

Tema: Soluționarea problemelor de optimizare în MATLAB.

Scopul lucrării: de studiat principii de optimizare în pachetul de programe MATLAB.

Teoria:

Termenul de optimizare se referă la studierea problemelor în care se urmărește minimizarea (găsirea minimului) sau maximizarea (găsirea maximului) unei funcții reale prin căutarea sistematică în interiorul unei mulțimi admisibile. Această problemă poate fi reprezentată astfel:
Se dă: o funcție $f: A \rightarrow R$ unde $A \subseteq R$

Se cere: Să se găsească un element $x_0 \in A$ astfel încât $f(x_0) \leq f(x), \forall x \in A$ (minimizare), sau, astfel încât $f(x_0) \geq f(x), \forall x \in A$ (maximizare).

O astfel de formulare se numește problemă de optimizare. De obicei, mulțimea A este o submulțime a mulțimii R^n , specificată printr-un set de restricții de tip egalitate sau inegalitate pe care elementele lui A trebuie să le respecte. Dacă $A=R^n$, problema de optimizare se numește **fără restricții**.

Funcția f se numește **funcție obiectiv** sau **funcție cost**. Elementele lui A se numesc **soluții posibile** sau **soluții candidat**, iar o soluție posibilă care minimizează sau maximizează funcția obiectiv (în funcție de scopul problemei) se numește **soluție optimală**. Mulțimea A se numește spațiu de căutare.

A. Minimizări fără restricții

MATLAB-ul pune la dispoziția utilizatorului două funcții pentru optimizarea fără restricții. În continuare sunt prezentate două funcții MATLAB utilizate pentru rezolvarea problemelor fără restricții.

1. Funcția **fminunc**.

Este utilizată pentru găsirea minimului unei funcții reale fără a impune nici o restricție asupra soluției. Vom descrie modul de utilizare al acestei funcții folosind un exemplu.

Exemplu: Să se găsească minimul funcției de două variabile:

$$f(x_1, x_2) = e^{x_1}(4x_1^2 + 2x_2 + 4x_1x_2 + 2x_2 + 1)$$

Pentru aceasta mai întâi se creează un fișier MATLAB care returnează valorile funcției:

```
function f = functieObiectiv(x)
f = exp(x(1)) * (4*x(1)^2 + 2*x(2) + 4*x(1)*x(2) + 2*x(2) + 1);
```

Apoi, în alt fișier, se apelează funcția **fminunc**:

```
x0 = [-1, 1]; % Condiția inițială
options = optimset('LargeScale', 'off');
[x, valF, exitFlag, dateIesire] = fminunc(@functieObiectiv, x0, options);
```

Condiția inițială **x0** este necesară pentru inițializarea algoritmului de căutare a minimului. Acest program va genera soluția:

```
x =  
0.5000 -1.0000
```

Valoarea funcției în punctul **x** este returnată în variabila **valF**:

```
fval = 1.3030e-10
```

Indicatorul **exitFlag** dă informații despre convergența algoritmului. O valoare **exitFlag>0** înseamnă că a fost găsit un minim local. Variabila **dateIsire** este o structură ce conține informații detaliate despre optimizare. Ea include numărul de iterații efectuate (în elementul **iterations**), numărul de evaluări ale funcției (în elementul **funcCount**), valoarea finală a pasului de căutare (în elementul **stepsize**), norma infinit a gradientului soluției ce reprezintă o măsură de ordin unu a optimalității (în elementul **firstorderopt**), și tipul algoritmului utilizat pentru optimizare (în elementul **algorithm**).

Dacă există mai mult de un minim local condiția inițială **x0** afectează atât numărul de evaluări ale funcției cât și punctului în care se găsește soluția.

Parametrul **options** este utilizat pentru a seta anumiți parametri ai algoritmului de optimizare, cum ar fi eroarea maximă admisă pentru găsirea soluției, unde să fie utilizat gradientul furnizat de utilizator, numărul maxim de iterații, etc.

2. Funcția **fminsearch**.

Găsește minimul unei funcții reale multivariabile. Funcția poate fi apelată cu una din următoarele

sintaxe:

```
x = fminsearch(fun,x0)  
x = fminsearch(fun,x0,options)  
[x, valF] = fminsearch(fun,x0,options)  
[x, valF, exitFlag] = fminsearch(fun,x0,options)  
[x, valF, exitFlag, dateIsire] = fminsearch(fun,x0,options)
```

Semnificația parametrilor care apar în aceste sintaxe este aceeași cu ce de la funcția **fminunc**. Parametrul **fun** este o funcție MATLAB ce definește funcția obiectiv.

Exemplu: Să se găsească minimul funcției de o singură variabilă: $f(x) = \sin(x) + 3$. Se crează un fișier MATLAB ce returnează valorile funcției:

```
function f=fctSin(x)  
f=sin(x)+3;
```

Pentru a găsi valoarea minimă din apropierea lui se apelează funcția **fminsearch**
x=fminsearch(@fctSin,2)

Funcția **fminsearch** utilizează metoda simplex de căutare a minimului. Aceasta este o metodă directă de căutare ce nu utilizează gradienti analitici sau numerici. Funcția **fminsearch** este în general mai puțin eficientă decât decât **fminunc** pentru probleme de ordin mai mare decât 2, dar atunci când funcția obiectiv este discontinuă, **fminsearch** este mai robustă.

Observație: Dacă se dorește găsirea maximului unei funcții se utilizează aceleași funcții utilizate pentru găsirea minimului dar se va furniza algoritmului de optimizare funcția $-f$ în locul funcției f . Astfel, orice problemă de maximizare se transformă într-o problemă de minimizare.

B. Minimizarea funcțiilor de o variabilă pe un interval fixat

Acest tip de problemă poate fi sintetizat prin următoarea exprimare:

$\min f(x)$ astfel încât $x_1 \leq x \leq x_2$.

Funcția MATLAB corespunzătoare este **fminbnd**. Variabilele x_1 și x_2 reprezintă capetele intervalului pe care se dorește minimizarea funcției obiectiv f . Pentru utilizarea acestei funcții poate fi utilizată una din următoarele sintaxe:

```
x = fminbnd(fun,x1,x2)
x = fminbnd(fun,x1,x2,options)
[x,valF] = fminbnd(fun,x1,x2,options)
[x,valF,exitFlag] = fminbnd(fun,x1,x2,options)
[x,valF,exitFlag,dateTiesire] = fminbnd(fun,x1,x2,options)
```

Parametrii utilizați au aceeași semnificație cu cea specificată pentru funcțiile **fminsearch** și **fminunc**. Algoritmul de optimizare folosit de această funcție se bazează pe metoda "Secțiunii de Aur". Funcția obiectiv trebuie să fie continuă.

2

Exemplu: Să se găsească minimul funcției $f(x) = (x - 3)^2 - 1$ pe intervalul $[0, 5]$.

Se creează fișierul *fct.m* care conține funcția obiectiv: **function f = fct(x)**

```
f = (x-3).^2 - 1;
```

după care se apelează în alt program funcția **fminbnd**.

```
[x,valF] = fminbnd(@fct,0,5)
```

Acest program generează soluția

```
x
=3
```

```
valF
= -1
```

adică funcția f are un minim local egal cu -1 în punctul $x=3$.

Sacina:

1. Să se ruleze exemplele prezentate în lucrare.
2. Să se găsească minimul funcției $f(x_1, x_2) = x_1 + x_2$. Folosind funcția **mesh** să fie trasat graficul funcției $f(x_1, x_2) = x_1 + x_2$.

3. Să fie găsit maximul funcției $f(x) = \frac{\sin(x)}{x}$ pornind din punctul $x=1$. Aceeași problema x

pornind din punctul $x=6$. Să fie trasat graficul funcției **f** pe intervalul $[-20, 20]$.

4. Să fie găsit maximul funcției $f(x_1, x_2) = e^{x_1}(4x_1^2 + 2x_2 + 4x_1x_2 + 2x_2 + 1)$. Să fie reprezentat graficul funcției.

5. Să fie găsit maximul și minimul funcției $f(x_1, x_2) = x_1 * \exp(-x_1^2 - x_2^2)$. Folosind funcția **mesh** să fie trasat graficul funcției.

6. Să fie găsit maximul funcției $f(x_1, x_2) = \frac{\sin(\sqrt{x_1^2 + x_2^2})}{\sqrt{x_1^2 + x_2^2}}$. Folosind funcția **mesh** să fie trasat graficul funcției.