

Հ.Ա. ՃԱՆՃԱՊԱՆՅԱՆ

ԴԻՆԱՄԻԿ ԵՐԹՈՒՂԱՎՈՐՄԱՆ ԽԱՌՆԱԾԻՆ ԱԼԳՈՐԻԹՄԻ ՄՇԱԿՈՒՄ

Մշակվել է դինամիկ երթուղավորման խառնածին ալգորիթմ, որը հիմնված է թռիչքների նվազագույն քանակի հաշվման վրա, ինչը երթուղու որոշման ժամանակ հնարավորություն է տալիս փոքրացնել փնտրման տիրույթը: Օգտագործվել է տոպոլոգիական տվյալների հենք՝ ինչպես Դեյկստրայի ալգորիթմում, բայց որպես երթուղու լավարկման ցուցանիշ ընտրվել է թռիչքների (հոպերի) քանակը: Մշակված ալգորիթմն ունի Դեյկստրայի ալգորիթմի առավելությունները և զերծ է հեռավորության վեկտորով ալգորիթմների թերություններից:

Առանցքային բառեր. երթուղի, դինամիկ երթուղավորում, ալգորիթմ, կողմնորոշված գրաֆ, արձանագրություն, հոպերի քանակ:

Երթուղիների ընտրման ալգորիթմները և տվյալների հենքը, որ օգտագործում են դրանք, ցանցային մակարդակի նախագծման հիմնական նպատակներն են: Երթուղավորման ալգորիթմները բաժանվում են երկու դասի՝ չհարմարվող և հարմարվող [1]: Չհարմարվող ալգորիթմները երթուղի ընտրելիս ժամանակ հաշվի չեն առնում ցանցի տոպոլոգիան և ընթացիկ վիճակը: Երթուղու ընտրությունը յուրաքանչյուր զույգ կայանների համար կատարվում է նախօրոք՝ ավտոմատ ռեժիմում, և ցանցի բեռնավորման ժամանակ երթուղիների ցուցակը մուտքագրվում է երթուղավորիչներ: Այսպիսի գործընթացը հաճախ անվանում են ստատիկ երթուղավորում:

Հարմարվող ալգորիթմները, հակառակ վերը բերվածին, երթուղու ընտրության որոշումը կարող են փոխել ցանցի տոպոլոգիայի փոփոխման ժամանակ, ինչպես նաև՝ ըստ կապուղիների բեռնավորման, այսինքն, դրանց միջոցով իրականացվում է դինամիկ երթուղավորում:

Ներկայումս կարևորվում են դինամիկ երթուղավորման ալգորիթմները, որոնք օգտագործվում են երթուղավորման արձանագրություններում: Տարբերում են դինամիկ երթուղավորման DVP (distance vector protocol) արձանագրություններ, որոնց աշխատանքի հիմքում ընկած են DVA ալգորիթմներ, և LSP (link state protocol) արձանագրություններ, որոնք օգտագործում են LSA ալգորիթմներ:

LSA ալգորիթմներում երթուղավորման աղյուսակների կառուցումը տրոհվում է երկու խոշոր փուլերի: Առաջին փուլում յուրաքանչյուր երթուղավորիչ կառուցում է ցանցի կապերի գրաֆը, որի գագաթները համարվում են երթուղավորիչները և IP-ցանցերը, իսկ կողերը՝ երթուղավորիչների ինտերֆեյսները: Բոլոր երթուղավորիչները փոխանակում են իրենց հարևանների հետ ցանցի գրաֆի մասին տվյալ պահին եղած տեղեկությունը: Այդ հաղորդագրությունները կոչվում են router links advertisement (հայտարարություն երթուղավորիչների կապերի մասին): Ցանցի

գրաֆի մասին տեղեկությունը պահվում է երթուղավորիչի տոպոլոգիական տվյալների հենքում:

Երկրորդ փուլում ստացված գրաֆի միջոցով գտնվում է լավարկային երթուղին: Լավարկային ճանապարհի որոնման խնդիրը գրաֆի միջոցով բավականին բարդ և աշխատատար է: OSPF (Open Shortest Path First) արձանագրության մեջ օգտագործվում է Դեյկստրայի ալգորիթմը, որը համարվում է կապերի վիճակի ալգորիթմի արդիական իրականացում [2]: Ալգորիթմը հիմնված է տվյալների հենքի վրա, որտեղ պահվում են հարևաններից ամենաուշ ստացած LSPs (link state paths)-երը, (աղ. 1):

Աղյուսակ 1

Կապերի վիճակների տվյալների բազան

A	B	C	D	E	F	G
B/6	A/6	B/2	A/2	B/1	C/2	C/5
D/2	C/2	F/2	E/2	D/2	E/4	F/1
	E/1	G/5		F/4	G/1	

[3]-ում ևս մշակված է LSA տիպի ալգորիթմ, որը որոշում է տված որակին բավարարող երթուղին:

Աշխատանքում թռիչքների և ճանապարհների հաշվման ալգորիթմը ևս հենվում է տոպոլոգիական տվյալների կառուցվածքի վրա (աղ. 2), սակայն այն տարբերությամբ, որ այստեղ չեն ներկայացվում կապի վիճակի արժեքները:

Որպես մաթեմատիկական ապարատ ընտրված է գրաֆների տեսությունը, որից, մասնավորապես, օգտագործված են կցության մատրիցի միջոցով ճանապարհի առկայության և երկարության որոշման մեխանիզմները [4]:

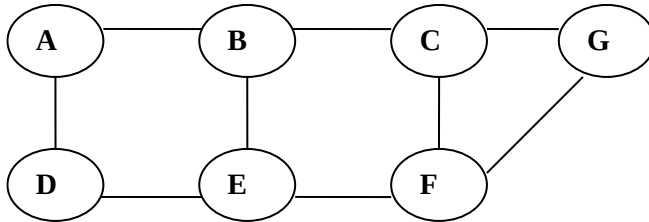
Աղյուսակ 2

Տոպոլոգիական տվյալների բազան

A	B	C	D	E	F	G
B	A	B	A	B	C	C
D	C	F	E	D	E	F
	E	G		F	G	

Մշակվող ալգորիթմը խառնածին է, այն օգտագործում է տոպոլոգիական տվյալների հենք, որը բնորոշ է LSP տիպի արձանագրություններում օգտագործվող ալգորիթմներին, սակայն որպես լավարկման ցուցանիշ (մետրիկ) վերցվում է թռիչքների քանակը, որն օգտագործվում է DVA ալգորիթմներում:

Աղ. 2-ի տոպոլոգիական տեղեկատվության հիման վրա ցանցը ներկայացվում է $G(V,E)$ սիմետրիկ կողմնորոշված գրաֆի տեսքով (նկ. 1),



Նկ. 1. Ցանցի գրաֆը

որտեղ V -ն գագաթների բազմությունն է, իսկ E -ն կողերի ենթաբազմությունը [4]: E ենթաբազմությունը կողերի բազմության համաչափ փակումն է:

Գրաֆի կցության մատրիցը ցույց է տրված նկ. 2-ում:

	A	B	C	D	E	F	G
A	0	6	0	2	0	0	0
B	6	0	2	0	1	0	0
C	0	2	0	0	0	2	5
D	2	0	0	0	2	0	0
E	0	1	0	2	0	4	0
F	0	0	2	0	4	0	1
G	0	0	5	0	0	1	0

ա)

բ)

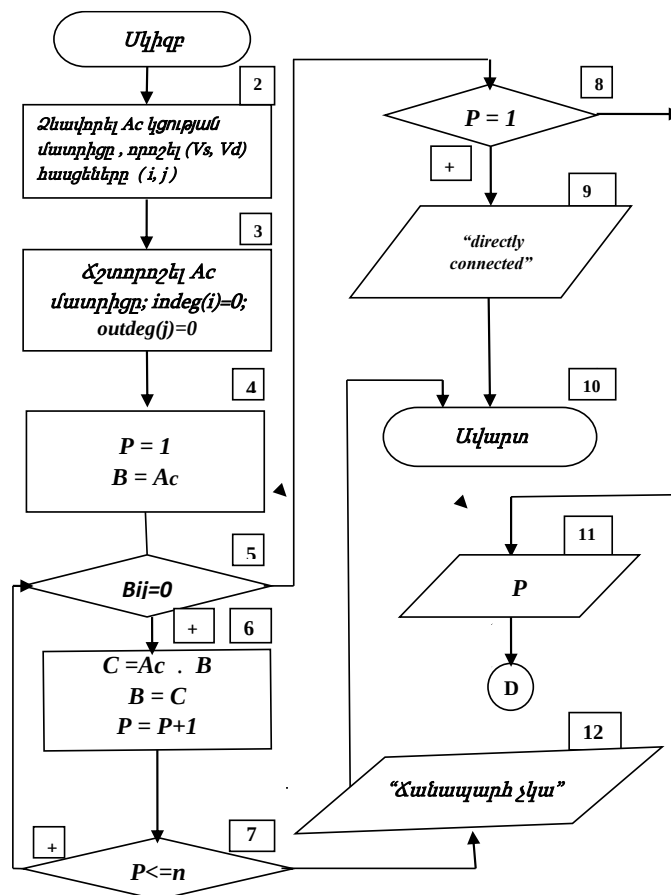
Նկ. 2. Գրաֆի կցության մատրիցը՝

ա) Դեյկստրայի ալգորիթմում, բ) մշակվող ալգորիթմում

Նկ. 3-ում բերված է թռիչքների նվազագույն քանակի որոշման ալգորիթմի բլոկ-սխեման: Համառոտ նկարագրենք դրա աշխատանքը:

Կցության մատրիցը (A_c) ձևավորելուց հետո որոշվում են նպատակային (destination) և ուղարկող կամ աղբյուր (source) հանգույցների (V_d, V_s) հասցեները՝ (i, j), այնուհետև ճշտորոշվում է մատրիցը, այսինքն՝ դրվում են $\text{indeg}(V_s)=0$ և $\text{outdeg}(V_d)=0$, որը նշանակում է՝ անտեսվում են աղբյուրին եկող և նպատակային հանգույցից դուրս եկող կապերը (օպերատորներ 2, 3), և կատարվում է անցում օպերատոր 4-ին: Վերջինս տեղադրում է թռիչքների քանակի սկզբնական արժեքը P -ում, կրկնօրինակում է կցության A_c մատրիցը B -ում, և ղեկավարումը փոխանցվում է օպերատոր 5-ին, որը ստուգում է P քանակով թռիչքի կամ թռիչքների առկա-

յությունը: Եթե օպերատոր 5-ով ստուգվող պայմանը բավարարվում է, ապա 5,6,7,5 ցիկլ կազմող օպերատորների միջոցով կցության մատրիցը այնքան անգամ է բազմապատկվում ինքն իրենով, մինչև որ մատրիցի (i,j) տարրը 0-ից տարբեր է ստացվում, որից հետո ստուգվում է հեռավորության վեկտորի մեկից տարբեր լինելը (օպերատոր 8), այսինքն՝ նպատակային և ուղարկող կամ աղբյուր հանգույցները ուղիղ են միացված, թե ոչ: Օպերատոր 8-ի պայմանի բավարարման դեպքում դուրս է բերվում հաղորդագրություն “directly connected”, և ավարտվում է ալգորիթմի աշխատանքը (օպերատորներ 8,9,10): Հակառակ դեպքում՝ դուրս է բերվում P -ի արժեքը, որը և փոխանցվում է հաջորդ ալգորիթմին (օպերատորներ 8,12): Եթե 5,6,7,5 օպերատորների ցիկլով բազմապատկումների քանակը, որով որոշվում է թռիչքների քանակը, գերազանցում է n -ի արժեքը (n -ը ցանցի հանգույցների կամ, որ նույնն է, գրաֆի գագաթների քանակն է), ապա ալգորիթմի աշխատանքն ավարտվում է՝ դուրս բերելով հաղորդագրություն՝ “Ճանապարհ չկա” (օպերատորներ 13,10):



Նկ. 3. Թռիչքների նվազագույն քանակի որոշման ալգորիթմի բլոկ-սխեման

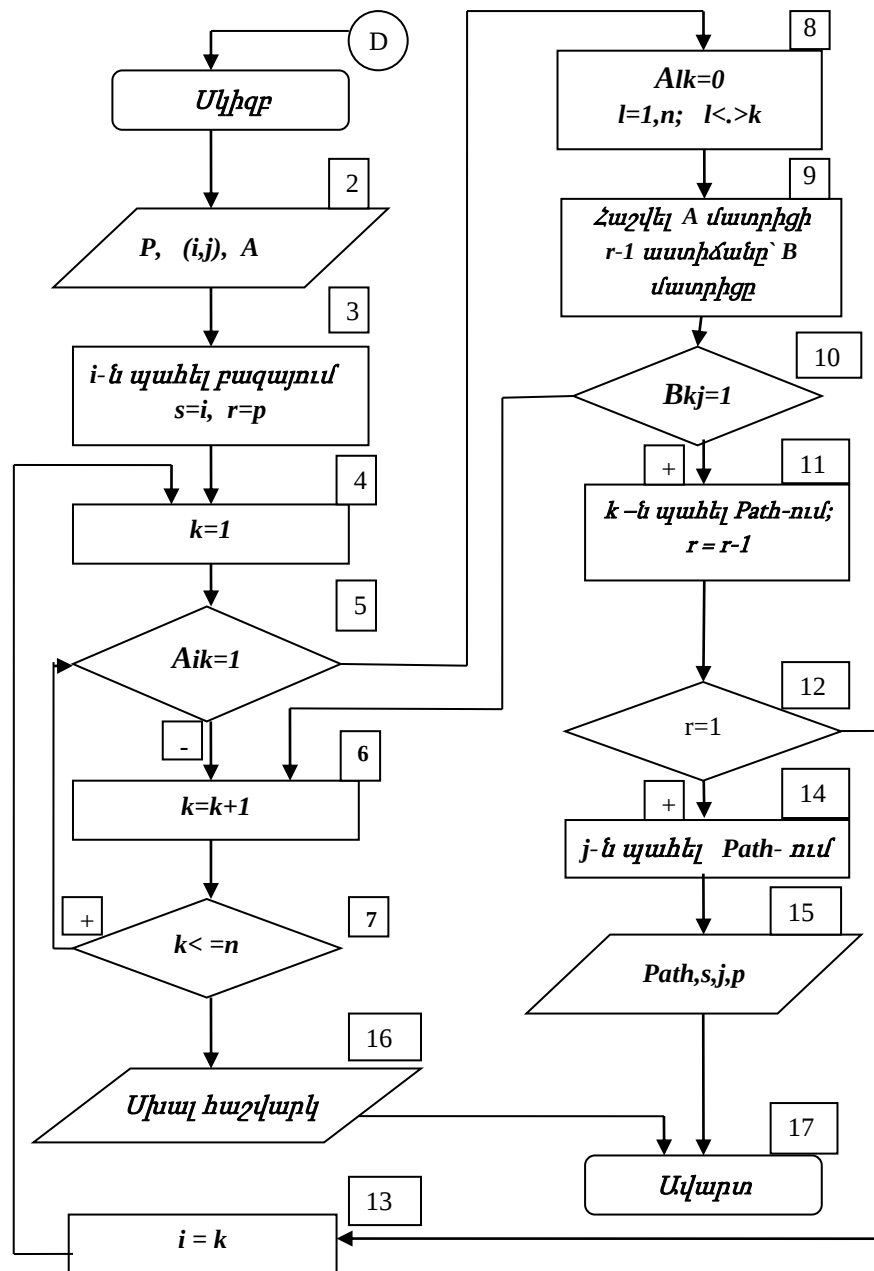
P պարամետրի արժեքը ծառայում է որպես մուտքային տվյալ ճանապարհի որոշման ալգորիթի համար:

Նկ. 4-ում բերված է թռիչքների նվազագույն քանակի վրա հիմնված ճանապարհի որոշման ալգորիթի բլոկ-սխեման: Պարզաբանենք ալգորիթի աշխատանքը: Այն սկսվում է սկզբնական տվյալների ստացումից, որոնք են նախորդ ալգորիթի արդյունք հանդիսացող P պարամետրը, կցության ճշտորոշված A մատրիցը և ուղարկող-ստացող հանգույցների հասցեները (i,j), (օպերատոր 2), որից հետո աղբյուր հանգույցի հասցեն գրանցվում է Path հենքում (օպերատոր 3)՝ որպես երթուղու առաջին հանգույց: 5,6,7,5 օպերատորներով իրականացվող ցիկլով ստուգվում են հերթական հանգույցի հարևան հանգույցները: Մինչև նպատակային հանգույցը P-1 քանակով թռիչքներով հանգույց հայտնաբերելու դեպքում այդ հարևանի հասցեն ավելացվում է Path հենքում, համարը դրվում է որպես նոր սկիզբ (օպերատորներ 8, 9, 10, 11, 12, 13), և ալգորիթի աշխատանքը վերսկսվում է օպերատոր 4-ից: Եթե հարևան հանգույցը նախավերջինն է, ապա հենքում ավելացվում է նաև վերջին հանգույցի հասցեն, դուրս են բերվում օպտիմալ երթուղու հանգույցների հասցեները կամ, որ նույնն է, գրաֆի գագաթների համարները, և ավարտվում է ալգորիթի աշխատանքը (օպերատորներ 12, 14, 15, 17): Ալգորիթը կարող է ավարտվել անարդյունք, եթե սկզբնական տվյալներում առկա են սխալներ: Այս դեպքի համար նախատեսված է համապատասխան հաղորդագրության դուրս- բերում (օպերատոր 16):

Եթե նվազագույն թռիչքների քանակով մեկից ավելի երթուղի կա, իսկ դա որոշվում է նկ. 3-ի ալգորիթով՝ մատրիցների բազմապատկման ժամանակ տրամաբանականի փոխարեն թվաբանական գումարում կատարելու դեպքում, ապա բլոկ-սխեմայի քիչ լրացումով կարելի է որոշել նաև մյուս երթուղիները:

Այս ալգորիթը խառնածին է, քանի որ այն օգտագործում է տոպոլոգիական տվյալների հենք, ինչպես բոլոր LSA ալգորիթները, և ճանապարհի լավարկման համար որպես մետրիկ՝ թռիչքների քանակը, ինչը հատուկ է DVA ալգորիթներին: Այն ունի Դեյկստրայի ալգորիթի առավելությունները և զերծ է հեռավորության վեկտորով ալգորիթների թերություններից: Թերությունը հաշվումների բարդությունն է՝ RIP-ի հետ համեմատած:

Ալգորիթը կարող է օգտագործվել RIP արձանագրությանը նման արձանագրության դեպքում:



Նկ. 4. Նվազագույն թռիչքներով քանակով երթուղու որոշման ալգորիթմի բլոկ-սխեման

ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ

1. **Таненбаум Э.** Компьютерные сети.- М.: ИС, 2005.- 993 с.
2. **Radia Perlman:** Interconnections. Second edition; Bridges, Routers, Switches and Interworking Protokols.-Boston, San-Francisco, Toronto, Montreal, London and so on. - 2004.-532p.
3. **Ճանճապանյան Հ.Ա.** Կապի վիճակի հաշվառումով երթուղավորման մի ալգորիթմի մասին//Հայաստանի ճարտարագիտական ակադեմիայի ԼՐԱԲԵՐ. Գիտատեխնիկական աշխատանքների ժողովածու,- Երևան, 2013.- Հատոր 3.- 5 էջ:
4. **Anderson A.** Discrete Mathematikc.- New-York, 2004. - 840 p.
5. **Michael J. Martin** Understanding the Network a Practocal Guide to Interworking.- Indianapolis, 2000.-687 p.

А.А. ДЖАНДЖАПАНЯН

РАЗРАБОТКА “МИКС” АЛГОРИТМА ДИНАМИЧЕСКОЙ МАРШРУТИЗАЦИИ

Разработан “микс” алгоритм динамической маршрутизации, который основан на определении минимального количества гофов. Используется топологическая база данных, как и в алгоритме Дейкстры, но вычисление минимального количества гофов дает возможность сократить область поиска оптимального маршрута. Разработанный алгоритм имеет все достоинства алгоритма Дейкстры и лишен недостатков DVA алгоритмов.

Ключевые слова: маршрут, динамическая маршрутизация, алгоритм, ориентированный граф, протокол, количество гофов.

Н.А. TSHANTSHAPANYAN

DEVELOPING A MIX ALGORYTHM FOR DYNAMIC ROUTING

A mix algorithm of dynamic routing based on the determination of minimal numbers of hops is developed. A topological database is used as in case of Dijkstra’s algorythm, but the determination of minimal number of hops gives an opportunity to reduce the search space of the optimal route. The developed algorithm has all the advantages of Dijkstra’s algorithm and is devoid of DVA algorithm flaws.

Keywords: route, dynamic routing, algorithm, oriented graphs, protocol, number of bops.