

IMPLEMENTATION OF A UNIFIED LIBRARY OF SOLVERS FOR NONLINEAR STRUCTURAL ANALYSIS

Mauro Poliotti^a, Oscar Möller^{a,b}, Juan P. Ascheri^a and Pablo L. Sierra^a

^a*Instituto de Mecánica Aplicada y Estructuras (IMAE), Facultad de Cs.Ex., Ingeniería y Agrimensura, Universidad Nacional de Rosario, Riobamba y Berutti, 2000 Rosario, Argentina, poliotti@fceia.unr.edu.ar, <http://www.imaefceia.unr.edu.ar>*

^b*Consejo de Investigaciones (CIUNR), Universidad Nacional de Rosario, Maipú 1065, 2000 Rosario, Argentina, <http://www.unr.edu.ar>*

Keywords: Unified Non-linear Library, Load Control, Displacement Control, Work Control, Arc-length.

Abstract. The nonlinear analysis of structural systems is a well known field among structural engineers and researchers. A large variety of solution schemes exists nowadays in literature each one with their advantages and disadvantages but none is suitable for all nonlinear problems and for different types of nonlinearities. Analysis tools have to be able to trace the full equilibrium path in order to so they have to include more than one solution scheme. As most of the control schemes can be written under the form of a constraint equation a unified framework allows the implementation of several control methods with the same algorithm enabling the user to choose the method that best suits their nonlinear problem. In this work a unified library of nonlinear solver schemes developed by other authors is implemented in a finite element environment. The library has the ability to introduce different nonlinear solution schemes with the same algorithm by means of a unified framework. Four control methods: load, displacement work and arc-length are presented and their limitations and capabilities are announced. Finally the library is tested through simple application examples and conclusions are presented.

1 INTRODUCTION

A general nonlinear structural problem may include several “singularities” as can be load or displacement limit points as well as snap-back problems, see Fig. (1). The nonlinear behavior may arise from different sources: geometrical, material, and nonlinearities of the boundary conditions as is the case of contact problems.

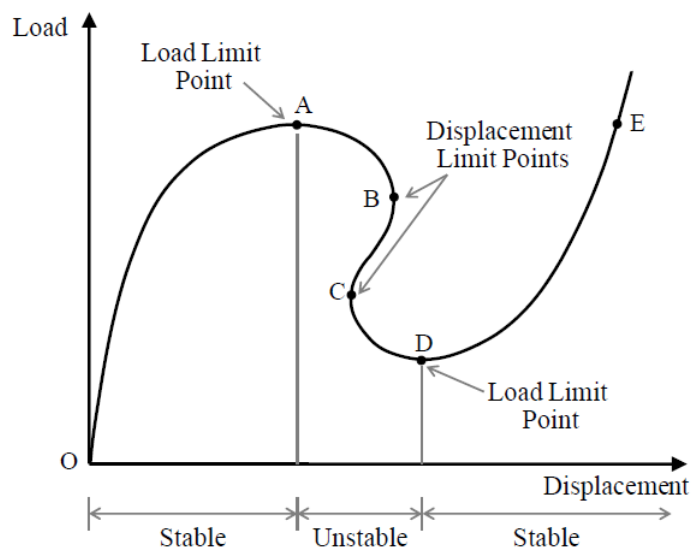


Figure 1: General Non Linear Problem

Solving the nonlinear problem is a well known field and several nonlinear solution schemes exists nowadays in literature, arc-length methods (Crisfield, 1983; Lam and Morley, 1992; Souza Neto and Feng, 1999; Ritto-Correa and Camotim, 2008), work-control (Bathe and Dvorkin, 1983), orthogonal residual procedure (Krenk, 1995), generalized displacement control method (Yang and Shieh, 1990), and classical methods as Newton-Raphson or displacement control. Each of the previous solutions has its own advantages and disadvantages, and the choice of the most suitable schemes is dependent on the nonlinear problem.

As none of the existing solutions schemes can be used for all nonlinear problems, any robust code for structural analysis should include as many solutions schemes as possible in order to allow the user to choose the one that best suits his problem. Also it is desirable that more than one scheme can be used during an analysis. For example, switching between schemes may be very useful to reduce the computational effort, like smooth parts of the equilibrium path, that may be traced by simple solution schemes and highly nonlinear parts, that may be followed by more complex solutions.

In this context, Leon et al. (2011), Leon (2010) developed a unified nonlinear library that includes several solution schemes with the same framework by means of a single algorithm. This library enables the user to choose and switch between different solution schemes and its object-oriented structure simplifies the implementation of new solution schemes.

In this work the unified framework is first described in its general form, also the implementation into a FE environment is made. Then, four solution schemes are presented and their limitations and capabilities are remarked. In order to test the library and the presented schemes two test are performed showing the differences between methods. Finally conclusions and further developments are summarized.

2 UNIFIED NONLINEAR FRAMEWORK

The unified nonlinear formulation is a usual incremental-iterative procedure where the load is applied in incremental steps and iterations are performed at each incremental step until the convergence criterion is satisfied. See Fig.(2).

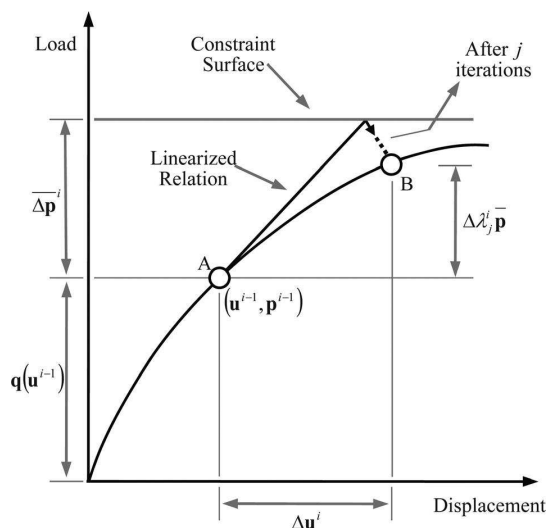


Figure 2: Incremental-iterative procedure

In structural nonlinear analysis the unbalanced force vector to be minimized in the i -incremental step and in the j -iteration can be written as follows:

$$\mathbf{r}_j^i = \mathbf{p}_j^i - \mathbf{q}(\mathbf{u}_j^i) \quad (1)$$

Where \mathbf{p}_j^i , is the vector of applied external forces and \mathbf{q}_j^i is the vector of internal forces which is a function of the displacements \mathbf{u}_j^i . In the incremental-iterative procedure the loads and displacements are computed by adding to the previous converged step the incremental updates at the current j -iteration.

$$\begin{aligned} \mathbf{u}_j^i &= \mathbf{u}^{i-1} + \Delta \mathbf{u}_j^i \\ \mathbf{p}_j^i &= \mathbf{p}^{i-1} + \Delta \mathbf{p}_j^i \end{aligned} \quad (2)$$

On each iteration the incremental updates presented in (2) are calculated in a similar way by adding the contribution of the previous iteration and the iterative updates at the current iteration.

$$\begin{aligned} \Delta \mathbf{u}_j^i &= \Delta \mathbf{u}_{j-1}^i + \delta \mathbf{u}_j^i \\ \Delta \mathbf{p}_j^i &= \Delta \mathbf{p}_{j-1}^i + \delta \mathbf{p}_j^i \end{aligned} \quad (3)$$

Taking (2) into account the residual vector (1) can be rewritten as:

$$\mathbf{r}_j^i = \mathbf{p}^{i-1} + \Delta \mathbf{p}_j^i - \mathbf{q}(\mathbf{u}^{i-1} + \Delta \mathbf{u}_j^i) \quad (4)$$

Finally the equation to be solved at the j -iteration of i -increment that governs the response of the structure is:

$$\mathbf{K}_{j-1}^i \delta \mathbf{u}_j^i = \mathbf{p}_j^i - \mathbf{q}_{j-1}^i \quad (5)$$

Where \mathbf{K}_{j-1}^i , is the tangent matrix of the structure.

Introducing into equations (2) and (3) the load factor parameter λ and the reference load vector $\bar{\mathbf{p}}$, the iterative form of the external applied load vector and the governing equation of the system can be rewritten as follows:

$$\begin{aligned}\mathbf{p}_j^i &= \mathbf{p}^{i-1} + \Delta \mathbf{p}_{j-1}^i + \delta \lambda_j^i \bar{\mathbf{p}} \\ \mathbf{K}_{j-1}^i \delta \mathbf{u}_j^i &= \mathbf{r}_{j-1}^i + \delta \lambda_j^i \bar{\mathbf{p}}\end{aligned}\quad (6)$$

Equation (6) is a set of N equations, being N the degrees of freedom of the system, with $N + 1$ unknowns: the N displacements $\delta \mathbf{u}_j^i$ and the load parameter $\delta \lambda_j^i$. In order to solve the system an additional constrain equation must be added.

$$\mathbf{a}_j^i \cdot \delta \mathbf{u}_j^i + b_j^i \delta \lambda_j^i = c_j^i \quad (7)$$

The system of $N + 1$ unknowns and $N + 1$ equations formed by (6) and (7) can be summarized in a matrix form as follows:

$$\begin{bmatrix} \mathbf{K}_{j-1}^i & -\bar{\mathbf{p}} \\ (\mathbf{a}_j^i)^T & b_j^i \end{bmatrix} \begin{Bmatrix} \delta \mathbf{u}_j^i \\ \delta \lambda_j^i \end{Bmatrix} = \begin{Bmatrix} \mathbf{r}_{j-1}^i \\ c_j^i \end{Bmatrix} \quad (8)$$

The new system's matrix presented in (8) is no longer symmetric and also the bandwidth has been modified from the original tangent stiffness matrix. This may cause a low computational efficiency when solving the system by means of traditional linear solvers. To overcome this problem the decomposition of the displacement iteration update is introduced as follows:

$$\begin{aligned}\delta \mathbf{u}_j^i &= \delta \lambda_j^i \delta \mathbf{u}_{p_j}^i + \delta \mathbf{u}_{r_j}^i \\ \mathbf{K}_{j-1}^i \delta \mathbf{u}_{p_j}^i &= \bar{\mathbf{p}} \\ \mathbf{K}_{j-1}^i \delta \mathbf{u}_{r_j}^i &= \mathbf{r}_{j-1}^i\end{aligned}\quad (9)$$

Finally the load parameter, which is needed to compute the total displacement for the j -iteration at i -increment, can be calculated using the constraint equation (7) and the displacement decomposition (9) as follows:

$$\delta \lambda_j^i = \frac{c_j^i - \mathbf{a}_j^i \cdot \delta \mathbf{u}_{r_j}^i}{\mathbf{a}_j^i \cdot \delta \mathbf{u}_{p_j}^i + b_j^i} \quad (10)$$

The benefit of using this unified framework is the fact that each nonlinear solution scheme differs only in the constraint equation and they can be solved by the same algorithm where only the calculation of the parameters of equation (7) and consequently the calculation of the load factor are different for each scheme.

The algorithm is implemented in a FE environment in MATLAB language, following the flow chart in Fig.(3). Additionally, for each nonlinear solution scheme, a function to calculate the load factor parameter is implemented. It is remarkable that the code allows the simple implementation of other nonlinear solutions by means of this "object" structure.

In the following sections four different control methods are presented under the form of the unified framework previously presented. Capabilities and limitations of each one are remarked.

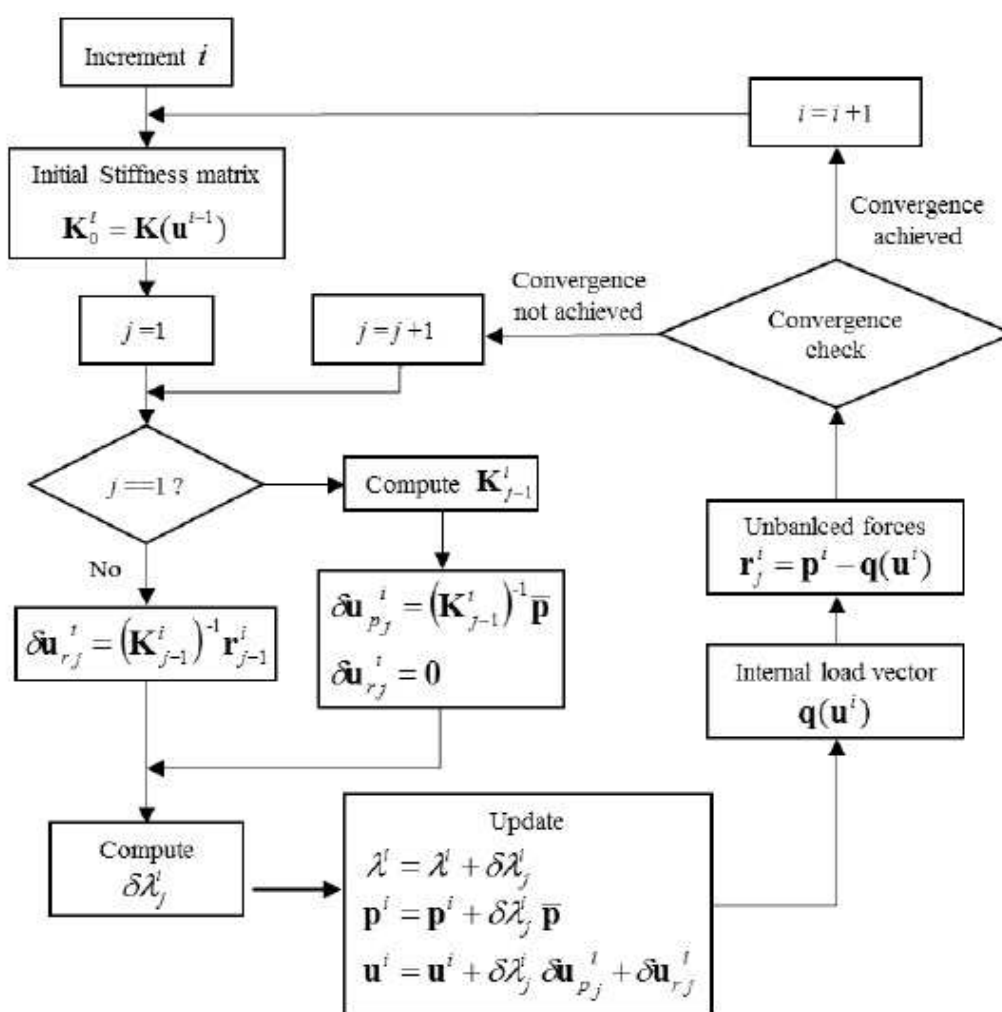


Figure 3: Unified scheme flow chart

3 LOAD CONTROL METHOD

The load control method (LCM) or Newton-Raphson method, where the external loads are computed at the first iteration of each increment and held constant in the remaining iterations, can be written in the unified scheme using the following constraint parameters:

$$\begin{aligned} a_j^i &= 0 \\ b_j^i &= 1 \end{aligned} \tag{11}$$

$$c_j^i = \begin{cases} \Delta \bar{\lambda} & \text{for } j = 1 \\ 0 & \text{for } j \geq 2 \end{cases} \tag{12}$$

Where $\Delta \bar{\lambda}$ is the prescribed initial load factor. The geometrical interpretation in a unidimensional case can be seen in Fig. (4)

The load control method is the most used in nonlinear analysis because of its simplicity and robustness. The weakness of the method comes from the fact that applied loads are kept constant during iterations, so the method has difficulties near load limits points and softening parts of the structural response cannot be traced.

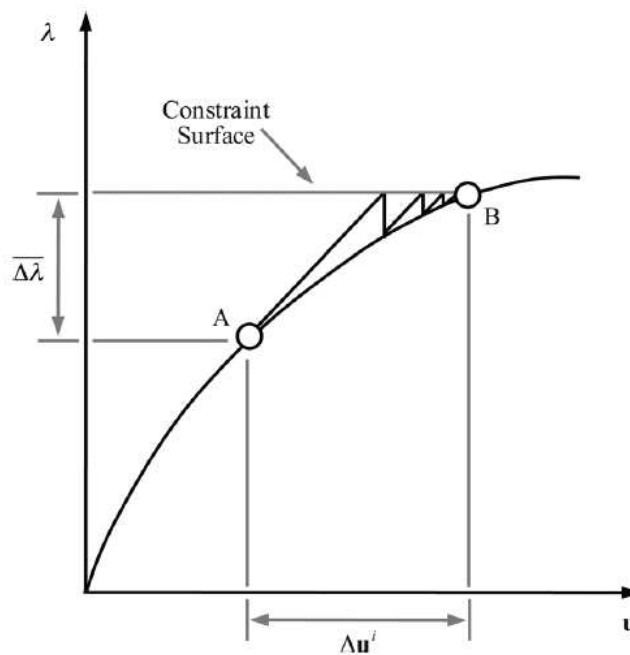


Figure 4: Load control method

4 DISPLACEMENT CONTROL METHOD

The displacement control method (DCM) uses a fixed displacement component as the control parameter to trace the structural response. The constraint parameters of the displacement control method are:

$$\begin{aligned}
 \mathbf{a}_j^i &= [0, 0, \dots, 1, \dots, 0, 0] \\
 b_j^i &= 0 \\
 c_j^i &= \begin{cases} \Delta \bar{u} & \text{for } j = 1 \\ 0 & \text{for } j \geq 2 \end{cases}
 \end{aligned}
 \tag{13}$$

Where $\Delta \bar{u}$, is the prescribed initial displacement. Finally by means of equation (10) the load factor is computed as:

$$\delta \lambda_j^i = \begin{cases} \frac{\Delta \bar{u}}{\delta u_{p_j^i}^{ctrl}} & \text{for } j = 1 \\ -\frac{\delta u_{r_j^i}^{ctrl}}{\delta u_{p_j^i}^{ctrl}} & \text{for } j \geq 2 \end{cases}
 \tag{14}$$

A geometric interpretation of the displacement control method for a one dimensional problem is shown in Fig.(5).

The displacement control method can capture load limit points and softening response but it fails near displacement limit points so it fails to reproduce snap-back phenomenon.

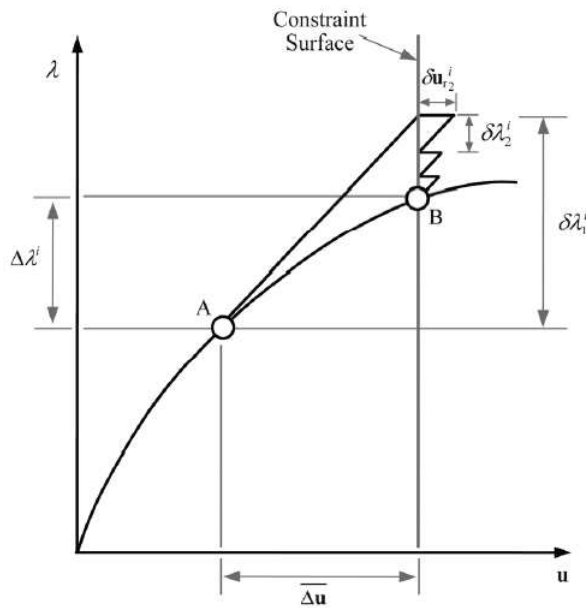


Figure 5: Displacement control method

5 ARC-LENGTH CONTROL METHOD

The arc-length control method (ALCM) considers simultaneous iterations on the load and displacements variables. The method is based in the constraint of the solution to an arc-length, where, this arc length is calculated as a norm of the increment.

$$\Delta \mathbf{u}_j^i \cdot \Delta \mathbf{u}_j^i + \eta (\Delta \lambda_j^i)^2 = (\Delta s_j^i)^2 \tag{15}$$

An initial arc-length is determined by equation (15) and the following iterations lie on the constrained surface created by the arc. See Fig. (6)

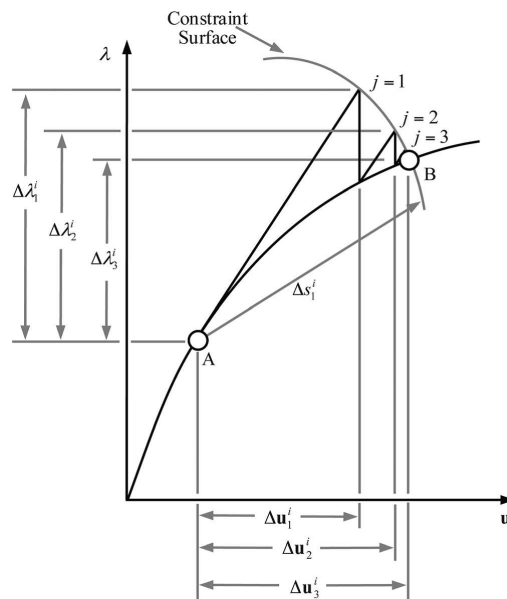


Figure 6: Arc-length control method

Expression (15) includes several version of the arc-length method by means of the positive

scalar parameter η . When η is equal to one, expression (15) becomes the equation of a hypersphere in the N+1 space and the method correspond to the spherical arc-length method. If η is equal to zero the equation (15) represents a hypercylinder in the N+1 space and the method corresponds to the cylindrical arc-length method. Finally the most general form is the elliptical arc-length method where $\eta > 0$ and $\eta \neq 1$. The surface is now a hiperellipse in the N+1 space. Figure (7) shows a geometrical interpretation in a two dimensional case.

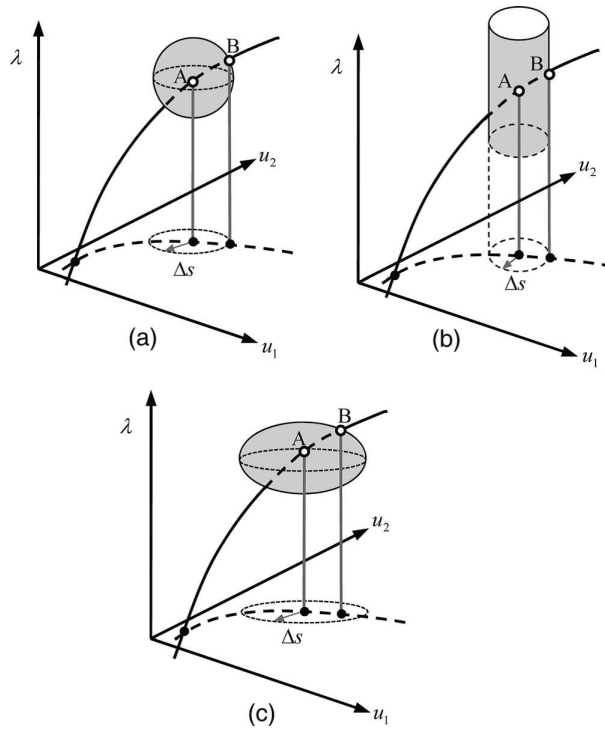


Figure 7: Arc-length control method: (a) spherical, (b) cylindrical, (c) elliptical

The constraint equation can be rewritten using the iteration updates instead of the increment update and the corresponding parameters of the constraint equation in the unified scheme are computed using this version of the arc-length constraint equation then, it results as follows:

$$\delta \mathbf{u}_1^i \cdot \delta \mathbf{u}_j^i + \eta \delta \lambda_1^i \lambda_j^i = (\Delta s_j^i)^2 \tag{16}$$

$$\begin{aligned} \mathbf{a}_j^i &= \delta \mathbf{u}_1^i = \delta \lambda_1^i \delta \mathbf{u}_{p1}^i \\ b_j^i &= \eta \delta \lambda_1^i \end{aligned} \tag{17}$$

$$c_j^i = \begin{cases} (\Delta \bar{s})^2 & \text{for } j = 1 \\ 0 & \text{for } j \geq 2 \end{cases}$$

Finally the corresponding load factor is computed for each iteration as follows:

$$\delta \lambda_j^i = \begin{cases} \pm \frac{\Delta \bar{s}}{\sqrt{\delta \mathbf{u}_{p1}^i \cdot \delta \mathbf{u}_{p1}^i + \eta}} & \text{for } j = 1 \\ -\frac{\delta \mathbf{u}_1^i \cdot \delta \mathbf{u}_{rj}^i}{\delta \mathbf{u}_1^i \cdot \delta \mathbf{u}_{pj}^i + \eta \delta \lambda_1^i} & \text{for } j \geq 2 \end{cases} \tag{18}$$

The sign of the load factor is not specified for the first iteration and it depends on whether the system is being loaded or unloaded. In order to choose the correct sign, many authors presented different techniques. In this work the technique introduced by Feng et al. (1995) is used where the sign of the load factor in the first iteration is determined by the sign of the determinant of the stiffness tangent matrix.

The arc-length method is able to reproduce complex nonlinear problems with load and displacement limit points, one of the weakness of the method is associated to the units of the terms in expression (18) where a scalar variable $\delta\lambda$ and the displacements $\delta\mathbf{u}$ are used. The displacement vector has as components, displacements and rotations which are of different orders of magnitude. This may cause that near displacement limit points the sign of $\delta\lambda$ changes incorrectly, causing divergence from the correct solution.

6 WORK CONTROL METHOD

The work control method (WCM) uses a constant work increment through the iterations of an increment. This method was introduced to overcome the unit problem in the arc-length method, that was mentioned in the previous section. The constraint equation of this control method is:

$$\delta W_j^i = \delta\lambda_j^i \bar{\mathbf{p}} \cdot \delta \mathbf{u}_j^i \quad (19)$$

The constraint parameters of this method are:

$$\begin{aligned} \mathbf{a}_j^i &= \delta\lambda_j^i \bar{\mathbf{p}} \\ b_j^i &= 0 \\ c_j^i &= \begin{cases} \Delta \bar{W} & \text{for } j = 1 \\ 0 & \text{for } j \geq 2 \end{cases} \end{aligned} \quad (20)$$

Where $\Delta \bar{W}$, is the prescribed work increment. The corresponding load factor can be computed as follows:

$$\delta\lambda_j^i = \begin{cases} \pm \sqrt{\frac{\Delta \bar{W}}{|\bar{\mathbf{p}} \cdot \delta \mathbf{u}_{p1}^i|}} & \text{for } j = 1 \\ -\frac{\bar{\mathbf{p}} \cdot \delta \mathbf{u}_{rj}^i}{\bar{\mathbf{p}} \cdot \delta \mathbf{u}_{pj}^i} & \text{for } j \geq 2 \end{cases} \quad (21)$$

The sign of the load factor in the first iteration is easily determined by the sign of the term inside the square root in expression (21).

The work control method, as the arc-length control method, is able to trace the structural response in complex nonlinear problems with load and displacement limit points. A potential weakness occurs in small systems when the displacement associated with the major forcing direction tends to snap-back. In that case $\bar{\mathbf{p}} \cdot \delta \mathbf{u}_{pj}^i$ will tend to zero and the load factor will tend to infinity.

7 APPLICATION EXAMPLES

Two simple numerical tests are presented in this section in order to check the implementation and capabilities of each method. The first is a one dimensional structure known as the Von Mises

truss in which, the analytic solution of the structure has two load limits points. The second numerical test is a two dimensional highly nonlinear problem with no physical interpretation, which has several load and displacement limit points.

7.1 One dimensional test - Von Mises truss

In order to solve the problem presented in Figure (8), the small perturbation hypothesis is made. The numerical problem is posed as the minimization of the potential energy, being the equilibrium equation, the first derivative of the energy and the tangent stiffness being, the second derivative of the energy. Both are presented in expressions (22) and (23).

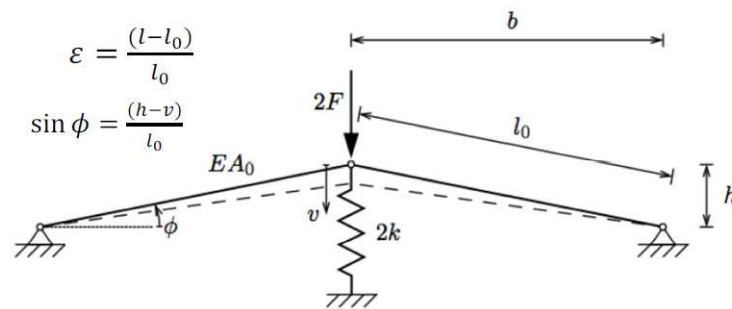


Figure 8: One dimensional Von Mises Truss (WCM)

$$E'(v, F) = EA \sin \phi \epsilon - v k + F \quad (22)$$

$$E''(v) = -\frac{EA}{l} \sin^2 \phi - \frac{EA}{l_0} \epsilon - k \quad (23)$$

The problem was solved with the different nonlinear solvers implemented. Figure (9) shows the numerical structural response and Table (1) summarizes the results of the different schemes.

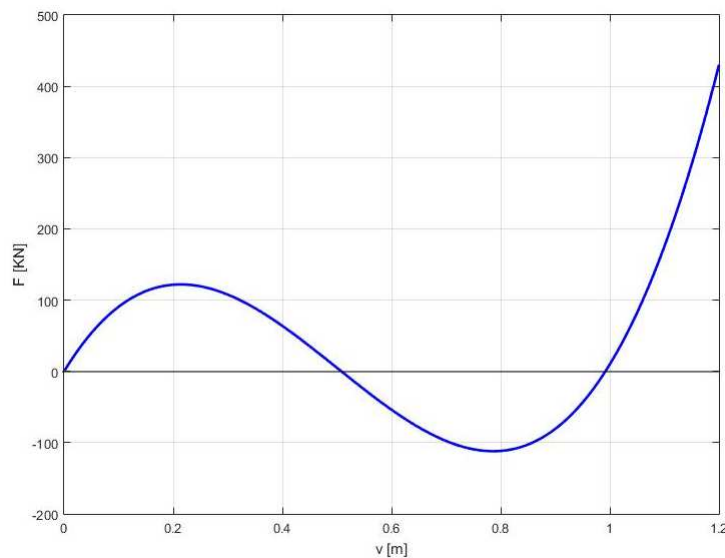


Figure 9: Response Von Mises Truss

Method	Control Factor	Steps	Behavior
LCM (NR)	0.1	-	Snaps Through
DCM	0.1	120	Fully Converged
ALCM	0.01	120	Fully Converged
WCM	0.15	75	Fully Converge

Table 1: Results for different control methods - Von Mises Truss

It can be seen in Fig.(9) that the structural response has two load limits point. At the first point the load control method, (classical Newton-Raphson), snaps directly to the hardening branch losing the softening response. The rest of the control methods are able to trace the full response of the structure. In this case the work control method reproduces accurately the response with the fewest number of steps being the most suitable for this particular problem.

7.2 Two dimensional numerical test

A pure numerical two dimensional test with load and displacement limit points in both degrees of freedom is carried out using the following vectors of external and internal applied “forces”.

$$\mathbf{p} = \begin{pmatrix} 40 \\ 15 \end{pmatrix} \quad (24)$$

$$\mathbf{q}(\mathbf{u}) = \begin{pmatrix} 10u_1 + 0.4u_2^3 - 5u_2^2 \\ 0.4u_1^3 - 3u_1^2 + 10u_2 \end{pmatrix}$$

The derivative of the internal forces vector with respect to the degrees of freedom gives the components of the tangent stiffness matrix.

$$\mathbf{K}(\mathbf{u}) = \begin{bmatrix} 10 & 1.2u_2^2 - 10u_2 \\ 1.2u_1^2 - 6u_1 & 10 \end{bmatrix} \quad (25)$$

Figure (10) shows the equilibrium paths of both degrees of freedom traced with the implemented code and also Table (2) summarizes the presented results.

This example shows a highly nonlinear problem with both load and displacement limit points. The LCM and DCM fails to reproduce the structural response after the first limit point. The ALCM and the WCM prove to be able to fully trace the path. For this particular problem the ALCM needs fewer steps than the WCM to trace accurately the response, being the ALCM the most suitable for this problem.

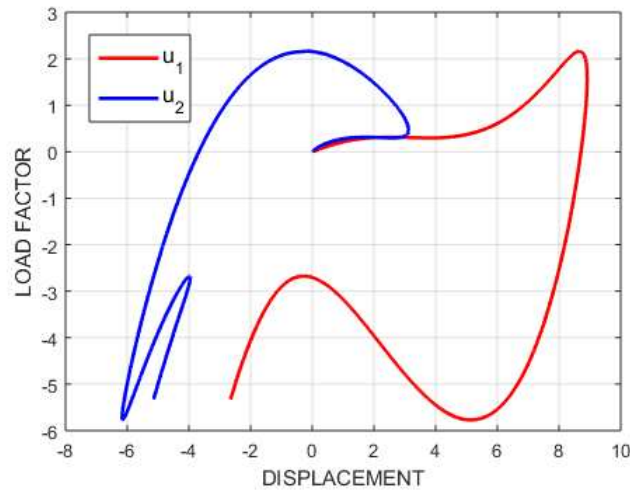


Figure 10: Response two dimensional test (ALCM)

Method	Control Factor	Steps	Behavior
LCM (NR)	0.01	-	Fails at load limit point
DCM	0.01	-	Fails at displacement limit point
ALCM	0.1	350	Fully Converged
WCM	0.0015	2600	Fully Converge

Table 2: Results for different control methods - Two dimensional test

8 CONCLUSIONS

The unified library of nonlinear solvers, developed by Leon et al. (2011), proves to be able to include several control methods in a unified manner. The library can be easily extended to include different solution schemes. The object oriented coding permits the implementation of new control methods by writing just the computation of the load factor in terms of the parameters of the constraint equation.

The library allows the user to choose the control method that best suits his nonlinear problem. Switching between methods it is also possible during a nonlinear analysis, so smooth parts of a response can be traced by simple methods as LCM or DCM and highly nonlinear branches can be traced with more complex methods as with different versions of the ALCM or WCM.

Numerical examples showed the limitations and capabilities of the different solution schemes implemented. The choice of the most suitable control method depends on the nonlinear problem. In the first example the WCM needed fewer steps than the other methods, but in the second test the ALCM showed a faster convergence than the WCM.

Further work can be done including other control methods as the Generalized Displacement Control Method and the Orthogonal Residual Procedure.

REFERENCES

- Bathe K.J. and Dvorkin E.N. On the automatic solution of nonlinear finite element equations. *Computer and Structures*, 17(5-6):871–879, 1983.
- Crisfield M. An arc-length method including line searches and accelerations. *International Journal for Numerical Methods in Engineering*, 19(9):1269–1289, 1983.
- Feng Y., Peric D., and Owen D. Determination of travel directions in path-following methods. *Mathl. Comput. Modelling*, 21(7):43–59, 1995.
- Krenk S. An orthogonal residual procedure for non-linear finite element equations. *International Journal for Numerical Methods in Engineering*, 38(5):823–839, 1995.
- Lam W.F. and Morley C.T. Arc-length method for passing limit points in structural calculation. *Journal of Structural Engineering*, 118(1):169–185, 1992.
- Leon S. *A Unified Library of Nonlinear Solutions Schemes: An Excursion Into Nonlinear Computational Mechanics*. Master's Thesis, University of Illinois at Urbana-Champaign, 2010.
- Leon S., Paulino G., Pereira A., Menezes I., and Lages E. A unified library of nonlinear solution schemes. *Applied Mechanics Review*, 64, 2011.
- Ritto-Correa M. and Camotim D. On the arc-length and other quadratic control methods: Established, less known and new implementation procedures. *Computer and Structures*, 86(11-12):1353–1368, 2008.
- Souza Neto E.A. and Feng Y.T. On the determination of the path direction for arc-length methods in the presence of bifurcations and snap-backs. *Comput. Methods Appl. Mech. Engrg*, 179:81–89, 1999.
- Yang Y.B. and Shieh M.S. Solution method for nonlinear problems with multiple critical points. *Journal of the American Institute of Aeronautics and Astronautics*, 28(12):2110–2116, 1990.