

# **Chapter 5:**

# **Basic Routing Protocols for**

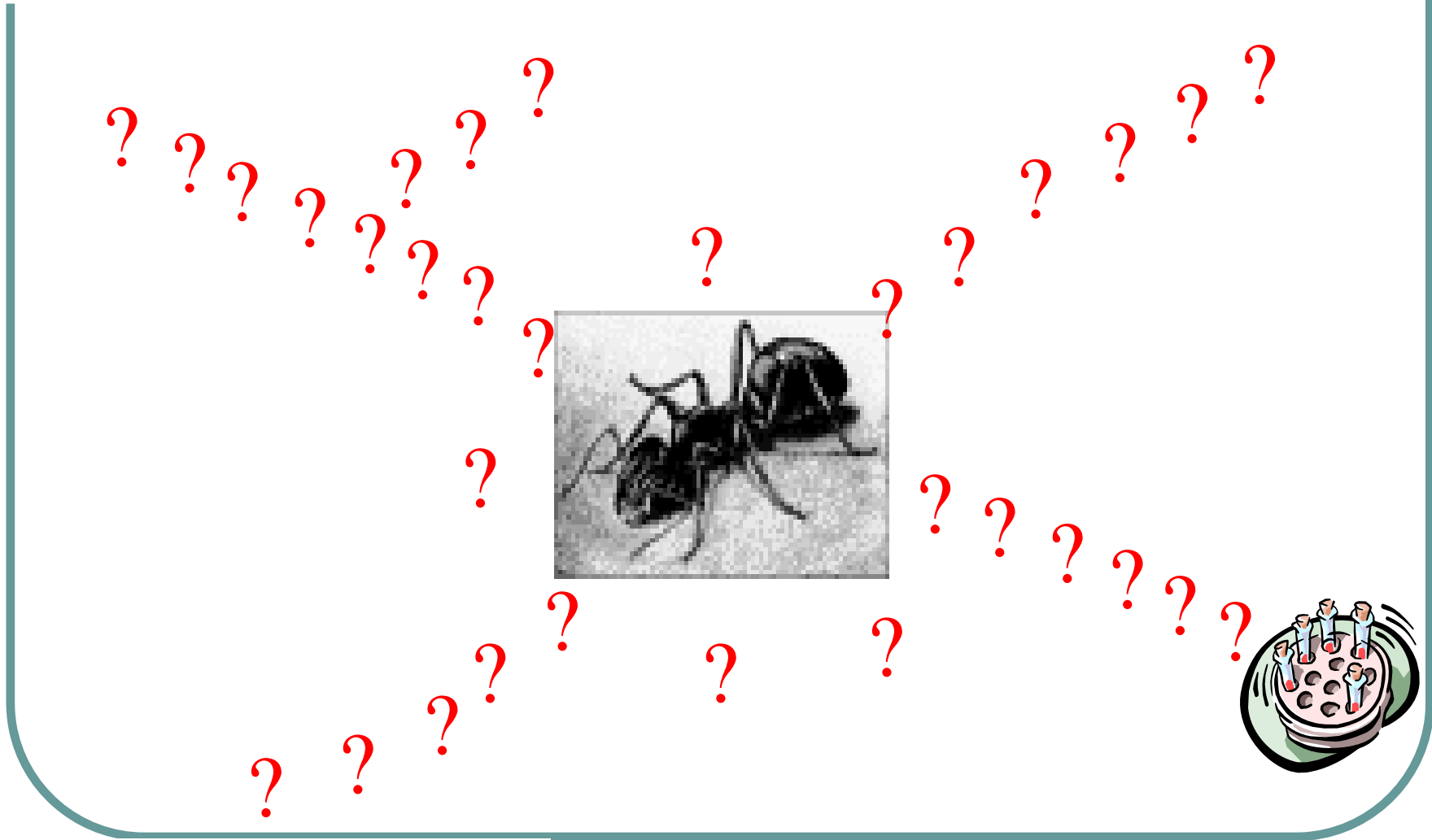
# **Ad Hoc Mobile Wireless Networks**

**Prof. Yuh-Shyan Chen**  
Department of Computer Science  
and Information Engineering  
National Taipei University

# Outline

- Routing = Ants Searching for Food
- Introduction to Ad Hoc Wireless Networks
- Can existing internet routing protocols be used for ad hoc ?
- Some Ad-Hoc Routing Protocols  
(Proactive vs. Reactive)

# Routing = Ants Searching for Food







Click on the buttons below to navigate through our tour, and learn more about the capabilities of MeshNetworks' mobile broadband solution.



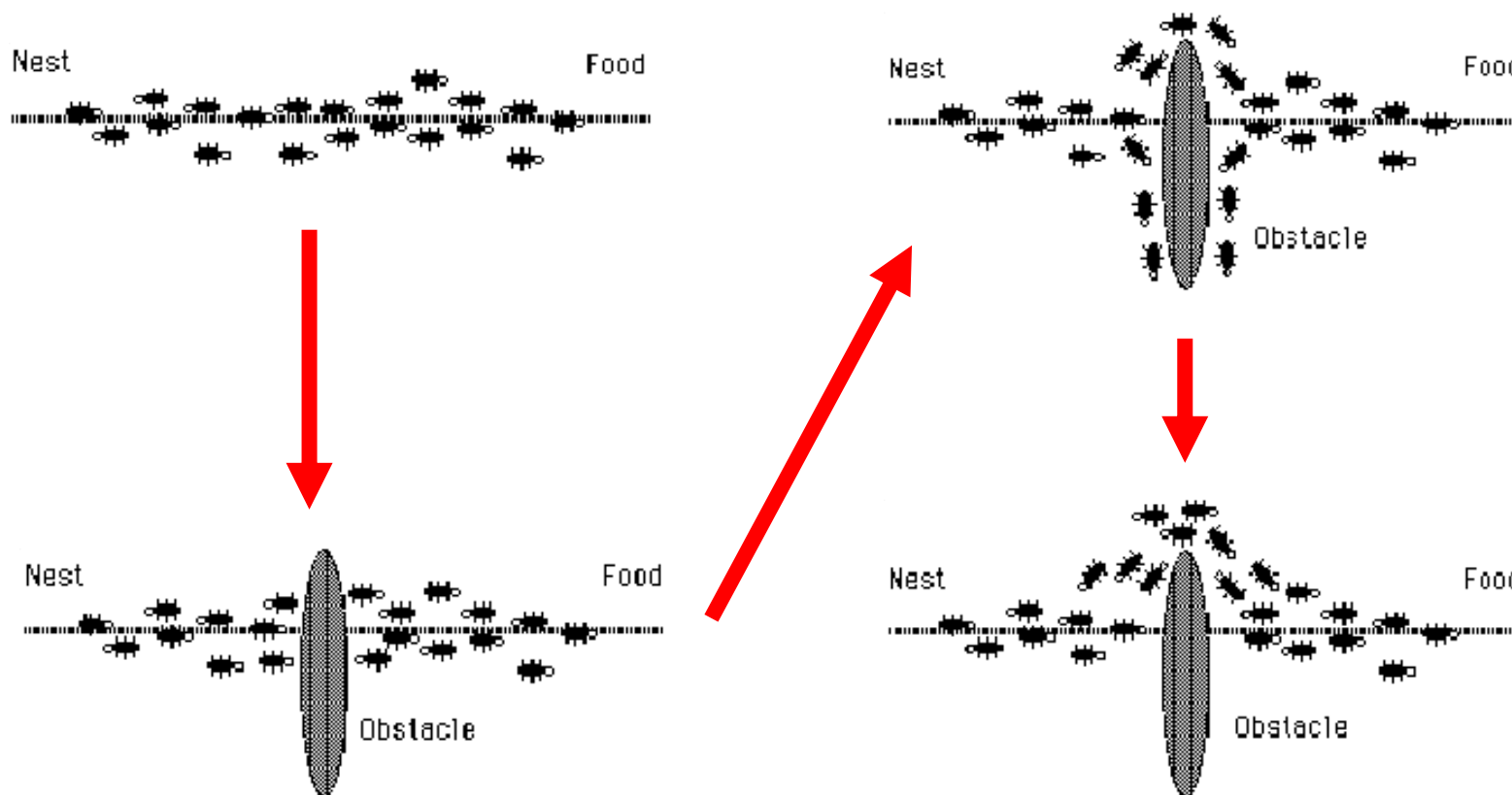
Members of a peer-to-peer group can also hop onto the Internet or telephone network anytime.

TO INTERNET  
AND PSTN

#### COMPONENT KEY

-  Subscriber Device
-  Wireless Router • WR
-  Intelligent Access Point • IAP
-  Mobile internet Switching Center • MISC

# Example:



# Three Main Issues in Ants' Life

- **Route Discovery:**
  - Searching for the places with food
- **Packet Forwarding:**
  - Delivering foods back home
- **Route Maintenance:**
  - When foods move to new place

# Mobile Ad Hoc Networks

- Infrastructureless mobile network
- No fixed routers, no base stations
- All nodes can move and be connected dynamically
- All nodes is treated as routers
- Application
  - battlefield, disaster areas

# Mobile Ad Hoc Networks





# Challenge of Ad Hoc NETs

- No centralized entity
- Host is no longer just an end system
- Acting as an intermediate system
- Changing network topology over time
- Every node can be mobile

# Can existing internet routing protocols be used for ad hoc?

- Link-state Routing
- Distance-vector Routing

# Link State Routing

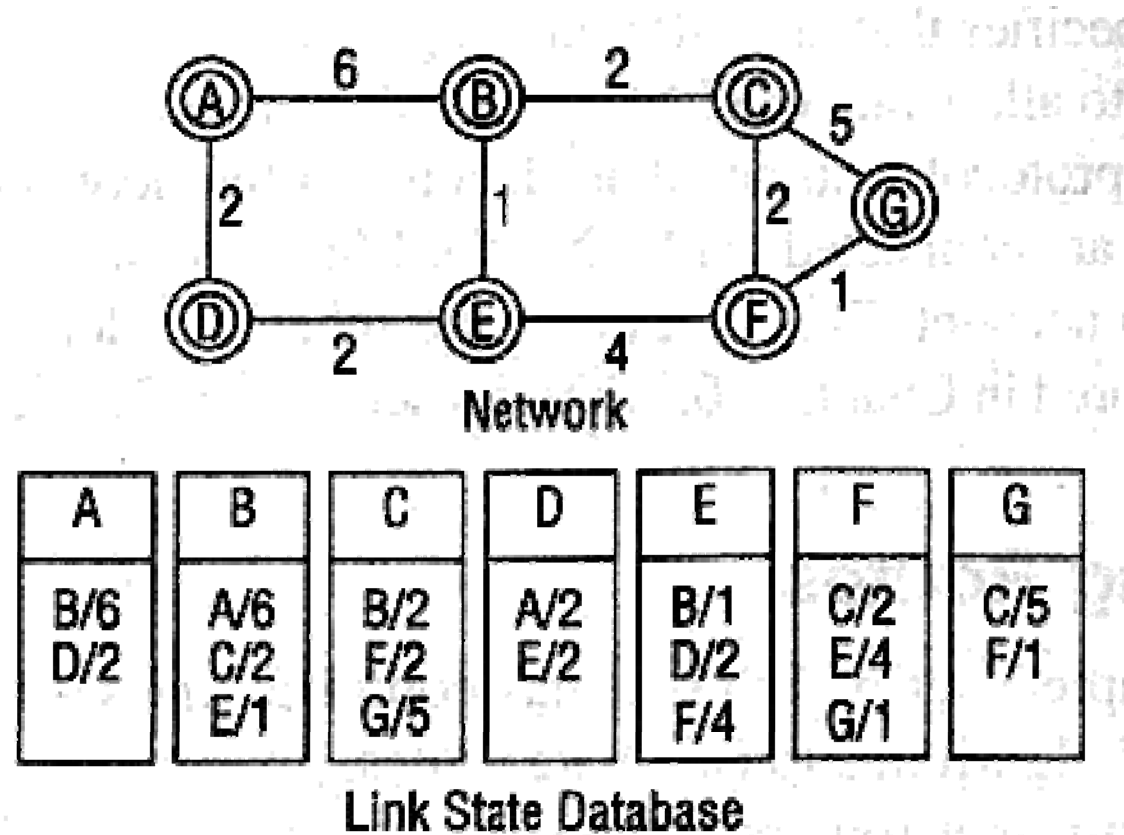
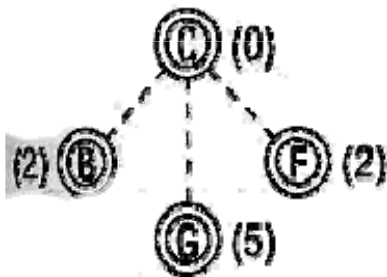
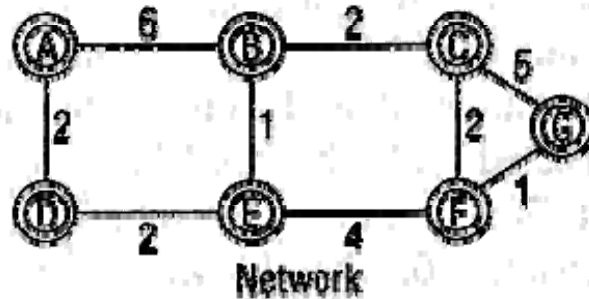


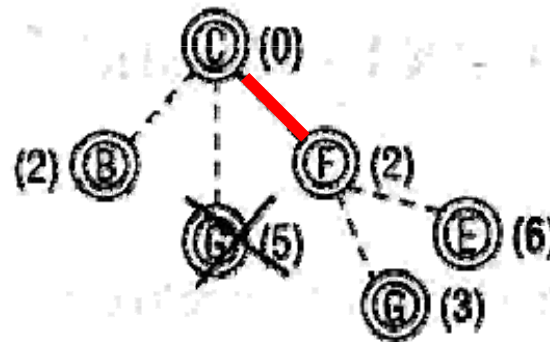
Figure 9.13

# Dijkstra computing as done by C

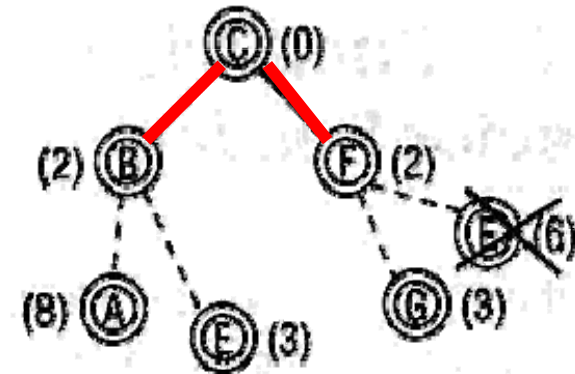
Dijkstra's algorithm refers in "Introduction to algorithm" by C, L, & R. pp 527



1. Place C in path.  
Examine C's LSP.



2. Place F in path.  
Examine F's LSP.  
Better path to G found.



3. Place B in path.  
Examine B's LSP.  
Better path to E found.

# Shortest Path (cont.)

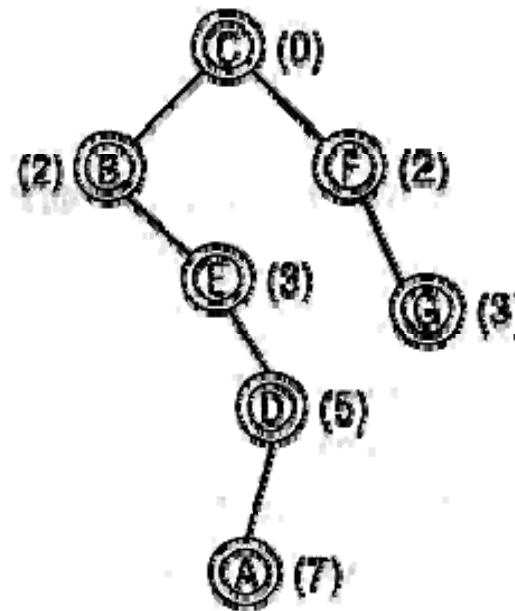
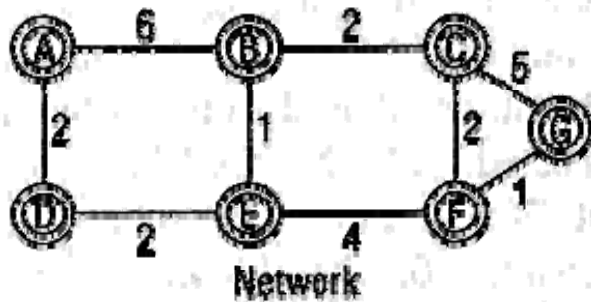
Network

4. Place E in path.  
Examine E's LSP.

5. Place G in path.  
Examine G's LSP.

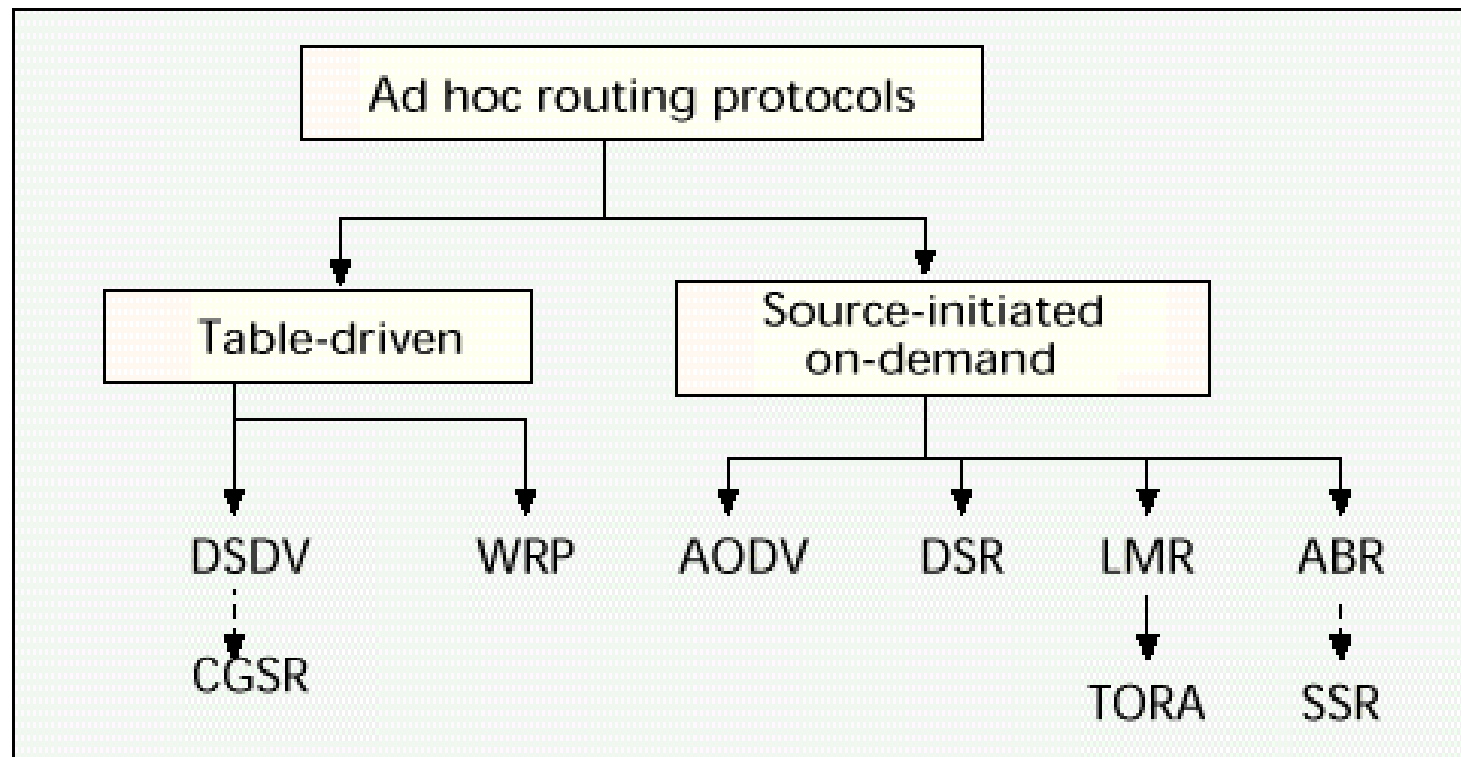
6. Place D in path.  
Examine D's LSP.  
Better path to A found.

# Dijkstra computing (cont.)



7. Place A in path.  
Examine A's LSP.  
No nodes left.  
Terminate.

# Overview of current approaches



# Proactive vs. Reactive Routing

- **Proactive Routing Protocol**
  - Continuously evaluate the routes
  - Attempt to maintain consistent, up-to-date routing information
    - When a route is needed, one may be ready immediately
  - When the network topology changes
    - The protocol responds by propagating updates throughout the network to maintain a consistent view
- **Reactive Routing Protocol**
  - Ex: DSR, AODV



# On-demand vs Table-driven

<i>Parameters</i>	<i>On-demand</i>	<i>Table-driven</i>
Availability of routing information	Available when needed	Always available regardless of need
Routing philosophy	Flat	Mostly flat, except for CGSR
Periodic route updates	Not required	Required
Coping with mobility	Use localized route discovery as in ABR and SSR	Inform other nodes to achieve a consistent routing table
Signaling traffic generated	Grows with increasing mobility of active routes (as in ABR)	Greater than that of on-demand routing
Quality of service support	Few can support QoS, although most support shortest path	Mainly shortest path as the QoS metric

# Table-Driven Routing

- **DSDV**: Destination Sequence Distance Vector
- WRP: Wireless Routing Protocol
- **CGSR**: Clustered Gateway Switch Routing

# DSDV

- Destination Sequenced Distance Vector
  - Table-driven
  - Based on the distributed **Bellman-Ford** routing algorithm
  - Each node maintains a routing table
    - **Routing hops to each destination**
    - **Sequence number**

# DSDV

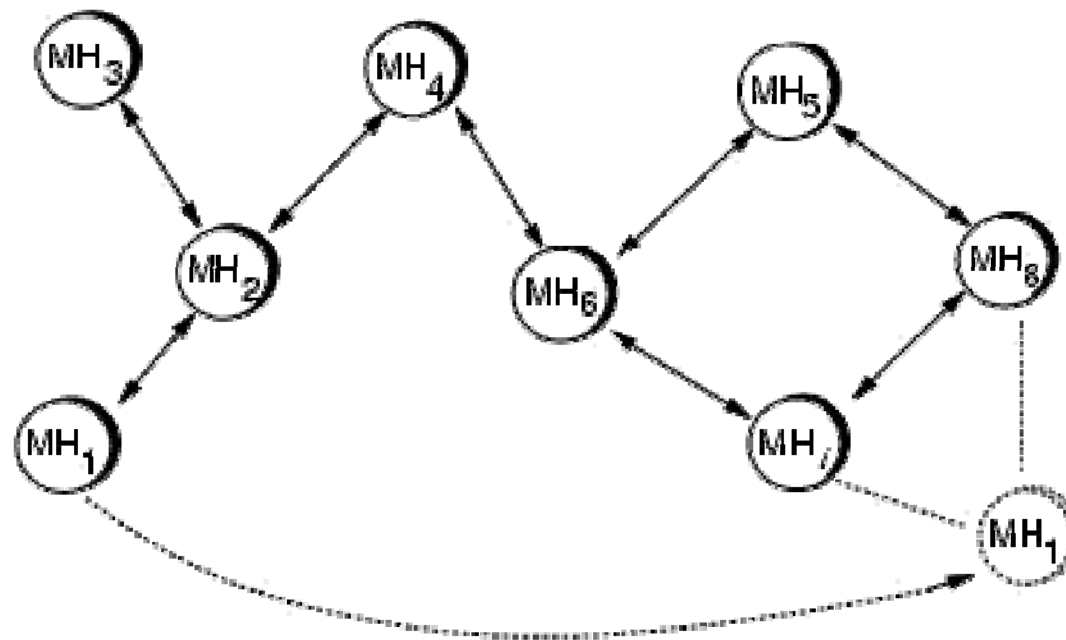


Figure 1: Movement in an ad-hoc network

# DSDV(cont.)

Destination	NextHop	Metric	Sequence number	Install	Stable_data
$MH_1$	$MH_2$	2	S406_ $MH_1$	T001_ $MH_4$	Ptr1_ $MH_1$
$MH_2$	$MH_2$	1	S128_ $MH_2$	T001_ $MH_4$	Ptr1_ $MH_2$
$MH_3$	$MH_2$	2	S564_ $MH_3$	T001_ $MH_4$	Ptr1_ $MH_3$
$MH_4$	$MH_4$	0	S710_ $MH_4$	T001_ $MH_4$	Ptr1_ $MH_4$
$MH_5$	$MH_6$	2	S392_ $MH_5$	T002_ $MH_4$	Ptr1_ $MH_5$
$MH_6$	$MH_6$	1	S076_ $MH_6$	T001_ $MH_4$	Ptr1_ $MH_6$
$MH_7$	$MH_6$	2	S128_ $MH_7$	T002_ $MH_4$	Ptr1_ $MH_7$
$MH_8$	$MH_6$	3	S050_ $MH_8$	T002_ $MH_4$	Ptr1_ $MH_8$

Table 1: Structure of the  $MH_4$  forwarding table

Destination	Metric	Sequence number
$MH_1$	2	S406_ $MH_1$
$MH_2$	1	S128_ $MH_2$
$MH_3$	2	S564_ $MH_3$
$MH_4$	0	S710_ $MH_4$
$MH_5$	2	S392_ $MH_5$
$MH_6$	1	S076_ $MH_6$
$MH_7$	2	S128_ $MH_7$
$MH_8$	3	S050_ $MH_8$

Table 2: Advertised route table by  $MH_4$

# DSDV(cont.)

Destination	NextHop	Metric	Sequence number	Install	Stable_data
$MH_1$	$MH_6$	3	$S516\_MH_1$	$T810\_MH_4$	$Ptr1\_MH_1$
$MH_2$	$MH_2$	1	$S238\_MH_2$	$T001\_MH_4$	$Ptr1\_MH_2$
$MH_3$	$MH_2$	2	$S674\_MH_3$	$T001\_MH_4$	$Ptr1\_MH_3$
$MH_4$	$MH_4$	0	$S820\_MH_4$	$T001\_MH_4$	$Ptr1\_MH_4$
$MH_5$	$MH_5$	2	$S502\_MH_5$	$T002\_MH_4$	$Ptr1\_MH_5$
$MH_6$	$MH_6$	1	$S186\_MH_6$	$T001\_MH_4$	$Ptr1\_MH_6$
$MH_7$	$MH_5$	2	$S238\_MH_7$	$T002\_MH_4$	$Ptr1\_MH_7$
$MH_8$	$MH_5$	3	$S160\_MH_8$	$T002\_MH_4$	$Ptr1\_MH_8$

Table 3:  $MH_4$  forwarding table (updated)

Destination	Metric	Sequence number
$MH_4$	0	$S820\_MH_4$
$MH_1$	3	$S516\_MH_1$
$MH_2$	1	$S238\_MH_2$
$MH_3$	2	$S674\_MH_3$
$MH_5$	2	$S502\_MH_5$
$MH_6$	1	$S186\_MH_6$
$MH_7$	2	$S238\_MH_7$
$MH_8$	3	$S160\_MH_8$

Table 4:  $MH_4$  advertised table (updated)

# DSDV

- Problem

- A lot of control traffic in the network

- Solution

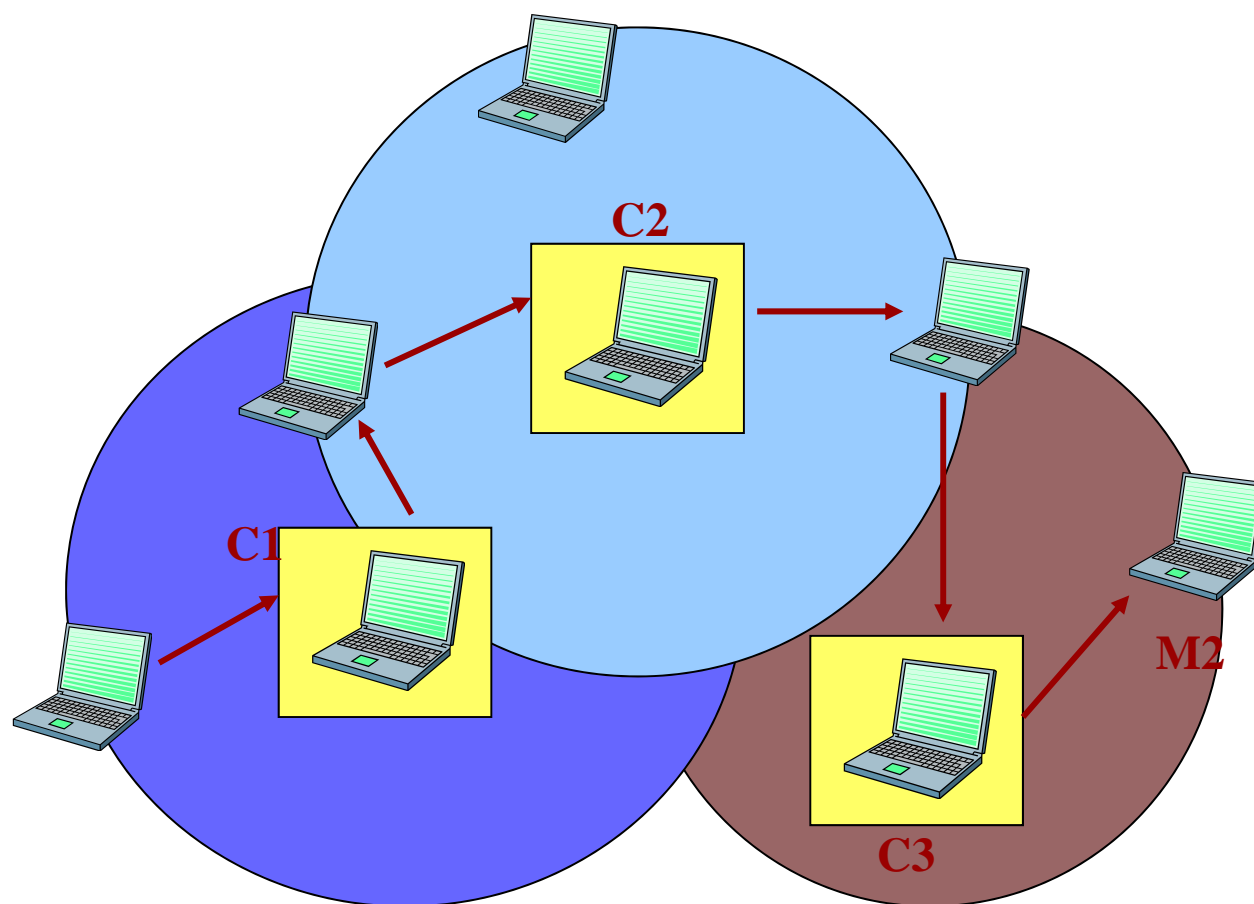
- two types of route update packets
    - Full dump
      - All available routing information
    - Incremental
      - Only information changed since the last full dump

# CGSR

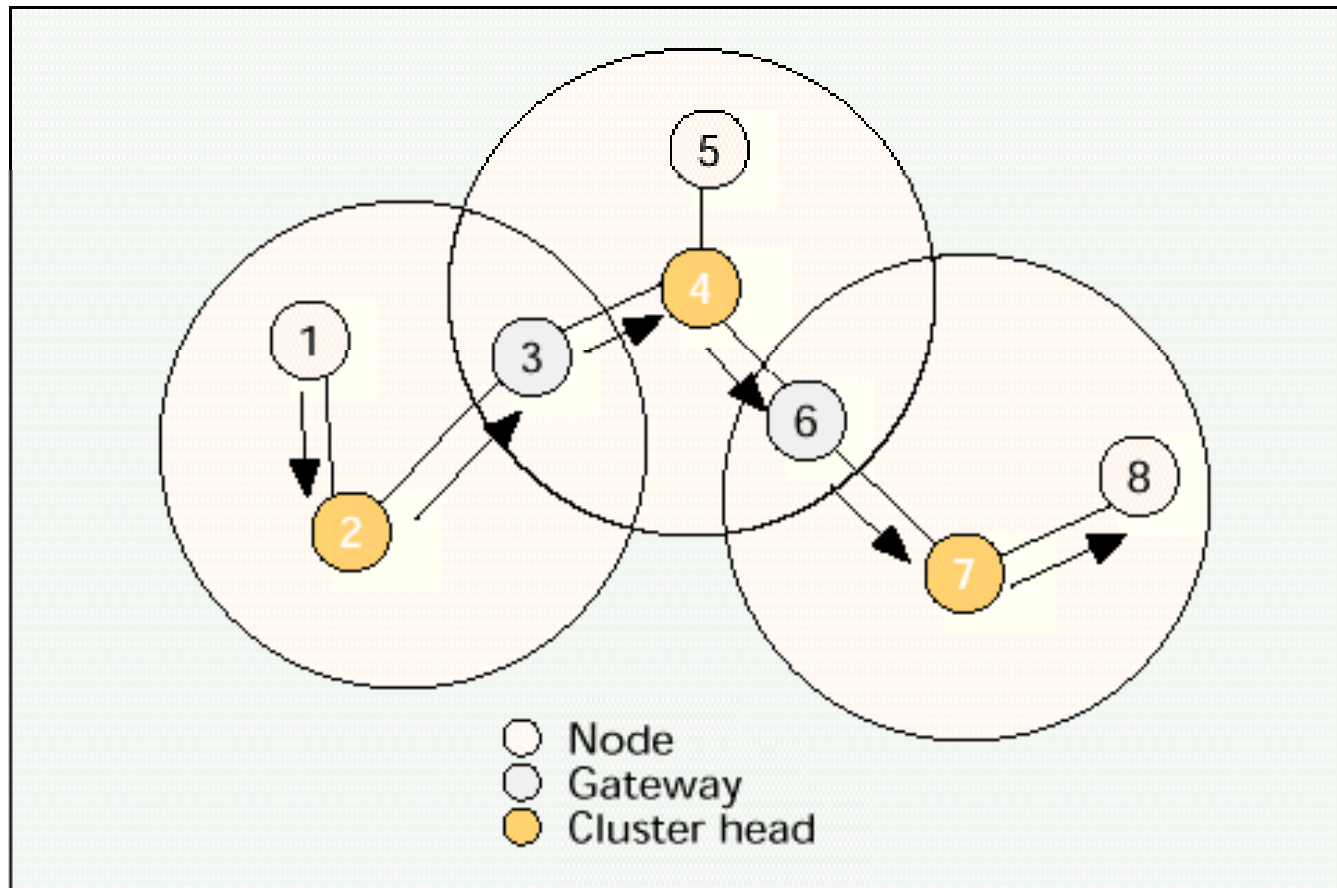
- Cluster Gateway Switch Routing
  - Table-driven
  - Uses DSDV as the underlying routing scheme
  - Modification
    - A hierarchical cluster-head-to-gateway routing approach



# Example



# CGSR (Clustered Gateway Switch Routing)



# CGSR(cont.)

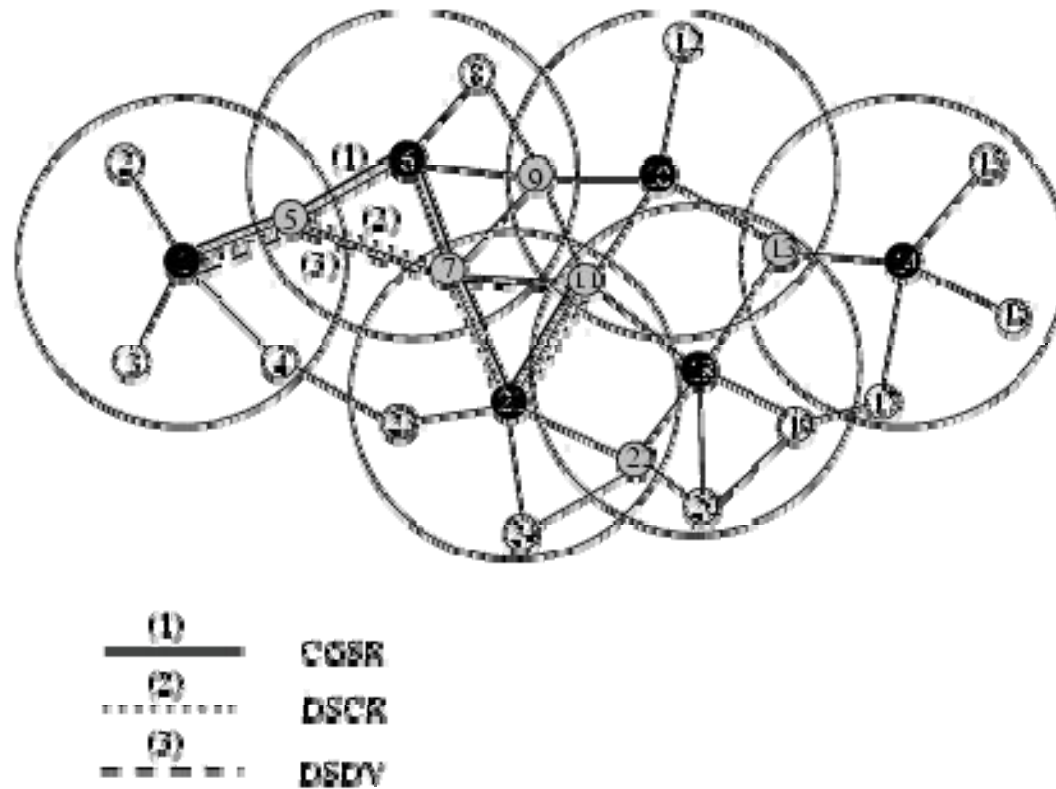


Figure 5: Routing examples (from node 1 to node 11)

# CGSR

- Problem
  - A cluster head is changing frequently
- Solution
  - The least cluster change algorithm
    - A cluster only change when
      - Two cluster heads come into contact
      - A node moves out of all other cluster heads

# On-Demand Routing

- **DSR**(Dynamic Source Routing)
- AODV(Ad Hoc On-Demand Distance Vector)
- **TORA**(Temporally Ordered Routing Algorithm)
  - MER-**TORA** (micromobility protocol)
    - IP Mobility
    - Support handover solutions
- LAR(Location-Aware Routing)
- ZRP(Zone Routing Protocol)
- SSA(Signal Stability-Based Adaptive Routing)

# DSR (Dynamic Source Routing)

# DSR

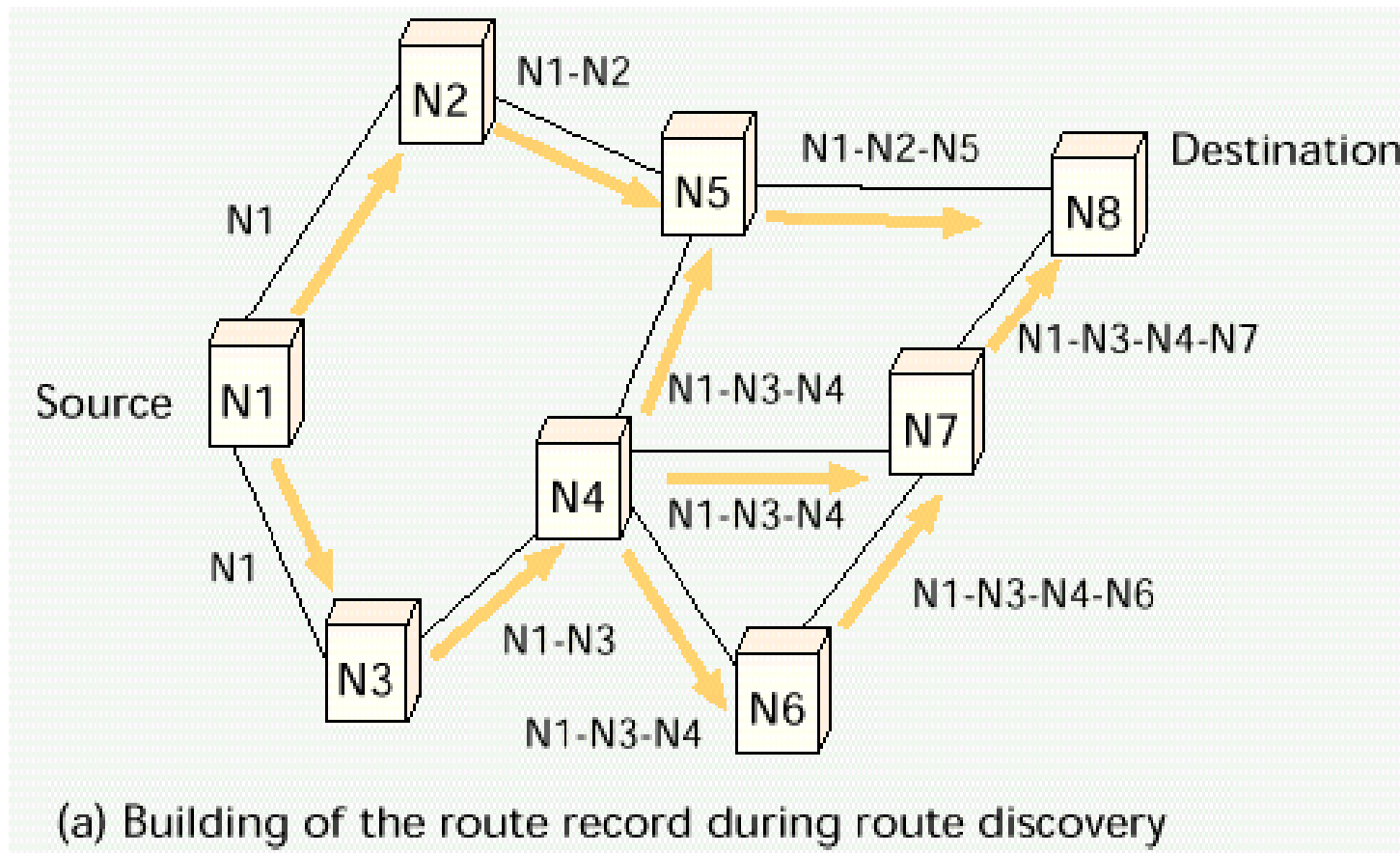
- Dynamic **Source Routing** [1996]
  - On-demand driven
  - Based on the concept of **source routing**
  - Required to maintain route caches
  - Two major phases
    - Route discovery
    - Route maintenance
      - A route error packet

# Route Discovery of DSR

- When a host has a packet to send, it first consults its **route cache**.
  - If there is an unexpired route, then it will use it.
  - Otherwise, a route discovery will be performed.
- **Route Discovery:**
  - A **ROUTE\_REQUEST** packet is sent by **flooding**.
  - There is a “**route record**” field in the packet.
    - Each node will append its address to the record.



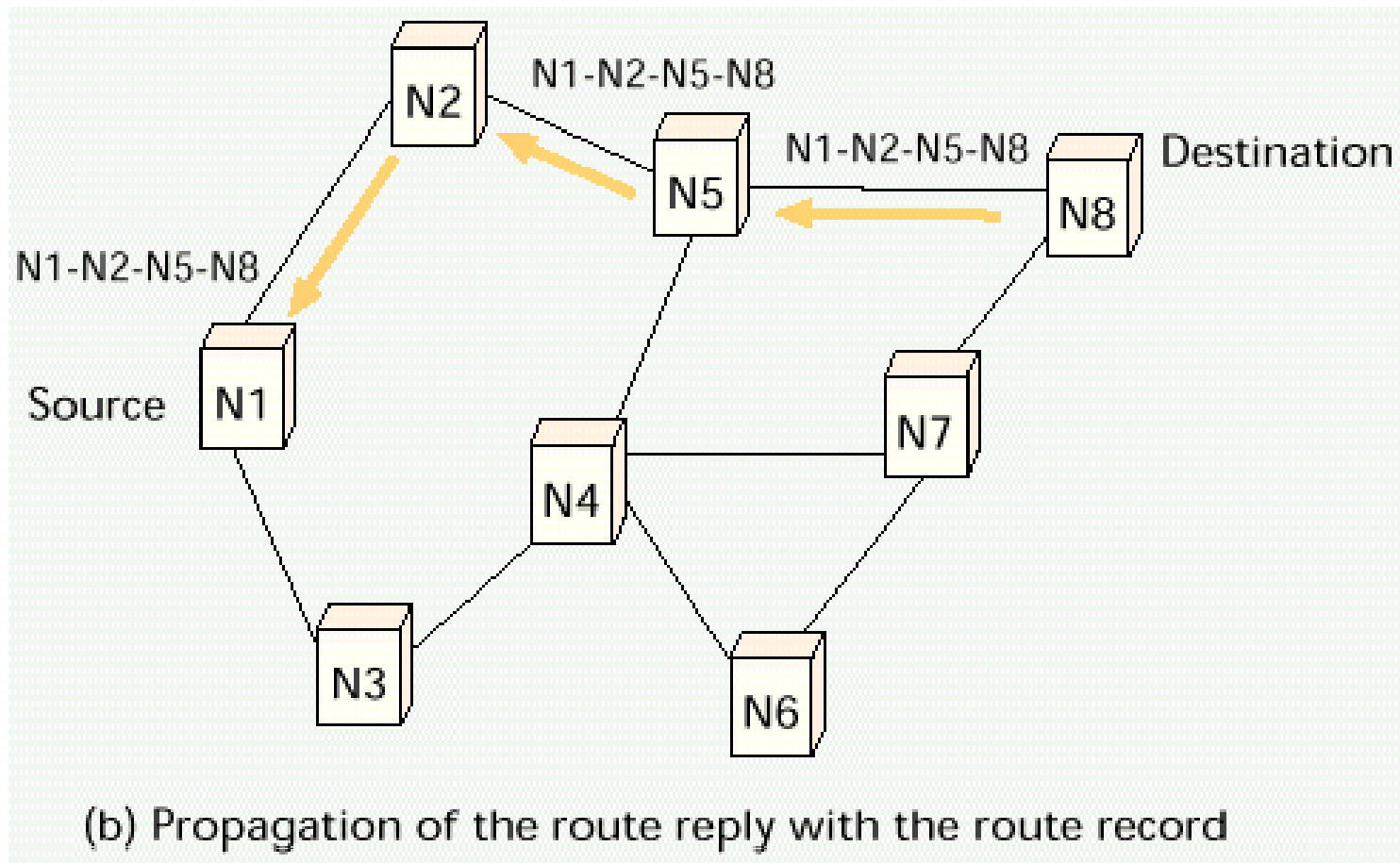
# DSR



# Route Reply of DSR

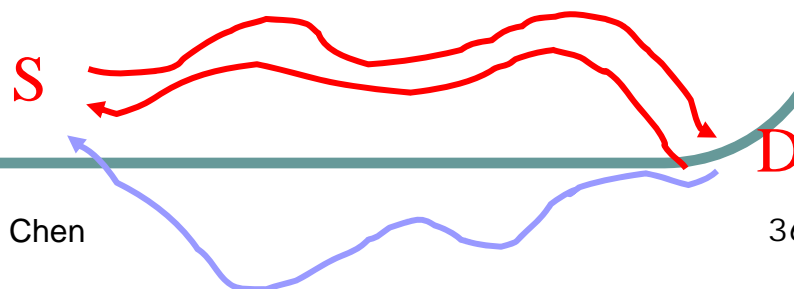
- A **ROUTE\_REPLY** packet is generated when
  - the route request packet reaches the destination
  - an intermediate host has an “unexpired” route to the destination
- A route is then generated in two manner:
  - **from destination:**
    - the route traversed by the ROUTE\_REQUEST packet
  - **from intermediate host:**
    - the route traversed by the ROUTE\_REQUEST packet concatenated with the route in the intermediate host's route cache

# DSR(cont.)



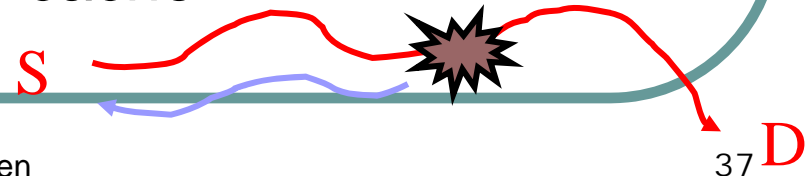
# Path of ROUTE\_REPLY

- Which way should be taken by the ROUTE\_REPLY?
- Two possibilities:
  - **symmetric path:**
    - follow the same route in the reverse order to reach the source
  - **asymmetric path:**
    - need to discover a new route to the source by initiating a ROUTE\_REQUEST to the source
    - piggyback the discovered route to the ROUTE\_REQUEST packet



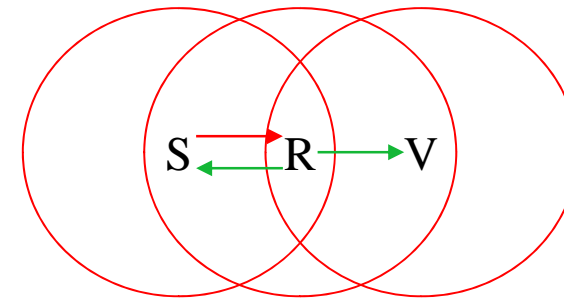
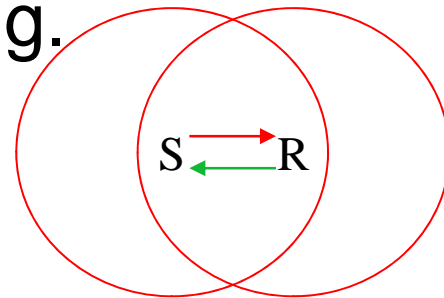
# Route Maintenance of DSR

- When the data link layer encounters a link breakage, a **ROUTE\_ERROR packet** will be initiated.
  - The packet will traverse in the backward direction to the source.
  - The source will then **initiate another ROUTE\_REQUEST**.
- Maintenance of route cache:
  - All routes which contain the breakage hop have to be removed from the route cache.



# How to Detect a Link Breakage

- Active Acknowledge:
  - The receiver of a packet actively sends an ACK to the sender.
- Passive Acknowledge:
  - The sender passively listen to the receiver's sending.



→ data packets  
→ active/passive ACK

# DSR: Route Request(RREQ)

Type=REQ	Option Length		Idetification
Target Address			
index1	index2	index3	index4
Address1			
Address2			
Address3			
Address4			

Figure 1: The ROUTE\_REQ packet used in DSR.  
The *option length* field can be used to calculate the number of addresses appended.

# DSR: Route Reply(RREP)

Type=REPLY	Option Length	R	F	Reserved
Target Address				
Index1	Index2	Index3	Index4	
Address1				
Address2				
Address3				
Address4				

Figure 2: The ROUTE\_REPLY packet used in DSR.



# DSR: Data Packet

R	Option Length			Idetification
index1	index2	index3	index4	
Address1				
Address2				
Address3				
Address4				

Figure 3: Format of data packet header in DSR. Duplicating the index and address fields, if necessary, can increase the route length by 4.

# DSR: Error Packet

Type=ERROR	Option Length	Index
Originator Address		
From Hop Address		
Next Hop Address		

Figure 4: The ERROR packet used in DSR. The *originator address* field indicates the source of the data packet experiencing error, and the *from hop* and *next hop* addresses identify the two end nodes of the broken link.

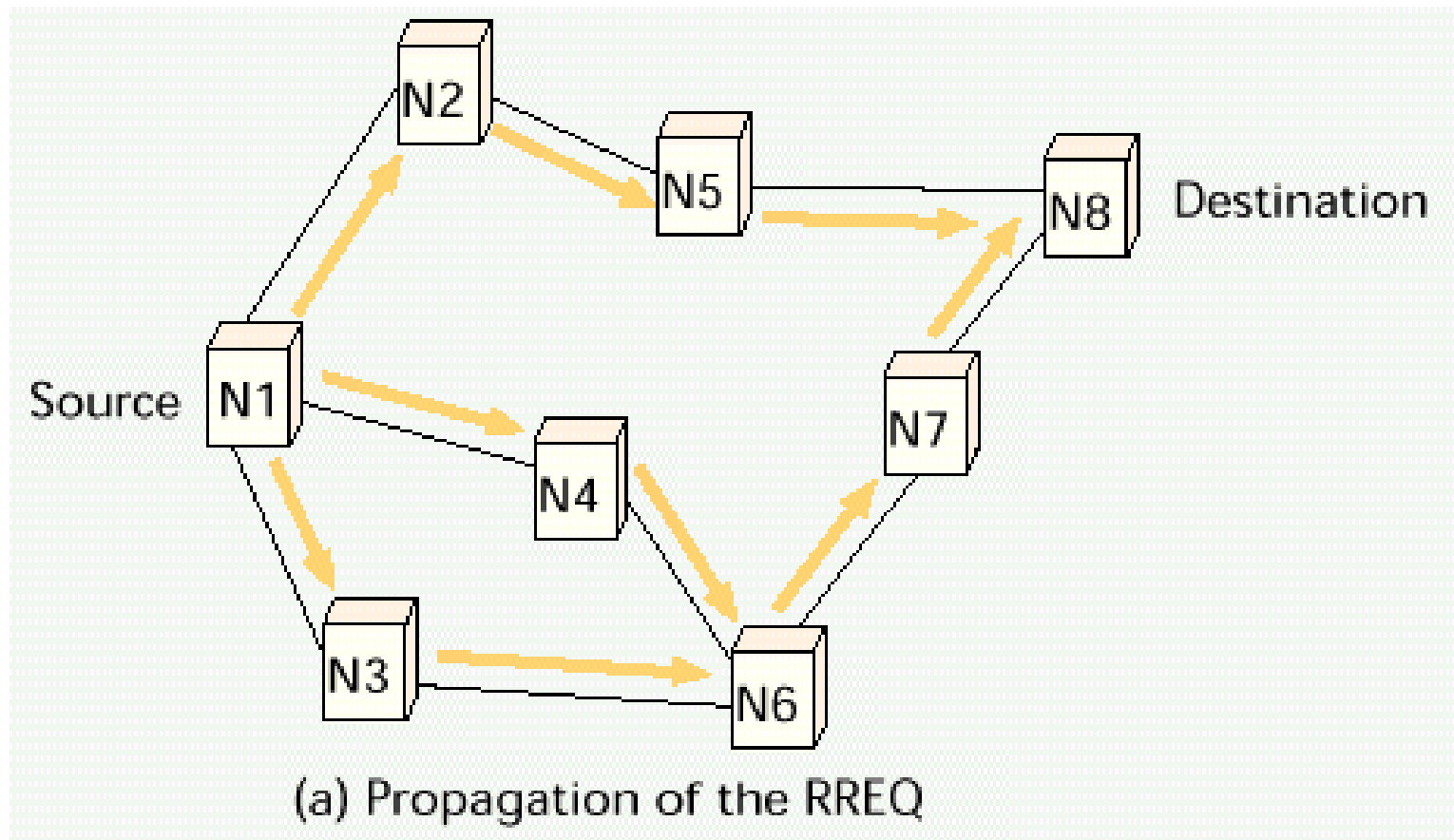
# AODV(Ad Hoc On-Demand Distance Vector)

## MAODV (Multicast AODV)

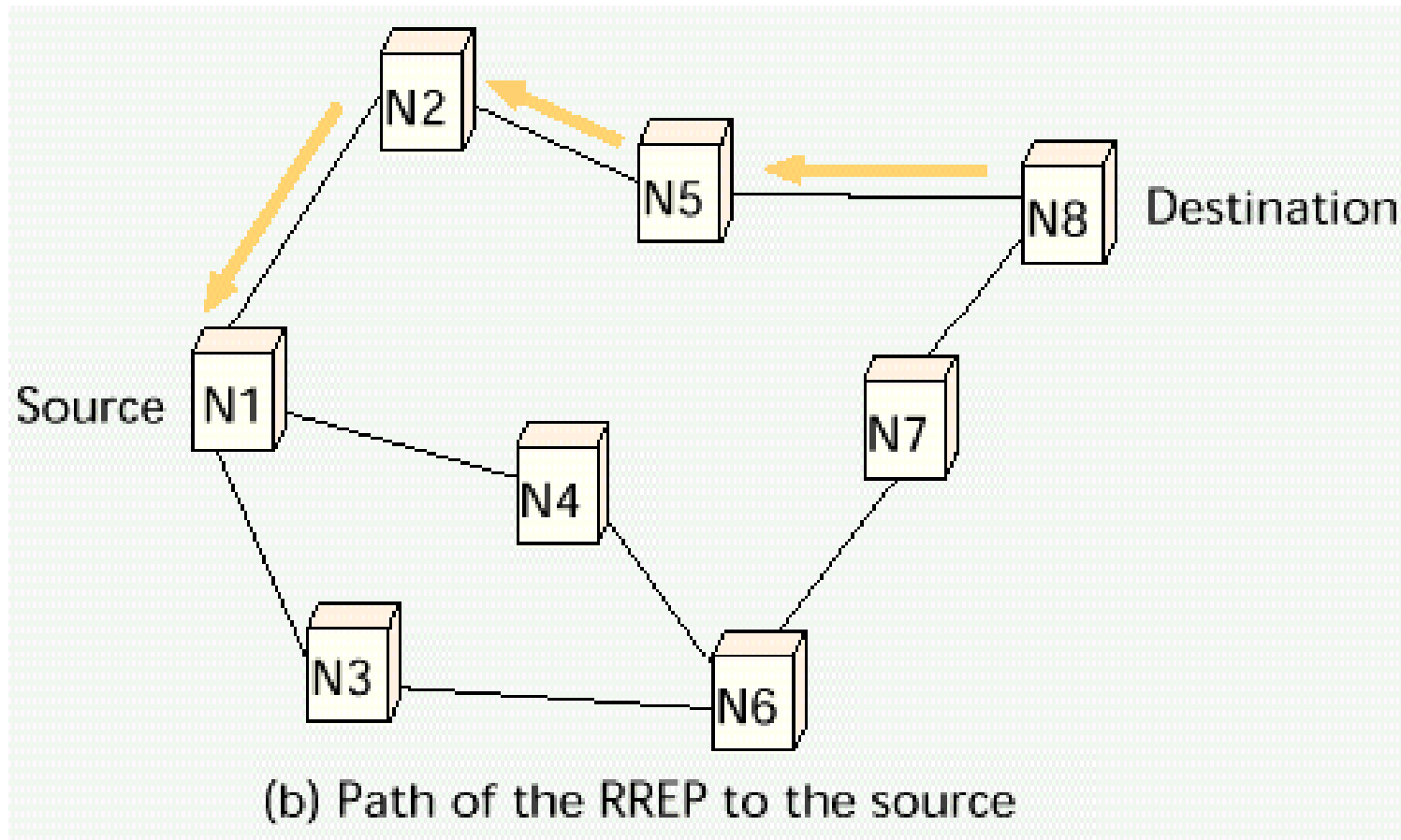
# AODV

- Ad hoc On-demand Distance Vector
  - On-demand driven
  - Nodes that are not on the selected path do not maintain routing information
  - Route discovery
    - The source node broadcasts a route request packet (RREQ)
    - The destination or an intermediate node with “fresh enough” route to the destination replies a route reply packet (RREP)

# AODV



# AODV (cont.)



# AODV: Route Request

Type	Reserved	Hop Count
Broadcast ID		
Destination IP address		
Destination Sequence Number		
Source IP address		
Source Sequence Number		

Figure 8: The ROUTE\_REQ packet used in AODV.

# AODV: Route Reply

Type	L	Reserved	Hop Count
Destination IP address			
Destination Sequence Number			
Lifetime			

Figure 9: The ROUTE\_REPLY packet used in AODV.



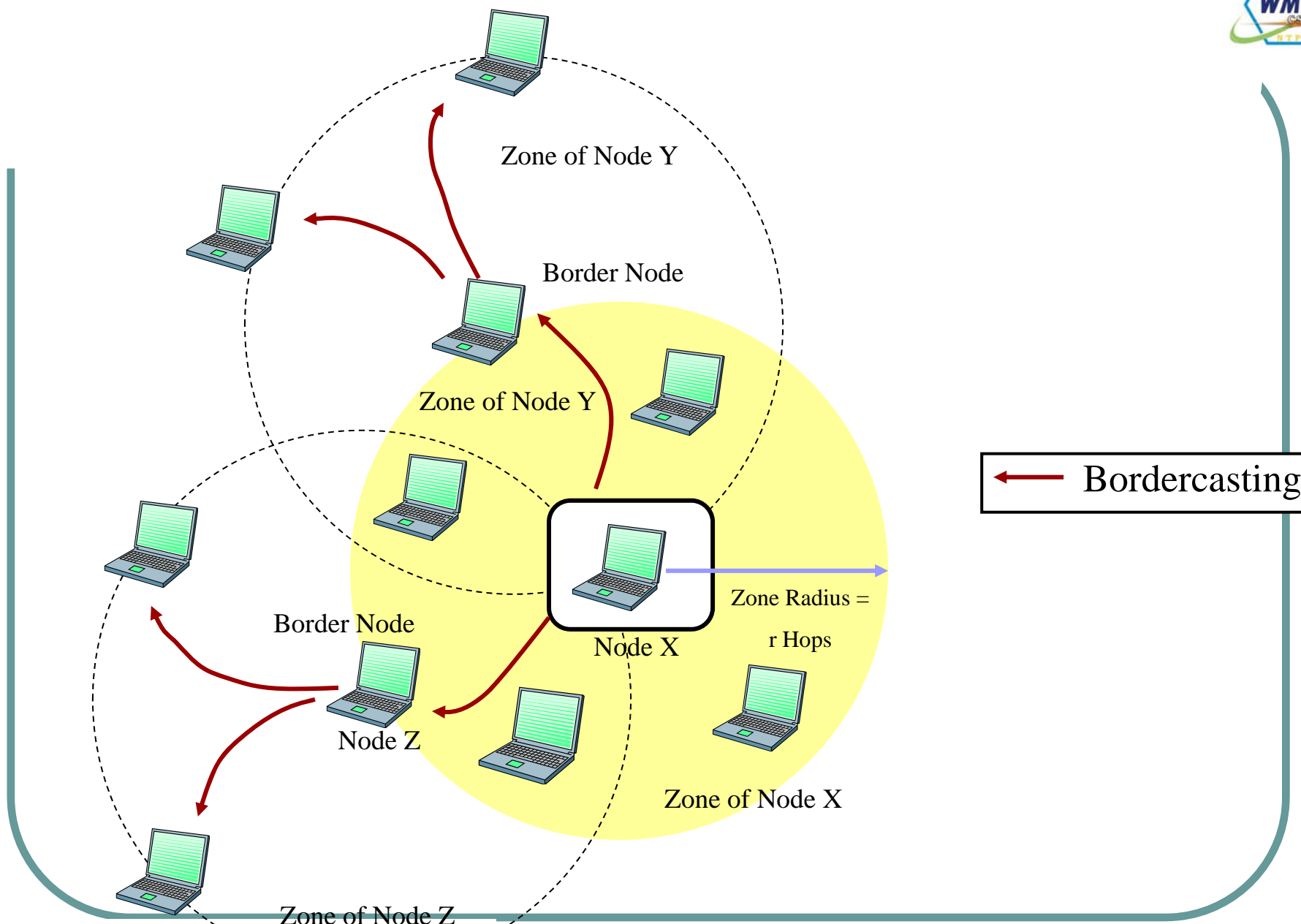
# AODV

- Problem
  - A node along the route moves
- Solution
  - Upstream neighbor notices the move
  - Propagates a link failure notification message to each of its active upstream neighbors
  - The source node receives the message and re-initiate route discovery

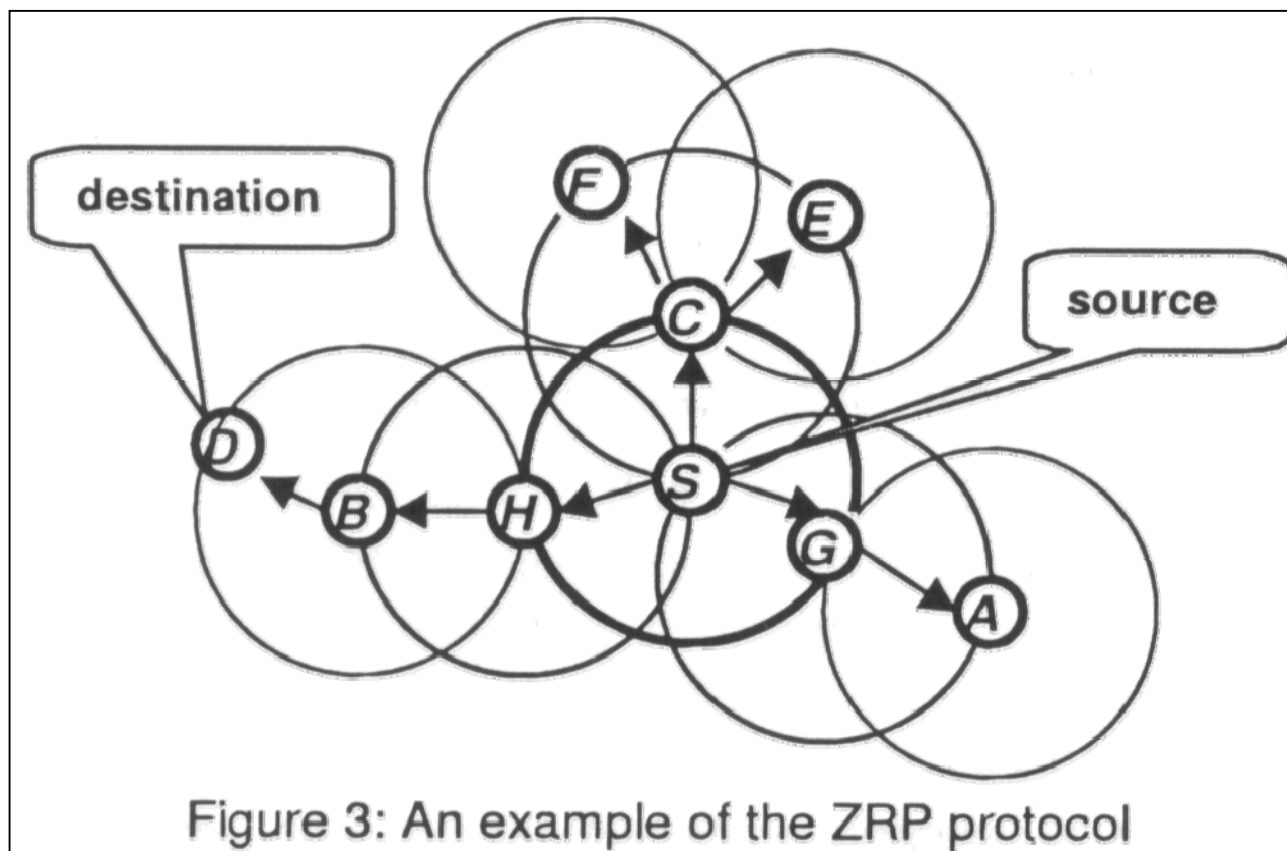
# ZRP (Zone Routing Protocol)- Combining Pro-active and Reactive

# ZRP

- Zone Routing Protocol
  - Hybrid protocol
    - On-demand
    - Proactive
  - ZRP has three sub-protocols
    - Intrazone Routing Protocol (IARP)
    - Interzone Routing Protocol (IERP)
    - Bordercast Resolution Protocol (BRP)



# ZRP



# Location-Aware Routing

(1) LAR (Mobicom'98)

(2) DREAM

(3) GRID

(Yu-Chee Tseng *et al.*, Telecommunication Systems (2001))

# Location-Aware Routing

- LAR
  - Mobicom'98
- DREAM
- GRID
  - Yu-Chee Tseng *et al.*, Telecommunication Systems (2001)

# Notebook + GPS

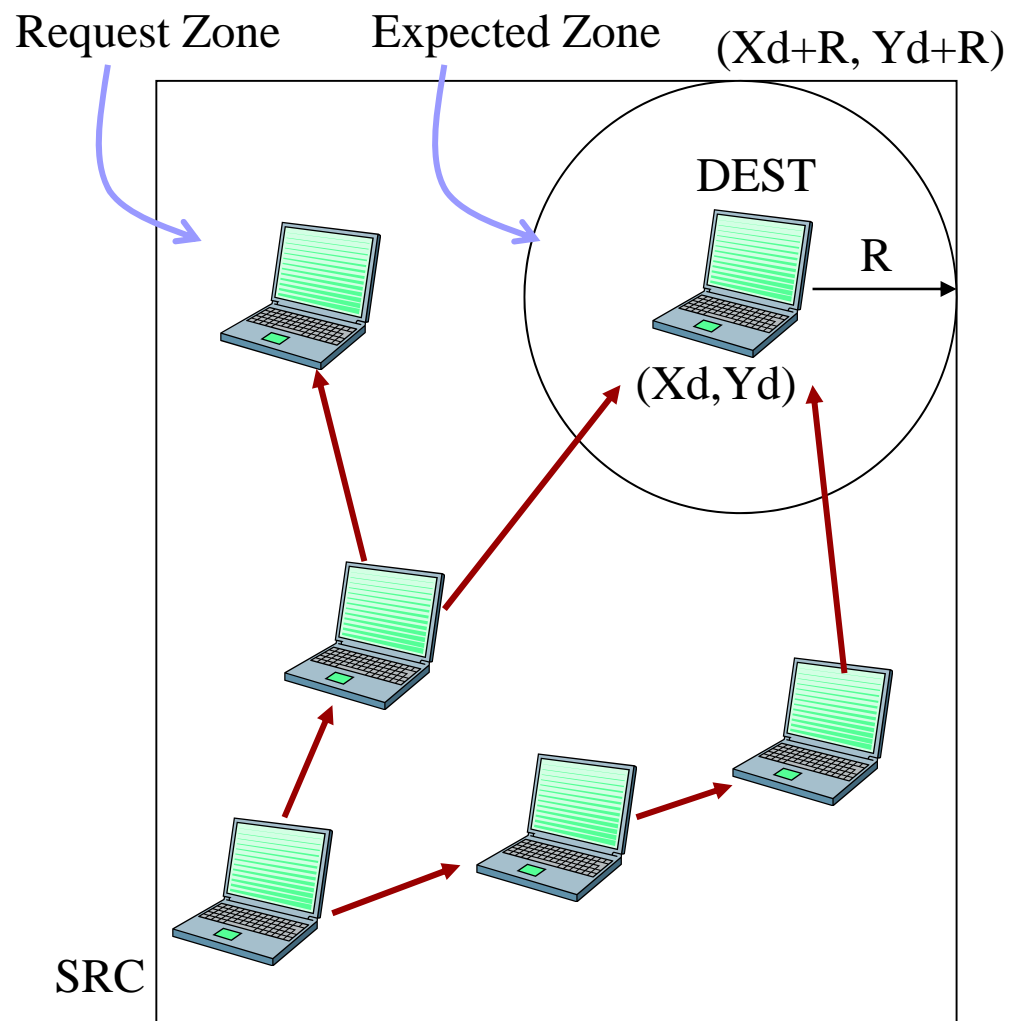




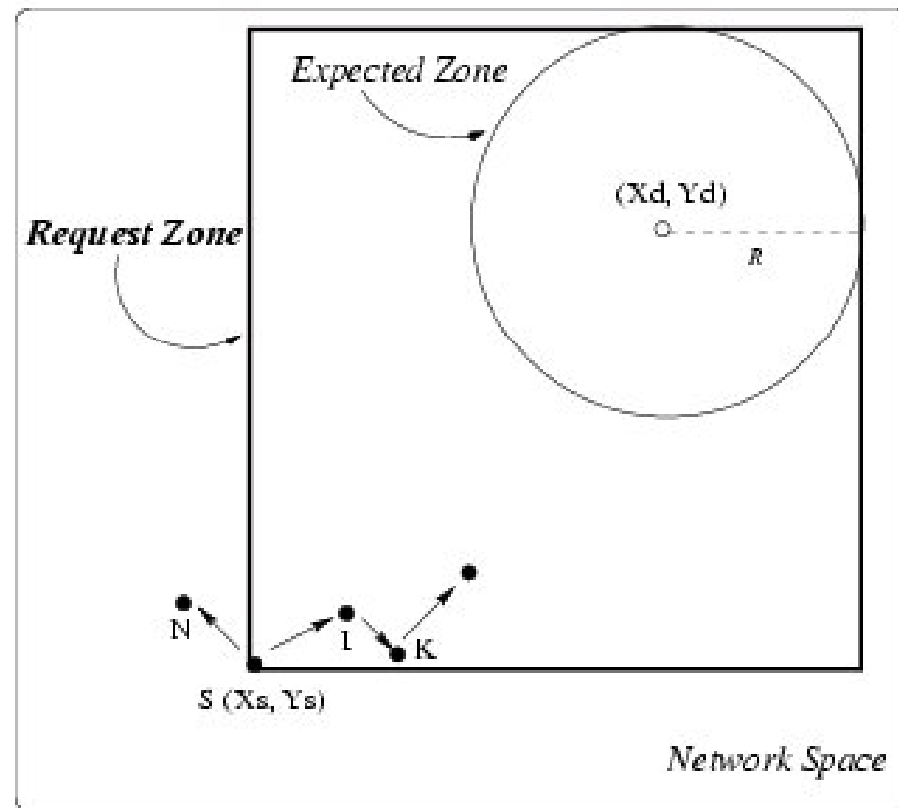
# LAR

- Location-Aided Routing [**Mobicom'98 best paper**]
  - Location information via GPS
  - Shortcoming
    - GPS availability is not yet worldwide
    - Position information come with deviation

# Reduce the flooding area

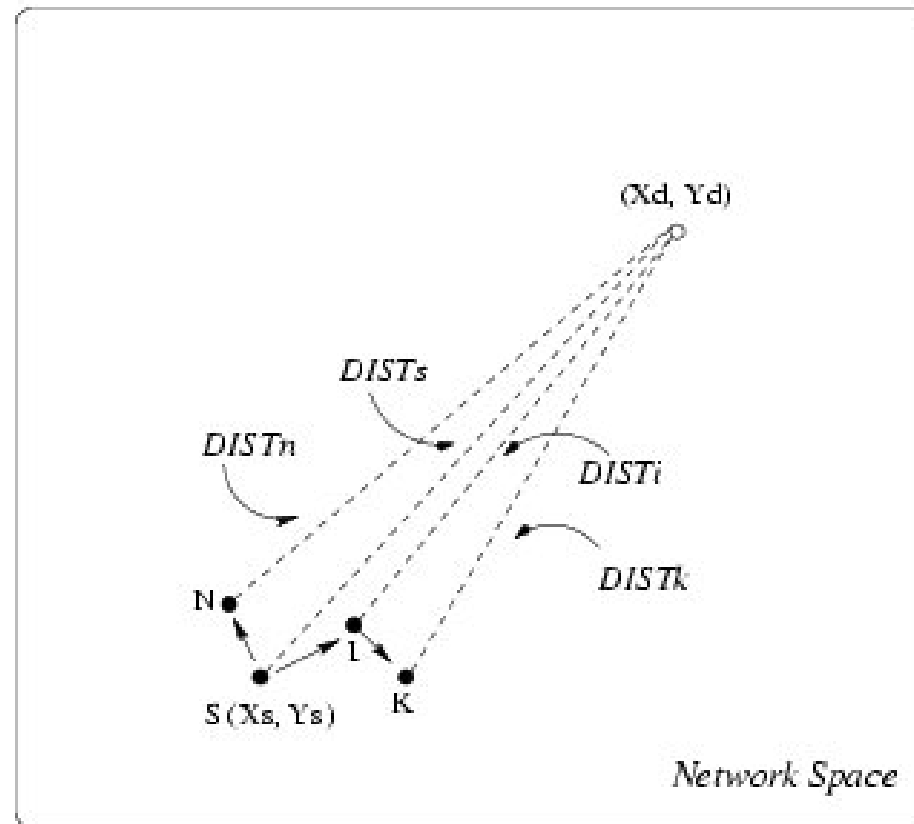


# LAR



(a) LAR scheme 1

# LAR(cont.)



(b) LAR scheme 2

# DREAM

- Distance Routing effect Algorithm for mobility
  - Position-based
  - Each node
    - maintains a position database
    - Regularly floods packets to update the position
      - Temporal resolution
      - Spatial resolution

# SSA (Signal Stability-Based Adaptive Routing)

# SSA

Host	Signal Strength	Last	Clicks	Set
Y				
Z				

Table 2: The Signal Stability Table (SST)

Destination	Next Hop
Y	
Z	

Table 3: The Routing Table (RT)

# SSA(cont.)

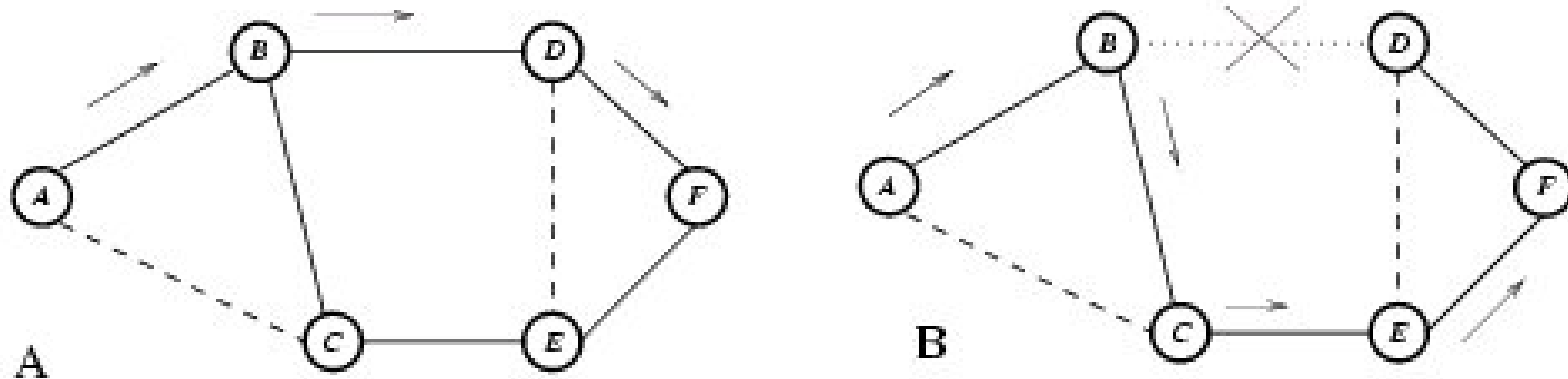


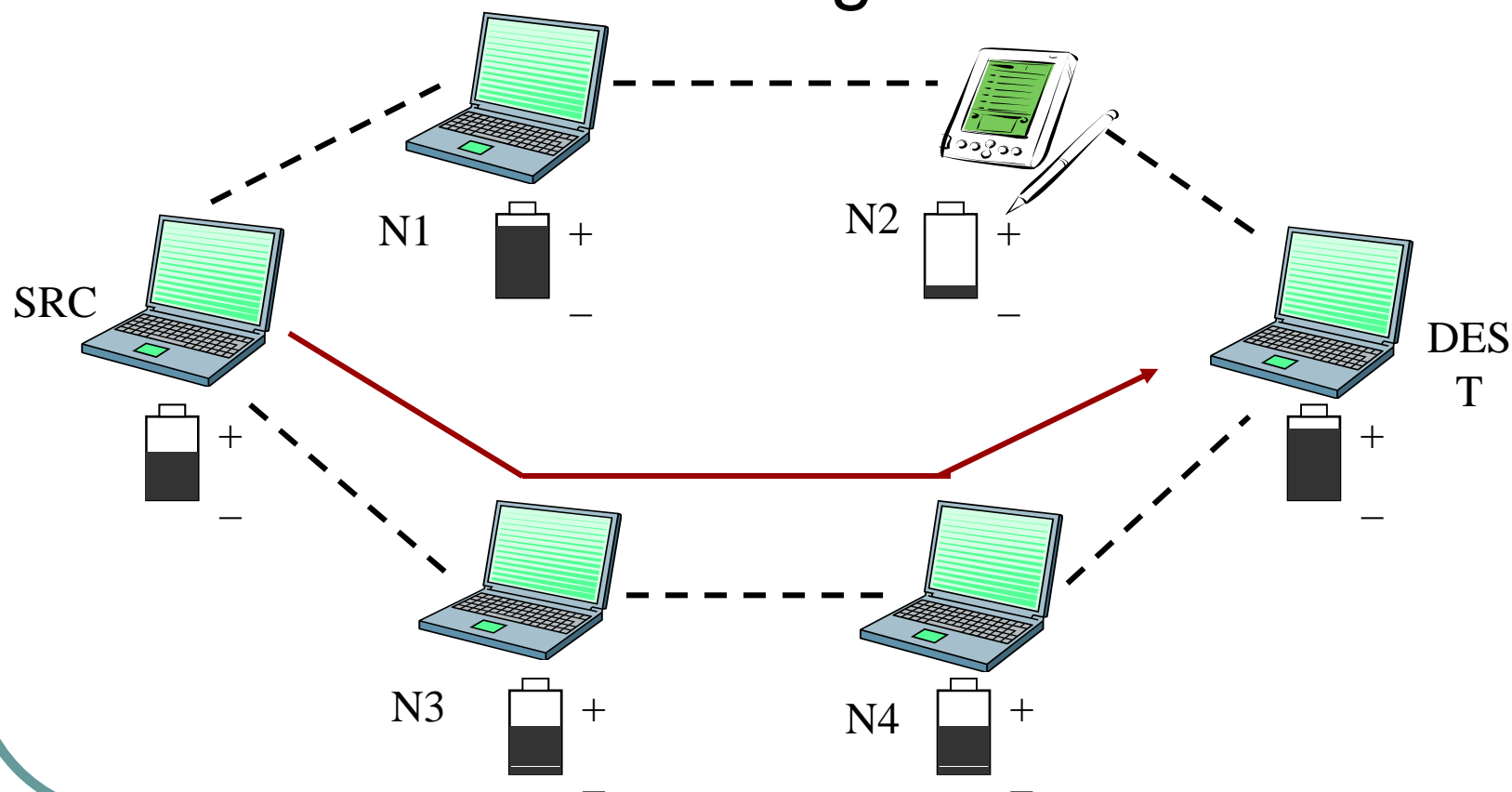
Figure 1: An Example Network



# Power-Aware Routing

# PAR

## ● Power-Aware Routing



## Homework #5:

- What's difference of proactive and reactive routing protocols in MANETs ?
- What's difference of DSR and AODV protocols ?
- What's the location-aware routing protocol in MANET ?
- What's power-aware routing protocol ?