MEF-AI.3.1

ALGORITM DE ÎNTOCMIRE A UNUI PROGRAM DE MODELARE ŞI REZOLVARE CU MEF

CUPRINS

MFF-AI.3.1.1 INTRODUCERE

MEF a apărut si s-a dezvoltat rapid, ca urmare a apariţiei calculatoarelor numerice cu capacităţi mari de stocare şi viteze mari de lucru. Acestea au permis inginerilor rezolvarea, prin calcul numeric aproximativ, a unor probleme tot mai complexe şi dificile privind evaluarea deformaţiilor şi tensiunilor mecanice, temperaturilor, presiunilor, vitezelor etc., în condiţiile unor materiale diversificate, având comportament liniar sau neliniar, cu condiţii limită complicate şi cu acţiuni externe staţionare sau nestaţionare.

Există un mare număr de programe performante de calculator, generalizate, destinate rezolvării problemelor din domenii diferite. Aceste probleme conţin un număr mare de instrucţiuni, necesită calculatoare cu memorii şi viteze de calcul mari, sunt foarte scumpe şi necesită instruiri speciale ale personalului utilizator.

În practică se întâlnesc frecvent probleme specifice, pentru a căror rezolvare nu se justifică achiziţionarea de programe generalizate şi pentru care „merită” să se întocmească programe specializate mai mici, relativ uşor de elaborat, testat, exploatat şi care realizează timpi de calculator acceptabili.

Deoarece cunoaşterea conceptelor de bază ale teoriei metodei elementelor finite, nu este suficientă pentru întocmirea unui program de calculator, în continuare se vor urmări pas cu pas, etapele necesare întocmirii acestuia, pentru cazul unei probleme de analiza structurilor plane. Ideea că partea de întocmire a unui program de calculator să fie realizată de către un specialist programator nu este întotdeauna oportună, deoarece acesta trebuie să cunoască în detaliu concepte de bază ale MEF şi mai ales trebuie să cunoască bine ipotezele şi principiile domeniului problemei de rezolvat. Ideal, pentru realizarea unui program care are la bază MEF, este necesar ca specialiştii în domeniul problemei de rezolvat să colaboreze cu specialiştii în informatică. Pentru buna colaborare a acestora, este necesară instruirea interdisciplinară în domeniile ingineriei şi informaticii.

MFF-AI.3.1.2 ETAPELE UNUI ALGORITM DE MODELARE ŞI REZOLVARE CU MEF

La elaborarea unor programe care au la bază MEF, trebuie să se ţină cont, pe cât posibil, de următoarele cerinţe:

- organizarea modulară a programului, după un algoritm care să-i confere generalitate şi posibilitate de dezvoltare;
- necesitatea unor subprograme, pentru testări privind corectitudinea datelor de intrare şi limitările introduse de program, care în caz de eroare atenţionează utilizatorul;
- elaborarea unor subprograme pentru necesarul de resurse hardware şi necesarul de memorie;

- posibilitatea integrării şi colaborării programului de întocmit în programe mai mari şi, respectiv, cu alte programe.
- Pentru o mai bună înţelegere a etapelor principale necesare la întocmirea unui program, care are la bază metoda elementelor finite, se consideră mai multe exemple.

Etapele principale ale algoritmului de rezolvare a problemelor de analiza structurilor plane sunt sintetizate în schema din fig. 1. În continuare, se prezintă aceste etape neinsistându-se asupra aspectelor de introducere, stocare şi modificare a datelor.

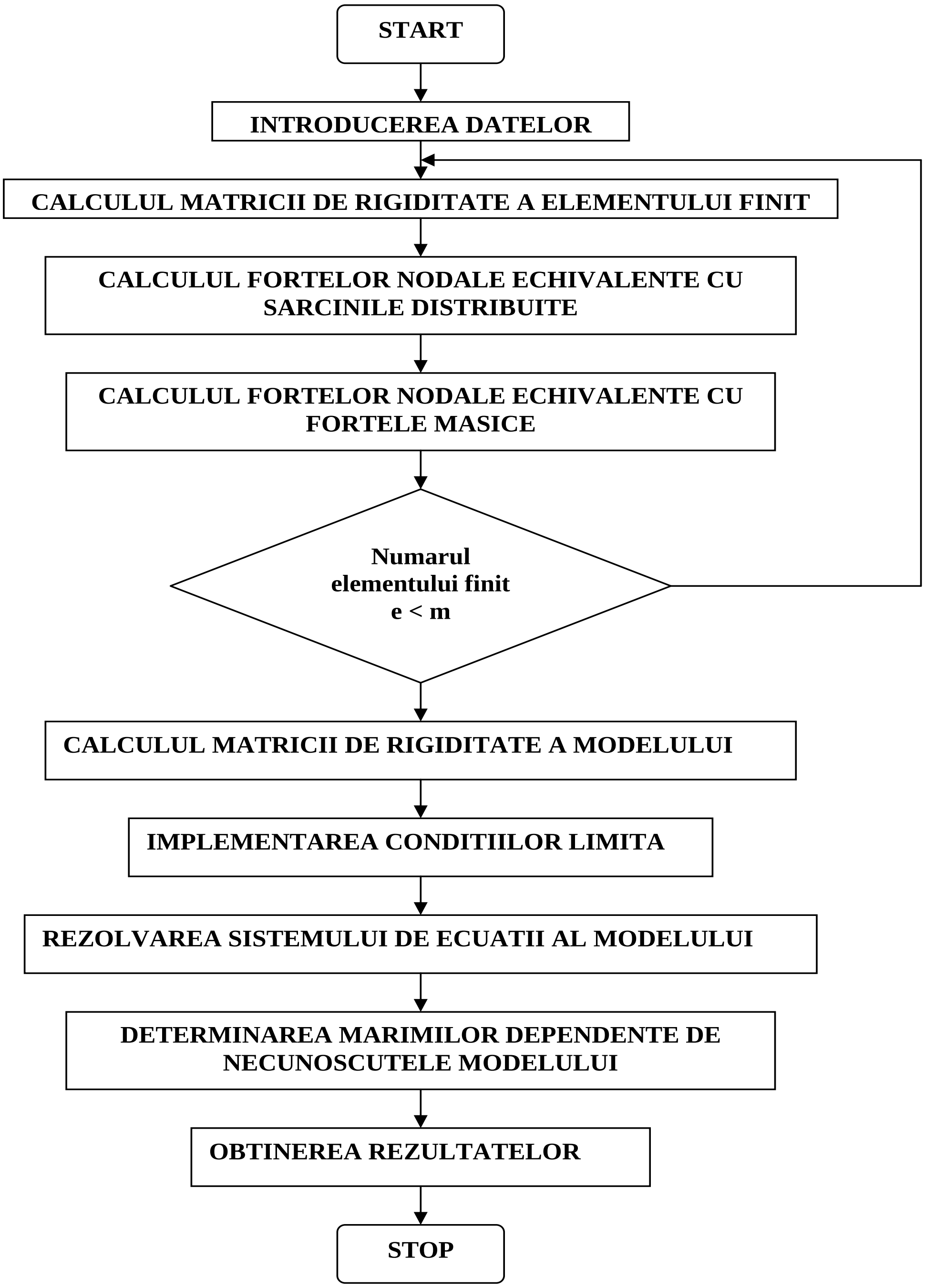


Fig. 1

MFF-AI.3.1.3 INTRODUCEREA DATELOR

Pentru a se putea evidenţia clar etapele algoritmului, se discretizează structura folosind elemente finite de tip patrulater numai cu noduri exterioare primare. De asemenea, în acelaşi scop se consideră un număr minim de elemente finite. În fig. 2. se prezintă structura idealizată prin discretizare, raportată la sistemul de coordonate global xOy.

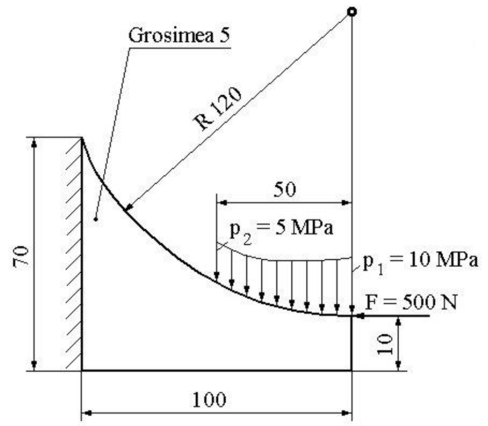


Fig. 2

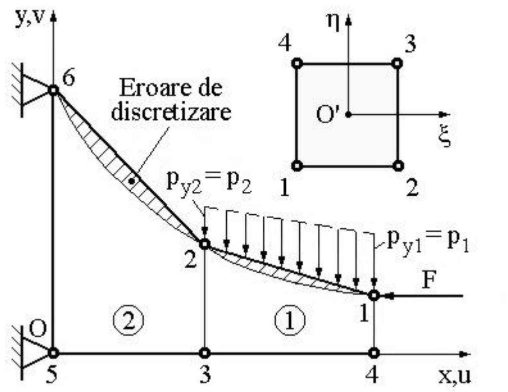


Fig. 3

Datele de intrare necesare pentru descrierea structurii idealizate, în vederea modelării cu elemente finite, se pot sintetiza în cinci grupe.

a. Date generale despre structură şi matricele de rigiditate. Parametrii care dau informaţii generale despre structura de analizat şi matricele de rigiditate au următoarele notaţii, valori şi semnificaţii:

- **nne = 4** - numărul de noduri ale elementului finit de referinţă;
- **ngl = 2** - numărul gradelor de libertate (necunoscute) ale unui nod, care în cazul exemplului considerat sunt deplasările liniare u şi v după axele sistemului global (fig. 3);
- **ne = 2** - numărul de elemente finite ale structurii;
- **nn = 6** - numărul de noduri ale structurii;
- **$[\ cn\ ]_{nne,ne}$ = [(cn(I,j),j = 1,nne] = [(1, 2, 3, 4), (3, 2, 6, 5)]** – matricea de conexiuni a elementelor finite ale structurii, care se completează cu numerele nodurilor elementelor finite respectând sensul antiorar;
- **necel = nne • ngl = 8** - numărul de linii (coloane) ale matricei de rigiditate a elementului finit (ordinul matricelor de rigiditate ale elementelor finite);
- **nec = nn • ngl = 12** - numărul de linii (coloane) ale matricei structurii (ordinul matricei de rigiditate a structurii);
- **lband = ngl (1+ max((| cn(i,j) – cn(i,j + 1)|,j = 1,nne -1), i = 1,ne) = 10** – lăţimea semibenzii matricei de rigiditate, dată de produsul dintre diferenţa maximă a numerelor din perechile posibile cu numerele nodurilor elementelor finite şi numărul gradelor de libertate corespunzătoare unui nod.

Lăţimea semibenzii, astfel definită, se calculează cu subprogramul LBANDA din fig. 4.

```
Procedure LBANDA(var ne,nne,lband,ngl:integer;cn:mat2);
{type mat2=array[1..ne,1..nne] of integer;}
var i,j,jj,dif:integer;
begin
   lband:=0;
   for i:=1 to ne do
      for j:=1 to nne-1 do
          for jj:=1 to nne do
             begin
                dif:=abs(cn[i,j]-cn[i,jj]);
                if dif > lband then lband:=dif;
             end;
   lband:=ngl+ngl*lband;
end;
```

Fig. 4

Pentru realizarea unei lăţimi minime a semibenzii, este necesar ca numerotarea codurilor structurii să se facă în aşa fel încât diferenţa din relaţia de definire a semibenzii să fie minimă. Astfel, pentru structura din fig. 3 numerotând nodurile ca în fig. 5, rezultă lband = 8. Unele programe de calcul au subprograme speciale de renumerotare a nodurilor structurii, în vederea realizării unei lăţimi minime a semibenzii şi deci a avantajelor rezultate din aceasta.

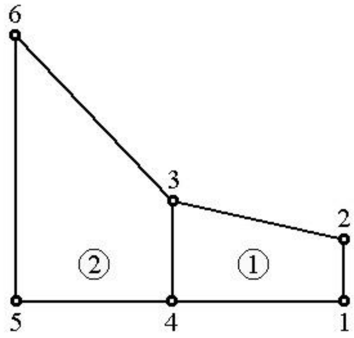


Fig. 5

b. Date despre geometria structurii. Aceasta se poate descrie cu ajutorul următorilor parametri:

- **$[\,xg\,]_{1,nn}$ = [xg(i), I = 1,nn] = [x_1 x_2...x_{nn}] = [100 50 50 100 0 0]** – matricea absciselor nodurilor structurii;
- **$[\,yg\,]_{1,nn}$ = [yg(i), I = 1,nn] = [y_1 y_2...y_{nn}] = [10 18 0 0 0 70]** – matricea ordonatelor nodurilor structurii;
- **h = 5** – grosimea plăcii (fig. 2)

c. Date despre încărcarea structurii. Considerarea încărcării structurii se poate realiza prin următorii parametri:

- **$[\,fc\,]_{1,nec}$ = [fc(i), I = 1,nec] = [fx_1 fy_1 ... fx_{nn} fy_{nn}] = [-500 0 0 0 0 0 0 0 0 0 0 0]** – matricea componentelor, după axele sistemului de coordonate global, ale forţelor concentrate nodale;
- **$[p]_{1,nec}$ = [p(i),I = 1,nec] = [px_1 py_1 ... px_{nn} py_{nn}] = [0 -5 0 -10 0 0 0 0 0 0 0 0]** – valorile componentelor încărcărilor distribuite, din planul structurii, considerate în nodurile de frontieră în care acţionează.

Descrierea cu acurateţe mărită a încărcărilor neuniform distribuite presupune considerarea unui număr mărit de noduri în zona de acţiune a acestora.

d. Date despre condiţiile limită. Condiţiile limită geometrice se cuantifică prin înlocuirea în matricea deplasărilor nodale corespunzătoare nodurilor structurii a valorilor deplasărilor nodale cunoscute (inclusiv cele nule). În cazul problemei din exemplul considerat, formele sintetică şi extinsă ale matricei deplasărilor nodale, cu considerarea valorilor deplasărilor nodale cunoscute, sunt:

$[d]_{1,nec}$ = [d(i),i=1,nec] = [u_1 v_1 ... u_{nn} v_{nn}] = [x x x x x x x x 0 0 0 0],

în care cu x s-au notat valorile necunoscute ale deplasărilor care urmează a fi aflate. Numărul minim al deplasărilor nodale nule trebuie să elimine mişcarea generală a corpului solid. Pentru structura plană, numărul minim al deplasărilor nodale nule este 3. Corelat cu matricea deplasărilor nodale de mai sus, pentru uşoara identificare a deplasărilor se generează matricea:

$[el]_{1,nec}$= [el(i), i=1,nec] = [0 0 0 0 0 0 0 0 1 1 1 1],

care ia valori egale cu 1, corespunzător deplasărilor cunoscute şi valori nule, corespunzător deplasărilor necunoscute.

e. Date despre proprietăţile fizice ale materialelor. În cazul exemplului considerat, cu structura realizată dintr+un singur tip de material, este necesar să se cunoască următoarele date:

- **neps = 3** – dimensiunile (ordinul= matricei de elasticitate corespunzătoare problemei particulare aplicată în cazul acestui exemplu;
- **el = 2,1 • 10^5** – modulul de elasticitate longitudinală al materialului structurii, în MPa;
- **cp = 0,3** – coeficientul (contracţiei transversale) Poisson al materialului structurii;
- **dens = 7800** – densitatea materialului structurii, în kg-m^3;
- **g = 9,8** – acceleraţia gravitaţională, îm m-s^2;
- **$[E]_{1,neps}$ = [(E(I,j),j=1,neps), i=1,neps]** – matricea de elasticitate pentru starea plană de tensiuni, calculată cu subprogramul MAT_E din fig. 6.

Datele de intrare despre structură se stochează în memoria auxiliară (fixă) a calculatorului cu posibilitatea de citire, modificare şi scriere.

```
Procedure MAT_E(var El,cp:real;E:mat5);
{type mat5=array[1..neps,1..neps] of real;}
var el:real
begin
   el:=El/(1-cp*cp);
   E[1,1]:=el;E[1,2]:=el*cp;E[1,3]:=0;
   E[2,1]:=E[1,2];E[2,2]:=E[1,1];E[2,3]:=0;
   E[3,1]:=0;E[3,2]:=0;E[3,3]:=el*(1-cp)/2;
end;
```

Fig. 6

MFF-AI.3.1.3 MATRICEA DE RIGIDITATE A ELEMENTULUI FINIT

În cadrul acestei etape se pune problema explicitării matricei de rigiditate a elementului finit la general care urmează să fie evaluată pentru fiecare element finit în parte. Pentru această problema matricea de rigiditate elementală are forma,

$$\left[k^e\right] = \int_{V^e} [B]^T [E][B] dV \, . \tag{1}$$

Pentru calculul acestei integrale este necesar să se cunoască matricea de transformare deformaţii-deplasări $[B]$ şi volumul dV.

Modelările geometrică şi a parametrilor fizici (deplasările nodale) necunoscuţi s-au realizat având la bază elementul finit „părinte” de tip patrulater, în coordonate locale naturale, cu ajutorul relaţiilor:

$$\begin{aligned} x(\xi,\eta) &= \sum_{i=1}^{4} N_i(\xi,\eta) x_i , \\ y(\xi,\eta) &= \sum_{i=1}^{4} N_i(\xi,\eta) y_i \end{aligned} \tag{2}$$

şi respectiv,

$$\begin{aligned} u_i(\xi,\eta) &= \sum_{i=1}^{4} N_i(\xi,\eta) u_i , \\ v_i(\xi,\eta) &= \sum_{i=1}^{4} N_i(\xi,\eta) v_i , \end{aligned} \tag{3}$$

în care x_i, y_i şi u_i, v_i, cu i = 1,2,…,4, reprezintă coordonatele nodurilor şi, respective, necunoscutele nodale, corespunzătoare nodurilor şi, respective, necunoscutele nodale, corespunzătoare unui element finit oarecare:

$$\begin{aligned} N_1(\xi,\eta) &= \frac{1}{2}(1-\xi)(1-\eta), \qquad & N_2(\xi,\eta) &= \frac{1}{2}(1|+\xi)(1-\eta), \\ N_3(\xi,\eta) &= \frac{1}{2}(1+\xi)(1+\eta), \qquad & N_4(\xi,\eta) &= \frac{1}{2}(1-\xi)(1+\eta) \end{aligned} \tag{4}$$

reprezintă funcţiile de formă asociate elementului finit „părinte” considerat.
Determinarea elementelor matricei $[B]$ se realizează pornind de la derivatele parţiale

$$\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y},$$

Deoarece parametrii fizici necunoscuţi sunt modelaţi luând ca variabile independente coordonatele locale ξ, η pentru determinarea derivatelor parţiale de mai sus, se porneşte de la relaţiile:

$$\begin{aligned} \frac{\partial u}{\partial \xi} &= \frac{\partial u}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial u}{\partial y},\frac{\partial y}{\partial \xi}, \\ \frac{\partial u}{\partial \eta} &= \frac{\partial v}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial v}{\partial y},\frac{\partial y}{\partial \eta}, \end{aligned} \tag{5}$$

care, în forma matriceală, se pot scrie sub forma

$$\begin{bmatrix} \dfrac{\partial u}{\partial \xi} \\ \dfrac{\partial u}{\partial \eta} \end{bmatrix} = [J] \begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial u}{\partial y} \end{bmatrix}, \tag{6}$$

cu

$$[J] = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} \end{bmatrix} \tag{7}$$

matricea Jacobi de transformare,

$$\begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial u}{\partial y} \end{bmatrix} = [J]^{-1} \begin{bmatrix} \dfrac{\partial u}{\partial \xi} \\ \dfrac{\partial u}{\partial \eta} \end{bmatrix}. \tag{8}$$

Elementele matricei Jacobi se calculează cu relaţiile:

$$\begin{aligned} \frac{\partial x}{\partial \xi} &= \sum_{i=1}^{4} \frac{\partial N_i}{\partial \xi} x_i, \\ \frac{\partial y}{\partial \xi} &= \sum_{i=1}^{4} \frac{\partial N_i}{\partial \xi} y_i, \\ \frac{\partial x}{\partial \eta} &= \sum_{i=1}^{4} \frac{\partial N_i}{\partial \eta} x_i, \\ \frac{\partial y}{\partial \eta} &= \sum_{i=1}^{4} \frac{\partial N_i}{\partial \eta} y_i, \end{aligned} \tag{9}$$

în care:

$$\frac{\partial N_1}{\partial \xi} = -\frac{1}{4}(1-\eta), \quad \frac{\partial N_2}{\partial \xi} = +\frac{1}{4}(1-\eta), \quad \frac{\partial N_3}{\partial \xi} = +\frac{1}{4}(1+\eta), \quad \frac{\partial N_4}{\partial \xi} = -\frac{1}{4}(1+\eta),$$

$$\frac{\partial N_1}{\partial \eta} = -\frac{1}{4}(1-\xi), \quad \frac{\partial N_2}{\partial \eta} = -\frac{1}{4}(1+\xi), \quad \frac{\partial N_3}{\partial \eta} = +\frac{1}{4}(1+\xi), \quad \frac{\partial N_4}{\partial \eta} = +\frac{1}{4}(1-\xi),$$

Determinantul matricei Jacobi (7)

$$|J| = \frac{\partial y}{\partial \eta}\frac{\partial x}{\partial \xi} - \frac{\partial y}{\partial \xi}\frac{\partial x}{\partial \eta}, \tag{10}$$

după înlocuirea relaţiilor (9), devine:

$$\begin{aligned} |J| &= \sum_{i=1}^{4} \frac{\partial N_i}{\partial \eta} y_i \sum_{i=1}^{4} \frac{\partial N_i}{\partial \xi} x_i - \sum_{i=1}^{4} \frac{\partial N_i}{\partial \xi} y_i \sum_{i=1}^{4} \frac{\partial N_i}{\partial \eta} x_i = \\ &= \sum \sum (y_i (\frac{\partial N_i}{\partial \eta}\frac{\partial N_j}{\partial \xi} - \frac{\partial N_i}{\partial \xi}\frac{\partial N_j}{\partial \eta}) x_i) \end{aligned} \tag{11}$$

Această relaţie se mai poate scrie şi sub forma,

$$|J| = [y_1\ y_2\ \ y_3\ y_4]\ [a]\ [x_1\ x_2\ \ x_3\ x_4]^T, \tag{12}$$

în care, matricea [a] are termenii cu forma,

$$a_{ij} = \frac{\partial N_i}{\partial \eta}\frac{\partial N_j}{\partial \xi} - \frac{\partial N_i}{\partial \xi}\frac{\partial N_j}{\partial \eta} \tag{13}$$

Pentru funcţiile de formă considerate, matricea [a] este

$$[a] = -\frac{1}{8}\begin{bmatrix} 0 & 1-\eta & \eta-\xi & \xi-1 \\ \eta-1 & 0 & \xi+1 & -(\eta+\xi) \\ \xi-\eta & -(\xi+1) & 0 & \eta+1 \\ 1-\xi & \xi+1 & -(1+\eta) & 0 \end{bmatrix} \tag{14}$$

Relaţia (8), ţinând cont de expresia inversei matricei Jacobi, devine

$$\begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial u}{\partial y} \end{bmatrix} = \frac{1}{|J|}\begin{bmatrix} \dfrac{\partial y}{\partial \eta} & -\dfrac{\partial y}{\partial \xi} \\ -\dfrac{\partial x}{\partial \eta} & \dfrac{\partial x}{\partial \xi} \end{bmatrix}\begin{bmatrix} \dfrac{\partial u}{\partial \xi} \\ \dfrac{\partial u}{\partial \eta} \end{bmatrix}, \tag{15}$$

din care rezultă,

$$\begin{aligned} \frac{\partial u}{\partial x} &= \frac{1}{|J|}(\frac{\partial y}{\partial \eta}\frac{\partial u}{\partial \xi} - \frac{\partial y}{\partial \xi}\frac{\partial u}{\partial \eta}, \\ \frac{\partial u}{\partial y} &= \frac{1}{|J|}(\frac{\partial x}{\partial \eta}\frac{\partial u}{\partial \xi} - \frac{\partial x}{\partial \xi}\frac{\partial u}{\partial \eta}. \end{aligned} \tag{16}$$

Derivatele relaţiilor (3),

$$\begin{aligned} \frac{\partial u}{\partial \xi} &= \sum_{i=1}^{4}\frac{\partial N_i}{\partial \xi}u_i, \\ \frac{\partial u}{\partial \eta} &= \sum_{j=1}^{4}\frac{\partial N_j}{\partial \xi}u_j, \\ \frac{\partial u}{\partial \xi} &= \sum_{i=1}^{4}\frac{\partial N_i}{\partial \xi}v_i, \\ \frac{\partial u}{\partial \eta} &= \sum_{j=1}^{4}\frac{\partial N_j}{\partial \xi}v_j \end{aligned} \tag{17}$$

şi relaţiile (4.9), înlocuite în (4.16), determină expresiile:

$$\begin{aligned} \frac{\partial u}{\partial x} &= \frac{1}{|J|}\begin{bmatrix} y_1 & y_2 & y_3 & y_4 \end{bmatrix}[a]\begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix}^T = \varepsilon_x, \\ \frac{\partial u}{\partial y} &= \frac{1}{|J|}\begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}[a]\begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix}^T. \end{aligned} \tag{18}$$

Similar, pornind de la relaţiile:

$$\frac{\partial v}{\partial \xi} = \frac{\partial v}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial v}{\partial y}\frac{\partial y}{\partial \xi},$$
$$\frac{\partial v}{\partial \eta} = \frac{\partial v}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial v}{\partial y}\frac{\partial y}{\partial \eta}, \tag{19}$$

aplicând acelaşi procedeu, rezultă:

$$\frac{\partial v}{\partial x} = \frac{1}{|J|}\begin{bmatrix} y_1 & y_2 & y_3 & y_4 \end{bmatrix}[a]\begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix}^T,$$
$$\frac{\partial v}{\partial y} = \frac{1}{|J|}\begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}[a]\begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix}^T = \varepsilon_y. \tag{20}$$

Înlocuind relaţiile (19) şi (20) în expresia matricei deformaţiilor,

$$[\varepsilon] = \left[\frac{\partial u}{\partial x}\frac{\partial v}{\partial y}\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right]^T,$$

rezultă relaţia,

$$[\varepsilon] = [B]\begin{bmatrix} u_1 & v_1 & u_2 & v_2 & u_3 & v_3 & u_4 & v_4 \end{bmatrix}^T \tag{21}$$

în care, elementele matricei $[B]$ au expresiile:

$$\begin{aligned}
&B(1,2j-1) = \frac{1}{|J|}\sum_{i=1}^{4} y_i a(i,j), \text{ pentru } j = 1,2,3,4;\\
&B(1,j) = 0, \text{ pentru } j = 2,4,6,8;\\
&B(2,2j) = -\frac{1}{|J|}\sum_{i=1}^{4} x_i a(i,j), \text{ pentru } j = 1,2,3,4;\\
&B(2,j) = 0, \text{ pentru } j = 1,3,5,7;\\
&B(3,j) = B(2,j+1), \text{ pentru } j = 1,3,5,7;\\
&B(3,j) = B(1,j-1), \text{ pentru } j = 2,4,6,8.
\end{aligned} \tag{22}$$

Dezvoltând sumele şi notând $x_m - x_n = x_{mn}$ şi $y_m - y_n = y_{mn}$, elementele nenule ale matricei $[B]$ devin:

$$\begin{aligned}
&B(1,1) = B(3,2) = \frac{1}{8|J|}(y_{24} + \xi y_{43} + \eta y_{32}),\\
&B(1,3) = B(3,4) = \frac{1}{8|J|}(y_{31} + \xi y_{34} + \eta y_{14}),\\
&B(1,5) = B(3,6) = \frac{1}{8|J|}(y_{42} + \xi y_{12} + \eta y_{41}),\\
&B(1,7) = B(3,8) = \frac{1}{8|J|}(y_{13} + \xi y_{21} + \eta y_{23}),\\
&B(2,2) = B(3,1) = \frac{1}{8|J|}(x_{42} + \xi x_{34} + \eta x_{23}),\\
&B(2,4) = B(3,3) = \frac{1}{8|J|}(x_{13} + \xi x_{43} + \eta x_{41}),\\
&B(2,6) = B(3,5) = \frac{1}{8|J|}(x_{24} + \xi x_{21} + \eta x_{14}),\\
&B(2,8) = B(3,7) = \frac{1}{8|J|}(x_{31} + \xi x_{12} + \eta x_{32}),
\end{aligned} \tag{23}$$

Volumul elementar dV, având în vedere că grosimea structurii este constantă, se poate calcula cu relaţia,

$$dV = h\,dx\,dy \tag{24}$$

care transpusă în sistemul de coordonate local devine,

$$dV = h|J|d\xi d\eta . \tag{25}$$

Prin înlocuirea acestei relaţii în expresia matricei de rigiditate, se obţine

$$\left[k^e\right] = h\int_A [B]^T [E][B]|J|d\xi d\eta \tag{26}$$

Integrala din această relaţie se evaluează cu metoda Gauss, folosind formula cu două puncte Gauss şi deci matricea de rigiditate a elementului finit se poate calcula cu relaţia

$$\begin{aligned}\left[k^e\right] = h(+\ & w_1 w_2 \left[B(\xi_1\eta_1)\right]^T [E]\left[B(\xi_1\eta_1)\right]|J(\xi_1\eta_1)| + \\ & w_1 w_2 \left[B(\xi_1\eta_2)\right]^T [E]\left[B(\xi_1\eta_2)\right]|J(\xi_1\eta_2)| + \\ & w_2 w_1 \left[B(\xi_2\eta_1)\right]^T [E]\left[B(\xi_2\eta_1)\right]|J(\xi_2\eta_1)| + \\ & w_2 w_2 \left[B(\xi_2\eta_2)\right]^T [E]\left[B(\xi_2\eta_2)\right]|J(\xi_2\eta_2)|)\end{aligned} \tag{27}$$

în care, coeficienţii de pondere şi coordonatele punctelor Gauss sunt: $w_1=w_2=1$ şi, respectiv,

$$\begin{aligned}\xi_1 &= -0{,}57735,\\ \eta_1 &= -0{,}57735,\\ \xi_2 &= -\xi_1,\\ \eta_2 &= -\eta_1.\end{aligned}$$

În fig. 7 se prezintă subprogramul MAT_RIG1, care permite calculul elementelor matricei de rigiditate a elementului finit, conform relaţiei (27). Dacă se utilizează alt tip de element finit, similar se poate dezvolta o nouă expresie pentru matricea de rigiditate a elementului finit şi deci şi un nou subprogram de calcul.

MFF-AI.3.1.4 FORŢE NODALE ECHIVALENTE

Forţe nodale echivalente sarcinilor distribuite, aplicate pe muchiile elementului finit

Echivalarea sarcinilor distribuite pe frontiera unui element finit, cu un sistem de forţe concentrate în nodurile elementului finit, se realizează egalând energia potenţială a celor două cazuri de încărcare. În cazul acestui exemplu, frontiera elementului finit este compusă din patru muchii.

Potenţialul corespunzător unui element de arie de pe o muchie a elementului finit (fig.8, a) este dat de relaţia

$$dV_p = -\,(p_x\ dA)\,u - (p_v\,dA)\,v, \tag{28}$$

în care u,v şi p_x, p_v sunt componentele matricelor deplasare şi, respectiv, sarcinii distribuite (presiune), într-un punct M ales arbitrar pe muchiile elementului finit.

Dacă deplasările u şi v ale punctelor muchiei sunt modelate în raport cu sistemul de coordonate naturale ξ – naturale – în funcţie de deplasările nodurilor muchiei $u_{1,2}$ şi $v_{1,2}$ – cu relaţia,

$$\begin{bmatrix} u \\ v \end{bmatrix} = [N]\begin{bmatrix} u_1 & v_1 & u_2 & v_2 \end{bmatrix}^T, \tag{29}$$

în care,

$$[N]=\begin{bmatrix} N_1 & 0 & N_2 & 0 \\ 0 & N_1 & 0 & N_2 \end{bmatrix}. \tag{30}$$

```
Procedure MAT_RIG1(var nne:integer;xi,yi,det:vect1;ke:mat;E:mat5;
h:real);
{type mat=array[1..necel,1..necel] of real;vect1=array[1..nne] of
real;mat5=array[1..neps,1..neps];}
var i,j,k,i1:integer;G:array[1..4,1..4]of real;a,c:array[1..8,1..8]
of real;B,cc:array[1..3,1..8] of real;d,csi,eta,de,x:real;
begin
   {matricea ponderilor  i punctelor Gauss, G}
   G[1,1]:=1.0;G[1,2]:=1.0;G[1,3]:=-0.57736;G[1,4]:=G[1,3];
   G[2,1]:=1.0;G[2,2]:=1.0;G[2,3]:=-G[1,3];G[2,4]:=G[1,3];
   G[3,1]:=1.0;G[3,2]:=1.0;G[3,3]:=-G[1,3];G[3,4]:=-G[1,3];
   G[4,1]:=1.0;G[4,2]:=1.0;G[4,3]:=G[1,3];G[4,4]:=-G[1,3];
   for i:=1 to 8 do
   for j:=1 to 8 do ke[i,j]:=0;
   for i1:=1 to 4 do
      begin
         csi:=G[i,3];eta:=G[i,4];
         {calculul determinantului matricei Jacobi, notat det}
         a[1,1]:=0;a[1,2]:=(eta-1)/8;a[1,3]:=(csi-eta)/8;
         a[1,4]:=(1-eta)/8;a[2,1]:=-a[1,2];a[2,2]:=0;
         a[2,3]:=-(csi+1)/8;a[2,4]:=(eta+csi)/8;a[3,1]:=-a[1,3];
         a[3,2]:=-a[2,3];a[3,3]:=0;a[3,4]:=-(eta+1)/8;
         a[4,1]:=-a[1,4];a[4,2]:=-a[2,4];a[4,3]:=-a[3,4];a[4,4]:=0;
         det[i1]:=0;
         for i:=1 to 4 do
            begin
               d:=0;
               for j:=1 to 4 do
                  d:=d+yi[j]*a[i,j];
                  d:=d*xi[i];
                  de:=de+d;
            end;
         det[i1]:=de;
         {calculul matricei B}
         for i:=1 to 3 do
            for j:=1 to 8 do
               B[i,j]:=0;
         x:=8*det[i1];
        B[1,1]:=(yi[2]-yi[4]+csi*(yi[4]-yi[3])+eta*(yi[3]-yi[2]))/x;
        B[1,3]:=(yi[3]-yi[1]+csi*(yi[3]-yi[4])+eta*(yi[1]-yi[4]))/x;
        B[1,5]:=(yi[4]-yi[2]+csi*(yi[1]-yi[2])+eta*(yi[4]-yi[1]))/x;
        B[1,7]:=(yi[1]-yi[3]+csi*(yi[2]-yi[1])+eta*(yi[2]-yi[3]))/x;
        N[1,1]:=(1-csi)*(1-eta)/2;N[1,3]:=(1+csi)*(1-eta)/2;
        N[1,5]:=(1+csi)*(1-eta)/2;N[1,7]:=(1-csi)*(1+eta)/2;
        N[2,2]:=N[1,1];N[2,4]:=N[1,3];N[2,6]:=N[1,5];
        N[2,8]:=N[1,7];
        {calculul produsului matricelor: N transpusă şi Bg}
        for i:= 1 to 8 do
           begin
              qim[i]:=0;
              for k:=1 to 2 do qi[i]:=qi[i]+N[k,i]*Bg[k];
           end;
        {calculul integralelor}
     for i:=1 to 8 do qim[i]:=qim[i]+G[i1,1]*G[i1,2]*qi[i];
   end;
end;
```

Fig. 7

este matricea funcţiilor de formă, care pentru punctele muchiei se descriu cu ajutorul variabilei η (pe această muchie variabila $\xi=1$), cu relaţiile:

$$N_1=\frac{1-\eta}{2},\qquad N_2=\frac{1+\eta}{2}. \tag{31}$$

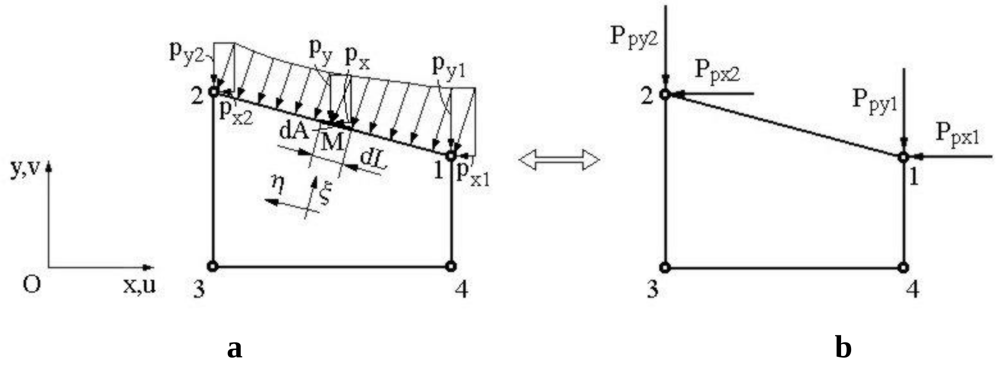


Fig. 8

Forma matriceală a expresiei energiei potenţiale (28),

$$V_p = -\int_A [u \quad v] \begin{bmatrix} p_x \\ p_y \end{bmatrix} dA, \tag{32}$$

folosind relaţiile (29) şi dA = h dL, devine,

$$V_p = -h[u_1 \quad v_1 \quad u_2 \quad v_2] \int_L [N]^T \begin{bmatrix} p_x \\ p_y \end{bmatrix} dL. \tag{33}$$

Pe de altă parte, energia potenţială a sistemului de forţe concentrate, echivalent forţelor distribuite (fig. 8, b), este

$$V_p = -[u_1 \quad v_1 \quad u_2 \quad v_2] \begin{bmatrix} P_{px1} \\ P_{py1} \\ P_{px2} \\ P_{py1} \end{bmatrix} = [u_1 \quad v_1 \quad u_2 \quad v_2][P_{p12}] \tag{34}$$

Comparând relaţiile (33) şi (34), rezultă expresia matricei forţelor nodale echivalente

$$[P_{p12}] = h\int_L [N]^T \begin{bmatrix} p_x \\ p_y \end{bmatrix} dL. \tag{35}$$

Lungimea dL şi proiecţiile acesteia dx şi dz, corespunzătoare muchiei încărcare cu sarcina distribuită, au expresiile:

$$dL = \sqrt{dx^2 + dy^2},$$

$$dx = \frac{\partial x}{\partial \xi} d\xi + \frac{\partial x}{\partial \eta} d\eta = \sum_{i=1}^{4} \frac{\partial N_1}{\partial \xi} x_i d\xi + \sum_{i=1}^{4} \frac{\partial N_1}{\partial \eta} x_i d\eta, \tag{36}$$

$$dy = \frac{\partial y}{\partial \xi} d\xi + \frac{\partial y}{\partial \eta} d\eta = \sum_{i=1}^{4} \frac{\partial N_1}{\partial \xi} y_i d\xi + \sum_{i=1}^{4} \frac{\partial N_1}{\partial \eta} y_i d\eta.$$

Deoarece pentru punctele muchiei cu sarcini distribuite $d\xi = 0$, $N_3 = N_4 = 0$, din relaţia (31),

$$\frac{\partial N_1}{\partial \eta} = -\frac{1}{2}, \quad \frac{\partial N_2}{\partial \eta} = \frac{1}{2},$$

după înlocuire, expresia lungimii elementare dL devine,

$$dL = \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}{2} d\eta = \frac{L_{12}}{2} d\eta, \tag{37}$$

unde, L_{12} este lungimea muchiei pe care acţionează forţa distribuită.

Astfel, matricea forţelor nodale echivalente, (35), devine,

$$\begin{bmatrix} P_{px1} \\ P_{py1} \\ P_{px2} \\ P_{py1} \end{bmatrix} = h\frac{L_{12}}{2}\int_{-1}^{1} \begin{bmatrix} N_1 & \\ & N_1 \\ N_2 & \\ & N_2 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix} d\eta, \tag{38}$$

de unde, rezultă relaţiile de calcul ale forţelor nodale echivalente:

$$\begin{aligned} P_{px1} &= h\frac{L_{12}}{2}\int_{-1}^{1} N_1(\eta) p_x d\eta, \\ P_{py1} &= h\frac{L_{12}}{2}\int_{-1}^{1} N_1(\eta) p_y d\eta, \\ P_{px2} &= h\frac{L_{12}}{2}\int_{-1}^{1} N_2(\eta) p_x d\eta, \\ P_{py2} &= h\frac{L_{12}}{2}\int_{-1}^{1} N_2(\eta) p_y d\eta, \end{aligned} \tag{39}$$

Pentru calculul integralelor de mai sus este necesară cunoaşterea expresiilor componentelor forţei distribuite $p_{x,y}$. În cazul MEF, şi variaţiile încărcărilor distribuite pe frontieră se aproximează cu funcţii simple, similare sau chiar aceleaşi cu cele folosite la modelările geometrică şi a parametrilor fizici necunoscuţi. De exemplu, pentru cazul structurii idealizate din fig. 3, se aproximează variaţia sarcinii distribuite p_z ($p_x = 0$) cu o variaţie liniară, de forma,

$$p_y = c_1 + c_2\eta. \tag{40}$$

Prin particularizarea acestei relaţii, cu valori ale variabilelor corespunzătoare nodurilor muchiei $(\eta = -1, p_y = p_{y1}; \eta = 1, p_y = p_{y2})$, se determină constantele $c_{1,2}$ şi rezultă pentru p_y următoarea expresie

$$p_y = \frac{1-\eta}{2} p_{y1} + \frac{1+\eta}{2} p_{y2}, \tag{41}$$

După înlocuirea relaţiei (41) în expresiile forţelor nodale, (39), şi calculul integralelor, rezultă,

```
Procedure FORTE_D1(var xi,yi,pi,qi,qid :vect1;h:real);
{type vect1=array[1..nne,1..nne] of real;}
var  l:array[1..4] of real;
begin
l[1]:=sqrt((xi[2]-xi[1])*(xi[2]-xi[1])+(yi[2]-yi[1])*(yi[2]-y
l[2]:=sqrt((xi[3]-xi[2])*(xi[3]-xi[2])+(yi[3]-yi[2])*(yi[3]-y
l[3]:=sqrt((xi[4]-xi[3])*(xi[4]-xi[3])+(yi[4]-yi[3])*(yi[4]-y
l[4]:=sqrt((xi[4]-xi[1])*(xi[4]-xi[1])+(yi[4]-yi[1])*(yi[4]-y
qid[1]:=h*l[1]*(4*pi[1]+pi[2]+pi[4])/6;
qid[2]:=h*l[1]*(pi[1]+4*pi[2]+pi[3])/6;
qid[3]:=h*l[1]*(4*pi[3]+pi[4]+pi[2])/6;
qid[1]:=h*l[1]*(4*pi[4]+pi[1]+pi[3])/6;
end;
```

Fig. 8

$$P_{py1} = \frac{hL_{12}}{6}(2p_{y1} + p_{y2}),$$
$$P_{py2} = \frac{hL_{12}}{6}(2p_{y1} + p_{y2}), \tag{42}$$

Subprogramul FORTE_D1, din fig. 8, permite calculul forţelor nodale echivalente, pentru cazul simplificat prezentat mai sus. Algoritmul şi subprogramul de calcul prezentat se poate dezvolta în direcţia considerării de forţe distribuite oarecare, atât ca direcţie cât şi ca tip de variaţie. Astfel, pentru elemente finite de tip patrulater neliniare, se pot realiza aproximări neliniare şi pentru variaţia încărcărilor distribuite.

Forţe nodale echivalente forţelor masice

Forţele masice sunt forţe care acţionează în interiorul structurii şi se concretizează ca forţe de greutate proprie şi/sau ca forţe centrifugale. Determinarea forţelor nodale echivalente forţelor masice se realizează prin echivalarea energiei potenţiale de deformare a structurii elementului finit sub acţiunea celor două tipuri de forţe.

Potenţialul de deformare a elementului de volum dV, din domeniul elementului finit din fig. 9, a, sub acţiunea forţei masice specifice B, cu componentele B_x, şi B_y, se calculează cu relaţia

$$dV_B = -B_x dVu - B_y dVv, \tag{43}$$

în care u şi v sunt deplasările corespunzătoare centrului elementului de volum dV considerat. Din această relaţie, după înlocuirile şi prelucrările similare ca în cazul etapei precedente, pentru potenţialul elementului finit (4.43), se obţine expresia

$$V_B = -h[u_1 \quad v_1 \quad u_2 \quad v_2 \quad u_3 \quad v_3 \quad u_4 \quad v_4]\iint_A [N]^T \begin{bmatrix} B_x \\ B_y \end{bmatrix} dA \tag{44}$$

în care

$$[N] = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{bmatrix} \tag{45}$$

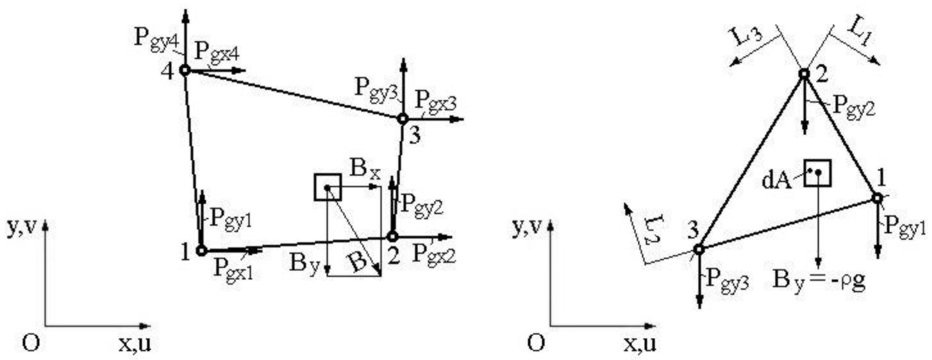
este matricea funcţiilor de formă, cu expresiile (4.4).

Sub acţiunea forţelor nodale echivalente

$$[P_g] = [P_{gx1} \quad P_{gy1} \quad P_{gx2} \quad P_{gy2} \quad P_{gx3} \quad P_{gy3} \quad P_{gx4} \quad P_{gy4}]^T, \tag{46}$$

energia potenţială de deformaţie este

$$V_B = -[u_1 \quad v_1 \quad u_2 \quad v_2 \quad u_3 \quad v_3 \quad u_4 \quad v_4][P_g]. \tag{47}$$



a **b**

Fig. 9

Din relaţiile (44) şi (47), rezultă matricea forţelor nodale echivalente,

$$\left[P_g\right] = h\int_A \left[N\right]^T \begin{bmatrix} B_x \\ B_y \end{bmatrix} dA \tag{48}$$

sau ţinând cont de relaţia $dxdy = |J| d\xi d\eta$, aceasta devine,

$$\left[P_g\right] = h\int_A \left[N\right]^T \begin{bmatrix} B_x \\ B_y \end{bmatrix} |J| d\xi d\eta . \tag{49}$$

Aplicând metoda Gauss de integrare în două puncte, se obţine

$$\left[P_g\right] = h\sum_{i=1}^{2}\sum_{j=1}^{2} w_i w_j \left[N(\xi_i \eta_j\right]^T \begin{bmatrix} B_x \\ B_y \end{bmatrix} |J(\xi_i, \eta_j)|, \tag{50}$$

unde coeficienţii de pondere, $w_{i,j}$ şi coordonatele ξ_i, η_j au valorile conform tab. 1. În cazul exemplului considerat $B_x = 0$ şi Bz = -ρg.

Subprogramul FORTE_M1, fig. 10, permite calculul forţelor nodale echivalente forţelor masice, folosind relaţia (50).

Şi în cazul forţelor masice sunt posibile mai multe relaţii de echivalare şi programe corespunzătoare, în funcţie de tipul elementului finit sau în funcţie de tipul şi forma de variaţie a forţelor masice.

Dacă discretizarea se realizează folosind elemente finite triunghiulare (fig. 10,b) raportate la sisteme de coordonate L naturale, integralele din relaţia forţelor echivalente se pot calcula cu formula,

$$\int_{A^{'}} L_1^{\alpha} L_2^{\beta} L_3^{\gamma} dA = \frac{\alpha!\beta!\gamma!}{(\alpha+\beta+\gamma+2)!} 2A^e$$

Tab. 1

Elemente finite	**Noduri**		
	i	**j**	**k**
1	1	3	2
2	2	3	4
3	4	3	5

Astfel, când se iau în considerare numai forţele de greutate (fig. 10,b), relaţia (48) devine,

```
Procedure FORTE_M1(var Bg:vect3;qim:vect4);
{type vect3=array[1..2] of real;vect4=array[1..8] of real;}
var i,j,i1,k:integer;G:array[1..4,1..4] of real;N:array[1..2,1..8]
of real;qi:array[1..8] of real;csi,eta:real;
begin
   {matricea ponderilor şi punctelor Gauss, G}
   G[1,1]:=1.0;G[1,2]:=1.0;G[1,3]:=-0.57736;G[1,4]:=G[1,3];
   G[2,1]:=1.0;G[2,2]:=1.0;G[2,3]:=-G[1,3];G[2,4]:=G[1,3];
   G[3,1]:=1.0;G[3,2]:=1.0;G[3,3]:=-G[1,3];G[3,4]:=-G[1,3];
   G[4,1]:=1.0;G[4,2]:=1.0;G[4,3]:=G[1,3];G[4,4]:=-G[1,3];
   for i:=1 to 8 do qim[i]:=0;
   for i1:=1 to 4 do
       begin
          csi:=G[i,3];eta:=G[i,4];
          {calculul elementelor matricei N }
          for i:=1 to 2 do
             for j:=1 to 8 do N[i,j]:=0;
          N[1,1]:=(1-csi)*(1-eta)/2;N[1,3]:=(1+csi)*(1-eta)/2;
          N[1,5]:=(1+csi)*(1-eta)/2;N[1,7]:=(1-csi)*(1+eta)/2;
          N[2,2]:=N[1,1];N[2,4]:=N[1,3];N[2,6]:=N[1,5];
          N[2,8]:=N[1,7];
          {calculul produsului matricelor: N transpusă şi Bg}
          for i:= 1 to 8 do
             begin
               qim[i]:=0;
               for k:=1 to 2 do qi[i]:=qi[i]+N[k,i]*Bg[k];
             end;
          {calculul integralelor}
       for i:=1 to 8 do qim[i]:=qim[i]+G[i1,1]*G[i1,2]*qi[i];
   end;
end;
```

Fig. 10

$$[P_g] = h\int_A \begin{bmatrix} L_1 & 0 \\ 0 & L_1 \\ L_2 & 0 \\ 0 & L_2 \\ L_3 & 0 \\ 0 & L_3 \end{bmatrix} \begin{bmatrix} 0 \\ -\rho g \end{bmatrix} dA = -\rho g h \int_A \begin{bmatrix} 0 \\ L_1 \\ 0 \\ L_2 \\ 0 \\ L_3 \end{bmatrix} dA \tag{51}$$

$$\int_{A^e} L_1 dA = \frac{1!0!0!}{3!} 2A^e = \frac{1}{3} A^e,$$

$$\int_{A^e} L_2 dA = \frac{0!1!0!}{3!} 2A^e = \frac{1}{3} A^e, \tag{52}$$

$$\int_{A^e} L_3 dA = \frac{0!0!1!}{3!} 2A^e = \frac{1}{3} A^e,$$

şi înlocuind aceste relaţii în matricea forţelor nodale echivalente (51) aceasta devine,

$$[P_g] = pghA^e \begin{bmatrix} 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \end{bmatrix}^T .$$
(53)

MFF-AI.3.1.5 MATRICEA DE RIGIDITATE A STRUCTURII

Operaţia de asamblare se realizează cu ajutorul matricei de identificare, ale cărei elemente indică poziţia elementelor matricei de rigiditate a elementului finit in matricea de rigiditate a modelului numeric global. Elementele matricei de identificare se pot calcula cu relaţia,

$$[id]_{ne,necel} = [(id(i,j), j=1,necel), i=1,ne] = [(ngl \cdot cn(i,j1) - j2), j2 = ngl-1,0)j1=1,ne].$$

Acest calcul este realizat de subprogramul MAT_IDEN, din fig.11. Pentru cazul exemplului considerat, matricea de identificare rezultă,

$$[id]_{2,8} = [(id(i,j), j=1,8), i=1,2] = [(12345678\ ),(5634111291\ 0)].$$

Asamblarea matricelor de rigiditate ale elementelor finite se poate realiza într-o matrice de rigiditate pătratică cu ordinul (nec) sau într-un mod mult mai economic într-o matrice dreptunghiulară cu ordinul (nec, lband). Procesele de asamblare în cele două cazuri se pot realiza cu subprogramele ASAMB şi, respectiv,

```
Procedure MAT_IDEN(var ne,nne,ngl:integer;var id:mat1;var cn:mat2);
{type mat1=array[1..ne,1..necel] of integer;mat2=array[1..ne,
1..nne] of integer;}
var i,j,jj:integer;
begin
   for i:=1 to ne do
      for j:=1 to nne do
         for jj:= ngl-1 downto 0 do
            id[i,ngl*j-jj]:=ngl*cn[i,j]-jj;
end;
```

Fig. 11

```
Procedure ASAMB_B(var m,necel:integer;id:mat1;ke:mat3;K:mat4);
{type mat1=array[1..ne,1..necel] of integer;mat3=array[1..necel,
1..necel] of real;mat4=array[1..nec,1..lband] of real;}
{m = 1,2, ... ,ne reprezintă  numărul elementului finit a cărui
matrice se asamblează;}
var i,j,j1:integer;
begin
   for i:=1 to necel do
      for j:=1 to necel do
         if id[m,j] >= id[m,i]+1 then
            begin
               j1:=id[m,j]-id[m,i]+1;
               K[id[m,i],j1]:=K[id[m,i],j1]+ke[i,j];
            end;
end;
```

Fig. 12

ASAMB_B din fig. 12. Simultan cu asamblarea matricelor de rigiditate ale elementelor finite, se realizează şi asamblarea matricelor termenilor liberi.

MFF-AI.3.1.6 IMPLEMENTAREA CONDIŢIILOR LIMITĂ

Introducerea condiţiilor limită, geometrice, de tip Dirichlet, se va face astfel încât să se opereze cât mai puţine modificări în modelul numeric (sistemul de ecuaţii liniare final). Din cele trei metode prezentate în § 6.4, în cadrul acestui exemplu se foloseşte a doua metodă, care pentru cazul condiţiilor limită sub formă de deplasări nule, implică modificări minime în matricea modelului. Astfel, implementarea constă în înmulţirea elementelor diagonalei principale a matricei de rigiditate a modelului, corespunzătoare deplasărilor nodale cunoscute, cu un număr foarte mare, de exemplu 10^{15} (în funcţie de posibilitatea echipamentului de calcul) şi înlocuirea cu zero a coeficienţilor corespunzători ai matricei termenilor liberi.

Subprogramul COND_LIM, de implementare a condiţiilor limită folosind acest algoritm, este prezentat fig. 13. Aplicând această metodă de implementare, pentru gradele de libertate corespunzătoare deplasărilor nule (u_5, v_5, u_6, v_6), sistemul de ecuaţii al modelului din exemplu considerat devine compatibil determinat.

AFF-AI.3.1.7 REZOLVAREA SISTEMULUI DE ECUAŢII AL MODELULUI

Sistemul de ecuaţii liniare corespunzător modelului numeric general se poate realiza prin mai multe metode. În cazul exemplului considerat, se foloseşte metoda Gauss. Daca matricea de rigiditate obţinută în urma asamblării este completă, pentru rezolvare se utilizează subprogramul GAUSS (fig. 14) sau dacă matricea de rigiditate este de tip banda se utilizează subprogramul GAUSS_B

MFF-AI.3.1.8 DETERMINAREA MĂRIMILOR DEPENDENTE DE NECUNOSCUTELE MODELULUI

Cunoscând matricea deplasărilor nodale, obținută la etapa precedentă, se pot calcula deformațiile specifice şi tensiunile. Acestea se determina pentru fiecare element finit, cu relațiile:

$$\left[\varepsilon^{\theta}\right]=[B]\left[d^{\theta}\right],$$

$$\left[\sigma^{\theta}\right]=[E]\left[\varepsilon^{\theta}\right].$$

Subprogramele DEF si TENS, care realizează calculul acestor parametri, sunt prezentate în fig.13.

MFF-AI.3.1.9 DATE DE IEŞIRE

Rezultatele obținute in cadrul etapelor precedente, corespunzătoare modurilor si elementelor finite ale modelului numeric, se pot prezenta sub forma tabelara sau sub forma grafica. Volumul de date furnizate de calculator este foarte mare, fiind direct proporțional cu numărul de noduri si elemente finite generate la discretizare. Pentru problema considerata ca exemplu, lista rezultatelor conţine: deplasările nodale, deformațiile si tensiunile corespunzătoare elementelor finite.

Datorita simplității aproximărilor liniare de clasa C°, la nivelul deplasărilor, rezulta valori constante pentru tensiunile din interiorul elementelor finite. Pentru valori diferite ale deformațiilor in diferite puncte ale elementului finit, corespund diferite valori ale tensiunilor. De aceea, în vederea apropierii de valorile reale, ce pot determina valorile tensiunilor în diferite puncte ale elementului finit: centrul de masă, punctele Gauss, mijloacele laturilor etc.

Pentru a avea o privire de ansamblu asupra rezultatelor, acestea se pun sub forma unor diagrame (grafice). În acest fel, câmpul continuu, real, de deformații sau de tensiuni se substituie cu un câmp de valori care pot prezenta discontinuităţi la granița dintre elementele finite. Astfel, în cazul exemplului considerat, cu elemente finite liniare, câmpul deplasărilor este continuu (fig. 15, a) şi câmpul tensiunilor (fig. 15, b) este discontinuu.

Algoritmul întocmirii programului matlab pentru rezolvarea structurilor robotice (fig. 16) presupune:

- introducerea datelor (subprogram intrare)

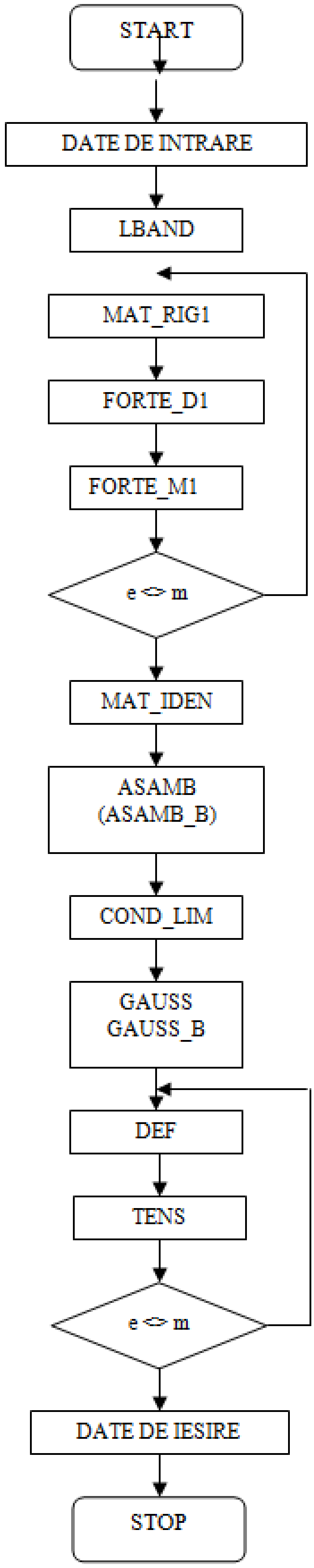


Fig. 16

```
Procedure COND_LIM(var K :mat4;d,el,Q :vect);
{type mat = array[1..nec,1..lband] of real;vect = array[1..nec];}
{matricea de rigiditate K este sub formă de bandă}
var i,j:integer;
begin
   for i:=1 to nec do
      if el[i] <> 0 then
         begin
            K[i,1]:=K[i,1]*1e10;
            Q[i]:=0;
         end;
end;
Procedure DEF(var ke :mat;B :mat6;ep :mat7;d :vect;id :mat1;i,nne,
necel :integer);
{type mat = array[1..necel,1..necel];mat6 = array[1..3,1..8];
mat7 = array[1..ne,1..3];vect = array[1..nec];mat1 = array[1..nne,
1..necel];}
{i este numărul curent al elementului finit}
var j,j1:integer;
begin
   for j:=1 to 3 do ep[i,j]:=0;
   for j1:=1 to 3 do
   for j:=1 to necel do
      ep[i,j1]:=ep[i,j1]+B[j1,j]*d[id[i,j]];
end;
Procedure TENS(ep,sigma :mat7;E :mat5;i :integer);
{type mat7 = array[1..ne,1..3];mat5 = array[1..neps,1..necel];}
{i este numărul curent al elementului finit}
var j,j1,k:integer;
begin
   for j:=1 to 3 do sigma[i,j]:=0;
   for j1:=1 to 3 do
   for k:=1 to 3 do
      sigma[i,j1]:=sigma[i,j1]+E[j1,k]*ep[i,k];
end;
```

Fig. 13

```
Procedure GAUSS(var n:integer;a:mat;r:vect);
{type mat=array[1..n,1..n] of real;vect=array[1..n] of real;}
var i,j,l,il:integer;
begin
   {triunghiularizare}
   for i:=1 to n-1 do
      begin
         if a[i,i] <> 0 then
            begin
               l:=i+1;
               r[i]:=r[i]/a[i,i];
               for j:=1 to n do a[i,j]:=a[i,j]/a[i,i];
               for il:=l to n do
                  begin
                     r[il]:=r[il]-a[il,i]*r[i];
                     for j:=l to n do
                        a[il,j]:=a[il,j]-a[il,i]*a[i,j];
                  end;
            end;
      end;
   r[n]:=r[n]/a[n,n];
   {retrosubstituţia}
   for i:=n-1 downto 1 do
      for j:=i+1 to n do r[i]:=r[i]-a[i,j]*r[j];
end;
```

Fig. 14

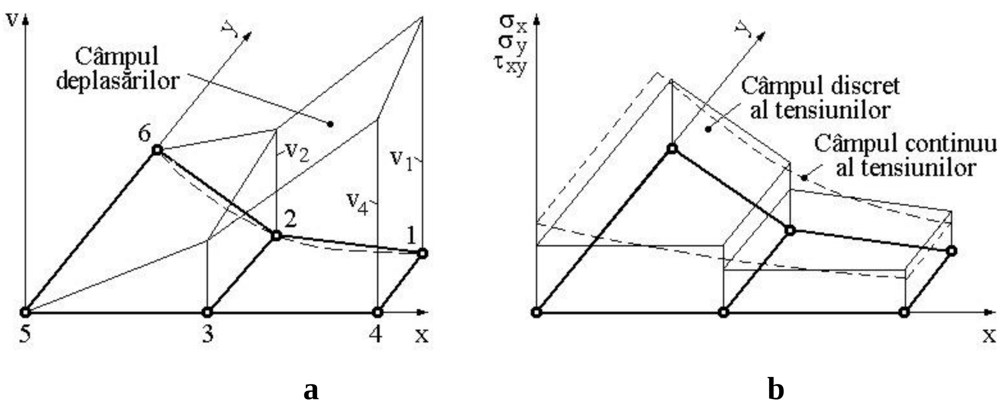


Fig. 15

- determinarea matricilor de rigiditate ale elementelor finite (subprogram mat_rig1);
- calculul forţelor nodale echivalente cu sarcinile distribuite (subprogram forte_d1);
- calculul forţelor nodale echivalente cu forţele masice (subprogram forte_m1);
- determinarea matricii de rigiditate a structurii (subprogram asamb);
- implementarea condiţiilor limită (cond_lim);
- rezolvarea sistemului de ecuaţii al modelului (gauss);
- determinarea mărimilor dependente de necunoscutele modelului (def şi tens);
- afişarea rezultatelor(subprogram ieşire).