

A method for solving a system of linear equations efficiently in order to optimise the analysis code

G.J. Cats

Research Department

March 1981

This paper has not been published and should be regarded as an Internal Report from ECMWF.
Permission to quote from it should be obtained from the ECMWF.



European Centre for Medium-Range Weather Forecasts
Europäisches Zentrum für mittelfristige Wettervorhersage
Centre européen pour les prévisions météorologiques à moyen

ABSTRACT

The modifications in the analysis code that are required to solve a linear system instead of inverting a matrix are described. A very preliminary estimate of CPU time savings is given.

1. INTRODUCTION

In the present analysis code, a large correlation matrix is inverted twice for each data box, once for data checking and once for calculating the analysis increments and errors.

For data checking indeed every element of the inverted matrix must be known, but for the calculation of analysis increments it suffices to solve a linear system which is much cheaper than inverting a matrix.

In this working paper, a survey of the work involved to solve the linear system is given, together with a very rough estimate of the CPU time savings.

2. THE SUBROUTINE TO SOLVE A LINEAR SYSTEM

A subroutine ("SYMSOL") has been written to solve a linear system. It is based on a Gaussian triangularisation and elimination. The subroutine assumes the matrix to be symmetric: further, if it is not positive definite, the subroutine will abort with an error code. For symmetric, positive definite matrices SYMSOL is extremely stable (e.g. Wilkinson, 1965).

The code for SYMSOL has been derived from that of SYMINV, the matrix inversion subroutine: only a small piece of code (vectorisable due to work by D. Dent) had to be written.

As opposed to SYMINV, SYMSOL does not return an estimated condition number. This is because a condition number is the product of the matrix norm and the norm of the inverted matrix: SYMSOL does by its nature not calculate the inverse. No method to estimate the condition number from the matrix itself has been found yet (Jennings, 1977).

Appendix 1 gives a listing of SYMSOL and appendix 2 gives some results about stability tests with SYMSOL.

Solving the system $Ax=b$, where A is a 191 x 191 matrix, takes about 0.15 sec on Cray with SYMSOL: the inversion of A with SYMINV takes about 0.30 sec.

3. USE OF SYMSOL IN THE ANALYSIS SUITE

3.1 Calculation of the increment

The analysis increment is given by the expression

$$a = \gamma A^{-1} \sigma \quad (1)$$

where A is the correlation matrix, γ the vector describing the correlation between the analysed quantity and the observed quantities, and σ the observation vector.

For a given set of observations, $A^{-1} \sigma$ is fixed, say $A^{-1} \sigma = w$. Therefore, with w the solution of the linear system $Aw = \sigma$ we get:

$$a = \gamma \cdot w \quad (2)$$

Thus, A does not have to be inverted but it is sufficient to solve the linear system $Aw = \sigma$.

3.2 Accuracy of the calculated increment

The relative accuracy of the solution x to $Ax = b$ is determined by three factors:

$$\frac{\|\delta x\|}{\|x\|} = K \Omega p \quad (3)$$

In here, K is the condition number of A :

$$K = \|A\| \|A^{-1}\|, \quad (4)$$

and Ω is determined by the stability of the solution algorithm. For positive definite A and Gaussian elimination

$$\Omega \lesssim n^3 \quad (5)$$

where n is the matrix order (Ralston and Rabinowitz, 1978). Further

$$p = \frac{\|\delta b\|}{\|b\|} \text{ is the computer accuracy (for Cray } \sim 710^{-15}\text{).}$$

Eq. (3) as it stands is not applicable for error estimates, because SYMSOL does not give an estimate for K.

The matrix A is the sum of two positive definite matrices:

$$A = c + \epsilon$$

where ϵ is the observation error correlation. The smallest eigenvalue of ϵ be η . Then $\|A^{-1}\| \leq \frac{1}{\eta}$, because c has only positive eigenvalues.

Further, in appendix 3 it will be shown that for all x

$$(\gamma \cdot x)^2 \leq x c x \tag{6}$$

The following estimate of the error in the increments, relative to the length of the observation vector $(\sigma^2)^{1/2}$ can be derived.

$$\begin{aligned} \frac{1}{(\sigma^2)^{1/2}} |\delta(\sigma A^{-1} \gamma)| &\leq \left[\frac{1}{\sigma^2} ((\delta \sigma) A^{-1} \gamma)^2 \right]^{1/2} + \|\delta(A^{-1} \gamma)\| \\ &\leq \left(\frac{\delta \sigma A^{-1} \delta \sigma}{\sigma^2} \right)^{1/2} + \|A^{-1}\| \frac{\|\delta \gamma\|}{\|\gamma\|} \|\gamma\| \Omega \end{aligned}$$

where a factor Ω appears due to the algorithm to invert A.

Using $\frac{\|\delta \sigma\|}{(\sigma^2)^{1/2}} \sim p$, $\frac{\|\delta \gamma\|}{\|\gamma\|} \sim p$ and $\|\gamma\| < \sqrt{n}$

(the last because every element of γ has absolute value less than 1) this results in

$$E = \frac{1}{(\sigma^2)^{1/2}} |\delta(\gamma A^{-1} \sigma)| \leq \frac{n^{7/2}}{\eta} p \tag{7}$$

If it is required that $E < 10^{-4}$, for all $n \leq 191$, η must exceed $\sim 10^{-2}$.

The matrix ϵ can be written as the sum of two positive definite matrices

$$\epsilon = \delta + \zeta$$

where δ is diagonal. The condition $\eta \geq 10^{-2}$ is then certainly met if every diagonal element of δ exceeds 10^{-2} . Therefore, in order to obtain a reasonably accurate increment, it is required that every observation with uncorrelated

observation errors be given a normalised observation error in excess of 0.1. If there are correlations between observation errors, the normalised observation error must exceed $0.1 / \sqrt{1 - \kappa}$ where κ is the largest observation error correlation between different data.

The present analysis scheme does in general fulfill these requirements: the observation error is seldom, if ever, less than 0.3 and κ is at present 0.8. But it is advised that the requirement:

$$\eta \geq \frac{pn^{7/2}}{E(1-\kappa)} \quad (8)$$

be checked and, if not fulfilled, be imposed for each datum.

3.3 Calculation of the analysis error

The analysis error is given by the expression

$$e = 1 - \gamma A^{-1} \gamma \quad (9)$$

If SYMSOL is used, $A^{-1}\sigma$ will be known but not $A^{-1}\gamma$. Therefore, it is not possible to estimate the analysis error when the analysis increments are calculated. However, during data checking each element of A^{-1} must be known and it would be very cheap to calculate the analysis error then. The only disadvantage is that during data checking, observations are used that have not yet received their final flag - although those observations have undergone a preliminary check. Therefore, the estimate of the analysis error might become slightly less accurate. This does not seem a serious drawback.

3.4 Time savings

The following CPU timings apply to an analysis of 15 January 1979, 0000h, made with the present code:

GAP	- data checking	200 s
GAP	- increments and errors	240 s
Analysis	- total	580 s

The increments and errors calculation time is split up as follows:

<u>SUBROUTINE</u>	<u>TASK</u>	
SELINF	Selection of influence boxes	20s
SETLHS	Construction of correlation matrix	55s
CHOOSE	Inversion of correlation matrix	65s
COEFFS	Calculation of $A^{-1}G$ and analysis error	35s
VALUES	Calculation of analysis increments	30s
	Selection of data, of grids, etc.	35s

If SYMSOL is used, CHOOSE would become twice as fast and COEFFS would no longer be needed, although a small portion of it would be required in GAP - data checking to calculate the analysis errors. The time saving is therefore estimated to be ~ 60s, i.e. 10% of the total CPU time.

3.5 Required changes in the analysis code

From the previous sections, it is seen that the following (extensive) changes must be made in the analysis code:

1. The analysis error calculation must be moved to the data checking run - this involves also reading in the error grid and writing the errors.
2. In the increments calculating run, the matrix inversion must be replaced by SYMSOL. This requires a different data selection algorithm because unchecked data must not enter the correlation matrix anymore. Further, the observation error must be checked using eq. (8).
3. The subroutines to calculate the increments need changes.

The burden of the work is presumably in the different data selection algorithm.

4. CONCLUSION

If the linear system $Aw = \gamma$ is solved instead of the matrix A inverted during the analysis increments calculation, CPU time savings of ~ 60 seconds per analysis run (about 10%) are expected. The required recoding involves many subroutines.

REFERENCES

Jennings, A., 1977: Matrix Computation for Engineers and Scientists, John Wiley & Sons, London.

Ralston, A., and Rabinowitz, P., 1978: A first course in numerical analysis, McGraw-Hill, New York.

Wilkinson, J.H., 1965: The algebraic eigenvalue problem, Clarendon Press, Oxford.

Appendix 1

```

Listing of SYMSOL SUBROUTINE SYMSOL(A,NDIM,N,POSD,X,B,DEPS)
DIMENSION A(NDIM,NDIM),X(NDIM),W(300),B(NDIM)
C WORK ARRAY W IS NOT NEEDED IN CAL VERSION SO IT IS NOT IN ARG LIST
C SOLVE AX=B
C A IS A SYMMETRIC POSITIVE DEFINITE MATRIX.
C THE UPPER TRIANGLE OF A (IE. A(I,J) I.LT.J) IS NOT USED OR ALTERED
C POSD (OUTPUT): IF A IS POS.DEF., POSD=1.;
C ELSE POSD=-FLOAT(I) WHERE I IS THE RANK OF THE
C FIRST PRINCIPAL MINOR OF A THAT IS NOT POS.DEF.
C DEPS IS OPTIONAL TO MAKE THE POS. DEFINITE TEST MORE SEVERE. IF
C NOT SUPPLIED, DEPS=0.
C*****THIS VERSION REQUIRES NUMARG*****
C
C THIS VERSION IS OPTIMIZED FOR THE CRAY CFT COMPILER
C A FASTER CAL VERSION OF SYMSOL EXISTS ON ECLIP
C REDUNDANT TESTS ON W ARE INCLUDED TO ENSURE ZERO TRIP DO LOOPS WORK
C
C IF(N.LT.1)RETURN
EPS=0.
IF (NUMARG(1).EQ.7) EPS=DEPS
C**** 1. GAUSS ELIMINATION TO OBTAIN LOWER TRIANGLE; OVERWRITES A & B
POSD=1.
IF (A(1,1).LE.EPS)GOTO91
X(1)=1./A(1,1)
IF(N.GT.1)GOTO10
A(1,1)=X(1)
X(1)=X(1)*B(1)
RETURN
10 DO 14 J=2,N
DO 11 I=J,N
11 A(I,J-1)=A(I,J-1)*X(J-1)
B(J-1)=B(J-1)*X(J-1)
IVL=N-J+1
DO 121 I=1,IVL
121 W(I)=A(J-1+I,J)
WB=B(J)
DO 13 K=2,J
DO 122 I=1,IVL
122 W(I)=(-A(K-1,K-1)*A(J,K-1)) * A(J-1+I,K-1) + W(I)
WB=-A(K-1,K-1)*A(J,K-1)*B(K-1)+WB
13 CONTINUE
DO 131 I=1,IVL
131 A(J-1+I,J)=W(I)
B(J)=WB
C CHECK FOR POSITIVE-DEFINITENESS AND INVERT DIAGONAL MATRIX D.
IF (A(J,J).LE.EPS)GOTO91
X(J)=1./A(J,J)
14 CONTINUE
B(N)=X(N)*B(N)
C
C**** 2. CONDITION NUMBER
C
C*****NOT KNOWN YET*****
C
C**** 3. SOLVE AX=B
C
DO 30 I=1,N
30 X(I)=B(I)
DO 32 J=N,1,-1
CDIRS IVDEP
DO 31 K=J-1,1,-1
31 X(K)=X(K)-A(J,K)*X(J)
32 CONTINUE
C
C**** 4. RETURN
C
4 RETURN
C
C**** 9. ERRORS
C
91 POSD=-FLOAT(J)
RETURN
END

```


Accuracy of SYMSOL vs. that of SYMINV

For several choices of the matrix A, the largest element of $E = |AA^{-1} - I|$ has been calculated for A^{-1} obtained with SYMINV and A^{-1} obtained from the SYMSOL solutions of $Ax = e_i$ $i = 1 \dots n$, where n is the matrix order and e_i the i^{th} unit vector. The matrix $AA^{-1} - I$ is not a very clean error estimator for A^{-1} , but it is suitable for a comparison of the two inversion methods used.

The matrices and the largest element of E are given in table A2.1. The Hilbert matrix has elements $A_{ij} = (i + j)^{-1}$ and is a notoriously ill-conditioned positive definite matrix. (Cray-1 refuses to invert it if its order is larger than 11.) The exp2 matrix has elements $A_{ij} = \exp(-(i-j)^2/5)$. Some matrices had their diagonal elements increased by ϵ to improve their condition. The upper bound on the condition number as estimated by SYMINV

$$(i.e. \max_{\substack{i=1..n \\ j=1..n}} |A_{ij}| \max_{\substack{i=1..n \\ j=1..n}} |A_{ij}^{-1}|)$$

is also shown in the table.

From the table, it is seen that SYMSOL behaves better than SYMINV, especially if the matrix is ill-conditioned. It is also seen that then both SYMINV and SYMSOL produce results that are extremely sensitive to the elements of the matrix, which is mathematically correct.

Table A2.1 Comparison of SYMINV and SYMSOL. Numbers in brackets are $\max_{i \neq j} |(AA^{-1})_{ij}|$: if these enter the table, the error in the diagonal elements of $|AA^{-1}|$ is certainly less than 10^{-3} .

Matrix			Cond.nr.	$\max_{i,j} (AA^{-1})_{ij} - \delta_{ij} $	
Type	Order	ϵ		SYMINV	SYMSOL
Hilbert	2	0	10^2	(0)	(0)
"	3	0	$5 \cdot 10^3$	$(9 \cdot 10^{-13})$	$(9 \cdot 10^{-13})$
"	5	0	10^7	$9 \cdot 10^{-10}$	$5 \cdot 10^{-10}$
"	8	0	10^{12}	$4 \cdot 10^{-4}$	$4 \cdot 10^{-6}$
"	8	10^{-6}	$2 \cdot 10^7$	$4 \cdot 10^{-9}$	$6 \cdot 10^{-10}$
"	10	0	10^{15}	1	10^{-2}
"	19	10^{-6}	$2 \cdot 10^8$	10^{-8}	$5 \cdot 10^{-10}$
"	191	10^{-6}	$2 \cdot 10^{10}$	$3 \cdot 10^{-8}$	$5 \cdot 10^{-10}$
exp 2	5	0	$2 \cdot 10^3$	$(9 \cdot 10^{-13})$	$(3 \cdot 10^{-13})$

Appendix 3

Proof of $(\mathbf{x} \cdot \boldsymbol{\gamma})^2 \leq \mathbf{x} \mathbf{C} \mathbf{x}$ for all \mathbf{x} .

Because $\boldsymbol{\gamma}$ is a vector of correlations and \mathbf{C} is a correlation matrix (say $n \times n$) the $(n+1) \times (n+1)$ matrix \mathbf{M} defined by

$$\mathbf{M} = \begin{pmatrix} 1 & \boldsymbol{\gamma}^T \\ \boldsymbol{\gamma} & \mathbf{C} \end{pmatrix}$$

is a correlation matrix. It is positive definite and for all $(n+1)$ -vectors $\boldsymbol{\zeta} = \begin{pmatrix} \xi \\ \mathbf{x} \end{pmatrix}$ where \mathbf{x} is an arbitrary n -vector:

$$\boldsymbol{\zeta} \mathbf{M} \boldsymbol{\zeta} \geq 0$$

i.e. $\xi^2 + 2\xi\boldsymbol{\gamma} \cdot \mathbf{x} + \mathbf{x} \mathbf{C} \mathbf{x} > 0$

for all ξ and \mathbf{x} . This is true if and only if for all \mathbf{x}

$$(\boldsymbol{\gamma} \cdot \mathbf{x})^2 \leq \mathbf{x} \mathbf{C} \mathbf{x} \tag{A3.1}$$

This completes the proof.

An interesting corollary of this theorem is that the analysis error (if there are no observation errors)

$$e = 1 - \boldsymbol{\gamma} \mathbf{C}^{-1} \boldsymbol{\gamma}$$

is positive.

For, taking $\mathbf{x} = \mathbf{C}^{-1} \boldsymbol{\gamma}$ in (A3.1) we get:

$$(\boldsymbol{\gamma} \mathbf{C}^{-1} \boldsymbol{\gamma})^2 \leq \boldsymbol{\gamma} \mathbf{C}^{-1} \boldsymbol{\gamma}$$

i.e. $|\boldsymbol{\gamma} \mathbf{C}^{-1} \boldsymbol{\gamma}| \leq 1 \tag{A3.2}$

Obviously, the proof of (A3.1) and (A3.2) holds also if an observation error is introduced.

Another interesting consequence of (A3.1) is that a small eigenvalue of c (say λ_i , with unit eigenvector $e_i : \lambda_i \ll 1$) does not contribute much to the analysis:

$$(\gamma \cdot e_i)^2 \ll \lambda_i$$

although it is a large eigenvalue of c^{-1} . This might be a starting point to the development of a scheme that does not invert the whole correlation matrix but only that part of the matrix that corresponds to a limited range of eigenvalues.