Journal of Numerical Linear Algebra with Applications Vol. 1, No. 1, (1992) 43–47

AN ALGORITHM FOR FINDING A SINGULAR MATRIX IN AN INTERVAL MATRIX

JIRI ROHN

Faculty of Mathematics and Physics, Charles University,
Malostranské nám. 25, 11800 Prague,
Czechoslovakia

Received June 1991

We present a finite algorithm for finding a singular matrix in a given interval matrix. We show that if started properly, it yields a singular matrix of a special form. The algorithm is however not general since it may fail.

Keywords: Interval matrix, singular matrix, finite algorithm.

The problem we are concerned with here is the following. Given an interval matrix $A^I=[\underline{A},\bar{A}]=\{A;\underline{A}\le A\le \bar{A}\}$ where $\underline{A}$, $\bar{A}$ are real $n\times n$ matrices with $\underline{A}\le\bar{A}$, find a singular matrix $A\in A^I$ provided it exists. This problem is by no means trivial since it was proved in [2] that the decision problem

Instance: An interval matrix A^I with rational entries,

Question: Does A^I contain a singular matrix?

is NP-complete (i.e. very unlikely to be solvable in a polynomial number of operations; cf. Garey and Johnson [1] for a survey of NP-completeness). In our paper [3, p. 69] we proposed an algorithm for solving the above decision problem. The algorithm presented in this paper is an elaborated version of the previous one, being improved in three respects: (1) it may be started from an arbitrary $A\in A^I$, (2) it no longer requires matrix inversion at each iteration since an updating procedure has been built in, and (3) if successful, it not only states the existence of a singular matrix, but also gives an explicit singular matrix of a special form. However, it may still fail, i.e. unable to continue even if a singular matrix in A^I exists; nevertheless, our computational experience is rather encouraging. The algorithm keeps and updates three $n\times n$ matrices A, B and C, of which it will later be proved that $C=A^{-1}$, and runs as follows (for the sake of brevity, we denote by A_{kj}, $A_{k.}$, $A_{.k}$ the kjth coefficient, the kth row and the kth column of a matrix A, respectively):

Algorithm

Step 0. Select an $A \in A^I$.
Step 1. If A is singular, terminate.
Step 2. Otherwise set $C := A^{-1}$.
Step 3. Define B by

$$B_{ij} = \underline{A}_{ij} \text{ if } C_{ji} \geqslant 0$$

$$B_{ij} = \overline{A}_{ij} \text{ if } C_{ji} < 0 \quad (i,j = 1, \ldots, n).$$

Step 4. Compute

$$\beta = \min_i (BC)_{ii} = (BC)_{kk}.$$

Step 5. If $\beta = 1$, terminate. The algorithm fails: a singular matrix has not been found.

Step 6. If $\beta \in (0, 1)$, set

$$C := C - \frac{1}{\beta} C_{.k} (B - A)_{k.} C$$

$$A_{k.} := B_{k.}$$

and go to Step 3.

Step 7. If $\beta \leq 0$, determine

$$m = \min \left\{ i;\ \sum_{j=1}^{i} B_{kj} C_{jk} + \sum_{j=i+1}^{n} A_{kj} C_{jk} \leq 0 \right\},$$

set

$$A_{kj} := B_{kj} (j = 1, \ldots, m-1)$$

$$A_{km} := -\left(\sum_{j=1}^{m-1} B_{kj} C_{jk} + \sum_{j=m+1}^{n} A_{kj} C_{jk} \right) / C_{mk}$$

and terminate: A is a singular matrix in A^I.

Before justifying the algorithm, we shall show that $C = A^{-1}$ always holds in Steps 3 and 4. This is obviously so during the first pass of the algorithm through these steps. Thus, assume by induction $C = A^{-1}$ to hold and let C', A' be the updated matrices at the end of Step 6. With e_k being the kth column of the unit matrix, we have

$$A' = A + e_k(B - A)_{k.}$$

$$C' = A^{-1} - \frac{1}{\beta} A^{-1} e_k (B - A)_{k.} A^{-1} = (A + e_k(B - A)_{k.})^{-1} = (A')^{-1}$$

in view of the Sherman–Morrison formula for inverting a rank one update since $\beta = (BA^{-1})_{kk} = 1 + (B - A)_{k.} A^{-1} e_k$; thus $C' = (A')^{-1}$. Also $A \in A^I$ implies $A' \in A^I$.

Now we prove that β always satisfies $\beta \leq 1$, so that Steps 5 to 7 exhaust all the possibilities. In fact, from the construction of the matrix B in Step 3, for each i,j we have $B_{ij} \leq A_{ij}$ if $A_{ji}^{-1} \geq 0$ and $B_{ij} \geq A_{ij}$ if $A_{ji}^{-1} < 0$, in both the cases $B_{ij} A_{ji}^{-1} \leq A_{ij} A_{ji}^{-1}$, hence $(BC)_{ii} = \sum_j B_{ij} A_{ji}^{-1} \leq \sum_j A_{ij} A_{ji}^{-1} = 1$ for each i, implying $\beta \leq 1$.

Next we shall show that the matrix A'', obtained at the end of Step 7 by updating a matrix A, is singular. First, since $\sum_{j=1}^{n} B_{kj} C_{jk} = \beta \leq 0$, we see that m is well-defined and satisfies

$$\sum_{j=1}^{m-1} B_{kj} C_{jk} + A_{km} C_{mk} + \sum_{j=m+1}^{n} A_{kj} C_{jk} > 0$$

$$\sum_{j=1}^{m-1} B_{kj} C_{jk} + B_{km} C_{mk} + \sum_{j=m+1}^{n} A_{kj} C_{jk} \leq 0,$$

hence $C_{mk} \neq 0$ and for the number

$$\alpha = -\left(\sum_{j=1}^{m-1} B_{kj} C_{jk} + \sum_{j=m+1}^{n} A_{kj} C_{jk} \right) / C_{mk},$$

we obtain by substituting into the previous inequalities that $(A_{km} - \alpha) C_{mk} > 0$ and $(B_{km} - \alpha) C_{mk} \leq 0$ hold, hence α belongs to the interval with endpoints A_{km} and B_{km} which shows that $\alpha \in [\underline{A}_{km}, \overline{A}_{km}]$. Now it can be easily seen that the matrix A'' is formed from A by

$$A'' = A + e_k(b - A_{k.}),$$

where $b = (B_{k1}, \ldots, B_{k,m-1}, \alpha, A_{k,m+1}, \ldots, A_{kn})$, hence $A'' \in A^I$ and since $bC_{.k} = 0$ in view of the definition of α, we have (E is the unit matrix)

$$\det A'' = \det(E + e_k(b - A_{k.})C) \det A = (1 + (b - A_{k.}) C_{.k}) \det A$$
$$= (bC_{.k}) \det A = 0,$$

hence A'' is singular.

It remains to be demonstrated that the algorithm is finite.

Theorem 1. For an arbitrary starting matrix $A^0 \in A^I$ in Step 0, the algorithm after a finite number of iterations either yields a singular matrix in A^I, or fails.

Proof. Since we already know that the matrix constructed in Step 7 is singular, we are left to prove that the algorithm cannot cycle infinitely in the loop between Step 3 and Step 6.

First, we show that the set of A's appearing during the algorithm is finite since each such a matrix satisfies

$$A_{kj} \in \{A^0_{kj}, \underline{A}_{kj}, \bar{A}_{kj}\}$$

$(k,j=1,\ldots,n)$. In fact this holds obviously at the outset; as soon as k has been chosen for the first time in Step 4, the updating in Step 6 enforces $A_{kj} \in \{\underline{A}_{kj}, \bar{A}_{kj}\}$ to hold and this property is being maintained in the subsequent steps.

Second, we show that no A can reappear in the course of the algorithm since the sequence of $|\det A|$ is strictly decreasing. Let A and A' be a current matrix and its update, respectively. Then, we have $A' = A + e_k(B-A)_{k.} = (E + e_k(B-A)_{k.}A^{-1})A$ for the respective k and B. Since $E + e_k(B-A)_{k.}A^{-1}$ differs only in the kth row from the unit matrix E, we have

$$\det(E + e_k(B-A)_{k.}A^{-1}) = 1 + ((B-A)A^{-1})_{kk} = (BA^{-1})_{kk} = \beta,$$

hence $\det A' = \beta \det A$ and since $\beta \in (0,1)$ in Step 6, we conclude that $|\det A'| < |\det A|$.

The two properties stated imply the finiteness of the algorithm. □

The formula $\det A' = \beta \det A$ justifies the choice of k in Step 4 aimed at incuring the steepest descent of the value of $|\det A|$.

In [3] we proved (Theorem 5.1, assertion (C7)) that if A^I contains a singular matrix at all, then it also contains a singular matrix of a special form

$$\begin{aligned} &A_{ij} \in \{\underline{A}_{ij}, \bar{A}_{ij}\} \text{ for each } (i,j) \neq (k,m) \\ &A_{km} \in [\underline{A}_{km}, \bar{A}_{km}] \end{aligned} \qquad (*)$$

for some (k,m) (a singular matrix of the form $A_{ij} \in \{\underline{A}_{ij}, \bar{A}_{ij}\}$ for each i,j need not exist; cf. the interval matrix $[-E, E]$ which contains the zero matrix but each matrix of the latter form satisfies $|\det A| = 1$). We shall show that our algorithm, if properly started and if it does not fail, finds a singular matrix in this form:

Theorem 2. Let the starting matrix A^0 in Step 0 satisfy $A^0_{ij} \in \{\underline{A}_{ij}, \bar{A}_{ij}\}$ for each i,j. Then the algorithm, provided it does not fail, constructs a singular matrix in the form (*).

Proof. From the definition of B in Step 3 and from the updating formula in Step 6, it follows that the property

$$A_{ij} \in \{\underline{A}_{ij}, \bar{A}_{ij}\} \text{ for each } i,j$$

is maintained throughout the algorithm and can be only violated during the construction of A_{km} in Step 7. Therefore, the final singular matrix has the form (*). □

According to our computational experience, the algorithm is rather effective, usually finding a singular matrix in a relatively small number of iterations. We ascribe this behavior to the steepest-descent-type technique built into the Step 4. In cases when a special form of the resulting singular matrix is not required, we recommend initiating the algorithm with the matrix $A^0 = \frac{1}{2}(\underline{A} + \bar{A})$; this will sometimes help to prevent the algorithm from failing at the early stages.

Example. Consider the interval matrix $A^I = [\underline{A}, \bar{A}]$ with

$$\underline{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \bar{A} = \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix}.$$

If started from A, the algorithm fails immediately. If initiated with $A^0 = \frac{1}{2}(\underline{A} + \bar{A})$, it constructs in one iteration the singular matrix

$$A = \begin{pmatrix} \frac{1}{2} & 1 \\ 1 & 2 \end{pmatrix}.$$

Final remark. We have given a theoretical description of the algorithm. If it is to be implemented in a finite-precision arithmetic, care is to be taken for the fact that the relation $C = A^{-1}$ will generally be violated and therefore the arguments based on it may not remain in force.

References

1. M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco (1979).
2. S. Poljak and J. Rohn, *Radius of Nonsingularity*, Technical Report, KAM Series 88–117, Charles University, Prague (1988) 11.
3. J. Rohn, *Systems of Linear Interval Equations*, Lin. Alg. Appls. **126** (1989) 39–78.