

CAPITOLUL IV

ELEMENTE DE PROGRAMARE NELINIARA

§ 1. Introducere

Caracteristic unei probleme de *programare liniară* este faptul că toate funcțiile implicate în ea – funcția *obiectiv* și *restricții* – sunt *liniare*. Deși *ipotezele de liniaritate* au loc în numeroase situații practice tot atât de frecvent ale nu sunt îndeplinite. De fapt, mulți economiști teoreticieni au constatat că un anumit *grad de neliniaritate* este *regula și nu excepția* în problemele de *planificare economică*.

Există deci o puternică *motivație economică* pentru studiul problemelor de *programare neliniară* a căror *formă canonică de prezentare* este:

$$(P) \left\{ \begin{array}{l} \text{Sa se determine } x^* = (x_1^*, x_2^*, \dots, x_n^*) \\ \text{care minimizeaza functia obiectiv :} \\ z = f(x_1, x_2, \dots, x_n) \\ \text{cu satisfacerea restrictiilor :} \\ g_i(x_1, x_2, \dots, x_n) \leq 0 \quad i = 1, 2, \dots, m \\ \text{si a conditiilor de nenegativitate :} \\ x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0 \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} \min f(x) \quad x \in R^n \\ g_i(x) \leq 0 \quad i = 1, 2, \dots, m \\ x \geq 0 \end{array} \right.$$

Dacă în cazul liniar ($\Leftrightarrow f$ și g_i sunt funcții liniare) există (cel puțin) o metodă generală de rezolvare – de exemplu metoda *simplex* – în cazul neliniar nu există o asemenea metodă. Totuși progrese substanțiale au fost făcute în unele cazuri speciale prin impunerea unor condiții asupra funcțiilor f și g_i . Aria acestor cazuri speciale este destul de mare astfel că nu ne vom putea opri decât asupra unora mai relevante.

§ 2. Neliniaritatea în modelarea proceselor economice. Câteva exemple

Să considerăm *problema firmei* al cărei obiectiv este determinarea unui *program de producție* astfel încât:

- necesarul de resurse pentru susținerea programului să se încadreze în disponibile *date*;
- *profitul total*, rezultat din *vânzarea* bunurilor produse să fie *maxim*.

În multe situații practice, *prețurile* și *costurile unitare de fabricație* pot fi considerate *constante* astfel că și profiturile unitare vor fi la fel. În consecință, în aceste cazuri funcția obiectiv, care reprezintă profitul total, va fi *liniară*:

$$f(x_1, x_2, \dots, x_n) = P_1 x_1 + P_2 x_2 + \dots + P_n x_n$$

(x_j reprezintă cantitatea produsă și vândută din bunul j ; P_j este profitul unitar corespunzător bunului j)

Nu întotdeauna tot ceea ce se produce dintr-un bun se poate și vinde la un anumit preț. Apare așa numita problemă a *elasticității prețului*: cantitatea de marfă vândută se află într-o *relație inversă* cu prețul cerut așa cum arată *curba preț – cerere* (fig. 2.1a)

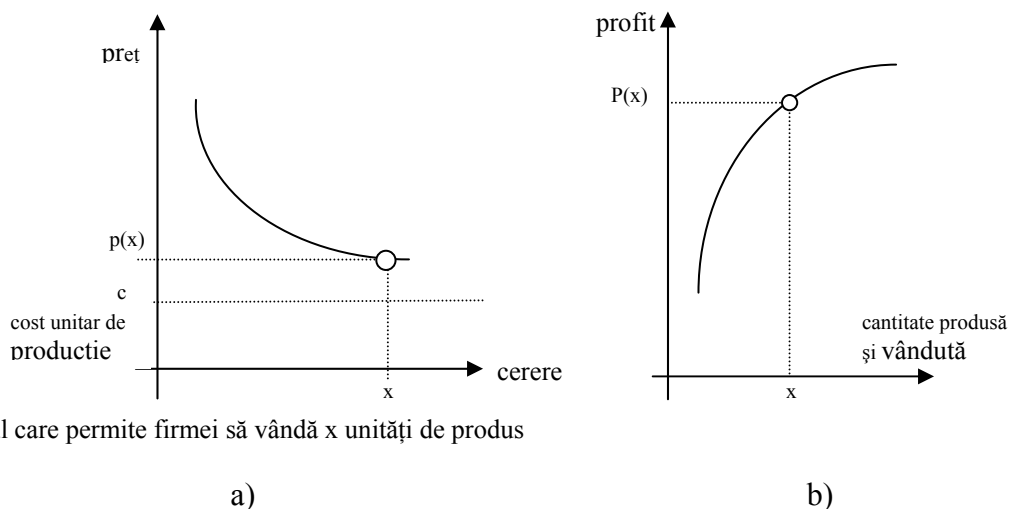


Figura 2.1

Dacă c este *costul unitar de producție*, cost pe care îl presupunem – pentru simplificarea expunerii – fix, atunci profitul firmei, rezultat din producerea și vânzarea cantității x este dat de funcția *neliniară* (vezi fig. 2.1b):

$$P(x) = x \cdot p(x) - c(x)$$

Dacă fiecare din produsele firmei are o asemenea funcție profit, notată $P_j(x_j)$, profitul total va fi exprimat prin funcția neliniară:

$$f(x_1, x_2, \dots, x_n) = \sum_{j=1}^n P_j(x_j)$$

O altă sursă de neliniarități în funcția obiectiv o constituie variația *costului marginal* - pentru producerea a încă unei unități dintr-un produs - în funcție de nivelul producției. Acest cost marginal poate să scadă în unele situații (ca urmare a trecerii la producția de serie, al acumulării experienței și al perfecționării procesului de producție) iar în altele poate să crească (ore suplimentare, utilizarea în regim de urgență a unor capacități de producție mai costisitoare pentru satisfacerea unor cereri imediate)

Neliniaritatea poate apare și în restricții într-o manieră asemănătoare. De exemplu, dacă există o *restricție bugetară* asupra costului producției, relația corespunzătoare va fi cu siguranță neliniară în cazul în care *costul marginal al producției este variabil*.

Pentru restricțiile privitoare la alte tipuri de resurse, neliniaritatea apare ori de câte ori *consumurile nu sunt direct proporționale cu nivelele de producție*.

Exemplul 2.1 În *problema transporturilor* se urmărește determinarea unui program pentru transportul unui produs de la diferite surse la diverși consumatori, cunoscându-se cantitățile disponibile și cererile, astfel încât costul total al transportului să fie minim. În programarea liniară, costul transportului unei unități de produs de la o anumită sursă la o anumită destinație a fost considerat *constant*, independent de cantitatea transportată. De multe ori se întâmplă ca la cantități mari să se acorde la transport anumite reduceri; în aceste situații, costul marginal al transportului unei unități suplimentare tinde să scadă și ca o consecință costul $C(x)$ al transportării cantității x este dat de o funcție neliniară. Să analizăm datele din fig. 2.2 a) și b).

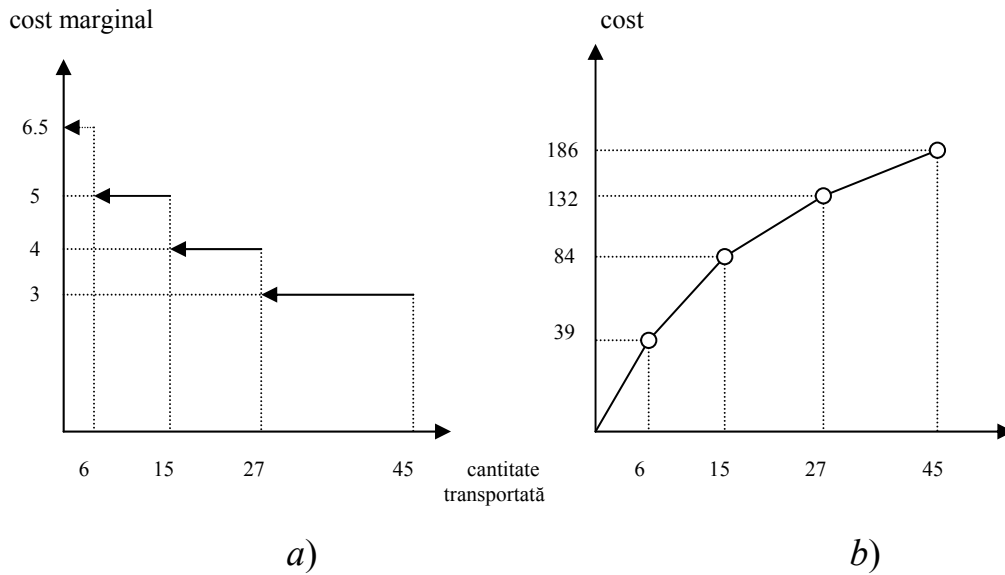


Figura 2.2

Fie x cantitatea transportată și $C(x)$ costul transportării cantității x .

- dacă $0 \leq x \leq 6$ costul unitar de transport este de 6.5 u.m. deci $C_1(x) = 6.5x$ u.m.;
- dacă $6 \leq x \leq 15$, 6 unități vor fi transportate cu costul 6.5 u.m. per unitate iar următoarele cu numai 5 u.m. per unitate astfel că :

$$C_2(x) = 6.5 \times 6 + 5.(x - 6) = 9 + 5.x$$
- dacă $15 \leq x \leq 27$, 15 unități vor fi transportate cu costul $C_2(15) = 84$ iar următoarele cu numai 4 u.m. per unitate. Costul total va fi :

$$C_3(x) = 84 + 4.(x - 15) = 24 + 4.x$$
- dacă $27 < x \leq 45$, 27 unități vor fi transportate cu costul $C_3(27) = 132$ iar următoarele cu 3 u.m. per unitate de unde costul total:

$$C_4(x) = 132 + 3.(x - 27) = 51 + 3.x$$

Prin urmare, costul $C(x)$ al transportării a x unități este dat de funcția:

$$C(x) = \begin{cases} 6.5x & 0 \leq x \leq 6 \\ 9 + 5.x & 6 < x \leq 15 \\ 24 + 4.x & 15 < x \leq 27 \\ 51 + 3.x & 27 < x \leq 45 \end{cases}$$

care este *neliniară* dar "*liniară pe porțiuni*". Din fig. 4.2b) rezultă că pantele porțiunilor liniare ale graficului funcției $C(x)$ sunt chiar costurile marginale evidențiate în fig. 4.2a).

Dacă fiecare cuplu (sursă i , destinație j) are o funcție cost similară, așa încât costul transportului a x_{ij} unități de la i la j este dat de funcția neliniară $C_{ij}(x_{ij})$ atunci costul total este reprezentat de funcția de asemenea neliniară;

$$f(x_1, x_2, \dots, x_n) = \sum_{i,j} C_{ij}(x_{ij})$$

Aceste considerații nu modifică restricțiile problemei de transport care rămân liniare:

$\sum_{j=1}^n x_{ij} = a_i \quad i = 1, 2, \dots, m \Leftrightarrow$ suma cantităților livrate de sursa i nu depășește disponibilul său;

$\sum_{i=1}^m x_{ij} = b_j \quad j = 1, 2, \dots, n \Leftrightarrow$ suma cantităților primite de consumatorul j acoperă cererea sa.

Exemplul 2.2 La terminalul unei conducte petroliere situat într-un port sosesc vasele A,B,C pentru a fi încărcate. Vasul A trebuie încărcat cu 15 mii tone în maximum 48 de ore, vasul B trebuie încărcat cu 20 mii tone în maximum 60 de ore iar vasul C trebuie încărcat cu 45 mii tone în maximum 70 de ore. (timpii de staționare pentru încărcare se numesc timpii de stalii și sunt prevăzuți în contractul de transport; orice depășire a acestor timpii (contrastalii) se penalizează, penalizarea depinzând de tipul navei etc) Terminalul dispune de pompe și conducte care însumează o capacitate de încărcare de 1200 tone pe oră. Se pune problema de a stabili ce debit trebuie afectat fiecărei nave astfel încât ele să fie încărcate în cel mai scurt timp.

Dacă se notează cu y_1, y_2, y_3 debitul repartizat vaselor A,B,C și cu x_1, x_2, x_3 numărul de ore necesar încărcării lor se obține ușor următorul program neliniar:

$$\left\{ \begin{array}{l} (\min) f = x_1 + x_2 + x_3 \\ x_1 y_1 = 15000 \quad x_2 y_2 = 20000 \quad x_3 y_3 = 45000 \\ y_1 + y_2 + y_3 \leq 1200 \\ x_1 \leq 48, x_2 \leq 60, x_3 \leq 70 \\ x_j, y_j \geq 0 \quad j = 1, 2, 3 \end{array} \right.$$

Eliminând y_1, y_2, y_3 rezultă modelul mai simplu:

$$\left\{ \begin{array}{l} (\min) f = x_1 + x_2 + x_3 \\ \frac{15000}{x_1} + \frac{20000}{x_2} + \frac{45000}{x_3} \leq 1200 \\ x_1 \leq 48, x_2 \leq 60, x_3 \leq 70 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 \end{array} \right.$$

§ 3. Dificultăți cauzate de neliniaritate

Considerăm problema de optimizare:

$$(P) \left\{ \begin{array}{l} \min f(x) \quad , \quad x \in R^n \\ g(x) \leq 0 \quad , \quad g(x) = (g_1(x), g_2(x), \dots, g_m(x)) \\ x \geq 0 \end{array} \right.$$

a cărei mulțime de soluții admisibile o notăm cu A:

$$A = \{x \in R^n \mid g(x) \leq 0 \Leftrightarrow g_1(x) \leq 0, g_2(x) \leq 0, \dots, g_m(x) \leq 0; x \geq 0\}$$

(P) se rescrie:

$$\text{Să se determine } x^* \in A \text{ cu proprietatea: } f(x^*) = \min\{f(x), x \in A\}$$

Este cunoscut faptul că dacă (P) este un program *liniar* (adică f și g_1, g_2, \dots, g_m sunt funcții *liniare*) atunci mulțimea A este *convexă* și mai mult chiar *poliedrală* (intersecție finită de semispații). Aceste proprietăți ale mulțimii A plus faptul că funcția obiectiv este și ea liniară ne asigură că:

- o soluție optimă – dacă există – este un punct de minim global adică:

$$f(x^*) \leq f(x) \quad (\forall) x \in A$$

- cel puțin o soluție este un vârf al mulțimii A

Cum numărul vârfurilor mulțimii poliedrale A este *finit* urmează că, pentru programul liniar (P), problema determinării unei soluții optime x^* din mulțimea, în general infinită, a tuturor soluțiilor admisibile se reduce la găsirea lui x^* în mulțimea finită a vârfurilor acestei mulțimi.

Metoda *simplex* realizează în mod *sistematic* această căutare oferind într-un număr finit de pași (iterații) soluția optimă x^* .

Neliniaritatea obiectivului sau a unora dintre restricții face ca unele din proprietățile relevate mai sus să dispară fapt care duce la *complicarea* sarcinii de determinare a optimului.

1) De la bun început vom sublinia faptul că în *programarea neliniară* – cu câteva excepții – metodele de rezolvare obțin “teoretic” soluția optimă ca *limită* a unui șir de soluții. Astfel, un proces *concret de optimizare neliniară* este *finit* nu datorită structurii problemei ci prin *voința utilizatorului* care limitează numărul pașilor în funcție de o serie întreagă de factori cum ar fi : complexitatea calcului, timpul disponibil, performanțele echipamentului de calcul etc.

2) este posibil ca funcția obiectiv din (P) să aibe mai multe *minime locale* pe mulțimea soluțiilor admisibile A . Pentru exemplificare considerăm problema:

$$(P) \begin{cases} \min f(x_1, x_2) = 2x_1 + 3x_2 \\ x_1^2 + x_2^2 \geq 4 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

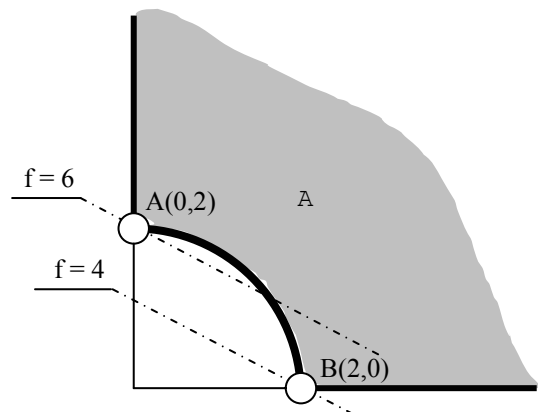


Figura 3.1

Mulțimea soluțiilor admisibile A este vizualizată în fig.3.1. Evident, A nu este o mulțime convexă. Punctul $A(0,2)$ oferă obiectivului valoarea 6 care este *cea mai mică valoare* a funcției f pe soluțiile admisibile situate în *imediate apropiere de A*! Totuși A nu este soluția optimă a problemei (P) deoarece în $B(2,0)$ f are valoarea $4 < 6$. Ca și A , B minimizează funcția obiectiv pe soluțiile admisibile “*vecine*” cu B dar, spre deosebire de A , B minimizează obiectivul pe întreaga mulțime A și în consecință reprezintă soluția optimă a problemei (P). Spunem că A și B sunt puncte de *minim local* ale funcției obiectiv f , B fiind chiar un punct de *minim global*.

Posibilitatea existenței mai multor minime locale ale funcției obiectiv reprezintă o serioasă dificultate în rezolvarea unui program neliniar. Într-adevăr, prin însăși formulare, într-o asemenea problemă se cere determinarea minimului global al obiectivului. Or, toate metodele de optimizare

neliniară cunoscute, nu reușesc să determine decât cel mult un minim local, neexistând garanția că acesta coincide cu minimul global căutat.

După cum vom vedea, dacă A este convexă iar funcția obiectiv este convexă și se minimizează atunci aceasta are cel mult un minim local care – dacă există – este automat global ! Din fericire, marea majoritate a aplicațiilor economice au proprietățile de mai sus, fapt care constituie un serios avantaj.

3) Chiar dacă restricțiile din (P) sunt *liniare* dar obiectivul rămâne *neliniar* însă *convex*, soluția optimă, deși se află pe frontiera lui A nu este neapărat *vârf*. Considerăm următorul exemplu:

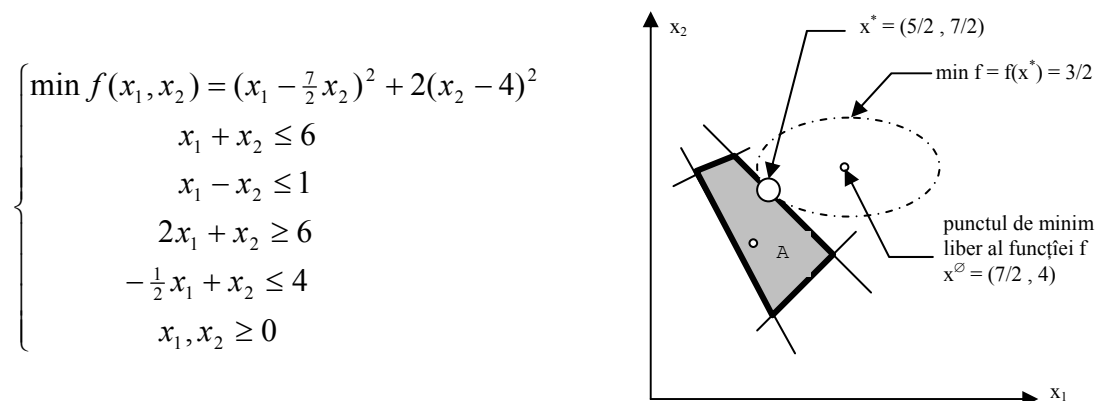


Figura 3.2

Curbele de nivel ale funcției obiectiv *patratice* f sunt *elipse centrate* în $x^{\emptyset} = (7/2, 4)$ care reprezintă și punctul de minim *nerestricționat* al lui f . Se poate arăta, prin mijloace algebrice elementare că f are pe A un minim *global* $x^* = (5/2, 7/2)$ care nu este vârf al lui A .

4) este posibil ca soluția optimă să fie situată în interiorul mulțimii A . Pentru exemplificare să atașăm restricțiilor din problema precedentă funcția

$$f(x_1, x_2) = (x_1 - 2)^2 + 2(x_2 - 3)^2$$

al cărei minim liber este punctul $x^{\emptyset} = (2, 3)$. Deoarece $x^{\emptyset} \in A$ (și mai precis $x^{\emptyset} \in \text{Int}(A)$) acest punct reprezintă soluția optimă x^* .

§ 4. Clase de probleme neliniare

1) Problemele de optimizare *fără restricții* au forma generală:

Să se determine $x^* \in \mathbb{R}^n$ care minimizează valoarea funcției

$$z = f(x_1, x_2, \dots, x_n)$$

minimul fiind luat după toți $x \in \mathbb{R}^n$ în care funcția f este definită.

2) **Probleme de optimizare cu *restricții liniare* și funcție obiectiv *neliniară*.** În această clasă un loc deosebit îl ocupă problemele de *programare patratice* în care funcția obiectiv este un *polinom* de gradul *doi* în variabilele sale:

$$f(x_1, x_2, \dots, x_n) = c_0 + \sum_{i=1}^n c_i x_i + \sum_{1 \leq i < j \leq n} c_{ij} x_i x_j$$

Importanța problemelor de programare patratice este motivată prin:

- faptul că modelează cu suficientă acuratețe multe situații practice;
- se rezolvă prin metode derivate din metoda simplex într-un număr finit de pași;
- rezolvarea multor probleme cu restricții liniare și funcție obiectiv neliniară se poate reduce la rezolvarea unei secvențe de probleme de programare patratică ale căror funcții obiectiv aproximează din ce în ce mai bine obiectivul neliniar original.

3) Problemele de programare *convexă* se caracterizează prin:

- funcție obiectiv *convexă* dacă aceasta se *minimizează* (echivalent: funcție obiectiv *concavă* dacă aceasta se *maximizează*);
- restricțiile *inegalități* sunt de forma $g_i(x) \leq 0$ în care g_i este o funcție *convexă* (echivalent $g_i(x) \geq 0$ cu g_i funcție *concavă*);
- eventualele restricții *egalități* sunt *liniare*, cerință motivată prin aceea că funcțiile liniare sunt singurele funcții simultan convexe și concave.

Problemele convexe au următoarele proprietăți fundamentale:

- mulțimea soluțiilor admisibile este *convexă*;
- funcția obiectiv admite cel mult un optim (minim sau maxim) *local*; automat acesta va fi un optim *global* și va reprezenta optimul problemei;
- dacă optimul *liber* (*nerestricționat*) al funcției obiectiv nu este o soluție admisibilă atunci optimul *restricționat* se găsește cu necesitate pe *frontiera* mulțimii A .

Importanța acestei clase de probleme este *covârșitoare*. Într-adevăr:

- programarea convexă include programarea liniară ;
- în acest domeniu a fost depus cel mai mare efort de cercetare și s-au obținut cele mai puternice rezultate *teoretice* (cum ar fi teoria dualității neliniare, condițiile de optimalitate Kuhn – Tucker) și *practice* (metode și algoritmi de optimizare);
- întregul formalism matematic al teoriei economice moderne se bazează pe ipotezele de convexitate.

4) Problemele de programare *separabilă* se caracterizează prin faptul că funcția obiectiv f ca și funcțiile g_I din restricții sunt *separabile* în sensul următoarei definiții:

Funcția $f(x_1, x_2, \dots, x_n)$ se numește separabilă dacă ea se poate scrie ca o sumă de funcții, fiecare de câte o singură variabilă:

$$f(x_1, x_2, \dots, x_n) = f_1(x_1) + f_2(x_2) + \dots + f_n(x_n)$$

Separabilitatea este importantă prin aceea că ușurează optimizarea. De exemplu, optimizarea unei funcții separabile fără restricții se reduce la optimizarea independentă a termenilor!

5) Problemele de programare *neconvexă* reunesc toate problemele care nu satisfac ipotezele de convexitate. Ele sunt “grele” în primul rând din cauza faptului că au mai multe minime locale. După cum s-a mai spus, metodele actuale pot determina un asemenea optim dar nu garantează că optimul găsit este cel global. Din fericire, există câteva tipuri de probleme neconvexe, utile în practică, care pot fi rezolvate fără dificultăți deosebite prin metode speciale. Printre aceste tipuri, se numără problemele de programare *fracționară*. Iată un exemplu:

$$\begin{cases} \min f(x) = \frac{cx + c_0}{dx + d_0} & x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \\ Ax \leq b \\ x \geq 0 \end{cases}$$

unde se presupune că $dx + d_0 > 0$ pe $A = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$.

O asemenea problemă se reduce la un program liniar uzual punând:

$$y = (y_1, y_2, \dots, y_n) = \frac{1}{dx + d_0} \cdot x, t = \frac{1}{dx + d_0} \text{ astfel că } x = \frac{y}{t}.$$

Rezultă programul liniar echivalent în $n + 1$ variabile $y = (y_1, y_2, \dots, y_n)$ și t :

$$\begin{cases} \min cy + c_0 t \\ Ay - bt \leq 0 \\ dy + d_0 t = 1 \\ y \geq 0, t \geq 0 \end{cases}$$

§ 5. Mulțimi și funcții convexe

Reamintim că o mulțime $C \subseteq \mathbb{R}^n$ se numește *convexă* dacă o dată cu două puncte conține și segmentul care le unește. Formal:

$$C \text{ este convexă} \Leftrightarrow_{\text{def.}} (\forall)x, y \in C, (\forall)\lambda \in [0,1] \Rightarrow (1-\lambda)x + \lambda y \in C$$

Fie C o mulțime convexă și f o funcție numerică definită în toate punctele mulțimii C . Spunem că f este *convexă* dacă:

$$(\forall)x, y \in C, (\forall)\lambda \in [0,1] \text{ avem:}$$

$$f((1-\lambda)x + \lambda y) \leq (1-\lambda)f(x) + \lambda f(y)$$

Vom zice că f este *concavă* dacă funcția *opusă* $-f$ este convexă. Aceasta revine la:

$$f((1-\lambda)x + \lambda y) \geq (1-\lambda)f(x) + \lambda f(y)$$

Funcția f se numește *strict convexă* (*strict concavă*) dacă inegalitățile de mai sus sunt *stricte* $(\forall)x, y \in C$ cu $x \neq y$ și $(\forall)\lambda \in (0,1)$.

Exemple de funcții convexe

1) Pentru funcțiile de o singură variabilă, convexitatea se traduce prin faptul că graficul "ține apa". Dacă I este un interval deschis (deci o mulțime convexă din \mathbb{R}) și f este o funcție definită pe I și care este de două ori derivabilă pe I atunci f este convexă $\Leftrightarrow f''(x) \geq 0 \quad (\forall)x \in I$.

2) Orice funcție liniară de n variabile

$$f(x_1, x_2, \dots, x_n) = a_1 x_1 + a_2 x_2 + \dots + a_n x_n + b$$

este simultan convexă și concavă.

3) Dacă f și g sunt funcții convexe cu același domeniu convex de definiție și $\alpha \geq 0$, $\beta \geq 0$ atunci combinația $\alpha f + \beta g$ este de asemenea o funcție convexă.

4) Să considerăm funcția patratrică în n variabile:

$$f(x_1, x_2, \dots, x_n) = \sum_{j=1}^n p_j x_j + \sum_{1 \leq i < j \leq n} c'_{ij} x_i x_j$$

definită în întreg \mathbb{R}^n . Dacă punem:

$$p = [p_1, p_2, \dots, p_n], x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, C = [c_{ij}], 1 \leq i, j \leq n, \text{ cu } \begin{cases} c_{ii} = 2c'_{ii}, i = 1, \dots, n \\ c_{ij} = c_{ji} = c'_{ij}, 1 \leq i < j \leq n \end{cases}$$

f se scrie condensat:

$$f(x) = px + \frac{1}{2} x^T C x$$

Notăm că prin construcție, matricea C este *simetrică*.

Cercetăm în ce condiții f este o funcție convexă. Deoarece "partea liniară" este convexă, chestiunea se reduce la a vedea în ce condiții funcția "pur patratrică" $\varphi(x) = x^T C x$ este convexă. Un calcul direct arată că φ satisface identitatea:

$$\varphi((1-\lambda)x + \lambda y) = (1-\lambda)\varphi(x) + \lambda\varphi(y) - \lambda(1-\lambda)\varphi(x-y) \quad (\forall)x, y \in \mathbb{R}^n, (\forall)\lambda \in \mathbb{R}$$

Acum, dacă $\lambda \in [0, 1]$, identitatea precedentă echivalează convexitatea funcției φ cu condiția:

$$z^T C z \geq 0 \quad (\forall)z \in \mathbb{R}^n$$

cerință care, după cum este cunoscut din algebra liniară, definește matricea C ca *matrice pozitiv semidefinită*.

Cu același raționament conchidem că φ este strict convexă dacă și numai dacă matricea C este *pozitiv definită* adică:

$$z^T C z \geq 0 \quad (\forall)z \in \mathbb{R}^n \text{ și } z^T C z = 0 \Rightarrow z = 0$$

Recapitulând obținem:

Funcția patratrică $f(x) = px + \frac{1}{2} x^T C x$ este convexă (strict convexă) dacă și numai dacă matricea simetrică C este pozitiv semidefinită (pozitiv definită).

Reamintim că matricea simetrică $C = [c_{ij}], 1 \leq i, j \leq n$ este pozitiv semidefinită (pozitiv definită) dacă și numai dacă toți minorii care se sprijină pe diagonala principală, adică:

$$c_{11} \quad ; \quad \begin{vmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{vmatrix} ; \quad \dots ; \quad \begin{vmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & c_{22} & \dots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \dots & c_{nn} \end{vmatrix}$$

sunt ≥ 0 (respectiv > 0).

6) Mai general, fie f o funcție definită și având derivate parțiale de ordinul doi în orice punct al unei mulțimi convexe deschise C din \mathbb{R}^n . Atunci f este *convexă* dacă și numai dacă matricea *hessiană*

$$H(x) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j} (x) \right]$$

este *pozitiv semidefinită*, în fiecare punct $x \in C$. Dacă $H(x)$ este *pozitiv definită* în orice punct din C , funcția f este *strict convexă*. Într-adevăr să fixăm un $x \in C$ și un $z \in \mathbb{R}^n$ oarecare. Considerăm funcția de o singură variabilă:

$$g(t) = f(x + tz)$$

definită pentru toți $t \in \mathbb{R}$ cu proprietatea că $x + tz \in C$ (acești t există cu siguranță și deoarece C este o mulțime deschisă, ei formează un interval deschis în \mathbb{R} ce conține 0). Evident, funcția f este convexă dacă și numai dacă funcția g este convexă, indiferent de alegerea lui $x \in C$ și a lui $z \in \mathbb{R}^n$. Ca și f , g este de două ori derivabilă și

$$g''(t) = z^T \cdot H(x + tz) \cdot z$$

Atunci, f este convexă $\Leftrightarrow (\forall) x \in C, (\forall) z \in \mathbb{R}^n$ și $(\forall) t \in \mathbb{R}$ cu $x + tz \in C$, $g''(t) = z^T \cdot H(x + tz) \cdot z \geq 0$, $\Leftrightarrow H(x)$ este pozitiv semidefinită $(\forall) x \in C$ etc.

În continuare, vom justifica unele proprietăți ale programelor convexe anunțate deja în secțiunea precedentă.

Propoziție Fie g_1, g_2, \dots, g_m funcții convexe definite în toate punctele unei mulțimi convexe $C \subseteq \mathbb{R}^n$. Atunci mulțimea:

$$A = \{x \in C \mid g_1(x) \leq 0, g_2(x) \leq 0, \dots, g_m(x) \leq 0\}$$

este convexă.

Demonstrație Deoarece :

$$A = \{x \in C \mid g_1(x) \leq 0\} \cap \dots \cap \{x \in C \mid g_m(x) \leq 0\}$$

și o intersecție de mulțimi convexe este încă o mulțime convexă, va fi suficient să arătăm că dacă g este o funcție convexă pe C atunci mulțimea :

$$A = \{x \in C \mid g(x) \leq 0\}$$

este convexă. Fie $x, y \in A$ și $\lambda \in [0, 1]$; atunci $g(x) \leq 0$, $g(y) \leq 0$ astfel că:

$$g((1-\lambda)x + \lambda y) \leq (1-\lambda)g(x) + \lambda g(y) \leq 0$$

Prin urmare $(1-\lambda)x + \lambda y \in A$.

Consecință Mulțimea soluțiilor admisibile ale unui program convex este o mulțime convexă.

Teoremă Considerăm problema de optimizare convexă: $\min\{f(x) \mid x \in A\}$ în care A este mulțimea convexă a soluțiilor admisibile iar f este o funcție convexă pe A . Dacă f are în $x^* \in A$ un *minim local* atunci x^* este un punct de *minim global* al funcției f pe întreg A .

Demonstrație Prin ipoteză, $f(x^*) \leq f(x)$ oricare ar fi $x \in A$, *suficient de aproape de x^** . Presupunem prin absurd că x^* nu este un minim global al funcției f : există atunci $x^{**} \in A$ astfel încât $f(x^{**}) < f(x^*)$. Să considerăm acum un punct interior variabil pe segmentul $[x^*, x^{**}]$:

$$z = (1-\lambda)x^* + \lambda x^{**} \quad \lambda \in (0,1)$$

Deoarece A este convexă, toate aceste puncte sunt în A . Funcția f fiind convexă avem:

$$\begin{aligned} f(z) = f((1-\lambda)x^* + \lambda x^{**}) &\leq (1-\lambda)f(x^*) + \lambda f(x^{**}) < \\ &< (1-\lambda)f(x^*) + \lambda f(x^*) = f(x^*) \end{aligned}$$

Pentru λ suficient de mic, z se va afla în imediata apropiere a lui x^* și deci, conform ipotezei, $f(z) \geq f(x^*)$, în contradicție cu cele arătate mai sus. Contradicție! Deci x^* este un minim global al funcției f .

§ 6. Optimizare fără restricții

6.1 Introducere Fie f o funcție numerică de n variabile pe care, pentru simplitate, o presupunem definită și cu derivate parțiale cel puțin de ordinul doi în fiecare punct din \mathbb{R}^n . Considerăm problema de optimizare *fără restricții*:

$$(P) \text{ Să se determine } x^* \in \mathbb{R}^n \text{ cu proprietatea: } f(x^*) = \{\inf f(x) \mid x \in \mathbb{R}^n\}$$

În legătură cu această problemă, analiza matematică clasică furnizează următorul rezultat fundamental:

Teoremă Dacă x^* este un punct de *extrem local* al funcției f , atunci $\nabla f(x^*) = 0$, unde

$$\nabla f(x) = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right] \quad (6.1)$$

este *gradientul funcției f* . Reciproc, dacă $x^* \in \mathbb{R}^n$ este un punct în care condiția (1) are loc și în plus matricea *hessiană* $H(x^*)$ este *pozitiv definită*, unde

$$H(x) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j} \right] \quad 1 \leq i, j \leq n$$

atunci x^* este un punct de *minim local* al funcției f .

Prin urmare, soluția optimă a problemei (P) trebuie căutată printre soluțiile sistemului de n ecuații în n variabile, în general neliniar:

$$\nabla f(x) = 0 \Leftrightarrow \frac{\partial f}{\partial x_1} = 0, \frac{\partial f}{\partial x_2} = 0, \dots, \frac{\partial f}{\partial x_n} = 0 \quad (6.2)$$

Însă *rezolvarea* sistemului (6.2), cel puțin în sensul uzual al termenului, este practic imposibil de făcut în cvasitotalitatea cazurilor practice. Chiar și în cazul fericit în care, prin mijloace *analitice* – adică “cu formule” – am putea găsi o soluție a sistemului (6.2) nu avem nici o garanție că aceasta este soluția optimă a problemei (P): ea ar putea fi foarte bine un punct de *maxim local* sau un *punct șa* sau, în cel mai bun caz, un *minim local* diferit de cel global! Astfel apare necesitatea considerării altor metode de abordare a problemei (P).

6.2 Principiul metodelor de optimizare fără restricții *Ideea comună* a tuturor metodelor de minimizare *nerestricționată (liberă)* este următoarea:

- Se generează un șir de puncte $x^0, x^1, x^2 \dots$ din \mathbb{R}^n , numite în continuare *aproximații*. Punctul *inițial* x^0 este *dat*, alegerea sa fiind făcută de *utilizator* în funcție de specificul problemei (P). O dată obținută aproximația x^k , noua aproximație x^{k+1} se determină astfel:

- Se alege o *direcție de deplasare* s^k precum și un *pas al deplasării* α_k ; s^k este un vector *nenul* din \mathbb{R}^n – de regulă $\|s^k\| = 1$; α_k este un scalar *pozitiv*.

- Se pune:

$$x^{k+1} = x^k + \alpha_k \cdot s^k \quad (6.3)$$

În principiu, vectorul s^k este astfel ales încât prin deplasarea din x^k în direcția s^k , să se obțină – cel puțin în prima fază a deplasării – o *descreștere* a valorii funcției de minimizat f . O dată stabilită direcția de deplasare s^k , pasul α_k se alege astfel încât:

$$f(x^{k+1}) < f(x^k)$$

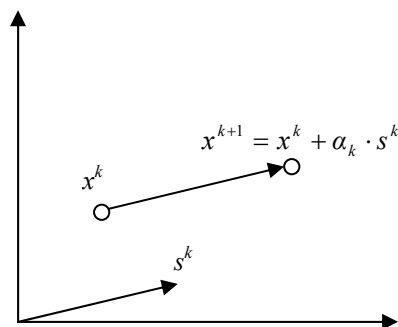


Figura 6.1

Prin urmare:

$$f(x^0) > f(x^1) > f(x^2) > \dots$$

Mai precis α_k se găsește prin minimizarea funcției de *o singură variabilă*:

$$g(\alpha) = f(x^k + \alpha s^k) \quad \alpha \geq 0 \quad (6.4)$$

Pentru a obține o metodă *efectivă* de minimizare este necesar să se precizeze:

- modul de alegere a direcției de deplasare s^k , $k = 0, 1, 2, \dots$;
- modul de alegere a pasului α_k altfel spus, modul în care se execută minimizarea funcției unidimensionale $g(\alpha)$ din (6.4);
- procedeul de oprire a procesului iterativ.

Decisivă în diferențierea metodelor concrete atât în ceea ce fundamental teoretic cât și performanțele numerice este prima chestiune, legată de alegerea direcției de deplasare.

Este foarte important de reținut că dacă funcția de minimizat nu are proprietăți adiționale “bune”, cum ar fi aceea de a fi convexă, rezultatul aplicării acestor metode se materializează în cel mai bun caz, într-un minim local – de regulă cel mai apropiat de punctul inițial x^0 . În practică, pentru a obține toate minimele locale și în particular minimul global, se procedează la *repetarea* minimizării din mai multe puncte inițiale, judicios alese.

Marea majoritate a metodelor de minimizare nerestricționată presupun diferențiabilitatea funcției obiectiv ; există totuși și scheme de minimizare pentru funcții nediferențiabile!

6.3 Gradientul unei funcții. Proprietăți În ipotezele și notațiile precedente, o hipersuprafață de nivel a funcției f este mulțimea punctelor $x = (x_1, x_2, \dots, x_n) \in R^n$ care satisfac o ecuație de forma: $f(x) = C$, unde C este o constantă. În cazul a două (respectiv trei) variabile vom vorbi despre curbe (respectiv, suprafețe) de nivel. Astfel pentru funcțiile:

$$\begin{aligned} a) f(x_1, x_2) &= 2x_1 + 3x_2 ; & b) f(x_1, x_2) &= -4x_1 + 2x_2 + x_1^2 + x_2^2 \\ c) f(x_1, x_2) &= 2x_1 - 16x_2 + x_1^2 + 4x_2^2 \end{aligned}$$

curbele de nivel sunt: a) drepte paralele; b) cercuri concentrice; c) elipse concentrice (vezi fig. 6.2)

Gradientul funcției f este vectorul:

$$\nabla f(x) = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right]$$

Acesta are următoarele proprietăți:

- În orice punct de extrem local x^* al funcției f avem $\nabla f(x^*) = 0$; reciproca nu este în general adevărată.
- Fie $\bar{x} \in R^n$ un punct în care $\nabla f(\bar{x}) \neq 0$. Atunci $s = -\nabla f(\bar{x})$ este direcția de deplasare din \bar{x} corespunzătoare celei mai rapide descreșteri a funcției f . Analog, $\nabla f(\bar{x})$ este direcția celei mai rapide creșteri.

Într-adevăr, să considerăm o direcție oarecare $s = (s_1, s_2, \dots, s_n) \in R^n$, $s \neq 0$ și funcția :

$$g(\alpha) = f(\bar{x} + \alpha s) \quad \alpha \geq 0$$

care indică variația funcției f pe direcția de deplasare s din \bar{x} .

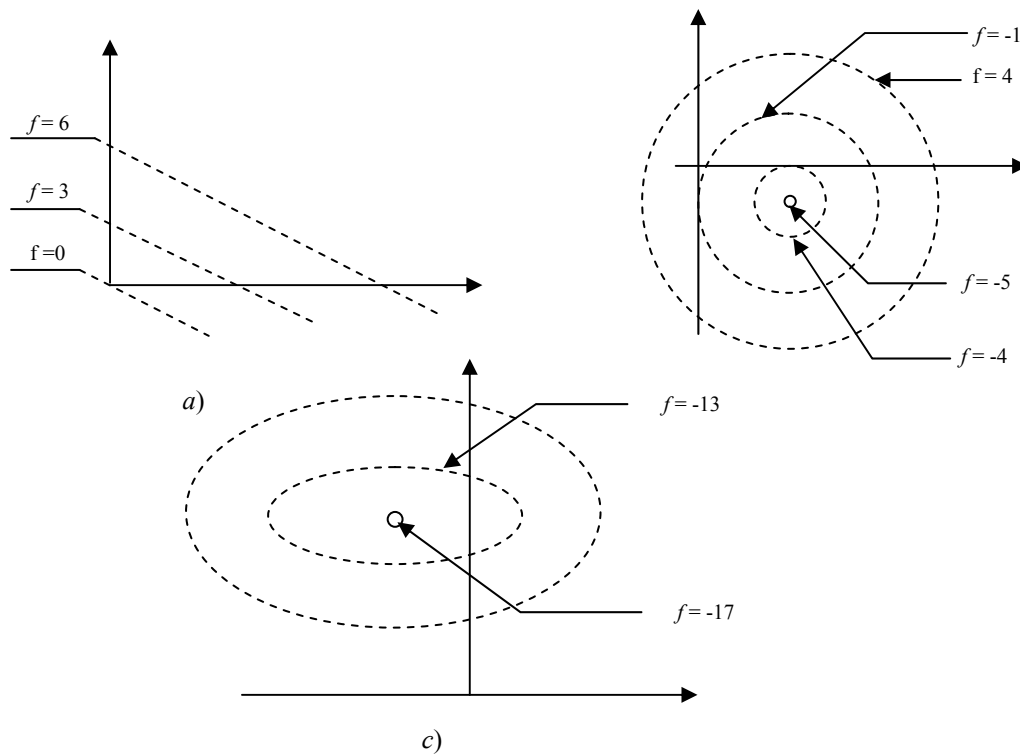


Figura 6.2

După cum se știe, *variația* (adică *creșterea* sau *descreșterea*) *potențială* sau *incipientă* a funcției f pe direcția s plecând din \bar{x} , este dată de derivata $g'(0)$.

În general:

$$g'(a) = \sum_{i=1}^n s_i \cdot \frac{\partial f}{\partial x_i}(\bar{x} + as) = s \cdot \nabla f(\bar{x} + as)$$

astfel că:

$$g'(0) = s \cdot \nabla f(\bar{x})$$

Din inegalitatea Cauchy – Buniakovski – Schwarz:

$$|s \cdot \nabla f(\bar{x})| \leq \|s\| \cdot \|\nabla f(\bar{x})\| \Leftrightarrow -\|s\| \cdot \|\nabla f(\bar{x})\| \leq g'(0) \leq \|s\| \cdot \|\nabla f(\bar{x})\|$$

rezultă că $g'(0)$ are cea mai mică valoare pentru $s = -\nabla f(\bar{x})$ și cea mai mare pentru $s = \nabla f(\bar{x})$.

• Dacă $\nabla f(\bar{x}) \neq 0$, atunci acest vector este *perpendicular* pe hipersuprafața de nivel $f(x) = f(\bar{x})$ ce trece prin \bar{x} (vezi fig. 4.8)

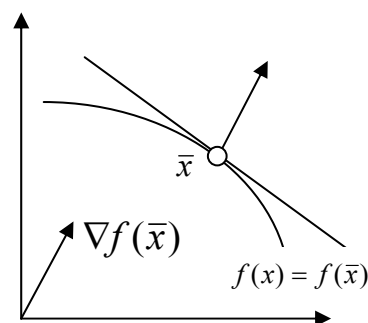


Figura 6.3

Exemplul 6.1 Considerăm funcția *patratrică*: $f(x_1, x_2) = 16x_1^2 + (x_2 - 4)^2$

Curbele sale de nivel sunt *elipse* cu centrul în punctul $x^* = (0, 4)$ care reprezintă și punctul de *minim global* al funcției. Gradientul lui f :

$$\nabla f(x_1, x_2) = [32x_1, 2x_2 - 8]$$

în punctul $\bar{x} = (1,1)$ este nenul: $\nabla f(\bar{x}) = (32, -6)$. Cea mai rapidă descreștere a funcției f plecând din \bar{x} are loc pe direcția $s = -\nabla f(\bar{x}) = (-32, 6)$ – vezi fig. 6.4

Atenție: *pe direcția celei mai rapide descreșteri, funcția nu descrește neîncetat!* Pentru ilustrare, în exemplul de mai sus, să considerăm un punct variabil pe direcția celei mai rapide descreșteri din $\bar{x} = (1,1)$:

$$x(\alpha) = \bar{x} - \alpha \nabla f(\bar{x}) = (1,1) - \alpha(32, -6) = (1 - 32\alpha, 1 + 6\alpha) \quad \alpha \geq 0$$

Pe această direcție, comportarea funcției f este descrisă de funcția:

$$g(\alpha) = f(x(\alpha)) = 16(1 - 32\alpha)^2 + (6\alpha - 3)^2 = 16420\alpha^2 - 1060\alpha + 25$$

a cărei variație este arătată în fig.6.5

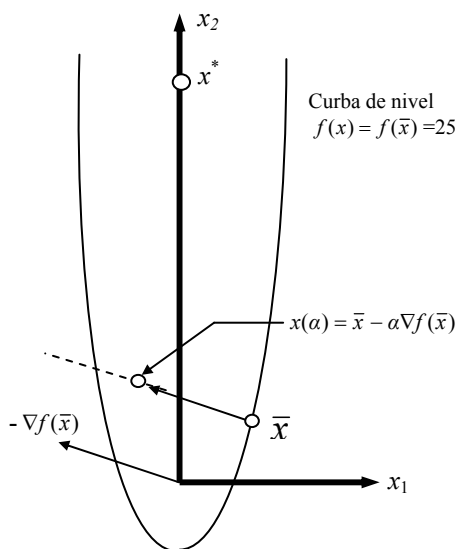


Figura 6.4

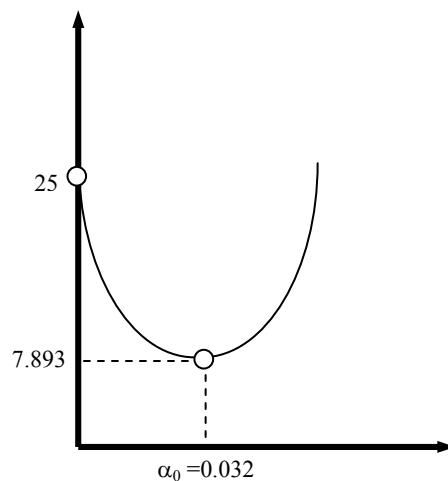


Figura 6.5

Se observă că atunci când α crește de la 0 la 0.032 (= punctul în care $g(\alpha)$ are valoarea minimă), funcția g și deci și funcția f scad de la valoarea 25 la valoarea 7.893 după care încep din nou să crească!

6.4 Criterii de oprire ale procesului iterativ de minimizare 1) Deoarece într-un punct de minim local avem $\nabla f(x^*) = 0$, procesul de minimizare se poate opri în momentul în care:

$$a) \left| \frac{\partial f}{\partial x_i}(x^k) \right| < \varepsilon \quad i = 1, 2, \dots, n$$

sau:

$$b) \sum_{i=1}^n \left[\frac{\partial f}{\partial x_i}(x^k) \right]^2 < \varepsilon$$

unde ε este un număr pozitiv foarte mic, ca de exemplu $\varepsilon = 10^{-5}$.

3) De multe ori este preferabil să oprim procesul iterativ în momentul în care variația funcției obiectiv în două aproximații succesive nu depășește o toleranță dată:

$$f(x^k) - f(x^{k+1}) < \varepsilon$$

3) Se poate întâmpla ca, din cauza unei nefericite alegeri a aproximației inițiale x^0 sau a structurii funcției de minimizat, procesul iterativ, deși convergent, să fie foarte lent. În asemenea situații, timpul de calcul poate atinge valori nepermis de mari. Vom evita acest inconvenient limitând din start numărul iterațiilor posibil de efectuat.

6.5 Minimizarea unei funcții de o singură variabilă În secțiunea introductivă am văzut că toate metodele de minimizare nerestricționată a unei funcții f de mai multe variabile necesită minimizarea unei funcții $g(\alpha)$ de o singură variabilă. Deoarece minimizarea unidimensională se efectuează la fiecare iterație a procesului de minimizare multidimensională este clar că eficacitatea procesului multidimensional depinde de calitățile procesului unidimensional.

În continuare vom presupune că în prealabil am localizat un punct de minim x^* al funcției g într-un interval (a, b) suficient de mic.

În general, metodele de minimizare unidimensională se bazează pe una din următoarele idei:

- Se aproximează funcția g pe intervalul (a, b) printr-un polinom de interpolare de grad doi sau trei al cărui punct de minim din (a, b) se determină ușor pe baza unei formule. Punctul de minim astfel calculat se consideră a fi o bună aproximare a punctului de minim x^* căutat. Construcția polinomului de interpolare se poate face folosind:

- numai evaluări ale funcției g (în câteva puncte);
- evaluări atât ale funcției g cât și ale derivatei sale.

De exemplu:

- putem construi un polinom de interpolare de gradul trei utilizând valorile funcției g și ale derivatei sale g' în punctele a și b ce "mărginesc" punctul de minim căutat.
- putem construi un polinom de interpolare de gradul doi utilizând valorile funcției g în a și b precum și într-un al treilea punct $c \in (a, b)$ – de obicei se ia $c = \frac{1}{2}(a + b)$.

- Se micșorează progresiv intervalul (a, b) la un interval (\bar{a}, \bar{b}) care de asemenea conține pe x^* și a cărui lungime este inferioară unei toleranțe date:

$$\bar{b} - \bar{a} < \varepsilon \quad \text{cu } \varepsilon > 0 \text{ număr foarte mic.}$$

Atunci orice punct \bar{x} din (\bar{a}, \bar{b}) aproximează x^* cu toleranța ε în sensul că:

$$|\bar{x} - x^*| < \varepsilon$$

Micșorarea intervalului (a, b) se face cu ajutorul evaluării funcției g și/sau ale derivatei sale într-un număr de puncte din interiorul lui (a, b) . Cele mai cunoscute metode de acest fel sunt metoda biseecției succesive, metoda secțiunii de aur și metoda Fibonacci.

În figura 6.6 este prezentată schema logică a metodei biseecției succesive ; justificarea metodei rezultă imediat din situațiile ilustrate în figurile 6.7a) și 6.7b).

Pentru a înțelege modul de acțiune al metodelor bazate exclusiv pe evaluarea funcției de minimizat în diferite puncte sunt necesare câteva pregătiri.

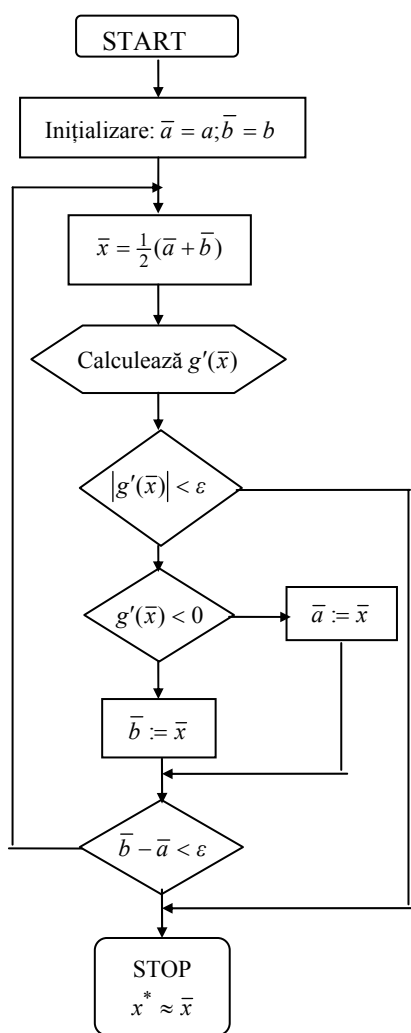


Figura 6.6

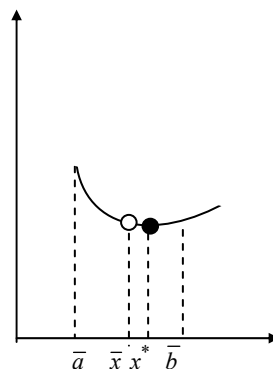


Figura 6.7 a)

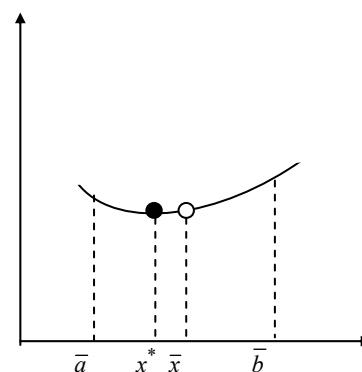


Figura 6.7 b)

Deoarece în intervalul de localizare (a, b) am presupus că f are un *unic* punct de *minim* x^* (vezi fig. 6.8) urmează că oricare ar fi $x_1 < x_2$ din (a, b) :

$$f(x_1) > f(x_2) \quad \text{dacă} \quad x_1 < x_2 < x^*$$

și

$$f(x_1) < f(x_2) \quad \text{dacă} \quad x^* < x_1 < x_2$$

Alegem două puncte x_S și x_D în (a, b) astfel încât $x_S < x_D$ pe care le vom numi *puncte interioare*. Dacă $f(x_S) \leq f(x_D)$, atunci cu necesitate x^* se va afla în intervalul (a, x_D) .

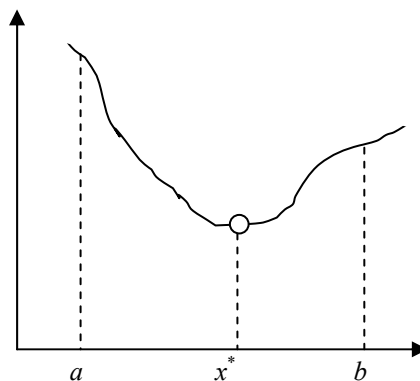


Figura 6.8

Dacă din contră, $f(x_S) > f(x_D)$, atunci cu siguranță x^* va fi în (x_S, b) . Cele două situații sunt ilustrate în figurile 6.9a) și 6.9b).

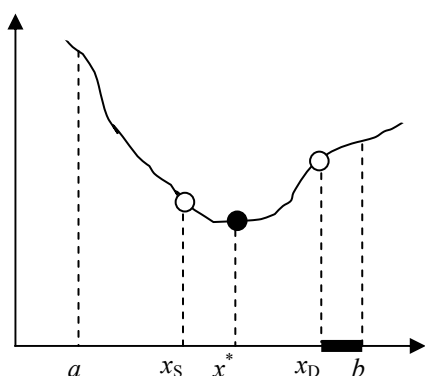


Figura 6.9a)

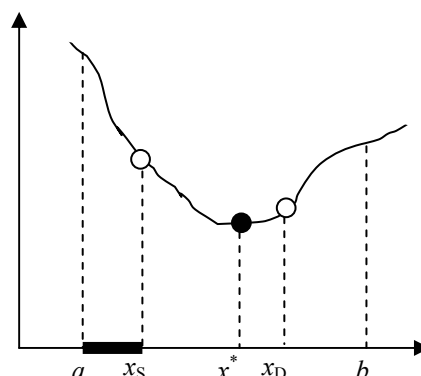


Figura 6.9b)

Observăm că noul interval mai mic conține cu necesitate unul din punctele interioare alese. Dacă în noul interval mai mic alegem încă un punct interior putem repeta considerațiile precedente.

Metodele de minimizare unidimensională bazate pe evaluări multiple se diferențiază prin modul de alegere a celor două puncte interioare x_S și x_D .

Vom descrie în continuare *metoda secțiunii de aur*. Fie $r = \frac{\sqrt{5}-1}{2} = 0.618034$.

START. Inițializăm:

- extremitățile intervalului curent (\bar{a}, \bar{b}) care conține punctul de minim x^* căutat:

$$\bar{a} \leftarrow a \quad \bar{b} \leftarrow b$$
- punctele interioare:

$$x_S \leftarrow \bar{a} + r^2 \cdot h \quad x_D \leftarrow \bar{a} + r \cdot h \quad \text{unde: } h \leftarrow \bar{b} - \bar{a}$$

Evaluăm funcția f în x_S și x_D .

Conținutul unei iterații:

Pasul 1. Se compară $f(x_S)$ cu $f(x_D)$. Dacă:

- $f(x_S) \leq f(x_D)$ se merge la pasul 2.
- $f(x_S) > f(x_D)$ se merge la pasul 3.

Pasul 2. (x^* se află în intervalul (\bar{a}, x_D) și tot aici se găsește și x_S)

Actualizăm:

$$\bar{b} \leftarrow x_D \quad ; \quad h \leftarrow \bar{b} - \bar{a}$$

Dacă $h < \epsilon$ se merge la pasul 4. Altminteri, actualizăm:

$$x_D \leftarrow x_S \quad ; \quad x_S \leftarrow \bar{a} + r^2 \cdot h$$

Evaluăm f în (noul) x_S . Ne întoarcem la pasul 1 în cadrul unei noi iterații.

Pasul 3. (x^* se află în intervalul (x_S, \bar{b})) și tot aici se găsește și x_D)

Actualizăm:

$$\bar{a} \leftarrow x_S \quad ; \quad h \leftarrow \bar{b} - \bar{a}$$

Dacă $h < \varepsilon$ se merge la pasul 4. Altminteri, actualizăm:

$$x_S \leftarrow x_D \quad ; \quad x_D \leftarrow \bar{a} + r \cdot h$$

Evaluăm f în (noul) x_D . Ne întoarcem la pasul 1 în cadrul unei noi iterații.

Pasul 4. $x^* \approx \frac{\bar{a} + \bar{b}}{2}$.

6.6 Schema logică de principiu a metodelor de optimizare nerestricționată

Considerațiile precedente conduc la schema logică de principiu din fig. 6.10. Elementele esențiale ale blocurilor **P** (alegerea **P**asului deplasării) și **O** (**O**prirea procesului iterativ) au fost deja discutate în secțiunile 6.4 și 6.5. În ceea ce privește blocul **D** (alegerea **D**irecției de deplasare) acesta diferă de la metodă la metodă. Cea mai simplă metodă de minimizare nerestricționată, datorată lui **Cauchy** va fi prezentată în continuare.

6.7 Metoda gradientului (Cauchy)

La fiecare iterație a algoritmului din fig.6.10 direcția de deplasare este dată de relația:

$$s^k = -\nabla f(x^k)$$

cu condiția ca $\nabla f(x^k) \neq 0$.

Această opțiune se bazează pe faptul că $-\nabla f(x^k)$ este direcția celei mai rapide descreșteri din x^k .

Caracteristic acestei metode este faptul că două direcții de deplasare consecutive sunt perpendiculare! Într-adevăr, dată direcția s^k , lungimea pasului α_k al deplasării se află, așa cum s-a mai spus, minimizând funcția unidimensională:

$$g(\alpha) = f(x^k + \alpha s^k) \quad , \quad \alpha \geq 0$$

Prin urmare $g'(\alpha_k) = 0$. Am văzut că: $g'(\alpha) = s^k \cdot \nabla f(x^k + \alpha s^k)$ - a se revedea secțiunea 6.3! - așa încât:

$$g'(\alpha_k) = 0 \Leftrightarrow s^k \cdot \nabla f(x^k + \alpha_k s^k) = 0 \Rightarrow s^k \cdot \nabla f(x^{k+1}) = 0 \Leftrightarrow s^k \cdot s^{k+1} = 0$$

Pentru funcțiile “sferice” de forma:

$$f(x_1, x_2, \dots, x_n) = c_0 + \sum_{i=1}^n c_i x_i + \sum_{i=1}^n x_i^2 = c_0 + cx + \|x\|^2 \quad c = (c_1, c_2, \dots, c_n)$$

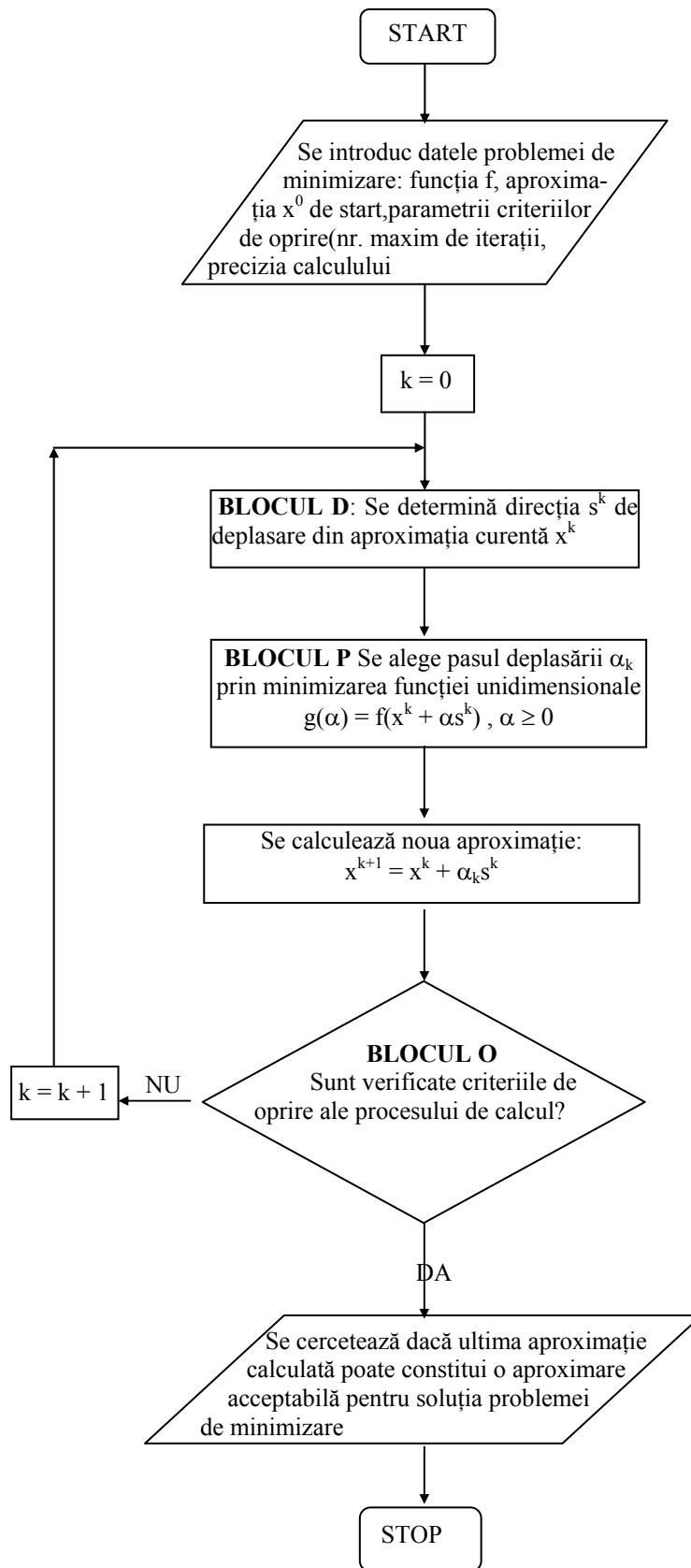


Figura 6.10

metoda gradientului oferă punctul de minim (global) într-o singură iterație, indiferent de aproximația inițială x^0 (vezi fig. 6.11a) Pentru orice altă funcție șirul de aproximații:

$$x^0, x^1, x^2, \dots \text{ pentru care } f(x^0) > f(x^1) > f(x^2) > \dots$$

conduce în general la un optim local.

Principalul neajuns al metodei gradientului constă în faptul că pașii succesivi sunt perpendiculari fapt care, în cazul anumitor funcții, conduce la o convergență foarte lentă. Mai precis, dacă hipersuprafețele de nivel au o oarecare “**excentricitate**” zig – zag procesul iterativ amendează convergența, deoarece în acest caz direcția gradientului este mult diferită de direcția către minim (vezi fig. 6.11b)

Există scheme de minimizare mult mai eficiente dintre care cea mai puternică pare a fi metoda **Davidon – Fletcher – Powell** [3]. În principiu, aceste metode garantează obținerea minimului unei funcții patratice de n variabile în cel mult n pași.

Exemplu numeric Vom efectua câteva iterații pentru căutarea minimului funcției:

$$f(x_1, x_2) = -x_2 + x_1^2 - 2x_1x_2 + 2x_2^2$$

cu metoda gradientului. Punctul de plecare $x^0 = (1,1)$

Gradientul funcției:

$$\nabla f = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2} \right] = [2x_1 - 2x_2, -1 - 2x_1 + 4x_2]$$

Iterația 1

- $\nabla f(x^0) = [0, 1]$
- $s^0 = -\nabla f(x^0) = [0, -1]$
- $x(\alpha) = x^0 + \alpha s^0 = [1, 1] + \alpha[0, -1] = [1, 1 - \alpha]$, $\alpha > 0$
- $g(\alpha) = f(x(\alpha)) = 2\alpha^2 - \alpha$ are un minim în $\alpha_0 = 1/4$
- $x^1 = x(1/4) = [1, 3/4]$

Iterația 2

- $\nabla f(x^1) = [1/2, 0]$
- $s^1 = -\nabla f(x^1) = [-1/2, 0]$
- $x(\alpha) = x^1 + \alpha s^1 = [1, 3/4] + \alpha[-1/2, 0] = [1 - \alpha/2, 3/4]$, $\alpha > 0$
- $g(\alpha) = f(x(\alpha)) = \frac{1}{4}\alpha^2 - \frac{1}{4}\alpha - \frac{1}{8}$ are un minim în $\alpha_1 = 1/2$
- $x^2 = x(1/2) = [3/4, 3/4]$

Iterația 3

- $\nabla f(x^2) = [0, 1/2]$
- $s^2 = -\nabla f(x^2) = [0, -1/2]$
- $x(\alpha) = x^2 + \alpha s^2 = [3/4, 3/4] + \alpha[0, -1/2] = [\frac{3}{4}, \frac{3}{4} - \frac{1}{2}\alpha]$, $\alpha > 0$
- $g(\alpha) = f(x(\alpha)) = \frac{1}{2}\alpha^2 - \frac{1}{4}\alpha - \frac{3}{16}$ are un minim în $\alpha_2 = 1/4$
- $x^3 = x(1/4) = [3/4, 5/8]$

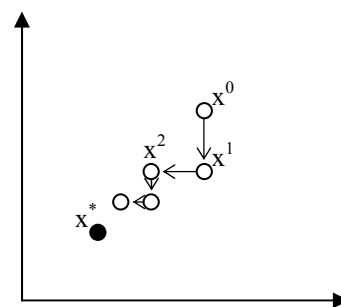


Figura 6.12

La iterația 4 se obține $[5/8, 5/8]$ ș.a.m.d. Funcția dată este convexă (exercițiu!) având un minim în $x^* = [1/2, 1/2]$. În figura 6.12 se poate constata apropierea “în zig – zag” a punctelor $x^0, x^1, x^3 \dots$ de x^* .

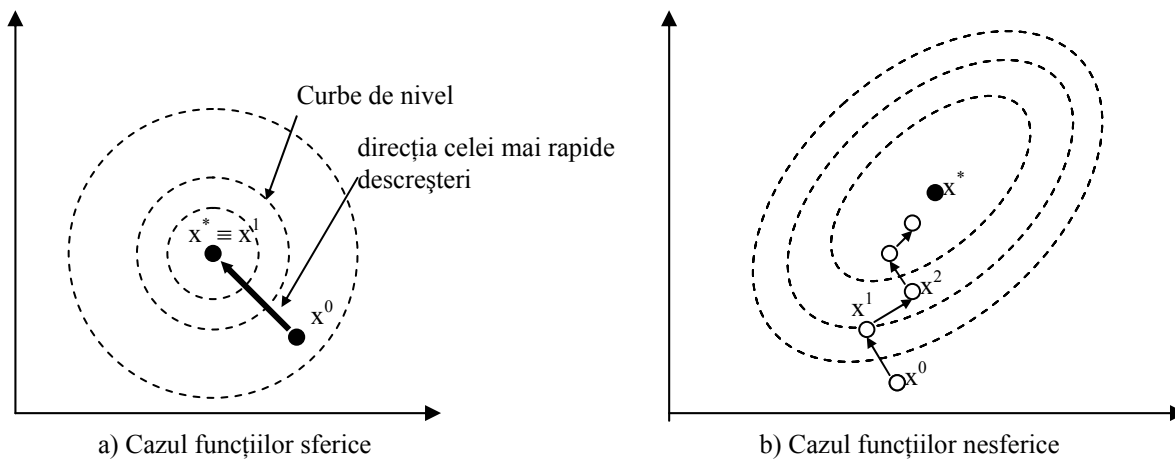


Figura 6.11

§ 7. Optimizarea neliniară fără restricții. Cazul convex

Reluăm problema de optimizare generală sub forma:

$$(P) \text{ Să se determine } x^* \in A \subseteq \mathbb{R}^n \text{ cu proprietatea: } f(x^*) = \inf \{f(x), x \in A\}$$

unde A ,numită și mulțimea soluțiilor admisibile ale problemei (P) este definită de o mulțime de restricții:

$$g_i(x) \leq 0 \quad i \in M = \{1, 2, \dots, m\}$$

Pentru simplificarea expunerii, eventualele condiții de nenegativitate $x_j \geq 0$ au fost incluse în blocul restricțiilor sub forma: $-x_j \leq 0$.

Presupunem că funcțiile f și g_1, g_2, \dots, g_m sunt definite în întreg spațiul \mathbb{R}^n , sunt **convexe** și **diferențiabile** și cel puțin una dintre ele este **neliniară**, altminteri (P) ar fi un program liniar. Astfel, (P) este o problemă de programare convexă.

Reamintim că în acest context:

- A este o mulțime **convexă** și **închisă**;
- orice **minim local** al funcției f pe mulțimea A este un **minim global**.

(vezi secțiunea 5)

Metodele de optimizare cu restricții se împart în trei categorii:

1) Metode bazate pe adaptarea schemei generale de optimizare nerestricționată la cazul prezenței restricțiilor; aceste metode poartă numele generic de **metode de gradient**.

2) Metode bazate pe utilizarea **funcțiilor de penalizare** : rezolvarea problemei (P) se reduce la o suită (teoretic infinită) de optimizări nerestricționate.

3) Metode bazate pe **plane de secțiune**; în principiu, aceste metode "aproximează" A printr-o mulțime **poliedrală** adică printr-o mulțime ce poate fi descrisă printr-un sistem de **inegalități liniare**; rezolvarea problemei (P) se reduce la o secvență (infinită) de **optimizări liniare** efectuate cu ajutorul **algoritmului simplex**.

7.1 Principiul metodelor de gradient

Aplicarea schemei generale de minimizare nerestricționată trebuie să țină seama de următoarele aspecte:

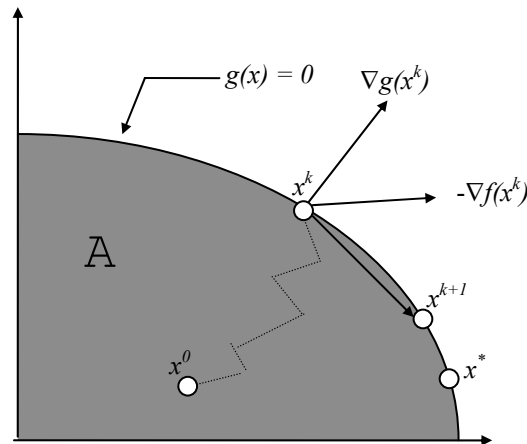


Figura 7.1

- de regulă soluția optimă x^* a problemei (P) se găsește pe **frontiera** lui A , adică satisface **cu egalitate** cel puțin una din restricțiile $g_i(x) \leq 0, i \in M$.
- chiar dacă se pleacă de la un punct inițial x^0 situat în **interiorul** lui A ($\Leftrightarrow g_i(x) < 0, i \in M$) se ajunge relativ repede la un punct x^k situat chiar pe frontieră altfel spus care satisface cu egalitate o parte din restricțiile problemei (P):

$$g_i(x^k) = 0, i \in I \subset M$$

Într-o atare situație, direcția celei mai rapide descreșteri $-\nabla f(x^k)$ s-ar putea să nu mai fie admisibilă adică:

$$x(\alpha) = x^k - \alpha \nabla f(x^k) \notin A \quad (\forall) \alpha > 0$$

(vezi fig. 7.1)

Introducem următoarea terminologie:

- o direcție s ($s \in \mathbb{R}^n, s \neq 0$) se va numi **admisibilă** relativ la punctul curent x^{k+1} dacă o deplasare suficient de mică din x^k pe direcția s ne menține în A ;
- direcția s se va numi **utilizabilă** dacă deplasarea din x^k pe direcția s duce la scăderea valorii funcției obiectiv.

Se poate arăta că o direcție s este admisibilă și utilizabilă relativ la punctul curent x^k dacă și numai dacă s satisface condițiile:

$$\begin{cases} s \cdot \nabla g_i(x^k) < 0 & i \in I \\ s \cdot \nabla f(x^k) < 0 \end{cases} \quad (7.1)$$

O dată stabilită direcția de deplasare s^k din x^k - direcție care verifică condițiile (7.1) - pasul α_k se alege astfel încât noua aproximație :

$$x^{k+1} = x^k + \alpha_k s^k$$

să aibe proprietățile:

$$x^{k+1} \in A \quad (\Leftrightarrow g_i(x^{k+1}) \leq 0, i \in M) \quad \text{și} \quad f(x^{k+1}) < f(x^k)$$

Cu acest amendament - legat de modul de alegere a direcției de deplasare - schema generală de minimizare nerestricționată funcționează și în cazul prezenței restricțiilor.

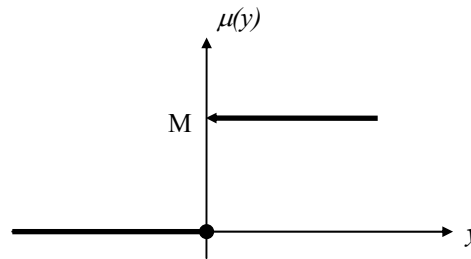


Figura 7.2

7.2 Principiul metodelor bazate pe funcții de penalizare

În esență, aceste metode încorporează restricțiile problemei (P0) în funcția obiectiv prin intermediul unei funcții care **penalizează nerespectarea** lor.

Pentru ilustrare să considerăm funcția:

$$\mu(y) = \begin{cases} 0 & \text{daca } y \leq 0 \\ M & \text{daca } y > 0 \end{cases} \quad \text{unde } M \gg 0$$

al cărei grafic este dat în fig 7.2. Considerăm problema de minimizare fără restricții:

$$(P') \begin{cases} \inf F(x) = f(x) + \sum_{i \in M} \mu(g_i(x)) \\ x \in R^n \end{cases}$$

Evident:

$$\varphi(y) = \begin{cases} 0 & \text{daca } y \leq 0 \\ y^2 & \text{daca } y > 0 \end{cases}$$

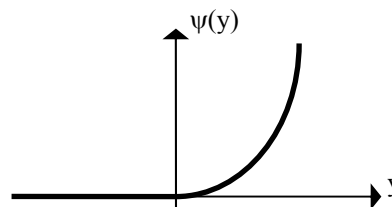


Figura 7.3

$$F(x) = \begin{cases} f(x) & \text{daca } x \in A \\ M & \text{daca } x \notin A \end{cases}$$

În consecință, orice procedeu de minimizare a funcției F se va orienta către punctele mulțimii A și deci va minimiza funcția originală f . Dificultatea majoră rezidă în faptul că funcția de penalizare μ are o **discontinuitate** în 0 care atrage după sine discontinuitatea funcției F pe frontiera lui A ! Cum de regulă punctul de optim x^* se află pe frontieră, metodele de minimizare bazate pe gradient nu sunt aplicabile deoarece gradientul funcției F nu este definit pe frontiera lui A .

Totuși ideea de a adăuga un termen la expresia funcției f care să penalizeze o eventuală "ieșire" din A și de a minimiza "liber" funcția rezultată poate fi salvată utilizând în locul funcției μ o funcție mai puțin "rigidă" care să aibe proprietățile de regularitate (continuitate, diferențiabilitate) reclamate de utilizarea metodelor de gradient. Un exemplu ar putea fi funcția:

cu ajutorul căreia construim problema de minimizare fără restricții

$$(P') \begin{cases} \inf F(x) = f(x) + \sum_{i \in M} \varphi(g_i(x)) \\ x \in R^n \end{cases}$$

Acum, penalizarea pentru ieșirea în afara mulțimii A nu mai este "infinită" astfel că soluția optimă a problemei (P') ar putea să nu fie în A , altfel spus ar putea încălca unele restricții. Pentru a diminua aceste încălcări amplificăm efectul penalizării înmulțind termenul de penalizare cu o constantă $r > 0$ suficient de mare. (P') devine:

$$(P'') \begin{cases} \inf F(x) = f(x) + r \sum_{i \in M} \varphi(g_i(x)) \\ x \in R^n \end{cases}$$

Introducerea multiplicatorului r mărește excentricitatea hipersuprafețelor de nivel ale funcției F ; or, este știut că excentricitatea pronunțată influențează negativ viteza de convergență a tuturor algoritmilor de minimizare nerestricționată.

Dificultățile semnalate au condus la următoarea schemă de lucru. În locul rezolvării unei singure probleme de minimizare fără restricții se va rezolva o secvență de asemenea probleme. Concret, se începe rezolvarea problemei (P'') cu o constantă r_0 nu prea mare; fie x^0 soluția optimă. Dacă $x^0 \notin A$ se reia (P'') cu un multiplicator $r_1 > r_0$ utilizând x^0 ca aproximație inițială. Fie x^1 noua soluție optimă. Dacă nici $x^1 \notin A$ schimbăm r_1 cu $r_2 > r_1$ etc. Se poate arăta că dacă $r_k \rightarrow \infty$ atunci șirul x^0, x^1, x^2, \dots converge către soluția optimă a problemei cu restricții (P) .

Să observăm că soluțiile intermediare x^0, x^1, x^2, \dots nu sunt admisibile și din acest punct de vedere metoda descrisă seamănă cu algoritmul simplex dual din programarea liniară. Acest fapt poate fi un serios dezavantaj deoarece dacă pentru anumite restricții nu sunt permise nici cele mai mici încălcări atunci nici o soluție intermediară nu poate fi reținută în caz de oprire a procesului iterativ.

§8 Condițiile de optimalitate Kuhn - Tucker în programarea convexă

8.1 Formularea condițiilor

Considerăm un **program convex** (P) pe care îl presupunem adus la următoarea formă zisă **canonică**:

$$(P) \begin{cases} \text{Să se determine } x^* \in R^n \text{ cu proprietatea:} \\ f(x^*) = \min f(x) \\ \text{minimul fiind luat după toți } x \in R^n \text{ care satisfac restricțiile:} \\ g_i(x) \leq 0 \quad i = 1, \dots, m \\ \text{și condițiile de nenegativitate:} \\ x \geq 0 \Leftrightarrow x_j \geq 0 \quad j = 1, \dots, n \end{cases}$$

Pentru simplitatea expunerii vom presupune că funcțiile convexe f și g_1, g_2, \dots, g_m sunt definite și diferentiabile în întreg spațiul R^n .

Asociem fiecărei restricții $g_i(x) \leq 0$ o variabilă nenegativă u_i și construim funcția:

$$L(x, u) = L(x_1, \dots, x_n; u_1, \dots, u_m) = f(x) + \sum_{i=1}^m u_i g_i(x)$$

Funcția L se numește **lagrangianul** problemei (P) iar u_1, \dots, u_m se numesc **multiplicatori Lagrange**. Se observă imediat că pentru $x \in \mathbb{R}^n$ fixat L este o funcție liniară în u_1, \dots, u_m iar pentru $u \geq 0$ fixat în \mathbb{R}^m , L este o funcție convexă și diferentiabilă.

Vom presupune îndeplinită următoarea condiție, numită **condiția de regularitate Slater**:

Există $\bar{x} \in \mathbb{R}^n$ cu proprietatea că $g_i(\bar{x}) < 0 \quad i = 1, \dots, m$ și $\bar{x}_j > 0 \quad j = 1, \dots, n$ altfel spus, mulțimea soluțiilor admisibile $A = \{x \in \mathbb{R}^n \mid g_i(x) \leq 0, x \geq 0\}$ are **interiorul relativ nevid**.

Cu aceste pregătiri putem enunța următoarea:

Teoremă (Kuhn - Tucker) Condiția necesară și suficientă ca $x^* \in \mathbb{R}^n$ să fie o soluție optimă a problemei (P) este să existe $u^* \in \mathbb{R}^m$ astfel încât cuplul (x^*, u^*) să verifice relațiile:

$\frac{\partial L}{\partial x_j} \geq 0 \quad (1_j)$	$x_j \frac{\partial L}{\partial x_j} = 0 \quad (1'_j)$	$i = 1, \dots, m$ $j = 1, \dots, n$
$\frac{\partial L}{\partial u_i} \leq 0 \quad (2_i)$	$u_i \frac{\partial L}{\partial u_i} = 0 \quad (2'_i)$	
$x_j \geq 0 \quad (3)$	$u_i \geq 0 \quad (3')$	

Condițiile încadrate sunt cunoscute sub numele de **condițiile de optimalitate Kuhn - Tucker**.

Deși este un rezultat de factură pur teoretică, teorema de mai sus este la baza multor algoritmi eficienți de rezolvare a programelor neliniare convexe cum sunt, de exemplu, cei utilizați în programarea patratrică.

Observație : Condiția de regularitate Slater intervine în probarea suficienței condițiilor de optimalitate K- T. Ea nu mai este necesară în cazul în care restricțiile programului (P) sunt liniare.

Exemplul 8.1 Considerăm programul neliniar patratric:

$$(P) \begin{cases} \max 2x_1 + x_2 - x_1^2 \\ x_1 + x_2 \leq 3 \\ 3x_1 - 2x_2 \leq 6 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

a cărui formă canonică este:

$$(P) \begin{cases} (\min) f = -2x_1 - x_2 + x_1^2 \\ x_1 + x_2 - 3 \leq 0 \\ 3x_1 - 2x_2 - 6 \leq 0 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

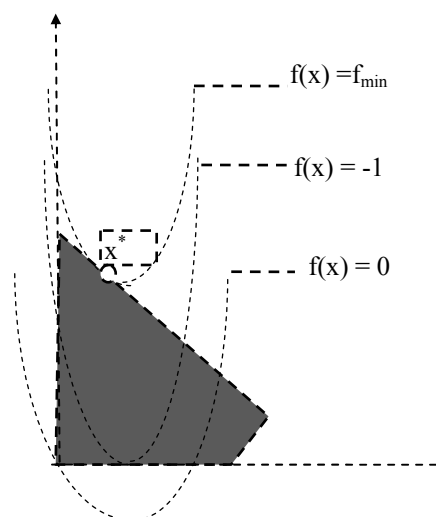


Figura 8.1

Fig. 8.1 confirmă faptul că (P) este un program convex: mulțimea soluțiilor admisibile este convexă, chiar poliedrală fiind definită prin inecuații liniare iar curbele de nivel $f(x) = c$ (constant) sunt parabole cu axa de simetrie comună $x = 1$. Desenul sugerează că soluția optimă x^* satisface cu egalitate restricția $x_1 + x_2 \leq 3$ și cu inegalitate strictă cealaltă restricție și condițiile de

nenegativitate. Aceste concluzii "calitative" și condițiile de optimalitate K - T ne vor permite să determinăm punctul x^* .

Asociem celor două restricții variabilele nenegative u_1 și u_2 și construim lagrangianul:

$$L = -2x_1 - x_2 + x_1^2 + u_1(x_1 + x_2 - 3) + u_2(3x_1 - 2x_2 - 6)$$

Scriem condițiile de optimalitate K - T:

$\frac{\partial L}{\partial x_1} \geq 0 \Leftrightarrow -2 + 2x_1 + u_1 + 3u_2 \geq 0 \quad (1.1)$	$x_1 \frac{\partial L}{\partial x_1} = 0 \Leftrightarrow x_1(-2 + 2x_1 + u_1 + 3u_2) = 0 \quad (1.1')$
$\frac{\partial L}{\partial x_2} \geq 0 \Leftrightarrow -1 + u_1 - 2u_2 \geq 0 \quad (1.2)$	$x_2 \frac{\partial L}{\partial x_2} = 0 \Leftrightarrow x_2(-1 + u_1 - 2u_2) = 0 \quad (1.2')$
$\frac{\partial L}{\partial u_1} \leq 0 \Leftrightarrow x_1 + x_2 - 3 \leq 0 \quad (2.1)$	$u_1 \frac{\partial L}{\partial u_1} = 0 \Leftrightarrow u_1(x_1 + x_2 - 3) = 0 \quad (2.1')$
$\frac{\partial L}{\partial u_2} \leq 0 \Leftrightarrow 3x_1 - 2x_2 - 6 \leq 0 \quad (2.2)$	$u_2 \frac{\partial L}{\partial u_2} = 0 \Leftrightarrow u_2(3x_1 - 2x_2 - 6) = 0 \quad (2.2')$
$x_1, x_2 \geq 0 \quad (3)$	$u_1, u_2 \geq 0 \quad (3')$

Reamintim interpretarea acestor condiții: $x^* = (x_1^*, x_2^*)$ este soluția optimă a programului (P) dacă și numai dacă există $u^* = (u_1^*, u_2^*)$ astfel încât cuplul (x^*, u^*) să satisfacă relațiile mai sus scrise. Deoarece la optim avem: $x_1 > 0$, $x_2 > 0$ și $3x_1 - 2x_2 < 6$ din relațiile (1.1'), (1.2') și (2.2') obținem: $-2 + 2x_1 + u_1 + 3u_2 = 0$, $-1 + u_1 - 2u_2 = 0$ și $u_2 = 0$ care împreună cu $x_1 + x_2 = 3$ constituie un sistem de patru ecuații cu patru variabile x_1 , x_2 , u_1 , u_2 . Rezultă ușor:

$$x^* = \left(\frac{1}{2}, \frac{5}{2}\right), \quad u^* = (1, 0) \text{ de unde } (min)f = -\frac{13}{4}.$$

8.2 Condițiile Kuhn - Tucker în programarea liniară

Evident, orice program liniar este un program convex și ca urmare teorema de optimalitate Kuhn - Tucker funcționează și pentru programele liniare. După cum vom vedea, în acest caz teorema amintită se suprapune peste un rezultat fundamental al dualității liniare.

Să considerăm un program liniar în **formă canonică de maximizare** (în notațiile uzuale):

$$(P) \begin{cases} \sum_{j=1}^n a_{ij}x_j \leq b_i & i = 1, \dots, m \\ x_j \geq 0 & j = 1, \dots, n \\ (max)f = \sum_{j=1}^n c_j x_j \end{cases} \Leftrightarrow \begin{cases} Ax \leq b \\ x \geq 0 \\ (max)f = cx \end{cases}$$

Rescriem (P) în forma canonică în care am studiat programele convexe și asociem celor m restricții variabilele u_1, u_2, \dots, u_m :

$$\begin{cases} \sum_{j=1}^n a_{ij}x_j - b_i \leq 0 \rightarrow u_i & i = 1, \dots, m \\ x_j \geq 0 \\ (min)-f = -\sum_{j=1}^n c_j x_j \end{cases} \Leftrightarrow \begin{cases} Ax - b \leq 0 \rightarrow u = (u_1, \dots, u_m) \\ x \geq 0 \\ (min)-f = -cx \end{cases}$$

Lagrangianul problemei (P) are expresia:

$$L(x_1, \dots, x_n; u_1, \dots, u_m) = -\sum_{j=1}^n c_j x_j + \sum_{i=1}^m u_i \left(\sum_{j=1}^n a_{ij} x_j - b_i \right) = -\sum_{i=1}^m u_i b_i + \sum_{j=1}^n \left(\sum_{i=1}^m u_i a_{ij} - c_j \right) x_j \Leftrightarrow$$

$$L(x, u) = -cx + u(Ax - b) = -ub + (uA - c)x$$

Condițiile de optimalitate K - T:

$\frac{\partial L}{\partial x_j} \geq 0 \Leftrightarrow \sum_{i=1}^m u_i a_{ij} - c_j \geq 0 \quad (1_j)$	$x_j \frac{\partial L}{\partial x_j} = 0 \Leftrightarrow \left(\sum_{i=1}^m u_i a_{ij} - c_j \right) x_j = 0 \quad (1'_j)$
$\frac{\partial L}{\partial u_i} \leq 0 \Leftrightarrow \sum_{j=1}^n a_{ij} x_j - b_i \leq 0 \quad (2_i)$	$u_i \frac{\partial L}{\partial u_i} = 0 \Leftrightarrow u_i \left(\sum_{j=1}^n a_{ij} x_j - b_i \right) = 0 \quad (2'_i)$
$x_j \geq 0 \quad (3)$	$u_i \geq 0 \quad (3')$

se interpretează astfel:

$x^* = (x_1^*, x_2^*, \dots, x_n^*) \in R^n$ este o soluție optimă a programului (P) dacă și numai dacă există $u^* = (u_1^*, u_2^*, \dots, u_m^*) \in R^m$ astfel încât cuplul (x^*, u^*) să verifice relațiile mai sus scrise.

Observăm că relațiile (2_i) $i = 1, \dots, m$ și (3) definesc x^* ca soluție admisibilă a problemei (P) în timp ce relațiile (1_j) $j = 1, \dots, n$ și (3') definesc u^* ca soluție admisibilă a **problemei duale**:

$$(Q) \begin{cases} \sum_{i=1}^m u_i a_{ij} \geq c_j & j = 1, \dots, n \\ u_i \geq 0 & i = 1, \dots, m \\ (\min) g = \sum_{i=1}^m u_i b_i \end{cases} \Leftrightarrow \begin{cases} uA \geq c \\ u \geq 0 \\ (\min) g = ub \end{cases}$$

Obținem următoarea reformulare a relațiilor K - T:

Condiția necesară și suficientă ca soluția admisibilă $x^* = (x_1^*, x_2^*, \dots, x_n^*)$ a problemei (P) să fie o soluție optimă este să existe o soluție admisibilă $u^* = (u_1^*, u_2^*, \dots, u_m^*)$ a problemei duale (Q) astfel încât cuplul (x^*, u^*) să verifice relațiile:

$$\begin{cases} \left(\sum_{i=1}^m u_i a_{ij} - c_j \right) x_j = 0 & j = 1, \dots, n \Leftrightarrow (uA - c)x = 0 \\ u_i \left(\sum_{j=1}^n a_{ij} x_j - b_i \right) = 0 & i = 1, \dots, m \Leftrightarrow u(Ax - b) = 0 \end{cases}$$

Am regăsit **teorema ecarterilor complementare** din teoria dualității liniare.

8.3 Condițiile de optimalitate Kuhn - Tucker în programarea patrată

Considerăm problema de programare patrată:

$$(P) \begin{cases} (\min) f = px + \frac{1}{2} x^T Cx \\ Ax \leq b \\ x \geq 0 \end{cases}$$

în care:

$$p = [p_1, p_2, \dots, p_n] \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad C = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & c_{22} & \dots & c_{2n} \\ \dots & \dots & \dots & \dots \\ c_{n1} & c_{n2} & \dots & c_{nn} \end{bmatrix} = \text{matrice simetrică}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

Vom presupune că matricea C este **pozitiv semidefinită**; știm atunci că funcția patrată f este convexă și ca urmare, programul (P) este convex.

Asociem blocului de restricții $Ax - b \leq b$ vectorul (linie) de multiplicatori Lagrange $u = [u_1, u_2, \dots, u_m]$ și construim lagrangianul problemei (P):

$$L(x, u) = px + \frac{1}{2} x^T Cx + u(Ax - b)$$

În scriere matricială, condițiile de optimalitate K - T pentru programul (P) arată astfel:

$\nabla_x L \geq 0 \Leftrightarrow p + x^T C + uA \geq 0 \quad (1)$	$(\nabla_x L) \cdot x = 0 \Leftrightarrow (p + x^T C + uA)x = 0 \quad (1')$
$\nabla_u L \leq 0 \Leftrightarrow Ax - b \leq 0 \quad (2)$	$u \cdot (\nabla_u L) = 0 \Leftrightarrow u(Ax - b) = 0 \quad (2')$
$x \geq 0 \quad (3)$	$u \geq 0 \quad (3')$

Transformăm inegalitățile (1) și (2) în egalități introducând vectorii de abatere:

$$v = [v_1, v_2, \dots, v_n] \stackrel{\text{def}}{=} p + x^T C + uA \geq 0 \quad \text{și} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \stackrel{\text{def}}{=} b - Ax \geq 0$$

Atunci:

$$x^T C + uA - v = -p \quad \text{și} \quad Ax + y = b$$

Se vede ușor că:

$$(1') \Leftrightarrow v \cdot x = 0 \Leftrightarrow v_j x_j = 0 \quad j = 1, \dots, n$$

$$(2') \Leftrightarrow u \cdot y = 0 \Leftrightarrow u_i y_i = 0 \quad i = 1, \dots, m$$

Cu aceste pregătiri putem formula următoarea interpretare a relațiilor K - T:

Condiția necesară și suficientă pentru ca $x^* \in R^n$ să rezolve programul patratric (P) este să existe $u^* \in R^m, v^* \in R^n, y^* \in R^m$ astfel încât (x^*, u^*, v^*, y^*) să verifice relațiile:

$$\begin{cases} x^T C + uA - v & = -p \\ Ax & + y = b \end{cases} \quad (4) \Leftrightarrow \begin{cases} \sum_{k=1}^n c_{kj} x_k + \sum_{i=1}^m a_{ij} u_i - v_j & = -p_j \quad j = 1, \dots, n \\ \sum_{k=1}^n a_{ik} x_k & + y_i = b_i \quad i = 1, \dots, m \end{cases}$$

$$x \geq 0, u \geq 0, v \geq 0, y \geq 0 \Leftrightarrow x_k \geq 0, u_i \geq 0, v_j \geq 0, y_i \geq 0$$

$$v x = 0 ; u y = 0 \quad (5) \Leftrightarrow v_j x_j = 0 \quad j = 1, \dots, n ; u_i y_i = 0 \quad i = 1, \dots, m$$

Se observă că (4) este un sistem liniar cu $m + n$ ecuații și $2m + 2n$ variabile. În concluzie:

Rezolvarea programului patratric (P) s-a redus la determinarea unei soluții admisibile (adică nenegative) a sistemului liniar (4) care să verifice relațiile de complementaritate (5).

În principiu, aceasta se poate face cu ajutorul algoritmului simplex în felul următor:

- se înmulțesc cu -1 acele egalități din (4) ai căror termeni liberi sunt < 0 ;
- dacă, după efectuarea operației precedente, matricea sistemului (4) nu conține o submatrice unitate de ordinul $m + n$, ea se completează cu coloanele care lipsesc adăugând - acolo unde este cazul - variabile artificiale nenegative. Astfel, sistemul (4) devine:

$$\begin{cases} x^T C + uA - v & + z^1 & = -p \\ Ax & + y & + z^2 = b \end{cases}$$

unde:

$$z^1 = (z_1^1, z_2^1, \dots, z_n^1) \geq 0 \quad \text{cu } z_j^1 = 0 \quad \text{daca } p_j \geq 0 \quad \text{și } z^2 = \begin{bmatrix} z_1^2 \\ z_2^2 \\ \vdots \\ z_m^2 \end{bmatrix} \geq 0 \quad \text{cu } z_i^2 = 0 \quad \text{daca } b_i \geq 0$$

- se rezolvă programul liniar:

$$\begin{cases} x^T C + uA - v & + z^1 & = -p \\ Ax & + y & + z^2 = b \\ x \geq 0, u \geq 0, v \geq 0, y \geq 0, z^1 \geq 0, z^2 \geq 0 \\ (\min) w = \sum_{j=1}^n z_j^1 + \sum_{i=1}^m z_i^2 \end{cases}$$

cu ajutorul algoritmului simplex, modificat cu următoarea regulă suplimentară care se referă la criteriul de intrare în bază:

La fiecare iterație, noua variabilă bazică va fi astfel aleasă încât:

- variabilele v_j și x_j să nu fie simultan bazice $j=1, \dots, n$;
- variabilele u_i și y_i să nu fie simultan bazice $i=1, \dots, m$.

La start se va pleca cu soluția:

$$x = 0, u = 0$$

$$v_j = \begin{cases} p_j & \text{daca } p_j \geq 0 \\ 0 & \text{daca } p_j < 0 \rightarrow z_j^1 = -p_j \end{cases}, \quad y_i = \begin{cases} b_i & \text{daca } b_i \geq 0 \\ 0 & \text{daca } b_i < 0 \rightarrow z_i^2 = -b_i \end{cases}$$

Deoarece $x = 0, u = 0$, relațiile de complementaritate (5) sunt verificate. Regula suplimentară ne asigură că la fiecare iterație:

$$v_j = 0 \text{ sau } x_j = 0 \text{ deci } v_j x_j = 0 \quad j = 1, \dots, n$$

$$u_i = 0 \text{ sau } y_i = 0 \text{ deci } u_i y_i = 0 \quad i = 1, \dots, m$$

Se poate arăta că dacă programul (P) este compatibil atunci într-un număr finit de iterații se ajunge la o soluție în care $(\min)w = 0 \Leftrightarrow$ toate variabilele artificiale introduse au valoarea zero. Este clar atunci că s-a obținut o soluție admisibilă a sistemului (4) care satisface relațiile de complementaritate (5). Componenta x^* a acestei soluții constituie soluția optimă a programului patratric (P).

Observație Considerațiile de mai sus constituie o descriere de principiu a algoritmului lui **Wolfe** pentru rezolvarea programelor patratrice convexe.

Exemplu numeric Vom arăta cum se aplică efectiv condițiile de optimalitate K - T și algoritmul simplex la rezolvarea programului patratric:

$$(P) \begin{cases} (\min) f = -15x_1 - 30x_2 - 4x_1x_2 + 2x_1^2 + 4x_2^2 \\ x_1 + 2x_2 \leq 30 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

Scriem matricial funcția obiectiv:

$$f(x_1, x_2) = [-15, -30] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{2} [x_1, x_2] \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Matricea simetrică $C = \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix}$ este chiar pozitiv definită astfel că f este o funcție strict convexă.

Lagrangianul problemei:

$$L(x_1, x_2) = -15x_1 - 30x_2 - 4x_1x_2 + 2x_1^2 + 4x_2^2 + u(x_1 + 2x_2 - 30)$$

Condițiile de optimalitate K - T:

$\frac{\partial L}{\partial x_1} = -15 - 4x_2 + 4x_1 + u \geq 0$	$x_1 \frac{\partial L}{\partial x_1} = 0 \Leftrightarrow x_1(-15 - 4x_2 + 4x_1 + u) = 0$
$\frac{\partial L}{\partial x_2} = -30 - 4x_1 + 8x_2 + 2u \geq 0$	$x_2 \frac{\partial L}{\partial x_2} = 0 \Leftrightarrow x_2(-30 - 4x_1 + 8x_2 + 2u) = 0$
$\frac{\partial L}{\partial u} = x_1 + 2x_2 - 30 \leq 0$	$u \frac{\partial L}{\partial u} = 0 \Leftrightarrow u(x_1 + 2x_2 - 30) = 0$
$x_1, x_2 \geq 0$	$u \geq 0$

Punem:

$$v_1 = \frac{\partial L}{\partial x_1} = -15 - 4x_2 + 4x_1 + u$$

$$v_2 = \frac{\partial L}{\partial x_2} = -30 - 4x_1 + 8x_2 + 2u$$

$$y = -\frac{\partial L}{\partial u} = 30 - x_1 - 2x_2$$

Condițiile K - T în forma (4) - (5):

$$\begin{cases} 4x_1 - 4x_2 + u - v_1 & = 15 \\ -4x_1 + 8x_2 + 2u & - v_2 & = 30 \\ x_1 + 2x_2 & & + y = 30 \end{cases} \quad (4)$$

$$x_1 \geq 0, x_2 \geq 0, u \geq 0, v_1 \geq 0, v_2 \geq 0, y \geq 0$$

$$x_1 \cdot v_1 = 0 \quad x_2 \cdot v_2 = 0 \quad u \cdot y = 0 \quad (5)$$

Știm că dacă (x^*, u^*, v^*, y^*) este o soluție admisibilă a sistemului (4) care satisface relațiile (5) atunci x^* este o soluție optimă a problemei date.

Introducem variabilele artificiale z_1 și z_2 în primele două egalități și rezolvăm programul liniar:

$$\begin{cases} (\min)w = z_1 + z_2 \\ 4x_1 - 4x_2 + u - v_1 & + z_1 & = 15 \\ -4x_1 + 8x_2 + 2u & - v_2 & + z_2 = 30 \\ x_1 + 2x_2 & & + y & = 30 \\ \text{Toate variabilele nenegative} \end{cases}$$

cu ajutorul algoritmului simplex, plecând de la soluția de bază inițială:

$$x_1 = x_2 = u = v_1 = v_2 = 0 \quad z_1 = 15 \quad z_2 = 30 \quad y = 30$$

care satisface vizibil condiția (5).

CB	VB	VVB	0	0	0	0	0	0	0	1	1	
			x_1	x_2	u	v_1	v_2	y	z_1	z_2		
1	z_1	15	4	-4	1	-1	0	0	1	0		ITERAȚIA 1 Poate intra x_2 deoarece $v_2 = 0$
1	z_2	30	-4	8	2	0	-1	0	0	1		
0	y	30	1	2	0	0	0	1	0	0		
	w	45	0	4	3	-1	-1	*	*	*		
1	z_1	30	2	0	2	-1	-1/2	0	1	1/2		ITERAȚIA 2 Poate intra x_1 deoarece $v_1 = 0$
0	x_2	15/4	-1/2	1	1/4	0	-1/8	0	0	1/8		
0	y	45/2	2	0	-1/2	0	1/4	1	0	-1/4		
	w	30	2	*	2	-1	-1/2	*	*	-1/2		
1	z_1	15/2	0	0	5/2	-1	-3/4	-1	1	3/4		ITERAȚIA 3 Poate intra u deoarece $y = 0$
0	x_2	75/8	0	1	1/8	0	-1/16	1/4	0	1/16		
0	x_1	45/4	1	0	-1/4	0	1/8	1/2	0	-1/8		
	w	15/2	*	*	5/2	-1	-3/4	-1	*	-1/4		
0	u	3	Nu mai este cazul!									
0	x_2	9										
0	x_1	12										
	w	0										

Prin urmare, o soluție a condițiilor de optimalitate K - T este $x^* = (12, 9)$ și $u^* = 3$. În concluzie

$$x_1^* = 12 \quad x_2^* = 9$$

este o soluție optimă a programului patratric dat de altfel și singura având în vedere faptul că funcția obiectiv f este strict convexă.

§ 9. Întrebări și probleme

1. Se dă o funcție numerică f definită în toate punctele unei mulțimi $C \subseteq \mathbb{R}^n$.

a) Ce înseamnă că mulțimea C este convexă? Presupunând C convexă, ce înseamnă că funcția f este convexă (strict convexă)?

b) Ce înseamnă că punctul $x^* \in C$ este un punct de minim local al funcției f ? Dar că x^* este un punct de minim global al funcției f pe C ?

c) Arătați că dacă C este o mulțime convexă iar f este o funcție strict convexă atunci f nu poate avea decât cel mult un punct de minim global pe C .

2. a) Se consideră funcția patratrică $\varphi(x) = x^T C x$, $x \in \mathbb{R}^n$ unde C este o matrice simetrică de ordinul n . Să se arate că pentru orice $x, y \in \mathbb{R}^n$ și orice $\lambda \in \mathbb{R}$ are loc identitatea:

$$\varphi((1 - \lambda)x + \lambda y) = (1 - \lambda)\varphi(x) + \lambda\varphi(y) - \lambda(1 - \lambda)\varphi(x - y)$$

b) Scrieți funcția patratrică:

$$f(x_1, x_2, x_3) = -x_1 + 2x_2 + x_3 + \frac{1}{2}x_1^2 - x_1x_2 + x_1x_3 + x_2^2 - x_2x_3 + x_3^2$$

în formatul matricial $f(x) = px + \frac{1}{2}x^T C x$, $x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \mathbb{R}^3$ și cercetați dacă f este o funcție convexă

(strict convexă)

3. Descrieți principiul metodelor de minimizare fără restricții

4. Elaborați programe de calculator pentru metodele de minimizare unidimensională prezentate în secțiunea 6.5

5. Se dă funcția patratrică $f(x_1, x_2) = -2x_1 - x_2 + \frac{1}{2}x_1^2 - x_1x_2 + 2x_2^2$.

a) Scrieți f în formatul matricial $f(x) = px + \frac{1}{2}x^T C x$, $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2$ și arătați că f este o funcție strict convexă.

b) Calculați gradientul ∇f și determinați minimul liber x^* al funcției f .

c) Determinați direcția celei mai rapide descreșteri a funcției f din $x^0 = (0,0)$ și prima aproximație x^1 din metoda gradientului.

d) Efectuați încă două iterații din metoda gradientului pentru minimizarea nerestricționată a funcției f . Comparați x^3 cu x^* obținut la b).

6. În procesul de minimizare nerestricționată a unei funcții numerice de trei variabile, efectuat cu metoda gradientului, printre direcțiile de deplasare s-au numărat și vectorii $s = (1, 1, -3)$, $s' = (2, -1, 1)$. Este posibil ca cele două direcții să fi fost obținute în două iterații consecutive?

7. Se consideră programul convex:

$$(P) \begin{cases} (\min) f = -3x_1 - 9x_2 + x_1^2 - x_1x_2 + x_2^2 \\ g(x_1, x_2) = x_1^2 + x_2^2 - 5 \leq 0 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

și punctul $\bar{x} = (1, 2)$ aflat pe frontiera mulțimii soluțiilor admisibile (deoarece $g(\bar{x}) = 0$)

a) Calculați gradientii funcțiilor f și g în \bar{x} .

b) Ce condiții trebuie să satisfacă un vector $s = (s_1, s_2) \in \mathbb{R}^2$ pentru a fi o direcție admisibilă și utilizabilă din \bar{x} ? Care din următorii vectori $s^1 = (-1, 1)$, $s^2 = (1, 1)$, $s^3 = (1, -1)$ satisfac aceste condiții?

8. Se dă programul neliniar patratric:

$$(P) \begin{cases} (\min) f = (x_1 - 4)^2 + (x_2 - 3)^2 \\ 3x_1 + 2x_2 \leq 12 \\ -2x_1 + 2x_2 \leq 3 \\ 2x_1 - x_2 \leq 4 \\ 2x_1 + 3x_2 \geq 6 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

a) Vizualizați mulțimea soluțiilor admisibile;

b) Ce formă au curbele de nivel ale funcției obiectiv? Stabiliți punctul de minim liber x^\emptyset al funcției f ;

c) Scrieți condițiile de optimalitate Kuhn - Tucker;

d) Stabiliți cu aproximație poziția soluției optime x^* a programului (P) și folosind condițiile K - T determinați efectiv x^* .

9. Se dă programul neliniar;

$$(P) \begin{cases} (\max) f = x_1^2 x_2 \\ x_1 + x_2^2 \leq 5 \\ 2x_1 - x_2 \geq 1 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

a) Reprezentați grafic mulțimea soluțiilor admisibile și curbele de nivel $f(x) = 1$, $f(x) = 4$ ale funcției obiectiv;

b) Aduceți programul (P) la forma canonică și scrieți condițiile de optimalitate K - T;

c) Determinați soluția optimă x^* a programului (P) știind că ea satisface cu egalitate numai prima restricție a acestuia.

10. În modelarea unui proces de stocare cu două produse și spațiu de depozitare limitat s-a ajuns la următorul program neliniar:

$$(P) \begin{cases} (\min) f(x_1, x_2) = x_1 + \frac{a_1^2}{x_1} + x_2 + \frac{a_2^2}{x_2} \\ x_1 + x_2 \leq I \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

unde $a_1, a_2, I > 0$ sunt constante.

a) Să se arate că f este o funcție convexă pe domeniul $\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 > 0, x_2 > 0\}$;
(se va observa că f este separabilă și se va cerceta convexitatea componentelor)

b) Scrieți condițiile de optimalitate K - T pentru programul convex (P);

c) Determinați soluția optimă a programului (P). Discuție.

11. Să se rezolve programul patratric:

$$(P) \begin{cases} (\min) f(x_1, x_2) = -x_1 - x_2 + x_1^2 - x_1 x_2 + \frac{1}{2} x_2^2 \\ x_1 + x_2 \leq 3 \\ 3x_1 + 2x_2 \geq 6 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

folosind condițiile de optimalitate K - T și algoritmul simplex (vezi secțiunea 8.2)