National Conference with International Participation
ENGINEERING MECHANICS 2007
Svratka, Czech Republic, May 14 – 17, 2007

UTILISATION EXTENDED BOUNDARY CONDITIONS IN FINITE ELEMENT ANALYSIS

L. Svoboda [1], B. Patzák [2]

Summary: *The authors are engaged in development of adaptive analysis of RC frames using the microplane model. The utilised finite element mesh is compound of 1D and 3D elements. In order to achieve a proper connection among these elements it was necessary to satisfy complex boundary conditions. This connection (boundary condition) was realized by "rigid arms" and "hanging nodes" which are special cases of "slave degree of freedom". Implementation and utilisation of the universal type of slave DOF will be discussed in this work.*

1. Introduction

The authors are engaged in development of adaptive analysis of RC frames with microplane joints. The analysis employes *d*-refinement which combines 1D and 3D geometrical model of RC beams. This approach leads to a complicated finite element (FE) mesh consisting of 1D (beam and truss) and 3D (brick) elements and brings two problems – generation of the mesh and connection of all finite elements into one compact unit. To satisfy the first one it was necessary to develop a special preprocessor capable to generate such composite FE mesh. In order to achieve true response it was necessary to provide suitable connection of 1D and 3D elements. There are two types of this connection. The first connection is between 1D and 3D model of the RC beam (segment). The second one is realised inside of 3D segment between 1D elements (representing bars of reinforcement) and 3D elements (representing concrete). To provide these connections the *rigid arm node* and *hanging nodes* was implemented in FE code. Both nodes were based on a so-called *slave degree of freedom* (sDOF). The implementation and utilisation of the hanging node, the rigid arm node, and the slave degree of freedom as well as design of the preprocessor will be discussed in this section.

2. Universal slave DOF

A common type of structural degree of freedom used in finite element packages supports only basic form of Dirichlet boundary conditions. It is typically able to simulate structural support (displacement is equal to zero), initial displacement, or prescribed displacement. In order to satisfy more complex boundary conditions the authors utilised the universal *slave degree of freedom* (sDOF) witch extends Dirichlet boundary condition.

[1] Ing. Ladislav Svoboda, Czech Technical University in Prague, Faculty of Civil Engineering, Department of Mechanics, Thákurova 7, Prague 6, 166 29, ladislav.svoboda@fsv.cvut.cz
[2] Doc. Dr. Ing. Bořek Patzák, Czech Technical University in Prague, Faculty of Civil Engineering, Department of Mechanics, Thákurova 7, Praha 6, 166 29, borek.patzak@fsv.cvut.cz

2.1. Definition

To illustrate the implementation we start from the general form of equilibrium conditions of the discretized system

$$\boldsymbol{K}^g \boldsymbol{r}^g = \boldsymbol{f}^g \quad in \quad \Omega\,, \tag{1}$$

with appropriate basic boundary conditions on $\delta\Omega$. The symbol $\boldsymbol{K}^g$ is the global stiffness matrix, $\boldsymbol{r}^g$ is the vector of unknown generalised displacements, and $\boldsymbol{f}^g$ is the generalised load vector. The dimension m of this set of linear algebraic equations is equal to number of unconstrained degrees of freedom. The unknown displacement vector $\boldsymbol{r}^g$ completed by known displacements (prescribed or equal to zero $\sim$ corresponding to constrained DOFs) can be rewritten into following form

$$\boldsymbol{r} = \{\, r_1^g\,;\, r_2^g\,;\, \ldots\,;\, r_m^g\,;\, \bar{r}_{m+1}\,;\, \bar{r}_{m+2}\,;\, \ldots\,;\, \bar{r}_n \,\}^T\,, \tag{2}$$

where and n is total number of DOFs (both constrained and unconstrained) and $(n-m)$ is number of constrained DOFs. In this context the discussed slave DOF $\hat{r}_i$ is defined as a linear combination of any other DOFs. In other words the slave is subordinated to other DOFs called master DOFs (masters). The general formulation of the displacement of *i*-th DOF which is slave DOF can be written as

$$\hat{r}_i = a_{i0} + \sum_j a_{ij} r_j\,, \quad j \in \langle 1, i) \cap (i, n\rangle, \quad i \in \langle 1, n\rangle \tag{3}$$

where a_{ij} are coefficients of the combination. The constant a_{i0} meaning is initial displacement. It is usually implemented already so it will not be further taken into account. The slave DOF $\hat{r}_i$ is in range $\langle 1, n\rangle$. It means it can be both constrained and unconstrained. If constrained, the “slave dependency” is converted to some other unconstrained master (s)DOF r_J which turns into sDOF if not yet. So if the displacement of the constrained slave DOF is written as (without the constant a_{i0})

$$const. = \hat{r}_i = a_{iJ} r_J + \sum_j a_{ij} r_j\,, \quad j \in \langle 1, i) \cap (i, J) \cap (J, n\rangle\,, \tag{4}$$

then the displacement of the new (unconstrained) slave DOF can be expressed as

$$r_J = \frac{1}{a_{iJ}} \hat{r}_i - \sum_j \frac{a_{ij}}{a_{iJ}} r_j\,, \quad j \in \langle 1, i) \cap (i, J) \cap (J, n\rangle\,, \quad a_{iJ} \neq 0\,. \tag{5}$$

After these modifications the Eq. (3) can be simplified into the form

$$\hat{r}_i = \sum_j a_{ij} r_j\,, \quad j \in \langle 1, i) \cap (i, n\rangle, \quad i \in \langle 1, m\rangle \tag{6}$$

In the case that the *j*-th master DOF is at once master and slave DOF, then the displacement r_j in Eq. (6) is replaced by the same equation. This recursive replacement is applied as long as there is no slave DOF on the right hand side of the equation. The obtained final linear combination has to be checked again to avoid cyclic dependence of the $\hat{r}_i$ on itself.

Application of Eq. (6) brings reduction of number of unknown DOFs so the final dimension of global stiffness matrix $\boldsymbol{K}^g$ becomes only $(m-s)$, where s is number of slave DOFs in given problem.

2.2. Implementation

There are two ways to apply the above defined slave DOF. At first we can do it on the global level by substitution of Eq. (6) into Eq. (1). The advantage of this way is its simple implementation. It is carried out by linear combination of rows of appropriate global matrix (e.g. stiffness matrix, mass matrix) and global vector (e.g. load vector). However, this implementation can be complicated in the case when there are several various external mathematical packages used for assembling and solving of governing equations. For this reason the second way was preferred – application on element level.

The Eq. (1) rewritten into element form follows

$$\boldsymbol{K}^s \boldsymbol{r}^s = \boldsymbol{f}^s \quad in \quad element\,, \tag{7}$$

Dimension r of this system of equations is equal to number of all DOFs on element. In order to get transparent expression, just one slave DOF $\hat{r}_i^s$, with s master DOFs, on element will be further supposed. Applying Eq. (6) on vector r^s and notating in matrix form the following equation can be written

$$\boldsymbol{r}^s = \boldsymbol{T} \boldsymbol{r}^x\,, \tag{8}$$

or itemised as

$$\begin{Bmatrix} r_1 \\ \vdots \\ r_{i-1} \\ \hat{r}_i \\ r_{i+1} \\ \vdots \\ r_r \end{Bmatrix} = \begin{bmatrix} 1 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & a_1 & \dots & a_s & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 1 \end{bmatrix} \begin{Bmatrix} r_1 \\ \vdots \\ r_{i-1} \\ r_1 \\ \vdots \\ r_s \\ r_{i+1} \\ \vdots \\ r_r \end{Bmatrix} \tag{9}$$

The superscript s means a *standard* vector or matrix, the superscript x means a one *expanded* by Eq. (6). Substituting Eq. (8) into Eq. (7), the relation

$$\boldsymbol{K}^s \boldsymbol{T} \boldsymbol{r}^x = \boldsymbol{f}^s \tag{10}$$

is obtained. To preserve usual symmetry of the set of equations in Eq. (7), Eq. (10) has to be multiplied by the transposed matrix $\boldsymbol{T}$ as follows

$$\boldsymbol{T}^T \boldsymbol{K}^s \boldsymbol{T} \boldsymbol{r}^x = \boldsymbol{T}^T \boldsymbol{f}^s\,. \tag{11}$$

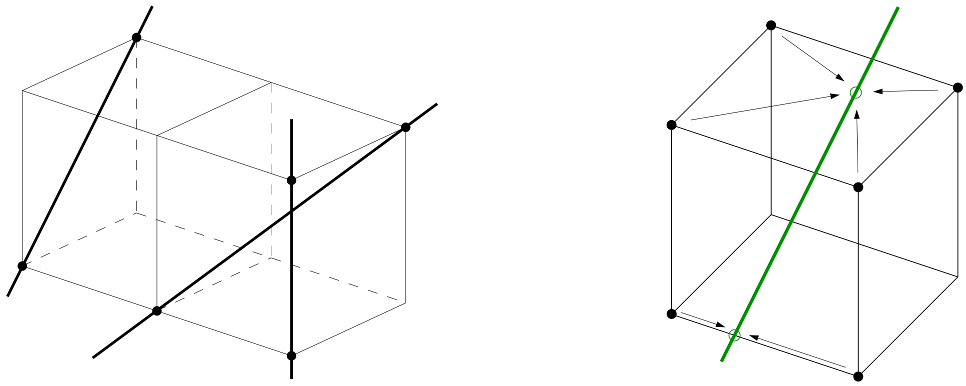
It can be rewritten in common compact form

$$\boldsymbol{K}^x \boldsymbol{r}^x = \boldsymbol{f}^x\,, \tag{12}$$

where

$$\boldsymbol{K}^x = \boldsymbol{T}^T \boldsymbol{K}^s \boldsymbol{T}\,, \tag{13}$$

$$\boldsymbol{f}^x = \boldsymbol{T}^T \boldsymbol{f}^s\,, \tag{14}$$



(a) ● - shared nodes. (b) ○ - hanging nodes and ● - master nodes.

Figure 1: Connection of brick and truss elements via (a) shared nodes and (b) hanging nodes.

2.3. Hanging node

A standard hanging node is node laying on an geometrical component of an element (edge, side or volume) but the node cannot be identified with an existing node on the element. Identical displacement of the hanging node and the element is desired. All DOFs of the element are slave DOFs and have the same set coefficient. The nodes of the corresponding component are masters. The set of coefficient is equal to interpolation functions of the component.

An universal hanging node can utilise the slave DOFs only in desired directions. The set of coefficient need not be equal to interpolation functions however the sum of the coefficients should be usually equal to 1.0.

2.4. Rigid arm

The RAN is connected to only one master. Its displacement, as the name suggests, is computed from both displacements and rotations of the master.

3. Connection inside of 3D segment

In order to create a correct and flexible model of 3D segments of a frame, it has to consist of two independent meshes. The primary mesh, compound of 3D elements (here hexahedrons $\sim$ bricks are used), represents concrete part of segment. Because microplane material constants have to be fitted according to experiments for one specific element size, all bricks have to be identical in size. Thus the primary mesh is regular. The secondary mesh is compound of truss elements and represents the reinforcement.

To guarantee the bond between concrete and reinforcement, both meshes have to be interconnected. In usual models this problem is solved during the creation of the primary mesh. It is generated so that the primary mesh is intersected by reinforcement right in vertices of bricks and nodes of the secondary mesh can be identified with the nodes of the primary one, see Fig. 1a.

In regular mesh, 3D elements are mostly intersected out of vertices. In such case, interaction between reinforcement bars and concrete bricks is ensured by hanging-nodes. It means that behaviour of a node of the secondary mesh is subordinate to behaviour of several nodes of the primary mesh. Practically all intersection points of a reinforcement bar and sides (edges) of bricks have to be found. Hanging nodes lie at these points and master nodes are identical with vertices of corresponding sides (edges), see Fig. 1b.

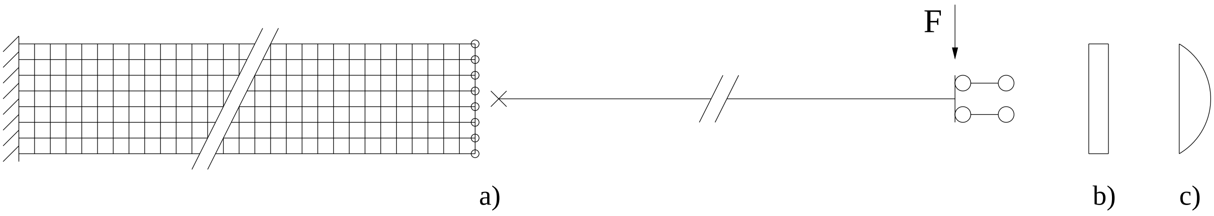


Figure 2: a) example b) uniform distribution c) parabolic distribution

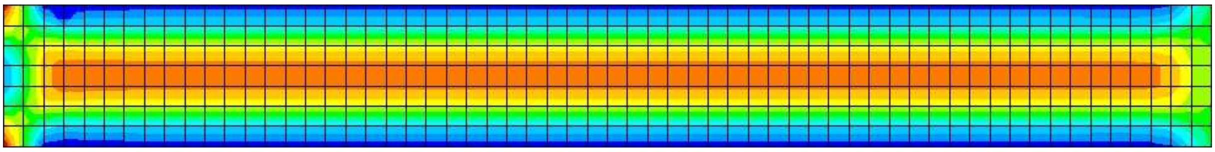


Figure 3: Stress τ_{xz} for uniform distribution.

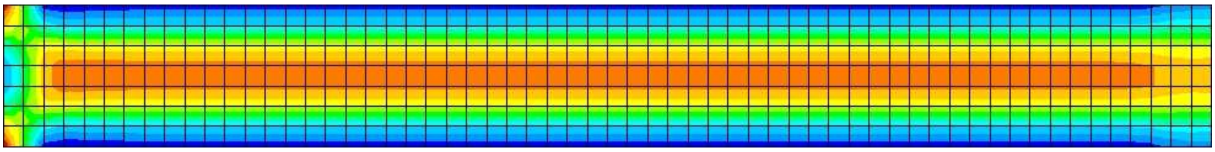


Figure 4: Stress τ_{xz} for parabolic distribution.

For this purpose a preprocessor was developed. At first, it finds all hanging nodes and divides rods of the secondary mesh, which are given as line segments or polygons, into elements. For each hanging node it finds its master nodes and computes natural coordinates inside corresponding brick, side or edge. In the case of a large amount of elements, it would be too slow to find hanging nodes going over all elements and finding possible intersection with each reinforcement polygon. That is why the preprocessor firstly maps complete connectivity of the primary mesh. Next it goes over each element and finds a master element for the first node of the polygon. Now the following intersection is searched always on elements adjacent to the last intersected (master) element only. In this manner it continues along the polygon to its end.

4. Connection 1D and 3D segment - example

A simple beam is used for demonstration potential of the slave DOF. The left hand side part of the beam is modelled by 3D elements, the second part is modelled by 1D beam elements. The geometry, the load, and the support are shown in Fig. 2a). To get same longitudinal displacement and same rotation in the midspan of the beam, all $\circ$ nodes (only DOFs in direction x) are modelled as the RAN and connected to $\times$ node. To get same transverse displacement and to allow transverse contraction of 3D segment in the midspan, the $\times$ node (only DOFs in direction z) is modelled as the HN and connected to $\circ$ master nodes. The distribution of the HN coefficients is uniform, see Fig. 2b), and parabolic, see Fig. 2c). The sum of the coefficients is 1.0. The parabolic distribution corresponds to real high-wise distribution of shear stress so it gives better response, see Fig. 3 and Fig. 4.

5. Conclusion

An extended boundary conditions was presented. It allows a displacement of particular DOF to be computed as linear combination of other DOFs. The proposed tool is general and powerful. It enables to simulate complex and complicated boundary conditions, as shown in this paper.

6. Acknowledgement

The support of the grant GA 103/05/2315 is gratefully acknowledged.