MDF (continuare): Model 1D, distribuit, al unui microcomutator acţionat electrostatic.

Analiza în regim staţionar elastostatic cuplat cu electrostatic.

Prof.dr.ing. Gabriela Ciuprina

Universitatea "Politehnica" Bucureşti, Facultatea de Inginerie Electrică

Suport didactic pentru disciplina *Algoritmi Numerici*, 2017-2018

Cuprins

Descrierea problemei

1. Înainte de a parcurge aceste slide-uri, trebuie să revedeţi contextul descris la studiul de caz prezentat în http://an.lmn.pub.ro/slides2017/06b_AN.pdf
   Acolo armătura superioară a condensatorului era rigidă, cu posibilitatea de a se mişca doar prin translaţie. Modelul 1D era **concentrat** într-un punct.
2. Acum, armătura superioară se consideră elastică, cu următorii parametri consideraţi cunoscuţi:
   - geometrie: lungime l, lătime w, înălţime h;
   - material: modul de elasticitate longitudinal E;

   În plus, armătura superioară se consideră încastrată la ambele capete şi are tensiuni iniţiale mecanice distribuite uniform şi constante pe parcursul deformării, descrise de mărimea locală S.

Descrierea problemei

La aplicarea unei tensiuni electrice între armăturile unui astfel de condensator, armătura superioară este atrasă de armătura fixă, dar ea se deformează, capetele ei fiind fixate.

Vom rezolva model 1D **distribuit** de-a lungul armăturii folosind:

1. MDF pentru discretizarea PDE $\Rightarrow$ sistem de ecuaţii algebrice neliniare;
2. metoda Newton pentru rezolvarea sistemului algebric asamblat.

Formularea problemei

Ecuaţiile care descriu fenomenele:

- Structural, elastostatic (MEC) - Ecuaţia Euler-Bernoulli

$$EI\frac{\partial^4 u}{\partial x^4} - S\frac{\partial^2 u}{\partial x^2} = F_{\text{ES}} \tag{1}$$

unde $I = wh^3/12$

- Electrostatic (ES) :

$$F_{\text{ES}} = -\frac{\varepsilon_0 w v^2}{2u^2} \tag{2}$$

Vom rezolva problema făcând un cuplaj strâns între fenomene.

Formularea problemei

Ecuaţiile care descriu fenomenele:

- (MEC) + (ES)

$$EI\frac{\partial^4 u}{\partial x^4} - S\frac{\partial^2 u}{\partial x^2} = -\frac{\varepsilon_0 w v^2}{2u^2} \tag{3}$$

şi rearanjat

- (MEC) + (ES)

$$u^2\left(S\frac{\partial^2 u}{\partial x^2} - EI\frac{\partial^4 u}{\partial x^4}\right) - \frac{\varepsilon_0 w v^2}{2} = 0 \tag{4}$$

unde $I = wh^3/12$.

Formularea problemei

Ecuaţii şi condiţii de frontieră

- (MEC) + (ES)

$$u^2\left(S\frac{\partial^2 u}{\partial x^2} - EI\frac{\partial^4 u}{\partial x^4}\right) - \frac{\varepsilon_0 w v^2}{2} = 0 \qquad (5)$$

unde $I = wh^3/12$.

- Condiţii la capetele domeniului

$$u(0) = u0 \qquad (6)$$

$$u(l) = u0 \qquad (7)$$

$$\frac{\mathrm{d}u}{\mathrm{d}x}(0) = 0 \qquad (8)$$

$$\frac{\mathrm{d}u}{\mathrm{d}x}(l) = 0 \qquad (9)$$

unde $u0$ este distanţa iniţială dintre armături.

Formularea problemei

Se dau:

- Geometria: l, w, h, $u0$;
- Materialele: (MEC): E, (ES): ε_0;
- Sursele: (MEC): S, (ES): v;
- Condiţiile de frontieră: (MEC) încastrare la ambele capete; (ES) între armătura superioară, mobilă şi cea inferioară, fixă se aplică o tensiune v.

Se cere:

- Deplasarea armăturii $u(x)$ pentru o tensiune v fixată, suficient de mică astfel încât comutatorul să nu se închidă;
- Analiza deplasarea armăturii $u(x)$ pentru o gamă de tensiuni v crescătoare, începând de la valoarea 0;
- Tensiunea minimă de acţionare V_{pi}.

Discretizarea domeniului

$$x \in [0, l]$$

n puncte interioare distribuite uniform

$$x_0 = 0 \quad x_0 = \Delta x \quad x_1 = 2\Delta x \quad \ldots \quad x_n = n\Delta x \quad x_{n+1} = l$$

$$\Delta x = \frac{l}{n+1}$$

Discretizarea ecuaţiei

Ecuaţii şi condiţii de frontieră

- (MEC) + (ES)

$$u^2\left(S\frac{\partial^2 u}{\partial x^2} - EI\frac{\partial^4 u}{\partial x^4}\right) - \frac{\varepsilon_0 w v^2}{2} = 0 \qquad (10)$$

unde $I = wh^3/12$.

- Condiţii la capetele domeniului

$$u(0) = g0 \qquad (11)$$

$$u(l) = g0 \qquad (12)$$

$$\frac{\mathrm{d}u}{\mathrm{d}x}(0) = 0 \qquad (13)$$

$$\frac{\mathrm{d}u}{\mathrm{d}x}(l) = 0 \qquad (14)$$

unde $g0 = u_0$ este distanţa iniţială (*gap*) dintre armături.

Derivata de ordin 2

Avem nevoie de formule de derivare numerică de ordinul 2 şi de ordinul 4 pentru reţele uniforme:

$$\frac{\mathrm{d}^2 f}{\mathrm{d}x^2} = \frac{1}{(\Delta x)^2}(f_{i-1} - 2f_i + f_{i+1}), \quad i = 1, n \tag{15}$$

În cazul nostru este convenabil să procedăm astfel:

$$\begin{aligned} \frac{\mathrm{d}^2 u}{\mathrm{d}x^2}(x_i) &= \frac{\mathrm{d}^2 (u - u_0)}{\mathrm{d}x^2}(x_i) = \\ &= \frac{1}{(\Delta x)^2}((u_{i-1} - u_0) - 2(u_i - u_0) + (u_{i+1} - u_0)), \\ & \qquad i = 1, n \end{aligned} \tag{16}$$

Acest artificiu ne permite să înlocuim cu uşurinţă condiţiile la capete, în această ecuaţie.

Derivata de ordin 2 + condiţii de frontieră

$i = 2, n-1$

$$\frac{\mathrm{d}^2 u_i}{\mathrm{d}x^2} = \frac{1}{(\Delta x)^2}\left((u_{i-1} - u_0) - 2(u_i - u_0) + (u_{i+1} - u_0)\right) \tag{17}$$

$i = 1$

$$\frac{\mathrm{d}^2 u_1}{\mathrm{d}x^2} = \frac{1}{(\Delta x)^2}\left(-2(u_1 - u_0) + (u_2 - u_0)\right) \tag{18}$$

$i = n$

$$\frac{\mathrm{d}^2 u_n}{\mathrm{d}x^2} = \frac{1}{(\Delta x)^2}\left((u_{n-1} - u_0) - 2(u_n - u_0)\right) \tag{19}$$

Derivata de ordin 2 + condiţii de frontieră

Scriere compactă

$$\mathbf{D2} = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & 0 & 0 & \cdots & 0 & 0 & 0 \\ & & \cdots & \cdots & \cdots & & & & \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & -2 \end{bmatrix} \quad \frac{\mathrm{d}^2\mathbf{u}}{\mathrm{d}\mathbf{x}^2} = \begin{bmatrix} \frac{\mathrm{d}^2\mathbf{u_1}}{\mathrm{d}\mathbf{x}^2} \\ \frac{\mathrm{d}^2\mathbf{u_2}}{\mathrm{d}\mathbf{x}^2} \\ \vdots \\ \frac{\mathrm{d}^2\mathbf{u_n}}{\mathrm{d}\mathbf{x}^2} \end{bmatrix} \tag{20}$$

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \quad \mathbf{u} - \mathbf{u_0} = \begin{bmatrix} u_1 - u_0 \\ u_2 - u_0 \\ \vdots \\ u_n - u_0 \end{bmatrix} \quad \Rightarrow \quad \frac{\mathrm{d}^2\mathbf{u}}{\mathrm{d}\mathbf{x}^2} = \mathbf{D2} * (\mathbf{u} - \mathbf{u_0}) \tag{21}$$

Derivata de ordin 4

Pentru reţele uniforme:

$$\frac{\mathrm{d}^4 f}{\mathrm{d}x^4} = \frac{1}{(\Delta x)^4}(f_{i-2} - 4f_{i-1} + 6f_i - 4f_{i+1} + f_{i+2}) \qquad (22)$$

Este convenabil să procedăm astfel:

$$\frac{\mathrm{d}^4 u}{\mathrm{d}x^4}(x_i) =$$

$$= \frac{1}{(\Delta x)^4}((u_{i-2} - u_0) - 4(u_{i-1} - u_0) + 6(u_i - u_0) - 4(u_{i+1} - u_0) + (u_{i+2} - u_0))$$

$$i = 2, n-1 \qquad (23)$$

Derivata de ordin 4 + condiţii de frontieră

Particularizăm formula pentru

- $i = 2$ şi impunem condiţia de frontieră $u(0) = u_0$
- $i = n$ şi impunem condiţia de frontieră $u(l) = u_0$
- $i = 1$ şi impunem condiţia de frontieră $du/dx(0) = 0 \Rightarrow u_{-1} = u_1$
- $i = n + 1$ şi impunem condiţia de frontieră $du/dx(l) = 0 \Rightarrow u_{n+2} = u_n$

Derivata de ordin 4 + condiţii de frontieră

i = 3,n-2

$$\frac{\mathrm{d}^4 u_i}{\mathrm{d}x^4} = \\ = \frac{1}{(\Delta x)^4}((u_{i-2}-u_0)-4(u_{i-1}-u_0)+6(u_i-u_0)-4(u_{i+1}-u_0)+(u_{i+2}-u_0)) \quad (24)$$

i = 2

$$\frac{\mathrm{d}^4 u_2}{\mathrm{d}x^4} = \frac{1}{(\Delta x)^4}(-4(u_1-u_0)+6(u_2-u_0)-4(u_3-u_0)+(u_4-u_0)) \quad (25)$$

i = n-1

$$\frac{\mathrm{d}^4 u_{n-1}}{\mathrm{d}x^4} = \frac{1}{(\Delta x)^4}((u_{n-3}-u_0)-4(u_{n-2}-u_0)+6(u_{n-1}-u_0)-4(u_n-u_0)) \quad (26)$$

Derivata de ordin 4 + condiţii de frontieră

i = 3,n-2

$$\frac{\mathrm{d}^4 u_i}{\mathrm{d}x^4} = \\ = \frac{1}{(\Delta x)^4}((u_{i-2}-u_0)-4(u_{i-1}-u_0)+6(u_i-u_0)-4(u_{i+1}-u_0)+(u_{i+2}-u_0)) \quad (27)$$

i = 1

$$\frac{\mathrm{d}^4 u_1}{\mathrm{d}x^4} = \frac{1}{(\Delta x)^4}(7(u_1-u_0)-4(u_2-u_0)+(u_3-u_0)) \quad (28)$$

i = n

$$\frac{\mathrm{d}^4 u_n}{\mathrm{d}x^4} = \frac{1}{(\Delta x)^4}((u_{n-2}-u_0)-4(u_{n-1}-u_0)+7(u_n-u_0) \quad (29)$$

Derivata de ordin 4 + condiţii de frontieră

Scriere compactă

$$\mathbf{D4} = \begin{bmatrix} 7 & -4 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ -4 & 6 & -4 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 1 & -4 & 6 & -4 & 1 & \cdots & 0 & 0 & 0 & 0 & 0 \\ & & \cdots & \cdots & \cdots & \cdots & \cdots & & & & \\ & & & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & & \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & -4 & 6 & -4 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & -4 & 6 & -4 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & -4 & 7 \end{bmatrix} \quad \frac{\mathrm{d}^4\mathbf{u}}{\mathrm{d}\mathbf{x}^4} = \begin{bmatrix} \frac{\mathrm{d}^4\mathbf{u_1}}{\mathrm{d}\mathbf{x}^4} \\ \frac{\mathrm{d}^4\mathbf{u_2}}{\mathrm{d}\mathbf{x}^4} \\ \vdots \\ \frac{\mathrm{d}^4\mathbf{u_n}}{\mathrm{d}\mathbf{x}^4} \end{bmatrix} \tag{30}$$

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \quad \mathbf{u} - \mathbf{u_0} = \begin{bmatrix} u_1 - u_0 \\ u_2 - u_0 \\ \vdots \\ u_n - u_0 \end{bmatrix} \quad \Rightarrow \quad \frac{\mathrm{d}^4\mathbf{u}}{\mathrm{d}\mathbf{x}^4} = \mathbf{D4} * (\mathbf{u} - \mathbf{u_0}) \tag{31}$$

Sistemul algebric neliniar obţinut

$$\mathbf{F}(\mathbf{u}) = \mathbf{0} \tag{32}$$

unde

$$\mathbf{F}(\mathbf{u}) = (\mathbf{u}.*\mathbf{u})*(S*\mathbf{D2}*(\mathbf{u}-\mathbf{u0})-E*I*\mathbf{D4}*(\mathbf{u}-\mathbf{u0}))-\frac{\varepsilon_0 w v^2}{2}*\mathbf{1} \tag{33}$$

unde

$$\mathbf{1} = \text{ones}(n,1) \qquad \mathbf{u}.*\mathbf{u} = \begin{bmatrix} u_1^2 \\ u_2^2 \\ \vdots \\ u_n^2 \end{bmatrix}$$

Ecuaţia obţinută este algebrică, neliniară

Metoda Newton

$$\mathbf{F}(\mathbf{u}) = \mathbf{0}$$

$$\mathbf{u}^{(n)} = \mathbf{u}^{(v)} - \left(\mathbf{F}'(\mathbf{u}^{(v)})\right)^{-1} \mathbf{F}(\mathbf{u}^{(v)})$$

Algoritm - la fiecare iteraţie

1. Se calculează corecţia $\mathbf{z}$ prin rezolvarea sistemului

$$\mathbf{F}'(\mathbf{u}^{(v)})\mathbf{z} = -\mathbf{F}(\mathbf{u}^{(v)})$$

2. Se actualizează soluţia

$$\mathbf{u}^{(n)} = \mathbf{u}^{(v)} + \mathbf{z}$$

Calculul Jacobianului

$$\mathbf{F}'(\mathbf{u}^{(\nu)}) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \cdots & \frac{\partial f_2}{\partial u_n} \\ \vdots & & & \\ \frac{\partial f_n}{\partial u_1} & \frac{\partial f_n}{\partial u_2} & \cdots & \frac{\partial f_n}{\partial u_n} \end{bmatrix} = ?$$

Implementare în Matlab

Codul (comentat) este disponibil pe pagina cursului.

```
1  % program principal
2   date   = citeste_date_punte();
3   date   = genereaza_grid_si_matriceDER(date);
4   [out] = solve_static_beam(date);
```

Implementare în Matlab

```
function date = citeste_date_punte()
%% geometrie
date.l = 610e-6; % [m] lungimea membranei
date.w = 40e-6;  % [m] latimea membranei
date.h = 2.2e-6; % [m] inaltimea membranei
date.u0 = 2.3e-6; %[m] gapul initial
%% materiale
date.E = 149e9;  % [Pa] modulul lui Young
date.S_overHW = -3.7e6; % tensiuni reziduale [Pa] (unde S = stress coefficient [N])
date.epsilon0 = 8.854187817e-12; % F/m - permitivitatea vidului
%% tensiunea de actionare
date.v = 8; %V - valoare folosita daca se va face simularea pentru o anumita valoare
      a tensiunii
%% parametrii pentru procedura neliniara (Newton)
date.err_impus = 1e-3; % eroarea impusa procedurii neliniare
date.it_max = 100; % nr max de pasi pentru procedura iterativa neliniara
%% finetea gridului de discretizare
date.n = 40;  % uniform pe OX, reprezinta nr de grade de libertate ale modelului
end
```

Implementare în Matlab

```
function date = genereaza_grid_si_matriceDER(date)
%%
n = date.n;
l = date.l;
%
dx = l/(n+1);
dx2 = dx^2;
dx4 = dx2^2;
date.dx = dx;
date.dx2 = dx2;
date.dx4 = dx4;
date.doidx = 2*dx;
% grid
date.x = linspace(0,l,n+2)';
%%
% derivata de ordinul 2 dupa x, se va aplica deplasarii (u-u0)
e = ones(n,1);
DER2 = spdiags([e -2*e e], -1:1, n, n);
% derivata de ordiul 4, centrata, dupa x, se va aplica deplasarii (u-u0)
DER2 = DER2/dx2;
DER4 = spdiags([e -4*e 6*e -4*e e], -2:2, n, n);
DER4(1,1) = 7;
DER4(n,n) = 7;
DER4 = DER4/dx4;
date.DER2 = DER2;
date.DER4 = DER4;
end
```

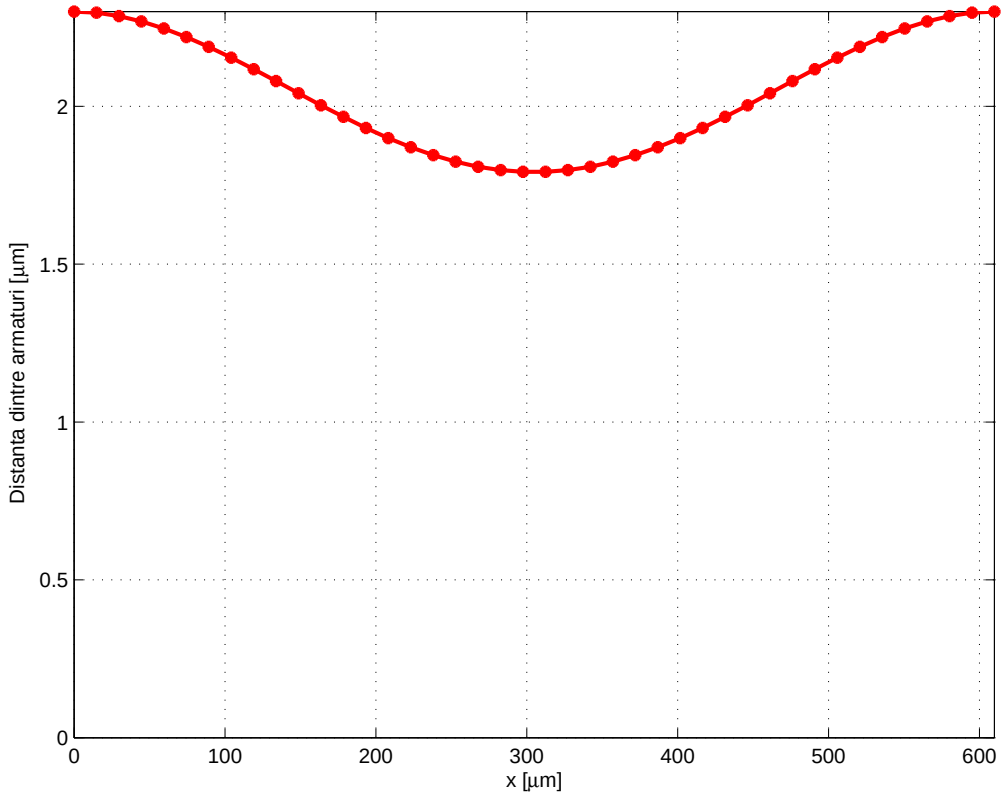
Implementare în Matlab

```
function [out] = solve_static_beam(date)
%%
out.errorCode = 0;
out.v = date.v;  out.x = date.x;
out.u = [];      % se va completa dupa rezolvare
out.gapMin = []; % valoarea la mijlocul barei
%
date.S = date.S_overHW*date.h*date.w; %[Pa*m^2]
I = date.w*date.h^3/12;
date.EI = date.E*I;
date.v2 = date.v^2; %V^2
u0 = date.u0;     n = date.n;
sol_v = u0*ones(n,1);  %initializare pentru Newton
err_impus = date.err_impus;     it_max = date.it_max;
pozitie_mijloc = floor(n/2) + 1;  % indexul punctului din mijlocul barei
u0 = date.u0;
I = date.I;
err = 2*err_impus;         knelin = 0;
while and((err > err_impus),knelin < it_max)
    knelin = knelin + 1;
    [Fdeu, jacobianF] = f_fder_static_matrix(sol_v,date);
    corectie = -jacobianF\Fdeu;
    sol_n = sol_v + corectie;
    err = norm(corectie)/norm(sol_v);
    fprintf('  corectia relativa = %e\n',err);
    sol_v = sol_n; % actualizare solutie veche
end
% vedeti codul complet pe pagina cursului
```

Implementare în Matlab

```
function [rhs, jacobian] = f_fder_static_matrix(u,date)
% calculeaza membrul drept si jacobianul sistemului de rezolvat
u0 = date.u0;  epsilon0 = date.epsilon0;
w = date.w;    v2 = date.v2;
S = date.S;    EI = date.EI;
%%
n = date.n;
dx2 = date.dx2;
dx4 = date.dx4;
%%
upatrat = u.^2;
uminusu0 = u - u0;
DER2 = date.DER2;
DER4 = date.DER4;
%% membrul drept al sistemului de rezolvat F(uv)
rhs1 = S*DER2*uminusu0 -EI*DER4*uminusu0;
rhs = upatrat.*rhs1 - epsilon0*w*v2/2;
%% jacobianul membrului drept F'(uv)
Sdx2 = S/dx2;
EIdx4 = EI/dx4;
diag0 = 2*u.*rhs1 + upatrat*(Sdx2*(-2) - EIdx4*6);
diag0(1) = diag0(1) + upatrat(1)*(-EIdx4);  % ca sa se faca 7 in loc de 6
diag0(end) = diag0(end) + upatrat(end)*(-EIdx4);
diag1 = upatrat*(Sdx2 + EIdx4*4);
diag2 = upatrat*(-EIdx4);
jacobian = spdiags([diag2 diag1 diag0 diag1 diag2], [-2 -1 0 1 2], n, n);
jacobian = jacobian'; % spdiags il creaza transpus, si nu e matrice simetrica
end
```

Implementare în Matlab



Temă

Temă facultativă

- Investigaţi convergenţa rezolvării sistemului neliniar de ecuaţii pentru diferite valori ale tensiunii aplicate. şi pentru diferite discretizări ale domeniului.
- Modificaţi codul astfel încât să puteţi detecta automat tensiunea minimă de acţionare.
- Ce modificări ar trebui să aduceţi implementării astfel încât să exploataţi simetria problemei.
- Rezolvaţi ecuaţia cu un alt software (de exemplu COMSOL) şi comparaţi soluţiile (deformarea la o anumită tensiune, tensiunea minimă de acţionare).
- Redactaţi un raport relevant.