C5 Routing Protocols

1. Internet Routing

- Router is a device used to interconnect networks and to forward packets by examining the destination address in the IP header of each packet
- Route path is determined from the routing table which is initialised following a routing protocol

1.1 Flooding

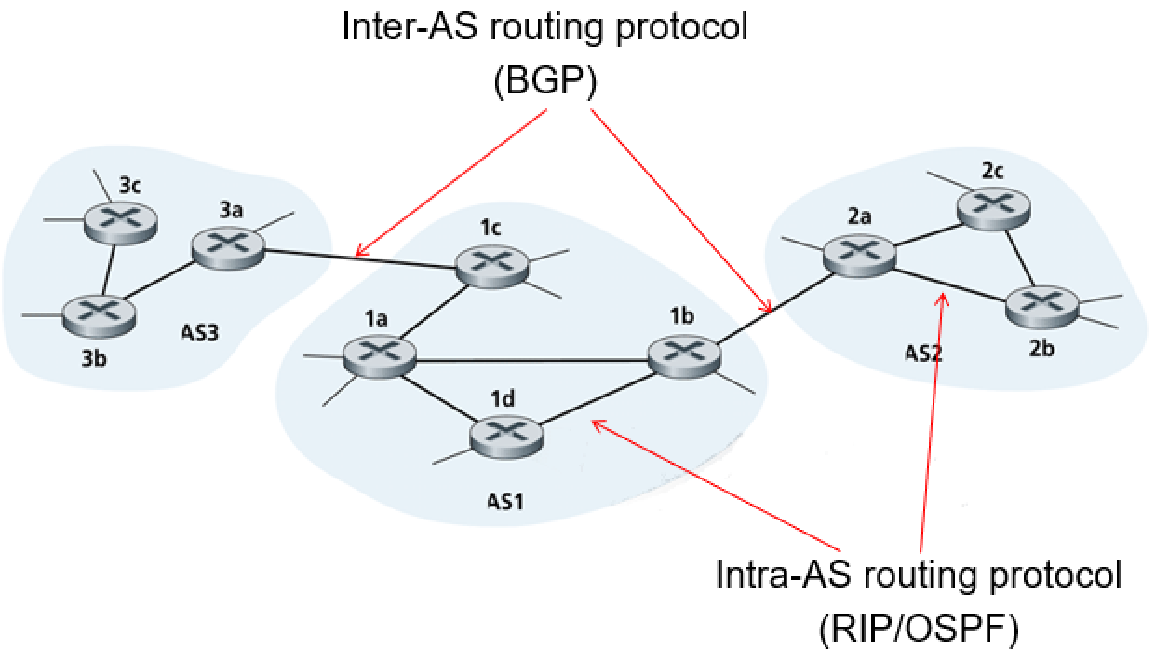
- When a node receives a packet, it will forward the packet to all other links except the incoming link, with a unique identifier attached to it.
- If packet returns to the same node, it gets discarded
- **Advantage** - Packet will always get through if one or more path (to destination) exists
- **Disadvantage** - Very wasteful of bandwidth, may cause serious congestion

1.2 Autonomous Systems

- Internet is divided into Autonomous Systems (AS) for routing purposes
  - AS refers to a group of routers under the authority of a single administration
  - Each AS is uniquely identified by a 2/4 byte AS number assigned by IANA

1.2 Intra AS + Inter AS routing

- Routing is done in a hierarchical manner (intra-AS and inter-AS)



- **Intra-AS routing**
  - Routing within an AS

    - Protocols are known as Interior Gateway Protocols (IGP)
    - Different AS can choose to run on their preferred protocols
- **Inter-AS routing**
    - Routing between AS
    - Protocols are known as Exterior Gateway Protocols (EGP)
    - **All AS** must run the same protocol

Differences

- **Policy**
    - Inter-AS: different admin wants control over how traffic are forwarded and who routes through its network
    - Intra-AS: Single admin deciding routing within AS; no policy decision needed
- **Performance**
    - Intra-AS: can focus on performance
    - Inter-AS: policy may dominate over performance
- **Scale**
    - Internet too large to be treated as a single routing domain

2. Intra AS routing

2.1 Distance Vector Routing

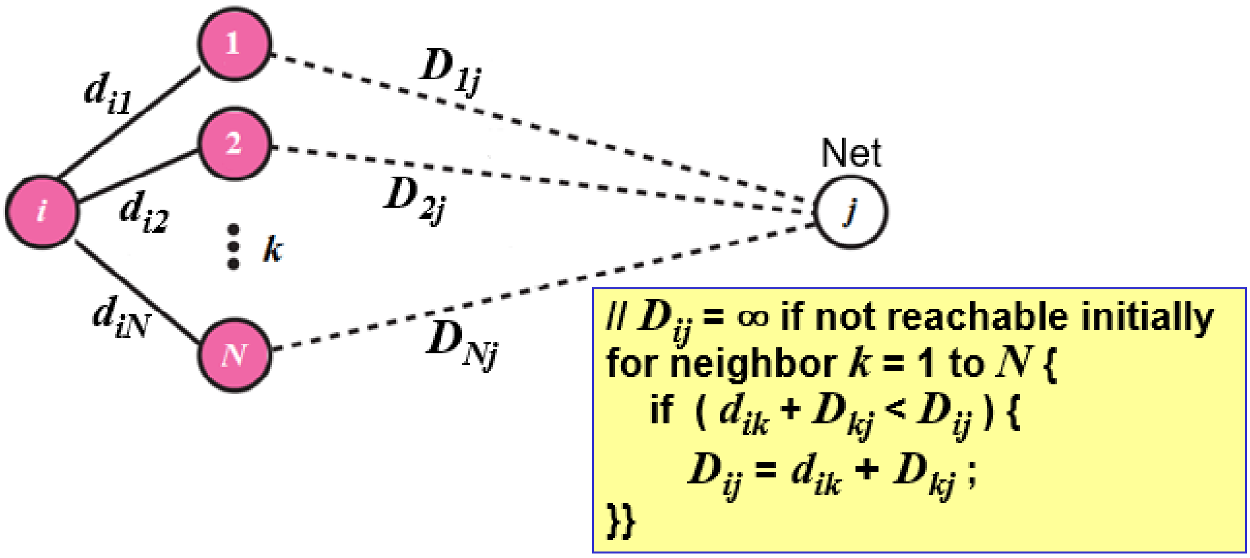
- Also known as 'Bellman-Ford' / 'old ARPANET' routing

1. Discover neighbours by multicasting request
2. Exchange distance vectors (routing information) with immediate neighbours
    - Response to request
    - Periodic updates
    - Triggered updates due to changes
3. Compute shortest-path using Bellman-Ford algorithm

General Idea

- Initially a router only has its own configured routing table
- Router will then send a multicast request to discover adjacent neighbours and exchange distance vectors

- Shortest path will be calculated using bellman-ford algorithm



d_{ik} = cost of going directly from node i to adjacent neighbor node k ;

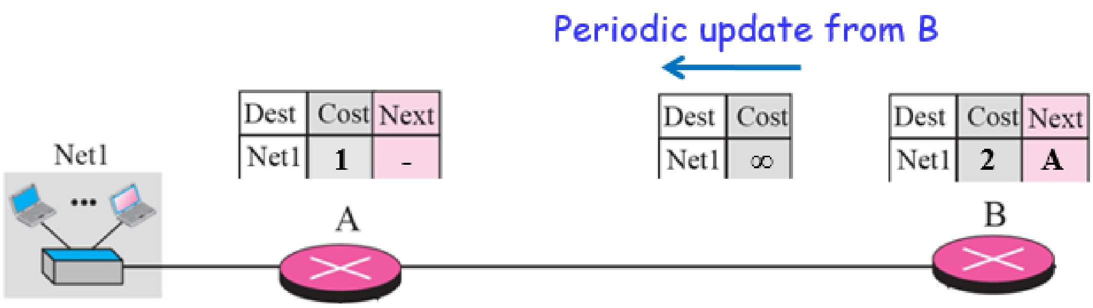
D_{ij} = least total cost of going from node i to destination j

- **Cost from A to B is not necessarily cost from B to A**

Count to infinity Problem

https://www.youtube.com/watch?v=f2ic7kVnhrs

- Hard to explain with words, watch link
- Idea is that two nodes will constantly update the other, which updates the count to the offline node, incrementing cost in their routing table to infinity.
- **Solution**
  - Split horizon with poisoned reverse
    - If B gets to Net1 via A, then update to A should indicate Net1 is unreachable



Routing Information Protocol

- Uses Distance Vector algorithm, and cost is based on number of hops, max being 15, 16 indicated as ∞, updated every 30s via Response Message
- No message heard after 180s $\rightarrow$ node / link declared dead / offline

2.2 Link State Routing

- Makes use of Dijkstra's Algorithm

1. Discover Neighbours by multicasting Hello
2. Construct Link State Advertisement Packet (LSA/LSP)
3. Flood LSA / LSP to all routers during initial start-up / when there is a change in topology
4. Construct Link state database
5. Compute Shortest-Path routes using Dijkstra's Algorithm

- Each router builds its own LS database to have a complete topology of the whole network

3. Inter AS routing

3.1 Path Vector Routing (Border Gateway Protocol)

1. Configure border router to know its neighbours
2. Exchange path vectors (routing information) with neighbours
3. Select path based on policy

- BGP routers will establish a TCP connection with their neighbours to exchange routing information
- https://www.youtube.com/watch?v=_aLmzq-23pE

Protocol Design Principles

- Scalable - backbone AS must be able to find the destination
- Loop free
- Autonomy of routing policy
- Broadly, Autonomous Systems can be classified into stub AS, multi-homed AS or transit AS
- **Stub AS**
    - Connected to only one other AS; typically customer connected to its provider. In fact, it's not necessary for stub AS to run BGP since it has only one path to its ISP
- **Multi-home AS**
    - Connected to more than one AS, but does not carry transit traffic; typically for customer requiring reliability
- **Transit AS**
    - Connected to other AS to carry transit traffic for its customers; mainly for providers (ISPs)

Configuring BGP routers

- External peers between different AS are normally adjacent to each other and share a subnet
- Internal peers may be in any subnet within the same AS
- Peers exchange routing information containing complete AS path to avoid loop problem

- Based on policy, BGP routers can decide to accept / decline offered paths; and to drop / advertise paths to their neighbours
    - import policy - may or may not select path offered
    - export policy - can filter routes you don't want to tell neighbours (e.g. don't want to route traffic to Z $\rightarrow$ don't advertise any route to Z)