1 SUMMARY

To carry out a rank one update to a given positive definite or semi-definite symmetric matrix which is stored in a factorized form $A = LDL^T$, i.e. given a rank one matrix $\sigma zz^T$ (z a real vector) forms $\tilde{A} = A + \sigma zz^T$.

The subroutine was written to be used by optimization subroutines and will also: (i) accumulate a sum of rank one updates, (ii) carry out projection and allied operations on $A$ which reduce the rank, and (iii) update rank deficient matrices where it is known from other considerations that the rank remains unchanged.

There are additional entry points which, factorize $A = LDL^T$, recover $A$ from its factors, compute $Ax$ or $A^{-1}x$, and obtain $A^{-1}$ in factored form.

The method is described in M.J.D. Powell and R. Fletcher, AERE TP.519.


2 HOW TO USE THE PACKAGE

The matrix $A$ is represented using the minimal storage of $n(n+1)/2$ elements where $n$ is the dimension of the problem. To facilitate operating with $A$, a number of independent subroutines have been provided with entry names MC11B/BD, MC11C/CD, MC11D/DD, MC11E/ED and MC11F/FD. These perform operations including reducing a matrix to its factors, multiplying out the factors, operating with the factors of $A$ on a vector $z$ to obtain either $Az$ or $A^{-1}z$, and replacing the factors of $A$ by the matrix $A^{-1}$. These facilities are described in more detail in §2.2.

2.1 Argument list

*The single precision version*

CALL MC11A(A,N,Z,SIG,W,IR,MK,EPS)

*The double precision version*

CALL MC11AD(A,N,Z,SIG,W,IR,MK,EPS)

$A$ is a REAL (DOUBLE PRECISION in the D version) array of $n(n+1)/2$ elements in which the matrix $A = LDL^T$ must be given in factored form. The order in which elements of $L$ and $D$ are stored is $d_1, l_{21}, l_{31}, ..., l_{n1}, d_2, l_{32}, ..., l_{n2}, ..., d_{n1}, l_{n,n-1}, d_n$. The factors of the matrix $\tilde{A} = A + \sigma zz^T$ will overwrite those of $A$ on exit.

$N$ is an INTEGER variable which must be set by the user to $n$ the dimension of the problem. Restriction: $n \geq 1$.

$Z$ is a REAL (DOUBLE PRECISION in the D version) array of length at least $n$ which must be set by the user to contain the vector $z$. The array $Z$ is overwritten by the subroutine.

$SIG$ is a REAL variable which must be set by the user to $\sigma$. The value of $\sigma$ is not restricted to $\pm 1.0$, but if $\sigma < 0$ then it must be known from other considerations that $\tilde{A}$ is positive definite or semi-definite, apart from the effects of round-off error.

$W$ is a REAL (DOUBLE PRECISION in the D version) array of $n$ elements. If $\sigma > 0$ then $W$ is not used, and the name of any array can be inserted in the calling sequence. If $\sigma < 0$ then $W$ is used as workspace. In addition for $\sigma < 0$ it may be possible to save time by setting in $W$ the vector $v$ defined by $Lv = z$. The ways in which this can occur are described under MK below.

$IR$ is an INTEGER variable which must be set by the user so that $|IR|$ is the rank of $A$. If the rank of $\tilde{A}$ is expected to be different from that of $A$, set $IR \leq 0$. On exit from MC11A/AD, $IR \geq 0$ will contain the rank of $\tilde{A}$.
MK is an INTEGER variable to be set by the user only when \( \sigma < 0 \), as follows. If the vector \( \mathbf{v} \) defined by \( \mathbf{L} \mathbf{v} = \mathbf{z} \) has not been calculated previously, set \( MK = 0 \). If MC11E/ED has been used previously to calculate \( \mathbf{A} \mathbf{z} \), then \( \mathbf{v} \) is a by-product of this calculation and is stored in the \( \mathbf{W} \) parameter of MC11A/AD. In this case transfer \( \mathbf{v} \) to the \( \mathbf{W} \) parameter of MC11A/AD and set \( MK = 1 \). If \( \mathbf{z} \) has been calculated as \( \mathbf{z} = \mathbf{A} \mathbf{u} \) for some arbitrary vector \( \mathbf{u} \) using MC11D/DD, then again \( \mathbf{v} \) is a by-product of the calculation and is available in the \( \mathbf{W} \) parameter of MC11D/DD. In this case (or any other in which \( \mathbf{v} \) is known) set \( \mathbf{v} \) in the \( \mathbf{W} \) parameter of MC11A/AD and set \( MK = 2 \).

EPS is a REAL (DOUBLE PRECISION in the D version) variable to be set only when \( \sigma < 0 \) and \( \hat{\mathbf{A}} \) is expected to have the same rank as \( \mathbf{A} \). In the ill-conditioned cases a nonzero diagonal element of \( \hat{\mathbf{D}} \) (where \( \mathbf{A} = \mathbf{L} \mathbf{D} \mathbf{L}^T \)) might become so small as to be indeterminate. Two courses of action are possible. One is to introduce a small perturbation in order that \( \hat{\mathbf{A}} \) keeps the same rank as \( \mathbf{A} \). This is the normal course of action and is achieved by setting \( EPS \) equal to the relative machine precision \( \varepsilon \). The other course of action is to let the rank of \( \hat{\mathbf{A}} \) be one less than the rank of \( \mathbf{A} \). This is achieved by setting \( EPS \) equal to zero.

### 2.2 The other entry points

Other entry points are provided to facilitate operating with \( \mathbf{A} \) which is stored in compact form. In all of these \( \lambda \) is a REAL one dimensional array of \( n(n+1)/2 \) elements where \( n \) is the dimension of the problem. Each entry point is an independent subroutine.

**MC11B/BD**  
Factorize a positive definite symmetric matrix.

*The single precision version*

```
CALL MC11B(A,N,IR)
```

*The double precision version*

```
CALL MC11BD(A,N,IR)
```

\( A \) is a REAL (DOUBLE PRECISION in the D version) array of \( n(n+1)/2 \) elements which must contain the elements of \( \mathbf{A} \) in the order \( a_{11}, a_{21}, \ldots, a_{n1}, a_{22}, a_{32}, \ldots, a_{n2}, \ldots, a_{n-1,n-1}, a_{nn} \): that is as successive columns of its lower triangle. On exit \( \mathbf{A} \) will be over-written by the factors \( \mathbf{L} \) and \( \mathbf{D} \) in the form described in §2.1, argument \( \lambda \).

\( N \) is an INTEGER variable which must be set by the user to \( n \) the dimension of the problem. **Restriction:** \( n \geq 1 \).

\( IR \) is an INTEGER variable set by MC11B/BD to the rank of the factorization. If the factorization has been performed successfully \( IR = N \) will be set. If on return \( IR < N \) then the factorization has failed because \( \mathbf{A} \) is not positive definite (possibly due to round-off error). In this case the factors of a positive semi-definite matrix of rank \( IR \) will be found in \( \hat{\mathbf{A}} \). However the results of this calculation are unpredictable, and MC11B/BD should not be used in an attempt to factorize positive semi-definite matrices.

**MC11C/CD**  
Multiply out the factors \( \mathbf{LDL}^T \) to obtain \( \mathbf{A} \).

*The single precision version*

```
CALL MC11C(A,N)
```

*The double precision version*

```
CALL MC11CD(A,N)
```

\( A \) is a REAL (DOUBLE PRECISION in the D version) array of \( n(n+1)/2 \) elements which must contain the elements of \( \mathbf{A} \) factored in the form described in §2.1, argument \( \lambda \). On return the factors will have been over-written by the explicit matrix \( \mathbf{A} \), the order of the elements being the same as that described for input to MC11B/BD.

\( N \) is an INTEGER variable which must be set by the user to \( n \) the dimension of the problem. **Restriction:** \( n \geq 1 \).
MC11D/DD Calculate the vector $z' = Az$ where $A$ is in factored form.

The single precision version

CALL MC11D(A,N,Z,W)

The double precision version

CALL MC11DD(A,N,Z,W)

$A$ is a REAL (DOUBLE PRECISION in the D version) array of $n(n+1)/2$ elements which must contain the elements of $A$ factored in the form described in §2.1, argument $A$.

$N$ is an INTEGER variable which must be set by the user to $n$ the dimension of the problem. **Restriction:** $n \geq 1$.

$Z$ is a REAL (DOUBLE PRECISION in the D version) array of $n$ elements which must be set by the user to the vector $z$. On exit, $Z$ contains the vector $z' = Az$.

$W$ is a REAL (DOUBLE PRECISION in the D version) array of $n$ elements which is set by the subroutine to the vector $v$ defined by $Lv = z'$. If this vector is not of interest, replace $W$ by $Z$ in the calling sequence to obviate the need to supply extra storage.

**MC11E/ED** Calculate the vector $z' = A^{-1}z$ where $A$ is in factored form.

The single precision version

CALL MC11E(A,N,Z,W,IR)

The double precision version

CALL MC11ED(A,N,Z,W,IR)

$A$ is a REAL (DOUBLE PRECISION in the D version) array of $n(n+1)/2$ elements which must contain the elements of $A$ factored in the form described in §2.1, argument $A$.

$N$ is an INTEGER variable which must be set by the user to $n$ the dimension of the problem. **Restriction:** $n \geq 1$.

$Z$ is a REAL (DOUBLE PRECISION in the D version) array of $n$ elements which must be set by the user to the vector $z$. On exit, $Z$ contains the vector $z' = A^{-1}z$.

$W$ is a REAL (DOUBLE PRECISION in the D version) array of $n$ elements which is set by the subroutine to the vector $v$ defined by $Lv = z'$. If this vector is not of interest, replace $W$ by $Z$ in the calling sequence to obviate the need to supply extra storage.

$IR$ is an INTEGER variable which must be set by the user to the rank of $A$.

**MC11F/FD** Calculate the explicit matrix $A^{-1}$ from the factors of $A$.

The single precision version

CALL MC11F(A,N,IR)

The double precision version

CALL MC11FD(A,N,IR)

$A$ is a REAL (DOUBLE PRECISION in the D version) array of $n(n+1)/2$ elements which must contain the elements of $A$ factored in the form described in §2.1, argument $A$. On exit this will be overwritten by the elements of the inverse matrix $A^{-1}$, in the order $a_{11}^{-1}, a_{21}^{-1}, \ldots, a_{nn}^{-1}$, as is done by MC11B/BD.

$N$ is an INTEGER variable which must be set by the user to $n$ the dimension of the problem. **Restriction:** $n \geq 1$.

$IR$ is an INTEGER variable which must be set by the user to the rank of $A$.

Notes:
(i) MC11F/FD should not be used to solve equations, in which case MC11E/ED should be used. MC11F/FD is intended for applications in which the explicit elements of $A^{-1}$ must be examined, for example in the use of variance-covariance matrices.

(ii) MC11E/ED and MC11F/FD both return without doing any calculation if $IR$ is not equal to $N$.

3 GENERAL INFORMATION

Use of common: None.

Workspace: $n(n+1)/2 + 2n$ words provided by the user in $A$, $Z$ and $W$. If $SIG>0$ the array argument $W$ is not used and may be dummied.

Other routines called directly: None.

Input/output: None.

Restrictions: None.

Timing: One call of MC11A/AD requires $\sim n^3$ multiplications, unless $\sigma<0$ and $MK=0$ when the figure is $\sim 1n^2$. One call of any of MC11B/BD, MC11C/CD or MC11F/FD requires $\sim n^3/6$ multiplications. One call of either MC11D/DD or MC11E/ED requires $\sim n^2/2$ multiplications.