

Wireless Networks II: Mesh Network Routing



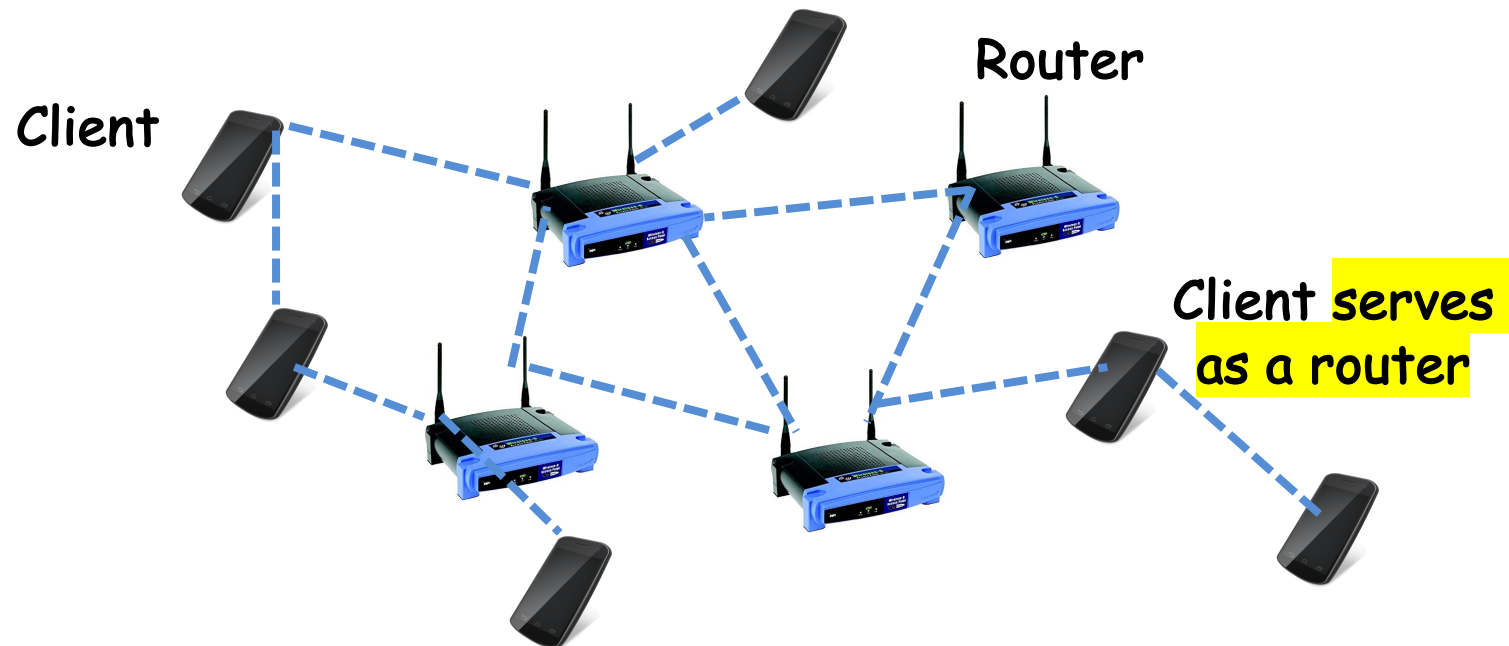
COS 461: Computer Networks
Lecture 19

Kyle Jamieson

[Parts adapted from I. F. Akyildiz, B. Karp]

Wireless Mesh Networks: Motivation

- Most wireless network traffic goes through **APs**
- Mesh networks **remove this restriction**
 - Multiple paths between most pairs: **Mesh topology**
- Big Impact: **Home Mesh, Satellite/Balloon Internet**

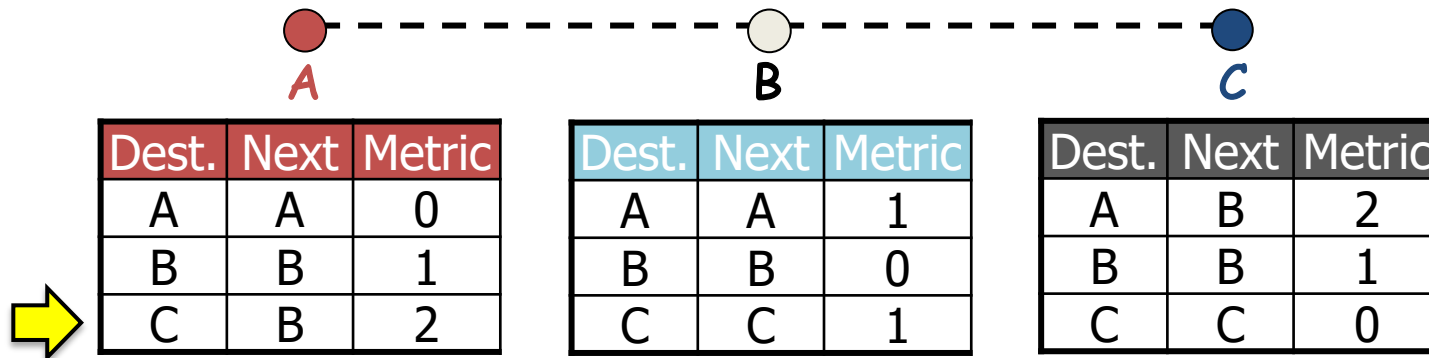


Today

1. Review Distance Vector Routing
 - New node join
 - Route changes
 - Broken link
2. Destination Sequenced Distance-Vector Routing (DSDV)
3. Dynamic Source Routing (DSR)

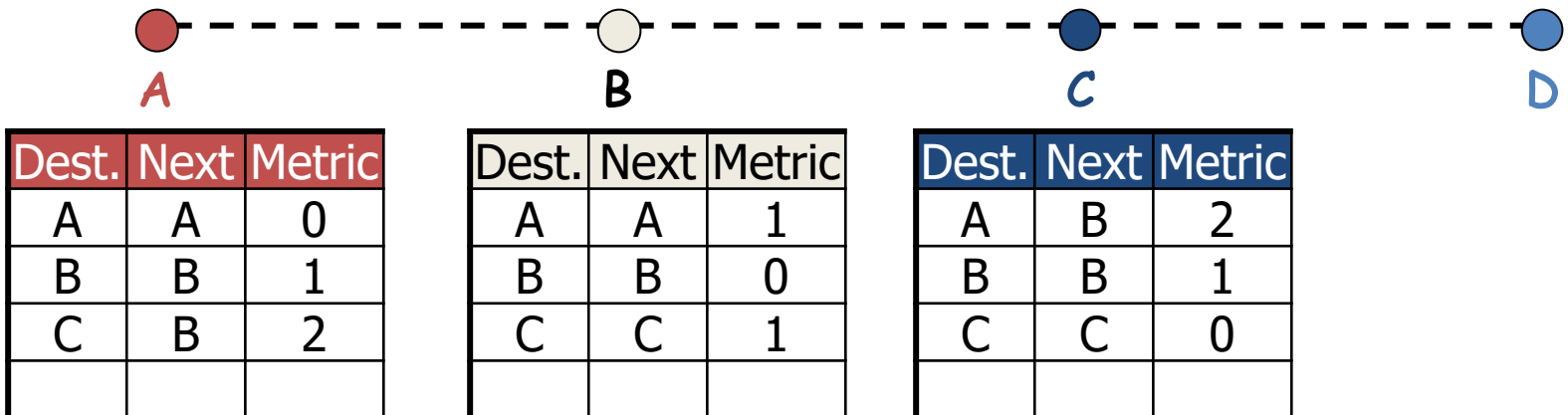
Distance Vector Routing: Review

- Every node maintains a **routing table**
 - For each *destination* node in the mesh:
 - The number of hops to reach the destination (**metric**)
 - The next node on the path towards the destination
- All nodes **periodically, locally broadcast** routing table, **learn about every destination in network**



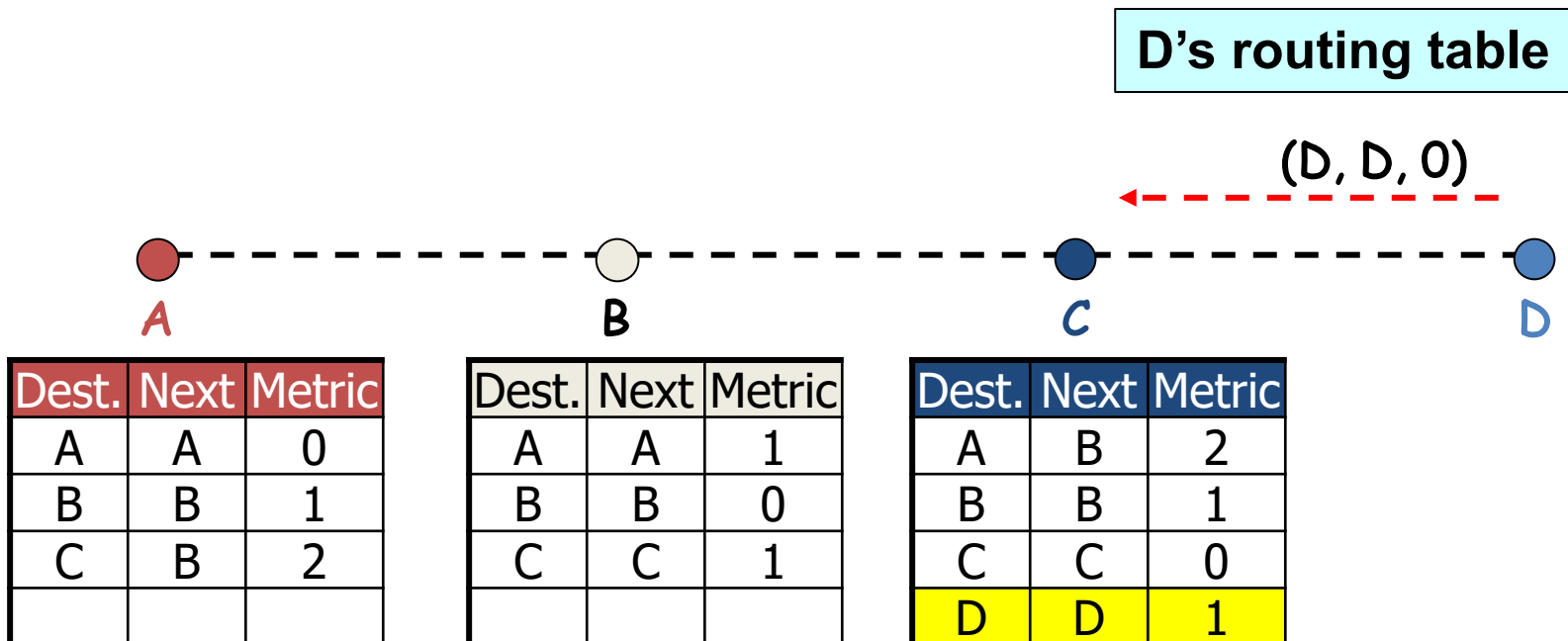
Distance Vector - New Node Join

- D joins the network



Distance Vector - New Node Join

- **D** joins the network
- D's broadcast first **updates C's table** with new entry for D

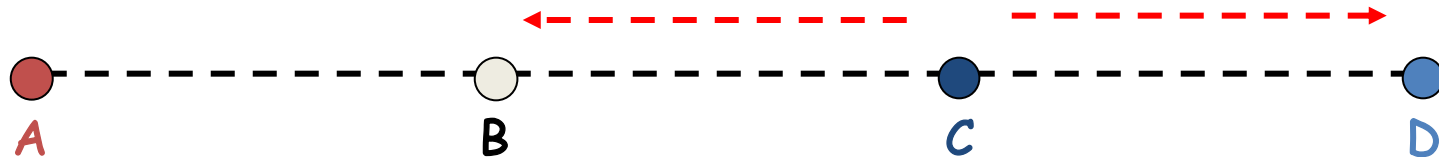


Distance Vector - New Node Join

- Now C broadcasts its routing table
 - B and D hear and **add new entries, incrementing metric**

C's routing table

(A, B, 2) (A, B, 2)
 (B, B, 1) (B, B, 1)
 (C, C, 0) (C, C, 0)
 (D, D, 1)



Dest.	Next	Metric
A	A	0
B	B	1
C	B	2

Dest.	Next	Metric
A	A	1
B	B	0
C	C	1
D	C	2

Dest.	Next	Metric
A	B	2
B	B	1
C	C	0
D	D	1

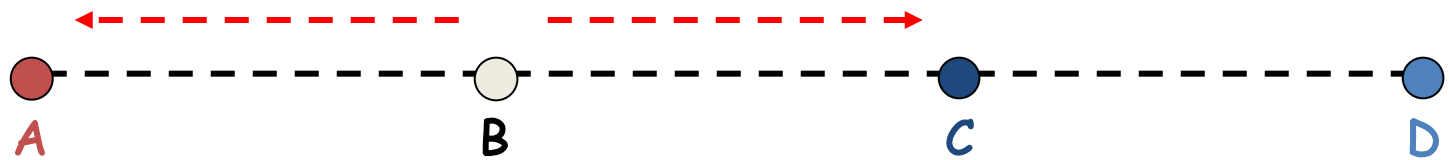
Dest.	Next	Metric
A	C	3
B	C	2
C	C	1
D	D	0

Distance Vector - New Node Join

- Now B broadcasts its routing table
 - A and C hear and **add new entries, if shorter route**

B's routing table

(A, A, 1) (A, A, 1)
 (B, B, 0) (B, B, 0)
 (C, C, 1) (C, C, 1)
 (D, C, 2) (D, C, 2)



Dest.	Next	Metric
A	A	0
B	B	1
C	B	2
D	B	3

Dest.	Next	Metric
A	A	1
B	B	0
C	C	1
D	C	2

Dest.	Next	Metric
A	B	2
B	B	1
C	C	0
D	D	1

Dest.	Next	Metric
A	C	3
B	C	2
C	C	1
D	D	0

Today

1. Review Distance Vector Routing

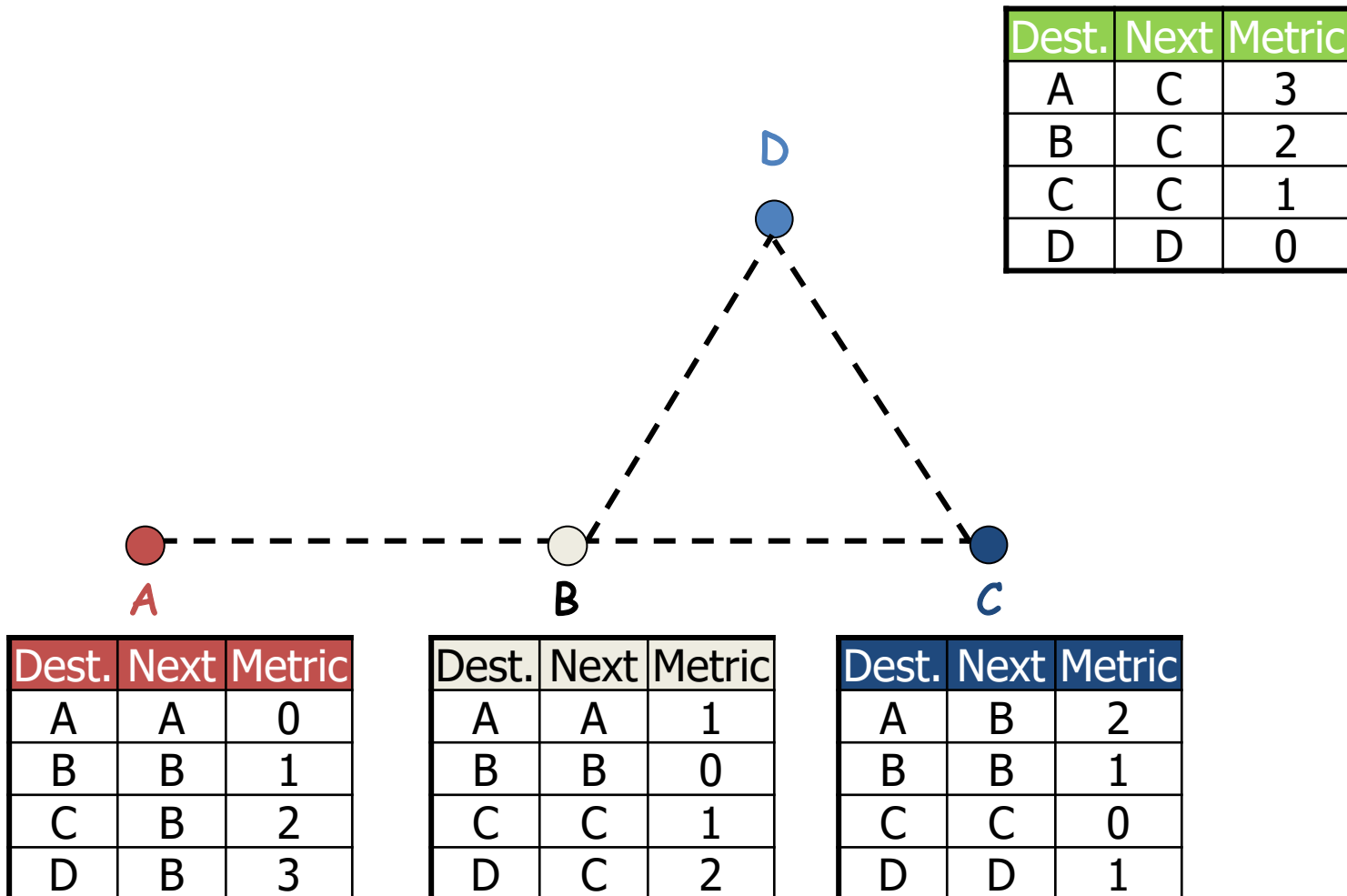
- New node join
- Route changes
- Broken link

2. Destination Sequenced Distance-Vector Routing (DSDV)

3. Dynamic Source Routing (DSR)

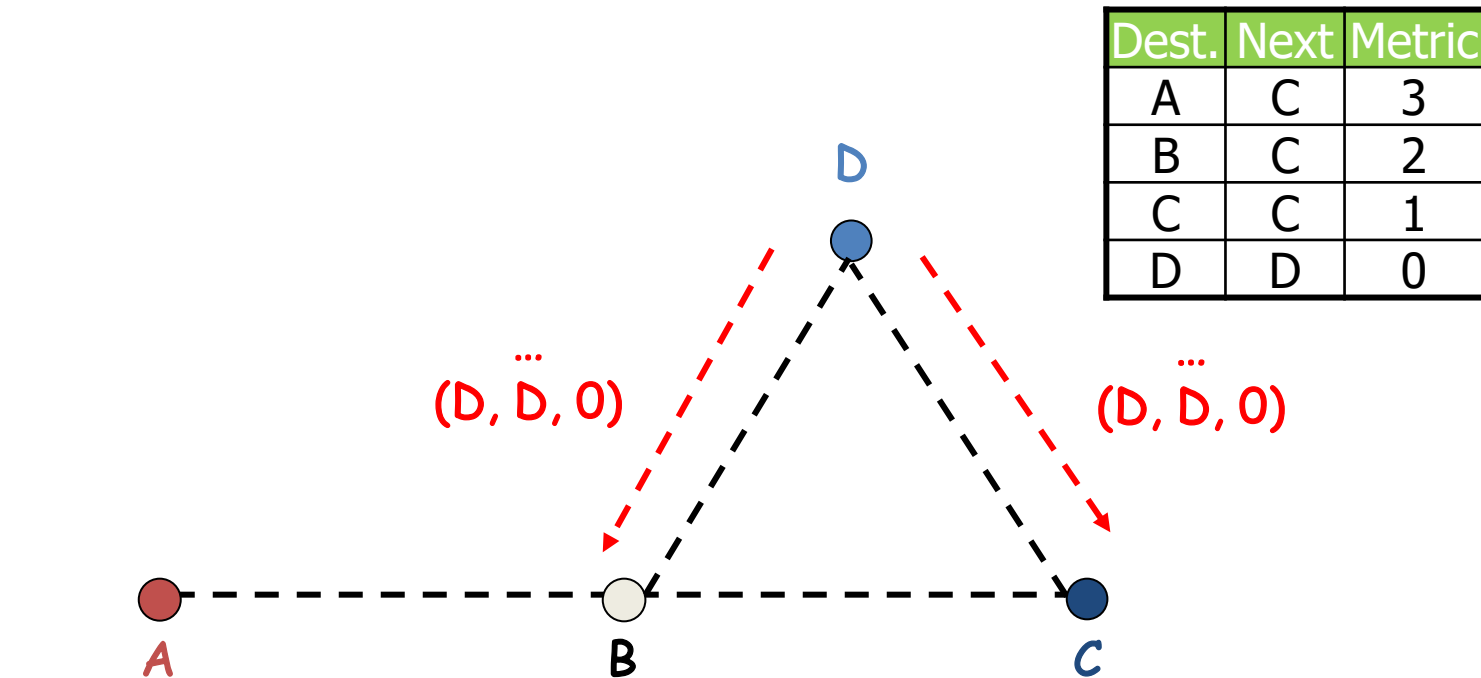
Distance Vector - Route Change

- **D** moves to another place and broadcast its routing table



Distance Vector - Route Change

- **D** moves to another place and broadcast its routing table



Dest.	Next	Metric
A	C	3
B	C	2
C	C	1
D	D	0

Dest.	Next	Metric
A	A	0
B	B	1
C	B	2
D	B	3

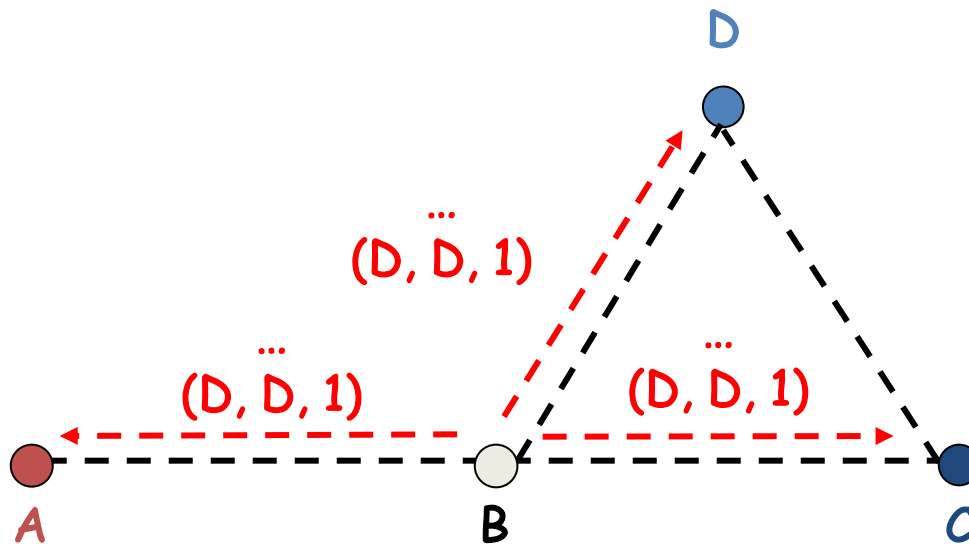
Dest.	Next	Metric
A	A	1
B	B	0
C	C	1
D	D	1

Dest.	Next	Metric
A	B	2
B	B	1
C	C	0
D	D	1

Distance Vector - Route Change

- **D** moves to another place and broadcast its routing table
- **B** broadcasts its routing table

Dest.	Next	Metric
A	B	2
B	B	1
C	C	1
D	D	0



Dest.	Next	Metric
A	A	0
B	B	1
C	B	2
D	B	2

Dest.	Next	Metric
A	A	1
B	B	0
C	C	1
D	D	1

Dest.	Next	Metric
A	B	2
B	B	1
C	C	0
D	D	1

Today

1. Review Distance Vector Routing

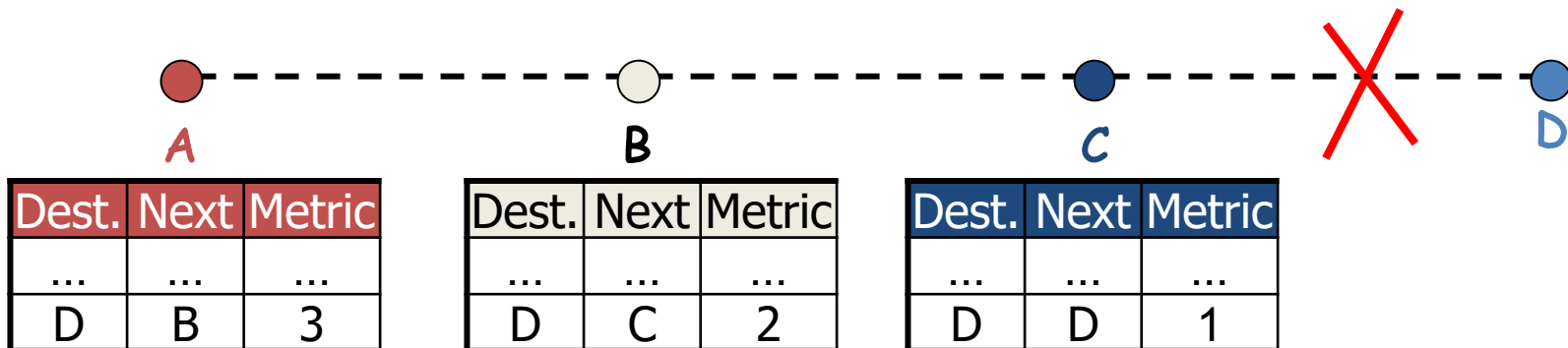
- New node join
- Route changes
- **Broken link**

2. Destination Sequenced Distance-Vector Routing (DSDV)

3. Dynamic Source Routing (DSR)

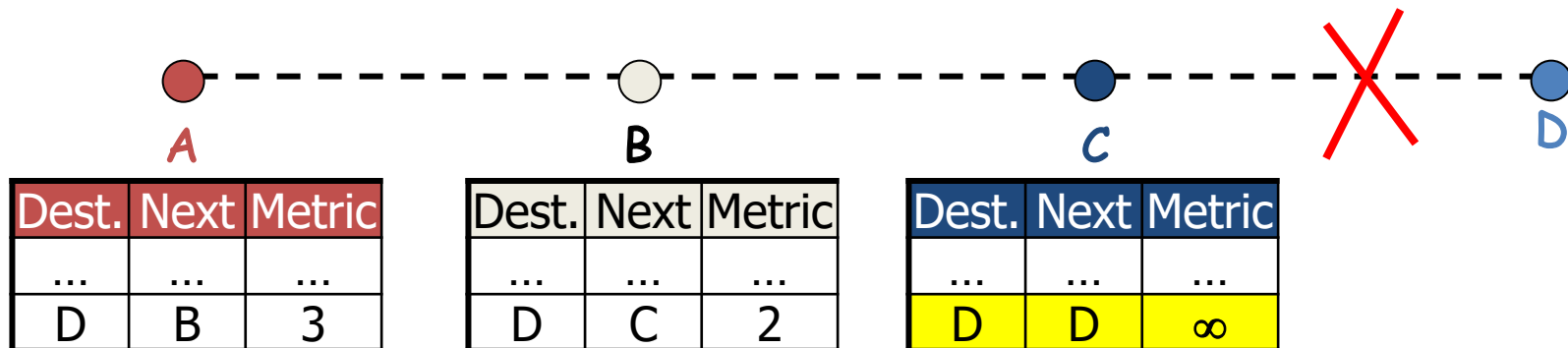
Distance Vector - Broken Link

- Suppose link $C \leftrightarrow D$ breaks



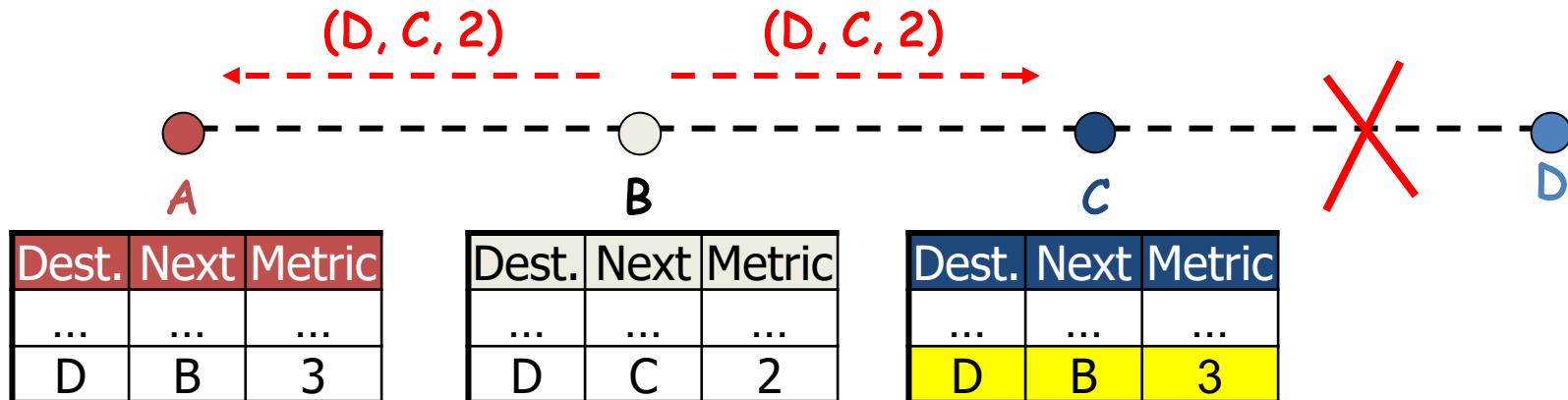
Distance Vector - Broken Link

1. **C hears no advertisement** from D for a *timeout period*
 - C sets D's metric to ∞



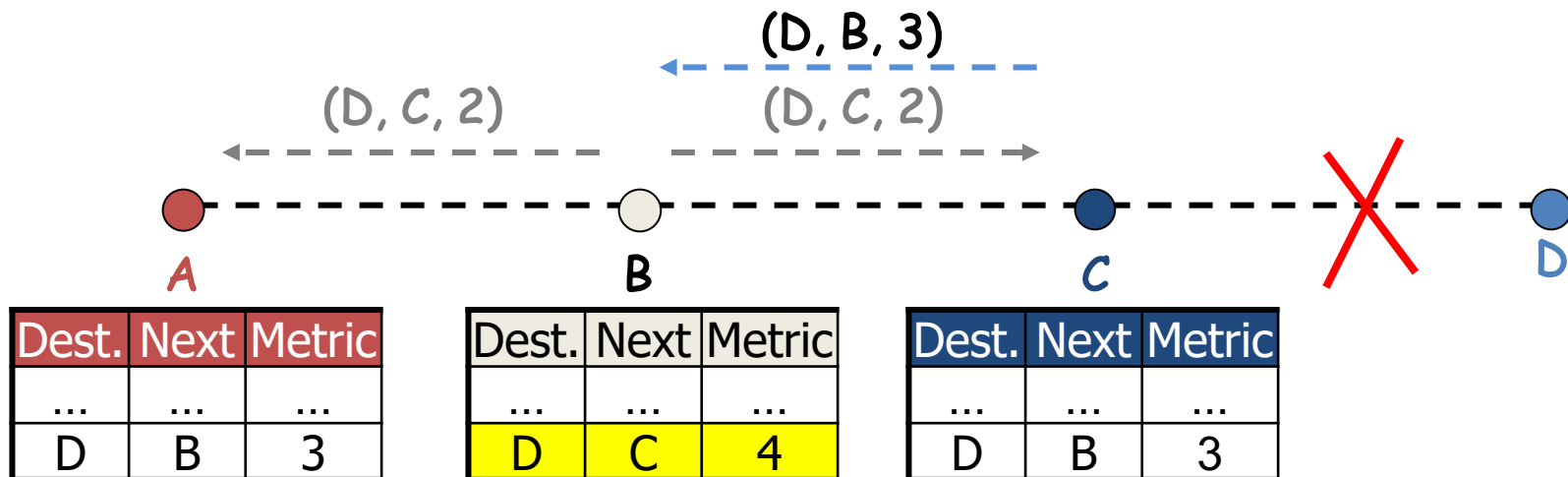
Distance Vector - Broken Link

1. C sets D's metric to ∞
2. B broadcasts its routing table
 - C now accepts B's entry for D ($3 < \infty$)



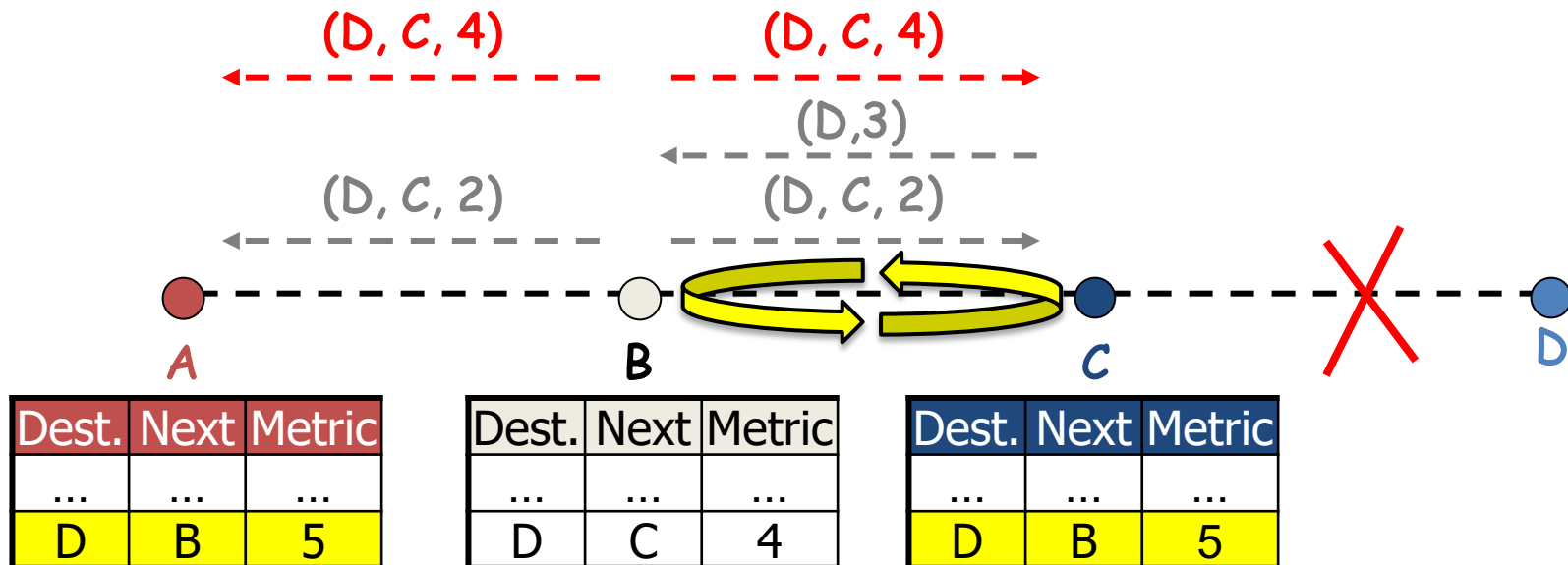
Broken Link: Counting to Infinity

1. C sets D's metric to ∞
2. B broadcasts its routing table
3. C broadcasts its routing table
 - B **accepts C's new metric** (B's previous next-hop was C)



Broken Link: Counting to Infinity

1. C sets D's metric to ∞
2. B broadcasts its routing table
3. C broadcasts its routing table
4. B broadcasts its routing table
 - A, C **accept** B's new metric (previous next-hops: B)



Today

1. Review Distance Vector Routing
2. Destination Sequenced Distance-Vector Routing (DSDV)
 - New node join
 - Broken link
 - Route change
3. Dynamic Source Routing (DSR)

Destination Sequenced Distance-Vector (DSDV) Routing

- Guarantees **loop freeness**
- **New routing table information: Sequence number**
 - Sequence number is **per-destination** information
 - Originated by destination
 - **Included and propagated** in routing advertisements

Destination	Next	Metric	Seq. Nr
A	A	0	550
B	B	1	102
C	B	3	588
D	B	4	312

DSDV: Route Advertisement Rule

Rules to set sequence number:

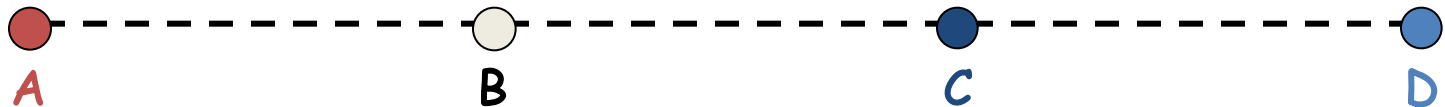
- Just before node N's broadcast advertisement:
 - Node N sets:
 - $Seq(N) \leftarrow Seq(N) + 2$
- Node N thinks **neighbor P is no longer directly reachable**
 - Node N sets:
 - $Seq(P) \leftarrow Seq(P) + 1$
 - $Metric(P) \leftarrow \infty$

DSDV - New Node

- D joins the network
- D's broadcast first updates C's table w/ new entry for D

1. D broadcast for first time
Send Sequence number 000

(D, D, 0, 000)



Dest.	Next	Metric	Seq.
A	A	0	550
B	B	1	104
C	B	2	590

Dest.	Next	Metric	Seq.
A	A	1	550
B	B	0	104
C	C	1	590

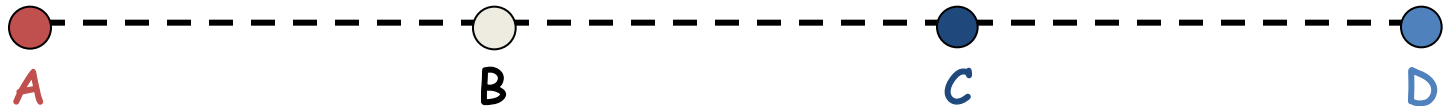
Dest.	Next	Metric	Seq.
A	B	2	550
B	B	1	104
C	C	0	590

DSDV - New Node

2.1: Insert entry for D with same sequence number 000
2.2: Triggered broadcast of its own table

1. D broadcast for first time
Send Sequence number 000

(D, D, 0, 000)

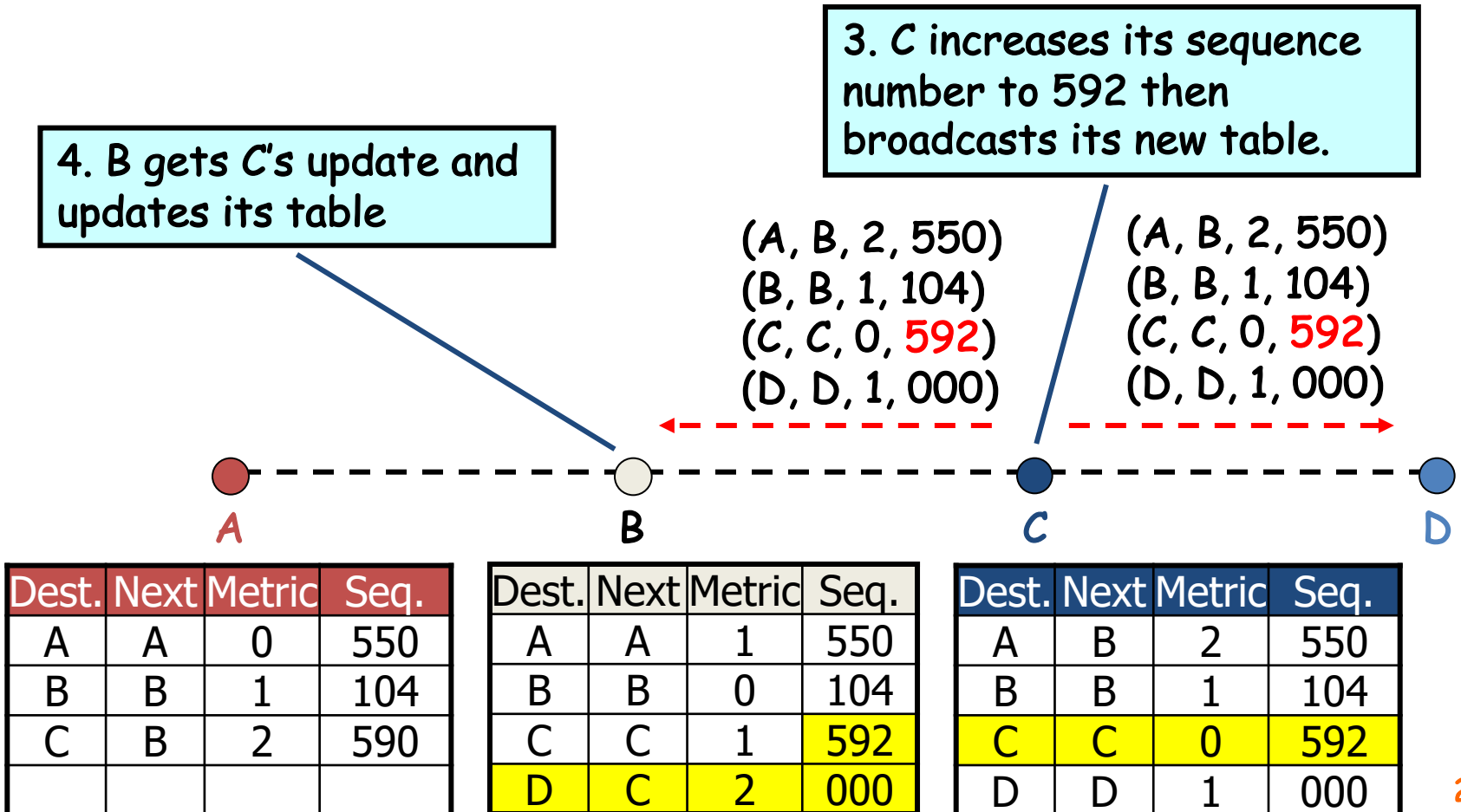


Dest.	Next	Metric	Seq.
A	A	0	550
B	B	1	104
C	B	2	590

Dest.	Next	Metric	Seq.
A	A	1	550
B	B	0	104
C	C	1	590

Dest.	Next	Metric	Seq.
A	B	2	550
B	B	1	104
C	C	0	590
D	D	1	000

DSDV - New Node

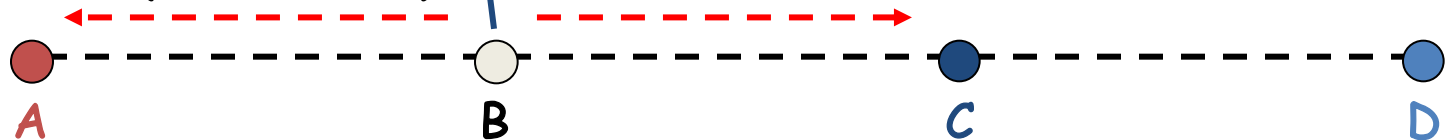


DSDV - New Node

4. B increases its own seqno and broadcasts its new table

(A, B, 1, 550)
 (B, B, 0, **106**)
 (C, C, 1, 592)
 (D, C, 2, 000)

(A, B, 1, 550)
 (B, B, 0, **106**)
 (C, C, 1, 592)
 (D, C, 2, 000)



Dest.	Next	Metric	Seq.
A	A	0	550
B	B	1	106
C	B	2	592
D	B	3	000

Dest.	Next	Metric	Seq.
A	A	1	550
B	B	0	106
C	C	1	592
D	C	2	000

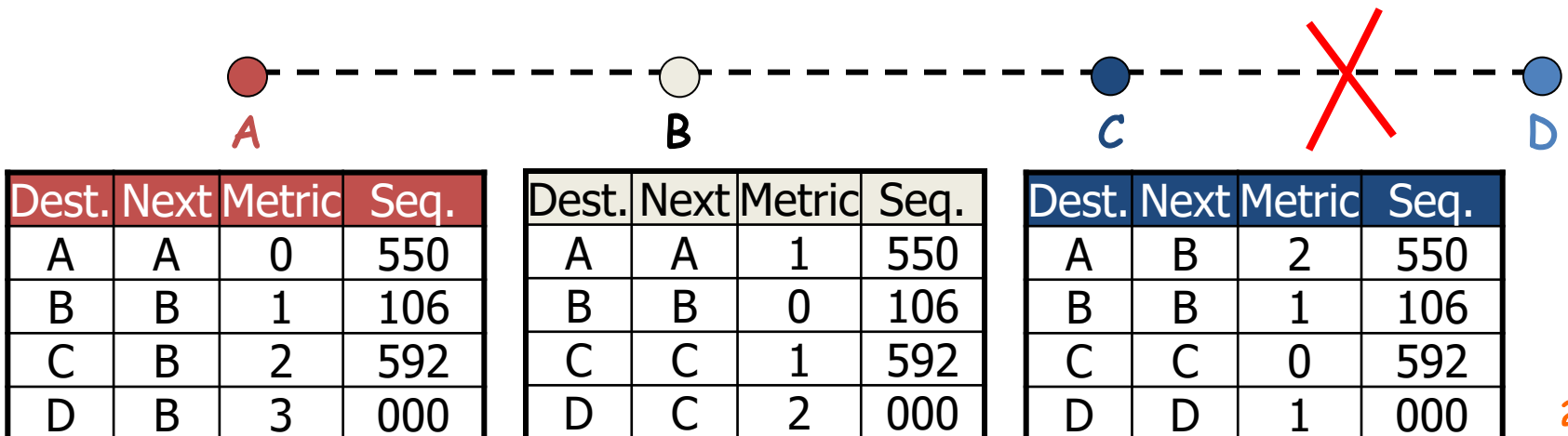
Dest.	Next	Metric	Seq.
A	B	2	550
B	B	1	106
C	C	0	592
D	D	1	000

Today

1. Review Distance Vector Routing
2. **Destination Sequenced Distance-Vector Routing (DSDV)**
 - New node join
 - **Broken link**
 - Route change
3. Dynamic Source Routing (DSR)

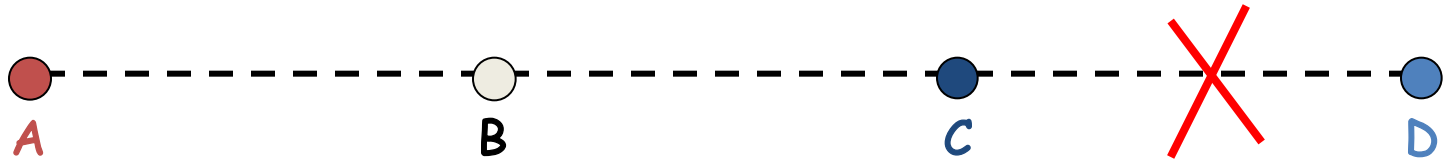
DSDV – Broken Link

- Suppose link $C \leftrightarrow D$ **breaks**



DSDV – Broken Link

1. Node C detects broken Link:
→ Increase Seq. No. by 1
(only case where not the destination sets the sequence number → odd number)



Dest.	Next	Metric	Seq.
A	A	0	550
B	B	1	106
C	B	2	592
D	B	3	000

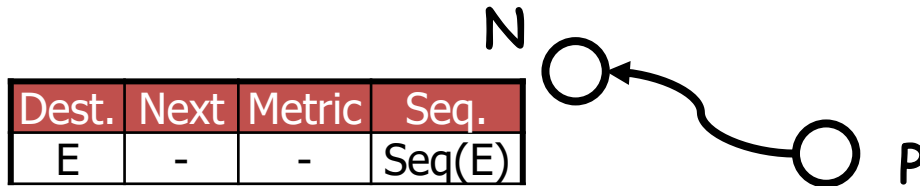
Dest.	Next	Metric	Seq.
A	A	1	550
B	B	0	106
C	C	1	592
D	C	2	000

Dest.	Next	Metric	Seq.
A	B	2	550
B	B	1	106
C	C	0	592
D	D	∞	001

DSDV: Routing Table Update Rule

Rules to update routing table entry:

- Node N gets routing advertisement from neighbor Node P:

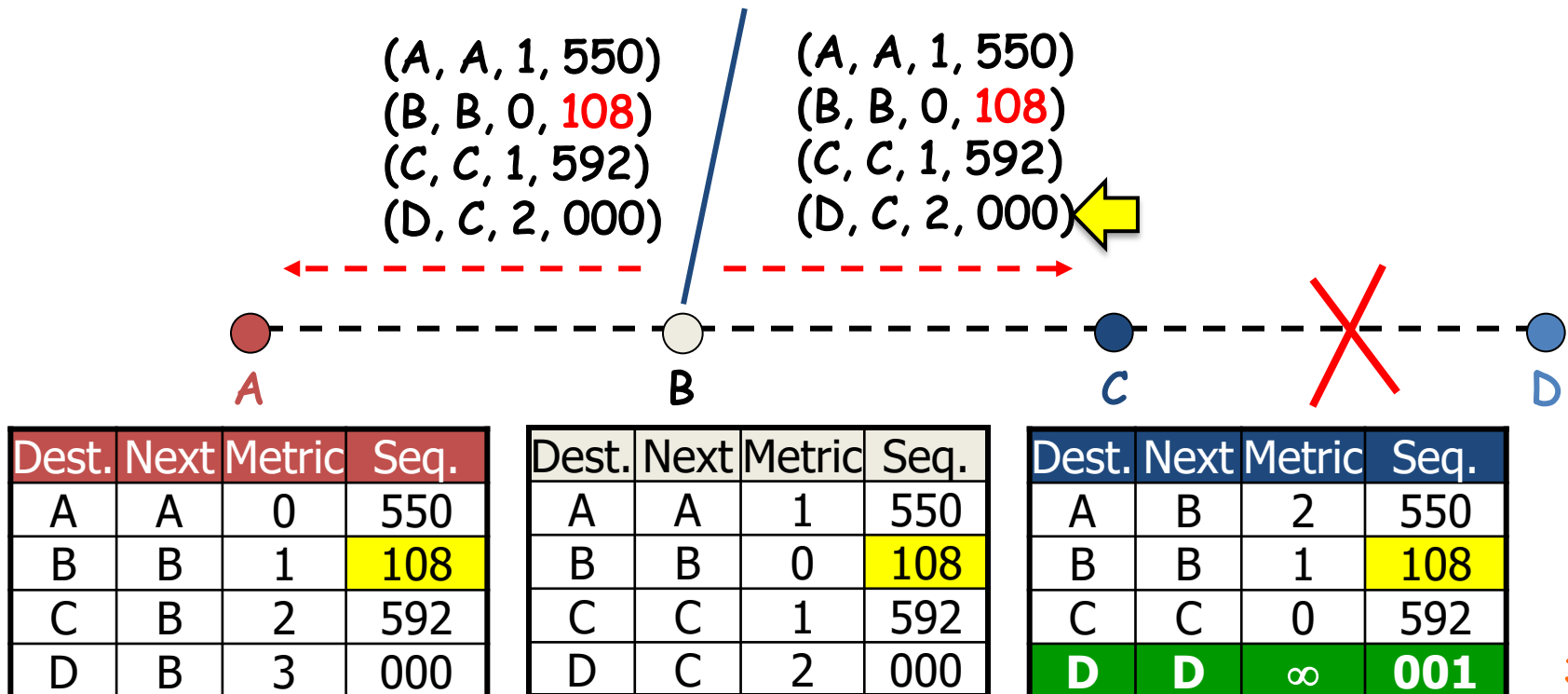


- Update routing table entry for node E when:
 - Seq(E) in P's advertisement > Seq(E) in N's table

DSDV - Broken Link

- B next broadcasts its routing table

- No effect on C's entry for D
(because 000 isn't greater than 001)
- No loop → no count to infinity



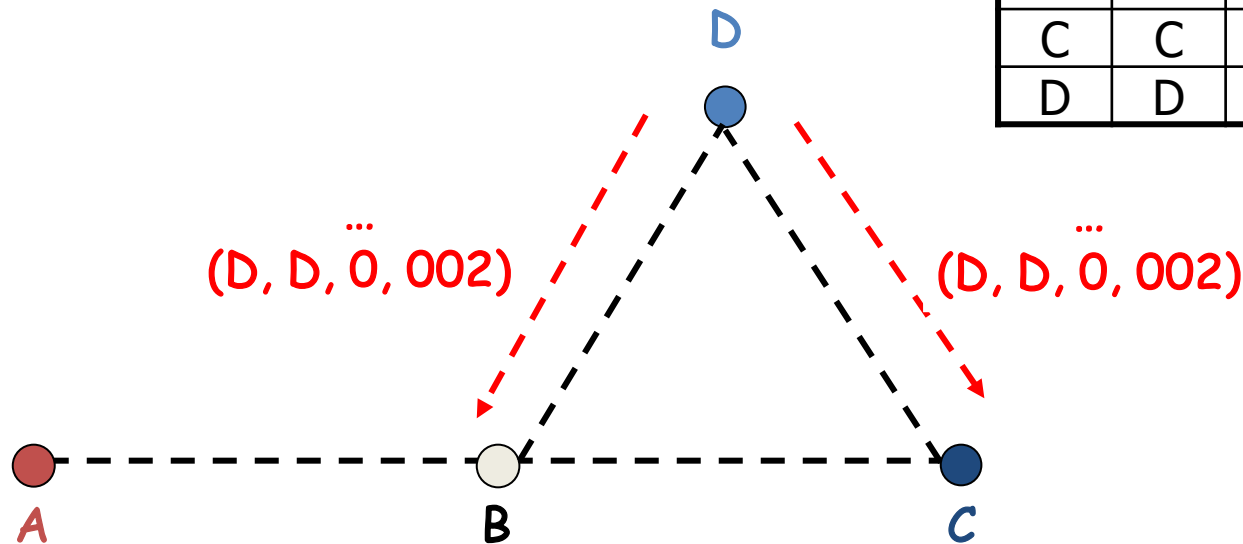
Today

1. Review Distance Vector Routing
2. **Destination Sequenced Distance-Vector Routing (DSDV)**
 - New node join
 - Broken link
 - **Route change**
3. Dynamic Source Routing (DSR)

Distance Vector - Route Change

- **D** moves to another place and broadcasts its routing table

Dest.	Next	Metric	Seq.
A	C	3	550
B	C	2	108
C	C	1	592
D	D	0	002



Dest.	Next	Metric	Seq.
A	A	0	550
B	B	1	108
C	B	2	592
D	B	3	000

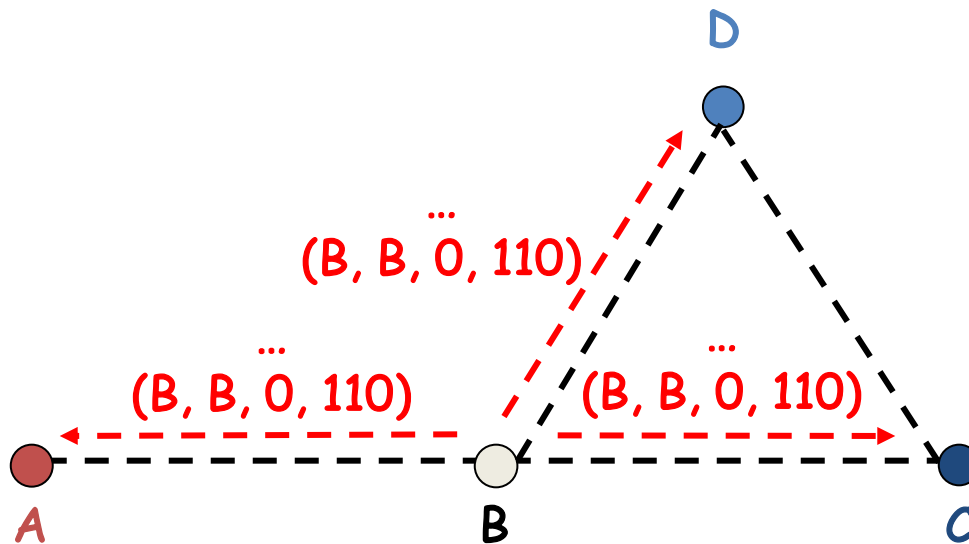
Dest.	Next	Metric	Seq.
A	A	1	550
B	B	0	108
C	C	1	592
D	D	1	002

Dest.	Next	Metric	Seq.
A	B	2	550
B	B	1	108
C	C	0	592
D	D	1	002

Distance Vector - Route Change

- **D** moves to another place and broadcasts its routing table
- **B** broadcasts its routing table

Dest.	Next	Metric	Seq.
A	B	2	550
B	B	1	110
C	C	1	592
D	D	0	002



Dest.	Next	Metric	Seq.
A	A	0	550
B	B	1	110
C	B	2	592
D	B	2	002

Dest.	Next	Metric	Seq.
A	A	1	550
B	B	0	110
C	C	1	592
D	D	1	002

Dest.	Next	Metric	Seq.
A	B	2	550
B	B	1	110
C	C	0	592
D	D	1	002

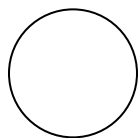
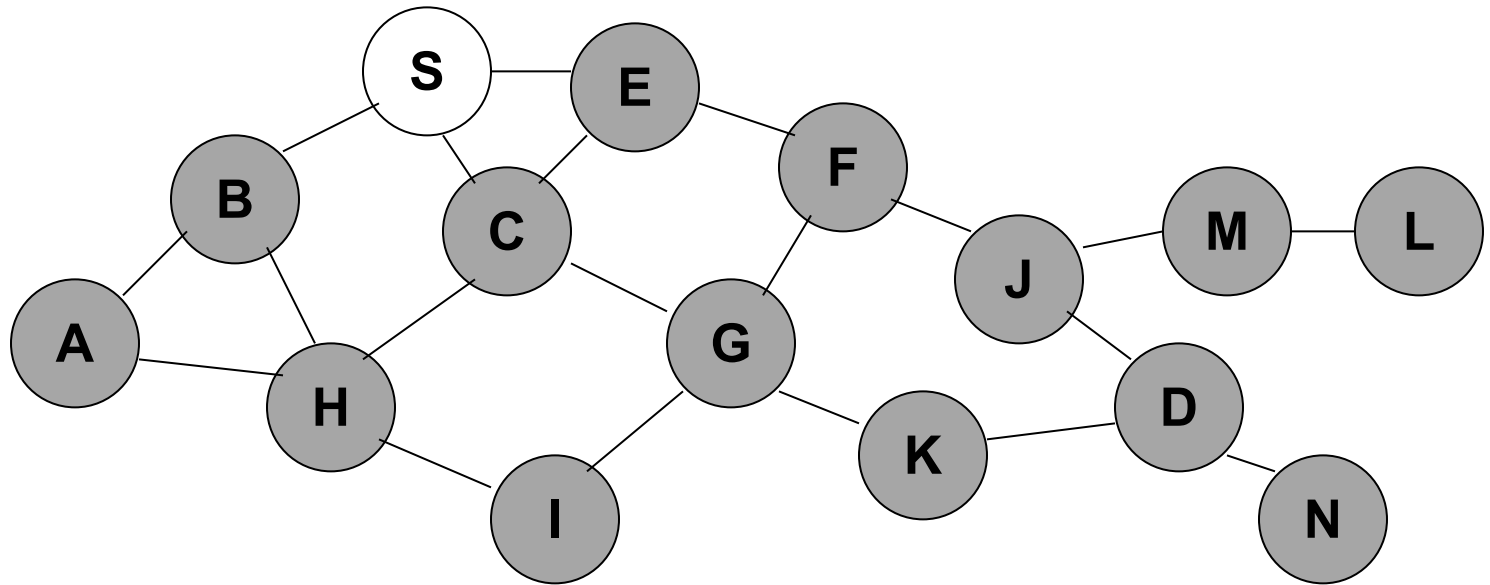
Today

1. Review Distance Vector Routing
2. Destination Sequenced Distance-Vector Routing (DSDV)
3. **Dynamic Source Routing (DSR)**

Dynamic Source Routing (DSR)

- No periodic “beaconing” from all nodes
- When node *S* wants to send a packet to node *D* (but doesn't know a route to *D*), *S* initiates a route discovery
- *S* network-floods a *Route Request (RREQ)*
 - Each node appends its own id when forwarding RREQ

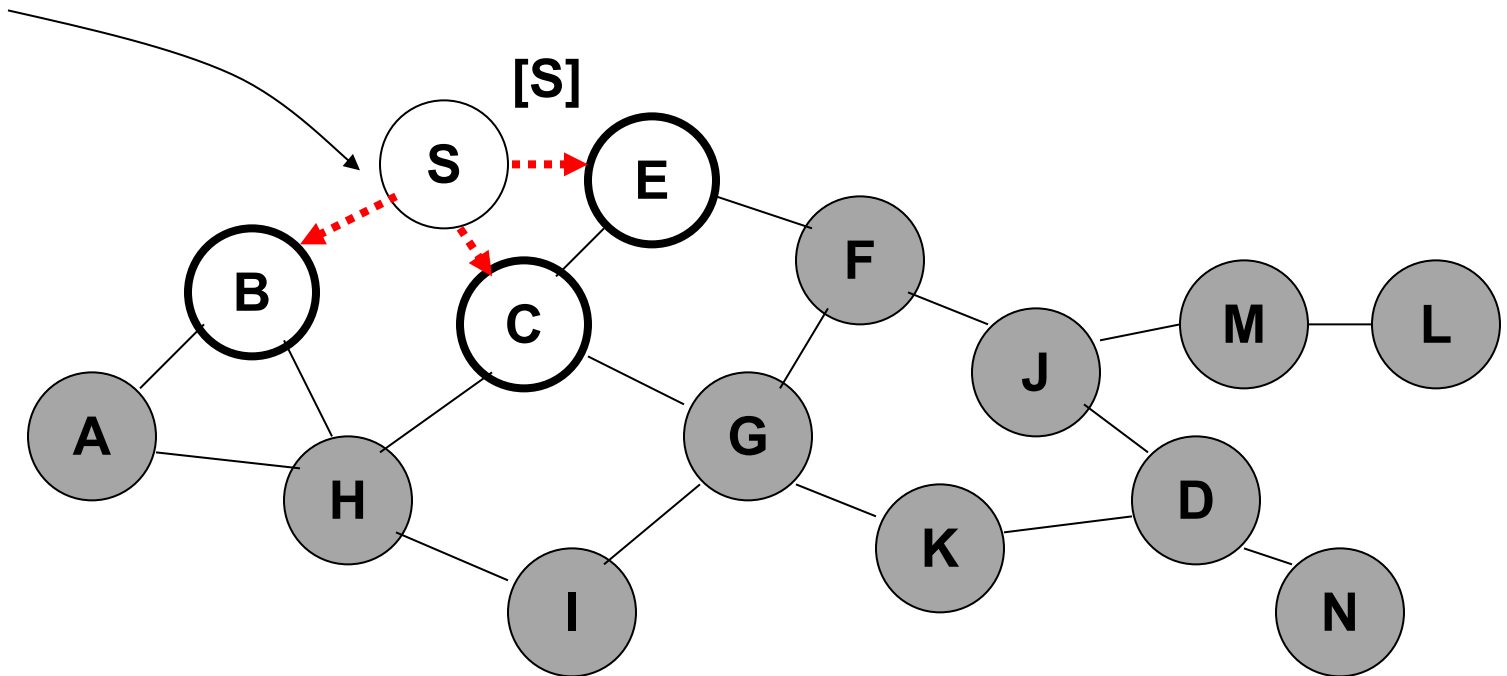
Route Discovery in DSR



Represents a node that has received RREQ for D from S

Route Discovery in DSR

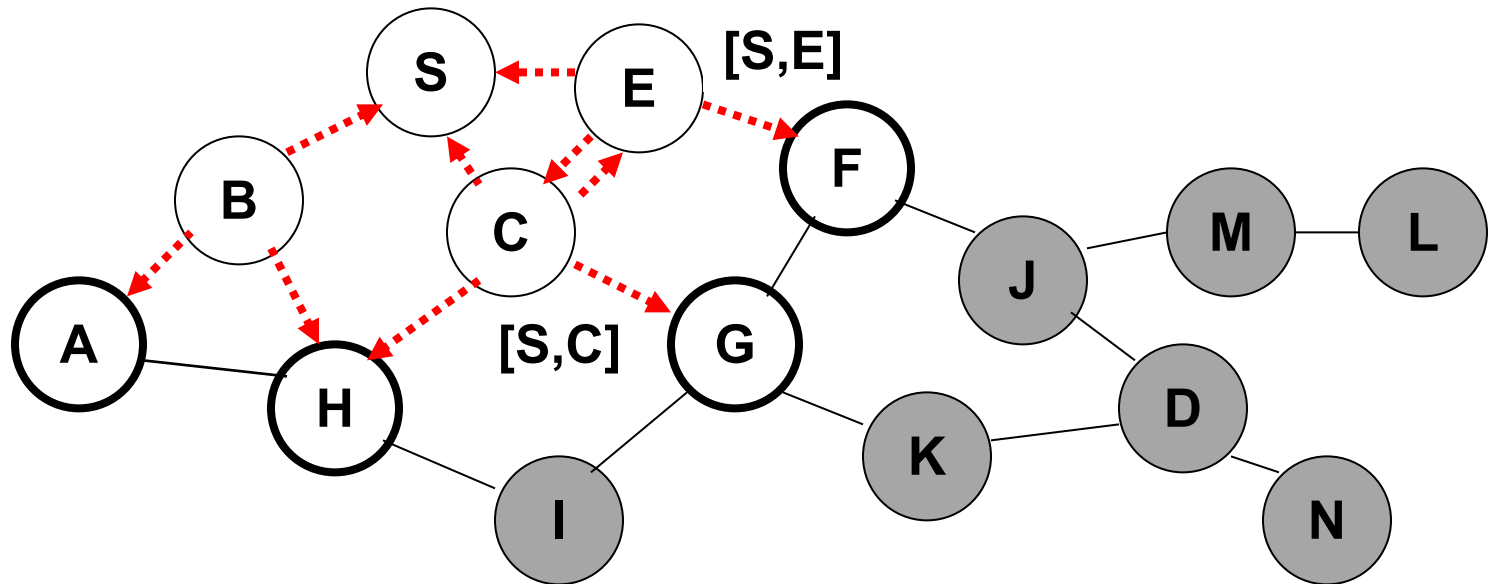
Broadcast transmission



.....→ Represents transmission of RREQ

[X,Y] Represents list of identifiers appended to RREQ

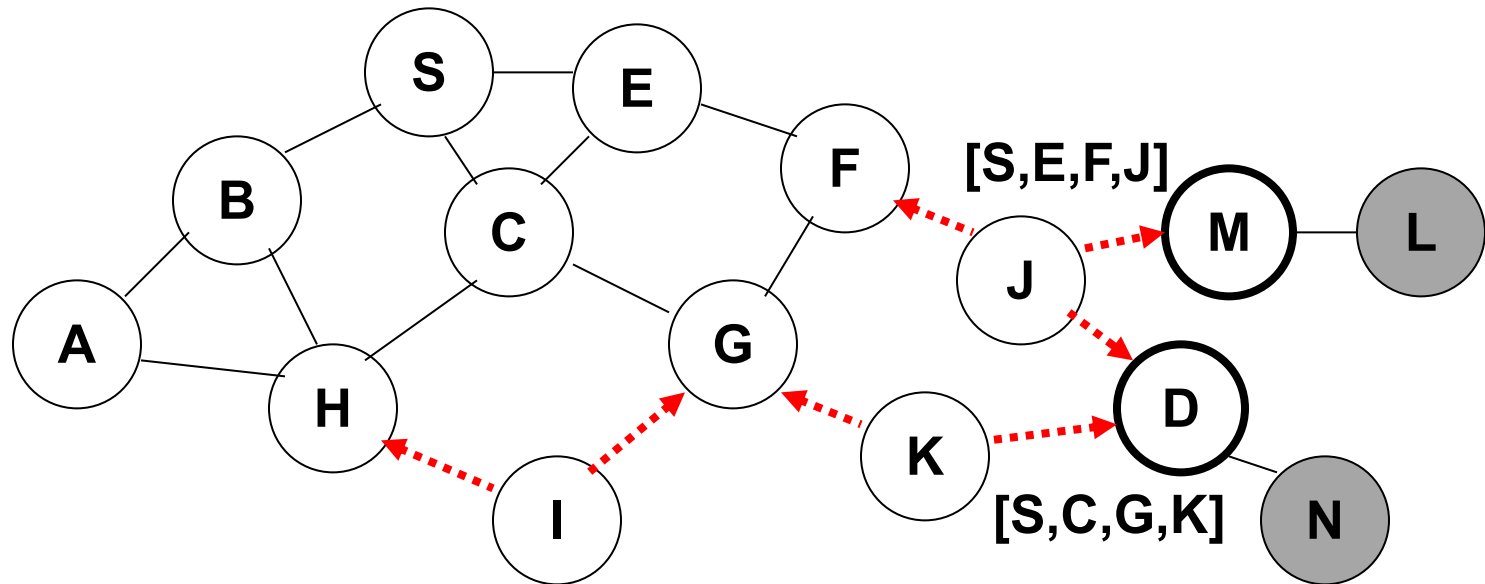
Route Discovery in DSR



.....→ Represents transmission of RREQ

[X,Y] Represents list of identifiers appended to RREQ

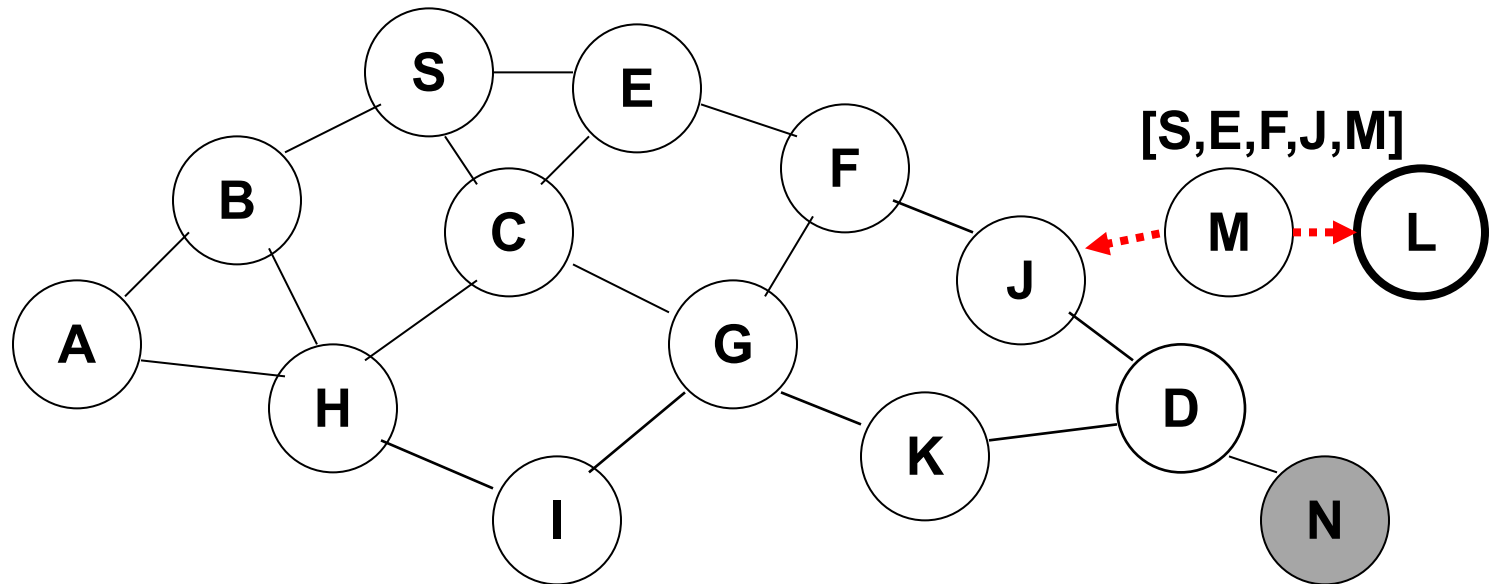
Route Discovery in DSR



.....→ Represents transmission of RREQ

- Nodes J and K both broadcast RREQ to node D
- Since nodes J and K are **hidden** from each other, their **transmissions may collide**

Route Discovery in DSR

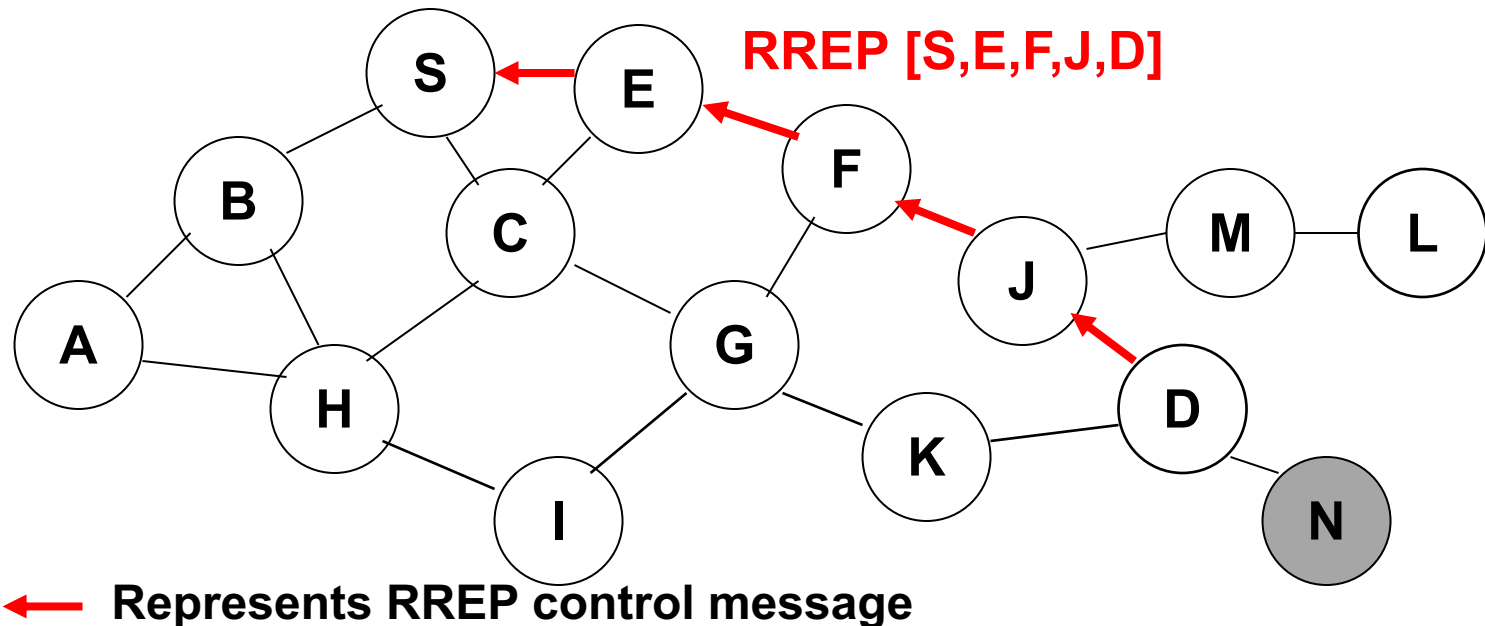


.....→ Represents transmission of RREQ

- Node D **does not forward** RREQ, because node D is the **intended target** of the route discovery

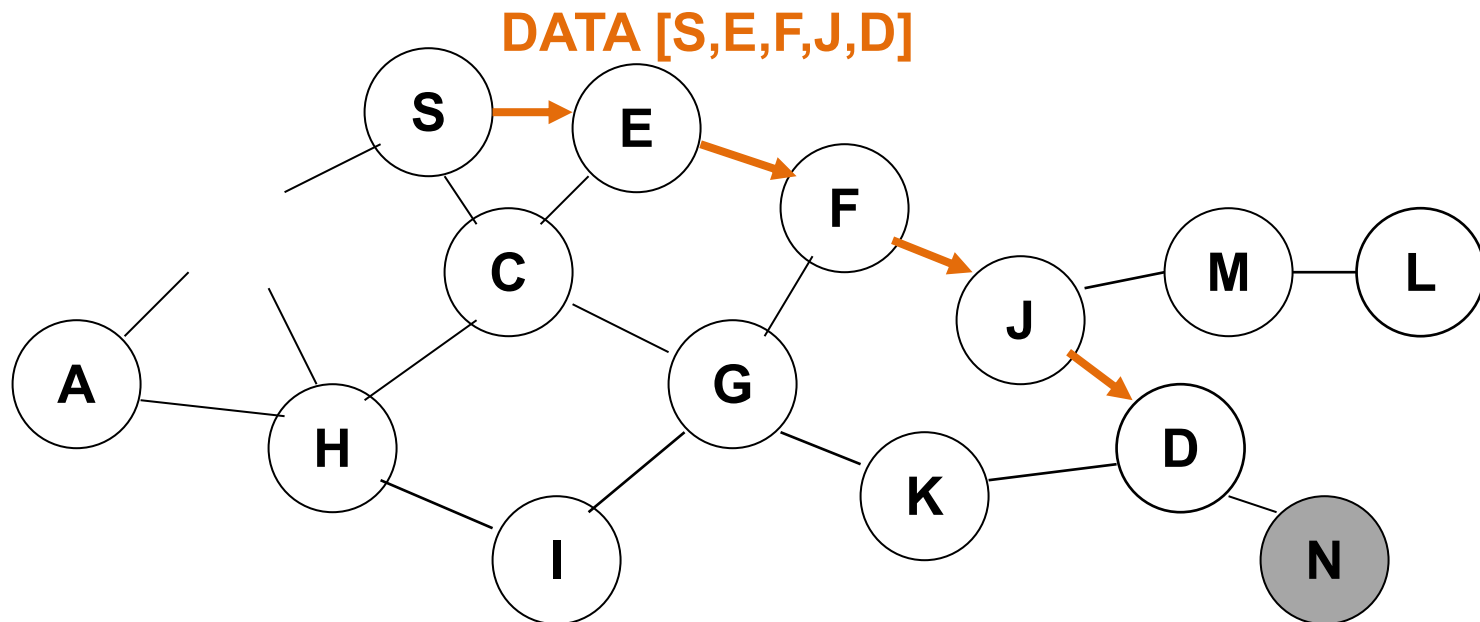
Route Reply in DSR

- On receiving first RREQ, D sends a *Route Reply (RREP)*
 - RREP sent on route obtained by *reversing* the route in the received RREQ
 - RREP *includes the route* from S to D over which D received the RREQ



Dynamic Source Routing (DSR)

- On receiving RREP, S caches route included therein
- When S sends a data packet to D, includes entire route in packet header
- Intermediate nodes use the **source route** included in packet to determine to whom packet should be forwarded



Summary

- DV reacts **poorly** to link failures, which are frequent in wireless
- DSDV is a **proactive** routing protocol, DSR **reactive**
- **Enable** wireless mesh routing, w/appl. in **recent/future**
 - Home mesh products
 - Satellite internet service providers
 - Balloon/UAV internet service providers