Institut für Technik der Informationsverarbeitung

Prof. Dr.-Ing. Dr. h.c. Jürgen Becker *(becker@kit.edu)*

Dr.-Ing. Jens Becker *(jens.becker@kit.edu)*

Institut für Technik der Informationsverarbeitung (ITIV)

Communication Systems and Protocols

Session 11: Networks

Clicker Session: Recapitulation

- https://arsnova.eu/mobile/#id/33969518

Recapitulation

- Automotive busses
  - FlexRay

Networks

Definition: Network

An arrangement of intersecting horizontal and vertical lines:
a spider constructs a complex network of several different kinds of threads

A number of interconnected computers, machines, or operations:
a computer network

1.Computers: A group of interconnected (via cable and/or wireless) computers and peripherals that is capable of sharing software and hardware resources between many users. The Internet is a global network of networks. See also local area network and wide area network.

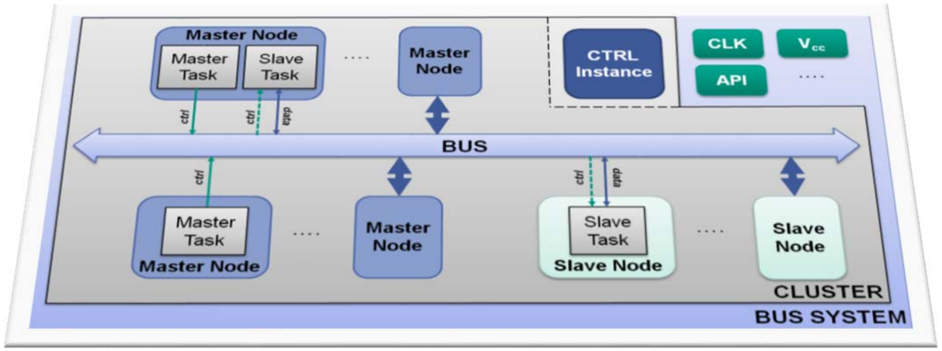
2.Communications: A system that enables users of telephones or data communications lines to exchange information over long distances by connecting with each other through a system of routers, servers, switches, and the like.

Sources: Oxford Dictionaries
http://www.businessdictionary.com/

Network vs. Bus

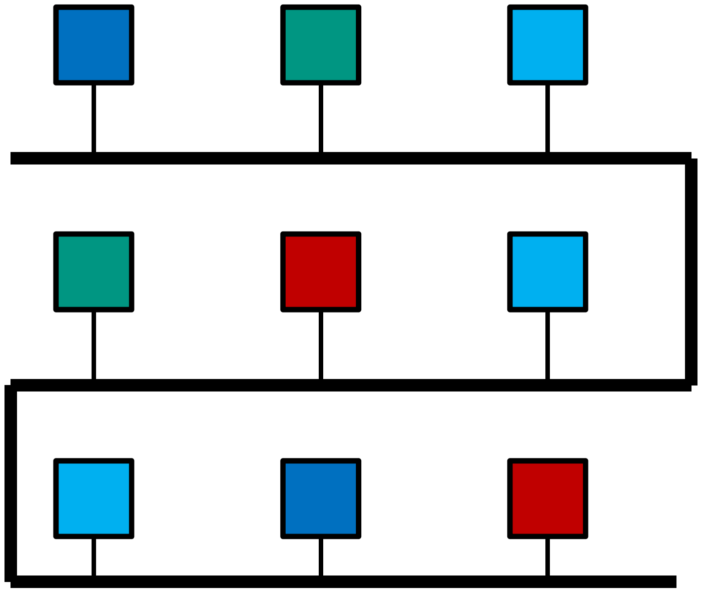
- Networks:
  - different and multiple communication channels are possible
  - Error resistance can be higher due to alternative routing
  - variable Bandwidth due to multiple, parallel channels

- Bus:
  - dedicated and fixed physical communication channel
  - bandwidth is limited by bus line
  - Communication bandwidth is constant for growing number of nodes → bad scalability

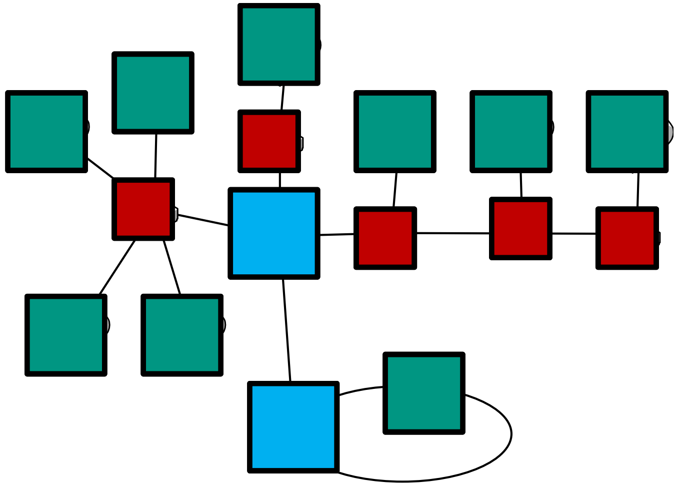


Benefits of Network

System Setup using a bus

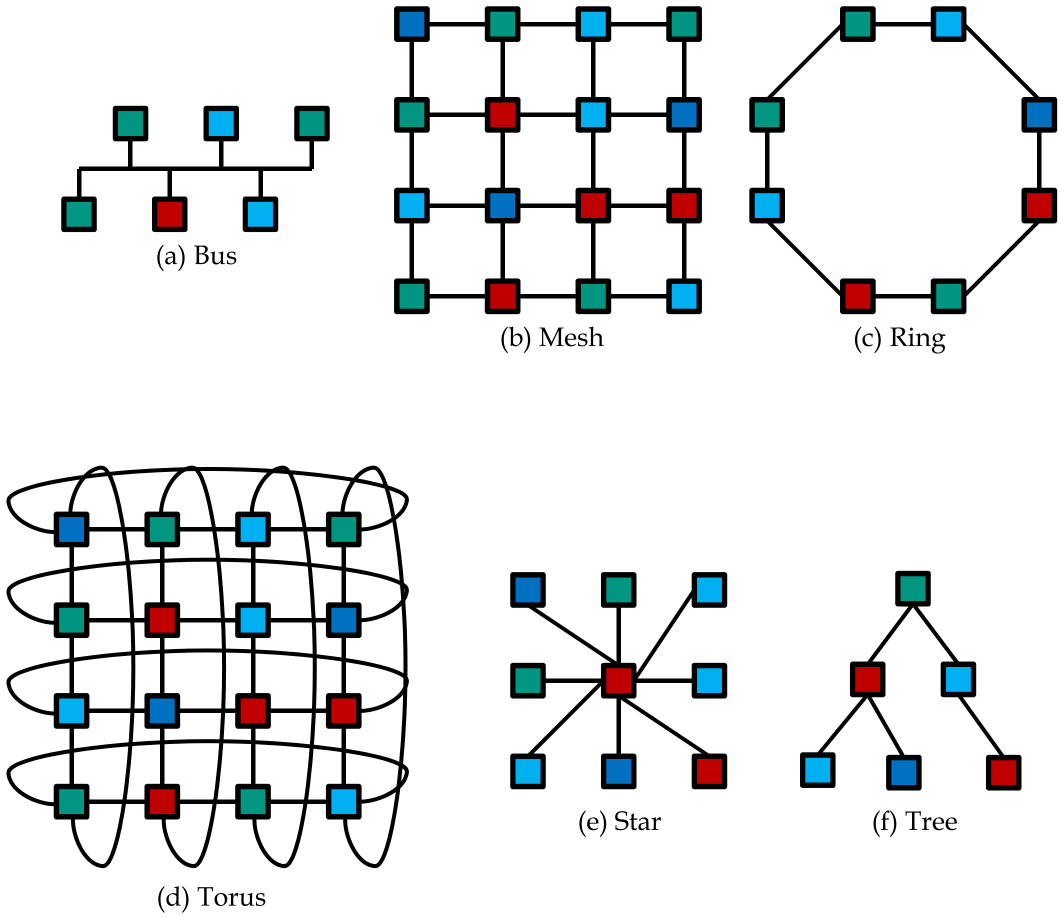


System Setup using a Network



- Less and shorter communication lines
- Multiple connections at the same time possible
- Very flexible
- Efficient use of bandwidth

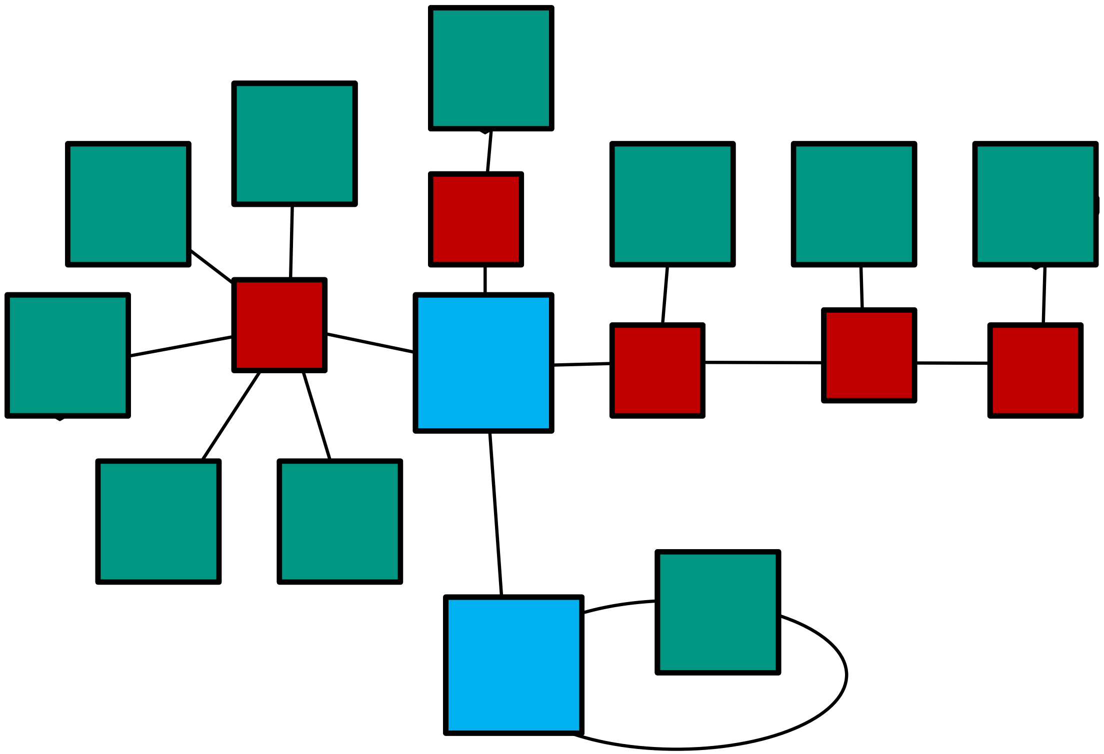
Networks – physical Structure



- Different components interconnected
  - Computing Nodes
  - Routers
  - Switches
  - …
- Differentiation between geometric / logic
- Geometric topologies can be ring, line, star, and any combinations

Example of a physical Structure

- Local Area Network

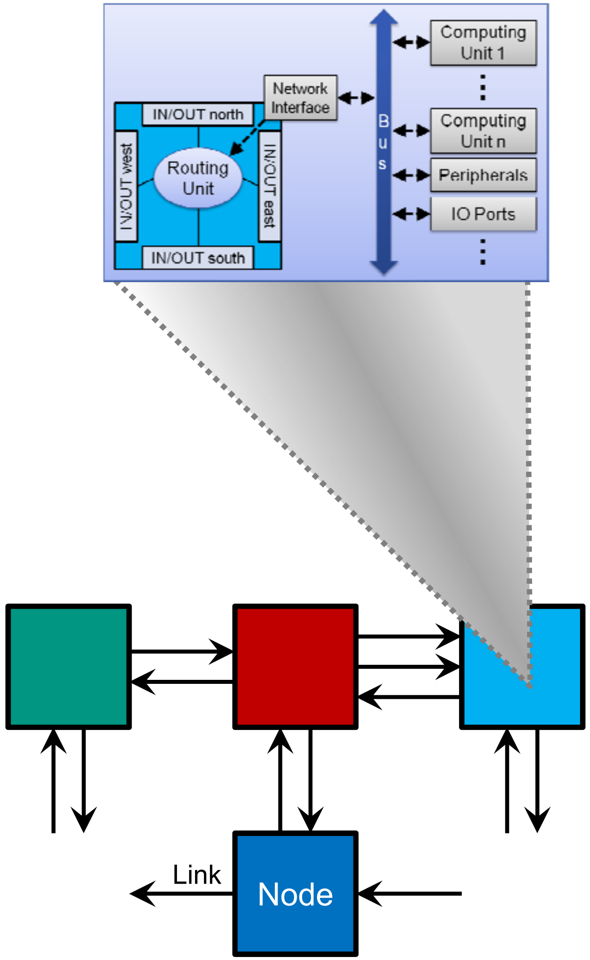


Definitions

Node
- Basic communication entity that gets connected in a network
- A network node (normaly) contains Ports,Routing Unit, Network Interface and a Tile

Tile
- A Tile is the units that produces and consumes the data that is to be transported over a network
- Computing Units, Peripherals, Memory, Input/Output ports, etc.

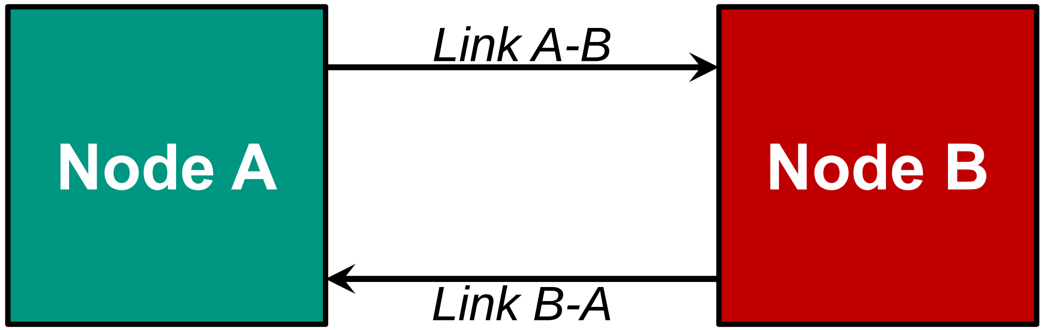
Link
- Communication path between neighboring nodes
- unidirectional
- multiple links between two tile can exist (in either direction)



Network Link

- Physical communication channel between neighboring nodes
- Unidirectional
- Link has fixed data width
- Multiple links between two nodes can exist (in either direction)
  - concurrent transmissions

  or
  - aggregation to achieve higher throughput/bandwidth



Definitions II

Hop

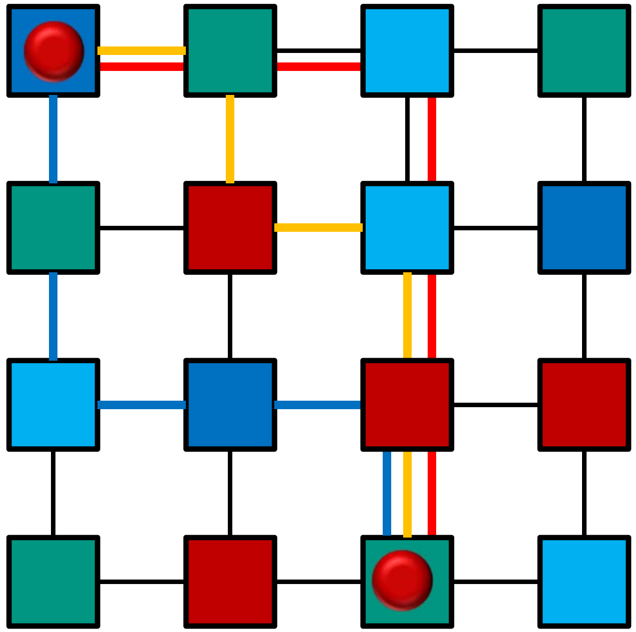
- The transmission of data from one node to the neighboring node

Diameter

- The Diameter of a network is the maximum number of hops between any pair of nodes in a network

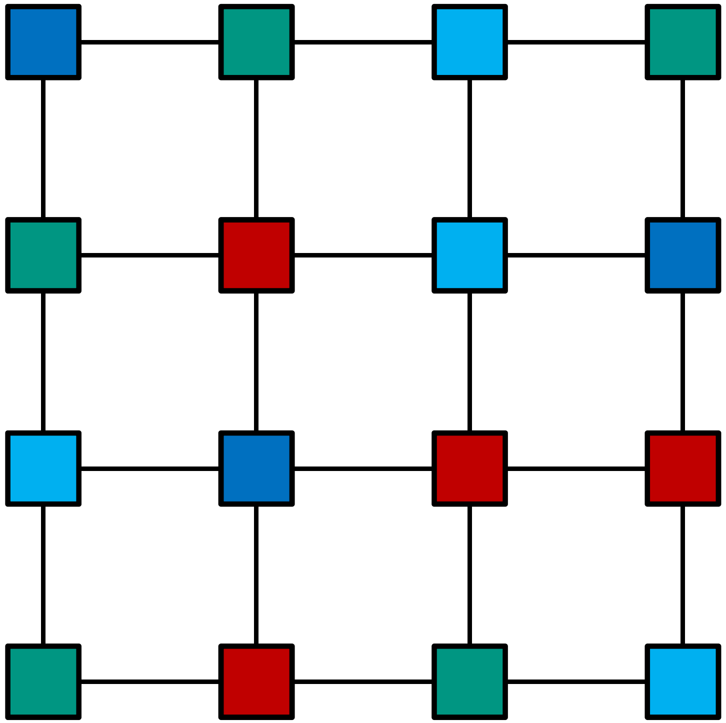
Manhattan Distance

- Layout of the network must be a mesh based grid
- physical distance following the edges
- all paths that always move towards the target have same distance



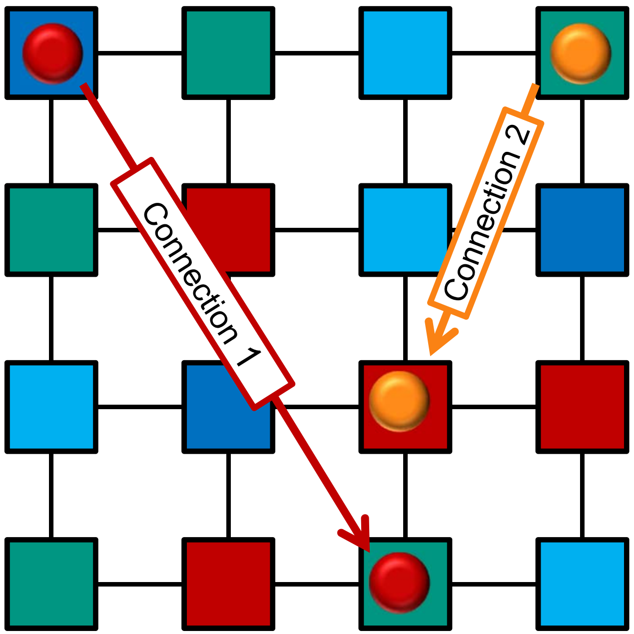
Communication Setup in Networks

- Need for logic connections between nodes (channels)
- Different paths can be conceived to connect nodes.
- Routing
  - find a suitable communication path



Connection Types

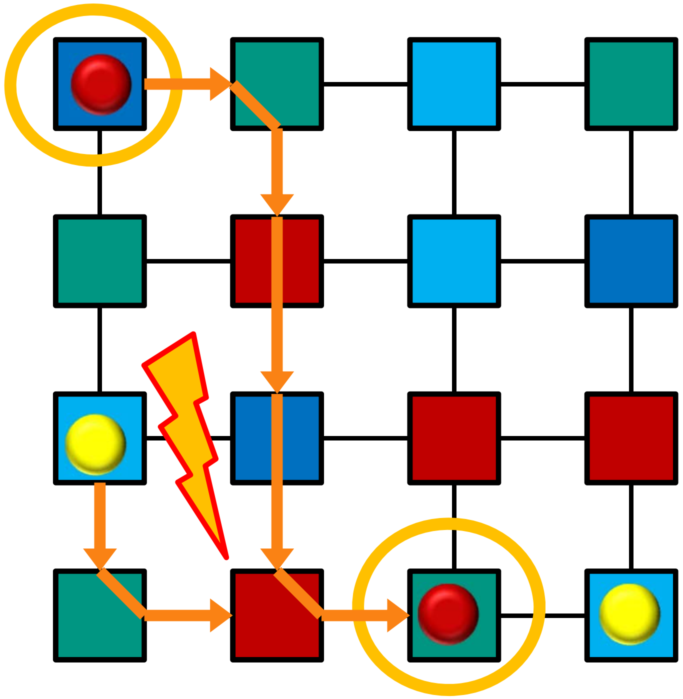
- Establish logical communication channel between nodes
- Multiple concurrent connections can exist



- **Two general types of connection**
  - Circuit Switching
  - Packet Switching

Circuit Switching

- Nodes communicate over a fixed path
- Path is established for the full communication cycle
- Path is used exclusively, can not be used for a second communication

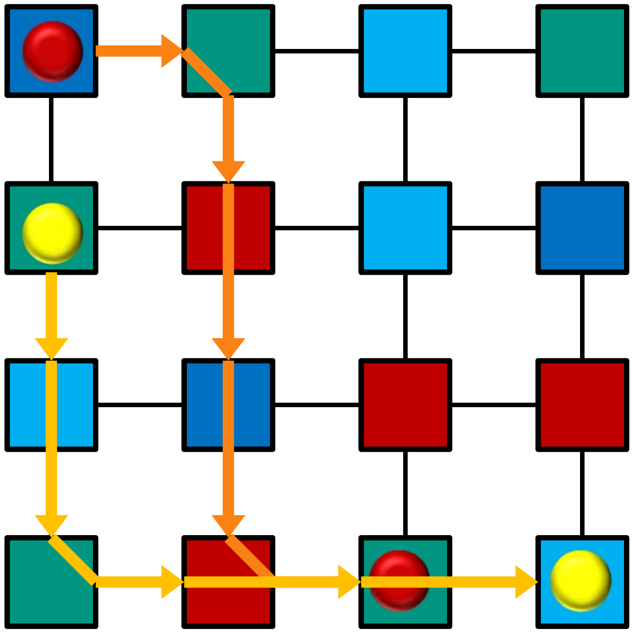


- **Pros & Cons**
  - No interference (on logic level)
  - Other communication paths may be blocked off
  - Guaranteed throughput possible
- **Example:** Analog telephone network

Packet Switching

Karlsruher Institut für Technologie

- Data is send in consecutive packets
- Path is established for the duration needed to send one packet
- Path can be used in a time multiplexed fashion by multiple communication processes



- **Pros & Cons**
  - Different routes of packets can lead to different arrival sequence
  - Ordering of packets may be necessary (packet ID)
  - Guarantees for network behavior difficult to give
  - Less blocking of communication paths

Comparison

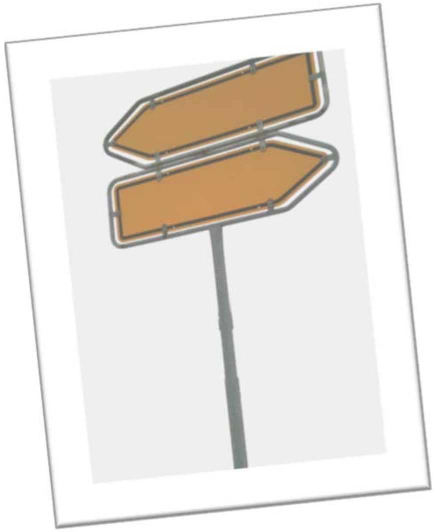
Circuit Switching		Packet Switching	
Pros	**Cons**	**Pros**	**Cons**
• good for streaming data • guaranteed throughput • fast • simple routers • high throughput • real-time capable	• can block other communication • inefficient use of communication resources (for sporadic communication) • Setup required prior to communication start	• good for sporadic data transfers • flexible • multiplexing of paths possible • efficient usage of communication resources (links) • Failure tollerant (multiple paths to target node)	• complex routers • additional control overhead • additional local storage needed • varying latencies possible • behavior hard to predict

Routing

The idea behind Routing

- Routing Problem
  - Connect data source with data sink
  - Decision on what paths to use
  - has to be integrated into the communication system (i.e. one or more network nodes, router,…)

- Possible Goals for Routing Algorithm:
  - Find short routes
  - Reduce routing overhead
  - Adaptability to…
    - changes in topology (node or link failures)
    - varying traffic loads
  - Avoid routing loops
  - …



Definitions

Loop

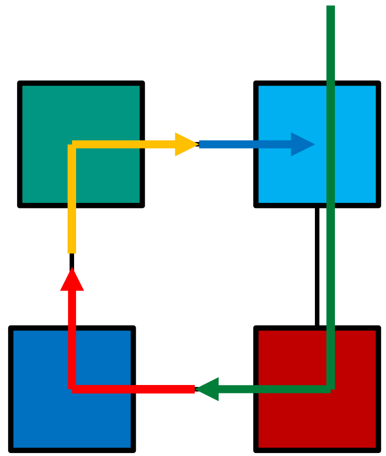
- A *Loop* is a path in a network that leads back to its starting point

Deadlock

- A situation where a link is blocked by one transmission that is waiting for the other transmission to finish is called a deadlock.
- No communication is possible in case of a deadlock

Livelock

- Data is forwarded through a network without reaching its destination
- Counterpart to deadlock



Routing Mechanism

- Each node needs to know what to do with a data packet
  - Pass it on or not?
  - Which outgoing link is to be used?

- Knowledge is gained from the routing algorithm
  - Source routing ← → distributed routing
  - Static routing ←→ Adaptive routing

Pathfinding / Routing

Static Routing

- predefined during design time
- generates fixed routing information for nodes

+ simple nodes
+ fast pathfinding due to precomputation
- not flexible
- susceptible to defective nodes
- no runtime adaptation

Adaptive Routing

- routing information is computed during runtime of the system
- the best path to a desired destination can change during operation
- knowledge of network status is necessary

+ highly flexible
+ scalable
- computationally expensive
- route finding slow due to internal computation

Pathfinding / Routing

Source Routing

- Sending node is defining the complete route for packet
- Information is part of the header of the packet

- + Optimal routings possible
- + Low complexity for intermediate routers
- – Protocol overhead, complete routing information is transported to the network
- – Overhead depends on path taken through the network

Distributed (cooperative) Routing

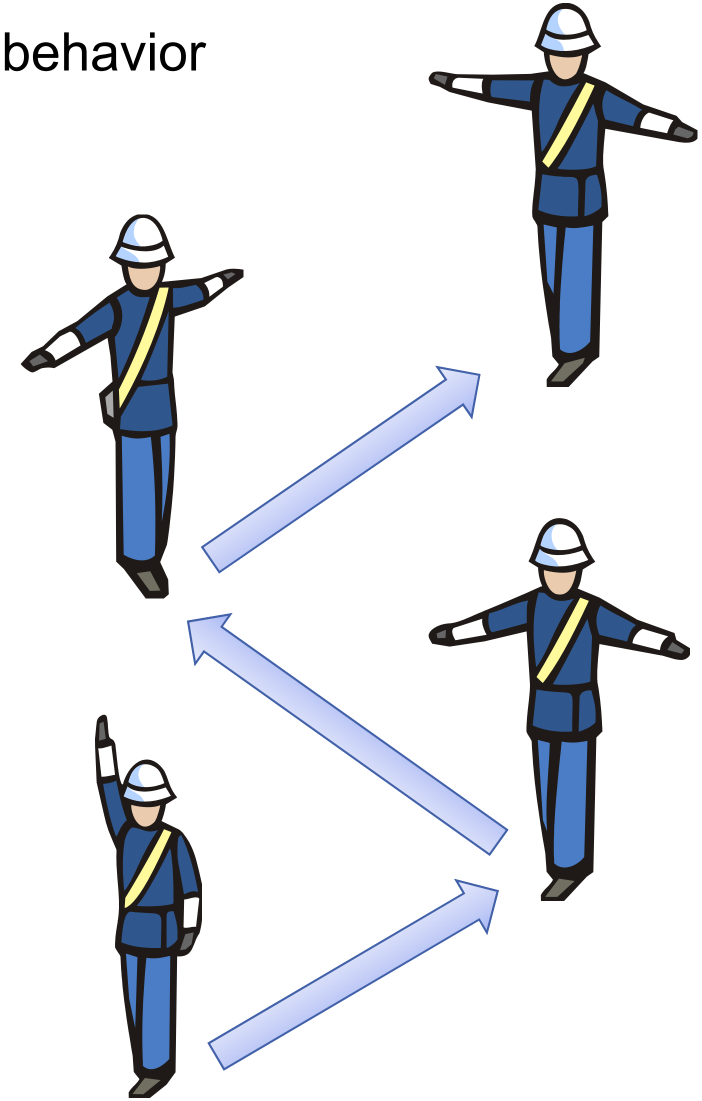
- routing computations is done in nodes
- cooperation of nodes by means of message exchanges possible

- + Easy adaptation to changing routing requirements during runtime possible
- + Constant protocol overhead → better scalability
- – Intermediate routers get more complex

Routing Algorithms

Routing Algorithms

- Routing Unit implements a (pre-)defined behavior
  - connects correct IN/OUT
  - implements the communication paths
- Routing has to be defined somehow
  - Routing algorithms
- Routing Optimization Goals
  - minimal latencies
  - short routing path
  - balanced network load
- Examples of Routing Algorithms
  - X-Y Routing
  - Flooding
  - Dijkstra-Algorithm



Routing Requirements

Usually multiple messages m_i are exchanged by multiple node pairs A_j and B_j

Routing goals

- Even Load Distribution over the network
- Minimization of *communication path or time (latency*
- Prevention of *Deadlocks*
- Minimization of *Contention* (delay of messages due to competing requirements)
- Minimization of *Congestions* (deletion of messages, for example caused by buffer overflows)

Classification of Routing Algorithms

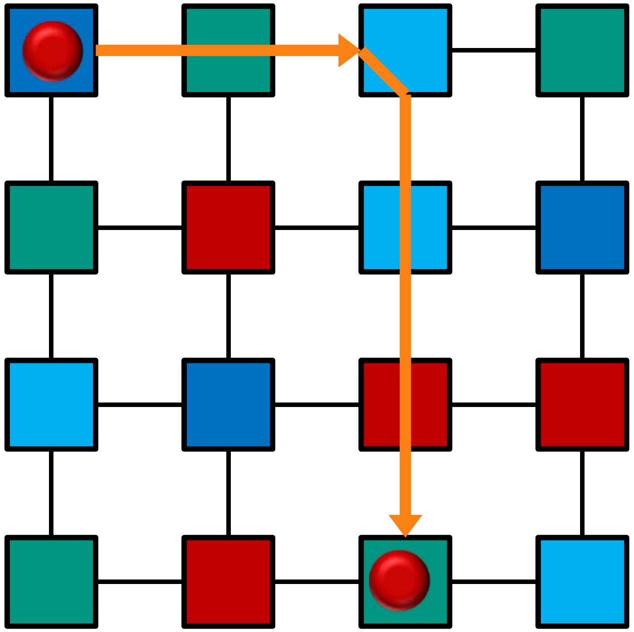
1. Depending on choice of path
   - Deterministic Routing
     - Unique communication path for a pair of nodes
   - Adaptive Routing
     - Multiple paths between two nodes are possible
2. Depending on length of path
   - Minimal Routing
     - always the shortest path is chosen
   - Non-minimal Routing
     - not always shortes path is chosen, for example because of congestion issues or load balancing, or ….

- Classifications are not correlated :

	Deterministic	Adaptiv
Minimal	Algorithm A	Algorithm B
Non-Minimal	Algorithm C	Algorithm D

X-Y Routing

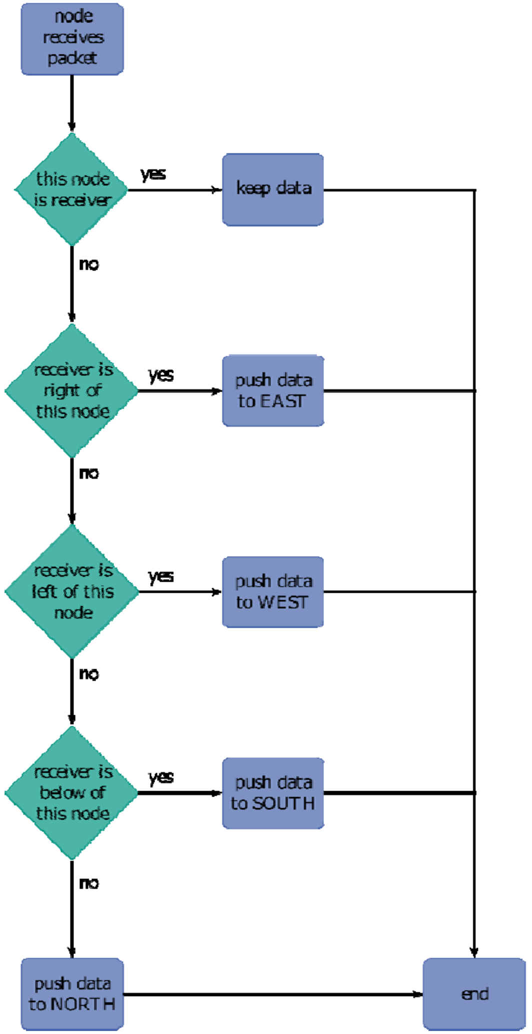
- Only applicable for mesh based networks
- Routing always towards destination
- First in X direction, then in Y direction

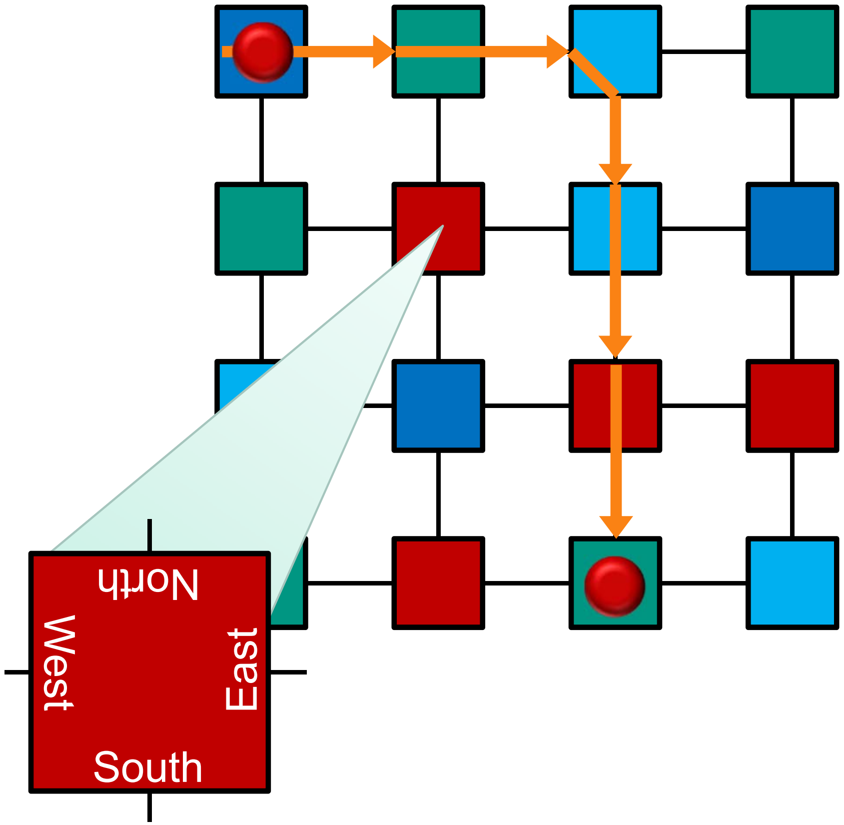


- Nodes have to know their position within the network
- Address has to be map able to the location within the network (coordinates)

+ Simple and small algorithm
+ Deadlock free
− Only works for meshes (rectangular)

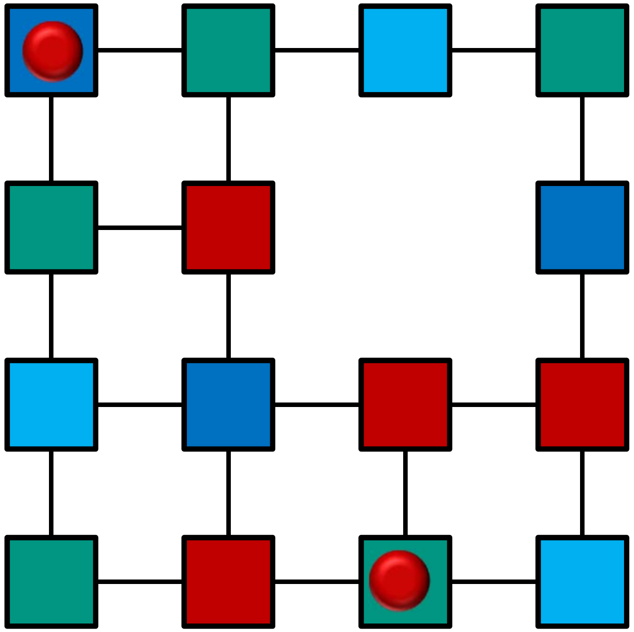
XY-Routing Implementation





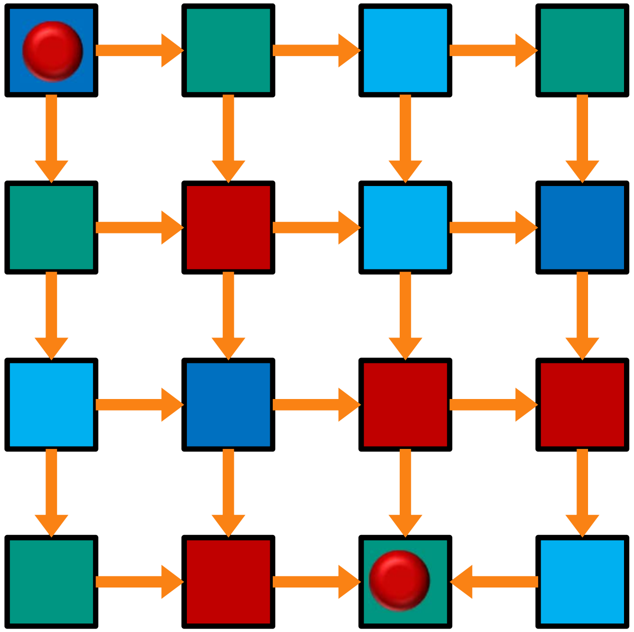
X-Y Routing: Limitations

- How can the destination be reached in case of a missing or faulty node?



Flooding

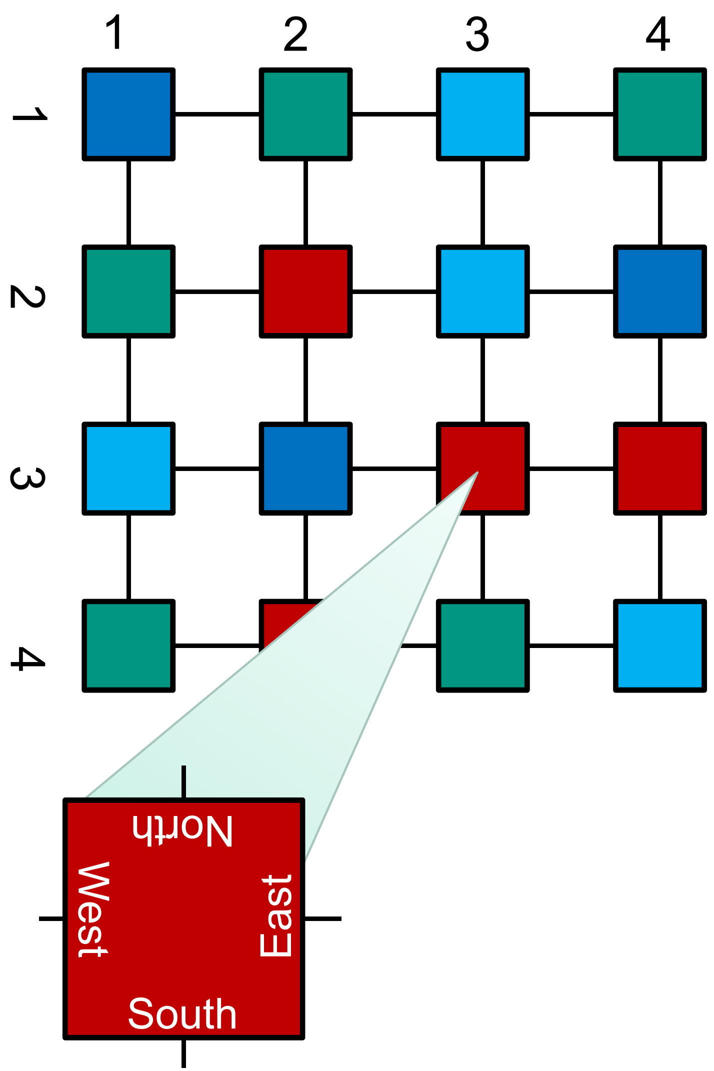
- Packets are forwarded to all ports except the receiving port
- Packet will eventually arrive at destination



- Time-to-live (TTL)
  - small number within a packet that is decreased by every router
  - if TTL = 0 packet is not forwarded
- Each packet is forwarded only once by a router
  - add Router ID to the packet when forwarding

\+ No need for routing table
\+ very robust
− May swamp network

Odd-Even Turn Routing

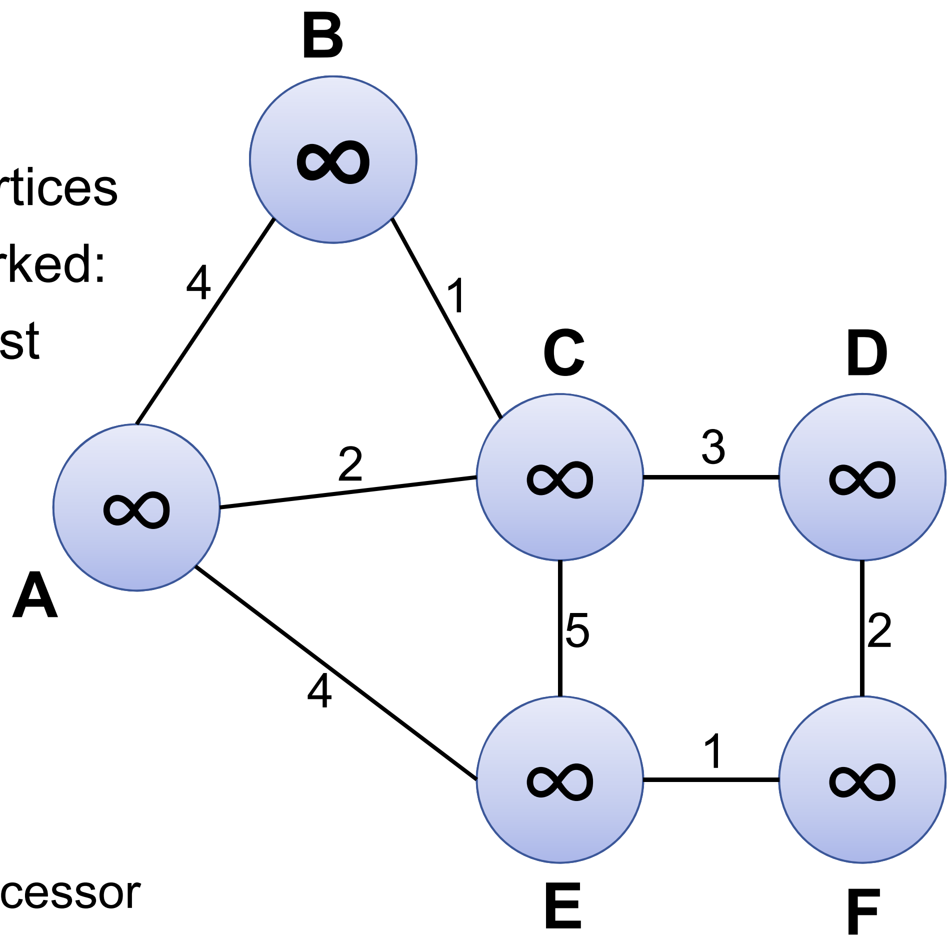


- Specialized XY Routing
- Columns and Rows are numbered
- Two rules apply:
  1. No East-North Turns in even columns
     No East-South Turns in even columns
  2. No South-West Turns in odd columns
     No North-West Turns in odd columns
- Adaptive routing:
  - Deadlock free
  - Load Balancing is possible
  - Error Tolerant
- Routing algorithm has to take care of reachability

Dijkstra-Algorithm

- Finds the shortest path from a starting node to all other nodes in a known topology

A. Set distance to infinity for all vertices
B. Repeat until all vertices are marked:
   1. Mark the vertex with smallest distance
   2. For all neighbors add current distance and edge weight
   3. If sum is smaller than current distance
      → update distance
      → set marked vertex as predecessor

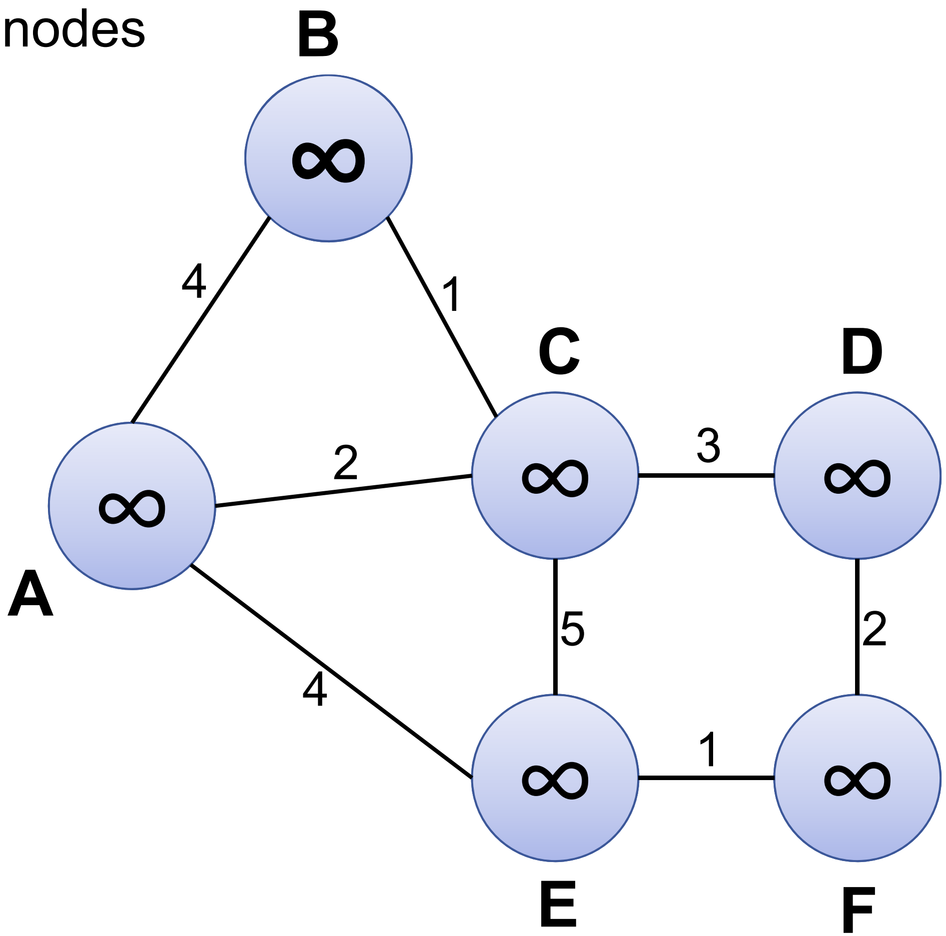


Dijkstra-Algorithm - Example

Step A

Vertex	Dist.	Pred.
A	∞	A
B	∞	-
C	∞	-
D	∞	-
E	∞	-
F	∞	-

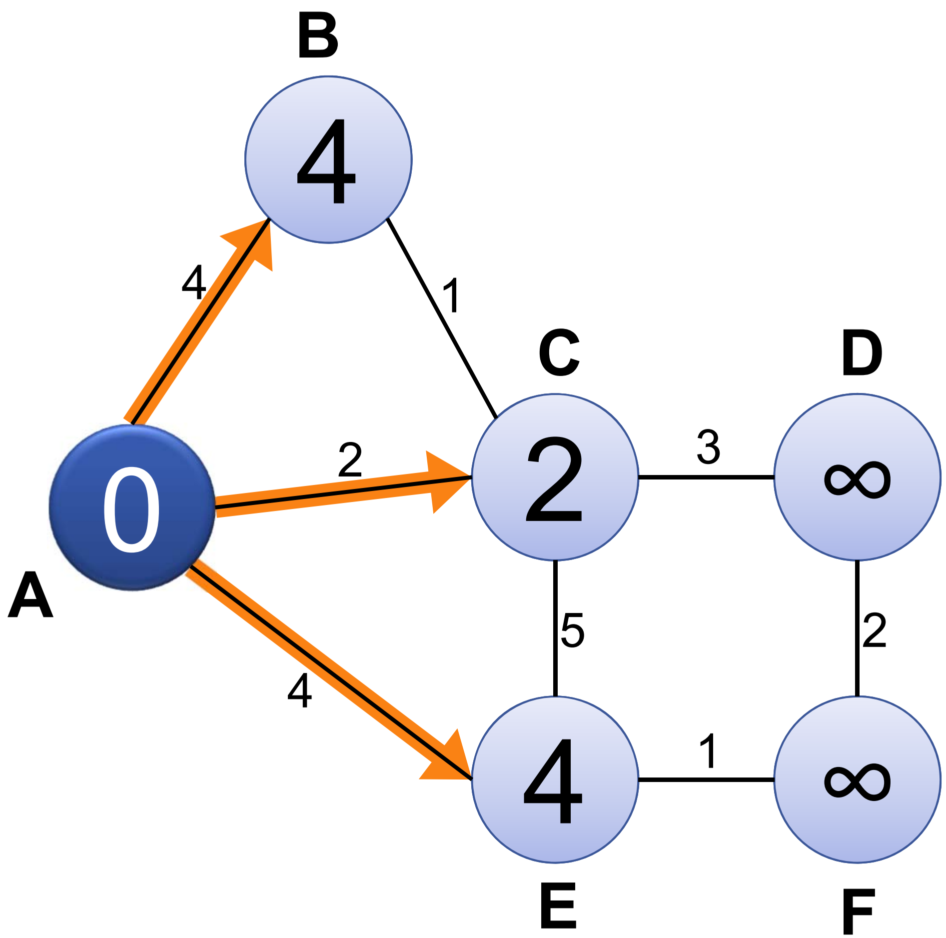
- Find the shortest path from node A to all other nodes



Dijkstra-Algorithm - Example

Step B

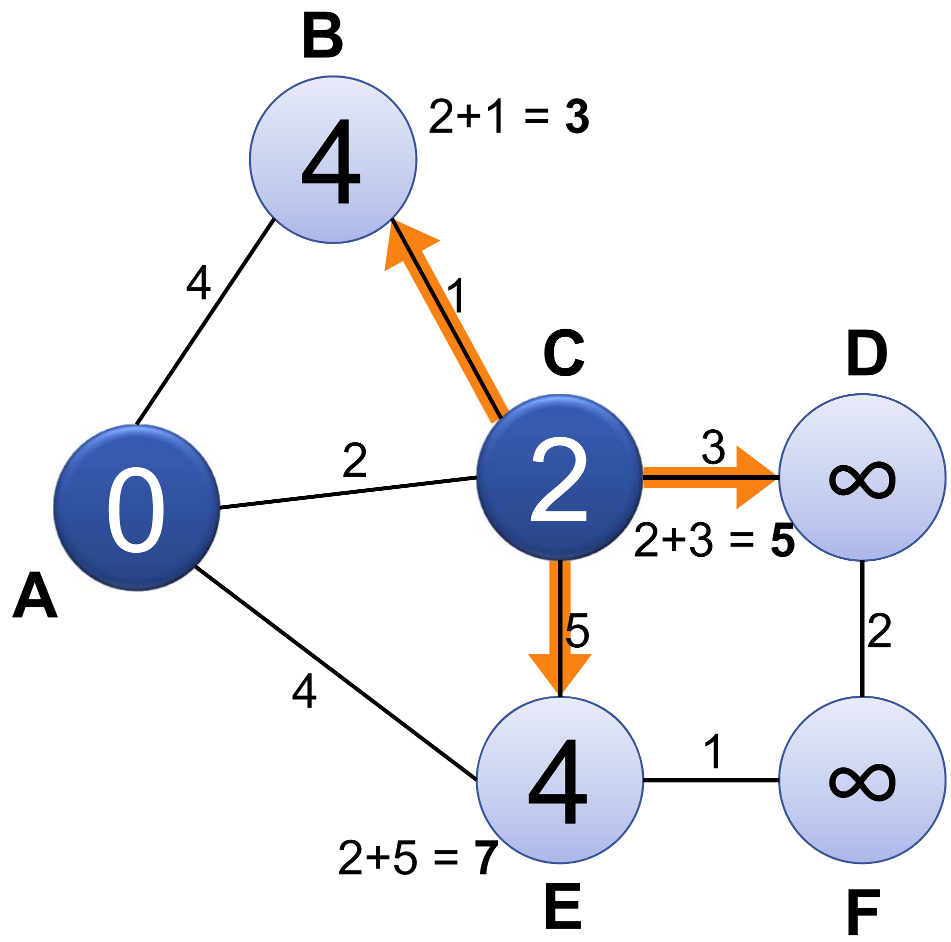
Vertex	Dist.	Pred.
A	0	A
B	4	A
C	2	A
D	∞	-
E	4	A
F	∞	-



Dijkstra-Algorithm - Example

Step B

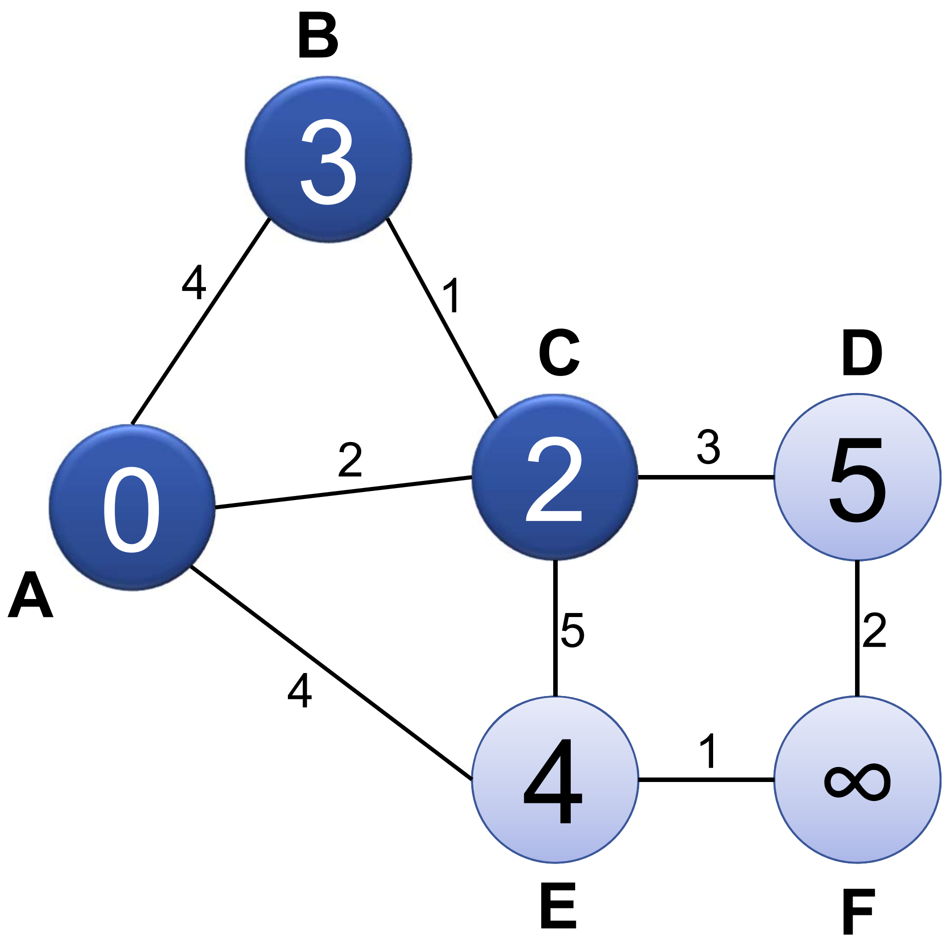
Vertex	Dist.	Pred.
A	0	A
B	3	C
C	2	A
D	5	C
E	4	A
F	∞	-



Dijkstra-Algorithm - Example

Step B

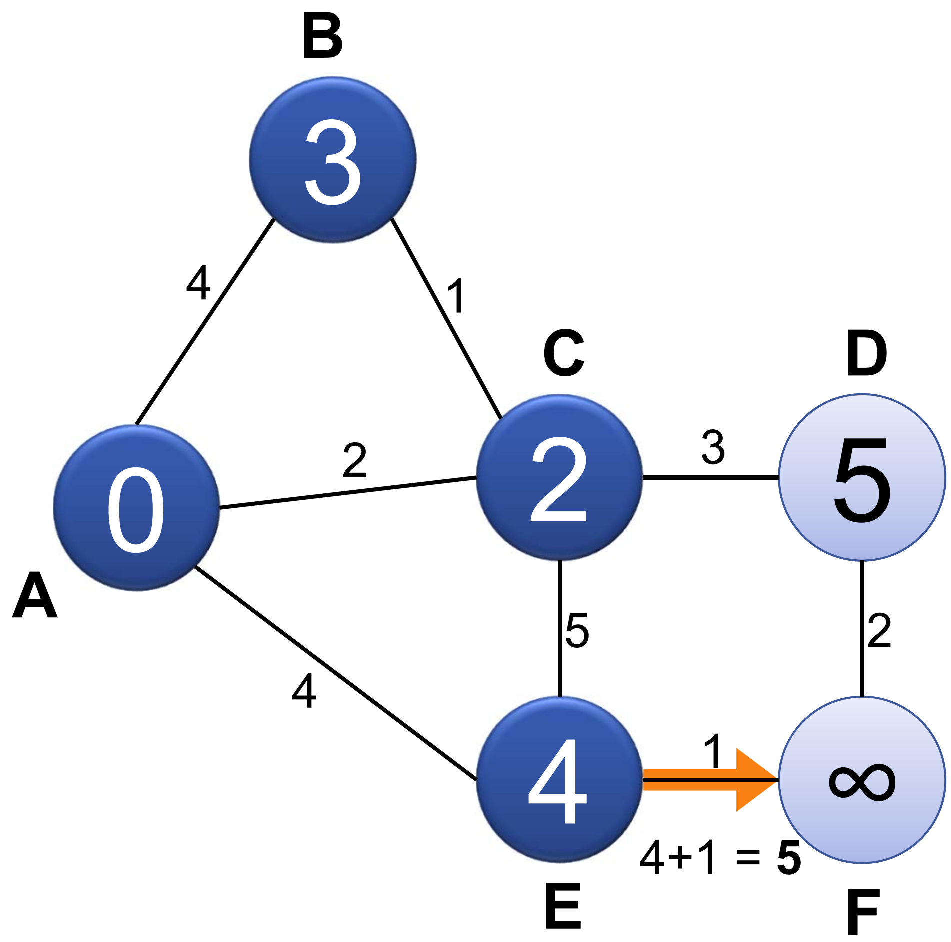
Vertex	Dist.	Pred.
A	0	A
B	3	C
C	2	A
D	5	C
E	4	A
F	∞	-



Dijkstra-Algorithm - Example

Step B

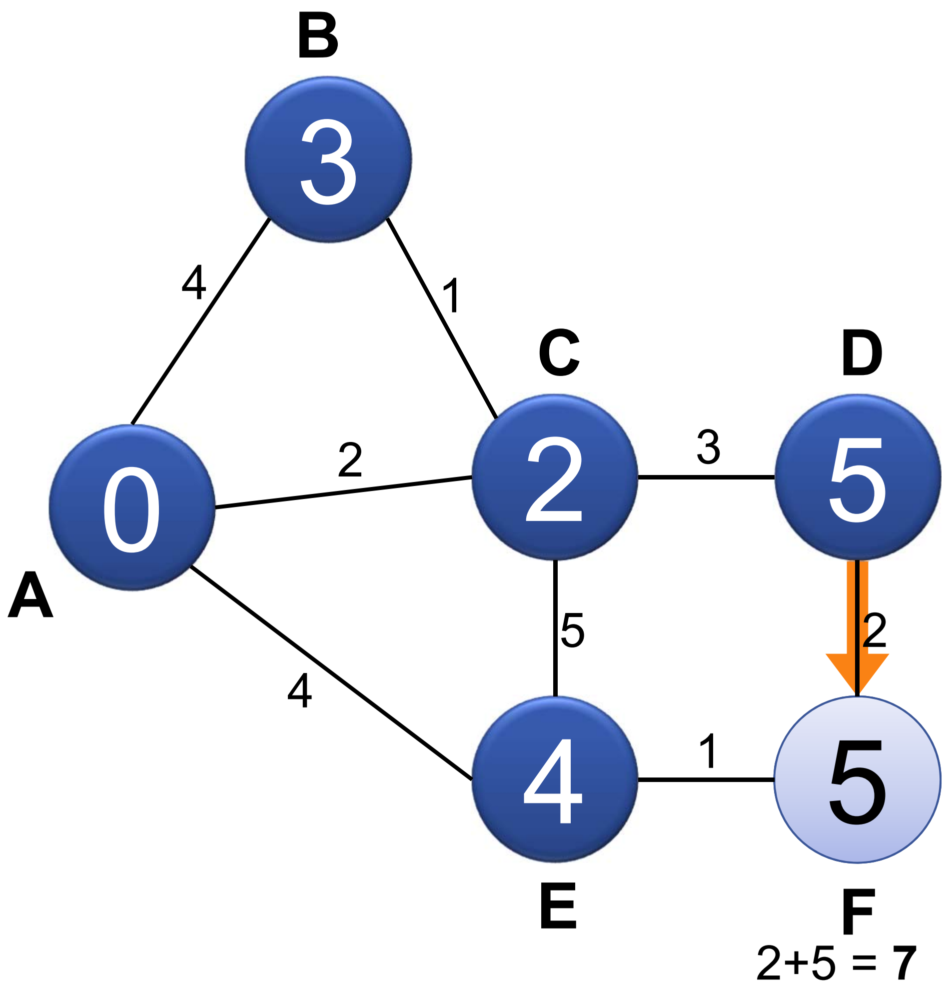
Vertex	Dist.	Pred.
A	0	A
B	3	C
C	2	A
D	5	C
E	4	A
F	5	E



Dijkstra-Algorithm - Example

Step B

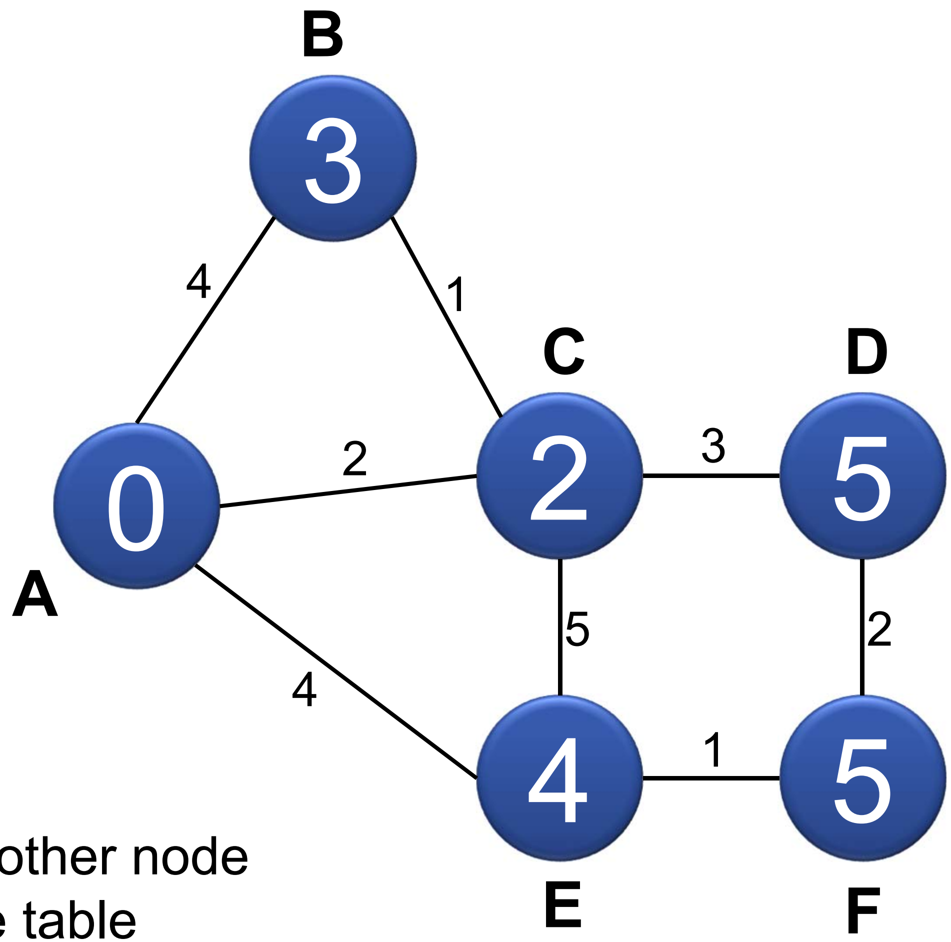
Vertex	Dist.	Pred.
A	0	A
B	3	C
C	2	A
D	5	C
E	4	A
F	5	E



Dijkstra-Algorithm - Example

Step B

Vertex	Dist.	Pred.
A	0	A
B	3	C
C	2	A
D	5	C
E	4	A
F	5	E



- The shortest path from A to any other node can be constructed now from the table
- For complete routing table, algorithm is repeated for all nodes

Djikstra – Why use it?

- Application for non-mesh-based networks
  - Distance may not be the only criteria
- Weights on Edges
  - traffic loads
  - bandwidth
  - latencies
  - …
- Djikstra can compute routes that are optimal regarding given constraints (weights) other than pure Manhattan distance

Evaluation criteria for routing algorithms

- Number of hops
  - How many links are used to reach the destination?
- Latency
  - Different links can have different bandwidth
  - Can be correlated with number of hops if same time per hop
- Guaranties on bandwidth, throughput, real-time capability, etc.
  - Latency has to be known
  - Waiting time inside router has to be limited
- Robustness, failure tolerance
  - System should be still operational if one or more links are broken
  - Flow control to make sure that packets reach destination

Outlook

- Networks
  - Node structure
- OSI Model
- Network coupling