Newton-Raphson scheme with sub-iterations and higher-order predictors for efficient nonlinear FEA simulations

Dr Chennakesava Kadapa
email: c.kadapa@napier.ac.uk

March 28, 2025

1 Introduction

The governing equations for the nonlinear elasticity are given by

$$-\nabla_{\boldsymbol{X}} \cdot \boldsymbol{P} = \boldsymbol{f}_0 \tag{1}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}}, \tag{2}$$

$$\boldsymbol{P} \cdot \boldsymbol{N}_0 = \bar{\boldsymbol{t}} \tag{3}$$

where $\boldsymbol{P}$ is the first Piola-Kirchoff stress.

By using the finite element discretisation, the semi-discrete equations for the nonlinear elasticity problem can be written as,

$$\mathbf{F}^{\text{int}}(\boldsymbol{u}) - \mathbf{F}^{\text{ext}} = \mathbf{0}, \tag{4}$$

where $\mathbf{F}^{\text{int}}(\boldsymbol{u})$ and $\mathbf{F}^{\text{ext}}$ are the vectors of internal and external forces.

The nonlinear equations (4) are solved using iterative schemes among which the Newton-Raphson scheme is one of the widely-used one. Using the Newton-Raphson scheme, the numerical solutions are computed iteratively, which requires the computation and inversion of the Jacobian matrix at each iteration in order to preserve its second-order convergence. While the computation and inversion of the Jacobian can be avoided in the modified Newton-Raphson scheme such a scheme requires smaller load steps for guaranteed convergence.

Often, it is impossible to obtain the numerical solution when the external load is applied all at once. Numerical solutions are computed by increasing the external load incrementally, and at every load increment, the nonlinear equations (4) are solved using the Newton-Raphson scheme.

Introducing the scalar parameter λ for controlling the amount of external load vector, the residual vector for the current load step $n+1$ can be written as,

$$\boldsymbol{R}(\boldsymbol{u}_{n+1}) = \mathbf{F}^{\text{int}}(\boldsymbol{u}_{n+1}) - \lambda_{n+1}\, \mathbf{F}^{\text{ext}} = \mathbf{0} \tag{5}$$

where $\boldsymbol{u}_{n+1}$ is the displacement field at the current load step. Starting with an initial guess $\boldsymbol{u}^1_{n+1}$, $\boldsymbol{u}_{n+1}$ is computed incrementally as

$$\boldsymbol{u}^{k+1}_{n+1} = \boldsymbol{u}^{k}_{n+1} + \Delta\boldsymbol{u}, \qquad \text{for} \quad k = 1, 2, 3, \dots \tag{6}$$

where k is the iteration count variable.

The parameter λ_{n+1} is specified by the user using the increment $\Delta\lambda$ from the previous load value λ_n, as

$$\lambda_{n+1} = \lambda_n + \Delta\lambda. \tag{7}$$

Using the Taylor series expansion of the internal force vector $\mathbf{F}^{\text{int}}(\boldsymbol{u}^{k+1}_{n+1})$ about $\boldsymbol{u}^k_{n+1}$, equation (5) can be written as,

$$\mathbf{F}^{\text{int}}(\boldsymbol{u}^k_{n+1}) + \left.\frac{\partial \mathbf{F}^{\text{int}}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{u}^k_{n+1}} \cdot \Delta\boldsymbol{u} - \mathbf{F}^{\text{ext}}_{n+1} = \mathbf{0} \tag{8}$$

where,

$$\mathbf{F}^{\text{ext}}_{n+1} = \lambda_{n+1}\,\mathbf{F}^{\text{ext}} \tag{9}$$

From (8), the incremental displacement Δu can be computed as,

$$\Delta\boldsymbol{u} = -\mathbf{K}^{-1}\,\mathbf{R}(\boldsymbol{u}^{k}_{n+1}), \tag{10}$$

where $\mathbf{K} = \left.\frac{\partial \mathbf{F}^{\text{int}}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{u}^{k}_{n+1}}$ is called as the stiffness matrix. Iterations are repeated until a predefined convergence criterion is satisfied.

The convergence of iterations is heavily dependent on the closeness of the predictor (initial guess) to the solution. Solution from the last converged load step ($\boldsymbol{u}_n$) is taken as the predictor. The pseudocode for the Newton-Raphson scheme is shown in Algorithm 1.

Algorithm 1 Algorithm for non-linear problems

```
Set: $\lambda_f$, $\Delta\lambda$, $n_{\max}$, $k_{\max}$ and tolerance $\epsilon$
for $n = 1$ to $n_{\max}$ do
    Compute: $\lambda_{n+1} = \lambda_n + \Delta\lambda$
    Predict: $\mathbf{u}^{(1)}_{n+1} = \mathbf{u}_n$
    for $k = 1$ to $k_{\max}$ do
        Compute: $\mathbf{K}(\boldsymbol{u}^{k}_{n+1})$ and $\mathbf{R}(\boldsymbol{u}^{k}_{n+1})$
        if $|\mathbf{R}(\boldsymbol{u}^{k}_{n+1})| \leq \epsilon$ then
            Converged, exit iteration loop
        end if
        Solve: $\mathbf{K}(\boldsymbol{u}^{(k)}_{n+1})\,\Delta\boldsymbol{u} = -\mathbf{R}(\boldsymbol{d}^{(k)}_{n+1})$
        Update: $\boldsymbol{u}^{(k+1)}_{n+1} = \boldsymbol{u}^{(k)}_{n+1} + \Delta\boldsymbol{u}$
    end for
    Store: $\mathbf{u}_n \leftarrow \mathbf{u}_{n+1}$
end for
```

Since u_{n+1} is closer to u_n for $n > 1$, the iterations are guaranteed to converge with the predictor $\boldsymbol{u}_{n+1} = \boldsymbol{u}_n$ as long as the load increment $\Delta\lambda$ is sufficiently small and assuming no critical/limit points on the load-displacement curve. However, there is an inherent problem in choosing the solution at the previously converged load step ($\boldsymbol{u}_n$) as the predictor for the current load step. For non-zero $\Delta\lambda$, solution at the current load step ($\boldsymbol{u}_{n+1}$) is certainly different from $\boldsymbol{u}_n$; therefore, $\boldsymbol{u}_n$ is not a sound choice as the initial guess for the current load step.

In the present work, the following predictors are proposed.

$$\boldsymbol{u}^{P_2} = 2\,\boldsymbol{u}_n - \boldsymbol{u}_{n-1} \tag{11}$$

$$\boldsymbol{u}^{P_3} = 3\,\boldsymbol{u}_n - 3\,\boldsymbol{u}_{n-1} + \boldsymbol{u}_{n-2} \tag{12}$$

$$\boldsymbol{u}^{P_4} = 4\,\boldsymbol{u}_n - 6\,\boldsymbol{u}_{n-1} + 4\,\boldsymbol{u}_{n-2} - \boldsymbol{u}_{n-3} \tag{13}$$

For the sake of completeness, $\boldsymbol{u}_n$ as the predictor is denoted as $\boldsymbol{u}^{P_1} = \boldsymbol{u}_n$. The proposed predictors are based on the higher-order forward different formulae, and the predictor is computed as an extrapolation of previous solutions. The above expressions are valid for uniform $\Delta\lambda$ values. For non-uniform load increments, see Fig. 1, which is common when adaptive load stepping is activated, the predictors need to be modified accordingly.

$$\boldsymbol{u}^{P_2} = \left(1 + \frac{l_1}{l_2}\right) \boldsymbol{u}_n - \left(\frac{l_1}{l_2}\right) \boldsymbol{u}_{n-1} \tag{14}$$

$$\boldsymbol{u}^{P_3} = (1 + b + c)\, \boldsymbol{u}_n - b\, \boldsymbol{u}_{n-1} - c\, \boldsymbol{u}_{n-2} \tag{15}$$

$$\tag{16}$$

where,

$$b = \frac{l_1\,(l_1 + l_2 + l_3)}{l_2\, l_3}, \qquad c = -\,\frac{l_1\,(l_1 + l_2)}{l_3\,(l_2 + l_3)}; \tag{17}$$

$$\boldsymbol{u}^{P_4} = (1 + b + c + d)\, \boldsymbol{u}_n - b\, \boldsymbol{u}_{n-1} - c\, \boldsymbol{u}_{n-2} - d\, \boldsymbol{u}_{n-3} \tag{18}$$

where,

$$b = \frac{l_1\,(l_1^2 + 2\,l_1\,l_2 + 2\,l_1\,l_3 + l_1\,l_4 + l_2^2 + 2\,l_2\,l_3 + l_2\,l_4 + l_3^2 + l_3\,l_4)}{l_2\,l_3\,(l_3 + l_4)}, \tag{19}$$

$$c = -\,\frac{l_1\,(l_1^2 + 2\,l_1\,l_2 + l_1\,l_3 + l_1\,l_4 + l_2^2 + l_2\,l_3 + l_2\,l_4)}{l_3\,l_4\,(l_2 + l_3)}, \tag{20}$$

$$d = \frac{l_1\,(l_1^2 + 2\,l_1\,l_2 + l_1\,l_3 + l_2^2 + l_2\,l_3)}{l_4\,(l_2\,l_3 + l_2\,l_4 + l_3^2 + 2\,l_3\,l_4 + l_4^2)} \tag{21}$$

with

$$l_1 = \Delta\lambda_{n+1}; \quad l_2 = \Delta\lambda_n; \quad l_3 = \Delta\lambda_{n-1}; \quad l_4 = \Delta\lambda_{n-2} \tag{22}$$

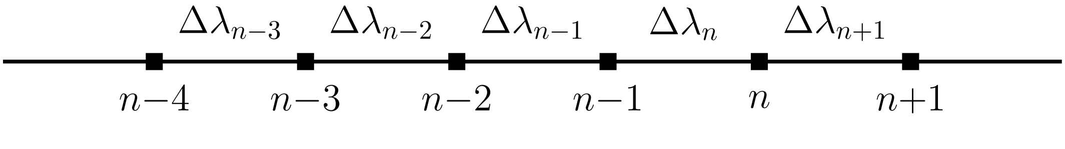

Figure 1: Sequence of load steps with non-uniform increments.

1.1 Newton-Raphson scheme

Let us consider the nonlinear equation

$$f(u) = 0 \tag{23}$$

where, f is a nonlinear function of a single variable u. Starting with an initial guess u_0, the root of (23) is obtained by using the Newton-Raphson scheme using following iterative form

$$u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)}, \qquad \text{for} \quad n = 0, 1, 2, \ldots \tag{24}$$

where, $f'(u_n)$ is the first derivative of f with respect to u evaluated at u_n. The iterations given by (24) converge quadratically provided that the initial guess is in the basin of attraction.

We can extend (24) for finding the roots of a set of (nonlinear) equations in multiple variables, $\boldsymbol{f}(\boldsymbol{u} = \mathbf{0}$, where $\boldsymbol{f} = \{f_1, f_2, \ldots, f_m\}$ and $\boldsymbol{u} = \{u_1, u_2, \ldots, u_m\}$, as

$$\boldsymbol{u}_{n+1} = \boldsymbol{u}_n - \mathbf{K}^{-1}\,\boldsymbol{f}(\boldsymbol{u}_n) \tag{25}$$

, where, $\mathbf{K}$ is the Jacobian matrix, which is computed as,

$$\mathbf{K} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial u_1} & \frac{\partial f_m}{\partial u_2} & \cdots & \frac{\partial f_m}{\partial u_m} \end{bmatrix} \tag{26}$$

NR scheme is one of the widely-used schemes in nonlinear FEM. To minimise the computational cost in computing the inverting the Jacobian matrix at every iteration for large-scale models, modified NR method in which the Jacobian is computed only at the first iteration is used in nonlinear FEM. Note, however, that the convergence order of such schemes is no longer quadratic.

1.2 Potra-Pták scheme

To find the root of (23), the Potra-Pták (PP) scheme [?] are given by

$$\left.\begin{aligned} v_n &= u_n - \tfrac{f(u_n)}{f'(u_n)} \\ u_{n+1} &= u_n - \tfrac{f(u_n)+f(v_n)}{f'(u_n)} \end{aligned}\right\}, \qquad \text{for} \quad n = 0, 1, 2, \ldots \tag{27}$$

The PP scheme makes use of the computed Jacobian for the second time to improve the accuracy of the solution at each iteration. Thus, the PP scheme might be interpreted as an extension of the NR scheme. For the purpose of computer implementation, the PP scheme can be considered as the NR scheme with two sub iteration at every iteration. Potra-Pták [?] demonstrated that the order of convergence of iterations given in (27) is three.

However, since the Jacobian is used twice at every iteration, the domain of convergence for the PP scheme is narrower when compared to the NR scheme; thus, requiring improved estimates for the initial guess.

1.3 Modified Potra-Pták scheme

In this work, we propose a modification to the PP scheme by adding one more sub-iteration to the original proposed scheme. Accordingly, the modified scheme is given by

$$\left.\begin{aligned} w_n &= u_n - \tfrac{f(u_n)}{f'(u_n)} \\ v_n &= u_n - \tfrac{f(u_n)+f(w_n)}{f'(u_n)} \\ u_{n+1} &= u_n - \tfrac{f(u_n)+f(w_n)+f(v_n)}{f'(u_n)} \end{aligned}\right\}, \qquad \text{for} \quad n = 0, 1, 2, \ldots \tag{28}$$

Iteration	$j_{\max} = 1$	$j_{\max} = 2$	$j_{\max} = 3$	$j_{\max} = 4$
1	5.3054 E+001	5.3054 E+001	5.3054 E+001	5.3054 E+001
2	1.9362 E+001	1.3296 E+001	1.0222 E+001	8.3206 E+000
3	6.9074 E+000 (1.0)	3.1051 E+000 (1.0)	1.7092 E+000 (1.1)	1.0297 E+000 (1.1)
4	2.3154 E+000 (1.1)	5.4355 E−001 (1.2)	1.2969 E−001 (1.4)	2.2806 E−002 (1.8)
5	6.5368 E−001 (1.2)	2.7454 E−002 (1.7)	9.4276 E−005 (2.8)	2.9138 E−009 (4.2)
6	1.1475 E−001 (1.4)	9.8732 E−006 (2.6)	4.0903 E−017 (3.9)	1.1009 E−043 (5.0)
7	5.7714 E−003 (1.7)	4.9267 E−016 (3.0)	1.4498 E−066 (4.0)	8.4758 E−216 (5.0)
8	1.6722 E−005 (1.9)	6.1218 E−047 (3.0)	2.2886 E−264 (4.0)	
9	1.4147 E−010 (2.0)	1.1744 E−139 (3.0)		
10	1.0125 E−020 (2.0)			
11	5.1867 E−041 (2.0)			
12	1.3610 E−081 (2.0)			
13	9.3720 E−163 (2.0)			

Table 1: Convergence of norm of the residual using the three different schemes for $f = \exp(u^2 + 7\,u - 30) - 1$ with $u_0 = 3.3$. i is the main iteration count and j is the sub-iteration count.

Iteration	$j_{\max} = 1$	$j_{\max} = 2$	$j_{\max} = 3$	$j_{\max} = 4$
1	$5.3054\ \times 10^{+001}$	$5.3054\ \times 10^{+001}$	$5.3054\ \times 10^{+001}$	$5.3054\ \times 10^{+001}$
2	$1.9362\ \times 10^{+001}$	$1.3296\ \times 10^{+001}$	$1.0222\ \times 10^{+001}$	$8.3206\ \times 10^{+000}$
3	$6.9074\ \times 10^{+000}$ (1.0)	$3.1051\ \times 10^{+000}$ (1.0)	$1.7092\ \times 10^{+000}$ (1.1)	$1.0297\ \times 10^{+000}$ (1.1)
4	$2.3154\ \times 10^{+000}$ (1.1)	$5.4355\ \times 10^{-001}$ (1.2)	$1.2969\ \times 10^{-001}$ (1.4)	$2.2806\ \times 10^{-002}$ (1.8)
5	$6.5368\ \times 10^{-001}$ (1.2)	$2.7454\ \times 10^{-002}$ (1.7)	$9.4276\ \times 10^{-005}$ (2.8)	$2.9138\ \times 10^{-009}$ (4.2)
6	$1.1475\ \times 10^{-001}$ (1.4)	$9.8732\ \times 10^{-006}$ (2.6)	$4.0903\ \times 10^{-017}$ (3.9)	$1.1009\ \times 10^{-043}$ (5.0)
7	$5.7714\ \times 10^{-003}$ (1.7)	$4.9267\ \times 10^{-016}$ (3.0)	$1.4498\ \times 10^{-066}$ (4.0)	$8.4758\ \times 10^{-216}$ (5.0)
8	$1.6722\ \times 10^{-005}$ (1.9)	$6.1218\ \times 10^{-047}$ (3.0)	$2.2886\ \times 10^{-264}$ (4.0)	
9	$1.4147\ \times 10^{-010}$ (2.0)	$1.1744\ \times 10^{-139}$ (3.0)		
10	$1.0125\ \times 10^{-020}$ (2.0)			
11	$5.1867\ \times 10^{-041}$ (2.0)			
12	$1.3610\ \times 10^{-081}$ (2.0)			
13	$9.3720\ \times 10^{-163}$ (2.0)			

Table 2: Convergence of norm of the residual using the three different schemes for $f = \exp(u^2 + 7\,u - 30) - 1$ with $u_0 = 3.3$. i is the main iteration count and j is the sub-iteration count.

Iteration	$j_{\max} = 1$	$j_{\max} = 2$	$j_{\max} = 3$	$j_{\max} = 4$
1	2.0645 E+001	2.0645 E+001	2.0645 E+001	2.0645 E+001
2	4.1669 E+000	1.9067 E+000	1.1331 E+000	7.5495 E−001
3	9.7827 E−001 (0.9)	2.0690 E−001 (0.9)	5.2400 E+002 (1.0)	1.1805 E−002 (1.3)
4	1.8642 E−001 (1.1)	4.3951 E−003 (1.7)	9.6936 E−006 (2.8)	5.5693 E−010 (4.0)
5	1.6714 E−002 (1.4)	8.7300 E−008 (2.8)	1.2193 E−020 (4.0)	9.6723 E−047 (5.0)
6	1.8807 E−004 (1.9)	6.0636 E−022 (3.0)	2.4941 E−080 (4.0)	1.1849 E−230 (5.0)
7	2.3824 E−008 (2.0)	1.8434 E−064 (3.0)	3.8147 E−319 (4.0)	
8	3.6951 E−016 (2.0)	4.8298 E−192 (3.0)		
9	8.6506 E−032 (2.0)			
10	4.6374 E−063 (2.0)			
11	1.3086 E−125 (2.0)			
12	1.0262 E−250 (2.0)			

Table 3: Convergence of norm of the residual using the three different schemes for $f_1 = u_1 + u_2 + u_3 - 3$; $f_2 = u_1^2 + u_2^2 + u_3^2 - 5$; $f_3 = \exp(u_1) + u_1\, u_2 - u_1\, u_3 - 1$; with $u_0 = [0.5,\, 5.0,\, 0.0]$. i is the main iteration count and j is the sub-iteration count.

NR			PP			PP-M		
i	j	$\lvert R\rvert$	i	j	$\lvert R\rvert$	i	j	$\lvert R\rvert$
1	1	2.064526E+01	1	1	2.064526E+01	1	1	2.064526E+01
2	1	4.166959E+00	1	2	4.166959E+00	1	2	4.166959E+00
3	1	9.782783E-01	2	1	1.906704E+00	1	3	1.906704E+00
4	1	1.864290E-01	2	2	4.249154E-01	2	1	1.133133E+00
5	1	1.671427E-02	3	1	2.069044E-01	2	2	2.354382E-01
6	1	1.880763E-04	3	2	2.116648E-02	2	3	1.050753E-01
7	1	2.382427E-08	4	1	4.395147E-03	3	1	5.240002E-02
8	1	4.965068E-16	4	2	1.403598E-05	3	2	1.905972E-03
			5	1	8.730011E-08	3	3	1.356704E-04
			5	2	3.614624E-15	4	1	9.693639E-06
						4	2	6.721270E-11
						4	3	9.155134E-16
Number of Jacobian factorisations								
7			5			4		

Table 4: Convergence of norm of the residual using the three different schemes for $f_1 = u_1 + u_2 + u_3 - 3$; $f_2 = u_1^2 + u_2^2 + u_3^2 - 5$; $f_3 = \exp(u_1) + u_1\, u_2 - u_1\, u_3 - 1$; with $u_0 = [0.5,\ 5.0,\ 0.0]$. i is the main iteration count and j is the sub-iteration count.

NR			PP			PP-M		
i	j	$\lvert R\rvert$	i	j	$\lvert R\rvert$	i	j	$\lvert R\rvert$
1	1	3.441156E+01	1	1	3.441156E+01	1	1	3.441156E+01
2	1	6.756503E+00	1	2	6.756503E+00	1	2	6.756503E+00
3	1	1.519694E+00	2	1	3.740711E+00	1	3	3.740711E+00
4	1	2.480969E-01	2	2	7.694325E-01	2	1	2.383320E+00
5	1	1.458661E-02	3	1	3.490065E-01	2	2	4.452440E-01
6	1	6.475655E-05	3	2	2.634494E-02	2	3	1.818573E-01
7	1	1.299243E-09	4	1	4.127506E-03	3	1	8.321461E-02
8	1	1.790181E-15	4	2	5.251887E-06	3	2	1.944991E-03
			5	1	1.337366E-08	3	3	9.198486E-05
			5	2	1.803899E-15	4	1	4.402890E-06
						4	2	6.006541E-12
						4	3	1.776357E-15
Number of Jacobian factorisations								
7			5			4		

Table 5: Convergence of norm of the residual using the three different schemes for $f_1 = 3\,u_1 - \cos(u_2\,u_3) - 0.5$; $f_2 = u_1^2 - 81\,(u_2 + 0.1)^2 + \sin(u_3) + 1.06$; $f_3 = \exp(-u_1\,u_2) + 20\,u_3 + \frac{(10\,\pi - 3)}{3}$; with $u_0 = [0.0,\, 0.5,\, 0.5]$. i is the main iteration count and j is the sub-iteration count.

1.4 Nonlinear FE examples

1.5 Geometrically exact beam in 2D

This example involves the bending of a cantilever beam under a prescribed end moment. This classic benchmark example has an analytical solution and has been used historically for demonstrating the accuracy of finite element formulations for nonlinear beam models, see [**?**, **?**, **?**].

The geometry and boundary conditions of the beam are as shown in Fig. 2 and the deformed shapes of the beam at different load values are shown in Fig. 3. The beam is of length $L = 10$ and of circular cross-section with diameter $d = 4$. The Young's modulus is $E = 79577$ and Poisson's ratio is $\nu = 0$. For the FEA, 20 geometrically-exact beam elements, see [**?**], are considered. The shear correction factor is $\kappa = 1$. The total amount of moment applied is $M = 20\,\pi$.

With a load step size of $\Delta\lambda = \pi$ and using the predictor $\boldsymbol{u}^{P_1}$, this problem requires 120 Jacobian factorisations using the standard NR scheme, i.e., with one sub-iterations. Increasing the number of sub-iterations results in divergence of iterations. With a reduced load step size of $\Delta\lambda = \pi/10$, a total of 800 Jacobian factorisations are required using the standard NR scheme; increasing the number of sub-iterations not only does not help but also results in divergence for two and four sub-iterations.

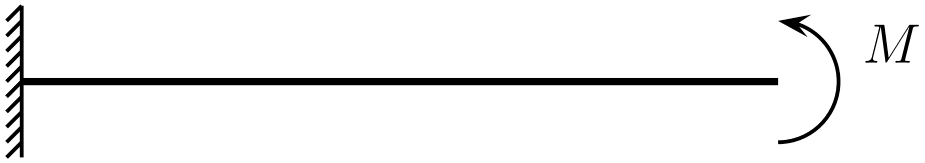

Figure 2: Beam 1D model: geometry and boundary conditions.

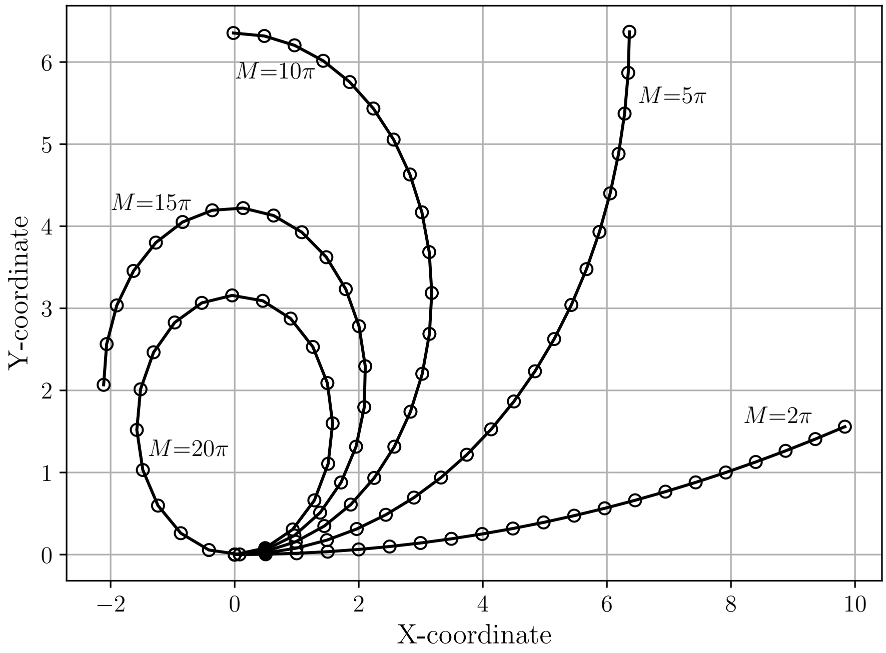

Figure 3: Beam 1D model: deformed shapes at different load values.

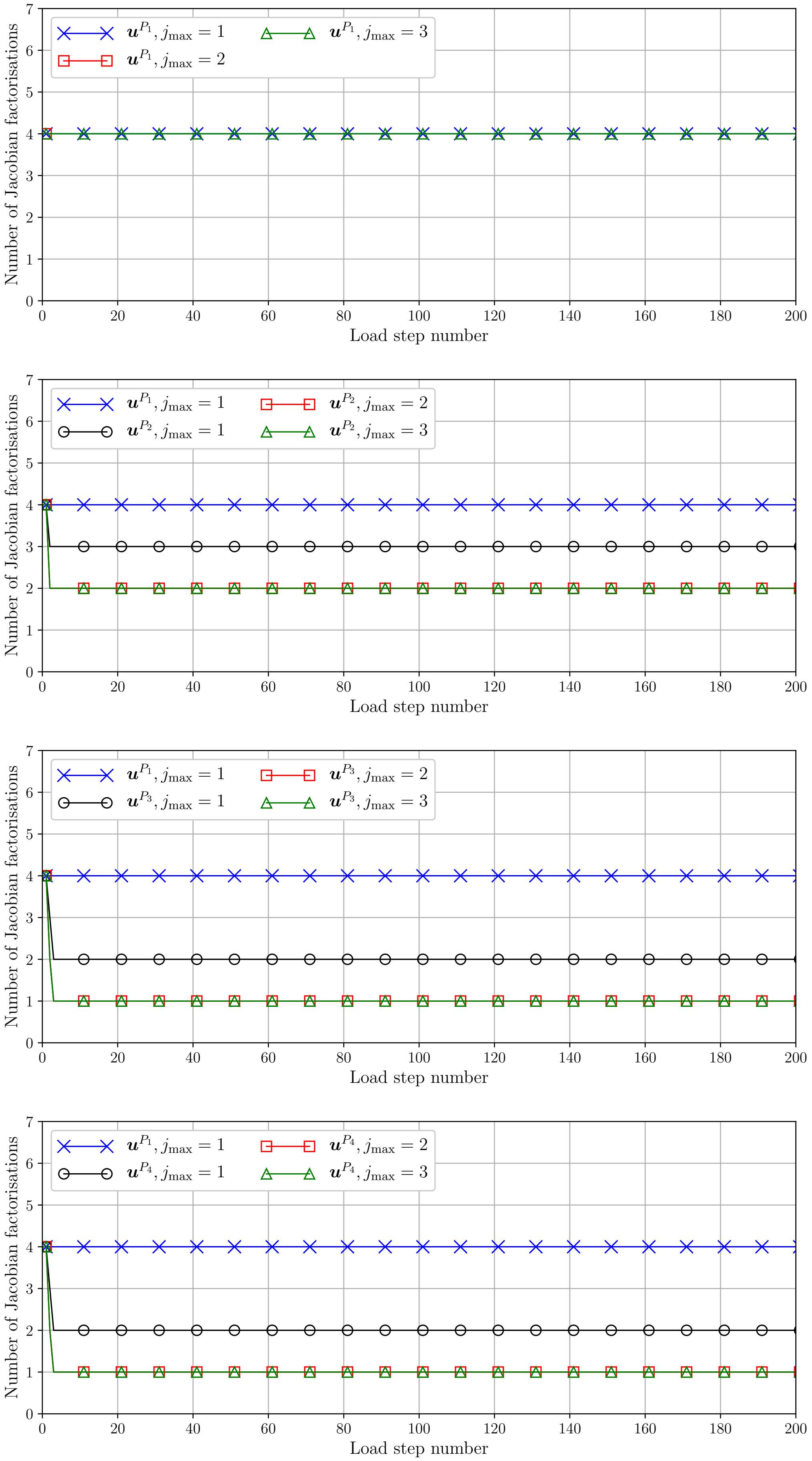

Figure 4: Beam 1D model: number of Jacobian factorisations at each load step for different cases with $\Delta\lambda = \pi/10$. Note that only every 10th instant is marked.

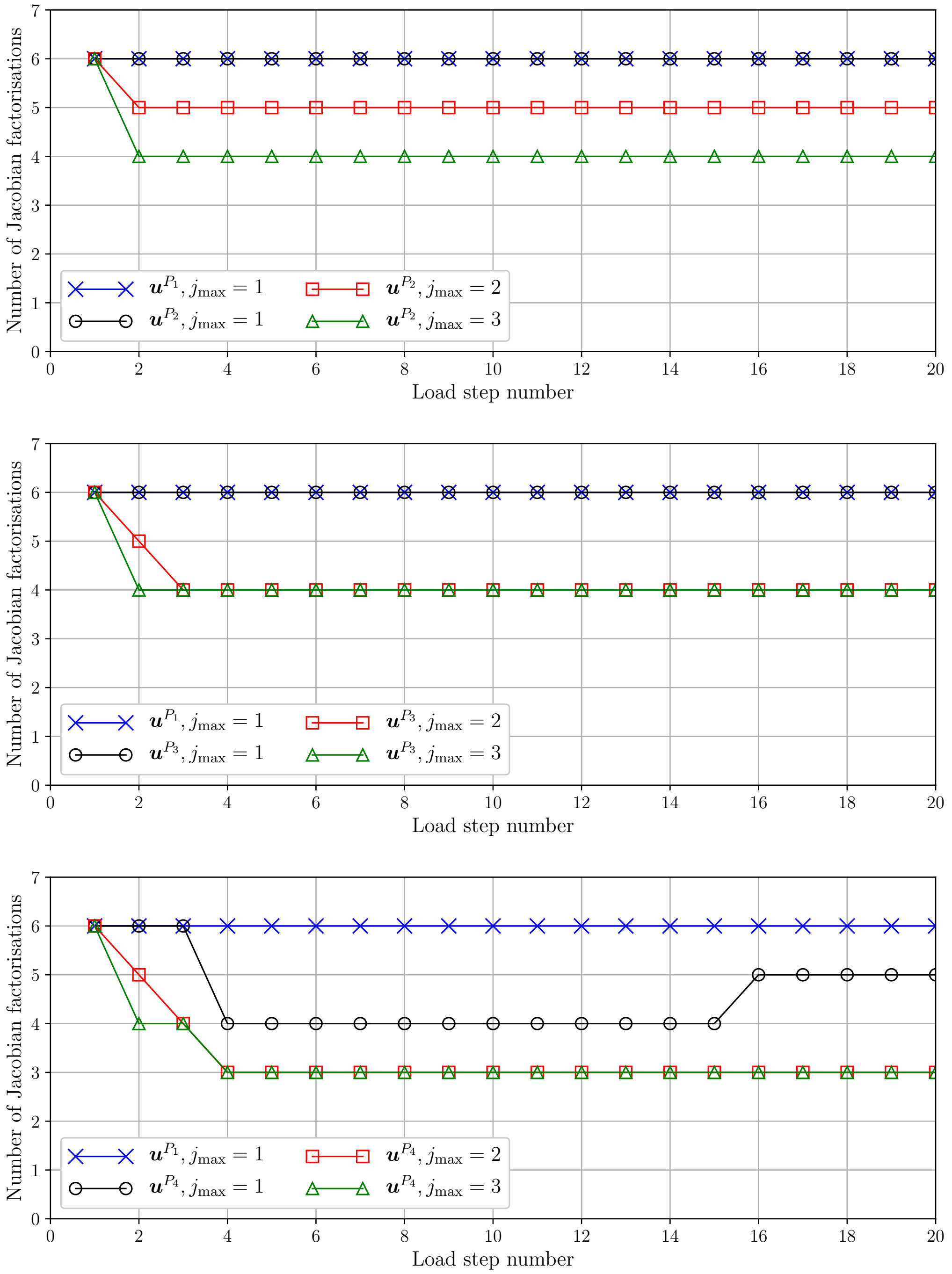

Figure 5: Beam 1D model: number of Jacobian factorisations at each load step for different cases with $\Delta\lambda = \pi$.

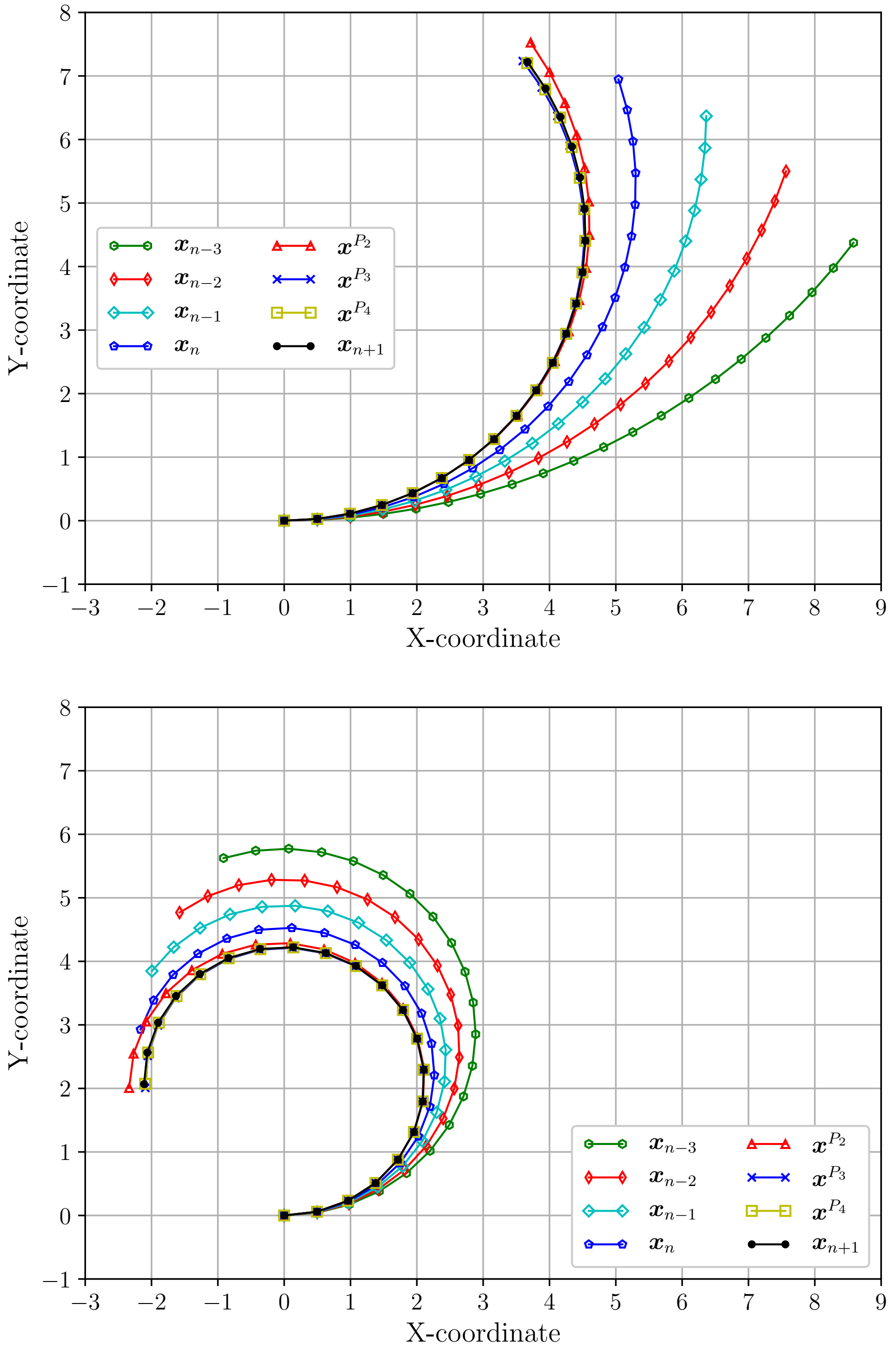

Figure 6: Beam 1D model: deformed shapes at different load values.