

Utilizarea algoritmului Dijkstra pentru determinarea rutelor de valoare minimă în rețele de comunicații

Ana Nistiriuc, Victor Ababii, Andrei Chihai, Dinu Țurcanu, Nicolae Bejan, Anatol Alexei, Ion Nistiriuc, Natalia Sharma Popovici, Pavel V.Nistiriuc
Universitatea Tehnica a Moldovei
andrei.chihai@fimet.utm.md

Abstract: In this paper is analyzed the use of the Dijkstra algorithm for determining the minimum length routes in the communication networks.

Termeni chei – Rețele de comunicații, algoritmul Dijkstra, rute de valoare minimă.

I. INTRODUCERE

Ideea de bază a algoritmului lui Dijkstra este de a eticheta într-o rețea de comunicații, relativ de nodul s , celelalte noduri în ordinea depărtărilor lor de la nodul s [1-3].

Astfel, fiecărui nod j al rețelei de comunicații i se asociază o variabilă $p(j)$, care în orice moment reține cea mai mică lungime a rutelor de la nodul s la nodurile i , $p(s)=0$.

De regulă, se elaborează două liste de coloane respectiv L și T . Lista coloanei L conține nodurile permanente, iar lista coloanei T conține nodurile temporare ale rețelei de comunicații.

Inițial, coloanele L și T se compun din acele noduri $j \in X$ pentru care ruta orientată (s,j) este permisă. Nodul i^* din coloana T a cărui variabilă $p(i^*)$ are valoarea cea mai mică se include în valorile coloanei L și nodul i^* devine permanent. În valorile coloanei T se includ apoi toate nodurile j ale rețelei de comunicații care nu sunt înscrise în această coloană și ruta (i,j) este PERMISĂ.

Pentru a reconstitui rutele de lungime minimă de la nodul s toate nodurile rețelei de comunicații ce pot fi atinse din nodul s , utilizând rute orientate PERMISE ale grafului rețelei de comunicații, se introduce pentru fiecare nod $j \neq s$ din graf un indicator $PRED(j)$. Prezentul indicator va reține în orice moment ultimul nod dinaintea lui j , pe ruta de la nodul s la nodul j care a fost depistată în acel moment posedă cea mai mică lungime $PRED(s)=0$.

Pentru a clarifica modalitatea de aplicare a algoritmului Dijkstra se elaborează un tabel cu coloanele următoare: iterația, lista coloanei L , nodul j , variabila $p(j)$ și indicatorul $PRED(j)$. Faptul, că un nod j este inclus în lista nodurilor permanente L , se marchează prin "*" în stânga tabelului (coloana L), în dreptul nodului j . În orice moment nodurile din tabel fără marcajul "*" aparțin listei nodurilor temporare T .

În stânga tabelului se precizează iterația în care nodul respective este permanentizat în lista L , prin marcarea "*".

Inițial, tabelul conține nodul s marcat cu "*" și acele noduri j pentru care ruta orientată (s,j) este permisă.

Corespunzător $p(s)=0$, $p(j)=V_{sj}$ și $PRED(j)=s$. Se considera

marcarea nodului s ca fiind iterația 0(zero).

Iterația este constituită din următorii trei pași:

●Pasul 1. Dacă în tabel, lista T este vidă, atunci toate nodurile rețelei de comunicații sunt permanente (marcate cu "*"). Graful rețelei de comunicații este analizat integral și algoritmul se oprește;

●Pasul 2. Se selectează $i^* \in T$ cu proprietatea:

$$p(i^*) = \min_{j \in T} p(j). \quad (1)$$

Dacă $i^* \equiv t$, algoritmul se oprește și se determină în rețeaua de comunicații ruta de lungime minimă de la s la t .

În caz contrar, nodul i^* este transferat în lista L , prin marcarea cu "*" și astfel, este eliminat din lista coloanei T . În lista coloanei T se înscriu toate nodurile $j \in T$ cu proprietatea, că ruta (i^*,j) este permisă. Se ia $p(j)=\infty$ și $PRED(j)=0$;

●Pasul 3. Se corectează valorile variabilelor corespunzătoare nodurilor rețelei de comunicații $j \in T$ adiacente cu nodul i^* comparând $p(j)$ cu $p(j^*)+V_{i^*j}$ și anume:

a) Dacă $p(j) > p(i^*) + V_{i^*j}$, atunci $p(j) = p(i^*) + V_{i^*j}$ și $PRED(j) = i^*$;

b) Dacă $p(j) = p(i^*) + V_{i^*j}$, atunci $i \in PRED(j)$;

c) Dacă $p(j) < p(i^*) + V_{i^*j}$ se trece la examinarea unui altui nod din lista coloanei T .

După epuizarea listei coloanei T se reia pasul 1 în cadrul următoarei iterații. După oprirea algoritmului și atingerea nodului t (în pasul 2) ruta de lungime minimă se reconstituie de la nodul t către nodul s cu ajutorul indicatorului $PRED(j)$. Dacă nodul t nu este evidențiat, după oprirea algoritmului se reconstituie, în același fel, graful rutelor de lungime minimă de la nodul s la orice alt nod în graful rețelei de comunicații.

II. PARTEA DE BAZĂ.

Obiectivul prezentei lucrări servește determinarea rutelor de valoare minimă de la nodul 3 la toate celelalte noduri în graful rețelei de comunicații din fig.1.

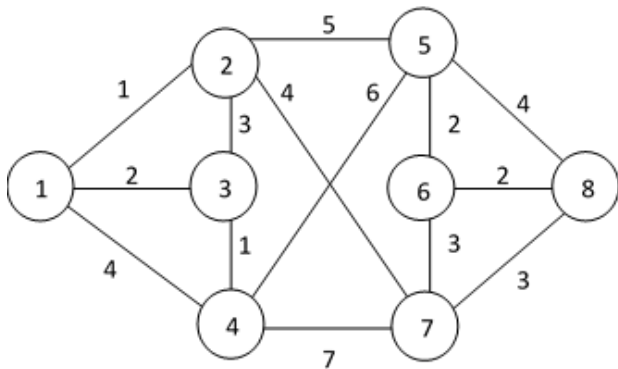


Fig.1. Graficul rețelei de comunicații

Valorile notate pe fiecare rută a rețelei de comunicații din fig.1, reprezintă lungimea tronsonului respectiv, în unități relative.

Pentru soluționarea obiectivului nominalizat vom utiliza algoritmul lui Dijkstra. Algoritmul poate fi urmărit prin parcurgerea iterațiilor următoare, rezultatele fiind sistematizate în tabelul 1-7. Inițial se întocmește tabelul 1, în care nodul 3 ca nod de start devine permanent în lista coloanei L: $p(3)=0$, $PRED(3)=0$. Nodul 3 se marchează cu (*).

TABELUL 1. RUTE ÎNȚIALIZATE

Iterația	L	j	P(j)	PRED(j)
0	*	3	0	-
		1	2	3
		2	3	3
		4	1	3

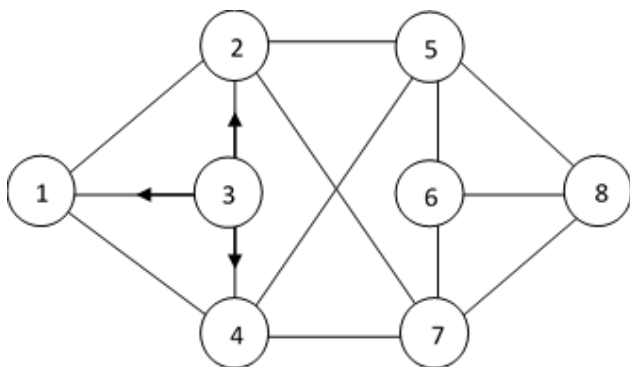


Fig.2. Rute inițializate în graficul rețelei de comunicații

Lista coloanei nodurilor temporare este $T=\{1,2,4\}$, $p(1)=2$, $p(2)=3$, $p(4)=1$ și toate nodurile 1,2 și 4 au ca nod precedent nodul 3, $PRED(1)=\{3\}$, $PRED(2)=\{3\}$, $PRED(4)=\{3\}$. Graful rețelei de comunicații cu primele rute inițializate este reprezentat în fig.2.

Analizăm iterațiile următoare:

Iterația 1.

Pasul 1. $T=\{1,2,3\} \neq \emptyset$.

Pasul 2. Se selectează, din lista coloanei T, nodul i^* cu valoarea $p(i^*)$ minimă:

$$\min_{j \in T} p(j) = 1 = p(4) \Rightarrow i^* = 4.$$

Nodul 4 se marchează cu (*), se include în lista nodurilor permanente L și se elimină din lista nodurilor temporare T. În lista T se înscriu nodurile 5 și 7, pentru că rutele (4,5) și (4,7) sunt permise. Se ia $p(5)=p(7)=\infty$ și $PRED(5)=PRED(7)=0$.

$T=\{1,2,5,7\}$.

Pasul 3. Se corectează valorile variabilelor $p(j)$ asociate nodurilor j din lista T adiacente cu nodul 4. În acest sens, se efectuează următoarele calcule și comparații.

$$p(1) = 2 < p(4) + v_{41} = 1 + 4 = 5 \text{ nu se corectează.}$$

$$p(5) = \infty > p(4) + v_{45} = 1 + 6 = 7 \Rightarrow p(5) = 7, PRED(5) = \{4\}.$$

$$p(7) = \infty > p(4) + v_{47} = 1 + 7 = 8 \Rightarrow p(7) = 8, PRED(7) = \{4\}.$$

În tabelul 2 și fig.3. sunt reflectate rezultatele algoritmului conform primei iterații.

TABELUL 2. RUTE CONFORM PRIMEI ITERAȚII.

Iterația	L	j	p(j)	PRED(j)
0	*	3	0	-
		1	2	3
		2	3	3
1	*	4	1	3
		5	∞ , 8	-4
		7	∞ , 6	-4

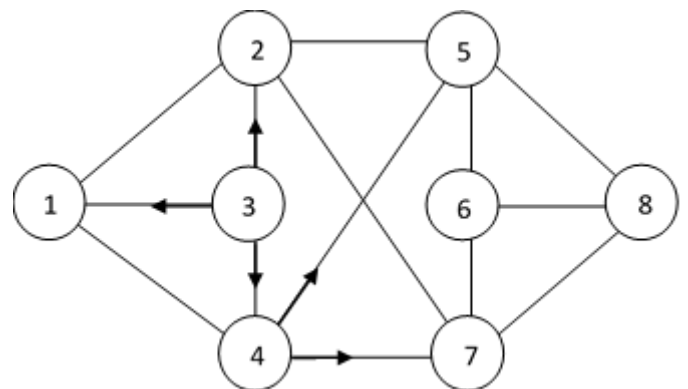


Fig.3. Rute conform primei iterații graficul rețelei de comunicații

Iterația 2.

Pasul 1. $T=\{1,2,5,7\} \neq \emptyset$.

Pasul 2. $\min p(j) = 2 = p(1) \Rightarrow i^* = 1$.

$$j \in T$$

Nodul 1 este inclus în lista L și eliminat în lista T. Noduri adiacente cu nodul 1, neincluse în lista T, nu există, deci, $T=\{2,5,7\}$.

Pasul 3. $p(2)=3=p(1)-V_{12}=2+1=3 \Rightarrow 1 \in \text{PRED}(2)$
 Rezultatele iterației 2 sunt reflectate în tabelul 3 și fig.4.

TABELUL 3. RUTE CONFORM ITERAȚIEI 2.

Iterația	L	j	p(j)	PRED(j)
0	*	3	0	-
2	*	1	2	3
		2	3	3;1
1	*	4	1	3
		5	$\infty, 7$	-,4
		7	$\infty, 8$	-,4

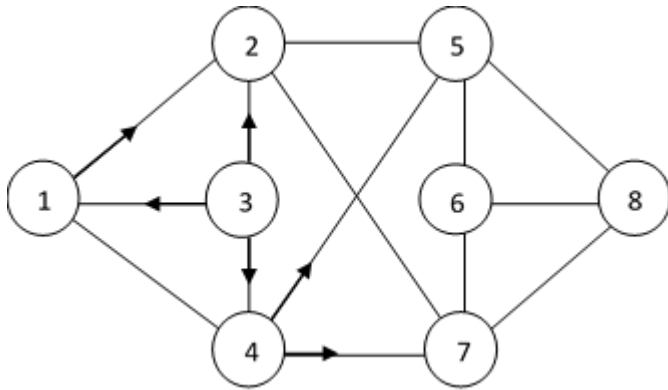


Fig.4. Rute conform iterației 2 în graful rețelei de comunicații
 Conform rezultatelor obținute observăm, că $\text{PRED}(2)=\{3,1\}$.

Iterația 3

Pasul 1. $T=\{2,5,7\} \neq \emptyset$.
 Pasul 2. $\min_{j \in T} p(j)=3=p(2) \Rightarrow i^*=2$.

Nodul 2 este marcat cu (*) în lista L. În lista T nu se introduce un alt nod adiacent, pentru că, acestea, sunt deja în listă, $T=\{5,7\}$.

Pasul 3. $p(5)=7 < p(2)+V_{25}=3+5=8$
 $p(7)=8 > p(2)+V_{27}=3+4=7 \Rightarrow p(7)=7, \text{PRED}(7)=\{2\}$.

După a treia iterație rezultatele sunt sistematizate în tabelul 4 și reprezentate în fig.5.

TABELUL 4. RUTE CONFORM ITERAȚIEI 3

Iterația	L	j	p(j)	PRED(j)
0	*	3	0	-
2	*	1	2	3
3	*	2	3	3;1
1	*	4	1	3
		5	$\infty, 7$	-,4
		7	$\emptyset, 8$,7	$\setminus, 4, 2$

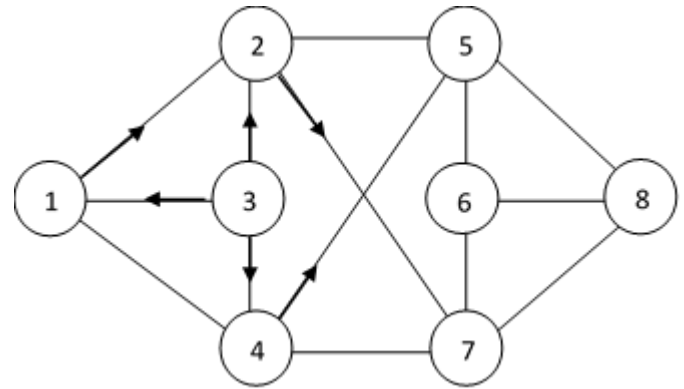


Fig.5. Rute conform iterației 3 în graful rețelei de comunicații

Iterația 4

Pasul 1. $T=\{5,7\} \neq \emptyset$.
 Pasul 2. $\min_{j \in T} p(j)=7=p(5)=p(7) \Rightarrow i^*=5$ și $i^*=7$

Astfel, se permanentizează și nodul 5 și nodul 7, marcându-le pe ambele cu (*) în lista L. În lista T se introduc nodurile 6 și 8 fiind adiacente cu nodurile 5 și 7. Se ia $p(6)=p(8)=\infty, \text{PRED}(6)=\text{PRED}(8)=\emptyset, T=\{6,8\}$.

Pasul 3. $p(6)=\infty > p(5)+V_{56}=7+2=9 \Rightarrow p(6)=9, \text{PRED}(6)=\{5\}$.

$p(6)=9 < p(7)+V_{76}=7+3=10$ - nu se corectează.

$p(8)=\infty > p(5)+V_{58}=7+4=11 \Rightarrow p(8)=11, \text{PRED}(8)=\{5\}$.

$p(8)=11 > p(7)+V_{78}=7+3=10 \Rightarrow p(8)=10, \text{PRED}(8)=\{7\}$.

Rezultatele iterației 4 sunt reflectate în tabelul 5 și fig.6.

TABELUL 4. RUTE CONFORM ITERAȚIEI 4

Iterația	L	j	p(j)	PRED(j)
0	*	3	0	-
2	*	1	2	3
3	*	2	3	3;1
1	*	4	1	3
4	*	5	$\infty, 7$	-,4
4	*	7	$\infty, 8, 7$	$\setminus, 4, 2$
		6	$\infty, 9$	-,5
		8	$\emptyset, 11,$ 10	$\setminus, 5, 7$

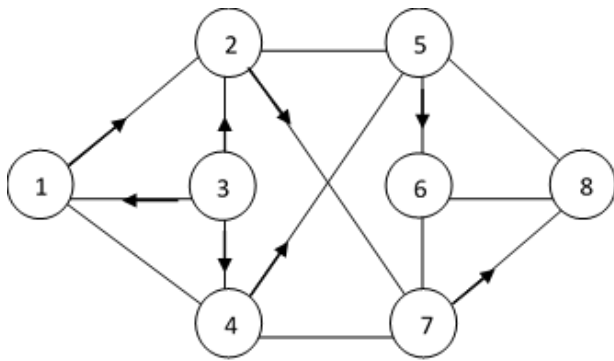


Fig.6. Rute conform iterației 4 în graful rețelei de comunicații

Iterația 5.

Pasul 1. $T = \{6, 8\} \neq \emptyset$.

Pasul 2. $\min_{j \in T} p(j) = 9 = p(6) \Rightarrow i^* = 6$

se permanentizează nodul 6, $T = \{8\}$.

Pasul 3. $p(8) = 10 < p(6) + v_{68} = 9 + 2 = 11$, nu se corectează.

Rutele nu se modifică în graful rețelei de comunicații și rezultatele conform iterației 5 sunt reflectate în tabelul 6.

TABELUL 6. RUTE CONFORM ITERAȚIEI 5.

Iterația	L	j	p(j)	PRE D(j)
0	*	3	0	-
2	*	1	2	3
3	*	2	3	3;1
1	*	4	1	3
4	*	5	$\infty, 7$	-4
4	*	7	$\infty, 8, 7$	-4,2
5	*	6	$\infty, 9$	-5
		8	$\infty, 11, 10$	-5,7

Iterația 6.

Pasul 1. $T = \{8\} \neq \emptyset$.

Pasul 2. $\min_{j \in T} p(j) = 10 = p(8) \Rightarrow i^* = 8$.

Nodul 8 se permanentizează și se elimină din lista nodurilor temporare T. $T = \emptyset$.

Rezultatele conform iterației 6 sunt reflectate în tabelul 7.

TABELUL 7. RUTE CONFORM ITERAȚIEI 6.

Iterația	L	j	p(j)	PRED(j)
0	*	3	0	-
2	*	1	2	3
3	*	2	3	3;1
1	*	4	1	3
4	*	5	$\infty, 7$	-4
4	*	7	$\infty, 8, 7$	-4,2
5	*	6	$\infty, 9$	-5
6	*	8	$\infty, 11, 10$	-5,7

Iterația 7.

Pasul 1. $T = \emptyset$.

III. CONCLUZII

Algoritmul lui Dijkstra reprezintă o metodă eficientă privind determinarea rutelor de valoare minimă în rețelele de comunicație. Rutele de valoare minimă determinate conform algoritmului lui Dijkstra în graful rețelei de comunicații analizată (fig.1) sunt reprezentate în fig.7.

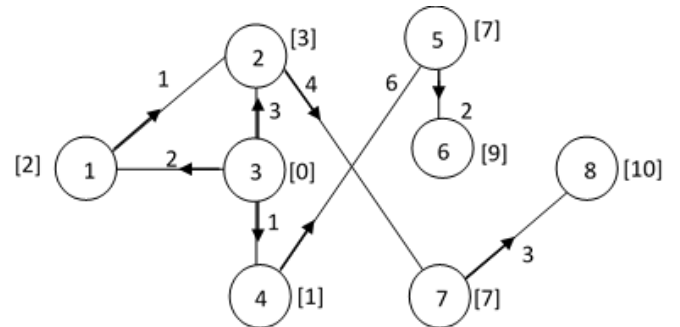


Fig.7. Rute de valoare minimă în graful rețelei de comunicații

Valorile minime sunt notate în paranteze pătrate la fiecare nod al rețelei de comunicații. Astfel, observăm, că de la nodul 3 la nodul 8 există două taseuri de lungime minimă egală cu 10 unități convenționale, după cum sunt $\mu_1 = [3, 1, 2, 7, 8]$ și $\mu_2 = [3, 2, 7, 8]$.

BIBLIOGRAFIE

- [1] <http://voyager8.blogspot.md/2018/01/book-algorithms-notes-for-professionals.html>
- [2] http://lmgfiles.com/iu0a2ppufl8r/Advanced_Analysis_Techniques_ebook3000.pdf.html
- [3] Bhatnagar S. K. Network Analysis Technique. John Wiley & Sons, New York, 2016. -912 pages.