Network Routing

Aashish Paliwal[1]

[1]C-STAR
International Institute of Information Technology, Hyderabad

01st September 2022

Outline

Outline

1. Definition
2. Desirable properties of routing
3. Design parameters
4. Routing strategies

Definition

- On the Internet, routing is the way IP packets of data travel from their origin to their destination.
- Any sufficiently large network (e.g. Internet) can have hundreds or even thousands of paths to reach from node A to node B. As network scale, routing becomes more important and complex in packet switched network design.
- Routing consists a huge chunk of network layers responsibility
- The process of constructing routing tables for switches is called routing

Routing vs Forwarding

- Routing
  - Computes path that packets will follow
  - Routers share data among themselves to create forwarding table
- Forwarding
  - a table lookup
  - Directing a packet to particular link
  - Uses forwarding tables extensively

Routing Game

- Require 20 volunteers
- Adapted rules from Scott Shenker, UC Berkley and then modified further to add more randomness
- You have 5-7 minutes to complete the task
- Rules:
  - Each slip says Your ID = ?. Send data packet to node_id. S = Your_id
  - Your job: Find path from source to destination
  - You are directly connected to three nodes standing closest to you
  - You may not: Leave your spot, pass your slip
  - You may: ask for advice, speak to other participants (anything), not curse
  - You must: Try (if you're a volunteer), help or provide advice (everyone), not cheat

Outline

Desirable properties

- Correctness
- Simplicity
- Robustness
- Stability
- Fairness and optimality
- Efficiency

Outline

Design Parameters

- **Performance Criteria**: number of hops, cost, delay, latency
- **Decision time**: per packet (Datagram), per session (Virtual Circuit)
- **Decision place**: each node (distributed), central node (centralised), originated node (source)
- **Network information source**: none, local, adjacent node, nodes along route, all nodes
- **Network information update timing**: continuous, periodic, major load change, topology change

Outline

Routing strategies

- a substantial number of routing strategies have developed over decades
- Some known routing strategies includes:
  - Static routing
  - Flooding
  - Random routing
  - Flow based routing
  - Dynamic routing

Static Routing

- a central routing matrix is created based on least cost path and is stored at network control center
- the matrix holds the next node on the route for each source destination pair
- Advantages
  - Simple
  - Works well in reliable networks with stable load
- Disadvantages
  - Unreliable - contains single point of failure
  - Does not respond to failures
  - Lacks flexibility

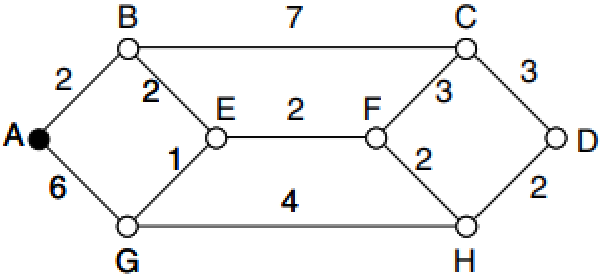
Least Cost Path

- Cost associated with each link e.g. - minimum hop, latency, queue length etc
- For each pair of attached nodes, the least cost path is looked for
- Several algorithms exists to find optimal path
  - Dijkstra's algorithm
  - Bellman-Ford algorithm

Dijkstra's algorithm

- Dijkstra's algorithm finds the shortest paths from a given source node to all other nodes in order of increasing path length
- The algorithm converges under static conditions of topology and cost
- Basic steps
  - Initialization
    - Cost of each node to ∞
    - Cost of source node to 0
  - while unknown node left in the graph
    - Select an unknown node b with the lowest cost
    - Mark b as known
  - For each node a adjacent to b
    - a's cost $=$ min(a's old cost, b's cost $+$ cost of (b, a))

Dijkstra's algorithm example



Flooding

- Does not require any network information
- Every incoming packet is forwarded to every outgoing link
- Advantages
  - Simple
  - Packet reaches every connected node
  - Highly robust (can be used for emergency messages)
  - Atleast one packet pass through shortest path
- Disadvantages
  - Generates huge amount of duplicate data
  - Require damping mechanism to be used in practise eg - hop count

Random Routing

- for every incoming packing, node selects one output link on random
- Advantages
  - Has simplicity and robustness of flooding while generation much less traffic
  - Does not need network information
- Disadvantages
  - Actual route will usually be much more expensive than least cost path

Flow Based Routing

- utilised information about topology and load information for routing
- If capacity and average flow is known for a give line, we can calculate mean packet delay
- From mean delay of all lines, we can calculate flow weighted average to get mean delay of subnet
- Routing algorithm then optimised path based on minimum average delay

Dynamic/Adaptive Routing

- Network state information must be exchanged among nodes
- Typical metrics - Number of hops, time delay, total number of packets queued in the path
- How much information is required?
  - More information exchange $\implies$ better routing $\implies$ more overhead
  - More frequent $\implies$ better routing $\implies$ more overhead
- Popular approaches
  - Distance Vector Routing
  - Link State Routing

Distance Vector Routing

- Can be divided into three steps
  - Identify neighbours
  - Send advertisements to neighbours consisting of tuples as
$$[(dest1, cost1), (dest2, cost2), (dest3, cost3).....]$$
  - Integration step uses Bellman-Ford algorithm to find least cost path
$$updated\_cost = min(current\_cost, link\_cost(source, advertiser) + advertised\_cost(advertiser, destination))$$
- Advantages
  - Simple
  - Works well for small networks
  - Only advertises best cost to each destination
- Disadvantages
  - Works only for small networks
  - Cannot determine topology of network

Link State Routing

- Can be divided into three steps
  - Identify neighbours
  - Send link-state advertisements (LSA) periodically

    $$[seq\#, (nbhr1, linkcost1), (nbhr2, linkcost2), ...]$$

  - Integration
    - if seq# in incoming LSA ¿ seq# in saved LSA for source node: Update LSA for node with new seq#, neighbour list rebroadcast LSA to neighbour (Flooding)
    - Remove saved LSA if seq# is very old
    - Result: Each node has complete current map of network
- Uses Dijkstra's algorithm at each node to compute shortest path
- Advantages
  - Scalable
  - Fast network convergence
- Disadvantages
  - High network overhead
  - Require more memory and processing power