EC 553
Communication Networks

Mohamed Khedr

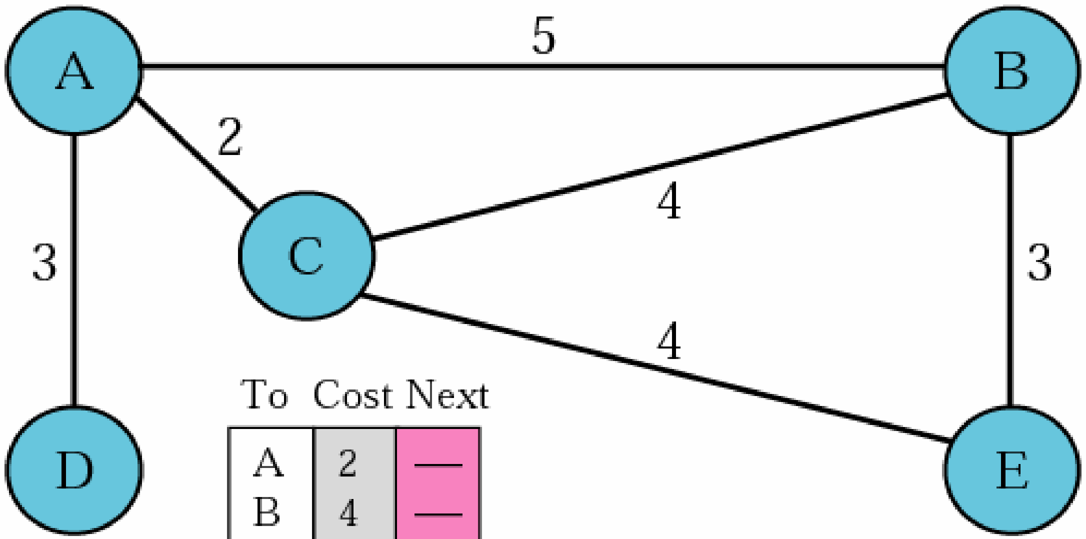
http://webmail.aast.edu/~khedr

Syllabus

- Tentatively

Week 1	Overview
Week 2	Packet Switching
Week 3	IP addressing and subnetting
Week 4	IP addressing and subnetting
Week 5	Introduction to Routing concept, Routing algorithms
Week 6	Routing protocols
Week 7	Multiple Access I
Week 8	Multiple access II
Week 9	LAN networks
Week 10	Token ring networks
Week 11	VOIP
Week 12	WLAN
Week 13	TCP
Week 14	Congestion control
Week 15	QOS

Figure 14.3 *Distance vector routing tables*



A's table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

B's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

D's table

To	Cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

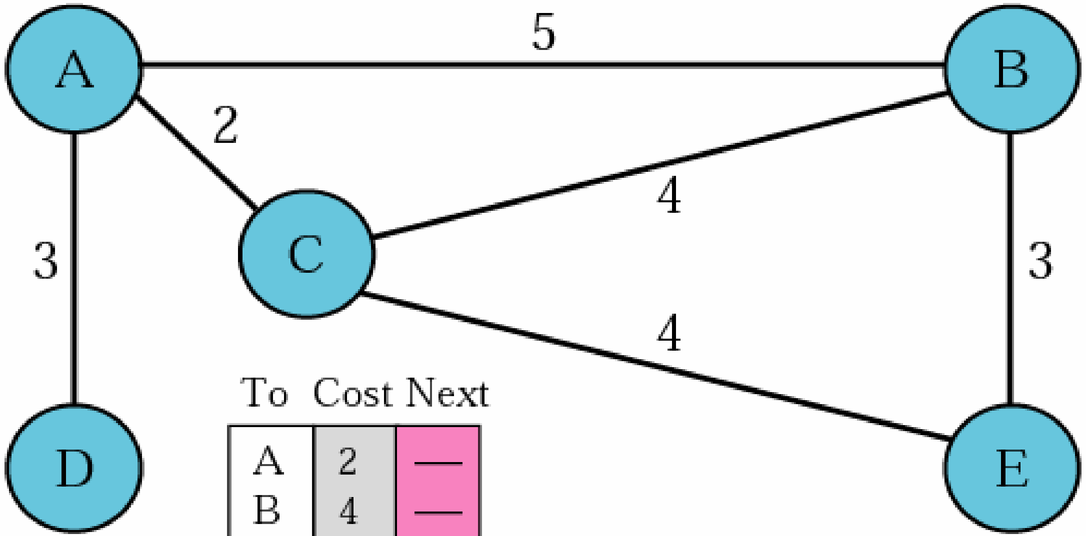
C's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

E's table

To	Cost	Next
A	6	C
B	3	—
C	4	—
D	9	C
E	0	—

Figure 14.4 *Initialization of tables in distance vector routing*



A's Table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	∞	

B's Table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	∞	
E	3	—

D's Table

To	Cost	Next
A	3	—
B	∞	
C	∞	
D	0	—
E	∞	

C's Table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	∞	
E	4	—

E's Table

To	Cost	Next
A	∞	
B	3	B
C	4	C
D	∞	
E	0	D

Note:

In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change.

Figure 14.5 *Updating in distance vector routing*

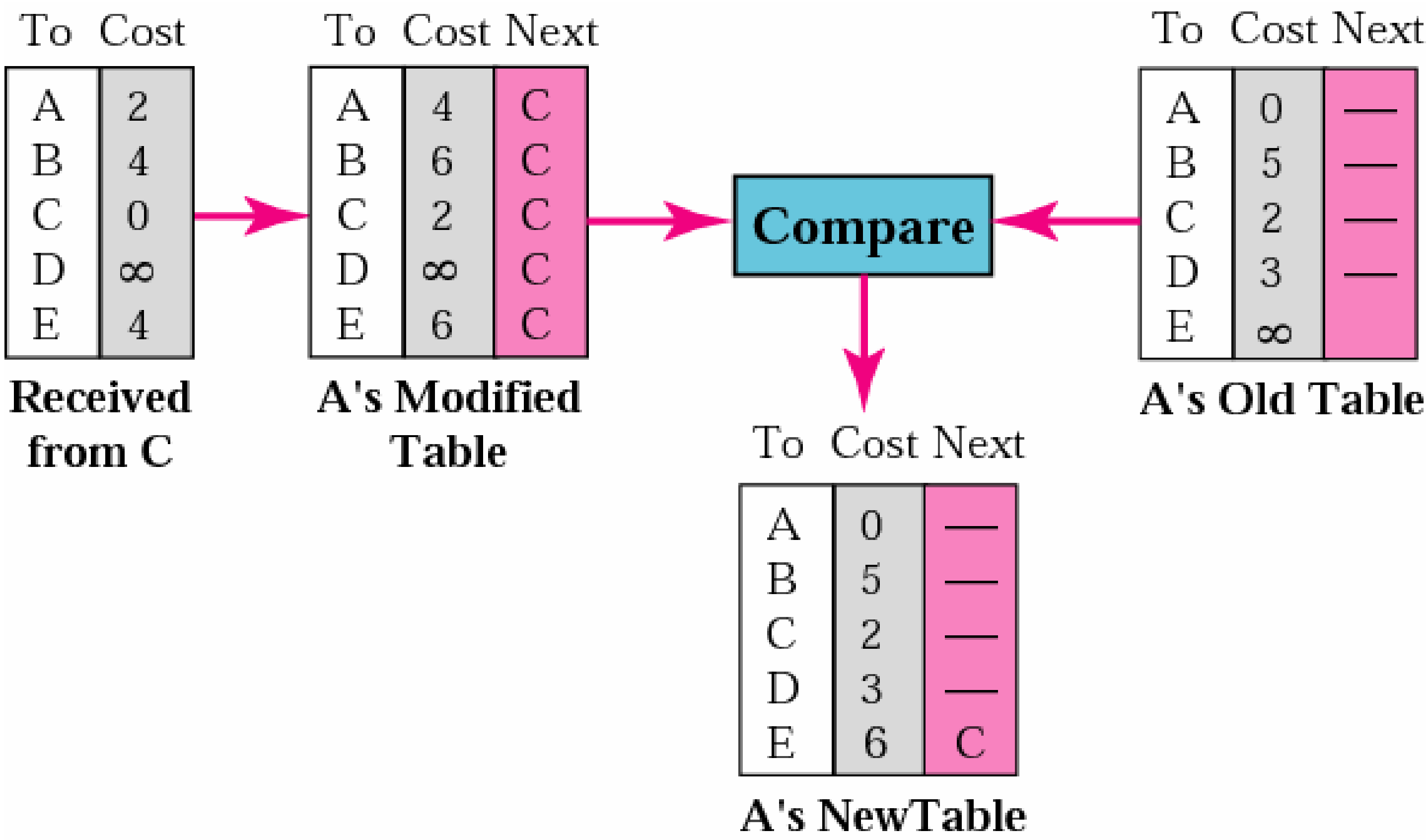


Figure 14.6 *Two-node instability*

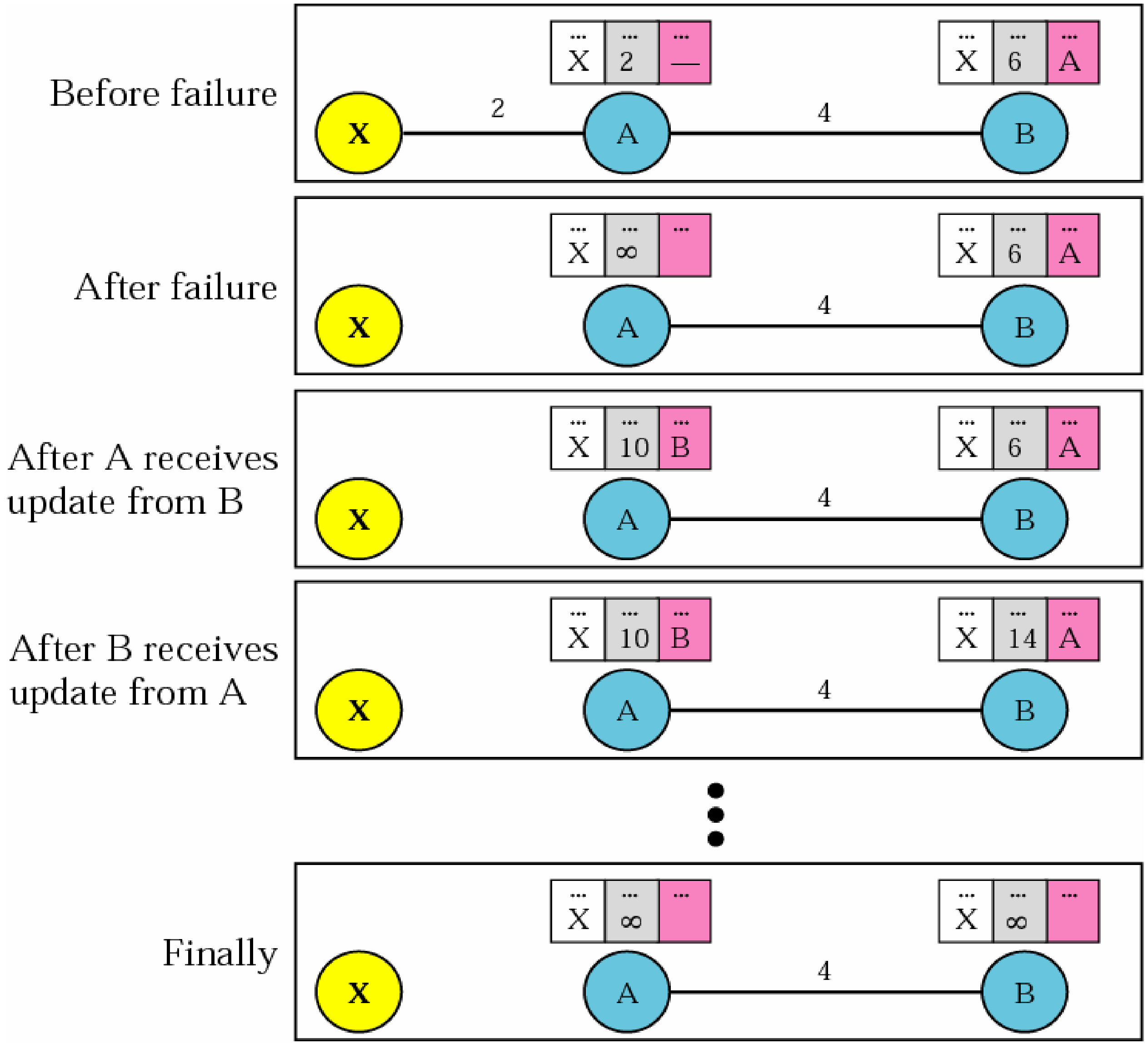


Figure 14.7 *Three-node instability*

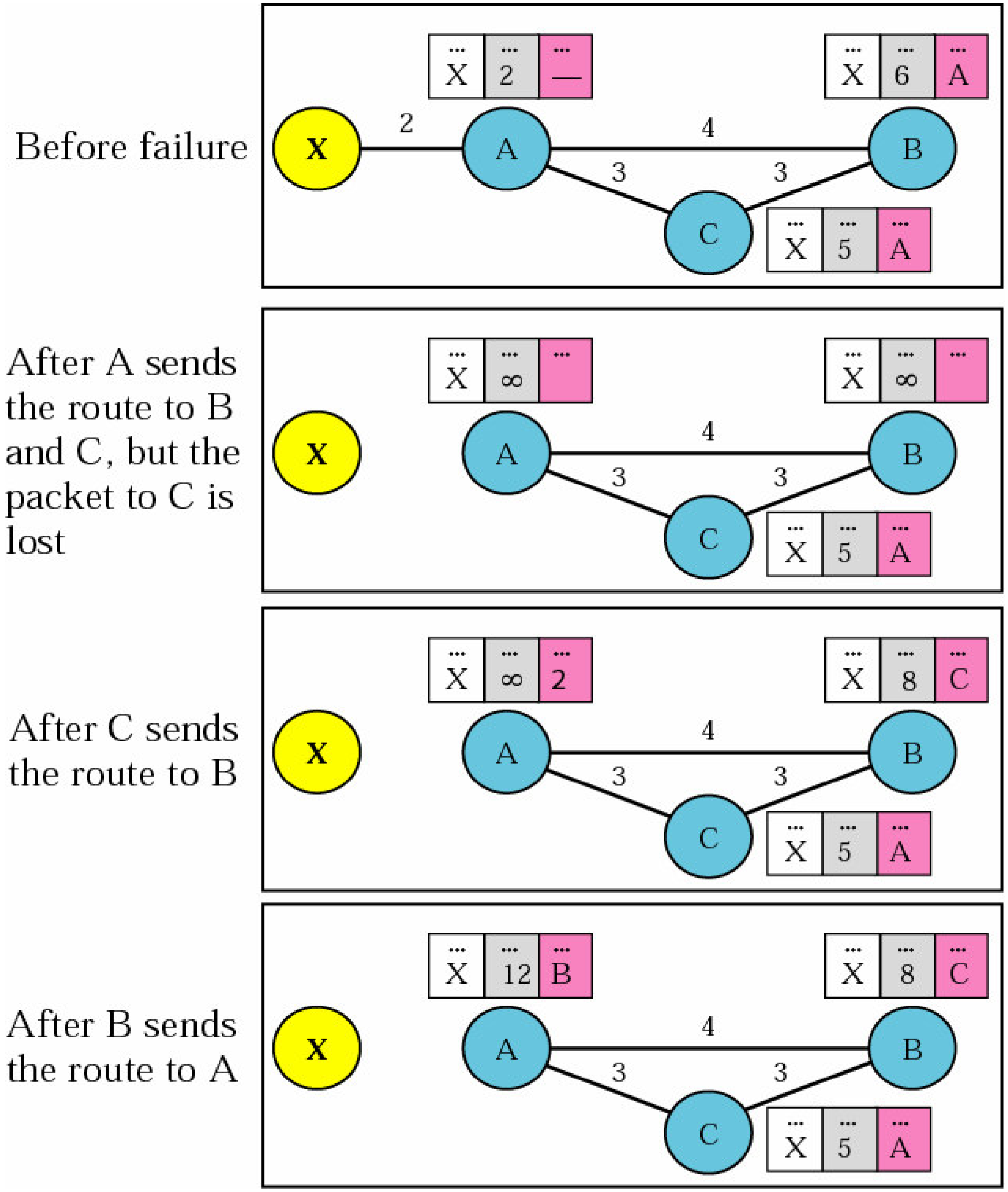
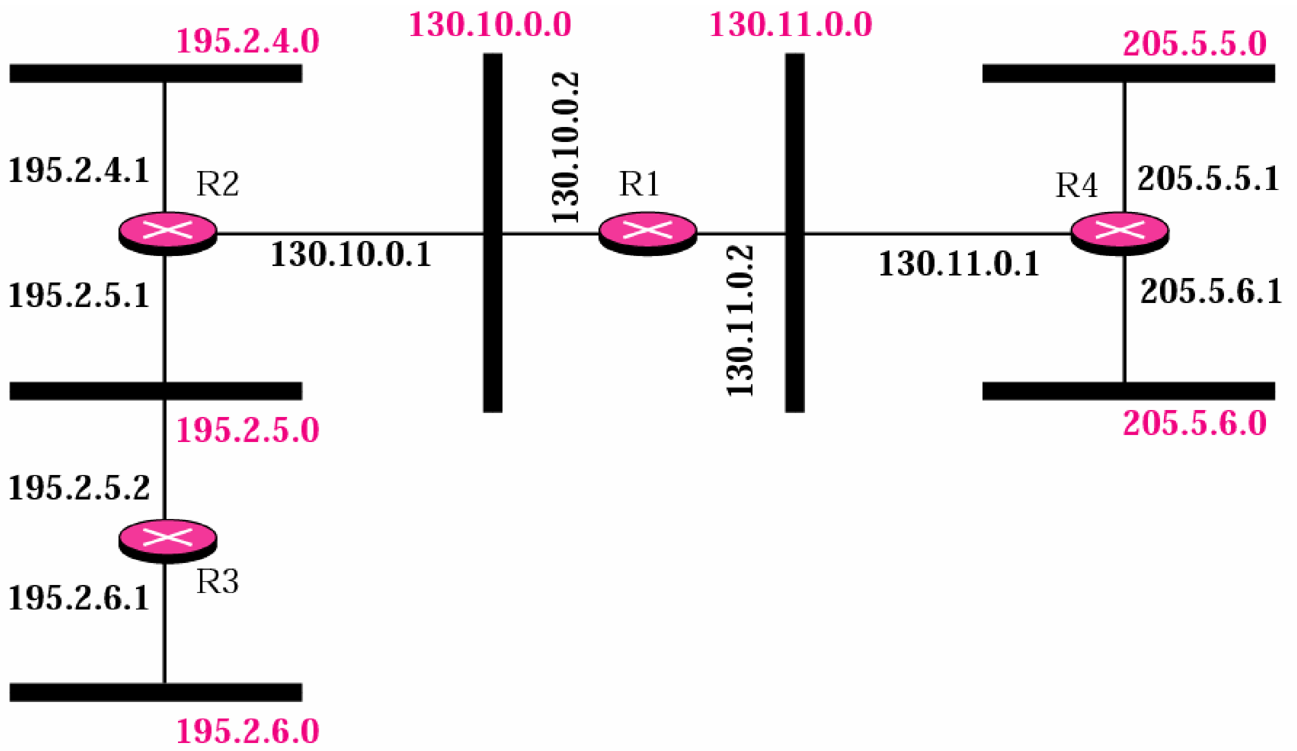


Figure 14.8 *Example of a domain using RIP*



R1 Table

Dest.	Hop	Next
130.10.0.0	1	———
130.11.0.0	1	———
195.2.4.0	2	130.10.0.1
195.2.5.0	2	130.10.0.1
195.2.6.0	3	130.10.0.1
205.5.5.0	2	130.11.0.1
205.5.6.0	2	130.11.0.1

R2 Table

Dest.	Hop	Next
130.10.0.0	1	———
130.11.0.0	2	130.10.0.2
195.2.4.0	1	———
195.2.5.0	1	———
195.2.6.0	2	195.2.5.2
205.5.5.0	3	130.10.0.2
205.5.6.0	3	130.10.0.2

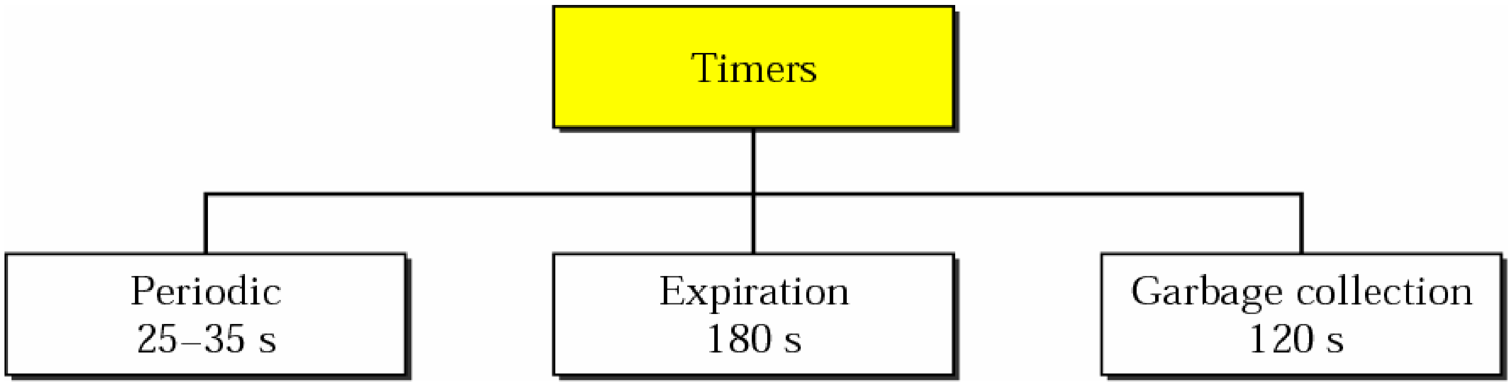
R3 Table

Dest.	Hop	Next
130.10.0.0	2	195.2.5.1
130.11.0.0	3	195.2.5.1
195.2.4.0	2	195.2.5.1
195.2.5.0	1	———
195.2.6.0	1	———
205.5.5.0	4	195.2.5.1
205.5.6.0	4	195.2.5.1

R4 Table

Dest.	Hop	Next
130.10.0.0	2	130.11.0.2
130.11.0.0	1	———
195.2.4.0	3	130.11.0.2
195.2.5.0	3	130.11.0.2
195.2.6.0	4	130.11.0.2
205.5.5.0	1	———
205.5.6.0	1	———

Figure 14.12 *RIP timers*



Example 2

A routing table has 20 entries. It does not receive information about five routes for 200 s. How many timers are running at this time?

Solution

The 21 timers are listed below:

Periodic timer: 1

Expiration timer: 20 – 5 = 15

Garbage collection timer: 5

14.4 LINK STATE ROUTING

In link state routing, if each node in the domain has the entire topology of the domain, the node can use Dijkstra's algorithm to build a routing table.

The topics discussed in this section include:

Building Routing Tables

Figure 14.15 *Concept of link state routing*

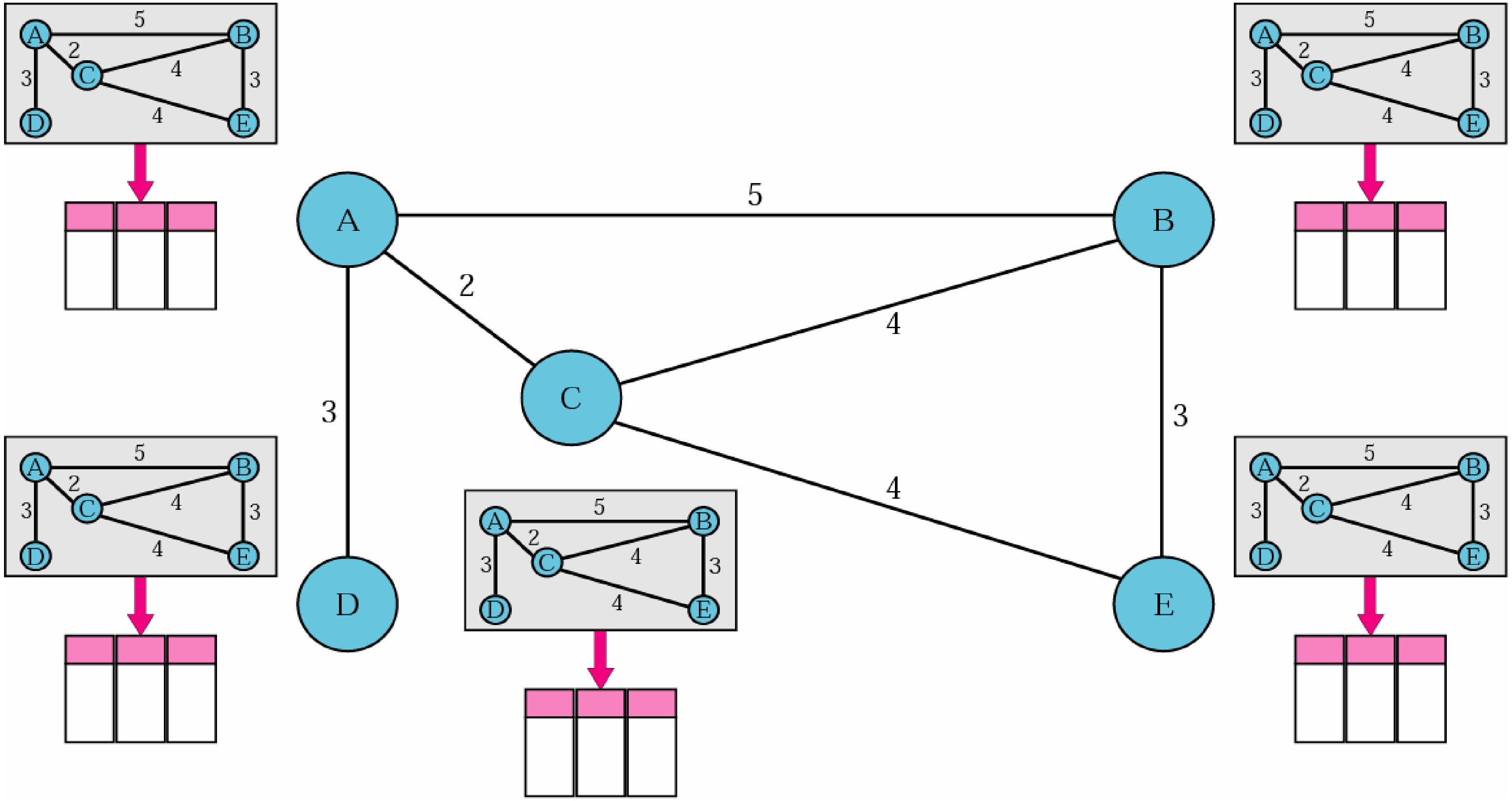


Figure 14.18 *Example of formation of shortest path tree*

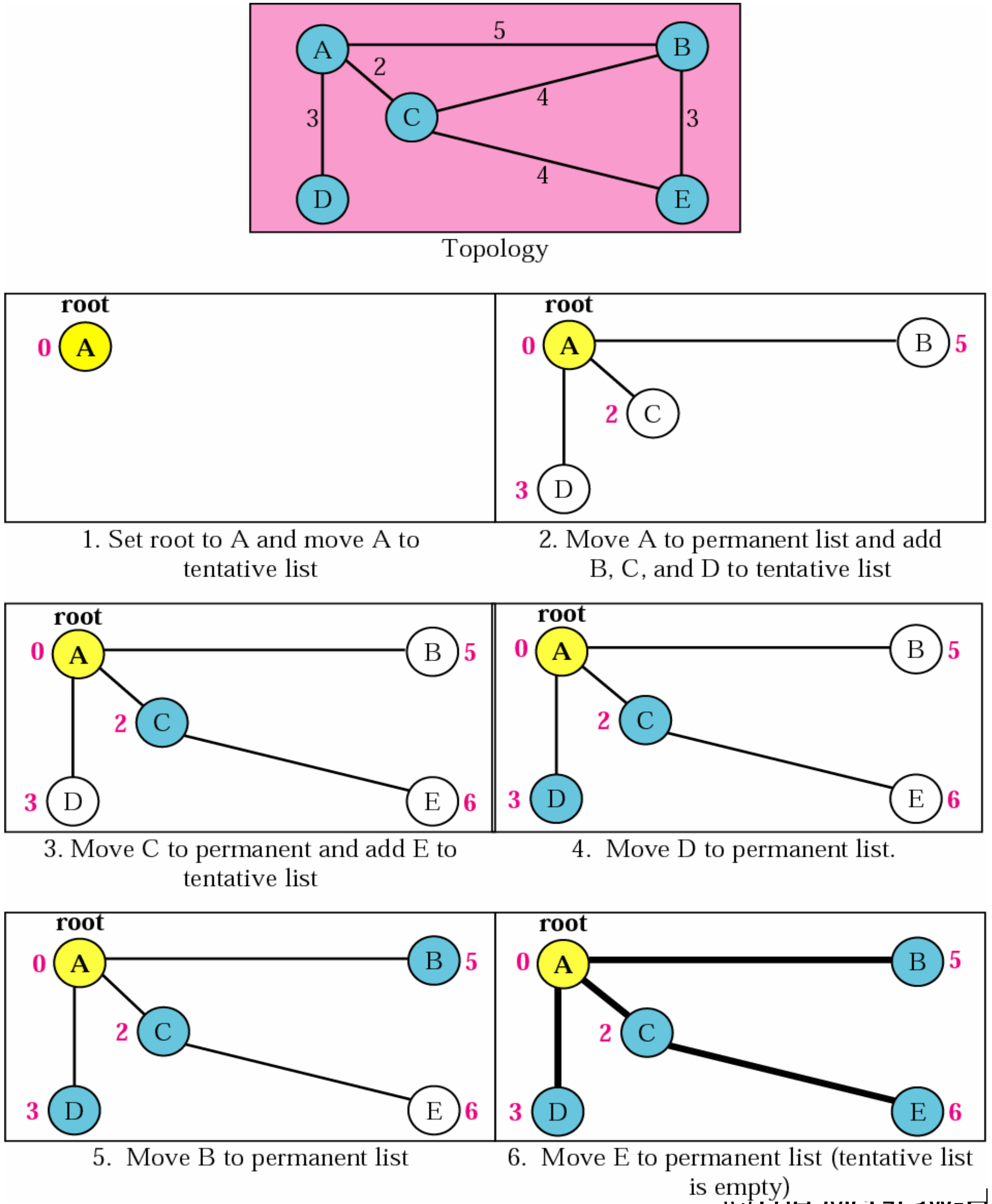


Table 14.1 ***Routing table for node A***

Node	*Cost*	*Next Router*
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

Open Shortest Path First (RFC2328)

- IGP of Internet
- replaced Routing Information Protocol (RIP)
- uses Link State Routing Algorithm
  - each router keeps list of state of local links to network
  - transmits update state info
  - little traffic as messages are small and not sent often
- uses least cost based on user cost metric
- topology stored as directed graph
  - vertices or nodes (router, transit or stub network)
  - edges (between routers or router to network)

Figure 21.8 Types of links

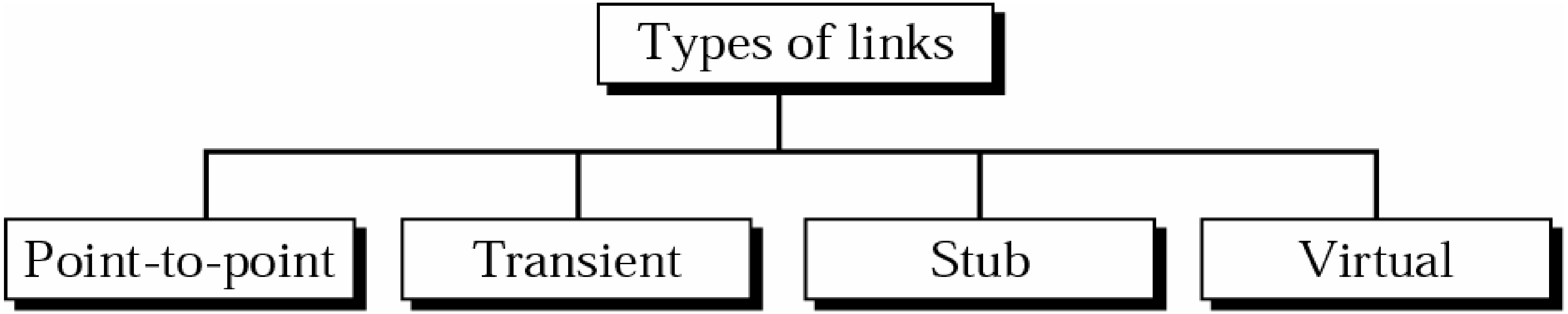


Figure 21.9 Point-to-point link

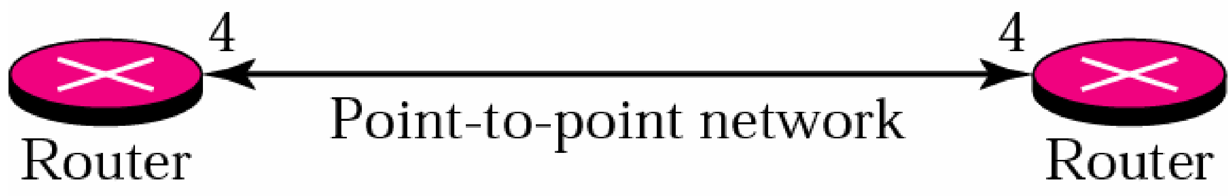


Figure 21.10 Transient link

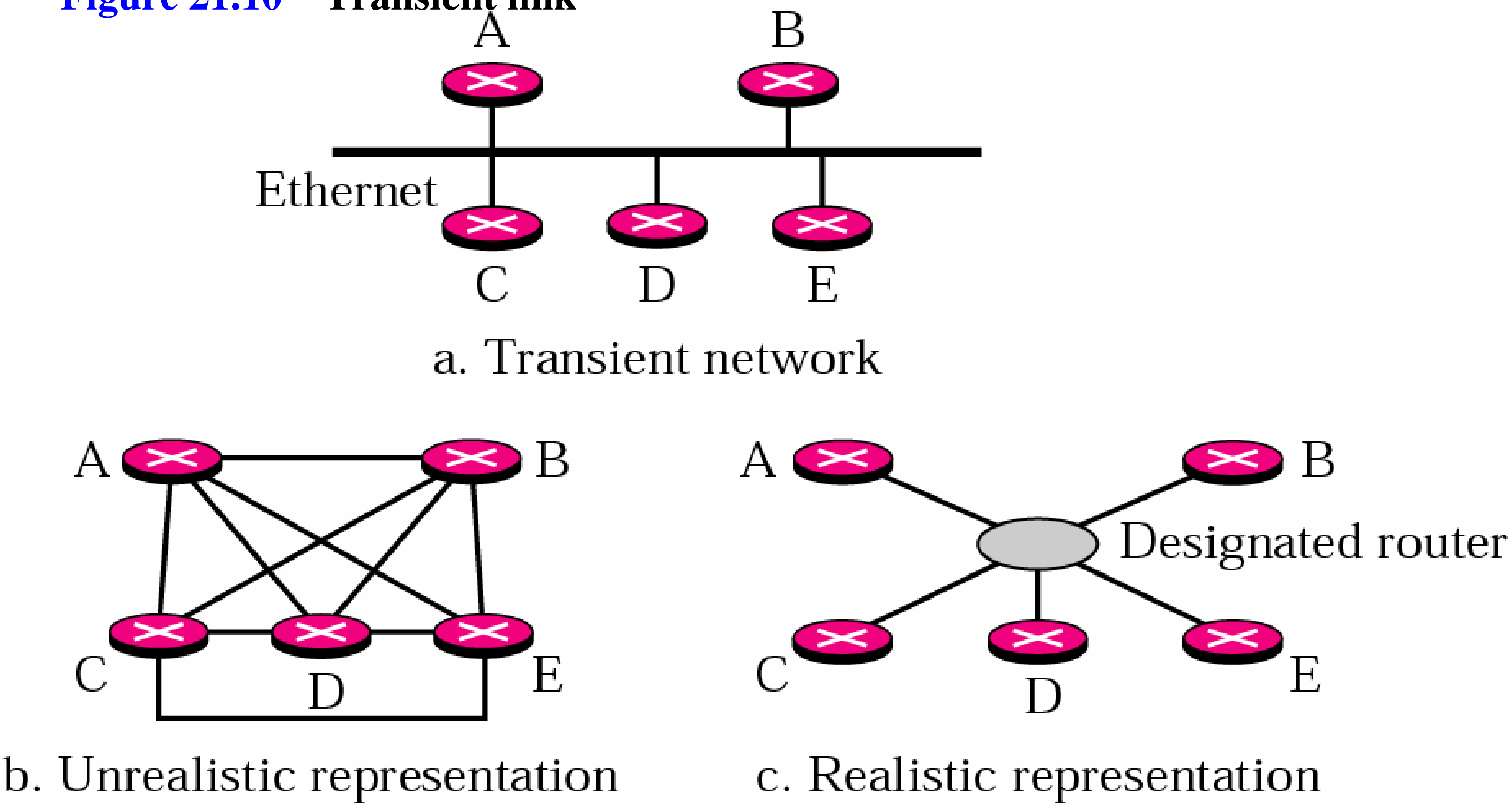
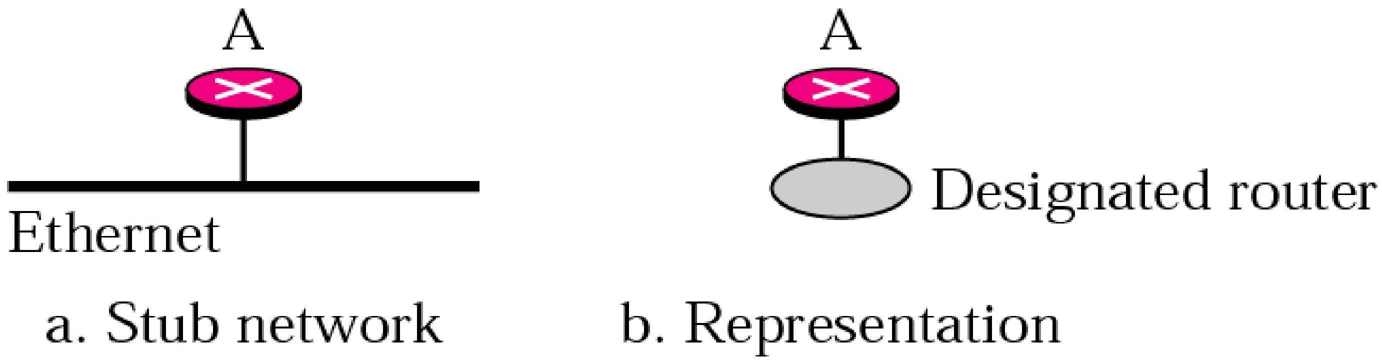


Figure 21.11 Stub link



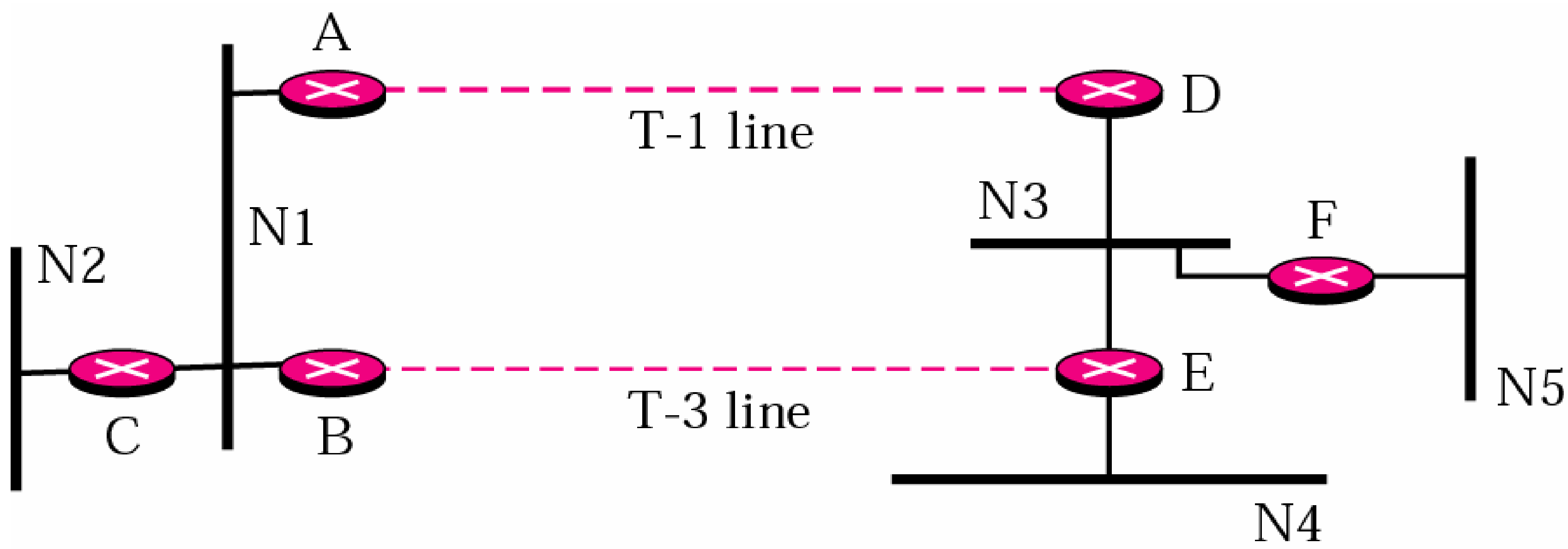
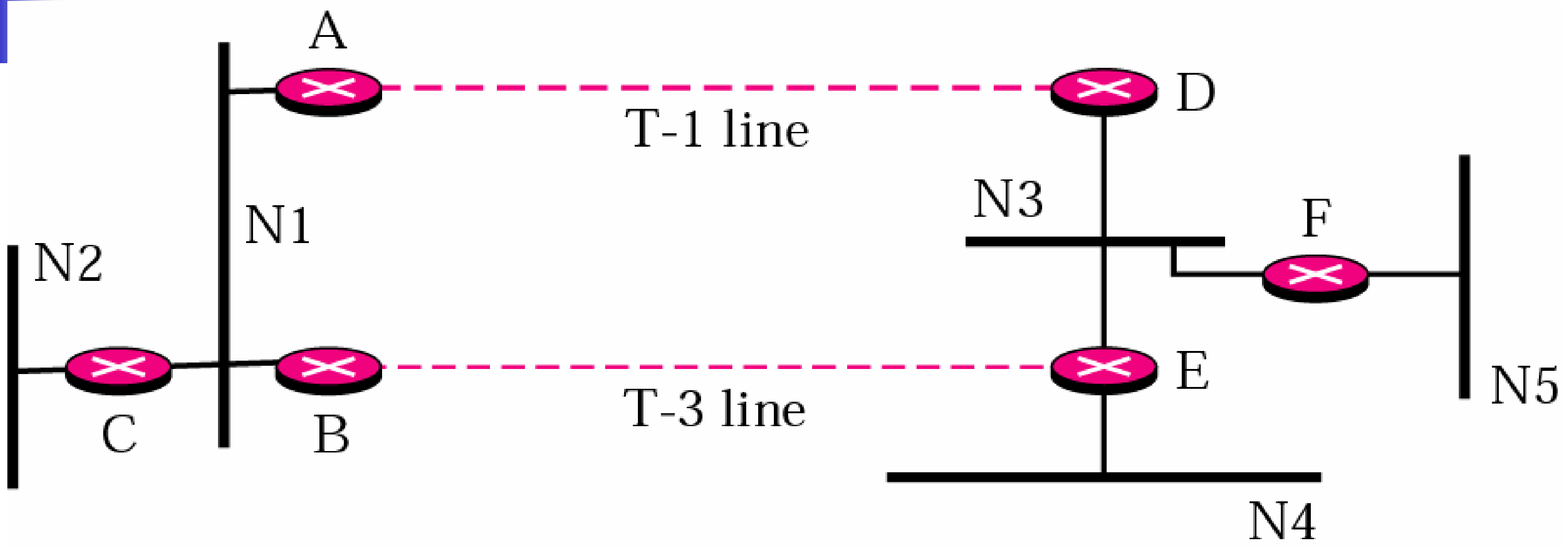


Figure 21.13 Graphical representation of an internet



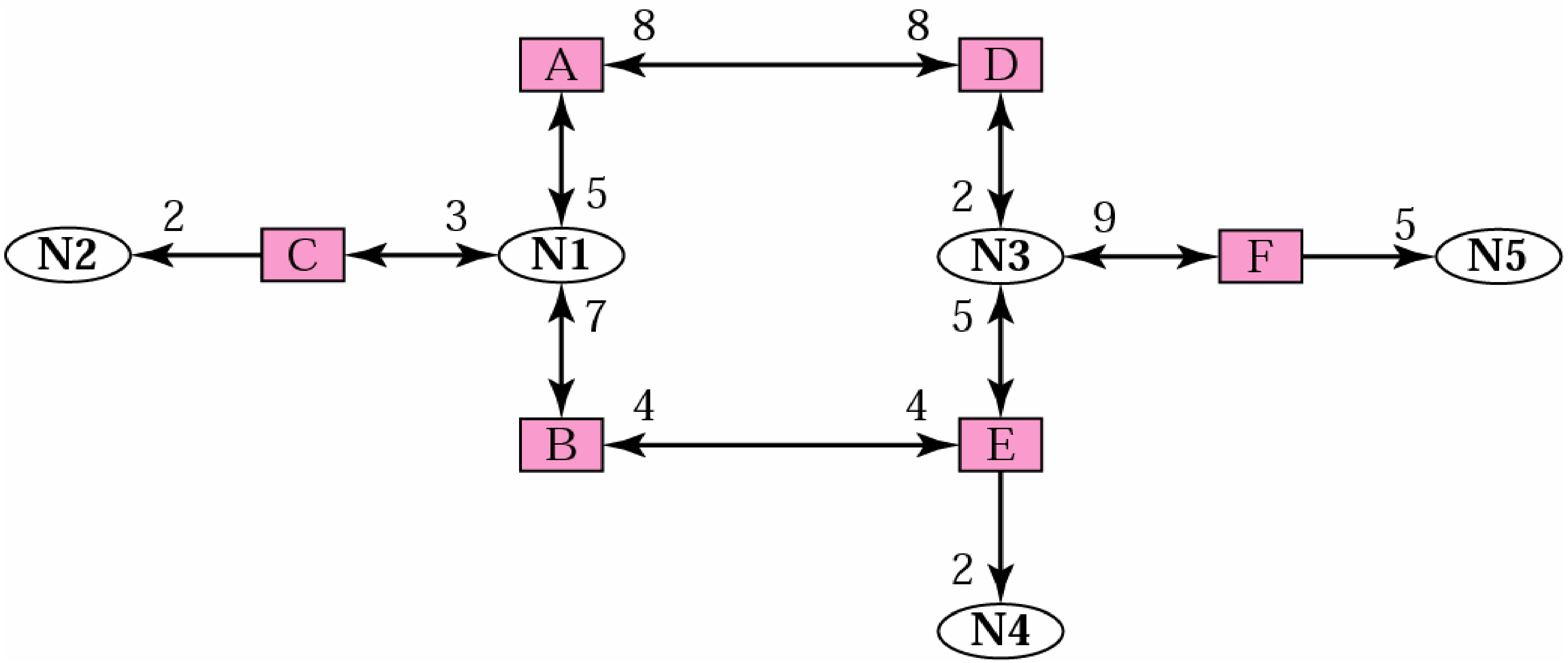


Figure 21.14 Types of LSAs

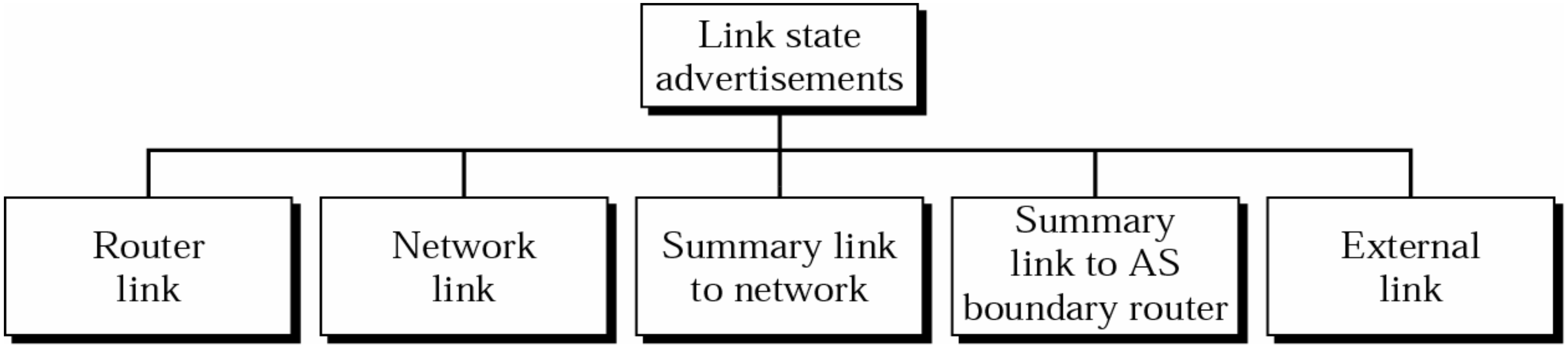


Figure 21.15 **Router link**

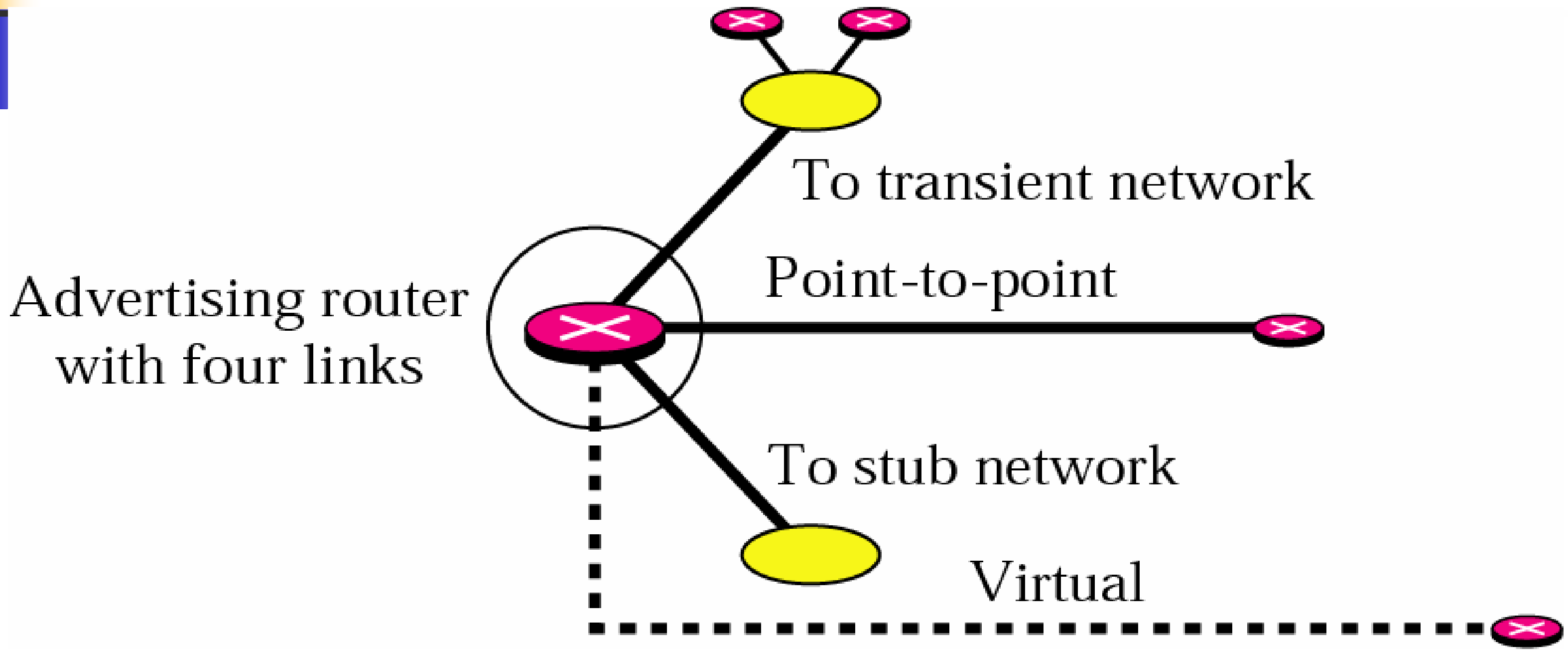


Figure 21.16 **Network link**

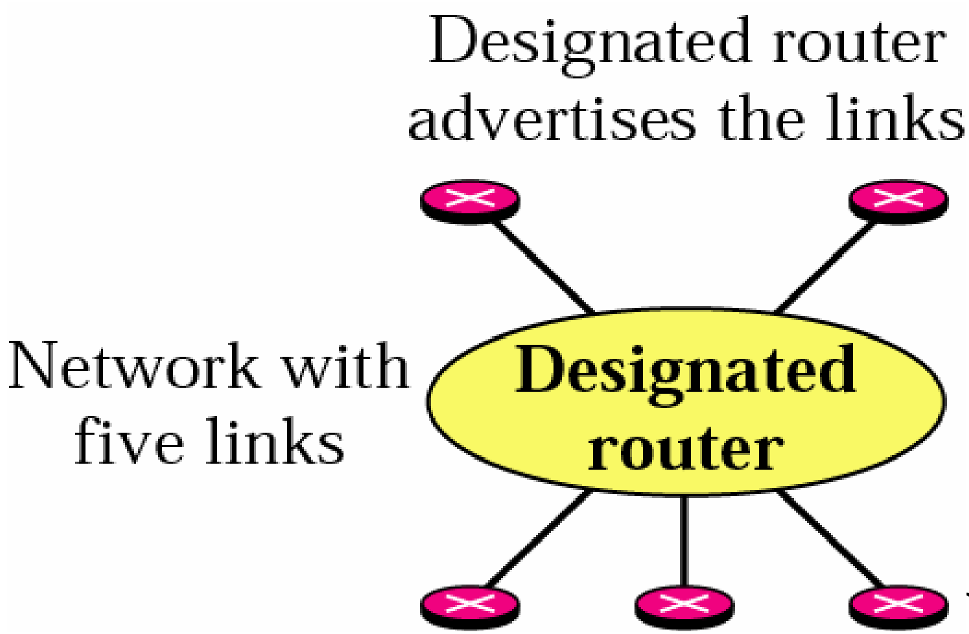


Figure 21.17 Summary link to network

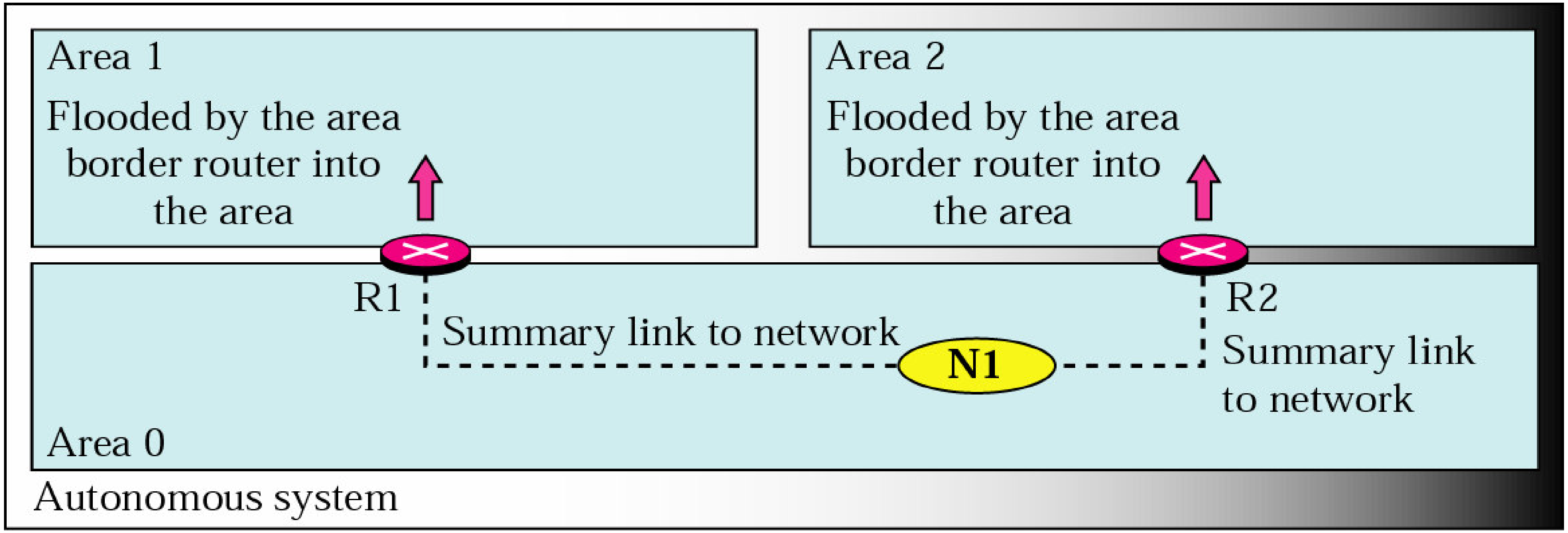


Figure 21.18 Summary link to AS boundary router

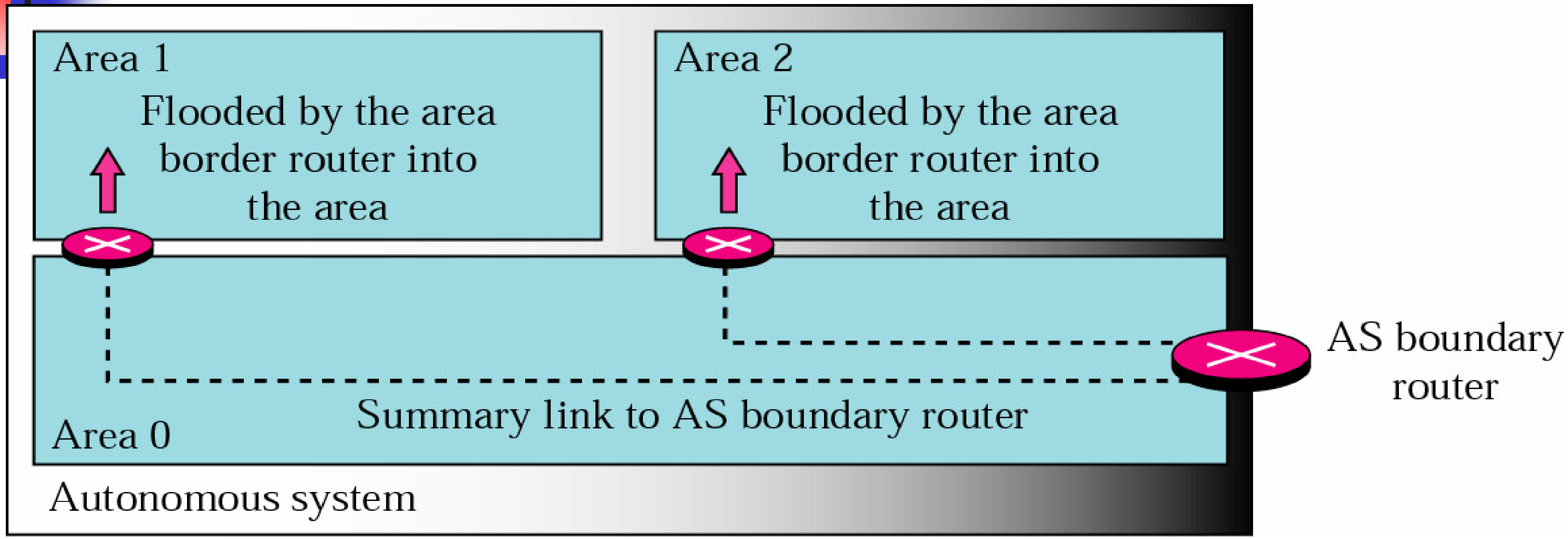
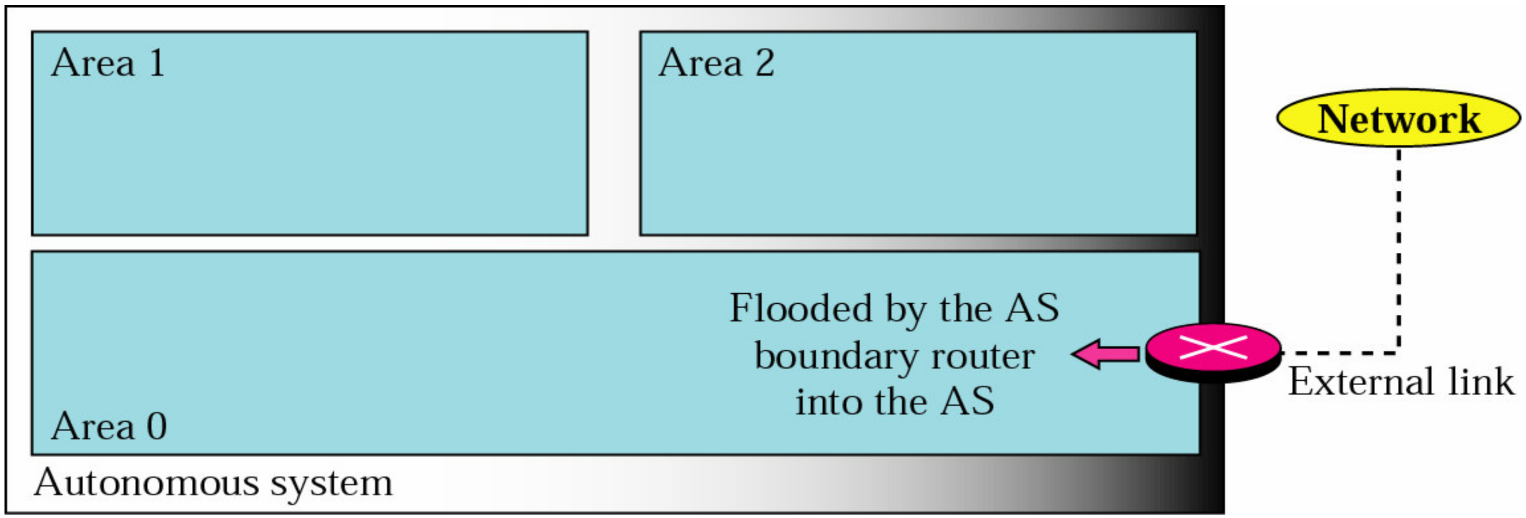
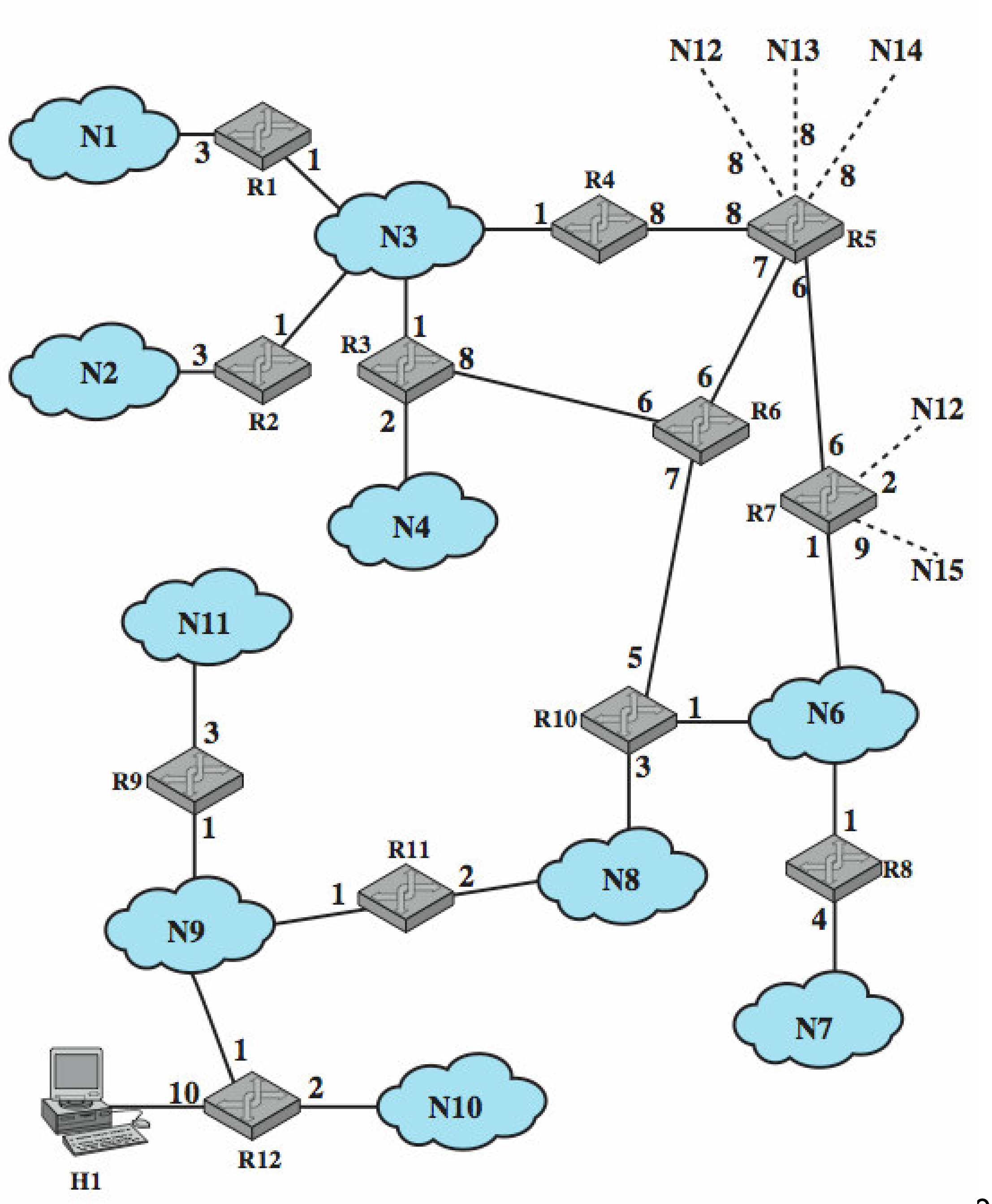


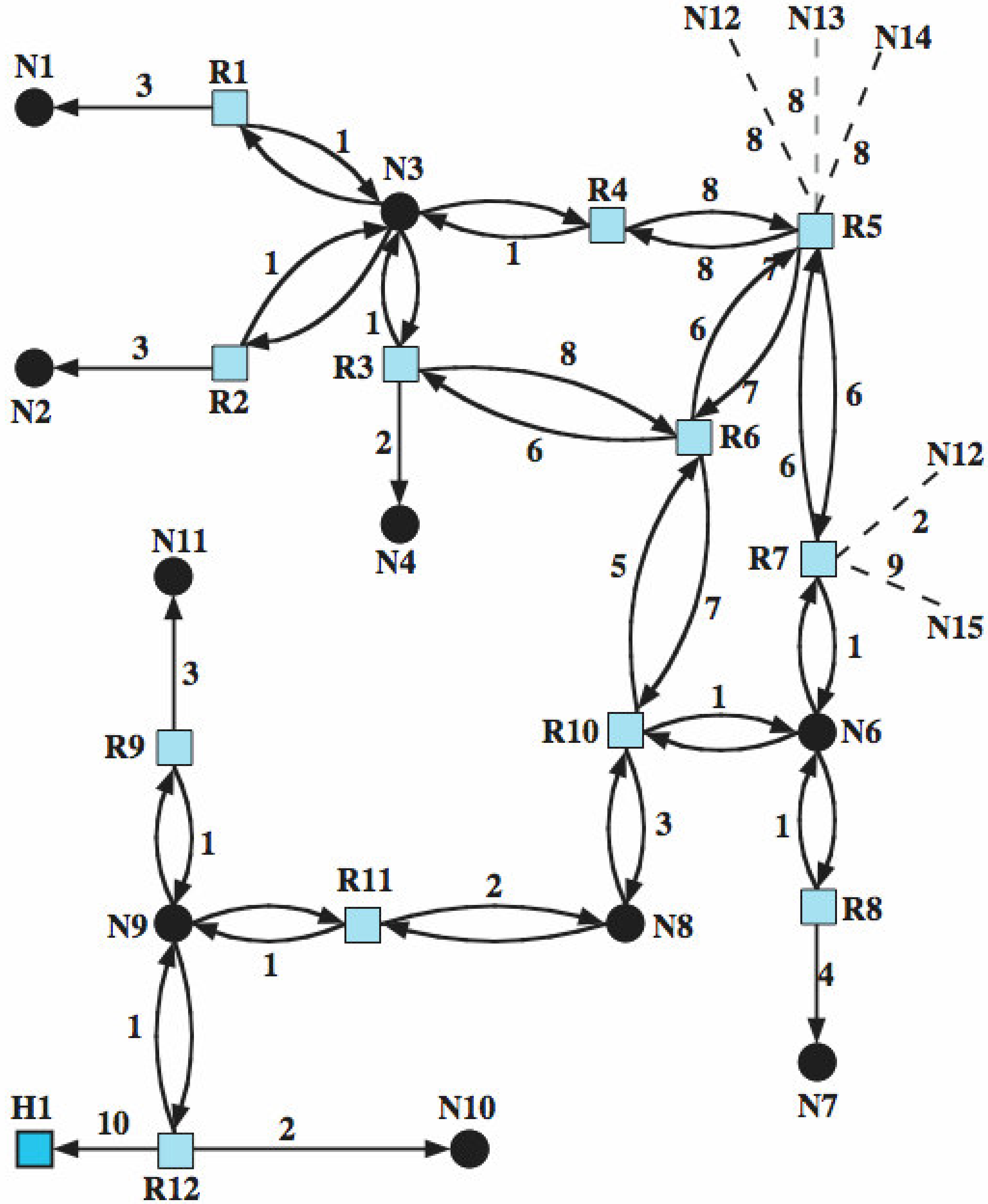
Figure 21.19 External link



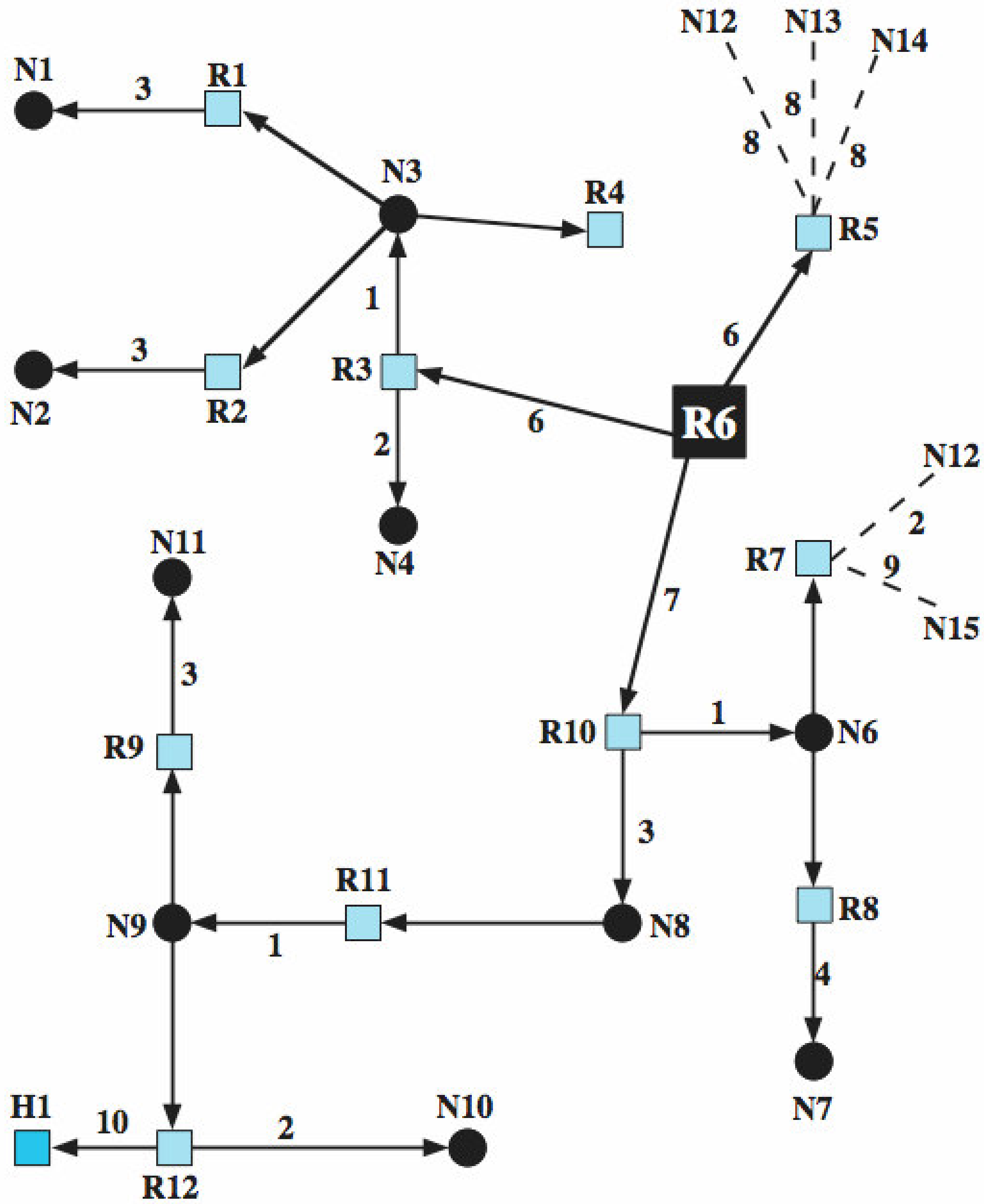
Example OSPF AS



Directed Graph of AS



SPF Tree for Router 6



14.7 BGP

Border Gateway Protocol (BGP) is an interdomain routing protocol using path vector routing. It first appeared in 1989 and has gone through four versions.

The topics discussed in this section include:

Types of Autonomous Systems
Path Attributes
BGP Sessions
External and Internal BGP
Types of Packets
Packet Format
Encapsulation

What Exterior Routing Protocols are not

- link-state and distance-vector not effective for exterior router protocol
- distance-vector
  - assumes routers share common distance metric
  - but different ASs may have different priorities & needs
  - but have no info on AS's visited along route
- link-state
  - different ASs may use different metrics and have different restrictions
  - flooding of link state information to all routers unmanageable

Exterior Router Protocols – Path-vector

- alternative **path-vector** routing protocol
  - provides info about which networks can be reached by a given router and ASs crossed to get there
  - does not include distance or cost estimate
  - hence dispenses with concept of routing metrics
- have list of all ASs visited on a route
- enables router to perform policy routing
  - eg. avoid path to avoid transiting particular AS
  - eg. link speed, capacity, tendency to become congested, and overall quality of operation, security
  - eg. minimizing number of transit ASs

Table 21.3 Path vector routing table

Network	Next Router	Path
N01	R01	AS14, AS23, AS67
N02	R05	AS22, AS67, AS05, AS89
N03	R06	AS67, AS89, AS09, AS34
N04	R12	AS62, AS02, AS09

Figure 21.21 Path vector messages

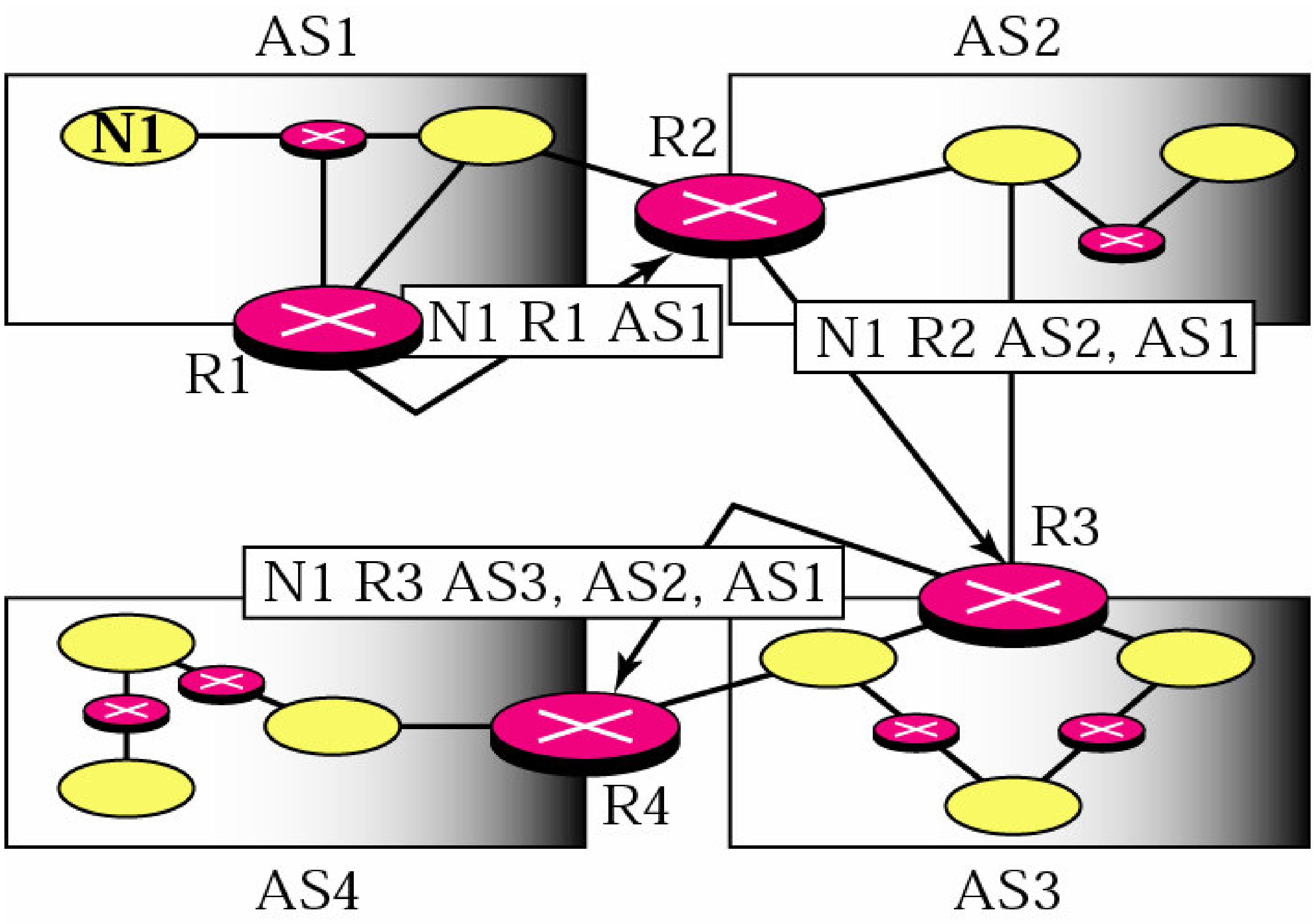
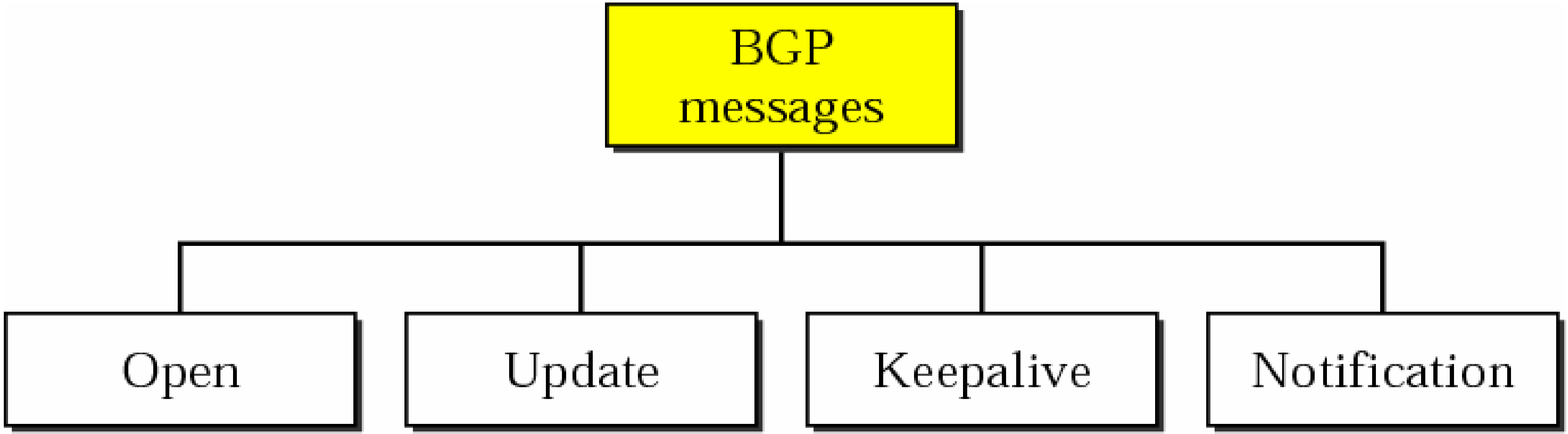


Figure 14.51 *Types of BGP messages*



Message Types - Open & KeepAlive

- router makes TCP connection to neighbor
- Open message
  - sent by connection initiator
  - includes proposed hold time
  - receiver uses minimum of own/sent hold time
  - max time between Keepalive and/or Update
- Keep Alive message
  - To tell other routers that this router is still here

Message Types - Update

- Update message conveys two info types:
  - Info about single routes through internet
  - List of routes being withdrawn
- info on a route uses 3 fields:
  - Network Layer Reachability Information (NLRI)
  - Total Path Attributes Length
  - Path Attributes
- withdraw route identified by dest IP address

Message Types - Update

- Origin - IGP or EGP
- AS_Path - list of AS traversed
- Next_hop - IP address of border router
- Atomic_Aggregate, Aggregator - implement route aggregation to reduce amount of info

Notification Message

- sent when some error condition detected:
- Message header error
- Open message error
- Update message error
- Hold time expired
- Finite state machine error
- Cease

BGP Routing Information Exchange

- within AS a router builds topology picture using IGP
- router issues Update message to other routers outside AS using BGP
- these routers exchange info with other routers in other AS
  - AS_Path field used to prevent loops
- routers must then decide best routes

Figure 14.48 *Initial routing tables in path vector routing*

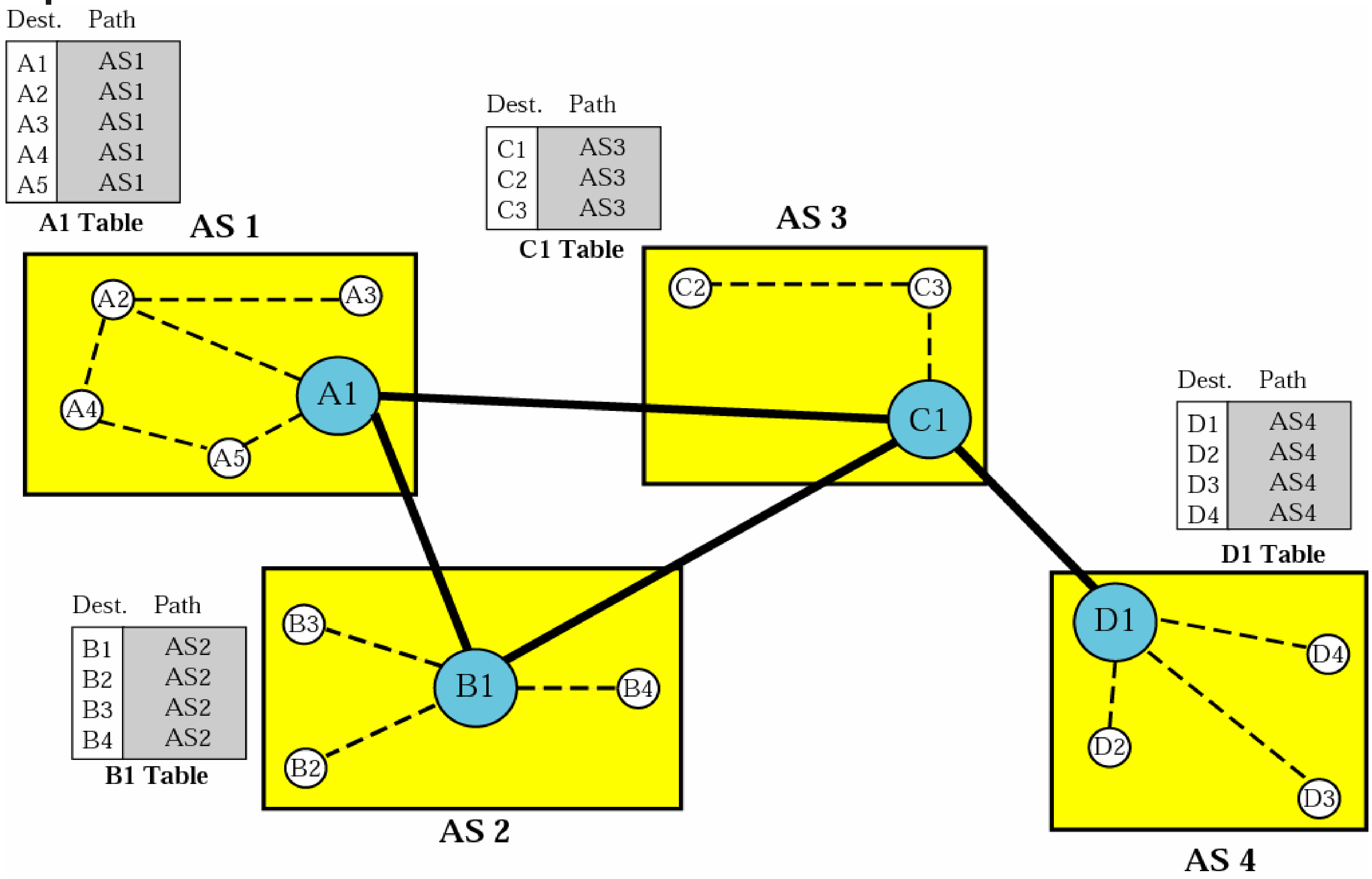


Figure 14.49 *Stabilized tables for four autonomous systems*

A1 Table

Dest.	Path
A1	AS1
...	
A5	AS1
B1	AS1-AS2
...	...
B4	AS1-AS2
C1	AS1-AS3
...	...
C3	AS1-AS3
D1	AS1-AS2-AS4
...	...
D4	AS1-AS2-AS4

B1 Table

Dest.	Path
A1	AS2-AS1
...	
A5	AS2-AS1
B1	AS2
...	...
B4	AS2
C1	AS2-AS3
...	...
C3	AS2-AS3
D1	AS2-AS3-AS4
...	...
D4	AS2-AS3-AS4

C1 Table

Dest.	Path
A1	AS3-AS1
...	
A5	AS3-AS1
B1	AS3-AS2
...	...
B4	AS3-AS2
C1	AS3
...	...
C3	AS3
D1	AS3-AS4
...	...
D4	AS3-AS4

D1 Table

Dest.	Path
A1	AS4-AS3-AS1
...	
A5	AS4-AS3-AS1
B1	AS4-AS3-AS2
...	...
B4	AS4-AS3-AS2
C1	AS4-AS3
...	...
C3	AS4-AS3
D1	AS4
...	...
D4	AS4

Chapter 13

Multiple Access

Figure 13.1 Multiple-access protocols

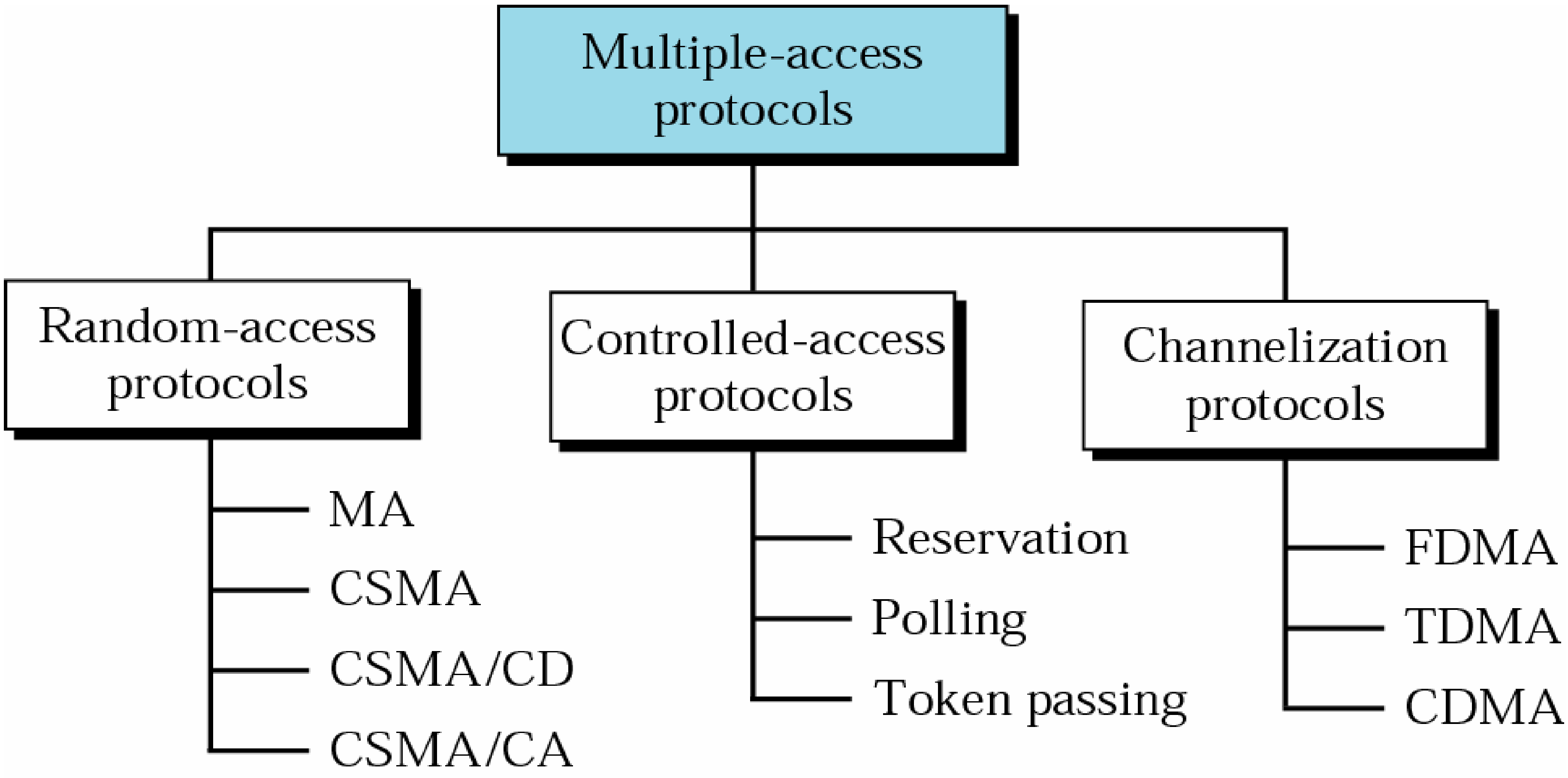


Figure 13.3 ALOHA network

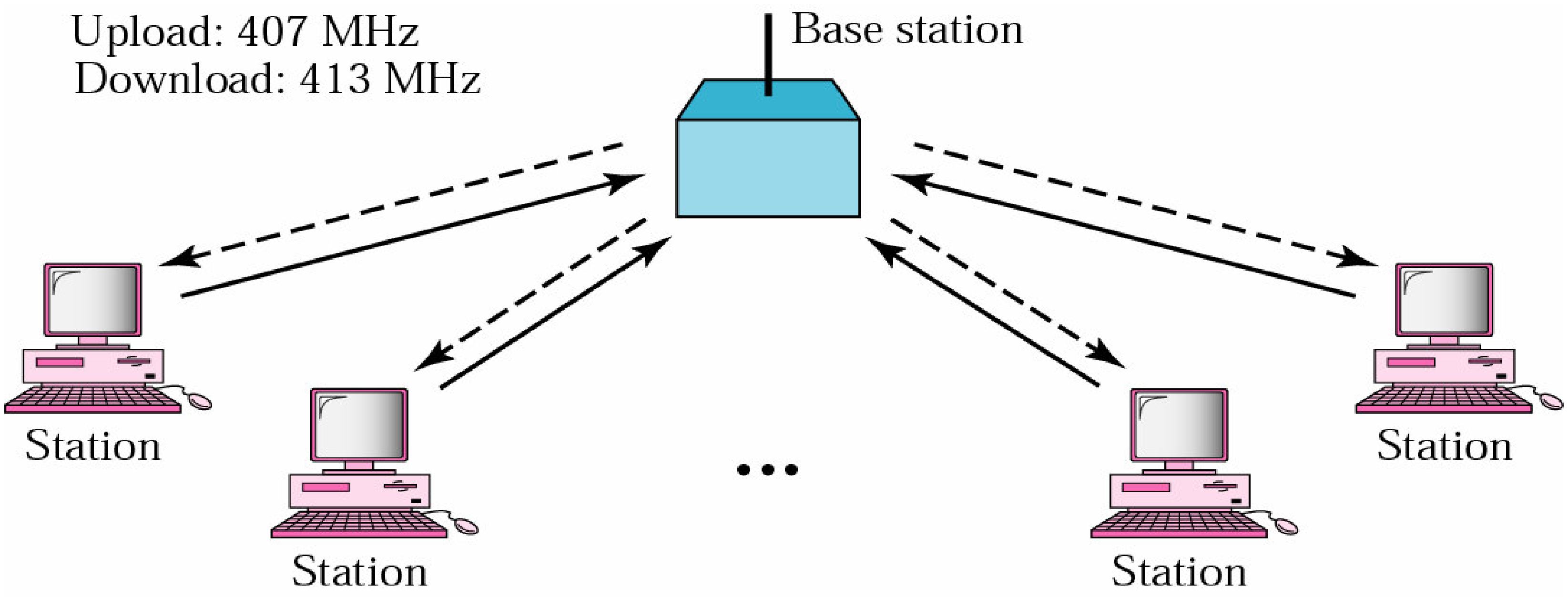


Figure 13.4 Procedure for ALOHA protocol

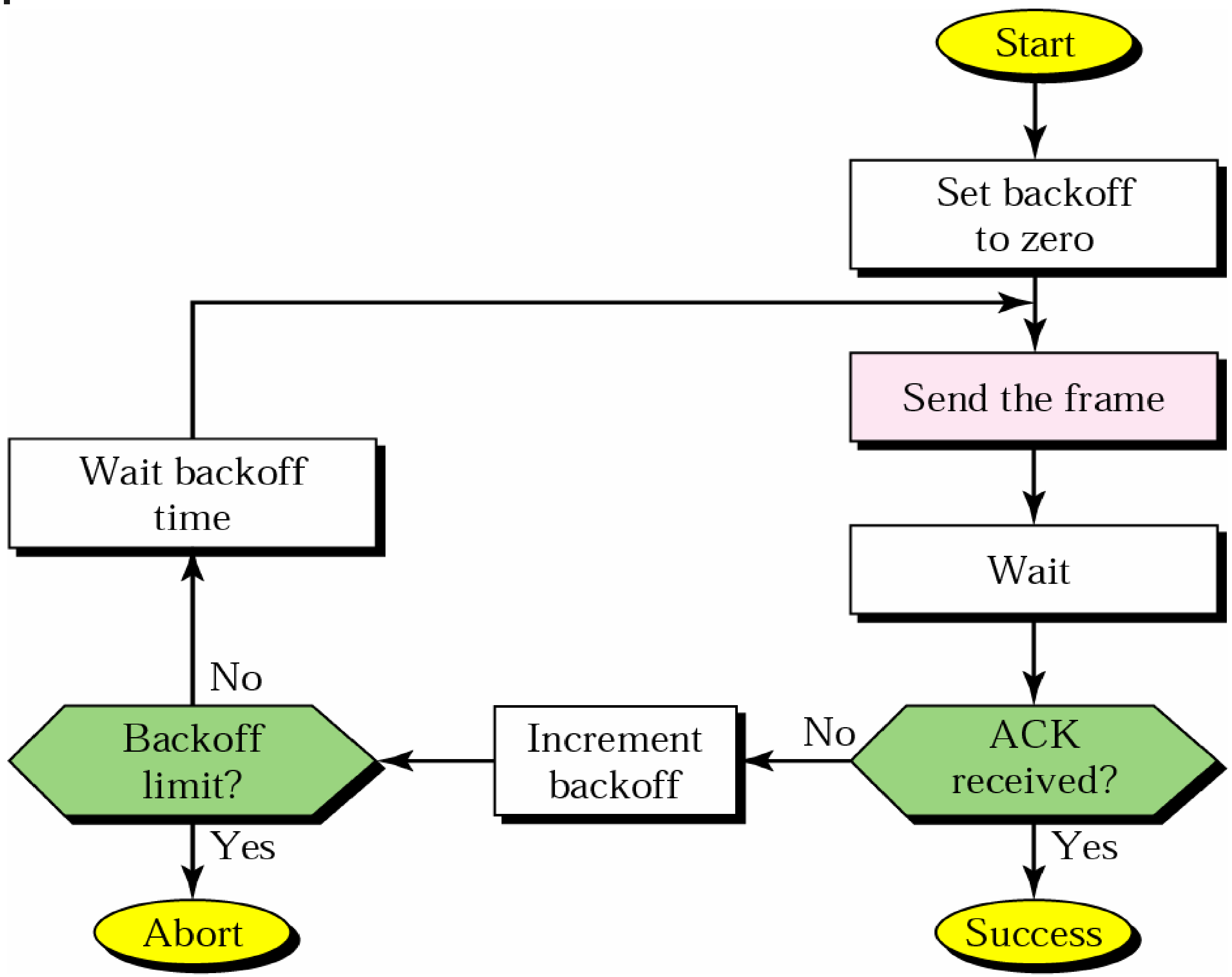


Figure 13.5 Collision in CSMA

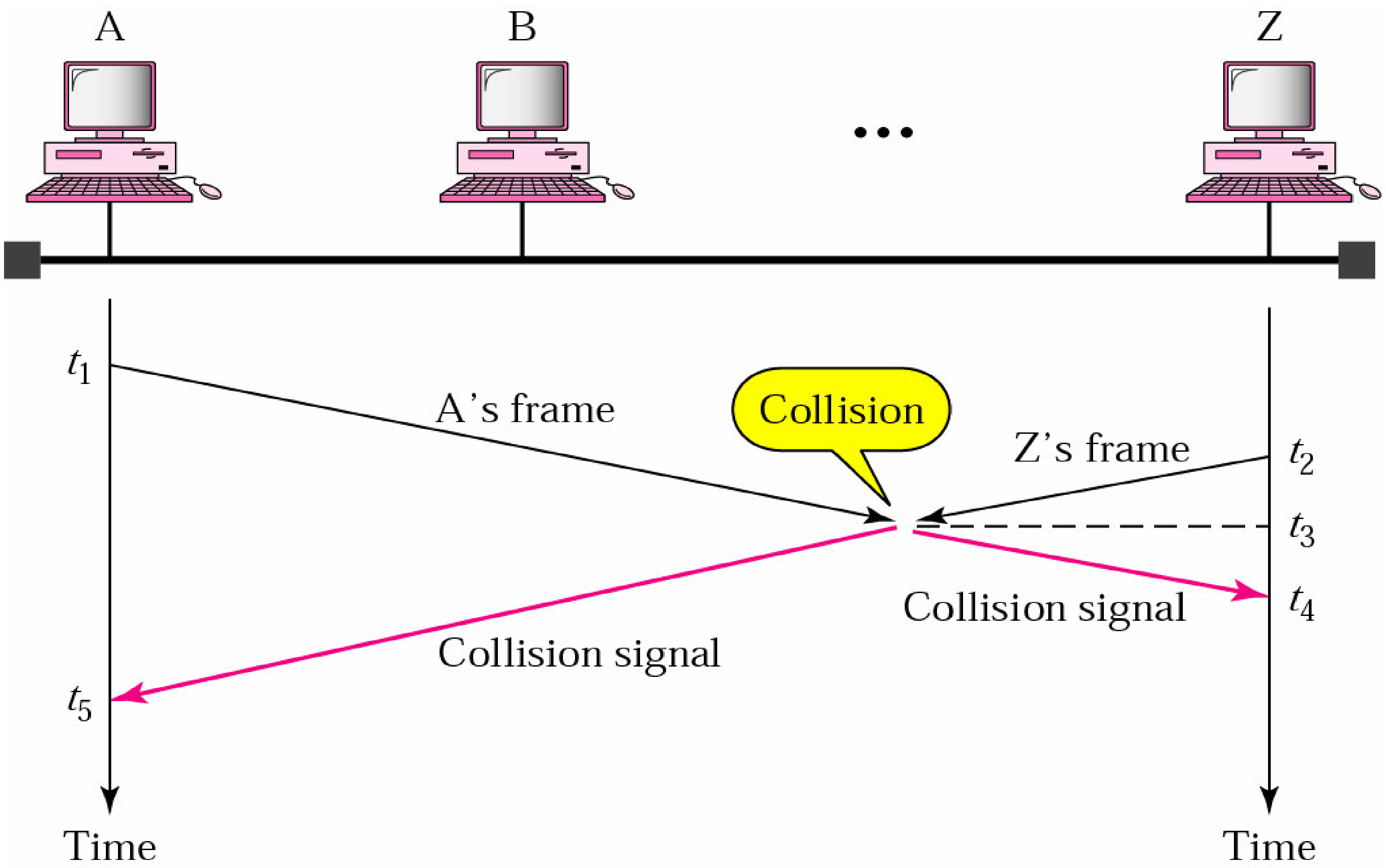
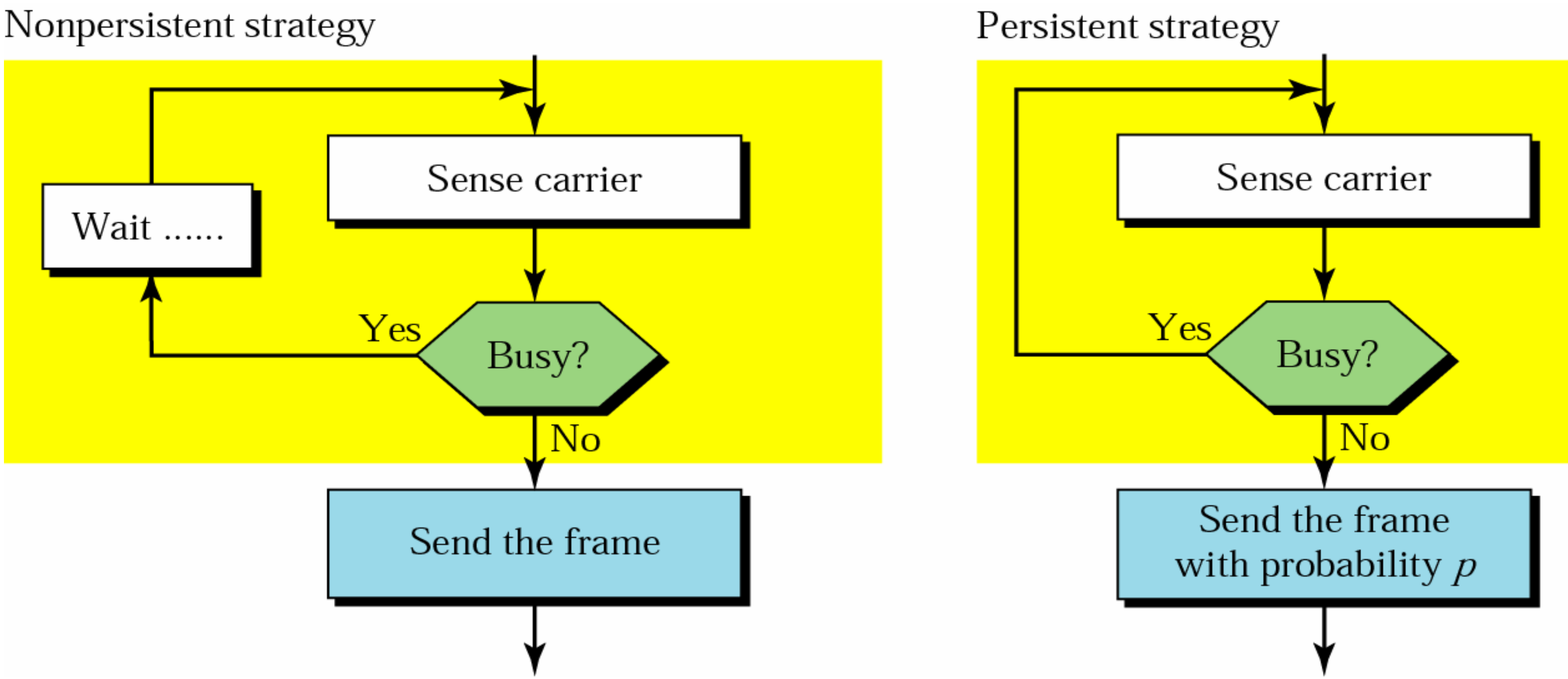


Figure 13.6 Persistence strategies



13.7 CSMA/CD procedure

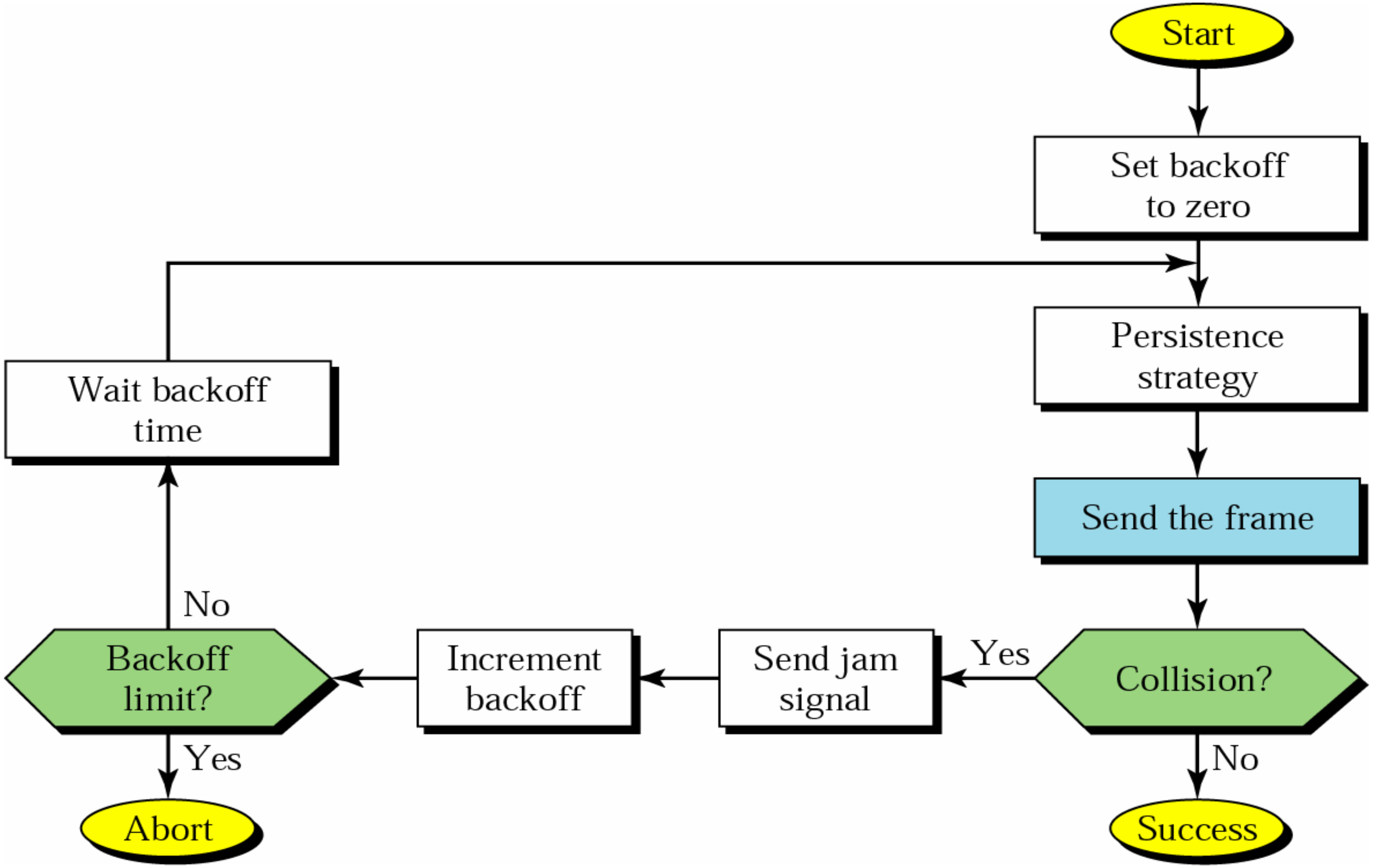
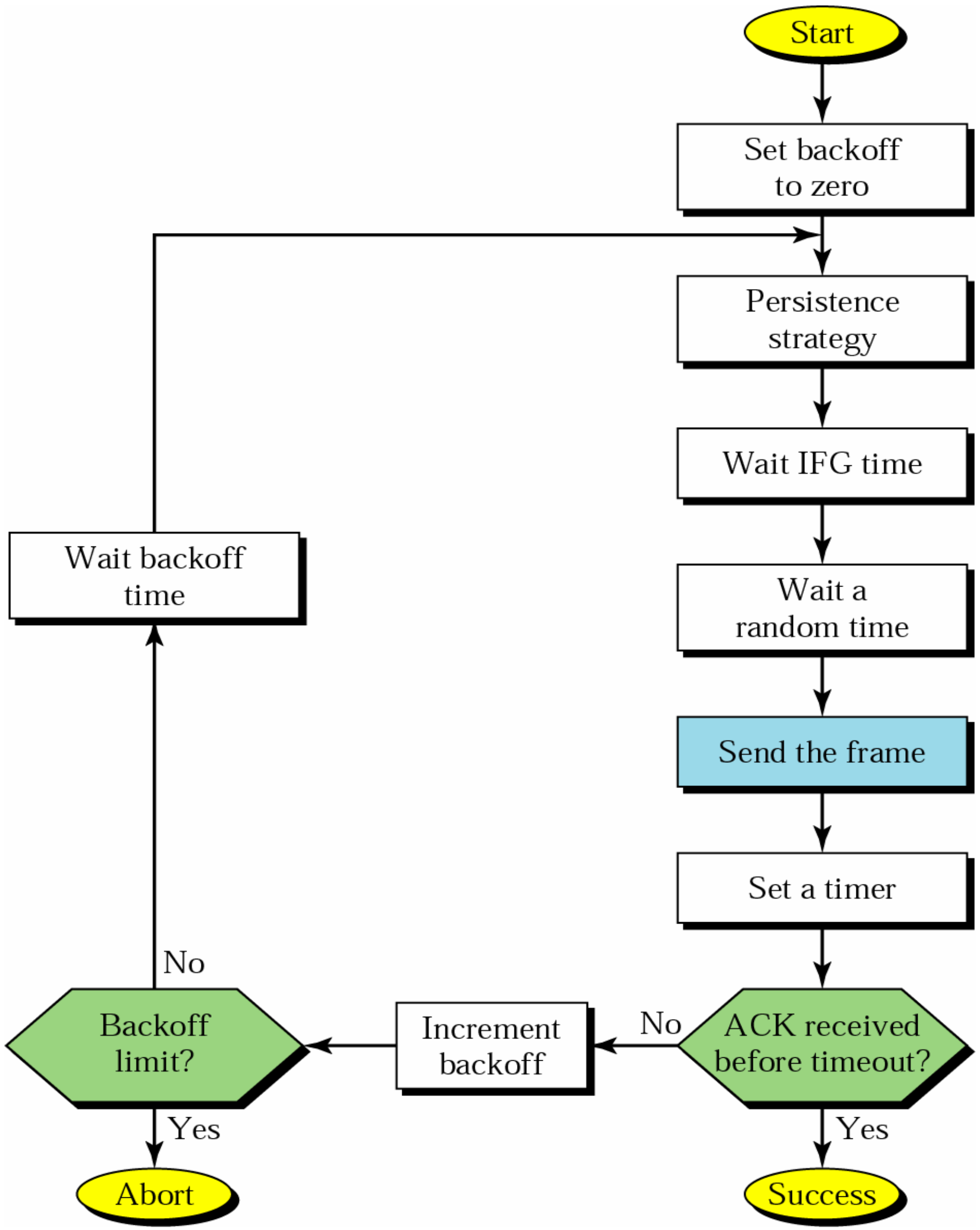


Figure 13.8 CSMA/CA procedure



13.2 Control Access

Reservation

Polling

Token Passing

Figure 13.9 Reservation access method

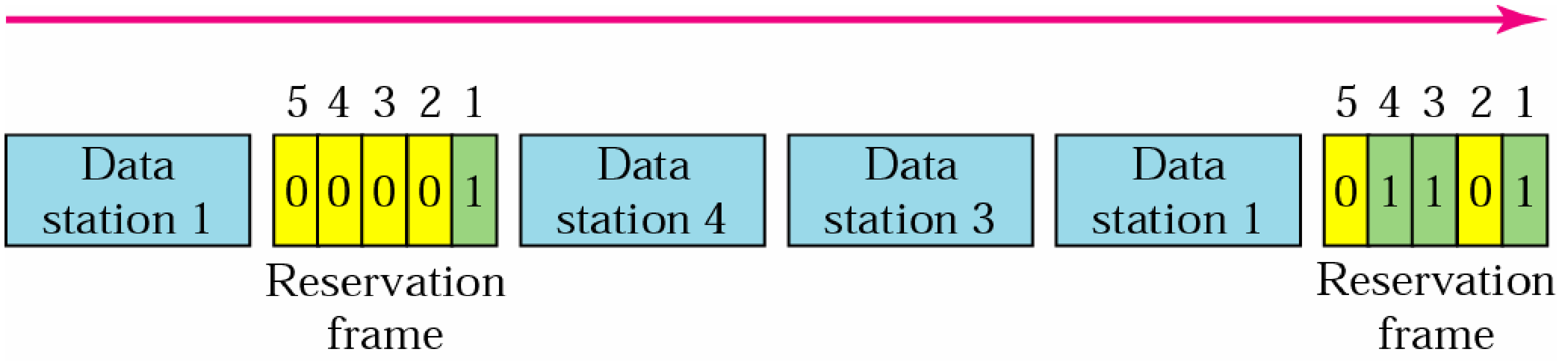


Figure 13.10 Select

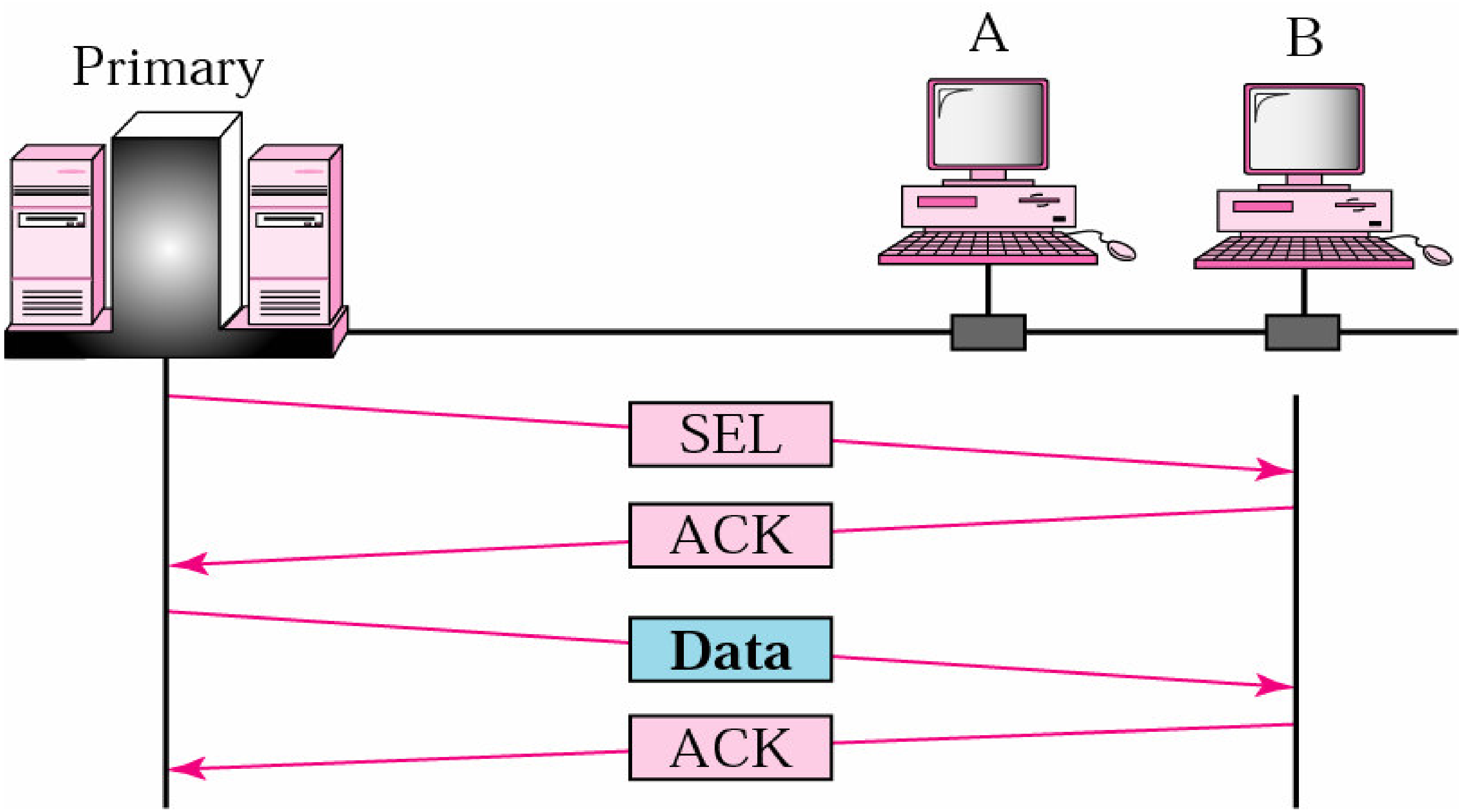


Figure 13.11 Poll

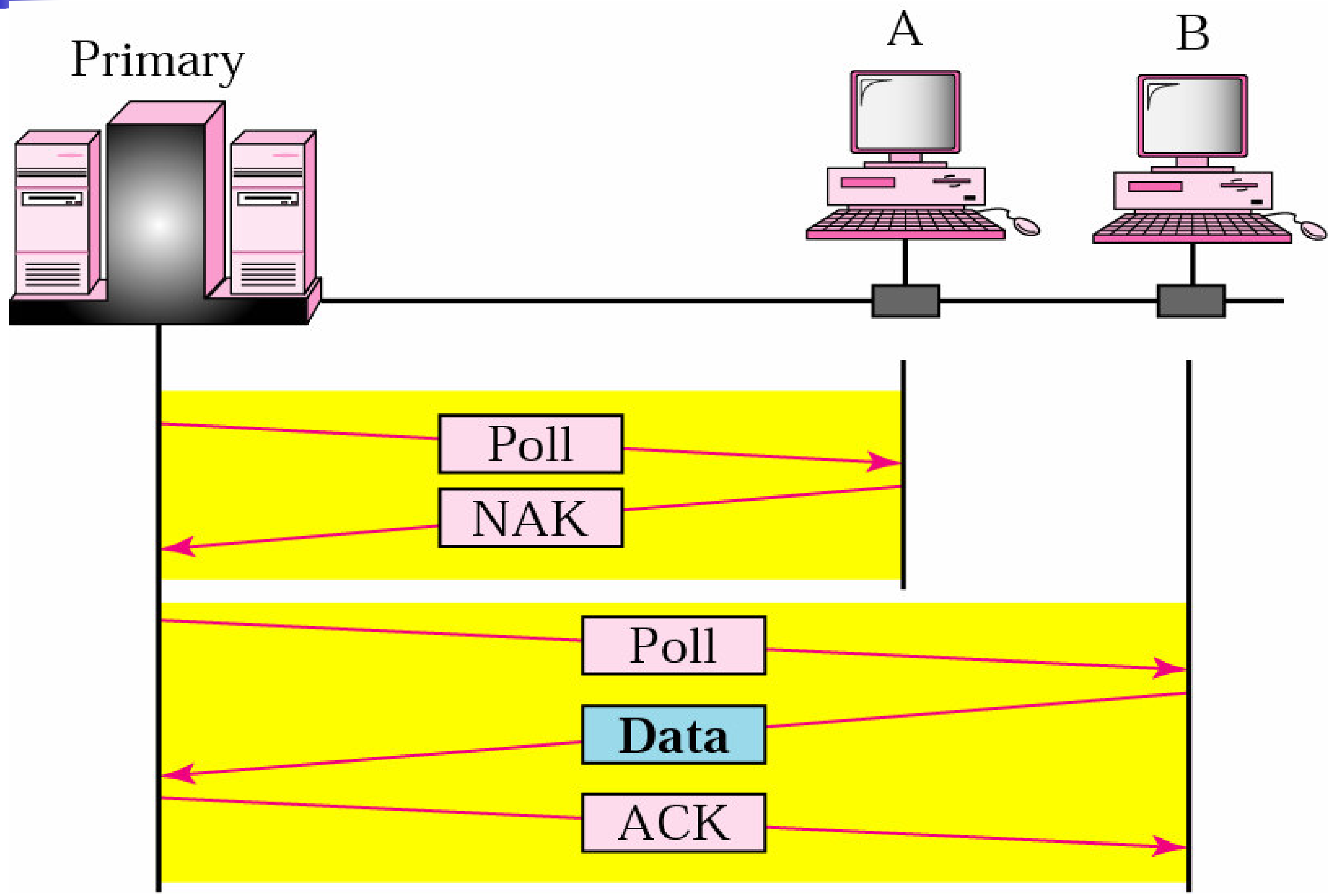


Figure 13.12 Token-passing network

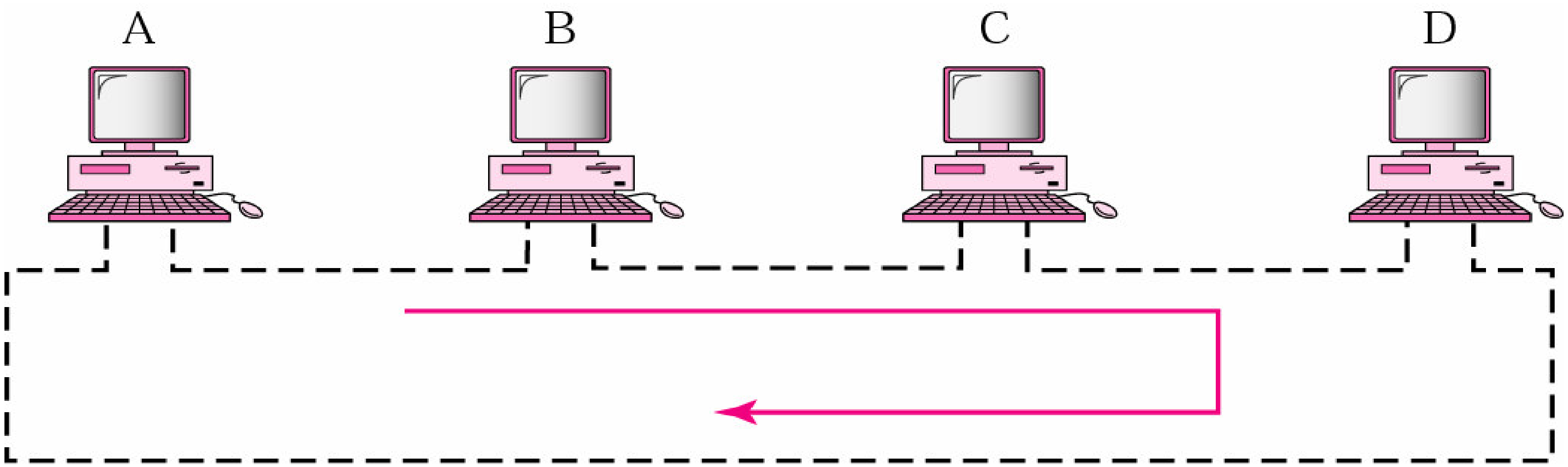
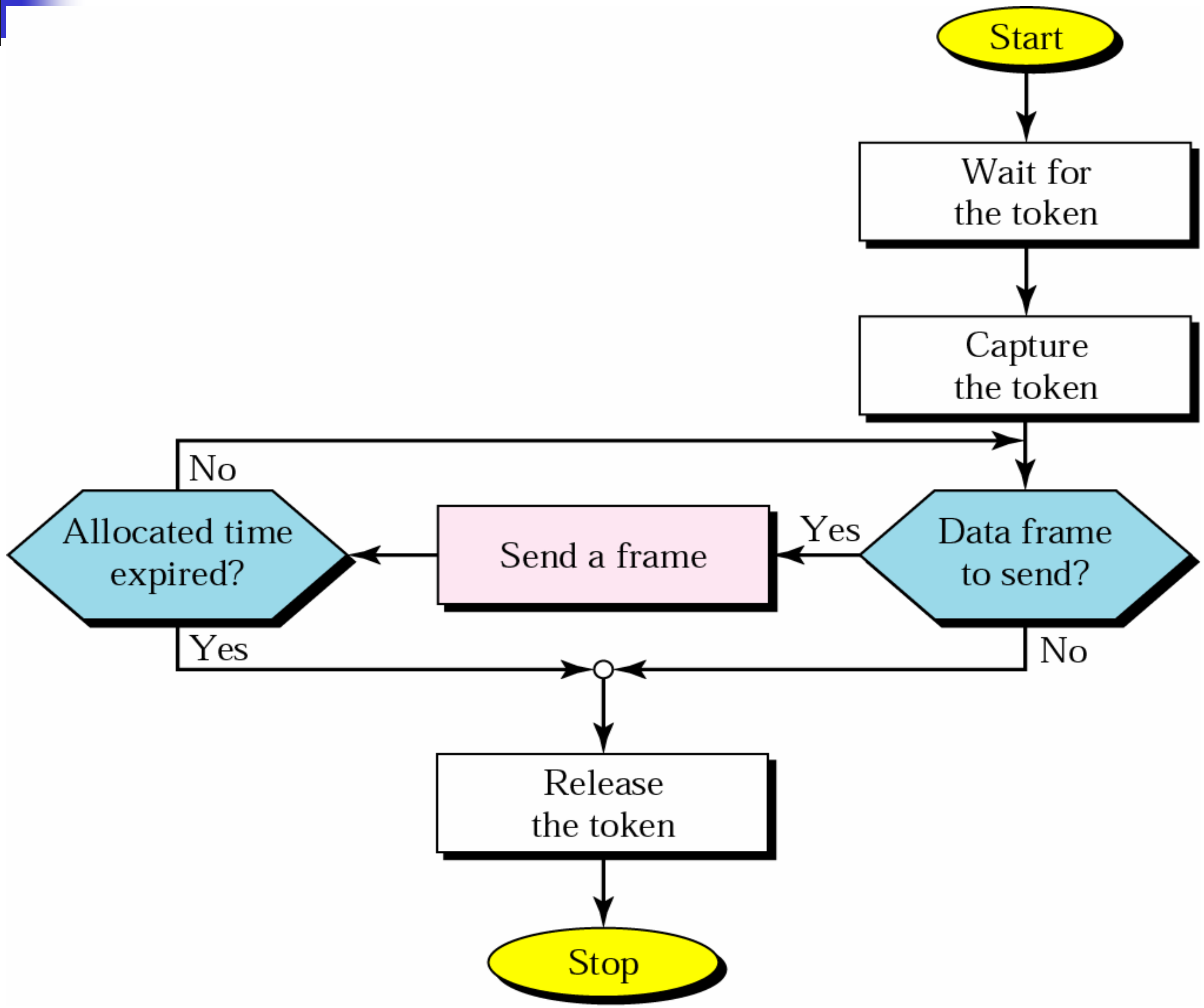


Figure 13.13 Token-passing procedure



13.3 Channelization

FDMA

TDMA

CDMA

Note:

In FDMA, the bandwidth is divided into channels.

Note:

In TDMA, the bandwidth is just one channel that is timeshared.

Note:

In CDMA, one channel carries all transmissions simultaneously.

Figure 13.14 Chip sequences

+1, +1, +1, +1	+1, −1, +1, −1	+1, +1, −1, −1	+1, −1, −1, +1
A	**B**	**C**	**D**

Figure 13.15 Encoding rules

Data bit 0 → −1   Data bit 1 → +1   Silence → 0

Figure 13.16 CDMA multiplexer

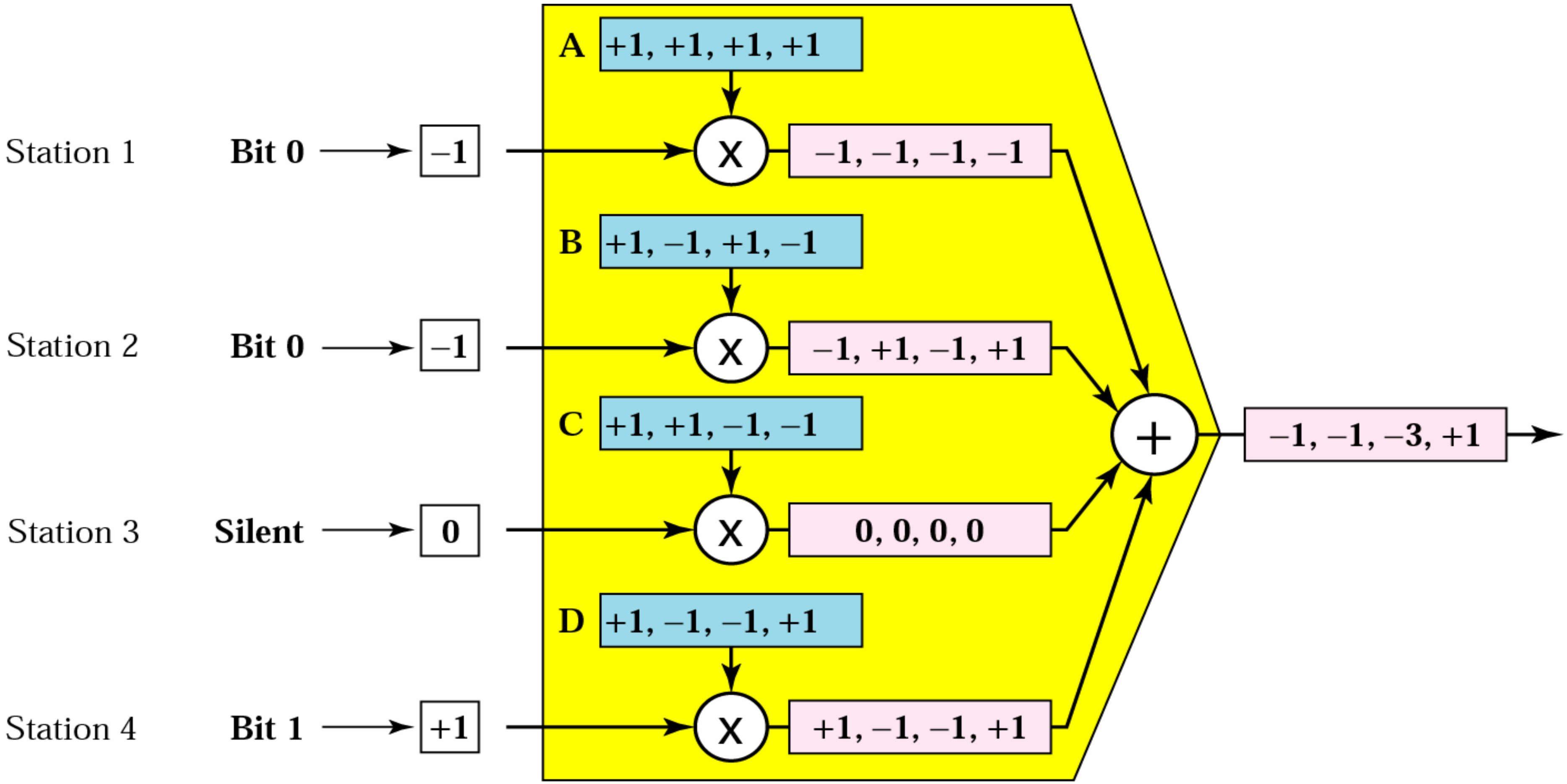


Figure 13.17 CDMA demultiplexer

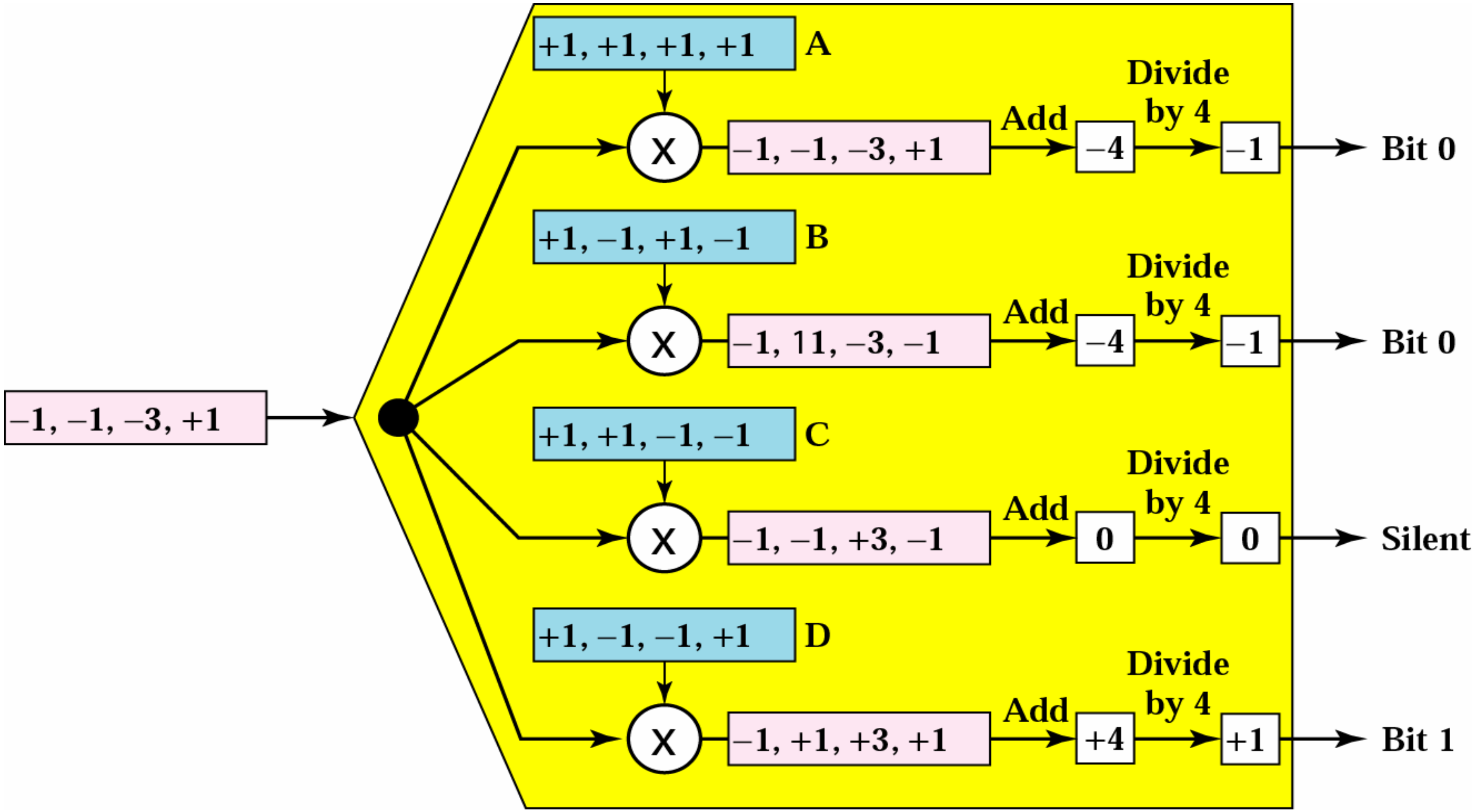


Figure 13.18 W1 and W2N

$$W_1 = \begin{bmatrix} +1 \end{bmatrix} \qquad W_{2N} = \begin{bmatrix} W_N & W_N \\ W_N & \overline{W_N} \end{bmatrix}$$

Figure 13.19 Sequence generation

$$W_1 = \begin{bmatrix} +1 \end{bmatrix}$$

$$W_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & -1 & +1 \end{bmatrix}$$

Example 1

Check to see if the second property about orthogonal codes holds for our CDMA example.

Solution

The inner product of each code by itself is N. This is shown for code C; you can prove for yourself that it holds true for the other codes.

$$\mathbf{C} \cdot \mathbf{C} = [+1, +1, -1, -1] \cdot [+1, +1, -1, -1] = 1 + 1 + 1 + 1 = 4$$

If two sequences are different, the inner product is 0.

$$\mathbf{B} \cdot \mathbf{C} = [+1, -1, +1, -1] \cdot [+1, +1, -1, -1] = 1 - 1 - 1 + 1 = 0$$

Example 2

Check to see if the third property about orthogonal codes holds for our CDMA example.

Solution

The inner product of each code by its complement is –N. This is shown for code C; you can prove for yourself that it holds true for the other codes.

C . (–C) = [+1, +1, –1, –1] . [–1, –1, +1, +1] = – 1 – 1 – 1 – 1 = –4

The inner product of a code with the complement of another code is 0.

B . (–C) = [+1, –1, +1, –1] . [–1, –1, +1, +1] = –1 + 1 + 1 – 1 = 0