STUDIA UNIVERSITATIS MOLDAVIAE, 2020, nr.2(132)
Seria "Științe exacte și economice" ISSN 1857-2073 ISSN online 2345-1033 p.36-44

CZU: 519.85/.86 **DOI: http://doi.org/10.5281/zenodo.3978140**

ALGORITMI DE SOLUȚIONARE A PROBLEMELOR NELINIARE DE TRANSPORT CU MAI MULȚI INDICI

Tatiana PAȘA
Universitatea de Stat din Moldova

În articol este formulată și studiată problema neliniară de transport cu funcții concave de cost cu patru indici descrisă de surse, destinații, tipuri de produse și tipuri de transport care circulă prin rețea de transport. Tot aici este formulată și studiată problema neliniară de transport cu funcții concave de cost cu cinci indici descrisă de surse, destinații, tipuri de produse și tipuri de transport. Sunt prezentați algoritmi euristici care permit soluționarea acestor probleme și rezultatele testărilor pentru algoritmii implementați în limbajul Wolfram.
Cuvinte-cheie: *funcție concavă de cost, problemă neliniară de transport, rețea de transport, problemă de transport cu n indici.*

ALGORITHMS FOR SOLVING THE NONLINEAR MULTY-INDEX TRANSPORT PROBLEM
In this paper we formulate and study the nonlinear transport problem with concave cost functions with four indices described by sources, destinations, types of products and types of transport circulating through the transport network. We also formulate and study the nonlinear transport problem with concave cost functions with five indices described by sources, destinations, types of products and types of transport. We present several heuristic algorithms that solve these problems and present the test results for the algorithms implemented in the Wolfram Language.
Keywords: *concave cost function, nonlinear transport problem, transport network, multy-index transport problem.*

Introducere

În zilele noastre, problema transportului produselor este una actuală, avându-se în vedere globalizarea și faptul că oamenii tind să cumpere produse din diferite zone ale lumii, iar prețul depinde de mai mulți factori care trebuie luați în calcul. În problema de transport se încadrează un număr larg de probleme din viața reală în care ca elemente de bază care o descriu sunt sursele și destinațiile. Din practică se cunoaște că, în cele mai dese cazuri, problema de transport poate fi descrisă de tipul de produse transportate, de tipul de transport utilizat, de distanța parcursă sau de tipul ambalajului utilizat.

Managementul aprovizionării devine tot mai important pentru funcționarea eficientă a companiilor. Problemele care presupun transportul produselor pot varia în dependență de tipurile de produse transportate (materiale de construcții, produse alimentare, produse petroliere), de numărul și tipul transportului utilizat pentru transportare [1] (tren, vapor, camion, avion), de numărul de surse și destinații, dar și de calea de transport (aer, apă, drumuri, țevi, cabluri) aleasă de transportator.

Cazul problemelor cu mai mulți indici a fost formulat de Ph.-X. Ninh [2], care a propus și o metodă exactă de soluționare, o extensie a metodei potențialilor prin coordonarea soluționării problemei primare și duale. Un rezultat important este formularea și demonstrarea teoremei condiției necesare și suficiente de existență a soluției. Printre alți autori care au studiat problema de transport cu mai mulți indici se numără K.B. Hayley [3], care a prezentat un algoritm de rezolvare și W.Junginer [4], care a efectuat un studiu privind caracteristicile problemei multiindex și a propus un șir de probleme logice pentru soluționarea problemelor de transport cu mai mulți indici.

I. Problema de transport cu 4 indici

În cele ce urmează, este considerată problema de transport pe rețea cu funcții concave de cost descrisă de:

- n surse $A_1, A_2, \dots, A_n$ care posedă cantitățile de produs respectiv $\alpha_1, \alpha_2, \dots, \alpha_n$;
- m destinații $B_1, B_2, \dots, B_m$ cu necesitățile respective de produs $\beta_1, \beta_2, \dots, \beta_m$;
- p tipuri de produse $P_1, P_2, \dots, P_p$ cu cantitățile respective $\gamma_1, \gamma_2, \dots, \gamma_p$;
- q tipuri de transport $T_1, T_2, \dots, T_q$ cu capacitățile de transport respective $\delta_1, \delta_2, \dots, \delta_q$.

Costul de transport este descris de funcții neliniare $\varphi_{ijkl}(x_{ijkl})$ care depind de cantitatea de produs transportat de la surse la destinații. Fluxul x_{ijkl} descrie cantitatea de produs P_l transportat din sursa A_i în destinația B_j utilizând tipul de transport T_k. Scopul problemei este de a minimiza costul total de transport al tuturor produselor disponibile în surse pentru aprovizionarea tuturor destinațiilor. Această problemă face parte din grupul de probleme de transport cu 4 indici (PT4I), unde în calitate de cei 4 indici sunt: sursele, destinațiile, tipurile de transport și tipurile de produse.

Problema neliniară de transport cu 4 indici (PNT4I) constă în determinarea unui flux x^* ce minimizează funcția:

$$F(x) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{p}\sum_{l=1}^{q} \varphi_{ijkl}(x_{ijkl}),$$

unde $\varphi_{ijkl}(x_{ijkl})$ sunt funcții concave secvențial-liniare nedescrescătoare de cost, adică se cere soluționarea problemei neliniare:

$$F(x) \to min, \tag{1}$$

$$\begin{cases} \sum_{j=1}^{m}\sum_{k=1}^{p}\sum_{l=1}^{q} x_{ijkl} = \alpha_i, \quad i = \overline{1,n}, \\ \sum_{i=1}^{n}\sum_{k=1}^{p}\sum_{l=1}^{q} x_{ijkl} = \beta_j, \quad j = \overline{1,m}, \\ \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{l=1}^{q} x_{ijkl} = \gamma_k, \quad k = \overline{1,p}, \\ \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{p} x_{ijkl} = \delta_l, \quad l = \overline{1,q}, \\ x_{ijkl} \geq 0, \quad (i,j,k,l), \end{cases} \tag{2}$$

pentru care sunt respectate condițiile de pozitivitate; deci, mărimile $\alpha_1, \alpha_2, \dots, \alpha_n$, $\beta_1, \beta_2, \dots, \beta_m$, $\gamma_1, \gamma_2, \dots, \gamma_p$ și $\delta_1, \delta_2, \dots, \delta_q$ sunt pozitive și nenule.

Forma generală a soluției optime pentru PNT4I (1) – (2) este $x^* = (x^*_{1111}, x^*_{1112}, \dots, x^*_{nmpq})$. În [5] sunt descrise proprietățile problemei și condiția de echilibru.

Problemele din acest grup descriu situația reală ce ține de activitatea unei întreprinderi, fapt ce provoacă interesul cercetătorilor de a le studia. Deoarece funcția de cost a problemei formulate este o funcție separabilă și neliniară, iar restricțiile sunt funcții separabile liniare, problema este neliniară separabilă și poate fi soluționată utilizând metodele respective.

Condiții suficiente pentru un optim global al problemei programării matematice la minim ar fi ca funcția obiectiv să fie convexă și mulțimea soluțiilor admisibile să fie o mulțime convexă. În caz că funcția obiectiv este concavă, iar mulțimea soluțiilor admisibile este mulțime convexă, putem vorbi doar despre un minim local. Metodele de soluționare a unor astfel de probleme presupun soluționarea unei sau a câtorva probleme ale programării liniare generate de problema inițială. Ele au la bază aproximarea funcției neliniare cu o funcție liniară sau secvențial-liniară și aplicarea metodei simplex sau simplex-modificată, în dependență de caz.

În cele ce urmează, soluționarea unei astfel de probleme presupune reduceri succesive ale problemei neliniare formulate la probleme de programare liniară. Pentru soluționarea problemei liniare, care are aceleași restricții ca și problema neliniară, poate fi aplicată metoda simplex adaptată cazului problemei cu 4 indici, ca, de exemplu, cea propusă de A.Djamel et al. [6] sau de R.Zitouni et al. [7].

Descrierea algoritmului E1

Pasul 1. Se construiește o soluție admisibilă inițială $x^0 = (x^0_{1111}, x^0_{1112}, \dots, x^0_{nmpq})$ a sistemului (2).

Pasul 2. Se determină valoarea funcției neliniare în x^0:

$$F(x^0) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{p}\sum_{l=1}^{q} \varphi_{ijkl}(x^0_{ijkl}),$$

și se calculează valoarea coeficienților:

$$C_{ijkl} = \begin{cases} \dfrac{\varphi_{ijkl}(x^0_{ijkl})}{x^0_{ijkl}}, & x^0_{ijkl} > 0, \\ \varphi'_{ijkl}(0), & x^0_{ijkl} = 0, \end{cases}$$

pentru fiecare (i, j, k, l).

Pasul 3. Se soluționează problema liniară de transport:

$$Z(x_{ijkl}) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{p}\sum_{l=1}^{q} C_{ijkl}x_{ijkl} \to min,$$

cu restricțiile (2) și se obține soluția optimă $x^1 = (x^1_{1111}, x^1_{1112}, \dots, x^1_{nmpq})$.

Pasul 4. Se compară valorile $Z(x^1)$ și $F(x^0)$. Dacă $Z(x^1) < F(x^0)$ sau $Z(x^1) = F(x^0)$ și $x^1 \neq x^0$, atunci x^0 se substituie cu x^1 și se trece la *Pasul 2*, ceea ce înseamnă că se repornește procedura de liniarizare cu o altă soluție inițială. Dacă $Z(x^1) > F(x^0)$ sau $Z(x^1) = F(x^0)$ și $x^1 = x^0$, atunci soluția optimă a problemei neliniare este considerată valoarea $x^* = x^0$. **STOP.**

În cazul liniarizării funcției obiectiv, soluția optimă este dependentă de soluția admisibilă inițială utilizată pentru aproximare, deci trebuie aleasă cu mare grijă. În cazul problemelor de dimensiuni mari, este complicat a realiza o alegere corectă, de aceea se utilizează o soluție aleatoare, ceea ce nu garantează obținerea soluției optime globale, ci doar a unui pseudo optim.

Se formulează și se demonstrează următoarele teoreme:

Teoremă 1. *Algoritmul E1 converge către un optim local.*

Demonstrație. *Algoritmul E1 pornește de la o soluție admisibilă care este unul dintre vârfurile poliedrului ce descrie mulțimea soluțiilor admisibile care satisfac sistemul de restricții liniare ale problemei neliniare cu funcții concave de cost. Soluția problemei liniare, care este problemă relaxată, satisface aceleași restricții, deci face parte din același domeniu de definiție și este unul dintre vârfurile poliedrului. Prin urmare, se poate spune că algoritmul converge către un optim local din domeniul de definiție a problemei inițiale neliniare.*□

Teoremă 2. *Algoritmul E1 necesită un volum de memorie de ordinul* $O(nmpq(n+m+p+q))$.

Demonstrație. *Graful* $G = (V, E)$ *este descris de o matrice de adiacență de dimensiunea* $nmpq \times (n+m+p+q)$. *Matricea care conține coeficienții liberi ai sistemului de restricții este de dimensiunea* $n+m+p+q$. *Soluția problemei este un tabel de dimensiunea* $nmpq$. *Prin urmare, implementarea algoritmului necesită memorie de ordinul* $O(nmpq(n+m+p+q))$.□

Realizarea algoritmului în limbajul Wolfram

Sistemul Wolfram permite implementarea algoritmului, în care se pot utiliza un șir de funcții standard care simplifică semnificativ problema. Pentru obținerea soluției inițiale a algoritmului se aplică funcția standard *FindInstace[]* care soluționează sistemul de ecuații și, ca rezultat, livrează o soluție admisibilă din mulțimea de soluții ale sistemului de restricții. *LinearPrograming[]* este o funcție standard care permite soluționarea problemei de programare liniară.

Rezultatele prezentate în tabelele 1 și 2 sunt bazate pe probleme de diferite dimensiuni, unde funcțiile de cost sunt funcții concave secvențial-liniare. Rețelele sunt descrise de n – surse, m – destinații, p – tipuri de produse transportate, q – tipuri de transport utilizat. Funcția de producere și consum presupune că totalul produselor disponibile în surse este egal cu totalul produselor necesare în destinații.

În Tabelul 1 poate fi vizualizat timpul de execuție a algoritmului E1 pentru probleme de diferite dimensiuni, numărul de surse, destinații, tipuri de produse și tipuri de transport, pentru a căror soluționare sunt utilizate de la 24 de necunoscute pentru problema descrisă de 2 surse, 2 destinații, 2 tipuri de produse și 3 tipuri de transport până la 2401 necunoscute pentru problema descrisă de 7 surse, 7 destinații, 7 tipuri de produse și 7 tipuri de transport.

Tabelul 1

Timpul de execuție pentru algoritmul E1

n/m/p/q (secunde)	**2/2/2/3**	**2/2/3/3**	**2/3/3/3**	**3/3/3/3**	**3/3/3/4**	**3/3/4/4**	**3/4/4/4**	**4/4/4/4**
	0.0156 0.0155 0.0156 0.0156 0.0155	0.6250 0.6250 0.6250 0.0781 0.6125	0.1718 0.1093 0.1250 0.0937 0.1250	0.2031 0.2187 0.1875 0.2031 0.2031	0.3437 0.3125 0.3281 0.3125 0.3327	0.5312 0.5468 0.5312 0.5156 0.4843	0.8906 0.9062 0.9062 0.9218 0.8750	3.1093 3.1875 3.2968 3.2031 3.1562
Necunoscute	24	36	54	81	108	144	192	256
n/m/p/q (secunde)	**4/4/4/5**	**4/4/5/5**	**4/5/5/5**	**5/5/5/5**	**5/5/5/6**	**5/5/6/6**	**6/6/7/7**	**7/7/7/7**
	4.5000 4.5937 4.7187 5.6093 4.7500	6.7343 6.9062 6.7187 6.7812 6.9218	9.6250 9.6406 9.8593 9.8125 9.6718	15.3125 15.2031 15.0938 15.2188 15.3594	25.0781 25.1719 25.3750 25.3438 25.4063	33.3750 33.1563 33.7344 33.6563 33.6406	178.5310 206.1250 198.1553 215.8750 209.5278	474.141 483.000 477.814 490.328 486.512
Necunoscute	320	400	500	625	750	900	1764	2401

Timpul de execuție în acest caz depinde și de soluția admisibilă inițială obținută la *Pasul 1*, deoarece și numărul de iterații depinde de aceasta. În cazul problemelor de dimensiuni mari, se poate face o alegere a acestei soluții din insuficiență de timp care ar putea lua acest procedeu și se operează cu o soluție admisibilă aleatorie. Astfel, algoritmul generează o soluție pseudo optimă a problemei formulate.

În Tabelul 2 sunt prezentate mai multe probleme de diferite dimensiuni pentru care valorile funcțiilor $F0$ și $Z1$ se modifică de la o iterație la alta. Analizând datele se poate observa că numărul de iterații necesar pentru obținerea soluției pseudo optime a problemei formulate nu depinde de dimensiunea problemei, ci de soluția admisibilă inițială cu care începe procesul de reduceri succesive ale problemei neliniare la probleme liniare.

Tabelul 2

Modificarea *F0* și *Z1* pentru algoritmul E1

n/m/p/q / **Iterația**	***F0 / Z1*** (u. c.)					
	4/4/4/4	**4/4/4/4**	**4/4/4/4**	**5/5/5/5**	**5/5/5/5**	**5/5/5/5**
I	27 / 20.80	28 / 19.91	29 / 23.76	34 / 26.87	33 / 22.36	34 / 27.9
II	20 / 16.70	18 / 16.47	22 / 19.5	19 / 18.94	20.6 / 19.30	22.66 / 21.25
III	14.5 / 14	15 / 13.91	15.5 / 15.1	18 / 17.4	17 / 16.83	21 / 20.96
IV	13.25 / 13.23	12.66 / 12.37	15 / 14.70	17.5 / 16.7	17 / 17	21 / 20.97
V	13.4 / 13.4	12 / 12	14.5 / 14.47	17 / 16.98	-	21 / 21
VI	-	12 / 12	14.3 / 14.3	16 / 16	-	-
VII	-	-	-	15.5 / 15.33	-	-
VIII	-	-	-	15 / 15	-	-

IX	-	-	-	15 / 14.83	-	-
1X	-	-	-	14 / 14	-	-
	6/6/6/6	**6/6/6/6**	**6/6/6/6**	**7/7/7/7**	**7/7/7/7**	**7/7/7/7**
I	47 / 29.06	41 / 29.78	43 / 30.66	43 / 26.16	52 / 34.83	43 / 36.48
II	21 / 17.93	27.1 / 23.03	24 / 23.2	23.33 / 20.87	26.33 / 25.21	30.6 / 29,27
III	18 / 17.50	18.5 / 18.33	22.33 / 20.43	17 / 16.33	23 / 23	27 / 27.2
IV	16.75 / 15.8	18 / 18	16.66 / 16.5	16 / 16	23 / 22.6	27 / 27
V	15 / 14.39	17.16 / 16.91	16 / 16	-	21.5 / 21	26.5 / 26.38
VI	13 / 13	16 / 16	15 / 15	-	20 / 20	26 / 26
VII	13 / 13	-	-	-	20 / 20	-

De la o iterație la alta, valoarea $F0$ este mai mică sau egală decât valoarea $F0$ obținută la iterația precedentă. Cazul în care $F0$ rămâne nemodificat pe parcursul a câtorva iterații demonstrează încă o dată că problemele de acest fel pot avea mai multe minime locale. Aceste puncte permit diferite aproximări ale funcției neliniare cu una liniară, ceea ce duce la o ieșire din blocaj, fiind obținut un alt minim local cu valoarea funcției de cost mai mică.

Aceeași problemă este discutată și în [8], unde autorul propune aplicarea unui algoritm genetic de soluționare.

II. Problema de transport pe rețea cu 5 indici

În cele ce urmează, se consideră problema descrisă de:

– n surse $A_1, A_2, \dots, A_n$ care posedă cantitățile de produs respective $\alpha_1, \alpha_2, \dots, \alpha_n$;
– m destinații $B_1, B_2, \dots, B_m$ cu necesitățile respective de produs $\beta_1, \beta_2, \dots, \beta_m$;
– p tipuri de produse $P_1, P_2, \dots, P_p$ de cantitățile respective $\gamma_1, \gamma_2, \dots, \gamma_p$;
– q tipuri de transport $T_1, T_2, \dots, T_q$ cu capacitățile respective $\delta_1, \delta_2, \dots, \delta_q$;
– g noduri (puncte) intermediare $v_r \in V_{int}$ în care nu are loc nici producerea, nici consumul din cantitatea de produs.

Costul de transport pe rețea este descris de o funcție neliniară $\varphi(x)$, pentru fiecare arc $e \in E$, care depinde de volumul fluxului de produs transportat. Scopul problemei este minimizarea costului total de transport al tuturor tipurilor de produse disponibile în surse pentru aprovizionarea tuturor destinațiilor cu necesarul respectiv utilizând toate tipurile de transport cu capacități diferite disponibile în rețea.

În acest caz, rețeaua de transport este descrisă de un graf orientat aciclic $G = (V, E)$, cu mulțimea de vârfuri $V = V_s \cup V_t \cup V_{int}$, unde V_s – mulțimea de surse, $|V_s| = n$, V_t – mulțimea de destinații $|V_t| = m$ și V_{int} – mulțimea de puncte intermediare $|V_{int}| = g$, iar E – mulțimea de arce, $|E| = u$. Această problemă face parte din grupul de probleme neliniare de transport cu 5 indici descrisă de surse, destinații, puncte intermediare, tipuri de produse și tipuri de transport.

Pentru a simplifica formularea problemei, dar și descrierea/implementarea algoritmului de soluționare, problema neliniară de transport descrisă de surse, destinații, tipuri de produse, tipuri de transport și puncte intermediare este formulată ca o problemă neliniară de transport pe rețea cu 3 indici (PNTR3I) descrisă de arce, tipuri de produse și tipuri de transport.

PNTR3I constă în determinarea unui flux x^* care minimizează funcția de cost:

$$F(x) = \sum_{e \in E} \sum_{k=1}^{p} \sum_{l=1}^{q} \varphi_{ekl}(x_{ekl}), \tag{3}$$

unde $\varphi_{ekl}(x_{ekl})$ sunt funcții concave secvențial-liniare nedescrescătoare de cost, adică se cere soluționarea problemei neliniare:

$$F(x) \to min, \tag{4}$$

$$\begin{cases} \sum_{e\in E^-(V_s)} \sum_{l=1}^{q} x_{ekl} = \gamma_k, \quad k = \overline{1,p}, \\ \sum_{e\in E^-(V_s)} \sum_{k=1}^{p} x_{ekl} = \delta_l, \quad l = \overline{1,q}, \\ \sum_{e\in E^-(s_i)} \sum_{k=1}^{p} \sum_{l=1}^{q} x_{ekl} = \alpha_i, \quad i = \overline{1,n}, \\ \sum_{e\in E^+(d_j)} \sum_{k=1}^{p} \sum_{l=1}^{q} x_{ekl} = \beta_j, \quad j = \overline{1,m}, \\ \sum_{e\in E^+(v_r)} \sum_{l=1}^{q} x_{ekl} - \sum_{e\in E^-(v_r)} \sum_{l=1}^{q} x_{ekl} = 0, \quad r = \overline{1,g}, k = \overline{1,p}, \\ \sum_{e\in E^+(v_r)} \sum_{k=1}^{p} x_{ekl} - \sum_{e\in E^-(v_r)} \sum_{k=1}^{p} x_{ekl} = 0, \quad r = \overline{1,g}, l = \overline{1,q}, \\ x_{ekl} \geq 0, \quad \forall (e,k,l), \end{cases} \tag{5}$$

pentru care sunt respectate condițiile de pozitivitate, deci mărimile $\alpha_1, \alpha_2, \dots, \alpha_n, \beta_1, \beta_2, \dots, \beta_m, \gamma_1, \gamma_2, \dots, \gamma_p$ și $\delta_1, \delta_2, \dots, \delta_q$ sunt pozitive și nenule.

Forma generală a soluției optime a *PNTR3I* are forma $x^* = (x^*_{111}, x^*_{112}, \dots, x^*_{upq})$ pentru o rețea descrisă de u arce.

În cele ce urmează drept scop este liniarizarea funcției obiectiv, deci se reduce soluționarea unei probleme a programării neliniare la o problemă a programării liniare. Pentru soluționarea problemei liniare care are aceleași restricții ca și problema neliniară poate fi aplicată metoda simplex.

Descrierea algoritmului E2

În cele ce urmează se propune algoritmul E2 de soluționare a problemei formulate care presupune reduceri succesive ale problemei neliniare la probleme liniare pentru care se cunosc metode de soluționare.

Algoritmul E2

Pasul 1. Se construiește o soluție admisibilă inițială $x^0 = (x^0_{111}, x^0_{112}, \dots, x^0_{upq})$ a sistemului de restricții (5).

Pasul 2. Se determină valoarea funcției în x^0:

$$F(x^0) = \sum_{e\in E} \sum_{k=1}^{p} \sum_{l=1}^{q} \varphi_{ekl}(x^0_{ekl}),$$

și se calculează valoarea coeficienților:

$$C_{ekl} = \begin{cases} \dfrac{\varphi_{ekl}(x^0_{ekl})}{x^0_{ekl}}, & x^0_{ekl} > 0, \\ \varphi'_{ekl}(0), & x^0_{ekl} = 0, \end{cases}$$

pentru orice (e, k, l).

STUDIA UNIVERSITATIS MOLDAVIAE, 2020, nr.2(132)
Seria "Științe exacte și economice" *ISSN 1857-2073* *ISSN online 2345-1033* *p.36-44*

Pasul 3. Se soluționează problema liniară de transport:

$$Z(x_{ekl}) = \sum_{e \in E} \sum_{k=1}^{p} \sum_{l=1}^{q} C_{ekl} x_{ekl} \rightarrow min,$$

cu restricțiile (5) și se obține soluția optimă $x^1 = (x_{111}^1, x_{112}^1, \ldots, x_{upq}^1)$ pentru o rețea descrisă de u arce.

Pasul 4. Se compară valorile $Z(x^1)$ și $F(x^0)$. Dacă $Z(x^1) < F(x^0)$ sau $Z(x^1) = F(x^0)$ și $x^1 \neq x^0$, atunci x^0 se substituie cu x^1 și se trece la *Pasul 2,* ceea ce înseamnă că se repornește procedura de liniarizare cu o altă soluție inițială. Dacă $Z(x^1) > F(x^0)$ sau $Z(x^1) = F(x^0)$ și $x^1 = x^0$, atunci soluția optimă a problemei neliniare este considerată valoarea $x^* = x^0$, se păstrează mărimea $F(x^0)$ și $x^* = x^0$ care îi corespunde și **STOP**.

În cazul liniarizării funcției obiectiv soluția optimă depinde de soluția admisibilă inițială utilizată pentru aproximare. Pentru probleme de dimensiuni mari este complicat a alege o soluție potrivită care ar garanta obținerea optimului global. Din acest motiv, pentru soluționarea problemei este utilizată o soluție admisibilă aleatorie care permite generarea unui pseudo optim.

Teoremă 3. *Algoritmul E2 converge către un optim local.*

Demonstrație. *Algoritmul E2 pornește de la o soluție admisibilă care este unul dintre vârfurile poliedrului ce descrie mulțimea soluțiilor admisibile care satisfac sistemul de restricții liniare ale problemei neliniare cu funcții concave de cost. Soluția problemei liniare, care este problemă relaxată, satisface aceleași restricții, deci face parte din domeniul de definiție și este unul dintre vârfurile poliedrului. Prin urmare, se poate spune că algoritmul converge către un optim local din domeniul de definiție a problemei inițiale neliniare.*□

Teoremă 4. *Algoritmul E2 necesită un volum de memorie de ordinul* $O(upq(n+m+g(p+q)))$.

Demonstrație. *Graful* $G = (V, E)$ *este descris de o matrice de adiacență de dimensiunea* $upq(n+m+2(p+q)+g(p+q))$. *Matricea care conține coeficienții liberi este de dimensiunea* $n+m+2(p+q)+g(p+q)$. *Soluția problemei este un tabel de dimensiunea* upq. *Ca urmare, implementarea algoritmului necesită memorie de ordinul* $O(upq(n+m+g(p+q)))$.□

Realizarea algoritmului în limbajul Wolfram

Rezultatele prezentate în tabelele 3 și 4 sunt bazate pe probleme de diferite dimensiuni, unde funcțiile de cost sunt funcții concave secvențial-liniare. Rețelele sunt descrise de n – surse, m – destinații, p – tipuri de produse transportate, q – tipuri de transport utilizate și g – puncte intermediare. Conform funcției de producere și consum, totalul produselor disponibile în surse este egal cu totalul produselor necesare în destinații și cu totalul de tipuri de produse care coincide cu capacitatea totală a tipurilor de transport din rețea.

Tabelul 3

Timpul de execuție pentru algoritmul E2

n/m/p/q/g (secunde)	**2/2/2/2/2**	**2/2/3/3/2**	**2/3/3/3/3**	**3/3/3/3/3**	**3/3/4/4/3**	**3/4/4/4/4**
	0.1093 0.1250 0.1250 0.1132 0.1520	0.4218 0.4062 0.3593 0.4531 0.3125	0.8437 2.5000 1.0156 2.0001 0.9687	3.6875 3.0937 3.1562 3.1875 2.8437	8.7343 13.4844 12.0938 10.1563 11.5000	22.7031 22.5313 23.2344 22.9688 31.2500
Necunoscute	48	102	234	261	432	896
n/m/p/q/g (secunde)	**4/4/4/4/4**	**4/4/5/5/4**	**4/5/5/5/5**	**5/5/5/5/5**	**5/5/6/6/5**	**6/6/6/6/6**
	31.4844 34.7500 38.2500 48.0000 38.2969	108.6090 95.4063 100.0160 105.8440 88.1075	226.3750 208.5160 211.8280 236.0780 205.8130	346.4690 308.8910 336.7750 321.4250 303.2660	783.063 816.953 873.813 802.891 848.469	2216.44 2217.25 2208.83 2107.88 2294.14
Necunoscute	864	1240	2225	2125	2850	4428

În Tabelul 3 este dat timpul de execuție a algoritmului E2 pentru probleme de diferite dimensiuni, pentru a căror soluționare sunt utilizate de la 48 de necunoscute pentru problema cu 2 surse, 2 destinații, 2 tipuri de produs, 2 tipuri de transport și 2 puncte intermediare până la 4428 de necunoscute pentru problema cu 6 surse, 6 destinații, 6 tipuri de produs, 6 tipuri de transport și 6 puncte intermediare.

După cum se observă, algoritmul permite obținerea unor soluții pseudo optime în timp rezonabil în cazul rețelelor de transport de dimensiuni mari. Soluția furnizată de algoritm depinde de soluția admisibilă generată la *Pasul 1* de inițializare a algoritmului. De soluția inițială depinde numărul de iterații necesar pentru soluționarea problemei neliniare și mai puțin de dimensiunea acesteia.

În Tabelul 4 sunt prezentate rezultatele executării mai multor probleme de transport pe rețea de diferite dimensiuni, pentru care se observă cum se modifică de la o iterație la alta valorile $F0$ și $Z1$ care descriu valoarea funcției neliniare de transport în x^0 și a funcției liniare în x^1 respectiv.

Tabelul 4

Modificarea *F0* / *Z1* pentru algoritmul E2

n/m/p/q/g / Iterația	***F0* / *Z1*** (u. c.)					
	3/3/3/3/3	**3/3/3/3/3**	**3/3/3/3/3**	**4/4/4/4/4**	**4/4/4/4/4**	**4/4/4/4/4**
1	49 / 41.96	67 / 51.39	58 / 42.90	74 / 54.79	65 / 61.66	62 / 46.42
2	41 / 34.67	46 / 44.67	39 / 35.36	50 / 50	57 / 55.70	44 / 39.36
3	29 / 29	35.5 / 31.33	35 / 33.78	46 / 44.66	47.5 / 45.8	39 / 39
4	-	29.66/ 29.66	31 / 31	44 / 42.76	41 / 41	37 / 37
5	-	-	-	44 / 44	39 / 39	-
6	-	-	-	44 / 44	-	-
	5/5/5/5/5	**5/5/5/5/5**	**5/5/5/5/5**	**6/6/6/6/6**	**6/6/6/6/6**	**6/6/6/6/6**
1	69 / 54.82	90 / 78	82 / 63.94	93 / 72.86	106 / 84.60	36 / 24.89
2	53 / 51.47	69 / 69	56.5 / 54.07	66.2 / 65.68	70.5 / 63.16	20.5 / 19.42
3	53 / 50.78	67 / 66.33	51 / 51	58 / 56.64	49 / 49	18 / 17.56
4	50 / 49.33	65 / 65	49.66 / 49	53 / 53	49 / 49	18 / 17.3
5	44.66 / 44.12	65 / 65	45 / 45	53 / 53	-	16 / 14.08
6	44 / 44	-	45 / 45	-	-	14 / 13.95
7	44 / 44	-	-	-	-	13.5 / 13.33
8	-	-	-	-	-	13 / 13
9	-	-	-	-	-	13 / 13

După cum se observă din datele prezentate în Tabelul 4, numărul de iterații până la îndeplinirea condiției de oprire a algoritmului nu depinde de dimensiunea problemei, ci doar de soluția inițială de la *Pasul 1*. De la o iterație la alta, valoarea $F0$ este mai mică sau egală decât valoarea $F0$ obținută la iterația precedentă. Cazul când $F0$ rămâne nemodificat pe parcursul a câtorva iterații demonstrează că problemele de acest tip pot avea mai multe minime locale. Pe de altă parte, aceste puncte permit diferite aproximări ale funcției neliniare cu una liniară, ceea ce duce la o ieșire din blocaj și la obținerea unui alt minim local cu valoarea funcției de cost mai mică. În cazul în care $F0$ și $Z1$ coincid, dar algoritmul nu se stopează, putem spune că aceste valori sunt obținute în puncte diferite.

Testele sunt efectuate pe o mașină Intel i5-2500 cu 4 Cores și 8 GB memorie DDR3 în Wolfram Mathematica 12.

Concluzii

1. Algoritmul E1 permite soluționarea problemei neliniare de transport cu 4 indici, prin reducerea acesteia la o problemă liniară, în timp rezonabil pentru probleme de până la 1000 de necunoscute.
2. Algoritmul E2 permite soluționarea problemei neliniare de transport pe rețea cu 3 indici, prin reducerea acesteia la o problemă liniară, în timp rezonabil pentru probleme de până la 1000 de necunoscute.
3. Soluțiile furnizate de algoritmii E1 și E2 depind de soluția generată la *Pasul 1* al fiecăruia dintre algoritmi, fapt ce presupune că o alegere corectă duce la obținerea unei soluții mai bune pentru care valoarea funcției de cost este mai mică.

4. Rezultatele experimentale demonstrează corectitudinea algoritmilor descriși.
5. Executând mai multe iterații se poate obține o soluție bună, chiar dacă uneori se pot observa blocaje, din care algoritmul iese cu succes utilizând operațiile de selecție, încrucișare și mutație.

Referințe:

1. DIAZ-PARA, O., RUIZ-VANOYE, A., LORANCA, B. B., FLUENTES-PENNA, A., BARRERA-CAMARA, R. A. A survey of transportation problem, In: *HPC J. of AM*, 2014, ID 848129, 17 pages, http://dx.doi.org/10.1155/2014/848129. Disponibil: http://downloads.hindawi.com/journals/jam/2014/848129.pdf.
2. NINH, P.-X. N index transportation problem. In: *Mathematical Journal*, 1979, vol.7, no.1, p. 18-25.
3. HALEY, K.B. The solid transportation problem. In: *Operat. Research*, 1962, no.10, p.448-463.
4. JUNGINGER, W. On reprezentative of multi-index transportation problems. In: *European Journal of Operational Research*, 1993, no.66, p.353-371. DOI:10.1016/0377-2217(93)90223-A
5. PAȘA, T. Multi-index transport problem with non-linear cost functions. In: *Romai J.*, 2018, vol.14, no.2, p.129-137. Disponibil: https://rj.romai.ro/arhiva/2018/2/Pasa.pdf .
6. DJAMEL, A., AMEL, N., HOAI, L. T., AHMED, Z. A modified clasical algorithm ALPT4C for solving a capacitated four-index transportation problem. In: *ACTA Mathematica Vietnamica*, 2012, vol.37, no.3, p.379-390. Disponibil: http://journals.math.ac.vn/acta/_images/stories/pdf1/Vol_37_No_3/Bai6_Dja_Amel_An_Ahmed_Acta_11_51.pdf.
7. ZITOUNI, R., KERAGHEL, A., BENTERKI, D. Elaboration and Implementation of an Algorithm Solving a Capacitated Four-Index Transportation Problem. In: *Applied Mathematical Sciences*, 2007, vol.1, no.53, p.2643-2657. Disponibil: https://www.researchgate.net/publication/267118025_Elaboration_and_implantation_of_an_algorithm_solving_a_capacitated_four-index_transportation_problem.
8. PAȘA, T., UNGUREANU, V. Solving the non-linear 4-index transportation problem. In: *The Fifth Conference of Mathematical Society of the Republic of Moldova*, 2019, p.221-224. Chișinău: Vladimir Andrunachievici Institute of Mathematics and Computer Science. Disponibil: https://ibn.idsi.md/sites/default/files/imag_file/221-224_9.pdf

Date despre autor:
Tatiana PAȘA, lector universitar, Facultatea de Matematică și Informatică, Universitatea de Stat din Moldova.
E-mail: pasa.tatiana@yahoo.com

Prezentat la 28.04.2020