Basic Routing Concepts

Surasak Sanguanpong
nguan@ku.ac.th
http://www.cpe.ku.ac.th/~nguan

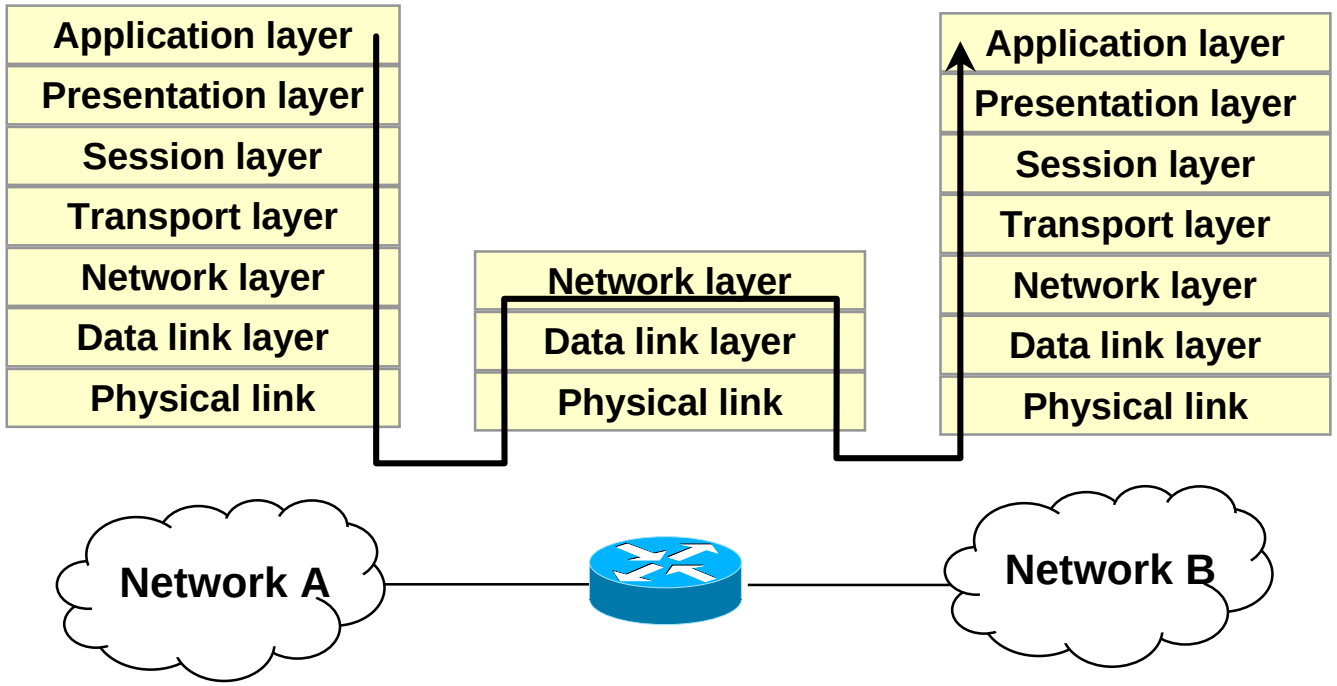
Last updated: July 30, 1999

Agenda

- Basic concepts
- Routing components
- Classes of routing protocol

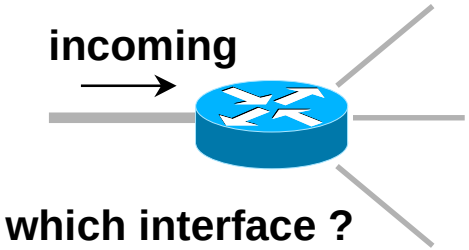
What's routing

- routing - path finding from one end to the other
  - routing occurs at layer 3
  - bridging occurs at layer 2



IP Routing

- IP performs:
  - search for a matching host address
  - search for a matching network address
  - search for a default entry



- Routing done by IP router, when it searches the routing table and decide which interface to end a packet out.

Routing Tables

- Routing is carried out in a router by consulting routing table.
- No unique format for routing tables, typically table contains:
  - address of a destination
  - IP address of next hop router
  - network interface to be used
  - subnet mask for the this interface
  - distance to the destination

Routing Component

- three important routing elements :
  - algorithm
  - database
  - protocol
- algorithm : can be differentiate based on several key characteristics
- database : table in routers or routing table
- protocol: the way information for routing to be gathered and distributed

Routing algorithm

- design goals
  - optimality - compute the best route
  - simplicity/low overhead - efficient with a minimum software and utilization overhead
  - robustness/stability- perform correctly in the face of unusual circumstances
  - rapid convergence- responds quickly when the network changes
  - flexibility- accurate adapt to a variety of network

Routing Protocols

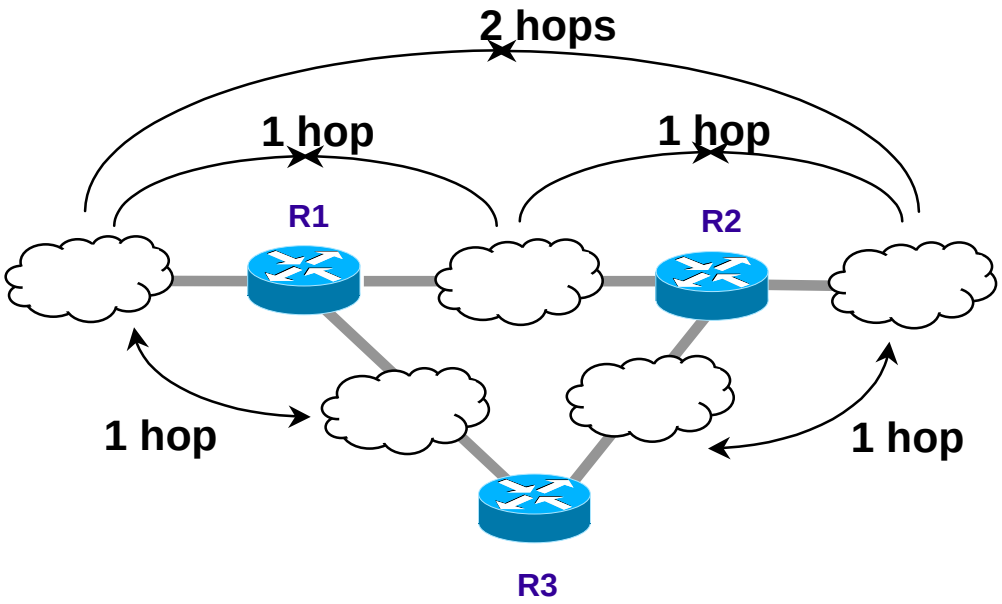
- **Routing protocol - protocol to exchange of information between routers about the current state of the network**
- **routing protocol jobs**
  - **create routing table entries**
  - **keep routing table up-to-date**
  - **compute the best choice for the next hop router**

Routing metrics

- How do we decide that one route is better than another?
- Solution : using a metric as a measurement to compare routes
- Metrics may be distance, throughput, delay, error rate, and cost.
- Today IP supports Delay, Throughput, Reliability and Cost (DTRC)

Hop Count

- A hop is defined as a passage through one router
- For some protocols, hop count means the number of links, rather than the number of routers

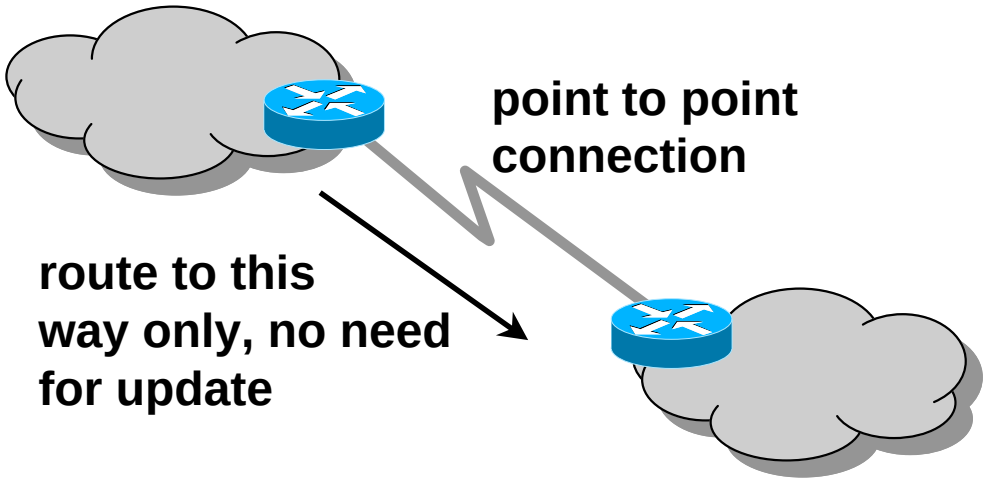


Routing algorithm types

- static V.S. dynamic
- source routing V.S. hop-by-hop
- centralize V.S. distributed
- distance vector V.S. link state

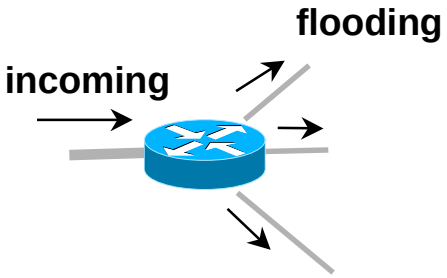
Routing algorithm: static route

- manually config routing table
- can't react dynamically to network change such as router's crash
- work well with small network or simple topology
- unix hosts use command route to add an entry



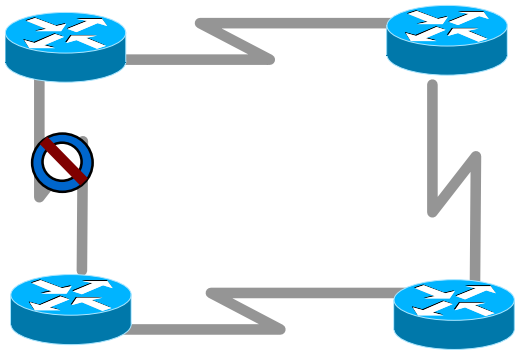
Routing algorithm: static technique

- **Flooding**
  - every incoming packet is sent out every outgoing
  - retransmit on all outgoing at each node
  - simple technique, require no network information
  - generate vast numbers of duplicate packet



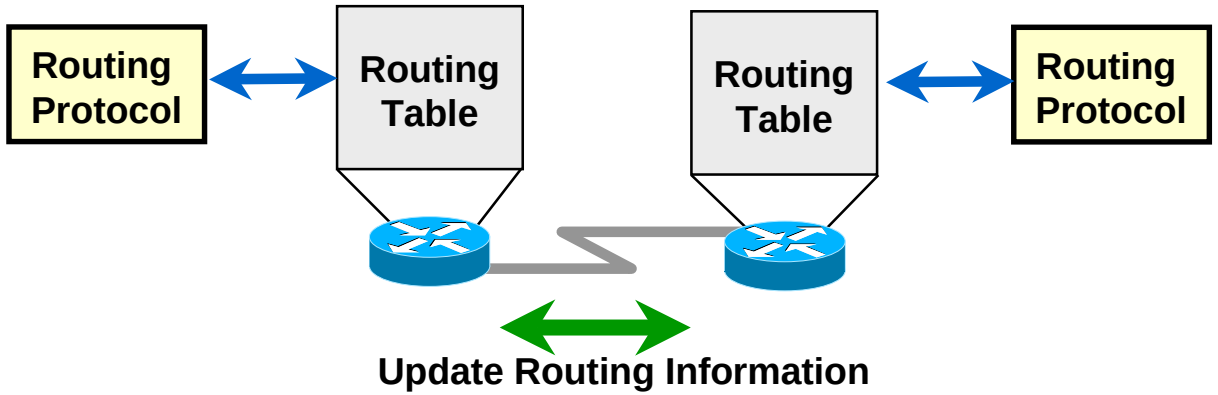
Routing algorithm: Dynamic Route

- **Dynamic route**
  - **network protocol adjusts automatically for topology or traffic changes**
  - **unix hosts run routing daemon** routed **or** gated



Routing algorithm: Dynamic Route operation

- **Routing protocol maintains and distributes routing information**



Routing algorithm: source routing

- source routing
  - source will determine the entire route
  - routers only act as sore-forward devices
- hop-by-hop
  - routers determine the path based on theirs own calculation

Routing algorithm: distance vector

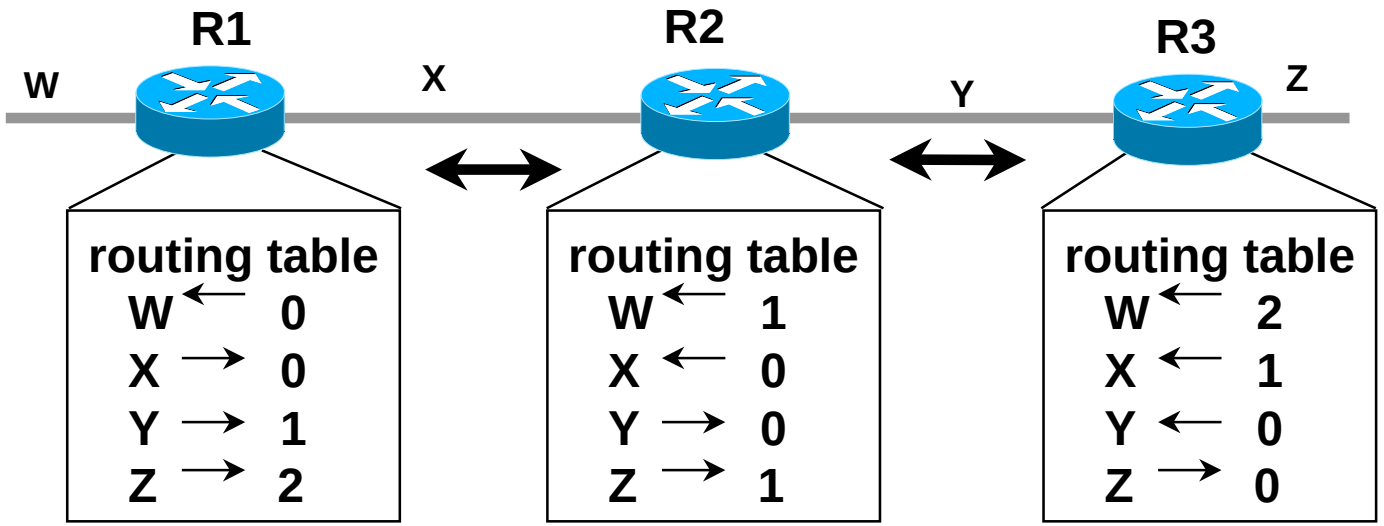
- **distance** means routing metric
- **vector** means destination
- flood routing table only to its neighbors
- RIP is an example
- also known as **Bellmann-Ford algorithm** or **Ford-Fulkerson algorithm**

Routing algorithm: link state

- flood routing information to all nodes
- each router finds who is up and flood this information to the entire routers
- use the link state to build a shortest path map to everybody
- OSPF is an example
- also known as Shortest Path First (SPF) algorithm

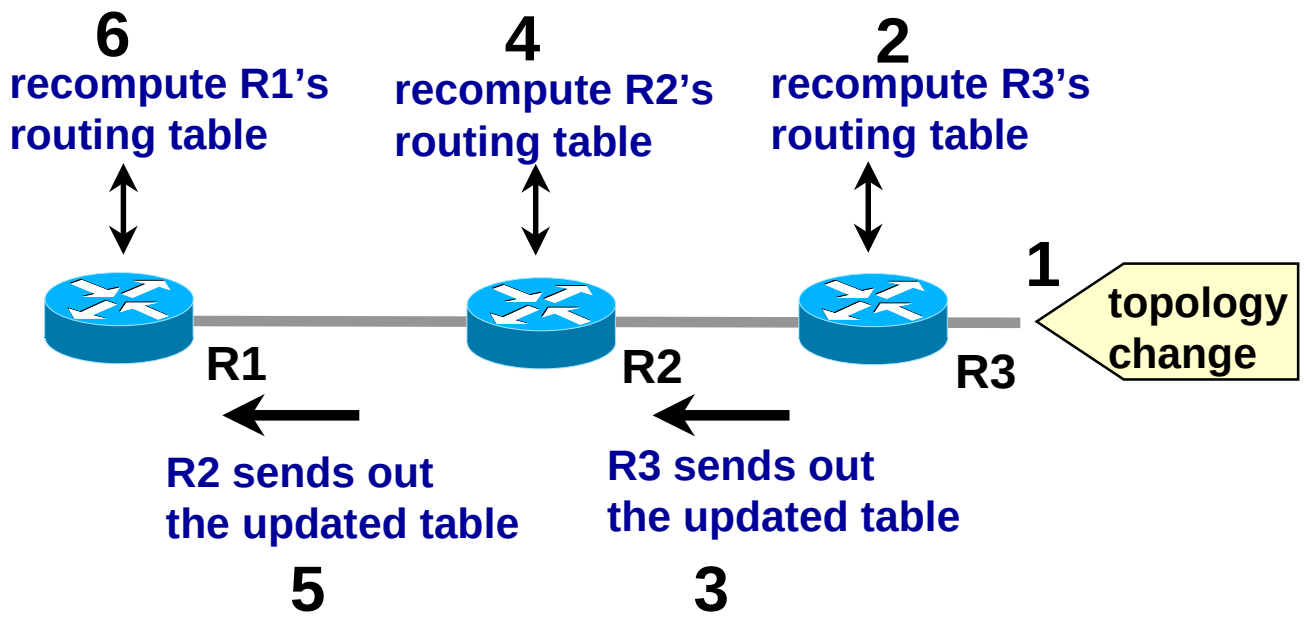
Distance vector algorithm

- using hop count as a metric
- each router periodically sends a copy of its routing table to neighbors
  - send <network X, hopcount Y>



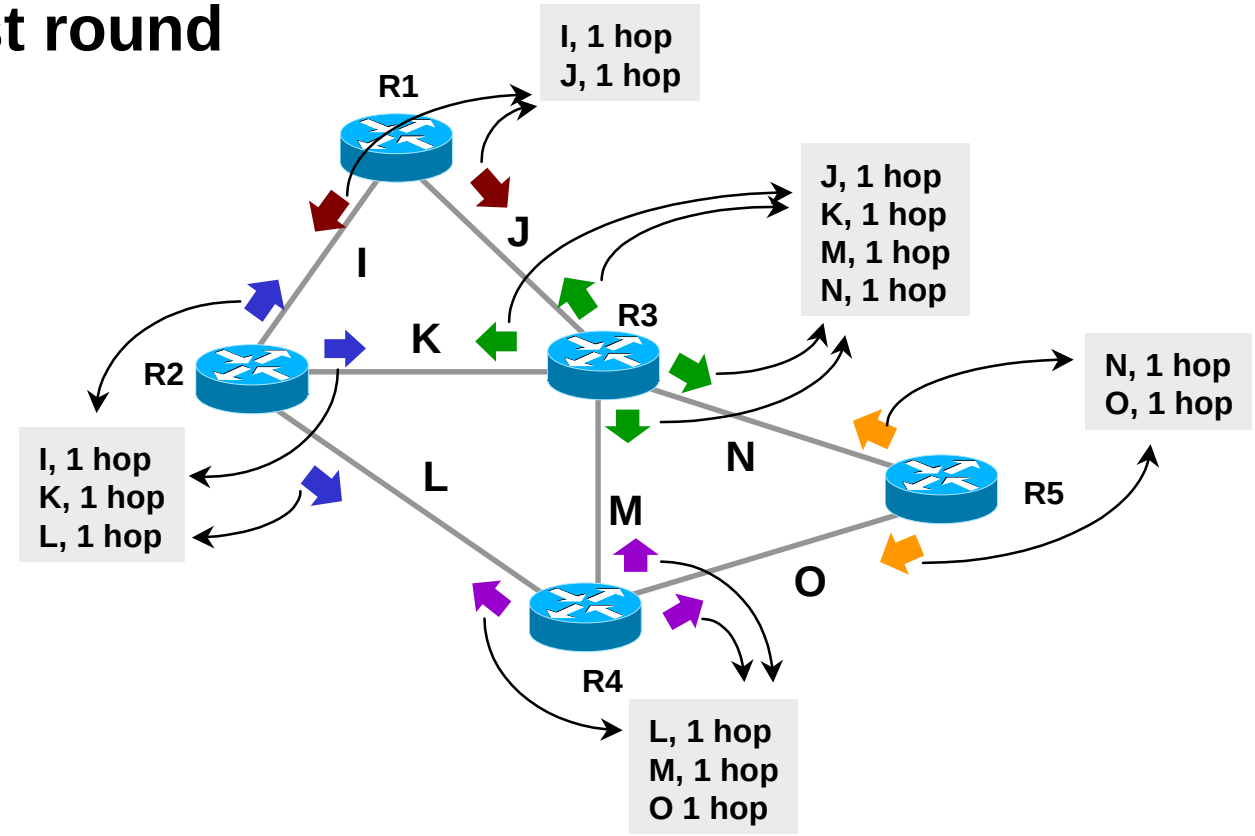
Distance vector routing update

- step by step from router to router
- slow convergence



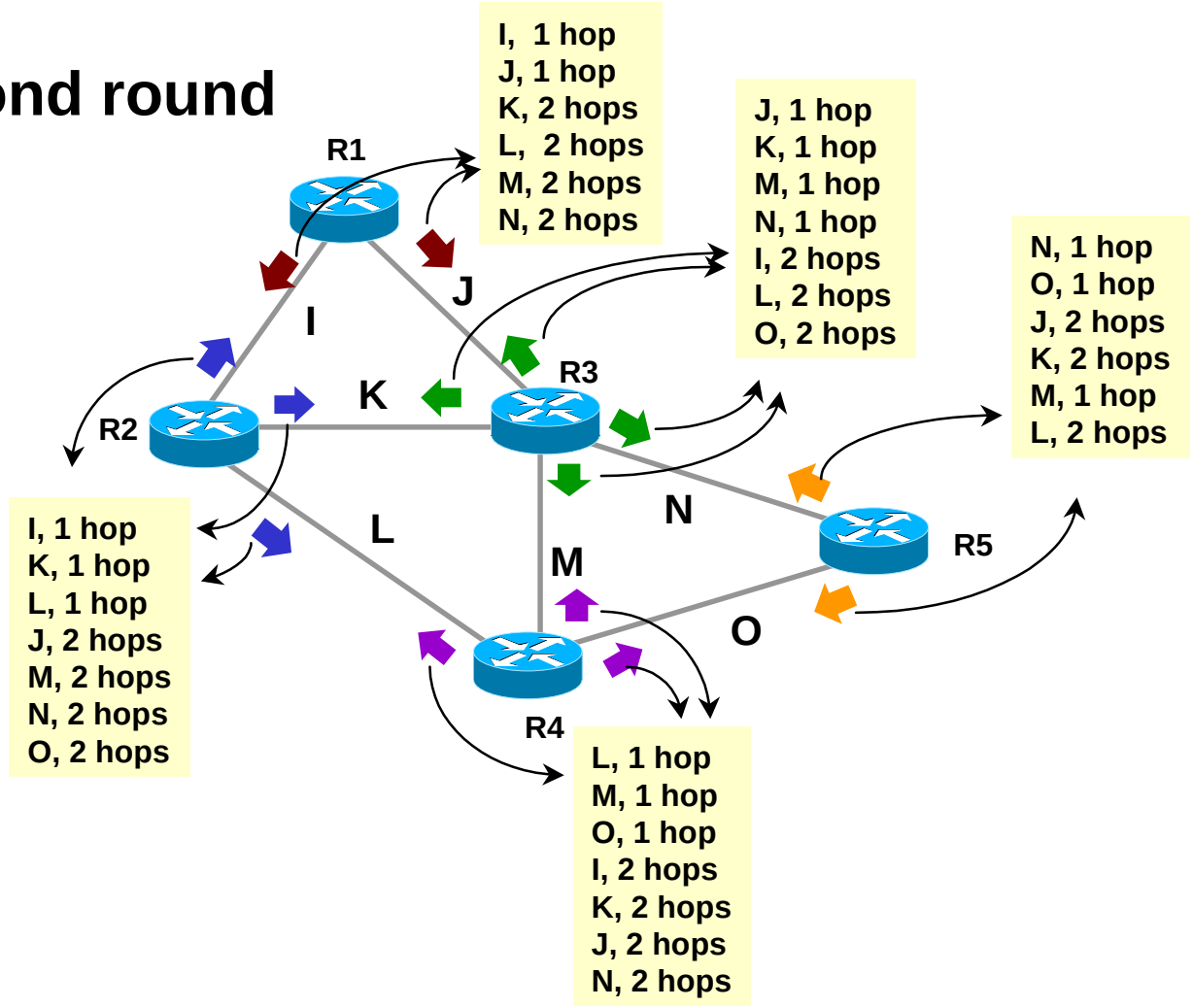
Distance vector: broadcast (I)

- the first round



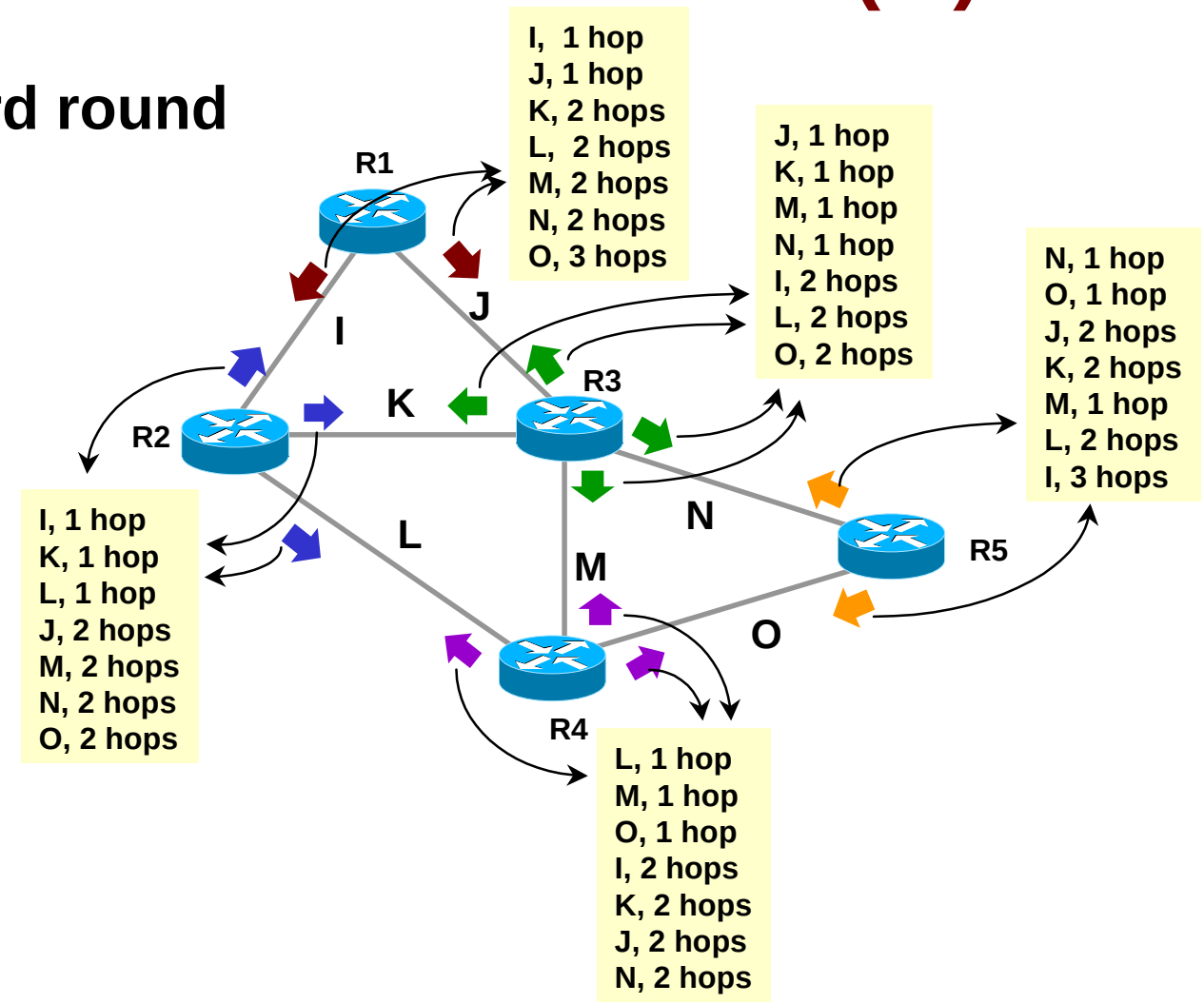
Distance vector: broadcast (II)

- the second round



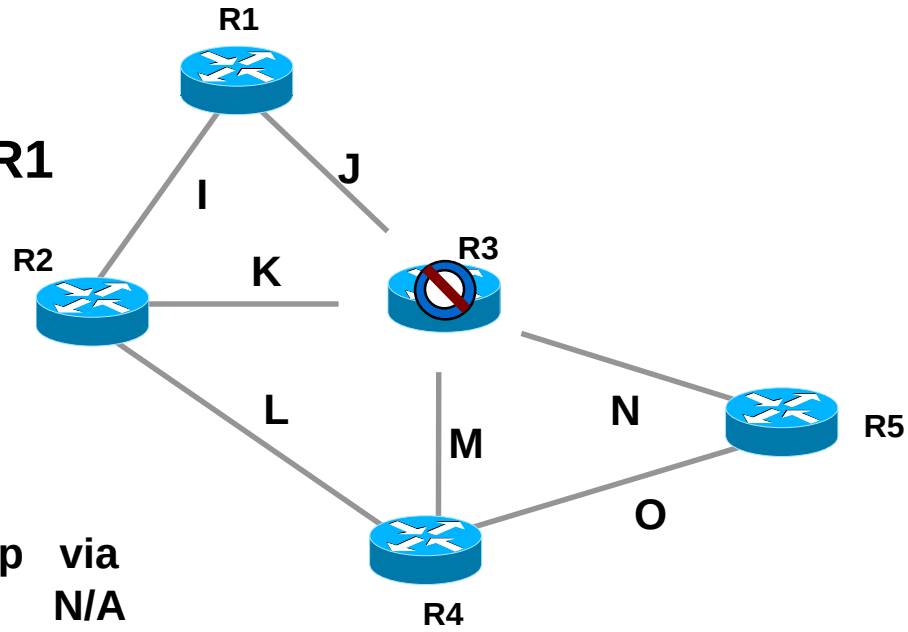
Distance vector: broadcast (III)

- the third round



Distance vector: crashed recovery

- **R3 crashed**
- **new complete route of R1**



R1 routing table

net	hop	via
I	1	N/A
J	1	N/A
K	2	R2
L	2	R2
M	2	R3
N	2	R3
O	3	R5

net	hop	via
I	1	N/A
J	1	N/A
K	2	R2
L	2	R2
M	3	R2
N	4	R2
O	3	R2

Count to infinity

R3 crashed

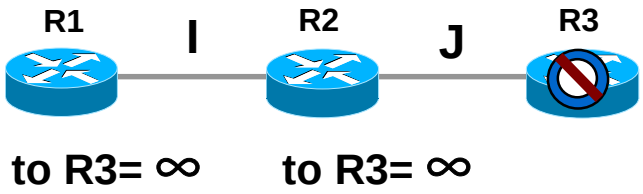
2	1	initial
2	3	1st round
4	3	2nd round
4	5	3rd round

hop count to R3

- R2 does not hear any thing from R3
- R1 says : don’t worry, I can reach R3 in 2 hops, R2 update hop count to 3
- R1 sees R2’s update, then update itself to 4 and so on , …

Solving count to infinity

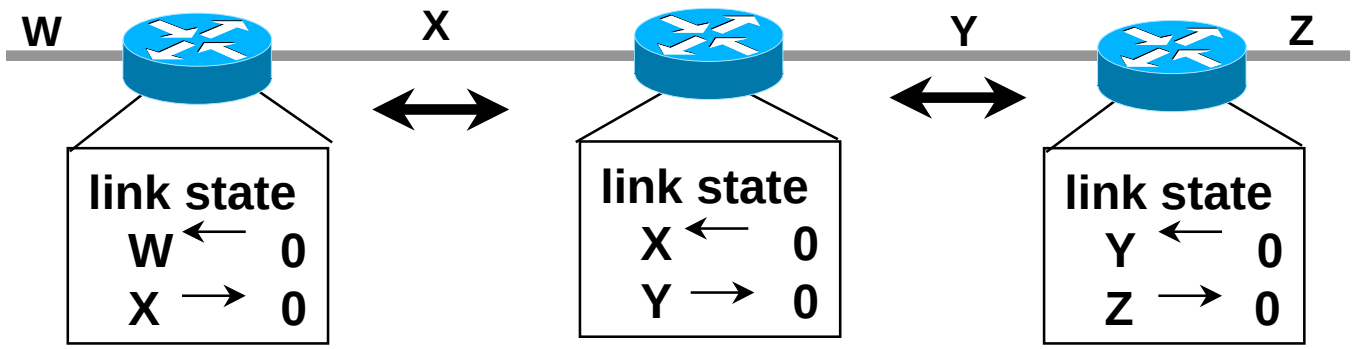
- solve by set distance “16” as infinity
- no destination can be more than 15 hops away from any other
- split-horizon : distance to X is not reported on the line that packet for X are sent
- Poison-reverse: send a route update that specifies that the distance is infinity



R3 crashed

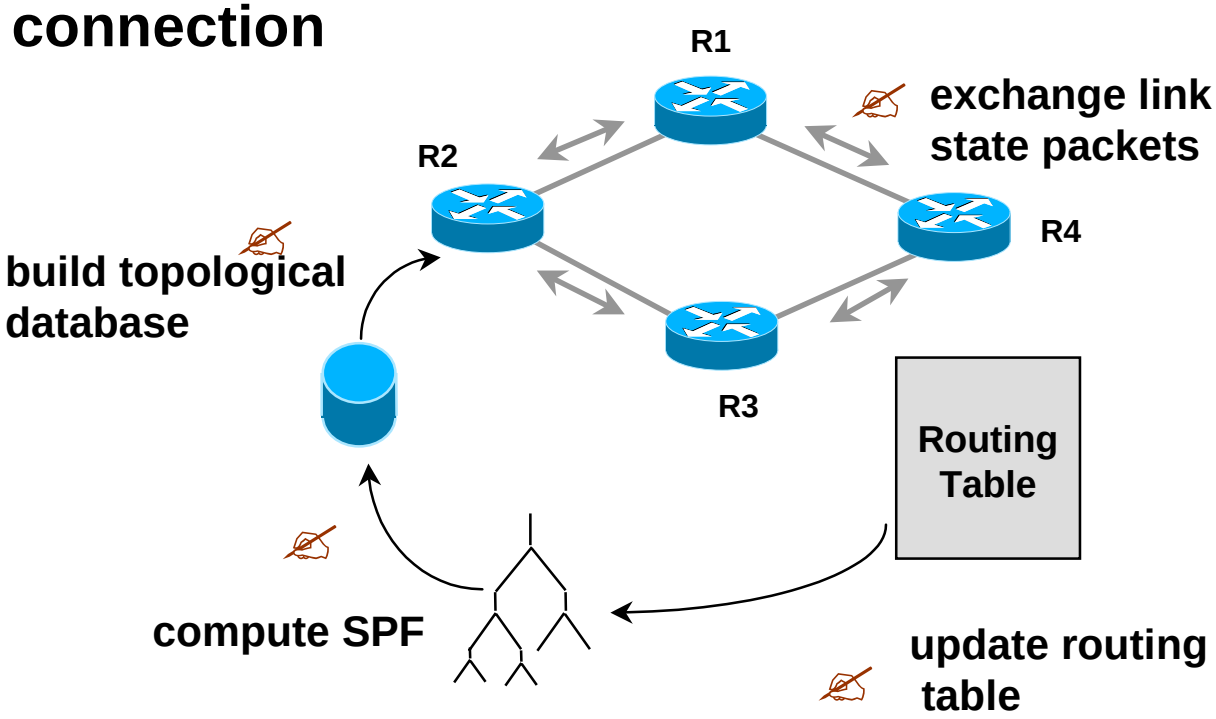
Link State Overview

- using cost as a metric
- exchange its connection and cost to its neighbors
- each router compute the set of optimum path to all destination (Shortest Path First)



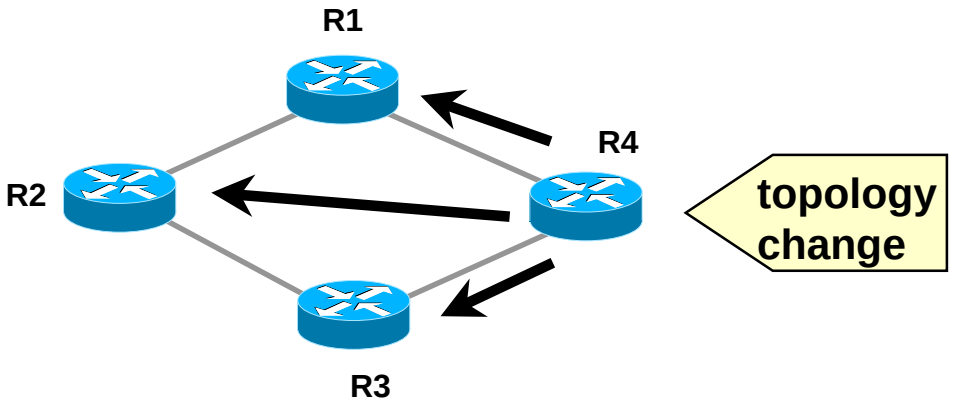
Link State concept

- each router initially begins with directly connected network
- determine full knowledge of distant routers and theirs connection



Link State routing update

- send information to other routers
- fast convergence



Comparison

Distance Vector	Link State
pass a copy of whole routing table	pass links state update
add metric from router to router	calculate the shortest path to other routers
frequent periodic update: slow convergence	event updated: fast convergence