

Мрежов слой

Маршрутизация

1

Маршрутизация

- Определяне/изчисляване на оптимален маршрут за пренасяне на пакети през дадена мрежа от даден подател към определен/и получател/и, въз основа на определен/и критерий/и за оптималност.
- **Видове**
 - Определяне на следващия маршрутизатор/скок (*next-hop routing*) – най-често използван вид!
 - Определяне на целия маршрут от подателя до получателя (*source routing*) – използва се от хостовете в определени случаи, например:
 - Еталонно тестване на мрежи (*benchmarking*)
 - Маршрутизация в някои видове безжични мрежи (multi-hop ad hoc networks, MANETs)
- **Маршрутизиращи алгоритми**
 - Изчисляват оптимален маршрут (най-кратък път) през мрежата
 - Попълват и актуализират маршрутизиращи таблици
- **Маршрутизиращи таблици**
 - Поддържани в мрежовите възли
 - Съдържат информация за най-краткия път до всяка дестинация, с адреса на следващия маршрутизатор, към който трябва да се предадат пакетите.
 - Могат да се актуализират чрез обмен на информация между възлите за състоянието на мрежата

2

Маршрутизация: Критерии и метрики

Критерий	Метрика
MIN транзитно време / MIN време на закъснение	Закъснение на предаването (<i>msec</i>) / Географско разстояние (<i>km</i>)
MIN брой скокове	Брой на скоковете
MIN брой на пакетите в опашките по маршрута	Средно закъснение поради изчакване в опашките (<i>msec</i>)
MAX пропускателна способност	Скорост на предаване (<i>bps</i>)
MAX надеждност / сигурност	BER / ниво на шифриране
MIN комуникационни разходи	€ / \$

Обикновено се използват съставни метрики, чрез комбиниране на прости метрики в претеглена формула, с цел образуване на еднина метрика за стойността (*cost*).

3

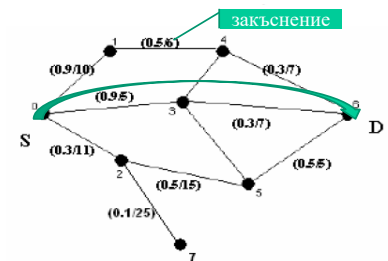
Маршрутизация: Типове метрики (1)

$M_{i,j}$ is the metric for link $\{i, j\}$

$M(P)$ = metric of path

- **Добавящи се** (*additive*)
(напр., закъснение, вариация на закъснението (*jitter*), (логаритъм на) брой успешни предавания, брой скокове, цена)
- Метриката на даден път е сума от метриките на съставните му комуникационни линии.

- **Additive** : $M(P) = M_{i,j} + M_{i,k} + \dots + M_{l,m}$



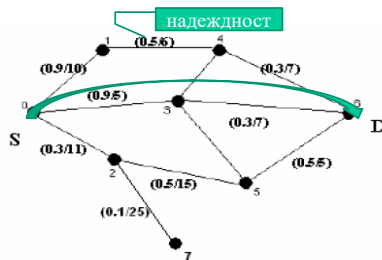
4

Маршрутизация: Типове метрики (2)

$M_{i,j}$ is the metric for link $\{i, j\}$

$M(P)$ = metric of path

- **Умножаващи се** (*multiplicative*)
(напр., надеждност, вероятност за успешно предаване)
- Метриката на даден път е произведение от метриките на съставните му комуникационни линии.



5

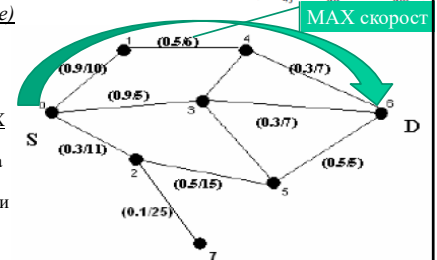
Маршрутизация: Типове метрики (3)

$M_{i,j}$ is the metric for link $\{i, j\}$

$M(P)$ = metric of path

- **С MIN-MAX** (*concave*)
(например, скорост)
- Метриката на даден път е MIN или MAX стойност на метриките на съставните му комуникационни линии.

- **Concave** : $M(P) = \min\{M_{i,j}, M_{i,k}, \dots, M_{l,m}\}$

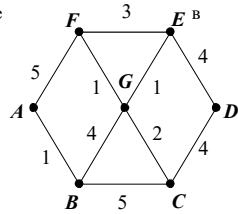


6

Маршрутизация:

Определяне на най-краткия път

- Всяка линия има **стойност** на използване в зависимост от използваните метрики:
 - Дължина
 - Закъснение
 - Степен на задръстване
 - Цена (лв/€/\$)
- Стойността** може да е различна в двете посоки!
- Стойността на даден път е функция на стойностите на съставните му линии
- Най-кратък е пътят с MIN стойност!**
- Определяне на най-краткия път:
 - Построяване на граф на мрежата
 - Вземе му са маршрутизаторите
 - Дъгите му са комуникационните линии
 - Изчисляване на цената на всяка дъга като претеглена функция на всички метрики, използвани от маршрутизиращия алгоритъм.
 - Напр. $cost = [(K1 * bandwidth) + (K2 * bandwidth) / (256 - load) + (K3 * delay)] * [(K5 / (reliability + K4)) * 256]$
 - Избиране на маршрут, съответстващ на най-краткия път през мрежовия граф.



7

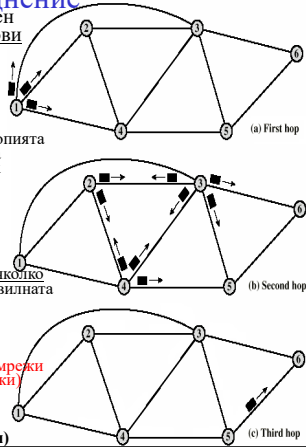
Класически маршрутизиращи стратегии

- Наводнение (*flooding*)
- По случаен начин
- Статична/фиксирана
- Динамична

10

Наводнение

- Всепакет се комутира от даден маршрутизатор към всички негови портове с изключение на този, на който е пристигнал.
 - Няколко копия на пакета пристигат до получателя
 - Всепакет е уникално номериран за установяване и отстраняване на копията (маршрутизаторите помнят номерата на пакетите вече преминали през тях и премахват повторните копия, за да намалят натоварването на мрежата)
 - Пакетите могат да съдържат и броя на скоковете, направени от тях.
- Варияции
 - Всепакет се предава само към няколко линии, водещи приблизително в правилната посока.
- Предимства**
 - MIN настройка
 - Определяне (подходяща за военни мрежи или безжични/Bluetooth mesh мрежи)
- Недостатъци**
 - Задръствания (огромен брой дублирани пакети)



11

Маршрутизация по случаен начин

- Всепакет маршрутизатор избира само 1 изходящ порт за препредаване на входящ пакет
 - Селекцията може да бъде по случаен признак или по ред на номерата в кръг (*round robin*)
- Подобрение
 - Избиране на изходящ порт въз основа на вероятности
- НЕ е необходима информация за мрежата**
 - Няма маршрутизиращи таблици
- Проста стратегия с много по-малко натоварващ трафик от наводнението
- НЕ се минимизират разходите/цената!**

13

Статична маршрутизация

- НЕ е адаптивна**
 - НЕ се основава на измервания или оценки на текущия мрежов трафик и топология
- Маршрути
 - Предварително изчислени (*off-line*)
 - Зареждат се в маршрутизаторите при инициране на мрежата
- Маршрутизиращи таблици
 - Съдържат статична информация
 - Създадени и дефинирани преди функциониране на мрежата
 - Рядко актуализирани
 - Само при значителна промяна в топологията на мрежата
 - Въз основа на очаквания трафик или капацитет
- Предимства**
 - Простота
- Недостатъци**
 - Липса на гъвкавост
 - Липса на реакция при повреда или претоварване/задръстване на мрежата
- Приложения
 - Малки мрежи
 - Опорни мрежи

ф. Иван

14

Динамична маршрутизация

- Използвана от почти всички видове мрежи с комутация на пакети
- Адаптивна**
 - Променя се във времето с цел отразяване на промените в топологията и трафика на мрежата, причинени от повреди и претоварвания/задръствания.
 - Изисква информация за мрежата
 - Маршрутизиращите таблици се актуализират често и автоматично
- Предимства**
 - Подобрена производителност
 - Подпомагане контрола на задръстванията (*congestion control*)
- Недостатъци**
 - Решенията са по-сложни
 - Компромис между качеството на информацията за мрежата и допълнителните разходи (режийните)
 - Твърде бързата реакция може да предизвика осцилации (колебания) в маршрутизирането
 - Твърде бавната реакция може да доведе до използване на информация, която вече е остаряла.
 - Теоретичните ползи може да НЕ се осъществят!
- 2 вида контрол
 - Централизиран
 - Разпределен

15

2

Динамична маршрутизация: Централизиран контрол

- 1 управляващ възел изчислява всички маршрути (по зададен алгоритъм) и актуализира маршрутизиращите таблици на всички останали мрежови възли
- Нова информация за мрежата (необходима за актуализация) се получава чрез специални пакети за поддръжка (*maintenance packets*)
 - Информират за закъснения, откази на линии/възли, задръствания и др.
- НЕ е толкова надежден колкото разпределения контрол
 - Единична точка на отказ
 - Необходимост от резервиране
- **Няма задръствания!**

16

Динамична маршрутизация: Разпределен контрол

- Всеки маршрутизатор сам изчислява маршрути (по зададен алгоритъм) и актуализира своя собствена маршрутизираща таблица, въз основа на информация:
 - Получена от другите маршрутизатори
 - Извлечена от преминаващите пакети
- **По-надежден от централизирания контрол**
- Може да предизвика задръствания!

17

Динамични разпределени маршрутизиращи алгоритми: Видове

- **С използване на дистанционни вектори** (*distance vectors*)
 - Вектори, съдържащи дистанцията/стойността и пътя (следващия маршрутизатор) до всички известни дестинации
 - Съхраняват се във всеки маршрутизатор и се обменят само със съседните маршрутизатори
 - **Недостатъци:**
 - Бавна сходимост на алгоритъма
 - Нестабилност
 - Тенденция за създаване на (затворени) маршрутни кръгове в мобилни мрежи
- **С използване на състоянието на линиите** (*link state*)
 - Поддържане на информация за топологията на (интер)мрежата в маршрутизаторите чрез периодично наводнение от страна на всеки маршрутизатор с информация за състоянието на неговите линии
 - **Недостатъци:**
 - Голям трафик, предизвикван от наводненията.
 - Огромно количество ресурси за изчисляване на маршрути в маршрутизаторите
 - Мобилността води до чести наводнения!

18

Дистанционно-векторна маршрутизация (*distance vector routing*)

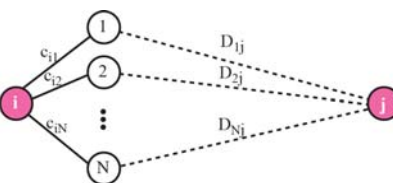
- Споделяне на информация:
 - за **цялата** (интер)мрежа
 - само между **съседите**
 - на **редовни** интервали
- Определяне на най-краткия път чрез използване на:
 - алгоритъм на **Bellman** (1957)
 - алгоритъм на **Ford-Fulkerson** (1962)

19

Алгоритъм на Bellman-Ford

- Ако всеки съсед k на възел i знае най-краткото разстояние D_{kj} до възел j , тогава най-краткото разстояние D_{ij} между възлите i и j може да бъде намерено чрез добавяне на разстоянието c_{ik} между възел i и всеки негов съсед k към съответното разстояние D_{kj} и избиране след това на MIN разстояние измежду всичките.

$$D_{ij} = \text{minimum} \{(c_{i1} + D_{1j}), (c_{i2} + D_{2j}), \dots, (c_{iN} + D_{Nj})\}$$



Legend

D_{ij} Shortest distance between i and j
 c_{ij} Cost between i and j
 N Number of nodes

Figure 11.4

Forouzan, B.A., TCP/IP Protocol Suite, 4th ed., McGraw-Hill, 2010

20

Дистанционно-векторна маршрутизация: В интермрежа (интернет)

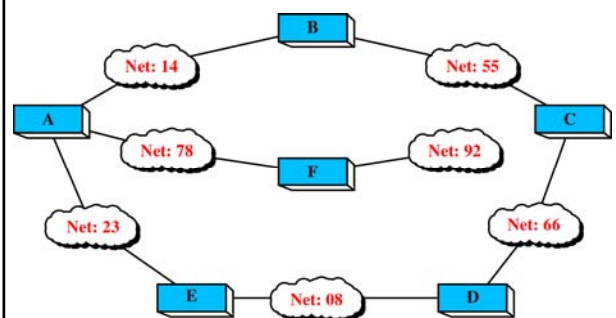


Figure 21-17

Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

21

Дистанционно-векторна маршрутизация:

Концепция

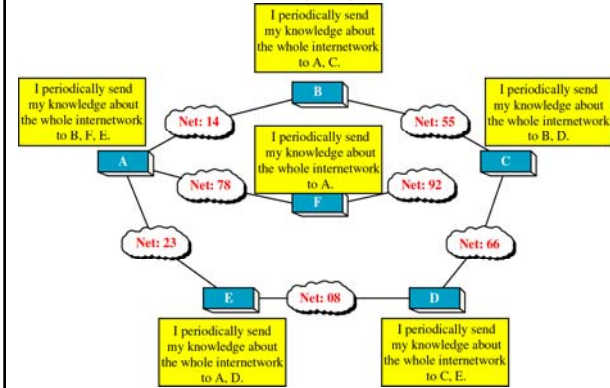


Figure 21-18

Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

22

Дистанционно-векторна маршрутизация:

Маршрутизиращи таблици

➤ ВСЕКИ МАРШРУТИЗАТОР ПОДДЪРЖА СВОЯ СОБСТВЕНА ТАБЛИЦА

Network ID	Distance	Next Hop
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

➤ КОЛОНИ

▪ **NETWORK ID** - АДРЕС НА ВСЯКА МРЕЖА ИЗВЕСТНА НА МАРШРУТИЗАТОРА

▪ **Distance** - НАЙ-КРАТКОТО РАЗСТОЯНИЕ ДО СЪОТВЕТНАТА МРЕЖА (измерено с приетата метрика, т.е. брой скокове, закъснение, цена, ...)

▪ **NEXT HOP** - АДРЕС НА СЛЕДВАЩИЯ МАРШРУТИЗАТОР, КЪМ КОЙТО ТРЯБВА ДА БЪДАТ ПРЕДАДЕНИ ПАКЕТИТЕ, ЗА ДА ДОСТИГНАТ ДО КРАЙНОТО СИ МЕСТОНАЗНАЧЕНИЕ.

23

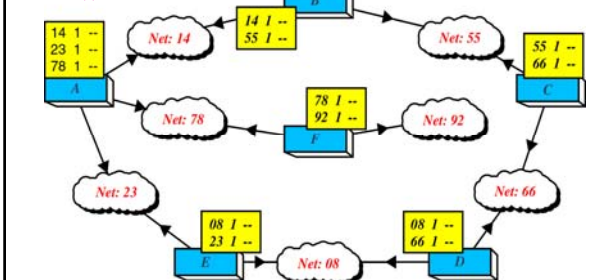
ПЪРВОНАЧАЛНО ЗАПЪЛВАНЕ НА

МАРШРУТИЗИРАЩИТЕ ТАБЛИЦИ (0. итерация)

➤ ВСЕКИ МАРШРУТИЗАТОР ИМА ТОЛКОВА ЛОГИЧЕСКИ АДРЕСА, КОЛКОТО СА МРЕЖИТЕ, КЪМ КОИТО Е СВЪРЗАН.

➤ ЧРЕЗ ИНСПЕКТИРАНЕ НА ЛОГИЧЕСКИТЕ СИ АДРЕСИ, МАРШРУТИЗАТОРЪТ ОТКРИВА КЪМ КОИ МРЕЖИ Е СВЪРЗАН И КАКВИ СА СЪОТВЕТНИТЕ ЗНАЧЕНИЯ НА МЕТРИКИТЕ (в примера тук се използват скокове: броят на скоковете към директно свързаните мрежи е 1).

➤ ТРЕТАТА КОЛОНА "СЛЕДВАЩ СКОК" Е ПРАЗНА – ТЪЙ КАТО В НАЧАЛОТО (ПРИ ИНИЦИИРАНЕ НА ИНТЕРМЕЖАТА) МАРШРУТИЗАТОРЪТ ПОЗНАВА САМО МРЕЖИТЕ, СВЪРЗАНИ ДИРЕКТНО КЪМ НЕГО.



24

ЗАПЪЛВАНЕ НА ТАБЛИЦАТА НА

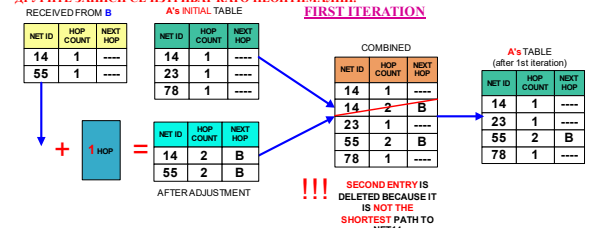
МАРШРУТИЗАТОР А (1. итерация)

➤ СЪСЕДНИТЕ МАРШРУТИЗАТОРИ НА А (т.е. B, E и F) ИЗПРАЩАТ ПЪРВОНАЧАЛНИТЕ СИ МАРШРУТИЗИРАЩИ ТАБЛИЦИ НА А

➤ А НАГЛАЖДА ИНФОРМАЦИЯТА, СЪДЪРЖАЩА СЕ В ПОЛУЧЕНИТЕ ТАБЛИЦИ, Т.Е. ДОБАВЯ РАЗСТОЯНИЕТО ОТ НЕГО ДО СЪОТВЕТНИЯ СЪСЕД (В СЛУЧАЯ 1) КЪМ ВСЕКИ ПЪТ, ПОЛУЧЕН ОТ ТОЗИ СЪСЕД.

➤ А ОСТАВЯ САМО 1 ЗАПИС ЗА ВСЯКА МРЕЖА В ТАБЛИЦАТА СИ (В СЛУЧАЯ, ЗАПИСЪТ С МИНИМАЛЕН БРОЙ СКОКОВЕ).

➤ В СЛУЧАЯ НА 2 (ИЛИ ПОВЕЧЕ) ЗАПИСА СЪС СЪЩИЯ БРОЙ СКОКОВЕ ДО ДАДЕНА МРЕЖА, САМО ЗАПИСЪТ С МИНИМАЛНА СТОЙНОСТ НА СЛЕДВАЩАТА КООРДИНАТА (Т.Е. "NEXT HOP") АДРЕС НА СЛЕДВАЩИЯ МАРШРУТИЗАТОР) ОСТАВА В ТАБЛИЦАТА; ДРУГИТЕ ЗАПИСИ СЕ ИЗТРИВАТ КАТО НЕОПТИМАЛНИ.



25

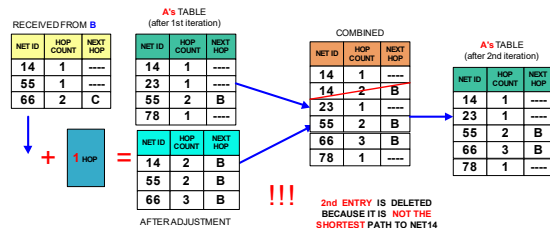
ЗАПЪЛВАНЕ НА ТАБЛИЦАТА НА

МАРШРУТИЗАТОР А (2. итерация)

➤ ПО ВРЕМЕ НА 1. ИТЕРАЦИЯ, В Е ОБМЕНИЛ ПЪРВОНАЧАЛНАТА СИ ТАБЛИЦА СЪС СВОЯ ДРУГ СЪСЕД С И СЕГА ИМА НОВА ТАБЛИЦА, КОЯТО ИЗПРАЩА НА А.

➤ ПРИЛАГАТ СЕ СЪЩИТЕ ПРАВИЛА КАКТО ПРИ 1. ИТЕРАЦИЯ

SECOND ITERATION



26

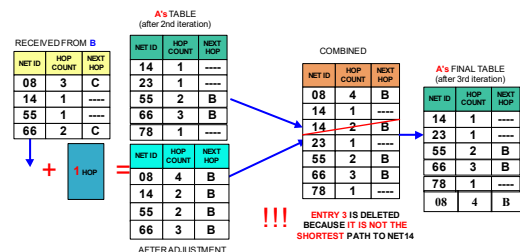
ЗАПЪЛВАНЕ НА ТАБЛИЦАТА НА

МАРШРУТИЗАТОР А (3. итерация)

➤ ПО ВРЕМЕ НА 2. ИТЕРАЦИЯ В Е ОБМЕНИЛ ИНФОРМАЦИЯ СЪС СВОЯ ДРУГ СЪСЕД С И ИМА НОВА ТАБЛИЦА, КОЯТО СЕГА ИЗПРАЩА НА А.

➤ ПРИЛАГАТ СЕ СЪЩИТЕ ПРАВИЛА КАКТО ПРИ ДРУГИТЕ ИТЕРАЦИИ

THIRD ITERATION



27

Окончателни маршрутизиращи таблици

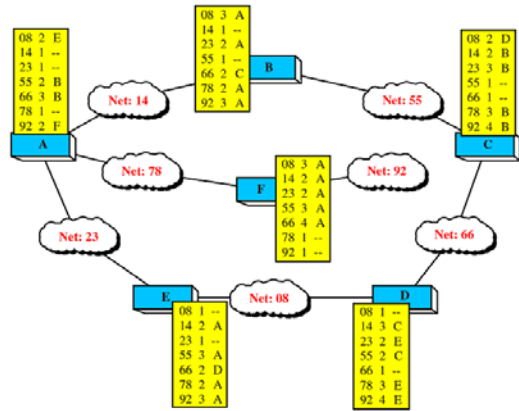


Figure 21-22

Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

28

Актуализиране на маршрутизиращите таблици

> **ПЕРИОДИЧНО** (по **ПРЕДВАРИТЕЛНО ЗАДАДЕНИ ИНТЕРВАЛИ**) СЪСЕДНИТЕ МАРШРУТИЗАТОРИ ПРОДЪЛЖАВАТ ДА ОБМЕНЯТ СВОИТЕ ТЕКУЩИ ТАБЛИЦИ, СЪДЪРЖАЩИ ИНФОРМАЦИЯ ЗА ВСИЧКИ ИЗВЕСТНИ ИМ МРЕЖИ.

> ИЗПОЛЗВАЙКИ ТАЗИ ИНФОРМАЦИЯ, ВСЕКИ МАРШРУТИЗАТОР АКТУАЛИЗИРА СВОЯТА ТАБЛИЦА ЧРЕЗ ПРИЛАГАНЕ НА СЛЕДНИЯ **АКТУАЛИЗИРАЩ АЛГОРИТЪМ**:

- ✓ Ако досега НЕ е разполагал с информация за новорекламираната дестинация, маршрутизаторът я **добавя** незабавно в таблицата си;
- ✓ В противен случай:

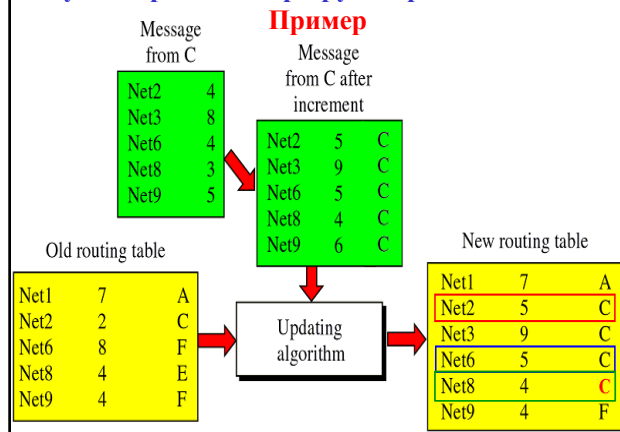
1) Ако **значението на полето NEXT HOP** е същото (което означава, че информацията е изпратена от същия рекламиращ я маршрутизатор), новата информация **заменя** стария запис в таблицата, **даже** ако новият запис е с **по-голямо РАЗСТОЯНИЕ!**

2) В противен случай (т.е. информацията идва от друг маршрутизатор):

- Ако **РАЗСТОЯНИЕТО** в новия запис е **по-малко** от това в съществуващия (стар) запис, новият запис **изменя** стария.
- Ако **РАЗСТОЯНИЕТО** в новия запис е **по-голямо** от това в съществуващия (стар) запис, старият запис **остава** (т.е. таблицата **не се променя**).
- Ако **РАЗСТОЯНИЕТО** в новия запис е **равно** на това в съществуващия (стар) запис, записът с **по-малко NEXT HOP** значение **остава** в таблицата.

29

Актуализиране на маршрутизиращите таблици:

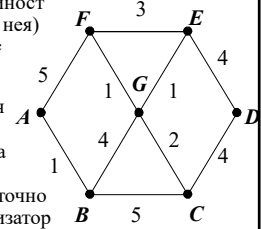


30

Дистанционно-векторна маршрутизация:

В мрежа

- Принципът е същият както при интермрежа
- Всяка комуникационна линия има стойност за използване (изпращане на пакет през нея)
 - Стойността може да бъде различна в двете посоки!
- Всеки маршрутизатор поддържа маршрутизираща таблица с информация за изпращане на пакети до всеки друг маршрутизатор в мрежата, въз основа на **минималната възможна стойност**.
- Маршрутизиращата таблица съдържа точно един запис за всеки един друг маршрутизатор (**r**) в мрежата, съдържащ:
 - 1) Разстоянието (**d**) до съответния маршрутизатор;
 - 2) Адреса/идентификатора на следващия маршрутизатор (**n**), към който трябва да се предават пакетите, предназначени за съответния маршрутизатор.



31

Първоначални маршрутизиращи таблици



- Първоначално всяка таблица съдържа информация само за съседите
 - Например, таблицата на маршрутизатор **A** съдържа информация само за маршрутизатори **F** и **B**
- Обменът на информация със съседите при следващите итерации завършва запълването на таблиците

ф. Иван

32

Маршрутизиращи таблици след обмен на първоначалните таблици (1. итерация)

A			B			C			D			E			F			G		
r	d	n	r	d	n	r	d	n	r	d	n	r	d	n	r	d	n	r	d	n
B	1	B	A	1	A	A	6	B	B	9	C	A	8	F	A	5	A	A	5	B
C	6	B	C	5	C	B	5	B	C	4	C	B	5	G	B	5	G	B	4	B
E	8	F	D	9	C	D	4	D	E	4	E	C	3	G	C	3	G	C	2	C
F	5	F	E	5	G	E	3	G	F	7	E	D	4	D	D	7	E	D	5	E
G	5	B	F	5	G	F	2	G	G	5	E	F	2	G	E	2	G	E	1	E
			G	4	G	G	2	G				G	1	G	G	1	G	F	1	F

- Всеки маршрутизатор изпраща своята таблица на съседите си
- Изпращаната таблица съдържа адреса на изпращащия маршрутизатор (записан в полето **n**), заедно с цената за изпращане на пакет от този маршрутизатор към всяка известна на него дестинация в мрежата.
- Маршрутизаторите променят таблиците си по такъв начин, че да сведат до **минимум стойността** до всяка конкретна дестинация.
- След получаване на таблицата на съсед **i**, маршрутизатор **j** добавя стойността на линията **i-j** към всяка нова (получена) дестинация.
- В таблицата остава само най-краткият маршрут към дадена дестинация.

33

Окончателни маршрутизиращи таблици

<i>A</i>			<i>B</i>			<i>C</i>			<i>D</i>			<i>E</i>			<i>F</i>			<i>G</i>		
<i>r</i>	<i>d</i>	<i>n</i>	<i>r</i>	<i>d</i>	<i>n</i>	<i>r</i>	<i>d</i>	<i>n</i>	<i>r</i>	<i>d</i>	<i>n</i>	<i>r</i>	<i>d</i>	<i>n</i>	<i>r</i>	<i>d</i>	<i>n</i>	<i>r</i>	<i>d</i>	<i>n</i>
B	1	B	A	1	A	A	6	B	A	10	C	A	6	G	A	5	A	A	5	B
C	6	B	C	5	C	B	5	B	B	9	C	B	5	G	B	5	G	B	4	B
D	10	B	D	9	C	D	4	D	C	4	C	C	3	G	C	3	G	C	2	C
E	6	B	E	5	G	E	3	G	E	4	E	D	4	D	D	6	G	D	5	E
F	5	F	F	5	G	F	3	G	F	6	E	F	2	G	E	2	G	E	1	E
G	5	B	G	4	G	G	2	G	G	5	E	G	1	G	G	1	G	F	1	F

- По време на следващите итерации маршрутизаторите добавят новопридобитата информация за мрежата към своите таблици и ги изпращат на съседите си.
- По този начин, всеки маршрутизатор получава обратно своята собствена информация заедно с нова информация за другите съседни на своите съседни.
- След определен брой итерации всеки маршрутизатор ще получи информация за всеки друг маршрутизатор в мрежата.
- След това **периодично** всеки маршрутизатор продължава да изпраща информацията си за цялата мрежа към своите съседни. Това споделяне на информация се извършва, независимо дали мрежата се е променила от последния път, или не.

34

Маршрутизация с използване на състоянието на линиите (link state routing)

- Всеки маршрутизатор споделя информация:
 - За **съседство**
 - Открива всички съседни чрез изпращане на пакети HELLO по всяка една своя комуникационна линия
 - Научава техните мрежови адреси
 - Оценява **състоянието на линията** към всеки един от тях (например, какво е закъснението, скоростта, цената и т.н.)
 - Например, закъснението се измерва чрез изпращане на ECHO пакети.
 - С **всички маршрутизатори** в (интер)мрежата
 - Създава **Link State Packet (LSP)**, съдържащ текущото състояние на всички свои локални линии.
 - Изпраща **LSP** до всички маршрутизатори, използвайки наводнение (**flooding**).
 - Всички **LSP** се потвърждават по линиите между всеки два маршрутизатора за предотвратяване на грешки
 - Веднага** когато има **значителна промяна** в състоянието на дадена линия
 - И **периодично**, за да се гарантира премахването на стара информация
 - Период $LSP > \text{Период } P_{LSP}$ (по-дългият период предпазва от създаването на прекалено голям трафик от наводненията)
- Всеки маршрутизатор формира база данни **LSD (Link State Database)**, създава **дърво с най-кратките пътища** (чрез изп. на алгоритъма на **Dijkstra**), създава/обновява маршрут. таблица.

35

Маршрутизация с използване на състоянието на линиите: Концепция

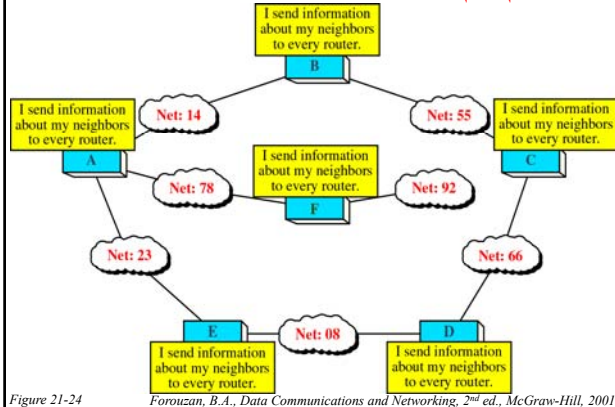


Figure 21-24

Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

36

Маршрутизация с използване на състоянието на линиите: Стойност

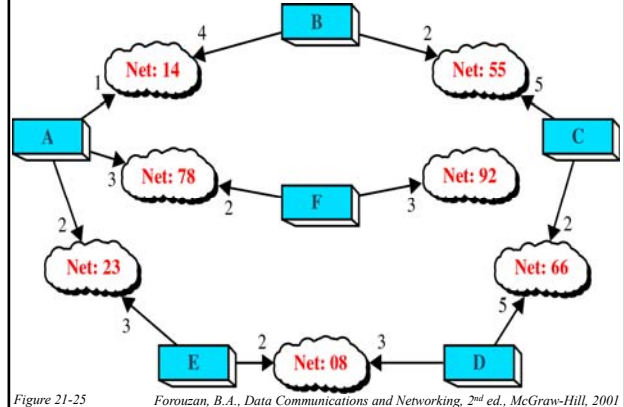


Figure 21-25

Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

37

Пакет за състоянието на линиите (Link State Packet, LSP)

- Увеличава се с 1 за всеки нов изпратен LSP
- Използва се за отличаване на нови от стари LSP

- Предотвратява съществуването на стари LSP за дълго време в мрежата
- Подобно на TTL полето в IP протокола

Advertiser	Network	Cost	Neighbor	Sequence No	Age
.....
.....
.....

ф. Иван

38

Наводнение с LSP пакет на маршрутизатор A

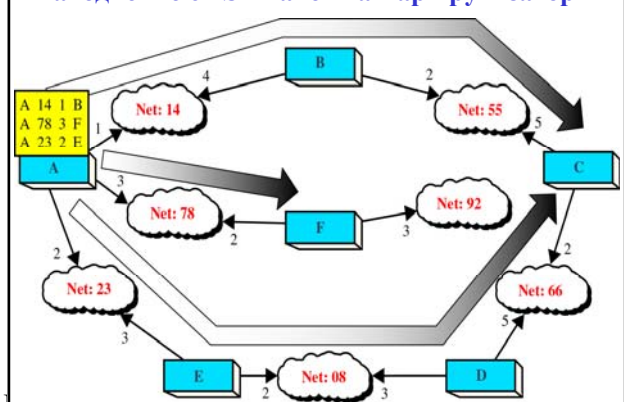


Figure 21-27

Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

39

База данни за състоянието на линиите (Link State Database, LSD)

Advertiser	Network	Cost	Neighbor
A	14	1	B
A	78	3	F
A	23	2	E
B	14	4	A
B	55	2	C
C	55	5	B
C	66	2	D
D	66	5	C
D	08	3	E
E	23	3	A
E	08	2	D
F	78	2	A
F	92	3	—

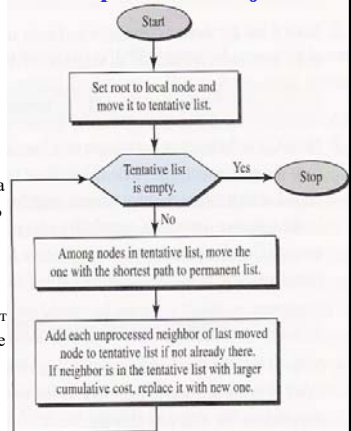
Figure 21-28 Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

40

Създаване на дърво с най-кратките пътища по сл. начин:

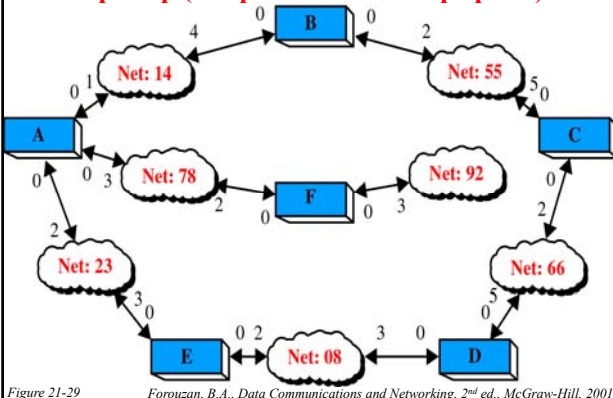
1. Идентифициране на корена на дървото (самия маршрутизатор).
2. Добавяне на всички възли, които могат да бъдат достигнати от този корен (т.е. първоначално всички съседни възли).
– Възлите и дъгите са временни на този етап.
3. Сравняване на временните дъги и идентифициране на дъгата, през която преминава път с най-ниска стойност.
– Тази дъга и възелът, към който води тя, стават постоянна част на дървото.
4. Разглеждане на LSD и идентифициране на всеки възел, до който може да се стигне директно от новодобавения възел.
– Новите възли и дъги се добавят временно към дървото.
5. Повторяне на последните две стъпки, докато всички възли в (интер)мрежата станат постоянна част на дървото.
– Постоянни дъги са само тези, по които преминават най-кратки пътища (с най-ниска стойност) до останалите възли.

Алгоритъм на Dijkstra



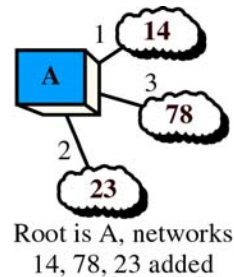
41

Алгоритъм на Dijkstra: Пример (за прилагане в интермрежа)

Figure 21-29 Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

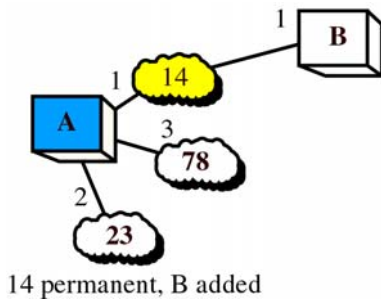
42

Определяне на най-краткия път: Част I

Figure 21-30, Part I Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

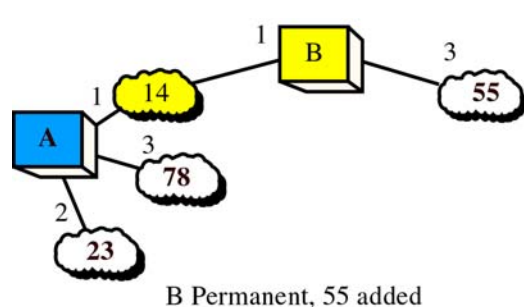
43

Определяне на най-краткия път: Част II

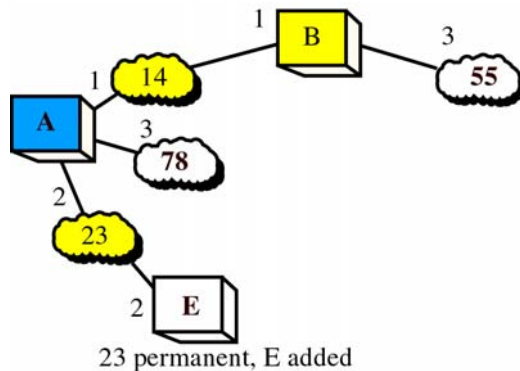
Figure 21-30, Part II Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

44

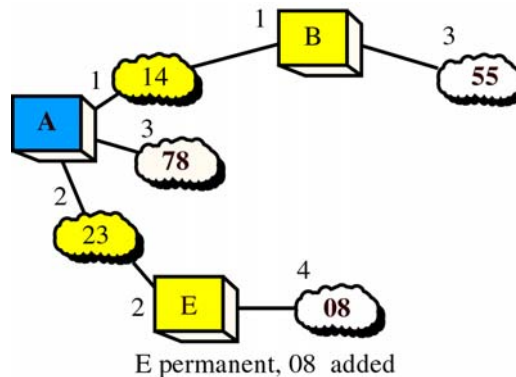
Определяне на най-краткия път: Част III

Figure 21-30, Part III Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

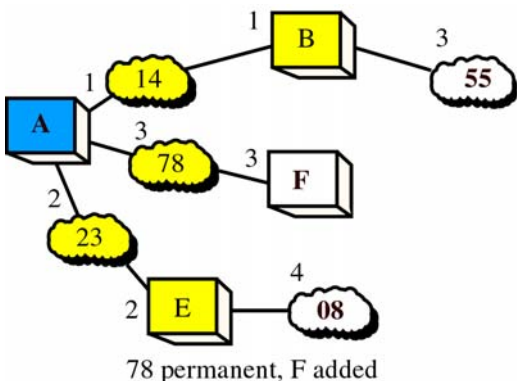
45

Определяне на най-краткия път: **Част IV**Figure 21-30, Part IV Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

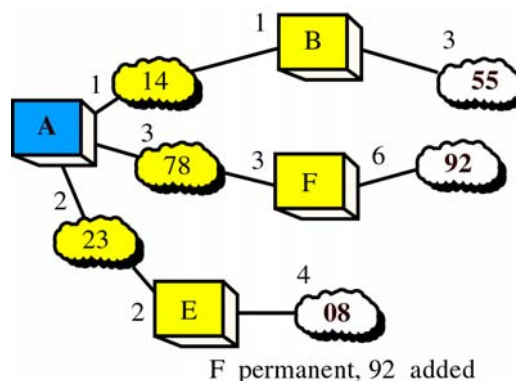
46

Определяне на най-краткия път: **Част V**Figure 21-30, Part V Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

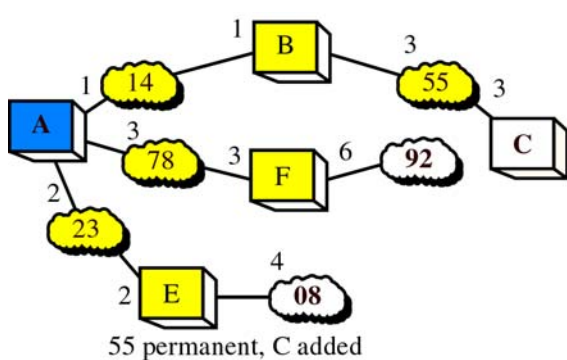
47

Определяне на най-краткия път: **Част VI**Figure 21-30, Part VI Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

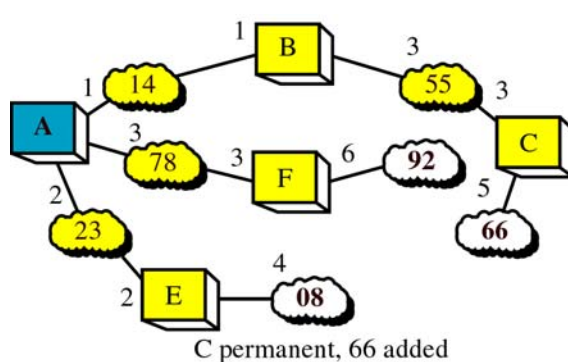
48

Определяне на най-краткия път: **Част VII**Figure 21-31, Part VII Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

49

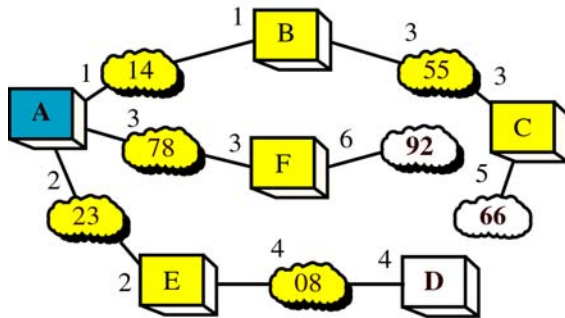
Определяне на най-краткия път: **Част VIII**Figure 21-31, Part I Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

50

Определяне на най-краткия път: **Част IX**Figure 21-31, Part II Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

51

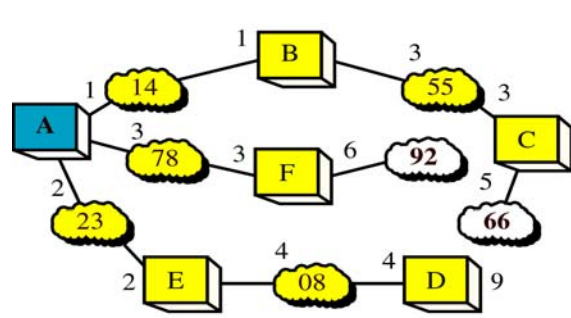
ф. Иван

Определяне на най-краткия път: **Част X**

08 permanent, D added

Figure 21-31, Part III Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

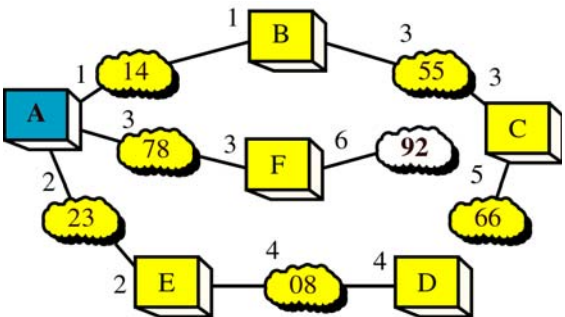
52

Определяне на най-краткия път: **Част XI**

D permanent, 66 added.

But $9 > 5$, so that link deletedFigure 21-31, Part IV Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

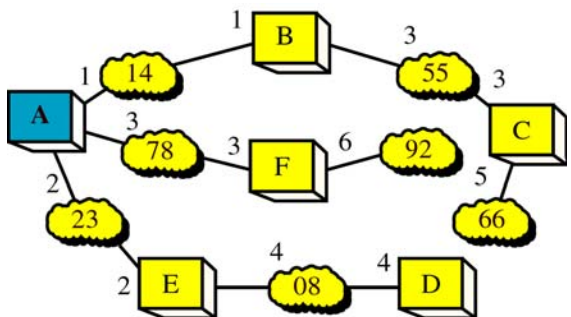
53

Определяне на най-краткия път: **Част XII**

66 permanent

Figure 21-31, Part V Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

54

Определяне на най-краткия път: **Част XIII**

92 permanent

Figure 21-31, Part VI Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

55

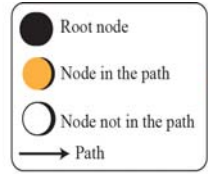
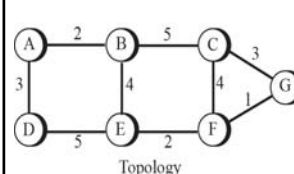
Окончателна маршрутизираща таблица
на маршрутизатор A

Net	Cost	Next router
08	4	E
14	1	--
23	2	--
55	3	B
66	5	B
78	3	--
92	6	F

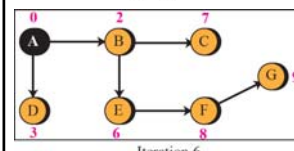
Figure 21-32 Forouzan, B.A., Data Communications and Networking, 2nd ed., McGraw-Hill, 2001

56

Алгоритъм на Dijkstra:

Пример (за прилагане в мрежа,
за маршрутизатор A)

Legend



Routing Table for Node A

Destination	Cost	Next Router
B	2	—
C	7	B
D	3	—
E	6	B
F	8	B
G	9	B

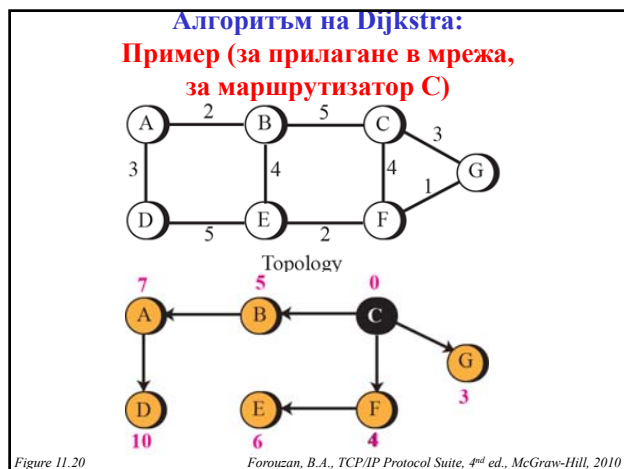
Figure 11.19

Forouzan, B.A., TCP/IP Protocol Suite, 4th ed., McGraw-Hill, 2010

57

ф. Иван

9



58