6 th International Symposium on Solid Mechanics

Select the box to select the Mini-Symposia:

[ ] Multiaxial and Fretting Fatigue (Edgar Mamiya and Lucival Malcher);

[ ] Composite Materials (Ricardo de Medeiros);

[ ] Constitutive Models I - Plasticity and Damage (Lucival Malcher, Miguel Vaz Jr. and Edgar Mamiya);

[ ] Constitutive Models II - Viscoelastic and Viscoplastic Solids: Models and Applications (Heraldo Mattos);

[ ] Impact Engineering (Marcílio Alves);

[ ] Structural Reliability Methods and Reliability-Based Design Optimization (André Beck);

[ ] Topology Optimization of Multifunctional Materials, Fluids and Structures (Eduardo Lenz Cardoso e Emilio Carlos Nelli Silva);

[ ] Topological Derivatives: Theory and Applications (André Novotny);

[ ] X-FEM, G-FEM and Meshfree Methods (Roberto Dalledone Machado and Rodrigo Rossi);

[ ] High Order Finite Elements (Marco Lúcio Bittencourt);

[X] Nonlinear Analyses (Eduardo Campello);

[ ] Other section;

MECSOL Joinville 2017
April 26-28, 2017
Joinville, Santa Catarina, Brazil

Computational Implementation of the Arc-Length Control Method Based on the Rates of Energy for Physically Nonlinear Analysis

Christian Frederic Jean
Structural Engineering Department
Federal University of Minas Gerais
christian@eng-estrut.mest.ufmg.br

Roque Luiz da Silva Pitangueira
Structural Engineering Department
Federal University of Minas Gerais
roque@dees.ufmg.br

Jamile Salim Fuina
Engineering and Architecture Department
FUMEC University
jamile@fumec.br

ABSTRACT

This paper presents the computational implementation of the arc-length control method based on the rates of the internal and the dissipated energy for physically nonlinear analysis. This implementation has been fulfilled in the numerical core of the INSANE (INteractive Structural ANalysis Environment), an open source system developed at the Structural Engineering Department of the Federal University of Minas Gerais. In nonlinear problems, most of the procedures for solving equilibrium equations are based on the iterative process of Newton-Raphson in which control methods are linked. The idea of these methods is to consider the loading parameter as a variable and to set a restraint condition into the system of equilibrium equations, in such a way that this parameter can be evaluated. In order to obtain a complete equilibrium path an arc-length method is often required due to the occurence of snap-through and snap-back phenomena. In this paper, an introduction to nonlinear analysis and the Newton-Raphson method responsible for the solution of incremental-iterative problems are presented. Next, the organization of the solution module of INSANE and the arc-length control method based on the rates of energy are presented. Finally, a comparative analysis with the classical control methods is presented along with general conclusions about the efficiency and the applicability of the energy rate control method.

Keywords: Arc-Length Control, Energy Rate, Incremental-Iterative Solutions, Physically Nonlinear Analysis

1. INTRODUCTION

This work presents the computational implementation of the arc-length control method based on the rates of the internal and the dissipated energy for physically nonlinear analysis. The nonlinear behavior of a material is hardly intuitive and can be fully described by an equilibrium path, which highlights the importance of control methods that are appropriate for its representation.

Given a random point on an equilibrium path defined by a displacement field (U) associated to a load factor (λ), the obtaining of other subsequent equilibrium points is accomplished using a incremental-iterative procedure in order to control one or more variables of the problem. The methods employed for this control are called nonlinear analysis control methods. The most frequently used methods are the well-known direct displacement control [1], the generalized displacement control [2] and the important class of methods known as the constant arc-length control which was first introduced by Ricks [3, 4]. New approaches of the arc-length control were then proposed by Ramm [5] and Crisfield [6, 7]. The choice of a proper control method for a random analysis must be made carefully as control methods can have certain limitations while passing through critical points. For example, the load control method is only able to trace the ascending branch of the loading path. The direct displacement control, on the other hand, is only effective for snap-through occurrences and fails whenever snap-back is encoutered.

In order to accurately obtain the representation of the equilibrium path, the algorithms must be able to detect the occurrence of limit points during the entire analysis. The arc-length control methods turns out to be indispensable in most situations due to their efficiency to handle multiple snap-through and snap-back phenomena.

In this paper, the new formulation proposed by de Borst, May e Vignollet [8] which switches between the internal and the dissipated energy was implemented and tested. This fully-energy based control method requires only two parameters in order to trace an complete equilibrium path and is especially robust for handling critical equilibrium points. In the following sections, the formulation and the computational implementation of the energy-based arc-length control method are presented. Next, the control method is applied to two different discrete models. A comparative analysis with the classical control methods is presented along with concluding remarks about the efficiency and the applicability of the discussed method.

2. INCREMENTAL-ITERATIVE METHOD FOR SOLVING NONLINEAR EQUATIONS

In a nonlinear analysis, one is confronted with the problem of solving an $n+1$ system equations, i.e. n equations of equilibrium and one constraint equation, for the $n+1$ system parameters, i.e. n displacements and one load parameter. To obtain this kind of solution, the use of an incremental-iterative procedure becomes necessary.

According to Fuina [9], the Newton-Raphson method, which is the main incremental-iterative procedure used for nonlinear problems, presupposes that the tangent stiffness matrix of the structure is recalculated at every iteration. In this case, the incremental equilibrium equation (1) for the *j*th iteration of the *i*th increment can be written as:

$$[K]^i_{j-1} \cdot \{\delta U\}^i_j = \delta\lambda^i_j \cdot \{P\} + \{Q\}^i_{j-1} \tag{1}$$

where $[K]^i_{j-1}$ is the tangent stiffness matrix which refers to the displacements vector $\{U\}^i_{j-1}$, $\{\delta U\}^i_j$ the displacement increment vector, $\delta\lambda^i_j$ the load increment parameter, $\{P\}$ the reference load vector, and $\{Q\}^i_{j-1}$ the unbalanced load vector.

Firstly a value for the load increment parameter is established which relies on the adopted control function. The displacement increment vector can then be obtained and conveniently replaced by the following:

$$\{\delta U\}_j = \delta\lambda^i_j \cdot \{\delta U^P\}_j + \{\delta U^Q\}_j \tag{2}$$

with

$$[K]_{j-1} \cdot \{\delta U^P\}_j = \{P\} \tag{3}$$

and

$$[K]_{j-1} \cdot \{\delta U^Q\}_j = \{Q\}_{j-1}. \tag{4}$$

At the end of every iteration, the convergence is verified through the size of the residual force vector and/or displacement increment vector. The procedure continues until the desired accuracy is obtained. It's worth pointing out that the load increment parameter must be obtained using a suitable constraint equation. According to Yang and Shieh [2], the general form of the constraint equation is based on the bounded nature of the loading parameter and is written as in equation 5:

$$H_j^i = \{C\}^T \{\delta U\}_j^i + k\delta\lambda_j^i, \tag{5}$$

where $\{C\}$ and k are constants and H_j^i is a parameter of the constraint equation.

In a multidimensional space, the relation between the load increment parameter and the constraint equation can then be represented by equation 6:

$$\delta\lambda_j^i = \frac{1}{\{C\}^T \cdot \{\delta U^P\}_j^i + k}(H_j^i - \{C\}^T \cdot \{\delta U^Q\}_j^i). \tag{6}$$

The updating of variables is fulfilled in accordance with equations 7 and 8:

$$\lambda_j = \lambda_{j-1} + \delta\lambda_j \tag{7}$$

$$\{U\}_j = \{U\}_{j-1} + \{\delta U\}_j. \tag{8}$$

The unbalanced force vector of the jth iteration is given by:

$$\{Q\}_j = \lambda_j \cdot \{P\} - \{F\}_j, \tag{9}$$

where $\{F\}_j$ is the internal force vector in the end of the jth iteration. In the first iteration of a step, the unbalanced load vector $\{Q\}_{j-1}$ is zero.

The diagram shown in Figure 1 illustrates the generic algorithm proposed by Yang and Shieh [2] which represents the Newton-Raphson procedure for solving nonlinear equations. The highlighted procedure refers to the obtaining of the load increment parameter $\delta\lambda_j^i$, which depends on the selected control method.

This formulation is quite generic and can be applied to various control methods simply by redefining the constraint equation.

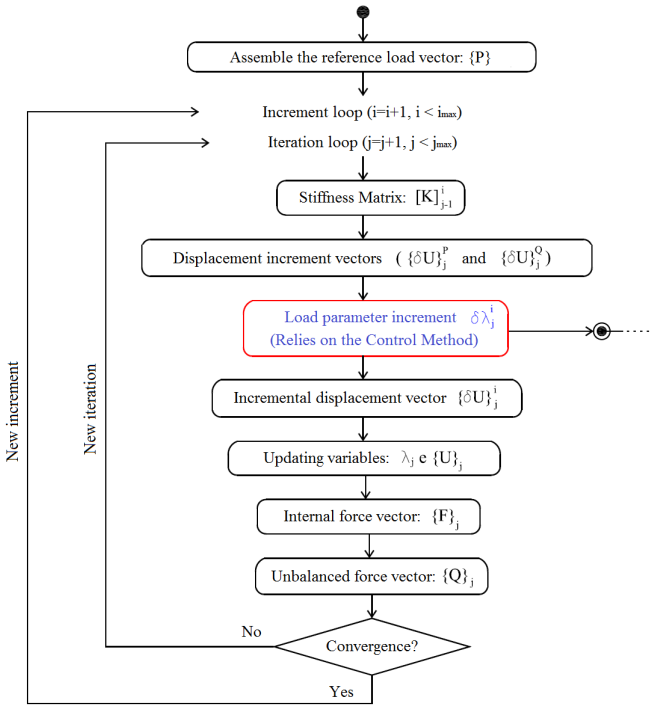

Figure 1. Algorithm for control methods (adapted from Fuina [9]).

3. ENERGY RATE CONTROL METHOD

The idea of a control method based on the rates of energy is motivated by the fact that under external forces the amount of dissipated energy can only increase monotonically. This fact has been exploited in the researches of Gutierrez [10] and has been applied to damage models. The idea was then pursued and improved by Verhoosel [11] to plasticity models and geometrically nonlinear problems with damage.

The energy-based arc-length control [8] switches between internal energy and dissipated energy, being entirely based on energy. This method allows one to obtain an equilibrium path using only two parameters and is robust handling multiple snap-through and snap-back phenomena.

3.1 Arc-Length Control based on the Rates of Dissipated Energy

This control method has been introduced in Gutierrez [10]. The procedure uses a load control at the beginning of the loading phase, and switches to an arc-length control based on the dissipation as soon as the dissipated energy reaches a fixed limit.

The dissipative conduct of a process can be detailed by the second law of thermodynamics. According to Jirasek and Bazant [12], in a local form for a steady temperature, the second law of thermodynamics can be written as in equation 10.

$$\dot{\mathscr{D}} = \sigma_{ij}\dot{\varepsilon}_{ij} - \dot{\psi} \geq 0, \tag{10}$$

where $\dot{\mathscr{D}}$ is the dissipation and ψ is the density of energy.

A damage law for the constitutive response between the stress and the strain is then adopted as in equation 11.

$$\sigma_{ij}(\varepsilon_{ij},d) = g(d)C_{ijkl}\varepsilon_{kl}, \tag{11}$$

where $d \in [0,1]$ is the damage parameter of the degradation function $g(d)$ and C_{ijkl} is the fourth order symmetric elasticity tensor.

Therefore, the energy density can be written as in equation 12:

$$\psi(\varepsilon_{ij},d) = \frac{1}{2}\sigma_{ij}(\varepsilon_{ij},d)\varepsilon_{ij} \tag{12}$$

so that:

$$\frac{\partial\psi}{\partial\varepsilon_{kl}} = \frac{1}{2}\frac{\partial\sigma_{ij}}{\partial\varepsilon_{kl}}\varepsilon_{ij} + \frac{1}{2}\frac{\partial\varepsilon_{ij}}{\partial\varepsilon_{kl}}\sigma_{ij} = \frac{1}{2}g(d)C_{ijkl}\varepsilon_{ij} + \frac{1}{2}g(d)C_{ijkl}\varepsilon_{kl}\delta_{ik}\delta_{jl} = g(d)C_{ijkl}\varepsilon_{ij} = \sigma_{kl}. \tag{13}$$

The elaboration of the time derivative of the energy density ψ in equation 10 can be performed in two ways. The first option is using the chain rule with equation 13 as presented in equation 14. The dissipation is then given by equation 15:

$$\dot{\psi} = \frac{\partial\psi}{\partial\varepsilon_{ij}}\dot{\varepsilon}_{ij} + \frac{\partial\psi}{\partial d}\dot{d} = \sigma_{ij}\dot{\varepsilon}_{ij} + \frac{\partial\psi}{\partial d}\dot{d} \tag{14}$$

$$\dot{\mathscr{D}} = -\frac{\partial\psi}{\partial d}\dot{d} \geq 0. \tag{15}$$

The second choice is to apply the product rule as in equation 16, which leads into equation 17:

$$\dot{\psi} = \frac{1}{2}\dot{\sigma}_{ij}\varepsilon_{ij} + \frac{1}{2}\sigma_{ij}\dot{\varepsilon}_{ij} \tag{16}$$

$$\dot{\mathscr{D}} = \frac{1}{2}\sigma_{ij}\dot{\varepsilon}_{ij} - \frac{1}{2}\dot{\sigma}_{ij}\varepsilon_{ij} \geq 0. \tag{17}$$

Presuming that there are no discontinuities in the body, the global forms of equations 15 and 17 can be written as in equation 18:

$$\dot{\tau}^D = \int_\Omega \dot{\mathscr{D}}\, dV = \int_\Omega (\frac{1}{2}\sigma_{ij}\dot{\varepsilon}_{ij} - \frac{1}{2}\dot{\sigma}_{ij}\varepsilon_{ij})\, dV = \int_\Omega -\frac{\partial\psi}{\partial d}\dot{d}\, dV, \tag{18}$$

where $\dot{\tau}^D$ is the dissipated energy rate.

It can be noticed that the dissipated energy τ^D increases monotonically, since $\dot{\tau}^D \geq 0$ is obtained from $\dot{d} \geq 0$ and $\partial\psi/\partial d \leq 0$.

One can then represent the dissipated energy rate according to the internal and external forces using $\varepsilon = [B]\{U\}$ as in equation 19:

$$\dot{\tau}^D = \int_\Omega \frac{1}{2}\{\dot{U}\}^T[B]^T\{\sigma\}\,dV - \int_\Omega \frac{1}{2}\{U\}^T[B]^T\{\dot{\sigma}\}\,dV = \frac{1}{2}\{\dot{U}\}^T\cdot\{f\}^{int}(U) - \frac{1}{2}\{U\}^T\cdot\{\dot{f}\}^{int}(U)$$
$$= \frac{1}{2}\{\dot{U}\}^T\cdot\lambda\{P\} - \frac{1}{2}\{U\}^T\cdot\dot{\lambda}\{P\}. \tag{19}$$

$\dot{\tau}^D$ can be used as a path parameter rate, leading to equation 20.

$$\frac{1}{2}(\lambda\{\dot{U}\}^T - \dot{\lambda}\{U\}^T)\cdot\{P\} - \dot{\tau}^D = 0, \tag{20}$$

which can be discretized as in equation 21:

$$\frac{1}{2}(\lambda^{i-1}\{U\}^{i^T} - \lambda^i\{U\}^{i-1^T})\cdot\{P\} - \delta\tau^D = 0. \tag{21}$$

Equation 21 can be used as the constraint equation $\varphi(U,\lambda)$ of a problem as following:

$$\varphi^D(U,\lambda) = \frac{1}{2}(\lambda^{i-1}\{U\}^T - \lambda\{U\}^{i-1^T})\cdot\{P\} - \delta\tau^D. \tag{22}$$

Equation 22 can now be formulated in accordance with equation 6 with the parameter H_j^i being the energy increment $\delta\tau^D$; i.e. the amount of energy which has to be dissipated in one increment.

In the first iteration of each step $(j = 1)$, the dissipated energy rate is incremented of a constant value. For the iterative steps $(j > 1)$, the dissipated energy rate is zero in order to ensure the adopted criteria. Thus, the value of the load factor increment $\delta\lambda_j^i$ can be calculated by equations 23 and 24:

$$\delta\lambda_j^i = \frac{2\delta\tau^D}{\lambda^i\{\delta U^P\}_j^{i^T}\cdot\{\widehat{f}\} - \{U\}^{i^T}\cdot\{\widehat{f}\}}, \quad for\ \ j = 1. \tag{23}$$

$$\delta\lambda_j^i = -\frac{\lambda^{i-1}\{\delta U^Q\}_j^i\cdot\{\widehat{f}\}}{\lambda^{i-1}\{\delta U^P\}_j^i\cdot\{\widehat{f}\} - \{U\}^{i-1^T}\cdot\{\widehat{f}\}}, \quad for\ \ j > 1. \tag{24}$$

It is important to note that the control is based on the energy that is dissipated during failure. Thus, it presents poor efficiency when the process does not show a significant amount of energy dissipation.

3.2 Arc-Length Control based on the Rates of Internal Energy

According to de Borst, May e Vignollet [8], this control function is specific for the regime when the dissipated energy rate due to the evolution of the damage variable is small, e.g., in the beginning

of the loading fase. Still presuming that there are no discontinuities in the body, equation 16 can be adapted in its global form to obtain the internal energy rate $\dot{\tau}^U$ in the matrix format:

$$\dot{\tau}^U = \int_\Omega \psi \, dV = \int_\Omega \frac{1}{2}\{\dot{U}\}^T [B]^T \{\sigma\} \, dV + \int_\Omega \frac{1}{2}\{U\}^T [B]^T \{\dot{\sigma}\} \, dV = \frac{1}{2}(\{\dot{U}\}^T \lambda + \{U\}^T \dot{\lambda}) \cdot \{P\}. \quad (25)$$

Applying the midpoint rule, equation 25 can be discretized as:

$$\frac{1}{2}(\lambda^i \{U\}^{i^T} - \lambda^{i-1} \{U\}^{i-1^T}) \cdot \{P\} - \delta\tau^U = 0 \quad (26)$$

which also can be used as constrait equation $\varphi(U, \lambda)$ as shown in equation 27.

$$\varphi^U(U, \lambda) = \frac{1}{2}(\lambda^i \{U\}^{i^T} - \lambda^{i-1} \{U\}^{i-1^T}) \cdot \{P\} - \delta\tau^U. \quad (27)$$

Analogously to the dissipated energy-based control, the parameter $\delta\tau^U$ in equation 27 can be interpreted as a prescribed step size for one increment; it prescibes the amount of internal energy which has to be introduced into the process in one increment.

In the first iteration $(j = 1)$ of the first step $(i = 1)$:

$$\left.\frac{\partial \varphi^U(U, \lambda)}{\partial \lambda}\right|_{j=1}^{i=1} = \left.\frac{1}{2}\{U\}^{i^T} \cdot \{P\}\right|_{j=1}^{i=1} = \frac{1}{2}\{U\}_1^{1^T} \cdot \{P\} = 0 \quad (28)$$

where $\{U\}_1^{1^T} = \{U\}_0 = 0$.

Thus, for the first increment, the arc-length control function reduces to a formula equivalent to load control, as in equation 29:

$$\varphi_1^F(\lambda) = \lambda - \delta\tau_1^F, \qquad \left.\frac{\partial \varphi_1^F(\lambda)}{\partial \lambda}\right|^{i=1} = 1. \quad (29)$$

After the first increment, the solution for $\{U\}_1$ and λ_1 is known and the internal energy rate $\delta\tau_1^U$ can then be obtained using equation 27 as follows:

$$\Delta\tau_1^U = \frac{1}{2}(\lambda_1 \{U\}_1^T - \lambda_0 \{U\}_0^T) \cdot \{P\} = \frac{1}{2}\lambda_1 \{U\}_1^T \cdot \{P\}. \quad (30)$$

Equation 30 gives $\delta\tau_1^U$ which will be used as the prescribed energy rate in the subsequent increments. For these increments, the value of $\delta\lambda_j^i$ can be written as in equations 31 and 32, for the incremental steps and iterative steps, respectively.

$$\delta\lambda_j^i = \frac{2\delta\tau^U}{\lambda^i \{\delta U^P\}_j^{i^T} \cdot \{\widehat{f}\} + \{U\}^{i^T} \cdot \{\widehat{f}\}}, \quad for \;\; j = 1. \quad (31)$$

$$\delta\lambda_j^i = -\frac{\lambda^{i-1} \{\delta U^Q\}_j^i \cdot \{\widehat{f}\}}{\lambda^{i-1} \{\delta U^P\}_j^i \cdot \{\widehat{f}\} + \{U\}^{i-1^T} \cdot \{\widehat{f}\} + \{\delta U^Q\}_j^i \cdot \{\widehat{f}\}}, \quad for \;\; j > 1. \quad (32)$$

4. COMPUTATIONAL IMPLEMENTATION

The previously presented arc-length control was incorporated into the project INSANE (INteractive Structural ANalysis Environment), an open source system developed at the Structural Engineering Department of the Federal University of Minas Gerais, which is avaliable at `http://insane.dees.ufmg.br`. This system has a numerical core that is able to solve discrete models of structural analysis. It also has resources to solve nonlinear problems which uses the Newton-Raphson method and a variety of control methods.

4.1 The INSANE Solution Module

The superclasses that represents the numerical core of the INSANE system are the *Assembler*, *Model* and *Persistence* interfaces and the abstract class named *Solution*. The relationship between these superclasses is illustrated in Figure 2. To facilitate understanding, class diagrams using the *Unified Modeling Language* (UML) scheme are used in this section.

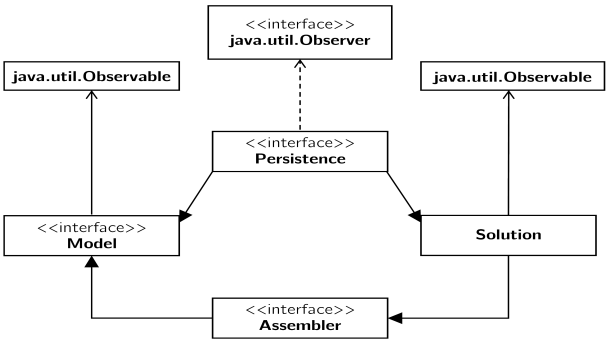

Figure 2. Organization of the numerical core of the INSANE system.

The *Persistence* interface is responsible for processing the incoming data and for persisting the output data whenever changes are made in the model. The *Assembler* interface is responsible for assembling the matrices and vectors of the model's equilibrium equation. The *Model* interface represents the finite element model and provides to *Assembler* the needed data to assemble the equilibrium equation, which will be solved by the *solution* module.

The main class of the *solution* module is the *Solution* abstract class (Figure 3). This class extends the *Observable* class, since its observed by the persistence. The class main method is the *execute()* method and is responsible for triggering the whole solution process.

The *SteadyState* class is the subclass of *Solution* that represents a linear static solution. The abstract class named *EquilibriumPath* is the subclass of *Solution* which generalizes a solution with the objective of determinating an equilibrium path.

The static nonlinear solution, subject of the present work, is represented by the *StaticEquilibriumPath* class, which implements an incremental-iterative procedure and uses a control method in order to obtain the equilibrium path. This class has an object of type *Step*, which has the needed methods to execute an incremental step and an object of type *IterativeStrategy*, which defines the control method that will be used during the analysis.

Step is an interface implemented by the *StandardNewtonRaphson* and *ModifiedNewtonRaphson* classes where, respectively, the incremental-iterative processes of Standard Newton Raphson and Modified Newton Raphson are used to obtain the convergence of an incremental step. These correlations are shown in Figure 4. Each of these classes has an object of *Assembler*, which provides the matrices and vectors of the equation to be solved; A *LinearEquationSystems* object, capable of solving the system of linear algebraic equations of each iteration; And an object *IterativeStrategy*, which

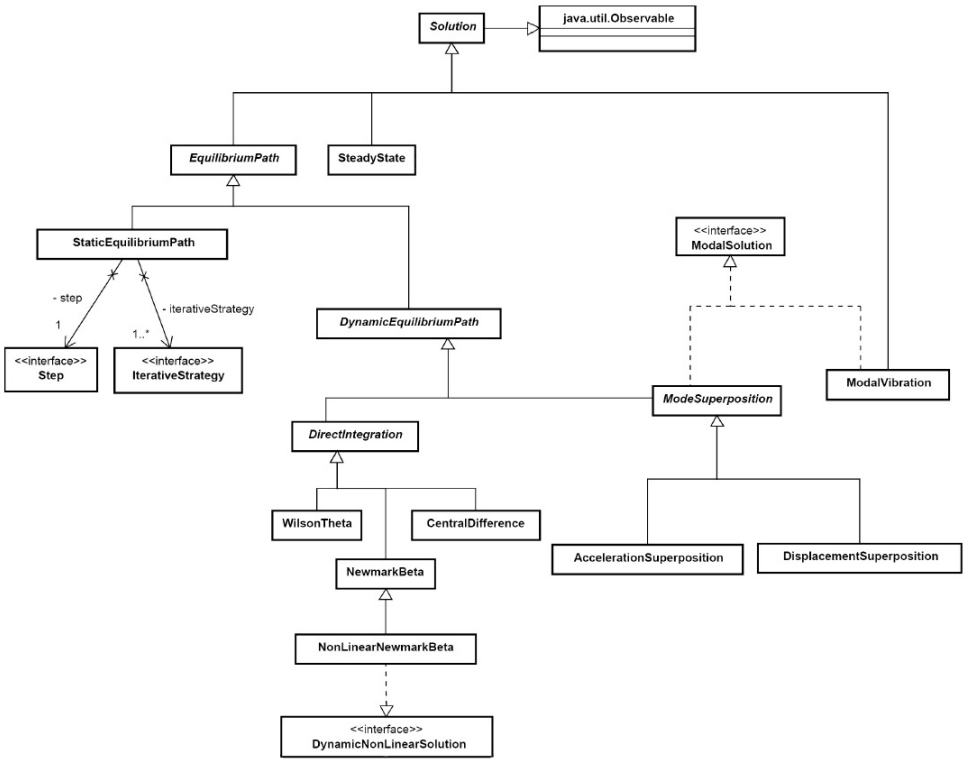

Figure 3. Diagram of the *Solution* class (adapted from Fuina [13]).

informs the adopted iteration strategy.

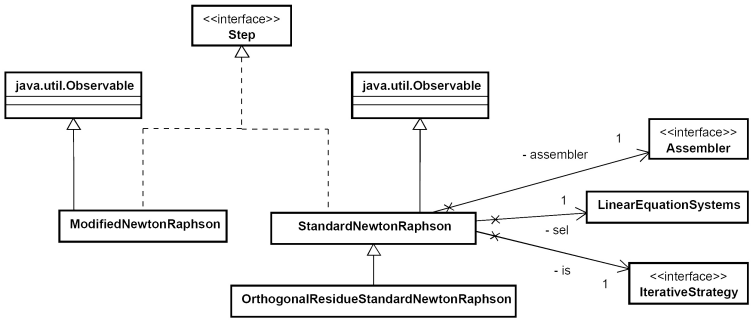

Figure 4. Inheritance and instance diagram of the *StandardNewtonRaphson* class [13].

The *IterativeStrategy* interface is implemented by classes that represents the nonlinear analysis control methods. In this work, a new class named *EnergyRateControl* was created in the solution module which represents the energy rate control method. The current implementations of *IterativeStrategy* on the INSANE system can be seen in Figure 5. A new method called *changeStrategy()* was introduced in the *IterativeStrategy* class and is invoked in *StandardNewtonRaphson* to provide the data needed to perform the switching between the dissipative and non-dissipative control functions.

4.2 *EnergyRateControl* Class

The classes that implement the nonlinear analysis control methods have the *getPredictor()* and *getCorrector()* methods, which calculate the load factor increment for the first and the subsequent iterations, respectively, according to the Yang and Shieh [2] algorithm. In the previous section, the constraint equations for the internal and dissipated energy rate-based control (equations 22 and 27) were formulated in accordance with the equation 6 with H_j^i being the energy rate increment. Equations 31 and 32 represents, for the internal energy control, the *getPredictor()* and *getCorrector()* methods,

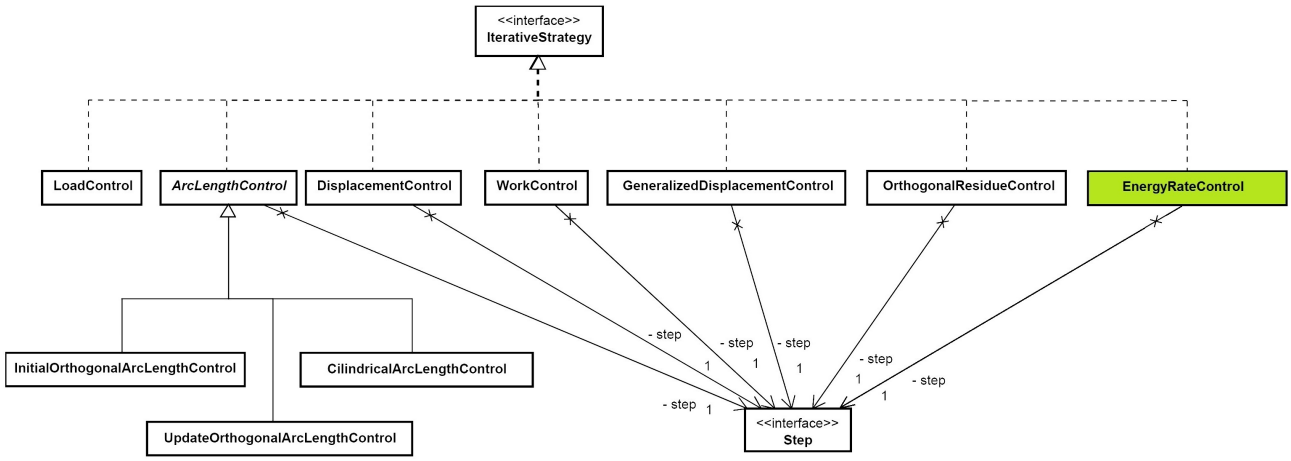

Figure 5. Implementation diagram of the *IterativeStrategy* interface.

respectively, and equations 23 and 24 represents the *getPredictor()* and *getCorrector()* methods for the dissipated energy control.

The swapping from one control function to another is achieved through the evaluation of the internal and dissipated energy rate of the structure. The Figure 6 illustrates where the change of method occurs. The simulation starts with the prescribed load factor $\delta\tau_1^F$, which gives the step size for the internal energy rate $\delta\tau_1^U$. $\delta\tau_1^U$ is then used for the subsequent step increments during the entire analysis. When the dissipated energy rate becomes higher than the internal energy rate, e.g., after point *B*, the process switches from the internal energy-based control to the dissipative energy-based control with $\delta\tau^D = \delta\tau_1^U$. The system's internal energy becomes negative between points *B* and *D*. At the time when the value of the internal energy increment $\delta\tau^U$ becomes again positive, the process switches back to the internal energy rate-based control with $\delta\tau^U = \delta\tau_1^U$.

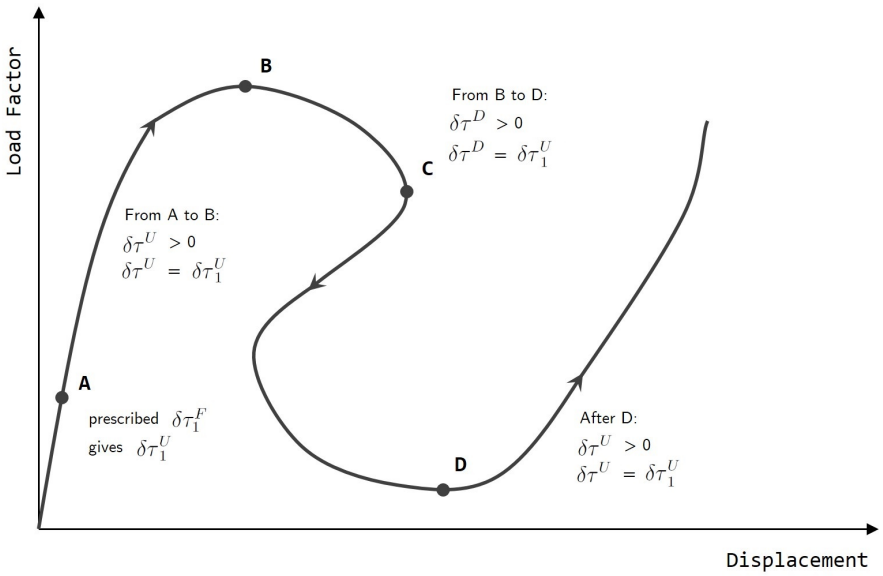

Figure 6. Obtaining equilibrium path using control method based on the rates of internal and dissipated energy (adapted from de Borst, May e Vignollet [8]).

5. NUMERICAL SIMULATIONS

This section has the purpose of evaluating the applicability of the arc-length control method based on the rates of energy. The results will then be compared with some of the most recommended classical control methods.

5.1 Direct Tension

Figure 7 shows the geometrical, load and restraint configurations for the direct tension example.

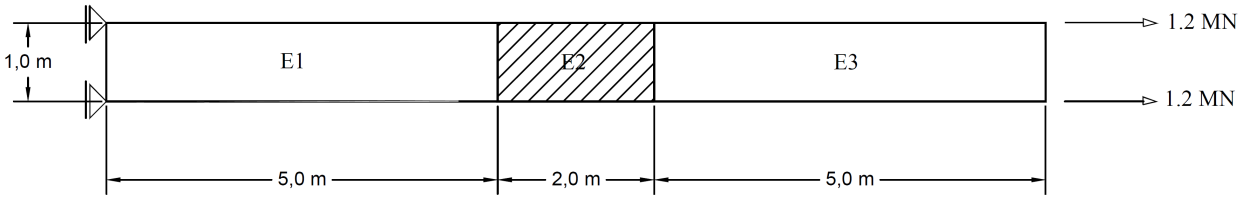

Figure 7. Direct tensile model.

The mesh used for this simulation is composed of three $Q4$ quadrilateral elements and the smeared crack constitutive model with law of Carreira and Chu [14] for tension was adopted. The material parameters for elements $E1$ and $E3$ are: $E_0 = 20000.0\ MPa$, $\nu = 0.2$, $f_t = 2.6\ MPa$, $\varepsilon_t = 0.0003$, $G_f = 0.000169\ MN/m$, length scale parameter $h = 0.1\ m$ and shear retention factor $\beta_r = 0,05$. The material parameters for element $E2$ are: $E_0 = 20000.0\ MPa$, $\nu = 0.2$, $f_t = 2.4\ MPa$, $\varepsilon_t = 0.0002$, $G_f = 0.000144\ MN/m$, $h = 0.1\ m$ and $\beta_r = 0,05$. The thickness of the structure is $1\ m$.

In order to obtain the solution of this model, the energy rate control method was used with an initial load factor of 10% of the maximum load and a tolerance for convergence of 1×10^{-4}. The load factor-displacement curve for the loaded face of the bar is shown on Figure 8.

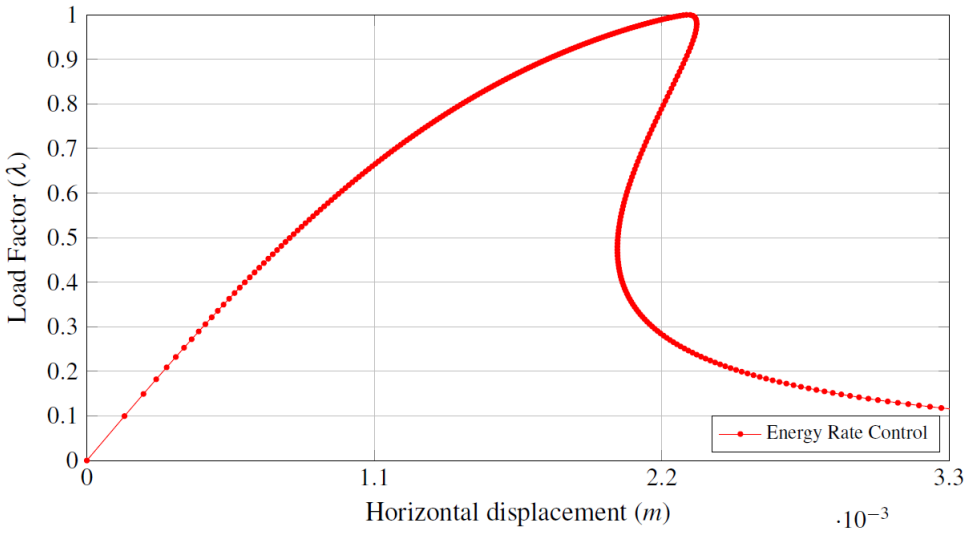

Figure 8. Equilibrium path response using the energy rate control.

The comparative analysis with the displacement control method, the cilindrical arc-length control method and the generalized displacement control method is shown on Figure 9.

It is observed that all methods were able to describe the post-critical behavior with ease, except for the displacement control method, which, as expected, failed to obtain convergence at the snap-back point and had its path interrupted. Furthermore, it can be seen that the energy rate method presents a smoother curve as it approaches the peak load, in relation to the classical methods.

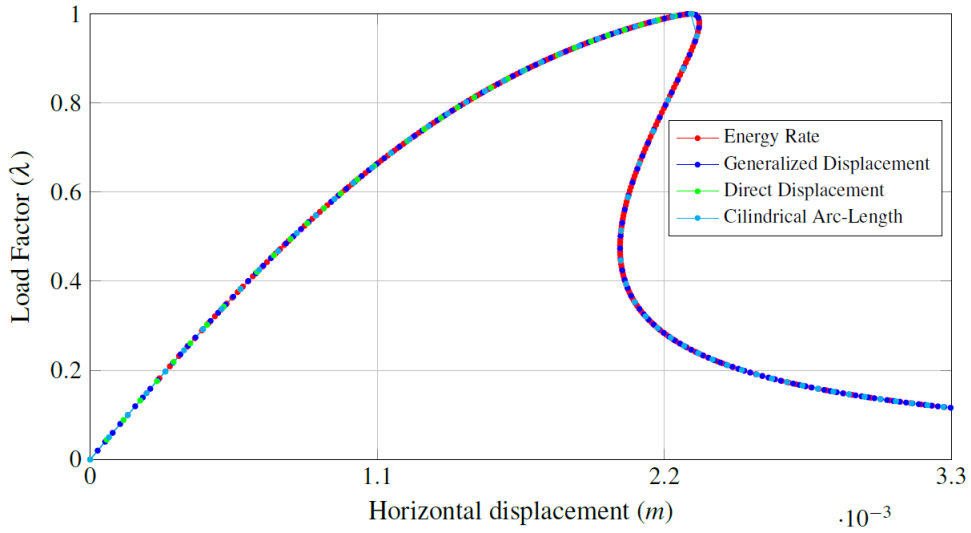

Figure 9. Equilibrium paths using the energy rate control and other classical methods.

5.2 Diametrical Compression

As a second numerical example, we consider a cylinder subject to the diametrical compression test as shown in Figure 10.

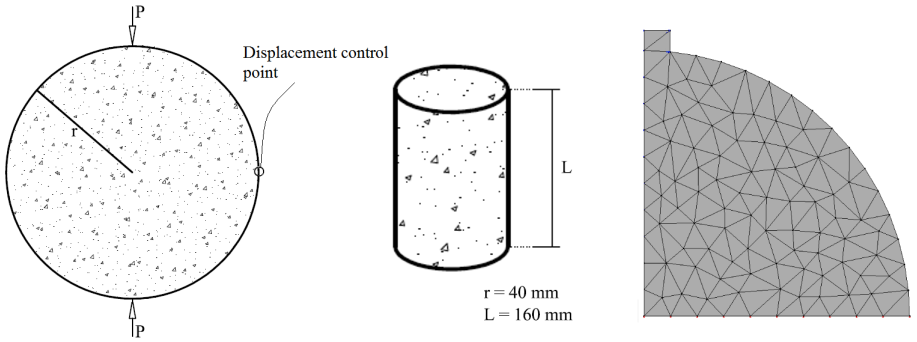

Figure 10. Diametrical compression: Configuration (Penna [15]).

In this analysis, the cross-section of the cylinder was modeled with 180 $T3$ triangular elements and plane stress was assumed. The constitutive model of smeared crack with concrete laws of Carreira and Chu [14], for compression, and Boone *et al.* [16], for tension, was adopted whose parameters are: $E_0 = 20000.0\ MPa$, $\nu = 0.2$, $f_c = 25.0\ MPa$, $f_t = 2.5\ MPa$, $\varepsilon_c = 0.01$, $G_f = 0.0001\ MN/m$, length scale parameter $h = 0.05\ m$ and shear retention factor $\beta_r = 0,05$.

The energy rate control was used with a reference load of $P = 0.5\ N$ and a tolerance for convergence of 1×10^{-3}. The load-displacements curves for the horizontal displacement of the control point and for the vertical displacement of the contact of the loading plate with the cylinder, respectively, are shown on Figures 11 and 12.

Figures 13 and 14 shows the comparative analysis of the 12 *mm* crack case with the classical control methods.

It can be observed in Figures 11-14 that bifurcations occurs in the descending branch for the 12 *mm* crack. These bifurcations, which were not successfully detected by the classical control methods, appear to be associated with the appearance of numerically induced strain localization.

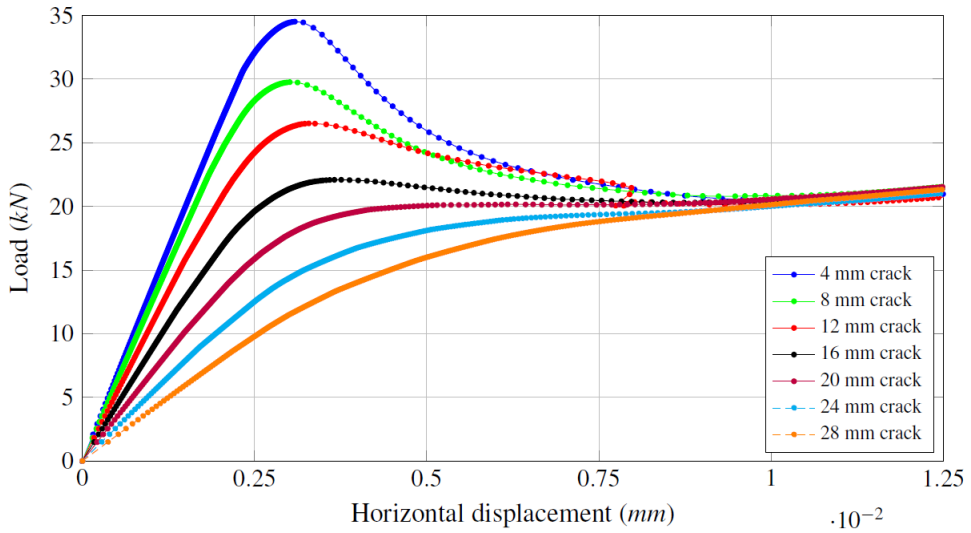

Figure 11. Equilibrium path of the horizontal displacement of the control point.

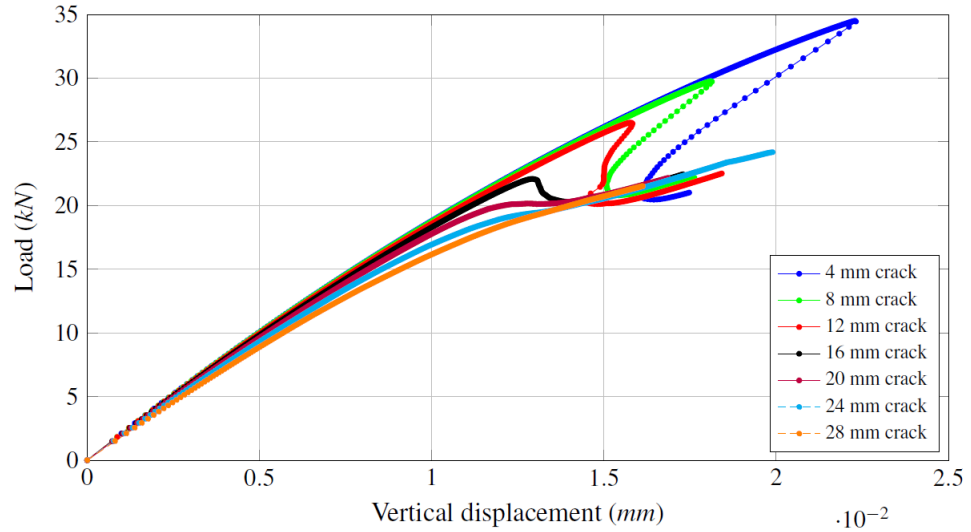

Figure 12. Equilibrium path of the vertical displacement of the contact of the plate with the cylinder.

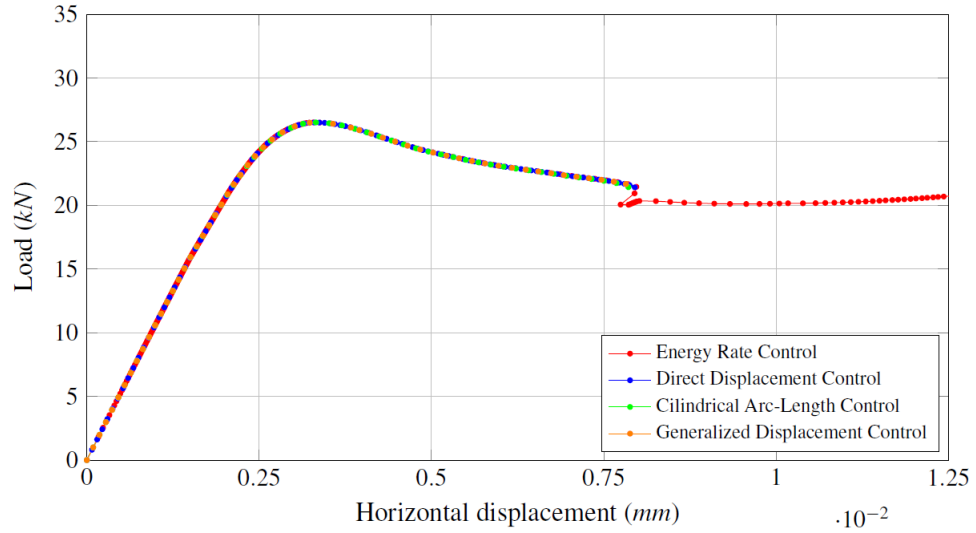

Figure 13. Horizontal displacement paths using the energy rate control and other classical methods.

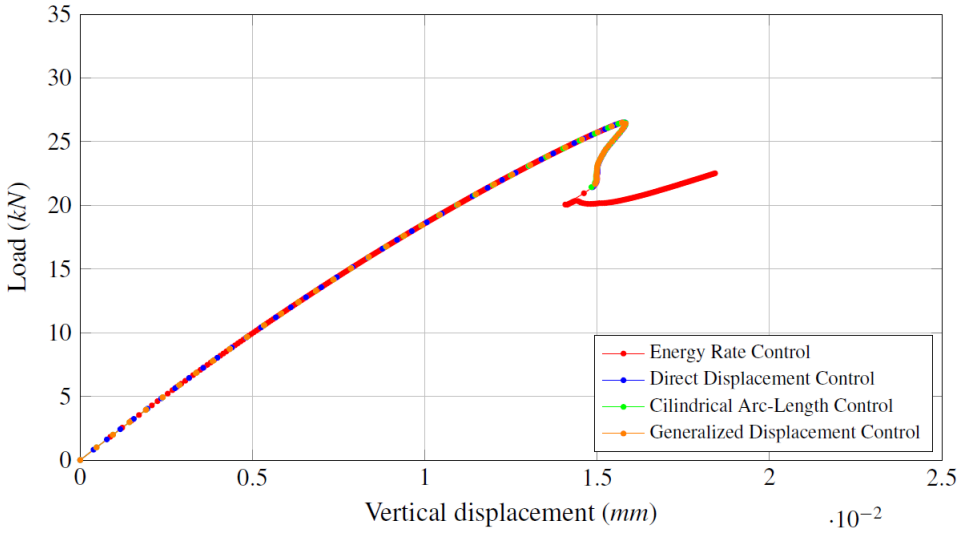

Figure 14. Vertical displacement paths using the energy rate control and other classical methods.

6. CONCLUSIONS

The use of an appropriate control method is essential in a numerical structural analysis. The arc-length control methods have become reference in the nonlinear analysis due to their efficiency to overcome the numerical instabilities caused by snap-through and snap-back phenomena found in equilibrium paths.

In this work, the energy-based arc-length control method proposed by de Borst, May e Vignollet [8] for non-linear analysis, which overcomes such phenomena, was implemented based on the Yang and Shieh [2] formulation. The method has shown to be superior for physically nonlinear analysis of brittle materials, since the accuracy of the response increases as it approaches the limit points. Furthermore, the method is simple to use and doesn't require prior knowledge of the behavior of the structure.

REFERENCES

[1] Batoz, J. L. and Dhat, G. (1979). "Incremental Displacement Algorithms for Nonlinear Problems." International Journal for Numerical Methods in Engineering, 14:1262–1267.

[2] Yang, Y. B. and Shieh, M. S. (1990). "Solution Method for NonLinear Problems with Multiple Critical Points." AIAA Journal, 28(12):2110–2116.

[3] Ricks, E. (1972). "The Application of Newton Method to the Problem of Elastic Stability." Journal of Applied Mechanics, pages 1060–1065.

[4] Ricks, E. (1979). "An Incremental Approach to the Solution of Snapping and Buckling Problems." International Journal of Solids and Structures, 15:529–551.

[5] Ramm, E. (1981). "Strategies for Tracing the Nonlinear Response Near Limit Points, in Nonlinear Finite Element Analysis in Structural Mechanics." pages 63–83.

[6] Crisfield, M. A. (1981). "A Fast Incremental-Iteractive Solution Procedure That Handles Snap-Through." Computers & Structures, 13:55–62.

[7] Crisfield, M. A. (1983). "An Arc Length Method Including Line Searches and Acelerations." International Journal for Numerical Methods in Engineering, 19:1269–1289.

[8] de Borst, R., May, S., and Vignollet, J. (2016). "A new arc-length control method based on the rates of the internal and the dissipated energy." Engineering Computations, 33:100–115.

[9] Fuina, J. S. (2004). Métodos de Controle de Deformações para Análise Não-Linear de Estruturas. Master's thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.

[10] Gutierrez, M. A. (2004). "Energy Release Control for Numerical Simulations of Failure in Quasi-Brittle Solids." Communications in Numerical Methods in Engineering, 20(1):19–29.

[11] Verhoosel, C. V., Remmers, J. J. C., and Gutierrez, M. A. (2009). "A Dissipation-Based Arc-Length Method for Robust Simulation of Brittle and Ductile Failure." International Journal for Numerical Methods in Engineering, 77(9):1290–1321.

[12] Jirasek, M. and Bazant, Z. P. (2001). Inelastic Analysis of Structures. Wiley, Chichester.

[13] Fuina, J. S. (2009). Formulações de Modelos Constitutivos de Microplanos para Contínuos Generalizados. Ph.D. thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.

[14] Carreira, D. J. and Chu, K. (1985). "Stress-strain relationship for plain concrete in compression." ACI Journal, 82:797–804.

[15] Penna, S. S. (2011). Formulação Multipotencial para Modelos de Degradação Elástica - Unificação Teórica, Proposta de Novo Modelo, Implementação Computacional e Modelagem de Estruturas de Concreto. Ph.D. thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.

[16] Boone, T., Wawrzynek, P. A., and Ingraffea, A. R. (1986). "Simulation of the fracture process in rock with application to hydrofracturing." International Journal of Rock Mechanics and Minig Science, 23 (3):255–265.

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.