

Study of large-deflection Beams using geometrically nonlinear C^0 Finite Elements

Fall 2018 Course Project for MECH 520 Nonlinear Finite Element Methods

Submitted by
Nidish Narayanaa Balaji
S01276643 (nb25)
Department of Mechanical Engineering
Rice University, Houston, TX 77005

December 1, 2018

1 Introduction

A large-deformation nonlinear formulation is used to implement a nonlinear finite element static & dynamic code. The only assumption that is made is that plane sections remain planar. Although invalid for beams of large cross-sections, this is justified for slender beams on a plane (no torsional and/or warping effects). Low amplitude responses are validated with linear analytical results for simple static cases. A low-amplitude modal analysis is conducted to verify the dynamic matrices. Although the code is inherently nonlinear, buckling under axial tip-load and snap-through response of buckled structures under transverse mid-point loads are the major nonlinear cases that are considered for response tracing and bifurcation analysis.

2 Formulation

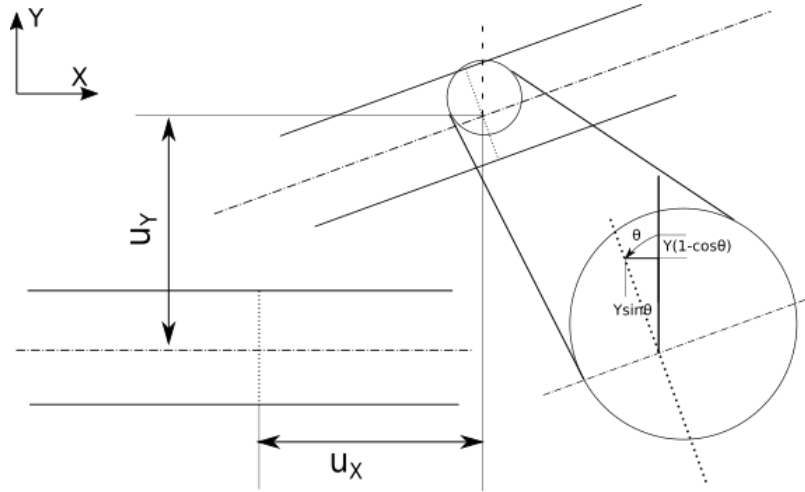


Figure 1: Schematic for the Kinematics

Figure 1 depicts a schematic representation of the beam kinematics. From the assumption that plane sections remain planar, the deformed coordinates of any material point may be given as,

$$\begin{Bmatrix} x \\ y \end{Bmatrix} = \begin{Bmatrix} X + u_X - Y \sin \theta \\ u_Y + Y \cos \theta \end{Bmatrix}. \quad (1)$$

Upper- and lower-case characters denote original and deformed coordinates respectively. The deformation gradient is

$$\mathbf{F} = \nabla \begin{Bmatrix} x \\ y \end{Bmatrix} = \begin{bmatrix} 1 + u'_X - Y \theta' \sin \theta & -\sin \theta \\ u'_Y - Y \theta' \cos \theta & \cos \theta \end{bmatrix}. \quad (2)$$

Primes (') denote differentiation with respect to X , the original X coordinate. The Green-Lagrange strain tensor (defined in the original coordinates) is given by,

$$2\mathbf{E} = \mathbf{F}^T \mathbf{F} - \mathbf{I} = \begin{bmatrix} (u'_X - Y\theta' \sin \theta)^2 + (u'_Y - Y\theta' \cos \theta)^2 + 2(u'_X - Y\theta' \sin \theta)(u'_Y - Y\theta' \cos \theta) & -(1 + u'_X) \sin \theta + (u'_Y) \cos \theta \\ -(1 + u'_X) \sin \theta + (u'_Y) \cos \theta & 0 \end{bmatrix} \quad (3)$$

As can be seen, the transverse strain ϵ_{YY} is identically 0. Denoting the axial strain as ϵ_{XX} and the shear strain (engineering definition, 2κ) as γ_{XY} , the independent strain components are written out as a polynomial in Y as follows:

$$\begin{aligned} \{\epsilon\} &= \begin{Bmatrix} \epsilon_{XX} \\ \gamma_{XY} \end{Bmatrix} = \begin{Bmatrix} \frac{1}{2} \left((u'_X - Y\theta' \sin \theta)^2 + (u'_Y - Y\theta' \cos \theta)^2 + 2(u'_X - Y\theta' \sin \theta)(u'_Y - Y\theta' \cos \theta) \right) \\ -(1 + u'_X) \sin \theta + (u'_Y) \cos \theta \end{Bmatrix} \\ &= \begin{Bmatrix} \frac{1}{2} (u'^2_X + u'^2_Y + 2u'_X u'_Y) \\ -(1 + u'_X) \sin \theta + (u'_Y) \cos \theta \end{Bmatrix} + Y \begin{Bmatrix} -\theta' ((1 + u'_X) \cos \theta + (u'_Y) \sin \theta) \\ 0 \end{Bmatrix} \\ &\quad + Y^2 \begin{Bmatrix} \frac{1}{2} \theta'^2 \\ 0 \end{Bmatrix} \\ &= \begin{Bmatrix} \epsilon_0 \\ \gamma_0 \end{Bmatrix} + Y \begin{Bmatrix} \epsilon_1 \\ 0 \end{Bmatrix} + Y^2 \begin{Bmatrix} \epsilon_2 \\ 0 \end{Bmatrix} \end{aligned} \quad (4)$$

The expansion shows that the strain state at any point may be captured with four functions of the displacement field, ϵ_0 , γ_0 , ϵ_1 , and ϵ_2 . The exact forms of the gradients and the Hessians of these functions are given in appendix B.

Using the constitutive law for a perfectly elastic linear isotropic homogeneous material, the corresponding stresses are written down in terms of the Young's modulus E and the shear modulus G (related through the Poisson's ratio ν as $G = E/2(1 + \nu)$).

$$\{\sigma\} = \begin{Bmatrix} E\epsilon_0 \\ G\gamma_0 \end{Bmatrix} + EY \begin{Bmatrix} \epsilon_1 \\ 0 \end{Bmatrix} + EY^2 \begin{Bmatrix} \epsilon_2 \\ 0 \end{Bmatrix} \quad (5)$$

2.1 Strain Energy Density

Using eqs. (4) to (5) one may write down the strain energy density in terms of the aforementioned scalar functions as,

$$\begin{aligned} \bar{\mathcal{V}} &= \frac{1}{2} \{\sigma\}^T \{\epsilon\} \\ &= \frac{1}{2} [(E\epsilon_0^2 + G\gamma_0^2) + Y(2E\epsilon_0\epsilon_1) + Y^2(E\epsilon_1^2 + 2E\epsilon_0\epsilon_2) + Y^3(2E\epsilon_1\epsilon_2) + Y^4(E\epsilon_2^2)] \end{aligned} \quad (6)$$

Further, if one makes the assumption that the central axis passes through the centroid for each section, $\bar{\mathcal{V}}$ may be integrated over each material section to give the linear strain energy density based on the zeroth (area), second, and fourth moments of the cross sectional area defined as follows.

$$A = \int Y^0 dA \quad (7)$$

$$I_2 = \int Y^2 dA$$

$$I_4 = \int Y^4 dA \quad (8)$$

It must be noted that the odd moments drop off if the coordinate Y is measured from the neutral plane.

Thence, the linear strain energy density is,

$$\mathcal{V} = \frac{1}{2} [(EA\epsilon_0^2 + GA\gamma_0^2) + (EI_2\epsilon_1^2 + 2EI_2\epsilon_0\epsilon_2) + (EI_4\epsilon_2^2)] \quad (9)$$

The internal forces and thus the governing differential equation may be obtained by imposing stationarity in the integral of the potential \mathcal{V} over the whole length of the beam.

2.2 Kinetic Energy Density

The kinetic energy density is,

$$\begin{aligned}\bar{\mathcal{T}} &= \frac{1}{2}\rho \left(\frac{d}{dt}(x - X)^2 + \frac{d}{dt}(y - Y)^2 \right) \\ &= \frac{1}{2}\rho \left(\frac{d}{dt}(u_X - Y \sin \theta)^2 + \frac{d}{dt}(u_Y + Y \cos \theta - Y)^2 \right) \\ &= \frac{1}{2}\rho(u_X^2 + u_Y^2) - Y \left(\rho \dot{\theta}(u_X \cos \theta + u_Y \sin \theta) \right) + Y^2 \left(\frac{1}{2}\rho \dot{\theta}^2 \right).\end{aligned}\quad (10)$$

Here, dots ($\dot{}$) represent differentiation with time.

Once again, integration over the section offers a much simpler expression for the linear kinetic energy density \mathcal{T} .

$$\mathcal{T} = \frac{\rho A}{2}(u_X^2 + u_Y^2) + \frac{\rho I_2}{2}\dot{\theta}^2 \quad (11)$$

Using the above two quantities, the linear Lagrangian density is given as,

$$\mathcal{L} = \mathcal{T} - \mathcal{V}. \quad (12)$$

Stationarity of the lagrangian with respect to the degrees of freedom in the formulation will allow us to obtain the “internal forces” or the equations of motion.

2.3 External Forcing

The external forcing can be of two broad types: follower and non-follower forces. Follower forces are applied in a direction that is fixed with respect to the beam locally, while non-follower forces are applied in a direction that is fixed with respect to the initial configuration of the beam. The virtual work done by each of these forces is used to develop the consistent finite element formulation. Figure 2 shows schematics for the same.

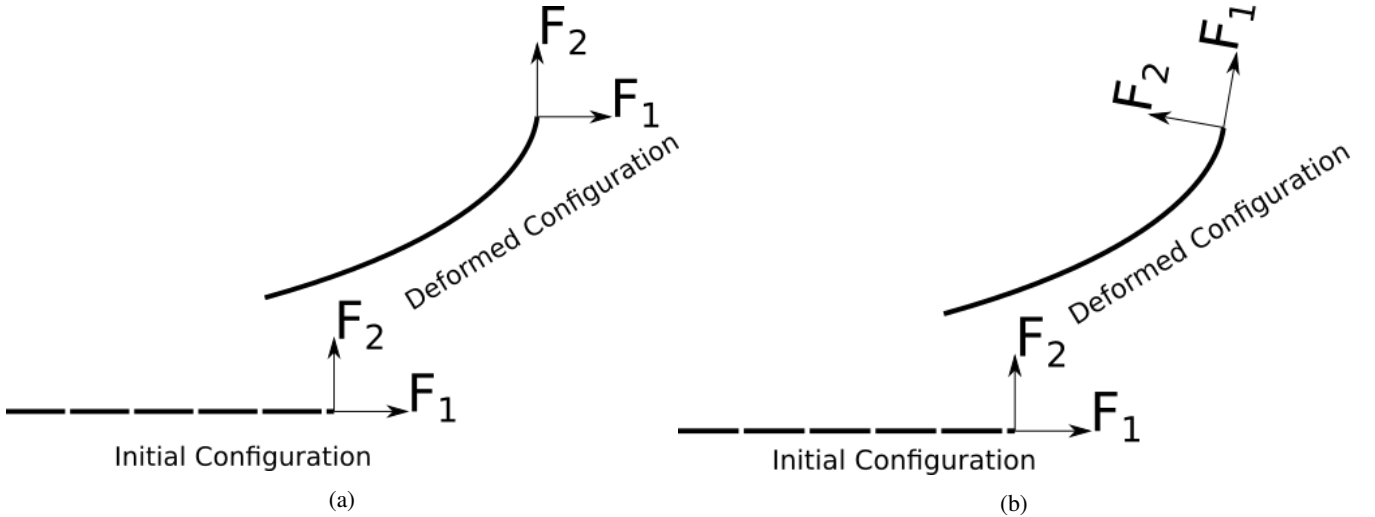


Figure 2: (a) Non-Follower and (b) Follower loads: Schematic

Non-Follower Forces The virtual work done by non-follower forces is only related to the global degrees of freedom u_X & u_Y . Hence the virtual work done will be,

$$\delta[w]_{nf} = f_1 \delta[x - X] + f_2 \delta[y - Y] = f_1 \delta[u_X - Y \sin \theta] + f_2 \delta[u_Y + Y \cos \theta - Y] + \frac{M}{A} \delta[\theta] \quad (13)$$

Integrating this over a section, and assuming $f_1(Y) = F_1/A$, & $f_2(Y) = F_2/A$, the virtual work done becomes,

$$\delta[W]_{nf} = F_1\delta[u_X] + F_2\delta[u_Y] + M\delta[\theta] \quad (14)$$

If applied in a nodal sense, the nodal force vector will have the two forces F_1 & F_2 and the moment M .

$$\{F^n\} = \begin{Bmatrix} F_1 \\ F_2 \\ M \end{Bmatrix}$$

If, on the other hand, the forcing is applied in a distributed manner, this will have to be integrated with the shape functions. The current study does not involve distributed forces and it will not be developed here.

Follower Forces The virtual work done by follower forces is given by,

$$\begin{aligned} \delta[w]_f &= f_1\delta[(x - X)\cos\theta + (y - Y)\sin\theta] + f_2\delta[-(x - X)\sin\theta + (y - Y)\cos\theta] + \frac{M}{A}\delta[\theta] \\ &= (f_1\cos\theta - f_2\sin\theta)\delta[u_X] + (f_1\sin\theta + f_2\cos\theta)\delta[u_Y] \\ &\quad + (-f_1u_X\sin\theta + f_1u_Y\cos\theta - f_2u_X\cos\theta - f_2u_Y\sin\theta + \frac{M}{A})\delta[\theta] \\ &\quad + Y(-f_1\cos\theta + f_2\sin\theta)\delta[\theta] \end{aligned} \quad (15)$$

Once again, integrating this over the section, and assuming constants as before, this becomes,

$$\begin{aligned} \delta[W] &= (F_1\cos\theta - F_2\sin\theta)\delta[u_X] + (F_1\sin\theta + F_2\cos\theta)\delta[u_Y] \\ &\quad + (F_1(-u_X\sin\theta + u_Y\cos\theta) - F_2(u_X\cos\theta + u_Y\sin\theta))\delta[\theta] \end{aligned} \quad (16)$$

Thus, the nodal forces become,

$$\{F^n\} = \begin{Bmatrix} F_1\cos\theta - F_2\sin\theta \\ F_1\sin\theta + F_2\cos\theta \\ F_1(-u_X\sin\theta + u_Y\cos\theta) - F_2(u_X\cos\theta + u_Y\sin\theta) \end{Bmatrix}$$

Since this depends on the state of the system, the corresponding Jacobian is also developed by differentiation as follows:

$$J = F_1 \begin{bmatrix} 0 & 0 & -\sin\theta \\ 0 & 0 & \cos\theta \\ -\sin\theta & \cos\theta & -(u_X\cos\theta + u_Y\sin\theta) \end{bmatrix} + F_2 \begin{bmatrix} 0 & 0 & -\cos\theta \\ 0 & 0 & -\sin\theta \\ -\cos\theta & -\sin\theta & (u_X\sin\theta - u_Y\cos\theta) \end{bmatrix}$$

2.4 Boundary Conditions

The specification of Dirichlet Boundary conditions at nodal locations is classified into two parts: (a) Homogeneous and (b) In-homogeneous boundary conditions.

For the homogeneous boundary conditions, the displacement of a particular node is specified to be exactly zero. For a case with N_d total degrees of freedom with N_h homogeneous boundary conditions, a transformation matrix \mathbf{B} is constructed from the N_d —identity matrix by removing the columns corresponding to the DOFs where the boundary conditions are specified. This can be used to specify a “reduced” vector of degrees of freedom u^{red} as,

$$u^{full} = \mathbf{B}u^{red}$$

Interpreting \mathbf{B} as a Galerkin projection matrix, the residual matrix and its derivatives are transformed as follows:

$$\begin{aligned} R^{red} &= \mathbf{B}^T R^{full} \\ R_{u^{red}}^{red} &= \mathbf{B}^T R_{u^{full}}^{full} \mathbf{B} \\ (R_{uu})_{ijk}^{red} &= \mathbf{B}_{li}\mathbf{B}_{lj}\mathbf{B}_{lk}(R_{uu})_{ijk}^{full} \end{aligned}$$

For inhomogeneous boundary conditions, the corresponding residual equation is replaced by,

$$u_k = d_i$$

where u_k is the nodal degree of freedom where the boundary condition has been specified, and d_i is the boundary specification. In the Jacobian matrix, all entries except for the one corresponding to the k^{th} column are set to zero in the k^{th} row. In order to avoid numerical ill-conditioning of the Jacobian matrix, one could pre-multiply the left- and right-hand sides of the equation by a constant in the order of magnitude of the rest of the entries in the Jacobian matrix. For the current work, a conditioning coefficient was introduced by looking at the order of diagonal elements of the Jacobian matrix of the internal force about the trivial solution. The corresponding third derivative entries are all set to zero.

2.5 Finite Element Formulation

For the finite element formulation, 2-noded elements will be used in the current study. Since the current approach is intended to be valid for large deformations and strains, the limits of the integral will be dependent on the degrees of freedom. This is calculated succinctly using a linear projection onto a line spanning from -1 to $+1$. This transformation is,

$$\begin{aligned} x(\xi) &= \underbrace{\frac{1-\xi}{2}}_{N_1(\xi)} x_1 + \underbrace{\frac{1+\xi}{2}}_{N_2(\xi)} x_2 \\ \implies u(x) &= u(x(\xi)) = u(\xi) = N_1(\xi)u_1 + N_2(\xi)u_2 \\ &= [N_1(\xi) \quad N_2(\xi)] \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \end{aligned} \quad (17)$$

The nodal degrees of freedom are represented by

$$\{d^n\} = \begin{Bmatrix} u_X \\ u_Y \\ \theta \end{Bmatrix}. \quad (18)$$

Thus, for any given 2-noded element, the element degrees of freedom are,

$$\{d^e\} = \begin{Bmatrix} u_{X_1} \\ u_{Y_1} \\ \theta_1 \\ u_{X_2} \\ u_{Y_2} \\ \theta_2 \end{Bmatrix}. \quad (19)$$

Thence, the Lagrangian over an element is,

$$L = \int_0^{L_e} \mathcal{L} dx = \int_{-1}^1 \mathcal{L} \frac{L_e}{2} d\xi \quad (20)$$

Hence, for the total Lagrangian formulation, the quantity that will have to be made stationary over the element is $\mathcal{E} = \frac{1}{2} \mathcal{L} L_e$. L_e is the element length, given by,

$$L_e = \sqrt{(X_2 + u_{X_2} - X_1 - u_{X_1})^2 + (u_{Y_2} - u_{Y_1})^2}. \quad (21)$$

The equations of motion, in terms of the element-nodal degrees of freedom, will be derived as,

$$\frac{d}{dt} \frac{\partial \mathcal{E}}{\partial \{\dot{d}^e\}} - \frac{\partial \mathcal{E}}{\partial \{d^e\}} = \{F^e\} \quad (22)$$

Appendix C gives the complete expressions for the equations of motion. In summary, the dynamical equations of the system will be represented as,

$$\mathbf{D}(d)\{\ddot{d}\} + \mathbf{C}(d, \dot{d})\{\dot{d}\} + \{F^{int}\} = \{F^{ext}\} \quad (23)$$

3 Path Following and Bifurcation Analysis

In order to understand the behavior of the system over a range of loading values (or any other control parameter), the current study uses numerical continuation and eigenvalue-based indirect critical point identification. The generalized algebraic system with an unknown vector u and the scalar parameter λ is represented by

$$u(\lambda) \quad s.t. \quad R(u, \lambda) = 0, \quad (24)$$

where $u \in \mathcal{R}^n$, $\lambda \in \mathcal{R}$, and $R(.,.) : \mathcal{R}^n \times \mathcal{R} \rightarrow \mathcal{R}^n$. Numerical continuation seeks to add an additional equation to this system so that the parameter λ may also be considered as an unknown in the stepping procedure.

Two strategies for this are followed presently, both based on placing constraints on the deviation of the incrementally stepped solution from the previous solution on the branch. The first strategy, termed as **arc-length continuation**, uses an equation that constrains the successively stepped solution to lie on a hyper-elliptical surface centered at the previous solution. Referring to the previous point with 0 as a subscript, the constraint for the successive point is represented by,

$$f(u, \lambda)|_{u_0, \lambda_0} = \sqrt{c(u - u_0)^T \mathbf{K}(u - u_0) + b\lambda^2} - \Delta s = 0. \quad (25)$$

The Jacobian at the previous step ($\partial R / \partial u|_{u_0, \lambda_0}$) is used as the weighting matrix \mathbf{K} . b and c are weighing constants with a functional relationship taken as,

$$q = \left. \frac{\partial R}{\partial u} \right|_{u_0, \lambda_0}^{-1} \left. \frac{\partial R}{\partial \lambda} \right|_{u_0, \lambda_0}$$

$$c = \frac{1 - b}{q^T \mathbf{K} q}$$

The second strategy, termed as the **pseudo arc-length continuation**, is based on ensuring that the successive point lies on a point on the response curve which makes a given step-length when perpendicularly projected onto the tangent vector at the previous step. The constraint equation is given by,

$$f(u, \lambda)|_{u_0, \lambda_0} = (u - u_0)^T \dot{u}_0 + (\lambda - \lambda_0) \dot{\lambda}_0 - \Delta s = 0. \quad (26)$$

The tangents are estimated through the first derivative of the residual function with respect to the arc-length,

$$\dot{R} = \frac{dR}{ds} = \frac{\partial R}{\partial u} \dot{u} + \frac{\partial R}{\partial \lambda} \dot{\lambda} = 0$$

$$z = \frac{\partial R}{\partial u}^{-1} \frac{\partial R}{\partial \lambda}$$

$$\dot{\lambda} = \frac{1}{\sqrt{1 + z^T z}}$$

$$\dot{u} = \dot{\lambda} z$$

The last few relations arise out of insisting on a unit 2-normed tangent.

For both cases, the predictor for each solution is given using the “tangent predictor”, i.e., $u^p = u_0 + \dot{u} \Delta s$ & $\lambda^p = \lambda_0 + \dot{\lambda} \Delta s$.

In order to detect bifurcations, the indeterminacy of the tangent is taken to be the primary criterion. This is monitored by looking at the eigenvalues of the tangent stiffness matrix $\partial R / \partial u$ at each point on the curve. When all the eigen-values are positive, it denotes that the current point is a locally stable solution since this indicates that it is at a local minimum point of the potential of the linearized forcing (Positive-Definiteness of stiffness \implies Positive-Definiteness of the Hessian of Potential). If a particular eigen-value (ordered by increasing value) changes sign, between consecutive steps along the iterations, a bisection search is conducted between the two steps (points) to obtain an estimate of the loading point where the eigen-value is exactly 0. The “direction of indeterminacy”, denoted by z , is estimated using the eigen-vector corresponding to the zero-eigenvalue and the tangent is represented as a linear combination of this and a “particular tangent”, denoted by y , in the null-space of this vector. The particular solution is estimated using the first order rate equations of the residual function as follows:

$$\dot{u} = (\sigma z + y) \dot{\lambda}$$

$$\dot{R} = \frac{\partial R}{\partial u} \dot{u} + \frac{\partial R}{\partial \lambda} \dot{\lambda} = 0$$

$$\implies \left(\frac{\partial R}{\partial u} (\sigma z + y) + \frac{\partial R}{\partial \lambda} \right) \dot{\lambda} = 0$$

$$\implies y = - \frac{\partial R}{\partial u}^{-1} \frac{\partial R}{\partial \lambda}$$

where σ is some weight representing a generic linear combination. For determining the appropriate value for this, the third derivative is employed:

$$\begin{aligned}
\ddot{R} &= \frac{\partial R}{\partial u} \ddot{u} + \frac{\partial^2 R}{\partial^2 u} \dot{u} + \frac{\partial R}{\partial \lambda} \ddot{\lambda} + \frac{\partial^2 R}{\partial^2 \lambda} \dot{\lambda} = 0 \\
&\text{Premultiplying by } z^T \text{ and substituting for } \dot{u}, \\
&\dot{\lambda}^2 \underbrace{[z^T R_{uu} z z]}_a \sigma^2 + \underbrace{z^T (R_{uu}(zy + yz) + 2R_{u\lambda}z)}_b \sigma + \underbrace{z^T (R_{uu}yy + 2R_{u\lambda}y + R_{\lambda\lambda})}_c = 0 \\
\Rightarrow a\sigma^2 + b\sigma + c &= 0 \\
\Rightarrow \sigma_{1,2} &= \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad a \neq 0 \\
\sigma_{1,2} &= 0, \infty \quad a = 0
\end{aligned} \tag{27}$$

The final set of equations represent a quadratic equation with respect to σ . When $a = 0$, it merely signifies that $\dot{u}_1 = y$ (along the “main” branch), and $\dot{u}_2 = z$, along the “bifurcated” branch.

Using these different tangents as the initial tangent for the predictor allows the continuation step to converge to the different solutions for sufficiently small step-sizes (determined empirically through trial and error in the current study). It must be noted that the term $\partial^2 R / \partial u^2$ (denoted R_{uu}), is a third order tensor. Although it is analytically expressed in the formulation, the exact equations are not given in the appendix for terseness of the document.

3.0.1 Dynamical Stability Estimation

In some cases, looking just at the eigenvalues of the static Jacobian need not necessarily represent the stability of the system. Thus, the dynamical equations (eq. (23)) are linearized to understand the perturbation behavior about the equilibrium. Compared to the static stability analysis, the dynamic stability analysis allows for perturbations in the velocity components independent of the displacements. Representing the displacements with u , velocities with \dot{u} and derivatives with subscripts, the state-space recast and linearization are represented as follows:

$$\begin{aligned}
\begin{Bmatrix} \dot{u} \\ \ddot{u} \end{Bmatrix} &= \begin{Bmatrix} \dot{u} \\ -\mathbf{D}^{-1}R - \mathbf{D}^{-1}\mathbf{C}\dot{u} \end{Bmatrix} \\
\begin{Bmatrix} \dot{\delta}_u \\ \dot{\delta}_{\dot{u}} \end{Bmatrix} &= \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{D}^{-1}\mathbf{R}_u - \mathbf{D}^{-1}_u R & -\mathbf{D}^{-1}\mathbf{C}_u \dot{u} - \mathbf{D}^{-1}\mathbf{C} \end{bmatrix}}_{\mathbf{J}_{lin}} \begin{Bmatrix} \delta_u \\ \delta_{\dot{u}} \end{Bmatrix}
\end{aligned}$$

where δ_u , & $\delta_{\dot{u}}$ represent perturbations in displacements & velocities; R represents the static residual $F^{int} - F^{ext}$. The derivatives of the matrices with respect to the displacement vector are computed as third order tensors (assembled and stitched the same way as the rest of the finite element quantities). See appendix D for some details on the third derivative calculations.

The stability of a static equilibrium is assessed using the eigen values of the linearized Jacobian \mathbf{J}_{lin} about the solution. If all the eigenvalues are in the left-half of the s-plane (negative real parts), the solution is said to be stable and if even one eigenvalue lies in the right-half (positive real part), the solution is said to be unstable. It must be noted that the stability of the linearized system corresponds to that of the original system only for hyperbolic equilibrium points. Thus, strictly, the linearization is **inconclusive** if the real parts of all the eigenvalues are exactly zero (see Hartman-Grobman Theorem).

4 Validation — Linear Test Cases

In order to validate the implementation, a set of three static cases are investigated. Figure 3 illustrates the test cantilever under consideration. The three cases will be the tip-loaded response of the cantilever subject to transverse force (F_Y), moment (M), and axial force F_X respectively. Non-conservative loading, i.e., fixed direction loading is employed for the current studies. Apart from these static cases, the low amplitude internal force Jacobian and inertia matrices are used to conduct an eigen analysis to validate the formulation against analytical results.

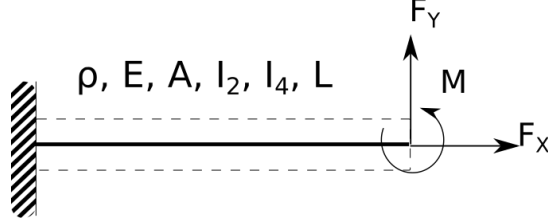


Figure 3: Cantilevered Beam Test Cases

A circular section with radius R is assumed throughout unless otherwise stated. The physical properties used for this are,

| Property | Value |
|----------|------------------------------|
| ρ | 7800 kg m^{-3} |
| E | $210 \times 10^9 \text{ Pa}$ |
| ν | 0.33 |
| G | $E/2(1 + \nu)$ |
| L | 1.0m |
| R | 10^{-2} m |
| A | πR^2 |
| I_2 | $\pi R^4/4$ |
| I_4 | $\pi R^6/8$ |

Table 1: Physical Properties

4.1 A Note on Shear-Locking

Considering the case with $M, F_X = 0$ and $F_Y \neq 0$, an issue that becomes apparent is what has been documented in the literature known as **shear locking**. The formulation requires a huge number of elements before it gives a sufficiently converged result. This is due to the fact that the section rotation angle θ and the transverse deflection u_Y are assumed to vary linearly across an element independently. This leads to an inconsistency since the section rotation is closely related to the first derivative of the transverse deflection. This is rectified by making θ constant over each element whilst not killing θ' . The following substitutions for the shape functions, that has been used with some success in the community, is employed here for the element quantities:

$$u_X(\xi) = N_1(\xi)u_{X1} + N_2(\xi)u_{X2} \quad (28)$$

$$u_Y(\xi) = N_1(\xi)u_{Y1} + N_2(\xi)u_{Y2} \quad (29)$$

$$\theta(\xi) = N_1(\xi)\theta_1 + N_2(\xi)\theta_2 = \frac{\theta_1 + \theta_2}{2} \quad (30)$$

$$u'_X(\xi) = \frac{u_{X2} - u_{X1}}{X_2 - X_1} \quad (31)$$

$$u'_Y(\xi) = \frac{u_{Y2} - u_{Y1}}{X_2 - X_1} \quad (32)$$

$$\theta(\xi) = \frac{\theta_2 - \theta_1}{X_2 - X_1} \quad (33)$$

While the first two are merely the linear interpolation on the element, the average value is used for the θ degree of freedom. The expressions for the derivatives, on the other hand, are not modified.

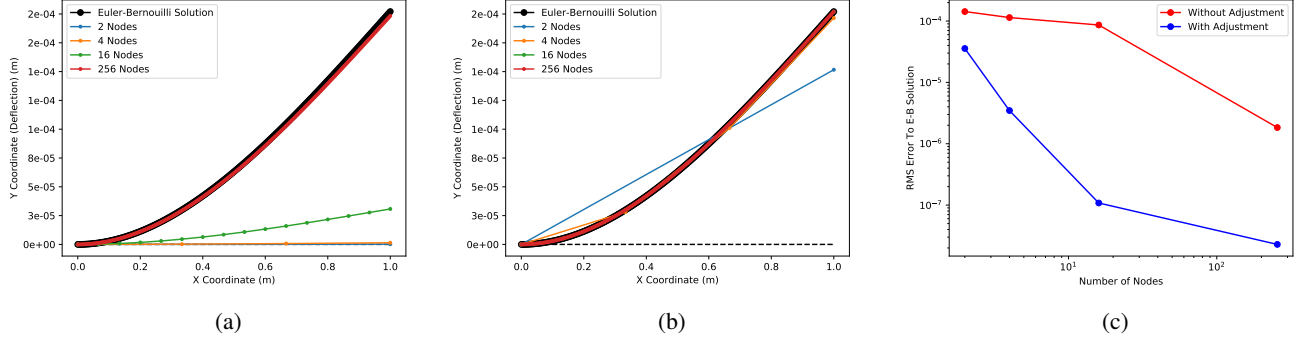


Figure 4: Effect of Shear-Locking Adjustment: (a) Without adjustment, (b) With Adjustment, (c) RMS Error in response for both cases

Since the internal force only contains θ and the derivatives of the degrees of freedom, after this substitution, the stress/strain over each element will be constant. Thus the above substitution is the same as using a single-point quadrature for the calculations. In the interest of generality, however, the substitution method is followed for the current study.

Figure 4 depicts the response against the Euler-Bernoulli reference solution for both cases. It can clearly be seen that the convergence of the adjusted formulation is much better.

4.2 Static validation test cases

For the three static validation test cases, the forcing amplitudes of $1N$, $1Nm$, and $1N$ are respectively used. Figure 5 presents the responses for each test case along with the corresponding low-amplitude analytical results for a 11-noded (10-element) simulation.

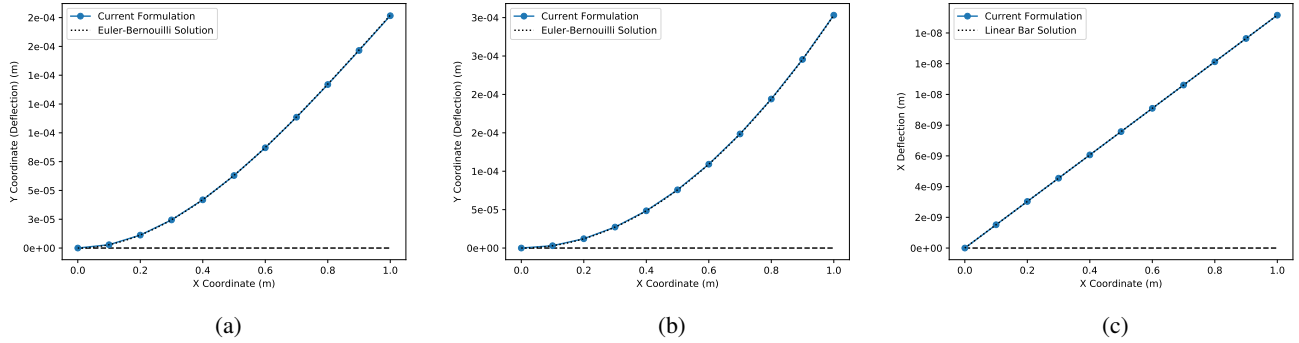


Figure 5: Static Validation Cases: (a) Case 1: $F_Y = 1N$, (b) Case 2: $M = 1Nm$, (c) Case 3: $F_X = 1N$

4.3 Low amplitude Modal Validation

In order to validate the inertial matrix, the stiffness matrix is formulated as the Jacobian when all the nodal degrees-of-freedom are set to 0, and the inertial matrix at the same condition is taken as the mass matrix. The mode-shapes and frequencies from the corresponding Eigen-analysis are presented in fig. 6.

The **longitudinal vibration modes** are set to happen at frequencies,

$$\omega_r = (2r - 1) \frac{\pi}{2} \sqrt{\frac{E}{\rho}} \quad (34)$$

Here, the first mode comes to $\omega_{L1} = 8150.46 rad/s$. It can be seen that Mode 6 from the simulations has a frequency of $8152.78 rad/s$ and corresponds to axial vibration of the structure.

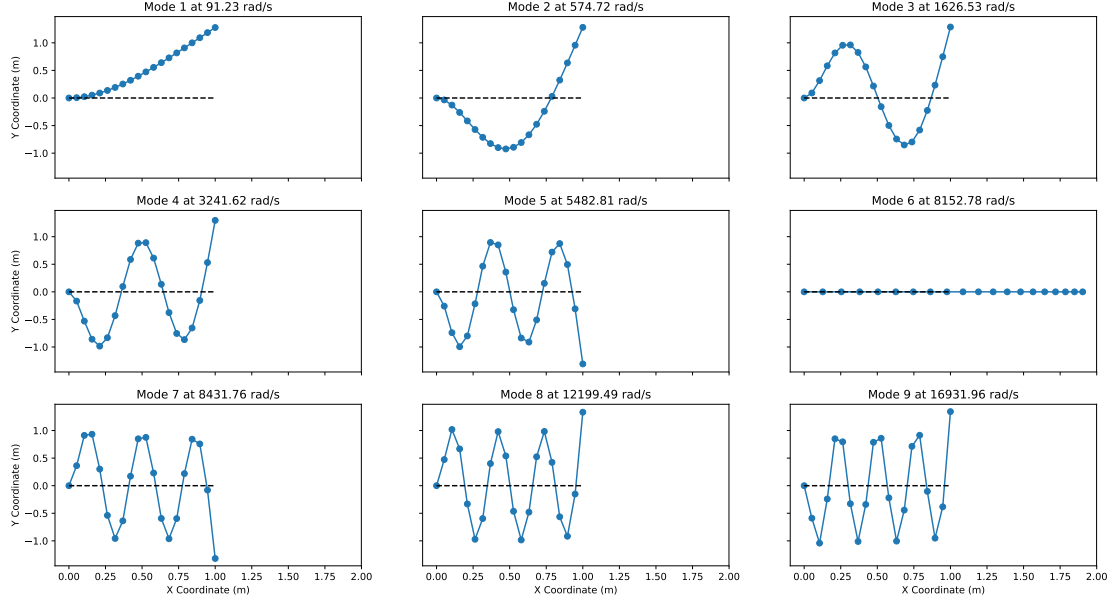


Figure 6: Low-amplitude Eigen-analysis

The **Transverse vibration modes** are set to occur at frequencies,

$$\omega_n = \left(\frac{\beta_n}{L} \right)^2 \sqrt{\frac{EI}{\rho A}} \quad \beta_n = 1.875, 4.694, 7.855, \dots \quad (35)$$

Analytically, the first three modes come out to be 91.21 rad/s , 571.63 rad/s , and 1600.75 rad/s respectively. This is verified in fig. 6 since the first three modes there fall at 91.23 rad/s , 574.72 rad/s , and 1626.53 rad/s respectively. Thus, the formulated matrices are valid, atleast for low amplitudes.

Before the application of the boundary conditions, it was also verified that the system shows 3 distinct zero-frequency, or rigid-body modes.

5 Nonlinear Application Cases

For the nonlinear application cases, the step-size Δs for continuation was adapted based on the number of iterations required for converging at each step. If a particular step took 5 iterations or less to converge, the step size was scaled up by a factor of 2.0; and if it took more than 10 iterations, the step size was scaled by 0.5.

In order to have well-conditioned matrices for continuation, the forces were all represented as $|f| = 10^6 \lambda$, with λ being the continuation parameter. The number 10^6 was obtained by looking at the order of the diagonal terms of the internal force Jacobian about the trivial solution. A similar conditioning number was introduced for the in-homogeneous boundary conditions (see section 2.4). The influence of the numerical ill-conditioning was most apparent around transcritical bifurcations such as the ones found in section 5.5, where the algorithm fails to turn around the limit-points when the problem is ill-conditioned.

5.1 Follower vs Non-Follower Loads

In order to demonstrate the difference between the response of the system under follower and non-follower loads, the same cantilever is forced using a follower and a non-follower load cases with $F_1 = 0N$ & $F_2 \neq 0$. The loads are applied as if concentrated on the node at the free end. Figure 7 shows the system responses for different forcing amplitudes. It can be seen that the influence becomes very pronounced as the force amplitude increases.

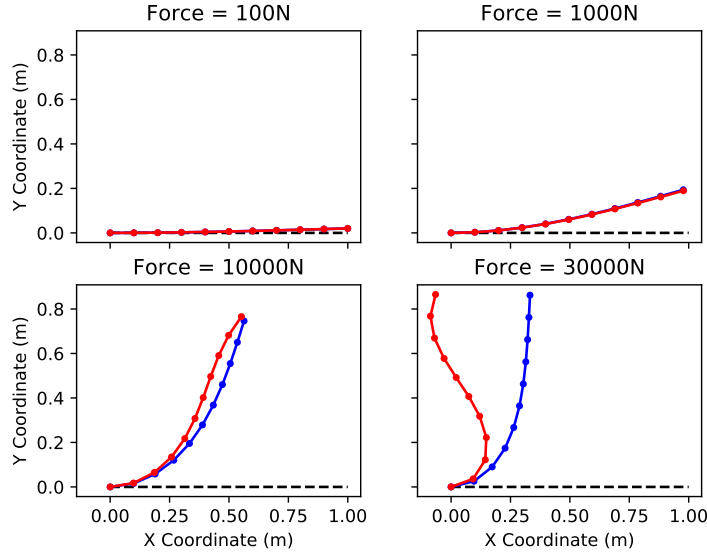


Figure 7: Difference between follower and non-follower load responses in the transverse direction. Blue corresponds to the non-follower case and red corresponds to the follower case.

5.2 Buckling - Cantilever

In order to investigate buckling, two cases are considered: with a non-follower load and with a follower load. The analytical solution for the Euler critical load for the cantilevered beam are,

$$P_{crit}^k = (2k - 1)^2 \frac{\pi^2 EI_2}{4L^2} \quad k = 1, 2, \dots$$

This solution can be compared against the non-follower loading case. However, buckling under follower loading for a cantilevered beam takes a slightly different form. The first mode here is given by,

$$P_{f,crit}^1 = 20.1 \frac{EI_2}{L^2}$$

Table 2 tabulates the critical loads as calculated from the bisection searches conducted as above for both the cases. It can be

| Mode | Euler-Critical | EB-Follower | Non-Follower | Follower |
|------|----------------|-------------|--------------|------------|
| 1 | 4,069.57 | 33,151.63 | 4,084.94 | 13,092.69 |
| 2 | 36,626.16 | | 37,901.94 | 65,615.03 |
| 3 | 101,739.34 | | 112,102.31 | 164,736.96 |

Table 2: Buckling Critical Loads (all loads in N)

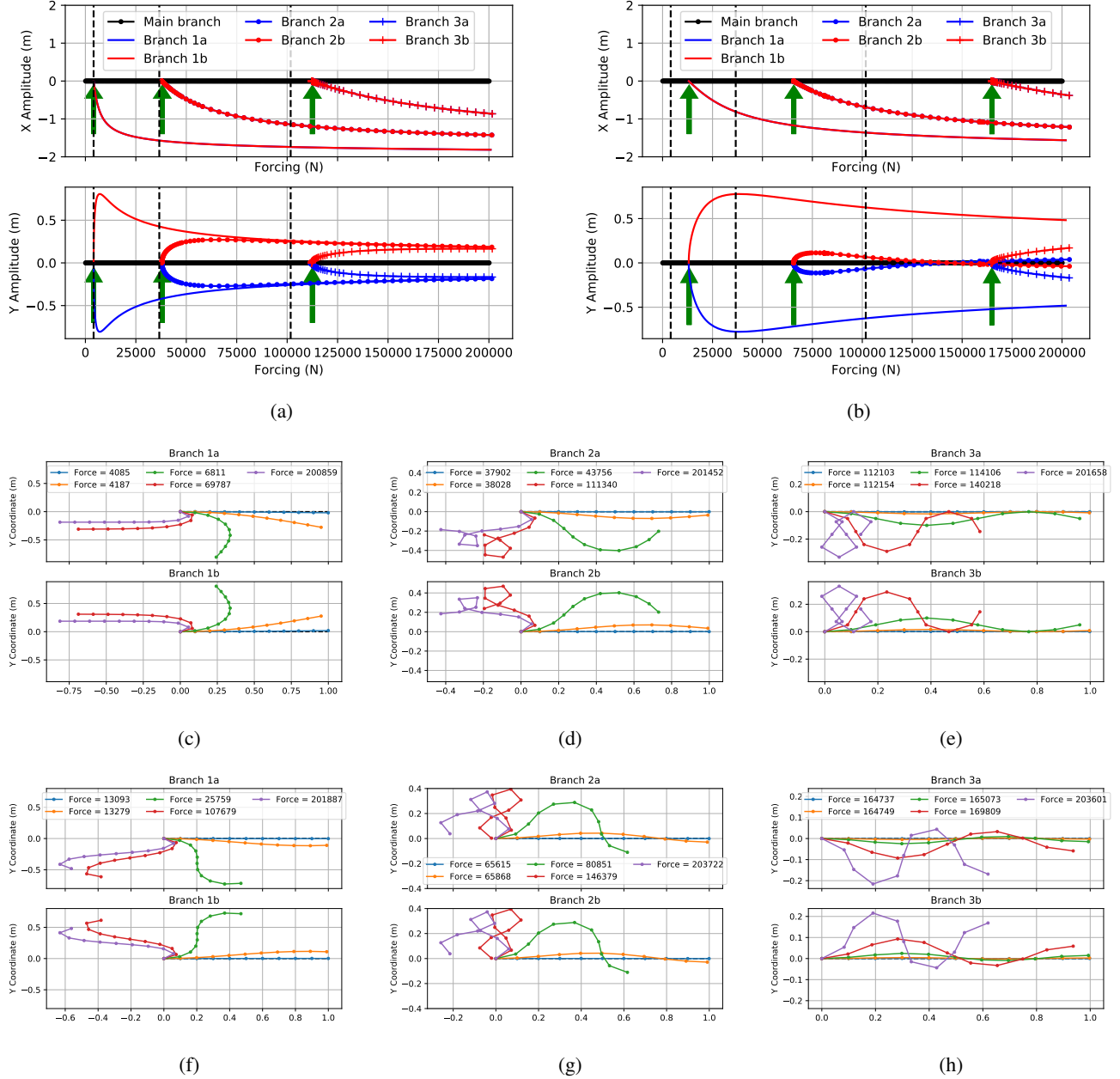


Figure 8: Bifurcation Diagrams for the cantilevered beam in the (a) non-follower and (b) follower loading cases. (c), (d), (e) depict the mode shapes for the non-follower case, and (f), (g), (h) depict the mode shapes for the follower case. In (a) & (b), the dashed vertical line represents the Euler Critical Load and the green arrows represent the critical loads corresponding to loss of stability through successive eigenvalues crossing the imaginary axis.

seen that the Euler-Critical load matches the non-follower case pretty well but the Euler-Bernouilli prediction for the follower buckling is way higher.

Figure 8 depicts the bifurcation diagrams for the two cases. The shapes are observed to be characteristically different for both the cases. Additionally, the green arrows denote the load levels where the “backbone” equilibrium branch loose dynamic stability through successive eigenvalues. In other words, the first green arrow corresponds to the load case where the first eigenvalue goes across to the right-half of the s-plane, and the second green arrow corresponds to the load case where the second eigenvalue crosses the imaginary axis. For the bifurcation tracking, however, only the static eigen values and vectors are employed.

Arc-length continuation was found to be better suited for continuing the main branch while the Pseudo arc-length formulation was found to be more robust for tracing out the bifurcated branches. The actual mode-shape corresponding to each branch for different load amplitudes are also plotted. The shape of the solution in the main branch is not plotted since it is merely a trivial compression mode. Since the Euler-Bernouilli buckling load predictions for the follower loading case is much higher than the calculated buckling loads, usage of the former is not admissible for design since the beam would buckle for a much lower load.

5.3 Buckling - Simply Supported

The simply supported case presents a more interesting case. Figure 9 depicts the schematic for the buckling problem with the simply supported case. The boundary conditions are zero axial & transverse displacements in the left end and zero transverse displacement at the right end.

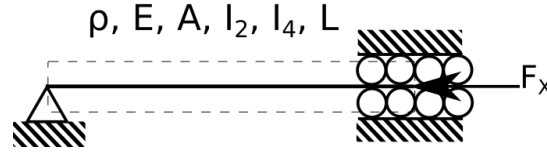


Figure 9: The Simply Supported Beam

The analytical solutions for the critical buckling loads (plotted in fig. 10) are,

$$P_{crit}^k = k^2 \frac{\pi^2 EI_2}{L^2} \quad k = 1, 2, \dots$$

Figure 10 shows the results of the bifurcation analysis conducted for this case. Once again, the green arrows represent the loss of dynamic stability through successive eigenvalues. For the non-follower loading case, the response seems to be similar in sense to the response encountered for the previous case. However, the non-follower case led to a particular load level for each buckling mode beyond which the solver kept failing. This could be due to the fact that a static equilibrium along the same branch ceases to exist beyond that load level. The right ends of the responses for the follower load case seems to be more “flat” or with lesser plane angle θ than for the non-follower case. This could be understood by the additional stiffness to section rotation coming from the application of follower forces.

In both of the above cases, the bifurcation points and the points of loss of dynamic stability happen at the same load level. It must be noted that the term “loss of stability” is used in a loose sense in that it denotes the point where the linearized analysis provides *sufficient* conditions for the loss of stability along atleast one direction in the state-space by making the equilibrium point hyperbolic in that direction. This is to say that the actual loss of stability of the nonlinear system could have happened before, but we can say for certain that the solution is not stable afterwards.

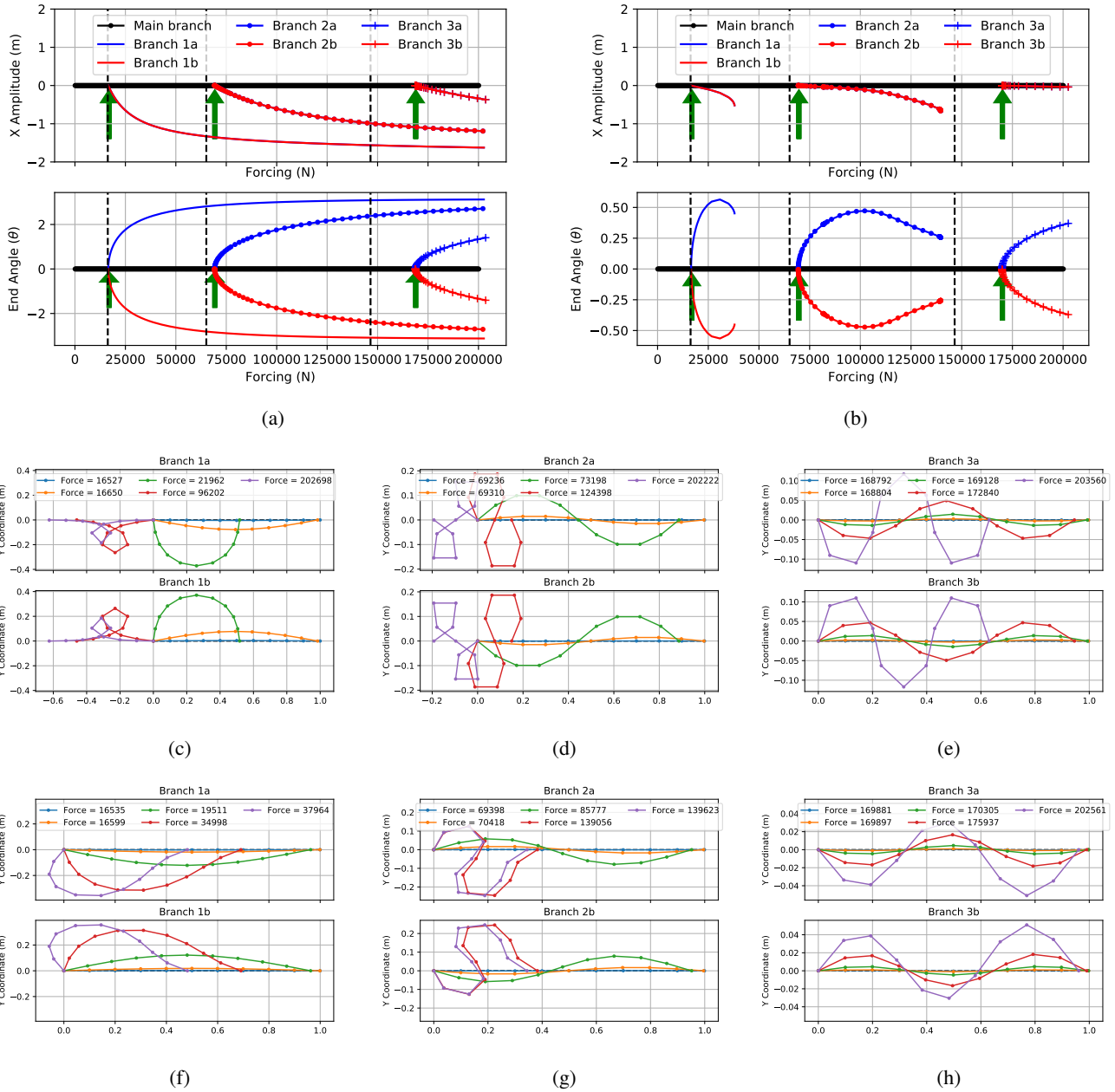


Figure 10: Bifurcation Diagrams for the simply supported beam in the (a) non-follower and (b) follower loading cases. (c), (d), (e) depict the mode shapes for the non-follower case, and (f), (g), (h) depict the mode shapes for the follower case. In (a) & (b) the dashed vertical line and green arrows are drawn at Euler-Critical load values and points of loss of dynamic stability respectively.

5.4 Buckling - Simply Supported Beam with Inhomogeneous Tip BC

For the current case, the simply supported beam in fig. 9 is employed but with “no forcing”. The input is specified based on the x deflection of the tip node of the beam. Figure 11 depicts the bifurcation diagram for this case with the four modes under investigation. Pitchfork bifurcations similar to the previous case are observed here too.

The main purpose of considering this case is to have a systematic definition for the definition of the pre-stressed initial configuration for the snap-through analysis. The odd modes are “symmetric” while the even ones are “unsymmetric” about the mid-point of the beam. This distinction is important to predict the nature of the snap-through response.

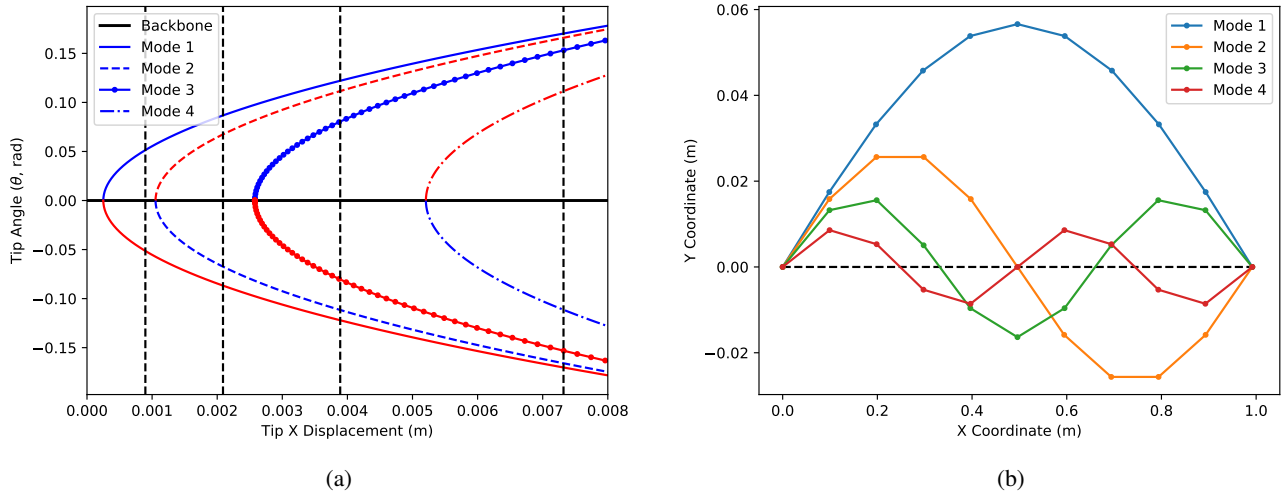


Figure 11: Buckling Behavior of the Simply Supported Beam under Tip deflection control: (a) Bifurcation Diagram; (b) Buckling Modes at maximal final X-deflection (blue branches in (a)). The dashed vertical lines in (a) indicate the load levels where snap-through analysis is performed (in section 5.5).

5.5 Snap-Through Behavior of a Buckled Simply-Supported Beam

For the snap-through analysis, the beam is first buckled to the first mode under an in-homogenous boundary condition (see section 5.4). Thus, whatever be the response, the axial and transverse deflections of the two ends of the beam are fixed — the left one to the origin, and the right one to the boundary condition. The only “active” degree of freedom for the ends are the sectional rotations. Following this, a transverse load in the $-\hat{y}$ direction is applied at the node in the center. Since an 11-noded model is used for these analyses, this transverse loading corresponds to node 6. The response of the structure, defined by the nodal y – deflection and the nodal section rotation θ , are investigated for the bifurcation study. Four different tip deflection boundary conditions are used, each corresponding to cases with the existence of progressively more number of buckling modes.

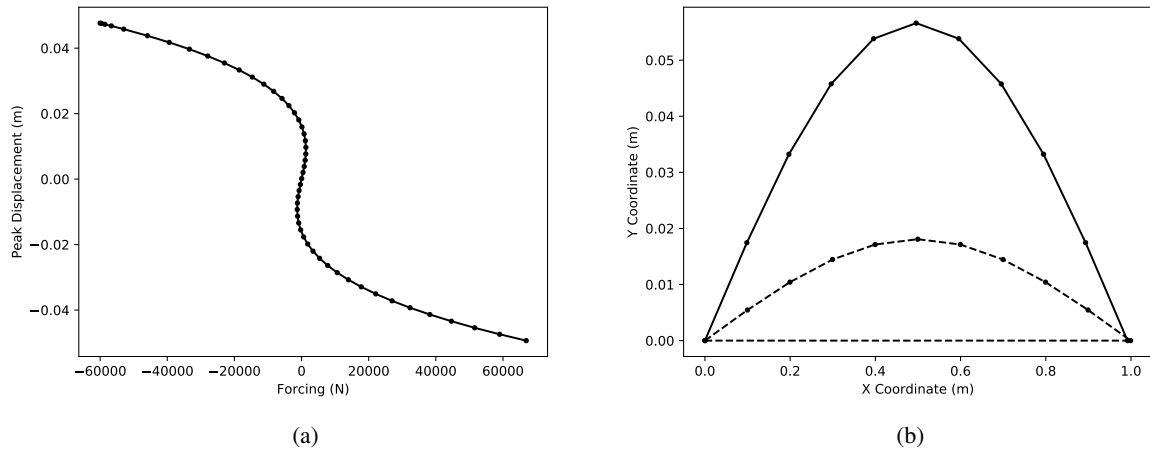


Figure 12: Snap-through at Buckling Level 1 ($\delta_{crit}^1 < \delta_x < \delta_{crit}^2$). (a) Load-Displacement diagram; (b) Deflection profiles. In (b) the dotted continuous line represents the deflection at zero snap-through load, while the dotted dashed-line represents the deflection before the limit point.

Figure 12 represents the snap-through behavior when the tip displacement $\delta_x = 896\mu m$. This is between the first and the second critical displacements. The existence of just a single buckling mode is reflected as the presence of the classical “snap-through” behavior, in the form of the existence of two limit-point (or transcritical) bifurcation points in the response. No

stable equilibrium exists in the branch between these points, and they are also the critical loads when the system snaps through to a shape that is buckled in the opposite direction.

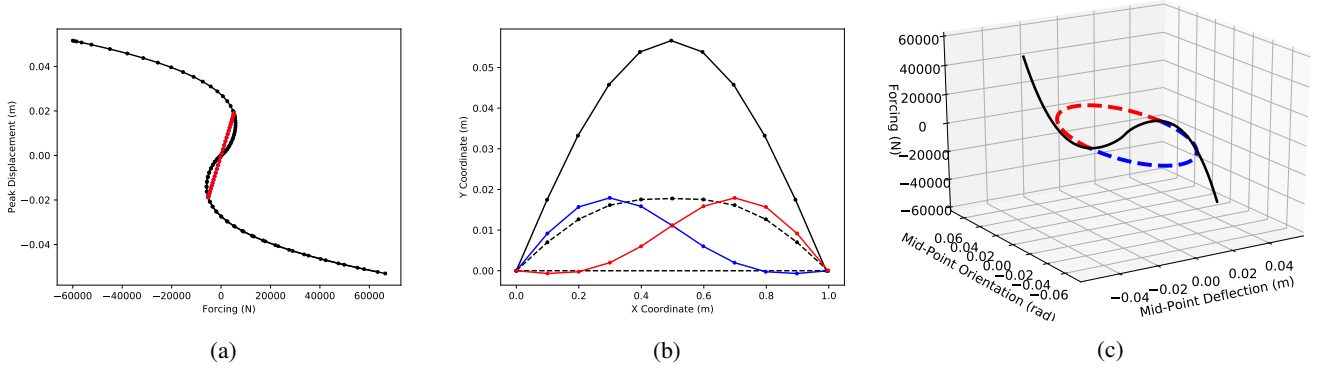


Figure 13: Snap-through at Buckling Level 2 ($\delta_{crit}^2 < \delta_x < \delta_{crit}^3$). (a) Load-Displacement diagram; (b) Deflection profiles; and (c) 3D Load-Displacement-Rotation diagram. In (b) the dotted continuous line represents the deflection at zero snap-through load; the dotted dashed-line represents the deflection before the limit point; the dotted blue & red lines show the deflection along the two bifurcated branches.

Figure 13 shows the snap-through behavior when the beam is initially buckled beyond its second (unsymmetric) critical mode ($\delta_x = 2089\mu m$). Now that an unsymmetric mode exists, the system loses stability before the transcritical point, wherein an unsymmetric transition between the snap-through states becomes more stable. As can be seen from fig. 13(b), two unsymmetric modes are present for each load level, corresponding to the symmetric parts of the pitchfork bifurcation. Since this did not happen for $\delta_x < \delta_{crit}^2$, it is surmised that such a mode of snap-through can exist only when the corresponding buckling mode exists “inherently” in the structure.

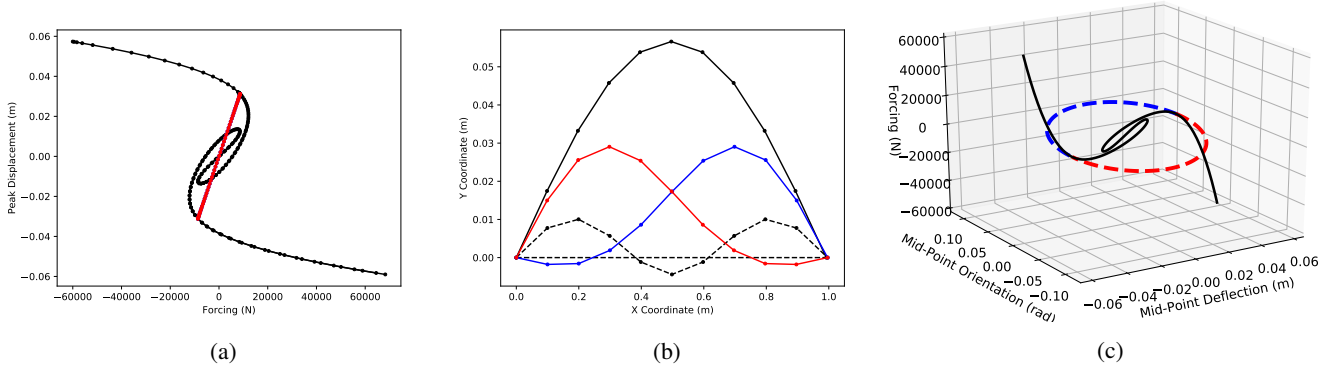


Figure 14: Snap-through at Buckling Level 3 ($\delta_{crit}^3 < \delta_x < \delta_{crit}^4$). (a) Load-Displacement diagram; (b) Deflection profiles; and (c) 3D Load-Displacement-Rotation diagram. In (b) the dotted continuous line represents the deflection at zero snap-through load; the dotted dashed-line represents the deflection beyond the second limit point; the dotted blue & red lines show the deflection along the two pitchfork branches.

Figure 14 shows the response of the system to the snap-through loading when it is initially prestressed beyond the second critical ($\delta_x = 3889\mu m$). In terms of bifurcations, the system has two additional turning points (transcritical bifurcations), while the number of symmetric bifurcations stays the same. It is surmised that the added existence of the symmetric buckling mode (mode 3) in the system does not let the beam take other unsymmetric paths, but introduces additional “snap-through” points by introducing additional symmetric equilibria for certain load cases.

Figure 15 shows the snap-through response of the system prestressed initially beyond its fourth critical axial deflection ($\delta_x = 7319\mu m$). Since the fourth buckling mode is unsymmetric, this introduces an additional pitchfork bifurcation, while keeping the number of transcritical points same (4). Figure 15(b) depicts all the different unsymmetric modes — it may be observed that these always occur in pairs.

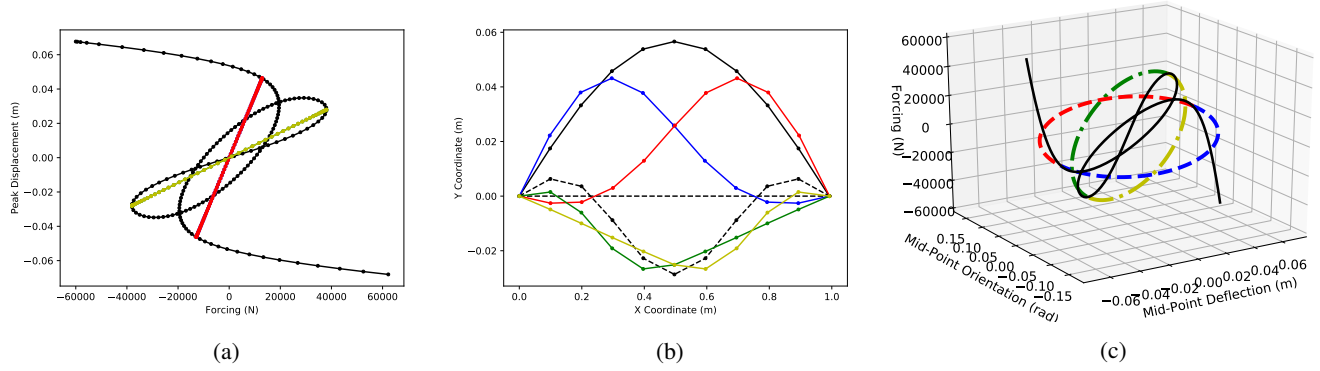


Figure 15: Snap-through at Buckling Level 4 ($\delta_{crit}^4 < \delta_x < \delta_{crit}^5$). (a) Load-Displacement diagram; (b) Deflection profiles; and (c) 3D Load-Displacement-Rotation diagram. In (b) the dotted continuous line represents the deflection at zero snap-through load; the dotted dashed-line represents the deflection beyond the second limit point; the dotted blue & red lines show the deflection along the first pitchfork pair; the dotted green & yellow lines show the deflection along the second pitchfork pair.

6 Discussions and Conclusions

A generic planar beam element was implemented in `Python` in such a manner that the only assumption is that plane sections remain planar. The Timoshenko beam formulation was employed, which resulted in shear-locking effects because of fully decoupling the section rotation and the transverse deflection. This effect was countered by altering the interpolation of the derivatives of the section angles by keeping them constant over the section (field-consistency approach). This thus makes the developed elements the so-called C^0 shear elements. Additionally, follower & non-follower forces, static homogeneous & in-homogeneous boundary conditions, and matrices necessary for full-nonlinear as well as linearized dynamic analyses were formulated and implemented. For solving the systems, a Newton's algorithm was implemented for sparse matrices (`scipy`'s optimization toolbox ¹ does not support sparse matrices for full Jacobian solvers yet); arc-length continuation was implemented with the ability to switch between two different arc-length constraints; the second order Hessian tensor of the quantities involved were implemented following analytical calculations for conducting bifurcation analyses and path-following. Appropriate conditioning coefficients were introduced in some of the above in order to keep the involved matrices numerically well-conditioned. It was found to be a little easier to have the orthogonal constraint for the continuation in most cases, but the Δs required for this was found to be much larger than the corresponding step-size in the conventional arc-length constraint for what seemed to be the same increment on the response curve.

Following some validation runs against linear test cases with analytical solutions, the code was used to perform buckling and snap-through response tracing for the cantilevered and simply supported configurations. Pitchfork bifurcations were successfully determined (indirectly) by estimating and monitoring the eigenvalues of the Jacobian matrix at each point along the response trajectory. The Hessian tensors were used to classify the bifurcations and determine the tangent along each branch. It must be noted that all of the bifurcations encountered in the current study are either transcritical (in the form of limit/turning points) or pitchfork bifurcations, which are both examples of “simple bifurcations”, and thus the Hessian tensor was sufficient to determine the tangent directions. For more complicated bifurcations, higher derivatives will be necessary.

¹<https://docs.scipy.org/doc/scipy/reference/optimize.html>

A Guide to Submitted Python Scripts

The following is a guide to the different Python scripts in the submission.

Scripts with Routine Definitions

STATICS.py This contains the major routines for the implementation: element shape functions, integration routines, element quantities, static residual, etc.

DYNAMICS.py This contains the definitions of the routines which were used for the dynamical analyses in the current study. Most of these routines require the routines in `STATICS.py`. The routines here include the dynamic matrices & their derivative tensors, and linearization routines.

SOLVERS.py Written pretty much independent of the rest of the scripts, this contains the definitions for the solution routines, including the sparse Newton solver, continuation routines (which rely on sparse Jacobians), and singular point tangent estimation routines.

Scripts intended for checking the code

check.py This contains checks for the various functions and their derivatives, comparing numerical differentiation to the analytical estimates. Apart from this, this also contains some initial test cases for most of the problems considered in the study.

continuation.py This contains a simple beam bending case for checking the continuation implementation. (*this may not work with the final form of the code*).

checkcontin.py This investigates the influence of numerical conditioning in the continuation of solutions beyond turning points by using a parabola to test the code.

Scripts with simulations feeding into the final report

locking.py This was the file used for generating the plots in section 4.1, assessing the influence of shear-locking.

validate.py This has the different (axial, transverse & moment) deflection cases for the validations in section 4.2.

linmodes.py This conducts the low-amplitude modal analysis for section 4.3.

follower.py This conducts the comparison between follower and non-follower loading cases in section 5.1.

buckling_follower.py Conducts the buckling path following and bifurcation analysis for the cantilevered beam under follower load. Dynamic stability exponents are also estimated here (results in section 5.2).

buckling_nonfollower.py Conducts the buckling path following and bifurcation analysis for the cantilevered beam under non-follower load. Dynamic stability exponents are also estimated here (results in section 5.2).

simpsupp_buckling.py This conducts the follower and non-follower tip load buckling analyses for the simply supported beam (results in section 5.3).

simpsupp_inhomogeneousbc.py Considers the buckling behavior of the simply supported beam under tip deflection boundary condition (results in section 5.4).

simpsupp_snapthrough.py Contains the scripts for the snap-through analyses conducted for section 5.5.

B Relevant functions and Derivatives

- Function ϵ_0 :

$$\begin{aligned}\epsilon_0 &= \frac{1}{2}(u_X'^2 + u_Y'^2 + 2u_X') \\ \frac{\partial \epsilon_0}{\partial \{d^e\}} &= \frac{1}{X_2 - X_1} \begin{pmatrix} -(1 + u_X') \\ -(u_Y') \\ 0 \\ (1 + u_X') \\ (u_Y') \end{pmatrix} \\ \left[\frac{\partial^2 \epsilon_0}{\partial d^{e^2}} \right] &= \frac{1}{(X_2 - X_1)^2} \left(\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right)\end{aligned}\quad (36)$$

with \otimes denoting the Kronecker product between matrices.

- Function γ_0 :

$$\begin{aligned}\gamma_0 &= -(1 + u_X') \sin \theta + u_Y' \cos \theta \\ \frac{\partial \gamma_0}{\partial \{d^e\}} &= \begin{pmatrix} \frac{\sin \theta}{X_2 - X_1} \\ -\frac{\cos \theta}{X_2 - X_1} \\ -((1 + u_X') \cos \theta + u_Y' \sin \theta) N_1 \\ -\frac{\sin \theta}{X_2 - X_1} \\ \frac{\cos \theta}{X_2 - X_1} \\ -((1 + u_X') \cos \theta + u_Y' \sin \theta) N_2 \end{pmatrix} \\ \left[\frac{\partial^2 \gamma_0}{\partial d^{e^2}} \right] &= \begin{bmatrix} 0 & 0 & \frac{N_1}{X_2 - X_1} \cos \theta & 0 & 0 & \frac{N_2}{X_2 - X_1} \cos \theta \\ 0 & 0 & \frac{N_1}{X_2 - X_1} \sin \theta & 0 & 0 & \frac{N_2}{X_2 - X_1} \sin \theta \\ \frac{N_1}{X_2 - X_1} \cos \theta & \frac{N_1}{X_2 - X_1} \sin \theta & f_1 & -\frac{N_1}{X_2 - X_1} \cos \theta & -\frac{N_1}{X_2 - X_1} \sin \theta & f_2 \\ 0 & 0 & -\frac{N_1}{X_2 - X_1} \cos \theta & 0 & 0 & -\frac{N_2}{X_2 - X_1} \cos \theta \\ 0 & 0 & -\frac{N_1}{X_2 - X_1} \sin \theta & 0 & 0 & -\frac{N_2}{X_2 - X_1} \sin \theta \\ \frac{N_2}{X_2 - X_1} \cos \theta & \frac{N_2}{X_2 - X_1} \sin \theta & f_2 & -\frac{N_2}{X_2 - X_1} \cos \theta & -\frac{N_2}{X_2 - X_1} \sin \theta & f_3 \end{bmatrix}\end{aligned}\quad (37)$$

Where the functions f_i' s are,

$$\begin{aligned}f_1 &= ((1 + u_X') \sin \theta - u_Y' \cos \theta) N_1^2 \\ f_2 &= ((1 + u_X') \sin \theta - u_Y' \cos \theta) N_1 N_2 \\ f_3 &= ((1 + u_X') \sin \theta - u_Y' \cos \theta) N_2^2\end{aligned}\quad (38)$$

- Function ϵ_1 :

$$\begin{aligned}
\epsilon_1 &= -\theta' [(1 + u'_X) \cos \theta + u'_Y \sin \theta] \\
\frac{\partial \epsilon_1}{\partial \{d^e\}} &= \left\{ \begin{array}{c} \frac{\theta' \cos \theta}{X_2 - X_1} \\ \frac{\theta' \sin \theta}{X_2 - X_1} \\ \theta' [(1 + u'_X) \sin \theta - u'_Y \cos \theta] N_1 + \frac{[(1 + u'_X) \cos \theta + u'_Y \sin \theta]}{X_2 - X_1} \\ -\frac{\theta' \cos \theta}{X_2 - X_1} \\ -\frac{\theta' \sin \theta}{X_2 - X_1} \\ \theta' [(1 + u'_X) \sin \theta - u'_Y \cos \theta] N_2 - \frac{[(1 + u'_X) \cos \theta + u'_Y \sin \theta]}{X_2 - X_1} \end{array} \right\} \\
\frac{\partial^2 \epsilon_1}{\partial d^{e^2}} &= \begin{bmatrix} 0 & 0 & g_1 & 0 & 0 & g_2 \\ 0 & 0 & g_3 & 0 & 0 & g_4 \\ g_1 & g_3 & g_5 & -g_1 & -g_3 & g_6 \\ 0 & 0 & -g_1 & 0 & 0 & -g_2 \\ 0 & 0 & -g_3 & 0 & 0 & -g_4 \\ g_2 & g_4 & g_6 & -g_2 & -g_4 & g_7 \end{bmatrix}
\end{aligned} \tag{39}$$

The functons g_i 's are,

$$\begin{aligned}
g_1 &= -\frac{\cos \theta}{\Delta X^2} - \frac{\theta' \sin \theta}{\Delta X} N_1 \\
g_2 &= \frac{\cos \theta}{\Delta X^2} - \frac{\theta' \sin \theta}{\Delta X} N_2 \\
g_3 &= -\frac{\sin \theta}{\Delta X^2} + \frac{\theta' \cos \theta}{\Delta X} N_1 \\
g_4 &= \frac{\sin \theta}{\Delta X^2} + \frac{\theta' \cos \theta}{\Delta X} N_2 \\
g_5 &= \theta' [(1 + u'_X) \cos \theta + u'_Y \sin \theta] N_1^2 - 2 \frac{N_1}{\Delta X} [(1 + u'_X) \sin \theta - u'_Y \cos \theta] \\
g_6 &= \theta' [(1 + u'_X) \cos \theta + u'_Y \sin \theta] N_1 N_2 - \frac{N_1 - N_2}{\Delta X} [(1 + u'_X) \sin \theta - u'_Y \cos \theta] \\
g_7 &= \theta' [(1 + u'_X) \cos \theta + u'_Y \sin \theta] N_2^2 + 2 \frac{N_2}{\Delta X} [(1 + u'_X) \sin \theta - u'_Y \cos \theta]
\end{aligned} \tag{40}$$

- Function ϵ_2 :

$$\begin{aligned}
\epsilon_2 &= \frac{1}{2} \theta'^2 \\
\frac{\partial \epsilon_2}{\partial \{d^e\}} &= \left\{ \begin{array}{c} 0 \\ 0 \\ -\frac{\theta'}{\Delta X} \\ 0 \\ \frac{\theta'}{\Delta X} \end{array} \right\} \\
\left[\frac{\partial^2 \epsilon_2}{\partial \{d^e\}^2} \right] &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\Delta X^2} & 0 & 0 & -\frac{1}{\Delta X^2} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\Delta X^2} & 0 & 0 & \frac{1}{\Delta X^2} \end{bmatrix}
\end{aligned} \tag{41}$$

C Finite Element Equations of Motion

For the two-noded finite element scheme, the equations of motion is given by,

$$\mathbf{D}(\{d\})\{\ddot{d}\} + \mathbf{C}(\{d\}, \{\dot{d}\})\{\dot{d}\} + F^{int}(\{d\}) = F^{ext} \quad (42)$$

The matrices \mathbf{D} & \mathbf{C} , referred to as the inertia matrix and the Chrostoffel symbols matrix, come from the kinetic energy density. After defining the shape functions as N_1 & N_2 ,

$$\begin{aligned} \frac{L_e}{2} \mathcal{T} &= \frac{1}{4} L_e \left(\rho A (u_X^2 + u_Y^2) + \rho I \dot{\theta}^2 \right) \\ &= \frac{L_e}{4} \begin{bmatrix} \dot{u}_X & \dot{u}_Y & \dot{\theta} \end{bmatrix} \begin{bmatrix} \rho A & 0 & 0 \\ 0 & \rho A & 0 \\ 0 & 0 & \rho I_2 \end{bmatrix} \begin{bmatrix} \dot{u}_X \\ \dot{u}_Y \\ \dot{\theta} \end{bmatrix} \\ &= \frac{L_e}{4} \{\dot{d}^e\}^T \underbrace{\begin{bmatrix} [N_1 \ N_2] \otimes \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{bmatrix}^T \begin{bmatrix} \rho A & 0 & 0 \\ 0 & \rho A & 0 \\ 0 & 0 & \rho I_2 \end{bmatrix} \begin{bmatrix} [N_1 \ N_2] \otimes \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{bmatrix}}_{\bar{\mathbf{D}}} \{\dot{d}^e\} \\ \mathbf{D} &= \int_{-1}^1 \bar{\mathbf{D}} \frac{L_e}{2} d\xi \\ \mathbf{C} &= \frac{1}{4} \int_{-1}^1 \mathbf{D} \dot{L}_e d\xi = \frac{1}{4} \int_{-1}^1 \left(\frac{\partial L_e}{\partial \{d^e\}}^T \{\dot{d}^e\} \right) \mathbf{D} d\xi \end{aligned} \quad (43)$$

The internal force vector $\{F^{int}\}$ is given by the partial derivative of the strain energy term with $\{d^e\}$.

$$\begin{aligned} F^{int} &= \frac{1}{2} \int_{-1}^1 \mathcal{U} \frac{\partial L_e}{\partial \{d^e\}} d\xi + \frac{1}{2} \int_{-1}^1 \frac{\partial \mathcal{U}}{\partial \{d^e\}} L_e d\xi \\ J^{int} &= \frac{1}{2} \int_{-1}^1 \mathcal{U} \frac{\partial^2 L_e}{\partial \{d^e\}^2} d\xi + \frac{1}{2} \left[\int_{-1}^1 \frac{\partial \mathcal{U}}{\partial \{d^e\}} \frac{\partial L_e}{\partial \{d^e\}}^T + \frac{\partial L_e}{\partial \{d^e\}} \frac{\partial \mathcal{U}}{\partial \{d^e\}}^T d\xi \right] + \frac{1}{2} \int_{-1}^1 \frac{\partial^2 \mathcal{U}}{\partial \{d^e\}^2} L_e d\xi \end{aligned} \quad (44)$$

The corresponding derivative quantities are,

$$\begin{aligned} \frac{\partial \mathcal{U}}{\partial \{d^e\}} &= (EA\epsilon_0 + EI_2\epsilon_2) \frac{\partial \epsilon_0}{\partial \{d^e\}} + (GA\gamma_0) \frac{\partial \gamma_0}{\partial \{d^e\}} + (EI_2\epsilon_1) \frac{\partial \epsilon_1}{\partial \{d^e\}} + (EI_2\epsilon_0 + EI_4\epsilon_2) \frac{\partial \epsilon_2}{\partial \{d^e\}} \\ \frac{\partial^2 \mathcal{U}}{\partial \{d^e\}^2} &= \left((EA\epsilon_0 + EI_2\epsilon_2) \frac{\partial^2 \epsilon_0}{\partial \{d^e\}^2} + (GA\gamma_0) \frac{\partial^2 \gamma_0}{\partial \{d^e\}^2} + (EI_2\epsilon_1) \frac{\partial^2 \epsilon_1}{\partial \{d^e\}^2} + (EI_2\epsilon_0 + EI_4\epsilon_2) \frac{\partial^2 \epsilon_2}{\partial \{d^e\}^2} \right) \\ &\quad + \left((EA \frac{\partial \epsilon_0}{\partial \{d^e\}} + EI_2 \frac{\partial \epsilon_2}{\partial \{d^e\}}) \frac{\partial \epsilon_0}{\partial \{d^e\}} + (GA \frac{\partial \gamma_0}{\partial \{d^e\}}) \frac{\partial \gamma_0}{\partial \{d^e\}} + (EI_2 \frac{\partial \epsilon_1}{\partial \{d^e\}}) \frac{\partial \epsilon_1}{\partial \{d^e\}} + (EI_2 \frac{\partial \epsilon_0}{\partial \{d^e\}} + EI_4 \frac{\partial \epsilon_2}{\partial \{d^e\}}) \frac{\partial \epsilon_2}{\partial \{d^e\}} \right) \end{aligned} \quad (45)$$

In the above notation, if two vectors A and B are given as AB , it denotes the product matrix AB^T .

D Third gradient Calculations

For the functions ϵ_0 , γ_0 , ϵ_1 , and ϵ_2 , the third derivatives are expressed analytically by expanding on the forms given earlier. The exact expressions are not provided here. The gradients of the matrix quantities are computed in tensorial form like the example below:

$$\begin{aligned}
\mathbf{D}_{il}\mathbf{D}_{lj}^{-1} &= \delta_{ij} \implies \mathbf{D}_{il,k}\mathbf{D}_{lj}^{-1} + \mathbf{D}_{il}\mathbf{D}_{lj,k}^{-1} = \mathbf{0}_{ijk} \\
\mathbf{D}_{il}\mathbf{D}_{lj,k}^{-1} &= -\mathbf{D}_{il,k}\mathbf{D}_{lj}^{-1} \implies \mathbf{D}_{mi}^{-1}\mathbf{D}_{il}\mathbf{D}_{lj,k}^{-1} = -\mathbf{D}_{mi}^{-1}\mathbf{D}_{il,k}\mathbf{D}_{lj}^{-1} \\
&\implies \mathbf{D}_{mj,k}^{-1} = -\mathbf{D}_{mi}^{-1}\mathbf{D}_{il,k}\mathbf{D}_{lj}^{-1}
\end{aligned}$$

The `einsum` method in `numpy` is employed to compute the tensor products ².

²<https://docs.scipy.org/doc/numpy/reference/generated/numpy.einsum.html>