

FSPAK90 - A FORTRAN 90 INTERFACE TO SPARSE-MATRIX PACKAGE FSPAK WITH DYNAMIC MEMORY ALLOCATION AND SPARSE MATRIX STRUCTURE

Ignacy Misztal¹, Miguel Perez-Enciso²

¹University of Georgia, Athens, USA

²Centre UdL-IRTA, 25005 Lleida, Spain

OBJECTIVE

Simplifying the use of the sparse matrix package FSPAK by providing a Fortran 90 interface

COMPUTING METHODS

FSPAK (Perez-Enciso *et al.*, 1994) is a sparse matrix package that performs a variety of computations with sparse symmetric semi-positive definite matrices, including obtaining solutions, Cholesky factorizations, sparse inverses, and determinants. However, the calling sequence to FSPAK is complicated, which makes FSPAK difficult to use.

Fortran 90 is a new version of the Fortran 77 programming language that contains a number of new useful features like dynamic data allocation, dense matrix operations, type structures, parameter overloading, default subroutine parameters, free source format, etc.

New features in FSPAK90, a Fortran 90 interface to FSPAK, include

- a) automatic memory allocation
- b) specification of sparse matrix by a structure rather than by individual components,
- c) optional use of most parameters,
- d) automatic execution of preceding steps,
- e) improved error handling.

USE

The structure IJA for sparse matrix storage is defined as:

```
type ija
  integer n,m           !n is number of equations; m is number of nonzeros
  integer, pointer::ia(:),ja(:) !will be ia(n+1), ja(m)
  real, pointer::a(:)    !will be a(m)
end type
```

A complete call to FSPAK90 is:

```
call fspak90(operation,ija,rs,sol,det,msglev,maxmem,rank)
```

where

```
operation=    "factorize"    - calculate sparse factorization
              "invert"      - calculate sparse inverse
```

“solve”	- solve a system of equation
“reset”	- reset the storage
“det”	- calculate determinant
“stat”	- print statistics

ija = sparse matrix structure in IJA form

rs = vector of right hand side

sol = vector of solutions

det = determinant

msglev=message level from 0 (minimum) to 3 (maximum); default=0

maxmem=maximum memory available in the system; default=infinite

rank=rank of the matrix

Examples

To factorize:

call fspak90('factor',ija) !all preceding steps done automatically

To invert:

call fspak90('invert',ija)

To solve:

call fspak90('solve',ija,rs,sol)

To deallocate the internal memory:

call fspak90('reset')

To solve only if maximum memory < 20,000k

call fspak90(ija,'solve',rs,sol, maxmem=20000)

Note that only relevant arguments for each step need to be included in calling FSPAK90. Reordering is performed the first time when FSPAK90 is called. Subsequent factorization except after the option “reset” will reuse the ordering. Subsequent solves will reuse the factorization.

Future developments

FSPAK90 may eventually become part of an object-oriented library for computations involving dense and sparse matrices in a variety of formats. Such a library would allow for development of programs as simple as created in matrix manipulation packages such as MATLAB or IML SAS, with efficiency of regular programming languages.

AVAILABILITY

FSPAK90 is available free of charge for educational institutions on anonymous FTP at num.ads.uga.edu or nce.ads.uga.edu, or via <http://nce.ads.uga.edu>.

COMPUTER ENVIRONMENT

FSPAK90 with FSPAK should compile under any Fortran 90 environment.

REFERENCES

Perez-Enciso, M., I. Misztal and M. A. Elzo (1994) Proc. 5th World Congress Gen. Appl. Livest. Prod. Vol. 22:77-78.