MINIMIZAREA NUMĂRULUI DE INTERSECŢII DE ARCE ÎN GENERAREA AUTOMATĂ A DESENULUI UNUI GRAF

Mat. Rădulescu Constanţa Zoie

Institutul de Cercetări în Informatică

Rezumat Articolul prezintă o metodă euristică de minimizare a numărului de intersecţii de arce în generarea cu ajutorul calculatorului a desenului unui graf. Este descrisă metoda baricentrică şi sînt detaliaţi algoritmul pentru grafurile cu două niveluri şi algoritmul pentru grafurile cu n niveluri.

Cuvinte cheie: graf, teoria grafurilor, modelare cu ajutorul unui graf, generarea automată a grafului, aplicaţii.

Introducere

În diverse domenii de activitate,precum sînt construcţiile, transporturile, finanţele, conducerea proiectelor de mari dimensiuni etc., există preocupări pentru modelarea cu ajutorul grafurilor a unor structuri.

Dezvoltarea vertiginoasă a reţelelor de transport şi comunicaţie a contribuit într-o mare măsură la impulsionarea teoriei grafurilor.

Problemele legate de analiza drumului critic reprezintă o clasă foarte importantă de probleme a căror rezolvare a contribuit la consacrarea metodelor de modelare bazate pe grafuri.

Există o întreagă teorie matematică dedicată rezolvării problemelor de optimitizare, formulate cu ajutorul grafurilor.

Este recunoscut empiric faptul că desenele grafurilor sînt un ajutor vizual pentru a înţelege imaginea generală a structurilor unor sisteme complexe. Este mai uşor a înţelege un graf dintr-o imagine decît dintr-o listă de noduri. Într-o imagine se pot găsi uşor noduri şi grupuri de noduri legate, se pot trasa drumuri în graf etc.

Obţinerea unor desene bune ale grafurilor ridică probleme dificile de matematică,mai precis de teoria grafurilor, programare matematică, teoria aproximării, analiza algoritmilor etc. De exemplu, problemele legate de minimizarea numărului de intersecţii de arce ale grafului din desen conduce la problema planităţii grafului, problemă care este recunoscută ca fiind deosebit de dificilă.

Într-un articol anterior [1], s-a prezentat problema de a genera automat desenul unui graf. S-au definit conceptele de bază (graf organizat pe n niveluri, matrice de interconectare, numărul de intersecţii de arce pentru un graf organizat pe n niveluri, conectivitate, baricentru) s-au stabilit reguli pentru generarea automată a desenului unui graf şi s-a formalizat o procedură în paşi, de a genera automat desenul unui graf. Obiectivul acestui articol este de a descrie algoritmii de minimizare a număruluide intersecţii de arce, pasul doi în procedura de desenare a unui graf.

Minimizarea numărului de intersecţii de arce

Algoritmii care să calculeze numărul minim de intersecţii de arce sînt mari consumatori de timp. De fapt, în practică este suficient să obţinem o imagine cît mai clară în care să fie un număr relativ mic de intersecţii de arce nu neapărat numărul minim. Acest lucru poate fi făcut utilizînd metode euristice.

Există mai multe tipuri de metode pentru rezolvarea minimizării numărului de intersecţii de arce [2], [3], [4], [5], [6].

Deoarece problema intersecţiei de arce este combinatorică prin natura ei soluţiile cer un timp de execuţie mare.

Din această cauză vom folosi un algoritm euristic numit **metoda baricentrică** (BC).

Fie un graf nivelat propriu G = (V,E, n, σ). Pentru orice i$\in$ {1,2,...,n} considerăm S_i mulţimea tuturor ordonatorilor posibile ale lui V_i.

Fie S = S_1x... xS_n. Atunci problema minimizării numărului de intersecţii de arce este formulată ca:

$$\min\left\{\sum_{i=1}^{n-1} K\left(M\left(\sigma_i,\sigma_{i+1}\right)\right) \mid \sigma_i\in S_i \quad i\in\left\{1,2,...n\right\}\right\}[1].$$

unde definirea noţiunilor de graf n nivelat propriu, matricea de interconectare, numărul de intersecţii de arce pentru un graf n nivelat propriu au fost definite în [1]. Aşa cum este formulată problema este combinatorică şi este dificil de a obţine o soluţie optimă cînd dimensiunea grafului este mare.

Algoritm pentru grafurile cu două niveluri

Metoda baricentrică se bazează pe reordonarea liniei

$\sigma_1 = v_1, v_2, ...v_{|v_1|}$

fixînd coloana σ_2 într-o matrice M(σ_1, σ_2). Ideea principală este de a reordona T1 în concordanţa cu linia baricentrelor

B^R_{1k}, k = 1,....... $|v_1|$

Conceptul de baricentru a fost definit în [1].

Acestea sînt ordonate de la cel mai mic la cel mai mare:

$B^R_{1S_1} \le B^R_{1S_2} .... \le B^R_{1S_{|v_1|}}$

Dacă sînt linii în care baricentrele sînt egale, atunci este preferată ordinea iniţială. Notînd cu σ_1' reordonarea lui σ_1 obţinem:

$\sigma_1' = v_{S_1}\ v_{S_2} ......v_{S|v_1|}$

Operaţia ce transformă M (σ_1, σ_2) în matricea ordonată

$M(\sigma_1', \sigma_2)$ este numită "ordonarea baricentrică" a liniilor şi o notăm cu β_R,

$$M\left(\sigma_1', \sigma_2\right) = \beta_R\left(M\left(\sigma_1, \sigma_2\right)\right)$$

Ordonarea baricentrică a coloanelor este definită similar şi o notăm cu β_C. Putem reduce numărul de intersecţii de arce repetînd ordonarea baricentrică pentru linii şi coloane.
Algoritmul constă în două faze. Faza 2 utilizează faza 1 ca un subalgoritm.
În prima fază se repeta ciclic ordonarea baricentrică a liniilor şi coloanelor. Este păstrată ordinea liniilor (sau coloanelor) care au baricentrul egal.
Totuşi, pentru a reduce numărul de intersecţii de arce trebuie să schimbăm şi această ordine şi vom folosi faza 2.
În faza 2 ordinea liniilor (sau coloanelor) este schimbată în fiecare mulţime (operaţie numită reversie a liniilor şi coloanelor şi notată cu $R_R(M)$ sau $R_C(M)$ apoi se execută din nou faza 1.
Fie M_0 o matrice de interconectare într-un graf de nivel doi şi fie M^* o matrice soluţie şi K^* numărul de intersecţii de arce a lui M^*.
Algoritmul este descris în paşi astfel:

Faza 1

pasul 1:

$M^* := M_0$, $K^* := K\left(M_0\right)$

pasul 2:

$M_1 := \beta_R\left(M_0\right)$

pasul 3:

Dacă $K\left(M_1\right) < K^*$ atunci $M^* := M_1$ şi $K^* := K\left(M_1\right)$

pasul 4:

$M_2 := \beta_c\left(M_1\right)$

pasul 5:

Dacă $K\left(M_2\right) < K^*$ atunci $M^* := M_2$ şi $K^* := K\left(M_2\right)$

pasul 6:

Dacă M_0 şi M_2 sînt egale sau dacă numărul de iteraţii din faza 1 atinge un număr dat iniţial, faza 1 se termină şi se merge la pasul 7. I_n caz contrar se merge la pasul 2.

Faza 2.

pasul 7:

$M_3 := R_R\left(M_2\right)$

pasul 8:

Cînd baricentrele coloanelor lui M_3 nu sînt aranjate într-o ordine crescătoare, mergi la pasul 11 cu $M_0 := M_3$; altfel mergi la pasul 9.

pasul 9:

$M_4 := R_c\left(M_3\right)$

pasul 10:

Cînd baricentrele liniilor lui M_4 nu sînt aranjate într-o ordine crescătoare, mergi la pasul 11 cu $M_0' := M_4$; altfel stop.

pasul 11:

Dacă numărul de iteraţii din faza 2 atinge un număr dat iniţial, atunci stop: altfel mergi la pasul 2.

În algoritmul descris, dacă alegem ca operaţie iniţiala pentru faza 1, reordonarea coloanelor β_c, în pasul 2 β_R este înlocuit cu β_c, iar βc din pasul 4 este înlocuit cu β_R.
La fel pentru faza 2, este utilizat R_c în loc R_R [în pasul 7 şi R_R este utilizat în loc de R_c în pasul 9].

Exemplu:

Considerăm un graf pe două nivele G_0 ilustrat în fig.1.a.

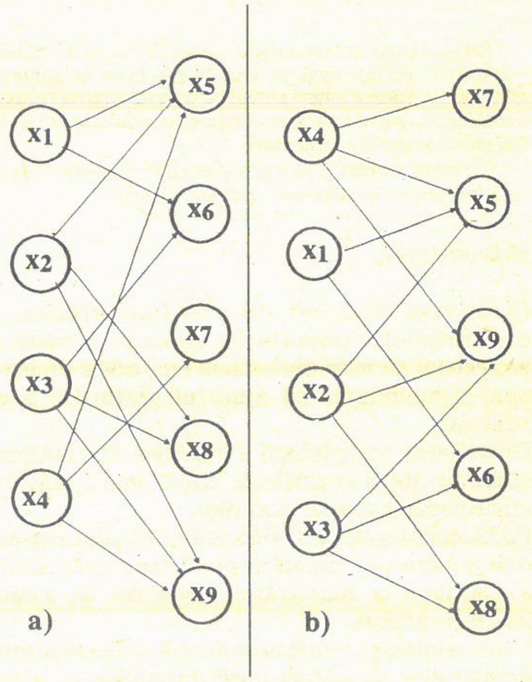

FIG. 1 (a) Graf iniţial
(b) Graf cu număr minim de intersecţii de arce

1) Matricea de interconectare M_0 a lui G_0 este:

		x_5	x_6	x_7	x_8	x_9	B_{ik}^R
	x_1	1	1	0	0	0	1,5
M_0 =	x_2	1	0	0	1	1	3,3
	x_3	0	1	0	1	1	3,7
	x_4	1	0	1	0	1	3,0
	B_{il}^c	2,3	2,0	4,0	2,5	3,0	

$$K\left(M_0\right) = \sum_{j=1}^{3} \sum_{k=j+1}^{4} \left(\sum_{\alpha=1}^{4} \sum_{\beta=\alpha+1}^{5} m_{j\beta}^{(0)} m_{k\alpha}^{(0)} \right) = 14$$

2) Ordonînd liniile x_2, x_3, x_4, se oţine M_1

$$M_1 = \begin{array}{c|ccccc|c} & x_5 & x_6 & x_7 & x_8 & x_9 & B^R_{ik} \\ \hline x_1 & 1 & 1 & 0 & 0 & 0 & 1{,}5 \\ x_4 & 1 & 0 & 1 & 0 & 1 & 3{,}0 \\ x_2 & 1 & 0 & 0 & 1 & 1 & 3{,}3 \\ x_3 & 0 & 1 & 0 & 1 & 1 & 3{,}7 \\ \hline B^c_{il} & 2{,}0 & 2{,}5 & 2{,}0 & 3{,}5 & 3{,}0 & \end{array}$$

$K(M_1) = 11$

3) Reordonînd coloanele x_7, x_6, x_9, x_8, se obține matricea M_2 (sfîrșitul fazei 1).

$$M_2 = \begin{array}{c|ccccc|c} & x_5 & x_7 & x_6 & x_9 & x_8 & \\ \hline x_1 & 1 & 0 & 1 & 0 & 0 & 2{,}0 \\ x_4 & 1 & 1 & 0 & 1 & 0 & 2{,}3 \\ x_2 & 1 & 0 & 0 & 1 & 1 & 3{,}3 \\ x_3 & 0 & 0 & 1 & 1 & 1 & 4{,}0 \\ \hline & 2{,}0 & 2{,}0 & 2{,}5 & 3{,}0 & 3{,}5 & \end{array}$$

$K(M_2) = 9$

4) Reordonînd coloanele x_5, x_7, a lui M_2 se obține M_3 (începutul fazei 2).

$$M_3 = \begin{array}{c|ccccc|c} & x_7 & x_5 & x_6 & x_9 & x_8 & \\ \hline x_1 & 0 & 0 & 1 & 0 & 0 & 2{,}5 \\ x_4 & 1 & 1 & 0 & 1 & 0 & 2{,}3 \\ x_2 & 1 & 0 & 0 & 1 & 1 & 3{,}7 \\ x_3 & 0 & 0 & 1 & 1 & 1 & 4{,}0 \\ \hline & 2{,}0 & 2{,}0 & 2{,}5 & 3{,}0 & 3{,}5 & \end{array}$$

$K(M_3) = 9$

5) Reordonînd liniile x_1 cu x_4 se obține M_4.

$$M_4 = \begin{array}{c|ccccc|c} & x_7 & x_5 & x_6 & x_9 & x_8 & \\ \hline x_4 & 1 & 1 & 0 & 1 & 0 & 2{,}3 \\ x_1 & 0 & 1 & 1 & 0 & 0 & 2{,}5 \\ x_2 & 0 & 1 & 0 & 1 & 1 & 3{,}7 \\ x_3 & 0 & 0 & 1 & 1 & 1 & 4{,}0 \\ \hline & 1{,}0 & 2{,}0 & 3{,}0 & 2{,}7 & 3{,}5 & \end{array}$$

$K(M_4) = 8$

6) Reordonînd coloanele x_6, x_9, se obține M_5 (sfîrșitul fazei 2)

$$M_5 = \begin{array}{c|ccccc|c} & x_7 & x_5 & x_9 & x_6 & x_8 & \\ \hline x_4 & 1 & 1 & 1 & 0 & 0 & 2{,}0 \\ x_1 & 0 & 1 & 0 & 1 & 0 & 3{,}0 \\ x_2 & 0 & 1 & 1 & 0 & 1 & 3{,}3 \\ x_3 & 0 & 0 & 1 & 1 & 1 & 4{,}0 \\ \hline & 1{,}0 & 2{,}0 & 2{,}7 & 3{,}0 & 3{,}5 & \end{array}$$

$K(M_5) = 7$

O imagine a grafului ce corespunde lui M5 este reprezentată în fig. 1.b.

$$\left(P^D_1\left(\sigma^*_1\right)\right) \ldots\ldots \left(P^D_{n-1}\left(\sigma^*_{n-1}\right)\right)$$

$$\text{sau } \left(P^U_{n-1}\left(\sigma^*_n\right)\right) \ldots\ldots \left(P^U_1\left(\sigma^*_2\right)\right)$$

se oprește cînd este îndeplinită una din următoarele condiții:

a) se obțin aceleași matrici de interconectare;

b) se atinge un număr de iterații dat inițial.

Pentru a rezolva subproblemele $\left(P^D_i\left(\sigma^*_i\right)\right)$ sau $\left(P^U_i\left(\sigma^*_{i+1}\right)\right)$ se aplică metoda baricentrelor.

Exemplu: Considerăm un graf nivelat G_0, ilustrat în figura2.a

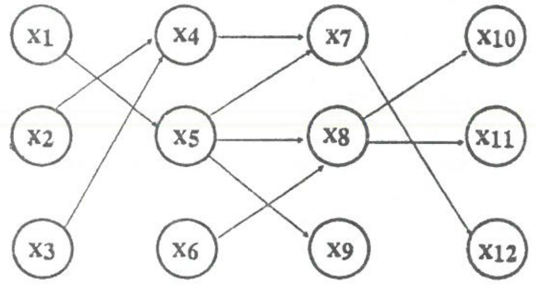

Fig. 2 (a) Graf inițial

Algoritm pentru grafurile cu n niveluri

Considerăm un graf nivelat propriu cu matricele de interconectare $M(\sigma_1, \sigma_2), M(\sigma_2, \sigma_3), \ldots, M(\sigma_{n-1}, \sigma_n)$ unde $n \geq 3$.

Problema (1) devine

$$\min f(z) = z_1 + z_2 + \ldots + z_{n-1}$$

$$\text{unde } z_i = K\left(M\left(\sigma_i, \sigma_{i+1}\right)\right), \quad i = 1, \ldots, n-1$$

$$\sigma_i \in S_i, \quad i = 1, \ldots, n$$

Observăm că funcția f(z) este separabilă (are variabile separate). Aceasta justifică descompunerea problemei în subprobleme pentru grafurile bipartite:

$$\left(P_i\right): \min\left\{k\left(M\left(\sigma_i, \sigma_{i+1}\right)\right) \mid \sigma_i \in S_i,\ \sigma_{i+1} \in S_{i+1}\right\}$$

$i = 1, \ldots, n-1$

Dar nu putem rezolva problema (P_i) separat pentru că este dependentă de problemele (P_{i-1}) și (P_{i+1}).

Introducem alte subprobleme:

$$\left(P^D_i\left(\sigma^*_i\right)\right): \min\left\{k\left(M\left(\sigma^*_i, \sigma_{i+1}\right)\right) \mid \sigma_{i+1} \in S_{i+1}\right\} \text{ sau}$$

$$\left(P^u_i\left(\sigma^*_{i+1}\right)\right): \min\left\{k\left(M\left(\sigma_i T, \sigma^*_{i+1}\right)\right) \mid \sigma_i \in S_i\right\} \text{ unde}$$

σ^*_1 sau σ^*_{i+1} sînt ordine date $i = 1, \ldots, n-1$

Subproblemele $\left(P^D_i\left(\sigma^*_i\right)\right)$ sau $\left(P^U_i\left(\sigma^*_{i+1}\right)\right)$ sînt mai slabe decît subproblemele $\left(P_i\right)$.

Formulăm algoritmul astfel:

pasul 1:

Este dată o ordine inițială σ_1^*. Fie i:=1

pasul 2:

Se rezolvă suproblema $\left(P_i^D\left(\sigma_i^*\right)\right)$ și notăm soluția ei cu σ_{i+1}^*

pasul 3:

Dacă i<n-1, atunci i := i+1 și mergi la pasul 2.
Dacă i = n-1, atunci mergi la pasul 4.

pasul 4:

Se rezolvă subproblema $\left(P_i^U\left(\sigma_{i+1}^*\right)\right)$ și notăm soluția ei cu σ_i^*

pasul 5:

Dacă i>1, atunci i := i-1 și mergi la pasul 4. Dacă i = 1, atunci stop.

Procedura de a rezolva

Matricele de interconectare sînt:

$$M^{(1)}: \begin{array}{c|ccc|} & x_4 & x_5 & x_6 \\ \hline x_1 & 0 & 1 & 0 \\ x_2 & 1 & 0 & 0 \\ x_3 & 1 & 0 & 0 \\ \hline & 2,5 & 1,0 & \end{array} \qquad M^{(2)}: \begin{array}{c|ccc|} & x_7 & x_8 & x_9 \\ \hline x_4 & 1 & 0 & 0 \\ x_5 & 1 & 1 & 1 \\ x_6 & 0 & 1 & 0 \\ \hline \end{array} \qquad M^{(3)}: \begin{array}{c|ccc|} & x_{10} & x_{11} & x_{12} \\ \hline x_7 & 0 & 0 & 1 \\ x_8 & 1 & 1 & 0 \\ x_9 & 0 & 0 & 0 \\ \hline \end{array}$$

$$K\left(M^{(1)}\right)+K\left(M^{(2)}\right)+K\left(M^{(3)}\right)=2+1+1+1=5$$

Reordonînd coloanele x_4, x_5, a lui $M^{(1)}$ și liniile x_4, x_5 ale lui $M^{(2)}$ se obține:

$$\begin{array}{c|ccc|} & x_5 & x_4 & x_6 \\ \hline x_1 & 1 & 0 & 0 \\ x_2 & 0 & 1 & 0 \\ x_3 & 0 & 1 & 0 \\ \hline & 1,0 & 2,5 & \end{array} \qquad \begin{array}{c|ccc|} & x_7 & x_8 & x_9 \\ \hline x_5 & 1 & 1 & 1 \\ x_4 & 1 & 0 & 0 \\ x_6 & 0 & 1 & 0 \\ \hline & 1,5 & 2,0 & 1,0 \end{array} \qquad \begin{array}{c|ccc|} & x_{10} & x_{11} & x_{12} \\ \hline x_7 & 0 & 0 & 1 \\ x_8 & 1 & 1 & 0 \\ x_9 & 0 & 0 & 0 \\ \hline \end{array}$$

$$K\left(M^{(1)}\right)+K\left(M^{(2)}\right)+K\left(M^{(3)}\right)=0+2+1+1+1=5$$

Ordonînd coloanele x_9, x_7, x_8, din $M^{(2)}$ și liniile x_9, x_7, x_8, din $M^{(1)}$ se obține:

$$\begin{array}{c|ccc|} & x_5 & x_4 & x_6 \\ \hline x_1 & 1 & 0 & 0 \\ x_2 & 0 & 1 & 0 \\ x_3 & 0 & 1 & 0 \\ \hline & 1,0 & 2,5 & \end{array} \qquad \begin{array}{c|ccc|} & x_9 & x_7 & x_8 \\ \hline x_5 & 1 & 1 & 1 \\ x_4 & 0 & 1 & 0 \\ x_6 & 0 & 0 & 1 \\ \hline & 1,0 & 1,5 & 2,0 \end{array} \qquad \begin{array}{c|ccc|} & x_{10} & x_{11} & x_{12} \\ \hline x_9 & 0 & 0 & 0 \\ x_7 & 0 & 0 & 1 \\ x_8 & 1 & 1 & 0 \\ \hline & 3,0 & 3,0 & 2,0 \end{array}$$

$$K\left(M^{(1)}\right)+K\left(M^{(2)}\right)+K\left(M^{(3)}\right)=1+1+1=3$$

Ordonînd coloanele x_{10}, x_{11}, x_{12}, se obține:

$$\begin{array}{c|ccc|} & x_5 & x_4 & x_6 \\ \hline x_1 & 1 & 0 & 0 \\ x_2 & 0 & 1 & 0 \\ x_3 & 0 & 1 & 0 \\ \hline \end{array} \qquad \begin{array}{c|ccc|} & x_9 & x_7 & x_8 \\ \hline x_5 & 1 & 1 & 1 \\ x_4 & 0 & 1 & 0 \\ x_6 & 0 & 0 & 1 \\ \hline \end{array} \qquad \begin{array}{c|ccc|} & x_{12} & x_{10} & x_{11} \\ \hline x_9 & 0 & 0 & 0 \\ x_7 & 1 & 0 & 0 \\ x_8 & 0 & 1 & 1 \\ \hline \end{array}$$

$$K\left(M^{(1)}\right)+K\left(M^{(2)}\right)+K\left(M^{(3)}\right)=1$$

Ordonînd x_{12}, x_{10}, se obțin aceleași matrici $M^{(1)}$, $M^{(2)}$, $M^{(3)}$

$$\begin{array}{c|ccc|} & x_5 & x_4 & x_6 \\ \hline x_1 & 1 & 0 & 0 \\ x_2 & 0 & 1 & 0 \\ x_3 & 0 & 1 & 0 \\ \hline & 1,0 & 2,5 & \end{array} \qquad \begin{array}{c|ccc|} & x_9 & x_7 & x_8 \\ \hline x_5 & 1 & 1 & 1 \\ x_4 & 0 & 1 & 0 \\ x_6 & 0 & 0 & 1 \\ \hline & 1,0 & 1,5 & 2,0 \end{array} \qquad \begin{array}{c|ccc|c} & x_{12} & x_{11} & x_{10} & \\ \hline x_9 & 0 & 0 & 0 & \\ x_7 & 1 & 0 & 0 & 1,0 \\ x_8 & 0 & 1 & 1 & 2,5 \\ \hline & 2,0 & 3,0 & 3,0 & \end{array}$$

Ordonînd liniile x_5, x_4 din $M^{(2)}$ și x_5, x_4, din $M^{(1)}$ se obține:

$$\begin{array}{c|ccc|c} & x_4 & x_5 & x_6 & \\ \hline x_1 & 0 & 1 & 0 & 2,0 \\ x_2 & 1 & 0 & 0 & 1,0 \\ x_3 & 1 & 0 & 0 & 1,0 \\ \hline \end{array} \qquad \begin{array}{c|ccc|c} & x_9 & x_7 & x_8 & \\ \hline x_4 & 0 & 1 & 0 & 2,0 \\ x_5 & 1 & 1 & 1 & 2,0 \\ x_6 & 0 & 0 & 1 & 3,0 \\ \hline \end{array} \qquad \begin{array}{c|ccc|c} & x_{12} & x_{11} & x_{10} & \\ \hline x_9 & 0 & 0 & 0 & \\ x_7 & 1 & 0 & 0 & 1,0 \\ x_8 & 0 & 1 & 1 & 2,5 \\ \hline \end{array}$$

$$K\left(M^{(1)}\right)+K\left(M^{(2)}\right)+K\left(M^{(3)}\right)=3$$

Ordonînd x_1, x_2, x_3, din $M^{(1)}$ se obține:

$$\begin{array}{c|ccc|c} & x_4 & x_5 & x_6 & \\ \hline x_2 & 1 & 0 & 0 & 1,0 \\ x_3 & 1 & 0 & 0 & 2,0 \\ x_1 & 0 & 1 & 0 & 2,0 \\ \hline & 1,0 & 3,0 & & \end{array} \qquad \begin{array}{c|ccc|c} & x_9 & x_7 & x_8 & \\ \hline x_4 & 0 & 1 & 0 & 2,0 \\ x_5 & 1 & 1 & 1 & 2,0 \\ x_6 & 0 & 0 & 1 & 3,0 \\ \hline & 2,0 & 1,5 & 2,5 & \end{array} \qquad \begin{array}{c|ccc|} & x_{12} & x_{11} & x_{10} \\ \hline x_9 & 1 & 1 & 1 \\ x_7 & 1 & 0 & 0 \\ x_8 & 0 & 1 & 1 \\ \hline \end{array}$$

$$K\left(M^{(1)}\right)+K\left(M^{(2)}\right)+K\left(M^{(3)}\right)=1$$

Ordonînd x_9, x_7 din $M^{(2)}$ și x_9, x_7 din $M^{(3)}$ se obține

$$\begin{array}{c|ccc|c} & x_4 & x_5 & x_6 & \\ \hline x_2 & 1 & 0 & 0 & 1,0 \\ x_3 & 1 & 0 & 0 & 1,0 \\ x_1 & 0 & 1 & 0 & 2,0 \\ \hline & 1,5 & 3,0 & & \end{array} \qquad \begin{array}{c|ccc|c} & x_7 & x_9 & x_8 & \\ \hline x_4 & 1 & 0 & 0 & 1,0 \\ x_5 & 1 & 1 & 1 & 2,0 \\ x_6 & 0 & 0 & 1 & 3,0 \\ \hline & 1,5 & 2,0 & 2,5 & \end{array} \qquad \begin{array}{c|ccc|} & x_{12} & x_{11} & x_{10} \\ \hline x_7 & 1 & 0 & 0 \\ x_9 & 0 & 0 & 0 \\ x_8 & 0 & 1 & 1 \\ \hline & 1,0 & 3,0 & 3,0 \end{array}$$

$$K\left(M^{(1)}\right)+K\left(M^{(2)}\right)+K\left(M^{(3)}\right)=0$$

Graful cu numărul intersecțiilor de arce egal cu zero este ilustrat în figura 2.b.

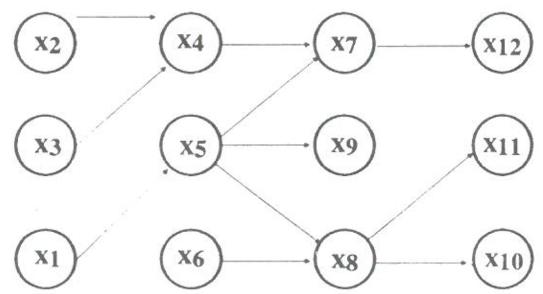

Fig. 2 (b) Graf cu număr minim de intersecții de arce

Concluzii

Algoritmii prezentați sînt relativ ușor de implementat pe sistemele de calcul existente, sînt rapizi și, deși obțin soluții suboptimale, în practică se constată că aceștia sînt suficient de apropiați cei optimi.

Desenele de grafuri obținute pe baza lor sînt ușor inteligibile, cu un număr mic de intersecții de arce.

Implementarea algoritmilor de minimizare a intersecțiilor de arce în produsul DESGRA (vezi [7], [8], [9]) contribuie la creșterrea calității desenelor de grafuri.

Bibliografie

1. **RĂDULESCU, C., POPA, A.** *Generarea automată a desenului unui graf.* În: Buletinul Român de Informatică, nr.1, I.C.I, București, 1990.
2. **SUGIYAMA, K., TAGAWA, S., TODA, M.** *Methods for Visual Understanding of Hierarchical System Structures.* În: IEEE Transations on Systems, Man and Cybernetics, vol.11, nr.2 februarie 1981, pp.109-125.
3. **WARFIELD,J.** *Crossing Theory and Hierarchy Mapping.* In: IEEE Transactions on Systems, Man and Cybernetics, vol.7, nr.7, iulie 1977, pp.505-523.
4. **CARPANO, M.J.** *Automatic Display of Hierarchiezed Graphs for Computer - aided Decision Analysis.* In: IEEE Transactions on Systems, Man and Cybernetics, vol.10, nr.11, noiembrie 1980, pp.705-715.
5. **GANSNER, E.R., NORTH, S.G., Vo, K.P.***Dag-a Program that Draws Directed Graphs.* In: Software Practice and Experience, vol. 18, nr.41, noiembrie 1988, pp. 1047-1062;
6. **ROWE,E., ş.a.** *A Browser for Directed Graphs.* In: Software Practice and Experience, vol.17, nr.1, ianuarie 1987, pp.61-76;
7. *** * * DESGRA.** *Produs - program de proiectare grafică asistată de calculator a rețelelor.* Manual de realizare. Septembrie 1988.
8. *** * * DESGRA.** *Produs-program de proiectare grafică asistată de calculator a rețelelor.* Manual de prezentare. Decembrie 1988.
9. *** * * DESGRA.** *Produs-program de proiectare grafică asistată de calculator a rețelelor.* Manual de utilizare. Decembrie 1988.

INSTITUTUL DE CERCETARE ÎN INFORMATIC

B-dul Mareșal Averescu 8 - 10, Sect. 1, cod 71316, Tel.: 17.79.78 - director, 65.45.65 - comercial, Telex: 11891 icp
Fax: 12.85.39, București-ROMÂNIA

LA DISPOZIȚIA DVS. PENTRU LUCRĂRI ÎN:

- inteligența artificială
- sisteme expert
- rețele locale
- rețele generale
- prelucrări distribuite
- baze și bănci de date
- birotică
- MIS
- sisteme suport de decizie
- sisteme în timp real
- conducerea proceselor tehnologice
- CAD/CAM/CAQ
- CIM

ORICÎND GATA SĂ PROIECTEZE PE BAZA SPECIFICAȚIILOR DVS.:

- programe
- sisteme informatice
- sisteme la cheie

ASIGURĂ:

- asistență tehnică
- școlarizare
- douăsprezece luni garanție

INTELIGENȚĂ
COMPETENȚĂ
INVENTIVITATE

PARTENERUL DVS. PE TERMEN LUNG ÎN INFORMATICĂ

LABORATORUL

BIROTICĂ ȘI SOFTWARE DE BAZĂ PE PC

Contactîndu-ne aflați lucruri utile despre:

- editare cu calculatorul
  - produsul PreText: procesor de text compatibil WordStar versiuni CP/m (limba romă și rusă) și MS-DOS (limba română)
  - drivere pentru utilizarea alfabetului românesc pe diverse imprimante
  - editor de fonte matriciale
  - convertoare de texte
  - corectarea ortografiei
  - colecție de fonte tipografice pentru VENTURA PUBLISHER
  - emulator de mașină de scris pe PC
- utilizare scanner pentru preluări imagini
- prelucrări imagini cu PICTURE PUBLISHER
- recunoaștere spectograme
- recunoaștere optică a caracterelor cu Omni Page
- produsul PreTAB (procesor de tabele în limba română, compatibil Lotru 1-2-3) și produ original DeFalc (procesor de tabele multidimensionale).

Numai cu noi puteți rezolva eficient problemele:

- protecției la copiere a programelor
  - soluții software și soluții hardware
- protecției antivirus:
  - produse originale, experiență îndelungată în lupta antivirus
- conversiei de date sau de suporți între diverse tipuri de calculatoare sau sisteme de operă medii de programare:
  - scriere/citire dischetă RSX (5 1/4") pe PC
  - conversie fișiere FORTRAN, COBOL, DTR din RSX în dBASE (CP/M și MS-DOS)
- dezvoltării de sisteme multitasking pentru comunicație pe COM, LPT, SUPER 8 (utiliz. 1-20 terminale pe PC) în aplicații tip tranzacțional sau culegere date (produsul MTK)
- biblioteci secvențial-indexate sub MS-DOS.

Printre beneficiarii noștri se numără:

- Ministerul Afacerilor Externe, Ministerul Culturii, Președenția României, Trustul de pre "Azi", ziarul "Acum", ziarul "Independentul", Agrinf SA (fostul Centru de Calcul al Ministeru Agriculturii), IPILF Fetești, ECOSOC SRL, FINRO SA, Comisia Națională de Informati RomSoft SA, CNDPI, etc.

Pentru informații suplimentare vă rugăm să vă adresați la sediul nostru sau la telefon 65.60 int.178 (șef laborator ing.Gheorghe Răuț).