ACTA UNIVERSITATIS LODZIENSIS
FOLIA OECONOMICA 48, 1985

Józef Białas*, Władysław Milo**, Zbigniew Wasilewski***

ON THE NUMERICAL PROPERTIES OF THE M-P GENERALIZED INVERSE ALGORITHMS

1. Introduction

A large number of algorithms for the calculation of a generalized inverse has been suggested in recent years. In this paper we shall confine our attention to the algorithms of the Moore-Penrose generalized inverse (the M-P generalized inverse). These algorithms in turn can be divided into two classes: non-iterative and iterative algorithms.

In formulating non-iterative algorithms for the calculation of the M-P generalized inverse A^+ of the real matrix A with m rows and n columns, i.e. $A \in R^{mxn}$ (where A^+ is such a matrix that $A^+AA^+ = A^+$; $AA^+A = A$; $(AA^+)' = AA^+$; $(A^+A)' = A^+A$) the factorization principle is used. This principle consists in the factorization of given matrix A with $\operatorname{rank}(A) = k_o$ into the product of two (or more), i.e.

$$(1.1)\quad A = DF,\ D \in R^{mxk_o},\ F \in R^{k_o xn},\ \operatorname{rank}(D) = \operatorname{rank}(F) = k_o.$$

Due to the properties of the M-P generalized inverse and the definitions of D, F we have

---

*Dr., Lecturer at the Institute of Mathematics, University of Łódź.
**Dr., Lecturer at the Institute of Econometrics and Statistics, University of Łódź.
***Senior Assistant at the Institute of Econometrics and Statistics, University of Łódź.

(1.2) $$A^+ = F^+D^+,$$

(1.3) $$F^+ = F'(FF')^{-1},\quad D^+ = (D'D)^{-1}D',$$

Three types of factorizations are very well known. The first type has the form

(f1) $$A = LU,$$

where $L \in R^{mxk_o}$ is a lower trapezoidal matrix with units on the diagonal and zeros above the diagonal, $U \in R^{k_o xn}$ is the upper trapezoidal. Therefore, due to (1.2), (1.3) it is

(1.4) $$A^+ = U^+L^+ = U'(UU')^{-1}(L'L)^{-1}L',$$

The second type has the form

(f2) $$A = QS,$$

$Q \in R^{mxk_o}$, $Q'Q = I_{(k_o)}$, S is the upper triangular, and

(1.5) $$A^+ = S'(SS')^{-1}Q' = S^{-1}Q'$$

(the decomposition (f2) can be done by the use of Housholder trasformations or the modified Gram-Schmidt procedure).

The third type has the form

(f3) $$A = U\Lambda V',$$

$U \in R^{mxk_o}$, $V = R^{nxk_o}$, $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_{k_o}) \in R^{k_o xk_o}$, $U'U = V'V = I_{(k_o)}$, is the diagonal matrix of nonzero square roots of eigen values of $A'A$ (or AA'), and

(1.6) $$A^+ = V\Lambda^{-1}U.$$

The factorization (f1) is used, among others, in calculating the M-P generalized inverse, when the Gauss elimination method with complete pivoting is used (see the description of the algorithms AWMEL and GEINW).

The factorization (f2) is used, among others, in the construction of the algorithm of L u e c k e [3].

The factorization (f3) is used, among others, in the formulation of the algorithm SVDZ described in this paper.

Iterative algorithms (to which the studied algorithm ABI belongs) are based on the idea of iterative monotonic two-sided approximations of Schultz.

The existence of the great number of the algorithms for calculation of the Moore-Penrose generalized inverse of the matrix implies the need of the comparative study of their properties. In the paper the authors give a description of four such algorithms and try to compare them by use of the Monte-Carlo experiments. The measures of algorithm precision, used in the experiments, were based on four conditions which define the Moore-Penrose generalized inverse. They are of the form:

a) $$\frac{\| A\hat{A}_1^+A - A \|}{\| A \|}$$

b) $$\frac{\| \hat{A}_1^+A\hat{A}_1^+ - \hat{A}_1^+ \|}{\| \hat{A}_1^+ \|}$$

c) $$\frac{\| (A\hat{A}_1^+)' - A\hat{A}_1^+ \|}{\| A\hat{A}_1^+ \|}$$

d) $$\frac{\| (\hat{A}_1^+A)' - \hat{A}_1^+A \|}{\| \hat{A}_1^+A \|}$$

where $\hat{A}_1^+$ denotes, given by the i-th algorithm, estimate of the M-P generalized inverse of the matrix A ($i = 1, \ldots, 4$), and $\|\cdot\|$ denotes Euclidean norm.

Besides this, we try to answer the question how much the changes in ill conditioning of the inverted matrix influence the form of the generalized inverse of this matrix, realized by means of the i-th algorithm. This has been done by means of a comparison of the Euclidean norms of the M-P generalized inverses $\hat{A}_i^+$, $i = 1, \ldots, 4$ for various degrees of ill-conditioning of the inverted matrix A.

2. The Description of the Algorithms

We confined our analysis to one iterative algorithm, marked here as ABI, and three non-iterative AWMEL, GEINW, SVDZ based on the idea of the factorization of the given matrix A of rank k_o. Algorithms AWMEL and GEINW are based on the factorization of the form:

$$A = LU, \quad A \in R^{mxn}, \quad \operatorname{rank}(A) = k_o \leq \min(m, n),$$

where $L \in R^{mxk_o}$ is lower trapezoidal matrix with units in main diagonal and zeros above it, $U \in R^{k_o xn}$ is upper trapezoidal matrix and $A^+ = U^+L^+ = U^*(UU^*)^{-1}(L^*L)^{-1}L^*$.

Algorithm SVDZ is based on the idea of the spectral factorization of the form

$$A = U\Lambda V^*, \quad U \in R^{mxk_o}, \quad V \in R^{nxk_o}, \quad \Lambda \in R^{k_o xk_o}, \quad V^*V = U^*U = I_{k_o}$$

where Λ denotes the diagonal matrix of nonzero square roots of the singular values of the matrix A^*A, and $A^+ = V\Lambda^{-1}U^*$.

Below we give a concise description of the successive steps of the Moore-Penrose inverse computation by means of each from the analysed algorithms.

2.1. Algorithm ABI (cf. B e n-I s r a e l [2])

S1: Using one of the algorithms for dealing with eigenvalue problem (e.g. the method of Jacobi) find the greatest eigenvalue of the matrix A^*A and denote it by λ^2_{max}.

S2: Find such a value α that

$$0 < \alpha < 2/\lambda^2_{max}.$$

S3: Form $X_0 = \alpha A^*$.

S4: For $j = 1, 2, \ldots$ compute

$$X_j := X_{j-1}(2I - AX_{j-1}),$$

until

$$\frac{\|x_j - x_{j-1}\|}{\|x_{j-1}\|} < \eta,$$

where η denotes the rank of the accuracy of the iterative process. If the convergence criterion is satisfied for $j = j_z$ then put $A^+ := x_{jz}$.

2.2. Algorithm AVMEL (cf. W i l l n e r [9])

S1: Using the row elementary transformations reduce the inverted matrix $A \in R^{mxn}$ to the normal Hermitean form, i.e. find the matrix $H = \begin{pmatrix} G \\ 0 \end{pmatrix}$, $H \in R^{mxn}$, $G \in R^{k_o xn}$, $k_o = \text{rank}(A) = \text{rank}(H)$, where:

. first k_o rows of matrix H are nonzero rows and elements of the rows $k_o + 1, \ldots, n$ are all equal zero,

.. the first nonzero element of the i-th row of the matrix H ($i = 1, \ldots, k_o$) is equal 1 and belongs to column n_i, $n_1 < n_2 < \ldots < n_{k_o}$,

... the only nonzero element in column n_i of the matrix H is 1 in the i-th row.

S2: Using the matrices A and H from matrix $P \in R^{mxk_o}$ according to the scheme

$$P^{(i)} := A^{(j)} \Longleftrightarrow H^{(j)} = 1^{(i)} \qquad \begin{cases} j = 1, \ldots, n, \\ i = 1, \ldots, k_o. \end{cases}$$

where $P^{(i)}$, $A^{(j)}$, $H^{(j)}$ denote respectively the i-th and j-th columns of the matrices P, A and H; $1^{(i)}$ denotes the unit vector with i-th component equal 1.

S3: Compute $A^+ = G^*(P^*AG^*)^{-1}P^*$.

2.3. Algorithm GEINW (cf. W a r m u s [7])

S1: Using row and column elementary transformations reduce the inverted matrix $A \in R^{m \times n}$ of rank $k_o \leq \min(m, n)$ to the matrix of the form

$$\begin{pmatrix} I_{k_o} & \vdots & 0 \\ \cdots & \vdots & \cdots \\ 0 & \vdots & 0 \end{pmatrix}$$

making respective row transformations on the matrix $G := I_m$ and column transformations on the matrix $F := I_n$.

S2: Using transformed matrices G and F, obtained in S1, compute

$$G^{-1} = \left(\Gamma_{m,n} \vdots \Delta_{m,m-n}\right); \quad V = (\Gamma'\Gamma)^{-1}\Gamma'\Delta,$$

$$F^{-1} = \begin{pmatrix} \varphi_{m,n} \\ \cdots \\ \psi_{n-m,n} \end{pmatrix}, \quad U = \Psi\varphi'(\varphi\varphi')^{-1},$$

$$W = UV$$

and form the matrix

$$Y = \begin{pmatrix} I_{k_o} & \vdots & V_{k_o,m-k_o} \\ \cdots & \vdots & \cdots \\ U_{n-k_o,k_o} & \vdots & W_{n-k_o,m-k_o} \end{pmatrix}.$$

S3: Compute $A^+ = FYG$.

2.4. Algorithm SVDZ (cf. G o l u b, R e i n s c h [8] M i ł o, W a s i l e w s k i [5])

S1: Using one of the algorithms for dealing with eigenvalue problem (e.g. the method of Jacobi) find the eigenvalues λ_i^2 and respective eigenvectors v_i of the matrix A^*A of rank k_o.

S2: Order eigenvalues, obtained in S1, in a decreasing manner, i.e. $\lambda^2_{r_1}, \ldots, \lambda^2_{r_{k_o}}$ and form matrices

$$\Lambda^2 = \mathrm{diag}\left(\lambda^2_{r_1}, \ldots, \lambda^2_{r_{k_o}}\right),$$

$$V = \left(v_{r_1}, \ldots, v_{r_{k_o}}\right).$$

S3: Compute $A^+ = V\Lambda^{-2}V^*A^*$.

3. Numerical Analysis of the Algorithms' Properties

The comparisons of the properties of the four algorithms described above, were confined to only one specific numerical structure of the inverted matrices. The analysis comprises the symmetric and rectangular matrices, and for each type of the matrices there were distinguished various degrees of ill-conditioning and collinearity. Each matrix A was generated in two stages. At first there was generated nonsingular tridiagonals basis matrix with units on diagonals and zeros elsewhere (denoted further as T). In the next stage the matrix T was enlarged. Ill-conditioning was introduced through symmetric enlargement of the matrix T by columns and rows differing from respective columns and rows of the matrix T only in main-diagonal element by near zero value ϵ. Singularity, in the case of symmetric matrices, was introduced through enlargement of the matrix T by columns and rows being the linear combinations of the successive two columns and rows of this matrix. Rectangular matrices were obtained by adding to matrix T its columns with increased by the constant value β elements of the last row, and by adding columns being linear combination of the successive two columns of the matrix T. Such a generation of the matrix A will fix in an arbitrary way the number of linearily dependent columns (the degree of the singularity of the matrix A) as well as the number of columns being the carrier of ill-conditioning and its magnitude. At the same time the general structure of the inverted matrices still remains the same. Describing the results of the experiments we will use the following parametric note characterizing the structure of the inverted matrix A

$$A(m,n,k,l,\varepsilon,c_1,c_2,\beta),$$

where:

m,n,k - respectively the dimensions and rank of the matrix A,

l,ε - respectively the number of columns which are the carrier of ill-conditioning and parameter defining the degree of this ill-conditioning,

c_1,c_2 - coefficients of the linear combination of the basis columns of A,

β - constant value used in forming rectangular matrices in the way described in the text above.

At first we considered the matrix of the form

$$A_1 = A(11, 11, 8, 0, 0, 0, 1, 0)$$

i.e. a singular but well-conditioned symmetrical matrix. For such defined matrix A all analysed algorithms characterized the same precision but iterative algorithm ABI appeared the most time-consuming and algorithm AWMEL the fastest of them. Considering matrices $A_2 = A(11, 11, 8, 0, 0, 0.0001, 1)$ and $A_3 = A(11, 11, 8, 0, 0, 0.01, 1)$ we stated rather small changes in the values of $\|A_2^+\|$ and $\|A_3^+\|$ in comparison with the value of $\|A_1^+\|$ which indicates the stable performance of the algorithms with regard to unsignificant changes in the values of elements in nonbasis columns of the inverted matrix A. The next group of analysed matrices consisted of the singular symmetrical matrices which characterized the various degree of ill-conditioning, i.e. matrices of the form

$$A(11, 11, 8, l, \varepsilon, 2, 1, 0) = A_{l,\varepsilon},$$

where: $l = 1,2,3$ and $\varepsilon = 0.01, 0.001, 0.0001$.

For such a defined group of inverted matrices we observed substantial differences between respective algorithms in dependence on the degree of ill-conditioning of these matrices. In the case of small level of ill-conditioning ($\varepsilon \leq 0.01$), despite the number of columns being its carrier ($l = 1,2,3$), the results of computations for all algorithms were, for each value of l, very similar ($\|A^+\|$ computed by means of respective algorithms differs only in the fourth place after point). The precision measures, we

had accepted, indicate a bit less precision of the computations which were done by SVDZ. Just in the case of $\varepsilon = 0.001$ and $\varepsilon = 0.0001$ computations done by means of SVDZ led to quite different form of $\|\hat{A}^+\|$ in comparison with these obtained by means of algorithms AWMEL, ABI and GEINW (cf. Table 1). The value of $\|\hat{A}^+_{SVDZ}\|$ declined from 164,4545 to 2.92117 when the value of the ill-conditioning parameter ε declined from 0.01 to 0.001 and remained stable despite further decline in ε. Yet, the values of $\|\hat{A}^+_{ABI}\|$, $\|\hat{A}^+_{AWMEL}\|$ and $\|\hat{A}^+_{GEINW}\|$ increased all the time proportionally to the decline in ε.

T a b l e 1

The influence of ill-conditioning on the value of $\|\hat{A}^+\|$ obtained by means of algorithms AWMEL, ABI, SVDZ and GEINW

Algorithm	The value of coefficient ε			
	0.1	0.01	0.001	0.0001
	$l = 1$			
AWMEL2	14.93864965	147.42738999	1 475.11192473	14 406.37441319
ABI	14.93865005	147.42794870	1 475.96259826	14 761.67377477
ASVDZ2	14.93865165	147.42873272	3.31496181	3.31490505
GEINW	14.93865005	147.42794849	1 475.96259019	14 761.67157197
	$l = 3$			
AWMEL2	26.71292776	267.83145887	2 679.08125581	27 992.02097376
ABI	26.71292755	267.83201591	2 680.03565124	26 802.17216366
ASVDZ2	26.71292739	267.83226888	1.82713458	1.82718446
GEINW	26.71292756	267.83201613	2 680.03567281	26 802.17405431

Despite such substantial differences between the value of $\|\hat{A}^+_{SVDZ}\|$ and the values of $\|\hat{A}^+_{ABI}\|$, $\|\hat{A}^+_{AWMEL}\|$, $\|\hat{A}_{GEINW}\|$, accepted measures of the computations' precision indicate a similar and quite good precision of all the algorithms (cf. Table 2).

The comparison of values of the precision measures for various values of ε indicate a small decline in precision when the value of ε declines. The most significant decline appeared in the case of the algorithm AWMEL. On the other hand, the best

Table 2

The influence of ill-conditioning of the inverted matrix A on the computational precision of the respective algorithms

Algorithm	Precision measures			
	$\frac{\|AA^+A-A\|}{\|A\|}$	$\frac{\|A^+AA^+-A^+\|}{\|A^+\|}$	$\frac{\|(AA^+)^T-AA^+\|}{\|AA^+\|}$	$\frac{\|(A^+A)^T-A^+A\|}{\|A^+A\|}$
	$\varepsilon = 0.1$			
AWMEL2	0.00000001	0.00000003	0.00000001	0.00000006
ABI	0.00000000	0.00000001	0.00000000	0.00000000
ASVDZ2	0.00000001	0.00000011	0.00000000	0.00000011
GEINW	0.00000000	0.00000000	0.00000000	0.00000000
	$\varepsilon = 0.01$			
AWMEL2	0.00000634	0.00000379	0.00000102	0.00001191
ABI	0.00000000	0.00000013	0.00000000	0.00000030
ASVDZ2	0.00000174	0.00000532	0.00000073	0.00001077
GEINW	0.00000000	0.00000000	0.00000001	0.00000000
	$\varepsilon = 0.001$			
AWMEL2	0.00027984	0.00057636	0.00013251	0.00045671
ABI	0.00000002	0.00000118	0.00000004	0.00000712
ASVDZ2	0.00004995	0.00000000	0.00000000	0.00000000
GEINW	0.00000005	0.00000001	0.00000004	0.00000004
	$\varepsilon = 0.0001$			
AWMEL2	0.01744044	0.02406894	0.00361627	0.04149807
ABI	0.00000012	0.00000987	0.00000027	0.00257619
ASVDZ2	0.00000499	0.00000000	0.00000000	0.00000000
GEINW	0.00000032	0.00000014	0.00000050	0.00000028

precision characterized algorithm GEINW. Algorithm ABI was, as compared with others, nearly 6-8 times more time-consuming, but its precision was rather high.

The last group of inverted matrices consists of rectangular matrices with different degree of ill-conditioning. We considered nine matrices of the form

$$A(6,11,4+l,1,\varepsilon,2,1,1) = A_{l,\varepsilon}$$

where: $l = 1,2,3$; $\varepsilon = 0.01,\ 0.001,\ 0.0001$.

In the case when only one column was the carrier of ill-conditioning, all the algorithms were characterized by high precision and all of them led to M-P generalized inverses of similar norms. In the case when two or three columns were the carriers of ill-conditioning, the situation was similar to that concerning symmetrical matrices. The highest precision was characteristic of algorithm GEINW and the worst one of algorithm AWMEL.

4. Final Remarks

Our investigations allow us, despite their very limited scope, to formulate some remarks concerning the analysed algorithms for computation of M-P inverses.

1° It is supposed that from the numerical point of view, measures of algorithm's precision based on four Moore-Penrose conditions which were accepted in the paper do not allow to indicate uniquely the best numerical approximation of the generalized M-P inverse of the ill-conditioned matrix.

2° All algorithms which were analysed except SVDZ were characterized by small robustness on high ill-conditioning of the inverted matrix.

3° From the computations' precision point of view algorithm GEINW was the best and algorithm AWMEL was the worst, but at the same time the fastest among the four algorithms being analysed.

4° The ill-conditioning of high degree caused a decline in the algorithms' precision, except in the case of SVDZ.

5° From Table 1,2 it is seen that the SVDZ is stable both in the sense of $\| \hat{A}_i^+ \|$ as well as in the sense of the degree of fulfilment of the four conditions defining the M-P generalized inverse A^+ when the ill-conditioning is increasing (that is when ε is declining from $\varepsilon = 0.001$ to $\varepsilon = 0.0001$). It is caused, probably, by the inherent properties of Jacobi procedure and the

assumption that $U_1 = AV_1\Lambda^{-1}$, where $U = (U_1 \vdots U_2)$, $U'U = UU' = I_{(m)}$, $U_1'U_1 = I_{(k_0)}$; $V = (V_1 \vdots V_2)$, $V'V = VV' = I_{(n)}$, $V_1'V_1 = I_{(k_0)}$. For drawing decisive conclusions about instability and its causes we are carrying out more structural experiments.

BIBLIOGRAPHY

[1] A l b e r t A. (1972), Regression and the Moore-Penrose Pseudoinverse, N.Y., Academic Press.

[2] B e n-I s r a e l A. (1966), A Note on an Iterative Method for Generalized Inversion of Matrices, Math. of Comput., p. 439-440.

[3] L u e c k e G. (1979), A Numerical Procedure for Computing the Moore-Penrose Inverse, Numer. Math. 32, p. 129-137.

[4] M a r c u s M., M i n c H. (1964), A Survey of Matrix Theory and Matrix Inequalities, Boston, Allyn and Bacon Inc.

[5] M i l o W., W a s i l e w s k i Z. (1980), Generalized Inverse Algorithms and Their Applications, Part I, Grant R. III.9.5.7, unpublished manuscript.

[6] M i t r a S., R a o C. (1971), Generalized Inverse of Matrices and Its Applications, N.Y., Wiley.

[7] W a r m u s M. (1972), Uogólnione odwrotności macierzy, Warszawa, PWN.

[8] W i l k i n s o n J., R e i n s c h J. (1971), Handbook for Automatic Computation. Vol. III: Linear Algebra, Berlin, Springer-Verlag.

[9] W i l l n e r L. (1967), An Elimination Method for Computing the Generalized Inverse, Math. Comput., p. 227-229.

Józef Białas, Władysław Milo, Zbigniew Wasilewski

O WŁASNOŚCIACH NUMERYCZNYCH ALGORYTMÓW MOORE'A-PENROSE'A UOGÓLNIONYCH ODWROTNOŚCI MACIERZY

Praca zawiera krótki opis trzech nieiteracyjnych i jednego iteracyjnego algorytmu obliczania uogólnionych odwrotności danej macierzy A^+ oraz rezultatów eksperymentów numerycznych zmierzających do ustalenia niektórych numerycznych własności tych algorytmów. Stwierdzono, iż

a) wszystkie analizowane algorytmy, oprócz bazującego na zmodyfikowanej dekompozycji wartości własnej, wykazywały małą odporność na złe uwarunkowanie odwracanej macierzy;

b) najszybszy algorytm Willnera był najmniej precyzyjny;

c) algorytm GEINV Warmusa był najbardziej precyzyjny pod względem spełnienia czterech warunków określających uogólnioną odwrotność macierzy A^+;

d) w przypadku wzrastającego złego uwarunkowania najbardziej stabilny okazał się (w sensie λ_i^+ i spełniania definicji A^+) algorytm oparty na zmodyfikowanej dekompozycji wartości własnej.