

Tehnici de Optimizare

Tema de laborator

1. Introducere

Scopul laboratorului consta in studierea principalelor metode de cautare unidimensionala, pentru functii reale $f : [a_0, b_0] \rightarrow \mathbb{R}$ ale caror minime nu pot fi determinate analitic.

O prima varianta de rezolvare a acestor probleme de optimizare, consta in evaluari succesive ale functiei obiectiv intr-un numar fixat de puncte apartinand intervalului initial de incertitudine. Pentru aceasta este nevoie de doua ipoteze restrictive de lucru: functia trebuie sa fie continua si unimodala pe intervalul de definitie, i.e., sa admita un singur minim relativ pe intervalul respectiv. Notam $x^* = \inf_{x \in [a_0, b_0]} f(x)$. Fie doua puncte $u, v \in [a_0, b_0]$ astfel incat $a_0 \leq v < u \leq b_0$. Atunci sunt posibile doua situatii, ca o consecinta esentiala a ipotezelor de lucru:

$$\begin{cases} f(u) < f(v), x^* \in [v, b_0] \\ f(u) \geq f(v), x^* \in [a_0, u] \end{cases} \quad (1.1)$$

Deci, functia odata evaluata in doua puncte, intervalul de cautare a minimului se poate restrange. Scopul este ca dupa un numar fixat N de evaluari ale functiei sa obtinem un interval de incertitudine cat mai mic posibil pentru x^* .

O a doua varianta de rezolvare a problemei gasirii minimelor functiilor reale, consta in aproximarea pe portiuni a functiei obiectiv cu functii polinomiale. Se porneste de la ipoteza ca functia obiectiv are un anumit grad de netezime. Se genereaza o curba neteda ce trece prin anumite puncte ale functiei obiectiv, evaluate a priori, pentru a determina o estimare a punctului de minim.

2. Descrierea problemei

Exista o structura fundamentala ce sta la baza aproape tuturor algoritmilor de descrestere:

- Se porneste dintr-un punct initial;
- Se determina o directie de deplasare, conform cu o regula fixa;
- Se efectueaza deplasarea in acea directie spre un minim relativ al functiei obiectiv pe directia respectiva;
- Din noul punct se determina o noua directie si procesul se repeta.

Procesul de determinare a minimului de-a lungul unei linii date se numeste *cautare liniara*. Pentru functiile ce nu pot fi minimizate analitic, rezolvarea problemei de optimizare consta in elaborarea unor algoritmi de cautare eficienta a minimului pe intervalul de definitie al functiei obiectiv.

2.1. Metoda sirului lui Fibonacci

In cadrul procesului de cautare efortul general de calcul este puternic influentat de efortul de calcul al evaluarii functiei obiectiv in diferite puncte alese. Metoda sirului lui Fibonacci realizeaza un interval de incertitudine finala impusa, evaluand functia obiectiv in cat mai putine puncte. Expresia recurenta a sirului lui Fibonacci este

$$F_N = F_{N-1} + F_{N-2}, \quad \text{cu } F_0 = F_1 = 1, \quad (2.1)$$

Consideram ca pentru iteratia k intervalul de incertitudine este $[a_k, b_k]$. Alegerea punctelor intermediare se face astfel:

$$\begin{aligned} u_k &= a_k + \frac{F_{N-k-1}}{F_{N-k+1}}(b_k - a_k), \\ v_k &= a_k + \frac{F_{N-k}}{F_{N-k+1}}(b_k - a_k). \end{aligned} \quad (2.2)$$

In urma comparatiei dintre valorile functiei obiectiv in cele doua puncte intermediare intervalul de incertitudine devine fie $[a_k, v_k]$ fie $[u_k, b_k]$. Lungimea acestor intervale este aceeasi:

$$\begin{aligned} v_k - a_k &= \frac{F_{N-k}}{F_{N-k+1}}(b_k - a_k), \\ b_k - u_k &= b_k - a_k - \frac{F_{N-k-1}}{F_{N-k+1}}(b_k - a_k) = \frac{F_{N-k+1} - F_{N-k-1}}{F_{N-k+1}}(b_k - a_k) = \frac{F_{N-k}}{F_{N-k+1}}(b_k - a_k). \end{aligned} \quad (2.3)$$

Raportul de contractie va fi:

$$\frac{b_{k+1} - a_{k+1}}{b_k - a_k} = \frac{F_{N-k}}{F_{N-k+1}} \quad (2.4)$$

Teorema. Metoda Fibonacci asigura realizarea unei lungimi minime posibile a intervalului $\frac{F_{N-k+1}}{F_N}(b_0 - a_0)$ pentru orice N , prin evaluarea functiei in N puncte.

Procedura se incheie cu ultimele doua puncte, in mijlocul penultimului interval de cautare. Acest pas nu necesita nicio evaluare a functiei.

Algoritmul 2.1. Algoritmul metodei sirului lui Fibonacci

Fixeaza $L > 0$ (lungimea finala a intervalului de incertitudine);

Determina N din conditia $F_N > \frac{b_0 - a_0}{L}$;

Calculeaza $u = a_0 + \frac{F_{N-2}}{F_N}(b_0 - a_0)$ si $v = a_0 + \frac{F_{N-1}}{F_N}(b_0 - a_0)$;

Calculeaza $f(u)$ si $f(v)$;

Pentru $k=1:N-3$

Daca $f(u) < f(v)$
 $b_0 \leftarrow v$;
 $v \leftarrow u$;
 $u \leftarrow a_0 + \frac{F_{N-k-2}}{F_{N-k}}(b_0 - a_0)$;
Altfel
 $a_0 \leftarrow u$;
 $u \leftarrow v$;
 $v \leftarrow a_0 + \frac{F_{N-k-1}}{F_{N-k}}(b_0 - a_0)$.

2.2. Metoda sectiunii de aur

Metoda sectiunii de aur constituie varianta limita a metodei precedente. Ea permite imbunatatirea preciziei fara reluarea calculelor. Pentru a reduce numarul de evaluari ale functiei, unul din punctele in care se evalueaza functia obiectiv la pasul curent ar trebui sa coincida cu unul din punctele de test de la pasul anterior. Problema consta in a stabili posibilitatea unei asemenea suprapuneri. Acest deziderat se obtine prin impunerea urmatoarelor conditii:

- Lungimea intervalului in oricare iteratie $b_{k+1} - a_{k+1}$ sa fie aceeaasi indiferent de rezultatul comparatiei dintre $f(u)$ si $f(v)$. Prin urmare este necesar ca $b_k - u_k = a_k - v_k \quad \forall k$ si deci daca consideram

$$u_k = a_k + (1 - \alpha)(b_k - a_k) \quad (2.5)$$

rezulta

$$v_k = a_k + \alpha(b_k - a_k) \quad \alpha \in (0, 1). \quad (2.6)$$

Prin urmare,

$$b_{k+1} - a_{k+1} = \alpha(b_k - a_k) \quad (2.7)$$

si deci raportul de contractie α ramane constant in tot procesul de cautare.

- La noua iteratie trebuie fie ca u_{k+1} sa coincida cu v_k fie ca v_{k+1} sa coincida cu u_k .

Algoritmul 2.2. Algoritmul metodei sectiunii de aur

Fixeaza $L > 0$ (lungimea finala a intervalului de incertitudine);

Fixeaza $\alpha = \frac{\sqrt{5}-1}{2}$ (numarul de aur);

Calculeaza $u_1 = a_0 + (1 - \alpha)(b_0 - a_0)$ si $v_1 = a_0 + \alpha(b_0 - a_0)$;

Calculeaza $f(u_1)$ si $f(v_1)$;

$k=1$;

Cat timp $b_k - a_k > L$

Daca $f(u_k) < f(v_k)$
 $a_{k+1} \leftarrow a_k$ si $b_{k+1} \leftarrow v_k$;
 $u_{k+1} = a_{k+1} + (1 - \alpha)(b_{k+1} - a_{k+1})$;
 Altfel
 $a_{k+1} \leftarrow u_k$ si $b_{k+1} \leftarrow b_k$;
 $v_{k+1} = a_{k+1} + \alpha(b_{k+1} - a_{k+1})$;
 $k = k + 1$;

Calculeaza $f(u_{k+1})$;
 $v_{k+1} \leftarrow u_k$ si $f(v_{k+1}) \leftarrow f(u_k)$;
 Altfel
 $a_{k+1} \leftarrow u_k$ si $b_{k+1} \leftarrow b_k$;
 $u_{k+1} \leftarrow v_k$ si $f(u_{k+1}) \leftarrow f(v_k)$;
 $v_{k+1} = a_{k+1} + \alpha(b_{k+1} - a_{k+1})$;
 Calculeaza $f(v_{k+1})$;
 $k=k+1$.

2.3. Metode de cautare prin interpolare

Se exploateaza faptul ca functiile obiectiv au zone netede. Functia obiectiv este aproximata pe portiuni cu functii polinomiale. Exista mai multe tehnici dezvoltate, in functie de cate puncte anterioare sunt folosite pentru a determina interpolarea, in functie de cunoasterea derivatelor si a valorilor functiei in diverse puncte. Toate aceste metode au ordin de convergenta supraunitar.

Interpolare patratica

Fie o functie $f: I \rightarrow \mathbb{R}$. Sunt date 3 puncte x_1, x_2 si x_3 si valorile functiei in aceste puncte $f(x_1) = f_1, f(x_2) = f_2, f(x_3) = f_3$. Se construiesc un polinom de gradul 2 ce trece prin aceste puncte:

$$q(x) = \sum_{i=1}^3 f_i \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)} \quad (2.8)$$

si apoi se determina urmatorul punct x_4 ca fiind extremul lui $q(x)$:

$$x_4 = \frac{1}{2} \frac{b_{23}f_1 + b_{31}f_2 + b_{12}f_3}{a_{23}f_1 + a_{31}f_2 + a_{12}f_3} \quad (2.9)$$

unde $a_{ij} = x_i - x_j$ si $b_{ij} = x_i^2 - x_j^2$. Se repeta procedura pentru noul set de puncte: (x_2, x_3, x_4) .

Metodele de interpolare nu sunt global convergente in formele lor pure. Pentru a asigura convergenta globala a procedurii se impune ca punctele initiale sa respecte conditiile: $x_1 < x_2 < x_3$ astfel incat $f(x_1) \geq f(x_2) \leq f(x_3)$. Cu alte cuvinte, valoare din mijloc este mai mica decat valoarea in oricare din capete. Un interpolator patratic care trece prin astfel de puncte va avea un minim in intervalul $[x_1, x_3]$. Punctul x_4 se obtine din relatia (2.9). Presupunem in continuare ca $x_2 < x_4 < x_3$ si avand in vedere ca f este unimodala vom avea doua posibilitati:

- $f(x_4) \leq f(x_2)$,
- $f(x_2) < f(x_4) \leq f(x_3)$.

In ambele cazuri, se poate determina un nou set de 3 puncte format din x_4 si 2 din punctele anterioare:

- $(\bar{x}_1, \bar{x}_2, \bar{x}_3) = (x_2, x_4, x_3),$
- $(\bar{x}_1, \bar{x}_2, \bar{x}_3) = (x_1, x_2, x_4).$

Cu acest set nou de puncte se gaseste un nou interpolator patratic si procedeul se reia.

Algoritmul 2.3. Cautare prin interpolare polinomiala

Fixeaza $x_1, x_2, x_3 \in I$ (punctele de initializare);

Fixeaza $\varepsilon > 0$ (distanța maxima între minimul gasit si minimul real);

Calculeaza x_4 cu formula (2.9);

Cat timp $f(x_4) - \min_{i=1,2,3} f(x_i) > \varepsilon$

Actualizeaza setul de puncte (x_1, x_2, x_3) (se va abandona acel x_k pentru care $f(x_k) = \max_{i=1,2,3} f(x_i)$);

Calculeaza noul punct x_4 cu formula (2.9);

Calculeaza $f(x_4)$;

Interpolare cubica

Fie o functie $f: I \rightarrow \mathbb{R}$. Se dau doua puncte $x_{k-1}, x_k \in I$ si valorile functiei respectiv ale derivatei de ordin 1 in aceste puncte: $f'(x_{k-1})$, $f(x_{k-1})$, $f'(x_k)$ si $f(x_k)$. Punctul din iteratia urmatoare corespunzatoare procedurii de cautare se poate gasi dupa legea de recurenta:

$$x_{k+1} = x_k - (x_k - x_{k-1}) \left[\frac{f'(x_k) + u_2 - u_1}{f'(x_k) - f'(x_{k-1}) + 2u_2} \right], \quad (2.10)$$

unde

$$u_1 = f'(x_{k-1}) + f'(x_k) - 3 \frac{f(x_{k-1}) - f(x_k)}{x_{k-1} - x_k},$$

$$u_2 = \left[u_1^2 - f'(x_{k-1}) f'(x_k) \right]^{\frac{1}{2}}.$$

Algoritmul 2.4. Cautare prin interpolare cubica

Fixeaza $x_1, x_2 \in I$ (punctele initiale);

Fixeaza $\varepsilon > 0$ (distanța maxima între minimul gasit si minimul real);

Calculeaza $f'(x_1)$, $f(x_1)$, $f'(x_2)$ si $f(x_2)$;

Calculeaza x_3 cu formula (2.10);

Cat timp $f'(x_3) > \varepsilon$

Actualizeaza setul de puncte $(\bar{x}_1, \bar{x}_2) \leftarrow (x_3, x_k)$, unde x_k este acel punct pt care $f(x_k) = \min(f(x_1), f(x_2))$;

Calculeaza noul punct x_3 cu formula (2.10);

Calculeaza $f'(x_3)$ si $f(x_3)$;

3. Sarcini de lucru

1) Implementati algoritmi 2.1 si 2.2 pentru functiile:

a) $f(x) = \frac{x^2 - 3x}{x^2 - 2x + 2}$;

b) $f(x) = x^2 - \sin x$;

c) $f(x) = e^x - \sin 2x$ pe intervalul $[-1, 1]$;

d) $f(x) = e^{-\frac{1}{9}x} \sin x$ pe intervalul $[-3, 1]$;

e) $f(x) = x^5 - \frac{4}{5}x + \frac{1}{2}$ pe intervalul $[0, 1]$;

Impuneti un interval final de lungime 10^{-4} . Afisati numarul de iteratii. Figurati grafic functiile in Matlab. Comparati rezultatele obtinute cu rutinele deja existente in Matlab: **fminbnd** si **fminsearch**.

2) Modificati algoritmul de la punctul 1. (doar pentru 2.1) astfel incat lungimea intervalului final sa fie dictata de un numar impus de elemente ale sirului lui Fibonacci. Modificati acest numar pana cand obtineti aproximativ acelasi numar de iteratii ca la punctul 1.

3) Implementati algoritmi 2.3 si 2.4 pentru functiile de la punctul 1.

4) Figurati grafic parcursul fiecarui algoritm pe diferite functii.