WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control

Outline

- Background
- WirelessHART Architecture
- Network Management and Data Management
- Implementation and Deployment
- Future Work

Background

Smart Control Systems
Smart Analytical
Smart Asset Optimization
Smart Final Control
Smart Wireless
Smart Measurement
Smart Machinery Health
Smart Safety

History

HART (Highway Addressable Remote Transducer)

- Bi-directional industrial field communication protocol
- Used to communicate between field devices and host systems
- The global installed base of HART-enabled devices is 20 million

WirelessHART (Released in Sept. 2007)

- Wireless extension of HART Standards
- The first open wireless communication standard for industrial process control applications

Properties

Real-Time

- TDMA Technology
- Centralized Network Management

Reliability

- Channel Hopping and Channel Blacklisting
- Mesh Networking

Security

- Data Integrity on MAC Layer
- Data Confidentiality on the Network Layer

WirelessHART Architecture

WirelessHART Architecture

OSI Layer	HART	
Application	Command Oriented. Predefined Data Types and Application Procedures	
Presentation		
Session		
Transport	Auto-Segmented transfer of large data sets, reliable stream transport, Negotiated Segment sizes	
Network		Power-Optimized Redundant Path, Mesh to the edge Network
Data Link	A Binary, Byte Oriented, Token Passing, Master/Slave Protocol	Secure, Time Synched TDMA/ CSMA, Frequency Agile with ARQ
Physical	Simultaneous Analog & Digital Signaling 4-20mA Copper Wiring	2.4 GHz Wireless, 802.15.4 based radios, 10dBm Tx Power
	Wired FSK/PSK & RS 485	**Wireless 2.4 GHz**

Summary of PDU Format

No. Bytes	Description
	Physical Layer
4	Preamble
1	Delimiter
1	Byte Count
	Data-Link Layer
1	"0x41"
1	Address Specifier
1	Sequence Number
2	Network ID
2/8	Destination
2/8	Source
1	DLPDU Specifier
	Network Layer
1	Control
1	TTL
2	ASN snippet
2	Graph ID
2/8	Destination
2/8	Source
2/4/6/8/10	(Optional) Proxy/Source Routing
	Security
1	Security Control
1/4	Counter
4	MIC
	Transport Layer
1	Transport Control
1	Device Status
1	Extended Device Status
	Application Layer
2	Command
1	Byte Count
--	Data
4	MIC
2	CRC

(a) WirelessHART

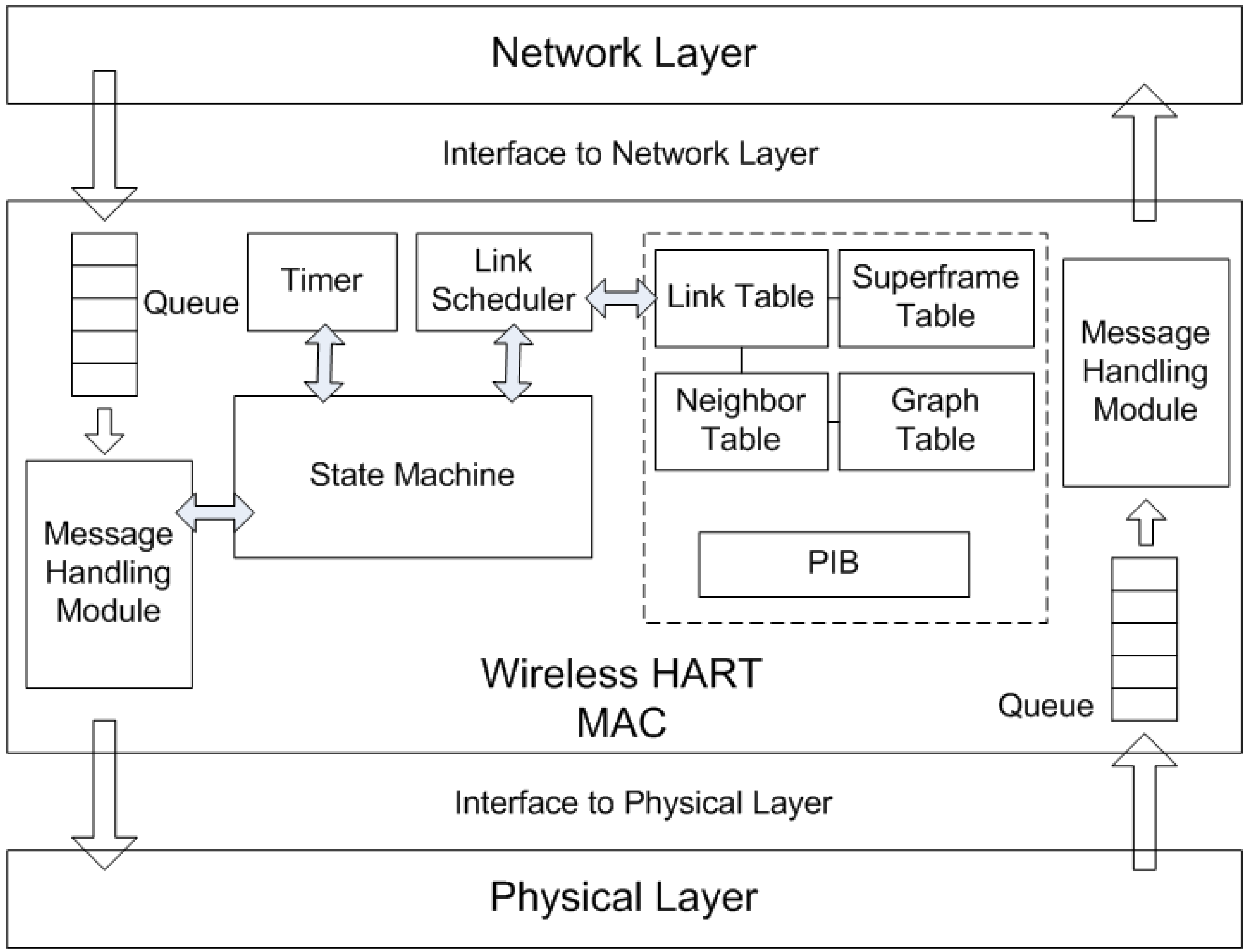
No. Bytes	Description
	Physical Layer
5 - 20	Preamble
	Data-Link Layer
1	Delimiter
1/5	Address
	Application Layer
1	Command
1	Byte Count
1	Response Code
1	Device Status
--	Data
1	Check Byte

(b) Wired HART

WirelessHART Architecture

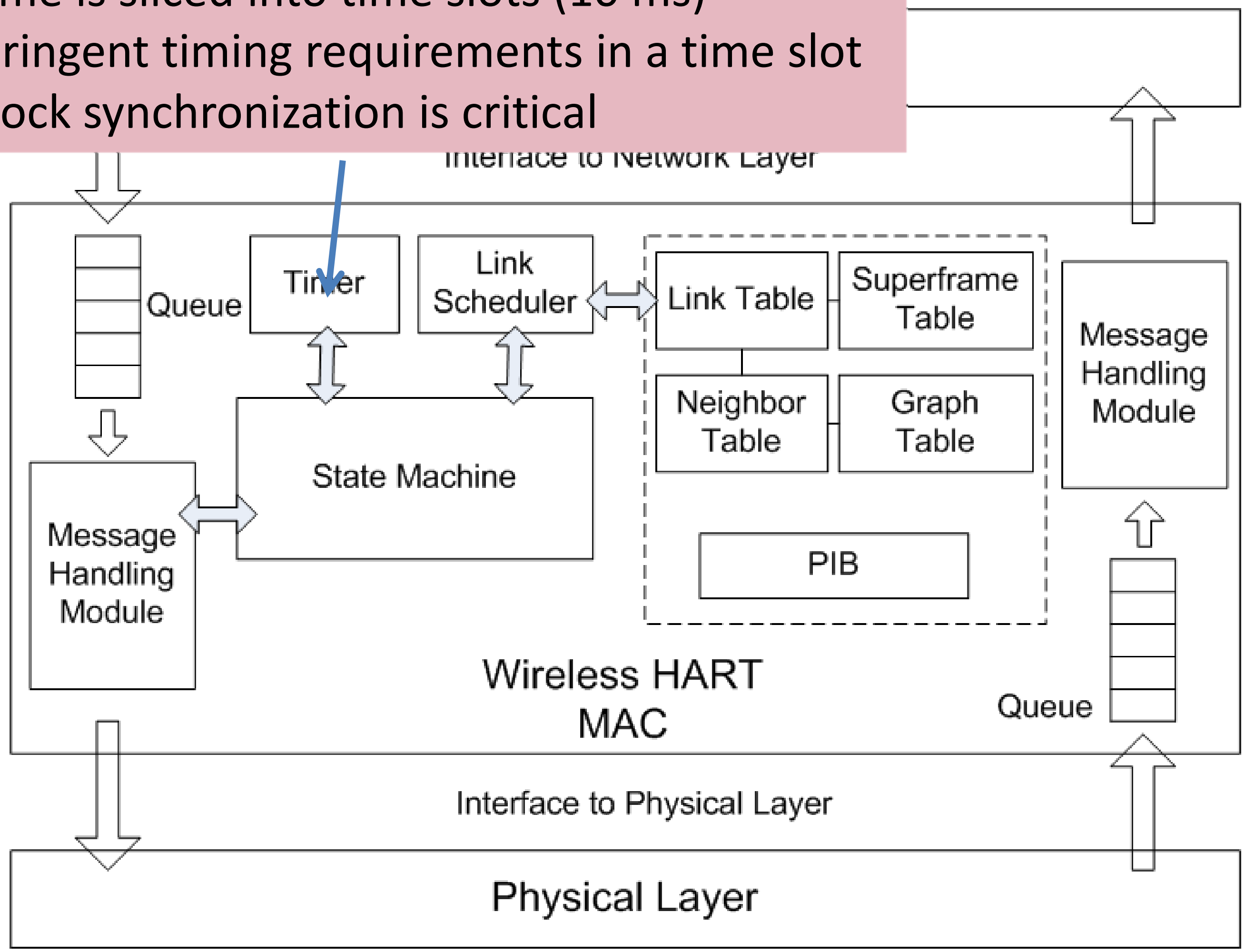
- Physical Layer (IEEE 802.15.4)
- Data Link Layer
- Network Layer and Transport Layer
- Application Layer
- Security

Data Link Layer



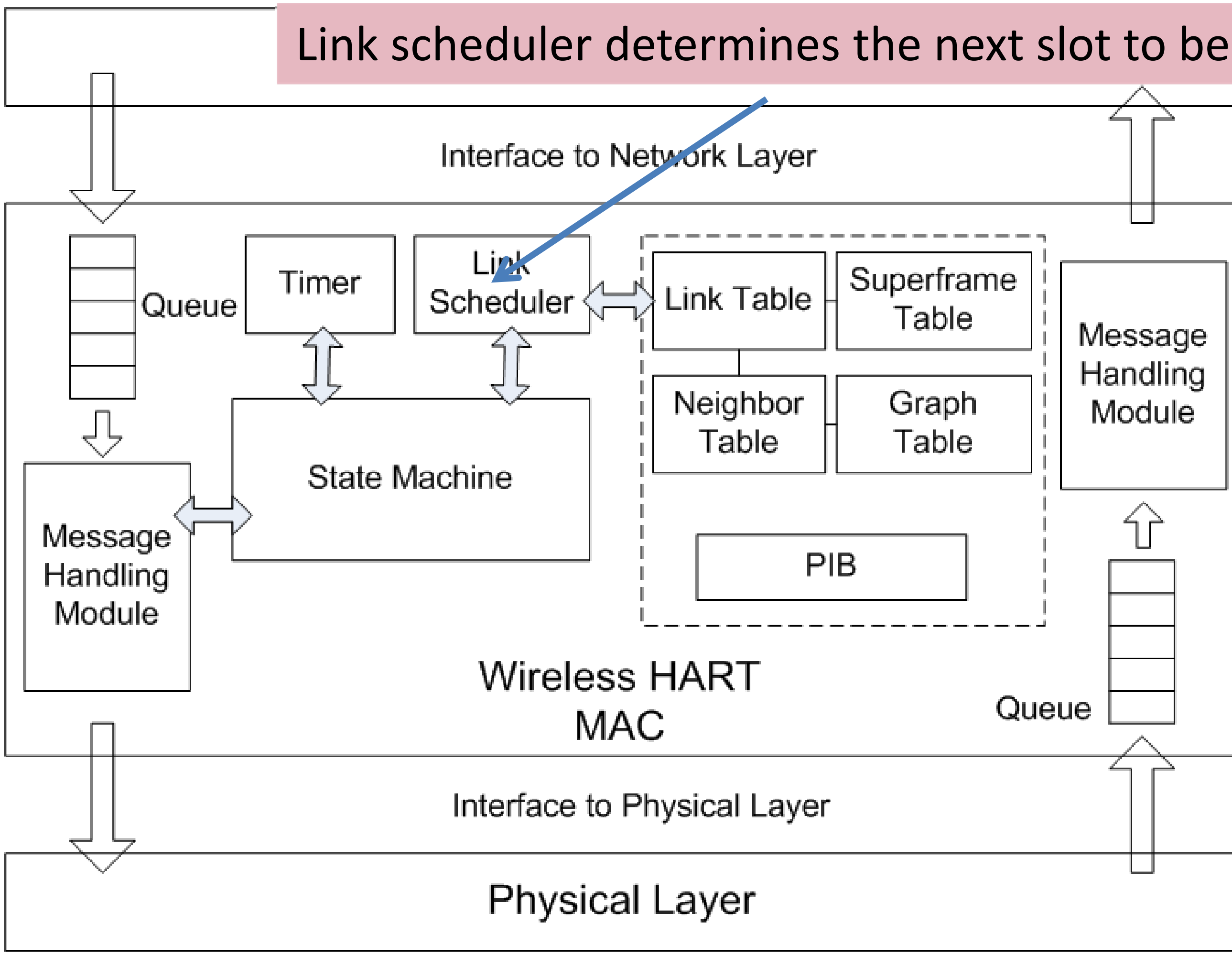
Data Link Layer

- Time is sliced into time slots (10 ms)
- Stringent timing requirements in a time slot
- Clock synchronization is critical

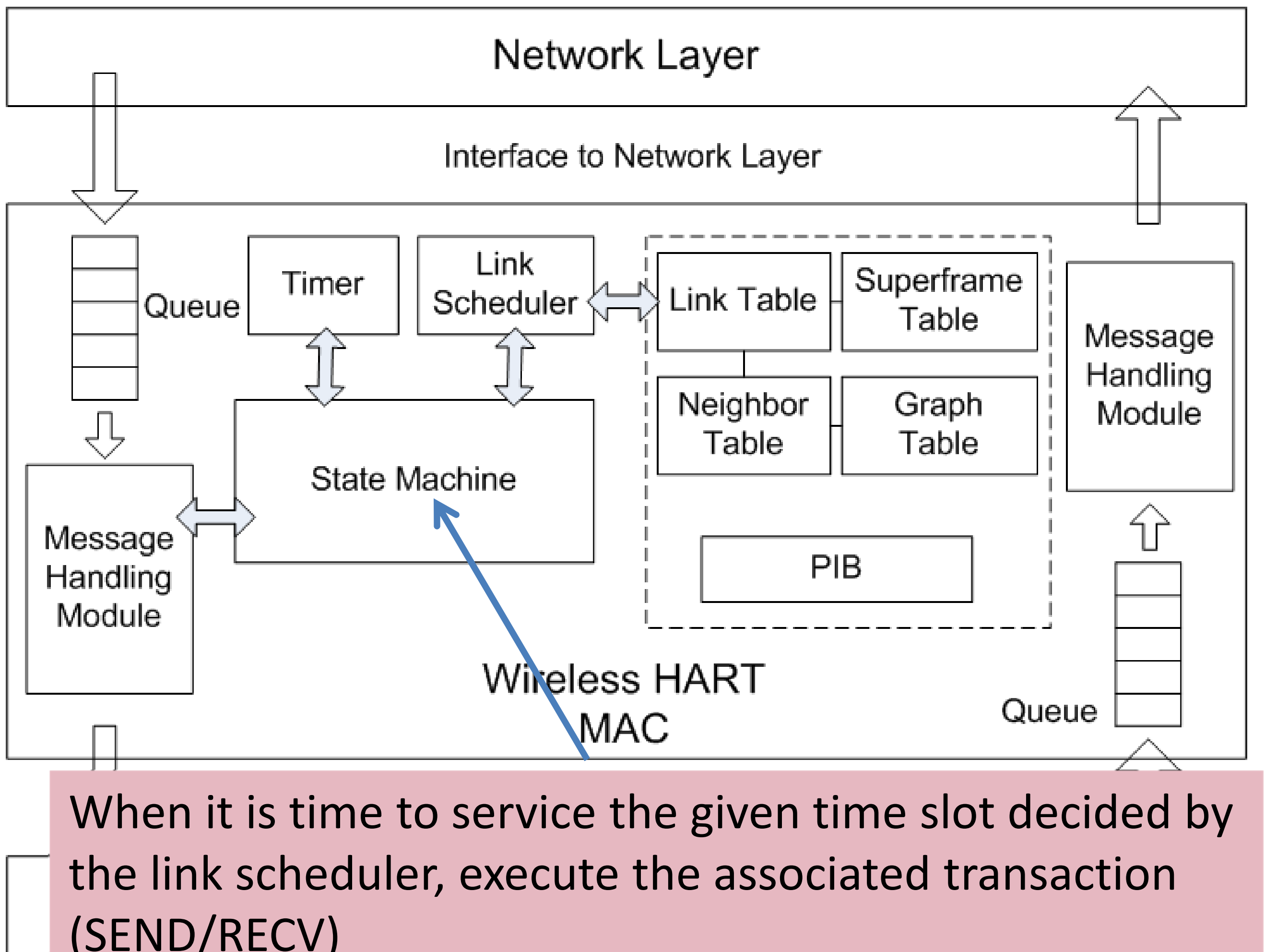


Data Link Layer

Link scheduler determines the next slot to be serviced

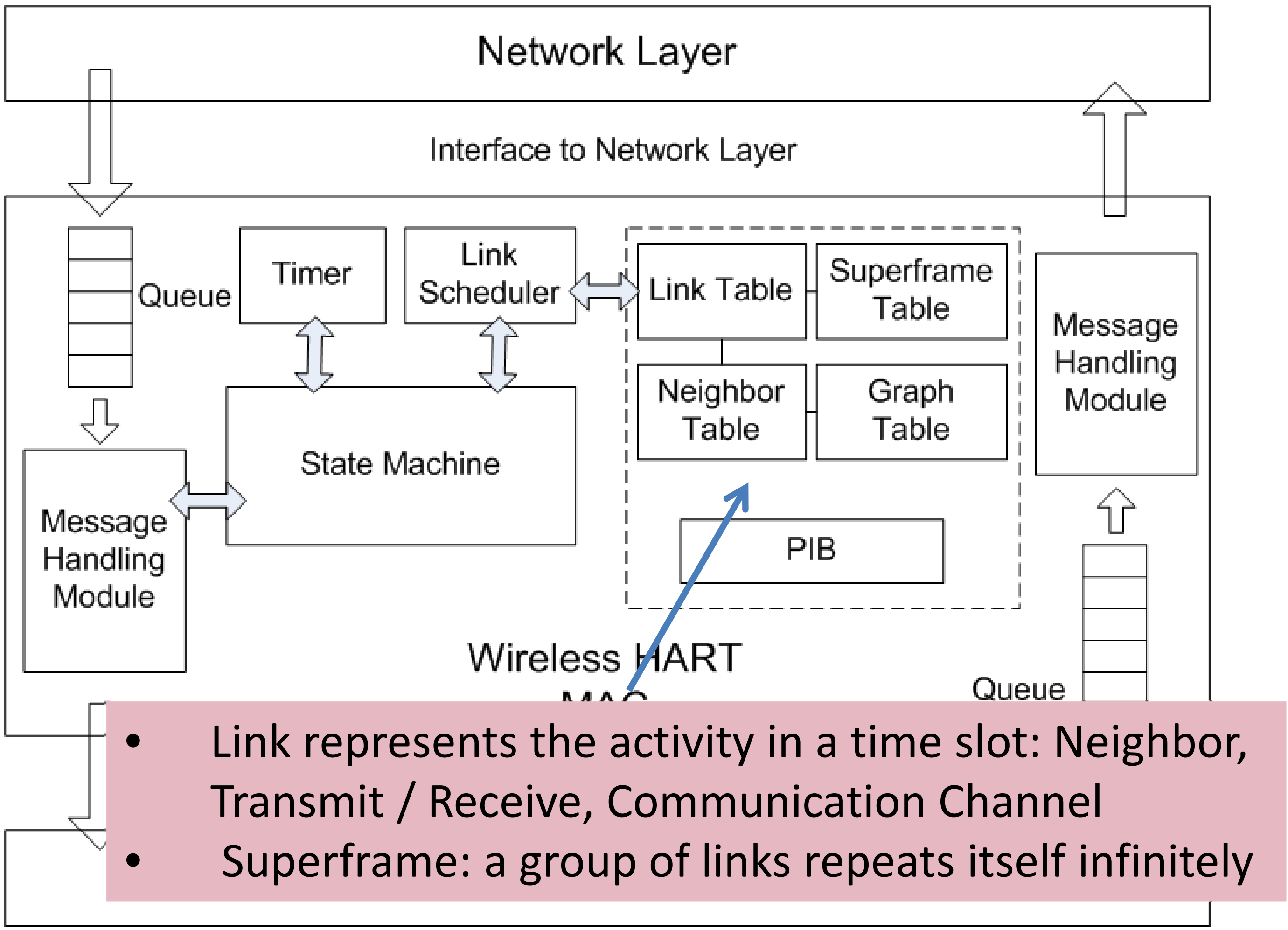


Data Link Layer



When it is time to service the given time slot decided by the link scheduler, execute the associated transaction (SEND/RECV)

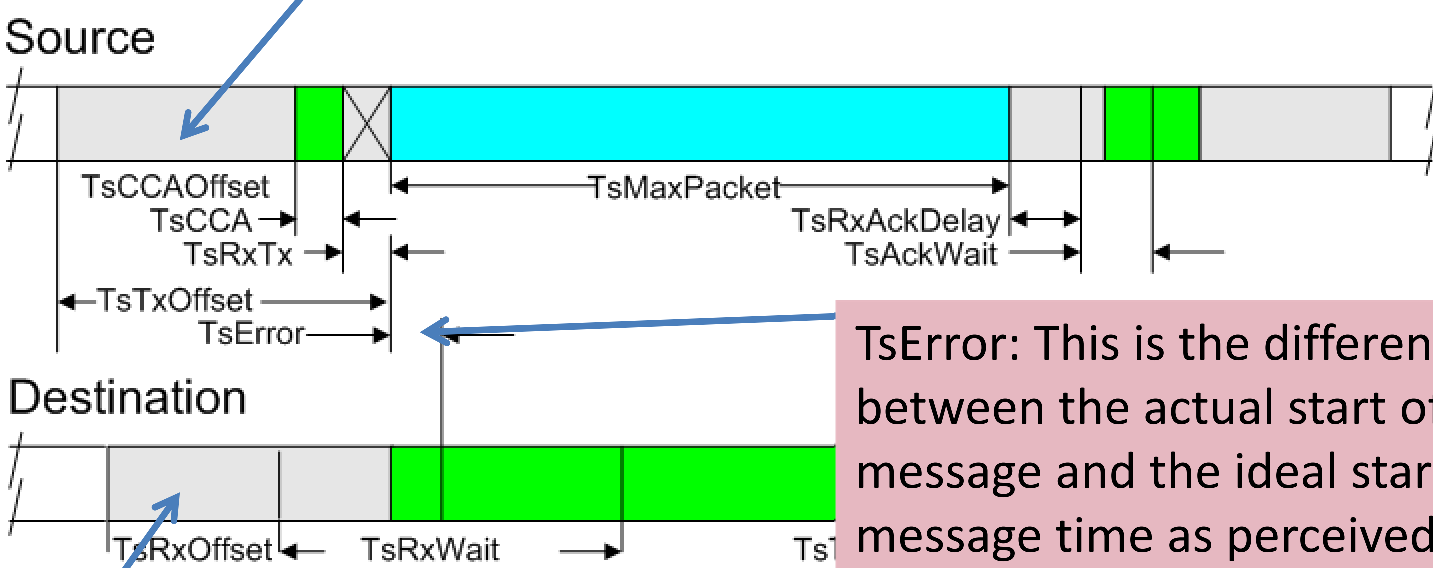
Data Link Layer



- Link represents the activity in a time slot: Neighbor, Transmit / Receive, Communication Channel
- Superframe: a group of links repeats itself infinitely

WirelessHART Slot Timing

TsCCAOffset: Start of slot to beginning of CCA

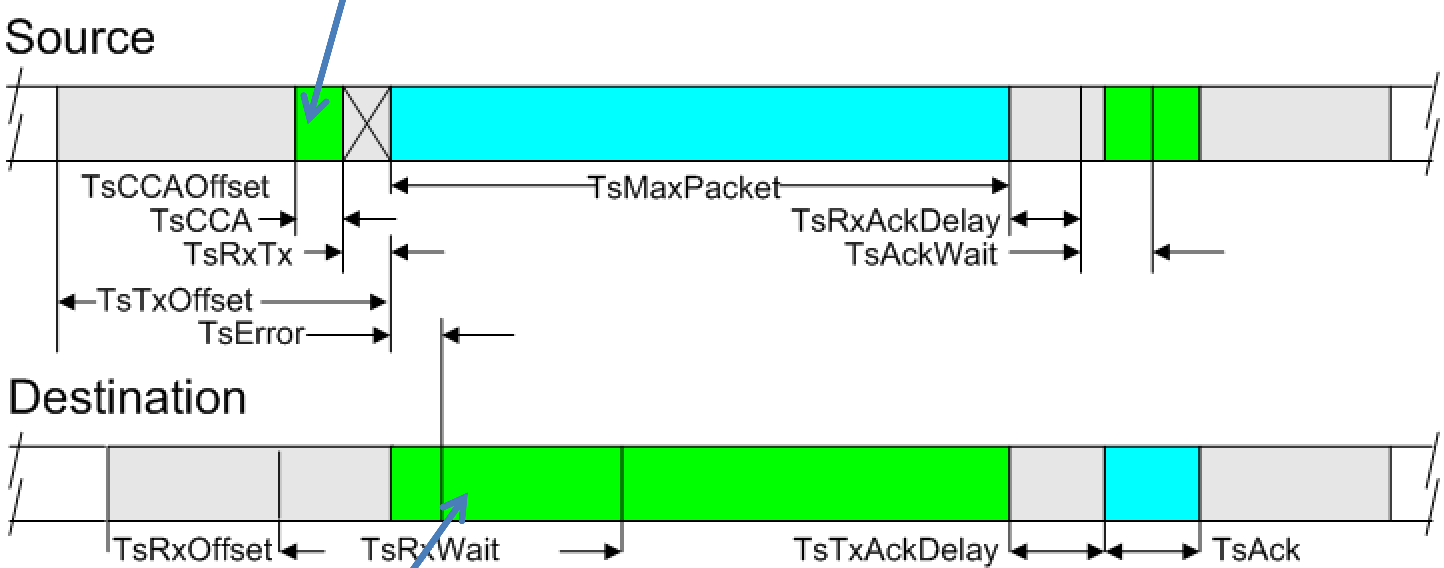


TsError: This is the difference between the actual start of message and the ideal start of message time as perceived by the receiving device.

TsRxOffset: Start of the slot to when transceiver must be listening.

WirelessHART Slot Timing

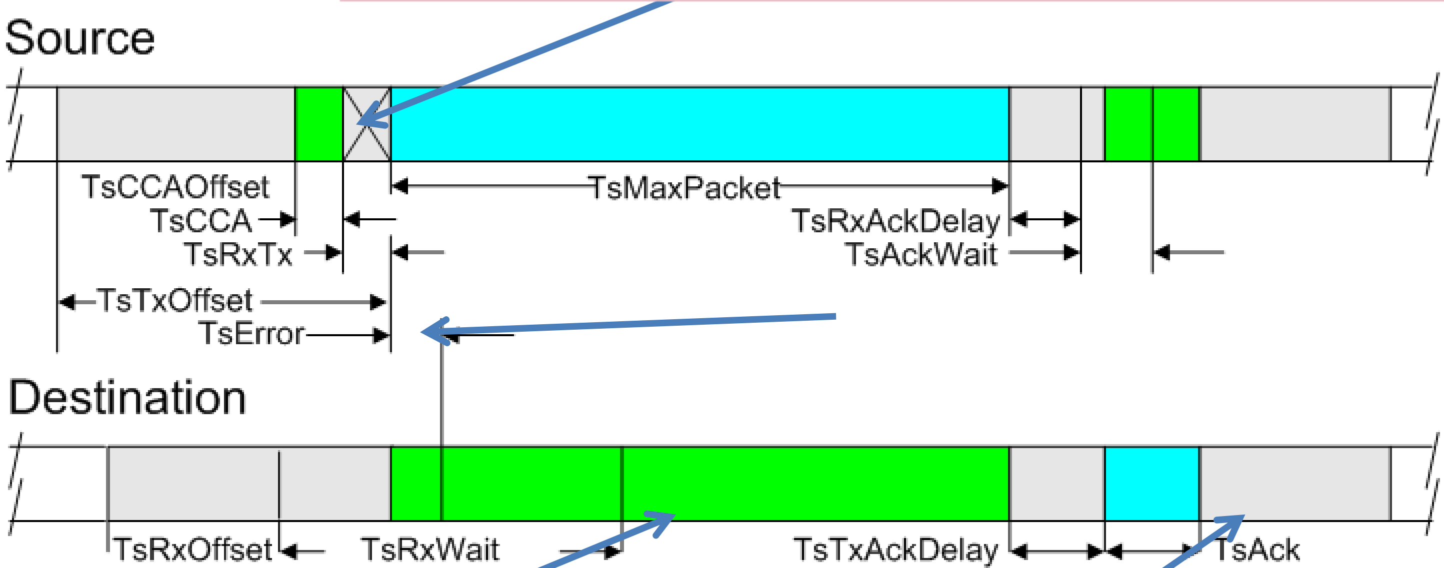
TsCCA: Time to perform CCA



TsRxWait :The minimum time to wait for start of message.

WirelessHART Slot Timing

TsRxTx: The longer of the time it takes to switch from receive to transmit or vice versa.



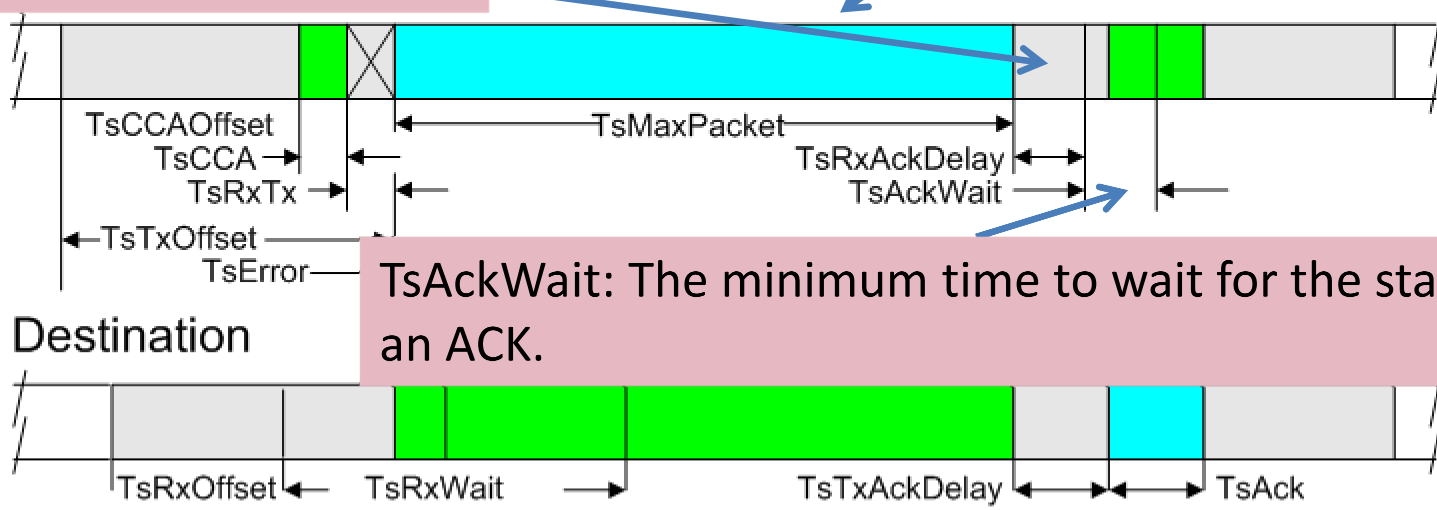
TsTxAckDelay: End of message to start of ACK.

TsAck: Time to transmit an ACK.

WirelessHART Slot Timing

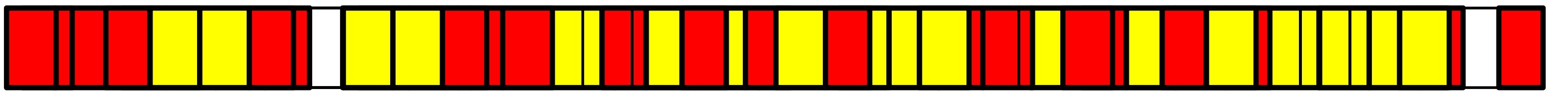
TsRxAckDelay: End of message to when transceiver must be listening for ACK.

TsMaxPacket: The amount of time it takes to transmit the longest possible message



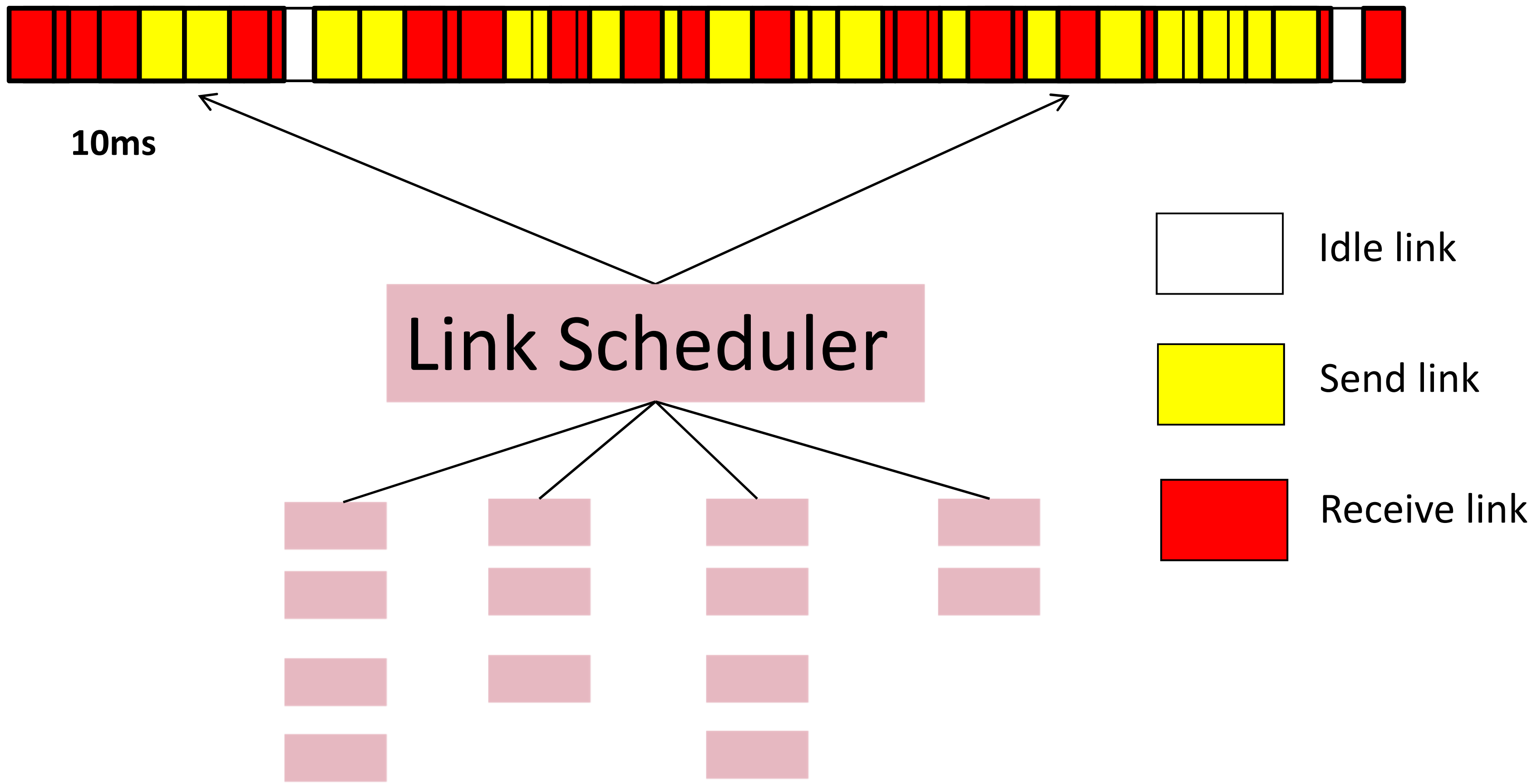
TsAckWait: The minimum time to wait for the start of an ACK.

TDMA-based Data Link Layer



- Link: activity in a time slot
  - Neighbor
  - Send/Receive
  - Communication channel
- Superframe: a group of links
  - Repeat itself infinitely
  - A device can support several superframes

TDMA-based Data Link Layer

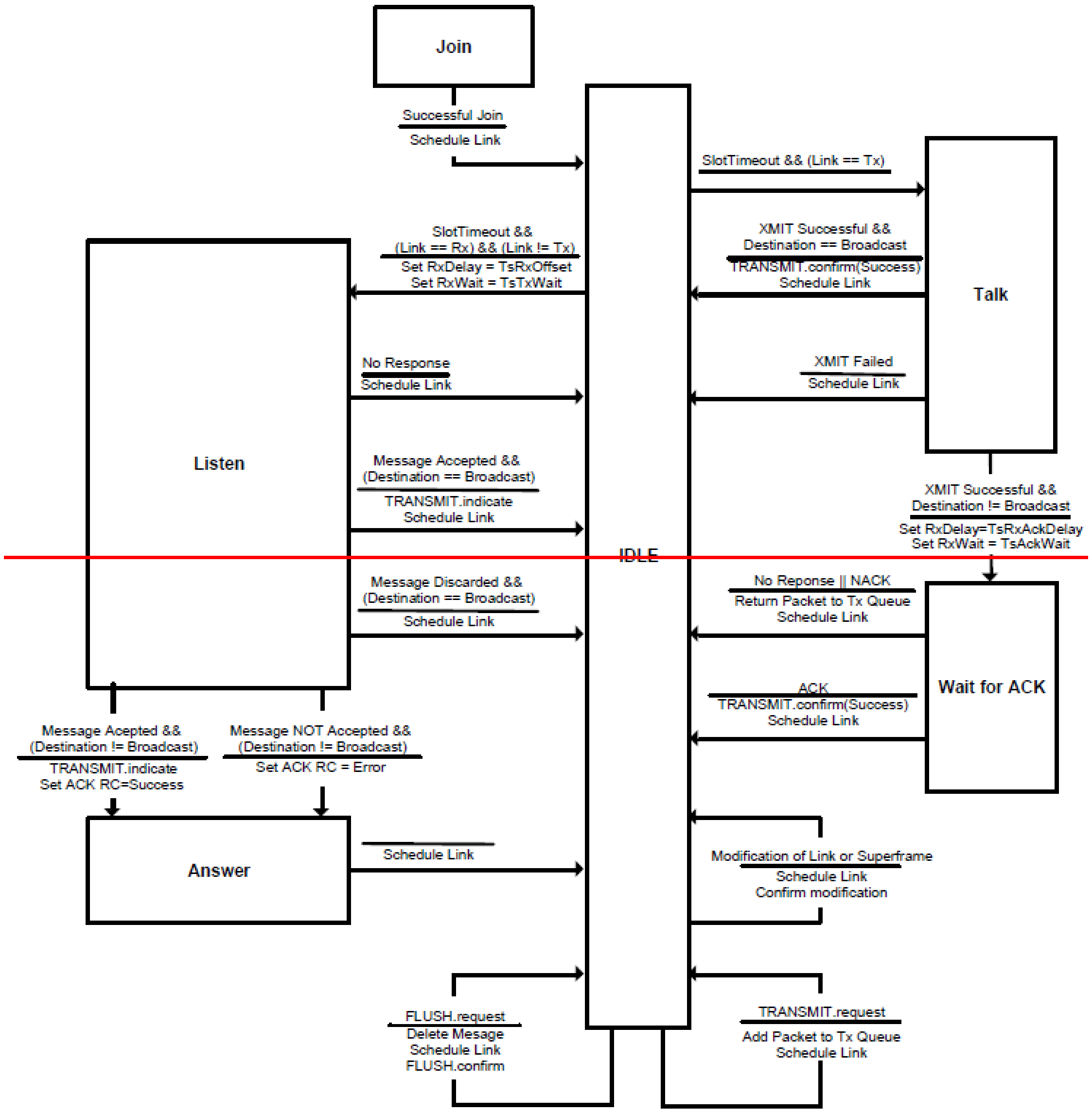


Priority queues for data link layer packets

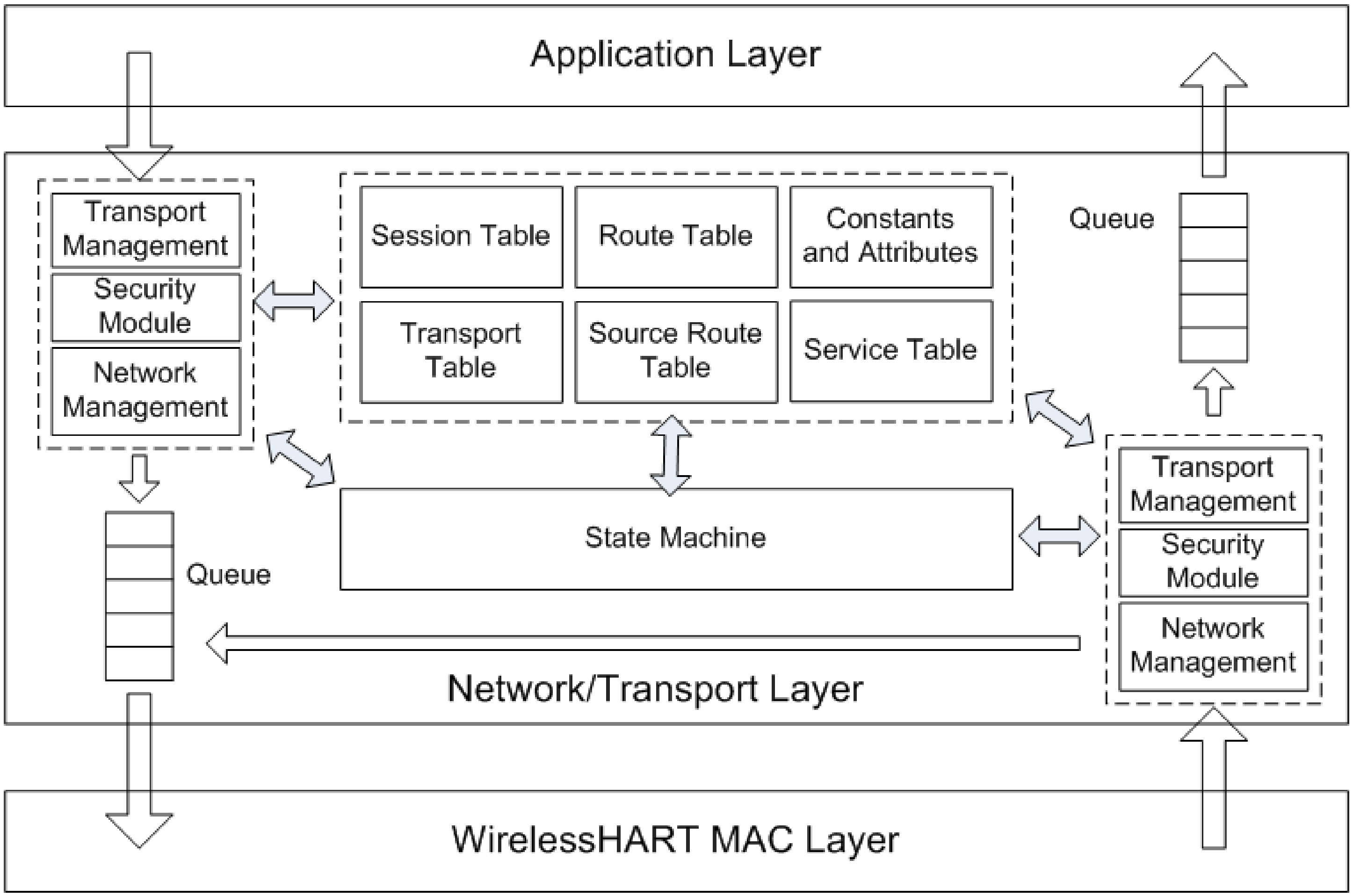
Channel Hopping and Blacklisting

- Channel Hopping
  - Spread WirelessHART communication in all active physical channels in 802.15.4 (up to 16 channels)
  - ActiveChannel = (ChannelOffset + ASN) % number of Active Channels
- Channel Blacklisting
  - Restrict channel hopping to selected channels in the RF band.
  - protect a wireless service that uses a fixed portion of the RF band.

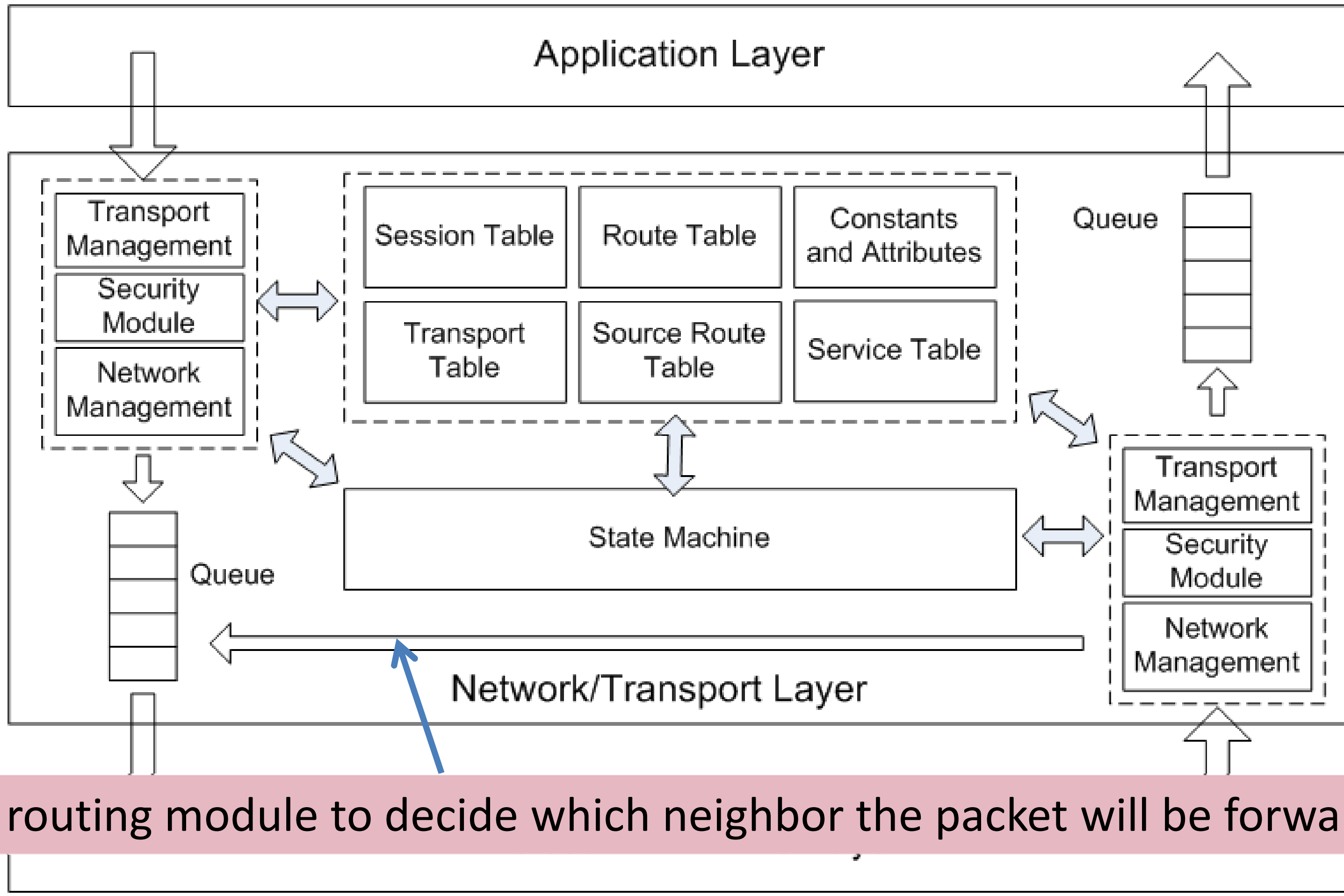
MAC State Machine



Network and Transport Layer

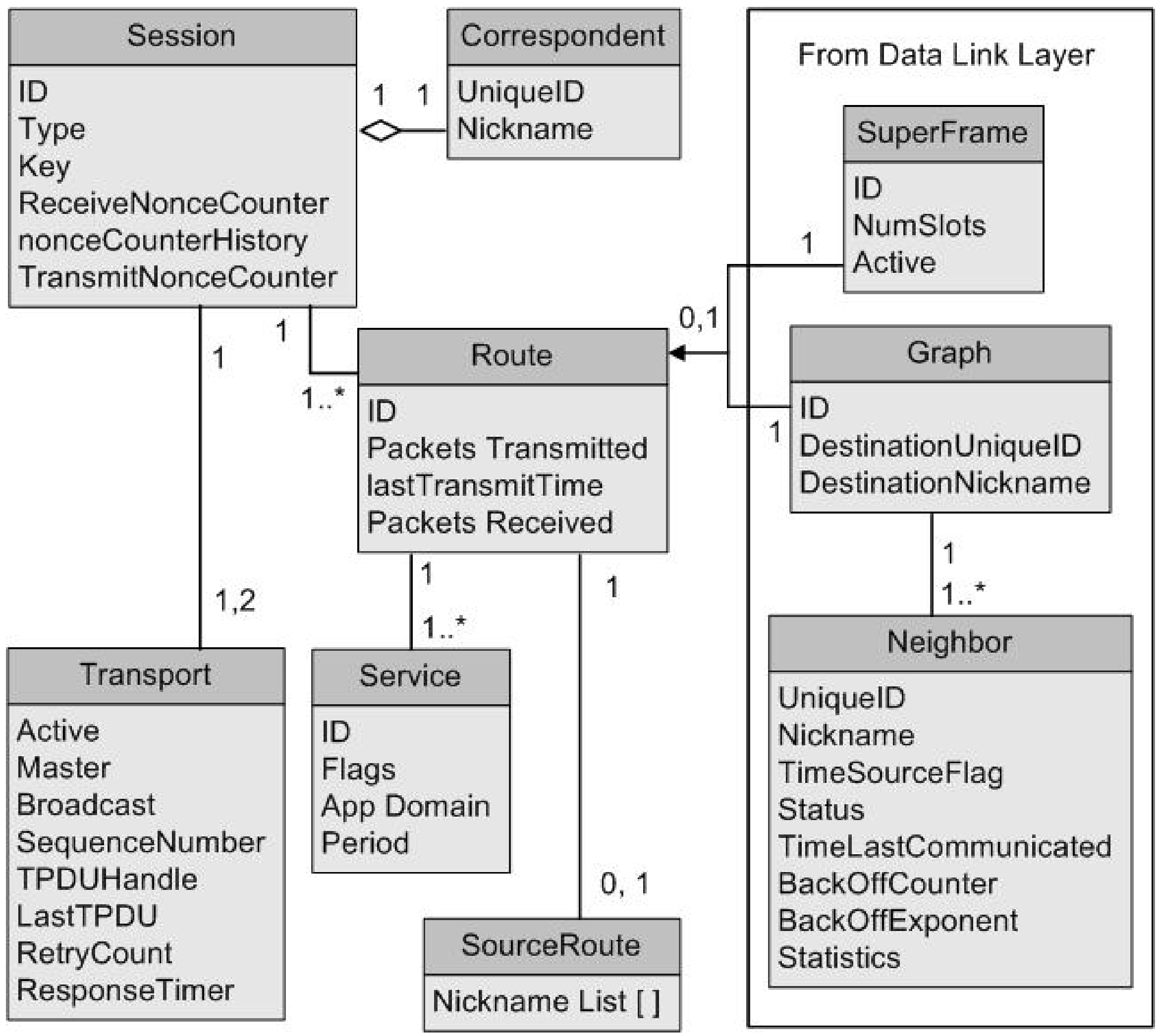


Network and Transport Layer



A routing module to decide which neighbor the packet will be forwarded to.

Network/Transport Layer Data Model



Graph and Source Routing

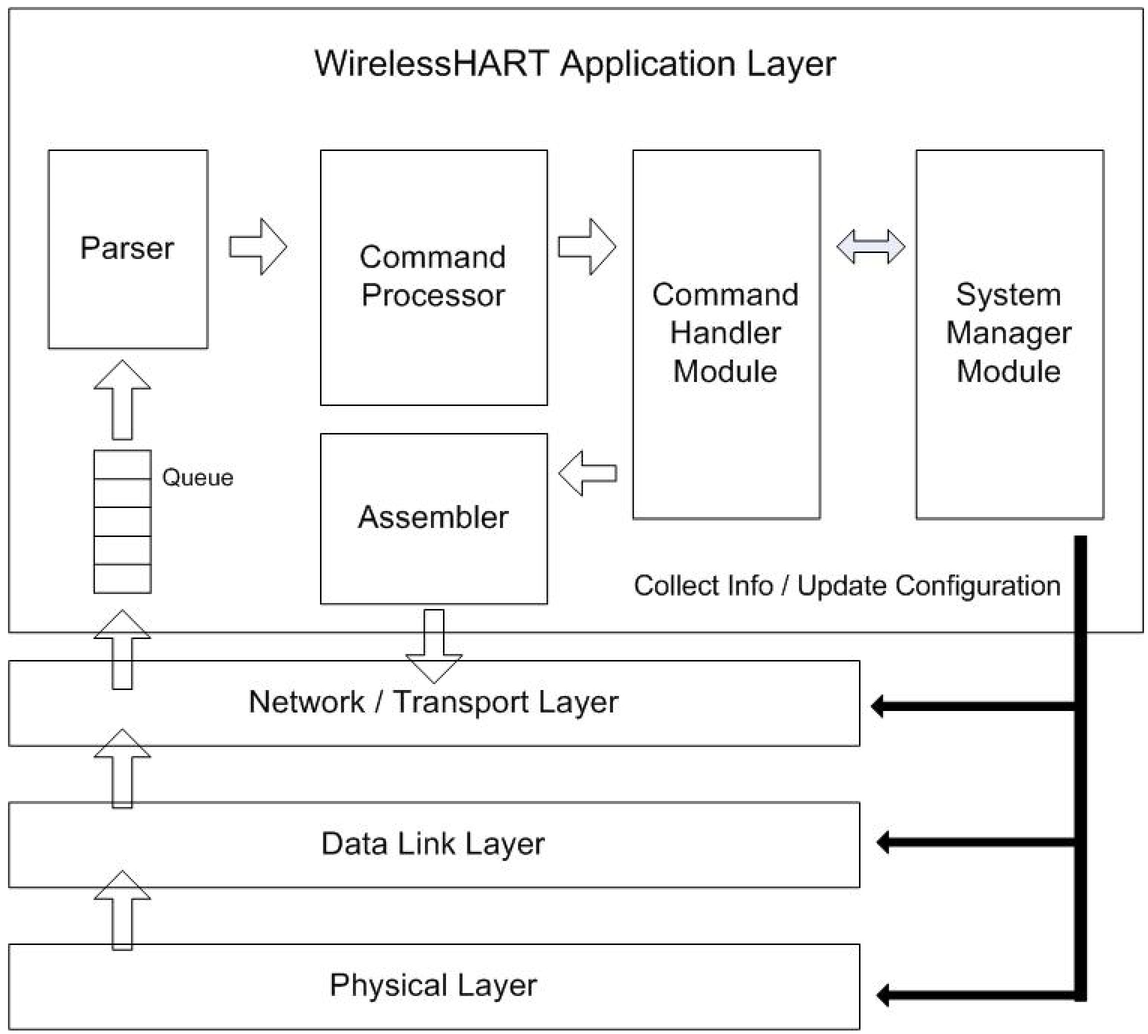
- Graph Routing
  - A graph is a collection of paths that connect devices.
  - Network Manager creates and downloads them to devices.
  - Strict reliability requirements
- Source Routing
  - Aims at network diagnostics.
  - The path is explicitly included in the header
- Proxy Routing and Superframe Routing

Application Layer

Commend-oriented: communications between peers are represented as command requests and responses.

- Three classes of Commands
  - Universal Commands
  - Common Practice Commands
  - Wireless Commands

Application Layer



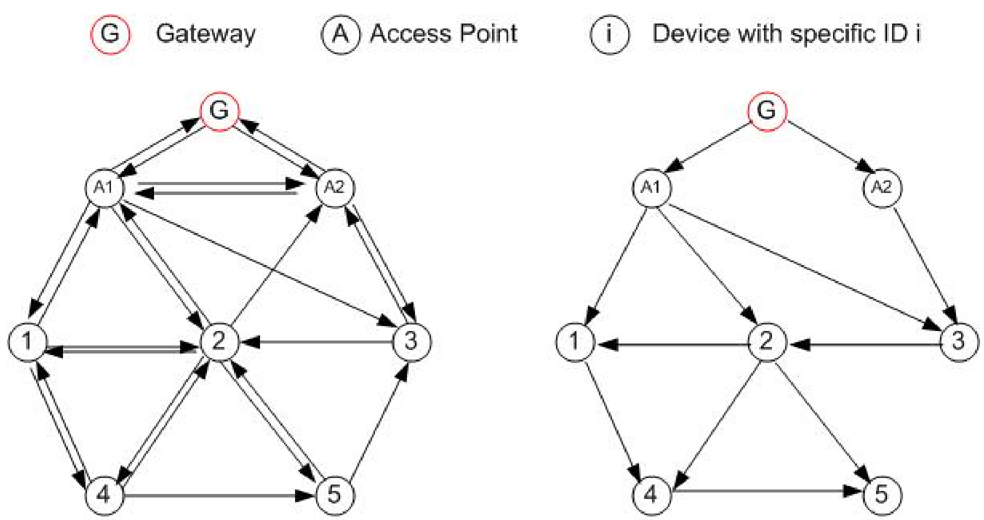
Network Management Techniques

Introduction

- Challenge
  - The complexity of network management is pushed to the centralized manager but engineering decisions can have large performance impact.
- Goals
  - Achieve reliable graph routing in WirelessHART network
  - Achieve real-time communication by deterministic link and channel assignment
  - Evaluate their performance in industrial environments

Reliable Graph Routing

- Reliable Broadcast Graph (G_B)
  - G_B is a graph connecting Gateway (GW) downward to all DEVs
  - Broadcasts common configuration and control messages
  - Each DEV has at least two parents in G_B

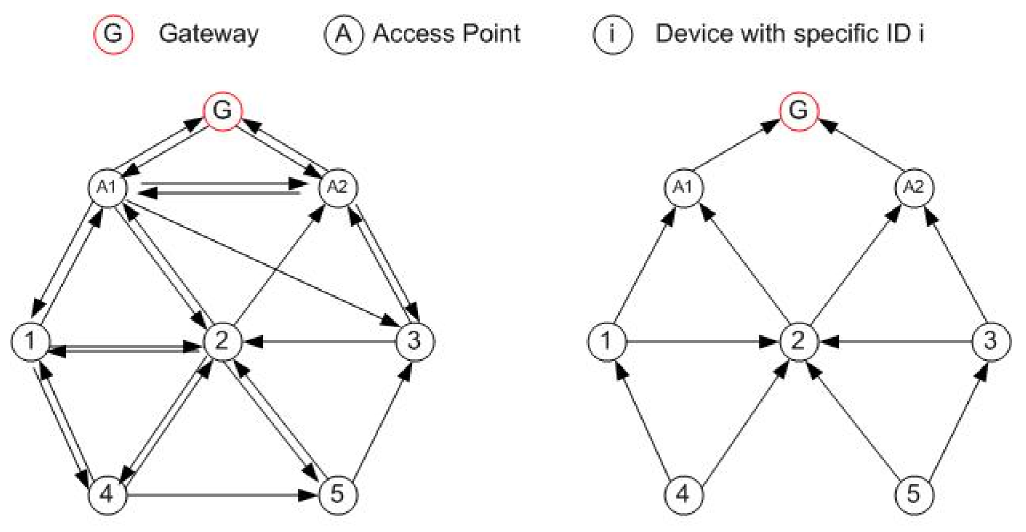


(a) Original network topology

(b) Broadcast graph

Reliable Graph Routing (Cont.)

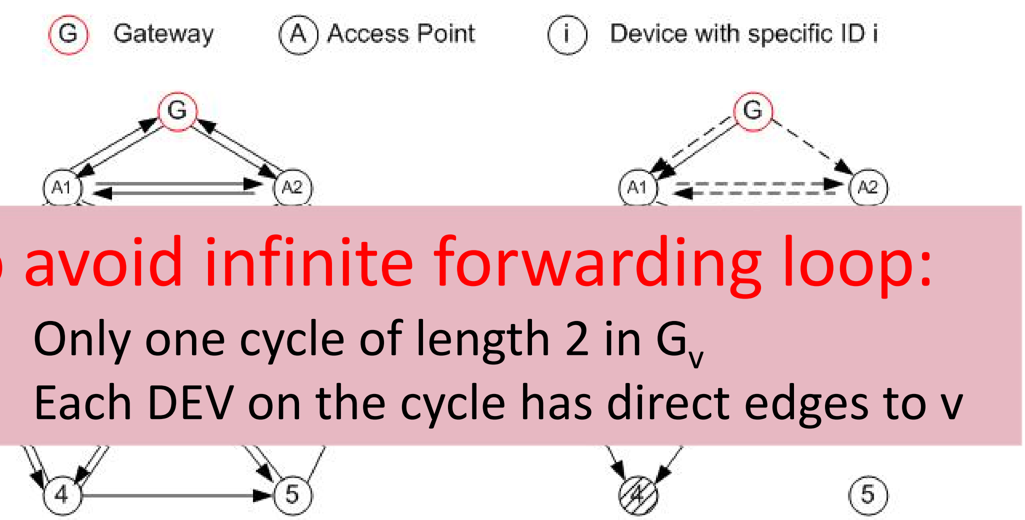
- Reliable Uplink Graph (G_U)
  - G_U is a graph connecting all DEVs upward to the Gateway
  - DEVs propagate periodic process data
  - Each DEV has at least two children in G_U
  - Both G_B and G_U have no fewer than 2 Access Points



(a) Original network topology (b) Uplink graph

Reliable Graph Routing (Cont.)

- Reliable Downlink Graph (G_v)
  - The graph from the Gateway to DEV v
  - Transmit unicast messages from the GW and NM to v
  - Each intermediate DEV has at least two children in G_v
  - There exists at least one directed cycle in G_v



To avoid infinite forwarding loop:

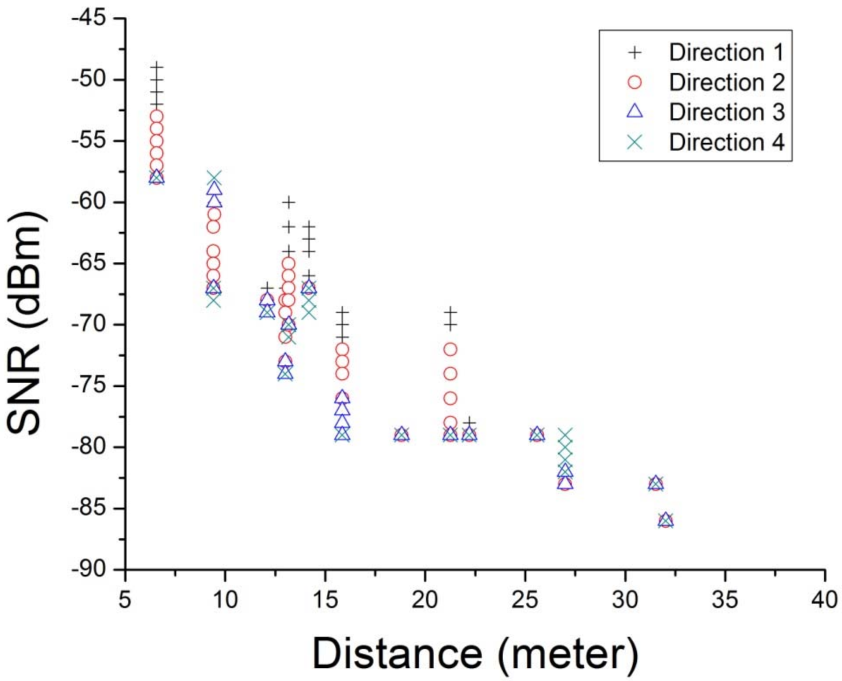
1) Only one cycle of length 2 in G_v
2) Each DEV on the cycle has direct edges to v

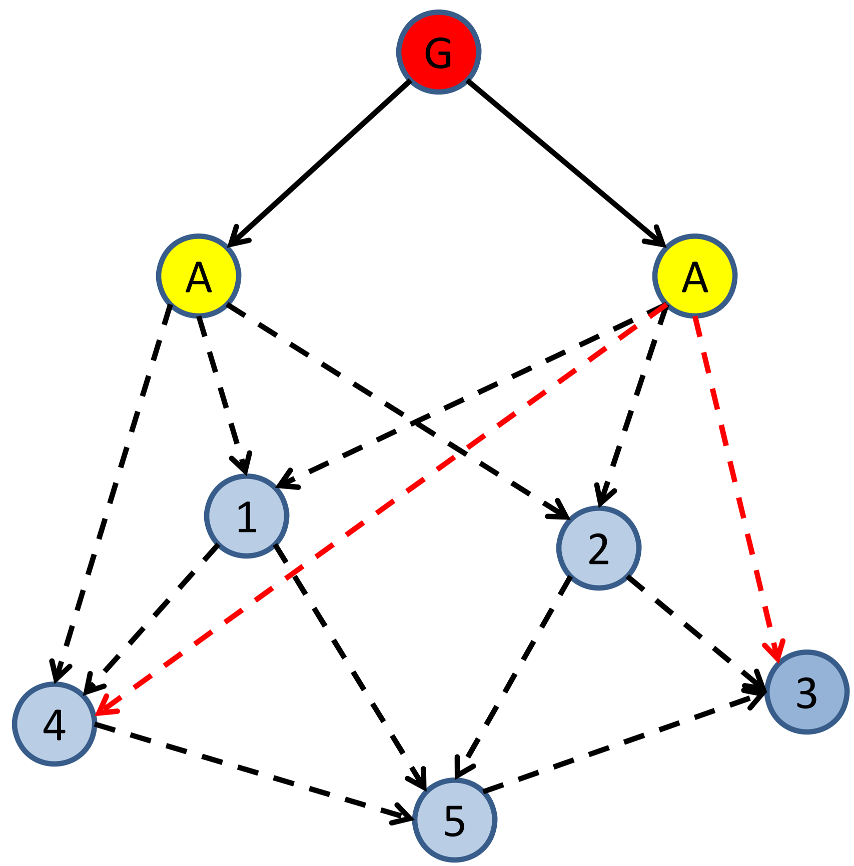
(a) Original network topology

(b) Downlink graph to Dev 3 and Dev 4

Constructing GB

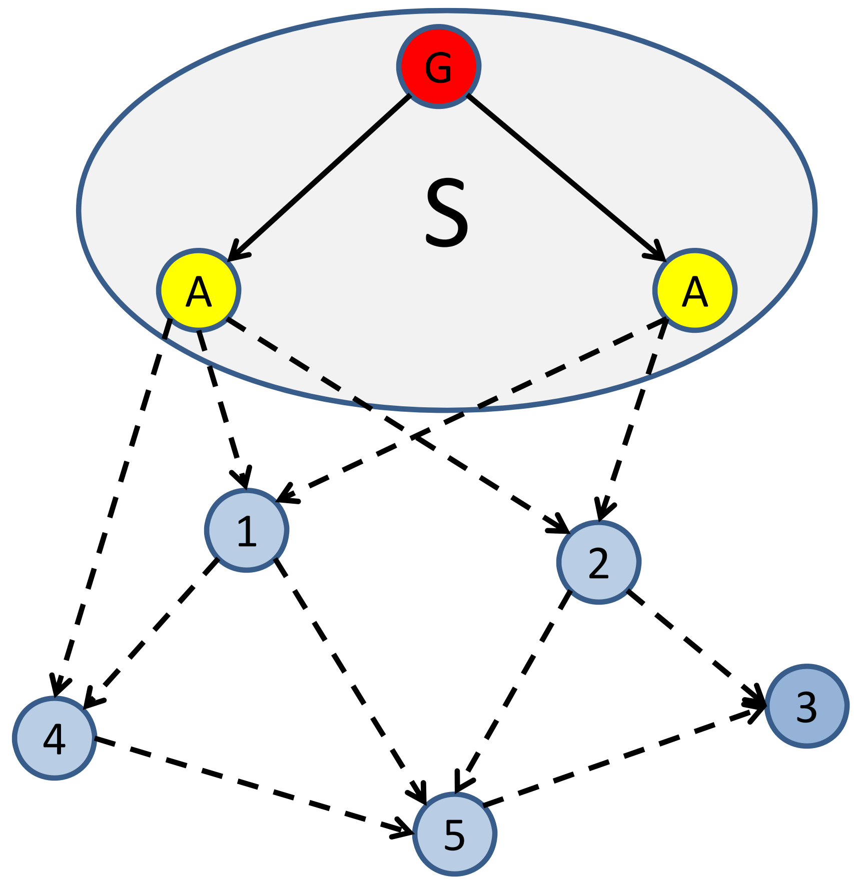
- Drop the links with low Receive Signal Strength (RSS) in the original network topology G





Constructing G_B

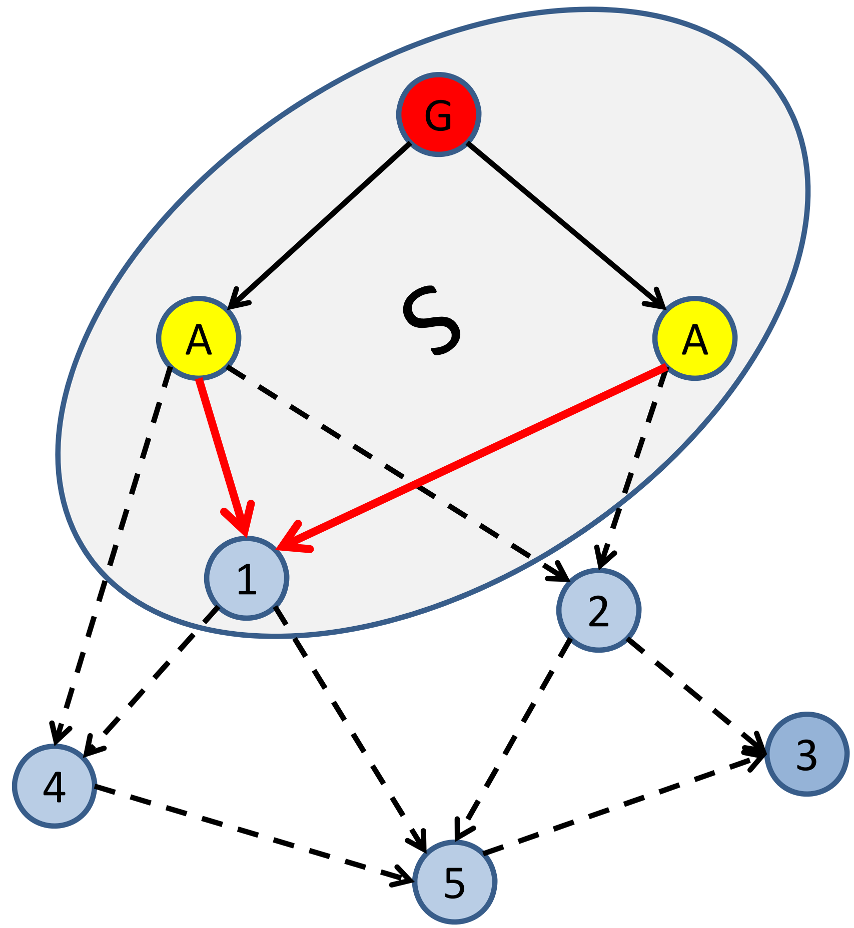
- Drop the links with low RSS in the original network topology G
- Maintain a set of explored node S, initially S = {G, APs}



Constructing GB

- Drop the links with low RSS in the original network topology G
- Maintain a set of explored node S, initially S = {G, APs}
- Grow S according to $\bar{h}_i = \frac{\sum_{k \in P_i} \bar{h}_k}{|P_i|} + 1$

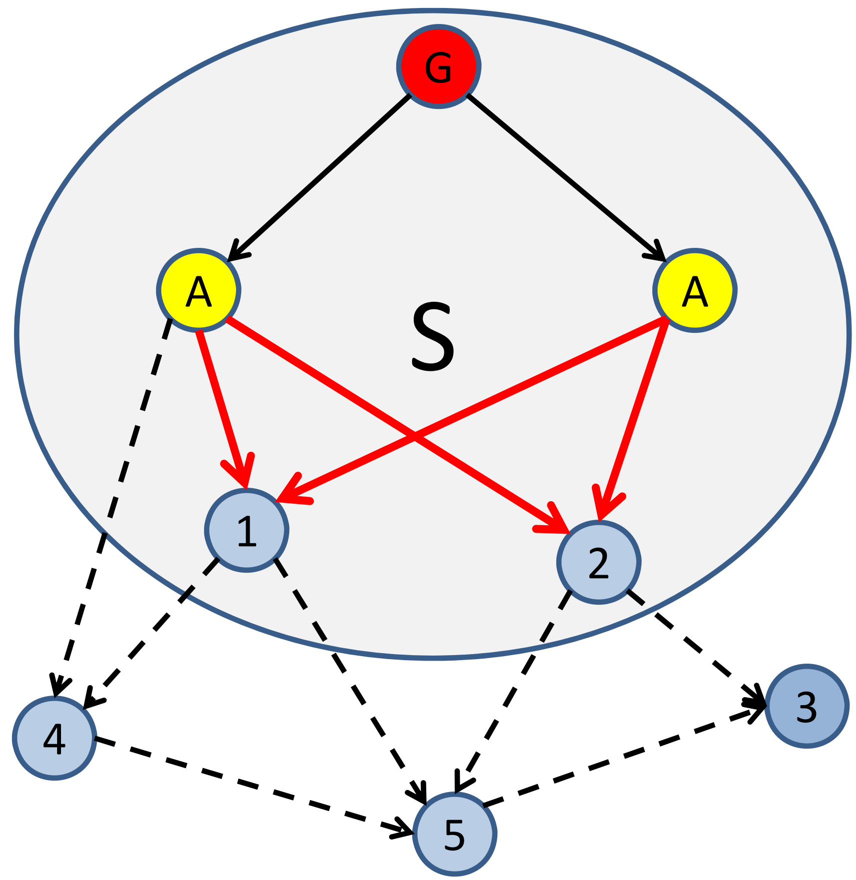
S = {G, Aps, 1}



Constructing G_B

- Drop the links with low RSS in the original network topology G
- Maintain a set of explored node S, initially S = {G, APs}
- Grow S according to $\bar{h}_i = \frac{\sum_{k \in P_i} \bar{h}_k}{|P_i|} + 1$

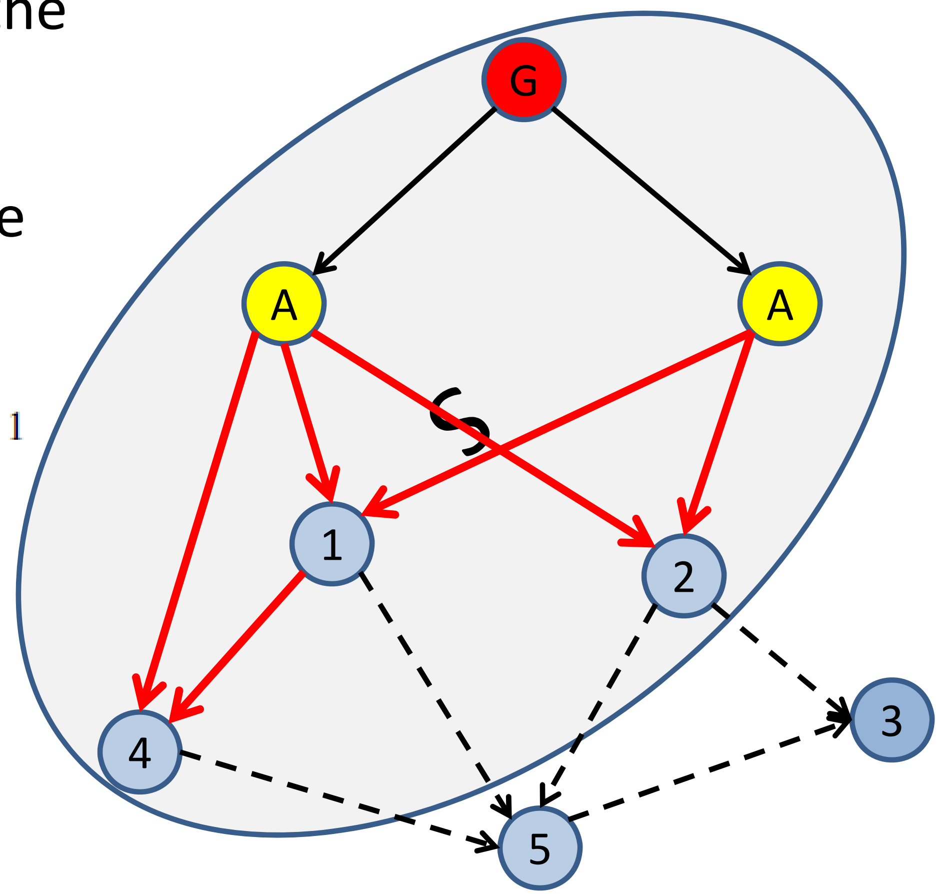
S = {G, Aps, 1, 2}



Constructing GB

- Drop the links with low RSS in the original network topology G
- Maintain a set of explored node S, initially S = {G, APs}
- Grow S according to $\bar{h}_i = \frac{\sum_{k \in P_i} \bar{h}_k}{|P_i|} + 1$

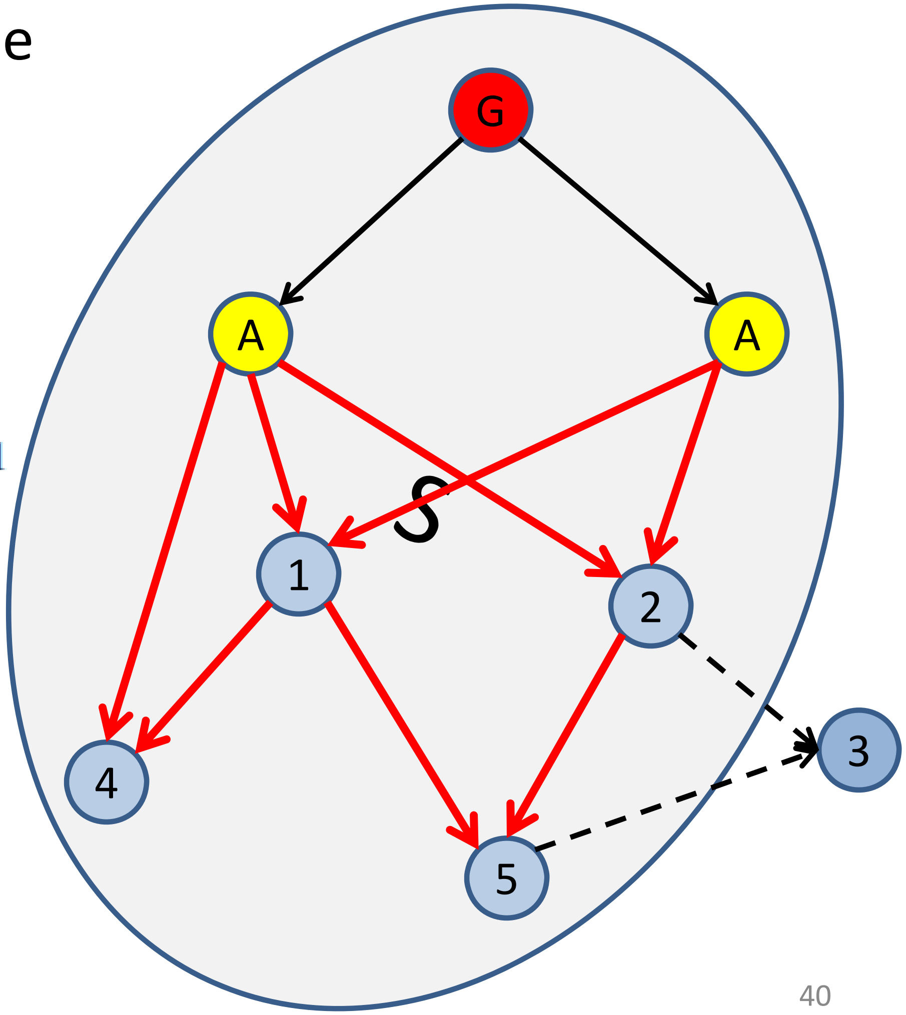
S = {G, Aps, 1, 2, 4}



Constructing GB

- Drop the links with low RSS in the original network topology G
- Maintain a set of explored node S, initially S = {G, APs}
- Grow S according to $\bar{h}_i = \frac{\sum_{k \in P_i} \bar{h}_k}{|P_i|} + 1$

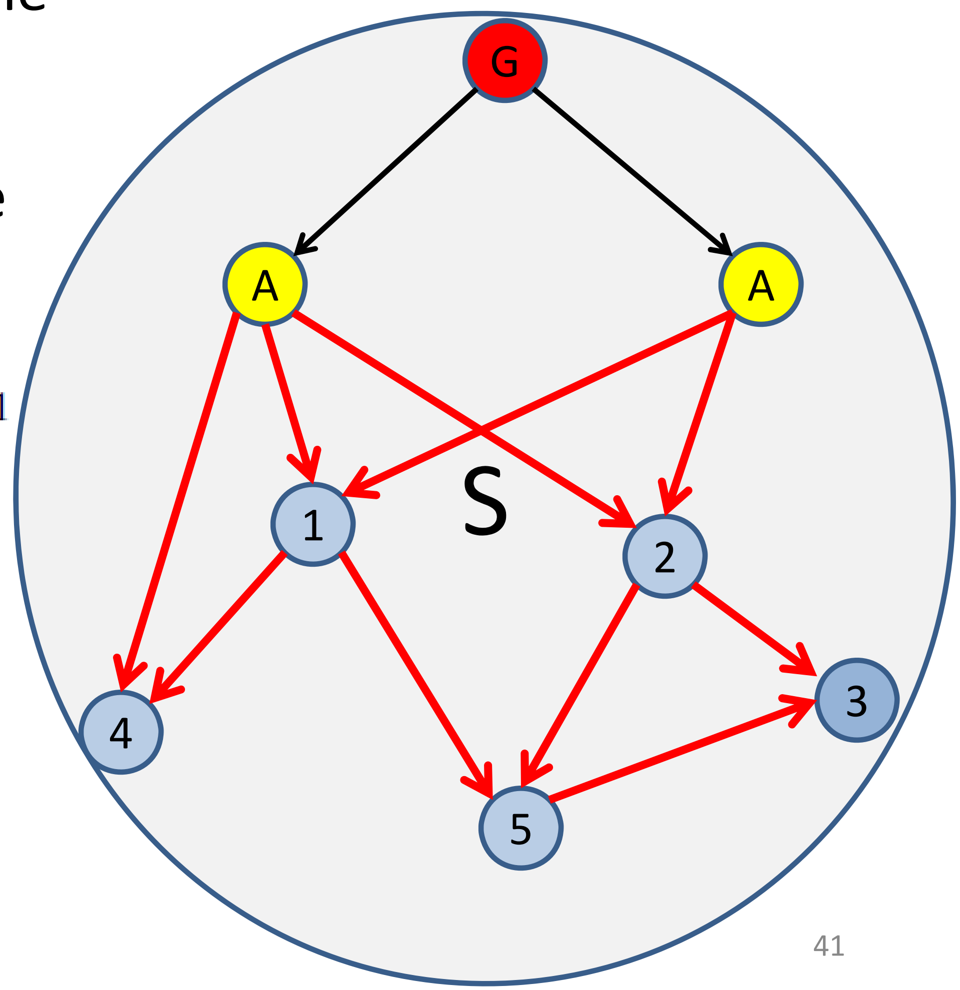
S = {G, Aps, 1, 2, 4, 5}



Constructing G_B

- Drop the links with low RSS in the original network topology G
- Maintain a set of explored node S, initially S = {G, APs}
- Grow S according to $\bar{h}_i = \frac{\sum_{k \in P_i} \bar{h}_k}{|P_i|} + 1$

S = {G, Aps, 1, 2, 4, 5, 3}



Construct G_V

- More complicated than G_B and G_U:
  - Only involves part of the nodes in G
  - The existence of cycle
  - Restrictions: One cycle (length 2) between the parents of destination node v
- Standard Reliable Downlink Graph
  - Construct a completely new graph from GW to DEV v
  - Configuration in intermediate nodes cannot be reused
  - High configuration cost and poor scalability

Sequential Reliable Downlink Routing (SRDR)

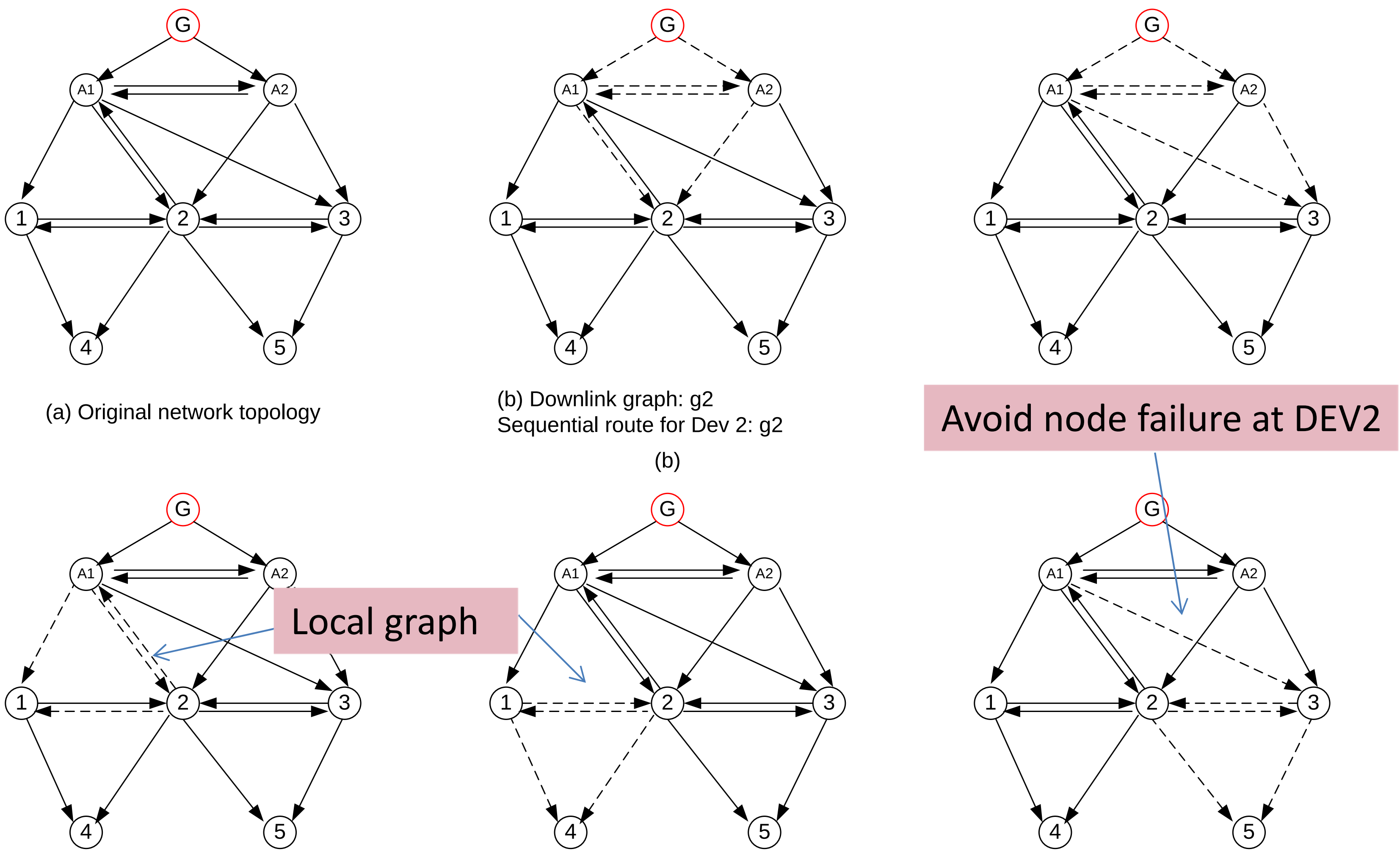
- Key Principles
  - Each node only keep a small local graph
  - Local graphs are reusable building blocks for constructing reliable downlink graph for multiple destinations

Low configuration cost

High Scalability

High Reliability

An example of SRDR



(a) Original network topology

(b) Downlink graph: g2
Sequential route for Dev 2: g2

(b)

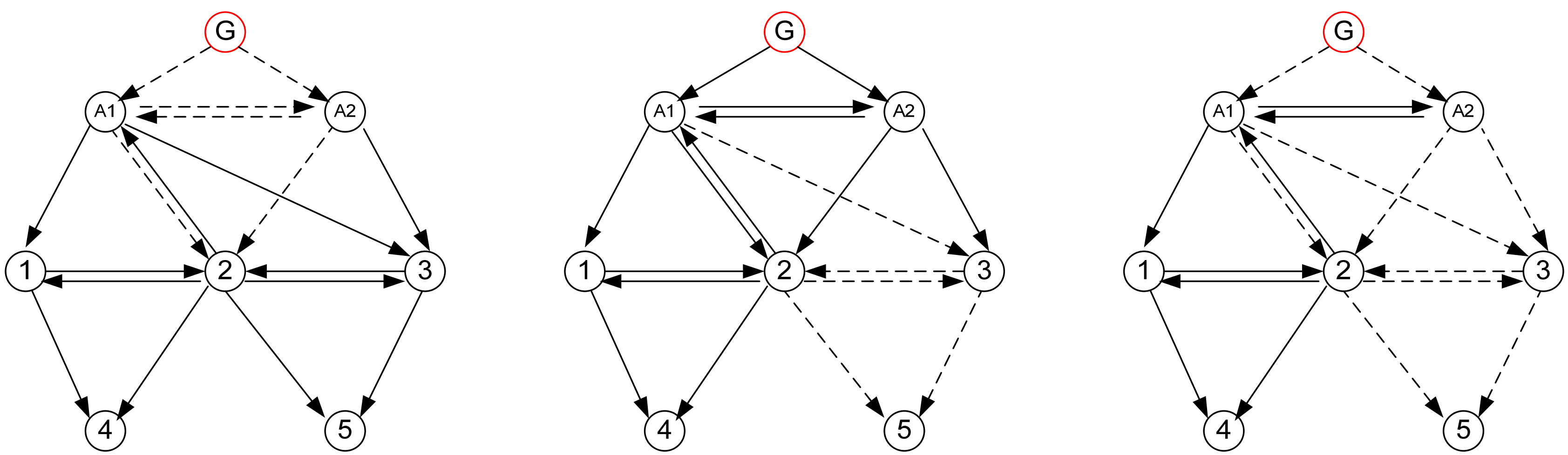
(d) Downlink graph: g1
Sequential route for Dev 1: g2, g1

(e) Downlink graph: g4
Sequential route for Dev 4: g2, g1, g4

(f) Downlink graph: g5
Sequential route for Dev 5: g2, g5

SRDR vs. Standard Downlink Graph

Configure cost is reduced by 3 links



(a) Downlink graph for Dev 2

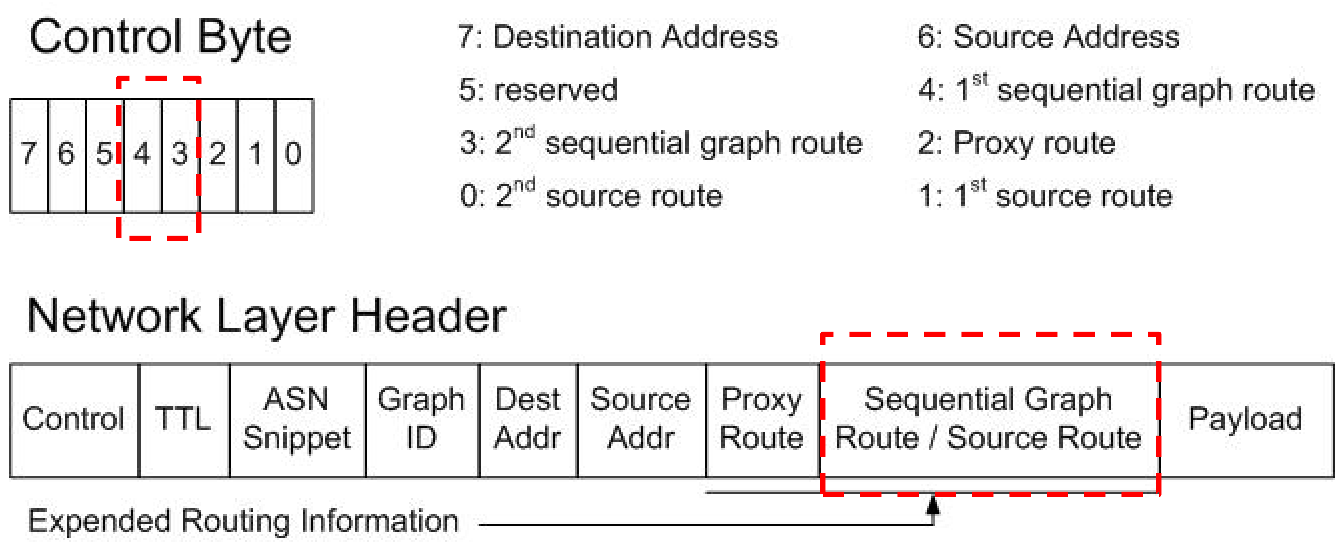
(b) Downlink graph: g5
Sequential route for Dev 5: g2, g5

(c) Standard downlink graph for Dev 5

More significant improvement in large scale networks

Sequential Reliable Downlink Routing (SRDR) Extensions

- Network layer header extension:



SRDR Extensions

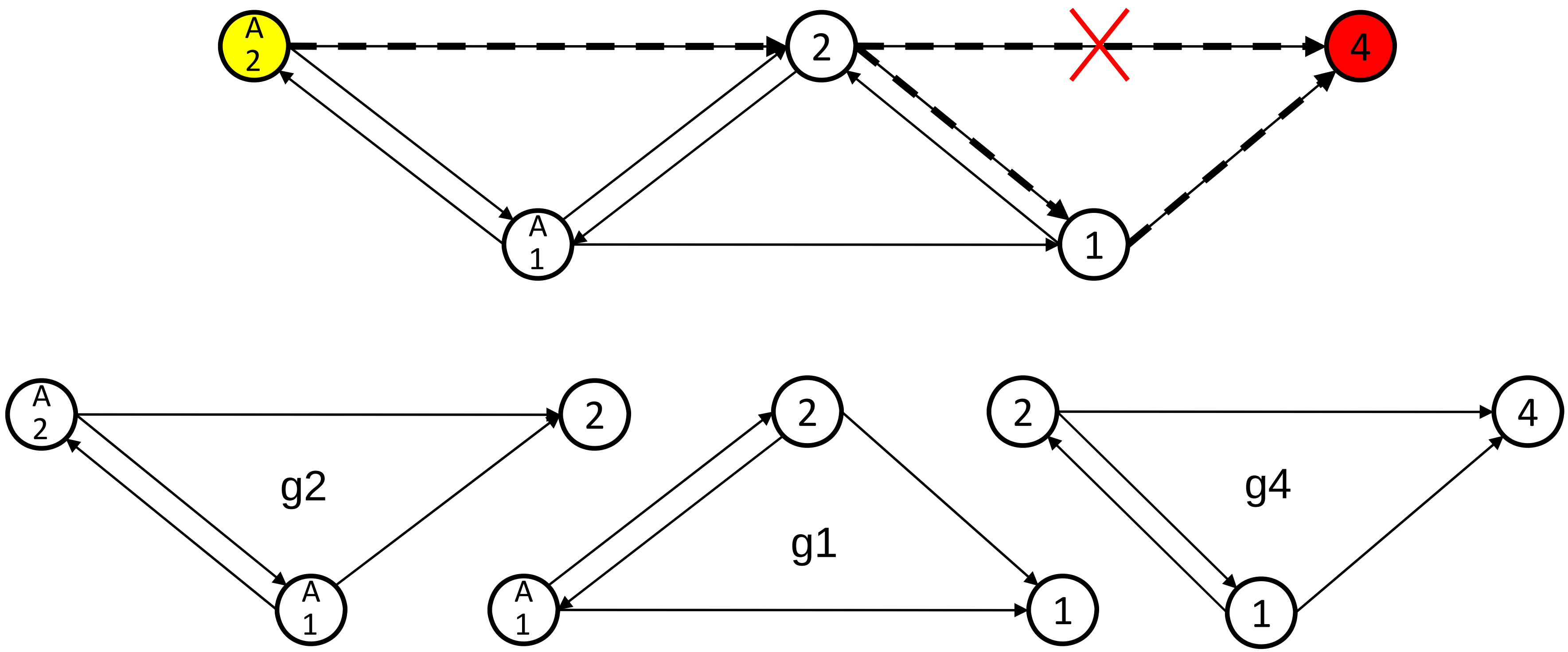
- Routing module extension:
  - retrieve the earliest graph ID in the graph list and route the packet on this graph
  - If current node is the sink of the graph, remove this graph ID and route the packet on the next earliest graph.
  - If routing is failed, remove this graph ID and try the next earliest graph ID if it has the corresponding edges.

Optimization on SRDR

- In SRDR, routing is performed strictly according to the sequence in the ordered graph list.

- SRDR-OPT
  - Observation: each node can keep graph info to multiple destination.
  - Have chance to take the "shortcut"
  - Principle: Search the ordered graph list *backward* and route the packet on the first graph ID that is stored in its table

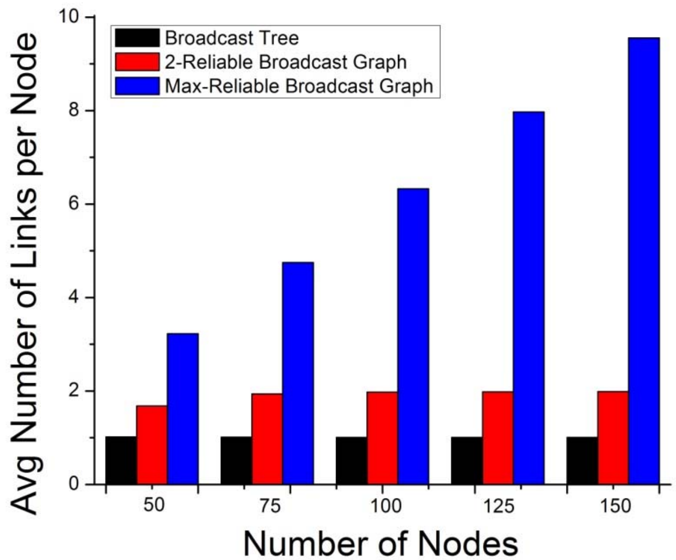
SRDR Optimization

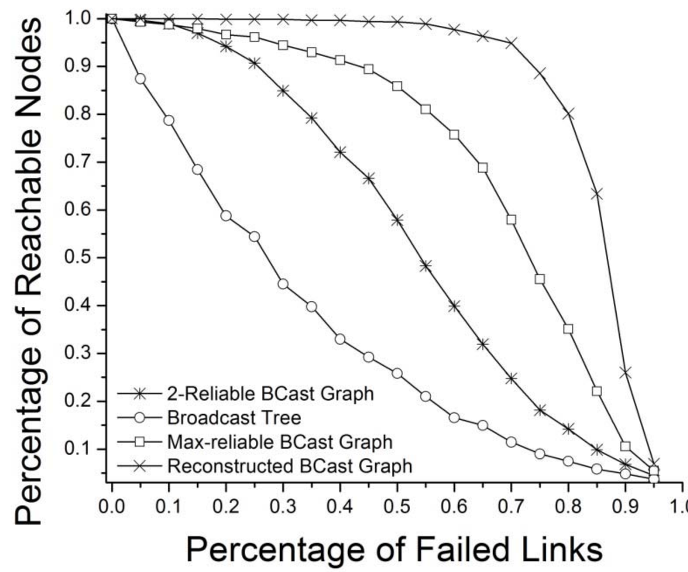


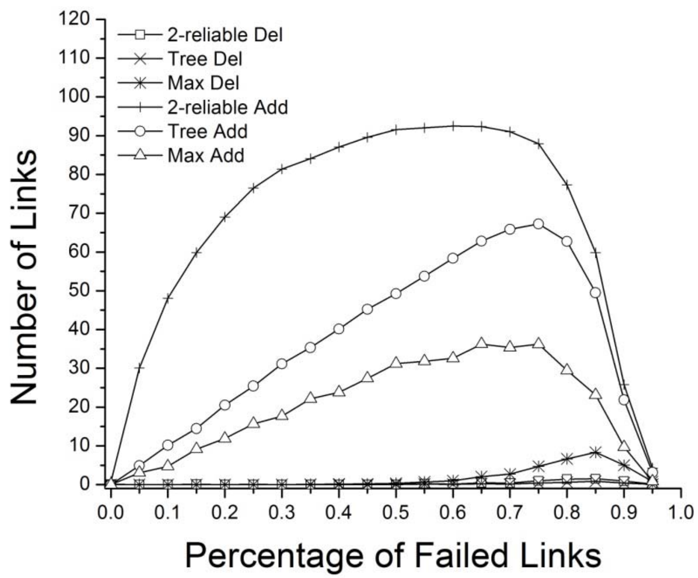
Control	TTL	ASN Snippet	Graph ID	Dest Addr	Source Addr	Proxy Route	g2	g1	g4		Payload

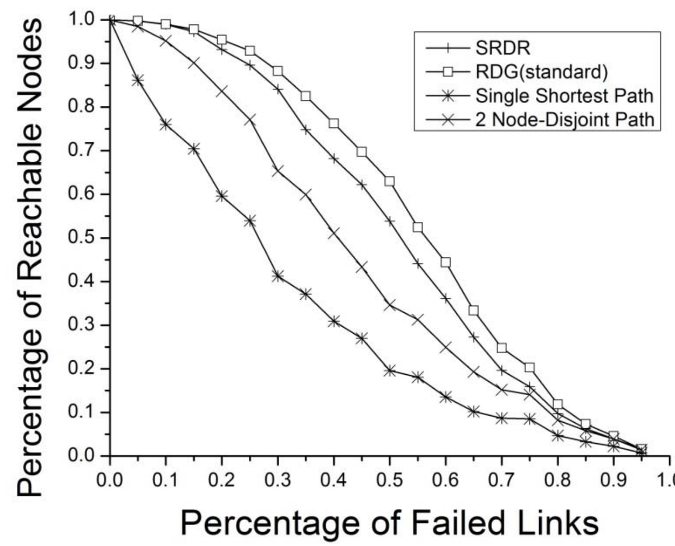
Extended Routing Information

Performance Evaluation

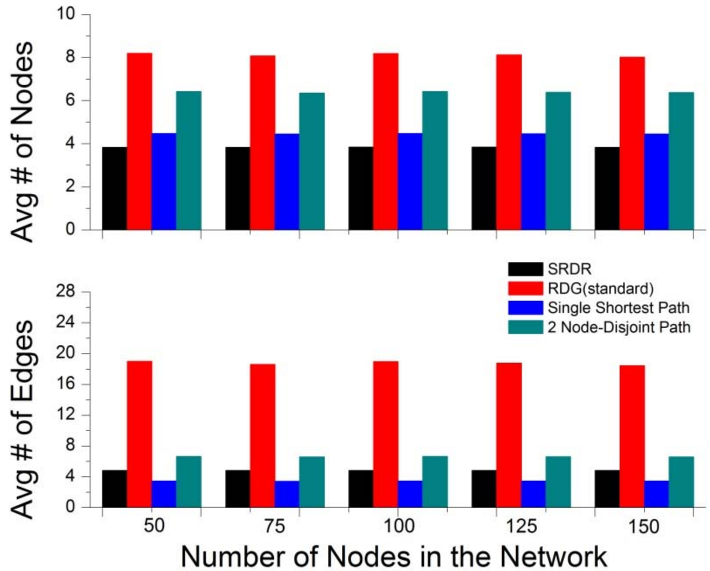


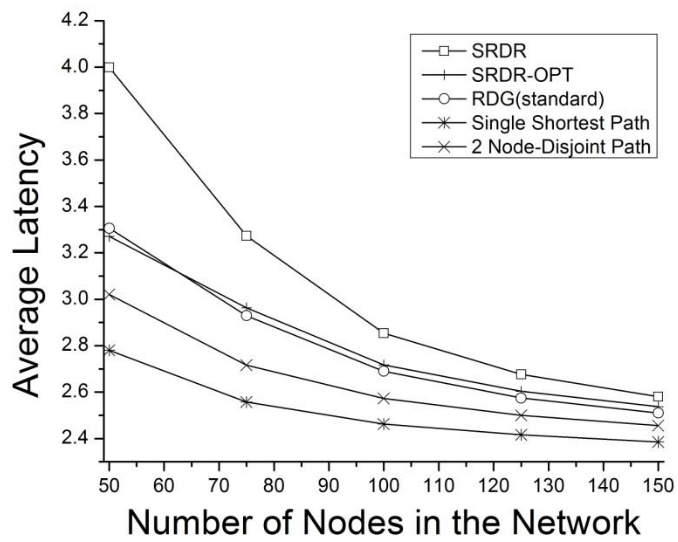


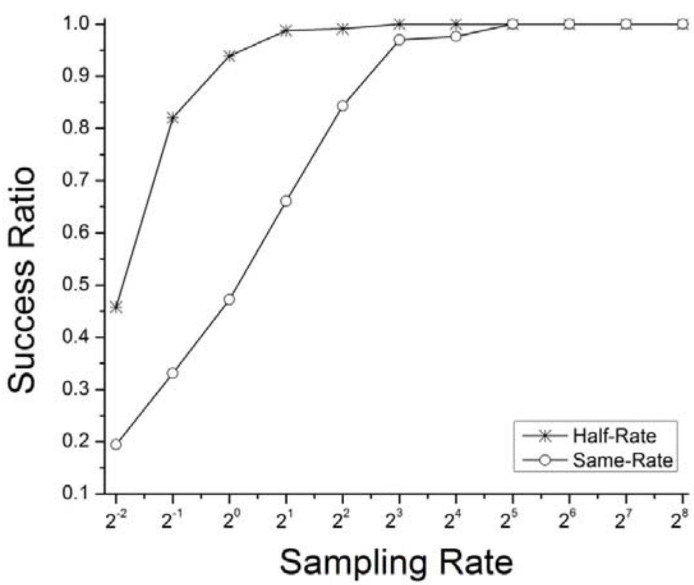


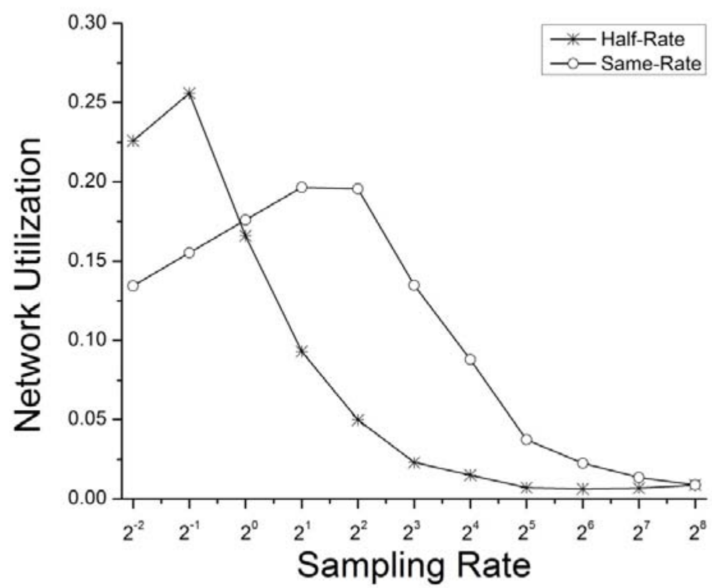


Performance Evaluation









System Design, Implementation and Deployment

WirelessHART Prototype System

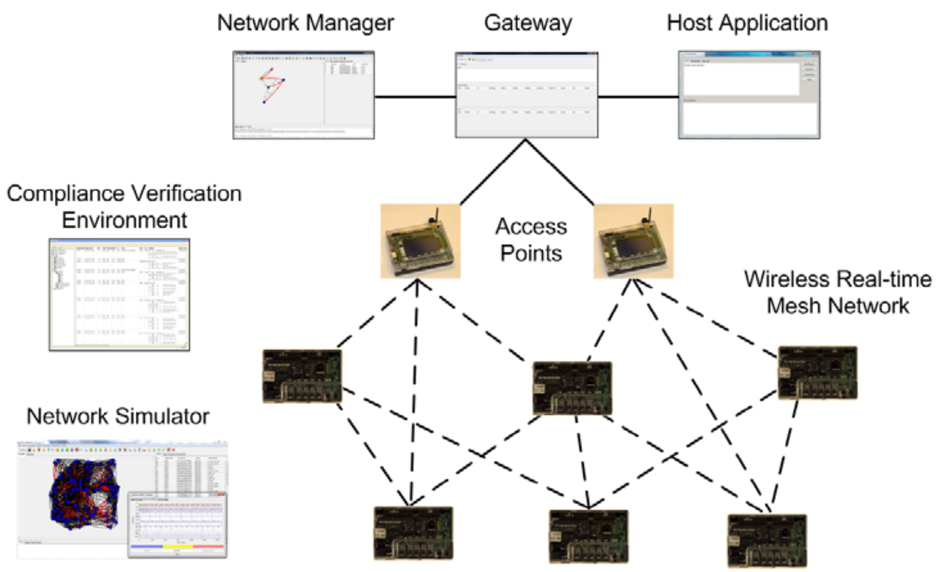
Major Components in the prototype :

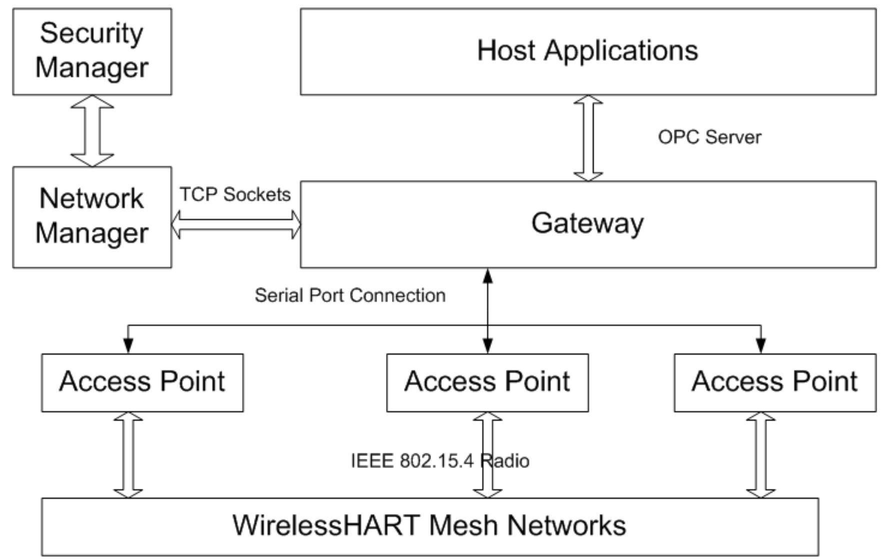
- Network Manager
- Gateway
- Host Application
- Access Point
- Device
- Sniffer

PC Side (Network Manager, Gateway, Host Application)

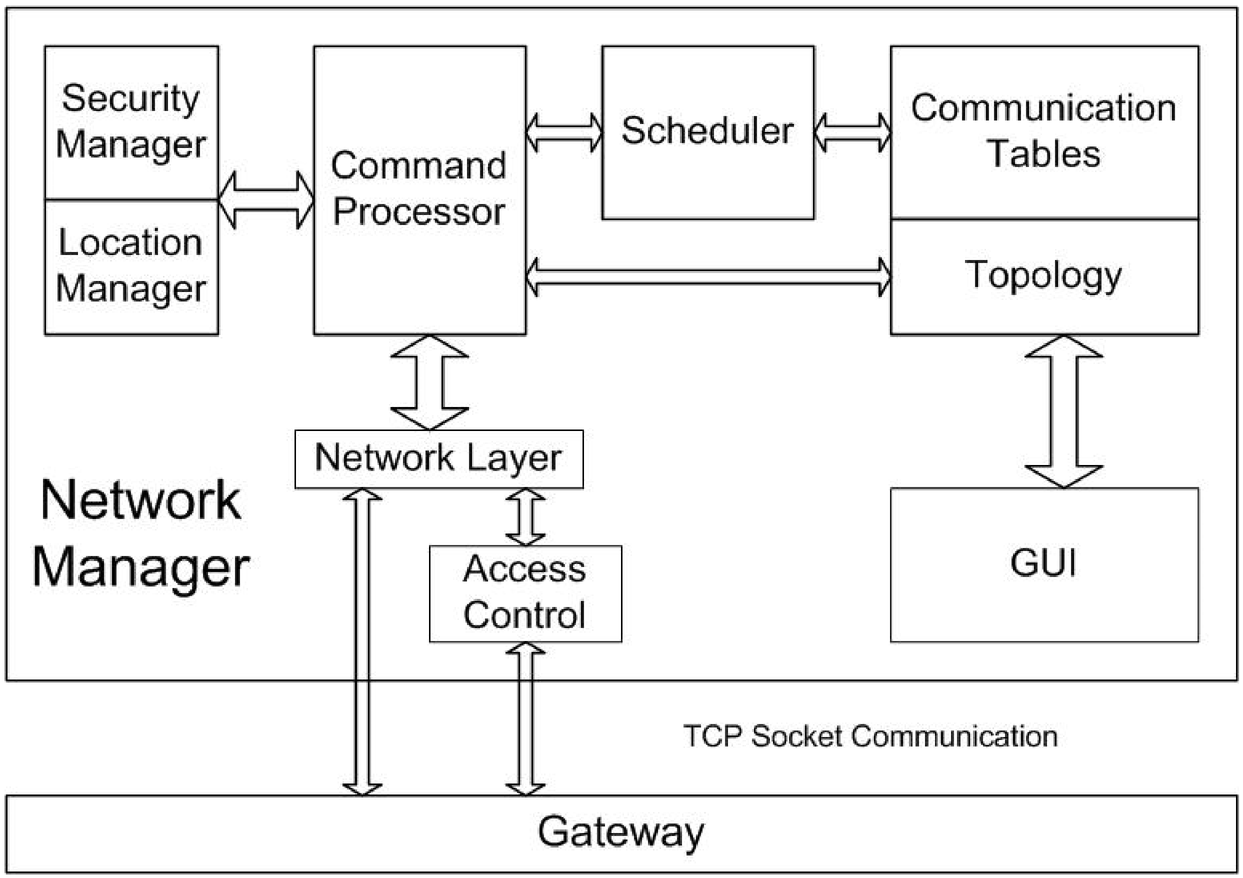
Embedded Side (Access Point, Device, Sniffer)

System Design, Implementation and Deployment





Design of the Network Manager



Functionalities in Network Manager

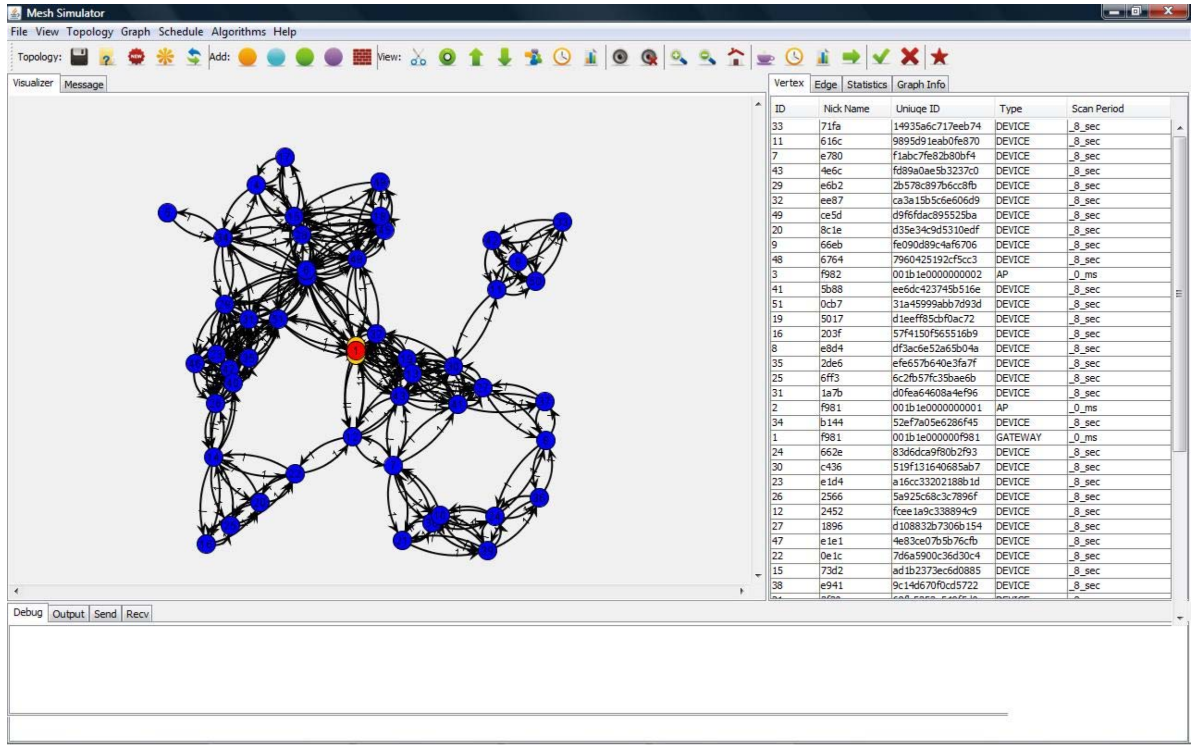
- Location Manager
- Security Manager and Access Control
- Maintaining Reliable Routing Graphs
- Maintaining Communication Schedule
- Friendly GUI

Network Manager Functionalities

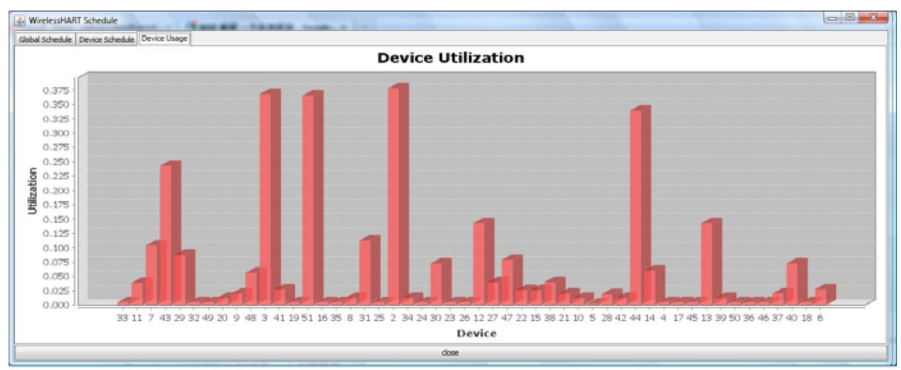
Network Topology

Routing Graphs

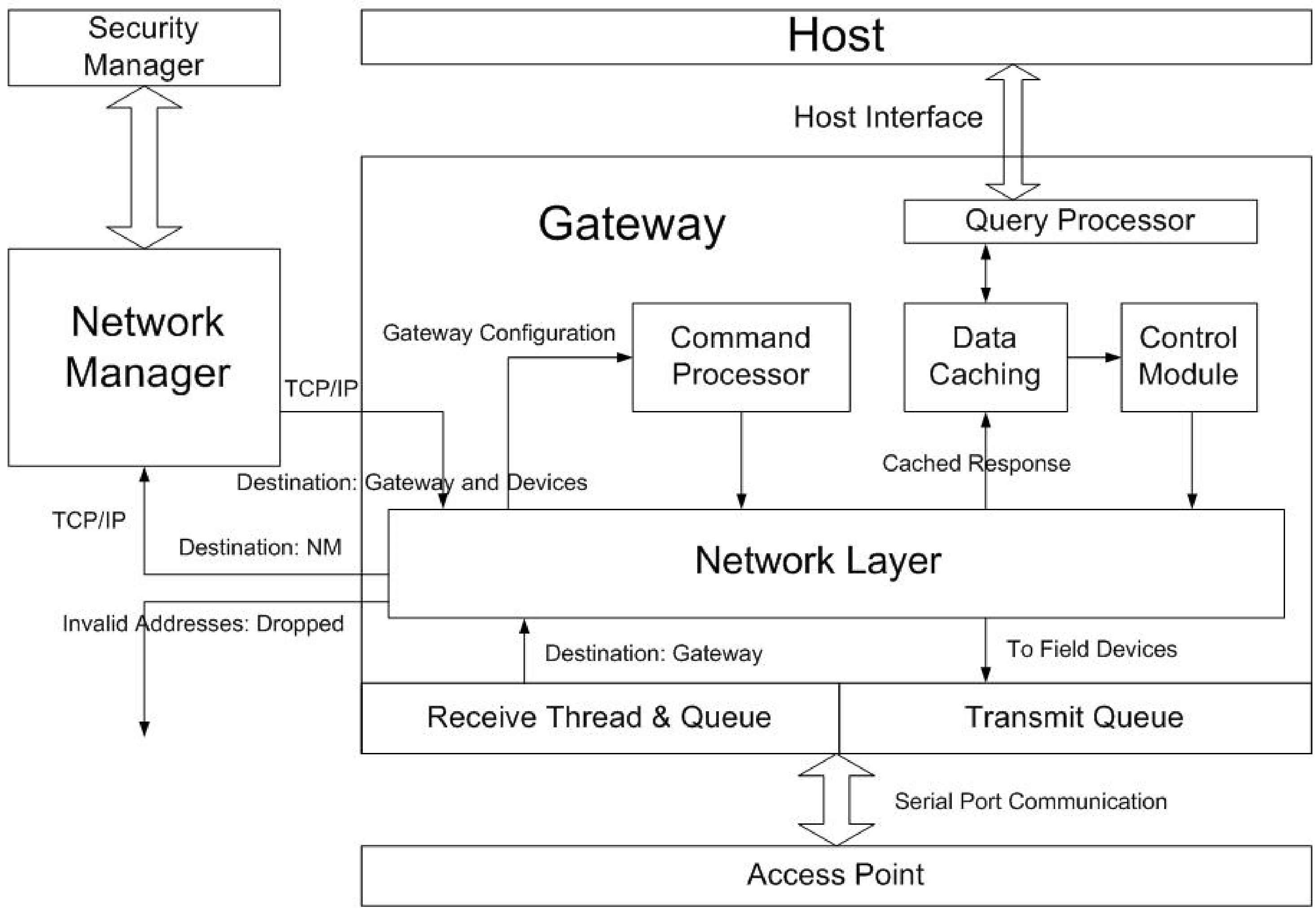
Device Configuration



Device Bandwidth



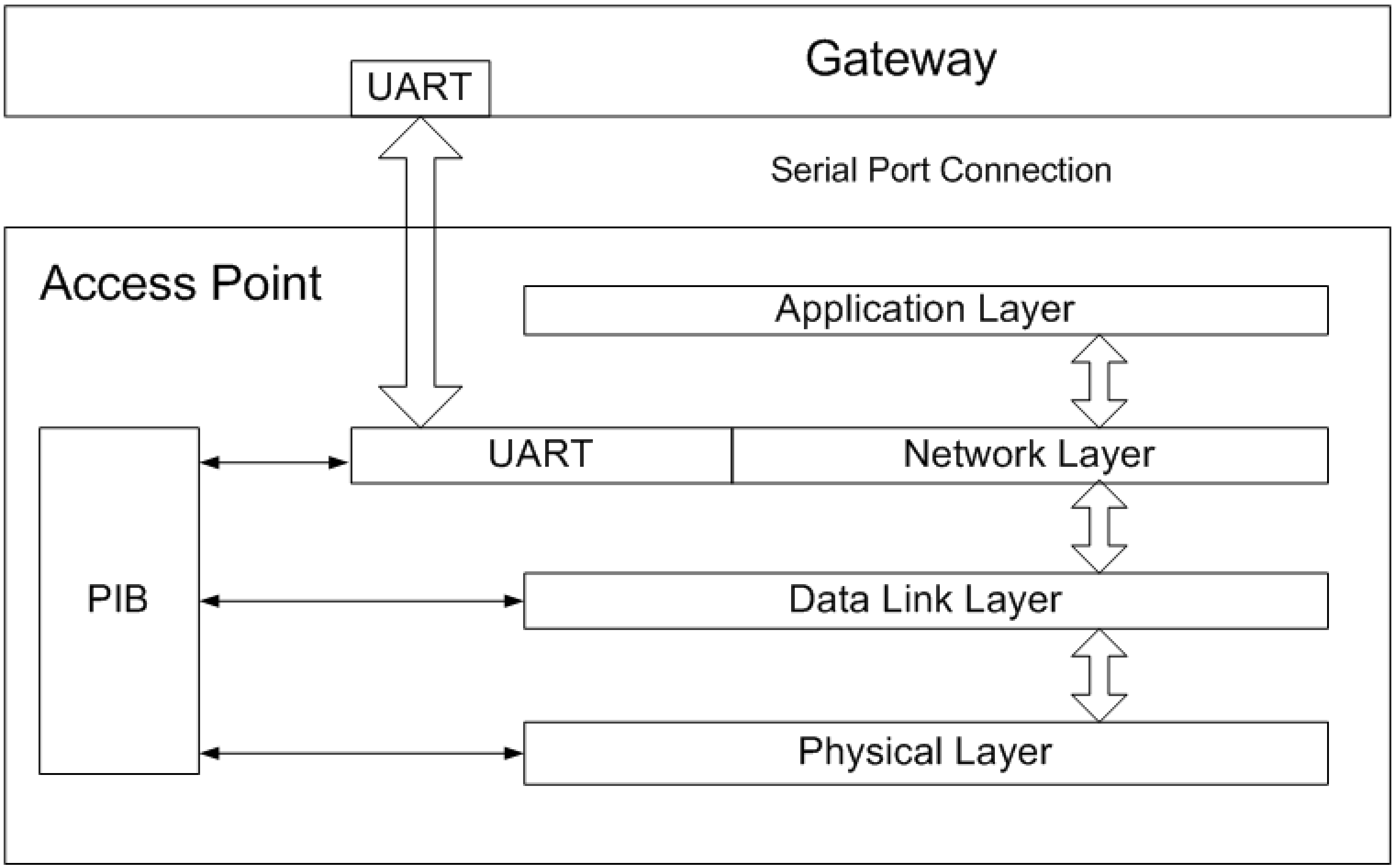
Design of the Gateway



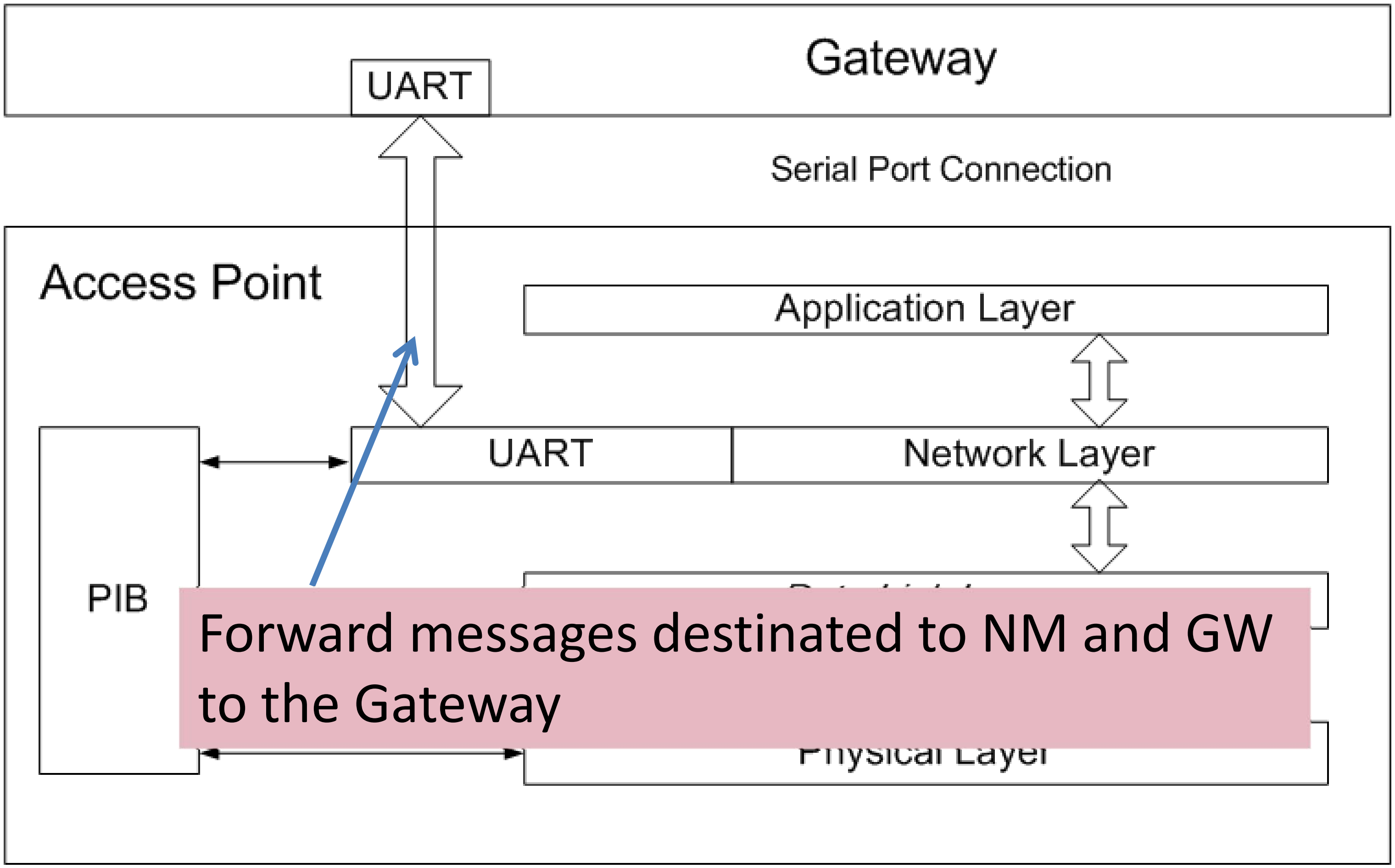
Functionalities in Gateway

- Processing Queries from Host Applications
- Data Caching for devices
- Multiple Access Points support
- Communication between GW and NM, AP and Host Applications
- Control in the Gateway

Design of the Access Point



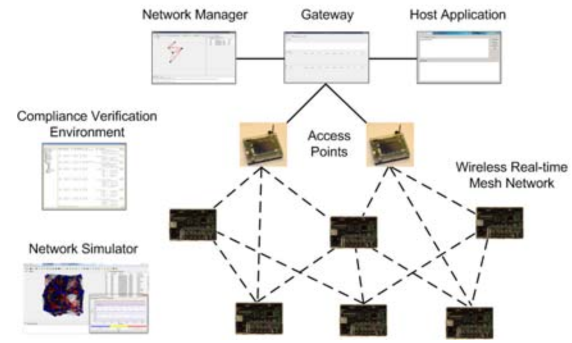
Design of the Access Point

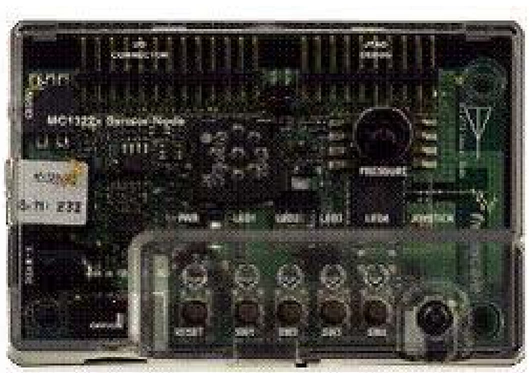


Forward messages destinated to NM and GW to the Gateway

System Design, Implementation and Deployment

Hardware Platforms





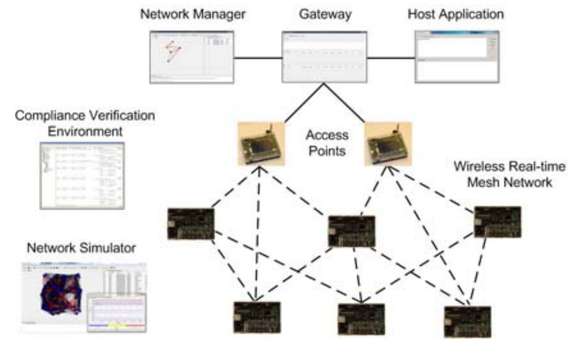
Freescale 1322x SRB Evaluation Board

Custom Designed Mother Board with Sensor Support

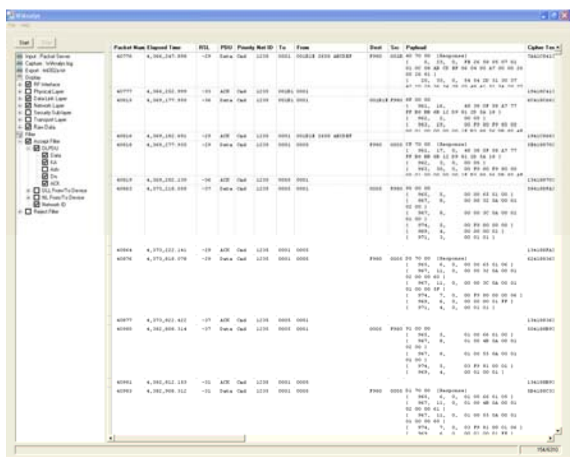
Custom Designed Board with EnergyMicro EFM32 MCU

System Design, Implementation and Deployment (Cont.)

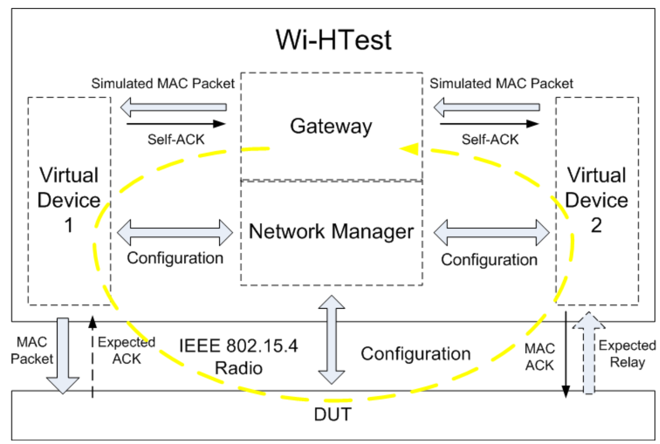
Compliance Testing Suite



Testing Engine

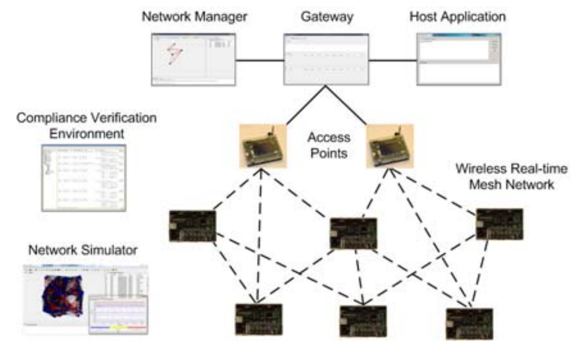


16-Channel Sniffer

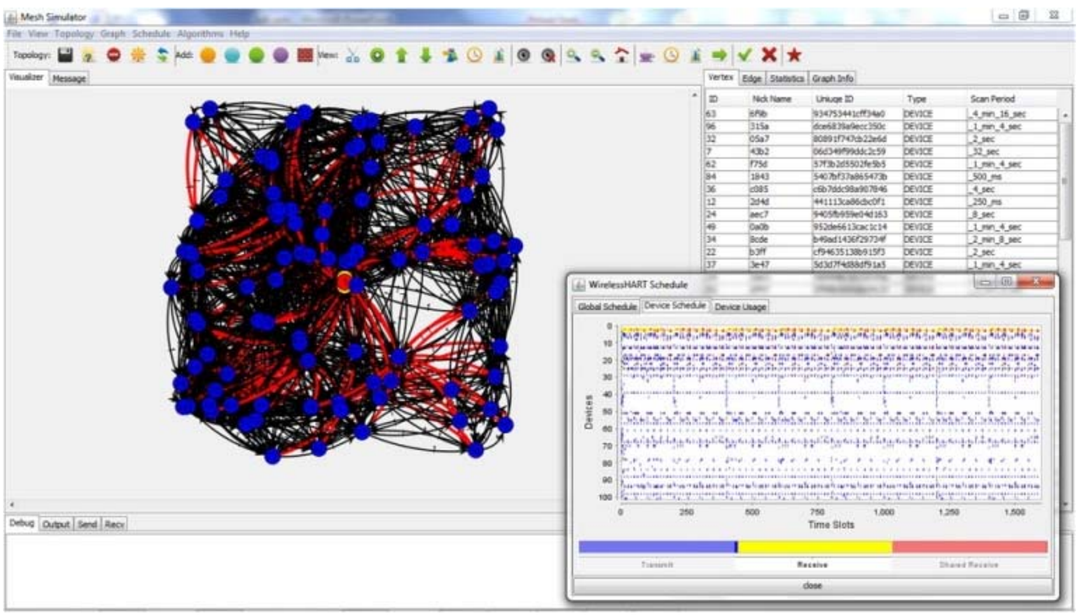


Virtual Network Approach

System Design, Implementation and Deployment (Cont.)



Network Manager and Simulator

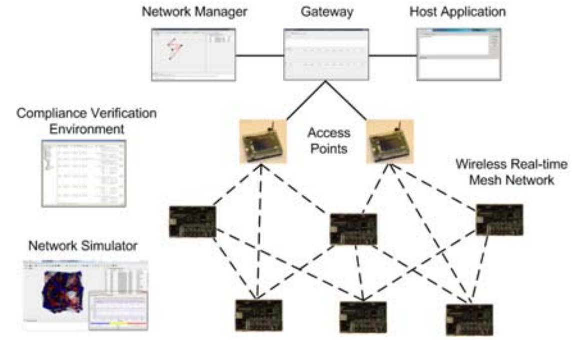


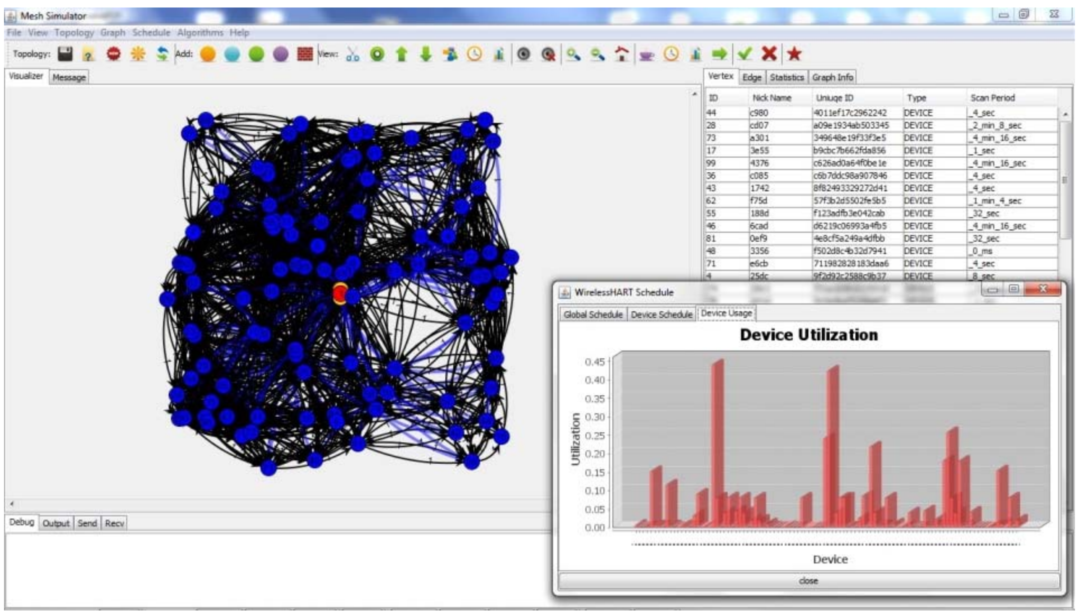
Simulating a real-time wireless network with 100 devices:

- reliable broadcast graph
- device communication schedule

System Design, Implementation and Deployment (Cont.)

Network Manager and Simulator

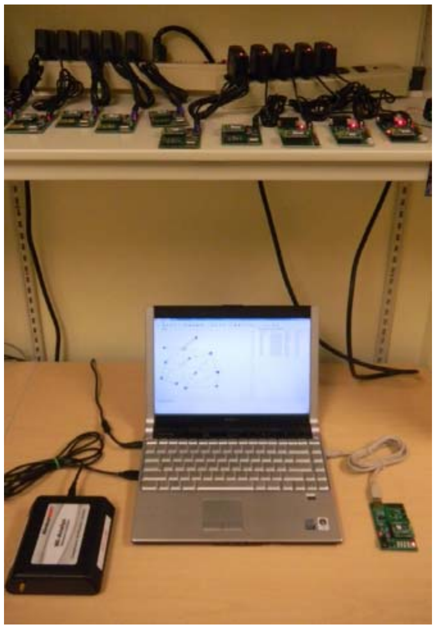




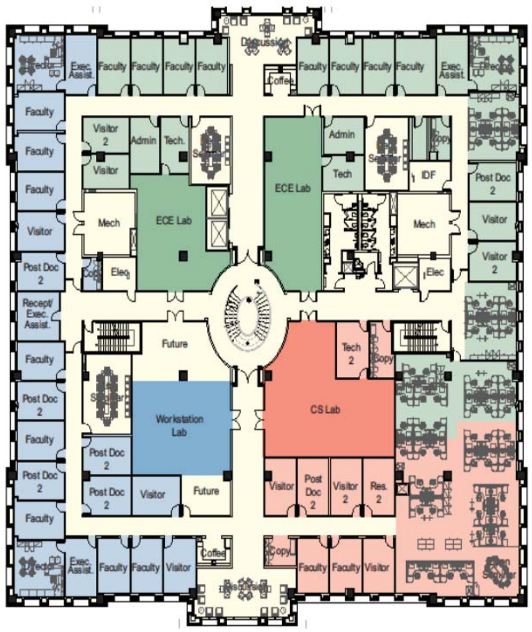
Simulating a real-time wireless network with 100 devices:

- reliable uplink graph
- device bandwidth utilization

System Design, Implementation and Deployment (Cont.)



10 Device Testbed



UT Austin ACES 5th floor

UT Pickle Research Center

UWO
Power House