vecteurs de la forme $p(A)r_0$ où p est un polynôme de degré strictement inférieur à k.

Ainsi, on a le

Théorème 1.1.2
Le sous-espace de Krylov $\mathcal{K}_k(A, r_0)$ sera de dimension $k = m$ si et seulement si le degré du polynôme minimal de A pour r_0 est supérieur ou égal à m.

On utilise dans la suite les résultats de Sadok et Heyouni [60, 34, 35]. Afin de calculer z_k, on construit une base $\{v_1, v_2, \ldots, v_k\}$ du sous-espace de Krylov $\mathcal{K}_k(A, r_0)$, et ceci à l'aide du processus de Hessenberg généralisé.

Le premier vecteur v_1 est choisi colinéaire à r_0, i.e. $r_0 = \beta_1 v_1$. A la j-ième étape, le vecteur v_{j+1} est calculé à l'aide de la formule suivante :

$$\beta_{j+1} v_{j+1} = A v_j - \sum_{i=1}^{j} h_{i,j} v_i, \tag{1.4}$$

le paramètre β_{j+1} étant appelé facteur normalisant de v_{j+1}.

La détermination des paramètres $\{h_{i,j}\}_{i=1}^{j}$ se fait en imposant une condition d'orthogonalité sur le vecteur v_{j+1}. On considère pour cela des vecteurs $w_1, \ldots, w_j$ appartenant à $\mathbb{R}^n$ qui peuvent être quelconques mais doivent être linéairement indépendants. Ces vecteurs w_i permettent de caractériser les vecteurs v_j de $\mathcal{K}_j(A, r_0)$, ceci pour $i = 1, \ldots, j$. Ainsi,

$$v_j \in \mathcal{K}_j(A, r_0) \text{ et } (w_i, v_j) = v_j^t w_i = 0 \text{ pour } i = 1, \ldots, j-1. \tag{1.5}$$

Les relations (1.4) et (1.5) peuvent être traduites matriciellement. On a respectivement :

1.
$$AV_j = V_{j+1}\overline{H}_j \tag{1.6}$$
ou encore
$$AV_j = V_j H_j + \beta_{j+1} v_{j+1} e_j^{(j)t}, \tag{1.7}$$
où H_j est une matrice de Hessenberg carrée de taille j de la forme :
$$H_j = \begin{pmatrix} h_{1,1} & h_{1,2} & \dots & \dots & h_{1,j} \\ \beta_2 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \beta_j & h_{j,j} \end{pmatrix} \tag{1.8}$$
et $\overline{H}_j = \begin{pmatrix} H_j \\ (0, \dots, 0, \ \beta_{j+1}) \end{pmatrix} = \begin{pmatrix} H_j \\ \beta_{j+1} e_j^{(j)t} \end{pmatrix} \in I\!R^{(j+1)\times j}$ est une matrice de Hessenberg supérieure. $e_j^{(j)}$ est le j-ième vecteur colonne de la matrice identité de taille $j \times j$.

2. La condition d'orthogonalité s'écrit :
$$W_j^t V_j = L_j \tag{1.9}$$
où L_j est une matrice triangulaire inférieure de taille j, donnée par
$$L_j = \begin{pmatrix} (w_1, v_1) & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ (w_j, v_1) & \dots & \dots & (w_j, v_j) \end{pmatrix} \in I\!R^{j\times j}. \tag{1.10}$$

On peut déduire de ces relations les deux égalités essentielles pour la suite :
$$W_j^t A V_j = L_j H_j \tag{1.11}$$
$$W_{j+1}^t A V_j = L_{j+1} \overline{H}_j. \tag{1.12}$$
Par souci d'homogénéité dans l'écriture de H_j et $\overline{H}_j$, le scalaire β_{j+1} sera noté $h_{j+1,j}$ Par conséquent $H_j = \{h_{i,l}\}_{1\leq i,l\leq j}$ et $\overline{H}_j = \{h_{i,l}\}_{\substack{1 \leq i \leq j+1 \\ 1 \leq l \leq j}}$.

On rappelle maintenant le processus de Hessenberg généralisé :

Algorithme 1.1 : $[V_{k+1}, \overline{H}_k]$ = Hessenberg généralisé (A, r_0, k, W_k)

1. initialisation :
   (a) choisir $\beta_1 \neq 0$ coefficient normalisateur de r_0,
   (b) calculer $\beta_1 v_1 = r_0$ et $\eta_1 = (w_1, v_1)$,
2. itérations : pour $j = 1, 2, \ldots, k$ faire
   (a) $\omega = Av_j$,
   (b) pour $i = 1, \ldots, j$ faire
   $h_{i,j} = (\omega, w_i)/\eta_i$,
   $\omega = \omega - h_{i,j} v_i$,
   fin pour,
   (c) choisir $h_{j+1,j}$ facteur normalisant de v_{j+1},
   (d) $v_{j+1} = \omega / h_{j+1,j}$ et $\eta_{j+1} = (w_{j+1}, v_{j+1})$,

   fin pour,
3. définir $V_{k+1} = [v_1, \ldots, v_{k+1}]$, $\overline{H}_k = \{h_{i,j}\}_{\substack{1 \leq i \leq k+1 \\ 1 \leq j \leq k}}$.

On obtient le résultat suivant :

Proposition 1.1.1
Supposons que k étapes du processus de Hessenberg généralisé soient réalisées, alors les vecteurs $v_1, \ldots, v_k$ calculés par l'algorithme 1.1 forment une base de $\mathcal{K}_k(A, r_0)$.

Une dégénérescence peut apparaître dans le processus lorsque

$$\eta_{j+1} = (w_{j+1}, v_{j+1}) = 0 \tag{1.13}$$

et l'algorithme s'arrête alors. η_{j+1} est nul dans deux cas :

1. $v_{j+1} = 0$, on parlera alors de "lucky breakdown". En effet, un sous-espace invariant a été trouvé et une solution exacte est calculée.
2. $v_{j+1} \perp w_{j+1}$ alors,
   (a) lorsque les w_j sont fixés à l'avance on calcule la solution x_j ainsi que le résidu correspondant r_j et on redémarre le processus avec $x_0 = x_j$.

(b) Si les vecteurs w_j ne sont pas connus à l'avance, on choisit un autre vecteur w_{j+1} qui n'est pas orthogonal à v_{j+1}.

On peut montrer le lemme suivant, utile pour la suite :

Lemme 1.1.1

$$H_{j+1} \textit{ est définie } \Leftrightarrow L_{j+1} \textit{ est inversible.}$$

Preuve :
Cette équivalence se montre à l'aide du processus de Hessenberg généralisé. On suppose tout d'abord que L_j est inversible, donc

$$L_{j+1} \text{ est inversible } \Leftrightarrow (w_{j+1}, v_{j+1}) \neq 0 \Leftrightarrow \eta_{j+1} \neq 0.$$

Or $h_{j+1,j+1} = (Av_{j+1}, w_{j+1})/\eta_{j+1}$, donc $h_{j+1,j+1}$ est défini si et seulement si $\eta_{j+1} \neq 0$. Ainsi, à l'aide de l'inversibilité de L_j, $h_{j+1,j+1}$ est défini si et seulement si H_{j+1} est définie, ce qui conclut la preuve. ■

De la même façon on peut montrer le

Lemme 1.1.2

$$\overline{H}_j \textit{ est définie } \Leftrightarrow L_{j+1} \textit{ est inversible.}$$

Une fois ces préliminaires établis, on se propose de développer les processus auxquels donne lieu le processus de Hessenberg généralisé.

1.2 Les cas particuliers du processus de Hessenberg généralisé

On a précisé précédemment que les différentes versions des méthodes de sous-espaces de Krylov provenaient du choix de $\mathcal{L}_k$. Les différents processus utilisés proviennent du choix des vecteurs $w_1, w_2, \ldots, w_k$ et ces choix répondent à des besoins classiques d'efficacité et d'économie en temps de calcul et de stockage.

1.2.1 Le processus d'Arnoldi

Un choix naturel consiste à prendre la famille $\{w_j\}_{j=1}^k$ égale à la famille $\{v_j\}_{j=1}^k$. Dans ce cas, la condition d'orthogonalité (1.9) s'écrit sous la forme :

$$W_j^t V_j = V_j^t V_j = L_j.$$

Comme L_j est triangulaire inférieure et $V_j^t V_j$ est symétrique, on en déduit que L_j est diagonale :

$$V_j^t V_j = D_j, \tag{1.14}$$

où $D_j = diag(v_1^t v_1, \ldots, v_j^t v_j)$.

De plus, on a vu que $h_{j+1,j}$ était souvent calculé comme facteur normalisant de v_{j+1}. Il suffit donc de choisir $h_{j+1,j}$ tel que $\|v_{j+1}\|_2 = 1$ ou encore $h_{j+1,j} = \|\omega\|_2$. De même, on choisira $\beta_1 = \|r_0\|_2$. Dans ce cas,

$$D_j = I_j \text{ et } V_j \in I\!R^{n\times j} \text{ est orthonormale.}$$

L'algorithme 1.1 se réécrit de la façon suivante :

Algorithme 1.2 : $[V_{k+1}, \overline{H}_k]$ = Arnoldi (A, r_0, k)

1. initialisation :
   calculer $v_1 = r_0/\|r_0\|_2$
2. itérations : pour $j = 1, 2, \ldots, k$ faire
   (a) $\omega = Av_j$,
   (b) pour $i = 1, 2, \ldots, j$ faire
       $h_{i,j} = (\omega, v_i)$,
       $\omega = \omega - h_{i,j} v_i$,
       fin pour,
   (c) $h_{j+1,j} = \|\omega\|_2$,
   (d) si $h_{j+1,j} \neq 0$ alors $v_{j+1} = \omega/h_{j+1,j}$,
       sinon stop,

   fin pour,
3. définir $V_{k+1} = [v_1, \ldots, v_{k+1}]$, $\overline{H}_k = \{h_{i,j}\}_{\substack{1 \le i \le k+1 \\ 1 \le j \le k}}$.

Si, à l'étape k du processus d'Arnoldi, l'égalité $v_{k+1} = 0$ est vérifiée alors on a le

Théorème 1.2.1
L'algorithme d'Arnoldi s'arrête à l'étape $k = m$ si et seulement si le degré du polynôme minimal de A pour r_0 est égal à m. De plus le sous-espace $\mathcal{K}_m(A, r_0)$ est invariant par A, i.e. $A\mathcal{K}_m(A, r_0) = \mathcal{K}_m(A, r_0)$.

L'algorithme 1.2 décrit le processus d'Arnoldi avec orthonormalisation.

Différentes versions de ce processus existent, notamment celles utilisant l'algorithme de Gram-Schmidt classique et l'algorithme de Gram-Schmidt modifié qui sont mathématiquement équivalents. On peut également utiliser l'algorithme de Householder très stable numériquement mais deux fois plus coûteux que les deux algorithmes précédents. La version utilisée ici est l'algorithme de Gram-Schmidt modifié.

Lorsque $h_{j+1,j} = 0$, c'est-à-dire lorsque l'algorithme dégénère à la j-ième itération, un sous-espace de Krylov invariant est trouvé :

$$A\mathcal{K}_j(A, r_0) = \mathcal{K}_j(A, r_0) \tag{1.15}$$

(l'invariance de $\mathcal{K}_j(A, r_0)$ par A découle du degré du polynôme minimal de A pour r_0).

Etant donné que V_j est orthonormale, la proposition 1.1.1 peut maintenant s'énoncer

Proposition 1.2.1
Supposons que k étapes soient réalisables dans le processus d'Arnoldi, alors les vecteurs $v_1, v_2, \ldots, v_k$ calculés par l'algorithme 1.2 forment une base orthonormale de $\mathcal{K}_k(A, r_0)$.

Les relations (1.11) et (1.12) s'écrivent à la k-ième itération :

$$V_k^t A V_k = H_k, \tag{1.16}$$

$$V_{k+1}^t A V_k = \overline{H}_k. \tag{1.17}$$

Si on appelle P_k le projecteur L_2-orthogonal sur $\mathcal{K}_k(A, r_0)$ et si on note par A_k la restriction de A à $\mathcal{K}_k(A, r_0)$, i.e. l'opérateur $A_k = P_k A_{|\mathcal{K}_k(A, r_0)}$, on peut vérifier que H_k n'est rien d'autre que la représentation matricielle de A_k dans la base $\{v_1, v_2, \ldots, v_k\}$.

Remarque 1.2.1

1. *Le processus d'Arnoldi nécessite à la j-ième étape $2nj$ multiplications ainsi qu'un produit matrice-vecteur dont le coût est de $n\nu$ multiplications avec :*
   (a) *pour une matrice dense, $\nu = n$,*
   (b) *pour une matrice creuse, $\nu = N_{nz}/n$ où N_{nz} est le nombre d'éléments non nuls de la matrice.*

   Au total, pour k itérations, le processus d'Arnoldi nécessite un coût algorithmique de l'ordre de $nk^2 + kn\nu$ multiplications.
2. *En ce qui concerne le stockage, on conserve k vecteurs de taille n ainsi que les données non nulles d'une matrice de Hessenberg, soit un total approximatif de $nk + (k+3)k/2$ éléments à stocker.*

1.2.2 Un cas particulier du processus d'Arnoldi : l'algorithme de Lanczos symétrique

L'algorithme de Lanczos symétrique peut être vu comme une simplification de la méthode d'Arnoldi pour le cas particulier où la matrice est symétrique (et plus généralement hermitienne). Ainsi, le principe de la méthode est le même que celui développé dans la section précédente dans le sens où c'est une technique de projection sur un sous-espace de Krylov.

Lorsque A est symétrique, H_k l'est également. La matrice H_k est donc tridiagonale et ses diagonales supérieures et inférieures sont identiques. Ainsi,

$$T_k \equiv H_k = \begin{pmatrix} \alpha_1 & \gamma_2 & 0 & \dots & 0 \\ \gamma_2 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \gamma_k \\ 0 & \dots & 0 & \gamma_k & \alpha_k \end{pmatrix}, \text{ en posant } \begin{cases} \alpha_j = h_{j,j} \text{ pour } 1 \le j \le k \text{ et} \\ \gamma_j = h_{j-1,j} = h_{j,j-1} \text{ pour } 2 \le j \le k \end{cases}.$$

Le processus d'Arnoldi se simplifie car seuls les vecteurs v_k et v_{k-1} sont nécessaires à la construction de v_{k+1}.

Algorithme 1.3 : $[V_{k+1}, T_k]$ = Lanczos symétrique (A, r_0, k)

1. initialisation :
   calculer $v_1 = r_0/\|r_0\|$,
   poser $\gamma_1 \equiv 0,\ v_0 \equiv 0$,
2. itérations : pour $j = 1, 2, \ldots, k$, faire
   (a) $\omega = Av_j - \gamma_j v_{j-1}$,
   (b) $\alpha_j = (\omega, v_j)$,
   $\omega = \omega - \alpha_j v_j$,
   (c) $\gamma_{j+1} = \|\omega\|_2$,
   (d) si $\gamma_{j+1} \neq 0$ alors $v_{j+1} = \omega/\gamma_{j+1}$,
   sinon stop,

   fin pour,
3. définir $V_{k+1} = [v_1, \ldots, v_{k+1}],\ T_k$.

On peut noter que cet algorithme garantit, au moins en arithmétique exacte, l'orthogonalité des vecteurs v_i pour $i = 1, \ldots, k$. On remarquera également que seul le stockage de trois vecteurs est nécessaire dans le cas où on n'a recours à aucune forme de réorthogonalisation.

1.2.3 Le processus de Lanczos non-symétrique

Le processus de Lanczos construit les vecteurs $\{w_j\}_{j=1}^k$ de façon progressive : ils sont calculés en même temps que les vecteurs $\{v_j\}_{j=1}^k$. Les vecteurs $\{w_j\}_{1\leq j\leq k}$ vont constituer une base du sous-espace de Krylov associé à la matrice A^t ainsi qu'au vecteur initial w_1, c'est-à-dire $\mathcal{K}_k(A^t, w_1)$ avec $(r_0, w_1) \neq 0$. Ce processus va alors construire simultanément une base $\{v_1, \ldots, v_k\}$ de $\mathcal{K}_k(A, r_0)$ et une base $\{w_1, \ldots, w_k\}$ de $\mathcal{K}_k(A^t, w_1)$ à l'aide de relations de même nature que (1.4), c'est-à-dire

$$h_{j+1,j} v_{j+1} = Av_j - \sum_{i=1}^{j} h_{i,j} v_i \tag{1.18}$$

$$\widetilde{h}_{j+1,j} w_{j+1} = A^t w_j - \sum_{i=1}^{j} \widetilde{h}_{i,j} w_i, \tag{1.19}$$

Les paramètres $\{h_{i,j}\}_{i=1}^{j}$ et $\{\widetilde{h}_{i,j}\}_{i=1}^{j}$ sont déterminés de façon à ce qu'on ait

$$v_{j+1} \perp W_j = [w_1, \ldots, w_j], \tag{1.20}$$

$$w_{j+1} \perp V_j = [v_1, \ldots, v_j]. \tag{1.21}$$

On peut écrire ces quatre dernières relations sous forme matricielle. On a respectivement :

$$AV_j = V_{j+1}\overline{H}_j = V_j H_j + h_{j+1,j} v_{j+1} e_j^{(j)t} \tag{1.22}$$

$$A^t W_j = W_{j+1}\overline{\widetilde{H}}_j = W_j \widetilde{H}_j + \widetilde{h}_{j+1,j} w_{j+1} e_j^{(j)t} \tag{1.23}$$

$$W_j^t V_j = L_j \tag{1.24}$$

$$V_j^t W_j = \widetilde{L}_j \tag{1.25}$$

où $\overline{H}_j$ et $\overline{\widetilde{H}}_j$ sont les matrices de Hessenberg supérieures de taille $(j+1) \times j$ formées de manière respective par les éléments $h_{i,l}$ et $\widetilde{h}_{i,l}$ pour $1 \leq i \leq j+1$ et $1 \leq l \leq j$. Les matrices L_j et $\widetilde{L}_j$ sont quant à elles triangulaires inférieures.

On déduit des deux dernières relations que $L_j = \widetilde{L}_j^t = diag(\eta_1, \ldots, \eta_j) = D_j$ avec $\eta_i = (w_i, v_i)$ pour $1 \leq i \leq j$. Les relations (1.22) et (1.23) donnent alors

$$W_j^t A V_j = D_j H_j, \tag{1.26}$$

$$V_j^t A^t W_j = D_j \widetilde{H}_j. \tag{1.27}$$

On en déduit que $(D_j \widetilde{H}_j)^t = D_j H_j$ et finalement, si D_j est inversible,

$$H_j = D_j^{-1} \widetilde{H}_j^t D_j = (D_j \widetilde{H}_j D_j^{-1})^t. \tag{1.28}$$

H_j étant une matrice de Hessenberg supérieure et $(D_j \widetilde{H}_j D_j^{-1})^t$ une matrice de Hessenberg inférieure, H_j et $\widetilde{H}_j$ sont par conséquent tridiagonales et (1.18) ainsi que (1.19) se simplifient :

$$h_{j+1,j} v_{j+1} = A v_j - h_{j,j} v_j - h_{j-1,j} v_{j-1}, \tag{1.29}$$

$$\widetilde{h}_{j+1,j} w_{j+1} = A^t w_j - \widetilde{h}_{j,j} w_j - \widetilde{h}_{j-1,j} w_{j-1}. \tag{1.30}$$

En utilisant les conditions de biorthogonalité sur les familles $\{v_j\}_{j=1}^{k}$ et $\{w_j\}_{j=1}^{k}$, il est possible de montrer que

$$h_{j,j} = \widetilde{h}_{j,j}, \tag{1.31}$$

$$h_{j-1,j} = \widetilde{h}_{j,j-1} \eta_j / \eta_{j-1}, \tag{1.32}$$

$$\widetilde{h}_{j-1,j} = h_{j,j-1} \eta_j / \eta_{j-1}. \tag{1.33}$$

Les facteurs normalisants de v_{j+1} et w_{j+1} seront choisis égaux, ce qui entraînera les égalités $h_{j,j-1} = \widetilde{h}_{j-1,j}$ ainsi que $h_{j-1,j} = \widetilde{h}_{j,j-1}$ à l'aide de (1.32) et (1.33). Par conséquent l'égalité des facteurs normalisants entraîne l'égalité des matrices H_j et $\widetilde{H}_j$ produites par le processus de Lanczos.

On peut à présent décrire le processus :

Algorithme 1.4 : $[V_{k+1}, \overline{H}_k]$ = Lanczos non-symétrique (A, r_0, k)

1. initialisation :
   poser $v_1 = r_0$,
   choisir w_1 tel que $(w_1, v_1) = 1$,
2. itérations : pour $j = 1, 2, \ldots, k$, faire
   (a) $\omega = Av_j$,
   $\tilde{\omega} = A^t w_j$,
   (b) pour $i = max(1, j-1), \ldots, j$ faire
   $h_{i,j} = (\omega, w_i)/\eta_i$,
   $\omega = \omega - h_{i,j} v_i$, $\tilde{\omega} = \tilde{\omega} - h_{i,j} v_i$,
   fin pour,
   (c) choisir $h_{j+1,j} \neq 0$,
   $v_{j+1} = \omega / h_{j+1,j}$, $w_{j+1} = \tilde{\omega}/h_{j+1,j}$, $\eta_{j+1} = (v_{j+1}, w_{j+1})$,

   fin pour,
3. définir $V_{k+1} = [v_1, \ldots, v_{k+1}]$, $\overline{H}_k = \{h_{i,j}\}_{\substack{1 \leq i \leq k+1 \\ 1 \leq j \leq k}}$.

On arrêtera bien-sûr l'algorithme précédent à l'étape j lorsque

$$\eta_{j+1} = (v_{j+1}, w_{j+1}) = 0.$$

On retrouve les dégénérescences introduites lors de la présentation du processus de Hessenberg généralisé, survenant dans le cas particulier du processus de Lanczos lorsque :

1. $v_{j+1} = 0$ ou $w_{j+1} = 0$,
2. $v_{j+1} \perp w_{j+1}$.

Dans le premier cas, la dégénérescence peut être évitée en choisissant un nouveau vecteur w_1. Dans le second, on est en présence d'un "serious breakdown", l'idée pour y remédier étant la suivante : on peut définir deux nouveaux vecteurs v_{j+i} et w_{j+i} ($i \geq 2$) qui vont satisfaire les différentes conditions du processus de Lanczos. Les approches proposées

se nomment “look-ahead Lanczos algorithms” [26] ou “Method of Recursive Zoom” [5] et reposent essentiellement sur le lien existant entre le processus de Lanczos et les polynômes orthogonaux.
Dans le cas où ce type de dégénérescences ne survient pas, on dispose du résultat suivant

Proposition 1.2.2
Supposons que k étapes du processus de Lanczos soient réalisables, alors les vecteurs $v_1, \dots, v_k$ (respectivement $w_1, \dots, w_k$) forment une base de $\mathcal{K}_k(A, r_0)$ (respectivement $\mathcal{K}_k(A^t, w_1)$).

On a également le

Théorème 1.2.2
Supposons que $\forall w \in \mathcal{K}_k(A, r_0)$, $\forall \hat{w} \in \mathcal{K}_k(A^t, w_1)$, $(\hat{w}, w) \neq 0$. Alors le processus de Lanczos non-symétrique s'arrête à l'étape $k = m$ si et seulement si le degré du polynôme minimal de A pour r_0 (ou celui de A^t pour w_1) est égal à m. De plus, le sous-espace $\mathcal{K}_m(A, r_0)$ (ou $\mathcal{K}_m(A^t, w_1)$) est invariant par A (ou par A^t).

Remarque 1.2.2

1. *Le processus de Lanczos nécessite à la j-ième itération $6n$ multiplications ainsi que deux produits matrice-vecteur dont le coût est de $n\nu$ multiplications. Au total, pour k itérations, le processus de Lanczos nécessite un coût algorithmique de l'ordre de $6kn + 2nk\nu$.*

2. *On stockera également k vecteurs de taille n ainsi que $3k - 1$ éléments des 3 diagonales de la matrice $\overline{H}_k$.*

Remarque 1.2.3
Si la matrice A est symétrique et si les vecteurs de départ v_1 et w_1 sont égaux, les processus de Lanczos et d'Arnoldi sont équivalents et donnent le processus de Lanczos symétrique.

1.2.4 Le processus de Hessenberg

Un troisième choix qui peut être fait dans le processus de Hessenberg généralisé consiste à prendre

$$w_j = e_j^{(n)} \text{ pour } j = 1, \dots, k,$$

où $e_j^{(n)}$ est le j-ième vecteur de la base canonique de $I\!R^n$. Les w_j sont donc connus à l'avance.
On écrit V_j sous la forme :

$$V_j = \begin{pmatrix} V_j^1 \\ V_j^2 \end{pmatrix} \tag{1.34}$$

où V_j^1 est la matrice formée des j premières lignes de V_j, de taille $j \times j$. On partitionne de la même façon W_j^t sous la forme

$$W_j^t = (I_j^{(j)}, 0).$$

Notre condition d'orthogonalité (1.9) devient :

$$W_j^t V_j = V_j^1 = L_j$$

ce qui implique que V_j^1 est triangulaire inférieure. Par conséquent, V_j est une matrice trapézoïdale inférieure. Le processus de Hessenberg généralisé s'écrit dans ce cas sous la forme d'un processus plus simple appelé processus de Hessenberg :

Algorithme 1.5 : $[V_{k+1}, \overline{H}_k]$ = Hessenberg (A, r_0, k, W_k)

1. initialisation :
   poser $v_1 = r_0$ et $\eta_1 = (v_1)_1$,
2. itérations : pour $j = 1, 2, \ldots, k$ faire
   (a) $\omega = Av_j$,
   (b) pour $i = 1, 2, \ldots, j$ faire
       $h_{i,j} = (\omega)_i / \eta_i$,
       $\omega = \omega - h_{i,j} v_i$,
       fin pour,
   (c) choisir $h_{j+1,j} \neq 0$ facteur normalisant de v_{j+1},
   (d) $v_{j+1} = \omega / h_{j+1,j}$ et $\eta_{j+1} = (v_{j+1})_{j+1}$,
       fin pour,
3. définir $V_{k+1} = [v_1, \ldots, v_{k+1}]$, $\overline{H}_k = \{h_{i,j}\}_{\substack{1 \le i \le k+1 \\ 1 \le j \le k}}$.

Des résultats similaires à ceux obtenus précédemment peuvent être introduits :

Théorème 1.2.3
L'algorithme de Hessenberg s'arrête à l'étape $k = m$ si et seulement si le degré du polynôme minimal de A pour r_0 est égal à m. De plus, le sous-espace $\mathcal{K}_m(A, r_0)$ est invariant par A, i.e. $A\mathcal{K}_m(A, r_0) = \mathcal{K}_m(A, r_0)$.

Une dégénérescence peut survenir dans l'algorithme précédent lorsque $(v_{j+1})_{j+1} = 0$ ou lorsque $(v_{j+1})_{j+1}$ est proche de zéro (on parlera alors de "near-breakdown"). Afin d'éviter ce problème, on utilise une version de l'algorithme précédent avec pivotage comme lors de la factorisation LU. La stratégie utilisée consiste à prendre pour $\{w_j\}_{j=1}^k$ la famille $\{e_j^{(n)}\}_{j=1}^k$ mais pas nécessairement dans son ordre naturel. η_{j+1} sera prise comme étant la valeur de la composante de v_{j+1} qui est la plus grande en valeur absolue par rapport à toutes les autres composantes.
Dans le cas où aucun type de dégénérescence ne survient, on a la

Proposition 1.2.3
Supposons que k itérations du processus de Hessenberg soient réalisables, alors les vecteurs $v_1, \ldots, v_k$ calculés par l'algorithme de Hessenberg forment une base du sous-espace de Krylov $\mathcal{K}_k(A, r_0)$.

Remarque 1.2.4

1. *Le processus de Hessenberg nécessite à la j-ième étape $nj - j(j+1)/2$ multiplications et un produit matrice-vecteur dont le coût est de $n\nu_j = n - j$ multiplications si la matrice est dense et $n\nu_j \leq N_{nz}$ dans le cas où A est creuse. Au total, pour k itérations, le processus de Hessenberg nécessite un coût algorithmique de l'ordre de $nk^2/2 - k^3/6 + \sum_{j=1}^{k} n\nu_j$ multiplications.*
2. *On doit stocker nk éléments : les $k(k+1)/2$ éléments supérieurs (respectivement les k éléments de la sous-diagonale) de $\overline{H}_k$ peuvent être stockés à la place des éléments nuls (respectivement égaux à l'unité) de la matrice V_{k+1}.*

Ces différents processus ayant été présentés, on peut désormais les placer dans le contexte des matrices singulières.

1.3 Le processus de Hessenberg généralisé et les systèmes singuliers

On désire étudier dans cette section le comportement du processus de Hessenberg généralisé en arithmétique exacte lorsqu'il est appliqué à la construction d'une base du sous-espace de Krylov $\mathcal{K}_k(A, r_0)$ où A est une matrice singulière.

1.3.1 Résultats théoriques

On considère le système (1.1) avec une matrice singulière $A \in I\!\!R^{n\times n}$. Soit $Im(A)$ l'image de la matrice A. On rappelle que si $b \in Im(A)$, le système (1.1) est dit consistant et admet une infinité de solutions. Dans le cas contraire, le système est dit inconsistant ou incompatible et n'admet aucune solution.
Soit $Sp(A)$ l'ensemble des valeurs propres de la matrice A,

$$Sp(A) = \{\lambda_1, \lambda_2, \ldots, \lambda_p\} \tag{1.35}$$

où λ_1 correspond à la valeur propre nulle. On suppose que $p \geq 2$ pour assurer qu'au moins une valeur propre de A est non nulle. Toutes les valeurs propres du spectre de A ne sont pas comptées avec leur multiplicité algébrique respective $\alpha_1, \alpha_2, \ldots, \alpha_p$ où $\alpha_i \geq 1,\ \forall i \in \{1, 2, \ldots, p\}$. On rappelle également que la singularité en arithmétique flottante induit l'existence de valeurs propres très proches de 0. On précise une nouvelle fois que les matrices faisant intervenir ce type de valeurs propres n'interviennent pas dans ce travail.

Rappelons que le polynôme caractéristique de A s'écrit dans ce cas

$$P_A(\lambda) = \lambda^{\alpha_1}(\lambda - \lambda_2)^{\alpha_2} \ldots (\lambda - \lambda_p)^{\alpha_p}$$

et vérifie d'après le théorème de Cayley-Hamilton l'égalité $P_A(A) = 0$.

On introduit également la notion d'indice de A :

Définition 1.3.1
Le plus petit entier ν_1 non négatif vérifiant

$$dim Im(A^{\nu_1}) = dim Im(A^{\nu_1+1})$$

est appelé l'indice de A et est décrit par $ind(A)$. Avec $\nu_1 = ind(A)$, on a la décomposition de $I\!\!R^n$ suivante :

$$I\!\!R^n = Im(A^{\nu_1}) \oplus Ker(A^{\nu_1}). \tag{1.36}$$

On applique maintenant le processus de Hessenberg généralisé à la construction d'une base du sous-espace de Krylov $\mathcal{K}_k(A, r_0)$ où $k = m$ est supposé être l'étape à laquelle le processus s'arrête. On aura donc, en arithmétique exacte, $h_{m+1,m} = 0$ et $h_{j+1,j} \neq 0$ pour $1 \leq j \leq m-1$.

On supposera dans la suite de ce paragraphe que H_m est définie, m étant l'itération à laquelle la relation $h_{m+1,m} = 0$ est vérifiée.

On peut introduire le lemme suivant :

Lemme 1.3.1

$$\forall k \in I\!N, \quad W_m^t A^k V_m = L_m H_m^k \tag{1.37}$$

Preuve :
Ce résultat est démontrable par récurrence. En effet, pour $k = 1$, $W_m^t A V_m = L_m H_m$ ce qui est vrai à l'aide de (1.11). On va supposer ce résultat vrai au rang k : $W_m^t A^k V_m = L_m H_m^k$. Alors au rang $k+1$, $W_m^t A^{k+1} V_m = W_m^t A^k A V_m$ et d'après la relation (1.7) avec $h_{m+1,m} = v_{m+1}\beta_{m+1} = 0$, on a $AV_m = V_m H_m$ ce qui implique que $W_m^t A^{k+1} V_m = W_m^t A^k V_m H_m = L_m H_m^k H_m = L_m H_m^{k+1}$ et la relation est démontrée. ■

Ce lemme se révèle très utile pour démontrer le résultat suivant, à savoir

Proposition 1.3.1

$$\forall A \in I\!R^{n\times n}, \quad \Pi_A(A) r_0 = 0 \;\Rightarrow\; \Pi_A(H_m) = 0, \tag{1.38}$$

où Π_A est le polynôme minimal de A pour r_0.

Preuve :
On sait que $h_{m+1,m} = 0$ implique simultanément

$$d^o \Pi_A = m, \tag{1.39}$$

$$\Pi_A(A) r_0 = 0, \tag{1.40}$$

où $d^o P = k$ signifie que le degré du polynôme P est de degré k.
Soit Π_A tel que

$$\Pi_A(\lambda) = a_0 + a_1\lambda + \ldots + a_m\lambda^m. \tag{1.41}$$

Or $\Pi_A(A) r_0 = 0$ et $r_0 = \|r_0\| V_m e_1 = \beta_1 v_1$ donc :

$$(a_0 V_m + a_1 A V_m + \ldots + a_m A^m V_m) e_1 = 0. \tag{1.42}$$

En multipliant à gauche chaque côté de l'égalité (1.42) par W_m^t, on obtient

$$(a_0 W_m^t V_m + a_1 W_m^t A V_m + \ldots + a_m W_m^t A^m V_m) e_1 = 0.$$

On utilise ensuite la relation du lemme 1.3.1, ce qui donne :

$$\begin{aligned}(a_0 L_m + a_1 L_m H_m + ... + a_m L_m H_m^m) e_1 = 0 \\ \Leftrightarrow L_m \Pi_A(H_m) e_1 = 0. \qquad (1.43)\end{aligned}$$

Le fait que H_m soit définie équivaut à l'inversibilité de L_m (voir lemme 1.1.1), donc on a

$$\Pi_A(H_m) e_1 = 0.$$

On multiplie ensuite à gauche cette égalité par H_m^j, $j = 0, 1, \ldots, m-1$ ce qui donne

$$\begin{aligned}H_m^j \Pi_A(H_m) e_1 = 0 \text{ pour } j = 0, 1, \ldots, m-1, \\ \Leftrightarrow \Pi_A(H_m) H_m^j e_1 = 0 \text{ pour } j = 0, 1, \ldots, m-1.\end{aligned}$$

Puisque $h_{j+1,j} \neq 0$ pour $j = 1, 2, \ldots, m-1$, l'espace $vect\{e_1, H_m e_1, \ldots, H_m^{m-1} e_1\}$ est égal à $I\!R^m$. Ceci implique que $\Pi_A(H_m) = 0$. ∎

On déduit à l'aide de cette proposition que le polynôme minimal associé à A et r_0 est égal à un polynôme annihilateur de H_m et, plus précisément, au polynôme caractéristique de H_m puisque son degré est égal à la dimension de la matrice H_m.

De manière générale, on sait que A peut être factorisée de la façon suivante :
$\exists X \in I\!R^{n\times n}$ non-singulière et $J \in I\!R^{n\times n}$ telles que :

$$A = XJX^{-1}, \qquad (1.44)$$

où J est une forme canonique de Jordan de A.
Le noyau de la matrice $(A - \lambda_i I)^{\alpha_i}$ est appelé sous-espace caractéristique de A associé à la valeur propre λ_i. Le polynôme minimal de A est alors défini par

$$Q_A(\lambda) = \lambda^{\nu_1}(\lambda - \lambda_2)^{\nu_2} \ldots (\lambda - \lambda_p)^{\nu_p} \qquad (1.45)$$

et vérifie à l'aide du théorème de Cayley-Hamilton, $Q_A(A) = 0$. ν_i est égale à la dimension du plus grand bloc de Jordan associé à la valeur propre λ_i, $\forall i \in \{1, 2, \ldots, p\}$. On pose dans la suite $N_{\lambda_i} = Ker[(A - \lambda_i I)^{\nu_i}]$ avec $\nu_i \geq 1$, $\forall i \in \{1, 2, \ldots, p\}$. Ces sous-espaces vérifient

$$I\!R^n = \bigoplus_{i=1}^{p} N_{\lambda_i}, \qquad (1.46)$$

le signe $\oplus$ signifie que les sous-espaces N_{λ_i} sont supplémentaires dans $I\!R^n$. Il est évident à l'aide de (1.36) et (1.46) que la multiplicité associée à la valeur propre 0 dans le polynôme minimal de A est égale à l'indice de A.
Le résidu initial r_0 peut s'écrire avec les notations précédentes

$$r_0 = X\beta = \sum_{i=1}^{n} \beta_i x_i = u_1 + u_2 + \ldots + u_p \qquad (1.47)$$

avec $u_i \in N_{\lambda_i}$ pour $i \in \{1, 2, \ldots, p\}$.

Dans le cas où A est diagonalisable, $\nu_i = 1$, $\forall i \in \{1, 2, \ldots, p\}$ et $I\!R^n = \bigoplus_{i=1}^{p} E_{\lambda_i}$, où $E_{\lambda_i} = Ker(A - \lambda_i I)$ ceci pour tout $i \in \{1, 2, \ldots, p\}$. On peut dorénavant présenter le résultat suivant

Proposition 1.3.2

Supposons que $A \in I\!R^{n\times n}$ soit singulière avec $b \in Im(A)$. Si $\lambda_1 = 0$ est telle que $\nu_1 = 1$ alors

$$r_0 = u_2 + u_3 + \ldots + u_p \tag{1.48}$$

dans le cas où aucune composante de r_0 sur N_{λ_i} n'est nulle pour $i \in \{2, 3, \ldots, p\}$.

<u>Preuve</u>:

Supposons que $r_0 = u_1 + u_2 + \ldots + u_p$. Etant donné que $\nu_1 = 1$, $u_1 \in Ker(A)$. On sait que $r_0 = b - Ax_0 \in Im(A)$. A l'aide de la formule du binôme on a

$$(A - \lambda_i I)^{\nu_i} = \sum_{k=0}^{\nu_i} C_{\nu_i}^k A^k (-\lambda_i I)^{\nu_i - k}, \ \forall i \in \{2, 3, \ldots, p\}$$

or $u_i \in N_{\lambda_i}$ $\forall i \in \{2, 3, \ldots, p\}$, ce qui implique

$$\sum_{k=0}^{\nu_i} C_{\nu_i}^k A^k (-\lambda_i I)^{\nu_i - k} u_i = 0, \ \forall i \in \{2, 3, \ldots, p\}$$

$$\Rightarrow -(-\lambda_i I)^{\nu_i} u_i = A \sum_{k=1}^{\nu_i} C_{\nu_i}^k A^{k-1} (-\lambda_i I)^{\nu_i - k} u_i, \ \forall i \in \{2, 3, \ldots, p\}$$

et finalement $u_i \in Im(A)$, $\forall i \in \{2, 3, \ldots, p\}$. On pose ensuite

$$r_0 = u_1 + w \text{ avec } w = u_2 + u_3 + \ldots + u_p,$$

ainsi $u_1 \in Im(A) \cap Ker(A)$. Par définition de ν_1, on a

$$I\!R^n = Im(A) \oplus Ker(A). \tag{1.49}$$

Puisque l'intersection est vide entre les deux sous-espaces, on conclut que

$$r_0 = u_2 + u_3 + \ldots + u_p. \ \blacksquare$$

Corollaire 1.3.1

Supposons que $A \in I\!R^{n\times n}$ soit singulière. Si A est diagonalisable et si la multiplicité algébrique de $\lambda_1 = 0$ est quelconque alors

$$r_0 = u_2 + u_3 + \ldots + u_p$$

dans le cas où aucune composante de r_0 sur E_{λ_i} n'est nulle pour $i \in \{2, 3, \ldots, p\}$.

Preuve :
A diagonalisable implique que $\nu_i = 1, \ \forall i \in \{1, 2, \dots, p\}$ et on retrouve ainsi les hypothèses de la proposition 1.3.2, d'où la conclusion. ■

On se fixe maintenant pour objectif d'établir un lien entre les multiplicités ν_i et les décompositions sur les sous-espaces caractéristiques N_{λ_i}, ceci pour $i \in \{1, 2, \dots, p\}$. On introduit tout d'abord $\mathbb{K}[\lambda]$ l'ensemble des polynômes à une indéterminée et à coefficients dans $\mathbb{K}$ (où $\mathbb{K}$ désigne bien-sûr $\mathbb{R}$ ou $\mathbb{C}$). On utilise ensuite le lemme suivant :

Lemme 1.3.2
Soient $P_1, P_2, \dots, P_r$ des polynômes de $\mathbb{K}[\lambda]$ tels que P_i et P_j soient premiers entre eux $\forall i, j \in \{1, 2, \dots, r\}$ et $P = P_1 \times P_2 \times \dots \times P_r$, alors

$$KerP = \bigoplus_{j=1}^{r} KerP_j. \tag{1.50}$$

Preuve :
Ce résultat bien connu peut être prouvé par récurrence à l'aide de l'identité de Bezout. ■

On sait que

$$\Pi_A(\lambda) = a_0 + a_1\lambda + \dots + a_m\lambda^m = (\lambda - \lambda_{i_1})(\lambda - \lambda_{i_2})\dots(\lambda - \lambda_{i_m})$$

où $\lambda_{i_j} \in \{\lambda_1, \lambda_2, \dots, \lambda_p\}, \forall j \in \{1, 2, \dots, m\}$. Alors

$$\Pi_A(\lambda) = \lambda^{\mu_1}(\lambda - \lambda_2)^{\mu_2}\dots(\lambda - \lambda_p)^{\mu_p} \tag{1.51}$$

avec $\mu_1 + \mu_2 + \dots + \mu_p = m$ et $0 \leq \mu_i \leq \nu_i \leq \alpha_i, \ \forall i \in \{1, 2, \dots, p\}$.
On obtient le résultat suivant :

Proposition 1.3.3
On suppose que $\Pi_A(A)r_0 = 0$ alors

$$u_i \in Ker[(A - \lambda_i I)^{\mu_i}], \ \forall i \in \{1, 2, \dots, p\}. \tag{1.52}$$

Preuve :
On utilise la définition du polynôme minimal de A pour r_0 :

$$\Pi_A(A)r_0 = 0 \Leftrightarrow A^{\mu_1}(A - \lambda_2 I)^{\mu_2}\dots(A - \lambda_p I)^{\mu_p}r_0 = 0.$$

En prémultipliant $\Pi_A(A)$ à gauche par $(A-\lambda_i I)^{\nu_i-\mu_i}$,$\forall i \in \{2,3,\ldots,p\}$, on obtient

$$A^{\mu_1}(A-\lambda_2 I)^{\nu_2}\ldots(A-\lambda_p I)^{\nu_p} r_0 = 0,$$

$$\Rightarrow A^{\mu_1}(A-\lambda_2 I)^{\nu_2}\ldots(A-\lambda_p I)^{\nu_p} u_1 = 0.$$

Par conséquent $u_1 \in Ker[A^{\mu_1}(A-\lambda_2 I)^{\nu_2}\ldots(A-\lambda_p I)^{\nu_p}]$, mais $(\lambda-\lambda_i)$ et $(\lambda-\lambda_j)$ sont premiers entre eux pour $i \neq j$, $\forall i,j \in \{1,2,\ldots,p\}$, ce qui implique en utilisant le lemme 1.3.2 que

$$u_1 \in Ker(A^{\mu_1}) \oplus \bigoplus_{i=2}^{p} Ker[(A-\lambda_i I)^{\nu_i}].$$

Etant donné que $u_1 \notin Ker[(A-\lambda_i I)^{\nu_i}]$, $\forall i \in \{2,3,\ldots,p\}$, on a $u_1 \in Ker(A^{\mu_1})$.
En utilisant la même technique pour les vecteurs $u_2, u_3, \ldots, u_p$ on montre ainsi que

$$u_i \in Ker[(A-\lambda_i I)^{\mu_i}],\ \forall i \in \{1,2,\ldots,p\}. \ \blacksquare$$

Ce résultat sera très utile pour la suite : il permet de démontrer en partie la

Proposition 1.3.4
Supposons que $A \in I\!R^{n\times n}$ soit singulière et considérons que $i_0 \in \{1,2,\ldots,p\}$ alors :

$$r_0 = \sum_{\substack{i=1 \\ i\neq i_0}}^{p} u_i \Leftrightarrow \mu_{i_0} = 0, \tag{1.53}$$

où u_{i_0} est la composante de r_0 sur le sous-espace $Ker[(A-\lambda_{i_0} I)^{\nu_{i_0}}]$ et μ_{i_0} est la multiplicité de λ_{i_0} dans le polynôme minimal Π_A de A pour r_0.

Preuve :
On montre l'équivalence (1.53) à l'aide de la double implication :
supposons tout d'abord que $r_0 = \sum_{\substack{i=1 \\ i\neq i_0}}^{p} u_i$ pour $i_0 \in \{1,2,\ldots,p\}$. Alors, à l'aide de la proposition précédente, on a pour $i \neq i_0$

$$u_i \in Ker[(A-\lambda_i I)^{\mu_i}],\ \forall i \in \{1,\ldots,p\}.$$

On utilise ensuite le lemme 1.3.2 et cela implique que

$$r_0 \in Ker[A^{\mu_1}\ldots(A-\lambda_{i_0-1} I)^{\mu_{i_0-1}}(A-\lambda_{i_0+1} I)^{\mu_{i_0+1}}\ldots(A-\lambda_p I)^{\mu_p}]$$

ou encore $\prod_{\substack{i=1\\ i\neq i_0}}^{p} (A-\lambda_i I)^{\mu_i} r_0 = 0.$

Par unicité du polynôme minimal de A pour r_0, on obtient $\mu_{i_0} = 0$.

Supposons maintenant que $\mu_{i_0} = 0$: on a alors

$$\Pi_A(A) r_0 = 0 \Rightarrow A^{\mu_1} \ldots (A-\lambda_{i_0-1} I)^{\mu_{i_0-1}} (A-\lambda_{i_0+1} I)^{\mu_{i_0+1}} \ldots (A-\lambda_p I)^{\mu_p} r_0 = 0.$$

En utilisant la proposition 1.3.4 il est évident que

$$\prod_{\substack{i=1\\ i\neq i_0}}^{p} (A-\lambda_i I)^{\mu_i} u_{i_0} = 0. \tag{1.54}$$

Or $u_{i_0} \in Ker[(A-\lambda_{i_0})^{\mu_{i_0}}]$ et ceci implique que $r_0 = \sum_{\substack{i=1\\ i\neq i_0}}^{p} u_i$ puisque (1.54) est impossible à réaliser. ■

1.3.2 Résultats sur la singularité de H_m

On va utiliser les résultats précédents pour établir un résultat plus général faisant le lien entre l'inversibilité de H_m, matrice définie, et r_0 le résidu initial. En effet la proposition 1.3.1 implique que le polynôme caractéristique de H_m est égal au polynôme minimal de A pour r_0. Ainsi $\mu_1 = 0$ est équivalent au fait que H_m soit inversible.

La dernière proposition 1.3.4 ainsi que la proposition 1.3.2 vont permettre d'affirmer le résultat suivant dans le cas où m est l'itération à laquelle $h_{m+1,m} = 0$ est vérifié :

Proposition 1.3.5
Supposons que $A \in \mathbb{R}^{n\times n}$ soit singulière, telle que $Ax = b$ avec $b \in Im(A)$. Si $\lambda_1 = 0$ est telle que $\nu_1 = 1$, alors la matrice H_m produite à la m-ième itération par le processus de Hessenberg généralisé est inversible.

Preuve :
On a vu que dans ce cas de figure,

$$r_0 = u_2 + u_3 + \ldots + u_p$$

à l'aide de la proposition 1.3.2. En utilisant la proposition 1.3.4 il y a équivalence avec $\mu_1 = 0$, donc

$$\Pi_A(\lambda) = (\lambda - \lambda_2)^{\mu_2} \dots (\lambda - \lambda_p)^{\mu_p} \text{ et } \Pi_A(0) \neq 0.$$

Donc H_m n'admet pas 0 comme valeur propre, ce qui conclut la preuve. ■

Ce cas de figure peut être intégré dans un cadre plus général donné par le

Théorème 1.3.1
Supposons que $A \in I\!R^{n\times n}$ soit singulière telle que $Ax = b$ avec $b \in Im(A)$. Alors

$$H_m \text{ est inversible } \Leftrightarrow r_0 \in Im(A^{\nu_1}). \tag{1.55}$$

<u>Preuve</u>:
Supposons tout d'abord que H_m soit inversible, ceci est équivalent à $\mu_1 = 0$. D'après la proposition 1.3.4, on a $r_0 = u_2 + u_3 + \dots + u_p$.

$$\begin{aligned} \Pi_A(A)r_0 = 0 &\Rightarrow (A - \lambda_2 I)^{\mu_2} \dots (A - \lambda_p I)^{\mu_p} r_0 = 0 \\ &\Rightarrow (A - \lambda_2 I)^{\nu_2} \dots (A - \lambda_p I)^{\nu_p} r_0 = 0 \\ &\Rightarrow r_0 \in Ker[(A - \lambda_2 I)^{\nu_2} \dots (A - \lambda_p I)^{\nu_p}]. \end{aligned}$$

En utilisant la propriété (1.36) vérifiée par l'indice de A, on a $I\!R^n = \bigoplus_{i=1}^{p} N_{\lambda_i} = Ker(A^{\nu_1}) \oplus \bigoplus_{i=2}^{p} Ker[(A - \lambda_i I)^{\nu_i}]$ ce qui implique que

$$Im(A^{\nu_1}) = \bigoplus_{i=2}^{p} Ker[(A - \lambda_i I)^{\nu_i}] \tag{1.56}$$

donc $r_0 \in Im(A^{\nu_1})$.
Supposons maintenant que $r_0 \in Im(A^{\nu_1})$. Cela entraîne

$$\begin{aligned} &r_0 \in \bigoplus_{i=2}^{p} Ker[(A - \lambda_i I)^{\nu_i}] \\ &\Rightarrow (A - \lambda_2 I)^{\nu_2} \dots (A - \lambda_p I)^{\nu_p} r_0 = 0 \end{aligned}$$

$\Rightarrow \mu_1 = 0$ car le polynôme minimal de A pour r_0 est un multiple du polynôme minimal de A. ■

Remarque 1.3.1
Le théorème 1.3.1 englobe la proposition 1.3.5. En effet, si $\nu_1 = 1$, on a vu que $r_0 \in Im(A)$, ce qui est équivalent au fait que H_m soit inversible.

1.3.3 Conclusion

On a donc réussi, à l'aide de résultats classiques d'algèbre linéaire, à établir des relations entre l'inversibilité de la matrice de Hessenberg H_m et les multiplicités ν_i intervenant dans l'écriture du polynôme minimal de A. Il est bien entendu nécessaire pour la validité de ces résultats que $h_{m+1,m} = 0$ ce qui, lors des expériences numériques réalisées avec de grands systèmes linéaires, est très rarement le cas. En effet, le critère d'arrêt du programme porte sur la norme résiduelle et si cette dernière est inférieure ou égale à une certaine précision ϵ, l'algorithme est arrêté. Par conséquent, il est nécessaire de généraliser nos résultats lorsque la méthode utilisée est stoppée prématurément. Lorsque la condition $h_{m+1,m} = 0$ est vérifiée, la connaissance du paramètre ν_1 est indispensable pour assurer l'inversibilité de H_m mais cette seconde condition est également difficile à observer. Les résultats obtenus ont, pour l'instant, un intérêt purement théorique mais établissent clairement que le choix du vecteur initial x_0 revêt dans l'algorithme de Hessenberg généralisé une importance capitale.

Chapitre 2

Les méthodes de sous-espaces de Krylov et les systèmes singuliers

La nécessité de travailler en arithmétique exacte constitue l'inconvénient majeur des démarches du chapitre précédent. On se doute que la détermination d'une solution de Krylov dans ce cadre particulier dépend essentiellement du caractère inversible de la matrice H_k. Cette matrice intervient de manière différente selon le type de méthode utilisé.
On introduit dans ce chapitre les deux principales classes de méthodes de résolution de systèmes linéaires reposant sur le processus de Hessenberg généralisé. Il s'agit de la classe des méthodes RO (ou méthodes de Galerkin) et de la classe des méthodes RM (ou méthodes de semi-minimisation du résidu). On verra lors de la présentation de ces méthodes que la première classe met en évidence le rôle primordial de l'inversibilité de H_k, tandis que la seconde utilise la matrice pseudo-inverse de H_k, ce qui induit le caractère non-singulier de $\overline{H}_k^t\overline{H}_k$. Lorsque la condition $h_{k+1,k} = 0$ est vérifiée, on a $\overline{H}_k^t\overline{H}_k = H_k^2$ et on peut utiliser les résultats du chapitre précédent pour les méthodes RM et RO. Lorsque l'algorithme est arrêté prématurément, la dernière égalité n'est plus vérifiée. On exprime alors le besoin de généraliser les résultats dégagés dans le chapitre 1 et cela sera réalisé grâce, notamment, aux rotations de Givens utilisées dans la factorisation QR de H_k.

Dans une première partie, on établit une classification des méthodes de sous-espaces de Krylov selon leur appartenance à la classe des méthodes RO ou à celle des méthodes RM. On considère dans la deuxième partie les méthodes de sous-espaces de Krylov dans le cas singulier, on met alors en évidence des résultats qui parviennent à englober ceux du chapitre 1. De nouvelles notions sur la convergence et sur la stagnation de la norme du résidu sont introduites et se voient validées dans une dernière partie traitant de la comparaison numérique des méthodes les plus fréquemment utilisées.

2.1 Les méthodes de sous-espaces de Krylov

Comme cela a été introduit précédemment, les méthodes de sous-espaces de Krylov se répartissent en deux classes, les méthodes de Galerkin et celles de semi-minimisation du résidu. Les méthodes appartenant à la seconde catégorie ont été créées afin de remédier au problème des dégénérescences survenant dans la classe des méthodes à résidus orthogonaux. Ces dégénérescences, fatales pour les algorithmes utilisés, apparaissent à la fois dans le processus de Hessenberg généralisé et lorsque la matrice H_m est singulière. Dans ce cas, l'alternative utilisée est le redémarrage de l'algorithme. Pour cette raison, on se consacrera dans les applications numériques aux méthodes à résidus minimaux. Néanmoins, on verra que les comportements de convergence des méthodes RO peuvent être déduits de ceux des méthodes RM (on pourra se référer à [18]), ceci dans le cadre des systèmes singuliers.

2.1.1 La méthode RO ou méthode des résidus orthogonaux

On a vu dans le chapitre 1 que les méthodes de Krylov consistaient à corriger le vecteur x_0 initial à l'aide d'un vecteur z_k de $\mathcal{K}_k(A, r_0)$.
Par conséquent, en posant

$$x_k = x_0 + z_k,$$

on a

$$x_k = x_0 + V_k y_k \tag{2.1}$$

étant donné que le vecteur correction z_k peut s'exprimer dans la base $\{v_1, v_2, \ldots, v_k\}$ sous la forme $z_k = V_k y_k$ avec $y_k \in I\!R^k$.
Ceci implique que

$$r_k = r_0 - AV_k y_k.$$

On sait que $r_0 = \beta_1 v_1$ avec $\beta_1 \neq 0$. Ainsi, en utilisant la relation (1.6) à l'étape k, on a

$$r_k = \beta_1 v_1 - V_{k+1}\overline{H}_k y_k = V_{k+1}(\beta_1 e_1^{(k+1)} - \overline{H}_k y_k). \tag{2.2}$$

Le résidu r_k dans le cadre des méthodes de Galerkin est complètement déterminé en imposant la condition d'orthogonalité :

$$r_k \perp v_i, \ \forall i \in \{1, 2, \ldots, k\} \tag{2.3}$$

qu'on appelle condition de Galerkin et qui peut s'écrire sous la forme matricielle

$$V_k^t r_k = 0. \tag{2.4}$$

Quand les vecteurs $v_1, v_2, \ldots, v_{k+1}$ forment une base orthonormée de $\mathcal{K}_{k+1}(A, r_0)$, V_{k+1} est orthonormale et la condition (2.3) implique, à partir de (2.2), l'égalité :

$$I_{k,k+1}(\beta_1 e_1^{(k+1)} - \overline{H}_k y_k) = 0 \tag{2.5}$$

où $I_{k,k+1}$ est la matrice identité de taille $k \times (k+1)$. On a donc

$$\beta_1 e_1^{(k)} - H_k y_k = 0. \tag{2.6}$$

Intervient alors à ce niveau le caractère non-singulier de la matrice H_k : quand H_k est inversible, la solution approchée x_k est définie et va s'écrire :

$$x_k = x_0 + V_k H_k^{-1}(\beta_1 e_1^{(k)}). \tag{2.7}$$

L'existence de cette solution à l'étape m, c'est-à-dire lorsque $h_{m+1,m} = 0$, équivaut par conséquent à la condition $r_0 \in Im(A^{\nu_1})$.

On peut maintenant écrire l'algorithme de la méthode RO :

Algorithme 2.1 : $[x_k, r_k, V_{k+1}, \overline{H}_k]$ = Hessenberg généralisé-RO(A, x_0, k, W_k)

1. initialisation :
   calculer $r_0 = b - Ax_0$,
2. $[V_{k+1}, \overline{H}_k]$ = Hessenberg généralisé (A, r_0, k, W_k),
3. calcul de la solution approchée :
   (a) calculer y_k solution de $H_k y_k = \beta_1 e_1^{(k)}$,
   (b) calculer $x_k = x_0 + V_k y_k$,
   (c) calculer $r_k = r_0 - AV_k y_k$.

On peut retrouver à l'aide de cet algorithme les trois méthodes explicites FOM (Full Orthogonalization Method), BCG (Bi-Conjugate Gradient) et la méthode de Hessenberg. Celles-ci sont obtenues en utilisant respectivement les processus d'Arnoldi, de Lanczos non-symétrique et de Hessenberg.

La méthode d'Arnoldi ou méthode FOM est due à Saad [53] :

Algorithme 2.2 : $[x_k, r_k, V_{k+1}, \overline{H}_k] = \text{FOM}(A, x_0, k)$

1. initialisation :
   calculer $r_0 = b - Ax_0$,
2. $[V_{k+1}, \overline{H}_k] = \text{Arnoldi } (A, r_0, k)$,
3. calcul de la solution approchée :
   (a) calculer y_k solution de $H_k y_k = \beta_1 e_1^{(k)}$,
   (b) calculer $x_k = x_0 + V_k y_k$,
   (c) calculer $r_k = r_0 - AV_k y_k$.

Notons qu'à chaque itération k de la méthode FOM, on a les égalités suivantes :

1. $h_{k+1,k} = \|\overline{w}\|_2$ et
2. $\|v_{k+1}\|_2 = 1$.

On a également

$$\tau_k \equiv \|r_k\|_2 = h_{k+1,k} |e_k^{(k)t} y_k|. \tag{2.8}$$

Saad a montré dans [52] que l'on peut calculer le résidu $r_k = r_k^{FOM}$ à chaque étape de l'algorithme sans avoir à calculer le vecteur correspondant y_k^{FOM}. Il a prouvé, en utilisant l'élimination de Gauss avec pivotage pour adapter la décomposition LU de H_k, la relation suivante :

$$\|r_k\|_2 = h_{k+1,k} \|r_0\|_2 |\prod_{i \in I} l_i / u_{k,k}| \tag{2.9}$$

où I est l'ensemble des indices des itérations où le pivotage n'a pas eu lieu, où l_i $(i \in I)$ sont les pivots successifs et $u_{k,k}$ le dernier élément de la matrice U.

La méthode BCG est due à Lanczos [42] et elle est donnée sous une forme mathématiquement équivalente par Fletcher [24]. C'est cette dernière que l'on considère :

Algorithme 2.3 : $[x_k, r_k, V_{k+1}, \overline{H}_k] = \text{BCG}(A, x_0, k)$

1. initialisation :
   calculer $r_0 = b - Ax_0$,

2. $[V_{k+1}, \overline{H}_k]$ = Lanczos non-symétrique (A, r_0, k),

3. calcul de la solution approchée :

   (a) calculer y_k solution de $H_k y_k = \beta_1 e_1^{(k)}$,

   (b) calculer $x_k = x_0 + V_k y_k$,

   (c) calculer $r_k = r_0 - AV_k y_k$.

Etant donné la forme tridiagonale de H_k construite à l'aide du processus de Lanczos, on n'a pas besoin de calculer ici de manière explicite la décomposition LU de cette matrice de Hessenberg.

L'un des inconvénients de cette méthode concerne l'utilisation du produit $A^t u$ impossible à calculer dans le cas où, par exemple, seule l'action de la matrice A sur un vecteur est disponible.

La méthode de Hessenberg est due à Hessenberg et a été développée par Sadok [60] :

Algorithme 2.4 : $[x_k, r_k, V_{k+1}, \overline{H}_k]$ = Méthode de Hessenberg (A, x_0, k, W_k)

1. initialisation : calculer $r_0 = b - Ax_0$,

2. $[V_{k+1}, \overline{H}_k]$ = Hessenberg (A, r_0, k, W_k),

3. calcul de la solution approchée :

   (a) calculer y_k solution de $H_k y_k = \beta_1 e_1^{(k)}$,

   (b) calculer $x_k = x_0 + V_k y_k$,

   (c) calculer $r_k = r_0 - AV_k y_k$.

Dans le cas où A est symétrique, on a vu qu'on pouvait utiliser l'algorithme de Lanczos symétrique pour construire une base de $\mathcal{K}_k(A, r_0)$. Supposons donc que la factorisation LU de la matrice H_k soit définie, on a alors

$$H_k = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ \delta_2 & 1 & \ddots & & \vdots \\ 0 & \delta_3 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \delta_k & 1 \end{pmatrix} \times \begin{pmatrix} \eta_1 & \gamma_2 & 0 & \dots & 0 \\ 0 & \eta_2 & \gamma_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \gamma_k \\ 0 & \dots & \dots & 0 & \eta_k \end{pmatrix} = L_k U_k. \tag{2.10}$$

On pose $\delta_1 = 0,\ \gamma_1 = 0,\ \eta_1 = \alpha_1$. On peut calculer les coefficients δ_j et η_j par récurrence en effectuant les calculs :

$$\left\{ \begin{array}{l} \delta_j = \gamma_j / \eta_{j-1} \\ \eta_j = \alpha_j - \delta_j \gamma_j \end{array} \right. ,\ \forall j \in \{2, 3, \ldots, k\}. \tag{2.11}$$

On peut à présent réécrire la solution approchée x_k sous la forme :

$$x_k = x_0 + V_k U_k^{-1} L_k^{-1}(\beta_1 e_1^{(k)}) = x_0 + P_k d_k$$

en posant $P_k = V_k U_k^{-1}$ et $d_k = L_k^{-1}(\beta_1 e_1^{(k)})$ où les vecteurs $p_1, p_2, \ldots, p_k$ seront définis comme étant les vecteurs colonnes de P_k. Du fait de la structure de L_k, on peut mettre d_k sous la forme :

$$d_k = \left(\begin{array}{c} d_{k-1} \\ \xi_k \end{array} \right)$$

avec $\xi_k = -\delta_k \xi_{k-1}$ et $d_{k-1} \in I\!R^{k-1}$. Cela permet d'exprimer x_k en fonction de x_{k-1} :

$$x_k = x_{k-1} + \xi_k p_k.$$

On obtient ainsi l'algorithme SDFOM (Symmetric Direct FOM) qui nécessite beaucoup moins de stockage que la version initiale. La matrice $\overline{H}_k$ n'est plus stockée mais remplacée par le stockage d'un petit nombre de coefficients. Cet algorithme s'écrit de la façon suivante :

Algorithme 2.5 : $[x_k, r_k] = \text{SDFOM}(A, x_0, k)$

1. initialisation :
   calculer $r_0 = b - Ax_0,\ \beta_1 = \|r_0\|_2,\ v_1 = r_0/\beta_1$,
   $\delta_1 = \gamma_1 = 0,\ \xi_1 = \beta_1$ et $p_0 = 0$,

2. itérations : pour $j = 1, 2, \ldots, k$ faire

   (a) $\omega = Av_j - \gamma_j v_{j-1}$,

   (b) $\alpha_j = (\omega, v_j)$ et $\omega = \omega - \alpha_j v_j$,

   (c) $\gamma_{j+1} = \|\omega\|_2$,
   si $\gamma_{j+1} \neq 0$ alors $v_{j+1} = \omega/\gamma_{j+1}$ sinon stop,

   (d) si $j > 1$ alors $\delta_j = \gamma_j/\eta_{j-1}$ et $\xi_j = -\delta_j \xi_{j-1}$,

   (e) $\eta_j = \alpha_j - \delta_j \gamma_j$ et $p_j = (v_j - \gamma_j p_{j-1})/\eta_j$,

   (f) calculer la solution approchée $x_j = x_{j-1} + \xi_j p_j$,

   fin pour.

Cet algorithme est mathématiquement équivalent à celui de FOM lorsque les matrices considérées sont symétriques. Il est alors possible en réécrivant SDFOM d'une certaine façon de retrouver l'algorithme du Gradient Conjugué de Hestenes et Stiefel [33].

On a ainsi défini la méthode RO et développé les cas particuliers auxquels elle donne lieu, ceci à l'aide des processus introduits dans le premier chapitre. Une étude similaire peut être développée dans le cadre de la méthode RM.

2.1.2 La méthode RM ou méthode des résidus minimaux

On se trouve maintenant dans le cas où le sous-espace des contraintes $\mathcal{L}_k$ vérifie :

$$\mathcal{L}_k = A\mathcal{K}_k(A, r_0)$$

et r_k, dans le cadre des méthodes RM, est complètement déterminé en imposant la condition d'orthogonalité :

$$(AV_k)^t r_k = 0. \tag{2.12}$$

On introduit ensuite la matrice Z_k symétrique semi-définie positive, définie par $Z_k = (V_{k+1}^L)^t V_{k+1}^L$. La matrice V_{k+1} est introduite dans le chapitre 1 et V_{k+1}^L est une matrice inverse généralisée à gauche de V_{k+1} (pour plus de détails, on se référera à [34]). Du fait des propriétés de la matrice Z_k, on peut définir le produit scalaire et la norme induite par cette dernière :

Définition 2.1.1
Le Z_k-produit scalaire et la Z_k-norme correspondante sont définis de manière respective par :

$$\forall x, y \in \mathcal{K}_{k+1}(A, r_0),\ (x, y)_{Z_k} = (x, Z_k y)\ \ et\ |x|_{Z_k} = ((x, x)_{Z_k})^{1/2}.$$

La condition d'orthogonalité (2.12) est alors équivalente à :

$$r_k = arg \min_{v \in \mathcal{K}_k(A, r_0)} |r_0 - Av|_{Z_k}. \tag{2.13}$$

On parlera alors de méthode de projection à résidu minimal (méthode RM). La condition (2.12) s'écrit :

$$\overline{H}_k^t(\beta_1 e_1^{(k+1)} - \overline{H}_k y_k) = 0, \tag{2.14}$$

en utilisant les relations (2.2) et (1.12). On retrouve les équations normales du problème

$$\left\{\begin{array}{l} y_k = arg \min\limits_{y \in I\!R^k} \|\beta_1 e_1^{(k+1)} - \overline{H}_k y\|_2, \\ r_k = arg \min\limits_{v \in \mathcal{K}_k(A, r_0)} |r_0 - Av|_{Z_k}. \end{array}\right. \tag{2.15}$$

Le calcul de y_k nécessite la résolution d'un problème des moindres carrés linéaire de dimension $(k+1) \times k$. Cela se fera à l'aide de la factorisation QR de la matrice $\overline{H}_k$. On a de plus :

$$x_k = x_0 + V_k \overline{H}_k^{\dagger}(\beta_1 e_1^{(k+1)}) \tag{2.16}$$

en utilisant (2.1) et (2.14), avec $\overline{H}_k^{\dagger} = (\overline{H}_k^t \overline{H}_k)^{-1} \overline{H}_k^t$ la matrice pseudo-inverse de $\overline{H}_k$. Ainsi, en modifiant l'étape 3 de la méthode RO, on obtient la méthode RM :

Algorithme 2.6 : $[x_k, r_k, V_{k+1}, \overline{H}_k]$ = Hessenberg généralisé-RM(A, x_0, k, W_k)

1. initialisation :
   calculer $r_0 = b - Ax_0$,
2. construction de la base : $[V_{k+1}, \overline{H}_k]$ = Hessenberg généralisé (A, r_0, k, W_k),
3. calcul de la solution approchée :
   (a) calculer y_k solution de $\min\limits_{y \in I\!R^k} \|\beta_1 e_1^{(k+1)} - \overline{H}_k y\|_2$,
   (b) calculer $x_k = x_0 + V_k y_k$,
   (c) calculer $r_k = r_0 - AV_k y_k$.

En calcul exact et en l'absence de dégénérescence dans le processus de Hessenberg généralisé, les itérés x_j pour $j = 1, 2, \ldots, k-1$, sont bien définis. De plus, x_k est la solution exacte. En effet, si $h_{k+1,k} = 0$, la matrice $\overline{H}_k$ vérifie $\overline{H}_k = \begin{pmatrix} H_k \\ O \end{pmatrix}$ et $\overline{H}_k^{\dagger} = (H_k^{-1}, 0)$. Ajoutons également que $AV_k = V_k H_k$ et $\overline{H}_k^{\dagger} e_1^{(k+1)} = H_k^{-1} e_1^{(k)}$. Par conséquent

$$\begin{aligned} r_k &= b - Ax_k = r_0 - AV_k y_k = r_0 - V_k H_k y_k \\ &= r_0 - \beta_1 V_k H_k \overline{H}_k^{\dagger} e_1^{(k+1)} \\ &= r_0 - \beta_1 V_k H_k H_k^{-1} e_1^{(k)} \\ &= r_0 - \beta_1 v_1 \\ &= 0 \end{aligned} .$$

On peut retrouver, à l'aide de l'algorithme 2.6, les trois méthodes GMRES, QMR et CMRH obtenues en utilisant respectivement les processus d'Arnoldi, de Lanczos non-symétrique et de Hessenberg, qui sont des méthodes explicites.

La méthode GMRES [57] est due à Saad et Schultz : elle minimise de façon optimale le résidu sur $\mathcal{K}_{k+1}(A, r_0)$ et elle est mathématiquement équivalente aux méthodes GCR (Generalized Conjugate Residual) [23] ainsi qu'à la méthode ORTHODIR [41] :

Algorithme 2.7 : $[x_k, r_k, V_{k+1}, \overline{H}_k] = \mathrm{GMRES}(A, x_0, k)$

1. initialisation :
   calculer $r_0 = b - Ax_0$,
2. construction de la base : $[V_{k+1}, \overline{H}_k] = \text{Arnoldi}\ (A, r_0, k)$,
3. calcul de la solution approchée :
   (a) calculer y_k solution de $\min_{y \in \mathbb{R}^k} \|\beta_1 e_1^{(k+1)} - \overline{H}_k y\|_2$,
   (b) calculer $x_k = x_0 + V_k y_k$,
   (c) calculer $r_k = r_0 - AV_k y_k$.

On montrera dans la suite, à l'aide de relations sur les résidus, que les convergences de FOM et GMRES sont très liées. Du fait de la condition de minimisation de la norme de son résidu, le GMRES est souvent préféré à FOM dont la norme résiduelle peut présenter des comportements oscillatoires.

La méthode QMR [28] est due à Freund et Nachtigal. Elle permet de surmonter les problèmes rencontrés par la méthode BCG, c'est-à-dire :

1. l'irrégularité de la convergence de la méthode BCG,
2. la possibilité de non-existence de la décomposition LU du système tridiagonal associé. Alliée à une stratégie de "look-ahead", la méthode QMR permet d'éviter les dégénérescences associées au processus de Lanczos.

La méthode s'écrit de la façon suivante :

Algorithme 2.8 : $[x_k, r_k, V_{k+1}, \overline{H}_k] = \mathrm{QMR}(A, x_0, k)$

1. initialisation :
   calculer $r_0 = b - Ax_0$,
2. construction de la base : $[V_{k+1}, \overline{H}_k] = \text{Lanczos non-symétrique}\ (A, r_0, k)$,

3. calcul de la solution approchée :
   (a) calculer y_k solution de $\min_{y \in I\!R^k} \|\beta_1 e_1^{(k+1)} - \overline{H}_k y\|_2$,
   (b) calculer $x_k = x_0 + V_k y_k$,
   (c) calculer $r_k = r_0 - AV_k y_k$.

Il existe des relations entre les deux méthodes BCG et QMR et plus particulièrement une interdépendance entre les résidus de la méthode BCG et les pseudo-résidus de la méthode QMR, prouvée par Cullum et Greenbaum [18].

La méthode CMRH [58] utilise également l'idée de quasi-minimisation du résidu. Elle est obtenue comme cas particulier de la méthode *RM* par utilisation du processus de Hessenberg avec stratégie de pivotage et s'écrit de la façon suivante :

Algorithme 2.9 : $[x_k, r_k, V_{k+1}, \overline{H}_k] = \mathrm{CMRH}(A, x_0, k, W_k)$

1. initialisation :
   calculer $r_0 = b - Ax_0$,
2. construction de la base : $[V_{k+1}, \overline{H}_k] = \mathrm{Hessenberg}\ (A, r_0, k, W_k)$,
3. calcul de la solution approchée :
   (a) calculer y_k solution de $\min_{y \in I\!R^k} \|\beta_1 e_1^{(k+1)} - \overline{H}_k y\|_2$,
   (b) calculer $x_k = x_0 + V_k y_k$,
   (c) calculer $r_k = r_0 - AV_k y_k$.

On utilisera ces trois méthodes dans la suite et on comparera leurs performances lorsqu'elles sont utilisées dans le cadre singulier.

Dans le cas où A est symétrique, $\overline{H}_k$ le sera également. La matrice $R_k = Q_k^t \overline{H}_k$ est donc triangulaire supérieure (à trois diagonales) et peut s'écrire ainsi :

$$R_k = \begin{pmatrix} \rho_1 & \sigma_2 & \tau_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \tau_k \\ & & & \ddots & \sigma_k \\ & & & & \rho_k \\ & & & & 0 \end{pmatrix}. \tag{2.17}$$

On a par conséquent $Q_k^t \overline{H}_k = \begin{pmatrix} \overline{R}_k \\ 0 \end{pmatrix}$ avec $\overline{R}_k \in I\!R^{k\times k}$ matrice triangulaire supérieure non-singulière en supposant H_k inversible, c'est-à-dire en supposant $r_0 \in Im(A^{\nu_1})$.

On pose

$$\Gamma_{k+1}^{(i)} \equiv Q_i^t \beta_1 e_1^{(k+1)} = \begin{pmatrix} \gamma_1 \\ \vdots \\ \gamma_i \\ \tilde{\gamma}_{i+1} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \text{ pour } i \in \{1, 2, \ldots, k\}. \tag{2.18}$$

A l'aide de la factorisation QR de $\overline{H}_k$, on a :

$$\overline{H}_k^\dagger = (\overline{H}_k^t \overline{H}_k)^{-1} \overline{H}_k^t = (R_k^t R_k)^{-1} R_k^t Q_k^t = \overline{R}_k^{-1} \overline{R}_k^{-t} R_k^t Q_k^t \text{ donc}$$

$$\begin{aligned} x_k &= x_0 + V_k \overline{R}_k^{-1} \overline{R}_k^{-t} R_k^t (\gamma_1, \ldots, \tilde{\gamma}_{k+1})^t \\ &= x_0 + V_k \overline{R}_k^{-1} \overline{R}_k^{-t} \overline{R}_k^t (\gamma_1, \ldots, \gamma_k)^t \\ &= x_0 + V_k \overline{R}_k^{-1} (\gamma_1, \ldots, \gamma_k)^t \\ &= x_0 + P_k (\gamma_1, \ldots, \gamma_k)^t \end{aligned} \tag{2.19}$$

avec $P_k = V_k \overline{R}_k^{-1}$. Si $p_1, p_2, \ldots, p_k$ sont les colonnes de la matrice P_k on a :

$$p_k = (v_k - \sigma_k p_{k-1} - \tau_k p_{k-2}) / \rho_k$$

et on peut, à l'aide de cette relation, exprimer x_k en fonction de x_{k-1} :

$$x_k = x_{k-1} + \gamma_k p_k. \tag{2.20}$$

La matrice Q_k est obtenue comme produit implicite des matrices de rotation de Givens $F_{i,k}$ pour $i \in \{1, 2, \ldots, k\}$, de dimension $k+1$. On a $Q_k^t = F_{k,k} F_{k-1,k} \ldots F_{1,k}$ avec

$$F_{i,k} = \begin{pmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & 1 & & & & & \\ & & & c_i & s_i & & & \\ & & & -s_i & c_i & & & \\ & & & & & 1 & & \\ & & & & & & \ddots & \\ & & & & & & & 1 \end{pmatrix}, \ \forall i \in \{1, 2, \ldots, k\}. \tag{2.21}$$

Si on appelle x_k^{RO} la solution obtenue par une méthode de résidu orthogonal à l'étape k et x_k^{RM} une solution obtenue par une méthode de minimisation du résidu à cette même étape, ces deux solutions approchées sont reliées entre elles par les sinus et les cosinus s_k et c_k des angles de la rotation de Givens $F_{k,k}$:

$$x_k^{RM} = s_k^2 x_{k-1}^{RM} + c_k^2 x_k^{RO}. \tag{2.22}$$

On a également les relations

$$\gamma_k = c_k \widetilde{\gamma}_k \tag{2.23}$$

$$\widetilde{\gamma}_{k+1} = -s_k \widetilde{\gamma}_k \tag{2.24}$$

avec $\widetilde{\gamma}_1 = \beta_1$.

On peut désormais écrire l'algorithme SDGMRES (Symmetric Direct GMRES) bien moins onéreux en coût de stockage que l'algorithme dont il provient :

Algorithme 2.10 : $[x_k, r_k] = \text{SDGMRES}(A, x_0, k)$

1. initialisation :
   calculer $r_0 = b - Ax_0$, $\beta_1 = \|r_0\|_2$, $v_1 = r_0/\beta_1, \widetilde{\gamma}_1 = \beta_1$,
   $p_{-1} = p_0 = 0$ et $c_{-1} = c_0 = s_{-1} = s_0 = 0$,
2. itérations : pour $j = 1, 2, \ldots, k$ faire
   (a) calcul de la k-ième colonne de H_k,
   $\omega = Av_j - \beta_j v_{j-1}$,
   $\alpha_j = (\omega, v_j)$ et $\omega = \omega - \alpha_j v_j$,
   $\beta_{j+1} = \|\omega\|_2$, si $\beta_{j+1} \neq 0$ alors $v_{j+1} = \omega/\beta_{j+1}$ sinon stop,
   (b) calcul de la k-ième colonne de R_k :
   $\tau_j = s_{j-2}\beta_j$ et $\varepsilon = c_{j-2}\beta_j$,
   $\sigma_j = c_{j-1}\varepsilon + s_{j-1}\alpha_j$ et $\varepsilon = -s_{j-1}\varepsilon + c_{j-1}\alpha_j$,
   calcul des coefficients c_j et s_j correspondant à ε et β_{j+1}
   $\rho_j = c_j\varepsilon + s_j\beta_{j+1}$,
   (c) mise à jour de Γ :
   $\gamma_j = c_j\widetilde{\gamma}_j$ et $\widetilde{\gamma}_{j+1} = -s_j\widetilde{\gamma}_j$,
   (d) calcul de la solution approchée :
   $p_j = (v_j - \sigma_j p_{j-1} - \tau_j p_{j-2})/\rho_j$,
   $x_j = x_{j-1} + \gamma_j p_j$,
   $r_j = r_{j-1} - \gamma_j A p_j$.

On a vu précédemment que l'algorithme du gradient conjugué se déduisait de la méthode FOM dans le cadre des matrices symétriques. On peut remarquer de la même façon, lorsque A est symétrique, que la méthode MINRES [47] peut être dérivée de la méthode GMRES.

Il est évident que la méthode RO utilisée va subir une dégénérescence suivie de l'arrêt de son algorithme lorsque H_k est singulière. Aucune étude ne permet pour l'instant de prévoir avec certitude ce type de singularité aussi se consacre-t-on essentiellement aux méthodes RM introduites pour pallier à ce genre de problèmes. On souhaite mettre en évidence les conditions nécessaires et suffisantes qui permettront d'assurer l'existence d'une solution de Krylov ainsi que sa détermination.

2.2 Résolution des systèmes singuliers à l'aide de la méthode RM

Le but de cette section est d'étudier le comportement des méthodes RM ou méthodes à résidu minimal dans le cadre des systèmes singuliers. On verra que l'arithmétique exacte n'est plus nécessaire pour établir des résultats concernant l'obtention d'une solution de Krylov. L'étude repose essentiellement sur la factorisation QR de la matrice de Hessenberg utilisée lors du processus de Hessenberg généralisé.

2.2.1 Résultats pratiques

2.2.1.a Préliminaires

On a vu dans la section précédente que

$$r_k = arg \min_{v \in \mathcal{K}_k(A,r_0)} |r_0 - Av|_{Z_k}. \tag{2.25}$$

Cette minimisation équivaut à déterminer y_k tel que

$$\overline{H}_k^t(\beta_1 e_1^{(k+1)} - \overline{H}_k y_k) = 0. \tag{2.26}$$

On considère les matrices de rotation de Givens $F_{j,k} \in I\!R^{(k+1)\times(k+1)}$ pour $j = 1, 2, \dots, k$ introduites précédemment, telles que $Q_k^t = F_{k,k}F_{k-1,k}\dots F_{1,k}$ transforme de façon générale $\overline{H}_k$ en une matrice triangulaire supérieure $R_k \in I\!R^{(k+1)\times k}$ dont la dernière ligne est composée uniquement de zéros. On a bien évidemment

$$Q_k^t\overline{H}_k = R_k = \begin{pmatrix} & \overline{R}_k & \\ 0 & \dots & 0 \end{pmatrix}. \tag{2.27}$$

$F_{j,k}$ et $F_{j,k+1}$ diffèrent uniquement par l'addition d'une ligne et d'une colonne de la matrice identité. Rappelons que

$$F_{k-1,k}F_{k-2,k}\dots F_{1,k}\overline{H}_k = \begin{pmatrix} * & \dots & * & * \\ & \ddots & \vdots & \vdots \\ & & * & * \\ & & 0 & \rho_k \\ 0 & & 0 & h_{k+1,k} \end{pmatrix}, \tag{2.28}$$

où un astérisque correspond à un élément calculé. Les coefficients de Givens sont choisis de telle sorte que

$$c_k = \frac{\rho_k}{(\rho_k^2 + h_{k+1,k}^2)^{1/2}} = \frac{\rho_k}{\gamma_k} \text{ et} \tag{2.29}$$

$$s_k = \frac{h_{k+1,k}}{(\rho_k^2 + h_{k+1,k}^2)^{1/2}} = \frac{h_{k+1,k}}{\gamma_k}. \tag{2.30}$$

A l'aide de la factorisation QR de la matrice $\overline{H}_k$, le problème des moindres carrés (2.15) est équivalent à

$$\|\beta_1 e_1^{(k+1)} - \overline{H}_k y_k\|_2 = \min_{y \in I\!R^k} \|\beta_1 Q_k^t e_1^{(k+1)} - R_k y\|_2. \tag{2.31}$$

On considère également que

$$g_k \equiv \beta_1 Q_k^t e_1^{(k+1)} = \begin{pmatrix} \overline{g}_k \\ g \end{pmatrix}$$

avec $\overline{g}_k \in I\!R^k$ et $g \in I\!R$. Ainsi, en supposant qu'au k-ième pas $\overline{R}_k$ est non-singulière, le vecteur y_k vérifiant (2.15) est tel que

$$\overline{R}_k y_k = \overline{g}_k. \tag{2.32}$$

Donc, si $\overline{R}_k$ est inversible, la solution approchée x_k de (1.1) est définie et est donnée par

$$x_k = x_0 + V_k(\overline{R}_k)^{-1}\overline{g}_k. \tag{2.33}$$

Dans ce cas,

$$\begin{aligned} r_k &= b - Ax_k \\ &= b - Ax_0 - AV_k(\overline{R}_k)^{-1}\overline{g}_k \\ &= r_0 - V_{k+1}\overline{H}_k(\overline{R}_k)^{-1}\overline{g}_k, \end{aligned}$$

en considérant l'égalité (1.6). On utilise ensuite la factorisation QR de la matrice de Hessenberg $\overline{H}_k$. Ainsi

$$\begin{aligned} r_k &= V_{k+1}(\beta_1 e_1^{(k+1)} - \overline{H}_k(\overline{R}_k)^{-1}\overline{g}_k) \\ &= V_{k+1}Q_k(Q_k^t\beta_1 e_1^{(k+1)} - R_k(\overline{R}_k)^{-1}\overline{g}_k) \end{aligned}$$

puisque Q_k est orthogonale. Finalement, en utilisant les résultats précédents, on trouve

$$r_k = V_{k+1}Q_k(g_k - \begin{pmatrix} \overline{g}_k \\ 0 \end{pmatrix}) = V_{k+1}Q_k \begin{pmatrix} 0 \\ \vdots \\ 0 \\ g \end{pmatrix}. \tag{2.34}$$

On est donc en mesure de calculer une estimation du résidu (ou, en passant aux normes, l'estimation de la norme de ce même résidu) de manière simple et peu coûteuse à chaque étape de l'algorithme. En passant à la norme 2 chaque côté de l'égalité (2.34), on obtient

$$\|r_k\|_2 = \|V_{k+1}Q_k \begin{pmatrix} 0 \\ \vdots \\ 0 \\ g \end{pmatrix}\|_2 \leq \|V_{k+1}\|_2|g|. \tag{2.35}$$

On peut montrer que $g = (\beta_1 e_1^{(k+1)t} Q_k) e_{k+1}^{(k+1)}$. Ainsi, en posant $Q_k = [q_1^{(k)}, q_2^{(k)}, \ldots, q_{k+1}^{(k)}]$, on obtient

$$\|r_k\|_2 \leq \|V_{k+1}\|_2 \beta_1 |(q_{k+1}^{(k)})^t e_1^{(k+1)}|.$$

Mais il a été montré par Brown dans [10] que

$$q_{k+1}^{(k)} = (\prod_{i=1}^{k} s_i, c_1 \prod_{i=2}^{k} s_i, c_2 \prod_{i=3}^{k} s_i, \ldots, c_{k-1}s_k, c_k)^t, \tag{2.36}$$

donc $|g| = \beta_1 |\prod_{i=1}^{k} s_i|$ et finalement

$$\|r_k\|_2 \leq \|V_{k+1}\|_2 \beta_1 |\prod_{i=1}^{k} s_i|. \tag{2.37}$$

On rappelle que tous ces calculs sont réalisables si et seulement si la matrice $\overline{R}_k$ est inversible. On peut donc se demander sous quelles conditions l'hypothèse précédente est vérifiée. Dans ce but, on souhaite exprimer les coefficients de la matrice $\overline{R}_k$ à l'aide des coefficients c_j et s_j intervenant dans les rotations de Givens utilisées dans la factorisation QR précédente [62]. On pose

$$\overline{R}_k = \begin{pmatrix} r_{1,1} & \ldots & r_{1,k} \\ & \ddots & \vdots \\ 0 & & r_{k,k} \end{pmatrix} \in I\!R^{k \times k} \tag{2.38}$$

et on s'intéresse à la j-ième colonne de cette matrice. On peut montrer que

$$\begin{pmatrix} r_{1,j} \\ \vdots \\ r_{j,j} \end{pmatrix} = F_{j,j}F_{j-1,j} \ldots F_{1,j} \begin{pmatrix} h_{1,j} \\ \vdots \\ h_{j,j} \end{pmatrix}, \text{ pour } j \in \{1, 2, \ldots, k\}.$$

On remarque alors que
$r_{i,j} = c_i[-s_{i-1}(-s_{i-2}(\ldots(-s_2(-s_1h_{1,j}+c_1h_{2,j})+c_2h_{3,j})+\ldots)+c_{i-2}h_{i-1,j})+c_{i-1}h_{i,j}]+s_ih_{i+1,j}$

et cette relation est vraie pour tout $i \in \{1,2,\ldots,j\}$, $\forall j \in \{1,2,\ldots,k\}$. Cette égalité peut s'écrire sous une forme plus compacte

$$r_{i,j} = c_i\left[\sum_{l=1}^{i}\left(\prod_{q=l}^{i-1}(-1)s_q\right)c_{l-1}h_{l,j}\right] + s_ih_{i+1,j} \text{ avec } i \le j,\ \forall i,j \in \{1,2,\ldots,k\}. \quad (2.39)$$

Il s'ensuit, en utilisant les relations (2.29) et (2.39), que

$$c_i = \frac{\rho_i}{\gamma_i} = \frac{\sum_{l=1}^{i}\left(\prod_{q=l}^{i-1}(-1)s_q\right)c_{l-1}h_{l,j}}{\gamma_i} \text{ avec } i \le j,\ \forall i,j \in \{1,2,\ldots,k\} \quad (2.40)$$

$$s_i = \frac{h_{i+1,j}}{\gamma_i} \text{ avec } i \le j,\ \forall i,j \in \{1,2,\ldots,k\}. \quad (2.41)$$

On suppose maintenant qu'aucun problème ne survient dans la résolution du système (2.32) jusqu'à l'étape $k-1$. Ainsi l'inversibilité de R_k dépend uniquement de $r_{k,k}$. On utilise alors (2.39) pour $i,j=k$ et cela implique les égalités

$$\begin{aligned} r_{k,k} &= c_k\left[\sum_{l=1}^{k}\left(\prod_{i=l}^{k-1}(-1)s_i\right)c_{l-1}h_{l,k}\right] + s_kh_{k+1,k} \\ &= \left(\left(\sum_{l=1}^{k}\left(\prod_{i=l}^{k-1}(-1)s_i\right)c_{l-1}h_{l,k}\right)^2 + h_{k+1,k}^2\right)^{1/2} = \gamma_k. \quad (2.42) \end{aligned}$$

On remarque que l'expression précédente fait intervenir les coefficients de $\overline{H}_k$. Cela suggère par conséquent qu'il existe une relation entre γ_k, H_k et $\overline{H}_k$. On peut prouver la proposition suivante

Proposition 2.2.1

$$\det(H_k) = c_k\prod_{i=1}^{k}\gamma_i \quad \forall k, \quad (2.43)$$

$$det(\overline{H}_k^t\overline{H}_k) = \prod_{i=1}^{k}\gamma_i^2 \quad \forall k. \quad (2.44)$$

Preuve:
On peut prouver les deux expressions précédentes par récurrence. Tout d'abord, en utilisant la relation (2.29) pour $k=1$, on obtient

$$c_1 = \frac{h_{1,1}}{(h_{1,1}^2 + h_{2,1}^2)^{1/2}} = \frac{det(H_1)}{\gamma_1}.$$

De plus, $det(\overline{H}_1^t\overline{H}_1) = h_{1,1}^2 + h_{2,1}^2 = \gamma_1^2$. Ainsi les relations (2.43) et (2.44) sont vraies pour $k = 1$.
On considère ensuite le déterminant de H_{k+1},

$$det(H_{k+1}) = \begin{vmatrix} h_{1,1} & h_{1,2} & \dots & \dots & h_{1,k+1} \\ h_{2,1} & \ddots & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & h_{k,k+1} \\ 0 & \dots & 0 & h_{k+1,k} & h_{k+1,k+1} \end{vmatrix}.$$

En développant ce déterminant de manière récurrente par rapport à la dernière ligne, on obtient

$$det(H_{k+1}) = h_{k+1,k+1}det(H_k) - h_{k+1,k}(h_{k,k+1}det(H_{k-1}) - h_{k,k-1}(h_{k-1,k+1}det(H_{k-2}) -$$
$$\dots - h_{3,2}(h_{2,k+1}det(H_1) - h_{2,1}h_{1,k+1}))).$$

Cette égalité peut, tout comme (2.39), s'écrire sous une forme plus compacte en posant auparavant $det(H_0) = 1$. Il s'ensuit que

$$det(H_{k+1}) = \sum_{l=1}^{k+1}\left(\prod_{i=l}^{k}(-1)h_{i+1,i}\right) det(H_{l-1})h_{l,k+1},\ \forall k. \tag{2.45}$$

On utilise ensuite (2.43), c'est-à-dire l'hypothèse de récurrence, ainsi que la relation (2.41) et on obtient

$$\begin{aligned} det(H_{k+1}) &= \sum_{l=1}^{k+1}\left(\prod_{i=l}^{k}(-1)s_i\gamma_i\right) c_{l-1}\prod_{i=1}^{l-1}\gamma_i h_{l,k+1} \\ &= \prod_{i=1}^{k}\gamma_i\sum_{l=1}^{k+1}\left(\prod_{i=l}^{k}(-1)s_i\right) c_{l-1}h_{l,k+1}. \end{aligned}$$

L'égalité (2.40) permet de conclure car la relation précédente est équivalente à $det(H_{k+1}) = c_{k+1}\prod_{i=1}^{k+1}\gamma_i$. Ainsi la première partie de la preuve est démontrée.
Considérons maintenant le déterminant de $\overline{H}_{k+1}^t\overline{H}_{k+1}$:

$$det(\overline{H}_{k+1}^t\overline{H}_{k+1}) = det((Q_{k+1}R_{k+1})^t(Q_{k+1}R_{k+1})) = det(R_{k+1}^tR_{k+1})$$

puisque $Q_{k+1} \in I\!R^{(k+1)\times(k+1)}$ est une matrice orthogonale. En utilisant la forme particulière de R_{k+1}, il suit que

$$R_{k+1}^tR_{k+1} = \begin{pmatrix} & 0 \\ \overline{R}_{k+1}^t & \vdots \\ & 0 \end{pmatrix}\begin{pmatrix} & \overline{R}_{k+1} & \\ 0 & \dots & 0 \end{pmatrix} = \overline{R}_{k+1}^t\overline{R}_{k+1}. \tag{2.46}$$

Ceci implique immédiatement que

$$det(\overline{H}_{k+1}^t \overline{H}_{k+1}) = det(\overline{R}_{k+1}^t \overline{R}_{k+1}) = [det(\overline{R}_{k+1})]^2. \tag{2.47}$$

Puisque $\overline{R}_{k+1}$ est triangulaire, on obtient

$$det(\overline{R}_{k+1}) = \prod_{j=1}^{k+1} r_{j,j}. \tag{2.48}$$

Finalement, en utilisant les résultats précédents, on a

$$det(\overline{H}_{k+1}^t \overline{H}_{k+1}) = \left(\prod_{j=1}^{k+1} r_{j,j} \right)^2 = r_{k+1,k+1}^2 \left(\prod_{j=1}^{k} r_{j,j} \right)^2 = r_{k+1,k+1}^2 det(\overline{H}_k^t \overline{H}_k). \tag{2.49}$$

On utilise l'hypothèse de récurrence (2.44) ainsi que la relation (2.42) (au rang k) pour conclure. En effet,

$$det(\overline{H}_{k+1}^t \overline{H}_{k+1}) = \prod_{i=1}^{k+1} \gamma_i^2.$$

La proposition est maintenant démontrée. ■

En utilisant les relations (2.42) et (2.49), on déduit que

$$r_{k,k}^2 = \gamma_k^2 = \frac{det(\overline{H}_k^t \overline{H}_k)}{det(\overline{H}_{k-1}^t \overline{H}_{k-1})}.$$

De plus

$$r_{k,k} = c_k \frac{det(H_k)}{\prod_{i=1}^{k-1} \gamma_i} + s_k h_{k+1,k}$$

à l'aide de (2.39) et (2.43) pour $i = k$. Cela fournit donc une nouvelle relation entre $det(\overline{H}_k^t \overline{H}_k)$ et $det(\overline{H}_{k-1}^t \overline{H}_{k-1})$, à savoir

$$det(\overline{H}_k^t \overline{H}_k) = \left(c_k \frac{det(H_k)}{\prod_{i=1}^{k-1} \gamma_i} + s_k h_{k+1,k} \right)^2 det(\overline{H}_{k-1}^t \overline{H}_{k-1}). \tag{2.50}$$

En utilisant simultanément les relations (2.43) et (2.44), on obtient

$$\frac{det(H_k^t H_k)}{det(\overline{H}_k^t \overline{H}_k)} = \frac{det(H_k)^2}{det(\overline{H}_k^t \overline{H}_k)} = c_k^2, \tag{2.51}$$

ces résultats ont également été obtenus par Sadok dans [59].

Il est possible à présent de connaître les conditions sous lesquelles l'hypothèse $r_{k,k} = 0$ est vérifiée. On a

$$r_{k,k} = \left(\frac{det(H_k)^2}{det(\overline{H}_{k-1}^t \overline{H}_{k-1})} + h_{k+1,k}^2 \right)^{1/2} = \gamma_k.$$

On en déduit la proposition suivante :

Proposition 2.2.2
Soit $\overline{H}_k = Q_k R_k$ la décomposition QR de $\overline{H}_k$. Si $r_{k,k}$ est le k-ième élément diagonal de R_k,

$$r_{k,k} = 0 \Leftrightarrow \left\{ \begin{array}{l} det(H_k) = 0, \\ h_{k+1,k} = 0. \end{array} \right. \tag{2.52}$$

On suppose alors que $r_{j,j} \neq 0, \ \forall j \in \{1, 2, \ldots, k-1\}$, c'est-à-dire que le problème (2.31) admet une solution pour toute étape précédant la k-ième itération. Le problème des moindres carrés (2.31) n'admettra ainsi aucune solution à la k-ième étape si et seulement si H_k est singulière et $h_{k+1,k} = 0$ (autrement dit si un sous-espace invariant par A a été trouvé). Ce cas ne peut jamais survenir lorsqu'on résout un système non-singulier. En effet, les deux conditions ne peuvent être vérifiées simultanément.

La relation (2.50) pose problème car elle s'exprime à l'aide des termes c_k et s_k. On souhaite utiliser uniquement les coefficients de $\overline{H}_k$. A l'aide de (2.29) ainsi que de l'égalité

$$\rho_k = \frac{det(H_k)}{\prod_{i=1}^{k-1} \gamma_i^2}$$

on a

$$c_k = \frac{det(H_k)}{\left(det(H_k)^2 + \prod_{i=1}^{k-1} \gamma_i^2 h_{k+1,k}^2 \right)^{1/2}}.$$

Or

$$c_k = \frac{det(H_k)}{(det(\overline{H}_k^t \overline{H}_k))^{1/2}}$$

donc

$$det(\overline{H}_k^t \overline{H}_k) = det(H_k^t H_k) + \prod_{i=1}^{k-1} \gamma_i^2 h_{k+1,k}^2.$$

En utilisant la relation (2.44), on obtient

$$det(\overline{H}_k^t \overline{H}_k) = det(H_k^t H_k) + det(\overline{H}_{k-1}^t \overline{H}_{k-1}) h_{k+1,k}^2. \tag{2.53}$$

Puisque $det(\overline{H}_{k-1}^t\overline{H}_{k-1})h_{k+1,k}^2 \geq 0$, on remarque que $det(\overline{H}_k^t\overline{H}_k) \geq det(H_k^tH_k)$ à chaque pas de la méthode RM.

Appelons maintenant valeurs singulières de Ritz et valeurs singulières adjacentes de Ritz les racines carrées des valeurs propres respectives de $H_k^tH_k$ et $\overline{H}_k^t\overline{H}_k$. On pourra voir [59] pour plus de détails. Soient respectivement $(\alpha_i^{(k)})$ et $(\beta_i^{(k)})$ pour $i = 1, 2, \ldots, k$, les valeurs singulières des matrices H_k et $\overline{H}_k$. On ordonne ces valeurs de la manière suivante :

$$\alpha_k^{(k)} \leq \alpha_{k-1}^{(k)} \leq \ldots \leq \alpha_1^{(k)} \text{ et } \beta_k^{(k)} \leq \beta_{k-1}^{(k)} \leq \ldots \leq \beta_1^{(k)}. \tag{2.54}$$

On donne dans la proposition ci-dessous quelques propriétés d'entrelacement reliant les valeurs singulières des matrices H_k et $\overline{H}_k$:

Proposition 2.2.3

1. $\beta_i^{(k)} \in [\alpha_i^{(k)}, \alpha_{i-1}^{(k)}]$ *pour* $i = 2, 3, \ldots, k$ *et* $\beta_1^{(k)} \geq \alpha_1^{(k)}$,
2. $\alpha_i^{(k)} \in [\beta_{i+1}^{(k)}, \beta_i^{(k)}]$ *pour* $i = 1, 2, \ldots, k-1$ *et* $\beta_k^{(k)} \geq \alpha_k^{(k)}$,
3. $\alpha_i^{(k-1)} \in [\alpha_{i+1}^{(k)}, \alpha_i^{(k)}]$ *pour* $i = 1, 2, \ldots, k-2$ *et* $\alpha_{k-1}^{(k)} \geq \alpha_{k-1}^{(k-1)}$,
4. $\beta_i^{(k-1)} \in [\beta_{i+1}^{(k)}, \beta_i^{(k)}]$ *pour* $i = 1, 2, \ldots, k-1$.

On trouvera la preuve de ces inégalités dans [59].

On sait que $det(\overline{H}_k^t\overline{H}_k) = 0 \Leftrightarrow r_{k,k} = 0 \Leftrightarrow \gamma_k = 0$. Si à l'étape k de l'algorithme la relation $r_{k,k} = 0$ est vérifiée, le problème des moindres carrés (2.31) n'admet aucune solution. Ainsi, le système (1.1) n'admet pas de solution de Krylov et les vecteurs x_k et r_k ne sont pas définis. Si $det(\overline{H}_k^t\overline{H}_k) = 0$, cela signifie qu'une valeur singulière adjacente de Ritz a convergé vers 0.

En suivant les notations de Cullum et Greenbaum dans [18], si $y_k^{(1)}$ décrit la solution du problème des moindres carrés

$$\min_{y \in \mathbb{R}^k} \|\overline{H}_k y - \beta_1 e_1^{(k+1)}\|_2$$

et si $\|\overline{H}_k y_k^{(1)} - \beta_1 e_1^{(k+1)}\|_2 = |r_k^{(1)}|_{Z_k}$, alors $|r_k^{(1)}|_{Z_k} = \beta_1 |s_1 s_2 \ldots s_k|$. Pour la méthode RO, si H_k est inversible et si $y_k^{(2)}$ décrit la solution du système linéaire $H_k y = \beta_1 e_1^{(k)}$, en

posant $\|\overline{H}_k y_k^{(2)} - \beta_1 e_1^{(k+1)}\|_2 = |r_k^{(2)}|_{Z_k}$, on a $|r_k^{(2)}|_{Z_k} = \beta_1 \frac{1}{c_k}|s_1 s_2 \dots s_k|$.
On dispose ainsi des relations ci-dessous :

Proposition 2.2.4

1. $|r_k^{(2)}|_{Z_k} = \dfrac{|r_k^{(1)}|_{Z_k}}{\left(1 - (|r_k^{(1)}|_{Z_k}/|r_{k-1}^{(1)}|_{Z_k})^2\right)^{1/2}}$ ou encore $\left(\dfrac{|r_k^{(1)}|_{Z_k}}{|r_k^{(2)}|_{Z_k}}\right)^2 + \left(\dfrac{|r_k^{(1)}|_{Z_k}}{|r_{k-1}^{(1)}|_{Z_k}}\right)^2 = 1.$

2. $\dfrac{|r_k^{(1)}|_{Z_k}}{|r_{k-1}^{(1)}|_{Z_k}} = |s_k|$

3. $\dfrac{|r_k^{(1)}|_{Z_k}}{|r_k^{(2)}|_{Z_k}} = |c_k|$

4. $\dfrac{|r_k^{(2)}|_{Z_k}}{|r_{k-1}^{(2)}|_{Z_k}} = |s_k|\dfrac{|c_{k-1}|}{|c_k|}$

<u>Preuve</u> :
On a vu à l'aide de (2.36) que

$$|g| = \beta_1 |\prod_{i=1}^{k} s_i|$$

donc $|r_k^{(1)}|_{Z_k} = \beta_1 |s_1 s_2 \dots s_k|$.
Pour le système linéaire $H_k y = \beta_1 e_1^{(k)}$, la solution $y_k^{(2)}$ vérifie

$$|r_k^{(2)}|_{Z_k} = \|\zeta_k\|_2 \text{ si } \zeta_k = \overline{H}_k H_k^{-1} \beta_1 e_1^{(k)} - \beta_1 e_1^{(k+1)}.$$

Le vecteur ζ_k est nul, excepté pour la dernière composante qui est égale au produit de $\beta_1 h_{k+1,k}$ par la $(k,1)$-ième entrée de H_k^{-1}. Factorisons H_k sous la forme $H_k = Q_k^{(2)} R_k^{(2)}$ où $Q_k^{(2)t} = F_{k-1,k}^{(2)} F_{k-2,k}^{(2)} \dots F_{1,k}^{(2)} R_k^{(2)}$ et $F_{i,k}^{(2)}$ est la sous-matrice principale de taille $k \times k$. On a, si H_k est inversible, $H_k^{-1} = R_k^{(2)-1} Q_k^{(2)t}$ et la $(k,1)$-ième entrée de cette matrice est égale à a/ρ_k où a est la $(k,1)$-ième entrée de $Q_k^{(2)t} = F_{k-1,k}^{(2)} F_{k-2,k}^{(2)} \dots F_{1,k}^{(2)}$ c'est-à-dire $|s_1 s_2 \dots s_{k-1}|$ et ρ_k est défini dans (2.29). Par conséquent, la seule entrée non nulle de ζ_k est égale à

$$\beta_1 \frac{h_{k+1,k}}{\rho_k} |s_1 s_2 \dots s_{k-1}| = \beta_1 \frac{|s_k|}{|c_k|} |s_1 s_2 \dots s_{k-1}| = \beta_1 \frac{1}{|c_k|} |s_1 s_2 \dots s_k|.$$

En utilisant la propriété essentielle sur les coefficients des matrices de Givens, c'est-à-dire $|c_k|^2 + |s_k|^2 = 1$, on obtient la relation 1.. Les relations 2., 3. et 4. se déduisent aisément des relations précédentes. ■

On peut dorénavant exprimer les rapports des normes résiduelles en fonction des déterminants des matrices de Hessenberg utilisées dans le processus de Hessenberg généralisé.

Proposition 2.2.5

1. $\dfrac{|r_k^{(1)}|^2_{Z_k}}{|r_{k-1}^{(1)}|^2_{Z_k}} = 1 - |c_k|^2 = 1 - \dfrac{det(H_k)^2}{det(\overline{H}_k^t \overline{H}_k)},$

2. $\dfrac{|r_k^{(1)}|^2_{Z_k}}{|r_k^{(2)}|^2_{Z_k}} = |c_k|^2 = \dfrac{det(H_k)^2}{det(\overline{H}_k^t \overline{H}_k)},$

3. $\dfrac{|r_k^{(2)}|^2_{Z_k}}{|r_{k-1}^{(2)}|^2_{Z_k}} = |s_k|^2 \dfrac{|c_{k-1}|^2}{|c_k|^2} = \dfrac{\left(det(\overline{H}_k^t \overline{H}_k) - det(H_k)^2\right)}{det(H_k)^2} . \dfrac{det(H_{k-1})^2}{det(\overline{H}_{k-1}^t \overline{H}_{k-1})}$

<u>Preuve</u>:
Il suffit pour 1., 2. et 3. d'utiliser les résultats de la proposition précédente ainsi que (2.51). ■

En utilisant la définition des valeurs singulières de H_k et $\overline{H}_k$, il est évident que

$$\frac{|r_k^{(1)}|^2_{Z_k}}{|r_{k-1}^{(1)}|^2_{Z_k}} = 1 - \frac{\prod_{i=1}^{k} \alpha_i^{(k)2}}{\prod_{i=1}^{k} \beta_i^{(k)2}}, \tag{2.55}$$

$$\frac{|r_k^{(1)}|^2_{Z_k}}{|r_k^{(2)}|^2_{Z_k}} = \frac{\prod_{i=1}^{k} \alpha_i^{(k)2}}{\prod_{i=1}^{k} \beta_i^{(k)2}}, \tag{2.56}$$

$$\frac{|r_k^{(2)}|^2_{Z_k}}{|r_{k-1}^{(2)}|^2_{Z_k}} = \frac{\prod_{i=1}^{k-1} \alpha_i^{(k-1)2}}{\prod_{i=1}^{k-1} \beta_i^{(k-1)2}} \left(\frac{\prod_{i=1}^{k} \beta_i^{(k)2}}{\prod_{i=1}^{k} \alpha_i^{(k)2}} - 1 \right). \tag{2.57}$$

Toutes ces relations permettent de comprendre certains phénomènes de convergence particuliers comme la stagnation de la norme du résidu.

2.2.1.b La stagnation de la norme résiduelle

La condition d'orthogonalité imposée dans la méthode RM est la suivante :

$$|r_k|_{Z_k} = \min_{x \in x_0 + \mathcal{K}_k(A, r_0)} |b - Ax|_{Z_k} \tag{2.58}$$

On a donc à l'aide des propriétés de la matrice V_{k+1} :

$$|b - Ax|_{Z_k} = \|V_{k+1}^L (b - Ax)\|_2 = \|\beta_1 e_1^{(k+1)} - \overline{H}_k y\|_2.$$

Dans le cadre des méthodes RM, il y aura stagnation de la norme Z_k du résidu lorsque $|s_k| \simeq 1$ ou, de manière équivalente, $|c_k| \simeq 0$ à l'aide de 2. de la proposition 2.2.4. Or, d'après (2.51), ceci est équivalent au fait que $det(H_k) = 0$. Cela peut également signifier, en précision finie, que $det(\overline{H}_k^t \overline{H}_k)$ est très grand par rapport à $det(H_k)$. On notera, toujours d'après Cullum et Greenbaum [18], que les plateaux observés lors de la représentation graphique de la norme résiduelle (à l'aide d'une échelle logarithmique) en fonction du nombre d'itérations de la méthode RM correspondent aux pics observés par la méthode de Galerkin ou la méthode RO correspondante. Si on utilise l'égalité (2.53), on a l'équivalence

$$det(\overline{H}_k^t \overline{H}_k) \gg det(H_k^2) \Leftrightarrow det(\overline{H}_{k-1}^t \overline{H}_{k-1}) \gg det(H_k^2).$$

On considère ensuite les équations (2.43) et (2.44) et il suit que

$$det(\overline{H}_k^t \overline{H}_k) \gg det(H_k^2) \Leftrightarrow \prod_{i=1}^{k-1} \gamma_i^2 h_{k+1,k}^2 \gg \prod_{i=1}^{k} \gamma_i^2 c_k^2$$

$$\Leftrightarrow h_{k+1,k}^2 \gg \gamma_k^2 c_k^2 = \rho_k^2.$$

Dans le cas où $det(H_k) = 0$, on a la relation

$$det(\overline{H}_k^t \overline{H}_k) = det(\overline{H}_{k-1}^t \overline{H}_{k-1}) h_{k+1,k}^2$$

$$\Leftrightarrow \gamma_k^2 = h_{k+1,k}^2 = \rho_k^2 / c_k^2$$

et cette égalité est en accord avec le fait que $h_{k+1,k}^2 \gg \rho_k^2$. On peut finalement préciser que $det(\overline{H}_k^t \overline{H}_k) \gg det(H_k)^2$ signifie qu'une valeur de Ritz a commencé à converger vers la valeur 0, ce qui n'est pas encore le cas d'une valeur singulière adjacente de Ritz.

2.2.2 Obtention d'une solution à l'aide du résidu initial

Il est clair que la non-singularité de $\overline{H}_k^t \overline{H}_k$ (ou celle de H_k en arithmétique exacte) joue un rôle essentiel dans l'obtention d'une solution de Krylov. Mais peut-on établir un lien avec le résidu initial comme cela a été fait dans le premier chapitre? Pour cela, on introduit le résultat suivant provenant de Ipsen et Meyer [40] :

Théorème 2.2.1
Un système carré linéaire du type $Ax = b$ admet une solution de Krylov si et seulement si $r_0 = b - Ax_0 \in Im(A^{\nu_1})$.

La condition $r_0 \in Im(A^{\nu_1})$ équivaut au fait que pour un certain pas k, la matrice $\overline{H}_k^t \overline{H}_k$ est non-singulière. En effet, on a vu que le problème des moindres carrés (2.31) admettait une solution si et seulement si $\overline{H}_k^t \overline{H}_k$ est inversible. Or cette solution des moindres carrés correspond à une solution de Krylov pour le problème $Ax = b$, l'équivalence est alors démontrée. Par conséquent, on a le

Corollaire 2.2.1
Si à une certaine étape k de la méthode de Hessenberg généralisée $det(\overline{H}_k^t \overline{H}_k) = 0$, ceci est équivalent au fait que $r_0 \notin Im(A^{\nu_1})$.

Ce corollaire est une conséquence directe du théorème (2.2.1) et englobe le théorème (1.3.1) du chapitre précédent en arithmétique exacte, c'est-à-dire, lorsque pour une certaine itération, on a $h_{k+1,k} = 0$ (ce qui n'est généralement pas le cas en précision finie).

2.2.3 La convergence des méthodes RM

2.2.3.a Notations utilisées

Dans le cas non-singulier, A^{-1} peut être exprimé comme un polynôme dépendant de A à l'aide de la définition du polynôme minimal associé à A. Ceci implique que la solution x^* du système $Ax = b$, c'est-à-dire $x^* = A^{-1}b$, appartient à un sous-espace de Krylov dont la dimension n'excède pas le degré de A^{-1} (vu comme un polynôme). On se pose maintenant la même question dans le cas singulier. On doit pouvoir disposer d'un inverse de A ou plus précisément d'un pseudo-inverse de A. On pense (voir Ipsen et Meyer [40]) tout d'abord à l'inverse de Moore-Penrose de A, mais il est inutilisable ici car cet inverse ne s'exprime généralement pas sous la forme d'un polynôme en A. Intervient alors à ce niveau l'inverse de Drazin.
L'inverse de Drazin de A est défini comme étant l'unique matrice A^D satisfaisant les trois propriétés suivantes :

$$A^D A A^D = A^D \tag{2.59}$$

$$AA^D = A^D A \tag{2.60}$$

$$A^{\nu_1+1} A^D = A^{\nu_1} \tag{2.61}$$

où ν_1 est, on le rappelle, la multiplicité de la valeur propre $\lambda_1 = 0$ dans le polynôme minimal de A. Pour $\nu_1 = 1$, l'inverse de Drazin coïncide avec le groupe inverse $A^\#$ qui, en plus des trois propriétés ci-dessus, satisfait $AA^\# A = A$. Cette dernière condition établit des liens importants avec les systèmes linéaires. En effet, toute matrice pseudo-inverse satisfaisant $AXA = A$ est une équation résolvant l'inverse dans le sens où Xb est une solution de $Ax = b$ pour tout $b \in Im(A)$.

$$\begin{aligned} Ax = b \quad &\Leftrightarrow \quad AXAx = b \\ &\Leftrightarrow \quad AXb = b \\ &\Leftrightarrow \quad Ax^* = b, \text{ avec } x^* = Xb. \end{aligned}$$

Remarque 2.2.1
Si A est non-singulière, $\nu_1 = 0$ et l'inverse de Drazin coïncide avec l'inverse ordinaire. Autrement dit, $A^D = A^{-1}$.

Plus généralement, $A^D b$ n'est pas une solution du système consistant si $\nu_1 > 1$ et $b \in Im(A) \backslash Im(A^{\nu_1})$.
On retrouve le même type de résultats lorsqu'on considère le polynôme minimal associé à A pour un vecteur r_0. Dans ce cas, $A^D r_0$ est une solution du système consistant $Ax = b$ si et seulement si $r_0 = b - Ax_0 \in Im(A^{\nu_1})$.
Par l'intermédiaire de la forme canonique de Jordan de A, on peut obtenir une formule explicite pour A^D. Il existe une matrice non-singulière S telle que

$$A = S \begin{bmatrix} R & \vdots & 0 \\ \dots & & \dots \\ 0 & \vdots & N \end{bmatrix} S^{-1} \tag{2.62}$$

où R est une matrice bidiagonale supérieure non-singulière et N est nilpotente d'indice ν_1, c'est-à-dire une matrice telle que $N^{\nu_1} = 0$. Alors

$$A^D = S \begin{bmatrix} R^{-1} & \vdots & 0 \\ \dots & & \dots \\ 0 & \vdots & 0 \end{bmatrix} S^{-1}. \tag{2.63}$$

Freund et Hochbruck [27] introduisent ensuite les notations suivantes : pour des sous-espaces complémentaires arbitraires $X,\ Y \subseteq I\!\!R^n$, on note par $P_{X,Y}$ le projecteur oblique dont l'image est X et le noyau Y. On utilisera comme projecteur

$$P := P_{Ker(A^{\nu_1}),Im(A^{\nu_1})} \tag{2.64}$$

Ce projecteur peut être représenté de la façon suivante :

$$P = S \begin{bmatrix} 0 & \vdots & 0 \\ \dots & & \dots \\ 0 & \vdots & I \end{bmatrix} S^{-1}. \tag{2.65}$$

Les dimensions des blocs sont les mêmes que dans (2.62) et I est la matrice identité. On peut alors vérifier les relations suivantes :

$$AA^D = A^D A = I - P = P_{Im(A^{\nu_1}),Ker(A^{\nu_1})}, \tag{2.66}$$

$$AP = PA, \tag{2.67}$$

$$Im(A^D) = Im(A^{\nu_1}). \tag{2.68}$$

2.2.3.b Convergence de la méthode RM (et de la méthode de Galerkin)

Puisque $\mathcal{K}_k(A, r_0) = \{\phi(A)r_0/\phi \in \Pi_{k-1}\}$, où Π_{k-1} est l'ensemble des polynômes de degré inférieur ou égal à $k-1$, les itérés

$$x_k \in x_0 + \mathcal{K}_k(A, r_0) \tag{2.69}$$

de la méthode RM peuvent s'exprimer en termes de polynômes :

$$x_k = x_0 + \phi_{k-1}(A)r_0 \text{ où } \phi_{k-1} \in \Pi_{k-1}. \tag{2.70}$$

Les vecteurs résiduels r_k correspondants sont donnés par

$$r_k = (I - A\phi_{k-1}(A))r_0 = \Phi_k(A)r_0,$$

où

$$\Phi_k(\lambda) \equiv 1 - \lambda\phi_{k-1}(\lambda). \tag{2.71}$$

On note que $\Phi_k \in \Pi_k$ et $\Phi_k(0) = 1$. Tout polynôme de la sorte est appelé polynôme résiduel de degré k. Les relations (2.70) et (2.71) permettent d'écrire

$$x_k = \Phi_k(A)x_0 + \phi_{k-1}(A)b. \tag{2.72}$$

Dans la suite, on notera $\Phi_k^{(j)}$ la dérivée j-ième de la fonction Φ_k.
Soit k_* l'indice de terminaison de la méthode RM. On pose alors

$$\Phi_k = \Phi_{k_*} \text{ et } \phi_{k-1} = \phi_{k_*-1} \text{ pour tout } k > k_*.$$

Cela garantit que les deux conditions (2.74) et (2.75) du théorème suivant sont toujours bien définies. R.W. Freund et M. Hochbruck ont établi le

Théorème 2.2.2
Soit $A \in I\!R^{n\times n}$ avec $ind(A) = \nu_1$ dont le polynôme minimal s'écrit

$$Q_A(\lambda) = \lambda^{\nu_1} \prod_{i=2}^{p} (\lambda - \lambda_i)^{\nu_i} \ \text{où } \lambda_l \neq \lambda_j \text{ pour } l \neq j \text{ et } \lambda_l \neq 0.$$

De plus, soit $P = P_{Ker(A^{\nu_1}),Im(A^{\nu_1})}$ le projecteur oblique sur $Ker(A^{\nu_1})$ parallèlement à $Im(A^{\nu_1})$. Alors les itérés

$$x_k = \Phi_k(A)x_0 + \phi_{k-1}(A)b$$

convergent vers

$$x = A^D b + P x_0 \quad \forall b,\ x_0 \in I\!R^n \tag{2.73}$$

si et seulement si les conditions

$$\lim_{k\to\infty} \Phi_k^{(j)}(\lambda_l) = 0 \ \text{pour } l = 2, 3, \ldots, p \text{ et } 0 \leq j \leq \nu_l - 1, \tag{2.74}$$

$$\lim_{k\to\infty} \Phi_k^{(j)}(0) = 0 \ \text{pour } 1 \leq j \leq \nu_1 \tag{2.75}$$

sont vérifiées.

Remarque 2.2.2
La condition (2.74) établit simplement que les polynômes résiduels doivent tendre vers 0 sur le spectre de A restreint aux valeurs propres non nulles de la matrice A. La condition (2.75) n'est pas satisfaite par les méthodes de sous-espaces de Krylov standard en général et, par conséquent, dans le cadre des matrices singulières, ces méthodes ne convergent pas vers le vecteur défini par (2.73). Pourtant, le corollaire 2.2.3 qui suit établit que la condition (2.75) n'est pas nécessaire dans le cas particulier des systèmes consistants d'indice 1.

On définit ensuite $e_k \equiv x - x_k$ l'erreur associée au vecteur x_k si x est une solution du système $Ax = b$. Les deux conditions (2.74) et (2.75) sont des conséquences de l'égalité

$$e_k = (I - P)\Phi_k(A)e_0 - P\phi_{k-1}(A)r_0 \tag{2.76}$$

avec $e_0 = A^D b + P x_0 - x_0 \in Im(A^{\nu_1})$.
On remarque que si $r_0 \in Im(A^{\nu_1})$, $P\phi_{k-1}(A)r_0 = \phi_{k-1}(A)Pr_0 = 0$ car $Pr_0 = 0$. Par conséquent $e_k = (I - P)\Phi_k(A)e_0$ et e_k converge vers 0 pour tout x_0. Ainsi $\lim_{k\to\infty}(I - P)\Phi_k(A) = 0$, ce qui correspond à la relation (2.74) et cela est toujours vérifié.

Hochbruck et Freund établissent également deux corollaires du théorème précédent consacrés aux systèmes linéaires singuliers consistants.

Corollaire 2.2.2
Soit $A \in I\!R^{n\times n}$ avec $ind(A) = \nu_1$, $x_0 \in Im(A^r)+Ker(A)$ et $b \in Im(A^s)$ où $0 \leq s, r \leq \nu_1$. Si les conditions (2.74) et

$$\lim_{k\to\infty} \Phi_k^{(j)}(0) = 0 \ pour\ 1 \leq j \leq \nu_1 - min\{s, r+1\} \tag{2.77}$$

sont satisfaites, alors les itérés (2.72) convergent vers x donné par (2.73).

Corollaire 2.2.3
Soit $A \in I\!R^{n\times n}$ avec $ind(A) = 1$ et $b \in Im(A)$. Alors :

1. *si la condition (2.74) est satisfaite, les itérés (2.72) convergent vers*

$$x := A^{\#}b + Px_0 \ où\ P := P_{Ker(A),Im(A)}, \tag{2.78}$$

2. *le vecteur x donné par (2.78) est toujours une solution de $Ax = b$.*
3. *Si les itérés (2.72) convergent et si*

$$z := \lim_{k\to\infty} x_k$$

est une solution de $Ax = b$, alors $z = x$ est la solution donnée par (2.73).

Toutes les preuves de ces résultats sont données dans [27].

2.2.3.c Critères d'arrêt de la méthode RM

On sait que le critère d'arrêt d'une méthode de sous-espaces de Krylov porte sur la précision de la solution obtenue par l'intermédiaire de la norme résiduelle associée à cette dernière. Les résultats établis précédemment permettent d'affirmer, dans le cadre des systèmes singuliers, qu'une solution n'est pas toujours déterminée.
Par conséquent, il est utile de considérer un nouveau critère d'arrêt qui ne porte plus sur la norme du résidu mais qui permet de stopper de manière rigoureuse l'algorithme utilisé. En supposant qu'à l'étape $(k-1)$ la matrice $\overline{H}_{k-1}^t \overline{H}_{k-1}$ est inversible, la relation (2.44) permet d'affirmer que $det(\overline{H}_k^t \overline{H}_k) = 0$ si et seulement si $\gamma_k = r_{k,k} = 0$. La résolution

du système (2.32) n'est plus possible et l'algorithme doit être arrêté. On peut aisément effectuer un test sur γ_k puisque cette valeur, égale à $r_{k,k}$, est calculée explicitement dans l'algorithme. Le critère une fois vérifié permettra d'affirmer, à l'aide du corollaire 2.2.1, que $r_0 \notin Im(A^{\nu_1})$ et ainsi qu'une solution de Krylov ne peut être obtenue par l'intermédiaire du vecteur intial choisi.

On peut également remarquer que le critère d'arrêt porte implicitement sur le scalaire $h_{k+1,k}$ puisque $det(\overline{H}_k^t \overline{H}_k) = det(H_k^2) + det(\overline{H}_{k-1}^t \overline{H}_{k-1}) h_{k+1,k}^2$. Ceci prouve que la condition $h_{k+1,k} = 0$ n'implique pas nécessairement l'existence d'une solution de Krylov.

2.2.4 Exemples numériques

On compare dans ce paragraphe les performances des méthodes RM, à savoir les méthodes QMR, GMRES et CMRH dans le cas singulier. On fait intervenir les versions non redémarrées et non tronquées de ces algorithmes.

Tous les tests effectués dans ce paragraphe ont été réalisés à l'aide de MATLAB sur une machine Sun.
L'algorithme GMRES a été implémenté de la même façon que dans [57] de Saad et Schultz, l'algorithme QMR basé sur la double récurrence (et sans stratégie de "look-ahead") est programmé à partir de [29] de Freund et Nachtigal. Enfin, l'algorithme CMRH est implémenté avec stratégie de pivotage comme dans [58] de Sadok.

Le système à résoudre par toutes ces méthodes s'écrit sous la forme $Ax = b$ où A est une matrice singulière et le second membre b est donné par $b = A \times [1, 1, \ldots, 1]^t$. Le vecteur initial x_0 est généré aléatoirement à l'aide de la fonction *rand* produite par MATLAB. Pour la méthode QMR, le vecteur y est également généré de manière aléatoire. Le critère d'arrêt portant sur la norme résiduelle relative est donné par $\|r_k\|_2 / \|r_0\|_2 \leq 10^{-8}$. Un test d'arrêt porte également sur γ_k et est donné par $|\gamma_k| \leq 10^{-8}$.
Les courbes apparaissant dans la suite donnent la norme résiduelle relative $\|r_k\|_2 / \|r_0\|_2$ dans une échelle logarithmique à chaque itération k. On donne également les tableaux de résultats correspondant à chaque courbe. On y développera le coût en opérations, l'itération d'arrêt, la raison de l'arrêt de l'algorithme ainsi que la précision obtenue sur la norme résiduelle relative. Les exemples sont consacrés à trois types de matrices :

1. les matrices diagonalisables,
2. les matrices non diagonalisables avec $\nu_1 = 1$,
3. les matrices non diagonalisables avec $\nu_1 > 1$.

Les matrices utilisées sont issues, pour la plupart, de la collection Harwell-Boeing [21].

Exemple 1

On considère dans cet exemple la matrice NEUMANN. C'est une matrice singulière extraite du problème discrétisé de Neumann obtenu à l'aide de l'opérateur à 5 points sur un maillage régulier. Elle admet un noyau monodimensionnel, c'est-à-dire que $dimKer(A) = 1$, et une multiplicité ν_1 égale à 1. 0 est donc une valeur propre simple. La matrice $A \in I\!R^{n \times n}$ est diagonalisable et admet comme dimension n, un carré parfait.

Exemple 1.1
On prend $n = 400$ et on obtient, pour les trois méthodes, les graphes suivants:

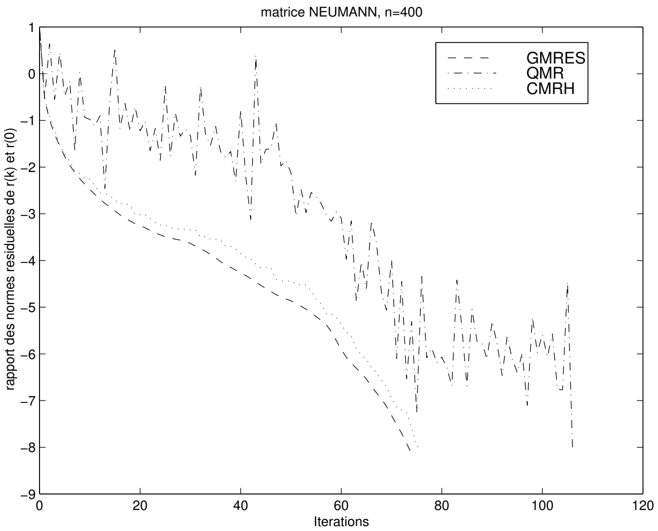


Figure 1.1 : résultats de l'exemple 1.1

	GMRES	QMR	CMRH
flops	7.7×10^6	2.7×10^6	5.5×10^6
itération d'arrêt	74	106	76
cause de l'arrêt	précision atteinte sur le résidu final	précision atteinte sur le résidu final	précision atteinte sur le résidu final
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	9.37×10^{-9}	9.98×10^{-9}	9.96×10^{-9}

Tableau 1.1 : résultats de l'exemple 1.1

On observe que la méthode QMR est plus lente à converger que les méthodes GMRES et CMRH de 30 itérations. La méthode GMRES semble la plus compétitive même si les courbes des deux méthodes sont assez voisines. Cet exemple numérique ne pose aucun problème pour les méthodes de sous-espaces de Krylov utilisées puisque la condition $r_0 \in Im(A^{\nu_1}) = Im(A)$ est vérifiée. On précise également que les solutions obtenues par les trois méthodes sont identiques. Cela sera vérifié dans les expériences suivantes lorsqu'il y a détermination d'une solution de Krylov.

Exemple 1.2

On prend cette fois-ci $n = 900$. On augmente la taille du système précédent pour vérifier que les résultats théoriques sont valides pour de grandes dimensions. Cela permet également d'observer les comportements de la convergence des méthodes pour des vecteurs initiaux différents. On obtient les résultats suivants :

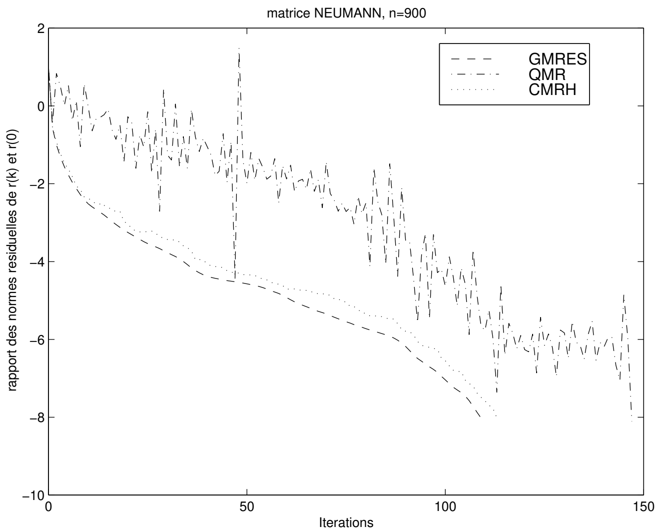


Figure 1.2 : résultats de l'exemple 1.2

	GMRES	QMR	CMRH
flops	3.6×10^7	8.5×10^7	2.6×10^7
itération d'arrêt	109	147	113
cause de l'arrêt	précision atteinte sur le résidu final	précision atteinte sur le résidu final	précision atteinte sur le résidu final
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	9.87×10^{-9}	9.9×10^{-9}	9.16×10^{-9}

Tableau 1.2 : résultats de l'exemple 1.2

Les résultats sont sensiblement les mêmes que dans l'exemple précédent. Les performances sont respectées et le GMRES est encore une fois la méthode la plus appropriée à la résolution de ce type de systèmes.

Exemple 2

On considère maintenant la matrice INVHESS. Elle est égale à l'inverse d'une matrice de Hessenberg supérieure. La multiplicité de la valeur propre nulle dans le polynôme minimal de A vaut ici 2. La matrice $A \in I\!R^{n\times n}$ a, dans les deux exemples suivants, les dimensions respectives $n = 100$ et $n = 300$.

Exemple 2.1
On obtient pour cet exemple les courbes suivantes :

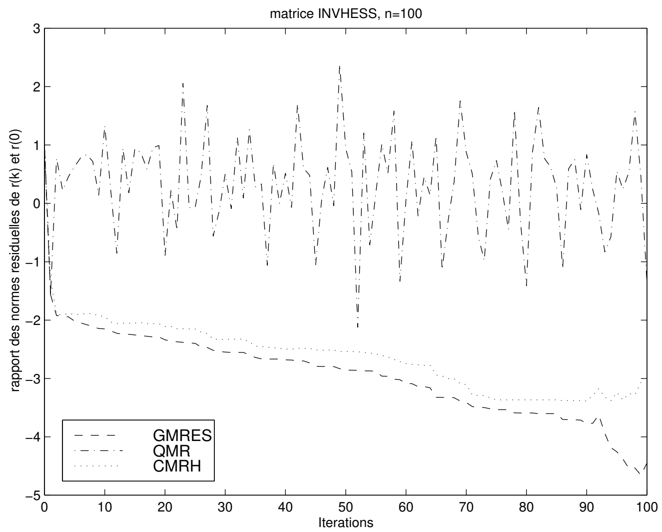


Figure 2.1 : résultats de l'exemple 2.1

	GMRES	QMR	CMRH
flops	7.6×10^6	6.4×10^6	6.2×10^6
itération d'arrêt	100	100	99
cause de l'arrêt	précision atteinte sur γ_k	non convergence	précision atteinte sur γ_k
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	2.01×10^{-5}	4.19×10^{-2}	8.6×10^{-4}

Tableau 2.1 : résultats de l'exemple 2.1

Dans ce cas de figure, les méthodes GMRES et CMRH sont les seules à voir le critère sur γ_k (et implicitement sur $h_{k+1,k}$) vérifié. La non-convergence des trois algorithmes s'explique simplement par le fait que $r_0 \notin Im(A^2)$. Le vecteur x_0 est mal choisi et ne permet pas d'assurer le caractère inversible de la matrice $\overline{H}_k^t\overline{H}_k$, ce qui implique la nullité programmée de γ_k. D'où l'arrêt de l'algorithme.

Exemple 2.2

La matrice est maintenant considérée comme étant de taille 300. Voici les résultats observés :

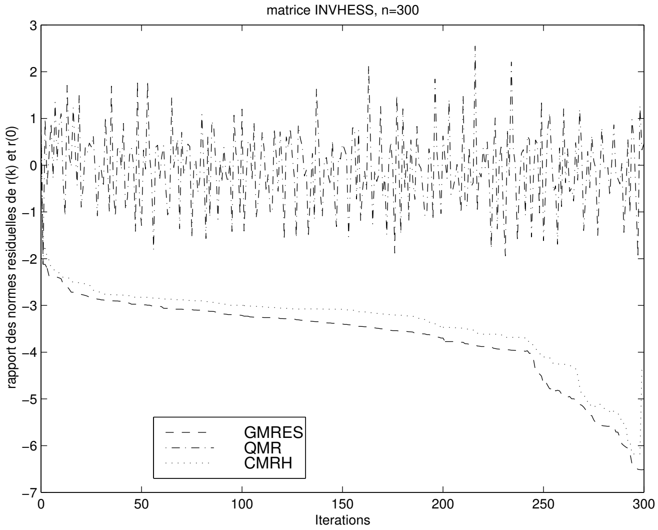


Figure 2.2 : résultats de l'exemple 2.2

	GMRES	QMR	CMRH
flops	2.0×10^8	1.7×10^8	1.6×10^8
itération d'arrêt	300	300	299
cause de l'arrêt	précision atteinte sur γ_k	non convergence	précision atteinte sur γ_k
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	3.04×10^{-7}	8.24×10^0	4.44×10^{-5}

Tableau 2.2 : résultats de l'exemple 2.2

Les conclusions à tirer de ces résultats sont exactement les mêmes que celles de l'exemple précédent. CMRH met une itération de moins que GMRES pour vérifier le critère sur γ_k. On remarque que la norme résiduelle de CMRH passe de 6.62×10^{-7} à la 298-ième itération à 4.44×10^{-5} à la 300-ième. Cette perte de précision soudaine ne peut s'expliquer que par des erreurs numériques.

Exemple 3

On considère maintenant la matrice CLEMENT. $A \in I\!R^{n \times n}$ est tridiagonale avec des zéros sur sa diagonale principale. Les valeurs propres de cette matrice sont connues et sont égales à plus ou moins les nombres $n-1, n-3, n-5, \ldots, 0$. La dimension n doit donc être choisie impaire pour rendre A singulière. ν_1 est une fois de plus égale à 1 et $dimKer(A) = 1$.

Exemple 3.1

On prend dans cet exemple $n = 101$. On obtient les courbes suivantes :

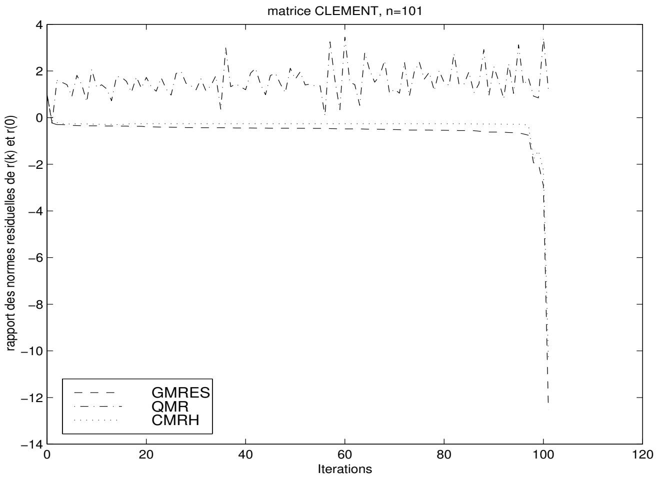


Figure 3.1 : résultats de l'exemple 3.1

	GMRES	QMR	CMRH
flops	7.8×10^6	6.6×10^6	6.4×10^6
itération d'arrêt	101	101	101
cause de l'arrêt	précision atteinte sur le résidu final	non convergence	précision atteinte sur le résidu final
précision obtenue sur $\\|r_k\\|_2/\\|r_0\\|_2$	2.92×10^{-13}	1.58×10^1	4.12×10^{-11}

Tableau 3.1 : résultats de l'exemple 3.1

Les méthodes GMRES et CMRH convergent en un même nombre d'itérations et plus précisément lorsque k atteint la dimension du système. La stagnation précédant la convergence s'explique par le fait que la matrice H_k, construite par les processus d'Arnoldi et de Hessenberg, est singulière. Ainsi, à l'aide du point 1. de la proposition 2.2.5, on peut vérifier que $|r_k|_{Z_k} \simeq |r_{k-1}|_{Z_k}$. La méthode QMR ne converge pas, alors que, théoriquement, elle doit déterminer une solution puisque $r_0 \in Im(A)$. On peut vérifier pour de petites dimensions de n que la convergence a bien lieu pour cette méthode. La non-convergence est uniquement due à la programmation de celle-ci.

Exemple 3.2

La dimension est cette fois égale à $n = 201$. Voici les résultats obtenus :

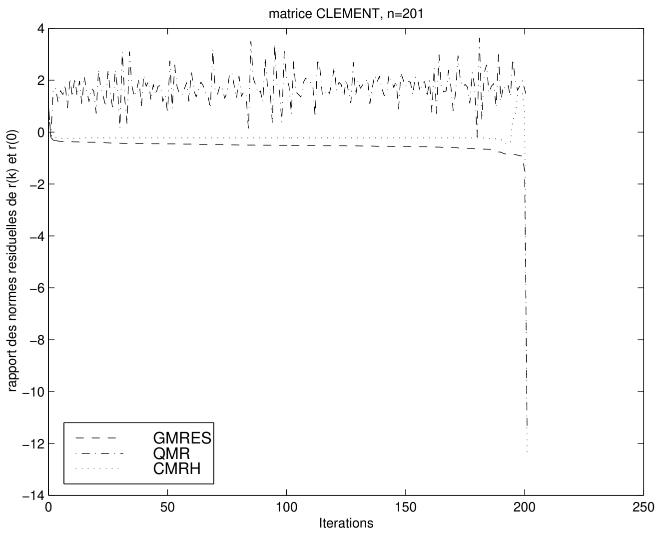


Figure 3.2 : résultats de l'exemple 3.2

	GMRES	QMR	CMRH
flops	6.0×10^7	5.0×10^7	4.9×10^7
itération d'arrêt	201	201	201
cause de l'arrêt	précision atteinte sur le résidu final	non convergence	précision atteinte sur le résidu final
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	4.58×10^{-12}	1.94×10^1	3.94×10^{-13}

Tableau 3.2 : résultats de l'exemple 3.2

Les résultats sont identiques à ceux de l'exemple précédent. Les méthodes les plus efficaces restent les méthodes GMRES et CMRH.

Exemple 4

On utilise dans cet exemple la matrice GEARM. $A \in I\!R^{n \times n}$ est une matrice avec des "1" sur ses diagonales inférieure et supérieure. De plus, si $sgn(i)$ est égal au signe de i pour i entier relatif, alors on a $A(1, abs(i)) = sgn(i)$ et $A(n, n+1-abs(j)) = sgn(j)$ avec $-n \leq i, j \leq n$. Dans le cas présent, pour rendre la matrice A singulière, on prendra $i = n$ et $j = -n$. La multiplicité ν_1 est alors égale à 2.

Exemple 4.1

Dans ce cas de figure, A est de dimension 100 et les résultats apparaissent sur la figure ci-dessous :

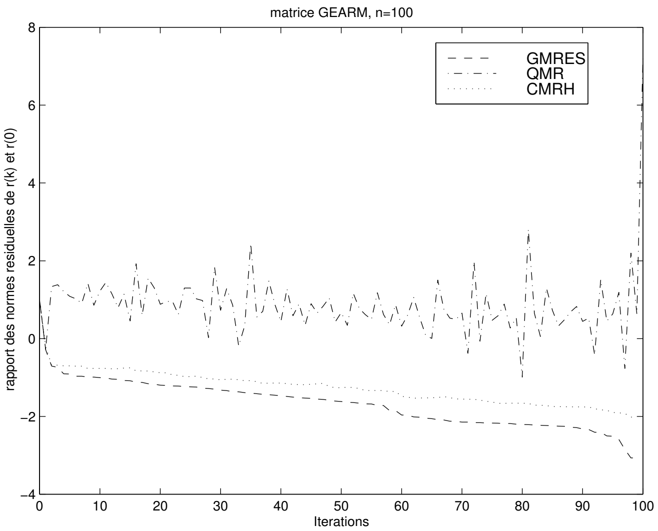


Figure 4.1 : résultats de l'exemple 4.1

	GMRES	QMR	CMRH
flops	7.4×10^6	6.4×10^6	6.1×10^6
itération d'arrêt	99	100	99
cause de l'arrêt	précision atteinte sur γ_k	non convergence	précision atteinte sur γ_k
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	2.2×10^{-3}	2.43×10^{7}	9.2×10^{-3}

Tableau 4.1 : résultats de l'exemple 4.1

Dans le cadre de cet exemple, aucune méthode ne converge. Les conclusions sont sensiblement les mêmes que celles de l'exemple 2. L'itération d'arrêt commune pour GMRES et CMRH est la 99-ième itération. Les deux algorithmes s'arrêtent car la précision est obtenue sur γ_k, ce qui signifie que la solution de (2.32) ne peut pas être déterminée. Comme $\gamma_k \simeq 0$, ceci équivaut à $det(\overline{H}_k^t \overline{H}_k) \ll det(\overline{H}_{k-1}^t \overline{H}_{k-1})$ ou encore à $det(\overline{H}_k^t \overline{H}_k) \simeq 0$. Cela signifie qu'une valeur singulière adjacente de Ritz a convergé vers 0. Le problème des moindres carrés (2.31) ne peut plus alors être résolu pour le vecteur initial choisi.

Exemple 4.2
On prend maintenant $n = 200$ et on obtient les courbes suivantes :

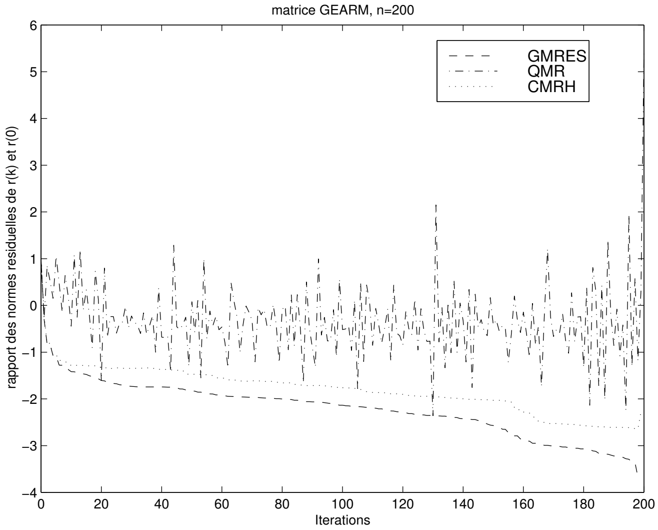


Figure 4.2 : résultats de l'exemple 4.2

	GMRES	QMR	CMRH
flops	5.9×10^7	4.9×10^7	4.8×10^7
itération d'arrêt	199	200	199
cause de l'arrêt	précision atteinte sur γ_k	non convergence	précision atteinte sur γ_k
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	3.20×10^{-4}	1.46×10^7	2.18×10^{-3}

Tableau 4.2 : résultats de l'exemple 4.2

Les résultats sont de même nature que ceux obtenus dans l'exemple précédent. Il y a non-convergence des méthodes CMRH et GMRES qui s'arrêtent malgré tout à l'avant-dernière itération, c'est-à-dire la 199-ième itération, car la précision est obtenue sur γ_k

et sur $h_{k+1,k}$.
Comme la précision est également atteinte sur $r_{k,k}$, le système triangulaire $\overline{R}_k y_k = \overline{g}_k$ ne peut pas être résolu. L'augmentation de la taille du système ne fournit aucun renseignement supplémentaire quant aux comportements de convergence observés.

Exemple 5

Cet exemple permet de vérifier que les algorithmes peuvent être très affectés numériquement par la singularité de A, déformant ainsi les notions concernant la stagnation de la norme résiduelle.

On considère la matrice

$$A = \begin{pmatrix} 0 & 1 & \dots & \dots & \dots & 1 \\ & \ddots & \ddots & & & \vdots \\ & & 0 & \ddots & & \vdots \\ & & & 1 & \ddots & \vdots \\ & 0 & & & \ddots & 1 \\ & & & & & 1 \end{pmatrix} \in I\!R^{n \times n}$$

qu'on appellera matrice STAG où $p = \nu_1$ est égal au nombre d'éléments nuls sur la diagonale principale.
Lorsque $p = 1$, les trois méthodes convergent sans problème particulier.
On prendra dans les deux exemples suivants $p = 5$.

Exemple 5.1
Voici les courbes obtenues par les méthodes GMRES, QMR et CMRH lorsque $n = 300$.

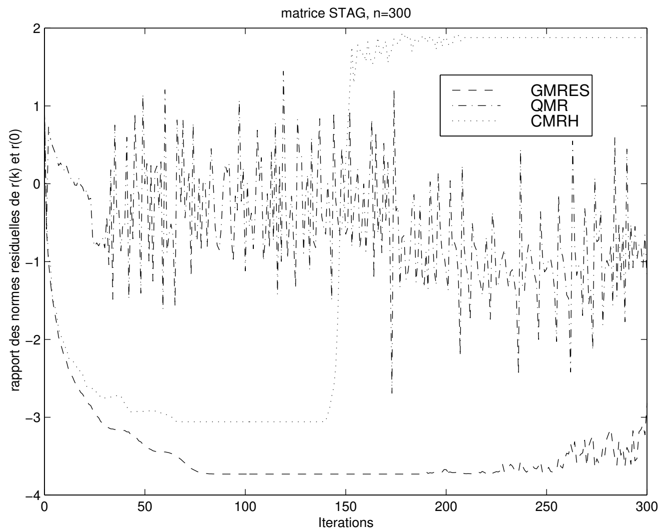


Figure 5.1 : résultats de l'exemple 5.1

	GMRES	QMR	CMRH
flops	2.0×10^8	1.6×10^8	1.6×10^8
itération d'arrêt	300	300	300
cause de l'arrêt	non convergence	non convergence	non convergence
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	1.51×10^{-3}	6.11×10^{-2}	7.54×10^{1}

Tableau 5.1 : résultats de l'exemple 5.1

On remarque immédiatement la non-convergence des trois méthodes qui ne vérifient même pas le critère sur γ_k. Des plateaux de stagnation font leur apparition après la 80-ième itération pour le GMRES et CMRH puis CMRH subit une détérioration de la précision de sa norme résiduelle : elle passe de l'ordre de 10^{-3} à 10^2 en une dizaine d'itérations, ceci de manière encore inexpliquée. La norme résiduelle du GMRES ne commence à subir des fluctuations qu'après la 230-ième itération, celles-ci étant certainement dues à des erreurs numériques. La méthode QMR montre une fois de plus la difficulté qu'elle a à vérifier les comportements prévus.

Exemple 5.2

On prend cette fois-ci comme dimension du système $n = 500$. On obtient les graphiques suivants :

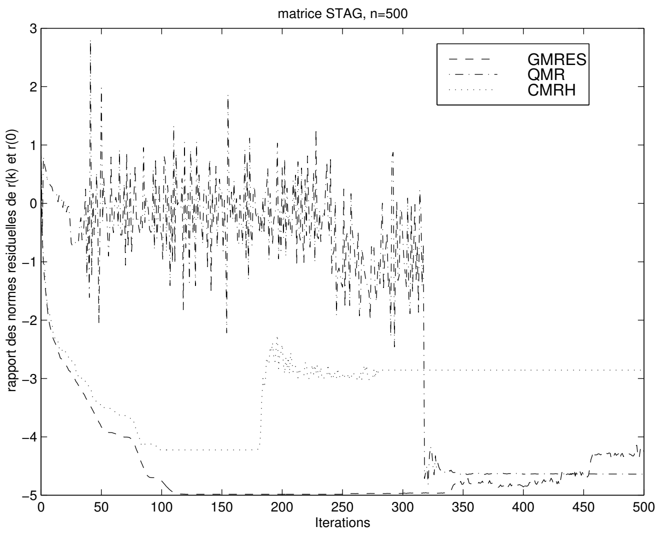


Figure 5.2 : résultats de l'exemple 5.2

	GMRES	QMR	CMRH
flops	9.2×10^8	7.6×10^8	7.5×10^8
itération d'arrêt	500	500	500
cause de l'arrêt	précision atteinte sur γ_k	non convergence	précision atteinte sur γ_k
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	5.52×10^{-5}	2.30×10^{-5}	1.39×10^{-3}

Tableau 5.2 : résultats de l'exemple 5.2

Les résultats semblent être ici un peu plus homogènes que dans l'exemple précédent. Les trois méthodes commencent à stagner après la 330-ième itération et atteignent la 500-ième sans converger. GMRES et CMRH vérifient bien le critère d'arrêt sur γ_k, ce qui concorde avec le fait que $r_0 \notin Im(A^5)$. Cette fois-ci, la méthode QMR stagne et vérifie par conséquent les phénomènes introduits dans la section 2.2.1.*b*. On remarque enfin que CMRH subit le même type de variation brutale que dans l'exemple précédent, ceci après la 180-ième itération.

Exemple 6

Le comportement erratique de la méthode QMR apparaissant dans les exemples précédents a de quoi surprendre. On peut aisément vérifier que cela est dû à l'implémentation de cette méthode. On propose alors, dans ce dernier exemple, d'utiliser l'algorithme QMR extrait de TEMPLATES [4]. Il y est inclus une relation de récurrence relativement peu coûteuse pour calculer le vecteur résidu.

Cet exemple est inspiré de [64]. Il est modifié dans le but de travailler avec une matrice singulière diagonalisable vérifiant $\alpha_1 > 1$. A est une matrice telle que $A = SBS^{-1}$ avec

$$S = \begin{pmatrix} 1 & \beta & & \\ & \ddots & \ddots & \\ & & \ddots & \beta \\ & & & 1 \end{pmatrix} \text{ et } B = \begin{pmatrix} 0 & & & & & \\ & \ddots & & & & \\ & & 0 & & & \\ & & & p+1 & & \\ & & & & \ddots & \\ & & & & & n \end{pmatrix}.$$

On nommera A la matrice SLCGM. On a $p = \alpha_1$ et on prend, dans les deux exemples qui suivent, $\alpha_1 = 10$ et $\beta = 1$.

Exemple 6.1

Dans le cas où $A \in I\!R^{300\times 300}$, on obtient les courbes suivantes :

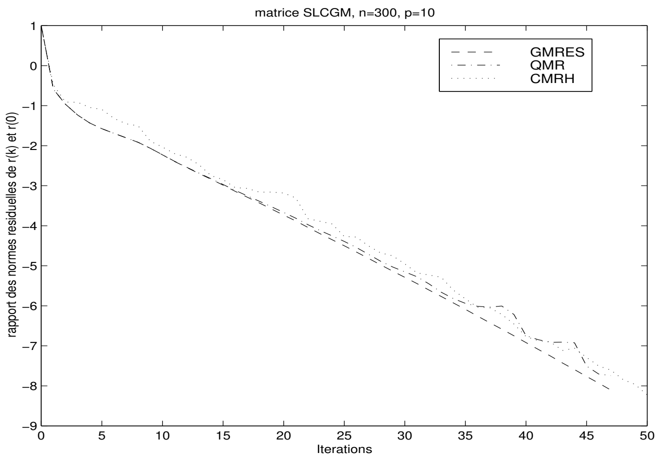


Figure 6.1 : résultats de l'exemple 6.1

	GMRES	QMR	CMRH
flops	1.9×10^7	5.3×10^7	2.5×10^7
itération d'arrêt	47	47	50
cause de l'arrêt	précision atteinte sur le résidu final	précision atteinte sur le résidu final	précision atteinte sur le résidu final
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	7.85×10^{-9}	9.98×10^{-9}	5.88×10^{-9}

Tableau 6.1 : résultats de l'exemple 6.1

On remarque immédiatement que la méthode QMR ne subit plus de fluctuations comme c'était le cas précédemment. On sait que A est diagonalisable. Par conséquent la convergence des trois méthodes est naturelle. Les performances des méthodes sont respectées comme dans le cas inversible, GMRES étant une fois de plus l'algorithme le plus compétitif. Ainsi, peu importe la dimension du noyau de A (égale à 10) lorsque la multiplicité associée à la valeur propre nulle dans le polynôme minimal de A vérifie $\nu_1 = 1$.

Exemple 6.2

On prend cette fois-ci comme dimension du système $n = 1000$ avec $p = 50$. On obtient les graphiques suivants :

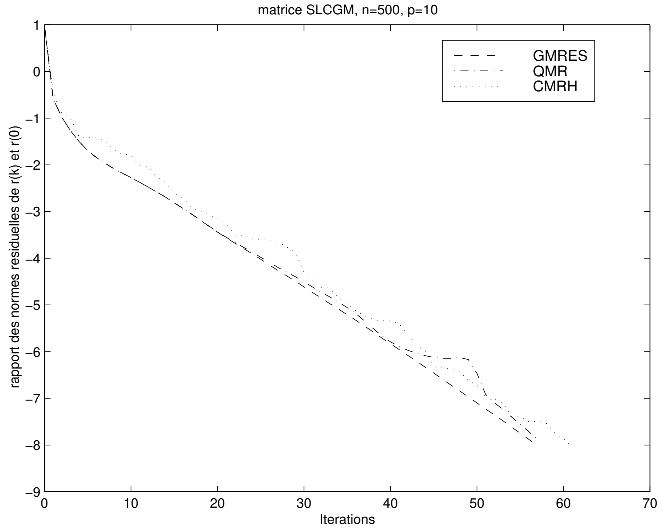


Figure 6.2 : résultats de l'exemple 6.2

	GMRES	QMR	CMRH
flops	6.3×10^7	2.0×10^8	9.3×10^9
itération d'arrêt	57	57	61
cause de l'arrêt	précision atteinte sur le résidu final	précision atteinte sur le résidu final	précision atteinte sur le résidu final
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	9.65×10^{-9}	9.99×10^{-9}	9.63×10^{-9}

Tableau 6.2 : résultats de l'exemple 6.2

Les conclusions sont les mêmes que précédemment. Les courbes de GMRES et CMRH sont très voisines et la méthode QMR nécessite un nombre d'itérations égal à 57 pour converger. L'augmentation de la taille du système ainsi que celle de α_1 ne jouent pas un grand rôle dans la convergence des trois méthodes.

On peut donc réaliser tous ces exemples numériques avec cette version du QMR. Les résultats obtenus sont identiques à ceux provenant de l'ancienne version. On notera néanmoins que le nombre d'itérations réalisé, lorsqu'il y a convergence, est bien moindre que celui observé avec la version précédente.

2.2.5 Conclusion

Après avoir défini les méthodes de sous-espaces de Krylov, puis établi de nouvelles relations portant essentiellement sur H_k et $\overline{H}_k$, on est en mesure de généraliser les résultats du chapitre 1. En effet, si $h_{k+1,k} = 0$ alors $\overline{H}_k^t \overline{H}_k = H_k^2$ et l'inversibilité de H_k est équivalente à celle de la matrice pseudo-inverse de $\overline{H}_k$. Lorsqu'on travaille en précision finie, la condition $h_{k+1,k} = 0$ n'est plus nécessairement vérifiée et les résultats du chapitre 2 voient toute leur utilité. La condition nécessaire et suffisante à l'obtention d'une solution de Krylov reste l'appartenance du résidu initial à $Im(A^{\nu_1})$ où ν_1 est l'indice de A, que ce soit en précision finie ou en arithmétique exacte.
Ce chapitre a également permis de comprendre les phénomènes de stagnation qui peuvent survenir lors de l'utilisation d'une méthode RM dans le cas singulier. La stagnation de la norme du résidu est indiquée par la singularité de H_k et implique une dégénérescence dans la méthode RO correspondante. Ces résultats restent vrais lorsque le système à résoudre est inversible.
Dans le chapitre suivant, on s'intéresse exclusivement à la méthode GMRES et on dégage, à l'aide du chapitre 2, de nouveaux résultats propres à cette méthode ainsi qu'à sa version redémarrée.

Chapitre 3

Le cas particulier de la méthode GMRES

La méthode GMRES reposant sur le processus d'Arnoldi a été définie dans le chapitre 2. De façon abstraite, cette méthode démarre avec une solution approchée x_0 ainsi que son résidu associé $r_0 = b - Ax_0$ et caractérise la k-ième solution approchée de (1.1) comme étant $x_k = x_0 + z_k$ où z_k résout le problème des moindres carrés

$$\min_{z \in \mathcal{K}_k} \|b - A(x_0 + z)\|_2 = \min_{z \in \mathcal{K}_k} \|r_0 - Az\|_2.$$

La méthode GMRES remplace alors ce problème par un nouveau problème des moindres carrés k-dimensionnel sans contraintes. La norme Z_k utilisée lors de la minimisation du résidu est la norme euclidienne. Cela permet de dégager des résultats particuliers au GMRES concernant essentiellement les dégénérescences qui peuvent survenir dans son algorithme. On verra qu'elles interviennent à des moments bien précis et dépendent du sous-espace de Krylov $\mathcal{K}_k$.

Ce chapitre est divisé en quatre sections, les trois premières portant sur les caractéristiques propres au GMRES. Elles concernent le nombre d'itérations nécessaire à la convergence puis les dégénérescences pouvant survenir dans l'algorithme, enfin la version redémarrée de la méthode. La dernière section est consacrée à la comparaison numérique du GMRES ainsi que de GMRES(m), sa version redémarrée toutes les m itérations, ceci dans le cas singulier. On verra qu'un nouveau critère d'arrêt peut être utilisé dans la version redémarrée du GMRES afin de gagner, de manière importante, du temps de calcul ainsi que de la place mémoire.

3.1 Résultats sur le nombre d'itérations

On a vu que

$$\mathcal{K}_k(A, r_0) = vect\{r_0, Ar_0, \ldots, A^{k-1}r_0\} = \mathcal{K}_k.$$

En supposant qu'on travaille en arithmétique exacte, on peut utiliser les notations et résultats du chapitre 1. On a

$$\begin{aligned} r_0 &= u_1 + u_2 + \ldots + u_p, \\ Ar_0 &= Au_1 + Au_2 + \ldots + Au_p, \\ &\vdots \\ A^{k-1}r_0 &= A^{k-1}u_1 + A^{k-1}u_2 + \ldots + A^{k-1}u_p, \ \forall k \geq 1. \end{aligned}$$

Ainsi, $\mathcal{K}_k \subset vect\{u_1, u_2, \ldots, u_p, \ldots, A^{k-1}u_1, A^{k-1}u_2, \ldots, A^{k-1}u_p\}$.

Considérons tout d'abord le cas A diagonalisable alors

$$A^j r_0 = A^j u_2 + A^j u_3 + \ldots + A^j u_p = \lambda_2^j u_2 + \lambda_3^j u_3 + \ldots + \lambda_p^j u_p \ , \forall j \in \{1, 2, \ldots, k-1\}.$$

Par conséquent $\mathcal{K}_k \subset vect\{u_1, u_2, \ldots, u_p\}$. Ceci implique l'inégalité et l'équivalence suivantes :

$$\begin{aligned} dim(\mathcal{K}_k) &\leq dim\ vect\{u_2, u_3, \ldots, u_p\} \\ &\Leftrightarrow k \leq p - 1. \end{aligned} \tag{3.1}$$

Supposons maintenant que A soit non diagonalisable et $\nu_1 = 1$. Alors, toujours en accord avec la proposition (1.3.2), on a l'inégalité

$$dim(\mathcal{K}_k) \leq dim\ vect\{u_2, u_3, \ldots, u_p, \ldots, A^{k-1}u_2, A^{k-1}u_3, \ldots, A^{k-1}u_p\} \text{ et}$$

ceci équivaut au fait que $k \leq (p-1)k \Leftrightarrow p \geq 2$, relation que l'on a déja posée au début du chapitre 2.

Enfin si A est non diagonalisable et $\nu_1 > 1$, on ne peut pas obtenir de relation sur le paramètre p.

On ne peut donc discuter du nombre d'itérations nécessaire à la convergence du GMRES que lorsque A est diagonalisable. En effet, en supposant que l'on connaisse parfaitement la décomposition du résidu initial r_0 sur les sous-espaces caractéristiques, l'inégalité (3.1) peut se préciser.
Dans le cas où $r_0 = u_2 + u_3 + \ldots + u_p$, il y a équivalence avec le fait que $\mu_i = 1, \ \forall i \in \{2, 3, \ldots, p\}$. Ainsi,

$$k = \mu_2 + \mu_3 + \ldots + \mu_p = p - 1.$$

Dans le cas où $l-1$ vecteurs dans la décomposition de r_0 ne sont pas définis avec $2 \leq l \leq p-1$ et $u_{i_j} \in \{u_2, u_3, \ldots, u_p\}$, $\forall j \in \{1, 2, \ldots, p-1\}$ on a l'égalité suivante :

$$r_0 = u_{i_l} + u_{i_{l+1}} + \ldots + u_{i_{p-1}}.$$

En utilisant le résultat de la proposition 1.3.4 ainsi que l'unicité du polynôme minimal de A pour r_0 on a :

$$\Pi_A(A)r_0 = (A - \lambda_{i_l} I)^{\mu_{i_l}} (A - \lambda_{i_{l+1}} I)^{\mu_{i_{l+1}}} \ldots (A - \lambda_{i_{p-1}} I)^{\mu_{i_{p-1}}} r_0 = 0.$$

Ceci entraîne que $\mu_{i_l} + \mu_{i_{l+1}} + \ldots + \mu_{i_{p-1}} = k$. Mais $\mu_{i_j} = 1$, $\forall j \in \{l, l+1, \ldots, p-1\}$ ainsi $\mu_{i_l} + \mu_{i_{l+1}} + \ldots + \mu_{i_{p-1}} = p - l = k$.
Finalement, quand A est diagonalisable, $0 < p - l \leq k \leq p-1$. On obtient le résultat suivant :

Proposition 3.1.1
Soit $A \in I\!R^{n \times n}$ une matrice singulière telle que $Ax = b$ avec $b \in Im(A)$. On suppose que A est diagonalisable et $x_0 \in I\!R^n$ est le vecteur initial. La méthode GMRES appliquée à (1.1) donne alors une solution des moindres carrés en $k = p-1$ itérations si $r_0 = u_2 + u_3 + \ldots + u_p$ et $0 < k = p - l$ itérations dans le cas où l vecteurs ne sont pas définis dans la décomposition de r_0.

3.2 Résultats sur les dégénérescences du GMRES

On introduit dans cette partie certains résultats de P.N.Brown et H.F.Walker [13]. On rappelle que trivialement

$$dim A(\mathcal{K}_k) \leq dim(\mathcal{K}_k) \leq k, \ \forall k \tag{3.2}$$

où $A(\mathcal{K}_k) = vect\{Ar_0, A^2 r_0, \ldots, A^k r_0\}$. Cette propriété implique le résultat suivant :

Théorème 3.2.1
On applique le GMRES à la résolution du système (1.1). Ainsi, s'il existe une itération telle que $dim(\mathcal{K}_k) = k$, soit

1. *le GMRES dégénère par déficience du problème des moindres carrés (2.13) sans déterminer de solution, c'est-à-dire $dim A(\mathcal{K}_k) < k$, soit*
2. *le GMRES détermine une solution sans dégénérescence et dégénère au pas suivant par dégénérescence du sous-espace de Krylov, c'est-à-dire $dim A(\mathcal{K}_k) = k$ et $dim(\mathcal{K}_{k+1}) < k+1$.*

On cherche dorénavant à savoir si on ne peut pas associer, de manière plus précise, une dégénérescence particulière avec les cas étudiés dans les chapitres précédents. Introduisons tout d'abord les deux types de solutions que le GMRES peut déterminer :

Définition 3.2.1
Une solution des moindres carrés est une solution $x^* \in I\!R^n$ pour laquelle $\|b - Ax^*\|_2$ est minimale.

Définition 3.2.2
La solution pseudo-inverse de $Ax = b$ est la solution $x^* \in I\!R^n$ des moindres carrés pour laquelle la norme $\|x^*\|_2$ est minimale.

On dispose du résultat suivant, toujours issu de [13] :

Proposition 3.2.1
Une solution (des moindres carrés) de $Ax = b$ est la solution pseudo-inverse si et seulement si elle appartient à $Ker(A)^\perp$.

Le théorème précédent s'applique également dans le cas non-singulier. Certaines remarques se dégagent alors :

1. si le système (1.1) n'est pas consistant, la dégénérescence par déficience du problème des moindres carrés peut éventuellement survenir.
2. Inversement, la dégénérescence du sous-espace de Krylov intervient si et seulement si le système (1.1) est consistant et la solution a été déterminée.
3. Si A est non-singulière alors le GMRES ne peut pas dégénérer par déficience du rang du problème des moindres carrés (2.31).

On a également le résultat ci-dessous :

Proposition 3.2.2
A la k-ième itération, le GMRES détermine une solution des moindres carrés de (1.1) sans dégénérer si et seulement si

$$dim A^t(\mathcal{K}_{k+1}) = dim A(\mathcal{K}_k) = k. \qquad (3.3)$$

Preuve:
D'après la seconde assertion du théorème (3.2.1), $dim A(\mathcal{K}_k) = k$ puisque le GMRES détermine une solution de Krylov sans dégénérescence . Il reste donc à montrer que x_k est une solution des moindres carrés de (1.1) si et seulement si $dim A^t(\mathcal{K}_{k+1}) = dim A(\mathcal{K}_k)$. On rappelle que

$$\min_{x \in x_0 + \mathcal{K}_k} \|b - Ax\|_2 = \min_{y \in I\!R^k} \|\beta_1 e_1^{(k+1)} - \overline{H}_k y\|_2 \tag{3.4}$$

donc, par définition, x_k est une solution des moindres carrés de (1.1) si et seulement si il existe $z \in \mathcal{K}_k$ tel que

$$A^t(b - A(x_0 + z)) = A^t(r_0 - Az) = 0. \tag{3.5}$$

La relation (3.5) est vérifiée si et seulement si $A^t r_0 \in A^t A(\mathcal{K}_k)$ ce qui équivaut à

$$A^t(\mathcal{K}_{k+1}) = A^t A(\mathcal{K}_k). \tag{3.6}$$

On sait que $dim A^t A(\mathcal{K}_k) \leq dim A(\mathcal{K}_k)$. Supposons que $dim A^t A(\mathcal{K}_k) < dim A(\mathcal{K}_k)$ alors $\exists\ w \in \mathcal{K}_k$ tel que $Aw \neq 0$ et $A^t A w = 0$. Mais on a dans ce cas

$$0 = w^t A^t A w = \|Aw\|_2^2$$

ce qui établit une contradiction. Donc $dim A^t(\mathcal{K}_{k+1}) = dim A(\mathcal{K}_k)$. Cela conclut la preuve. ■

Brown et Walker introduisent également une condition particulière sur A qui permet d'assurer au GMRES la détermination d'une solution de $Ax = b$ quand A est quelconque :

Théorème 3.2.2
Si $Ker(A^t) = Ker(A)$ et si une solution des moindres carrés est obtenue à un certain pas, alors le GMRES dégénère au pas suivant par dégénérescence du sous-espace de Krylov si (1.1) est consistant, et par déficience du problème des moindres carrés (2.31) sinon. De plus, si (1.1) est consistant et $x_0 \in Im(A)$, alors la solution obtenue est la solution pseudo-inverse.

Preuve:
On suppose que $Ker(A) = Ker(A^t)$. Alors, pour tout k, on a $dim A^t(\mathcal{K}_{k+1}) = dim A(\mathcal{K}_{k+1})$ et la relation (3.3) s'écrit :

$$dim A(\mathcal{K}_{k+1}) = dim A(\mathcal{K}_k) = k.$$

Cette relation est vraie pour $0 \leq k \leq n$ et il s'ensuit d'après la proposition (3.2.2) que le GMRES détermine une solution des moindres carrés x_k sans dégénérer à la k-ième itération. De plus, puisque $dim A(\mathcal{K}_{k+1}) = k$, le GMRES dégénère à la $(k+1)$-ième itération.

On conclut ainsi, à l'aide du théorème (3.2.1), que la dégénérescence qui survient est celle due à la dégénérescence du sous-espace de Krylov si le système (1.1) est consistant et celle par déficience du problème des moindres carrés (2.31) sinon. Si (1.1) est consistant alors x_k est une solution et, de plus, $\mathcal{K}_k \subseteq Im(A)$. Si on a également $x_0 \in Im(A)$, alors $x_k = x_0 + z_k \in x_0 + K_k \subseteq Im(A) = Ker(A)^\perp$. Puisqu'une solution (des moindres carrés) de (1.1) est la solution pseudo-inverse si et seulement si elle appartient à $Ker(A)^\perp$, il suit que x_k est la solution pseudo-inverse. ∎

On fait remarquer que les deux preuves précédentes sont issues de [13].

Notre objectif est maintenant d'associer chaque type de dégénérescence à un cas de figure bien particulier et plus précisément à la singularité de $\overline{H}_k^t \overline{H}_k$. On sait déja que le GMRES déterminera une solution des moindres carrés de (2.31) si et seulement si $det(\overline{H}_k^t \overline{H}_k) \neq 0$. Dans ce cas la solution est donnée par

$$x_k = x_0 + V_k(\overline{H}_k^t \overline{H}_k)^{-1} \overline{H}_k^t \beta_1 e_1^{(k+1)}.$$

Prouvons à présent le résultat suivant :

Proposition 3.2.3
On applique le GMRES à (1.1) avec A singulière et $b \in Im(A)$. Alors, à la k-ième itération, le GMRES détermine une solution des moindres carrés sans dégénérer et dégénère à l'itération suivante par dégénérescence du sous-espace de Krylov si et seulement si la matrice $\overline{H}_k^t \overline{H}_k$ est non-singulière.

Preuve:
On est amené à prouver à l'aide de (3.2) et (3.3) que les conditions $dim A(\mathcal{K}_l) = dim(\mathcal{K}_l) = l$, $\forall l \leq k$ et $dim(\mathcal{K}_{k+1}) < k+1$ ainsi que $dim A^t(\mathcal{K}_{k+1}) = dim A^t A(\mathcal{K}_k) = dim A(\mathcal{K}_k)$ sont équivalentes à la non-singularité de $\overline{H}_k^t \overline{H}_k$. Supposons tout d'abord qu'il existe un vecteur non nul $y \in I\!R^k$ tel que $\overline{H}_k^t \overline{H}_k y = 0$. Mais en utilisant (1.17) ainsi que l'orthonormalité de V_{k+1} on a

$$\overline{H}_k^t \overline{H}_k = \overline{H}_k^t V_{k+1}^t V_{k+1} \overline{H}_k = V_k^t A^t A V_k.$$

On sait que $z = V_k y \neq 0$ appartient à $\mathcal{K}_k$. Donc comme $dim A(\mathcal{K}_k) = k$, $AV_k y \neq 0$. Par hypothèse $dim A^t A(\mathcal{K}_k) = dim A(\mathcal{K}_k) = k$ et par conséquent $A^t A V_k y \neq 0$. Ensuite $\overline{H}_k^t \overline{H}_k y = 0$ entraîne le fait que $A^t A V_k y \in \mathcal{K}_{k+1}$. Mais, en considérant l'inégalité $dim(\mathcal{K}_{k+1}) < k+1$, il y a contradiction. Finalement, la seule possibilité est $y = 0$, ce qui implique que $\overline{H}_k^t \overline{H}_k$ est non-singulière.
La seconde implication est plus simple à démontrer : supposons que $\overline{H}_k^t \overline{H}_k$ soit non-singulière alors $x_k = x_0 + V_k(\overline{H}_k^t \overline{H}_k)^{-1} \overline{H}_k^t \beta_1 e_1^{(k+1)}$ est définie. Cela signifie que le GMRES

détermine une solution des moindres carrés de (1.1) sans dégénérer. De plus, l'égalité $dim A^t(\mathcal{K}_{k+1}) = dim A(\mathcal{K}_k) = k$ est vérifiée et (1.1) est consistant. Cela implique que $dim(\mathcal{K}_{k+1}) = k$, ce qui conclut la preuve. ■

On en déduit la proposition suivante

Proposition 3.2.4
On applique le GMRES à (1.1) avec A singulière et $b \in Im(A)$. Alors, à la k-ième itération, le GMRES dégénère par déficience du rang du problème des moindres carrés (2.31) sans déterminer de solution si et seulement si la matrice $\overline{H}_k^t \overline{H}_k$ est singulière.

On rappelle que le système (2.32) peut être résolu si et seulement si $det(\overline{H}_k^t \overline{H}_k) \neq 0$. Si ça n'est pas le cas, on obtient, grâce à la relation (2.50), $det(H_k) = 0$ et $h_{k+1,k} \neq 0$. Par conséquent il est facile de vérifier que $\gamma_k = r_{k,k} = 0$. Dans ce cas, il est clair que le GMRES ne peut pas déterminer de solution des moindres carrés de (1.1).

3.3 Résultats sur le GMRES redémarré

On sait que lorsque le nombre d'itérations k dans le processus d'Arnoldi augmente, le nombre de vecteurs nécessaire au processus augmente proportionnellement à k. Ainsi la méthode GMRES est utilisée de manière itérative et est redémarrée toutes les m itérations où m est un paramètre entier fixé. On appellera cycle un ensemble de m itérations. Voici l'algorithme de cette méthode :

Algorithme 3.1 : $[x_m, r_m, V_{m+1}, \overline{H}_m] = \text{GMRES}(A, x_0, m)$

1. initialisation :
   calculer $r_0 = b - Ax_0$,
2. construction de la base : $[V_{m+1}, \overline{H}_m] =$ Arnoldi (A, r_0, m),
3. calcul de la solution approchée :
   (a) calculer y_m solution de $\min_{y \in I\!R^m} \|\beta_1 e_1^{(m+1)} - \overline{H}_m y\|_2$,
   (b) calculer $x_m = x_0 + V_m y_m$,
   (c) calculer $r_m = r_0 - AV_m y_m$,

4. redémarrage :
   si ($\|r_m\|_2/\beta_1 < tol$) ou si ($iter = maxit$) alors stop,
   sinon poser $x_0 := x_m$ et $r_0 := r_m$ et aller à 2.

Le GMRES redémarré consiste tout d'abord à calculer la solution x_m puis à redémarrer l'algorithme avec $\tilde{x}_0 = x_m$ comme vecteur initial.
On s'intéresse ici au GMRES redémarré appliqué à des systèmes singuliers avec $\nu_1 > 1$ où ν_1 est l'indice de A. On se pose alors la question suivante :
"est-ce que la condition $r_0 \notin Im(A^{\nu_1})$ se transmet lors du redémarrage du GMRES ?"

On suppose tout d'abord que $r_k \equiv P_k(A)r_0$ avec $P_k \in \mathcal{P}_k$ où $\mathcal{P}_k$ est l'espace des polynômes de degré au plus k et $r_0 \notin Im(A^{\nu_1})$. On pose $P_k(t) \equiv \sum_{i=0}^{k} \alpha_i t^i$. Si $\alpha_0 = 0$ alors $r_k \in A(\mathcal{K}_k)$ mais par construction $r_k \perp A(\mathcal{K}_k)$ donc l'égalité $P_k(0) = 0$ est irréalisable sauf si $r_k = 0$. Ce cas implique que k a atteint le degré du polynôme minimal de A pour r_0. On supposera par la suite que $0 \le k \le m$ avec m inférieur strictement au degré de ce polynôme. Donc si $r_0 \notin Im(A^{\nu_1})$, $r_m \equiv \tilde{r}_0 \notin Im(A^{\nu_1})$ où $\tilde{r}_0$ est le nouveau résidu initial associé à $\tilde{x}_0$. Par conséquent, la condition $r_0 \notin Im(A^{\nu_1})$ se transmet lors des redémarrages. Grâce aux théorèmes (2.2.1) et (3.2.1), on peut facilement vérifier que la condition $r_0 \notin Im(A^{\nu_1})$ est équivalente au fait qu'à une certaine itération, $dimA(\mathcal{K}_k) < k$. Soit $\widetilde{\mathcal{K}}_k \equiv \mathcal{K}_k(A, \tilde{r}_0) \equiv vect\{\tilde{r}_0, A\tilde{r}_0, \ldots, A^{k-1}\tilde{r}_0\}$, le sous-espace de dimension k associé à A et $\tilde{r}_0$. Ainsi, lors du redémarrage du GMRES, à un certain pas, on obtient $dimA(\widetilde{\mathcal{K}}_k) < k$. On pose maintenant

$$\widetilde{H}_k = \{\tilde{h}_{i,j}\}_{\substack{1 \le i \le k \\ 1 \le j \le k}} \in I\!R^{k\times k} \text{ et } \overline{\widetilde{H}}_k = \begin{pmatrix} \widetilde{H}_k \\ 0 \ldots 0 \tilde{h}_{k+1,k} \end{pmatrix} \in I\!R^{(k+1)\times k}$$

comme étant les matrices de Hessenberg obtenues par le processus d'Arnoldi après chaque nouveau redémarrage. On a vu à l'aide de (2.52) que

$$det\left(\overline{\widetilde{H}}_k^t \overline{\widetilde{H}}_k\right) = 0 \Leftrightarrow \begin{cases} det\left(\widetilde{H}_k^t \widetilde{H}_k\right) = 0, \\ \tilde{h}_{k+1,k}^2 = 0. \end{cases}$$

Dans ce cas, aucune solution ne peut être déterminée. En arithmétique exacte, si on a $\tilde{h}_{k+1,k} \neq 0$, alors

$$det\left(\overline{\widetilde{H}}_k^t \overline{\widetilde{H}}_k\right) \neq 0.$$

Par conséquent si

$$\tilde{\gamma}_k^2 \equiv \frac{det\left(\overline{\widetilde{H}}_k^t \overline{\widetilde{H}}_k\right)}{det\left(\overline{\widetilde{H}}_{k-1}^t \overline{\widetilde{H}}_{k-1}\right)},$$

on obtient $\tilde{\gamma}_k \neq 0$ et ainsi, comme cela a été détaillé dans le second chapitre, GMRES(m) ne peut jamais dégénérer par déficience du rang du problème des moindres carrés (2.31). En précision finie, quand une valeur singulière de Ritz et une valeur singulière adjacente de Ritz convergent vers 0, $det\left(\widetilde{H}_k^t\widetilde{H}_k\right)$ et $det\left(\overline{\overline{H}}_{k-1}^t\overline{\overline{H}}_{k-1}\right)$ ont tendance à devenir très petits. Ceci implique que $det\left(\overline{\overline{H}}_k^t\overline{\overline{H}}_k\right) \simeq 0$ grâce à (2.53), même si $\tilde{h}_{k+1,k} \neq 0$. Ainsi l'inégalité $dimA(\widetilde{\mathcal{K}}_k) < k$ est pratiquement vérifiée et, puisque $A(\widetilde{\mathcal{K}}_k) \subset \widetilde{\mathcal{K}}_k$, on obtient $r_k \in \mathcal{P}_{k-1}$. Ceci entraîne que $r_k \in \widetilde{\mathcal{K}}_k$ et ainsi $Ar_k \in A(\widetilde{\mathcal{K}}_k)$. Or, par construction, r_k est orthogonal à $A(\widetilde{\mathcal{K}}_k)$ donc $r_k^tAr_k \simeq 0$. On suppose maintenant que l'on redémarre le GMRES, alors $\tilde{h}_{1,1} \equiv \tilde{r}_0^tA\tilde{r}_0 = r_k^tAr_k$. Par conséquent, lors du redémarrage, $\widetilde{H}_1$ est presque singulière.

On peut se demander à présent comment se transmet la singularité de la matrice de Hessenberg lorsque le processus continue.
On considère pour cela

$$\tilde{h}_{1,j} \equiv \tilde{v}_1^tA\tilde{v}_j = \frac{1}{\|\tilde{r}_0\|_2}\tilde{r}_0^tA\tilde{v}_j,\ \forall j \in \{1,2,\ldots,m\}. \tag{3.7}$$

On sait que $\tilde{v}_j \in \mathcal{K}_j$ donc $\tilde{v}_j = \sum_{i=0}^{j-1}\alpha_iA^i\tilde{r}_0$ et (3.7) peut s'écrire

$$\tilde{h}_{1,j} = \frac{1}{\|\tilde{r}_0\|_2}\tilde{r}_0^tA\sum_{i=0}^{j-1}\alpha_iA^i\tilde{r}_0 = \frac{1}{\|\tilde{r}_0\|_2}\sum_{i=0}^{j-1}\alpha_i(\tilde{r}_0^tA^{i+1}\tilde{r}_0). \tag{3.8}$$

Grâce à l'inégalité $dimA(\widetilde{\mathcal{K}}_k) < k$ on a $dim(\widetilde{\mathcal{K}}_{k+1}) < k$. On obtient alors $A^{i+1}(\widetilde{\mathcal{K}}_k) = A(\widetilde{\mathcal{K}}_k)$ pour $i \in \{0,1,\ldots,j-1\}$. Puisque $r_k \perp A(\widetilde{\mathcal{K}}_k)$, on a $r_k \perp A^{i+1}(\widetilde{\mathcal{K}}_k)$ et ainsi $r_k^tA^{i+1}r_k \simeq 0$. Il est donc évident que $\tilde{h}_{1,j} \simeq 0,\ \forall j \in \{1,2,\ldots,m\}$ à l'aide de (3.8). Par conséquent, $\widetilde{H}_j$ est presque singulière $\forall j \in \{1,2,\ldots,m\}$. Cela signifie que la singularité de la matrice de Hessenberg se transmet lors des redémarrages.

Il suffit par conséquent, à une certaine étape de l'algorithme, qu'une valeur singulière de Ritz approche une valeur propre nulle pour rendre définitive la stagnation de la norme résiduelle dans le GMRES redémarré. Cela signifie également que si la norme résiduelle de la version non redémarrée du GMRES stagne jusqu'à une certaine itération k et que le nombre d'itérations par cycle de la version redémarrée est inférieur à k, alors la stagnation du GMRES redémarré est inéluctable.

Sachant qu'aucune amélioration concernant la solution n'est possible lorsque $r_0 \notin Im(A^{\nu_1})$ lors des redémarrages, on ajoute un nouveau critère d'arrêt sur $\tilde{h}_{1,1}$: si la précision sur $\tilde{h}_{1,1}$ est acceptée, on stoppe le GMRES.

3.4 Applications numériques

Cette partie est consacrée uniquement à l'application de la méthode GMRES redémarrée à la résolution de systèmes singuliers. On réutilisera les matrices introduites dans les exemples numériques du chapitre précédent. La notation GMRES(m) signifiera que la méthode GMRES est redémarrée après chaque cycle de m itérations. Les tests numériques sont toujours effectués à l'aide de MATLAB sur une machine Sun dont la précision machine est égale à 2.22×10^{-16}.

De la même manière que dans le dernier chapitre, on tracera la norme résiduelle relative en fonction du nombre d'itérations sur un graphique, suivi d'un tableau récapitulatif et d'une discussion. Le nombre d'itérations maximal pour GMRES(m) peut varier en fonction des exemples mais il est en général fixé à 1000. Le vecteur initial x_0 est généré aléatoirement à l'aide de la fonction *rand*.

Exemple 1

On utilise la matrice NEUMANN. On a vu que cette matrice ne posait aucun problème pour les méthodes non redémarrées. On utilise tout d'abord $n = 400$ puis, dans le second exemple, $n = 900$.

Exemple 1.1

On superpose sur le graphique, les courbes de GMRES(50), GMRES(30) et GMRES(20) :

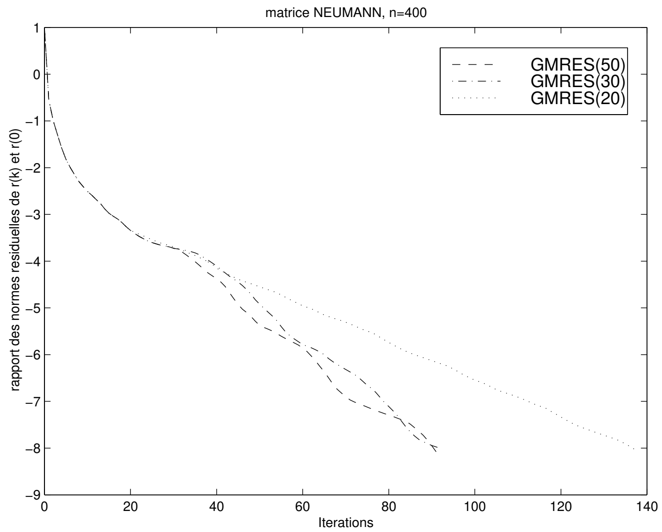


Figure 1.1 : résultats de l'exemple 1.1

	GMRES(50)	GMRES(30)	GMRES(20)
flops	1.9×10^7	8.0×10^6	6.7×10^6
temps CPU	2.44	1.88	2.03
itération d'arrêt	91	92	137
cause de l'arrêt	précision atteinte sur le résidu final	précision atteinte sur le résidu final	précision atteinte sur le résidu final
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	8.33×10^{-9}	9.79×10^{-9}	9.54×10^{-9}

Tableau 1.1 : résultats de l'exemple 1.1

On observe qu'il y a convergence pour les trois méthodes. On peut préciser également qu'elles déterminent la même solution et cela sera le cas pour tous les exemples introduits par la suite. Les algorithmes s'arrêtent car la précision est atteinte sur le résidu final. Le GMRES redémarré se comporte dans notre cas exactement de la même façon que lorsque le système à résoudre est inversible. On rappelle que A est ici diagonalisable et donc que le spectre de $\overline{H}_k^t \overline{H}_k$ ne peut contenir la valeur propre nulle. GMRES(50) est l'algorithme le plus coûteux aussi bien en temps CPU qu'en nombre d'opérations. Il est également celui qui effectue le moins d'itérations.

Exemple 1.2
On augmente la taille du système, n passe à 900 et les résultats suivants sont obtenus :

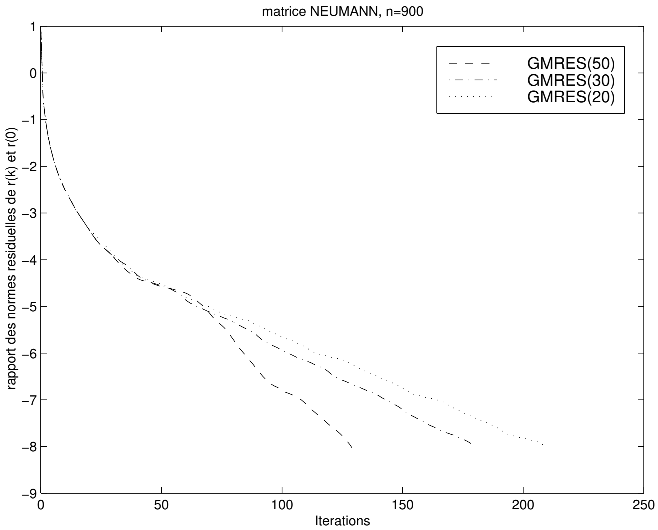


Figure 1.2 : résultats de l'exemple 1.2

	GMRES(50)	GMRES(30)	GMRES(20)
flops	3.8×10^7	2.7×10^7	2.0×10^7
temps CPU	7.38	8.10	8.46
itération d'arrêt	129	180	210
cause de l'arrêt	précision atteinte sur le résidu final	précision atteinte sur le résidu final	précision atteinte sur le résidu final
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	9.38×10^{-9}	9.56×10^{-9}	9.65×10^{-9}

Tableau 1.2 : résultats de l'exemple 1.2

Les conclusions sont bien-sûr identiques à celles de l'exemple précédent. GMRES(30) et GMRES(20) voient néanmoins leur temps CPU ainsi que leur coût algorithmique approcher sensiblement ceux de GMRES(50). Les solutions obtenues par les trois méthodes sont égales et le restent entre elles, lors de nouvelles expériences numériques, pour des vecteurs initiaux quelconques. L'augmentation de la taille du système a pour effet d'illustrer l'efficacité d'un long redémarrage dans l'algorithme GMRES.

Exemple 2

Le vecteur initial choisi ne nous permet pas de savoir si une solution de Krylov peut être déterminée. Dans le cas où il y a non-convergence de l'algorithme, le GMRES redémarré va continuer jusqu'à ce que le nombre maximal de cycles soit atteint. Il est alors utile d'arrêter l'algorithme pour éviter de stocker des vecteurs superflus et de perdre du temps. Le critère, portant sur $\widetilde{h}_{1,1}$, va stopper l'algorithme et nous permettre d'affirmer qu'une valeur singulière de Ritz a convergé vers zéro.

On cherche à résoudre $Ax = b$ avec A matrice CLEMENT. La non-parité de n assurera que A est bien singulière.

Exemple 2.1

On prend dans ce premier exemple $n = 501$: on applique les trois méthodes avec des tests différents sur $\widetilde{h}_{1,1}$. Le test effectué est $|\widetilde{h}_{1,1}| < \epsilon$ avec respectivement $\epsilon = 10^{-8}$, $\epsilon = 10^{-7}$ et $\epsilon = 10^{-6}$. On pourra donc observer les gains réalisés sur le coût algorithmique et le temps CPU. Dans le premier cas, on a les courbes suivantes :

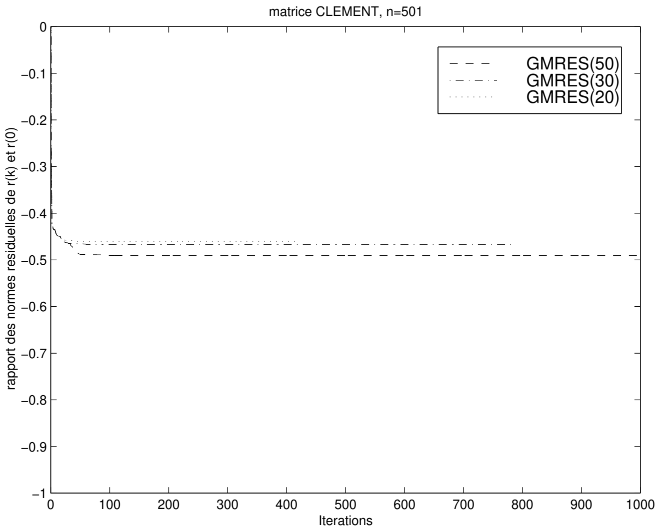


Figure 2.1 : résultats de l'exemple 2.1

	GMRES(50)	GMRES(30)	GMRES(20)
flops	1.3×10^9	8.7×10^8	2.2×10^8
temps CPU	88.52	60.25	25.28
itération d'arrêt	1000	781	341
cause de l'arrêt	non convergence	précision atteinte sur $\widetilde{h}_{1,1}$	précision atteinte sur $\widetilde{h}_{1,1}$
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	3.23×10^{-1}	3.41×10^{-1}	3.54×10^{-1}

Tableau 2.1.a : résultats de l'exemple 2.1 avec $\epsilon = 10^{-8}$

Les versions GMRES(20) et GMRES(30) s'arrêtent pour cause de précision atteinte sur $\widetilde{h}_{1,1}$. La stagnation de la norme du résidu est due à la singularité de la matrice de Hessenberg $\widetilde{H}_k$ construite à chaque redémarrage. GMRES(50) ne vérifie à aucun moment le critère sur $\widetilde{h}_{1,1}$, et atteint par conséquent le nombre maximal d'itérations fixé.

Le second test, effectué avec le même vecteur initial, donne le tableau suivant :

	GMRES(50)	GMRES(30)	GMRES(20)
flops	4.6×10^8	4.6×10^8	1.9×10^8
temps CPU	37.14	38.3	15.12
itération d'arrêt	601	751	321
cause de l'arrêt	précision atteinte sur $\widetilde{h}_{1,1}$	précision atteinte sur $\widetilde{h}_{1,1}$	précision atteinte sur $\widetilde{h}_{1,1}$
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	3.23×10^{-1}	3.41×10^{-1}	3.54×10^{-1}

Tableau 2.1.b : résultats de l'exemple 2.1 avec $\epsilon = 10^{-7}$

Une fois de plus, le critère sur $\widetilde{h}_{1,1}$ voit toute son utilité. Les gains en coût algorithmique et en temps CPU sont importants. GMRES(50) gagne 399 itérations par rapport à la version précédente.

Enfin, dans le dernier cas de figure, on obtient les résultats ci-dessous :

	GMRES(50)	GMRES(30)	GMRES(20)
flops	3.9×10^8	3.8×10^8	1.6×10^8
temps CPU	31.11	32.29	13.37
itération d'arrêt	501	631	281
cause de l'arrêt	précision atteinte sur $\widetilde{h}_{1,1}$	précision atteinte sur $\widetilde{h}_{1,1}$	précision atteinte sur $\widetilde{h}_{1,1}$
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	3.23×10^{-1}	3.41×10^{-1}	3.54×10^{-1}

Tableau 2.1.c : résultats de l'exemple 2.1 avec $\epsilon = 10^{-6}$

Les gains sont toujours aussi importants mais on peut se demander si le critère est assez puissant. En effet, $\widetilde{h}_{1,1}$ peut être petit sans impliquer nécessairement le caractère singulier de $\widetilde{H}_k$. Il se peut alors que l'algorithme s'arrête à tort. Pour cette raison, dans la suite, on prendra $\epsilon = 10^{-7}$ qui apparaît comme le meilleur compromis.

Exemple 2.2

On prend à présent $n = 502$. Dans ce cas A n'est plus singulière. On a laissé les critères d'arrêt propres au GMRES appliqué à des systèmes singuliers, c'est-à-dire ceux portant sur $\widetilde{\gamma}_k$, $\widetilde{h}_{k+1,k}$, $\|\widetilde{r}_k\|_2$ et $\widetilde{h}_{1,1}$. On obtient les résultats ci-dessous :

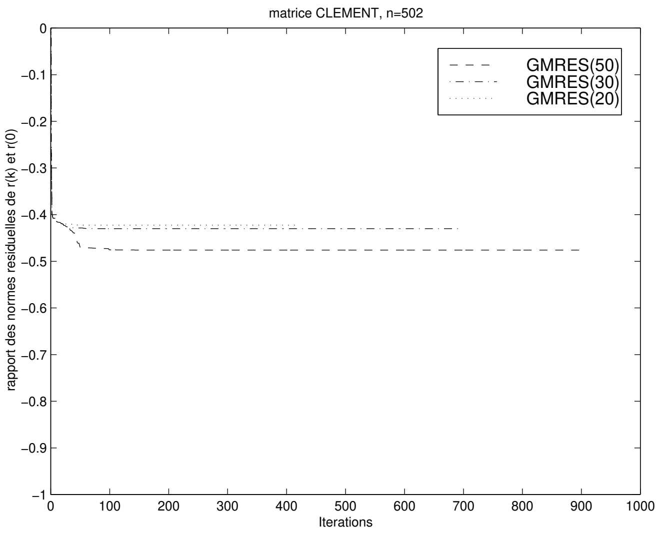


Figure 2.2 : résultats de l'exemple 2.2

	GMRES(50)	GMRES(30)	GMRES(20)
flops	1.2×10^9	7.7×10^8	4.6×10^8
temps CPU	80.65	54.09	31.13
itération d'arrêt	901	691	421
cause de l'arrêt	précision atteinte sur $\widetilde{h}_{1,1}$	précision atteinte sur $\widetilde{h}_{1,1}$	précision atteinte sur $\widetilde{h}_{1,1}$
précision obtenue sur $\|r_k\|_2/\|r_0\|_2$	3.34×10^{-1}	3.71×10^{-1}	3.78×10^{-1}

Tableau 2.2 : résultats de l'exemple 2.2

Ces résultats montrent que le critère d'arrêt portant sur $\widetilde{h}_{1,1}$ peut être utilisé dans le cas où A est inversible. Si le critère est vérifié dans ce cas, cela signifie que l'évolution sur la précision du résidu n'est plus possible. Par contre, cela n'implique pas que A soit singulière. H_k peut très bien être singulière à une certaine étape sans que le système $Ax = b$ soit singulier.

Exemple 3

Les trois exemples précédents illustrent bien l'intérêt de choisir une précision ϵ proche de 10^{-7}. On peut montrer que, si le critère sur $\widetilde{h}_{1,1}$ est trop puissant, l'arrêt de l'algorithme n'aura pas lieu alors que, théoriquement, il a tout lieu d'être.

Exemple 3.1
On considère la matrice

$$A = \begin{pmatrix} 1 & 1 & \cdots & \cdots & \cdots & 1 \\ & \ddots & \ddots & & & \vdots \\ & & 1 & \ddots & & \vdots \\ & & & 0 & \ddots & \vdots \\ & 0 & & & \ddots & 1 \\ & & & & & 0 \end{pmatrix} \in I\!R^{300\times 300}.$$

Cette matrice est très semblable à la matrice STAG utilisée dans le chapitre 2, les éléments sur la diagonale sont simplement inversés. On la nomme matrice STAGN. Le nombre d'éléments non nuls sur la diagonale principale est égal à p. Lorsque $p \geq 2$ la méthode GMRES ne peut pas converger si le vecteur initial n'est pas convenable. A admet pour rang $rg(A) = n - 1 = 299$. La précision ϵ intervenant dans le critère d'arrêt sur $\widetilde{h}_{1,1}$ est égale à 10^9. On obtient les courbes suivantes :

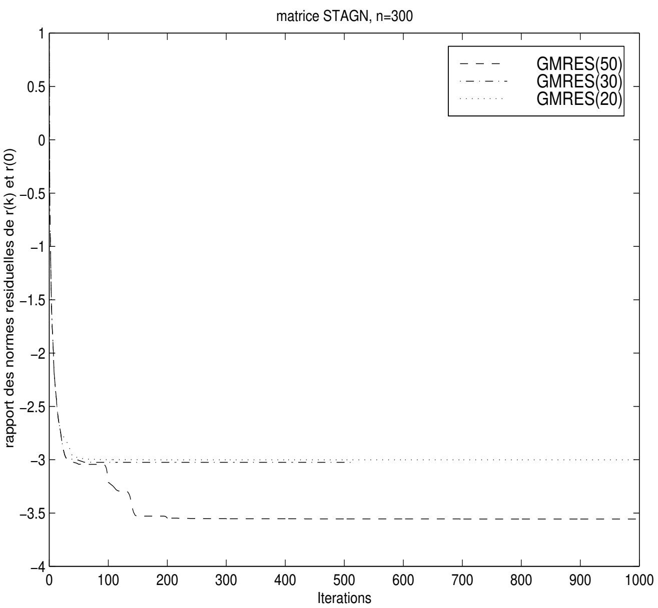


Figure 3.1 : résultats de l'exemple 3.1

	GMRES(50)	GMRES(30)	GMRES(20)
flops	4.1×10^8	1.3×10^8	2.2×10^8
temps CPU	33.78	11.74	19.09
itération d'arrêt	1000	511	1000
cause de l'arrêt	non convergence	précision atteinte sur $\widetilde{h}_{1,1}$	non convergence
précision obtenue sur $\\|r_k\\|_2/\\|r_0\\|_2$	2.78×10^{-4}	9.45×10^{-4}	9.98×10^{-4}

Tableau 3.1 : résultats de l'exemple 3.1

GMRES(50) et GMRES(20) atteignent le nombre maximal d'itérations fixé à 1000. Théoriquement, lors d'un cycle bien précis, la précision sur $\widetilde{h}_{1,1}$ doit être observée comme pour GMRES(30), ce qui montre que le critère d'arrêt est trop puissant. Il y a donc un compromis à faire sur le critère de $\widetilde{h}_{1,1}$ entre la rigueur de l'arrêt qu'il produit et les gains qu'il occasionne.
Le nombre d'itérations entre chaque redémarrage n'induit aucun résultat sur la vérification du critère d'arrêt.

Exemple 3.2
On utilise maintenant la même matrice que précédemment avec $n = 500$. Les résultats obtenus sont les suivants :

	GMRES(50)	GMRES(30)	GMRES(20)
flops	7.7×10^8	6.2×10^8	5.7×10^7
temps CPU	61.41	51.0	46.09
itération d'arrêt	1000	1000	1000
cause de l'arrêt	non convergence	non convergence	non convergence
précision obtenue sur $\\|r_k\\|_2/\\|r_0\\|_2$	4.61×10^{-1}	4.69×10^{-1}	4.74×10^{-1}

Tableau 3.2 : résultats de l'exemple 3.2

Les trois méthodes ne vérifient à aucun moment le critère d'arrêt sur $\widetilde{h}_{1,1}$. Théoriquement, il ne peut y avoir d'amélioration sur la norme résiduelle et l'arrêt semble être de bon augure. Pourtant, les méthodes atteignent le nombre maximal de cycles imposé. La discussion sur le choix de la précision de l'arrêt reste donc en suspens.

3.5 Conclusion

On a étudié dans ce chapitre les dégénérescences que peut occasionner la singularité d'un système lors de l'utilisation du GMRES. Chaque type de dégénérescence, qu'elle soit due à la déficience du problème des moindres carrés ou à la dégénérescence du sous-espace de Krylov, met en avant une propriété particulière du sous-espace de Krylov utilisé.
Des liens sont établis avec le chapitre précédent dans le sens où la définition de la matrice pseudo-inverse de $\overline{H}_k$ reste essentielle. La nature de la solution approchée initiale va régir les dégénérescences survenant dans le GMRES et son choix spécifique est une condition indispensable à la détermination d'une solution de Krylov.
La version redémarrée du GMRES va quant à elle souffrir du manque d'informations dû à son redémarrage. Si le nombre d'itérations des cycles reste fixe, le simple fait que le déterminant de $\widetilde{H}_k$ s'annule va impliquer une stagnation inéluctable de la norme du résidu. Pourquoi alors ne pas revenir en arrière dans l'algorithme, avant que le déterminant de $\widetilde{H}_k$ ne s'annule, et diminuer la valeur de m pour assurer la convergence du GMRES redémarré ? Le problème majeur de cette technique réside bien-sûr dans le choix du paramètre m. De plus, rien ne permet de certifier que la nullité de $det(\widetilde{H}_k)$ n'est pas programmée. L'intérêt d'un nouveau critère d'arrêt sur $\widetilde{H}_k$ est alors tout trouvé et voit, dans nos exemples numériques, sa réelle nécessité.
Il reste néanmoins à trouver une alternative à la connaissance indispensable de l'indice ν_1. On sait très bien que pour une matrice quelconque sa détermination sans débauche de temps et de calcul paraît illusoire. Notre optique sera dorénavant de trouver des cas pratiques où les systèmes singuliers interviennent, et d'observer quelles sont les méthodes utilisées pour permettre leur résolution.

Chapitre 4

Les systèmes singuliers dans la pratique

4.1 Introduction

Le contexte pratique qui nous intéresse dans ce chapitre est celui des équations aux dérivées partielles non linéaires dépendant d'un paramètre. Elles proviennent de l'étude de cas physiques très précis en mécanique des fluides. Elles induisent, lors de leur discrétisation par éléments finis, la résolution de systèmes d'équations linéaires non nécessairement inversibles ni consistants. Selon leur nature, leur traitement numérique diffère mais fait toujours appel à des techniques très élaborées.
Dans notre cas, on utilise les méthodes de continuation ou de cheminement pour résoudre ce type de problèmes. Connaissant un point sur la branche de solutions, un point voisin est déterminé par une prédiction de type Euler suivie d'une correction de type Newton. Ainsi, cette avancée progressive le long de la branche de solutions permet d'accéder aux points non évidents comme le sont les points de bifurcation ou les points de retournement. L'ajout au système d'une équation de normalisation supplémentaire permettra de passer ces points particuliers.

On étudie dans ce contexte particulier les méthodes utilisées et les résultats obtenus grâce à ces techniques par l'intermédiaire du GMRES et de sa version redémarrée. Le problème modèle considéré est le problème de Bratu. Il nous permettra de mettre en place les algorithmes utilisés pour sa résolution et pour celle de problèmes rentrant dans le cadre des systèmes singuliers consistants.

4.2 Quelques équations aux dérivées partielles non linéaires dépendant d'un paramètre

On s'intéresse dans cette section à la résolution de systèmes composés de n équations non linéaires dépendant de $n+1$ inconnues du type :

$$F(u, \lambda) = 0, \tag{4.1}$$

où F est une fonction non linéaire de $I\!R^{n+1}$ dans $I\!R^n$.
De tels systèmes apparaissent dans les problèmes d'homotopie ou lors de la discrétisation d'une EDP non linéaire dépendant d'un paramètre. En posant $U = \{u, \lambda\}$, la méthode de Newton appliquée à la résolution du système non linéaire d'équations

$$F(U) = 0 \tag{4.2}$$

s'utilise de la façon suivante :

Algorithme 4.1 : Méthode de Newton

1. poser U_0 un vecteur initial,
2. pour $k = 0, 1, \ldots$ jusqu'à convergence faire
   résoudre $J(U_k)\delta_k = -F(U_k)$,
   poser $U_{k+1} = U_k + \delta_k$,

où $J(U_k) = F'(U_k)$ est le système jacobien.

Pour des problèmes de grande taille, les méthodes itératives sont fréquemment utilisées pour résoudre (4.2) seulement de manière approximative. Cela donne naissance à des méthodes qui peuvent être vues comme des méthodes de Newton inexactes [19]. Pour résoudre le système intervenant dans l'étape 2. de l'algorithme, on utilise une méthode de sous-espaces de Krylov, ce qui permettra de faire le lien avec les chapitres précédents. Une méthode de sous-espaces de Krylov nécessite seulement l'action de la jacobienne J sur un vecteur v et non pas J de façon explicite. Dans les équations non linéaires, cette action peut être approchée par un quotient différence de la forme :

$$J(U)v \simeq \frac{F(U + \sigma v) - F(U)}{\sigma} \tag{4.3}$$

où U est l'approximation d'une racine de (4.2) et σ est un scalaire.
Le problème de cet algorithme est qu'il ne converge que lorsque le vecteur initial U_0 est

assez proche d'une solution. Des modifications dans cet algorithme sont donc nécessaires pour garantir la convergence de la méthode de Newton, quel que soit le vecteur de départ. Une méthode qui converge pour un vecteur intial quelconque sera appelée globalement convergente contrairement à une méthode localement convergente telle que l'itération de Newton non modifiée.
Plusieurs modifications sont utilisées pour assurer la convergence de la méthode de Newton :

1. on peut ajouter une procédure de recherche linéaire appelée "backtrack linesearch procedure" à l'algorithme basique suggérée par Dennis et Schnabel [20]. Ils proposent une stratégie globale permettant la détermination d'une racine U^* de (4.2) basée sur une méthode globalement convergente pour le problème

$$\min_{U \in \mathbb{R}^{n+1}} f(U) = \frac{1}{2} F(U)^t F(U), \tag{4.4}$$

2. on peut également utiliser une seconde approche appelée "model trust region". Une stratégie DOGLEG [12], basée sur la méthode de Powell hybride est utilisée avec restriction à un sous-espace de Krylov.

Les deux algorithmes DOGLEG et Linesearch Backtrack vont donc assurer la convergence de la méthode de Newton. Pourtant, ces deux méthodes ne vont pas permettre de déterminer des branches entières de solutions car ce sont des méthodes ponctuelles.

La seconde catégorie de méthodes permettant à l'algorithme de Newton de converger est celle des méthodes de continuation ou de cheminement. Ces méthodes permettent de déterminer des branches entières de solutions et d'atteindre ainsi les solutions non évidentes du système ainsi que les points singuliers (points de retournement ou de bifurcation). De nombreuses méthodes de cheminement ont déja été proposées : Kubiček [39], Keller [38], Glowinski, Keller et Reinhart [30], Allgower-Georg [1] (ces références proviennent de [16]). Ce sont toutes des méthodes de prédiction-correction (PC-méthodes). Il s'agit, connaissant un point solution de (4.1), d'en déterminer un nouveau voisin par une prédiction généralement de type Euler suivie d'une correction de type Newton.

Si ces méthodes ont toutes la même trame, elles diffèrent surtout par leur manière de traiter la matrice jacobienne non carrée du problème. On utilisera dans la suite la méthode de Keller, ancienne méthode de type PC qui reste encore très utilisée.

4.2.1 La méthode de Keller

4.2.1.a Notations

Soit le problème de bifurcation suivant :

$$\left\{ \begin{array}{l} \text{trouver } (u,\lambda) \in I\!R^n \times I\!R \text{ tel que} \\ F(u,\lambda) = 0 \end{array} \right. , \tag{4.5}$$

où F est une fonction non linéaire au moins de classe C^2 de $I\!R^n \times I\!R$ dans $I\!R^n$. On notera par la suite $D_uF^0 = D_uF(u_0,\lambda_0)$ et $D_\lambda F^0 = D_\lambda F(u_0,\lambda_0)$. On peut introduire les définitions suivantes provenant de [6, 7, 8] :

Définition 4.2.1
$(u_0,\lambda_0) \in I\!R^n \times I\!R$ est appelé point régulier de (4.5) si :

1. *$F(u_0,\lambda_0) = 0$*
2. *D_uF^0 est un isomorphisme de $I\!R^n$.*

A l'aide du théorème des fonctions implicites, on peut montrer qu'au voisinage de ce point passe une unique branche de solutions régulières $\{u(\lambda),\ \lambda \in]\lambda_0 - r, \lambda_0 + r[,\ r > 0\}$.

Définition 4.2.2
$(u_0,\lambda_0) \in I\!R^n \times I\!R$ est appelé point limite simple de (4.5) si :

1. *$F(u_0,\lambda_0) = 0$,*
2. *$dimKer(D_uF^0) = codimIm(D_uF^0) = 1$,*
3. *$D_\lambda F^0 \notin Im(D_uF^0)$.*

On rappelle que la codimension de l'image d'une matrice $A \in I\!R^{n\times n}$ ($codimIm(A)$) est égale à la dimension de l'image de A soustraite à la dimension de $I\!R^n$. Il est évident que si 2. est vérifiée, alors

$$I\!R^n = Ker(D_uF^0) \oplus Im(D_uF^0) \Leftrightarrow ind(D_uF^0) = 1$$

où $\mathrm{ind}(D_u F^0)$ est égal à l'indice de la matrice $D_u F^0$.
On désigne par $< .,. >$ le produit de la dualité entre $I\!R^n$ et son dual. Φ_0^* désigne l'élément du noyau de l'opérateur adjoint $(D_u F^0)^*$.

Définition 4.2.3
Soit (u_0, λ_0) un point limite simple de (4.5). Soient de plus :

1. *$\Phi_0 \in I\!R^n$ engendrant $Ker(D_u F^0)$,*
2. *$\Phi_0^* \in I\!R^n$ engendrant $Ker((D_u F^0)^*)$, tels que $\Phi_0^* \Phi_0 = 1$.*

Si $< \Phi_0^*, D_{uu}^2 F^0 \Phi_0 \Phi_0 > \neq 0$, (u_0, λ_0) est appelé point de retournement.

Définition 4.2.4
$(u_0, \lambda_0) \in I\!R^n \times I\!R$ est appelé point critique simple de (4.5) si :

1. *$F(u_0, \lambda_0) = 0$*
2. *$dim Ker(D_u F^0) = codim Im(D_u F^0) = 1$*
3. *$D_\lambda F^0 \in Im(D_u F^0)$*

Définition 4.2.5
Soit (u_0, λ_0) un point critique simple de (4.5). Soient de plus :

1. *$\Psi_0 \in Im(D_u F^0)$ tel que $D_u F^0 \Psi_0 = -D_\lambda F^0$*
2. *$A_0 \equiv < \Phi_0^*, D_{uu}^2 F^0 \Phi_0 \Phi_0 >$,*
3. *$B_0 \equiv < \Phi_0^*, D_{uu}^2 F^0 \Phi_0 \Psi_0 + D_{\lambda u}^2 F^0 \Phi_0 >$,*
4. *$C_0 \equiv < \Phi_0^*, D_{uu}^2 F^0 \Phi_0 \Phi_0 + 2 D_{\lambda u}^2 F^0 \Psi_0 + D_{\lambda\lambda}^2 F^0 >$.*

Si $B_0^2 - A_0 C_0 > 0$ alors (u_0, λ_0) est appelé point de bifurcation simple de (4.5).
Dans ce cas, au voisinage d'un point de bifurcation simple (u_0, λ_0), l'ensemble des solutions de (4.5) est constitué de deux branches qui se coupent transversalement en (u_0, λ_0).

Une fois définis tous les points particuliers survenant dans les branches de solutions, on a le

Lemme 4.2.1
Si $(u_0, \lambda_0) \in \mathbb{R}^n \times \mathbb{R}$ est un point régulier ou un point limite simple de (4.5), alors la matrice

$$DF^0 = (D_u F(u_0, \lambda_0), D_\lambda F(u_0, \lambda_0)) \tag{4.6}$$

est de rang n maximal.
Si (u_0, λ_0) est un point critique simple de (4.5), alors DF^0 est de rang $n-1$.

On peut trouver la preuve de ce lemme dans [16].

4.2.1.b Méthode de prédiction correction

On va à présent déterminer les solutions calculables de (4.1) à l'aide d'une prédiction d'Euler et une correction de type Newton. Connaissant $(u(\lambda^0), \lambda^0)$ solution de (4.1) par rapport au paramètre λ, on peut en déduire $\mathring{u}\,(\lambda^0)$ la dérivée de u par rapport à λ en λ^0, ceci à l'aide du système :

$$D_u F(u(\lambda^0), \lambda^0)\,\mathring{u}\,(\lambda^0) = -D_\lambda F(u(\lambda^0), \lambda^0)$$

On donne ensuite l'algorithme de la méthode de Keller :

Algorithme 4.2 : Méthode de Keller

λ^0, $\Delta\lambda$ donnés, $U^0 = (u^0, \lambda^0)$ donné tel que $F(u^0, \lambda^0) = 0$,
faire $k = 0$, nombre de points à calculer :

$$\left.\begin{array}{l} D_u F(u^k, \lambda^k)\,\mathring{u}\,(\lambda^k) = -D_\lambda F(u^k, \lambda^k) \\ y := u^k + \Delta\lambda\,\mathring{u}\,(\lambda^k), \\ \lambda^{k+1} := \lambda^k + \Delta\lambda, \end{array}\right\} \text{(prédiction d'Euler)} \tag{4.7}$$

$$\left.\begin{array}{l} \text{répéter} \\ D_u F(y, \lambda^{k+1}) z := F(y, \lambda^{k+1}) \\ y = y - z \\ \text{jusqu'à convergence} \end{array}\right\} \text{(correction de Newton)} \tag{4.8}$$

$u^{k+1} := y$, contrôler $\Delta\lambda$
fin faire.

Remarque 4.2.1

1. *Le choix du signe du pas de prédiction $\Delta\lambda$ dépend du sens dans lequel on veut parcourir la branche de solutions.*

2. *On peut, quitte à perdre la convergence quadratique de l'algorithme de Newton, limiter les factorisations de la matrice D_uF en utilisant une méthode de quasi-Newton (la jacobienne n'est pas remise à jour à chaque pas de l'algorithme de correction).*

On se propose de résoudre les systèmes linéaires survenant dans (4.7) et (4.8) à chaque pas de l'algorithme de cheminement à l'aide des méthodes de Krylov précédemment utilisées.

1. En présence d'un point régulier, $D_uF(u^k, \lambda^k)$ est inversible, donc la méthode de Krylov utilisée nous donnera théoriquement une solution.

2. Dans le cas d'un point limite simple et plus particulièrement un point de retournement, la jacobienne $D_uF(u^k, \lambda^k)$ est singulière de rang $n-1$. Or, le second membre n'appartient pas à l'image de la matrice. Par conséquent, le système n'est pas consistant et la méthode de Krylov ne pourra pas déterminer de solution en ce point, la prédiction d'Euler étant impossible à réaliser. On utilisera les méthodes détaillées dans la suite pour pallier à ce type de problème.

3. Dans le cas d'un point critique simple et plus particulièrement un point de bifurcation, la jacobienne est également singulière de rang $n-1$. Le second membre appartient cette fois-ci à l'image de la matrice $D_uF(u^k, \lambda^k)$. On a vu que dans ce cas, $\nu_1 = 1$ où ν_1 est la multiplicité de la valeur propre nulle dans le polynôme minimal de A. En utilisant les chapitres précédents, on peut affirmer que, théoriquement, la méthode de sous-espaces de Krylov utilisée va déterminer une solution en ce point.

Si le paramétrage naturel des solutions par rapport à λ est suffisant pour le cheminement sur une branche de points réguliers, il est clair qu'on ne réussira pas à passer un point de retournement. On rencontre le même problème au voisinage d'un point de bifurcation où deux branches de solutions se coupent transversalement : il se peut que l'on change de branche de solutions de manière involontaire.

On applique alors la méthode de cheminement au calcul numérique des branches de solutions régulières ou comportant un point de retournement du problème de bifurcation suivant :

$$\left\{ \begin{array}{l} \text{trouver } (u(s), \lambda(s)) \in I\!R^n \times I\!R \text{ tel que} \\ F(u(s), \lambda(s)) = 0, \;\; s \in J, \;\; F : I\!R^n \times I\!R \longrightarrow I\!R^n. \end{array} \right. \tag{4.9}$$

où s est un nouveau paramètre bien choisi. La fonction F est supposée suffisamment régulière (au moins de classe $\mathcal{C}^2$) et J est un intervalle ouvert de $I\!R$.

4.2.1.c Introduction d'une équation de normalisation

Une contrainte ou équation de normalisation est ajoutée au problème (4.9) pour rendre le système carré. On cherche à résoudre :

$$\begin{cases} F(u(s), \lambda(s)) = 0, \ F : \mathbb{R}^n \times \mathbb{R} \longrightarrow \mathbb{R}^n \\ N(u(s), \lambda(s), s) = 0, \ N : \mathbb{R}^n \times \mathbb{R}^2 \longrightarrow \mathbb{R}. \end{cases} \tag{4.10}$$

De manière équivalente, si l'on pose $x = (u, \lambda) \in \mathbb{R}^{n+1}$, on se ramène au problème :

$$P(x(s), s) = 0 \tag{4.11}$$

avec l'application P définie par

$$\begin{cases} P : \mathbb{R}^{n+1} \times \mathbb{R} & \longrightarrow & \mathbb{R}^{n+1} \\ \quad (x, s) & \longmapsto & (F(u, \lambda), N(u, \lambda, s)) \end{cases}$$

Plusieurs choix sont possibles pour la normalisation N. On présente ici la normalisation par pseudo-abscisse curviligne. Soit $(u_i, \lambda_i) = (u(s_i), \lambda(s_i))$ avec $i \geq 0$, une solution connue de (4.9). Pour le calcul d'un point voisin (u_{i+1}, λ_{i+1}), on définit localement l'application N par

$$N(u(s), \lambda(s), s) = \theta \ \mathring{u} \ (s_i)(u(s) - u(s_i)) + (1 - \theta) \ \mathring{\lambda} \ (s_i)(\lambda(s) - \lambda(s_i)) - (s - s_i) \tag{4.12}$$

où θ est un réel appartenant à $]0, 1[$. L'intérêt de ce choix se trouve dans le lemme suivant (cf Keller [38]) :

Lemme 4.2.2
Soit (u_0, λ_0) un point régulier ou un point limite simple du problème (4.9). Alors avec $(u(s_0), \lambda(s_0)) = (u_0, \lambda_0)$ et N définie par (4.12), il existe $\rho_0 > 0$ et une unique branche de solutions $(u(s), \lambda(s))$ de (4.10) *pour $|s - s_0| < \rho_0$. Sur cette branche de solutions, la matrice carrée de taille $(n+1, n+1)$*

$$D_x P(x(s), s) = \begin{pmatrix} F_u(u(s), \lambda(s)) & F_\lambda(u(s), \lambda(s)) \\ \theta \ \mathring{u}^* \ (s_0) & (1-\theta) \ \mathring{\lambda} \ (s_0) \end{pmatrix} \tag{4.13}$$

est non singulière.

Cela signifie que les points limites simples comme les points réguliers du problème (4.9) sont des points réguliers pour le problème (4.11). Les points de retournement ne posent donc plus de difficultés lors du cheminement, du moins localement.

Il est bien évident que le problème (4.11) se résout de la même façon que le problème (4.9) à l'aide de la méthode de Keller. On remplace simplement u par x, λ par s et F par P. On comparera par la suite, à l'aide d'exemples bien précis, la méthode de Keller appliquée à la fois à (4.5) et à (4.11).

4.2.2 Un problème simple : le problème de Bratu

4.2.2.a Formulation du problème, propriétés des solutions

Dans cette partie, on s'inspire du travail de Glowinski, Keller et Reinhart [30]. On considère la solution numérique d'un problème de Bratu modifié, i.e. trouver une solution du problème non linéaire avec valeurs aux bords :

$$\begin{cases} -\Delta u = \lambda e^u + f \text{ dans } \Omega \\ u = 0 \text{ sur } \partial\Omega \end{cases} . \tag{4.14}$$

Ici Ω est un domaine borné de $I\!R^n$, et $\partial\Omega$ son contour. On note également que $\overline{\Omega} = \Omega \cup \partial\Omega$. On décrit par $x = \{x_i\}_{i=1}^n$ le point générique de $I\!R^n$ et on définit la mesure dx par $dx = dx_1 \dots dx_n$. L'espace de Sobolev-Hilbert

$$H_0^1(\Omega) = \{v/v \in L^2(\Omega),\ \frac{\partial v}{\partial x_i},\ \forall i = 1, \dots, n,\ v = 0 \text{ sur } \partial\Omega\}$$

est muni du produit scalaire

$$(u, v)_{H_0^1(\Omega)} = \int_\Omega \nabla u \nabla v dx \tag{4.15}$$

et de la norme correspondante

$$\|v\|_{H_0^1(\Omega)} = (\int_\Omega |\nabla v|^2 dx)^{1/2}. \tag{4.16}$$

On peut introduire le résultat suivant :

Proposition 4.2.1
Il existe une valeur critique de λ appelée $\lambda^* > 0$ telle que :

1. *si $\lambda > \lambda^*$, alors le problème (4.14) n'admet aucune solution,*
2. *si $\lambda \in]0, \lambda^*[$, alors le problème (4.14) admet deux solutions appartenant à $H_0^1(\Omega) \cap W^{2,p}(\Omega)$, $\forall p \geq 1$, où*
$$W^{2,p}(\Omega) = \{v/v,\ \frac{\partial v}{\partial x_i},\ \frac{\partial^2 v}{\partial x_i \partial x_j} \in L^p(\Omega),\ \forall 1 \leq i, j \leq n\},$$
3. *si $\lambda = \lambda^*$, il existe une unique solution $u^* \in H_0^1(\Omega) \cap W^{2,p}(\Omega)$, $\forall p \geq 1$. De plus, $\{u^*, \lambda^*\}$ est un point limite simple pour l'équation $F(u, \lambda) = 0$ où F est l'opérateur défini sur $H_0^1(\Omega) \times I\!R$ par*
$$F(u, \lambda) = -\Delta u - \lambda e^u - f.$$

4.2.2.b Approximation par éléments finis du problème de Bratu

1. Formulation variationnelle du problème de Bratu - Triangulation de Ω - Espaces discrets fondamentaux

Une formulation variationnelle du problème de Bratu (4.14) qui convient aux approximations par éléments finis est donnée par :

$$\left\{ \begin{array}{l} \text{trouver } \{u, \lambda\} \in H_0^1(\Omega) \times I\!R \text{ tel que} \\ \int_\Omega \nabla u \nabla v dx = \int_\Omega (\lambda e^u + f) v dx \ \forall v \in H_0^1(\Omega) \end{array} \right. . \tag{4.17}$$

On décrit seulement l'approximation du problème (4.14) pour $n = 2$. On suppose également que Ω est un domaine polygonal de $I\!R^2$. On considère une famille standard d'éléments finis triangulaires $\{\tau_h\}_h$ de Ω, i.e. pour un h donné, τ_h est une ensemble fini de sous-triangles (fermés) T de Ω tels que

(a) $\cup_{T \in \tau_h} T = \overline{\Omega}$

(b) $\forall T, T' \in \tau_h,\ T \neq T'$, on a soit

  i. $T \cap T' = \emptyset$, soit

  ii. T, T' ont seulement un sommet en commun, soit

  iii. T, T' ont seulement une arête en commun,

(c) h est la longueur maximale des arêtes de $T \in \tau_h$.

On approche $H_0^1(\Omega)$ par l'espace de dimension finie

$$V_{0h} = \{v_h / v_h \in C^0(\overline{\Omega}),\ v_h|_T \in \mathcal{P}_1,\ \forall T \in \tau_h,\ v_h = 0 \text{ sur } \partial\Omega\} \tag{4.18}$$

où $\mathcal{P}_1$ est l'espace des polynômes à deux variables de degré ≤ 1. Il suit que $dim V_{0h} = n_{0h}$ où n_{0h} est le nombre de sommets de τ_h intérieurs à Ω et $V_{0h} \subset H_0^1(\Omega)$.

2. Formulation des problèmes approchés

On prend comme problème approché

$$\left\{ \begin{array}{l} \text{trouver } \{u_h, \lambda\} \in V_{0h} \times I\!R \text{ tel que} \\ \int_\Omega \nabla u_h \nabla v_h dx = \int_\Omega (\lambda e^{u_h} + f) v_h dx \ \forall v_h \in V_{Oh}(\Omega) \end{array} \right. . \tag{4.19}$$

Ce problème est équivalent à un système d'équations non linéaires dans $I\!R^{n_{oh}+1}$. Afin d'obtenir ce système, on suppose que l'ensemble Σ_{0h} des sommets de τ_h a été ordonné de telle sorte que

$$\Sigma_{0h} = \{P_i\}_{i=1}^{N_{0h}}, \tag{4.20}$$

où les P_i sont les points intérieurs à la triangulation utilisée. A chaque P_i de Σ_{0h} on associe la fonction ω_i satisfaisant

$$\omega_i \in V_{0h},\ \omega_i(P_j) = \delta_{i,j},\ 1 \leq i, j \leq n_{0h} \tag{4.21}$$

qui est la fonction chapeau ou fonction de Kronecker. La famille $B_{0h} = \{\omega_i\}_{i=1}^{n_{0h}}$ est une base de l'espace vectoriel V_{0h} et on a $\forall v_h \in V_{0h}$:

$$v_h = \sum_{i=1}^{n_{0h}} v_h(P_i)\omega_i. \tag{4.22}$$

En utilisant (4.22) dans (4.18), on obtient le système non linéaire :

$$\sum_{j=1}^{n_{0h}} (\int_\Omega \nabla\omega_i \nabla\omega_j dx) u_h(P_j) = \int_\Omega (\lambda e^{(\sum_{j=1}^{n_{0h}} u_h(P_j)\omega_j)} + f)\omega_i dx, \tag{4.23}$$

ceci pour $1 \leq i \leq n_{0h}$. Ici le vecteur inconnu est $\{\{u_h(P_i)\}_{i=1}^{n_{0h}}, \lambda\} \in I\!R^{n_{0h}} \times I\!R$. Puisque ∇_i et ∇_j sont des fonctions constantes, le calcul de la partie gauche de l'égalité précédente est assez simple. Les intégrales intervenant dans la partie droite de (4.23) peuvent être calculées exactement.

Cependant, dans le but de réduire les calculs, on évalue ces intégrales de manière approchée : on appliquera à

$$\int_\Omega e^{u_h}\omega_i dx$$

la règle trapézoïdale bi-dimensionnelle, c'est-à-dire

$$\int_T \Phi(x)dx \simeq \frac{1}{3} aire(T) \sum_{j=1}^{3} \Phi(P_{jT}), \tag{4.24}$$

où $P_{jT},\ j = 1, 2, 3$ sont les sommets de T. La formule sera exacte si $\Phi \in \mathcal{P}_1$.

Si τ_h est une triangulation régulière, l'utilisation de (4.24) pour calculer la partie droite de (4.22) recouvre les schémas classiques de différences finies pour la discrétisation de (4.14).

4.2.2.c Solution numérique du problème discret de Bratu

1. Résolution sans équation de normalisation

On veut résoudre le problème discret de Bratu (4.23), ce qui suggère l'utilisation de l'algorithme suivant :

Algorithme 4.3

- initialisation : on pose

$$\lambda^0 = 0. \tag{4.25}$$

La solution u_h^0 correspondante est l'unique solution du problème de Bratu linéaire discret donné ci-dessous sous forme variationnelle :

$$\left\{\begin{array}{l}\text{trouver } u_h^0 \in V_{0h} \text{ tel que} \\ \int_\Omega \nabla u_h^0 \nabla v_h dx = \int_\Omega f v_h dx \ \forall v_h \in V_{0h}.\end{array}\right. \tag{4.26}$$

Ceci équivaut au système linéaire obtenu dans (4.19) en posant $\lambda = 0$. La matrice de ce système est symétrique et définie positive.

• continuation : avec $\Delta\lambda$ (> 0) comme pas fixé dans la longueur d'arc, on définit pour $k \geq 0$, une approximation u_h^{k+1} ($\in V_{0h}$) de $\{u_h((k+1), \Delta\lambda)\}$ comme solution du système variationnel non linéaire suivant :

$$\left\{\begin{array}{l}\text{trouver } u_h^{k+1} \in V_{0h} \text{ tel que} \\ \int_\Omega \nabla u_h^{k+1} \nabla v_h dx = \int_\Omega (\lambda_{k+1} e^{u_h^{k+1}} + f) v_h dx \ \forall v_h \in V_{0h}.\end{array}\right. \tag{4.27}$$

La relation (4.27) provient bien évidemment de (4.19). En posant

$$G(u, \lambda) \equiv \int_\Omega \nabla u \nabla v_h dx - \int_\Omega (\lambda e^u + f) v_h dx,$$

on est amené à résoudre $G(u_h, \lambda) = 0$. On peut très facilement vérifier que trouver (u, λ) vérifiant $F(u, \lambda) = 0$ est équivalent à la résolution de $G(u, \lambda) = 0$. La résolution de (4.27) se fait à l'aide de la méthode de Keller définie dans 4.2.1.b.

Cette méthode se nommera méthode sans équation de normalisation (s.e.n).

2. Résolution avec équation de normalisation

On reprend l'algorithme de Glowinski, Keller et Reinhart [30], appliqué à la résolution du système (4.11) :

Algorithme 4.4

• initialisation : on pose

$$\lambda^0 = 0$$

et on résout (4.26). On prend $\{u_h^0, 0\}$ comme point sur l'arc de solutions $\{u_h(s), \lambda(s)\}$ pour lequel $s = 0$.
On décrit ensuite $\frac{dX(s)}{ds}$ par $\overset{\circ}{X}(s)$ pour $X = u_h$ ou $X = \lambda$. Alors, par différentiation de (4.19) par rapport à s, on obtient en $s = 0$:

$$\int_\Omega \nabla \overset{\circ}{u}_h (0) \nabla v_h dx = \overset{\circ}{\lambda}(0) \int_\Omega e^{u_h^0} v_h dx, \ \forall v_h \in V_{0h} \tag{4.28}$$

avec $u_h(0) \in V_{0h}$.
On prend comme définition de s :

$$\int_\Omega |\nabla \overset{\circ}{u}_h (0)|^2 dx + \overset{\circ}{\lambda}{}^2 (0) = 1. \tag{4.29}$$

On définit ensuite $\hat{u}_h$ comme la solution de

$$\begin{cases} \hat{u}_h \in V_{0h}, \\ \int_\Omega \nabla \hat{u}_h \nabla v_h dx = \int_\Omega e^{u_h^0} v_h dx, \ \forall v_h \in V_{0h}. \end{cases} \tag{4.30}$$

On a alors à l'aide des trois égalités précédentes

$$(\overset{\circ}{u_h(0)}) = \overset{\circ}{\lambda} (0) \ \hat{u}_h, \tag{4.31}$$

$$\overset{\circ}{\lambda}{}^2 (0) = (1 + \int_\Omega |\nabla \hat{u}_h|^2 dx)^{-1}. \tag{4.32}$$

Puisque $\lambda > 0$ et qu'on a posé $\lambda(0) = 0$, on doit orienter l'arc solution de telle façon que $\frac{d\lambda}{ds} = \overset{\circ}{\lambda} \geq 0$. Alors (4.32) va impliquer

$$\overset{\circ}{\lambda} (0) = (1 + \int_\Omega |\nabla \hat{u}_h|^2 dx)^{-1/2}. \tag{4.33}$$

• continuation : avec Δs (> 0) comme pas fixé dans la longueur d'arc, on définit pour $k \geq 0$ une approximation $\{u_h^{k+1}, \lambda^{k+1}\}$ ($\in V_{0h} \times I\!R$) de

$$\{u_h((k+1)\Delta s), \lambda((k+1)\Delta s)\}$$

comme la solution du système variationnel non linéaire suivant :

$$\text{trouver } \{u_h^{k+1}, \lambda^{k+1}\} \in V_{0h} \times I\!R \text{ tel que}$$

$$\int_\Omega \nabla u_h^{k+1} \nabla v_h dx = \int_\Omega (\lambda^{k+1} e^{u_h^{k+1}} + f) v_h dx, \ \forall v_h \in V_{0h}, \tag{4.34}$$

$$\int_\Omega \nabla (u_h^1 - u_h^0) \nabla \overset{\circ}{u}_h (0) dx + (\lambda^1 - \lambda^0) \overset{\circ}{\lambda} (0) = \Delta s \text{ si } k = 0, \tag{4.35}$$

$$\int_\Omega \nabla (u_h^{k+1} - u_h^k) \nabla (\frac{u_h^k - u_h^{k-1}}{\Delta s}) dx + (\lambda^{k+1} - \lambda^k)(\frac{\lambda^k - \lambda^{k-1}}{\Delta s}) = \Delta s \text{ si } k \geq 1. \tag{4.36}$$

Pour résoudre ce système composé de (4.34), (4.35) et (4.36), on utilise la méthode de Keller associée à la méthode GMRES.

Cette méthode se nommera méthode avec équation de normalisation (a.e.n).

4.2.3 Exemples numériques

4.2.3.a Les points réguliers et les points de retournement

1. Choix des paramètres

On utilise, pour illustrer les notions de points réguliers et de points de retournement définis précédemment, le problème de Bratu. La prédiction d'Euler ainsi que la correction de Newton font apparaître des systèmes linéaires qui seront résolus à l'aide de l'algorithme GMRES. Le critère d'arrêt de ces méthodes porte sur la norme résiduelle $\|r_k\|_2$: il est donné explicitement par

$$\|r_k\|_2 \leq 10^{-8}.$$

On utilise un critère absolu pour arrêter l'algorithme car, lors des expériences numériques, le temps mis pour atteindre la convergence à chaque pas de la méthode est très important. Un critère portant sur la norme résiduelle relative ne fait qu'accroître ce temps de calcul et les gains obtenus sur la précision de la solution sont négligeables.
On utilise la méthode de Keller afin de résoudre (4.9) et (4.11) dans le cadre du problème de Bratu. Les algorithmes sont programmés sur une machine Sun en fortran 77, en double précision.

On considère, dans le cadre de la méthode s.e.n (sans équation de normalisation), un pas constant jusqu'au voisinage du point de retournement et un pas adaptatif afin d'approcher ce point par la suite. On prendra $\Delta\lambda = 0.1$. L'action de la jacobienne sur un vecteur (4.3) est utilisée en prenant $\sigma = 10^{-7}$.
La méthode GMRES employée dans la méthode de Keller sera utilisée également en version redémarrée généralement toutes les 20 itérations au voisinage d'une branche de solutions régulières. Elle sera utilisée en version redémarrée ou non à l'approche d'un point de retournement ou de bifurcation. Dans tous les tests numériques, l'approximation initiale x_0 des systèmes linéaires associés à la méthode de Keller est initialisée à $(0, 0, \ldots, 0)^t$.

On considère $\Omega =]0,1[\times]0,1[$. Ce cas de figure a été approché en utilisant une triangulation τ_h constituée de 228 triangles réguliers. Les inconnues sont les valeurs prises par la solution u_h aux noeuds intérieurs de τ_h dont le nombre s'élève à 121 dans nos exemples numériques. Un nombre d'inconnues plus important peut être considéré mais cela augmente considérablement les temps de calcul. On utilisera donc ce maillage particulier qui confère aux solutions obtenues une précision satisfaisante. On pourra également se référer aux résultats obtenus à l'aide de maillage présenté dans [30].

On représentera généralement trois types de graphiques :

(a) la composante maximale $u_h(.5,.5)$ du vecteur solution en fonction du paramètre λ,

(b) la norme résiduelle relative $\|r_k\|_2/\|r_0\|_2$ à l'aide d'une échelle logarithmique en fonction du nombre d'itérations k,

(c) le rapport de deux résidus consécutifs $\|r_k\|_2/\|r_{k-1}\|_2$ en fonction du nombre d'itérations k.

2. Description des exemples numériques

Les méthodes s.e.n et a.e.n (avec équation de normalisation) sont utilisées l'une après l'autre et donnent, dans le cas du problème de Bratu avec $f = 0$, les résultats de la figure 1.

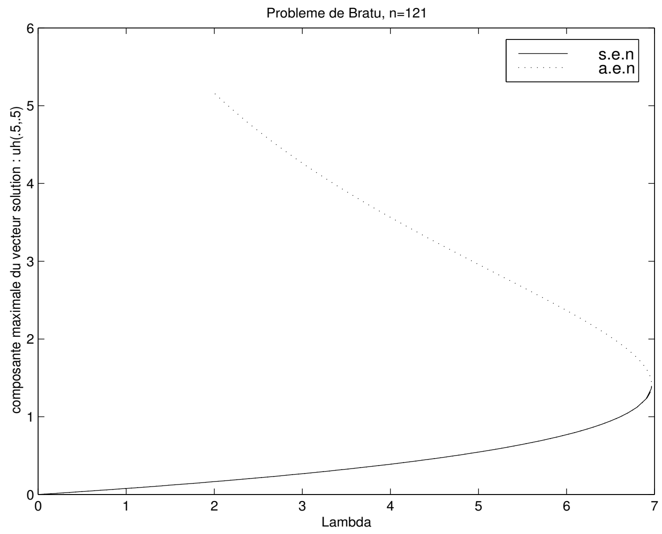


Figure 1 : composante maximale de la solution du problème de Bratu pour $f = 0$ en fonction de λ.

On peut situer le point de retournement à $\lambda \simeq 6.966$. Glowinski, Keller et Reinhart [30], en utilisant un maillage plus précis, ont déterminé ce même point à $\lambda \simeq 6.808$.

On remarque bien évidemment que la branche de solutions pour $\lambda \in [0, \lambda^*]$ n'est pas complète pour la méthode s.e.n car elle admet un point de retournement. La méthode GMRES doit alors résoudre, lors de la prédiction d'Euler, un système singulier non consistant : aucune solution de Krylov ne peut être déterminée. Puisque

la paramétrisation utilisée lors du cheminement dépend de λ, elle ne permettra pas de passer un point singulier ni d'atteindre la seconde partie de la branche de solutions.
La méthode a.e.n ne rencontre quant à elle aucun problème. Elle passe le point de retournement sans encombre puisqu'en un tel point, le système à résoudre est inversible. La seconde partie de la branche de solutions est déterminée par l'intermédiaire de la nouvelle paramétrisation qui dépend cette fois-ci de s.

On donne ensuite deux tableaux de comparaison du temps CPU mis par le GMRES dans les méthodes s.e.n et a.e.n. On y présente respectivement en fonction de la valeur λ, le nombre d'itérations nécessaire au GMRES pour atteindre la convergence dans la prédiction d'Euler, le temps que lui prend cette étape ainsi que le temps total utilisé lors de la continuation.
Le pas utilisé dans la méthode s.e.n est $\Delta\lambda = 10^{-1}$ et celui dans la méthode a.e.n est $\Delta s = 10^{-1}$.

valeur de λ	méthode s.e.n	méthode a.e.n
0.20	14 38.4127 134.65	17 24.5109 114.032
1.00	14 38.5101 762.047	18 26.0223 690.384
1.50	14 38.4969 1171.01	19 27.3363 1076.27
2.00	14 38.5895 1584.04	20 28.6354 1470.01
2.50	14 38.4686 2000.85	20 28.6831 1874.12
3.00	15 41.1078 2423.06	20 28.6483 2282.24

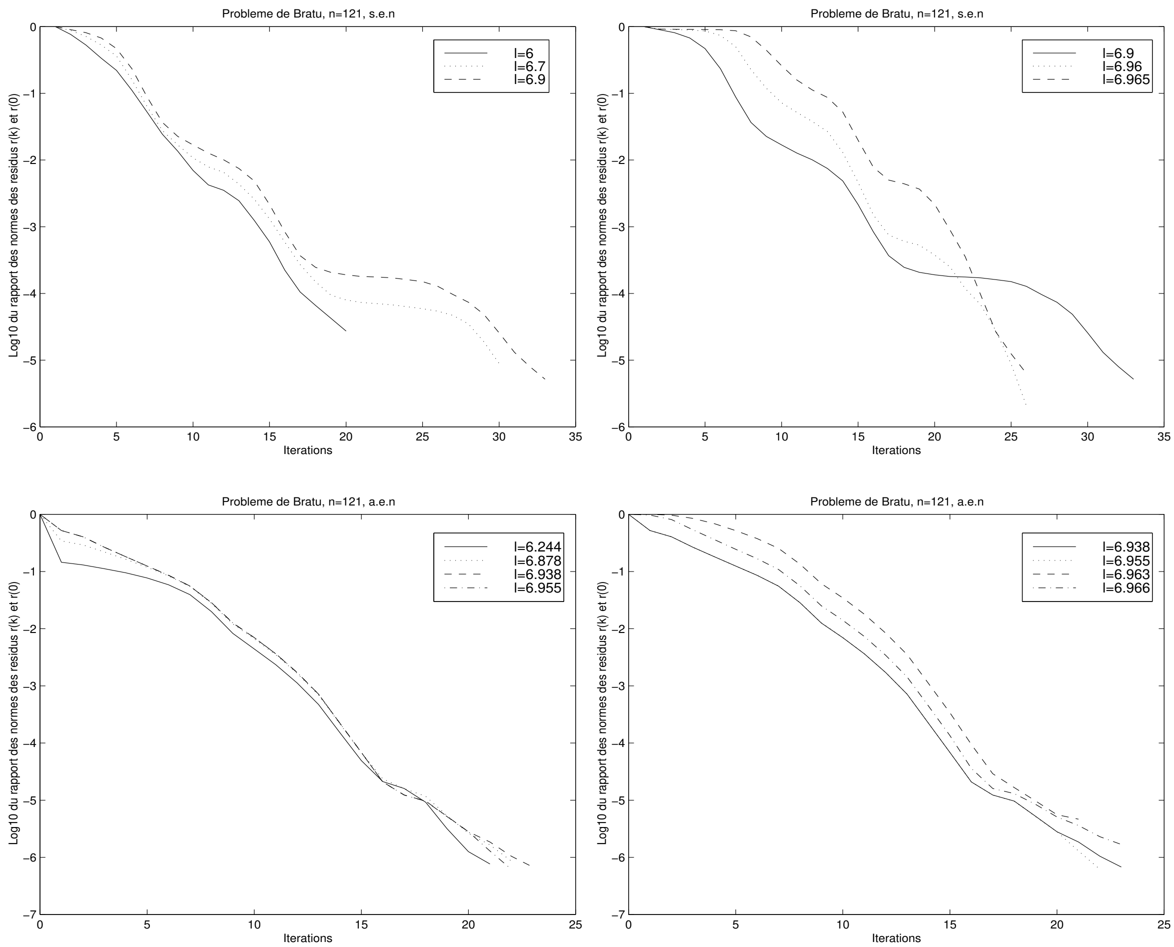
Tableau 1.a : résultats associés à la figure 1

valeur de λ	méthode s.e.n	méthode a.e.n
3.50	15 41.0098 2862.71	19 27.3420 2690.41
4.00	16 43.6859 3320.50	20 28.6290 3102.46
4.50	16 43.6242 3793.02	20 28.6697 3520.97
5.00	16 43.6476 4272.13	20 28.6707 3943.50
5.50	16 43.6948 4765.57	20 28.6635 4373.96
6.00	17 46.3143 5275.86	21 31.2913 5282.89
6.50	17 46.3831 5818.61	22 32.6019 5761.56

Tableau 1.b : résultats associés à la figure 1

Il est clair qu'à l'approche du point de retournement, les deux méthodes voient leur nombre d'itérations nécessaire à la convergence augmenter. Sur la branche de solutions régulières, elles passent de 14 à 17 pour la méthode s.e.n et de 17 à 22 pour la méthode a.e.n. Le temps utilisé dans la prédiction d'Euler augmente de façon régulière pour les deux méthodes : il est de l'ordre de 8 secondes pour passer de $\lambda = 0.2$ à $\lambda = 6.5$. On voit également que le temps CPU mis par la méthode a.e.n lors de la prédiction d'Euler est très inférieur à celui de la méthode s.e.n d'environ 15 secondes mais le temps global reste équivalent. La correction de Newton compense certainement cet écart. Les deux méthodes semblent donc avoir la même performance sur une branche de solutions régulières.

On trace ensuite pour plusieurs valeurs de λ (symbolisées par l dans les graphiques) la norme résiduelle relative associée à la solution x_k obtenue par le GMRES et les méthodes s.e.n et a.e.n lors de la prédiction d'Euler. On obtient les graphes suivants :



Figures 2.a,b,c,d

De manière générale, les figures doivent être ordonnées de gauche à droite puis de haut en bas.
La tendance globale de la méthode s.e.n qui se détache de la figure 2.a est que plus le paramètre λ s'approche de λ^*, plus le nombre d'itérations nécessaire au GMRES pour atteindre la précision dans la prédiction d'Euler augmente.

Ça n'est pas du tout le cas pour la méthode a.e.n (figure 2.c) qui semble n'avoir besoin que de 23 itérations au maximum pour vérifier le critère de convergence.

Pour la méthode s.e.n, on remarque également, à l'aide de ces figures, une tendance de plus en plus lente à converger dès le départ. Il y a stagnation de la norme résiduelle au commencement de l'algorithme, puis la convergence a lieu.
On remarque que les courbes obtenues pour $\lambda = 6.965$ et $\lambda = 6.96$, représentées

dans la figure 2.b, induisent un nombre d'itérations inférieur à celui de la courbe obtenue pour $\lambda = 6.9$. On peut expliquer ce phénomène en précisant que le pas utilisé lors du cheminement est plus petit. Ainsi, le nombre d'itérations réalisé pour obtenir la convergence est bien moindre puisque le critère d'arrêt ne porte pas sur la norme résiduelle relative.

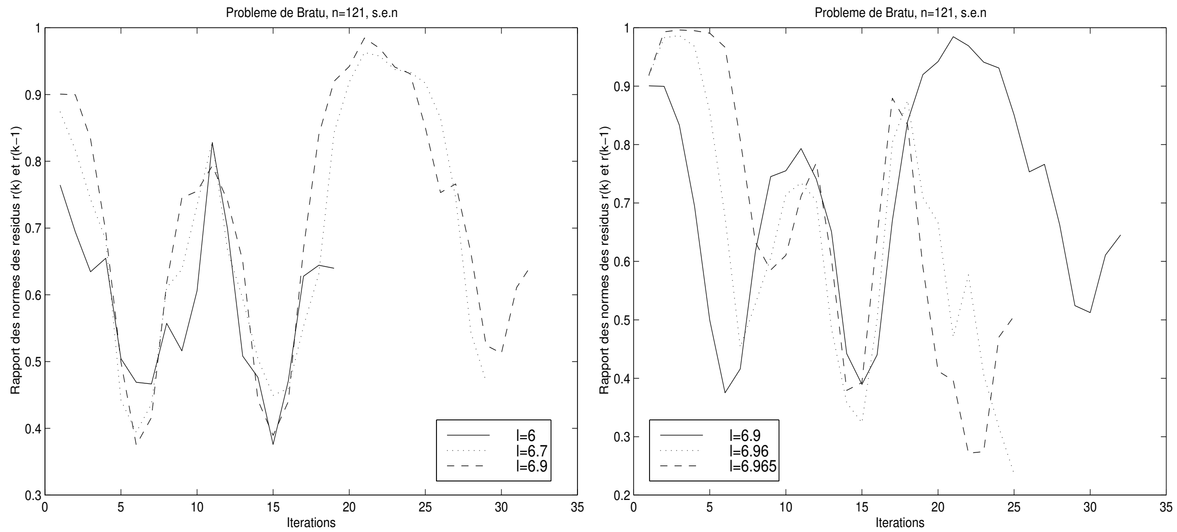
On donne ci-dessous un tableau récapitulant le nombre d'itérations nécessaire au GMRES (pour les méthodes s.e.n et a.e.n) afin d'atteindre la précision souhaitée, ceci pour des valeurs de λ comparables pour les deux méthodes :

λ	6.24	6.88	6.94	6.955	6.963	6.966
méthode s.e.n	16	19	24	26	29	31
méthode a.e.n	21	22	23	22	23	21

Tableau 2

Les résultats sont clairs : la méthode s.e.n éprouve de plus en plus de mal à voir sa norme résiduelle tendre vers zéro à l'approche d'un point de retournement. Le nombre d'itérations passe de 16 à 31, soit 15 itérations supplémentaires. Ce n'est pas le cas pour la méthode a.e.n qui voit son nombre d'itérations stagner même au voisinage du point singulier.

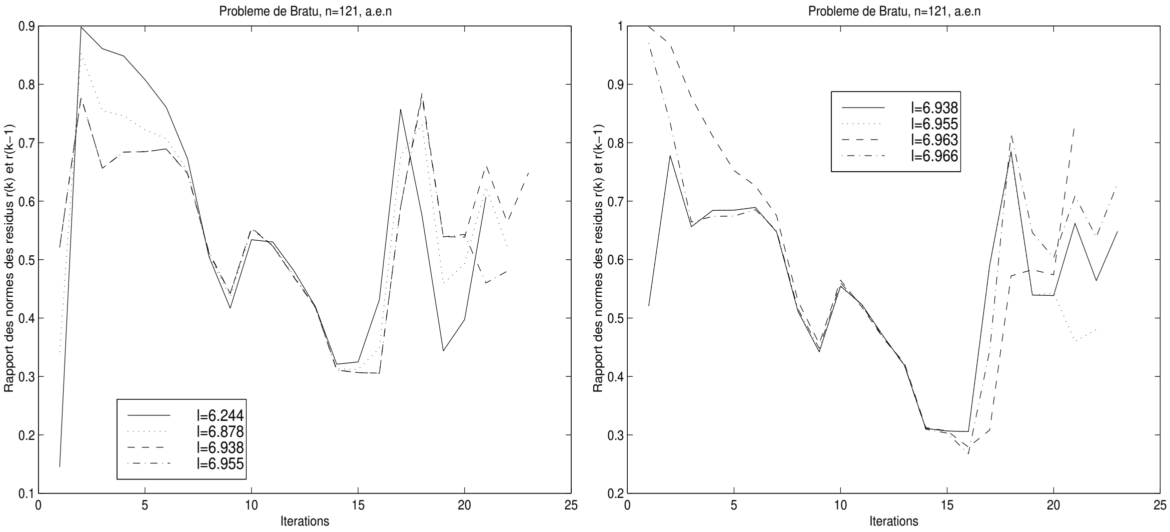
On peut également s'intéresser aux rapports de deux normes résiduelles consécutives associés aux figures précédentes. Lorsque la méthode s.e.n est utilisée, on obtient les figures suivantes :



Figures 3.a,b

Il ne semble pas à première vue qu'il y ait une aptitude générale du GMRES à converger de la même manière lorsque λ varie. On ne peut pas dégager de ces graphiques une allure générale.
On voit à l'aide de la figure de droite qu'il y a stagnation de la norme résiduelle lorsque $\lambda = 6.965$ pour $2 \leq k \leq 5$. Cela signifie que $det(\overline{H}_k^t \overline{H}_k) \gg det(H_k)^2$ à l'aide de (2.51). On ne peut pourtant pas affirmer qu'il y a une tendance à la stagnation lorsque λ approche la valeur limite λ^*.

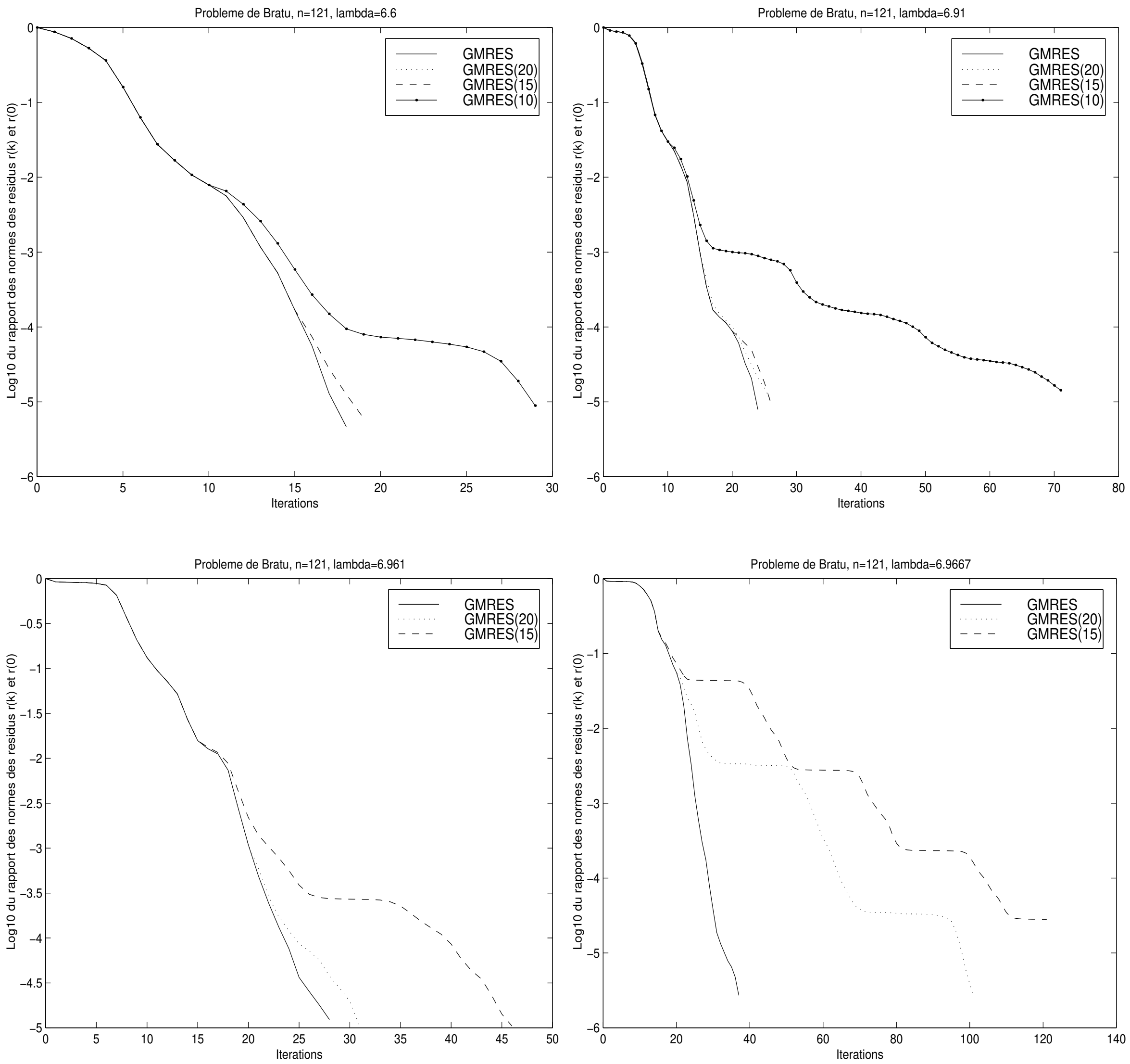
Lorsque la méthode a.e.n est utilisée, on obtient les graphes ci-dessous :



Figures 4.a,b

Cette fois-ci, un mouvement général apparaît, ce qui est évident puisque les courbes des figures 2.c et 2.d s'emboîtent pratiquement. Aucune stagnation n'apparaît même lorsque λ approche λ^*. $Det(H_k)$ n'est donc jamais petit par rapport à $det(\overline{H}_k^t \overline{H}_k)$ ce qui empêche à c_k, défini par (2.29), de vérifier $c_k \simeq 0$.

On s'interroge également sur l'utilité et l'efficacité du redémarrage du GMRES dans de tels cas. Pour cela on va utiliser GMRES(20), GMRES(15) et GMRES(10) lors de la résolution du système propre à la prédiction d'Euler à l'aide de la méthode s.e.n. On compare la convergence des trois méthodes redémarrées avec la méthode GMRES.



Figures 5.a,b,c,d

On remarque qu'à l'approche du point de retournement, plus le nombre d'itérations entre chaque redémarrage est petit, plus la convergence est longue à observer. On voit même apparaître dans la figure 5.d des plateaux de stagnation de manière régulière, laissant l'impression d'une convergence en escalier.
GMRES(10) n'apparaît plus dans les deux dernières figures car il y a stagnation de cette dernière, suivie de la non convergence de l'algorithme, le nombre maximal de cycles étant atteint. GMRES reste donc la méthode la plus performante pour la résolution de ce type de systèmes.

4.2.3.b Les problèmes de bifurcation via les bifurcations perturbées

1. Généralités

On a vu précédemment, à l'aide du problème de Bratu, comment intervenait un point de retournement dans une branche de solutions. On a vu également que passer un point de retournement sur la branche revenait à résoudre un système singulier linéaire non consistant. Ceci ne rentre pas dans le cadre général qu'on s'est fixé dans les chapitre précédents, mais la marche à suivre est désormais clairement définie. Afin de vérifier réellement la théorie dégagée sur les systèmes singuliers consistants, on introduit les problèmes de bifurcation.
On discute ici du traitement numérique des problèmes non linéaires du second ordre avec valeurs aux bords dont les branches de solutions mettent en évidence une bifurcation. On rappelle qu'au voisinage d'un point de bifurcation simple, l'ensemble des solutions de (4.5) est constitué de deux branches qui se coupent transversalement. Les branches de solutions sont calculées à l'aide des méthodes de cheminement précédentes. Le processus est basé sur l'utilisation d'une méthode de perturbation reliée au concept de bifurcation perturbée (on pourra voir Keener-Keller [37], Matkowsky-Reiss [43]) : en passant du problème perturbé au problème non perturbé (en relâchant le paramètre de bifurcation), on peut retrouver la branche de solutions qui bifurque.

2. Le problème de Dirichlet

Avec Ω un domaine borné de $I\!R^2$, on considère la solution dans $H_0^1(\Omega)$ du problème de Dirichlet non linéaire

$$\begin{cases} -\Delta u = \lambda u^2 + f \text{ dans } \Omega \\ u = 0 \text{ sur } \partial\Omega \end{cases} . \tag{4.37}$$

Ici f est le paramètre perturbateur. Le problème non perturbé (obtenu avec $f = 0$) admet deux branches distinctes de solutions pour $\lambda \geq 0$:

(a) la branche triviale $\{u, \lambda\} = \{0, \lambda\}$ $\forall \lambda \in I\!R$,

(b) une branche non triviale qui ne coupe jamais la première.

Par symétrie par rapport à l'origine, on obtient les solutions non perturbées de (4.37) correspondant à $\lambda \leq 0$. Donc, en général, le problème non perturbé admet trois branches de solutions disjointes : $u \equiv 0$, $\lambda \in I\!R$ et deux branches hyperboliques, l'une pour $\lambda > 0$ et l'autre pour $\lambda < 0$. Il suit d'après [30] que le problème perturbé admet seulement deux branches distinctes. La première est une perturbation d'une des branches hyperboliques et contient seulement des solutions régulières. La seconde est formée à partir de la branche triviale perturbée joignant la seconde branche hyperbolique et contient simplement un point limite.

On prend dans les applications numériques $f = 8$ comme paramètre perturbateur, on passe le point de retournement puis, lorsqu'on a suivi la seconde partie de la branche de solutions à l'aide de la méthode a.e.n de manière conséquente, on relâche la perturbation et on pose $f = 0$ dans l'algorithme.
On obtient la courbe suivante :

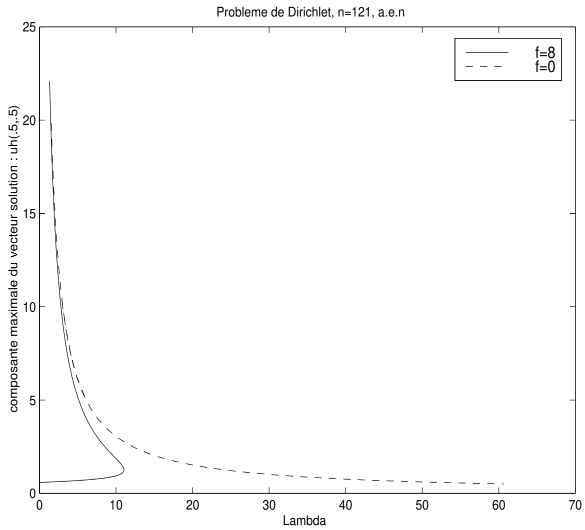
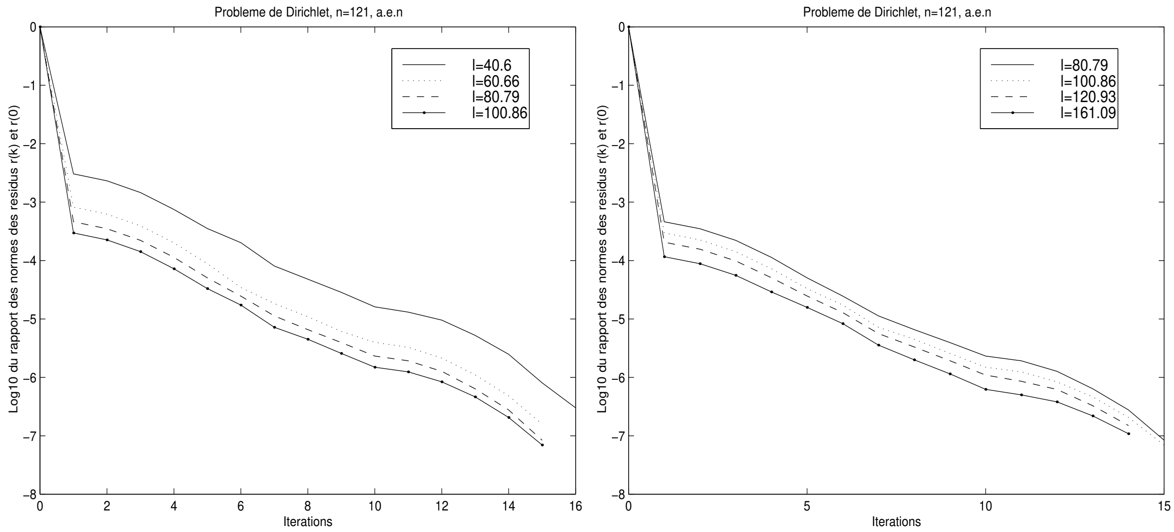


Figure 6 : composante maximale de la solution du problème de Dirichlet en fonction de λ.

Des expériences donnant des résultats similaires peuvent être obtenues en utilisant un paramètre perturbateur différent.

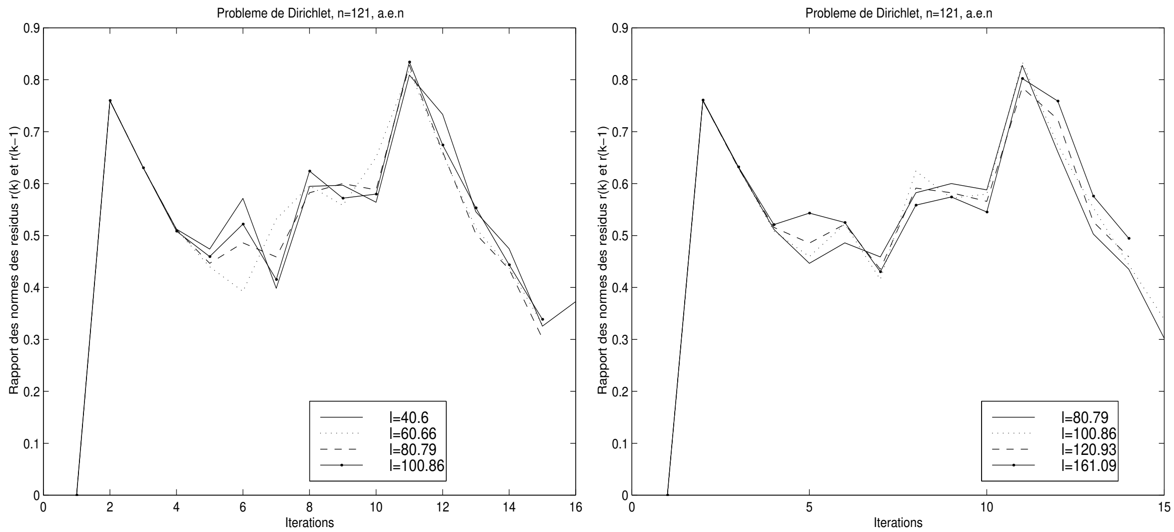
Le problème de Dirichlet non perturbé n'est pas techniquement un problème de bifurcation car il admet un point de bifurcation en $\lambda = +\infty$. Néanmoins, il fournit un exemple de problème de perturbation. La méthode utilisée pour résoudre (4.37) est bien évidemment la méthode a.e.n. Elle permet de passer sans encombre le point de retournement du problème perturbé.

On s'intéresse ensuite plus précisément au point de bifurcation “situé” à l'infini. De manière théorique, la prédiction d'Euler fait intervenir, pour $\lambda = +\infty$, un système singulier. Le pas, lors de cette prédiction, sera réalisé si la méthode de sous-espaces de Krylov employée résout ce système. Or, en ce point, D_xP admet pour multiplicité $\nu_1 = 1$, ce qui va assurer la résolution du système faisant intervenir cette jacobienne. On obtient, pour différentes valeurs de λ, les courbes suivantes :



Figures 7.a,b

Les courbes ont toutes la même allure et le nombre d'itérations nécessaire à la convergence diminue, ce qui paraît évident puisque le vecteur de départ est initialisé avec des composantes nulles et que la solution approche le vecteur $(0, \lambda)$.
Les courbes des rapports des normes des résidus associés apparaissent ci-dessous :



Figures 8.a,b

Elles sont également de même nature. Les temps de calcul diminuent : on a exposé les résultats obtenus pour des valeurs de λ plus importantes dans le tableau ci-dessous. On utilise le GMRES non redémarré, $f = 0$, $\Delta s = 10^{-1}$.

valeur de λ	valeur de $u_h(.5, .5)$	temps CPU
265.39	0.1146	16.6176
305.53	9.95×10^{-2}	16.6309
345.68	8.80×10^{-2}	15.3762
385.82	7.88×10^{-2}	15.3964
425.96	7.14×10^{-2}	15.4015
466.10	6.53×10^{-2}	14.1327
506.25	6.01×10^{-2}	14.0884
546.39	5.57×10^{-2}	14.1425

Tableau 3

On a calculé le temps mis par le GMRES lors de la prédiction d'Euler avant de converger : ce temps diminue et indique ainsi une facilité à la convergence. Cette diminution est néanmoins très lente. Elle peut peut-être s'expliquer par le fait que l'initialisation devient de plus en plus proche de la solution.

Le problème de Dirichlet apparaît alors seulement comme un problème de transition. Le problème traité ci-dessous fait intervenir cette fois des points de bifurcation pour des valeurs de λ finies.

3. Le problème de Von Karman

On étudie à présent un problème aux valeurs propres non linéaire avec bifurcations à partir de la branche triviale s'écrivant sous la forme

$$\left\{ \begin{array}{l} -\Delta u = \lambda u + u^3 + d \text{ dans } \Omega \\ u = 0 \text{ sur } \partial\Omega \end{array} \right. . \tag{4.38}$$

La fonction $f(u, \lambda) \equiv u^3$ satisfait

$$\begin{array}{l} f(0, \lambda) = 0, \ \forall \lambda \in I\!R \text{ et} \\ f'_u(0, \lambda) = 0, \ \forall \lambda \in I\!R. \end{array} \tag{4.39}$$

Pour $d = 0$, on note que $\{u, \lambda\} \equiv \{0, \lambda\}$ est solution de (4.38) pour tout $\lambda \in I\!R$. Cette branche est la branche triviale et on détermine des branches bifurquant à partir de cette dernière. On utilise la même technique de bifurcation que précédemment. On détermine les solutions de la branche perturbée avec $d = 1$ (différentes valeurs de d ont été considérées et donnent des résultats similaires). Lorsque λ devient grand, on relâche la perturbation pour tomber sur la branche non triviale du problème (4.38) avec $d = 0$. On a le comportement local indiqué sur la figure ci-dessous :

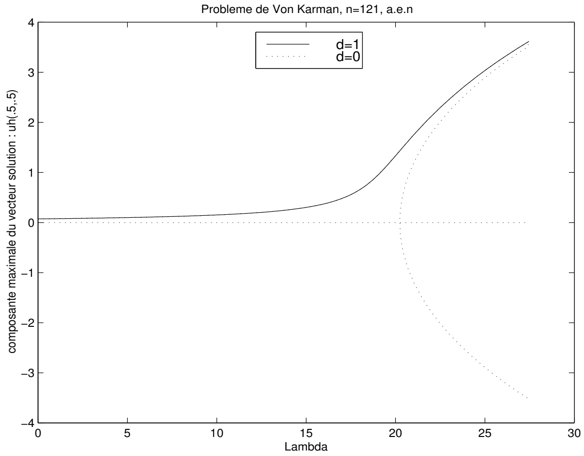
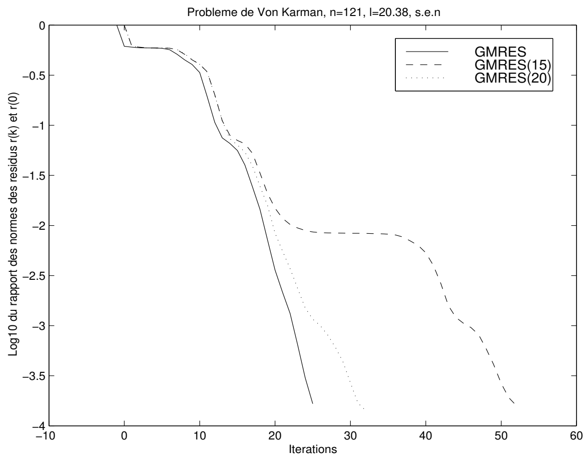
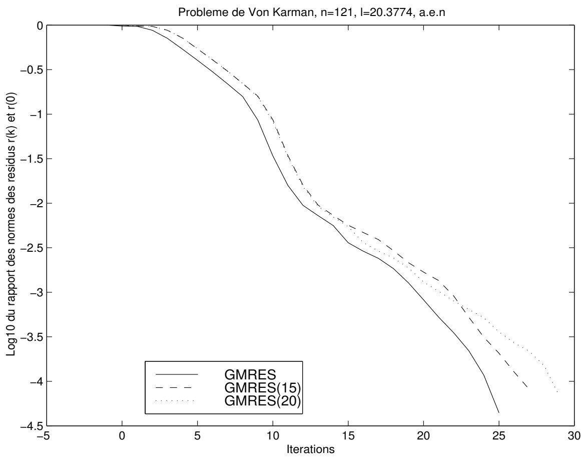
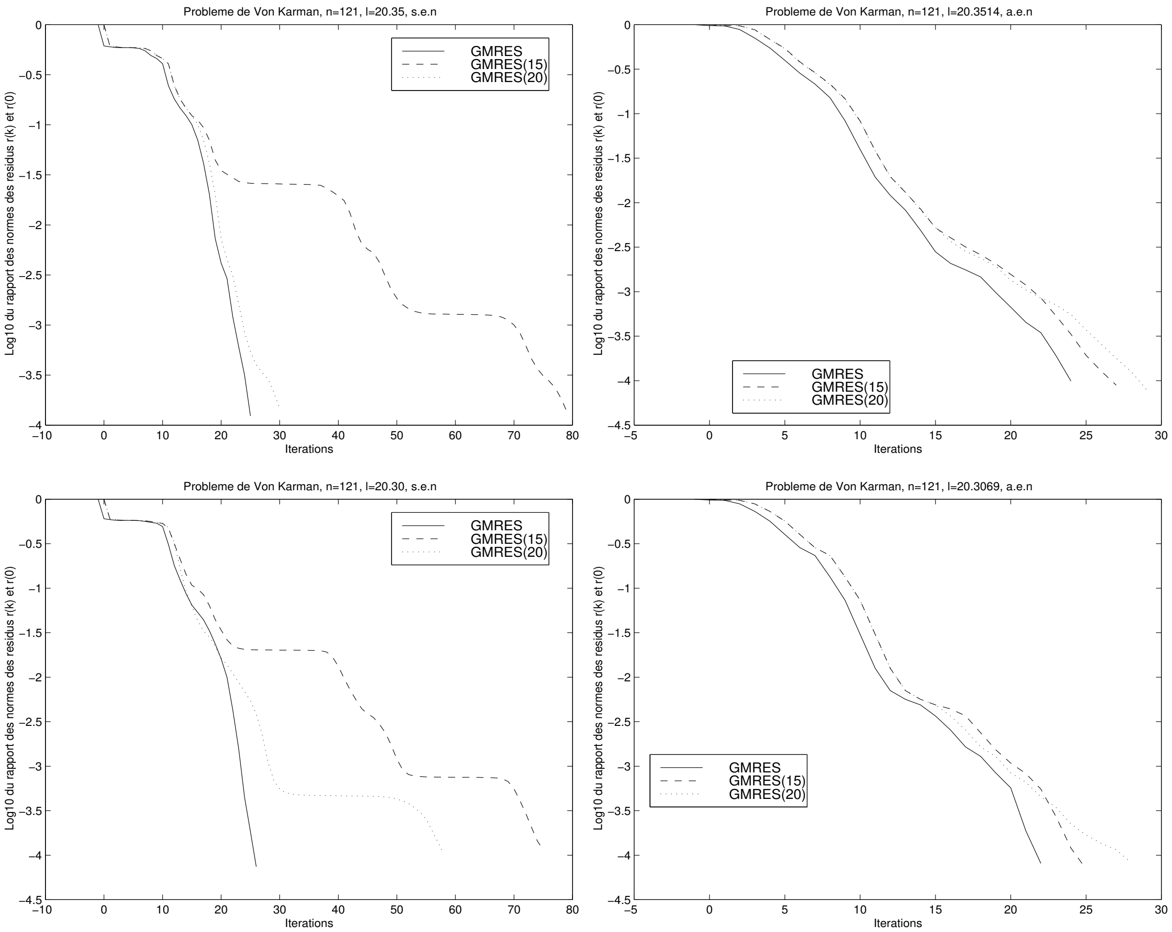


Figure 9 : composante maximale de la solution du problème de Von Karman en fonction de λ.

On situe le point de bifurcation de façon approximative à la position $(0; 20.24)$. L'idée est de voir, dans la suite, si les résultats observés sont du même type que ceux obtenus lors de la résolution du problème de Bratu à l'aide du GMRES et de sa version redémarrée. On approchera avec précaution le point de bifurcation et on observera ce qui s'y passe. On se place sur la branche de solutions non triviale du problème (4.38) avec $d = 0$ et on applique respectivement les méthodes s.e.n puis a.e.n par l'intermédiaire des algorithmes GMRES, GMRES(20) et GMRES(15). On obtient les courbes suivantes :







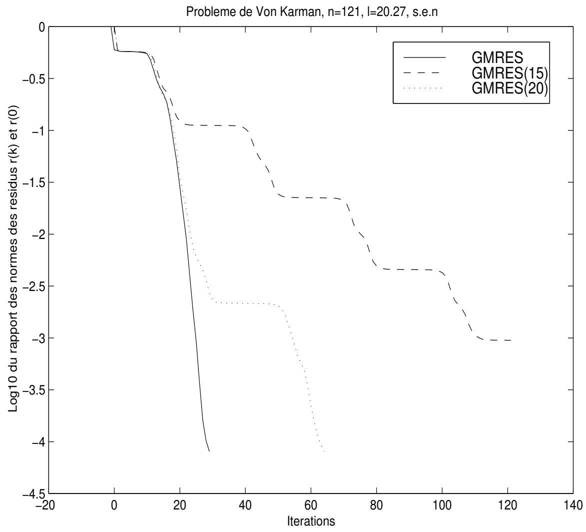
Figures 10.a,b,c,d,e,f

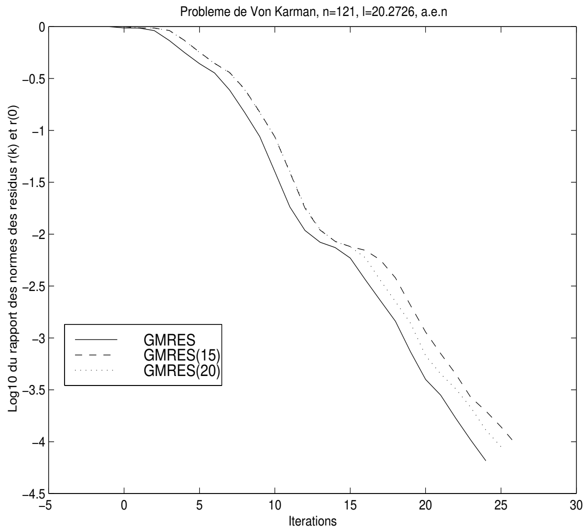
On remarque sur ces premiers graphiques les différents comportements de convergence des deux méthodes : dans le cadre de l'utilisation de la méthode s.e.n, on note que les plateaux de stagnation pour la version redémarrée du GMRES se font de plus en plus présents à l'approche du point critique comme c'était le cas pour le problème de Bratu.
Le nombre d'itérations nécessaire à la convergence du GMRES dans la prédiction d'Euler augmente sensiblement. On doit pouvoir prévoir le comportement de ces courbes lorsque le point de bifurcation est dépassé. A une certaine itération, le pas de la prédiction d'Euler qui sera utilisé va impliquer un changement de branche puisqu'il dépend de λ. On va donc se retrouver sur la branche triviale et il n'y a plus de raison pour que le GMRES redémarré ait du mal à converger.

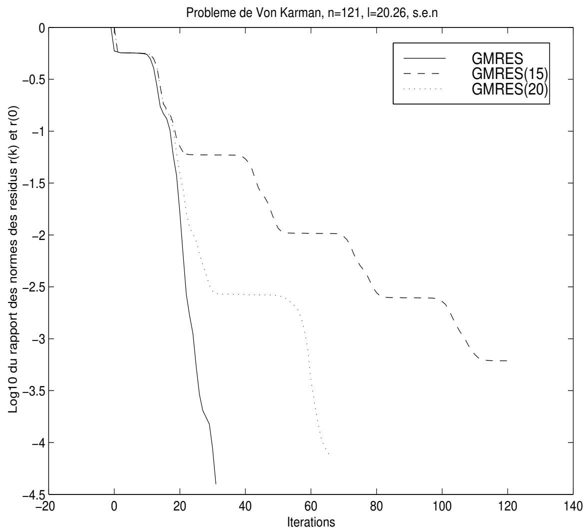
Pour la méthode a.e.n, ce problème de changement involontaire de branche ne se pose pas du fait de la paramétrisation utilisée. En fait, comme le point de bifurcation n'est pas un point limite simple, la jacobienne de (4.11) reste singulière

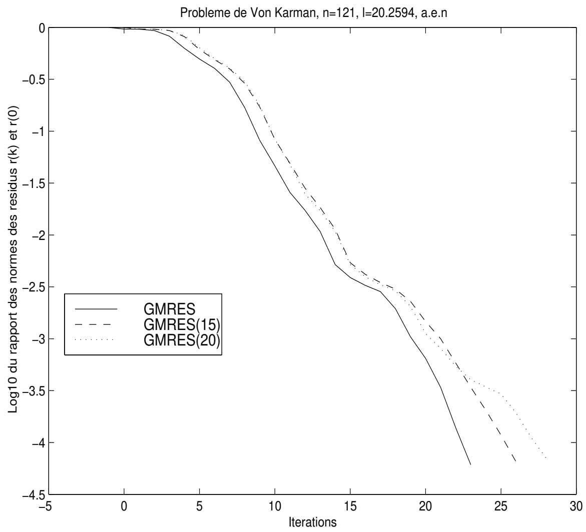
en ce point. La multiplicité de la valeur propre nulle dans le polynôme minimal de $D_u F$ est égale à 1 donc, il y a bien théoriquement et numériquement détermination de la solution qui nous intéresse.
Quand on regarde les trois figures associées à la méthode a.e.n, on ne remarque aucune évolution dans le comportement des courbes. Elles conservent le même profil pour les trois valeurs particulières de λ. Comparé aux courbes obtenues par la méthode s.e.n, le nombre d'itérations nécessaire à la convergence est très inférieur, surtout pour GMRES(15).

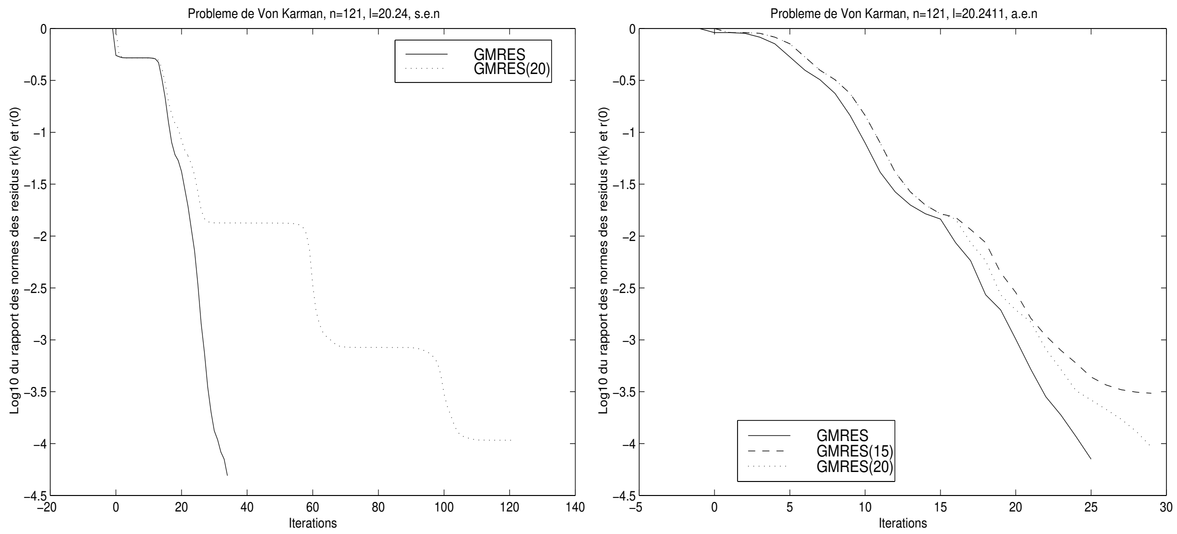
On s'intéresse maintenant à des valeurs de λ communes aux deux méthodes et très proches de la valeur critique $\lambda = 20.24$. On obtient les résultats ci-dessous :











Figures 11.a,b,c,d,e,f

Le nombre d'itérations k nécessaire à la convergence du GMRES dans la méthode s.e.n augmente en fonction de λ. Pour $\lambda = 20.38$, on obtient $k = 25$ et pour $\lambda = 20.24$, on a $k = 35$. On remarque que la convergence de GMRES(15) n'est pas atteinte avant le nombre de cycles maximal, ceci dans la figure 11.e. Il faut également 120 itérations à GMRES(20) pour vérifier le critère de convergence sur la norme résiduelle en $\lambda = 20.24$, alors qu'il n'en fallait que 32 lorsque $\lambda = 20.38$. Il est clair qu'on approche de très près un point difficile à calculer. On remarque néanmoins que la méthode GMRES ne rencontre aucun problème en ce point. On a vu que dans ce cas, $\nu_1 = 1$ et il y aura de manière théorique (et finalement pratique) l'obtention d'une solution de Krylov.

On peut se demander s'il se produit des comportements similaires dans la méthode a.e.n.
On a tracé la norme relative du résidu en fonction du nombre d'itérations pour des valeurs de λ approchant celles données par la méthode sans équation de normalisation. Les courbes obtenues (figures 11.$b.d.f$) continuent à garder un profil comparable, même à l'approche du point de bifurcation. Par contre, rien ne nous permet d'affirmer que l'on se trouve sur un tel point lors de l'utilisation des deux méthodes.

Les tableaux ci-après donnent respectivement le nombre d'itérations nécessaire à la convergence des méthodes utilisées dans la prédiction d'Euler ainsi que le temps mis pour réaliser cette étape :

s.e.n	20.38	20.35	20.30	20.27	20.26	20.24
GMRES	26 136.49	26 136.53	27 142.03	30 157.11	32 166.77	35 182.65
GMRES(20)	32 173.16	30 163.14	58 309.37	64 345.86	66 355.07	120 745.82
GMRES(15)	52 284.10	81 430.63	81 407.13	121 602.04	120 601.05	nombre de cycles atteint

Tableau 4

Il est évident que, pour les trois méthodes, l'approche du point de bifurcation a pour effet de ralentir la convergence. A proximité de ce point, pour $\lambda = 20.24$, la version redémarrée GMRES(15) ne converge pas, le nombre de cycles permis étant atteint. La méthode GMRES reste la méthode la plus performante pour résoudre ce genre de systèmes.
On s'intéresse désormais à des valeurs de λ approchant celles employées lors de l'utilisation de la méthode sans équation de normalisation, pour la méthode a.e.n. On donne les mêmes caractéristiques que précédemment dans le tableau ci-dessous :

a.e.n	20.3774	20.3514	20.3069	20.2726	20.2594	20.2411
GMRES	26 71.35	25 68.67	23 63.73	24 68.76	23 66.25	25 71.06
GMRES(20)	29 81.83	29 81.85	28 79.12	25 71.52	28 79.15	29 81.84
GMRES(15)	27 76.15	27 76.54	25 71.39	26 73.92	26 74.12	29 81.23

Tableau 5

Il y a bien concordance entre le temps CPU et le nombre d'itérations nécessaire à la convergence. La convergence est bien plus rapide que celle de la méthode s.e.n et le nombre d'itérations n'augmente pas. Le temps mis pour observer la convergence lors de la prédiction d'Euler reste sensiblement le même pour les trois versions redémarrées. Par conséquent, la méthode a.e.n est plus performante que la méthode s.e.n pour approcher les points de bifurcation.

Le point de bifurcation étant dépassé, on s'aperçoit de changements notables lors de la convergence dans la méthode s.e.n. On a vu qu'un pas mal choisi pouvait faire changer involontairement de branche de solutions et c'est ce qu'on observe dans la figure ci-dessous :

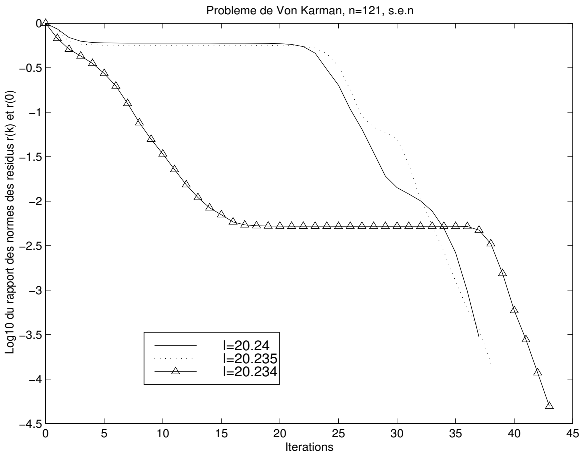


Figure 12

Le comportement de la norme résiduelle pour $\lambda = 20.234$ est radicalement différent de celui obtenu pour $\lambda = 20.34$ ou $\lambda = 20.235$. On sait donc graphiquement qu'on se trouve sur la branche triviale de solutions. Il y a une convergence bien plus rapide au départ, ceci jusqu'à la 15-ième itération. Elle est suivie d'un plateau de stagnation, puis le critère d'arrêt est vérifié. Le nombre d'itérations nécessaire à la convergence est plus important que pour les autres valeurs de λ mais la précision sur le vecteur solution est plus grande. L'évolution des résidus peut être représentée en traçant le rapport des normes résiduelles $\|r_k\|_2/\|r_{k-1}\|_2$ en fonction du nombre d'itérations correspondant à la figure précédente :

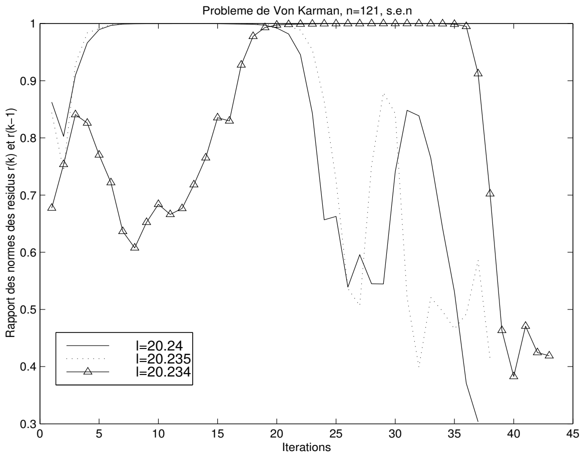


Figure 13

Il est clair d'après les chapitres précédents que la stagnation de la norme résiduelle apparaît lorsqu'une valeur singulière de Ritz approche de très près dans l'algorithme une valeur propre nulle.

Tout ce travail comparatif a pour but de démontrer qu'il est possible de trouver une alternative à la non-détermination d'une solution lorsqu'un système est singulier consistant ou non consistant. L'idée qui découle de cette étude est en fait très simple : pourquoi ne pas travailler avec un système augmenté qui possèderait les qualités requises pour sa résolution, à l'aide des méthodes de sous-espaces de Krylov ? L'inconvénient majeur de cette démarche est qu'elle nécessite un contexte fonctionnel. Ce dont on ne dispose pas lors de la résolution d'un système linéaire quelconque.

4.3 Conclusion

On s'aperçoit dans la pratique que de nombreux systèmes singuliers font leur apparition comme c'est le cas lors du traitement numérique de certaines équations aux dérivées partielles non linéaires. Ces systèmes peuvent être consistants, non consistants, homogènes, et leur résolution semble souvent difficile, voire impossible. La méthode utilisée ici fait appel à des techniques impliquant l'augmentation de la dimension des systèmes rencontrés. Elle permet, alliée à des méthodes de continuation, de passer des points limites et critiques induisant des matrices jacobiennes non-inversibles qui ne sont pas données sous forme explicite. En effet, seule l'action de ces dernières sur des vecteurs est connue. Pourtant, les méthodes de sous-espaces de Krylov et plus particulièrement le GMRES permettent, de manière théorique, la détermination des points de bifurcation puisque $\nu_1 = 1$. Cela est prouvé numériquement en comparant les performances des méthodes de cheminement, avec et sans équation de normalisation, au voisinage de ce type de points. Par contre, il est évident que les méthodes de continuation utilisant l'équation de normalisation permettront le cheminement complet des branches de solutions, ce qui ne sera pas le cas pour la méthode de Keller non modifiée.

Conclusion générale

Il a été montré dans cette thèse que la résolution d'un système singulier consistant à l'aide d'une méthode de sous-espaces de Krylov dépendait essentiellement du choix du vecteur initial dans l'algorithme. Cela a été prouvé dans le cadre de l'arithmétique exacte à l'aide de résultats classiques d'algèbre linéaire et en précision finie grâce à l'étude approfondie de la matrice de Hessenberg créée par la méthode. On a vu également que lorsque la norme utilisée dans la méthode RM était la norme 2, les résultats pouvaient se simplifier et fournir des informations très intéressantes sur les dégénérescences occasionnées par la méthode GMRES.
Des exemples pratiques faisant intervenir des systèmes singuliers ont été introduits dans un dernier chapitre comme c'est le cas pour certaines équations aux dérivées partielles non linéaires dépendant d'un paramètre. Leur résolution passe principalement par des techniques faisant appel à l'augmentation de la taille du système initial. Les résultats numériques obtenus montrent dans un premier temps la réelle efficacité de ces méthodes quant à la détermination complète des branches de solutions. D'autre part, ils permettent de vérifier que les méthodes de sous-espaces de Krylov sont à même de déterminer des points singuliers comme c'est le cas pour les points de bifurcation. En s'inspirant des idées dégagées par ces méthodes, il peut être intéressant d'étudier les conditions sous lesquelles la résolution d'un système admettant une matrice d'indice strictement supérieur à 1 est équivalente à celle d'un système induisant $\nu_1 = 1$.

Suite au travail qui a été effectué dans cette thèse, nous aimerions poursuivre l'étude des systèmes singuliers consistants dans le cadre de nouvelles méthodes. De nombreuses techniques restent disponibles comme c'est le cas des techniques de déflation et de redémarrage implicites. Ces méthodes permettent d'enrichir le sous-espace de Krylov de manière naturelle à l'aide de vecteurs propres associés aux valeurs propres les plus petites en module. Les vecteurs propres étant A-invariants, la construction de la solution implique que le résidu correspondant est orthogonal aux vecteurs propres. Morgan a utilisé cette idée pour introduire la méthode AGMRES (pour Augmented GMRES) [45]. Cette méthode pourrait peut-être fournir des idées quant à la résolution de systèmes dont la matrice est singulière.
Les systèmes singuliers peuvent également intervenir lors de la recherche d'éléments

propres. En effet, on est ramené à l'étude de systèmes du type $Ax = \lambda x$ et connaissant une valeur propre λ ou une approximation de cette dernière, on doit déterminer x tel que $Bx = 0$ avec $B = A - \lambda I$ matrice singulière. Le système est homogène, tous les résultats obtenus dans le cadre des systèmes singuliers consistants sont applicables dans ce cas de figure. La détermination d'un résidu convenable semble plus simple que dans le cadre des systèmes consistants puisque le second membre est nul. Cela laisse entrevoir des méthodes particulières adaptées à la résolution de ce type de systèmes.
On peut également, dans le cadre plus général des systèmes linéaires singuliers consistants, modifier les propriétés des matrices associées à l'aide du préconditionnement. Il est probable, à l'aide de cette technique, que l'on puisse changer la multiplicité de la valeur propre nulle dans le polynôme minimal de la matrice. Ceci nous permettrait ainsi de récupérer les conditions indispensables à la résolution du système considéré.

Bibliographie

[1] E. L. ALLGOWER and K. GEORG. *Numerical continuation methods, an introduction.* Springer Verlag, Berlin (1996).

[2] W. ARNOLDI. *The principle of minimized iterations in the solution of the matrix eigenvalue problem.* Quart. Appl. Math., 9 (1951), pp.17-29.

[3] V. A. BARKER. *Numerical solution of sparse singular systems of equations arising from ergodic Markov chains.* Commun. Stat. Stoch. models, 5 (1989), pp.335-381.

[4] R. BARRET and al. *TEMPLATES for the solution of linear systems : building blocks for iterative methods.*

[5] C. BREZINSKI, M. REDIVO ZAGLIA, H. SADOK. *A breakdown-free Lanczos type algorithm for solving linear systems.* Numer. Math., 63 (1992), pp.29-38.

[6] F. BREZZI, J. RAPPAZ and P. A. RAVIART. *Finite dimensional approximation of nonlinear problems, part I.* Numer. Math., 36 (1980), pp.1-25.

[7] F. BREZZI, J. RAPPAZ and P. A. RAVIART. *Finite dimensional approximation of nonlinear problems, part II.* Numer. Math., 37 (1981), pp.1-28.

[8] F. BREZZI, J. RAPPAZ and P. A. RAVIART. *Finite dimensional approximation of nonlinear problems, part III.* Numer. Math., 38 (1980), pp.1-30.

[9] P. N. BROWN. *A local convergence theory for combined inexact Newton/ finite difference projection methods.* SIAM J. Numer. Anal., 24 (1987), pp.407-434.

[10] P. N. BROWN. *A theoretical comparison of the Arnoldi and GMRES algorithms.* SIAM J. Sci. Stat. Comp., 12 (1992), pp.58-78.

[11] P. N. BROWN and Y. SAAD. *Hybrid Krylov methods for nonlinear systems of equations.* SIAM J. Sci. Stat. Comp., 11 (1990), pp.450-481.

[12] P. N. BROWN and Y.SAAD. *Convergence theory of nonlinear Krylov algorithms.* SIAM J. Optimization, 4 (1994), pp.297-330.

[13] P. N. BROWN and H. F. WALKER. *GMRES on (nearly) singular systems.* SIAM J. Mat. Anal. Appl., 18 (1997) pp.37-51.

[14] A. M. BRUASET. *A survey of preconditionned iterative methods.* Pitman, Research in mathematics series, 328 (1995).

[15] S. L. CAMPBELL, I. C. F. IPSEN, C. T. KELLEY and C. D. MEYER. *GMRES and the minimal polynomial.* BIT, 36 (1996), pp.664-675.

[16] L. CHERFILS. *Méthode de cheminement adaptative pour les problèmes semi-linéaires dépendant d'un paramètre.* PhD thesis, Laboratoire LMC-IMAG, Université Joseph Fourier - Grenoble I (1996).

[17] J. CULLUM. *Arnoldi versus nonsymmetric Lanczos algorithms for solving matrix eigenvalue problems.* BIT, 36 (1996), pp.470-493.

[18] J. CULLUM and A. GREENBAUM. *Relations between Galerkin and norm-minimizing iterative methods for solving linear systems.* SIAM J. Mat. Anal. Appl. Comp., 17 (1996), pp.223-247.

[19] R. S. DEMBO, S. C. EISENSTAT and T. STEIGHAUG. *Inexact Newton methods.* SIAM J. Numer. Anal., 19 (1982), pp.400-408.

[20] J. E. DENNIS JR and R. B. SCHNABEL. *Numerical methods for unconstrained optimization and nonlinear equations.* Prentice-Hall series in computational mathematics (1983), Englewood Cliffs, NJ.

[21] I. S. DUFF, R. G. GRIMES and J. G. LEWIS. *User's guide for the Harwell-Boeing sparse matrix collection.* Technical report TR/PA/92/86, CERFACS (1992).

[22] M. EIERMANN and O. ERNST. *Preliminary version : on some recurrent theorems concerning Krylov subspace methods.* Technical report (1998), Technische Universtät, Bergakademie Freiberg, Freiberg.

[23] H. C. ELMAN. *Iterative methods for large sparse nonsymmetric systems of linear equations.* PhD thesis (1982), Computer Science Dept., Yale univ., New Haven, CT.

[24] R. FLETCHER. *Conjugate gradient methods for indefinite systems in numerical analysis.* G. A. Watson ed. LNM 506, Springer Verlag, Berlin, pp.73-89 (1976).

[25] R. W. FREUND, G. H. GOLUB and N. M. NACHTIGAL. *Iterative solution of linear systems.* Acta Numerica (1991), pp.57-100.

[26] R. W. FREUND, M. H. GUTKNECHT and N. M. NACHTIGAL. *An implementation of the look-ahead Lanczos algorithm for non-hermitian matrices.* part I, technical report 90.45 (1990), RIACS, NASA Ames research Center, Moffet Field, CA.

[27] R. W. FREUND and M. HOCHBRUCK. *On the use of two QMR algorithms for solving singular systems and applications in Markov chain modeling.* Numer. Lin. Alg. Appl., 1 (1994), pp.403-420.

[28] R. W. FREUND and N. M. NACHTIGAL. *QMR : a quasi minimal residual method for non-hermitian linear systems.* Numer. Math., 60 (1991), pp.315-339.

[29] R. W. FREUND and N. M. NACHTIGAL. *An implementation of the QMR method based on coupled two-term recurrences.* SIAM J. Sci. Stat. Comp., 15 (1994), pp.425-448.

[30] R. GLOWINSKI, H. B. KELLER and L. REINHART. *Continuation-conjugate gradient methods for the least squares solution of nonlinear boundary value problems.* SIAM J. Numer. Anal., 6 (1985).

[31] A. GRIEWANK and G. W. REDDIEN. *Characterization and computation of generalized turning points.* SIAM J. Numer. Anal., 21 (1984), pp. 176-185.

[32] I. GUSTAFSSON. *A class of first order factorization methods.* BIT, 18 (1978), pp.142-156.

[33] M. R. HESTENES and E. L. STIEFEL. *Methods of conjugate gradients for solving linear systems.* J. Res. Nat. Bur. Stand., 49 (1952), pp.409-436.

[34] M. HEYOUNI. *Méthode de Hessenberg généralisée et applications.* PhD thesis, Laboratoire d'Analyse Numérique et d'Optimisation, Université des Sciences et Technologies de Lille (1996).

[35] M. HEYOUNI and H. SADOK. *On a variable smoothing procedure for Krylov subspace methods.* Lin. Alg. App., 268 (1998), pp.131-149.

[36] A. S. HOUSEHOLDER and F. L. BAUER. *On certain methods for expanding the characteristic polynomial.* Numer. Math., 1 (1959), pp.29-37.

[37] J. P. KEENER and H. B. KELLER. *Positive solutions of convex nonlinear eigenvalue problem.* J. Diff. Eqs., 16 (1974), pp. 103-125.

[38] H. B. KELLER. *Numerical solution of bifurcation and nonlinear eigenvalue problems.* App. Bif. Theory, P.H. Rabinowitz ed., Academic Press, New-York (1977).

[39] M. KUBIČEK. *Dependance of solution of nonlinear systems on a parameter, algorithm 502.* ACM Trans. Math. software, 2 (1976),pp 98-107.

[40] I. C. F. IPSEN and C. D. MEYER. *The idea behind Krylov methods.* Submitted.

[41] K. C. JEA and D. M. YOUNG. *Generalized conjugate gradient acceleration of nonsymmetrizable iterative methods.* Lin. Alg. Appl., 34 (1980), pp.159-194.

[42] C. LANCZOS. *Solution of systems of linear equations by minimized iterations.* J. Res. Nat. Bur. Stand., 49 (1952), pp.33-53.

[43] B. MATKOWSKY and E. L. REISS. *Singular perturbations of bifurcations.* SIAM J. Appl. Math., 33 (1977), pp 230-255.

[44] J. A. MEIJERNIK, H. A. VAN DER VORST. *An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix.* Math. Comp., 31 (1977), pp.148-162.

[45] R. B. MORGAN. *A restarted GMRES method augmented with eigenvectors.* SIAM J. Mat. Anal. Appl., 4 (1995), pp.1154-1171.

[46] N. M. NACHTIGAL. *A look-ahead variant of the Lanczos algorithm and its application to the quasi minimal residual method for nonhermitian linear systems.* PhD thesis (1991), Massachussets, Institute of technology.

[47] C. C. PAIGE and M. A. SAUNDERS. *Solution of sparse indefinite systems of linear equations unsymmetric matrices.* SIAM J. Numer. Anal., 12 (1975), pp.617-624.

[48] B. N. PARLETT. *A new look at the Lanczos algorithm for solving symmetric systems of linear equations.* Lin. Alg. Appl., 29 (1980), pp.323-346.

[49] B. N. PARLETT, D. R. TAYLOR and Z. A. LIU. *A look-ahead Lanczos algorithm for unsymmetric matrices.* Math. Comp., 44 (1985), pp.105-124.

[50] B. PHILIPPE, Y. SAAD and W. J. STEWART. *Numerical methods in Markov chains modeling.* Stoch. Proc. Appl., 40 (1992), pp.1156-1178.

[51] L. REINHART. *Sur la résolution numérique de problèmes aux limites non linéaires par des méthodes de continuation.* PhD thesis, Université Pierre et Marie Curie, Paris VI (1980).

[52] Y. SAAD. *Variations on Arnoldi's method for computing eigenelements of large unsymmetric matrices.* Lin. Alg. Appl., 34 (1980), pp.269-295.

[53] Y. SAAD. *Krylov subspace methods for solving large unsymmetric linear systems.* Math. Comp., 37 (1981), pp.105-126.

[54] Y. SAAD. *The Lanczos biorthogonalization algorithm and other oblique projection methods for solving large unsymmetric systems.* SIAM J. Numer. Anal., 19 (1982), pp.485-506.

[55] Y. SAAD. *Practical use of some Krylov subspace methods for solving indefinite and nonsymmetric linear systems.* SIAM J. Sci. Stat. Comp., 5 (1984), pp.203-228.

[56] Y. SAAD. *Iterative methods for sparse linear systems.* PWS Publishing Company (1996).

[57] Y. SAAD and M. H. SCHULTZ. *A generalized minimal residual algorithm for solving nonsymmetric linear systems.* SIAM J. Sci. Stat. Comp., 7 (1986), pp.856-869.

[58] H. SADOK. *CMRH : a new method for solving nonsymmetric linear systems based on the Hessenberg reduction algorithm.* Submitted.

[59] H. SADOK. *Analysis of the convergence of the minimal and the orthogonal residual methods.* Technical Report LMPA, Université du Littoral, Laboratoire de Mathématiques Pures et Appliquées, Joseph Liouville (1988).

[60] H. SADOK. *Méthodes de projection pour les systèmes linéaires et non linéaires.* Habilitation (1994).

[61] P. SONNEVELD. *CGS, a fast Lanczos-type solver for nonosymmetric linear systems.* SIAM J. Sci. Stat. Comp., 10 (1989), pp.36-52.

[62] L. SMOCH. *Some results about GMRES in the singular case.* To appear in Numer. Alg.

[63] H. A. VAN DER VORST. *Bi-CGSTAB : a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems.* SIAM J. Sci. Stat. Comp., 2 (1992), pp.631-644.

[64] H. A. VAN DER VORST and C. VUIK. *The superlinear convergence of GMRES.* J. Comp. Appl. Math., 48 (1993), pp.327-341.

[65] P. K. W. VINSOME. *ORTHOMIN : an iterative method for solving sparse sets of simultaneous linear equations.* In Proc. fourth symposium on reservoir simulation, Society of Petroleum engineers of AIME. (1976), pp.149-159.

[66] C. VUIK and H. A. VAN DER VORST. *A comparison of some GMRES-like methods.* Lin. Alg. Appl., 160 (1992), pp.131-162.

[67] H. F. WALKER. *Implementation of the GMRES method using Householder transformations.* SIAM J. Sci. Stat. Comp., 9 (1988), pp.152-163.

[68] R.WEISS. *Orthogonalization methods.* Technical report, 52/94 (1994), Universtät Karlsruhe, Rechenzentrum, Karlsruhe.

[69] J. H. WILKINSON. *The algebraic eigenvalue problem.* Claredon Press (1965), Oxford, England.