

LUCRAREA 1

OPTIMIZAREA CU MATLAB

1.1. Aspecte generale

Conducerea, în general, și a unor procese, în particular, este reprezentată printr-un șir succesiv de decizii, iar acestea reprezintă alegeri între mai multe alternative disponibile, cu scopul atingerii unuia sau mai multor obiective. Fundamentarea științifică a deciziei, indiferent de situația implicată, poate fi considerată ca un proces general, sistematic, constând din mai multe trepte: definirea problemei, evidențierea variantelor disponibile, evaluarea variantelor posibile din punct de vedere al consecințelor, fundamentarea și selectarea variantei optime.

Modelele de optimizare iau forma unor ecuații, care deși din punct de vedere matematic pot fi foarte complicate, au o exprimare formală foarte simplă:

$$\max (\min) F(X), \quad X = [x_1, x_2, \dots, x_n] \quad (1.1)$$

unde:

F – reprezintă valoarea criteriului (utilitatea) ce caracterizează funcționarea sistemului;

X – vectorul variabilelor controlate sau de optimizare (ale căror valori pot fi măsurate, modificate sau fixate arbitrar).

În plus, la modelul (1.1) se mai adaugă, de cele mai multe ori, una sau mai multe ecuații sau inecuații, care indică faptul că variabilele de optimizare trebuie să se încadreze între anumite limite. Restricțiile reprezintă relații de constrângere care trebuie satisfăcute pentru ca soluția să fie acceptabilă practic. O restricție de egalitate, exprimată implicit sau explicit are forma:

$$h_i(X) = 0, \quad i = 1, \dots, m \quad (1.2)$$

Teoretic, fiecare restricție de egalitate poate fi folosită pentru eliminarea unei variabile, lucru care uneori este foarte complicat sau chiar imposibil din punct de vedere algebric. O restricție de inegalitate are forma:

$$g_i(X) \leq (\geq) 0, \quad i = 1, \dots, p \quad (1.3)$$

Se poate menționa faptul că restricțiile de inegalitate au o importanță majoră în căutarea unei soluții mai bune pentru o problemă de optimizare.

Funcția F , numită și funcție obiectiv sau funcție scop, formează împreună cu restricțiile modelul sistemului care urmează a fi optimizat. Acesta reprezintă atât un model al sistemului cât și un model de decizie. Modelul respectiv poate fi folosit pentru a găsi exact sau aproximativ valorile optime ale variabilelor controlabile, adică acele valori care asigură cea mai bună performanță a sistemului.

Teoria optimizării s-a bazat, într-o primă etapă, pe dezvoltările matematice ale secolelor precedente, în care "lumea" era curată, alcătuită din funcții pătratice, cu restricții ideale și derivabilitate omniprezentă. Dar "lumea" reală este plină de discontinuități, spații și soluții perturbate, care se pretează cu dificultate proceselor iterative de determinare a soluțiilor perturbate, care se pretează cu dificultate proceselor iterative de determinare a soluțiilor optime. De aceea optimizarea bazată doar pe calcule matematice nu poate fi întotdeauna aplicată. Metodele principale folosite în cazul problemelor de optimizare se pot clasifica, pe baza tehnicilor de programare folosite, astfel:

- Metode bazate pe programarea liniară;
- Metode bazate pe programarea neliniară;
- Metode bazate pe programarea dinamică.

1.2. Optimizarea cu MatLab

Produsul MatLab este compus dintr-o serie de programe standard scrise pentru calcule matematice, modelare și simulare numerică, prelucrări de date statistice, reprezentări grafice asistate de calculator.

Cea mai importantă caracteristică a MatLab-ului este ușurința cu care poate fi extins. Prin aceasta, orice utilizator poate adăuga, propriile programe scrise în MatLab, la fișierele originale, dezvoltând aplicații specifice domeniului în care lucrează.

De asemenea, MatLab-ul include aplicații specifice, numite Toolbox-uri. Acestea sunt colecții extinse de funcții MatLab (fișiere *.m) care dezvoltă mediul de programare de la o versiune la alta, pentru a rezolva probleme din domenii variate. Structural, MatLab-ul este organizat sub forma unui nucleu de bază, cu interpretor propriu, în jurul căruia sunt construite toolbox-urile.

Din cadrul acestui grup de programe face parte și toolbox-ul Optim. Optim este o colecție de funcții utilizată pentru optimizarea liniară și neliniară, ce grupează următoarele tipuri de probleme:

- programare liniară și pătratică;
- determinarea minimului și maximului;
- funcții neliniare rezolvate în sensul celor mai mici pătrate;
- rezolvarea ecuațiilor neliniare;
- rezolvarea problemelor de minimax și semi-infinite;
- optimizarea multiobiectiv.

Fiecare componentă a setului de programe Optim este o funcție MatLab (fișier *.m). Acestea sunt:

1. programe pentru minimizarea funcțiilor neliniare:

- **fgoalattain** - probleme multi-obiectiv;
- **fmincon** - minimizare cu restricții;
- **fminbnd** - minimizare fără restricții, cazul unidimensional;
- **fminunc** - minimizarea fără restricții cu metode de tip gradient;
- **fminsearch** - rezolvarea ecuațiilor neliniare;

- **lsqnonlin** - aplicarea metodei celor mai mici pătrate;
- **fminimax** - căutarea soluției minimax;
- **fseminf** - minimizarea semi-inf.

2. programe pentru probleme de minim date matriciale:

- **linprog** - programare liniară;
- **quadprog** - programare pătratică;
- **lsqnonneg** - aplicarea metodei celor mai mici pătrate în condiții de nonnegativitate.

Setarea opțiunilor implicite de lucru pentru MatLab, precum și stabilirea configurației dorite de utilizator se realizează cu funcția `optimset`. Funcția `optimset` returnează vectorul `options`. Componentele acestuia sunt folosite de funcțiile de mai sus în totalitate sau parțial. În cele ce urmează se vor prezenta cele mai utilizate componente ale vectorului `options`.

- **Display** - dacă are valoarea "iter" atunci se afișează valorile de ieșire la fiecare iterație; dacă are valoare "final" se vor afișa rezultatele procesului de optimizare din ultima iterație;
- **MaxFunEvals** - numărul de evaluări al funcției obiectiv;
- **MaxIter** - numărul maxim de iterații;
- **TolFun** - toleranța de oprire pentru valoarea funcției obiectiv;
- **DiffMaxChange** - valoarea maximă de modificare pentru variabile în metoda diferențelor finite;
- **DiffMinChange** - valoarea maximă de modificare pentru variabile în metoda diferențelor finite;
- **GradObj** - gradientul funcției obiectiv definit de utilizator;
- **GradConstr** - gradientii pentru restricțiile neliniare definiți de utilizator;
- **Hessian** - Hessianul funcției obiectiv definit de utilizator;
- **LineSearchType** - indică folosirea algoritmului liniar de căutare;
- **TolCon** - toleranța de oprire la încălcarea restricțiilor.

Exemplu: options = optimset('Display','iter','TolFun',1e-8)

Funcția optimset returnează vectorul options în care parametrul Display este setat pe valoarea 'iter', iar parametrul TolFun este setat pe valoarea 1e-8.

1.3. Desfășurarea lucrării

1. Se studiază textul lucrării.
2. Se studiază sintaxele funcțiilor Matlab corespunzătoare rezolvării problemelor de programare liniară. Se vor identifica variabilele de intrare și variabilele de ieșire și structura acestora.
3. Se studiază funcțiile Matlab corespunzătoare rezolvării problemelor de programare neliniară. Se vor identifica variabilele de intrare și variabilele de ieșire, respectiv structura acestora.
4. Se vor identifica opțiunile implicite de lucru pentru MatLab, precum și stabilirea configurației dorite de utilizator.