Course in Nonlinear FEM

Geometric nonlinearity
Outline

Lecture 1 – Introduction
Lecture 2 – Geometric nonlinearity
Lecture 3 – Material nonlinearity
Lecture 4 – Material nonlinearity continued
Lecture 5 – Geometric nonlinearity revisited
Lecture 6 – Issues in nonlinear FEA
Lecture 7 – Contact nonlinearity
Lecture 8 – Contact nonlinearity continued
Lecture 9 – Dynamics
Lecture 10 – Dynamics continued
Nonlinear FEM

Lecture 1 – Introduction, Cook [17.1]:
  – Types of nonlinear problems
  – Definitions
Lecture 2 – Geometric nonlinearity, Cook [17.10, 18.1-18.6]:
  – Linear buckling or eigen buckling
  – Prestress and stress stiffening
  – Nonlinear buckling and imperfections
  – Solution methods
Lecture 3 – Material nonlinearity, Cook [17.3, 17.4]:
  – Plasticity systems
  – Yield criteria
Lecture 4 – Material nonlinearity revisited, Cook [17.6, 17.2]:
  – Flow rules
  – Hardening rules
  – Tangent stiffness
Nonlinear FEM

Lecture 5 – Geometric nonlinearity revisited, Cook [17.9, 17.3-17.4]:
- The incremental equation of equilibrium
- The nonlinear strain-displacement matrix
- The tangent-stiffness matrix
- Strain measures

Lecture 6 – Issues in nonlinear FEA, Cook [17.2, 17.9-17.10]:
- Solution methods and strategies
- Convergence and stop criteria
- Postprocessing/Results
- Troubleshooting
Nonlinear FEM

Lecture 7 – Contact nonlinearity, Cook [17.8]:
   – Contact applications
   – Contact kinematics
   – Contact algorithms

Lecture 8 – Contact nonlinearity continued, Cook [17.8]:
   – Issues in FE contact analysis/troubleshooting

Lecture 9 – Dynamics, Cook [11.1-11.5]:
   – Solution methods
   – Implicit methods
   – Explicit methods

Lecture 10 – Dynamics continued, Cook [11.11-11.18]:
   – Dynamic problems and models
   – Damping
   – Issues in FE dynamic analysis/troubleshooting
Typical Nonlinear Problem
1 D-O-F

\[ k = k_0 + k_N \]

- \( k_0 \) constant
- \( k_N \) function of \( u \)
Problem Statement

\[
\left( k_0 + k_N \right) u = P
\]

\[
k_N = f(u)
\]

Given \( P \) find \( u \).
Assume \( f(u) \) is a known function.
Geometric nonlinearity

Computational Mechanics, AAU, Esbjerg

Slope $k_0$

Hardening
$k_N > 0$

Softening
$k_N < 0$

$(k_N = 0)$
Definitions - PSTRES vs. SSTIF

The difference between PSTRES and SSTIF?

- SSTIF and PSTRES are essentially the same in what they do (calculate stress stiffness matrix). SSTIF triggers nonlinear equilibrium iterations, however, so this is where they differ.

- **PSTRES**
  - Solve \([K]x = F\)
  - Calculate and store \([K_{\sigma}]\)
  - (this means that "x" is solved for based on \([K]\) only)

- **SSTIF**
  - Solve \([K^{*}]x = F\)
  - Calculate and store \([K_{\sigma}]\)
  - Solve \([K+K_{\sigma}]^{*}x=F\)
  - iterate until force equilibrium is satisfied
  - (this means that "x" and corresponding stresses are based on updated \([K+K_{\sigma}]\))

- Although both PSTRES and SSTIF calculate \([K_{\sigma}]\), the latter triggers nonlinear equilibrium iterations, so that probably is where your difference in results lies.
Definitions

- Stress stiffening (only) - Strains and rigid-body motions are assumed to be small, and stiffening (or softening) of the structure due to the stress state is taken into account.

- Stress stiffening is needed for structures which are thin in one (or two) dimensions (e.g., bending stiffness of beams and shells small compared with in-plane/transverse stiffness) since a coupling between the two occurs (examples - membrane of a drum or a guitar string).

- It is also needed when doing other types of analyses like buckling or modal where the stress state affects the response of the system.

- In ANSYS, stress stiffening terms are constant, so be aware of this assumption (since in prestress modal or eigenvalue buckling, analyses are linear too).
Definitions

• Stress stiffening may also be known as geometric stiffness matrix, differential stiffness matrix, stability coefficient matrix, initial stress stiffness matrix, incremental stiff matrix, etc.

• Note that the commands SSTIF and PSTRES essentially do the same thing, but are used in different situations (PSTRES is used to request that a stress-stiffening matrix be written for use in a future eigenvalue buckling or pre-stressed modal analysis).

• For the 18x family of elements, stress-stiffening is not available independent of large deflection. For other elements, the decision to include stress-stiffening with large deflection is generally based on ease of convergence, since large deflection and stress-stiffening are redundant.
Definitions

Consistent Tangent Stiffness Matrix (CTS Matrix, for short)

- Matrix used in nonlinear problems which is comprised of
  - the "main tangent stiffness matrix" (the "regular" stiffness matrix we think of, especially in linear analyses),
  - the "initial displacement matrix" (accounts for shape changes in elements),
  - the "initial stress matrix" (the stress stiffening matrix; stiffness due to stress state), and
  - the "initial load matrix" (stiffness associated with change in follower force loads during deformation - pressure load stiffness for elements 154/181/188/189).

- The 181/188/189 elements always use a fully consistent tangent stiffness matrix. The rest of the 18x family of elements does include a stress stiffening matrix, but not necessarily a fully CTS matrix.
Definitions

• Hierarchy of terms:
  – finite strain (or large strain)
  – large deformation (or large rotation)
  – stress stiffening (there's also spin softening, which is not discussed here)
  – regular linear analysis

• So (1) encompasses (2)-(4), (2) encompasses (3)-(4), etc. This may be a bit of an oversimplification, but it may help to think of it in these terms. When SSTIF is not ON with NLGEOM,ON, it just leads to slower convergence; but that effect is included in the overall response/results (again, thin beams and shells may be exceptions).

• Note that, if you use NLGEOM,ON, SSTIF,ON will automatically be activated (at least for v5.5 and above). The defaults should be left alone (i.e., you do not need to manually activate SSTIF,ON). SSTIF,ON helps aid convergence, especially for thin beams and shells. Although it has no effect for the 18x elements, as noted above, it's just good practice to leave the defaults (i.e, default is SSTIF,ON when NLGEOM is ON)
Objectives

1. Review buckling theory
2. Eigenvector/value calculation
3. FE Buckling analysis
What is buckling?

- An element under compression will have different mechanical responses relating to the following factors:
  - Length vs cross-sectional dimensions
  - Eccentricity of loads
  - Applied load
  - etc
What is buckling?

- A short, thick element with a centrally balanced load will more than likely experience **simple** compression.

- Failure (if reached) will be due to the compressive loads only.
What is buckling?

- A thin member under high and/or eccentric loads is more likely to deflect in a different manner – it will buckle if the loading reaches a critical value \( (P_{cr}) \)
What is buckling?

• The eccentric loading and/or small imperfections in the material cause out of line displacement
What is buckling?

• Obviously the strength of the member is greatly reduced
• failure often follows with loading $> P_{cr}$
Euler columns

- Basic analysis of circular columns with pinned ends
- Assuming the element bends towards the positive direction – the resultant moment is the negative of load x distance:
  \[ M = -Py \]
Euler columns

• From basic beam theory

\[ \frac{M}{EI} = \frac{d^2 y}{dx^2} = - \frac{P}{EI} y \]

\[ \Rightarrow \frac{d^2 y}{dx^2} + \frac{P}{EI} y = 0 \]

• Which has the simple solution:

\[ y = A \sin \sqrt{\frac{P}{EI}} x + B \cos \sqrt{\frac{P}{EI}} x \]
Euler columns

- Applying the boundary conditions
  \( y = 0 \) at \( x = 0 \) and at \( x = L \)
- Therefore \( B = 0 \)
- We have the trivial solution \( A = 0 \)
- But otherwise

\[
\sin \sqrt{\frac{P}{EI}} x = 0
\]
Euler columns

• Giving:
\[ \sqrt{\frac{P}{EI}}L = N\pi \]

• Where \( N \) is an integer

• Therefore we have discrete solutions to the problem – similar shapes to harmonics in a guitar string
Euler columns

- For $N=1$ we can determine the minimal load that buckling can occur

$$\sqrt{\frac{P_{cr}}{EI}} L = \pi \Rightarrow P_{cr} = \frac{\pi^2 EI}{L^2}$$

- Which is the Euler column buckling formula
Euler columns

• Looking at the deflection (y)

\[ y = A \sin \left( \frac{\pi x}{L} \right) \]

• Which is half of the sine curve:
Euler columns

- From this simple example we can see that multiple (though discrete) solutions can be possible for a certain geometry and loading direction.
- Loading above the critical load could lead to buckling with higher order shapes (N>1)
- This means that the “steady state” problems we have been considering could \textit{bifurcate} if loading is slightly changed
Boundary conditions (BC’s)

- Boundary conditions will affect the shape significantly:

Nonlinear FEM
Computational Mechanics, AAU, Esbjerg

Geometric nonlinearity
Eigenvectors/values

• Just before looking at the FE formulation of buckling – the eigenvector analysis is revised

• If a problem can be written in the following form:

\[ \mathbf{KU} = \lambda \mathbf{U} \]

– for scalar (eigenvalue) \( \lambda \)
– Matrix \( \mathbf{K} \) (n x n)
– Eigenvector \( \mathbf{U} = U_1, U_2, \ldots U_n \)
Eigenvectors/values

• Re-arranging

\[ \mathbf{KU} - \lambda \mathbf{U} = 0 \]

• Expanding the matrix (e.g. 2 x 2 matrix)

\[
\begin{bmatrix}
k_{11} & k_{12} \\
k_{21} & k_{22}
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2
\end{bmatrix}
- \lambda
\begin{bmatrix}
u_1 \\
u_2
\end{bmatrix}
= 
\begin{bmatrix}
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
k_{11} - \lambda & k_{12} \\
k_{21} & k_{22} - \lambda
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2
\end{bmatrix}
= 
\begin{bmatrix}
0 \\
0
\end{bmatrix}
\]
Eigenvectors/values

• We have a homogeneous problem – which only has a non-trivial solution if the matrix determinant is zero

\[
\begin{vmatrix}
 k_{11} - \lambda & k_{12} \\
 k_{21} & k_{22} - \lambda
\end{vmatrix} = 0
\]

• Giving us a polynomial to solve for \( \lambda \)

\[(k_{11} - \lambda) (k_{22} - \lambda) - k_{12}k_{21} = 0\]
Eigenvectors/values

\[ \lambda^2 - (k_{11} + k_{22}) \lambda + k_{11} k_{22} - k_{12} k_{21} = 0 \]

- We can solve for the eigenvalues (\( \lambda_i \)) and determine possible relating eigenvectors

\[
\begin{bmatrix}
  k_{11} - \lambda & k_{12} \\
  k_{21} & k_{22} - \lambda
\end{bmatrix}
\begin{bmatrix}
  u_1 \\
  u_2
\end{bmatrix} = \begin{bmatrix}
  0 \\
  0
\end{bmatrix}
\]
FE Formulation

• Of course there are more complex theories of buckling but we will look at a discrete element and then a FE formulation

• As with the previous example, we need to assume there will be out of line deflection

• We either start with a slight bend (as before) or introduce a small force (or displacement) in a direction that will induce the desired deflection
FE Formulation

- Consider the following idealisation of a column under a compressive load. The springs represent the “desire” of the structure to retain its shape.
FE Formulation

- Consider the following idealisation of a column under a compressive load. The springs represent the “desire” of the structure to retain its shape.

\[
\begin{align*}
\text{Force Equilibrium} \\
F = kU_1 + kU_2
\end{align*}
\]
FE Formulation

Consider the following idealisation of a column under a compressive load. The springs represent the “desire” of the structure to retain its shape.
FE Formulation

- Moment equilibrium of the bottom bar gives:
  \[ FL \sin(\beta) + k_T \alpha = kU_1 L \cos(\beta) \]

- Moment equilibrium about the centre
  \[ FL[\sin(\beta) + \sin(\beta+\alpha)] = kU_1 L[\cos(\beta)+\cos(\alpha+\beta) + kU_2 L \cos(\beta)] \]

- Assuming that displacements are small we can relate \( L \sin(\ldots) \) to \( U_1 \) and \( U_2 \)
FE Formulation

• Giving the following matrix:

\[
\begin{bmatrix}
    kL + \frac{k_r}{L} & -\frac{2k_r}{L} \\
    -\frac{2k_r}{L} & kL + \frac{4k_r}{L}
\end{bmatrix}
\begin{bmatrix}
    u_1 \\
    u_2
\end{bmatrix} =
\begin{bmatrix}
    1 & -1 \\
    -1 & 2
\end{bmatrix}
\begin{bmatrix}
    u_1 \\
    u_2
\end{bmatrix}
\]

• Now this is not of the form \(K\mathbf{U} = \lambda\mathbf{U}\) BUT is still classified as an eigenvalue problem if both square matrices are symmetric.
FE Formulation

• Giving the following matrix:

\[
\begin{bmatrix}
kL + \frac{k_r}{L} & - \frac{2k_r}{L} \\
- \frac{2k_r}{L} & kL + \frac{4k_r}{L}
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2
\end{bmatrix} = F
\begin{bmatrix}
1 & -1 \\
-1 & 2
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2
\end{bmatrix}
\]

• Solving for the lowest eigenvalue $F_1$ will give the critical load
FE Formulation

• FE Beam formulation:

We introduce a linear perturbation ($\beta F$) and look at the change in stiffness and loading on the structure using nonlinear analysis
FE Formulation

- FE Beam formulation:

As the structure deforms it is assumed that the stiffness of the structure is a linear function of those of previous iterations. For time \( \tau \):

\[
\tau K = t^{\Delta t}K + \lambda(tK - t^{\Delta t}K)
\]
FE Formulation

• FE Beam formulation:

  • For non zero vectors $\mathbf{u}$
    \[ \tau \mathbf{K} \mathbf{u} = 0 \]
    • this tangent stiffness is singular due to the buckling

• Therefore:
  \[ t^{-\Delta t} \mathbf{K} \mathbf{u} = \lambda (t^{-\Delta t} \mathbf{K} - t^0 \mathbf{K}) \mathbf{u} \]

• Where $\mathbf{u}$ are the possible buckling modes of the problem
FE Formulation

• FE Beam formulation:

\[ \tau R = \Delta t R + \lambda (R_t - \Delta t R) \]

Also the loading will vary through the analysis. For time \( \tau \):
FE Formulation

- Therefore we have eigenvalue problems to solve for each step of the iteration.
- If we choose a value of $\beta$ that is sufficiently small then results will be very close to those predicted by the Euler column formula.
- The nonlinear iteration method will be introduced further later.
Buckling Summary

- Buckling in structure? (eccentricity of loading, thin/thick, material imperfections)
- Simple beams under a buckling load can be analysed with Euler’s theory
- Rigid bar formulation leading to eigenvalue problem
- Introduced the FE formulation of a linear perturbation giving eigenvalue problems for each iteration.
EIGEN BUCKLING

\[ P_{cr} = \frac{\pi^2 EI}{(L_e)^2} \]

\( I = 2\text{nd Moment of Area about weak axis.} \)
\( E = \text{Young’s Modulus} \)

Load, \( P \)

Deflected shape

Le
The effective length, $L_e$, depends on the Boundary Conditions:

<table>
<thead>
<tr>
<th>Theoretical</th>
<th>$L_e = L$</th>
<th>$L_e = 0.707L$</th>
<th>$L_e = 0.5L$</th>
<th>$L_e = L$</th>
<th>$L_e = 2L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum AISC</td>
<td>$L_e = L$</td>
<td>$L_e = 0.80L$</td>
<td>$L_e = 0.65L$</td>
<td>$L_e = 1.2L$</td>
<td>$L_e = 2.1L$</td>
</tr>
</tbody>
</table>

Geometric nonlinearity
Computational Mechanics, AAU, Esbjerg
Find the *Buckling load* for a pin-ended aluminum column 3m high, with a rectangular x-section as shown:

**Weak axis:**
\[
I_{yy} = 100 \frac{(50)^3}{12} = 1.04 \times 10^6 \text{ mm}^4
\]

\[
P_{cr} = \frac{\pi^2 (72000)(1.04 \times 10^6)}{(3000)^2}
\]

\[
= 82246 \text{ N}
\]
Elastic Buckling Prediction

• Numerical Methods
  – finite element, finite strip ([www.ce.jhu.edu/bschafer](http://www.ce.jhu.edu/bschafer))

• Hand Methods (for use in a traditional Specification)
  – Local Buckling
    • Element methods, e.g. k=4
    • Semi-empirical methods that include element interaction
  – Distortional Buckling
    • Proposed (Schafer) method, rotational stiffness at web/flange juncture
    • Hancock’s method
    • AISI (k for Edge Stiffened Elements per Spec. section B4.2)
## Elastic Buckling Comparisons

<table>
<thead>
<tr>
<th></th>
<th>Local</th>
<th>Distortional</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(fcr)true</td>
<td>(fcr)true</td>
</tr>
<tr>
<td></td>
<td>(fcr)element</td>
<td>(fcr)interact</td>
</tr>
<tr>
<td>All Data</td>
<td>avg.</td>
<td>1.34</td>
</tr>
<tr>
<td></td>
<td>st.dev.</td>
<td>0.13</td>
</tr>
<tr>
<td></td>
<td>max</td>
<td>1.49</td>
</tr>
<tr>
<td></td>
<td>min</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>count</td>
<td>149</td>
</tr>
</tbody>
</table>

(fcr)true = local or distortional buckling stress from finite strip analysis  
(fcr)element = minimum local buckling stress of the web, flange and lip via Eq.'s 1-3  
(fcr)interact = minimum local buckling stress using the semi-empirical equations (Eq.'s 4-6)  
(fcr)Schafer = distortional buckling stress via Eq.'s 7-15  
(fcr)Hancock = distortional buckling stress via Lau and Hancock (1987)  
(fcr)AISI = buckling stress for an edge stiffened element via AISI (1996) from Desmond et al. (1981)

1 For a wide variety of cold-formed steel lipped channels, zees and racks

*For members with slender webs and small flanges the Lau and Hancock (1987) approach conservatively converges to a buckling stress of zero (these members are ignored in the summary statistics given above)
Effective Width Methods

$$A_{\text{eff}} = \sum b_{\text{eff}} t$$

$$\frac{b_{\text{eff}}}{b} = \left(1 - 0.22 \sqrt{\frac{f_{\text{cr} \ell}}{f}}\right) \left(\sqrt{\frac{f_{\text{cr} \ell}}{f}}\right) \text{ for } \sqrt{\frac{f}{f_{\text{cr} \ell}}} > 0.673 , \text{ else } b_{\text{eff}} = b . (17)$$

where:  
- $b_{\text{eff}}$ is the effective width of an element with gross width $b$
- $f$ is the yield stress ($f = f_y$) when interaction with other modes is not considered, otherwise $f^*$ is the limiting stress of a mode interacting with local buckling
- $f_{\text{cr} \ell}$ is the local buckling stress

* for Euler (long column) interaction $f = F_{cr \ell}$ of the column curve used in AISC Spec. (the notation for $f$ is $F_n$ in the AISI Spec. but the same column curve is employed) 

Nonlinear FEM  
Computational Mechanics, AAU, Esbjerg
Direct Strength Methods

Local

\[
\frac{P_n}{P} = \left( 1 - 0.15 \left( \frac{P_{cr\ell}}{P} \right)^4 \right) \left( \frac{P_{cr\ell}}{P} \right)^4 \quad \text{for} \quad \sqrt{\frac{P}{P_{cr\ell}}} > 0.776, \quad \text{else} \quad P_n = P. \quad (19)
\]

where: \( P_n \) is the nominal capacity
\( P \) is the squash load \( (P = P_y = A_g f_y) \) except when interaction with other modes is considered, then \( P = A_g f \), where \( f \) is the limiting stress of the interacting mode.
\( P_{cr\ell} \) is the critical elastic local buckling load \( (A_g f_{cr\ell}) \)

Distortional

\[
\frac{P_n}{P} = \left( 1 - 0.25 \left( \frac{P_{crd}}{P} \right)^6 \right) \left( \frac{P_{crd}}{P} \right)^6 \quad \text{for} \quad \sqrt{\frac{P}{P_{crd}}} > 0.561, \quad \text{else} \quad P_n = P. \quad (16)
\]

where: \( P_n \) is the nominal capacity in distortional buckling
\( P \) is the squash load \( (P = P_y = A_g f_y) \) when interaction with other modes is not considered, otherwise \( P = A_g f \), where \( f \) is the limiting stress of a mode that may interact
\( P_{crd} \) is the critical elastic distortional buckling load \( (A_g f_{crd}) \)
Some Solution Methods

1. Direct Substitution
2. Direct Substitution with Relaxation
3. Newton-Raphson (N-R)
4. Modified Newton-Raphson
5. Incremental Methods
6. Quasi-Newton Methods (Inverse Broyden)
Newton-Raphson (NR)

NAFEMS p. 248: Figure 5.4.

Figure 5.4 Standard Newton-Raphson method (NR) — the nth load increment \((f^n - f^{n-1})\) is applied — then using the tangential stiffness matrix \(K^n_{T0}\), the iterative displacements \(\delta d^n_0\) are found and hence the residual forces \(r^n_0\) — the tangential matrix \(K^n_{T1}\) is then evaluated and the iterative displacements \(\delta d^n_1\) and residual forces \(r^n_2\) are found — this process is repeated until convergence is achieved.
The arc-length method

\[ \lambda_i, \lambda_n \]

\[ \Delta \lambda, \Delta u_i^I, \Delta \lambda_i^I, \Delta u_{n} \]

\[ (n+1) \text{ converged solution} \]

\[ \text{spherical arc at substep } n \]

\[ u_n (\text{converged solution at substep } n) \]
Direct Substitution Method

1. Let load $P_A$ be applied to a softening spring ($k_N < 0$)
2. Assume $k_N = 0$ for the first iteration.
3. Compute first approximation to displacement: $u_1 = \frac{P_A}{k_0}$
4. Use $u_1$ to compute new stiffness:
   $$k = k_0 + f(u_1)$$
5. Compute next approximation to displacement: $u_2 = \frac{P_A}{k}$
6. Generate sequence of approximations.
Sequence of Operations

\[ u_1 = k_0^{-1} P_A \]
\[ u_2 = \left( k_0 + k_{N_1} \right)^{-1} P_A \]
\[ u_3 = \left( k_0 + k_{N_2} \right)^{-1} P_A \]
\[ \vdots \]
\[ u_{i+1} = \left( k_0 + k_{N_i} \right)^{-1} P_A \]
Geometric nonlinearity

Computational Mechanics, AAU, Esbjerg
Essentially a secant method
Geometric nonlinearity

Computational Mechanics, AAU, Esbjerg
Example:

\[ k = 0.2 - u \]

\[ P = 0.006 \]

\[ u_1 \]

<table>
<thead>
<tr>
<th>k</th>
<th>u</th>
<th>Del u</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2000000000</td>
<td>0.0300000000</td>
<td></td>
</tr>
<tr>
<td>0.1700000000</td>
<td>0.0352941176</td>
<td>15.000000000%</td>
</tr>
<tr>
<td>0.1647058824</td>
<td>0.0364285714</td>
<td>3.11418685%</td>
</tr>
<tr>
<td>0.1635714286</td>
<td>0.0366812227</td>
<td>0.68877551%</td>
</tr>
<tr>
<td>0.1633187773</td>
<td>0.0367379679</td>
<td>0.15445930%</td>
</tr>
<tr>
<td>0.1632620321</td>
<td>0.0367507370</td>
<td>0.03474506%</td>
</tr>
<tr>
<td>0.1632492630</td>
<td>0.0367536116</td>
<td>0.00782121%</td>
</tr>
<tr>
<td>0.1632463884</td>
<td>0.0367542587</td>
<td>0.00176085%</td>
</tr>
<tr>
<td>0.1632457413</td>
<td>0.0367544045</td>
<td>0.00039645%</td>
</tr>
<tr>
<td>0.1632455955</td>
<td>0.0367544373</td>
<td>0.00008926%</td>
</tr>
<tr>
<td>0.1632455627</td>
<td>0.0367544447</td>
<td>0.00002010%</td>
</tr>
<tr>
<td>0.1632455553</td>
<td>0.0367544463</td>
<td>0.00000452%</td>
</tr>
<tr>
<td>0.1632455537</td>
<td>0.0367544467</td>
<td>0.00000102%</td>
</tr>
<tr>
<td>0.1632455533</td>
<td>0.0367544468</td>
<td>0.00000023%</td>
</tr>
<tr>
<td>0.1632455532</td>
<td>0.0367544468</td>
<td>0.00000005%</td>
</tr>
</tbody>
</table>
Load - Deflection

Geometric nonlinearity

Nonlinear FEM
Computational Mechanics, AAU, Esbjerg
Direct Substitution Alternative

1. Let load $P_A$ be applied to a softening spring.
2. Assume $k_N = 0$ for the first iteration.
3. Compute first approximation to displacement:
   $$u_1 = \frac{P_A}{k_0}$$
4. Take nonlinear term to other RHS.
5. Compute next approximation to displacement:
   $$u_2 = \frac{(P_A - k_N u_1)}{k_0}$$
6. Generate sequence of approximations.
Sequence of Operations

\[ u_1 = k_0^{-1} P_A \]
\[ u_2 = k_0^{-1} (P_A - k_{N1} u_1) \]
\[ u_3 = k_0^{-1} (P_A - k_{N2} u_2) \]
\[ \vdots \]
\[ u_{i+1} = k_0^{-1} (P_A - k_{Ni} u_i) \]
Geometric nonlinearity

\[ P = k_0 a \]

\[ P_A = P_1 \]

\[ u_1 \]
Geometric nonlinearity

Computational Mechanics, AAU, Esbjerg
Example

\[ k = 0.2 - u \]

\[ P = 0.006 \]

\[ u_1 \]

<table>
<thead>
<tr>
<th>i</th>
<th>u</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>1</td>
<td>0.0300000000</td>
</tr>
<tr>
<td>2</td>
<td>0.0345000000</td>
</tr>
<tr>
<td>3</td>
<td>0.0359512500</td>
</tr>
<tr>
<td>4</td>
<td>0.0364624619</td>
</tr>
<tr>
<td>5</td>
<td>0.0366475556</td>
</tr>
<tr>
<td>6</td>
<td>0.0367152167</td>
</tr>
<tr>
<td>7</td>
<td>0.0367400357</td>
</tr>
<tr>
<td>8</td>
<td>0.0367491511</td>
</tr>
<tr>
<td>9</td>
<td>0.0367525005</td>
</tr>
<tr>
<td>10</td>
<td>0.0367537315</td>
</tr>
</tbody>
</table>
Comparison

1. First approach requires $[K]$ to be formulated and reduced in each step.

2. Second approach requires 1 formulation and reduction of $[K_0]$

3. Second approach usually has more iterative cycles than first approach.
<table>
<thead>
<tr>
<th>i</th>
<th>u</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>1</td>
<td>0.0300000000</td>
</tr>
<tr>
<td>2</td>
<td>0.0352941176</td>
</tr>
<tr>
<td>3</td>
<td>0.0364285714</td>
</tr>
<tr>
<td>4</td>
<td>0.0366812227</td>
</tr>
<tr>
<td>5</td>
<td>0.0367379679</td>
</tr>
<tr>
<td>6</td>
<td>0.0367507370</td>
</tr>
<tr>
<td>7</td>
<td>0.0367536116</td>
</tr>
<tr>
<td>8</td>
<td>0.0367542587</td>
</tr>
<tr>
<td>9</td>
<td>0.0367544045</td>
</tr>
<tr>
<td>10</td>
<td>0.0367544373</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>i</th>
<th>u</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>1</td>
<td>0.0300000000</td>
</tr>
<tr>
<td>2</td>
<td>0.0345000000</td>
</tr>
<tr>
<td>3</td>
<td>0.0359512500</td>
</tr>
<tr>
<td>4</td>
<td>0.0364624619</td>
</tr>
<tr>
<td>5</td>
<td>0.0366475556</td>
</tr>
<tr>
<td>6</td>
<td>0.0367152167</td>
</tr>
<tr>
<td>7</td>
<td>0.0367400357</td>
</tr>
<tr>
<td>8</td>
<td>0.0367491511</td>
</tr>
<tr>
<td>9</td>
<td>0.0367525005</td>
</tr>
<tr>
<td>10</td>
<td>0.0367537315</td>
</tr>
</tbody>
</table>
Under-Relaxation

\[
\begin{align*}
\mathbf{u}_{i+1} &= \mathbf{u}_i + \beta (\Delta \mathbf{u}_i) \\
\Delta \mathbf{u}_i &= \mathbf{u}_{i+1} - \mathbf{u}_i \\
\mathbf{u}_{i+1} &= \beta \mathbf{u}_{i+1} + (1 - \beta) \mathbf{u}_i \\
0 &< \beta < 1
\end{align*}
\]
Newton-Raphson Approach

\[
\left(k_0 + k_{NA}\right)u_A = P_A \\
k_{NA} = f(u_A) \\
One Term Taylor Series: \\
f(u_A + \Delta u_1) = f(u_A) + \left(\frac{dP}{du}\right)_A \Delta u_1
\]
Newton-Raphson Approach

\[ f(u_A + \Delta u_1) = f(u_A) + \left( \frac{dP}{du} \right)_A \Delta u_1 \]

\[
\frac{dP}{du} = \frac{d}{du} \left( k_0 u + k_N u \right) = k_0 + \frac{d(k_N u)}{du}
\]

**Tangent Stiffness**

\[
\frac{dP}{du} = k_t
\]

\[ k_t - \text{Tangent stiffness} \]
Newton-Raphson Approach

Seek:

\[ \Delta u_1 \] such that:

\[ f(u_A + \Delta u_1) = P_B \]

\[ P_B = P_A + (k_{tA})\Delta u_1 \]

\[ (k_{tA})\Delta u_1 = P_B - P_A \]

\[ P_B - P_B \] - Load imbalance
Newton-Raphson Approach

\[
\left( k_t \right)_i \Delta u_i = P_B - P_i
\]

\[
u_{i+1} = u_i + \Delta u_i
\]

\( P_B - P_i \) - Load imbalance
Geometric nonlinearity

Computational Mechanics, AAU, Esbjerg
Newton Raphson

<table>
<thead>
<tr>
<th>$u_A$</th>
<th>$u_B$</th>
<th>$k_{NA}$</th>
<th>$k_t$</th>
<th>$P_A$</th>
<th>$P_B$</th>
<th>$\Delta L u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0000000000</td>
<td>0.0300000000</td>
<td>0.000000</td>
<td>0.200000</td>
<td>0.000000</td>
<td>0.0060</td>
<td>0.0300000000</td>
</tr>
<tr>
<td>0.0300000000</td>
<td>0.0364285714</td>
<td>-0.030000</td>
<td>0.140000</td>
<td>0.005100</td>
<td>0.0060</td>
<td>0.0064285714</td>
</tr>
<tr>
<td>0.0364285714</td>
<td>0.0367536116</td>
<td>-0.036429</td>
<td>0.127143</td>
<td>0.005959</td>
<td>0.0060</td>
<td>0.0003250401</td>
</tr>
<tr>
<td>0.0367536116</td>
<td>0.0367544468</td>
<td>-0.036754</td>
<td>0.126493</td>
<td>0.006000</td>
<td>0.0060</td>
<td>0.0000008352</td>
</tr>
<tr>
<td>0.0367544468</td>
<td>0.0367544468</td>
<td>-0.036754</td>
<td>0.126491</td>
<td>0.006000</td>
<td>0.0060</td>
<td>0.0000000000</td>
</tr>
</tbody>
</table>
Modified Newton-Raphson Approach

Do not update $k_t$ every iteration!
Newton-Raphson Approach

\[
\left( k_{t(\text{old})} \right) \Delta u_i = P_B - P_i
\]

\[
u_{i+1} = u_i + \Delta u_i
\]

\[P_B - P_B \quad \text{- Load imbalance}\]
Geometric nonlinearity

NR

Nonlinear FEM
Computational Mechanics, AAU, Esbjerg
Modified N-R

![Diagram showing Modified N-R process with points a and b, and vectors Δu₁ and Δu₂.](image)
Comparison

1. Modified N-R has less calculations per iteration.
2. Modified N-R has more iterations.
Incremental Approach

1. Apply loads in a number of small increments.
2. Iterate and Converge for each increment.
3. Create entire load-displacement history.
Purely incremental approach with no corrections.
Euler’s Method

\[ P = f(u) \]

\[ k_t = \frac{dP}{du} \]

Load increments: \( \Delta P \)
Euler’s Method

Start at $P = 0$ and $u = 0$

Euler's Method

\[ u_1 = 0 + \left(k_t\right)_0^{-1} \Delta P_1 \quad \text{where} \quad \left(k_t\right)_0 = k_t \text{ at } u = 0 \]
\[ u_2 = u_1 + \left(k_t\right)_1^{-1} \Delta P_2 \quad \text{where} \quad \left(k_t\right)_1 = k_t \text{ at } u = u_1 \]
\[ u_3 = u_2 + \left(k_t\right)_2^{-1} \Delta P_3 \quad \text{where} \quad \left(k_t\right)_2 = k_t \text{ at } u = u_2 \]
\[ \vdots \]
\[ u_{i+1} = u_i + \left(k_t\right)_i^{-1} \Delta P_{i+1} \quad \text{where} \quad \left(k_t\right)_i = k_t \text{ at } u = u_{i-1} \]
Incremental with Load Correction

\[ u_{i+1} = u_i + (k_t)_i^{-1}[\Delta P_{i+1} + (P_i - P_{iR})] \]

\[ P_i = \sum \Delta P_1 \quad \text{Externally Applied Load} \]

\[ P_{iR} = \left(k_0 + k_{Ni}\right)u_i \quad \text{Resisting Load of the Spring} \]
Incremental approach with Load Corrections.