

## 0.1 Elasticita montáže

### 0.1.1 Rovnice pro rovnováhu, Hookův zákon a Lamého rovnice

Pohybová rovnice pro pevnou látku, zapsaná v indexové notaci, je:<sup>1</sup>

$$\rho u_{r,tt} = \sigma_{rs,s} + \rho f_r$$

kde  $u_r$  označuje *posunutí*, index  $r = 1..3$ , resp. do počtu dimenzí, označuje složky souřadnic,  $\rho$  hustotu,  $\sigma_{rs}$  tenzor napětí (čili vnitřní síly působící na jednotku plochy,  $[\sigma_{rs}] = \text{Pa}$ ),  $f_r$  síly působící v objemu tělesa (vyjádřené na jednotku objemu,  $[f_r] = \text{N/m}^3$ ), například by se mohlo jednat o gravitaci.

Vynecháním časových derivací získáme *rovnici pro rovnováhu*:

$$\sigma_{rs,s} + \rho f_r = 0 \quad \text{v } \Omega, \quad (1)$$

která musí platit v celém objemu  $\Omega$  tělesa.

Tuto parciální diferenciální rovnici musíme doplnit o konkrétní *okrajovou podmínku*, aby řešení bylo jednoznačné:

$$\sigma_{rs} n_s = g_r \quad \text{na } \Gamma, \quad (2)$$

platicí na hranici  $\Gamma$  objemu  $\Omega$ , kde působí povrchové síly  $g_r$  (vyjádřené na jednotku plochy,  $[g_r] = \text{Pa}$ );  $n_s$  zde označuje normálu k ploše. Pro volný povrch by bylo samozřejmě  $g_r = 0$ .

*Hookův zákon* v nejjednodušší podobě je  $\sigma = E\varepsilon$ , kde  $\sigma$  označuje napětí (sílu na jednotku plochy, resp. průřezu),  $E$  Youngův modul pružnosti (též elastický modul,  $[E] = \text{Pa}$ ) a  $\varepsilon$  je deformace (bezrozměrná, změna délky lomená délkou). Pro popis deformací ve třech rozměrech však potřebujeme jeho složitější podobu:

$$\sigma_{rs} = \lambda \varepsilon_{qq} \delta_{rs} + 2\mu \varepsilon_{rs}, \quad (3)$$

kde  $\delta_{rs}$  je Kroneckerovo delta (pro  $r = s$  je  $\delta_{rr} = 1$  jinak  $\delta_{rs} = 0$ ) a  $\lambda$  a  $\mu$  jsou Laméovy konstanty, mající určitý vztah k modulu pružnosti  $E$  a Poissonově poměru  $\nu$ , jež jsou obvykle udáványmi charakteristikami materiálů:

$$\mu = \frac{E}{2(1+\nu)}, \quad \lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}.$$

<sup>1</sup> Indexy za čárkou označují v této notaci derivace  $_{,t} \equiv \frac{\partial}{\partial t}$ ,  $_{,s} \equiv \frac{\partial}{\partial x_s}$ , při totožných indexech platí sumační pravidlo  $_{,qq} \equiv \sum_q$ .

Tenzor deformací je zaveden jako:

$$\varepsilon_{rs} \equiv \frac{1}{2}(u_{r,s} + u_{s,r}).$$

Dosažením Hookova zákona do rovnice pro rovnováhu obdržíme *Laméovu rovnici*, čili parciální diferenciální rovnici pro posunutí  $u_r$  („skrytá“ v tenzoru deformace):

$$(\lambda \varepsilon_{qq} \delta_{rs})_{,s} + (2\mu \varepsilon_{rs})_{,s} + \rho f_r = 0, \quad (4)$$

neboli:

$$(\lambda u_{q,q} \delta_{rs})_{,s} + [\mu(u_{r,s} + u_{s,r})]_{,s} + \rho f_r = 0.$$

Tuto rovnici můžeme též zapsat abstraktněji pomocí diferenciálního operátoru:

$$\mathcal{L}(u_r) = 0. \quad (5)$$

### 0.1.2 Slabá formulace problému

Namísto (zatím neznámého) řešení  $u_r$  rovnice (5) hledáme jeho aproximaci (Langtangen 2003):

$$u \doteq \hat{u}_r = \sum_{j=1}^M u_{j,r} N_{j,r},$$

kde  $N_{j,r}$  jsou bázové funkce a  $u_{j,r}$  hledané koeficienty;  $M$  je počet bodů sítě, kterou triangulujeme objem  $\Omega$ . Pro aproximaci však nebude rovnice (5) platit přesně, nýbrž:

$$\mathcal{L}(\hat{u}_r) \neq 0.$$

Požadujeme alespoň, aby součet všech reziduí byl nulový:

$$\int_{\Omega} \sum_{r=1}^3 \mathcal{L}(\hat{u}_r) W_{ir} d\Omega = 0,$$

kde  $W_{ir}$  jsou vhodné váhy, nazývané též *testovací funkce*. Tato rovnice se nazývá *slabá formulace*. Integrace přes objem  $\Omega$  zde odpovídá sumaci  $\sum_{i=1}^M$ . Sumy přes body sítě  $i$  a přes dimenze  $r$  lze sdružit do jedné, abychom tu druhou nemuseli stále vypisovat.

V Galerkinově metodě bereme za váhy přímo bázové funkce,  $W_{ir} \equiv N_{ir}$ , takže:

$$\int_{\Omega} \mathcal{L}(\hat{u}_r) N_{ir} d\Omega = 0.$$

Pro naši rovnici rovnováhy (1), máme:

$$\int_{\Omega} \sigma_{rs,s} N_{ir} d\Omega + \int_{\Omega} \rho f_r N_{ir} d\Omega = 0.$$

Pro její úpravu využijeme Greenovo lemma:

$$\int_{\Omega} \sigma_{rs,s} N_{ir} d\Omega = - \int_{\Omega} \sigma_{rs} N_{ir,s} d\Omega + \int_{\Gamma} \overbrace{\sigma_{rs} n_s}^{g_r} d\Gamma$$

a zároveň dosadíme z Hookova zákona (3) a šikovně z okrajové podmínky (2):

$$- \int_{\Omega} (\lambda \varepsilon_{qq} \delta_{rs} + 2\mu \varepsilon_{rs}) N_{ir,s} d\Omega + \int_{\Gamma} g_r N_{ir} d\Gamma + \int_{\Omega} \rho f_r N_{ir} d\Omega = 0, \quad (6)$$

čímž získáme základní rovnici pro metodu konečných prvků.

### 0.1.3 Implementace v programu FreeFem++

V programu FreeFem++ (Hecht 2012) je implementace velmi snadná, protože umožňuje symbolický zápis integrálů přes  $\Omega$  a  $\Gamma$  (viz níže). Funkce  $\hat{u}_r$  jsou v programu značeny  $u_1, u_2, u_3$ , testovací funkce  $N_{ir}$  pak  $v_1, v_2, v_3$ . Aby bylo zápisu dobře rozumět, některé členy rovnice (6) rozepíšeme:

$$\begin{aligned} \sum_q \varepsilon_{qq} &= \sum_q \frac{1}{2} \left( \frac{\partial u_q}{\partial x_q} + \frac{\partial u_q}{\partial x_q} \right) = \frac{\partial u_1}{\partial x} + \frac{\partial u_2}{\partial y} + \frac{\partial u_3}{\partial z}, \\ \sum_r \sum_s \delta_{rs} N_{ir,s} &= \sum_r \frac{\partial N_{ir}}{\partial x_r} = \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v_3}{\partial z}, \\ \sum_r \sum_s \varepsilon_{rs} N_{ir,s} &= \sum_r \sum_s \frac{1}{2} \left( \frac{\partial u_r}{\partial x_s} + \frac{\partial u_s}{\partial x_r} \right) \frac{\partial N_{ir}}{\partial x_s} = \\ &= \frac{1}{2} \left( \frac{\partial u_1}{\partial x_1} + \frac{\partial u_1}{\partial x_1} \right) \frac{\partial v_1}{\partial x_1} + \frac{1}{2} \left( \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) \frac{\partial v_1}{\partial x_2} + \frac{1}{2} \left( \frac{\partial u_1}{\partial x_3} + \frac{\partial u_3}{\partial x_1} \right) \frac{\partial v_1}{\partial x_3} + \\ &+ \frac{1}{2} \left( \frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} \right) \frac{\partial v_2}{\partial x_1} + \frac{1}{2} \left( \frac{\partial u_2}{\partial x_2} + \frac{\partial u_2}{\partial x_2} \right) \frac{\partial v_2}{\partial x_2} + \frac{1}{2} \left( \frac{\partial u_2}{\partial x_3} + \frac{\partial u_3}{\partial x_2} \right) \frac{\partial v_2}{\partial x_3} + \\ &+ \frac{1}{2} \left( \frac{\partial u_3}{\partial x_1} + \frac{\partial u_1}{\partial x_3} \right) \frac{\partial v_3}{\partial x_1} + \frac{1}{2} \left( \frac{\partial u_3}{\partial x_2} + \frac{\partial u_2}{\partial x_3} \right) \frac{\partial v_3}{\partial x_2} + \frac{1}{2} \left( \frac{\partial u_3}{\partial x_3} + \frac{\partial u_3}{\partial x_3} \right) \frac{\partial v_3}{\partial x_3} = \\ &= \left[ \frac{\partial u_1}{\partial x}, \frac{\partial u_2}{\partial y}, \frac{\partial u_3}{\partial z}, \frac{1}{\sqrt{2}} \left( \frac{\partial u_2}{\partial z} + \frac{\partial u_3}{\partial y} \right), \frac{1}{\sqrt{2}} \left( \frac{\partial u_1}{\partial z} + \frac{\partial u_3}{\partial x} \right), \frac{1}{\sqrt{2}} \left( \frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \right] \cdot \\ &\cdot \left[ \frac{\partial v_1}{\partial x}, \frac{\partial v_2}{\partial y}, \frac{\partial v_3}{\partial z}, \frac{1}{\sqrt{2}} \left( \frac{\partial v_2}{\partial z} + \frac{\partial v_3}{\partial y} \right), \frac{1}{\sqrt{2}} \left( \frac{\partial v_1}{\partial z} + \frac{\partial v_3}{\partial x} \right), \frac{1}{\sqrt{2}} \left( \frac{\partial v_1}{\partial y} + \frac{\partial v_2}{\partial x} \right) \right] \end{aligned}$$

---

```

// read mesh from file

load "msh3";
load "medit";

mesh3 Th = readmesh3("cube.1.msh");

real a = 0.1; // cross section dimensions [m]
real b = 0.15;

/*****/

// material parameters (and units)

real MPa = 1.e6; // Pa
real ag = 9.81; // gravitational acceleration [m/s^2]

real E = 21.5e4 * MPa; // Young modulus [Pa], steel
real nu = 0.29; // Poisson ratio [], steel
real rho = 7800.; // density [kg/m^3], steel
real f = rho * ag; // volumetric force [Pa/m = N/m^3], i.e. gravity
real m = 100.; // mass attached at the end of the bend [kg]
real g = m*ag / (a*b); // surface force [Pa = N/m^2]

real mu = E/(2*(1+nu)); // Lamé's constants
real lambda = E*nu/((1+nu)*(1-2*nu));

/*****/

// finite elements space (vectorial)

fespace Vh(Th, [P1,P1,P1]);
Vh [u1,u2,u3], [v1,v2,v3];

// macros simplifying the problem

real sqrt2 = sqrt(2.);
macro epsilon(u1,u2,u3) [ dx(u1), dy(u2), dz(u3), (dz(u2)+dy(u3))/sqrt2,
  (dz(u1)+dx(u3))/sqrt2, (dy(u1)+dx(u2))/sqrt2 ] // end-of-macro
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // end-of-macro

/*****/

// weak formulation of the problem and solution

solve Lamé([u1,u2,u3], [v1,v2,v3]) =
  int3d(Th)(
    lambda * div(u1,u2,u3) * div(v1,v2,v3) // divergence term
    + 2.*mu*( epsilon(u1,u2,u3) '* epsilon(v1,v2,v3) ) // ' ) // vectorial *
  )
- int3d(Th) (-f*v3) // volumetric forces
- int2d(Th,3) (-g*v3) // surface forces
+ on(1, u1=0, u2=0, u3=0); // Dirichlet boundary condition

```

```

/*****/

// save output and show maximum displacement

savemesh(Th, "output.sol.mesh");
savesol("output.sol", Th, [u1,u2,u3], order=1);

real umax = max(abs(u1[] .min), u1[] .max);
cout << "\numax = " << umax << " m = " << umax*1.e6 << " microns\n" << endl;

/*****/

// display amplified mesh

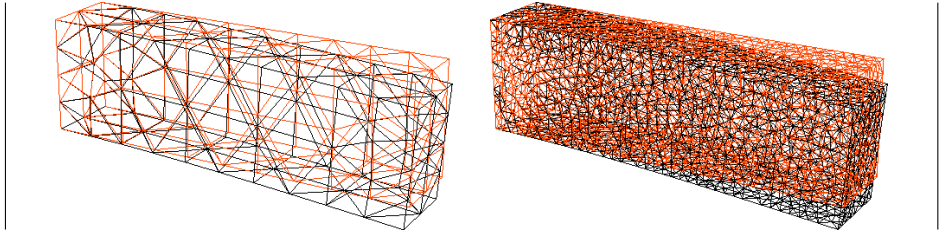
real coef= 0.2*b/umax;
mesh3 Thamplified = movemesh3(Th, transfo = [x+u1*coef, y+u2*coef, z+u3*coef]);

int[int] ref0 = [1,0, 2,0, 3,0, 4,0, 5,0, 6,0]; // change labels
int[int] ref2 = [1,2, 2,2, 3,2, 4,2, 5,2, 6,2];
Th = change(Th, label=ref2);
Thamplified = change(Thamplified, label=ref0);

plot(Th, Thamplified, wait=1, cmm = "amplification = " + coef);

```

Program lze spustit jednoduše z příkazové řádky FreeFem++ `lame.edp`. Příklad výsledku výpočtu je na obr. 1.



**Obr. 1** — Porovnání nedeformované a deformované sítě, kde posunutí spočtená programem FreeFem++ byla zvětšena faktorem asi 4000. Maximální posunutí dosahuje  $u_{\max} = 7,7 \mu\text{m}$ . Vlevo síť hrubá a vpravo jemná, s počtem vrcholů  $n_{\text{vertices}} = 88$ , resp. 3018.

### 0.1.4 Vytvoření sítě programem Tetgen

Objem, vymezený povrchovou sítí ve formátu MESH, je nutné triangulovat čtyřstěny. Pomocí programu `tetgen` (Si 2006) to můžeme provést jako: `tetgen -Vgpnq1.2a1.e-6 cube.mesh > tetgen.log`, kde parametr  $q = 1,2$  omezuje poměr stran čtyřstěnu a  $a = 10^{-6}$  jejich maximální objem. Pro import do programu FreeFem++ se osvědčilo použít formát MSH: `./nodeeleface2msh.awk cube.1.node cube.1.ele cube.1.face > cube.1.msh`.

### 0.1.5 Testovací příklad s jednoduchým nosníkem a závažím

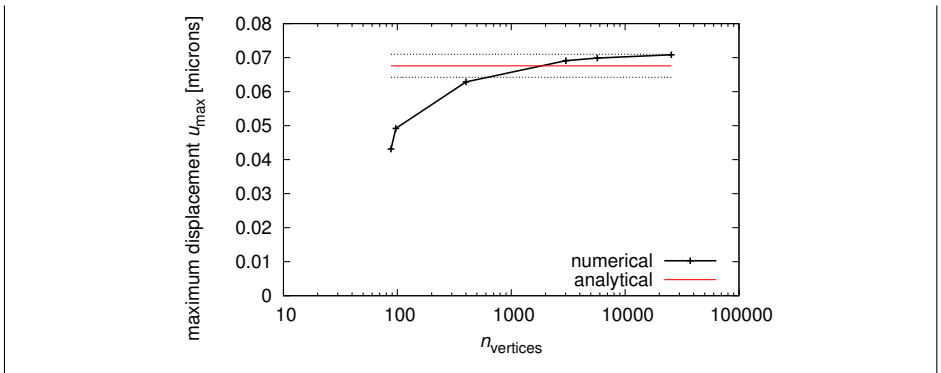
Analytické řešení lze použít pro kontrolu implementace. Pro nosník na jednom konci upevněný a na druhém konci zatížený, je výchylka konce:

$$x_l = \frac{Fl^3}{3EI_a},$$

kde  $F = ma_g$  je síla působící svisle na konec nosníku,  $m$  hmotnost závaží,  $a_g$  tíhové zrychlení,  $l$  délka nosníku,  $E$  modul pružnosti a  $I_a$  kvadratický moment průřezu. Pro obdélníkový průřez o rozměrech  $a$ ,  $b$  (měřených vodorovně a svisle) a pro osu procházející středem je:

$$I_a = \frac{ab^3}{12}.$$

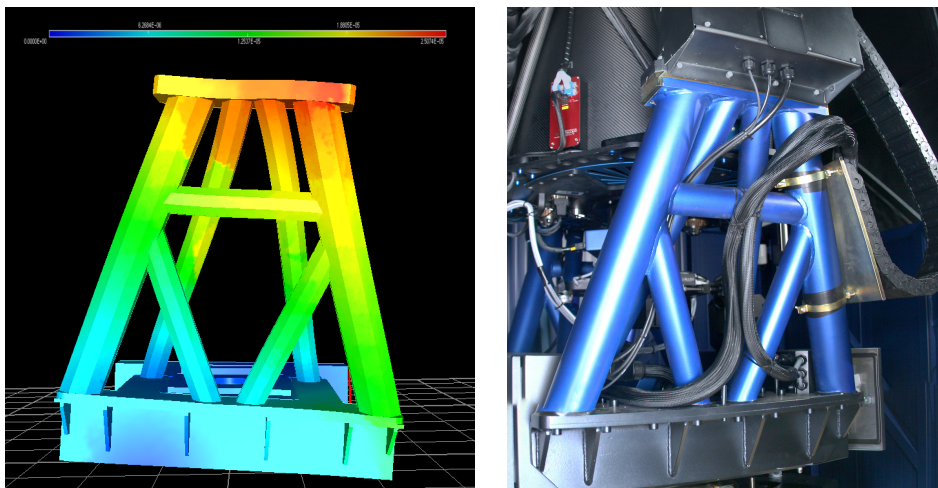
Rozdíly mezi analytickým a numerickým modelem dosahují méně než 5% pro počet vrcholů  $n_{\text{vertices}} \gtrsim 10^3$ . Mohou být dány jednak diskretizací numerického modelu a jednak omezeními analytického modelu.



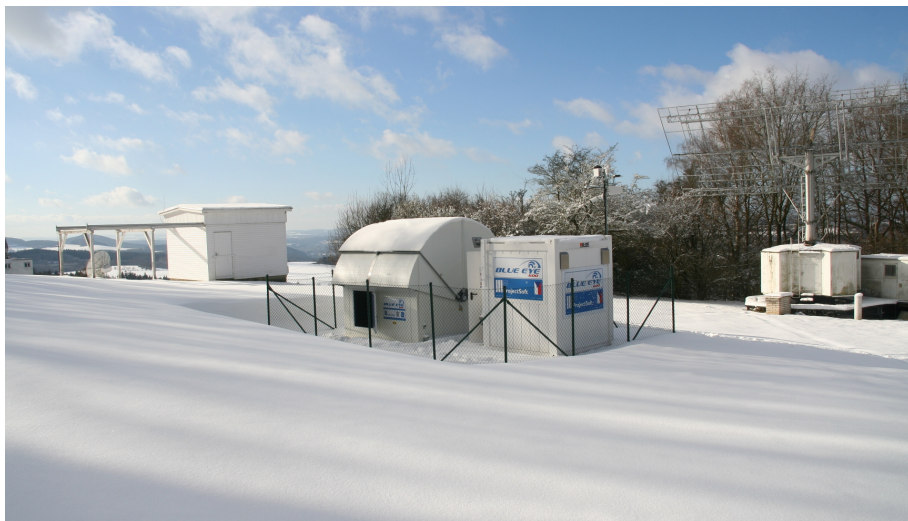
**Obr. 2** — Porovnání numerického a analytického modelu deformace jednoduchého nosníku. Hodnoty parametrů byly v obou případech zvoleny následovně:  $l = 0,5 \text{ m}$ ,  $a = 0,1 \text{ m}$ ,  $b = 0,15 \text{ m}$ ,  $m = 1 \text{ kg}$ ,  $a_g = 9,81 \text{ m} \cdot \text{s}^{-2}$ ,  $E = 21,5 \cdot 10^4 \text{ MPa}$ ,  $\nu = 0,29$ .

### 0.1.6 Výpočet deformace vidlice

Pro vidlici (montáž) lze postupovat obdobně. Před exportem z programu CAD je vhodné konstrukci zjednodušit. Pak stačí konvertovat příslušnou síť do formátu MESH a přitom zadat působení silou určitého směru na určitou plochu (označenou pomocí sloupce label v síti). Výsledek triangulace a výpočtu je ukázán na obr. 3; celou observatoř pak vidíme na obr. 4.



**Obr. 3** — Spočtené deformace vidlice, respektive části alt–azimutální montáže, při působení síly 1300 N na horní část, kde budou umístěny domečky pro momentové motory. Maximální posunutí činí 20  $\mu\text{m}$ . Vpravo výsledná podoba montáže.



**Obr. 4** — Autonomní oservatoř BlueEye 600, vyrobená firmou Projectsoft v rámci grantu Technologické agentury ČR, byla umístěna v Ondřejově.





# Literatura

Učebnice

Reference