27/28th International Conference
ENGINEERING MECHANICS 2022
Milovy, Czech Republic, May 9 –12, 2022
Paper **#92**, pp. 301–304, doi: 10.21495/51-2-301

COMPARISON OF CO-ROTATIONAL AND NONLINEAR FINITE ELEMENT SIMULATIONS WITH GEOMETRIC NONLINEARITIES

Páleník R.[*], Molčan M.[**], Poruba Z.[***]

Abstract: *The co-rotational formulation offers a fast and numerically stable pseudolinear solution technique for structural problems with large displacements but small strains. The aim of this paper is to demonstrate capabilities of the implemented algorithm with the consistent element independent co-rotational formulation in a geometrically nonlinear static analysis. The co-rotational formulation incorporates linear finite elements into a co-rotating local frame following the rigid body motion of the element and the geometric nonlinearities are accounted for via the rotation of this local frame. With the use of a hexahedral element with linear shape functions, the main steps of the co-rotational and nonlinear algorithm are compared. Additionally, the extra shape functions may be easily added into the co-rotational element in order to avoid shear locking. Finally, the algorithms are numerically compared on a demonstration example of a cantilever solid block undergoing large displacements. The results of the co-rotational and nonlinear algorithm are almost the same.*

Keywords: Co-rotational formulation, Finite element method, Geometrically nonlinear analysis, Newton-Raphson iterations, Extra shape functions.

1. Introduction

With the increasing demand on light-weight design and high performance, many modern structures are becoming slenderer, longer, and more complex in shape such as aircraft wings and wind turbine blades. During the operation of such structures, large displacements can arise. As a result, the strain-displacement relation becomes nonlinear and cannot be solved by a simple linear algorithm assuming the linear Cauchy strain. Conventionally, these geometrically nonlinear problems are solved by nonlinear (NL) algorithms often utilizing the total Lagrangian formulation of kinematics and the nonlinear Green-Lagrange strain as described in, e.g., (Bhatti, 2006). However, if strains remain small, algorithms utilizing the co-rotational (CR) formulation can be used. This brings faster and numerically more stable simulations (Müller, 2004).

A unified theoretical framework for CR formulation is excellently presented in (Felippa et al., 2005) in a wide context. The basic principle of CR formulation lies in splitting the total displacements of an element into a rigid body motion part (i.e., translations and rotations) which can be arbitrarily large and into a small local deformational part which can be processed using the Cauchy strain. This enables a reuse of already developed small-strain elements in a geometrically nonlinear analysis, which is a great advantage of CR formulation.

This paper focuses only on static problems. However, when coupled with a proper time-integration scheme, CR formulation can be also used in geometrically nonlinear dynamic analyses.

Applications of CR formulation include aircraft wings, wind turbine blades, transmission towers with electrical cables, various springs, foams, and lattice structures, or a real-time computer graphics.

[*] Ing. Radek Páleník: IT4Innovations National Supercomputing Center & Department of Applied Mechanics, VSB – Technical University of Ostrava; 17. listopadu 2172/15; 708 00, Ostrava-Poruba; CZ, radek.palenik@vsb.cz

[**] Ing. Michal Molčan: IT4Innovations National Supercomputing Center & Department of Applied Mechanics, VSB – Technical University of Ostrava; 17. listopadu 2172/15; 708 00, Ostrava-Poruba; CZ, michal.molcan@vsb.cz

[***] doc. Ing. Zdeněk Poruba, Ph.D.: Department of Applied Mechanics, Faculty of Mechanical Engineering, VSB – Technical University of Ostrava; 17. listopadu 2172/15; 708 00, Ostrava-Poruba; CZ, zdenek.poruba@vsb.cz

2. Description of three static algorithms for 3D solids: linear, CR, and NL

Three algorithms utilizing the finite element method for solution of static problems were implemented and tested. In this section, the consistent element independent CR algorithm according to (Moita et al., 1996) is compared with a conventional linear algorithm and with the NL algorithm according to (Bhatti, 2006), chapter 9. Although the algorithms were tested on the hexahedral element with linear shape functions, they are valid for all 3-dimensional solid elements. A linear elastic material is assumed.

For problems without nonlinearities, a simple linear algorithm can be used. Such systems are described by a constant global stiffness matrix $\mathbf{K}$, which is assembled from linear element stiffness matrices $\mathbf{K}_e$ each belonging to the e-th element of the mesh. Then the boundary conditions are applied. When a linear system is loaded by a global external force vector $\mathbf{f}_{\mathrm{ext}}$, the resulting vector of global displacements $\mathbf{u}$ of all mesh nodes is obtained in a single step as

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{f}_{\mathrm{ext}} \,. \tag{1}$$

Unfortunately, geometrically nonlinear problems cannot be described by a constant global stiffness matrix because their stiffness varies during the large displacements. Therefore, a conventional iterative Newton-Raphson (NR) procedure is incorporated in the NL and the CR algorithm. In each NR iteration a global tangent stiffness matrix $\mathbf{K}_{\mathrm{t}}$ and a global internal force vector $\mathbf{f}_{\mathrm{int}}$ are required.

Comparison of static displacements computation using the NL and CR algorithms is provided in Table 1. The main difference between the algorithms is that in the CR algorithm the linear element stiffness matrices $\mathbf{K}_e$ are precomputed for all elements only once and then, during the nonlinear NR iterations, they are only modified inside CR formulation onto the element tangent stiffness matrices $\mathbf{K}_{\mathrm{t}e}$. The modification is done based on the initial element coordinates $\mathbf{x}_{0e}$ and the current nodal displacements $\mathbf{u}_e$. In NL formulation, on the other hand, the $\mathbf{K}_{\mathrm{t}e}$ matrices and the element internal force vectors $\mathbf{f}_{\mathrm{i}e}$ are computed using the Gauss quadrature integration in every NR iteration. The integration procedure starts from the constitutional matrix $\mathbf{C}$ and utilizes the deformation gradient, the Green-Lagrange strain tensor, and the second Piola-Kirchhoff stress tensor. Therefore, the NL algorithm is computationally more expensive and introduces numerical instabilities.

Table 1: Main steps of a static analysis with the NL and CR algorithm.

NL algorithm	**CR algorithm**
Prepare constitutional matrix $\mathbf{C}$	Compute linear element stiffness matrices $\mathbf{K}_e$
Start load steps loop, inside of it start NR iterations	
$(\mathbf{C}, \mathbf{x}_{0e}, \mathbf{u}_e) \xrightarrow{\text{NL formulation}} (\mathbf{K}_{\mathrm{t}e}, \mathbf{f}_{\mathrm{i}e})$	$(\mathbf{K}_e, \mathbf{x}_{0e}, \mathbf{u}_e) \xrightarrow{\text{CR formulation}} (\mathbf{K}_{\mathrm{t}e}, \mathbf{f}_{\mathrm{i}e})$
Assemble global tangent stiffness matrix $\mathbf{K}_{\mathrm{t}}$ from $\mathbf{K}_{\mathrm{t}e}$ and global internal forces $\mathbf{f}_{\mathrm{int}}$ from $\mathbf{f}_{\mathrm{i}e}$ and apply boundary conditions	
Compute the vector of residual forces $\mathbf{r} = \mathbf{f}_{\mathrm{ext}} - \mathbf{f}_{\mathrm{int}}$	
NR iterations of displacements: $\mathbf{u}_{\mathrm{new}} = \mathbf{u}_{\mathrm{old}} + \mathbf{K}_{\mathrm{t}}^{-1}\mathbf{r}$	
Exit NR iterations if $\lVert\mathbf{r}\rVert / \lVert\mathbf{f}_{\mathrm{ext}}\rVert$ is smaller than predefined NR tolerance	
Increase external forces $\mathbf{f}_{\mathrm{ext}}$ up to full load and then exit load steps loop	

As proposed in (Crisfield, 1997), the implemented CR formulation also calculates the deformation gradient, but only in the centroid of the element and to subsequently extract the rotation matrix of the local element frame using the polar decomposition. The extracted orthogonal rotation matrix is then used to express local displacements within the co-rotating local frame and the transformation matrix, which transforms the element local internal forces to the element global internal forces $\mathbf{f}_{\mathrm{i}e}$. Since the local displacements are small, the element local internal forces are expressed using $\mathbf{K}_e$ as in the linear algorithm, where the Cauchy strain is utilized.

Adding the extra shape functions (ESFs)

The implemented standard hexahedral element with linear shape functions is not suitable for bending dominant loading. This problem, typically referred to as shear locking, e.g., in (Bhatti, 2006), can be overcome by a very fine mesh or more effectively by adding extra shape functions (ESFs), also referred to as incompatible modes in (Moita et al., 1996). The addition of 3 quadratic shape functions to the original 8 linear ones softens the element bending stiffness, and it also helps with near-incompressibility problems.

The ESFs do not belong to any element node and are eliminated at the element level by the static condensation. Thanks to the element independent CR formulation, no modification had to be done to the CR formulation because of this change of the linear element stiffness matrix.

3. Numerical tests on cantilever solid block undergoing large displacements

A straight three-dimensional cantilever block of the length of 2 m and square cross-section 0.1 m × 0.1 m is considered. A linear material model with Young's modulus of 200 GPa and Poisson's ratio of 0.3 is assumed in the whole range of strains. The block is discretized by identical hexahedral elements with linear shape functions in 2 different meshes: coarse (5 elements) and fine (320 elements). One end of the block is kept fixed while the other one is loaded by a static force acting in the negative direction of the coordinate z-axis.

For the sake of a clear visual comparison of the block true scale displacements, the block is designed not to be quite thin and when relatively large loading forces are applied, the resulting strains might not remain small. However, if the CR algorithm performs well in these exaggerated testing cases, it will do even better when used for slenderer structures with similarly large translations and rotations.

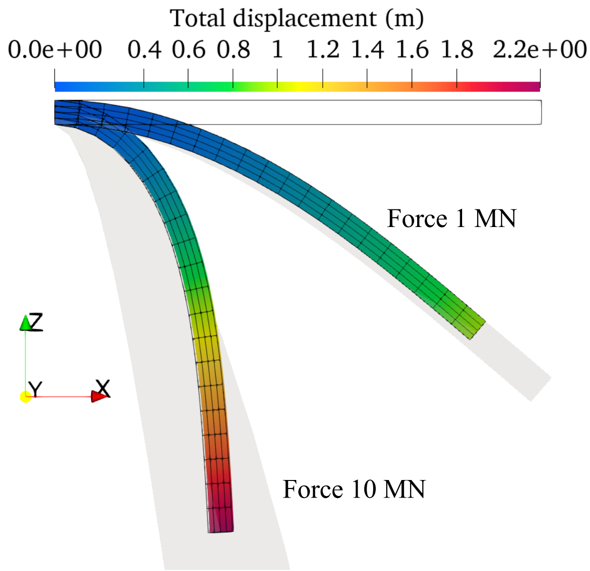

Fig. 1: Total displacements (true scale) computed without ESFs by NL (mesh), CR (color), and linear (grey) algorithm.

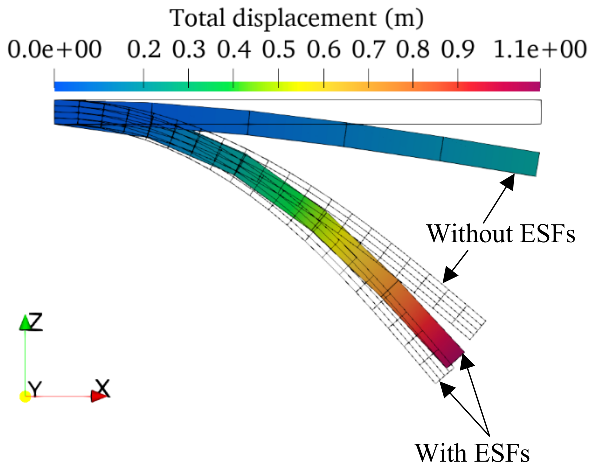

Fig. 2: Total displacements (true scale) for loading force 1 MN, computed without and with ESFs on coarse and fine meshes.

Comparison of NL, CR, and linear algorithm results

The described cantilever block is meshed by the fine mesh and successively loaded by 2 different forces. The resulting total displacements computed without ESFs by 3 different algorithms are depicted in Fig. 1:. In case of loading force of 1 MN, the linear algorithm artificially prolongs the beam and increases its cross-section which is an unacceptable result, while the CR algorithm produces almost identical results as the NL one. In case of loading force of 10 MN, the artificial increase in volume of the linear solution is so enormous that it does not fit in the picture and the CR solution starts to visibly differ from the NL one (the mesh representing the NL solution is more deflected than the color solid representing the CR solution). However, the relative error between the maximal total displacements of the CR and NL algorithm is still much smaller than 1 %, see Table 2. For both the CR and the NL algorithms and both loading forces, only 1 load step was required for the convergence. The CR algorithm reached NR tolerance smaller than 10^{-9} always in less iterations than NL one.

Influence of ESFs and mesh coarseness

Fig. 2: demonstrates the influence of ESFs and mesh coarseness on the total displacements. Now, only the CR algorithm (with and without ESFs) and loading force of 1 MN are employed. When considering only the coarse mesh results (color), ESFs decrease the beam stiffness significantly and enable much larger displacements, which is more accurate. Both fine mesh displacements are much closer to each other than the coarse ones. If an extremely fine mesh would be used, the displacements computed with and without ESFs will be almost the same.

Table 2: Comparison of the NL and CR algorithm without ESFs.

Force (MN)	**1**		**10**	
Algorithm	**NL**	**CR**	**NL**	**CR**
Max. total displacement (m)	0.9364	0.9358	2.1784	2.1741
Displacement relative error (%)	-	-0.06	-	-0.20
Number of NR iterations (-)	10	7	16	10

4. Conclusions

The linear, CR, and NL algorithm were implemented and numerically tested on an example of cantilever solid block undergoing large displacements. The solid block is discretized by hexahedral elements with linear shape functions. The results of the CR and NL algorithm are almost the same while the linear algorithm results are unacceptable. With both the CR and NL algorithms large load steps can be applied and for the CR algorithm a faster convergence has been observed. It was shown that for the bending dominant problems ESFs significantly improve performance of the hexahedral element with linear shape functions and unless an extremely fine mesh is used, ESFs should be employed to achieve trustworthy results. The CR algorithm was implemented and tested with a goal of future transient analyses on the high-performance computing architectures.

Acknowledgement

The presented work was supported by: the Doctoral grant competition VSB-TUO, No. CZ.02.2.69/0.0./0.0/19_073/0016945 within the Operational Program Research, Development and Education, under project DGS/TEAM/2020-033; the Ministry of Education, Youth and Sports from the National Program of Sustainability (NPS II) project „IT4Innovations excellence in science - LQ1602“; the Support for science and research in the Moravian-Silesian Region (RRC/10/2021).

References

Bhatti, M. A. (2006) *Advanced topics in finite element analysis of structures with Mathematica and MATLAB computations*. Wiley, New York.

Crisfield, M. A. (1997) *Non-linear finite element analysis of solids and structures, Volume 2: Advanced topics*. Wiley, Chichester.

Felippa, C. A. and Haugen, B. (2005) A unified formulation of small-strain corotational finite elements: I. Theory. *Computer methods in applied mechanics and engineering*, 194, pp. 2285-2335.

Moita, G. F. and Crisfield, M. A. (1996) A finite element formulation for 3-D continua using the co-rotational technique. *International journal for numerical methods in engineering*, 39, pp. 3775-3792.

Müller, M. and Gross, M. (2004) Interactive Virtual Materials, in: McCool, M.D., ed., *Proceedings of Graphics Interface 2004*, Canadian human-computer communications society, London, Ontario, pp. 239 - 246.