

**Lucrarea nr.1**  
**UTILIZAREA PDE-ase2**  
**in rezolvarea problemelor de camp electromagnetic**

**1 Metode de rezolvare a sistemului de ecuații  
al câmpului electromagnetic**

Metodele de rezolvare a problemelor de câmp se clasifică în :

- metode analitice ;
- metode grafice ;
- metode numerice.

Cu ajutorul metodelor analitice se obțin soluții sub formă de funcții cunoscute având avantajul interpretării calitative a rezultatelor. Dezavantajul metodelor analitice constă în numărul relativ restrâns de configurații în care pot fi aplicate. Rezolvarea analitică implică adoptarea unor ipoteze ce nu întotdeauna sunt verificate practic.

Odată cu aparitia calculatoarelor, și în consecință a posibilităților de rezolvare numerică a problemelor de câmp cu o precizie foarte mare, metodele numerice de rezolvare au înlocuit metodele analitice. În principiu, metodele numerice se aplică pentru orice configurație cu o eroare ce depinde de metoda de calcul numeric aplicată.

Dintre aceste metode cele mai frecvent utilizate sunt :

- metoda diferențelor finite ;
- metoda bazată pe principii variaționale.

În general, soluțiile problemelor de câmp sunt funcții de toate coordonatele spațiale (probleme tridimensionale), iar volumul de calcul și capacitatea calculatorului este foarte mare. Din acest motiv problemele de câmp se reduc la două coordonate (probleme bidimensionale).

Ecuațiile câmpului electromagnetic sunt ecuații cu derivate parțiale asociate mărimilor scalare ale câmpului. Formularea variațională a ecuației cu derivate parțiale constă în asocierea unei funcționale  $J(u)$  egală cu integrala definită efectuată asupra domeniului  $D$ , integrandul fiind o formă pătratică în raport cu mărimea scalară  $U$  a câmpului, derivatele acesteia și coordonatele domeniului.

Expresia funcționalei  $J(u)$  se stabilește astfel încât anularea variatiei  $\frac{\partial J}{\partial U} = 0$  conduce la satisfacerea de către mărimea scalară  $U$  a ecuației cu derivate parțiale și a condițiilor pe frontieră.

**2 Modelarea numerică a ecuațiilor diferențiale  
prin metoda elementelor finite (MEF)**

Unul din principalele modele de cunoaștere a mediilor continue a fost și rămâne cel de aproximare prin discretizare. Acesta constă în descompunerea domeniului de analiză într-un număr finit de elemente discrete și aproximarea întregului prin ansamblul elementelor componente (principiul localizării acțiunilor fizice aplicat spre exemplu la definirea densității de masă). Avantajul esențial constă în lucrul cu un număr finit de elemente, precum și îmbunătățirea aproximării prin creșterea numărului de elemente.

**2.1. Metoda elementelor finite (MEF) constă în :**

1. Identificarea unui număr finit de puncte în domeniul mărimii  $U$  de aproximat și specificarea valorilor acestei mărimi în punctele considerate. Aceste puncte se numesc *noduri* și au coordonatele globale  $X_i \in D$ ,  $X_i = \overline{1, N}$ , unde  $N =$  numărul de noduri. Valorile mărimii  $U$  în aceste puncte se numesc *valori nodale*, notate cu  $U_i$ .

Domeniul de analiză este reprezentat ca o colecție finită de subdomenii interconectate de dimensiuni finite denumite *elemente finite*. Distribuția inițială a nodurilor se face începând cu frontieră domeniului până la trasarea completă a frontierei în segmente îmbinate. Apoi se distribuie (se împrăștie) nodurile interioare ca o plasă fină, având grijă să nu aibă aceeași dimensiune

segmentele din interior cu cele de pe frontieră. Fiecare nod i se asociază o valoare scalară egală cu lungimea celui mai mic segment conectat la nod. Pornind de pe frontieră se conectează nodurile în plasă triunghiulară până la unirea tuturor nodurilor. Se selectează astfel un număr finit de elemente (celule) fiecare element fiind cuprins între trei noduri. Fiecare nod este caracterizat local prin coordonatele locale  $x_j \in e$ , ce aparțin elementului finit e.

2. Se stabilesc relațiile de discretizare a domeniului de aproximare în elemente finite. Relațiile prezintă dependența între coordonatele locale ale nodurilor  $j \in e$  și coordonatele globale ale elementului  $X_i \in D$ . Dependența este dată prin matricea de incidență a coordonatelor locale la coordonatele globale ale nodurilor domeniului, matrice având coeficienții :

$$\Omega_{ji}^e = \begin{cases} 1, & \text{daca nodul local } j \text{ al elementului } (e) \text{ este incident la nodul globali} \\ 0, & \text{în caz contrar} \end{cases}$$

La nivelul elementului e se poate scrie:

$$[X_j]_{j \times 1} = [\Omega_{ji}]_{j \times i} \cdot [X_i]^T_{i \times 1}$$

Ecuațiile matriciale de la nivelul elementului e grupate pentru toate elementele finite conduc la un sistem de ecuații între coordonatele locale și globale cu dependență exprimată de *matricea de conexiuni* ce este [o matrice booleană de transformare.

3. Se stabilesc apoi *relațiile de asamblare* ale fiecarei elemente finite "e" în domeniul de aproximare D. Aceste relații prezintă dependența între coordonatele globale  $X_i$  și coordonatele locale  $x_j$ . Dependența este dată prin inversa matricii de incidență. Coeficienții acestei matrici notați  $\Lambda_{ij}^e$ , la nivelul elementului "e" au proprietatea :

$$\Lambda_{ij}^e = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$

La nivelul elementului finit, relațiile de asamblare a coordonatelor nodurilor globale de cele locale sunt :

$$[X_i]_{i \times 1} = [\Lambda_{ij}^e]_{i \times j} \cdot [X_j]^T_{j \times 1}$$

Se obține astfel un sistem de ecuații între coordonatele globale și locale, iar dependența este dată de *matricea de asamblare* denumită Jacobian-ul transformării coordonatelor.

4. Se asociază sistemului local de noduri și un sistem local de coordonate denumite coordonate naturale. Originea acestui sistem local de coordonate este în centrul de greutate al elementului caz în care se obțin :  $\xi \in [-1,1]$  respectiv  $\eta \in [-1,1]$  cu  $\xi = \frac{X - X_G}{a}$ , ( $2a = \text{lungimea elementului}$ ),  $\eta = \frac{Y - Y_G}{a}$ .

Alegând drept origini a sistemului local de coordonate nodurile elementului, se obține sistemul L-natural cu :

$$\begin{aligned} X &= N_1 \cdot X_1 + N_2 \cdot X_2 + N_3 \cdot X_3 \\ Y &= N_1 \cdot Y_1 + N_2 \cdot Y_2 + N_3 \cdot Y_3 \end{aligned}$$

unde : -  $N_1 + N_2 + N_3 = 1$

sau matricial :

$$X = \mathbf{N}^T \cdot \mathbf{X}$$

$$Y = \mathbf{N}^T \cdot \mathbf{Y}$$

relații ce arată caracterul de funcție de pondere al coordonatelor N exprimate prin relațiile conform figurii 1 :

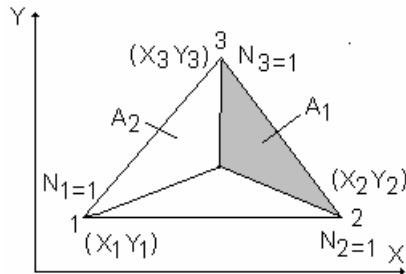


Fig. 1

$$N_1 = \frac{A_1}{A} = \frac{1}{2A} (a_1 + b_1 \cdot x + c_1 \cdot y)$$

$$N_2 = \frac{A_2}{A} = \frac{1}{2A} (a_2 + b_2 \cdot x + c_2 \cdot y)$$

$$N_3 = \frac{A_3}{A} = \frac{1}{2A} (a_3 + b_3 \cdot x + c_3 \cdot y)$$

unde :

$$\begin{cases} a_1 = X_2 \cdot Y_3 - X_3 \cdot Y_2 \\ b_1 = Y_2 - Y_3 \\ c_1 = X_3 - X_2 \\ A = \frac{1}{2} \cdot \begin{vmatrix} 1 & X_1 & Y_1 \\ 1 & X_2 & Y_2 \\ 1 & X_3 & Y_3 \end{vmatrix} \end{cases}$$

Ceilelalte coeficienți se obțin prin permutări ciclice ale indicilor.

## 2.2. Obținerea ecuațiilor cu elemente finite prin metoda rezidurilor ponderate

Rezidurile ponderate (The method of weighted residuals) constituie o metodă de obținere a soluțiilor aproximative pentru ecuațiile diferențiale liniare și neliniare. Prin această metodă se caută o soluție a problemei pe întreg domeniul de analiză.

Problemele descrise de ecuația (parabolică, hiperbolică sau eliptică) :

$$L(U) + S = 0$$

unde :

$L$  - operator diferențial de orice ordin ;

$S$  - sursa câmpului ;

sunt transformate în sisteme de ecuații nodale de forma :

$$\iiint_D L(U) \cdot \Psi_N \cdot dV + \iiint_D S \cdot \Psi_N \cdot dV = 0$$

unde  $\Psi_N$  este funcția de pondere asociată nodurilor  $n$  și volumului elementar  $dV$  ( $dV = dx \cdot dy$  în problemele bidimensionale sau  $r \cdot dr \cdot dz$  în bidimensional cilindric).

Considerând  $\hat{U}(\bar{r}, t) = U_0(\bar{r}, t) + \sum_{i=1}^n a_i(t) \cdot \Psi_i(r)$  o soluție aproximativă de încercare pe care o dorim convergentă spre soluția exactă  $U_{00}(\bar{r}, t)$  pe măsură ce  $n \rightarrow \infty$ . Soluția aproximativă se alege astfel încât să satisfacă în mod exact condițiile initiale și la limită ale problemei. Pentru  $i = 1, 2, \dots, n$ , Funcțiile  $\psi_i(\bar{r})$  constituie un sistem de funcții liniar independente pe domeniul  $D$ , iar  $a_i(t)$  sunt coeficienți ce urmează a fi determinați. Introducerea soluției

aproximativa în ecuație, problema nu mai este satisfăcută în mod exact, și determină un reziduu de calcul  $R(r,t,a_i)$ :

$$L(U) + S = R(r,t,a_i)$$

unde :

-  $R(r,t,a_i)$  este un reziduu de calcul.

Întrucât reziduu de calcul este nul ( $R = 0$ ) pentru soluția exactă, atunci  $R$  poate fi considerat o măsură a erorii de aproximare. Deoarece soluția exactă nu poate fi cunoscută, mărimea și distribuția rezidiului în domeniul  $D$  pot fi folosite pentru evaluarea acurateții soluției aproximative folosite.

Metoda rezidurilor ponderate caută să determine coeficienții  $a_i$  astfel încât eroarea  $R$  pe întreg domeniul considerat să fie cât mai mică. Aceasta se obține prin formarea unei medii ponderate a erorii pe întreg domeniul de analiză și impunerea condiției :

$$\int_D R(r,t,a_i) \cdot W_k \cdot dV = 0, k = \overline{1,n}$$

unde :

$W_k$  - reprezintă setul de funcții de pondere care depind de aceleași variabile independente ca și funcțiile  $\Psi_i$ .

Distribuția erorii pe domeniul  $D$  depinde de alegerea funcțiilor de pondere  $W_k$ , iar procedeele de selectare a acestor funcții sunt :

- *procedeul subdomeniilor* :

$$W_k = \begin{cases} 1 & \text{în domeniile elementului finit;} \\ 0 & \text{în rest;} \end{cases}$$

- *procedeul colocației* :

$$W_k = \delta \cdot (r - r_k)$$

- *procedeul celor mai mici pătrate* :

$$W_k = \frac{\partial R}{\partial a_k}$$

- *procedeul Galerkin* :

$$W_k = N_k \quad \text{unde : } N_k \text{ - sunt funcțiile de interpolare ale modelului cu elemente finite .}$$

Ecuatiile câmpului sunt descrise de sursele câmpului ce pot avea forme de exprimare locală prin divergență și rotor, asociate mărimilor câmpului. Aplicând procedeul Galerkin, integrala divergenței sau a rotorului la nivelul elementului finit se reduce prin folosirea formulei integrale a divergenței și rotorului astfel :

$$\iiint_{L^e} \operatorname{div}(\bar{F}) \cdot N_k \cdot dV = \oint_{\partial L^e} (\bar{n} \cdot \bar{F}) \cdot N_k \cdot dS - \iiint_{L^e} (\bar{F} \cdot \nabla \cdot N_k) \cdot dV$$

unde :  $dS$  - aria elementului finit.

$$\iiint_{L^e} \operatorname{rot}(\bar{F}) \cdot N_k \cdot dV = \oint_{\partial L^e} N_k \cdot (\bar{n} \times \bar{F}) \cdot dS + \iiint_{L^e} (\nabla \times N_k \cdot \bar{p}) \cdot \bar{F} \cdot dV$$

unde :  $\bar{p}$  - este un vector unitate (versor) perpendicular pe planul de calcul.

### 3. Descrierea problemei in PDE-ase2

Orice problemă de câmp se descrie într-un fișier descriptor de problemă în limbajul ușor de învățat natural al PDEase 2 printr-un editor de text sau procesor word capabil să creeze fișier text ASCII pe disc ,fișier ce descrie în mod matematic problema de rezolvat. Fișierul descriptor poate avea orice nume care trebuie urmat de extensia “*.pde*” sau “*.mfe*” Orice fișier trebuie să conțină următoarele secțiuni predefinite în PDEase.

<b>Title</b>	Este optional utilizat pentru a specifica TITLUI PROBLEMEI IN fisierele de ieșire
<b>Select</b>	Optional setata si poate controla eroarea de calcul , nr noduri etc
<b>Coordinates</b>	Se utilizeaza pentru specificarea sistemul de coordonate si numele variabilei spatiale independente.In sistem cartezian nu se defineste ci numai in <i>sistem cilindric (z,r)</i> prin declaratii de tipul : <i>x cylinder</i> caz in care orienteaza cilindrul pe axa x (z,r) <i>y cylinder</i> orienteaza vertical axa cilindrului (r,z)
<b>Variables</b>	Sunt declarate variabilele dependente ce corespund ecuațiilor diferențiale .Numele atribuit variabilelor trebuie să înceapă cu un caracter alfabetic
<b>Definitions</b>	Se declară constantele numerice ,functiile sau expresiile asignand numele
<b>Initial Values</b>	.Se definesc valorile initiale in problemele dependente de timp.Declaratia este optionala in problemele de regim permanent
<b>Equations</b>	.Se descriu ecuațiile diferențiale de rezolvat
<b>Constraints</b>	.Se specifică restricțiile la care-i supușă soluția (relații de conservarea a masei , sarcinii etc)
<b>Boundaries region 1</b>	Descrierea geometriei fizice a problemei ,a regiunilor din care-i realizată , cu specificarea valorilor pe frontieră, a surselor lineice și locale.
<b>Time</b>	Coordinatele temporale valoarea inferioară și limita precum și numărul de pași în problemele cu dependență de timp
<b>Monitors</b>	Afisează mărimile în timpul prelucrării
<b>Plots</b>	Afisează mărimile după prelucrare
<b>End</b>	

#### 3.1 Cuvinte rezervate in PDEase2

PDEase2 are un număr de cuvinte rezervate și anume

ABS	ARCTAN	CONSTANTS	COSH
ANGLE	BINTEGRAL	CONSTRAINTS	CROSS
ARC	BOUNDARIES	CONTOUR	CURL
ARCCOS	BY	COORDINATES	DEFINITIONS
ARCSIN	CENTER	COS	DEL2

DIFF	FOR	MONITORS	SURFACE
DIRECTION	GLOBALMAX	NATURAL	TABLE
DIV	GLOBALMIN	NORMAL	TAN
DOT	GRAD	OUTLINE	TANGENTIAL
ELEVATION	GRID	PI	TANH
ELSE	HISTORIES	PLOTS	THEN
END	HISTORY	POINT	TIME
ENDTIME	IF	RADIUS	TITLE
EQUATIONS	INITIAL	REGION	TO
ERF	INTEGRAL	SELECT	UPULSE
ERFC	LINE	SIGN	URAMP
EXCLUDE	LN	SIN	USTEP
EXP	LOAD	SINH	VALUE
FEATURE	LOG10	SQRT	VALUES
FINISH	MAGNITUDE	STAGE	VARIABLES
FIT1	MAX	START	VECTOR
FIT2	MIN	STARTTIME	

### 3.2 Expresii matematice în PDEase2

PDEase2 utilizează expresii matematice ce pot fi:

◆ Constante numerice

- - *întregi* (conțin maxim nouă cifre)
- - *constante zecimale* cu delimitatorul zecimal ”,” ce primește până la 308 cifre în ambele părți dar numai 15 cifre sunt considerate semnificative
- - *constante inginerești* formate din constante zecimale și separatorul de exponent ”E”

PDEase 2 utilizează toate *operațiile aritmetice* precum adunarea , scăderea ,înmulțirea ,împărțirea ridicarea la putere , extragerea rădăcinii

◆ Funcții matematice

- *Funcții continue*

ABS(arg)	Valoare absolută.
ARCCOS(arg)	Arc cosinus in radiani.
ARCSIN(arg)	Arc sinus in radiani.
ARCTAN(arg)	Arc tangent in radiani.
ATAN2 (arg1,arg2)	Returnează unghiul in radiani al căruia cosinus si sinus este proporțional cu arg1 si arg2. Poate fi definit si ATAN(arg1[arg2]), exceptând cazul când arg2=0.
COS(arg)	
COSH(arg)	
ERF(arg)	Funcția erorilor.
ERFC(arg)	Funcția erorilor complement.
EXP(arg)	Funcția exponențială.
LOG (arg)	Logaritm natural ; sinonim LN(arg).
LOG10(arg)	Log in baza 10.
LN(arg)	Logaritm natural
SIN(arg)	
SINH(arg)	
SQRT(arg)	Radical.
TAN(arg)	
TANH(arg)	
- *Funcții non-analitice*

MAX(arg1,arg2) funcția maximum necesită două argumente ,fiecare din ele putând fi o expresie sau o constantă. MAX este evaluată în orice punct și este egală cu cel mai mare dintre cele două argumente în fiecare punct.

**MIN(arg1,arg2)** funcția minimum necesită două argumente ,fiecare din ele putând fi o expresie sau o constantă. MIN este evaluată în orice punct și este egală cu cel mai mic dintre cele două argumente în fiecare punct.

**GLOBALMAX(arg)** funcția maximum global necesită un argument și este egală cu valoarea algebrică cea mai mare a argumentului pe domeniul problemei

**GLOBALMIN(arg)** funcția minimum global necesită un argument și este egală cu valoarea algebrică cea mai mică a argumentului pe domeniul problemei

**SIGN(arg)** funcția semn este egală cu 1 dacă argumentul este pozitiv și egală cu -1 dacă argumentul este negativ.

- **Funcții unitare**

**USTEP(arg)** funcția treaptă necesită un argument și este 0 pentru valori negative ale argumentului și 1 pentru valori pozitive ale argumentului.

**UPULSE(arg1,arg2)** funcția puls necesită două argumente și este 1 când argumentul 1 este pozitiv și argumentul 2 este negativ și este 0 în rest.

**URAMP(arg1,arg2)** funcția rampă necesită două argumente și este 0 când argument1 și argument 2 sunt negative, crește linear de la 0 la 1 când argument1 devine pozitive iar argument2 este negativ, și este 1 când argument1 și argument2 sunt pozitive. Dacă arg1 este mai mare decât arg2 URAMP evoluază până la USTEP(arg1).

- **Funcții și operatori vectoriali**

**CROSS(arg1,arg2)** produs vectorial necesită două argumente de tip vector și returnează o valoare scalară egală cu componenta vectorială a produsului vectorial arg1 și arg2 normală pe planul de calcul

**DOT(arg1,arg2)** produsul scalar necesită doi vectori argument și returnează o valoare scalară egală cu magnitudinea produsului arg1 și arg2. În PDEase2D  $\text{GRAD}(A)^{**2}$  este  $\text{DOT}(\text{GRAD}(A), \text{GRAD}(A)) = \text{GRAD}(V)^{**2}$

**MAGNITUDE(arg)** funcția magnitudine necesită un argument vector și returnează o valoare scalară egală cu magnitudinea argumentului vector.

**NORMAL(arg)** funcția componentă normală necesită un argument vector și returnează o valoare scalară egală cu componenta normală a argumentului vector. Are înțeles doar în raport cu un argument vector pe o limită internă sau externă bine definită

**TANGENTIAL(arg1)** funcția componentă tangențială necesită un argument vector și returnează o valoare scalară egală cu componenta tangențială a argumentului vector. Are înțeles doar în raport cu un argument vector pe o limită internă sau externă bine definită

**VECTOR(arg1,arg2)** funcția vector necesită două argumente și construiește un vector ale cărui argumente sunt arg1 și arg2

- **Funcții integrale**

**INTEGRAL(arg1,[ "arg2"])** funcția integrală de suprafață. INTEGRAL necesită unul sau două argumente. Când este folosit doar un argument valoarea funcției reprezintă integrala de suprafață a argumentului evaluat pe întreg domeniul spațial al problemei . când se folosesc două argumente al doilea trebuie să fie o regiune specificată în secțiunea BOUNDARIES a fișierului descriptor al problemei.

**BINTEGRAL(arg1,[ "arg2"])** funcția integrală de linie cere unul sau două argumente .Când este folosit un singur argument valoarea funcției este integrala liniară a argumentului peste toate limitele problemei (circulația pe frontieră) Când sunt două argumente al doilea trebuie să fie o cale specificată în secțiunea BOUNDARIES a fișierului descriptor al problemei.

### ◆ Operatori diferențiali

*Notăția PDEase2D      Notăția standard      Semnificația*

**DT(v)**                   $d v / d t$                   Derivată parțială în raport cu timpul

**DX(v)**                   $\frac{\partial v}{\partial x}$                   Derivată parțială de ordin I în raport cu x.

**DXX(v)**                 $\frac{\partial^2 v}{\partial x^2}$                   Derivată parțială de ordin II în raport cu x.

$$DY(v) \quad \frac{\partial v}{\partial y} \quad \text{Derivată parțială de ordin I în raport cu } y.$$

$$DYY(v) \quad \frac{\partial^2 v}{\partial y^2} \quad \text{Derivată parțială de ordin II în raport cu } y.$$

$$DEL2(v) \quad \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \quad \text{Laplace-anul } v.$$

♦ **Operatori diferențiali vectoriali**

$$DIV(F) \quad \text{Div}(F) \quad \text{Divergența unui vector.}$$

$$GRAD(v) \quad \text{Grad}(v) \quad \text{Gradientul unui scalar. Rezultatul este vector.}$$

CURL(F)    Curl(F)                      Rotorul unui vector . Rezultatul este magnitudinea scalară a unui vector normal pe planul de calcul.

CURL(V)    Curl(v)                      Rotorul componentei perpendiculare pe planul de calcul a unui vector axial . Rezultatul este un vector.

▪ **Expresiile in coordonate carteziene si cilindrice ale operatorilor vectoriali**  
*Operatorul de divergență*

In coordonate carteziene

$$\text{div}(\vec{F}) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y}$$

In coordonate cilindrice

$$\text{div}(\vec{F}) = \frac{1}{r} \frac{\partial r F_r}{\partial r} + \frac{1}{r} \frac{\partial F_\Phi}{\partial \Phi} + \frac{\partial F_z}{\partial z} \Rightarrow \frac{1}{r} \frac{\partial r F_r}{\partial r} + \frac{\partial F_z}{\partial z}$$

*Operatorul rotor a unui vector axial*

In cartezian

$$[\text{curl}(\vec{F})]_x = \frac{\partial F_z}{\partial y} \quad \text{and} \quad [\text{curl}(\vec{F})]_y = - \frac{\partial F_z}{\partial x}$$

In sistem cilindric

$$[\text{curl}(\vec{F})]_r = - \frac{\partial F_\Phi}{\partial z} \quad \text{and} \quad [\text{curl}(\vec{F})]_z = \frac{1}{r} \frac{\partial(r F_\Phi)}{\partial r}$$

*Operatorul rotor al unui vector perpendicular pe plan de calcul*

In coordonate carteziene

$$[\text{curl}(\vec{F})]_z = \frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y}$$

In coordonate cilindrice

$$[\text{curl}(\vec{F})]_\phi = \frac{\partial F_r}{\partial z} - \frac{\partial F_z}{\partial r}$$

♦ Specificatori de condiții limită

• *Segmente de frontieră*

*value(variable) = expression* Specificator ce impune valoarea variabilei în ecuația asociată ei la valoarea redată de expresie

*natural(variable) = expression* Specificator ce impune conservarea componentei normale în ecuația asociată prin divergență iar în ecuația asociată prin rotor impune conservarea componente tangențiale , la valoarea redată de expresie

*load(variable) = expression*

• *Puncte pe frontieră*

*Point value(var) = expression.* Specificator ce impune valoarea variabilei în ecuația asociată ei la valoarea redată de expresie în punctul indicat.

♦ **Expresii condiționale**

Fișierele descriptor de probleme ale PDEase 2 pot conține expresii condiționale prin instrucțiunile *if then else* ce sunt evaluate ca adevărate sau false . Expresiile condiționale constau în două expresii valorice legate de un operator condițional. Acești operatori sunt

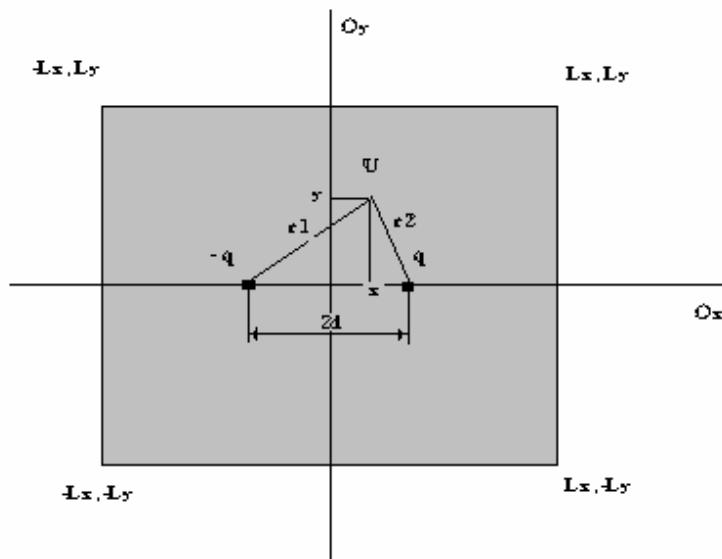
= egal; < mai mic; > mai mare; <= mai mic sau egal; >= mai mare sau egal; <> diferit de

### 3.3 Exemplu de aplicare a soft-ului PDE-ase2 .

Pentru rezolvarea unei probleme în PDEase se parcurg urmatorii pasi

1. Se lansează din programul **Macsyma**

2. Din bara programului *FILE* se deschide un fisier nou ce conține sectiunile predefinite ale PDEase-ului *Nu toate sectiunile predefinite sunt obligatorii* Spre exemplificare considerăm două sarcini egale și opuse dispuse la distanța 2d pe axa Ox . Urmărим să trasăm liniile de potential în domeniul (Lx,Ly) . În acest sens, se alege în sistemul de coordonate cartezian, dimensiunea domeniului în care trasăm liniile de potential



Potentialul  $U$  intr-un punct din interiorul domeniului are expresia  $U = \frac{1}{4\pi\epsilon} \left( \frac{q}{\sqrt{(x-d)^2 + y^2}} - \frac{q}{\sqrt{(x+d)^2 + y^2}} \right)$ . Pentru calculul acestui potential trebuie sa cunoastem :

-dimensiunea domeniului  $Lx$  respectiv  $Ly$

-distanta dintre sarcini  $d$  si dispozitiva geometrica a acestora

-valorile sarcinilor  $q$  respectiv  $-q$  si -permittivitatea mediului  $\epsilon$

In PDEase din sectiunile predefinite folosim :

**Title** ‘ exemplu’ { numele problemei }

**Definitions**  $Lx=1$   $Ly=1$   $d=0.2$   $eps=8.8e-12$   $q=1$   $c=1/(4*eps*pi)$  {dimensiuni domeniu }

$U= c* [q/sqrt(x-d)**2+y**2] - q/sqrt(x+d)**2+y**2)]$  {expresia matematica a potentialului}

**Boundaries** { in aceasta sectiune se descrie domeniul de analiza }

region 1

start(-Lx,-Ly)

line to (Lx,-Ly) to (Lx,Ly) to (-Lx,Ly) to finish

**Plots**

contour(  $U$  ) {conturul liniilor de potential}

surface( $U$ ) {suprafata de potential}

**End**