Voxel format for the finite element matrices

The nodes of voxel grids can be indexed by the triples (I,J,K),

$$1 \le I \le NX,\ \ 1 \le J \le NY,\ \ 1 \le K \le NZ$$

or enumerated by 1D numbers

$$INDN = I + (J-1) * NX + (K-1) * NX * NY.$$

For 3D elasticity problems, we have additionally 3 degrees of freedom (DOF) in each node. These DOF correspond to the displacements in the directions X, Y, Z.

The overall number of degrees of freedom is

$$ND = 3 * NX * NY * NZ$$

and the degrees of freedom at the node with index INDN can be indexed by the numbers

$$INDX = 3 * INDN - 2\,,$$

$$INDY = 3 * INDN - 1\,,$$

$$INDZ = 3 * INDN\,.$$

The discretization is given by the decomposition of the domain into eight-node bricks and consequently into tetrahedral finite elements. Hence, the stiffness matrix (each brick is divided into 6 tetrahedra) has all nonzero entries within a 27-node (81-DOF) regular stencil.

Supposing the symmetry of the matrix, we can additionally store only the upper triangular part of the matrix. It can be done row-by-row by using regular 42 element stencil for storage of the nonzero matrix entries.

The row numbers are given by the order of the records, column numbers have not to be stored due to the regular stencil. More exactly, the column numbers for the regular stencil and for the row with index I will be the following:

If ***I= INDX*** for some DOF, then the column numbers used in the 42 element stencil are subsequently:

$$J = I, I+1, \dots, I+5$$

and further

$$J = K, K+1, \dots, K+8 \quad \text{for } K = I+K1, I+K2, I+K3, I+K4,$$

where

$$K1 = 3 * NX - 3,$$

$$K2 = 3 * NX * NY - 3 * NX - 3,$$

$$K3 = 3 * NX * NY - 3,$$

$$K4 = 3 * NX * NY + 3 * NX - 3.$$

If ***I=INDY*** for some DOF, then the last entry in the stencil is not used and for the remaining positions, we have,

$$J = I, I + 1, \ldots, I + 4$$

and further

$$J = K, K + 1, \ldots, K + 8 \quad \text{for } K = I - 1 + K1, I - 1 + K2, I - 1 + K3, I - 1 + K4.$$

If ***I=INDZ*** for some DOF, then the last two entries in the stencil are not used and

$$J = I, I + 1, \ldots, I + 3$$

and further

$$J = K, K + 1, \ldots, K + 8 \quad \text{for } K = I - 2 + K1, I - 2 + K2, I - 2 + K3, I - 2 + K4.$$

In the FKBC.G32 file, one record involves three stencils for the rows which correspond to three degrees of freedom in one node. This enlarging of the record improves the speed of I/0 operations.

The use of the regular stencil enables us not to store the column numbers, which halves the storage demands (both column numbers and matrix entries require the same amount of storage - 4B). Because the decrease of the storage is very important (the matrix have to be transferred from the external memory within each iteration), we can also consider more economical storage for some special cases as e.g. rectangular grids, uniform division of the bricks to tetrahedral finite elements. But these possibilities are not implemented in this code.

The use of the matrix storage scheme can be also seen from the subroutines for conversion of FKBC.G32 and FRBC.G32, BINARY files to ASCI files and especially from the subroutine MXV for matrix by vector multiplication.