 **Advanced Computer Networks**

Internal routing - distance vector protocols

Prof. Andrzej Duda
duda@imag.fr

<http://duda.imag.fr>

1

Contents

- Principles of internal routing
- Distance vector (Bellman-Ford)
 - principles
 - case of link failures
 - count to infinity
 - split horizon
- RIP
- RIP v2
- IGRP

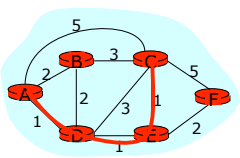
2

Routing

Routing protocol
Goal: determine "good" path (sequence of routers) thru network from source to dest.

Graph abstraction for routing algorithms:

- graph nodes are routers
- graph edges are physical links
 - link cost: delay, \$ cost, or congestion level



- "good" path:
 - typically means minimum cost path
 - other def's possible

3

Routing algorithms

- Problem
 - find the **best** route to a destination
- What does it mean the best?
 - metric to measure how a route is good
 - hops
 - link capacity
 - performance measures: link load, delay
 - cost
- Graph optimization - Shortest Path
 - find the shortest path in a graph
 - shortest in the sense of a metric

4

Main algorithms

- Distance vector (Bellman-Ford)
 - routers only know their local state
 - link metric and neighbor estimates
 - internal routing protocols (RIP, IGRP)
- Link state
 - knowledge of the global state
 - metrics of all links
 - global optimization (Shortest Path First - Dijkstra)
 - internal routing protocols (OSPF, PNNI (ATM))
- Path vector
 - knowledge of the global state
 - path: sequence of AS with attributes
 - global optimization and policy routing
 - external routing protocols (BGP)

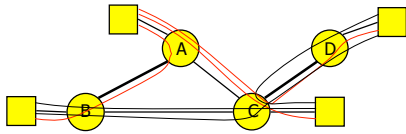
5

Metrics

- Static - do not depend on the network state
 - number of hops
 - link capacity and static delay
 - cost
- Dynamic - depend on the network state
 - link load
 - current delay

6

Traffic matrix



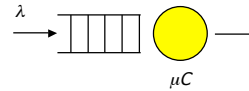
	A	B	C	D
A	0	8	2	3
B		0	6	4
C			0	1
D				0

	A	B	C	D
A		AB	AC	ACD
B			BC	BCD
C				CD
D				

7

Traffic

- Link model
 - queueing system M/M/1
 - exponentially distributed service and interarrival times



$$T = \frac{1}{\mu C - \lambda}$$

8

Delay

- Parameters
 - 1 Mb/s and 0.5 Mb/s links
 - mean packet length $1/\mu$ - 5 Kbytes (40 000) bits
 - transmission time on 1 Mb/s link: 40 ms
 - transmission time on 0.5 Mb/s link: 80 ms

	λ pq/s	C Mb/s	μC pq/s	T
AB	8	1	25	58 ms
AC	5	0.5	12.5	133 ms
BC	10	0.5	12.5	400 ms
CD	8	1	25	58 ms

9

Flooding

- Simple and robust routing
 - no need for routing tables
 - each packet duplicated on each outgoing link
 - packet duplication
 - duplicated packets destroyed at destination
 - robust - tolerates link or router failures
 - optimal in some sense
 - the first packet has found the shortest path to the destination
 - cannot be compared to the shortest path calculated by Link State - no packet duplication
- Problem
 - increased load due to packet duplication
- Used in OSPF to distribute link state information and in ad hoc routing protocols (AODV, OLSR)

10

Distance vector

- Dynamic routing based on distributed estimation of the distance to the destination
 - uses the distributed algorithm by Bellman-Ford (dynamic programming)
 - each router receives aggregated information from its neighbors
 - estimates the local cost to its neighbors
 - computes the best routes
 - no global network states
- Distance
 - number of hops
 - delay

11

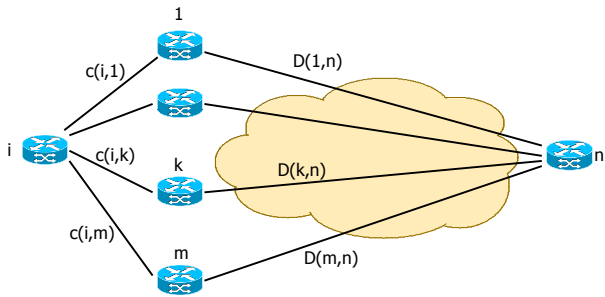
Bellman-Ford algorithm

- Bellman-Ford algorithm
 - node i knows cost $c(i,k)$ to its immediate neighbours ($+\infty$ for most values of k)
 - distance $D(i,n)$ is given by: $D(i,n) = \min_k (c(i,k) + D(k,n))$
 - in the worst case, convergence after $N-1$ iterations
- Distributed Bellman-Ford algorithm
 - initially: $D(i,n) = 0$ if i directly connected to n and $D(i,n) = +\infty$ otherwise
 - node i receives from neighbour k latest values of $D(k,n)$ for all n (distance vector)
 - node i computes the best estimates

$$D(i,n) = \min_k (c(i,k) + D(k,n))$$

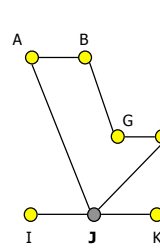
12

Bellman-Ford algorithm



13

Example of Bellman-Ford



	A	I	H	K	
A	0	24	20	21	
B	12	36	31	28	
G	18	31	6	31	
H	17	20	0	19	
I	21	0	14	22	
J	9	11	7	10	
K	24	22	22	0	
J:	8	10	12	6	

Table of J

8	A
20	A
18	H
12	H
10	I
0	-
6	K

computation of G : 18+8=26, 31+10=41, 6+12=18, 6+31=37
 → choice of 18, H

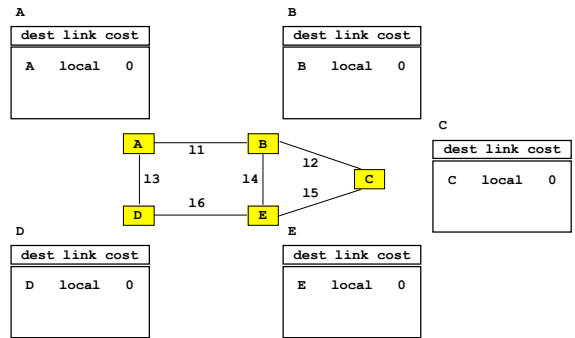
14

Distance vector example

- Simple network
 - routers connected by links
 - destinations = subnetworks connected to routers
 - symmetric links
 - cost = number of hops

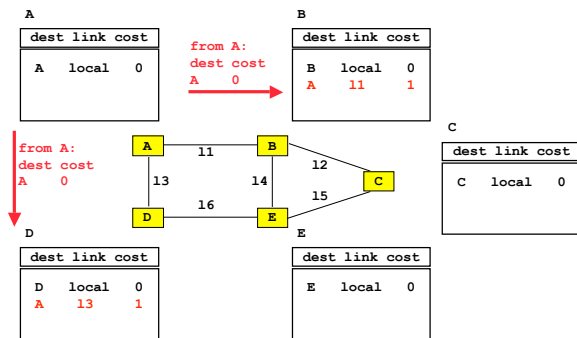
15

Initialization



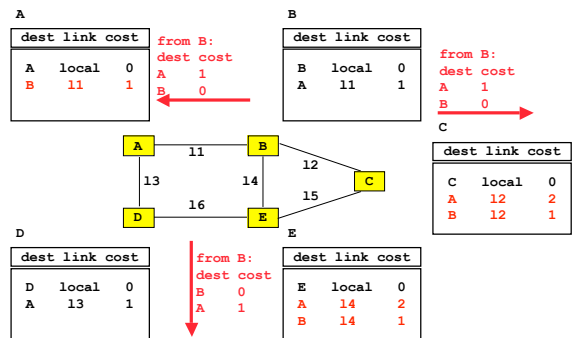
16

Distance vector announcement



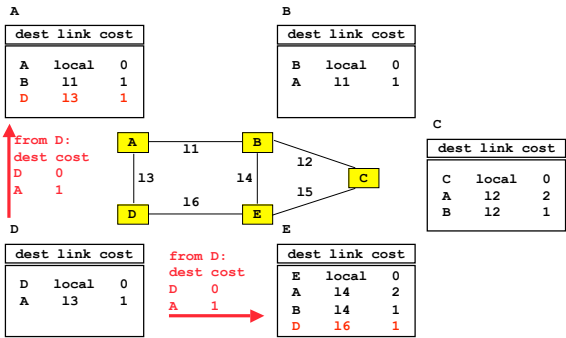
17

Distance vector announcement



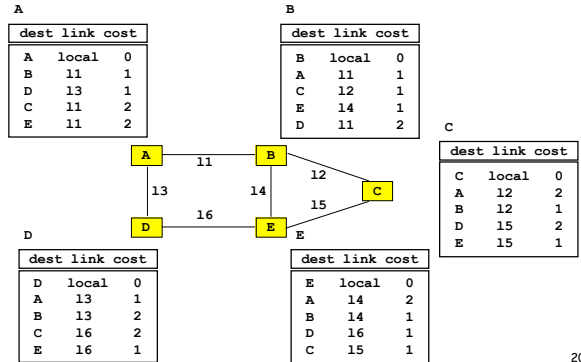
18

Distance vector announcement



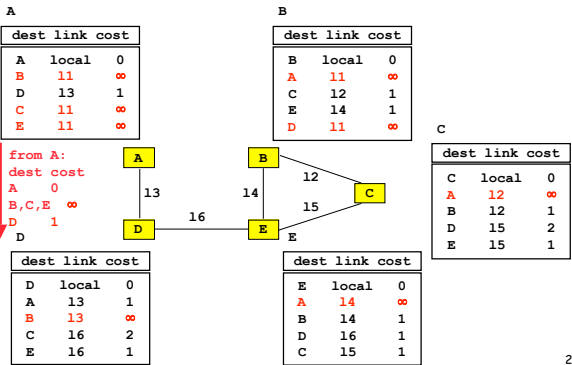
19

Final



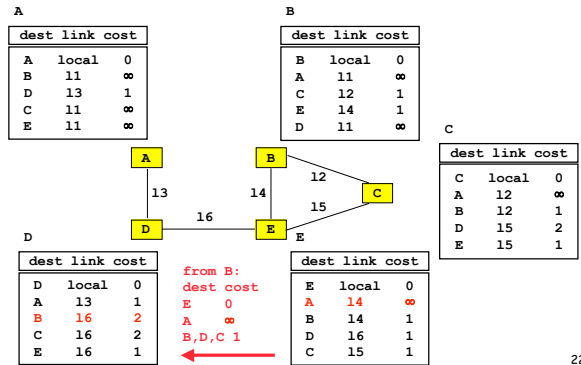
20

Link failure



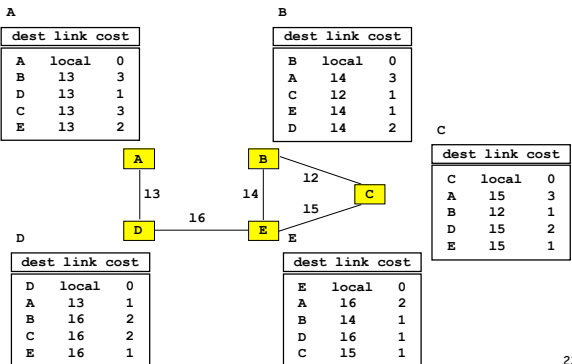
21

Link failure



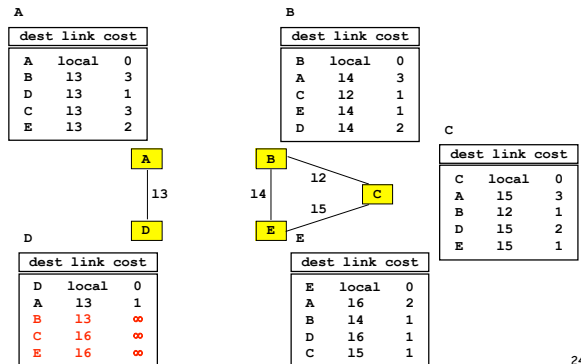
22

Final state after failure



23

Equal link costs - link failures



24

Counting to infinity

A

dest	link	cost
A	local	0
B		13
D		13
C		13
E		13

from A:
dest cost

A	0
B,C	3
D	1
E	2

D

dest	link	cost
D	local	0
A		13
B		13
C		13
E		13

- Loop between A and D
- Exchange of routes, costs increase by 2 each cycle
- Convergence to a stable state
 - ∞ = large number
 - e.g. RIP: ∞ = 16

25

Split horizon

- Minimize the effects of bouncing and counting to infinity
- Rule
 - if A routes packets to X via B, it does not announce this route to B

26

Example of split horizon

A

dest	link	cost
A	local	0
B		13
D		13
C		13
E		13

B

dest	link	cost
B	local	0
A		14
C		12
E		14
D		14

C

dest	link	cost
C	local	0
A		15
B		12
D		15
E		15

D

dest	link	cost
D	local	0
A		13
B		13
C		16
E		16

E

dest	link	cost
E	local	0
A		16
B		14
D		16
C		15

27

Split horizon

A

dest	link	cost
A	local	0
B		13
D		13
C		13
E		13

from A:
dest cost

A	0
---	---

D

dest	link	cost
D	local	0
A		13
B		13
C		16
E		16

- Split horizon cuts the process of counting to infinity

28

Split horizon

A

dest	link	cost
A	local	0
B		13
D		13
C		13
E		13

from D:
dest cost

D	0
B,C,E	∞

D

dest	link	cost
D	local	0
A		13
B		13
C		16
E		16

- Split horizon cuts the process of counting to infinity

29

Split horizon may fail

B

dest	link	cost
B	local	0
A		14
C		12
E		14
D		14

from E:
dest cost

A	∞
B	1
C	1
D	∞

C

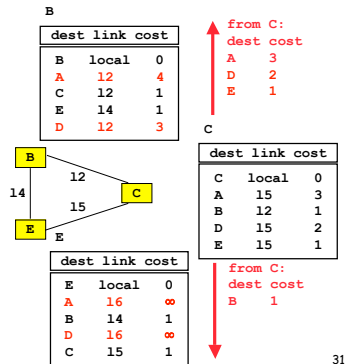
dest	link	cost
C	local	0
A		15
B		12
D		15
E		15

E

dest	link	cost
E	local	0
A		16
B		14
D		16
C		15

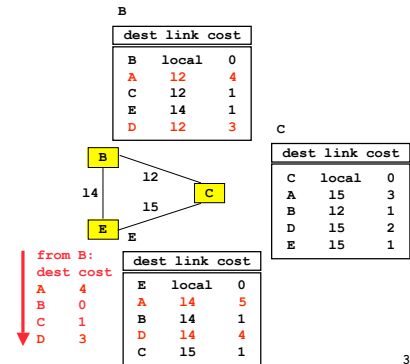
30

Split horizon may fail



31

Split horizon may fail



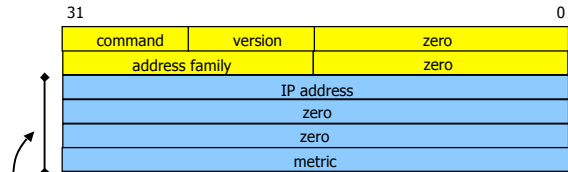
32

RIP v1

- Distance vector protocol
- Metric - hops
- Network span limited to 15
 - ∞ = 16
- Split horizon
- Destination network identified by IP address
 - no prefix/subnet information - derived from address class
- Encapsulated as UDP packets, port 520
- Largely implemented (routed on Unix)
- Broadcast every 30 seconds or when update detected
- Route not announced during 3 minutes
 - cost becomes ∞

33

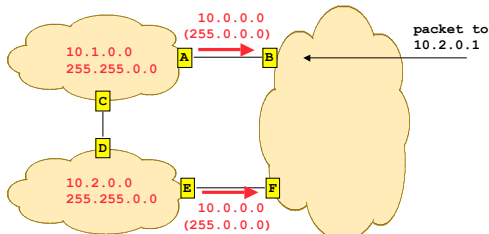
Message format



- May be repeated 25 times
- Command
 - REQUEST - 1 (sent at boot to initialize)
 - RESPONSE - 2 (broadcast each 30 sec)

34

Missing netmask



- A and E can forward to 10.0.0.0
- Packet to 10.2.0.1 can go through F or B
 - if sent to B, it goes through A and C
- If link C-D broken, no route to destination

35

RIP v2 (RFC 2453)

- Subnetworks
 - take into account CIDR prefixes and netmasks
- Authentication
- Multicast
 - 224.0.0.9 mapped to MAC 01-00-5E-00-00-09
 - on LAN only, no need for IGMP

36

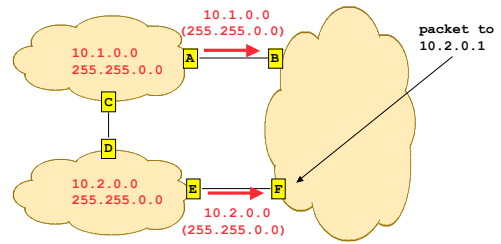
Message format

31		0	
command	version	unused	
address family		route tag	
IP address			
netmask			
next router			
metric			

- Command, version unchanged
- One address family - authentication
- Next router
 - used at the border of different routing domains (e.g. RIP and OSPF)
- Route tag
 - for external routes (used by BGP)

37

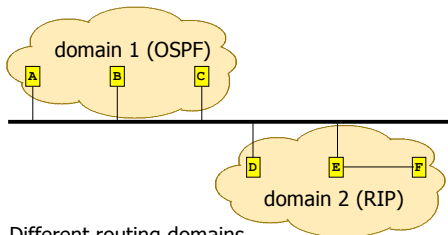
Announcing netmasks



- E can forward to 10.2.0.0
- Packet to 10.2.0.1 can go through F

38

Routing domains



- Different routing domains
 - e.g. routers under different administrations that run different routing protocols (RIP, OSPF)
- If A wants to send a packet to F, it goes through D and E
- When announcing F, D adds E as **next router**

39

Simple authentication

31		0	
command	version	unused	
xFFFF		authentication type = 2	
password on 16 bytes			

- Configuration of gated (/etc/gated.conf)


```
rip yes {
    interface all
    version 2 multicast
    authentication simple "qptszwmz"
}
```

40

MD5 authentication

31		0	
command	version	unused	
xFFFF		authentication type = 3	
packet length	key Id	auth. length	
increasing sequence no.			
zero			
zero			
route info			
xFFFF		x01	
seal			

41

MD5 authentication

- Seal
 - MD5 digest on the message using a shared secret
 - sequence number avoids replay attacks
- Configuration of gated (/etc/gated.conf)


```
rip yes {
    interface all
    version 2 multicast
    authentication md5 "qptszwmz"
}
```

42

Conclusion

- Main distance vector protocols
- Largely deployed (Unix BSD `routingd`)
- Simplicity
- Slow convergence
- Not suited for large and complex networks
 - Link State protocols should be used instead

43

44