SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP

Linear vs. nonlinear problems - Part I

Miroslav Halilovič, Bojan Starman, Janez Urevc, Nikolaj Mole

Faculty of Mechanical Engineering, University of Ljubljana

June/2021

Univerza *v Ljubljani*

TECHNISCHE UNIVERSITÄT WIEN

CINECA consorzio interuniversitario

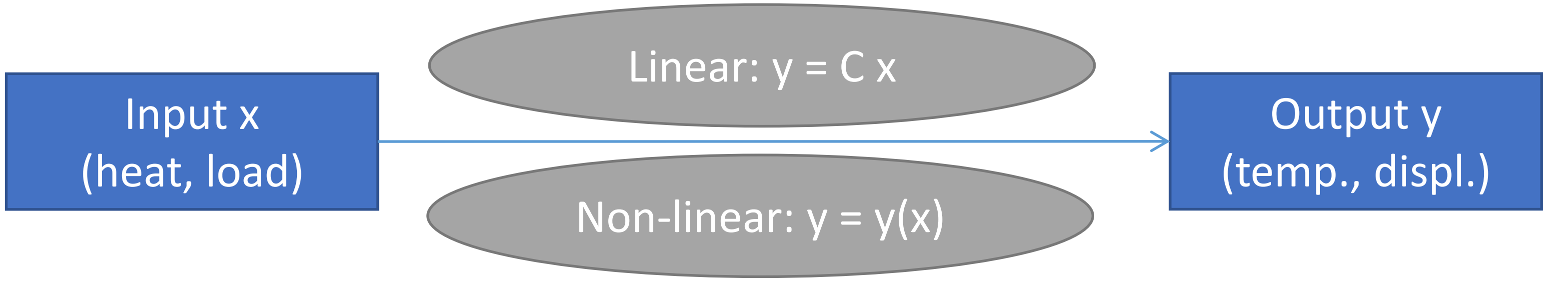
VSB TECHNICAL UNIVERSITY OF OSTRAVA | IT4INNOVATIONS NATIONAL SUPERCOMPUTING CENTER

Co-funded by the Erasmus+ Programme of the European Union

This project has been funded with support from the European Commission.
This publication [communication] reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Linear vs. nonlinear

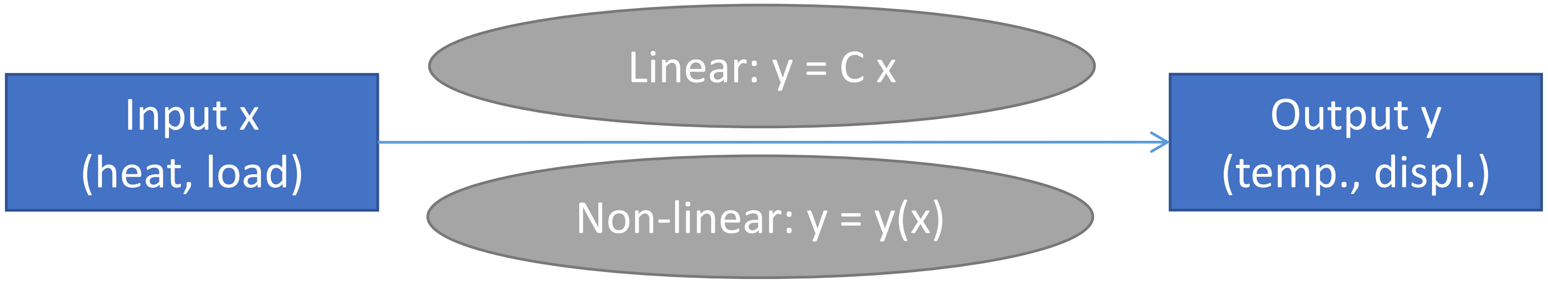
SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP



$2x = 6$ $x^2 - 3x + 2 = 0$ $\ln(2\sin(x) + x^3) + 3x\cos(x) = 6$

- Linear: unique solution
- Nonlinear: multiple solutions (or no solution!)

Linear vs. nonlinear

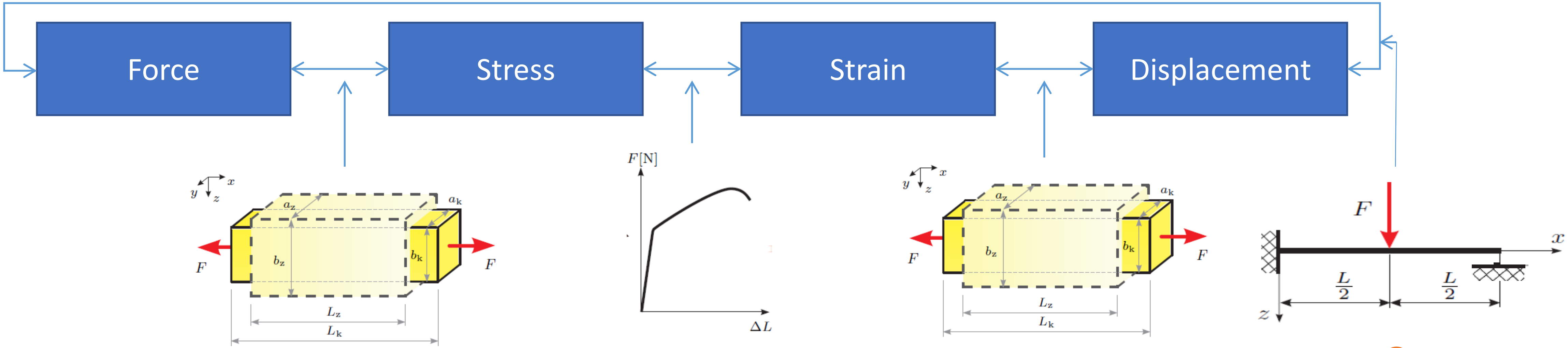


FEM: $$K\,U = F \rightarrow \begin{bmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{bmatrix} \begin{Bmatrix} \vdots \end{Bmatrix} = \begin{Bmatrix} \vdots \end{Bmatrix}$$

- Linear: *K* = const.
- Nonlinear: *K* = *K*(*U*), *F* = *F*(*U*) -> linearize -> iteratively solving linear systems

Types of nonlinearities

SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP



$F = A\sigma$ $\quad$ $\sigma = E\varepsilon$ $\quad$ $\Delta L = L\varepsilon$ $\quad$ $u = cF$

Constants. Really?

Types of nonlinearities

SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP

Types of non-linearities

- Geometric nonlinearity
- Material nonlinearity
- Kinematic (constrain, contact) nonlinearity

Assumptions for linearity:

- Small strains
- Small displacements
- Small rotations
- Linear stress-strain relations
- Constant applied loads
- No contact conditions

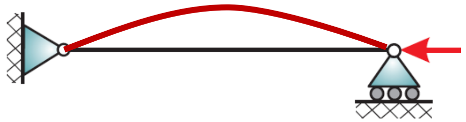
Types of nonlinearities

How small? Rules of thumb:

- Strains < 5 %
- Displacements < half of the beam/plate thickness
- Small rotations < 5°

Magnitude of strains, displacements and rotations is not the only criterion:

- Buckling
- Membrane state during bending



[1]

[1] https://www.amazon.com/Two-Way-Foldable-Rebounder-Trampoline-included/dp/B00GZ0IVEG

Types of nonlinearities

Why not always performing analyses that include all kind of nonlinearities?

Advantages of linear analyses:

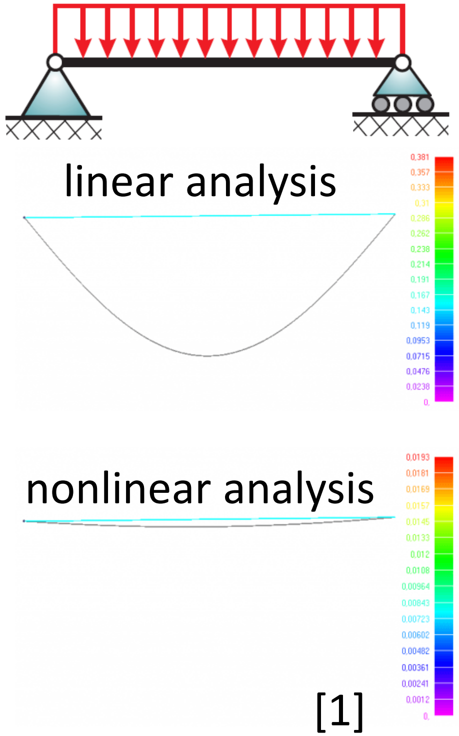
- analytical solutions are available for many linear problems (model validation)
- they facilitates better conceptual understanding of the problem
- procedures are prescribed by standards
- they are computationally fast and robust

Take advantage of computational simplicity of linear analyses when possible...

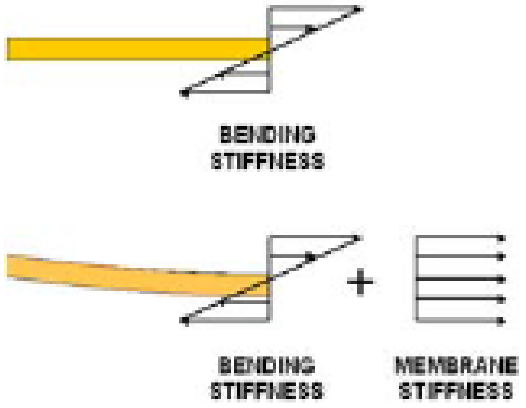
... but do not analyse nonlinear problems with linear analyses!

Geometric nonlinearity

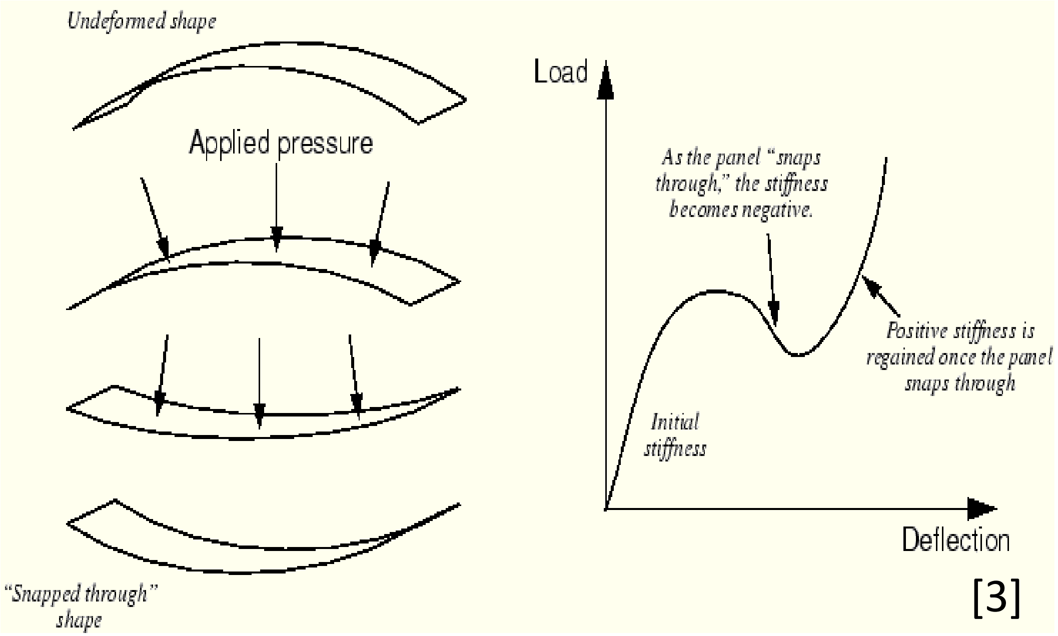
SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP



[1]



[2]



[3]

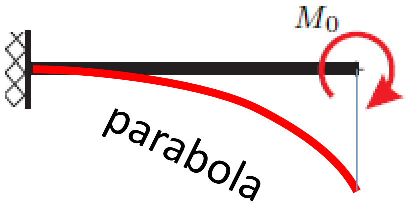
[1] https://enterfea.com/nonlinear-fea-introduction/

[2] https://caendkoelsch.wordpress.com/2019/03/14/geometric-nonlinearity-what-does-it-mean-post-1-2/

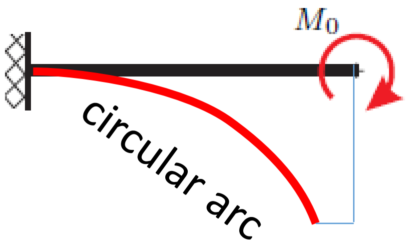
[3] Abaqus manual, Simulia

Geometric nonlinearity

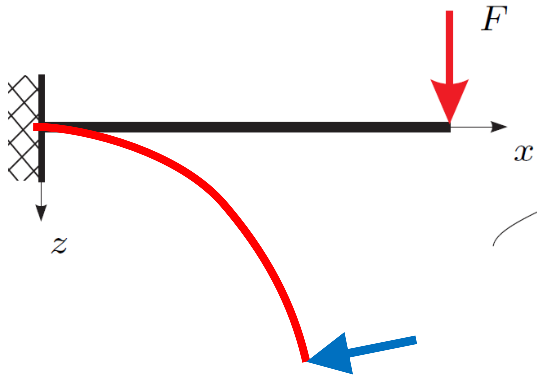
linear analysis



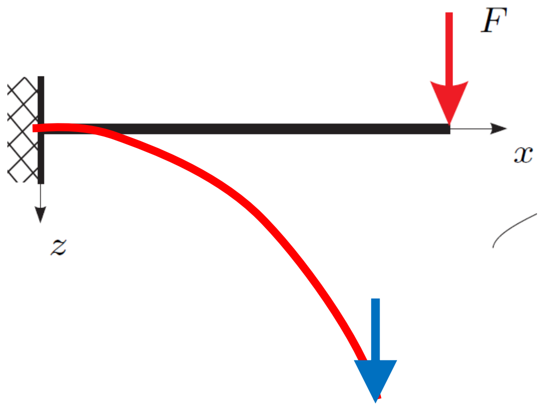
nonlinear analysis



nonlinear analysis

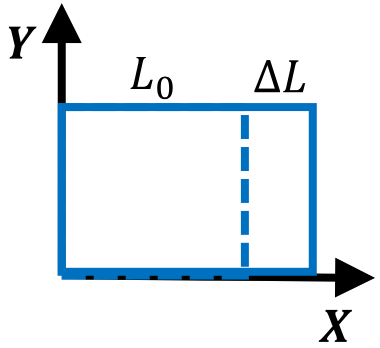


follower force

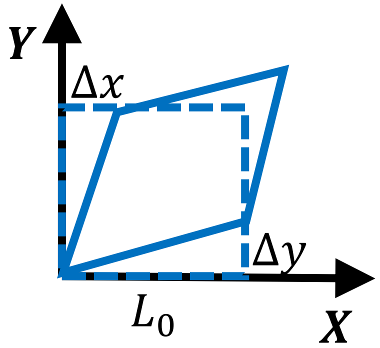


non-follower force

Geometric nonlinearity



$$\varepsilon = \frac{\Delta L}{L_0} \rightarrow \varepsilon_{xx} = \frac{\partial u_x}{\partial x}$$



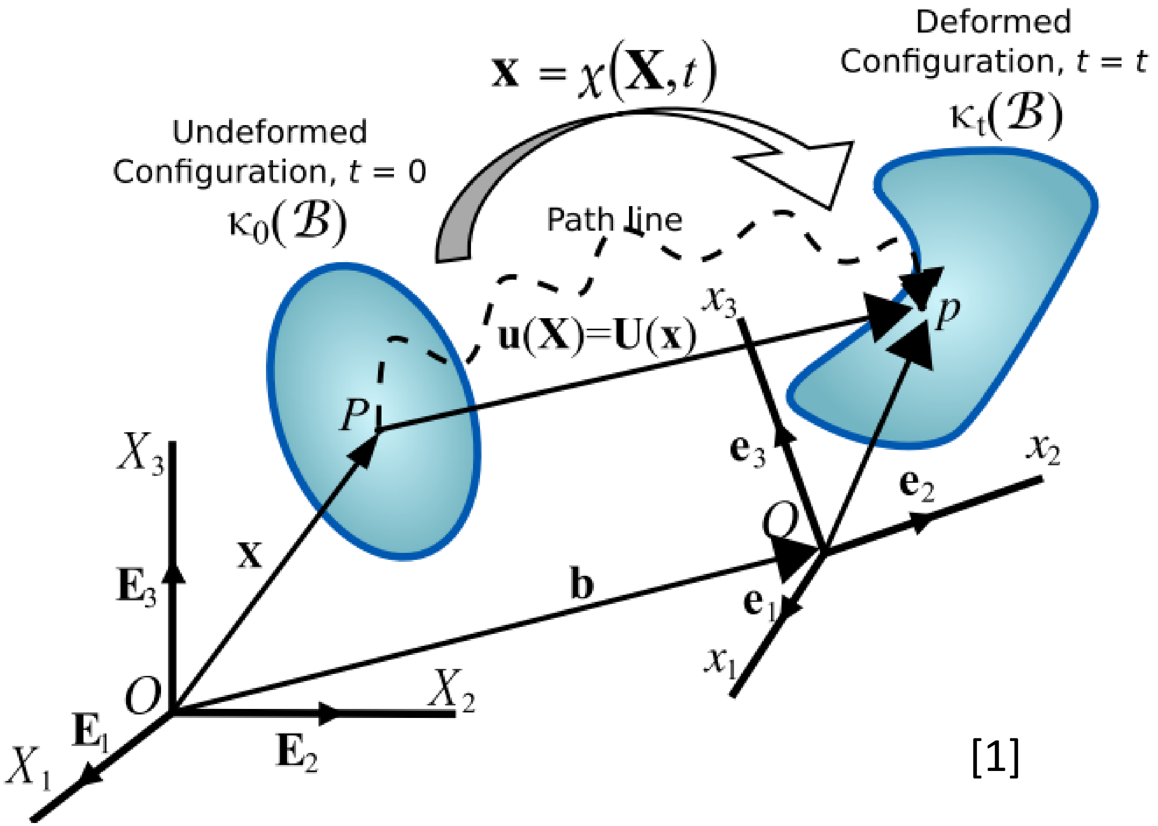
$$\gamma = \frac{\Delta x + \Delta y}{L_0} \rightarrow \gamma_{xy} = \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}$$

$$\varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right)$$

small (infinitesimal) strain

Geometric nonlinearity

SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP



[1]

displacement ≠ deformation

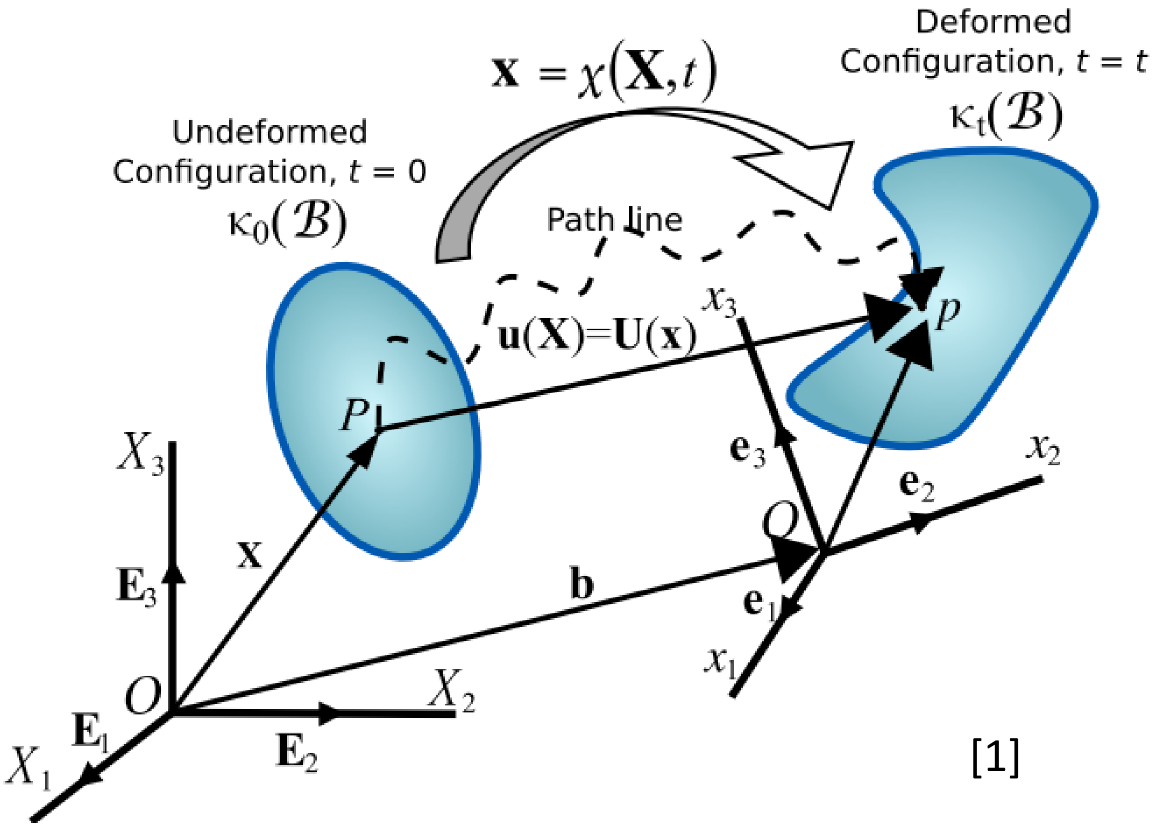
small (infinitesimal) strain:

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

which frame?

[1] https://en.wikipedia.org/wiki/Finite_strain_theory

Geometric nonlinearity



[1]

Deformation gradient

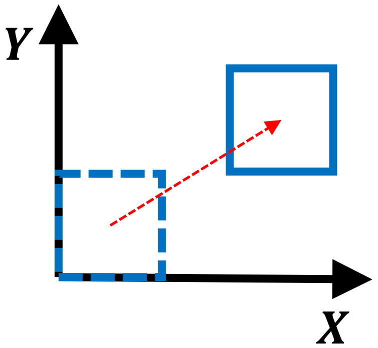
$$F_{ij} = x_{i,j} = \frac{\partial x_i}{\partial X_j}$$

$$u_i = x_i - X_i \;\rightarrow x_i = X_i - u_i$$

$$F_{ij} = \delta_{i,j} + u_{i,j} \qquad \ldots \qquad \mathbf{F} = \mathbf{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}}$$

[1] https://en.wikipedia.org/wiki/Finite_strain_theory

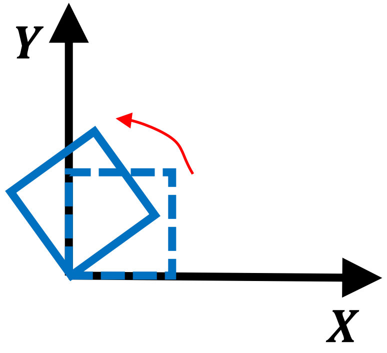
Geometric nonlinearity



$$\mathbf{F} = \mathbf{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}}$$

$$\mathbf{F} = \mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

... rigid body displacement (translation)



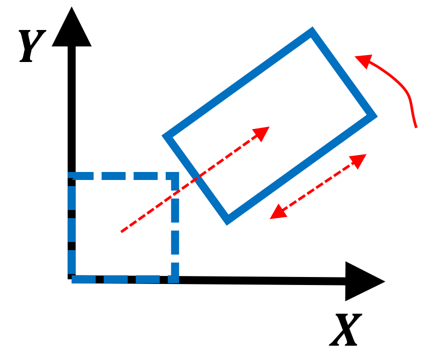
$$\mathbf{F} = \mathbf{R} = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix}$$

... rigid body rotation

$$\varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right) \quad \rightarrow \quad \boldsymbol{\varepsilon} = \frac{1}{2}(\mathbf{F} + \mathbf{F}^T) - \mathbf{I} = \begin{bmatrix} \cos(\varphi) - 1 & 0 \\ 0 & \cos(\varphi) - 1 \end{bmatrix}$$

$$\varphi = 90° \rightarrow \varepsilon_{xx} = \varepsilon_{yy} = -1$$

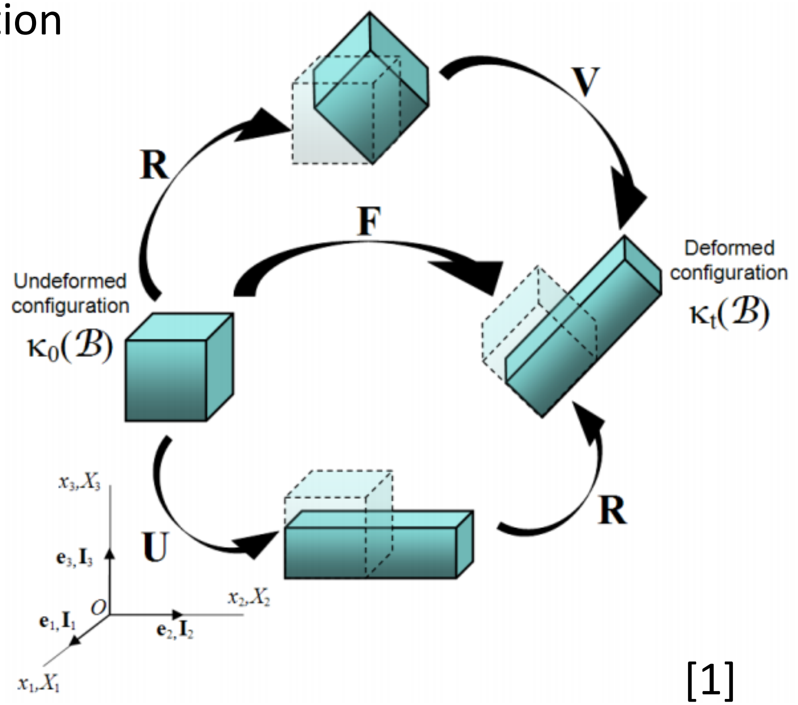
Geometric nonlinearity



$\mathbf{F} = \mathbf{R}\,\mathbf{U} = \mathbf{V}\,\mathbf{R}$ ... polar decomposition

$\mathbf{U}$ ... right stretch tensor

$\mathbf{V}$ ... left stretch tensor



$$\mathbf{F}^{\mathbf{T}}\,\mathbf{F} = (\mathbf{R}\,\mathbf{U})^{\mathbf{T}}\mathbf{R}\,\mathbf{U} = \mathbf{U}^{\mathbf{T}}\,\underbrace{\mathbf{R}^{\mathbf{T}}\,\mathbf{R}}_{\mathbf{I}}\,\mathbf{U} = \mathbf{U}^{\mathbf{T}}\,\mathbf{U} = \mathbf{C}$$

... right Cauchy–Green deformation tensor

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}\,\mathbf{F}^{T} - \mathbf{I})$$

... Green-Lagrange strain tensor

[1]

[1] https://en.wikipedia.org/wiki/Finite_strain_theory

Geometric nonlinearity

SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP

Green-Lagrange strain

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}\,\mathbf{F}^T - \mathbf{I})$$

$$E_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i} + u_{k,i}\,u_{k,j})$$

small strain

$$\boldsymbol{\varepsilon} = \frac{1}{2}(\mathbf{F} + \mathbf{F}^T) - \mathbf{I}$$

$$\varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i})$$

$\mathbf{C} = \mathbf{F}^\mathbf{T}\,\mathbf{F}$ ... right Cauchy–Green deformation tensor

$\mathbf{B} = \mathbf{F}\,\mathbf{F}^\mathbf{T}$ ... left Cauchy–Green deformation tensor

$\mathbf{f} = \mathbf{F}^{-1}\,\mathbf{F}^{-\mathbf{T}}$ ... Finger deformation tensor

$\mathbf{c} = \mathbf{F}^{-\mathbf{T}}\,\mathbf{F}^{-1}$ ... Cauchy deformation tensor

$\mathbf{E} = \frac{1}{2}(\mathbf{C} - \mathbf{I})$ ... Green-Lagrange strain tensor

$\mathbf{e} = \frac{1}{2}(\mathbf{I} - \mathbf{c})$ ... Almansi strain tensor

$\boldsymbol{E_{Biot}} = \mathbf{U} - \mathbf{I}$ ... Biot strain tensor

$\boldsymbol{\varepsilon_{true}} = \ln(\mathbf{U})$ ... logarithmic (natural, true, Hencky) strain tensor

Material nonlinearity

SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP

Work conjugates

$$w = \int_{\Omega_0} \mathbf{\Pi}\,\dot{\mathbf{F}}\,d\Omega_0 = \int_{\Omega_0} \mathbf{S}\,\dot{\mathbf{E}}\,d\Omega_0 = \int_{\Omega} \boldsymbol{\sigma}\,\mathbf{D}\,d\Omega = \int_{\Omega_0} \boldsymbol{\tau}\,\mathbf{D}\,d\Omega_0$$

$\mathbf{\Pi}$ ... First Piola Kirchhoff stress tensor

$\mathbf{S}$ ... Second Piola Kirchhoff stress tensor

$\boldsymbol{\sigma}$ ... Cauchy (true) stress tensor

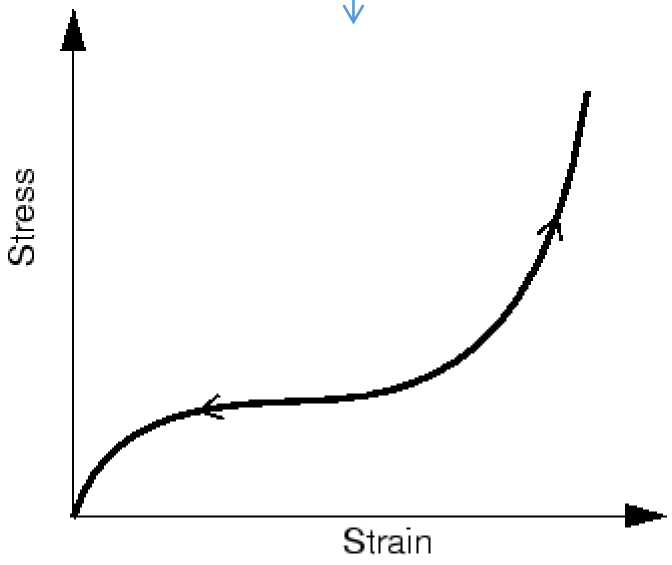
$\boldsymbol{\tau}$ ... Kirchhoff stress tensor

$\mathbf{D} = \frac{1}{2}\left(\dot{\mathbf{F}}\,\mathbf{F}^{-1} + \mathbf{F}^{-\mathrm{T}}\dot{\mathbf{F}}\right) = \dot{\boldsymbol{\varepsilon}}_{\boldsymbol{true}}$ ... rate of deformation tensor

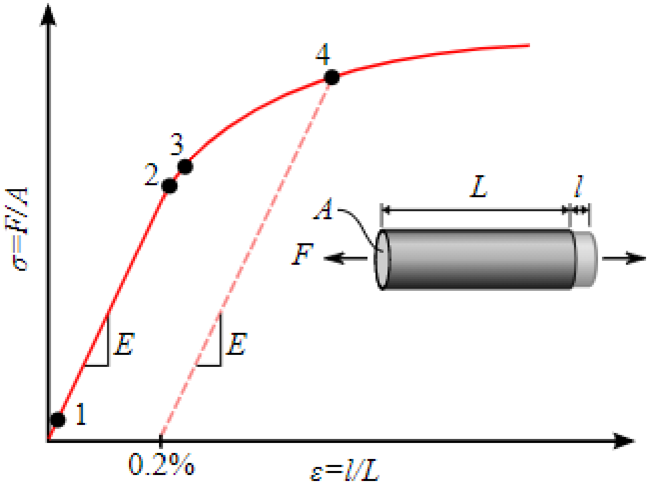
Material nonlinearity

Constitutive law: Stress ⟷ (linear or nonlinear) ⟷ Strain

- Linear: *Linear elasticity*
- Nonlinear: *Nonlinear elasticity (hyperelasticity), elastoplasticity*



[1]



[2]

[1] Abaqus manual, Simulia
[2] https://en.wikipedia.org/wiki/Plasticity_(physics)

Material nonlinearity

SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP

Linear elasticity, isotropic material (Hooke's model):

$$\sigma_{ij} = \frac{E}{(1+\upsilon)}\left(\varepsilon_{ij} + \frac{\upsilon}{(1-2\upsilon)}\varepsilon_{kk}\delta_{ij}\right)$$

Elastoplasticity and damage (Gurson-Tvergaard-Needleman's model):

$$\Phi = \frac{(\sigma_{eq})^2}{(\sigma_y)^2} + 2\, f\, q_1\, Cosh\left(\frac{3 q_2\, \sigma_H}{2\sigma_y}\right) - (1 + q_3\, f^2)$$

$$\sigma_{eq} = a\Big(F_1(\sigma_{22}-\sigma_{33})^2 + G_1(\sigma_{33}-\sigma_{11})^2 + H_1(\sigma_{11}-\sigma_{22})^2 + 2\big(L_1\sigma_{12}^2 + M_1\sigma_{13}^2 + N_1\sigma_{23}^2\big)\Big)^{\frac{1}{2}} +$$

$$\sqrt{2}(1-a)\Big(F_2^2(\sigma_{22}-\sigma_{33})^4 + G_2^2(\sigma_{33}-\sigma_{11})^4 + H_2^2(\sigma_{11}-\sigma_{22})^4 + 2\big(L_2^2\sigma_{12}^4 + M_2^2\sigma_{13}^4 + N_2^2\sigma_{23}^4\big)\Big)^{\frac{1}{4}}$$

$$\mathrm{d}f = (1-f)\,\mathrm{d}\varepsilon_{kk}^p + A_n\,\mathrm{d}\bar{\varepsilon}_m^p \;;\; \mathrm{d}\varepsilon_{ij}^p = \frac{\partial\Phi}{\partial\sigma_{ij}}\mathrm{d}\lambda$$

$$\mathrm{d}\sigma_{ij} = C_{ijkl}\left(\mathrm{d}\varepsilon_{kl} - \frac{\partial\Phi}{\partial\sigma_{kl}}\mathrm{d}\lambda\right) \;;\; \mathrm{d}\bar{\varepsilon}_m^p = \frac{\sigma_{ij}\,\mathrm{d}\varepsilon_{ij}^p}{(1-f)\,\sigma_y}$$

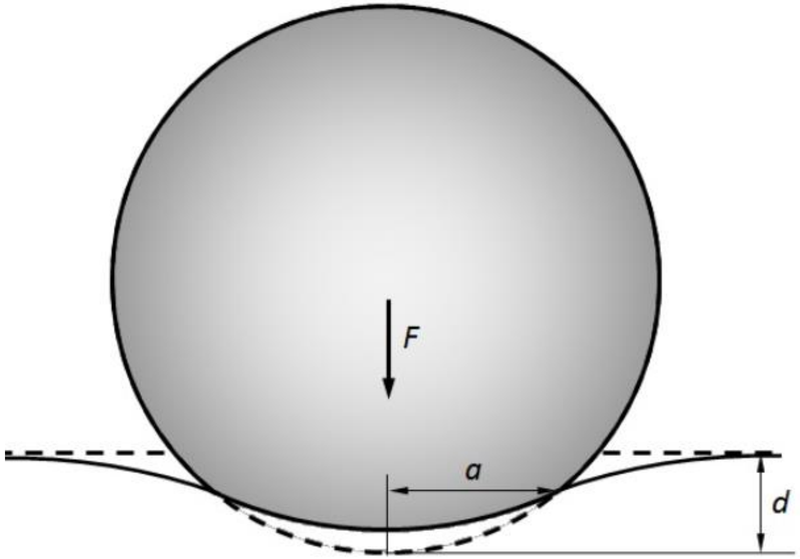
$$\sigma_{ij} = C_{ijkl}\varepsilon_{kl}^e \quad ; \quad C_{ijkl} = C_{ijkl}\left(\mathrm{E}_{11},\mathrm{E}_{22},\mathrm{E}_{33},\mathrm{G}_{12},\mathrm{G}_{13},\mathrm{G}_{23},\nu_{12},\nu_{13},\nu_{23}\right)$$

$$\mathrm{d}\bar{\varepsilon}_m^m = \sigma_{ij}P_{ijkl}\,\mathrm{d}\varepsilon_{kl}^p / (\sigma_y(1-f))$$

Contact nonlinearity

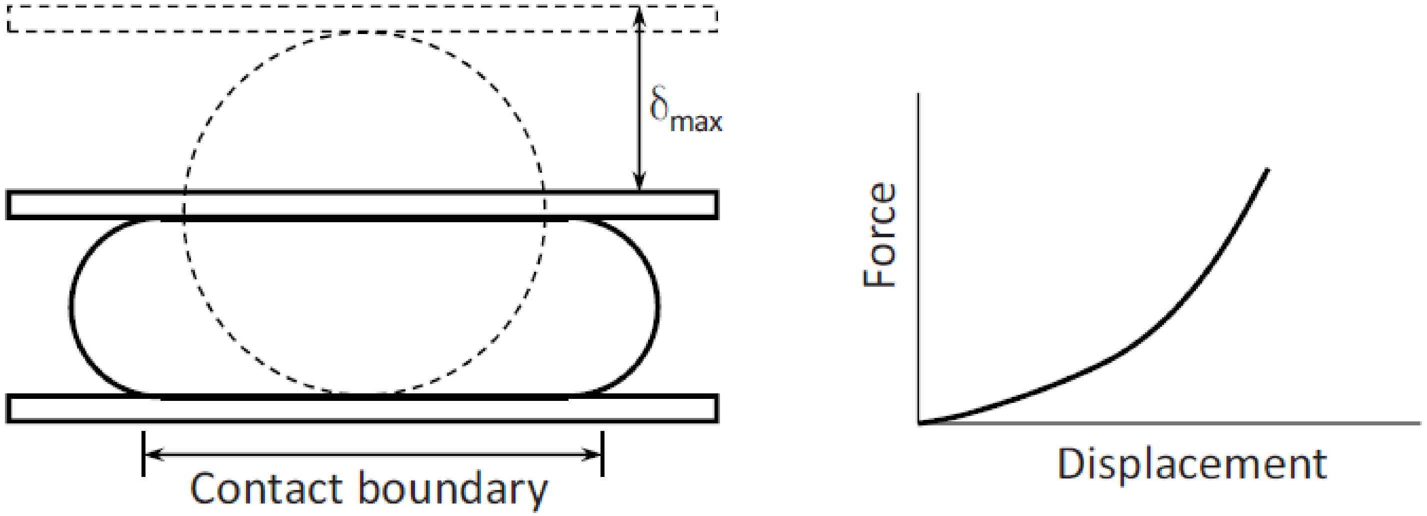
SCtrain | SUPERCOMPUTING KNOWLEDGE PARTNERSHIP

Hertz contact theory $F \propto \sqrt{d^3}$



[1]

Displacement dependent boundary conditions

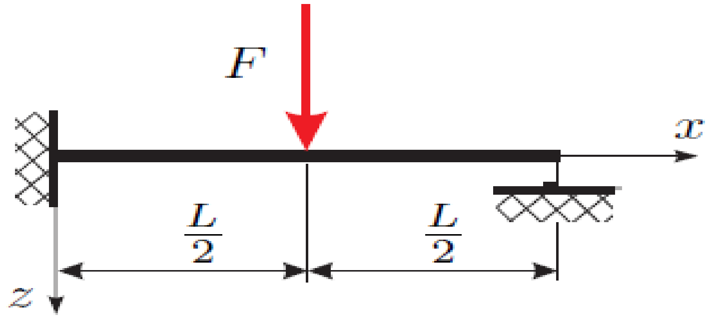


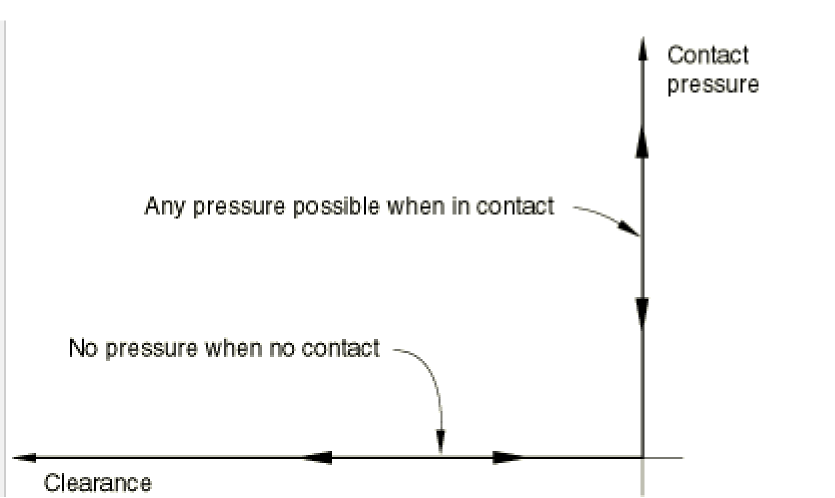
[2]

[1] https://en.wikipedia.org/wiki/Contact_mechanics
[2] https://mae.ufl.edu/nkim/egm6352/Chap2.pdf

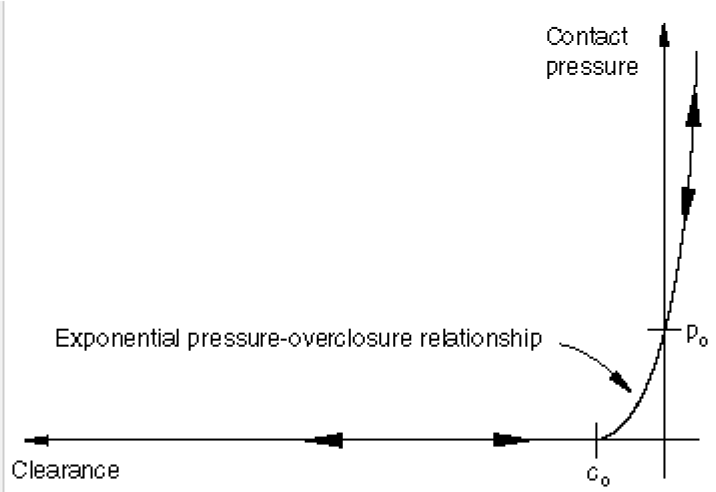
Contact nonlinearity

Establishing a contact





hard contact



soft contact [1]

[1] Abaqus manual, Simulia

Thank you for your attention!

http://sctrain.eu/

Univerza v Ljubljani
TECHNISCHE UNIVERSITÄT WIEN
CINECA consorzio interuniversitario
VSB TECHNICAL UNIVERSITY OF OSTRAVA | IT4INNOVATIONS NATIONAL SUPERCOMPUTING CENTER

Co-funded by the Erasmus+ Programme of the European Union

This project has been funded with support from the European Commission.
This publication [communication] reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.