ISAE SUPAERO, 2018-19 SM301

Finite Element Analysis : the engineer's way

Author : Prof. **Joseph MORLIER** : `joseph.morlier@isae-supaero.fr`

Keywords : Big Picture of FEA and best practices in Computational Structural Mechanics aka CSM

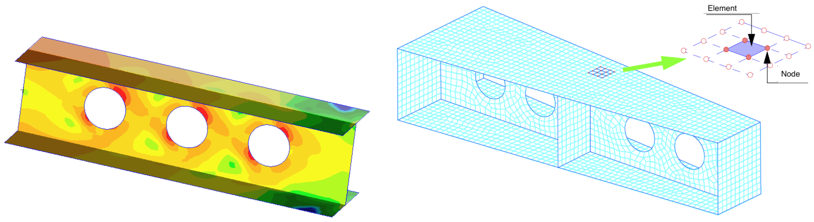

Fig. 1: Stress Analysis (Nastran) Nice colors ? But before we need to mesh... .

Table des matières

1 Overview

The key idea is to subdivide the structure into elements and nodes (finite numbers), displacements and rotations (if they exist) are the unknowns, results in huge systems of algebraic equations

Assemble element stiffness matrices
What is different is that we use elements other than frame elements (e.g. membrane, plate or shell elements)
The finite element method generally provides an approximate solution whose accuracy increases as more element, and hence unknowns, are used.

This method can be used for fluid dynamics (CFD), thermal analysis and/or any multiphysics analysis. Visit Wolfram website `http://reference.wolfram.com/language/FEMDocumentation/tutorial/SolvingPDEwithFEM.html`. In fact FE method is capable of Solving Partial Differential Equations (see First course). This manuscript is dedicated to Computational Structural Mechanics (CSM) only.

1.1 Hand vs. computer analysis of structure

Let's take the example of semicircular 2-pinned arch.

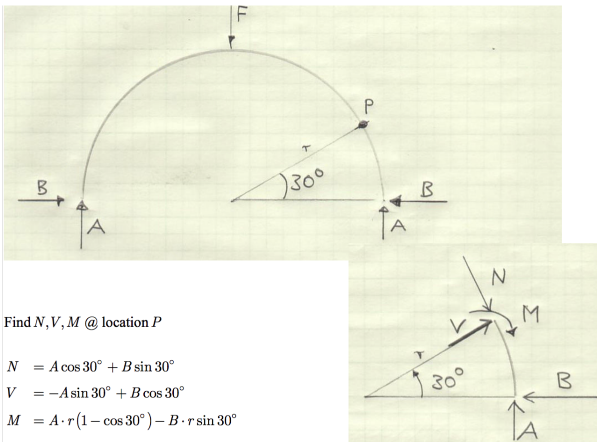

Fig. 2: Semicircular continuous approach .

In this case the level of structure indetermination is of degree one.

• use three equilibrium equations and one compatibility equation to solve for reactions, then cut free body and determine N,V,M

• process is difficult to automate

for example if we had 3 pinned arch, so statically determinate four EQ and zero compatibility equation!!

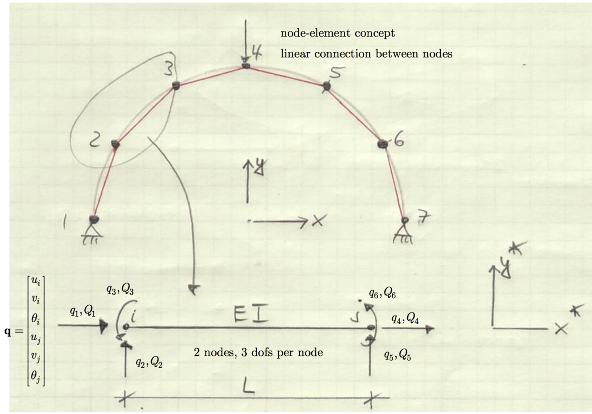

Fig. 3: Semicircular discrete approach .

2 A bit of theory

subdivide structure into nodes and elements (probably straight), node-element concept, internal forces are N,V,M (see strength of materials/aeronautical structures) leads into the matrix problem (assume linear structural behavior) :

$\mathbf{K}q = f$ or $\overline{K}\widetilde{q} = \widetilde{f}$ or $[K]\{q\} = \{f\}$ my favourite is $\underline{\underline{K}} * \underline{q} = \underline{f}$

with q vector of element nodal displacements, f vector of element nodal forces, K element stiffness matrix. The above equation is a force-displacement relation in matrix form, which forms the basis of CSM. In order to get K , we need :

(1) stress-strain relation (constitutive relation)

(2) strain-displacement relation (kinematic relation)

(3) Equilibrium

Those three fundamental relations of structural analysis are part of the force method, the slope-deflection method, the solution to the differential equation or any conventional method of structural analysis, i.e. we can use any of these methods to derive K .

Note that this course will demonstrate bar, beam and membrane element. For plate/shell let's see another advanced course or ref.

NOTE —

The time derivative of v will be noted : $\dot{v}$

The space derivative of v will be noted : v'

3 (Re)view of Rayleigh-Ritz method

> Assume displaced shape (trial functions) and use minimum potential energy to find the best solution among the trial functions
> Mathematically this step consists of finding the constants c of the trial functions
> Trial functions must satisfy geometric boundary conditions
> If the set of trial functions contains the exact solution, then the method will find that solution

We start by writing a short review of beam's theory.

Side-trip: Beam kinematics

ϕ : curvature
θ : angle of rotation of cross section
v' : angle of rotation of beam axis

The concept of plane section remain plane leads to

$\varepsilon(x,y) \quad = -y\,\phi(x) = -y\,\theta'(x)$

If the cross section is assumed to remain perpendicular to the beam axis (i.e. if we ignore shear deformation), we have

$\varepsilon(x,y) \underbrace{=}_{\theta=v'} -\,y\,v''(x)$

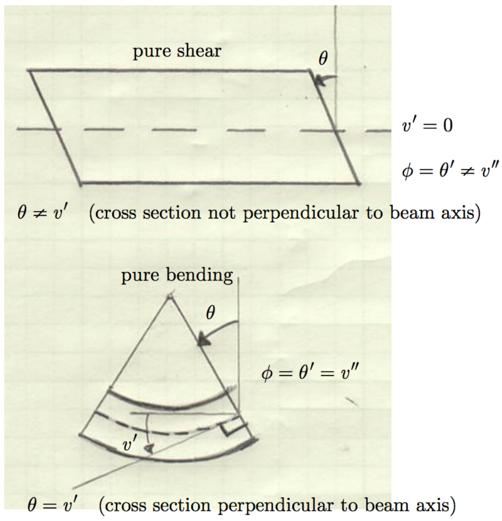

Fig. 4: Review of linear beam static's theory .

The internal energy stored in a beam element (neglecting shear deformation) is :

$W_i = \frac{1}{2}\int_0^L \sigma\varepsilon dV$

then the work done by external element loading along the element displacements :

$W_e = \frac{1}{2}\int_0^L v\,(x)\,p\,(x)\,dx$

Let's try to solve a Cantilever with uniformly end load Problem. We'll find the deflected shape and the moment diagram of structure. The demonstration will be made in course for a quadratic trial function. You'll nee to test your skills on a cubic trial function with homework 1.

Let's try to solve the problem of the figure with $v(x) = cx^2$

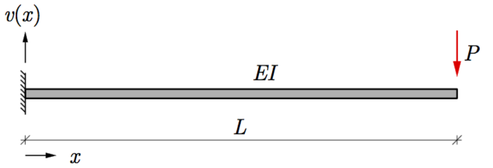

Fig. 5: Cantilever beam example .

so :
$v'(x) = 2cx$
$v''(x) = 2c = \phi(x) = \text{curvature!!!}$

$$\begin{aligned}
\mathrm{W}_i &= \frac{1}{2}\int_0^L \sigma\varepsilon dV = \frac{1}{2}\int_0^L \varepsilon E\varepsilon dV \\
&= \frac{1}{2}\int_0^L \phi(x)yE\phi(x)ydV = \frac{1}{2}\int_0^L \phi(x)yE\phi(x)ydV \\
&= \frac{1}{2}\int_0^L \phi(x)yE\phi(x)ydAdx = \frac{1}{2}\int_0^L EI\phi(x)^2dx \ (1)
\end{aligned}$$

Using Hooke's law and minimum of potential energy, we can write
$W = W_i - W_e = \frac{1}{2}\int_0^L EI\phi(x)^2dx - Pv(L)$
by integrating and subtituing $\phi(x)$ it gives $W = \frac{1}{2}EI4cL^2 + PcL^2$
Thus deriving
$\frac{\partial W}{\partial c} = 4EIcL + PL^2 = 0 \rightarrow c = -\frac{PL}{4EI}$
and expressing $M(x) = EIv''(x) = EI(-\frac{PL}{4EI}) = -\frac{PL}{2}$

Note that we could have used chain rules for avoiding integrate the square of Phi :
$\frac{\partial W_i}{\partial c} = \frac{2}{2}EI\int_0^L \phi(x)\frac{\partial\phi(x)}{c}dx = EI * 2c * 2L$;)
Finally we got : $v(L) = -\frac{PL^3}{4EI}$.

The exact value is $v(L) = -\frac{PL^3}{3EI}$, try cubic trial function $c_1x^2 + c_2x^3$: It means 2 unknowns... see homework 1, Illustration of potential energy too.

We have seen that we obtain a more accurate solution is to increase the degree of the polynomial of the trial functions. We have a second option : That is decrease the size of the element for which each trial functions is valid, that is subdivide the beam into several elements and assume independent trial functions for each of those elements. In the RR method with free constants c it is difficult to enforce the boundary conditions between adjacent elements. A natural way to enforce the boundary conditions is to express the trial function in terms of nodal parameters (displacement and rotation) rather the free constants as in the Rayleigh-Ritz method. This is the key idea of the finite element method. We will demonstrate this concept in the next chapter using beam elements, the simplest family of finite elements.

4 Beam finite element

First course introduce the direct stiffness method to find the stiffness matrix of a bar (1D, works in traction/compression only).

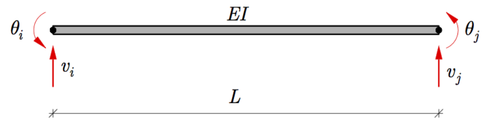

Fig. 6: 4DOFs beam .

In this chapter we demonstrate the basics of the finite element method using the simple beam above.

The four nodal parameters allow us to interpolate the deflection by a cubic polynomial. As in the Rayleigh-Ritz method, we start by using free constants c.

Let's first define ξ adimensional (see mapping...) length equal to x/L.

Then we can write (the second is easy to derive function of x ;)

v(ξ) $= c_1 + c_2 * \xi + c_3 * \xi^2 + c_4 * \xi^3$

$v'(\xi) = (1/L) * (c_2 + c_3 * \xi + c_4 * \xi^2)$ (2)

or in matrix form : $v(\xi) = [1, \xi, \xi^2, \xi^3]\underline{c}$

with $\underline{c} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}$

Next, we use the four boundary conditions of the previous figure to write :

$v(x = 0) = v_i$

$v'(x = 0) = \theta_i$

$v(x = L) = v_j$

$v'(x = L) = \theta_i$

In matrix form using

$v(\xi) = [1, \xi, \xi^2, \xi^3]\underline{c}$

it gives : $\begin{bmatrix} v_i \\ \theta_i \\ v_j \\ \theta_j \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/L & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1/L & 2/L & 3/L \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}$ or in matrix form :

$\underline{q} = \underline{\underline{G}} * \underline{c}$

thus $G^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & L & 0 & 0 \\ -3 & -2L & 3 & -L \\ 2 & L & -2 & L \end{bmatrix}$

We substitute in (4) the inverse of this relation $\underline{c} = \underline{\underline{G}}^{-1} * \underline{q}$ to obtain :
$v(\xi) = [1, \xi, \xi^2, \xi^3]\underline{\underline{G^{-1}q}}$
so
$v(\xi) = N(\xi)\underline{q}$
with
$N(\xi) = [1 - 3\xi^2 + 2\xi^3, L(\xi - 2\xi^2 + \xi^3), 3\xi^2 - 2\xi^3, L(\xi^3 - \xi^2)]$

where N is a 4x1 row vector of shape functions. Note that the concept of shape functions for beam elements is to fit a curve between two points at which both the ordinate (q1,q3) and the slope (q2,q4) are known (four data items define a cubic polynomial).

Shape functions for beam element with four degrees of freedom can be computed using this code untitled shape1.m on LMS : x is xi ?

```
syms x L real % symbolic variable
P = inline('[1 x x^2 x^3]') % third order polynomial
dP = inline(diff(P(x))) % symbolic derivation
Pn = [ P(0); dP(0); P(L) ; dP(L) ] % Nodes evaluation
N = inline(( P(x) * inv(Pn))) % Shape Functions
dN = inline((dP(x) * inv(Pn))) % Derivatives
% graphs
x = 0:0.01:1;
subplot(2,1,1), plot(x, N(1,x')), title(' Shape functions N1 N2 ...
    N3 N4 ')
subplot(2,1,2), plot(x,dN(1,x')), title(' Derivative dN1 dN2 dN3 ...
    dN4 ')
end
```

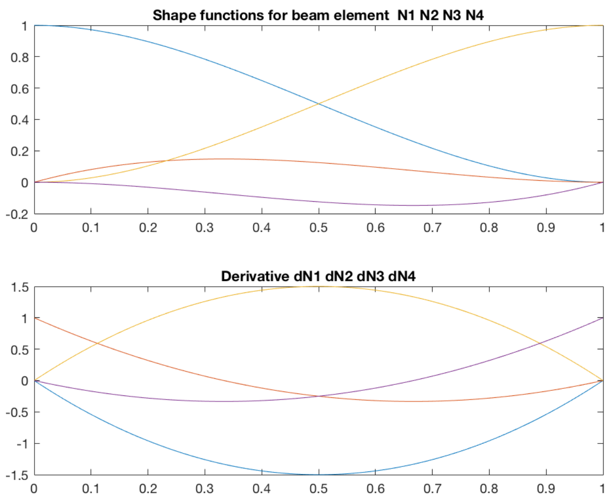

Fig. 7: Shape functions, easy to analytically derive, isn't it.

BUT —

how using this, can I postprocess the strain/Stress to obtain nice colors ?

In structural analysis of beams, we work with a generalized strain, the curvature. In linear structural analysis (small displacement theory) and ignoring shear deformation, the curvature (x) is the second derivative of the deflected shape v(x), thus

$$\phi(x) = \frac{d^2v(\xi)}{dx^2} = \frac{d^2}{dx^2}N(\xi)q = \frac{1}{L^2}\frac{d^2}{dx^2}N(\xi)q = B(\xi)q \tag{3}$$

with $\xi = x/L$

The 4x1 vector B can be written :

$$B(\xi) = \frac{1}{L^2}\frac{d^2}{dx^2}N(\xi) = [\frac{1}{L^2}(-6+12\xi); \frac{1}{L}(-4+6\xi); \frac{1}{L^2}(6-12\xi); \frac{1}{L}(-2+6\xi)] \tag{4}$$

is commonly referred to as strain-displacement transformation vector since it relates the strain (curvature) and nodal displacements q.

Then we should use the law of linear elasticity (HOOKE) which relates the curvature, the generalized strain for beam elements, to the bending moment (the corresponding generalized stress)

$M(\xi) = EI\phi(\xi) = EIB(\xi)q$

BUT —

We discuss a lot about $\underline{\underline{K}} * \underline{q} = \underline{f}$, How can I construct $\underline{\underline{K}}$?

Remember the RR approach :

$W_i = \frac{1}{2}\int_0^L \sigma\varepsilon dV = \frac{1}{2}\int_0^L \varepsilon E\varepsilon dV$
$= \frac{1}{2}\int_0^L \phi(x)yE\phi(x)ydV = \frac{1}{2}\int_0^L \phi(x)yE\phi(x)ydV$
$= \frac{1}{2}\int_0^L \phi(x)yE\phi(x)ydAdx = \frac{1}{2}\int_0^L EI\phi(x)^2dx = \frac{1}{2}\int_0^L B(x) * q * B(x) * q * EIdx$

using matrix.vector product :

$$\frac{1}{2}\int_0^L q^T * B^T(x) * B(x) * q * EIdx \tag{5}$$

Note that $\epsilon(y) = \phi * y$ and $\int y^2dA = I$

The external work done on a beam element is the work done by the nodal forces Q along the nodal displacements (general force and moment matrix) and the work done by external element loading F along the element displacements (if element loading p(x) is present).

$$We = q^T*Q+\int_O^L N(x)*q*p(x)dx = q^T*Q+q^T\int_O^L N^T(x)*p(x)dx = q^T*Q+q^T*F \tag{6}$$

with $F = \int_O^L N^T(x) * p(x)dx$

The vector F contains forces applied to the nodes that are equivalent (in an energy sense) to the element loading.It is thus often referred to as vector of equivalent nodal forces. Note that the potential energy is now a function of the nodal displacements q , in the Rayleigh-Ritz method it is a function of the free constants c . Taking derivative with respect to q yields

$$\frac{\partial W}{\partial q} = \begin{bmatrix} \frac{\partial W}{\partial q_1} \\ \frac{\partial W}{\partial q_2} \\ \vdots \\ \frac{\partial W}{\partial q_n} \end{bmatrix} = \int_0^L B^T(x) * EI * B(x)dx * q - Q - F = 0 \tag{7}$$

Isn't it $\underline{\underline{K}} * \underline{q} = \underline{f}$ with $\underline{f} = Q + F$
and

$$\underline{\underline{K}} = \int_0^L B^T(x) * EI * B(x)dx \tag{8}$$

We then try to construct the first element of this 4x4 matrix (4DOFs)=
it comes :

$$k_{11} = \int_0^L B_1^T(x) * EI * B_1(x)dx = \int_0^1 B^T(\xi) * EI * B(\xi)Ld\xi$$

$=\int_0^1 (1/L^2)(6 - 12\xi) * EI * (1/L^2)(6 - 12\xi) * Ld\xi$
$=(\mathrm{EI/L}^3) \int_0^1 (6 - 12 * \xi)^2 d\xi$
$=(\mathrm{EI/L}^3)(-1/(12 * 3)) \left[(6 - 12\xi)^3\right]_0^1$
$= \frac{12EI}{L^3}$(9)
finally,

$$\underline{\underline{K}} = (EI/L^3) \begin{bmatrix} 12 & 6L & -12 & 6L \\ & 4L^2 & -6L & 2L^2 \\ sym & & 12 & -6L \\ & & & 4L^2 \end{bmatrix} \tag{10}$$

Similarly one can obtain Mass matrix

$$\underline{\underline{M}} = \int_0^L N^T(x) * \rho * A * N(x)dx \tag{11}$$

with final computation leads to :

$$\underline{\underline{M}} = (\rho * A * L/420) \begin{bmatrix} 156 & 22L^2 & 54 & -13L \\ & 4L^2 & 13L & -3L^2 \\ sym & & 156 & -22L \\ & & & 4L^2 \end{bmatrix} \tag{12}$$

Only valid If the beam is made of the same homogeneous material and the same section.

One other remark concerning the mass matrix is that we can associate half of the total mass of the element with the degrees of translation of each node. We only take the translations because the rotations do not produce forces of inertia. The concentrated mass matrix is thus written as follows :

$$\underline{\underline{M}} = (\rho * A * L/420) \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{13}$$

It should be noted that the two concentrated and distributed matrices give the same total mass with the sum of the components associated with degrees of freedom of translation : M(1, 1) M(1, 3) M(3, 1) M(3, 3)

BUT —

Identify ?

By writing Potential and Kinematic energy of the beam we can directly identify $\underline{\underline{K}}$ and $\underline{\underline{M}}$ using the quadratic form per element.

$$V_e = \frac{1}{2}\int_0^L EI*v''^2 dx = \frac{1}{2}\int_0^L q^T*B^T(x)*EI*B(x)*q dx = \frac{1}{2}q_e^T*\underline{\underline{K_e}}*q_e \tag{14}$$

$$T_e = \frac{1}{2}\int_0^L \rho*A*\dot{v}^2 dx = \frac{1}{2}\int_0^L q^T*N^T(x)*\rho*A*N(x)*q = \frac{1}{2}\dot{q_e}^T*\underline{\underline{M_e}}*\dot{q_e} \tag{15}$$

4.1 Equivalent nodal forces and examples, see course handwritten exercice

As an example, we calculate the first component of the vector of equivalent nodal forces **F** due to a trapezoidal load with intensities *p1* and *p2* .

```
%%
% *Equivalent nodal forces*
%
% As an example, we calculate the first component of the vector ...
    of equivalent
% nodal forces  *F* due to a trapezoidal load with intensities  ...
    _p1_ and  _p2_
% .

syms z
syms L
L=1;
N_1 = 1 - 3 * z^2 + 2 * z^3
figure
fplot(N_1, [0, 1])

N_3 = 3 * z^2 - 2 * z^3
```

```
figure
fplot(N_3, [0, 1])
%%
%%
N_2 = L * z * (z - 1)^2
figure
fplot(N_2, [0, 1])
N_4=-L*(z^2-z^3);
figure
fplot(N_4, [0, 1])
%%
% Works in real coordinate
%%

N_1 = subs(N_1, z, X/L)
N_2 = subs(N_2, z, X/L)
N_3 = subs(N_3, z, X/L)
N_4 = subs(N_4, z, X/L)

int(N_1, X)

int(N_1, X, sym(0), L)

N = [N_1; N_2; N_3; N_4]

int(N, X)

syms Feq
Feq == symhold('p1*int(N, X, 0, L) + (p2 - p1)*int((X*N)/L, X, ...
    0, L)')
%%
% Finally
%%
syms p1 p2
Feq = p1 * int(N, X, sym(0), L) + (p2 - p1) * int((X * N)/L, X, ...
    sym(0), L)

simplify(Feq)
%%
% Special case were p1=p2=p
%%
syms p
Feq = subs(subs(Feq, p1, p), p2, p)
%%
%  That's all
```

Now, we should work in real coordinate, and obtain the results :

```

N_1 = subs(N_1, z, X/L)
```

$\mathrm{N}_1 = 2X^3 - 3X^2 + 1$

```
N_2 = subs(N_2, z, X/L)
```

$\mathrm{N}_2 = X(X-1)^2$

```
N_3 = subs(N_3, z, X/L)
```

$\mathrm{N}_3 = 3X^2 - 2X^3$

```
N_4 = subs(N_4, z, X/L)
```

$\mathrm{N}_4 = X^3 - X^2$

```

int(N_1, X)
```

$$\mathrm{ans} = \frac{X^4}{2} - X^3 + X$$

```

int(N_1, X, sym(0), L)
```

$$\mathrm{ans} = \frac{1}{2}$$

```

N = [N_1; N_2; N_3; N_4]
```

$$\mathrm{N} = \begin{pmatrix} 2X^3 - 3X^2 + 1 \\ X(X-1)^2 \\ 3X^2 - 2X^3 \\ X^3 - X^2 \end{pmatrix}$$

```

int(N, X)
```

$$\mathrm{ans} = \begin{pmatrix} \frac{X^4}{2} - X^3 + X \\ \frac{X^2\left(3X^2-8X+6\right)}{12} \\ -\frac{X^3(X-2)}{2} \\ \frac{X^3(3X-4)}{12} \end{pmatrix}$$

```

syms Feq
Feq == symhold('p1*int(N, X, 0, L) + (p2 - p1)*int((X*N)/L, X, ...
    0, L)')
```

$$\mathrm{ans} = \mathrm{Feq} = p_1 \int N \mathrm{d}X + (p_2 - p_1) \int \frac{XN}{L} \mathrm{d}X$$

Finally,

```
syms p1 p2
Feq = p1 * int(N, X, sym(0), L) + (p2 - p1) * int((X * N)/L, X, ...
    sym(0), L)
```

$$\mathrm{Feq} = \begin{pmatrix} \frac{7p_1}{20} + \frac{3p_2}{20} \\ \frac{p_1}{20} + \frac{p_2}{30} \\ \frac{3p_1}{20} + \frac{7p_2}{20} \\ -\frac{p_1}{30} - \frac{p_2}{20} \end{pmatrix}$$

```

simplify(Feq)
```

$$\mathrm{ans} = \begin{pmatrix} \frac{7p_1}{20} + \frac{3p_2}{20} \\ \frac{p_1}{20} + \frac{p_2}{30} \\ \frac{3p_1}{20} + \frac{7p_2}{20} \\ -\frac{p_1}{30} - \frac{p_2}{20} \end{pmatrix}$$

Try to solbe a special case were p1=p2=p

```
syms p
Feq = subs(subs(Feq, p1, p), p2, p)
```

$$\mathrm{Feq} = \begin{pmatrix} \frac{p}{2} \\ \frac{p}{12} \\ \frac{p}{2} \\ -\frac{p}{12} \end{pmatrix}$$

That's all!

5 Gauss Integration

In finite element analysis, the element stiffness matrix is obtained by evaluating an integral. For line element (also termed 1-D elements, like rod, truss, beam, frame), we integrate over the length of the element, since the integration over the cross section has already been carried out by using cross section properties like the area A and the second moment of inertia I . For 2D elements, the integral is over the area of the element, for volume elements we integrate over the volume. In finite element analysis, the integrand usually becomes quite complicated and it is not possible to solve the integrals in closed form. We have to use numerical procedures in these situations. In finite element calculations, simple numerical integration schemes such as the trapezoidal rule or Simpson's formula do not work very well. The Gauss numerical integration scheme or Gauss quadrature, has becomes the standard tool to calculate element stiffness matrices. We derive the Gauss formula for one-dimensional integrals in the following section and then easily extend it to two and three-dimensional integrals.

6 plane stress and strain

7 Deriving differential equation

8 The 4-node membrane element

9 Patran/Nastran/What else ?

NASA STRuctural ANalysis, founded in the 1960s, marketed in 1969 (COSMIC) is the most widely used EF program in the aerospace industry. MSC-Nastran (Macneal Swendler Corp.) is the most popular version. The software will create the stiffness matrix K, the mass matrix M from user input (E, nu, rho, formulation and type of elements - bar -beam-plate-shell, plus thicknesses). Then Nastran will launch the resolution of the mechanical problem (inversion of K in the static case, eigenvalue problem for modal / buckling analysis).

PATRAN : pre and post-processor of MSC.Software for NASTRAN (similar to a CAD program).
NASTRAN : calculation code to solve problems by the finite element method. NASTRAN is not a graphical tool, to perform a calculation it is necessary to create an ASCII data file which describes the problem = it is the role of Patran.

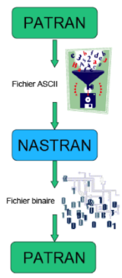

Fig. 8: Chaine de calcul patran/nastran .

10 Hello Doc ?

First, the Linear Static User Guide is a wealth of information : Type of elements, materials, modeling guides, Model verification, organization of a Nastran dataset ...

You can find relevant information about : Nastran : MSC.Nastran 2016 - Quick Reference Guide - MSC.Software that can be downloaded : Type on google "MSC nastran 2016 quick reference guide 2016". You will find it on the LMS on the SM201 course page, along with a number of other official supports of the 2016 version. Regarding the use of Patran, you can online help directly in the software .

MSC does not provide information on the design of its elements, you can find documentation on the more theoretical part :

— «Modélisation des structures – Calcul par éléments finis», Jean-Charles Craveur, Dunod, 2e édition, 2001.

- «Analyse des structures par éléments finis» de J.-f. Imbert, 1991, Cépaduès, SUP'AERO
- "The Finite Element Method third edition", O.C. Zienkiewitch, McGraw-Hill, London, 1977
- "Concepts and Applications of Finite Element Analysis", Robert D. Cook and al., , John Wiley and Sons, Inc., 2001.

To finish on internet, we can find a number of examples Nastran, Patran on the site MSC.

11 Units system ?

The FE software does not have a predefined unit system. They are universal. It is the user who defines his system of units and ensures its consistency. A consistent system defines the units of all properties required for the model : Length, Area, Volume, Section Moment Force, Moment, Acceleration, Mass, Density Young Modulus, Constraint, etc. In Nastran there are no default units. It is up to the user to always work with textbf consistent units : Patran only works with numbers!!! For structural mechanics problems the user chooses the units of length and force. For problems where time intervenes, it will have to choose the unit of time.

SI Units : Lenght -(meters), Force - (Newtons) Mass (Kg)
Then one will analyze constraints in Pa and the density must be specified in Kg / m^3 to obtain s^-1 Hz

Lenght-(mm), Force - (Newtons) Mass (Kg)
Then we will analyze constraints in MPa and the density must be specified in t / mm^3 to obtain s^-1 or Hz

12 Types of analysis

Each type of analysis available is called a "solution sequence" (SOL). The most famous "SOL" are :

- **101 - Linear Static**
- **103 - Modal**
- **105 - Linear Buckling**
- 106 - Static nonlinear
- 108 - Frequency response (direct)
- 109 - Transient response (direct)
- 111 - Frequency response (modal)
- 112 - Transient response (modal)
- 200 - Sensitivity and design analysis
- 400 - The new BIG solver (implicit)

Those used in this course are in bold.

13 Let's start...

- Copy the Patran and Nastran shortcuts to the desktop.
- Create a working folder on the local disk (textbf not on the network)
- Specify the path of your working folder to Patran when you create the 1st template.
- Sotcker all templates to use in this directory

All I/O files generated by Patran / Nastran will be stored in this working directory.

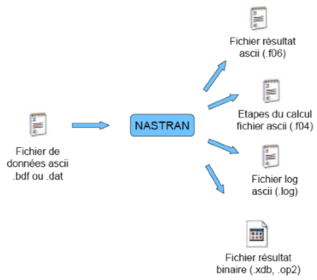

Fig. 9: Fichiers générés .

.f06
Results in ascii format. Also indicates errors and warnings, always look at it (look for the words "fatal" and "warning" in this file).

.f04
Resumes the steps of resolution (matrix assembly, elimination Blocked d.d.l, ...) with their execution time.

.log
Summarizes the configuration of the calculation (paths of files and executables, configuration of machine, ...)

.xdb, .op2
Results files in binary format. The .xdb and .op2 are readable by Patran But we can also create op2 readable by IDEAS, ...

.ses, .jou
All The nastran / patran commands are stored. This is the equivalent of macro under excel for example.

14 Step by step

CAD on Patran

For simple models it is entirely possible to create entirely the model under patran. It is then enough to follow from left to right the icons of the horizontal menu.

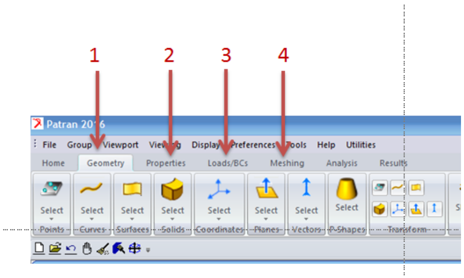

Fig. 10: Pre-treatment steps on Patran .

— Step 1 : Creating the geometry : point, curve, surface, set of axis

— Step 2 : Creation of the material and physic properties :
Plate thickness, beam section, and assignment to geometric entities (line, surface, ..)

— Step 3 : Definition of the boundary conditions in movement and loading
Creation then assignment to geometric entities which will be the supports of the nodes and elements on which these BCs will bear. It is possible to create different load cases from different BCs.

— Step 4 : Creation of the mesh :
Control of the mesh by the Meshseeds definition, then creation of the mesh with association to the physical properties and materials desired. Do not forget to do equivalence to ensure the correct junction of the meshes between the different entities that have been meshed.

Starting the computation

— Launching the computation.

In the section ≪**Analysis**≫ :
`analysis > entire model > full run` The following topics are used for :
* Translation parameters : specific parameters of the calculation, choice on the creation of the file (long or short formats, what output files, ...)

* Typical solution : choice of solution type (static, buckling, or modal analysis)
* Direct text input : section to add specific commands (options for "home" solutions for example)
* Subcases : choice of what we will ask in computation results for each subcase (constraints, displacements, efforts, ...)
* Subcase select : choice of load cases to be used for the calculation

— Checking

In the compute directory look for the NASTRAN calculation files and open the file `.f06`
Search in this file the term "**FATAL**", if it is present in the file : the calculation does not pass it needs to correct the model according to the information contained in this file
* Also search for "**warning**", it may indicate problems when calculating
* Search in file "**epsilon**", there is a residue for the calculation, if the value is too high (greater than 1e-4) it can be assumed that there is a problem in the mesh and Must be analyzed with great caution.

* Search the term **SPCFORCE RESULTANT** : one finds there the sum of the reactions of support, it is necessary to find the sum of the forces applied. Always verify that the sum of the forces applied is equal to the sum of the forces measured at the boundary conditions (mechanical equilibrium).
* Request the **OLOADs** Result (load applied) and the **SPCFORCEs** Result (Reaction to embedding points) :
If all these points are validated, you can look at the results with confidence.

— Postprocessing

Before the post-processing under Patran we must reread the result file generated by Nastran (in our case the .xdb file) : * In the "Analysis" section : action "attach xdb" By default, Patran averaged the constraints between two neighboring elements to create a continuous pattern. It is a "trap" because this visualization can smooth local peaks of constraints and lead to an underestimation of the value in an element. To avoid this, you should choose not to use the "**Fringe**" option in an option and to see the values at the integration points.

Finally, in the "**Results**" section :
* **Quick Plot** : fast isovate plot (suitable for displacements but limited for constraints)
* **Deformation** : view the distorted

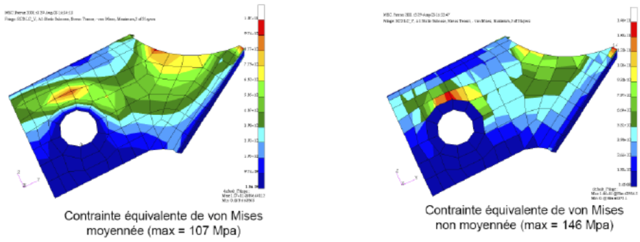

Fig. 11: Fringe ? .

* **Fringe** : a more advanced isov value plot (useful for viewing non-averaged constraints)
* **Marker** : plotting forces in the form of tensors (arrows)
* **Cursor** : Lets you know either a Scalar, Vector or Tensor value
* **Graphs** : Allows you to draw a nodal or elementary result along a path
* **Report** : display results in a file for use in excel for example.

Case of triangles : The calculations made with the linear interpolation triangle elements must lead to the following remark : constraints are constant in a triangle element. This observation is quite normal : the stress is calculated from the interpolation gradient, and for the triangles with linear interpolation the interpolation gradient is constant :

$N(x, y) = [1, -x, -y, x, y]$

$\nabla N(x, y) = [-1, 1, 0; -1, 0, 1]$

15 Finally, some tips

* Pay attention to the use of triangles (and tetrahedrons for 3D meshes) which allow only a uniform field of stresses in the element (cf. above). In sensitive areas it is necessary to mesh finer
* Always take a critical look at the results of a calculation, they only reflect the instructions you gave to the calculation software

Patran and Nastran allow to display on the model of the structure the maps of visualization of the fields of constraints. The maps giving the variation of the stress components (sx, sy ...) give only a partial image and do not allow any interpretation on the risks of exceeding Re. Moreover, they depend on the reference point in which they are Expressed. The equivalent constraint of Von Mises is a combination of these components and does not depend on the benchmark.

Only the image of the field of the equivalent constraints of Von Misès makes it possible to clearly visualize the zone (s) subjected to the risk of plastification. Be careful, however, the Von Mises constraint does not apply to composite materials.

The display of stresses or deformations requires special attention insofar as Patran proposes various options, in particular for calculating averages.

Patran uses a number of options by "**default**" which can show values different from those calculated by the solver. If we consider that the values provided by the solver are the reference values, it is good to know how to find these values in Patran. Patran's online documentation can help.

`Book: Results Postprocessing > Chapter: 13 Numerical Methods >`