A TABLE METHOD AND COMPUTER REALIZATION OF SOLVING THE LARGEST G-INVERSE OF FUZZY MATRIX

Yi Chong-Xin
Dept. of Basis. Qiqihar College of Light Industry
Qiqihar, CHINA

Wang Hong-Xu
Dept. of Basis. Liaoyang College of Petrochemistry
Liaoyang, CHINA

ABSTRACT

The paper gives a table method and computer realization of solving the largest of all the g-inverse of fuzzy matrix on the basis of the definition in the document [1].

For high order fuzzy matrix, the paper introduces a processing method using files system by which this algolrithm realized in micro-computer, too.

1 INTRODUCTION

K.H.Kim and F.W.Roush have put forward the concept of the generalized inverse of the fuzzy matrix in the document [1]. Luo Ching-Zhong has given the definitoon method and the decision condition of solving the largest of all the g-inverse of the fuzzy matrix. The definition method is very complicated. Hence the hawd calculation is very dificult in patricular as the order of the matrix is very large.

This paper is aimed at this weak point of the definition method and gives a table method of solving the largest of all the g-inverse of the fuzzy matrix.

2 THE CONCEPTS, THEOREMS AND SOLVING PROCESS BY THE DEFINITION

For the convenince of understanding the algolrithm by the readers, the paper gives out the relative propositions (the concepts and the theorems) without the proof.

Definition 1 [1]. For any given fuzzy matrix $A_{n\times m}$, if there is at least a fuzzy matrix $G = (g_{ij})_{n\times m}$ that make the relation

$$AGA = A$$

holds trul, then the matrix G is called an generalized inverse matrix of fuzzy matrix A or simply g-inverse of the A.

Definition 2 [2]. For any given fuzzy matrix $A = (a_{ij})_{n\times m}$ Suppose there are g-inverse of the A and we state all the inverse of the A as $\mathcal{A}$. If there is a matrix $G_0 \in \mathcal{A}$, for any matrix $G \in \mathcal{A}$, the relation

$$G \subseteq G_o$$

holds true, then the fuzzy matrix G_o is called the largest g-invesre of the matrix A.

Definition 3 [2]. For any given fuzzy matrix $A = (a_{ij})_{n \times m}$, let

$$\bar{x}_{jk} = \wedge \{ a_{st} \mid a_{st} < (a_{sj} \wedge a_{kt}) \} \quad j=1,2,\ \ldots\ ,m;\ k=1,2,\ \ldots\ ,n.$$

and specify the infimum of the null set is equal to 1, then all the $\bar{x}_{jk}$ compose a fuzzy matrix $\bar{X} = (\bar{x}_{jk})_{m \times n}$, too.

Definition 4. For any given fuzzy matrix $A = (a_{ij})_{n \times m}$; we call the matrix $\bar{X} = (\bar{x}_{jk})_{m \times n}$ defined in the definition 3 a fuzzy matrix connected with the A or simply f-matrix of the A.

On the basis of the definitions above, we can esily prove that the following theorems are all correct.

Theorem 1 [1]. If the fuzzy matrix A has g-inverse, then
(i) the A has the largest g-inverse necessarily;
(ii) the largtst inverse of the A is the only one;
(iii) the f-matrix of the A given in the definition 3 is namely the largest inverse of the A.

Theorem 2 [2]. The f-matrix $\bar{X} = (\bar{x}_{jk})_{m \times n}$ of the A given by the definition 3 is the largest g-inverse of the fuzzy matrix A if and only if the ralation

$$A\bar{X}A = A$$

holds true.

On the basis of the proposition above, when any fuzzy matrix is given out, we may first solve for its f-matrix $\bar{X} = (\bar{x}_{jk})_{m \times n}$ given by the definition 3, and then prove the relation

$$A\bar{X}A = A$$

if hold true. When the formula is satisfied, the $\bar{X}$ is exictly the largest inverse of the A or else the fuzzy matrix A has not any g-inverse.

3 THE TABLE ALGOLRITHM AND COMPUTER REALIZATION

We analysis the definition 3 now. Suppose $A = (a_{ij})_{n \times m}$ is any fuzzy matrix. On the basis of the definition 3:

$$\bar{x}_{jk} = \wedge \{ a_{st} \mid a_{st} < (a_{sj} \wedge a_{kt}) \} \quad j=1,2,\ \ldots\ ,m;\ k=1,2,\ \ldots\ ,n.$$

We deply its into all the terms and have formula

$$\begin{aligned} \bar{x}_{jk} = \wedge \{ a_{st} \mid\ & (a_{1j} \wedge a_{k1}) > a_{11},\ \ldots\ ,(a_{1j} \wedge a_{km}) > a_{1m}, \\ & (a_{2j} \wedge a_{k1}) > a_{21},\ \ldots\ ,(a_{2j} \wedge a_{km}) > a_{2m}, \\ & \ldots\ldots\ldots \\ & (a_{nj} \wedge a_{k1}) > a_{n1},\ \ldots\ ,(a_{nj} \wedge a_{km}) > a_{nm} \}. \end{aligned}$$

From this we may construct a table as shown by the table 1 consisting from the matrix A and jth column and the kth row of the A. And we treat the table by the deferent way. Thus we have

table 1

	a_{k1}	a_{k2}	...	a_{km}
a_{1j}	a_{11}	a_{12}	...	a_{1m}
a_{2j}	a_{21}	a_{22}	...	a_{2m}
...	...	...	...	...
a_{nj}	a_{n1}	...	...	a_{nm}

(i) Reconstructing sets—general table algolrithm.

Step 1. To reconstruct set B.

We reconstruct set B by the content of the table 1. The elements of the set are taken out in way: We drow respectively a horizontal line to left (the 0th column) and a vertical line up(the 0th row) from every element a_{il}($i=1, 2, \dots ,n$; $l=1,2, \dots ,m$) of the matrix A constructing set A in the table 1. And we compare a_{il} with the corresponding element a_{ij} in the 0th column and a_{kl} in the 0th row respectively. We put a_{il} into B if a_{ij} and a_{kl} both are greater than the a_{il} or else then put the null value Φ into the set B.

Step 2. To solve for infimum.

We solve for the infimum of the set B reconstructed from the content of the table 1 in the above way and record it by $\bar{x}_{jk}$ (the infimun is equal to 1 if elements of the set B is all null value Φ).

Step 3. To construct the f-matrix.

In this way , after treating all the table 1 consisted from all the j columns ($j=1,2, \dots ,m$) and the k rows ($k=1,2, \dots ,n$) throughout the matrix A ,we construct row by row the matrix $\bar{X}$ with all the $\bar{x}_{jk}$ abtained above. The matrix $\bar{X}_{m\times n}$ is namely f-matrix of the matrix $A_{n\times m}$.

The procedure that we reconstruct the set B with reference to the jth column and the k row to solve for $\bar{x}_{jk}$ is shown by the program flowchart 1.

See figure 1.

(ii) To mark out the rows and the columns--simplied table algolrithm.

Step 1. To mark out the columns.

We compare respectively all the elements (except for the 0th) a_{il}($i=1,2, \dots , n$) in the every column (except for Oth column, i.e. l=1,2, ... ,m) with the Oth element a_{kl} of the one and mark out the elements not less than a_{kl} in the table 1. I.e. so-called marking-on the columns.

Step 2. Marking-on the rows.

We compare respectively all the elements (except for Oth) a_{il}(l=1,2, ... ,m) of the every row (except for the Oth row, i.e. i=1,2, ... ,n) with the Oth elements a_{ij} of the one and mark out the elements not less than a_{ij} in the table 1. I.e. so-called marking-on the rows.

Step 3. Solving for the infimum.

We solve for the infimum of the set B obtained marking out the rows and the columns. And we record it by $\bar{x}_{jk}$ (the infimum is the equal to 1 if the set B is null set).

Step 4. Constructing the f-matrix.

We take all the j column(j=1,2, ... ,m) and k row (k=1, 2, ... , n) throughout the A and are going to obtain $n \times m$ same tables with the table 1 on the structure (as shown in the table 2). After recpectively treating them in the same way, we construct row by row matrix $\bar{X}$ with all of the $\bar{x}_{jk}$ obtained above. The matrix $\bar{X}_{m \times n}$ is exactly f-matrix of the matrix $A_{n \times m}$.

table 2

	a_{11} ... a_{1m}	a_{21} ... a_{2m}	......	a_{n1} ... a_{nm}
a_{11} ⋮ a_{n1}	$\left\{\begin{matrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{matrix}\right\}$	$\left\{\begin{matrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{matrix}\right\}$	......	$\left\{\begin{matrix} a_{11} & \cdots & a_{nm} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{matrix}\right\}$
...	...	...	...	...
a_{1m} ⋮ a_{nm}	$\left\{\begin{matrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{matrix}\right\}$	$\left\{\begin{matrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{matrix}\right\}$	......	$\left\{\begin{matrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nm} \end{matrix}\right\}$

The procedure marking out the rows and the columns of the matrix $B_{n \times m}$ with referance to the jth column and the kth row is shown by the diagram 2.

See figure 2.

From mentioned algolrithm and diagrams above, to solve for the largest g-inverse of the fuzzy matrix A or prove none of any g-inverse for one, we give following all the program flowchart (shown in the figure 3).

4 THE PROCESSING METHOD OF THE LARGE FUZZY MATRIX.

(i) Analysis of required internal space.

From diagram one or two , we can know that a fuzzy matrix $A_{n \times m}$ stored source data, a fuzzy matrix $B_{n \times m}$ participating in the intermediate calculation and a fuzzy matrix $\bar{X}_{m \times n}$ stored the results is needed at least as solving the matrix $\bar{X}$ with the computer.

After solving for the matrix $\bar{X}_{m \times n}$, to judge if

$$A\bar{X}A = A$$

holds true, a intermediate fuzzy matrix $C_{n \times n}$ stored the result of the $A\bar{X}$ will be required yet.

Let
$$p = \vee\{m,n\}$$
then allthe numbers of the required internal memory space is $3mn + n^2$. Therefore the space complexity of the algolrithm is $o(p^2)$. For this reason, in the case m and n are all very large, it is in fact very dificult that the algolrithm is realized with microcomputer having limited memory capacity in particular.

(ii) Adopted maasures.

In view of the analysis above, in order to realize the processing for large fuzzy matrix, we shold adopted following measures:

. To establish random files so as to store a large party of the data matries.

We shall store source data matrix $A_{n \times m}$, result matrix $\bar{X}_{m \times n}$, intermediate result matrix $C_{n \times n}$ in desk by means of the random files. Once matrix elements are required, the computer call its one by one (or by record (row or column)) in internal memory from the desk. And twinkling result data will be returned to the desk in time.

In this way, there is only a matrix $B_{n \times m}$ participating in the intermediate calculation at all time in internal memory . Hence the internal memory space required for the calculation will be decreased greatly.

. To translate the real number into an integer one for the convenince of the store and calculation.

As we know, the occupied memory space (4 byte) by a real number is four times as many as one (1 byte) by a integer one in BASIC language of the micro computer IBM/PC. Hence we first time multiply source data matrix $A_{n \times m}$ by a proper number m' to make all the real number in the A integer one as to store and calculate. Before the computer needs to output calculating result, the integer must divided by m' in order to recover and disply the result. Then the memory space has been decresed further. And we can easily realize processing for large fuzzy matrix with micro computer.

5 THE EXAMPLES.

Example 1. Let

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0.2 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

(i) Solving process by definition 3.

From the definition 3 (and 4), we have following formulas

$$\bar{x}_{11} = \wedge \{ (a_{21} \wedge a_{12}) > a_{22},\ (a_{21} \wedge a_{13}) > a_{23},\ (a_{31} \wedge a_{12}) > a_{32},$$
$$(a_{31} \wedge a_{13}) > a_{33} \} = \wedge \{ \Phi, \Phi, \Phi, \Phi \} = 1$$

$$\bar{x}_{12} = \wedge \{ (a_{11} \wedge a_{22}) > a_{12},\ (a_{11} \wedge a_{23}) > a_{13},\ (a_{31} \wedge a_{22}) > a_{32},$$
$$(a_{31} \wedge a_{23}) > a_{33} \} = \wedge \{ 0, 0, \Phi, \Phi \} = 0$$

$$\bar{x}_{13} = \wedge \{ (a_{11} \wedge a_{32}) > a_{12},\ (a_{11} \wedge a_{33}) > a_{13},\ (a_{21} \wedge a_{32}) > a_{22},$$
$$(a_{21} \wedge a_{33}) > a_{23} \} = \wedge \{ 0, 0, \Phi, \Phi, \} = 0$$

$$\bar{x}_{21} = \wedge \{ (a_{22} \wedge a_{11}) > a_{21},\ (a_{22} \wedge a_{13}) > a_{23}, (a_{32} \wedge a_{11}) > a_{31},$$
$$(a_{32} \wedge a_{13}) > a_{33} \} = \wedge \{ 0.2, \Phi, 0, \Phi \} = 0$$

$$\bar{x}_{22} = \wedge \{ (a_{12} \wedge a_{21}) > a_{11},\ (a_{12} \wedge a_{23}) > a_{13},\ (a_{32} \wedge a_{21}) > a_{31},$$
$$(a_{32} \wedge a_{23}) > a_{33} \} = \wedge \{ \Phi, \Phi, 0, \Phi \} = 0$$

$$\bar{x}_{23} = \wedge \{ (a_{12} \wedge a_{31}) > a_{11},\ (a_{12} \wedge a_{33}) > a_{13},\ (a_{21} \wedge a_{31}) > a_{21},$$
$$x \quad (a_{22} \wedge a_{33}) > a_{23} \} = \wedge \{ \Phi, \Phi, \Phi, \Phi \} = 1$$

$$\bar{x}_{31} = \wedge \{ (a_{23} \wedge a_{11}) > a_{21},\ (a_{23} \wedge a_{12}) > a_{22},\ (a_{33} \wedge a_{11}) > a_{31},$$
$$(a_{33} \wedge a_{12}) > a_{32} \} = \wedge \{ 0.2, \Phi, 0, \Phi \} = 0$$

$$\bar{x}_{32} = \wedge \{ (a_{13} \wedge a_{21}) > a_{11},\ (a_{13} \wedge a_{22}) > a_{12},\ (a_{33} \wedge a_{21}) > a_{31},$$
$$(a_{33} \wedge a_{22}) > a_{32} \} = \wedge \{ \Phi, \Phi, 0, \Phi \} = 0$$

$$\bar{x}_{33} = \wedge \{ (a_{13} \wedge a_{31}) > a_{11},\ (a_{13} \wedge a_{32}) > a_{12},\ (a_{23} \wedge a_{31}) > a_{21},$$
$$(a_{23} \wedge a_{32}) > a_{22} \} = \wedge \{ \Phi, \Phi, \Phi, \Phi \} = 1$$

Then, we found out f-matrix of the matrix A, i.e.

$$\bar{X} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

Whereas

$$A \bar{X} A = \begin{pmatrix} 1 & 0 & 0 \\ 0.2 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0.2 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0.2 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0.2 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0.2 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} = A$$

therefore the matrix $\bar{X}$ is the largest g-inverse of the matrix A.

(ii) Solving process by marking-on the rows and the columns.

Firstry, we table (as shown in the table 3) by the form shown in the table (or table 2), for all the rows and the columns of the matrix A. And then mark out all the rows and columns in the table 3.

table 3

	1 0 0	0.2 1 1	0 1 1
1 0.2 0	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{11}$	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{12}$	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{13}$
0 1 1	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{21}$	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{22}$	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{23}$
0 1 1	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{31}$	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{32}$	$\left\{\begin{matrix}1&0&0\\0.2&1&1\\0&1&1\end{matrix}\right\}_{33}$

The sloping lines "/ " and "\" are shown respectively marking-on the rows and making-on the columns in the table 3. After marking-on the columns and the rows, we solve for the infimum for the set shown by every braces $\{\quad\}_{jk}$ in order to find out $\bar{x}_{jk}$. Finally, we got

$$\bar{X} = \begin{pmatrix}1&0&0\\0&0&1\\0&0&1\end{pmatrix}$$

all the same.

(iii) Solving process with the computer.

First , we make the program LEI run on the computer that can realize the table algolrithm solving for the largest g-inverse of the fuzzy matrix. And next answer the computer's hint and input data into the computer by one's request, then may get following disply and result direct.

```
ENTER THE FUZZY MATRIX A:
          1    0  0
        0.2    1  1
          0    1  1
  F-MATRIX OF THE A IS X:
```

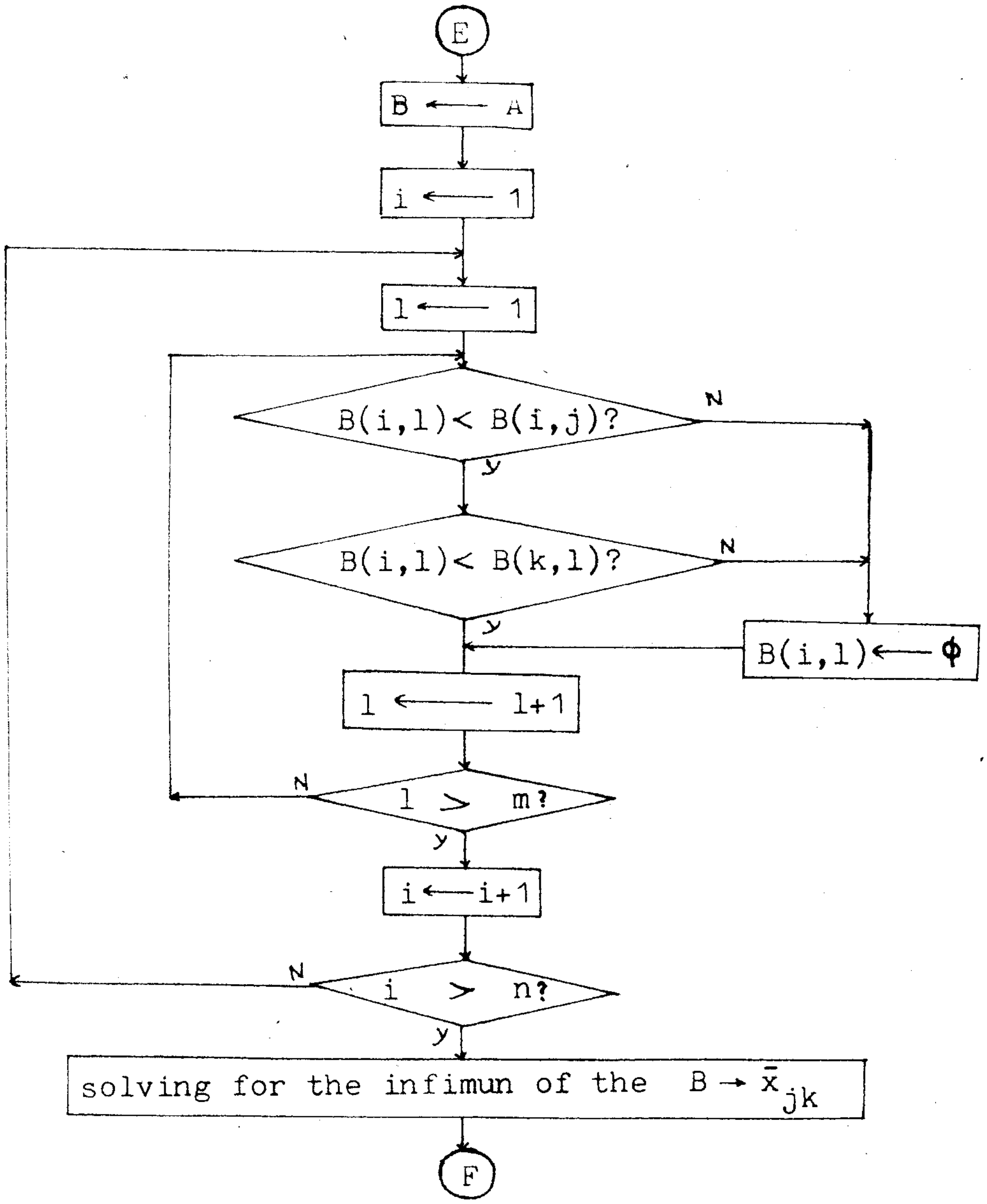

Fig.1. Flowchart showing the procedure of reconstructing set B and solving for the $\bar{x}_{jk}$

```
              1   0   0
              0   0   1
              0   0   1
LARGEST G-INVERSE OF THE A IS X.
DO YOU CONTINUE ? (Y/N)  y/↵
ENTER THE FUZZY MATRIX A:
```

```
   1    0    1    0
 0.2    1  0.1    1
   0    1  0.3    0
   0    0    1  0.1
THERE IS NOT ANY G-INVERES FOR A.
DO YOU CONTINUE ?(Y/N)  N /↵
END.
```

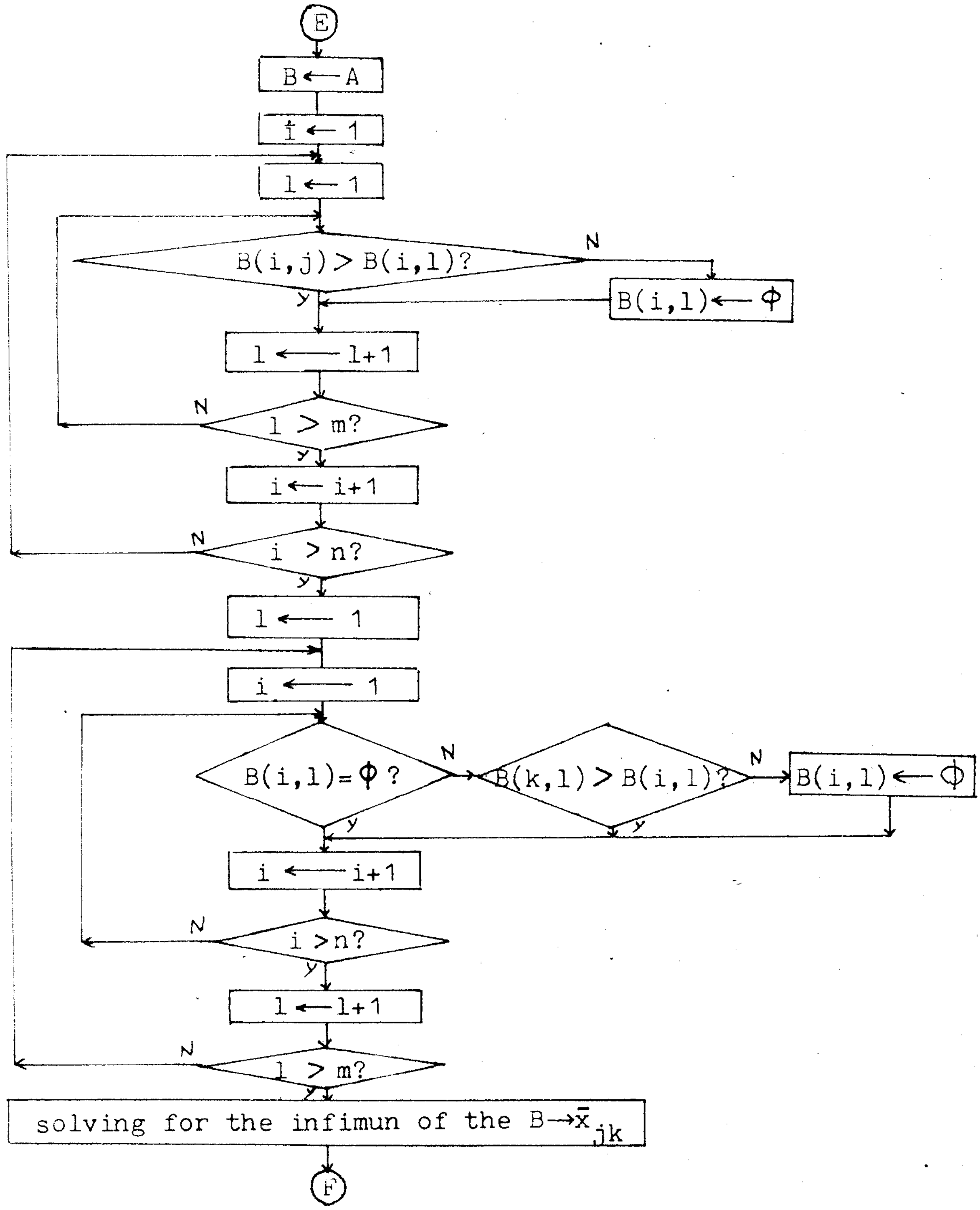

Fig.2. Marking out the rows and the columns of the B with reference to the jth column and kth row. And solving for the $\bar{x}_{jk}$.

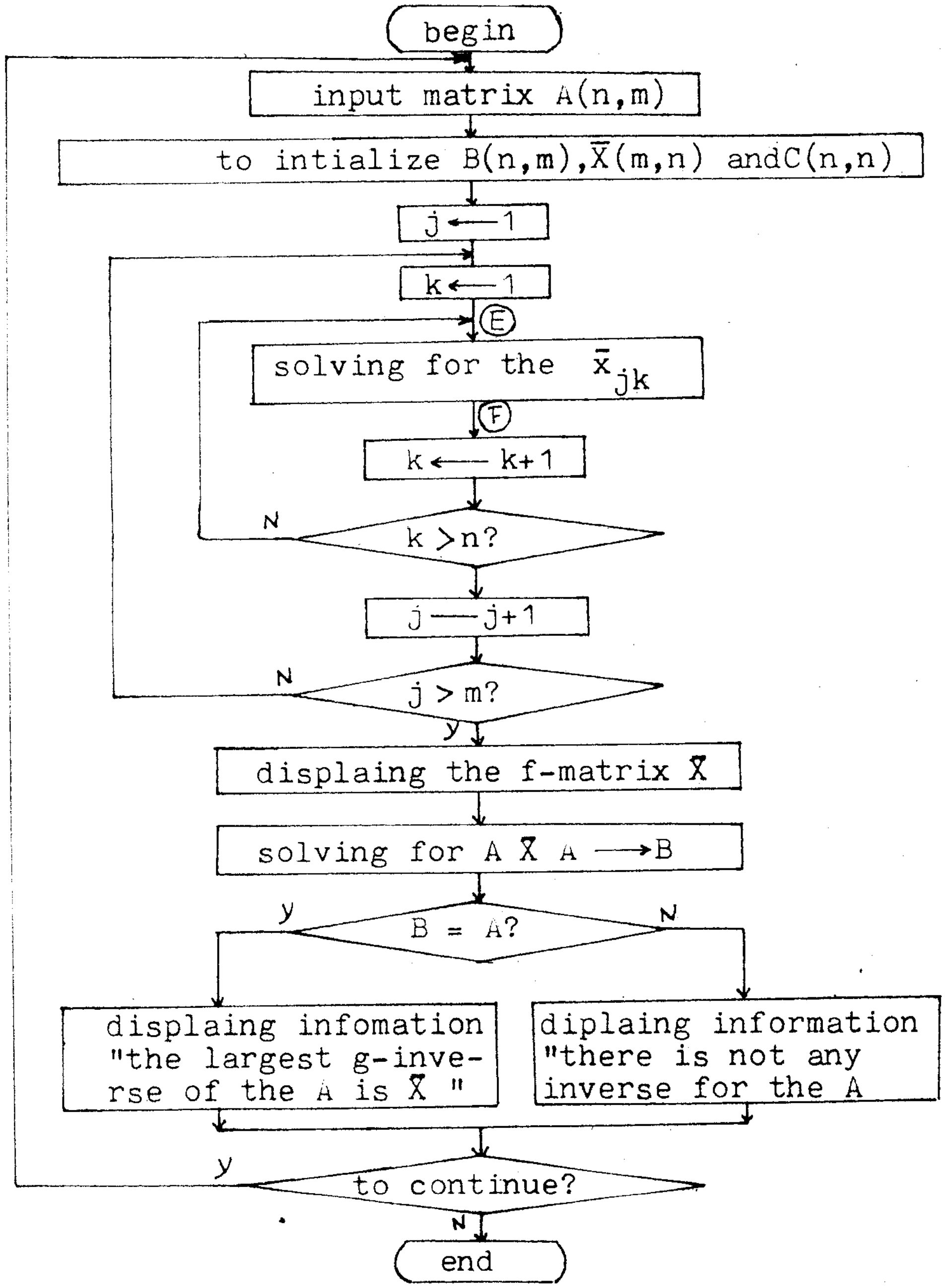

Fig.3. General flowohart of solving the largest g-inverse of the fuzzy matrix $A_{n \times m}$.

Note: The parts drawn under line and the matrix A are all inputed by the operator.

REFERENCES

[1] Kim.K.H & Roush.F.W., Generalized fuzzy matrix, Fuzzy Sets And System, 4(1980), p293-315

[2] Luo Cheng-Zhong, Solving process of generalized fuzzy inverse matrix, Fuzzy Mathematics, 1(1981), p31-38