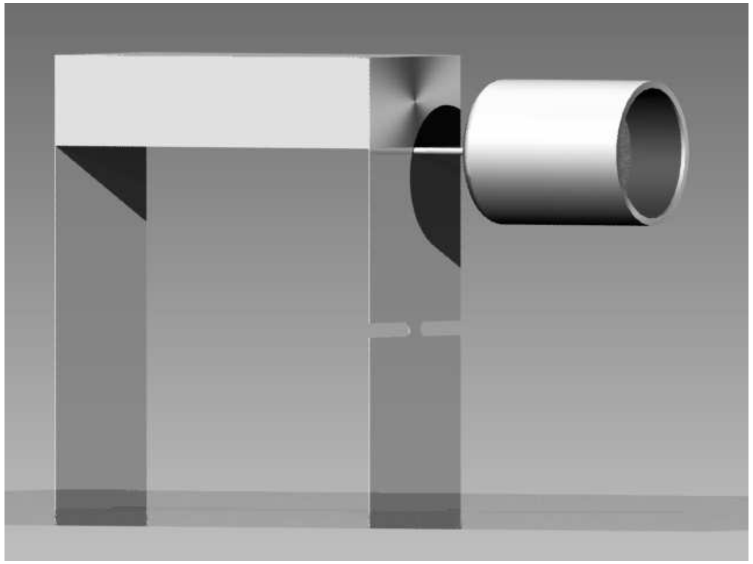
Prototype modelling of mechanic systems

An introduction for Modules 5&8

Ronald Aarts
Mechanical Automation /
Werktuigbouwkundige Automatisering (Wa)
Horstring Z.229

See also BlackBoard!!!

Overview



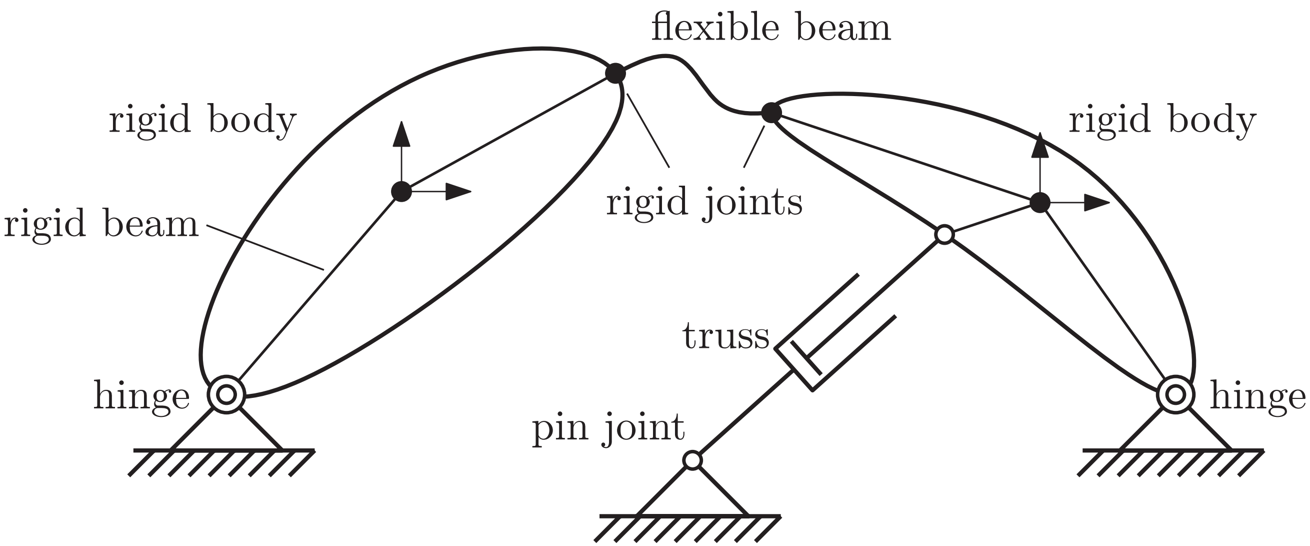
- Introduction prototype modelling.
  - $\rightarrow$ Example system.
  - $\rightarrow$ SPACAR software package:
    - Create and edit models from a GUI
    - Analysis via MATLAB/SIMULINK interface
    - Visualisation
- Simple 1-DOF mass-spring model
  - $\rightarrow$ Non-linear finite elements, nodal coordinates and deformation parameters.
  - $\rightarrow$ Selection of degrees-of-freedom (DOF's).
- Two-dimensional model: 1-DOF and more-DOF.
- Three-dimensional model: 1-DOF and more-DOF.

Introduction

- Modelling and analysis enable designers to test whether design specifications are met
  $\longrightarrow$ with varying level of detail.
- In the early, conceptual stage: high level analysis when only a few design details are known.
- Simple prototype models with a few degrees of freedom:
  - Capture only the relevant systems dynamics
    $\longrightarrow$ offer insight.
  - Quick to evaluate, quick to change
    $\longrightarrow$ immediate feedback on design decisions.
  - Comprehensive exploration of design alternatives
    $\longrightarrow$ well-considered selection of “best” design concept
    $\longrightarrow$ to be analysed in more detail (e.g. with ANSYS).

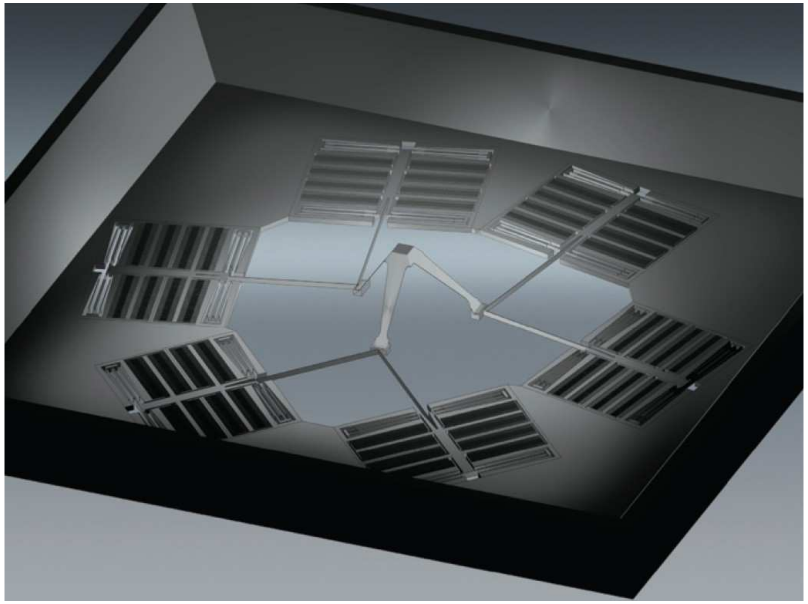
Flexible multibody systems and structures

- Equations of motion expressed in terms of system's degrees of freedom (DOF's) $\rightarrow$ Lagrange equations.



- Stationary and equilibrium solutions (structures).
- Linearised equations of motion $\rightarrow$ State-space equations.

Software package SPACAR



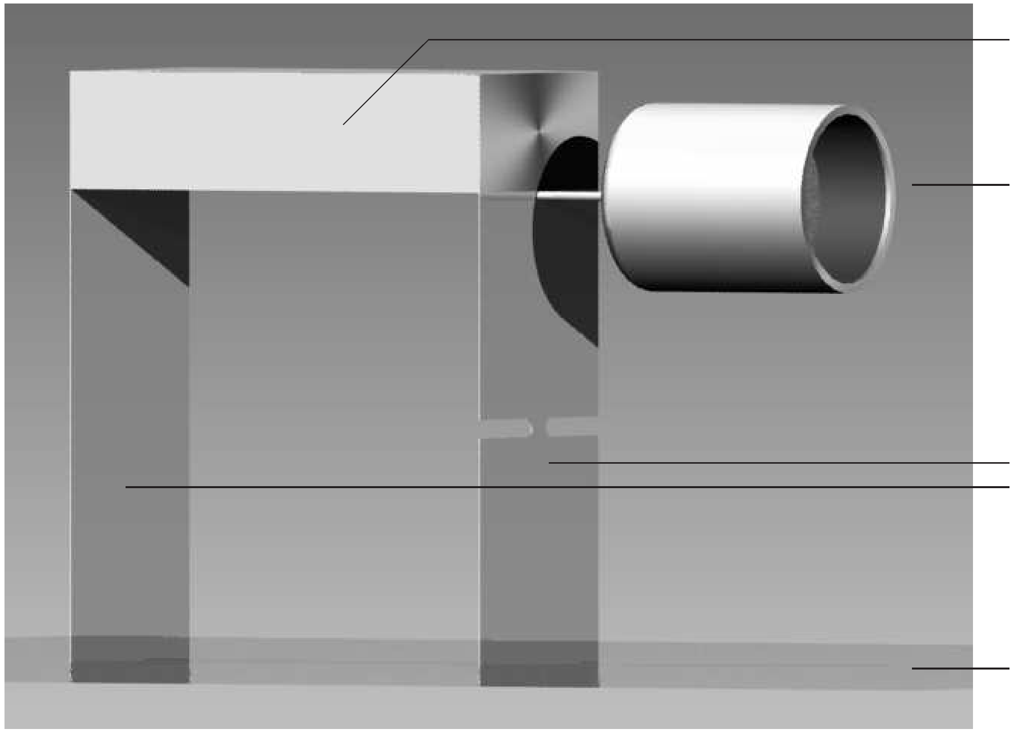
- Kinematic and dynamic analysis of
  - flexible multibody systems,
  - flexible structures.
- Based on the finite element method
- MATLAB user interface for the analysis of (linear) systems including visualisation.
  $\rightarrow$ Available for download and installation, see BlackBoard.
- Stand-alone GUI to build and edit models.
  $\rightarrow$ Educational license available, see BlackBoard.

Mechatronic system design

- Conceptual design
  - Kinematic analysis.
- Dimensioning the concepts
  - Natural frequencies and mode shapes
  - Static stability (buckling)
  - State space input output formulations (SISO or MIMO)
  - Simulation of the dynamic behaviour.
- Computer aided prototyping.
- Final design (fine tuning, e.g. with ANSYS).

Example system

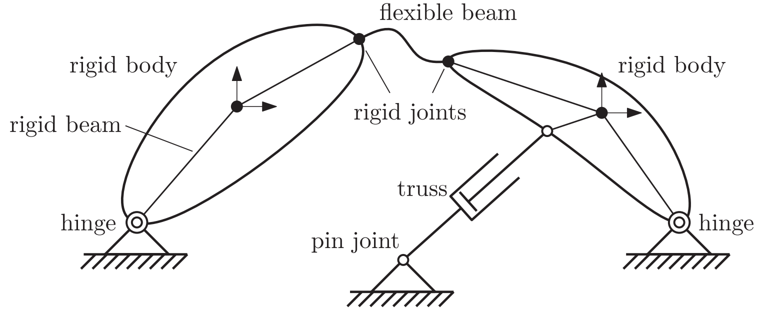
- One degree of freedom (1-DOF) VCM-driven support mechanism with elastic leaf springs. Both springs are fixed at the bottom (clamped support).



- This system will be analysed with an increasing degree of complexity using SPACAR.

Analysis of example system with SPACAR

- Introduction of the finite element concept with *nodal coordinates* and *element deformations*.

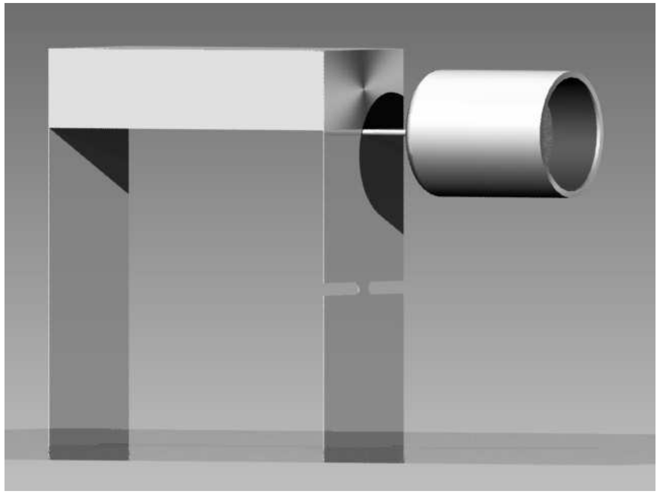


- Two-dimensional (planar) and three-dimensional (spatial) models with (a small number) truss and beam elements.
- Each element has nodal points: The coordinates of translational and rotational nodal points describe the element's position and orientation.
- For each element a fixed number of independent (discrete) deformation modes are defined as functions of the nodal coordinates. Deformation modes are always invariant for rigid body movements of the element.
- Systems are defined in SPACAR input files (e.g. using the GUI) and after the call to `spacar` the results are available in MATLAB variables and stored in output files.

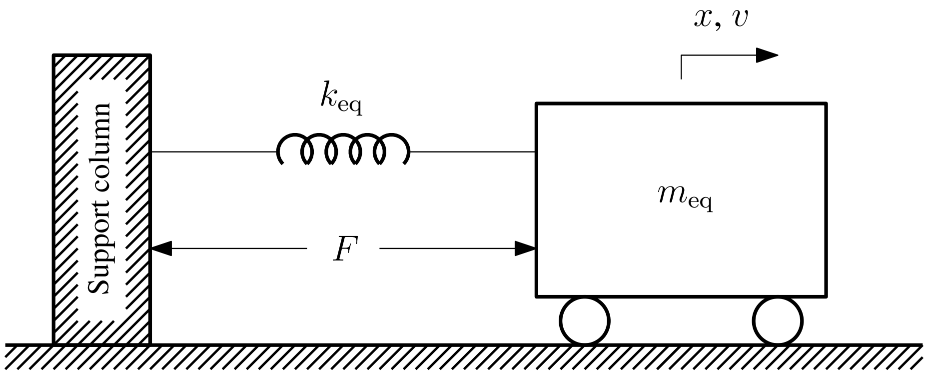
Getting started

- Download and install the software:
- An educational time-limited license is available for the stand-alone GUI, see BlackBoard.
  $\rightarrow$ Any MS Windows version
  $\rightarrow$ Unpack ZIP-file anywhere on your PC's harddisk and run `spacar.exe`
  $\rightarrow$ On-line help.
  $\rightarrow$ Built-in update manager.
- The MATLAB toolbox for the numerical analysis must be downloaded separately.
  $\rightarrow$ Follow the installation instructions.
  $\rightarrow$ Only for 32-bit/64-bit MS Windows MATLAB versions.
  $\rightarrow$ Additional help and demo's available.

Mass-spring model



$\leftrightarrow$

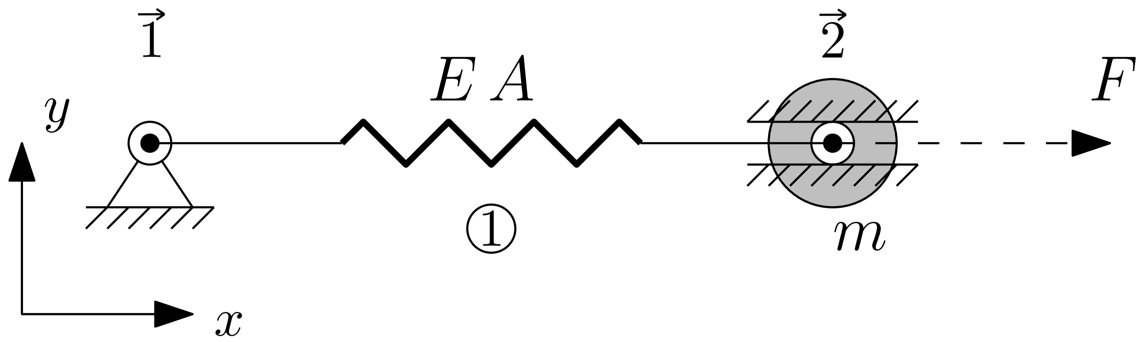


- Model system as a simple one degree-of-freedom mass-spring system.
- All mass is lumped in a single equivalent mass m_{eq} and the equivalent stiffness k_{eq} represents all elastic components.
- Input VCM force F and output position x:

$$G(s) = \frac{x}{F} = \frac{1}{m_{\text{eq}}\, s^2 + k_{\text{eq}}}.$$

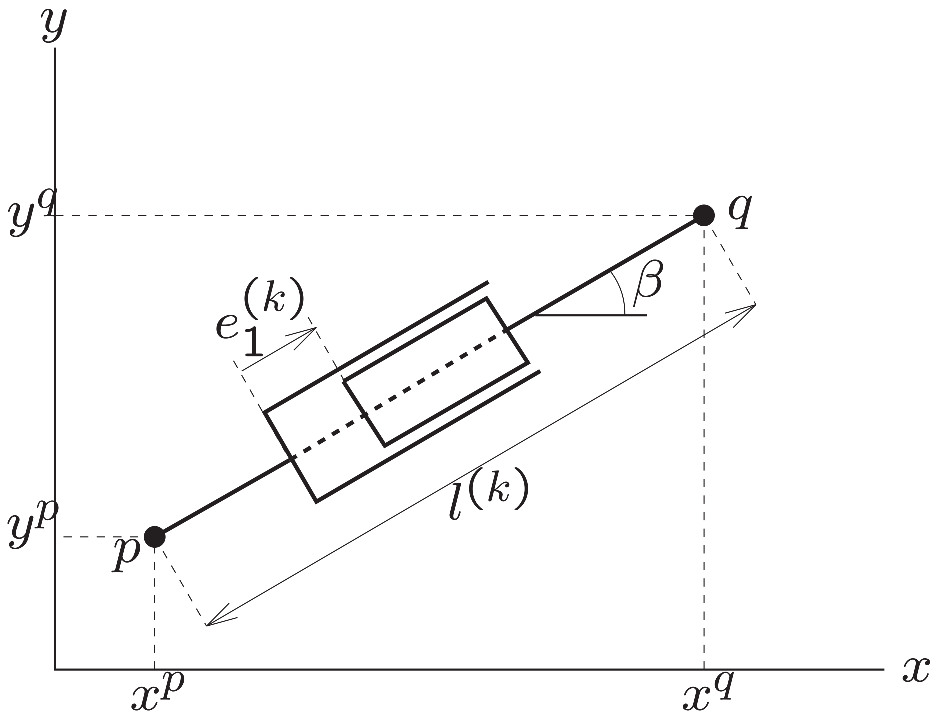
First SPACAR model: x degree-of freedom

- The mass-spring system can be modelled as a one-dimensional SPACAR model: Identify elements with their coordinates and deformations.



- Two translational nodal points: $\vec{1}$ and $\vec{2}$.
- One element for the spring: a (two-dimensional) *truss* element ①.

Planar truss element (PLTRUSS)



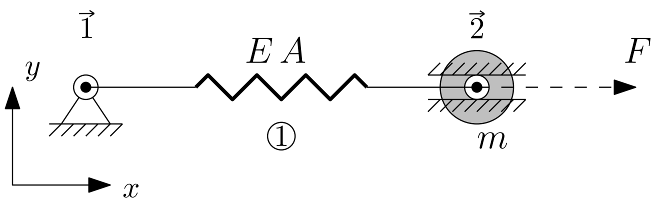
- Two translational nodal points p and q with two coordinates each:

$$\boldsymbol{x}_{\text{truss}}^{(k)} = \begin{bmatrix} \boldsymbol{x}^p \\ \boldsymbol{x}^q \end{bmatrix} = [x^p, y^p, x^q, y^q]^T \,.$$

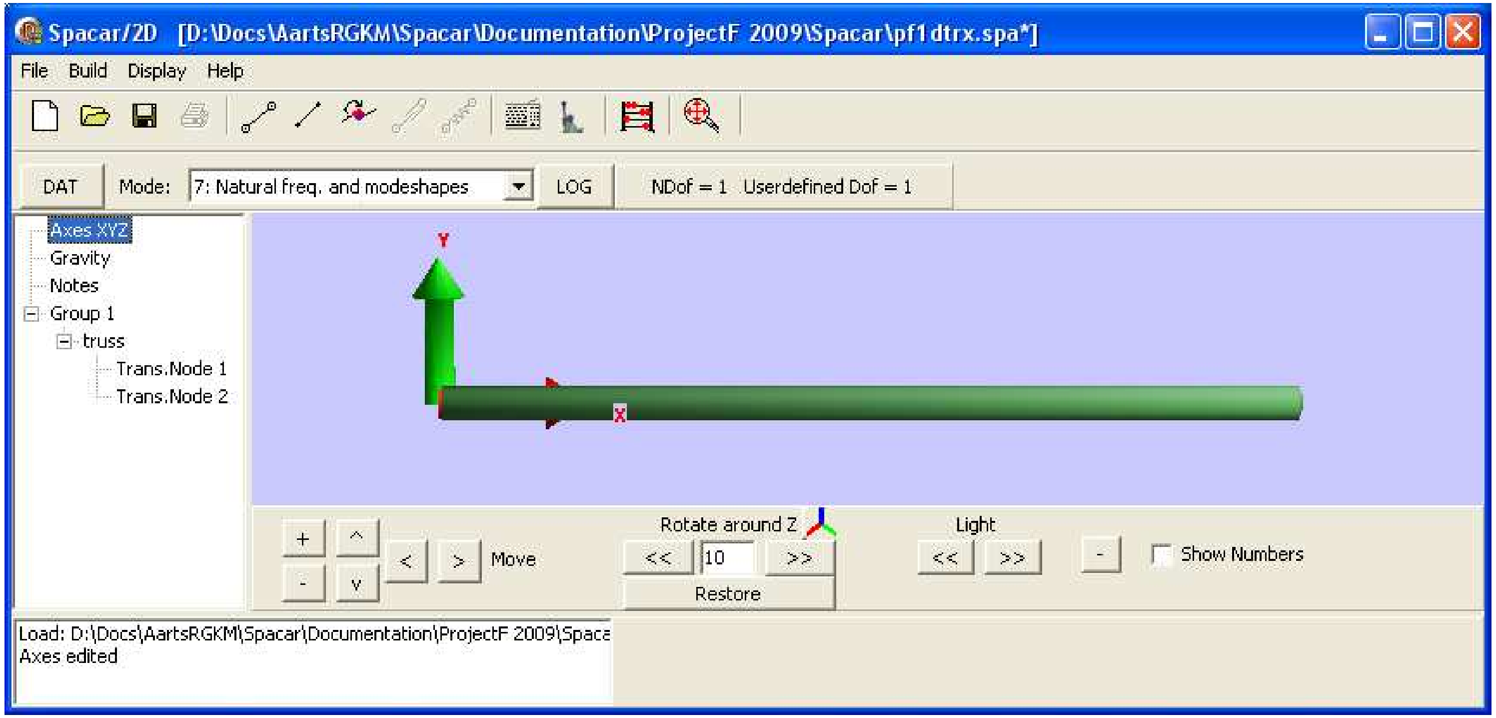
- A single deformation mode e_1 of this element represents the elongation:

$$e_1^{(k)} = l^{(k)} - l_0^{(k)}, \text{ with } l^{(k)} = \sqrt{(x^p - x^q)^2 + (y^p - y^q)^2}.$$

Building model `pf1dtrx.spa` – Creating the truss

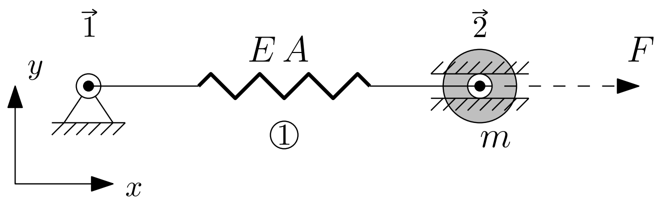


- Create the truss.
- The end nodes are `New`
  $\rightarrow$ specify the (initial) coordinates $(0.0, 0.0)$ and $(0.1, 0.0)$.
- Edit the truss to set its dimensions.



Building model `pf1dtrx.spa`
– Constraint coordinates and released deformations

- Nodal point coordinates are by default “`Dependent`”, i.e. can move freely depending on the rest of the system.
  $\rightarrow$ Set support coordinates to `Fixed`: Both `X` and `Y` in $\vec{\texttt{1}}$ and `Y` in $\vec{\texttt{2}}$.



- Element deformations are prescribed `Zero` unless defined otherwise.
  $\rightarrow$ Set the truss elongation, i.e. deformation e_1, to `Released`.

Building model `pf1dtrx.spa` – Counting the DOF's

- SPACAR computes the number of degrees of freedom NDOF in the system from

  $$\mathtt{NDOF} = \mathtt{NX} - \mathtt{NXO} - \mathtt{NEO},$$

  $\mathtt{NX} = 4$ is the number of nodal coordinates,

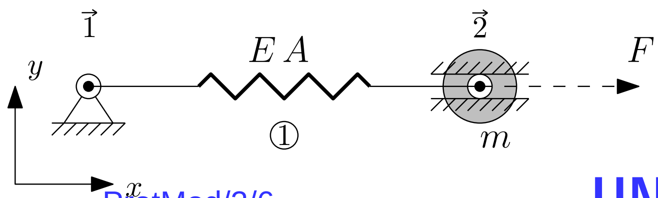
  $\mathtt{NXO} = 3$ the number of *absolute constraints*: `Fixed` coordinates,

  $\mathtt{NEO} = 0$ the number of *relative constraints*: The remaining `unReleased` element deformation parameters.

  So $\mathtt{NDOF} = 4 - 3 - 0 = 1$ degree of freedom has to be defined.

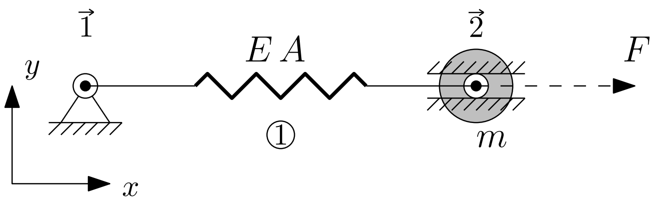
- Define e.g. an *absolute degree of freedom*:
  $\rightarrow$ Set the X coordinate of node $\vec{2}$ to be a `Dynamic DOF`.

- A *necessary* (though not *sufficient* condition for a valid system definition is that the computed number of DOF's equals the number of DOF's explicitly defined by the user.

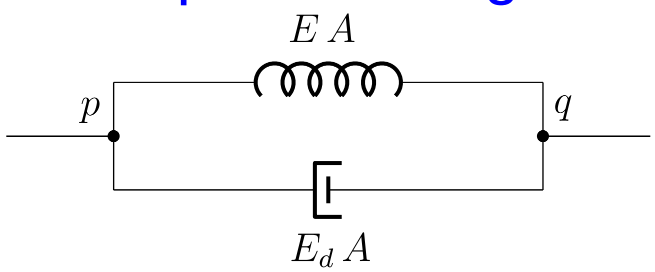


Building model `pf1dtrx.spa` – Adding mass

- Masses and inertial properties can be specified as a *lumped* mass attached at a nodal point or as a *distributed* mass present along an element.
- Lumped mass and inertia are defined by `Edit`ing the nodal point.
  $\rightarrow$ Set the `Mass` of node $\vec{2}$ to `0.206` kg.
- Distributed mass and inertia (per unit length) are defined by `Edit`ing the `Mass & Inertia` properties of the element.
  $\rightarrow$ Set the `Mass per Length` of the truss element `1` to `0.1413` kg/m (representing both leaf springs).



Building model `pf1dtrx.spa` – Adding stiffness and damping

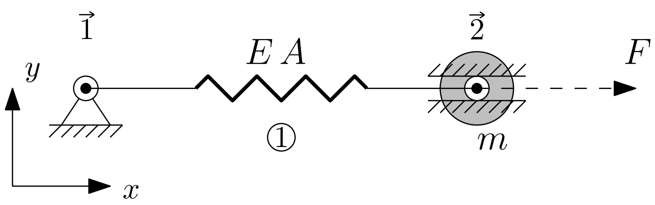


- `Edit` the `Stiffness & Damping` properties of the truss to set the axial rigidity EA and damping E_dA of the truss element.
- Knowing the equivalent longitudinal stiffness $k_{\mathrm{eq}} = 945$ N/m:
  $(EA)_{\mathrm{eq}} = k_{\mathrm{eq}} l_0$.
- For the considered spring leaves the damping is computed assuming a *relative damping* ζ in the range of 0.01 to 0.001 and knowing that the damping $d_{\mathrm{eq}} = 2\zeta\sqrt{k_{\mathrm{eq}} m_{\mathrm{eq}}}$.

  Next $(E_dA)_{\mathrm{eq}} = d_{\mathrm{eq}} l_0$.

Building model `pf1dtrx.spa` – Adding an external force

- `Edit` a node to define an external force, e.g.g. a horizontal force of `1` N in node $\vec{2}$.



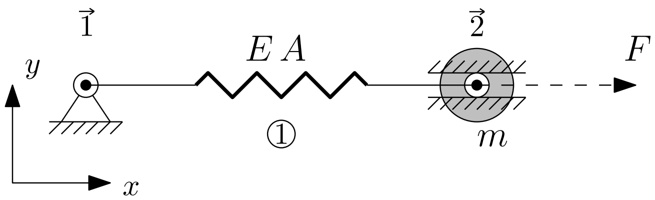
Building model `pf1dtrx.spa` – Specification of input and output

- Click the `Edit additional Dynamic commands` button to enter

```
INPUTF  1  2  1
OUTX    1  2  1
```

The keywords `INPUTF` and `OUTX` define a single input force F and a single output coordinate X, respectively.
The first parameter of both keywords is the input or output number that corresponds with its position in the input vector $\boldsymbol{u}$ or output vector $\boldsymbol{y}$.
The other two parameters define the input's or output's nodal point number and the corresponding Cartesian nodal coordinate.

Model pf1dtrx.spa – SPACAR results



F9: Create `dat` file and launch MATLAB.

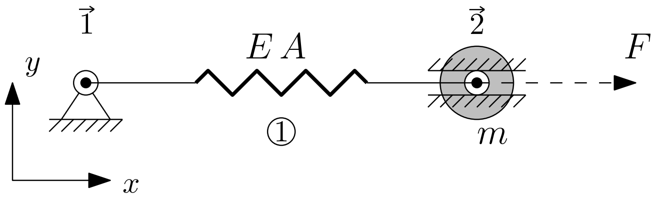
Start SPACAR from the MATLAB command prompt in mode “7”: Static analysis and the computation of the natural frequencies and mode shapes of the system:

```
>> spacar(7,'pf1dtrx')
```

Output:

- MATLAB plot window with the system in its equilibrium configuration.
- Log file `pf1dtrx.log`.
- Binary data files `pf1dtrx.sbd` and `pf1dtrx.sbm`.
- MATLAB variables with results read from these files.

Model `pf1dtrx.spa` – SPACAR results (2)



Horizontal displacement of node 2:

```
>> x(lnp(2,1))                  =   0.1011
```

The normal force in the truss element:

```
>> sig(le(1,1))                 =   1.0000
```

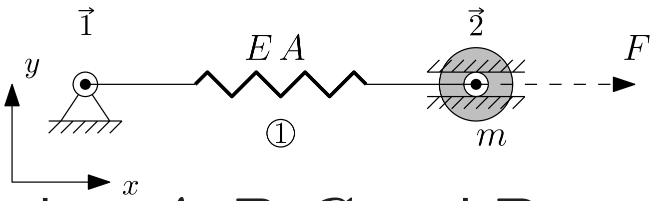
Mass, stiffness, and (first and only) natural frequency:

```
>> m0                           =   0.2107
>> k0                           = 945
>> sqrt(eig(k0,m0))/2/pi =  10.6584
```

Visualisation:

```
>> spavisual('pf1dtrx');
```

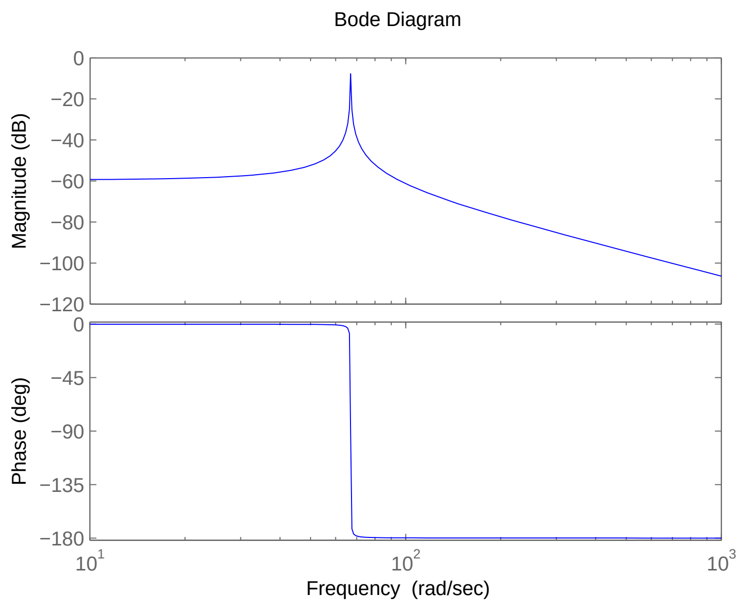
Model pf1dtrx.dat – SPACAR results (3)




```
>> spacar(9,'pf1dtrx')
```

With a mode “9” run the state space matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ are computed and stored in binary data file `pf1dtrx.ltv`. They can be read with the `getss` command.

```
>> [A,B,C,D] = getss(pf1dtrx);
>> G=tf(getss(pf1dtrx))
Transfer function:
        4.746
---------------------
s^2 + 0.1732 s + 4485
>> bode(G)
```



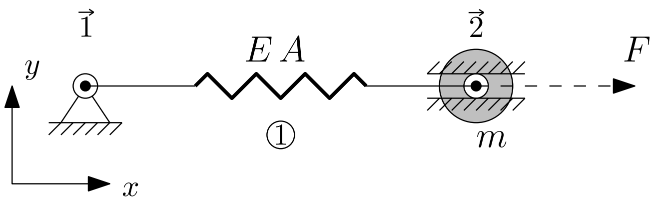
The latter command is used to create a Bode plot.

Second SPACAR model: e degree-of freedom

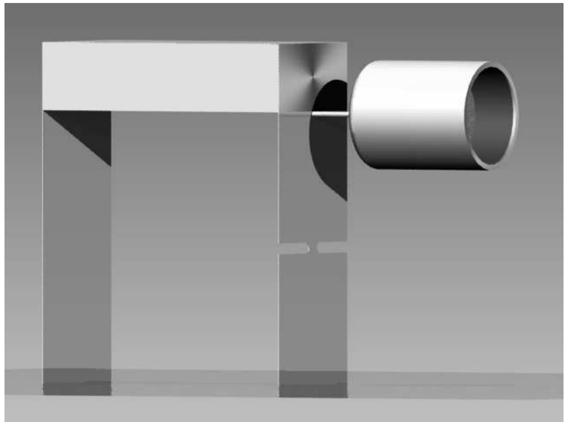
Alternatively, a *relative* degree of freedom can be defined by setting a deformation to be a `Dynamic DOF`.
Of course $\texttt{NDOF} = \texttt{NX} - \texttt{NXO} - \texttt{NEO}$ has to be satisfied.

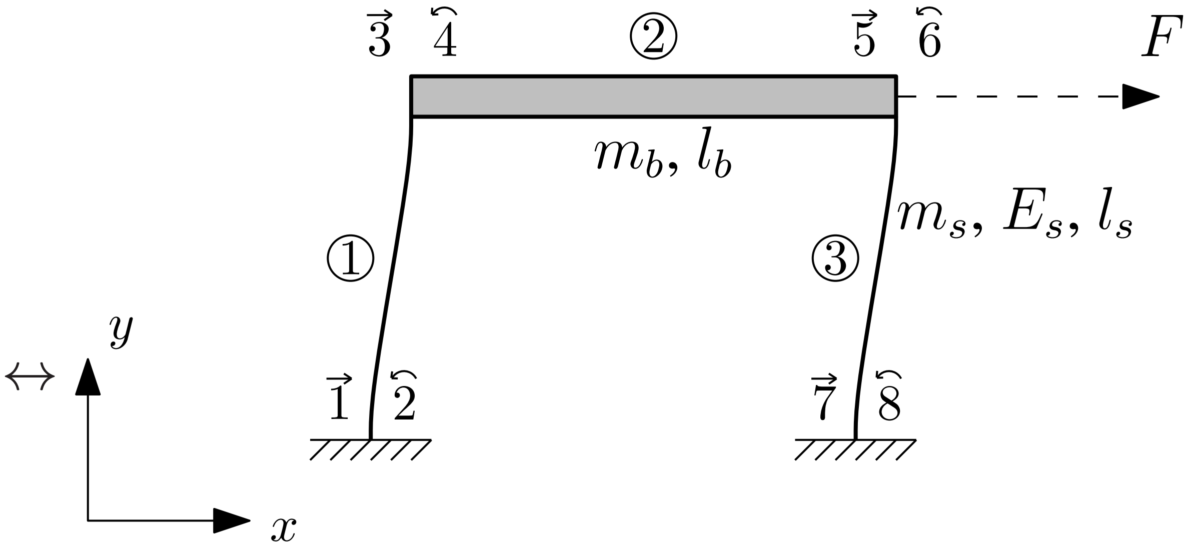
- Change `Coorinate type` of the `X` coordinate of nodal point $\vec{2}$ into its default type `Dependent`.
- Change the `Deformation type` of the `Elongation` of the truss element into a `Dynamic DOF`.

Outcome as before ...



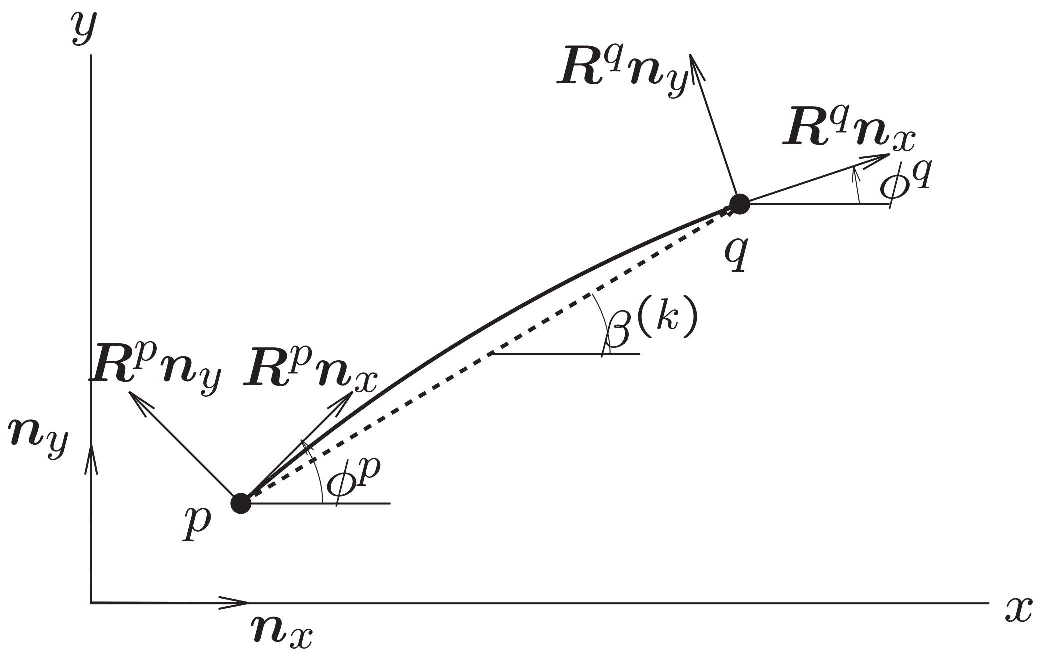
Two-dimensional model





- Bending of the leaf springs modelled more accurately using planar *beam* elements.
- Translational and *rotational* planar nodal points.

Planar beam element (PLBEAM)



- Four Cartesian coordinates (x^p, y^p), (x^q, y^q) describing the position of the beam in the (x, y)-coordinate system.
- Two rotation angles ϕ^p and ϕ^q representing the angular orientation of the triads $(\boldsymbol{R}^p\boldsymbol{n}_x, \boldsymbol{R}^p\boldsymbol{n}_y)$ and $(\boldsymbol{R}^q\boldsymbol{n}_x, \boldsymbol{R}^q\boldsymbol{n}_y)$ at the nodes p and q respectively.

Planar beam element (PLBEAM) (2)

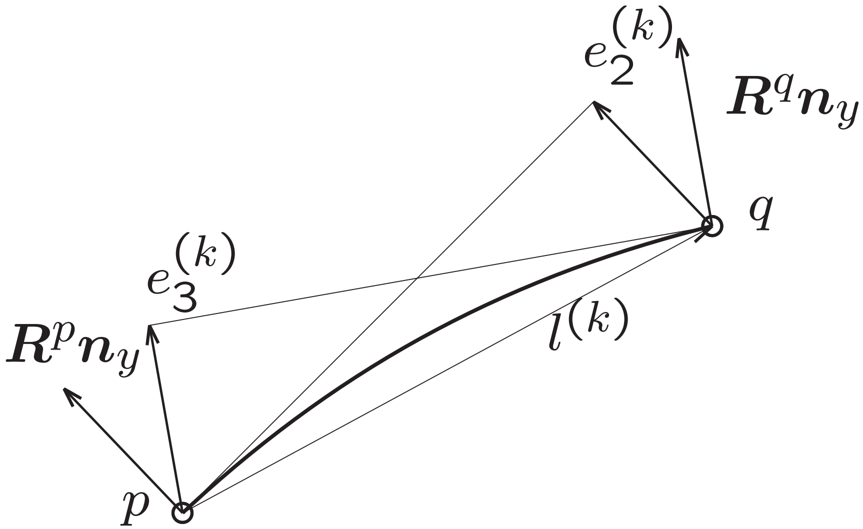
Nodal coordinates: $\boldsymbol{x}^{(k)}_{\text{beam}} = \begin{bmatrix} \boldsymbol{x}^p \\ \phi^p \\ \boldsymbol{x}^q \\ \phi^q \end{bmatrix} = [x^p, y^p, \phi^p, x^q, y^q, \phi^q]^T$.

Deformation parameters:

elongation: $e_1^{(k)} = l^{(k)} - l_0^{(k)}$,

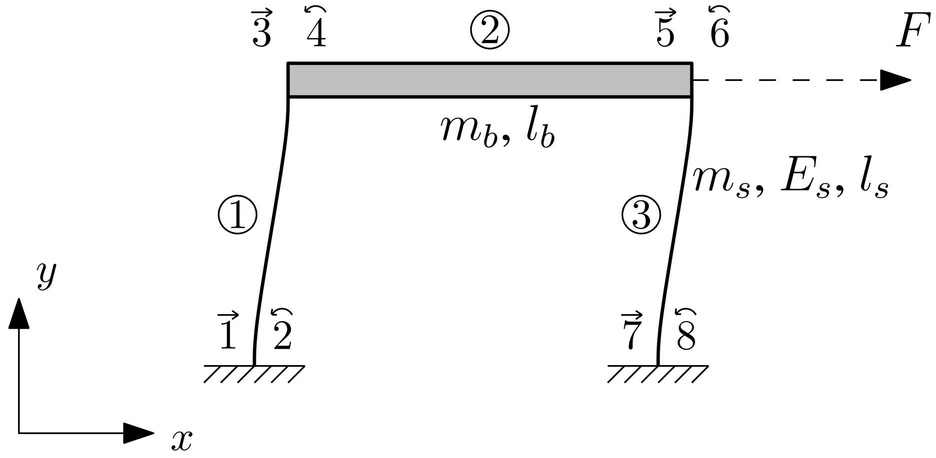
bending: $e_2^{(k)} = -(\boldsymbol{R}^p \boldsymbol{n}_y, \boldsymbol{l}^{(k)})$,

$e_3^{(k)} = (\boldsymbol{R}^q \boldsymbol{n}_y, \boldsymbol{l}^{(k)})$,



The deformations are independent for rigid body movements of the element and have a clear physical meaning.

1-DOF model



Three beams:

- Common nodes
  $\rightarrow$ rigid connection.
- Fixed support: Translational nodes 1 and 7, rotational nodes 2 and 8.
- Solid bar: All deformations `Zero (default)`.
- In both leaf springs: `Release` bending modes e_2 and e_3.

$\texttt{NDOF} = \texttt{NX} - \texttt{NXO} - \texttt{NEO} = 1$ as

$\texttt{NX} = 12$ is the number of nodal coordinates $(4 \times 2 + 4 \times 1)$,
$\texttt{NXO} = 6$ the number of absolute constraints (`Fixed` $2 \times 2 + 2 \times 1$),
$\texttt{NEO} = 5$ the number of relative constraints, so $3 \times 3 - 4$
So $\texttt{NDOF} = 1$: E.g. horizontal position of node 3.

1-DOF model (2)

Distributed inertia of the solid bar ② in a lumped mass representation:

1. The mass of the element should be equal to the sum of the lumped masses.
2. The center of mass of the element and that of the discrete mass model should coincide.
3. The rotational inertia of the element and that of the lumped system should be equal.

m
$J_c = \frac{1}{12}ml^2$

Solid bar

$m^p = m/2$
$J^p = -\frac{1}{12}ml^2$

$m^q = m/2$
$J^q = -\frac{1}{12}ml^2$

Discrete mass model

Solid bar with mass m and length l: the rotational inertia relative to the centre of mass $J_c = \frac{1}{12}ml^2$. Equivalent lumped masses and rotational inertias are then:

$$m^p = m^q = \frac{1}{2}m \quad \text{and} \quad J^p = J^q = -\frac{1}{12}ml^2.$$

1-DOF model (3)

- Define lumped and/or distributed masses and inertia properties.
- For the leaf springs it is convenient to use the `Calculate Inertia` button, provided the beam dimensions and density are defined correctly.
- Define the axial rigidity EA (related to longitudinal stiffness EA/l) and flexural rigidity EI (related to bending stiffness EI/l^3).
- The `Calculate` button offers a user-friendly automatic calculation of the stiffness properties.
- Damping properties need to be defined manually, see previous model with relative damping ζ.

1-DOF model (4)

- SPACAR analysis mode 7 and 9 as before: Still 1 natural frequency!

- SPACAR analysis mode 8 for buckling analysis:

The critical loading parameter λ_i is computed, such that the buckling load

$$f_i = \lambda_i f_0,$$

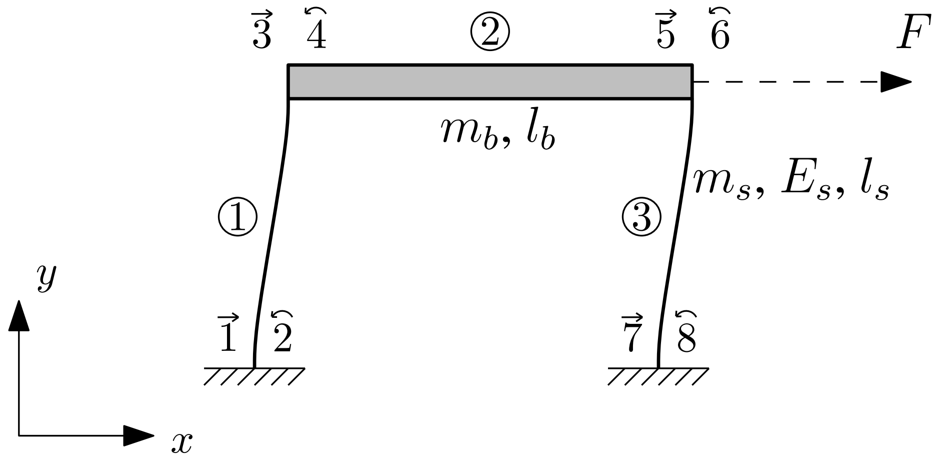
where f_0 represents a static reference loading vector of nodal forces/torques.

`Edit` node 3 to apply a vertical force with a magnitude of 1 N in the negative y direction.

Determine the equilibrium configuration and the critical loading parameter λ_1:

```
>> spacar(8,'pf2db')
>> eig(-k0,g0)          = 78.7500
>> spavisual('pf2db')
```

3-DOF model



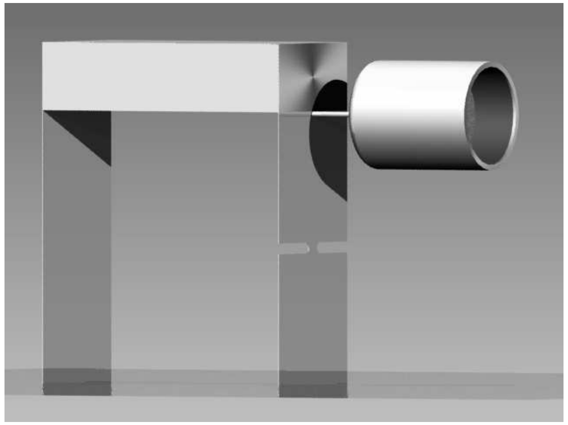
- `Release` elongations e_1 of both leaf springs as well.

$\mathtt{NDOF} = \mathtt{NX} - \mathtt{NXO} - \mathtt{NEO} = 3$ as

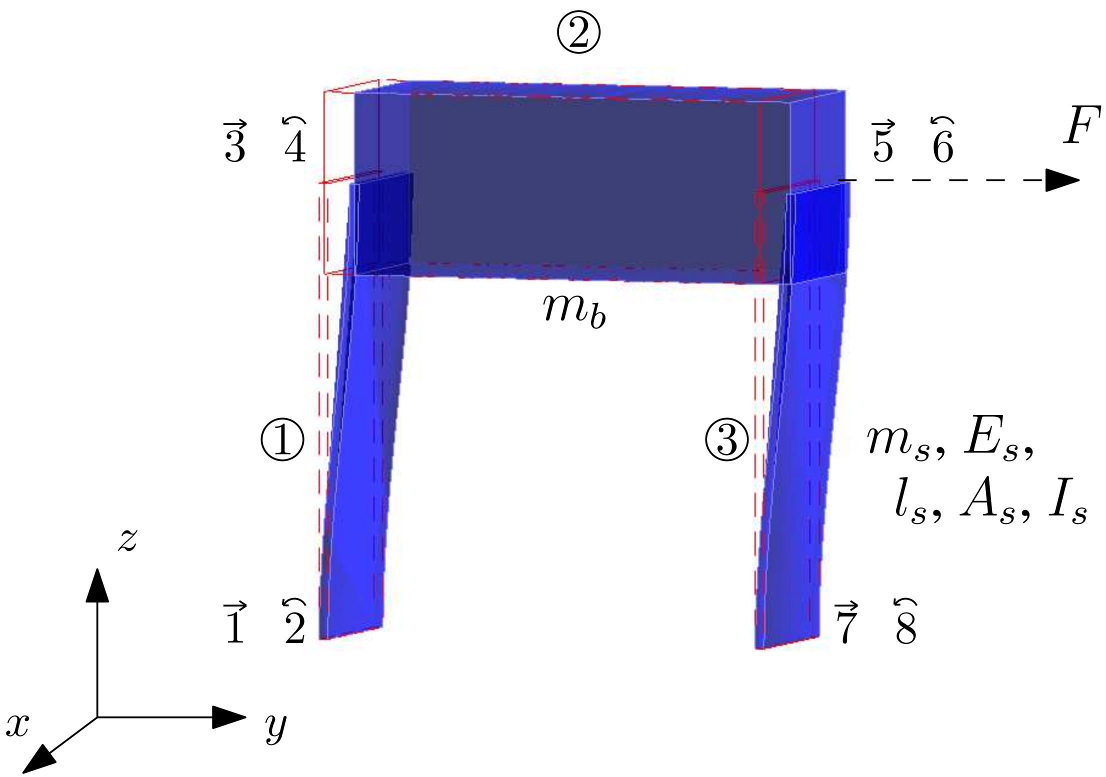
$\mathtt{NX} = 12$ is the number of nodal coordinates $(4 \times 2 + 4 \times 1)$,
$\mathtt{NXO} = 6$ the number of absolute constraints (`Fixed` $2 \times 2 + 2 \times 1$),
$\mathtt{NEO} = 3$ the number of relative constraints, so only the solid bar
So $\mathtt{NDOF} = 3$: E.g. X and Y coordinates of node 3 plus rotation of node 4.

Next to the first natural frequency (10.6 Hz), now also higher natural frequencies are found (2129 Hz and 3585 Hz), justifying the previous 1-DOF approximation.

Three-dimensional model

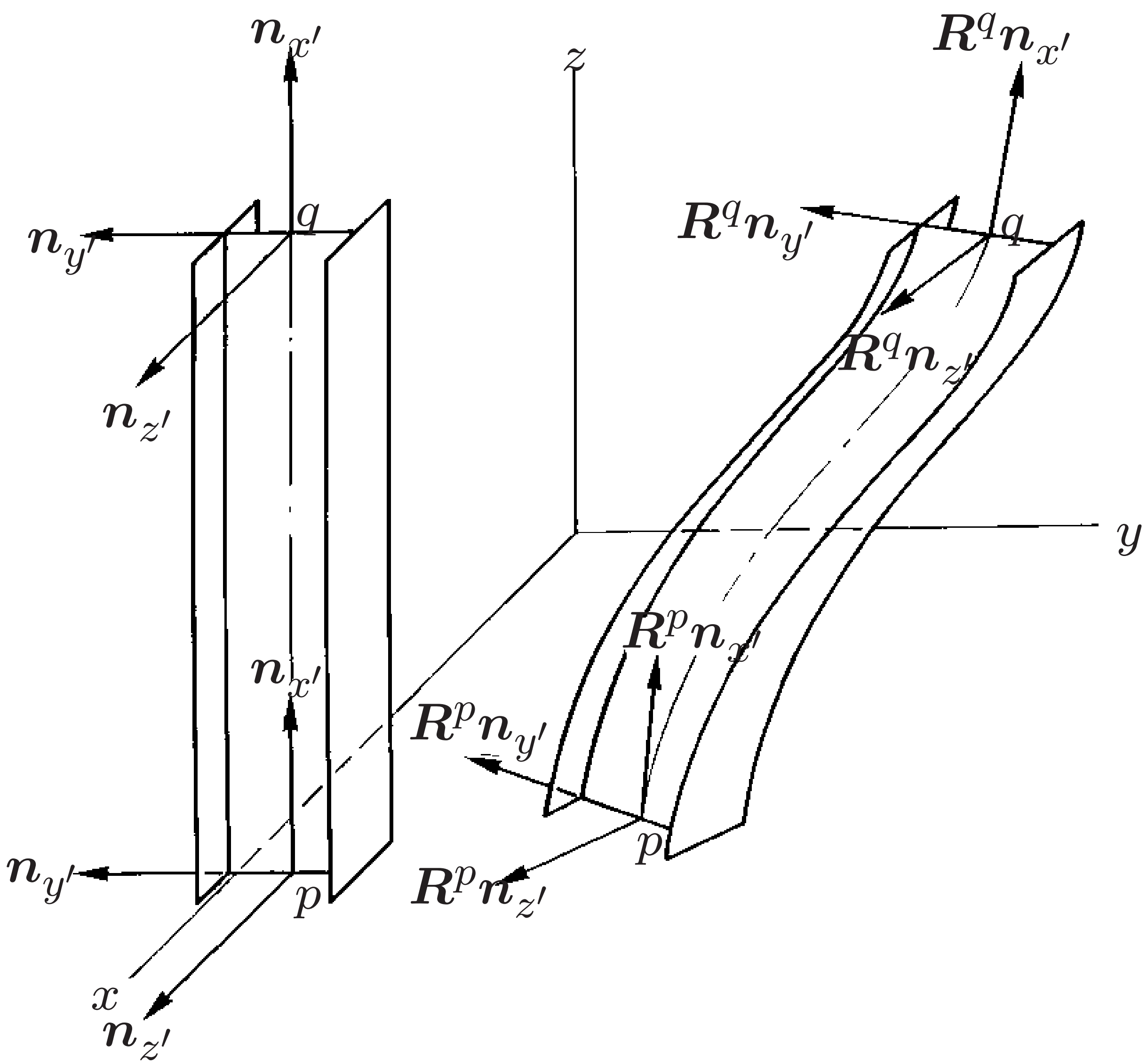


$\leftrightarrow$



- Modelled with *spatial* elements.
- Translational and rotational *spatial* nodal points.

Spatial beam element (BEAM)



Spatial beam element (BEAM) (2)

- Six translational coordinates are from two position vectors $\boldsymbol{x}^p$ and $\boldsymbol{x}^q$ describing the position of the beam in the fixed inertial coordinate system.
- Six independent rotational coordinates as the orientation of each rotational node in three dimensions is given by three independent rotation coordinates collected in the vectors $\boldsymbol{\lambda}^p$ and $\boldsymbol{\lambda}^q$ respectively.
  $\rightarrow$ orientation of the triads $(\boldsymbol{n}_{x'}, \boldsymbol{n}_{y'}, \boldsymbol{n}_{z'})$ at the nodes p and q.
- Spatial beam: 12 (independent) nodal coordinates.
  As a rigid body: 6 degrees of freedom.
  So $12 - 6 = 6$ independent deformation parameters:

elongation: $e_1^{(k)} = l^{(k)} - l_0^{(k)}$,

torsion: $e_2^{(k)} = \frac{1}{2} l_0^{(k)} \left[(\boldsymbol{R}^p \boldsymbol{n}_{z'}, \boldsymbol{R}^q \boldsymbol{n}_{y'}) - (\boldsymbol{R}^p \boldsymbol{n}_{y'}, \boldsymbol{R}^q \boldsymbol{n}_{z'}) \right]$,

bending:

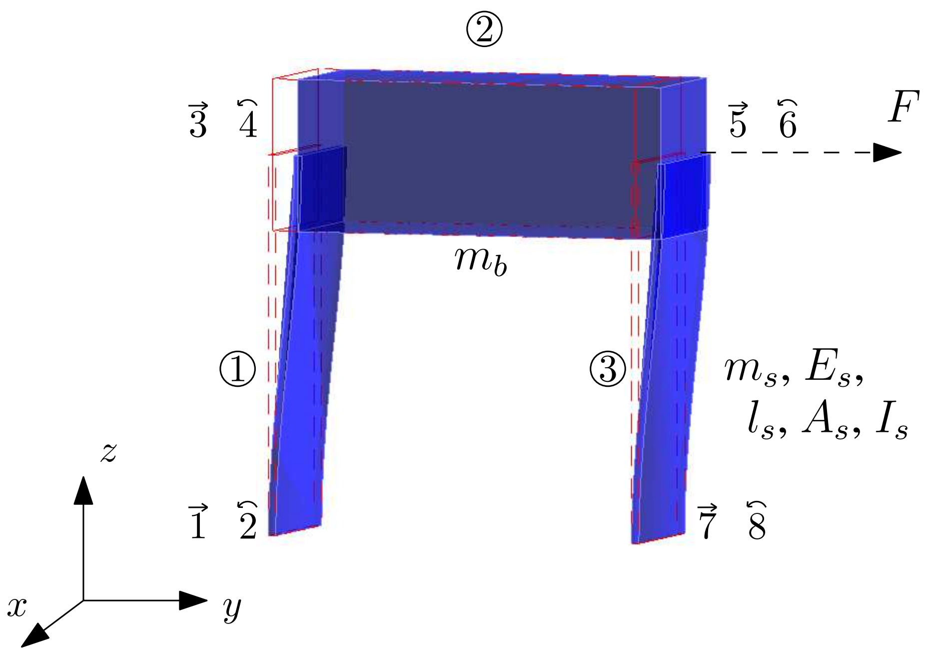
$$e_3^{(k)} = -(\boldsymbol{R}^p \boldsymbol{n}_{z'}, \boldsymbol{l}^{(k)}), \qquad e_4^{(k)} = (\boldsymbol{R}^q \boldsymbol{n}_{z'}, \boldsymbol{l}^{(k)}),$$

$$e_5^{(k)} = (\boldsymbol{R}^p \boldsymbol{n}_{y'}, \boldsymbol{l}^{(k)}), \qquad e_6^{(k)} = -(\boldsymbol{R}^q \boldsymbol{n}_{y'}, \boldsymbol{l}^{(k)}).$$

1-DOF model

$$\begin{aligned} \texttt{NDOF} &= \texttt{NX} - \texttt{NXO} - \texttt{NEO} \\ &= 24 - 12 - 11 \\ &= 1, \end{aligned}$$

Note: Internally in SPACAR *rotational nodal points are described by so-called* Euler parameters *(4 in each node)*
$\Rightarrow$ NX *and* NEO *are counted differently.*



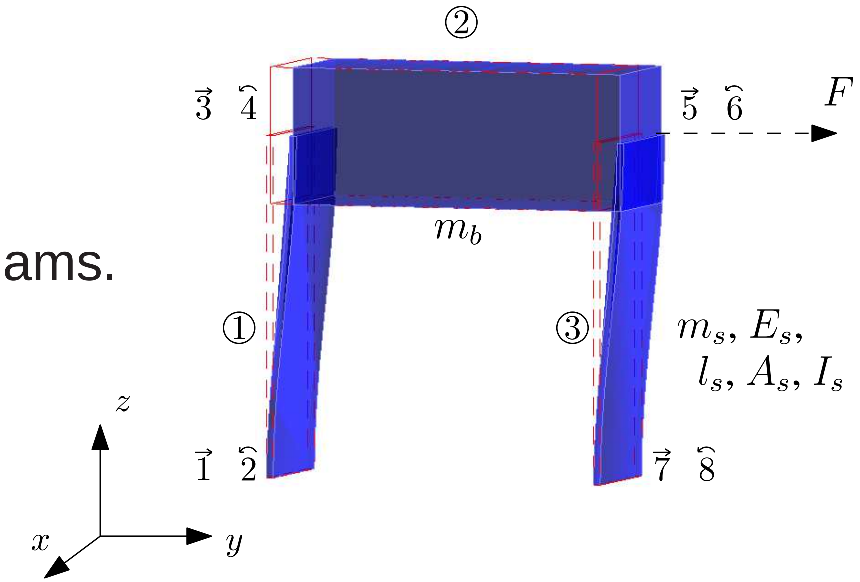
NX $= 24$ is the number of nodal coordinates $(4 \times 3 + 4 \times 3)$,

NXO $= 12$ the number of absolute constraints (`Fixed` $2 \times 3 + 2 \times 3$),

NEO $= 11$ the number of relative constraints, so $3 \times 6 - 11 = 7$ deformation parameters have to be released,

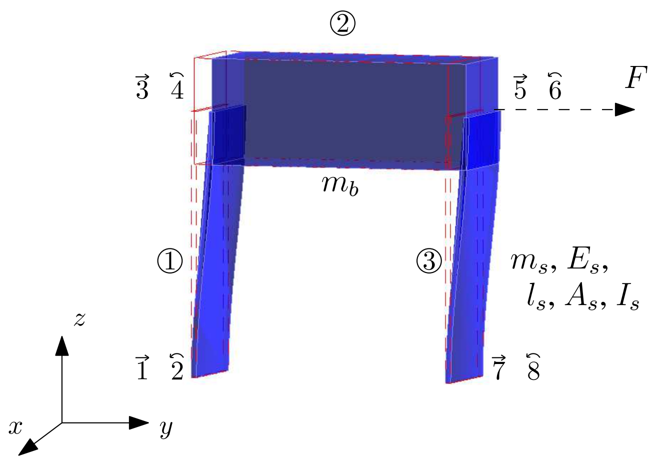
NDOF $= 1$: Horizontal position of node 3.

1-DOF model (2)



- Create a 3D model with 3 *spatial* beams. The beams' principle y' axes must be specified.
- Set the coordinates of the supports to `Fixed`.
- `Release` the bending modes e_5 and e_6 of both leaf springs.
- In addition the torsion and remaining bending modes of element 3 are released in accordance with the local cut in the leaf spring.

1-DOF model (3)

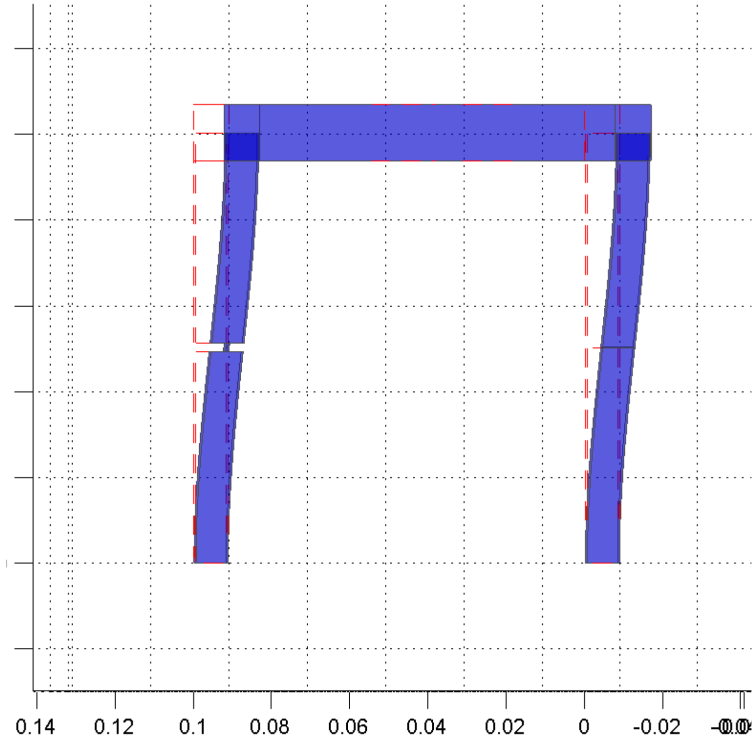


- Define lumped and/or distributed masses and inertia properties.
  $\rightarrow$ `Calculate Inertia` button.
- Define the axial rigidity EA (related to longitudinal stiffness EA/l), torsional rigidity GI_t (related to the (equivalent) torsional stiffness GI_t/l^3), and flexural rigidities $EI_{y'}$ and $EI_{z'}/l^3$ (related to bending stiffnesses $EI_{y'}/l^3$ and $EI_{z'}/l^3$).
  $\rightarrow$ `Calculate` button.
- Damping properties.

Analysis in MATLAB as before ...

19-DOF model

Modelling the leaf spring2 more accurately with five beams and releasing 25 deformations:

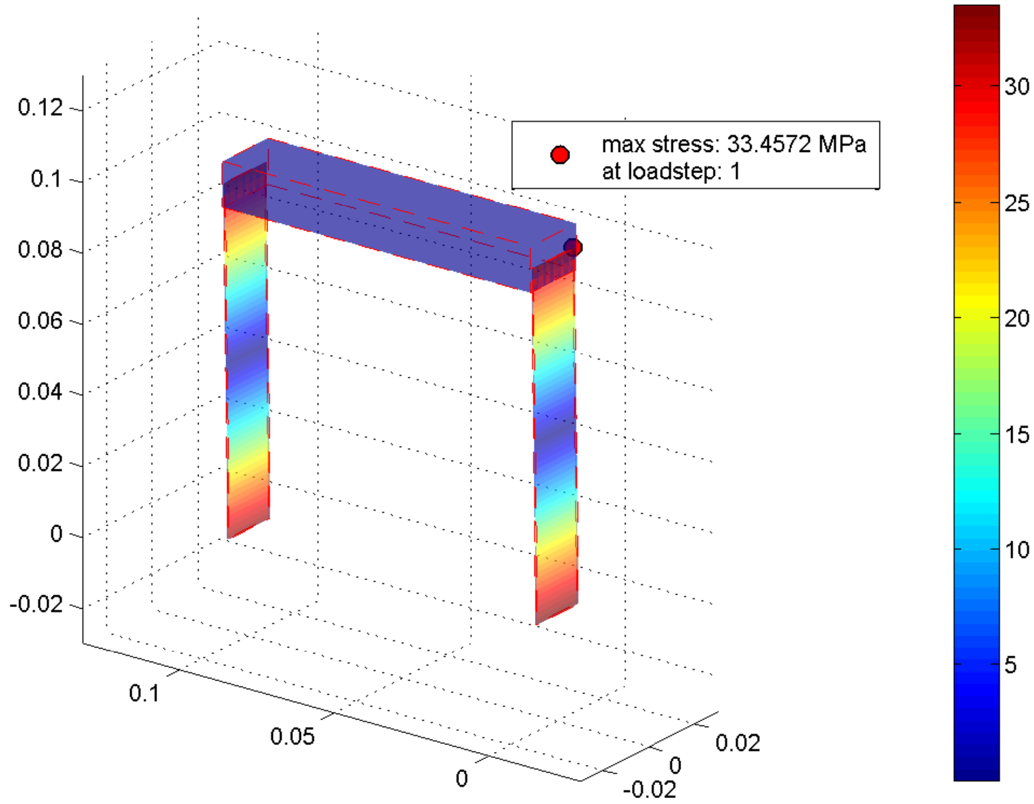


$$\begin{aligned}\mathtt{NDOF} &= \mathtt{NX} - \mathtt{NXO} - \mathtt{NEO} \\ &= 2*7*3 - 12 - (6+5) \\ &= 19,\end{aligned}$$

$\mathtt{NX} = 42$ is the number of nodal coordinates $(7 \times 3 + 7 \times 3)$,
$\mathtt{NXO} = 12$ the number of absolute constraints ($\mathtt{FIX}$ $2 \times 3 + 2 \times 3$),
$\mathtt{NEO} = 11$ the number of relative constraints (only solid bar and elongations),
$\mathtt{NDOF} = 19$: Translation of solid bar plus additional deformations.

Von Mises stresses

SPAVISUAL can also visualise the stress distribution in flexible spatial beams, e.g. in the 19-DOF model with a horizontal force of 1 N:



Help, I get lost in defining DOF's!

A SPACAR model refuses to run until you have managed to define the correct number of DOF's. Some hints:

- Start with a simple low dimensional model. Next remove the constraint on additional elastic deformations by defining these as dynamic DOF.
- In the GUI it is possible to “display a complete overview of all DOF” (F5).
- If all else fails, the SPACAR mode 0 may help. It offers a kinematic analysis to detect overconstrained and underconstrained systems. Visualisation with SPAVISUAL gives insight that helps to achieve exact constrained design.

Help, is this ... result correct?

- Note that computed stiffnesses, stresses, etcetera in beams depend strongly on the assumption that the beam model may be applied.

Help, I need to model a belt or pulley!

For *planar* systems a `PLBELT` element is available.
Example: PLBELT to model an elevator: