RAIRO. RECHERCHE OPÉRATIONNELLE

J. B. LASSERRE

Sur l'inversion de matrice

RAIRO. Recherche opérationnelle, tome 26, nº 2 (1992), p. 177-182

<http://www.numdam.org/item?id=RO_1992__26_2_177_0>

Recherche opérationnelle/Operations Research
(vol. 26, n° 2, 1992, p. 177 à 182)

SUR L'INVERSION DE MATRICE (*)

par J. B. LASSERRE [1]

Résumé. — *On montre que la méthode directe d'inversion de matrice qui consiste à inverser, de façon récursive, des sous-matrices de la matrice initiale de taille croissante est de complexité algorithmique* $o(n^3)$*, donc aussi efficace que les méthodes (de factorisation) de Gauss et Jordan.*

Mots clés : Analyse numérique; algèbre linéaire; inversion de matrice; méthodes directes.

Abstract. — *We show that the direct method which consists of iteratively inverting sub-matrices of increasing size, has same* $o(n^3)$ *computational complexity as Gauss or Jordan methods.*

Keywords : Numerical analysis; linear algebra; matrix inversion; direct methods.

1. INTRODUCTION

On présente une méthode directe d'inversion de matrice (à coefficients réels ou complexes). Alors que les méthodes de Gauss et Jordan factorisent la matrice initiale, on calcule ici l'inverse directement. On inverse itérativement des sous-matrices de dimension croissante $A^{(k)}$ (avec éventuellement des permutations de colonnes). $(A^{(k+1)})^{-1}$ est calculé de façon économique à partir de $(A^{(k)})^{-1}$ en utilisant une propriété élémentaire d'Algèbre Linéaire. La méthode exige $o(n^3)$ multiplications comme pour les méthodes d'élimination de Gauss et de diagonalisation de Jordan [1, 2]. Le nombre de pivots (n) et l'espace mémoire ($o(n^2)$) utilisés sont les mêmes que dans la méthode de Gauss. Cette méthode permet aussi comme la méthode de Gauss de résoudre un système linéaire avec une complexité de $o(n^3/3)$ ($o(n^3/2)$ pour la méthode de Jordan). En outre, la méthode est particulièrement adaptée à l'inversion

(*) Reçu mai 1991.
[1] L.A.A.S.-C.N.R.S., 7, avenue du Colonel-Roche, 31077 Toulouse Cedex, France.

Recherche opérationnelle/Operations Research, 0399-0559/92/02 177 06/$ 2.60

de matrices obtenues à partir d'une matrice initiale, par concaténation de lignes et de colonnes supplémentaires.

2. LA MÉTHODE

Avant de décrire la méthode, nous rappelons une propriété bien connue en Algèbre linéaire. Soit Ω la matrice

A	b
c	d

où A est une matrice carrée régulière, b un vecteur colonne, c un vecteur ligne et d un scalaire. Ω est régulière si et seulement si

$$\delta = -cA^{-1}b + d \neq 0 \tag{1}$$

L'inverse de Ω est alors

$A^{-1} + \frac{1}{\delta} A^{-1} bc A^{-1}$	$-\frac{1}{\delta} A^{-1} b$
$-\frac{1}{\delta} c A^{-1}$	$\frac{1}{\delta}$

2.1. La procédure récursive

On suppose que l'on doit inverser une matrice A de dimension n. Supposons que l'on connaisse l'inverse d'une sous-matrice de A de taille k, du coin gauche supérieur. On peut toujours commencer avec $k=1$ car il y a au moins un coefficient A_{1i}, $i=1, \ldots, n$ different de zéro sinon A est singuliére. On permute alors les colonnes 1 et i si $A_{11}=0$. Appelons $A^{(k)}$ cette sous-matrice de taille k. De plus, appelons $A^j(k)$ [respectivement $A_j(k)$] le vecteur colonne A_{ij}, $i=1, \ldots, k$ (respectivement le vecteur ligne A_{ji}, $i=1, \ldots, k$). On a :

PROPOSITION : *Il existe au moins une colonne j, $j \geqq k+1$, telle que la matrice $A^{(k+1)}$ définie ci-dessous*

$A^{(k)}$	$A^j(k)$
$A_{k+1}(k)$	$A_{k+1\,j}$

est régulière.

Démonstration. — Si elle est singulière, et comme $A^{(k)}$ est régulière, tous les vecteurs $(A^j(k+1), j=k+1, \ldots, n$ sont des combinaisons linéaires des colonnes de la matrice

$$\begin{array}{|c|}\hline A^{(k)} \\ \hline A_{k+1}(k) \\ \hline \end{array}$$

En utilisant ces combinaisons linéaires dans la matrice initiale A, on doit encore trouver une matrice régulière alors que l'on obtient la matrice

$$\begin{array}{|c|c|c|}\hline A^{(k)} & 0 & 0 \\ \hline A_{k+1}(k) & 0 & 0 \\ \hline \begin{array}{c} A_{k+2}(k) \\ \ldots\ldots \\ A_n(k) \end{array} & \begin{array}{c} A_{k+2j} \\ . \\ A_{nj} \end{array} & L \\ \hline \end{array}$$

où L est la sous-matrice $(n-k-1, n-k-1)$ obtenue depuis la sous-matrice correspondante dans A par ces combinaisons linéaires. Cette matrice est singulière, en contradiction avec l'hypothèse. □

Détection de la colonne j : Le problème est donc de trouver une telle colonne j. On examine successivement toutes les colonnes $j=k+1, \ldots. n$. En appliquant la propriété précédente, la colonne j est candidate si et seulement si

$$\delta^j(k) = -A_{k+1}(k)(A^{(k)})^{-1} A^j(k) + A_{k+1j} \neq 0 \tag{2}$$

Donc, une colonne $j(k) \in \{k+1, \ldots, n\}$ tel que (2) est satisfait est sélectionnée et on permute dans A les colonnes $j(k)$ et $k+1$. $\delta^{j(k)}(k)$ est appelé le pivot à l'étape k. L'inverse de $A^{(k+1)}$ est alors

$$\begin{array}{|c|c|}\hline (A^{(k)})^{-1}\left(I + \dfrac{1}{\delta^{j(k)}(k)} A^{j(k)}(k) A_{k+1}(k)(A^{(k)})^{-1}\right) & -\dfrac{1}{\delta^{j(k)}(k)}(A^{(k)})^{-1} A^{j(k)}(k) \\ \hline -\dfrac{1}{\delta^{j(k)}(k)} A_{k+1}(k)(A^{(k)})^{-1} & \dfrac{1}{\delta^{j(k)}(k)} \\ \hline \end{array}$$

On a donc inversé une matrice carrée de dimension $k+1$ à partir d'une sous-matrice carrée de dimension k. Notons de plus que l'on a la relation de récurrence

$$[(A^{(k+1)})^{-1} A^i(k+1)](k) = (A^{(k)})^{-1} A^i(k) - \frac{\delta^i(k)}{\delta^{j(k)}(k)} (A^{(k)})^{-1} A^{j(k)}(k), \quad (3)$$

$$[(A^{(k+1)})^{-1} A^i(k+1)]_{k+1} = \frac{\delta^i(k)}{\delta^{j(k)}(k)}, \qquad i=k+2, \ldots, n \quad (4)$$

Cette relation de récurrence permet d'itérer la procédure avec le minimum de calculs. Quand $k=n$, $(A^{(n)})^{-1}$ est l'inverse de A aux permutations près des lignes qui correspondent aux colonnes éventuellement permutées durant la procédure. Notons en passant que

$$\det(A) = \prod_{k=0}^{n-1} \delta^{j(k)}(k)$$

où $\delta^{j(k)}(k)$ est le pivot à l'itération k and $\delta^{j(0)}(0) = A_{ij(0)}$.

2.2. Complexité algorithmique

Considérons le nombre d'opérations élémentaires, *i.e.*, multiplications et divisions, nécessaires. Supposons que l'on est à la fin de l'étape k et que l'on connaisse :

- la matrice $(A^{(k)})^{-1}$
- les vecteurs $(A^{(k)})^{-1} A^j(k)$, $j=k+1, \ldots, n$

Avec cette notation, le vecteur $A^j(k)$ peut éventuellement être le résultat d'une permutation déjà effectuée. Les différentes opérations de l'étape $k+1$ sont :

1. *Sélection de la nouvelle ligne :* Il y a $n-k-1$ produits scalaires $\delta^j(k+1)$ de dimension $k+1$ à faire, soit $(k+1)(n-k-1)$ multiplications.

La colonne j^* est alors sélectionnée.

2. *Calcul de* $(A^{(k+1)})^{-1}$.

- $(A^{(k)})^{-1} A^{j^*}(k)$ a déjà été calculé à l'étape précédente et donc il suffit de faire k divisions par $\delta^{j^*}(k)$ pour avoir $(A^{(k)})^{-1} A^{j^*}(k)/\delta^{j^*}(k)$.
- $A_{k+1}(A^{(k)})^{-1}/\delta^{j^*}(k)$ exige k^2 multiplications et k divisions.
- $\frac{1}{\delta^{j^*}(k)}(A^{(k)})^{-1} A^{j^*}(k) A_{k+1}(k)(A^{(k)})^{-1}$ exige au plus k^2 multiplications.

donc au total il faut au plus $2k^2$ multiplications et k divisions.

3. *Calcul de* $(A^{(k+1)})^{-1} A^j(k+1)$, $j=k+2, \ldots, n$. En utilisant la relation de récurrence (3)-(4), ce calcul exige au plus $n-k-1$ divisions et $(k+1)(n-k-1)$ multiplications.

Finalement, puisqu'il y a $n-1$ itérations on obtient au total

$$\sum_{k=2}^{n} 2k^2+2(k+1)(n-k-1)=o(n^3)$$

multiplications et n^2 divisions dans le pire des cas.

Espace mémoire requis : A cause du calcul itératif de $(A^{(k)})^{-1} A^j(k)$, on n'a pas besoin des vecteurs $A^j(k)$ et donc les vecteurs $(A^{(k)})^{-1} A^j(k)$ sont stockés à la place des vecteurs $A^j(k)$. L'espace mémoire requis est donc $o(n^2)$ (la matrice initiale est remplacée petit à petit par la matrice inverse). On a donc la même complexité algorithmique que la méthode de Gauss ou Jordan et le même espace mémoire requis.

Remarque : Supposons qu'après avoir inversé A on veuille inverser la matrice

A	B
C	D

où B, C, et D sont des matrices de dimension (n, p), (p, n) et (p, p) respectivement. Alors avec cette méthode, il suffit de calculer les vecteurs $A^{-1} B^j$, j $=1, \ldots, p$, ce qui exige pn^2 multiplications et la procédure peut continuer, comme si on avait directement traité la matrice agrandie. Avec la méthode de Gauss, il faudrait calculer les produits $A^{-1}B$, CA^{-1}, $CA^{-1}B$, puis appliquer l'algorithme de Gauss à la matrice $CA^{-1}B-D$ et faire les produits

$$A^{-1}B(CA^{-1}B-D)^{-1}, \qquad (CA^{-1}B-D)^{-1}CA^{-1}$$

et

$$A^{-1}B(CA^{-1}B-D)^{-1}CA^{-1}$$

pour avoir l'inverse de la matrice augmentée. La complexité algorithmique est encore de $o(n+p)^3$ pour les deux méthodes, mais la première apparaît plus adaptée à ce problème.

2.3. Résolution d'un système linéaire d'équations

Si on ne veut pas inverser la matrice A mais simplement résoudre un système d'équations linéaires $A x = b$, alors dans la procédure précédente :

- on rajoute le vecteur colonne b comme $n+1$-ième colonne de A
- à chaque étape, on supprime les calculs de $A_{k+1}(k)\,(A^{(k)})^{-1}$ et de $(A^{(k)})^{-1}$ puisque l'on ne veut que $(A^{(n)})^{-1} b$ [relation de récurrence (3)-(4)].

Au bout de n itérations on a calculé le vecteur $(A^{(n)})^{-1} b = A^{-1} b$. Dans ce cas, le nombre de multiplications est alors de $\sum_k 2(k+1)(n-k-1) = o(n^3/3)$, comme dans la méthode de Gauss.

BIBLIOGRAPHIE

1. N. Gastinel, Analyse numérique linéaire, *Hermann*, Paris, 1966.
2. G. Jacques, Mathématiques pour l'Informatique : Algorithmique numérique, *Armand Colin*, Paris, 1971.
3. J. B. Lasserre, On matrix inversion, LAAS-report, octobre 1990.