Wireless Communication

Dr. Gianluca Franchino

Scuola Superiore Sant’Anna
Pisa, Italy

Email: g.franchino@sssup.it

RETIS LAB

Real-Time Systems Laboratory

Retis Real-Time Systems Laboratory

Scuola Superiore Sant'Anna Pisa

Graduate Course on Embedded Control Systems: Theory and Practice

Wireless Communication in Real-Time Embedded Systems

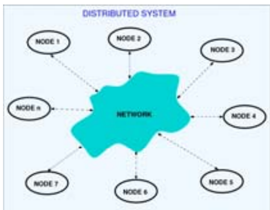
Gianluca Franchino

g.franchino@sssup.it

Outline

- **Introduction to:**
  - Distributed Embedded Systems
  - QoS and Real-Time Communication
- **Wireless Communication:**
  - MAC protocol techniques: CSMA/CA, TDMA (scheduling approach) and IEEE 802.15.4/ZigBee
  - Real-time MAC protocols
- **Capacity of Wireless Networks:**
  - Capacity of adhoc networks
  - Real-Time Capacity of Fixed-Priority schemes
  - Capacity of Implicit EDF

Distributed System

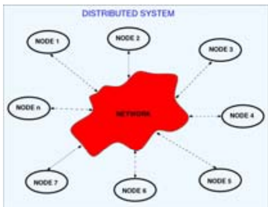


- Distributed System: is an application that executes a collection of protocols to coordinate the actions of multiple processes on a network, such that all components cooperate together to perform a single or small set of related tasks.

Distributed Systems

- **Why is a distributed architecture desirable?**
  - **Composability:** the system is built by composing/integrating sub-systems
  - **Scalability:** a new system function can be obtained adding a new node. A system function can be replicated in the same way
  - **Information Processing close to data sources/sinks:** in-node data elaboration: intelligent sensors and actuators.
  - **Dependability:**
    - **Robustness:** a node failure does not jeopardize the system operation (no single point of failure);
    - **Maintainability:** tanks to system modularity, a node can be replaced easily

Distributed Systems



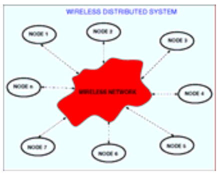
- The **network** is a fundamental part of a **Distributed System**
- In general, a **node failure does not compromise** system services
- In general, **loss of network operation jeopardizes** system services
- **THE NETWORK IS THE KERNEL OF A DISTRIBUTED SYSTEM**

Applications

- **Wireless Sensor/Actuator Networks**
  - Home automation (domotic systems)
  - Wearable WSN for health-care
  - Environment monitoring
- **Multi-robot systems**
- **Control Systems for Cars, Aircrafts, Trains** etc.
- **Factory Automation**
- Etc

Wireless Distributed Systems

- Wireless Distributed System: Distributed System where the network is composed by wireless nodes
- With respect to a wired channel, the management of a wireless channel is more difficult
- A wireless channel is characterized by:
  - High bit error rate → e.g > 10^{-3}
  - Asymmetric links: NODE1 - - - - NODE2
  - Variable Channel Capacity (Bandwidth) , both over time and node by node



Communication Stack (ISO/OSI Model)

- Network systems designed by a modular methodology-> layered stack
- Each layer is delegated to specific functionalities
- Each layer implements:
  - Protocols to manage the communication with the corresponding layer in other nodes
  - Services provided to adjacent layer through service interfaces

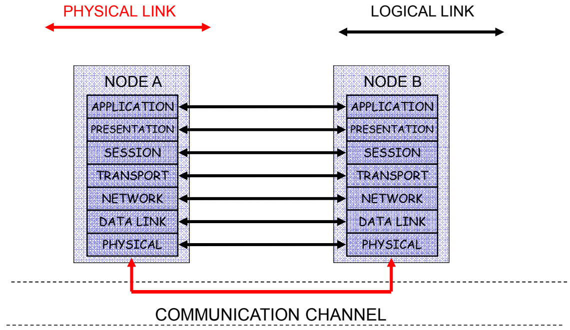
NODE

NODE
APPLICATION
PRESENTATION
SESSION
TRANSPORT
NETWORK
DATA LINK
PHYSICAL

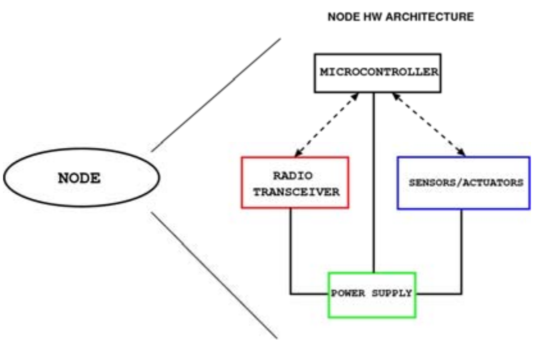
Communication Stack

NODE	
APPLICATION	Services (Protocols) for Applications
PRESENTATION	Data semantics
SESSION	Remote Actions
TRANSPORT	End-to-End Communication
NETWORK	Routing and Logical Addressing
DATA LINK	LLC, MAC, Physical addressing
PHYSICAL	Bit Encoding, Modulation, Physical Channel, Transceiver Control

Communication Stack



Node Architecture (Embedded Systems)



Real-Time Communication

- **Real-time Communication:**
  - Efficient communication of Short Data: Sensor Data (few bytes)
  - Periodic transmission with low Jitter: Control, Sensor and Monitoring Data: Time Triggered Transmission
  - Fast transmission of Event Data (Asynchronous Data, e.g. alarms): Event Triggered Transmission
- **Mixed traffic Communication:**
  - Coexistence of best-effort traffic (non real-time traffic as log data, configuration data …) and Real-Time traffic

Real-Time Communication

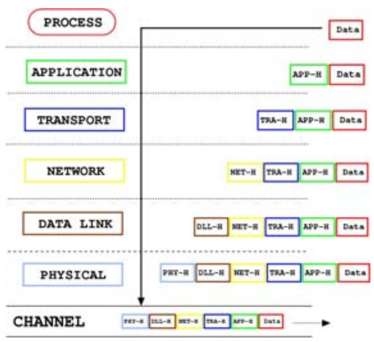
- **End-to-End** communication **delay** must be **bounded**
- Each layer introduces **computational** and **communication overheads** (header bytes)
- All **layer services** must be **time-bounded**
- Thus **not all stack layers** are **implemented**:
  - **Short Messages**: message fragmentation/reassembly is not needed (**no Transport Layer**)
  - When there is only a **single-hop domain**, the **network layer** is not implemented (**no routing**)
  - **Application Layer** interfaces the **Data Link Layer** directly (when there is no need of **Network layer**)

Communication Efficiency

- *CEff*: **Communication Efficiency**
- *Data_length* (**payload**): is the **length** (time units) of **data** generated by the application running into the node
- *Comm_length*: is the **time length** of the **message transaction** (**end-to-end delay**). It comprises **layer services overhead** plus **transmission overhead** due to the **control characters** (**packet headers**)

$$CEff = \frac{Data\_length}{Comm\_length}$$

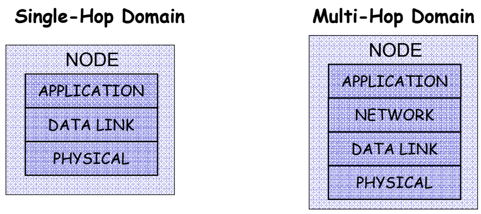
Communication Overhead



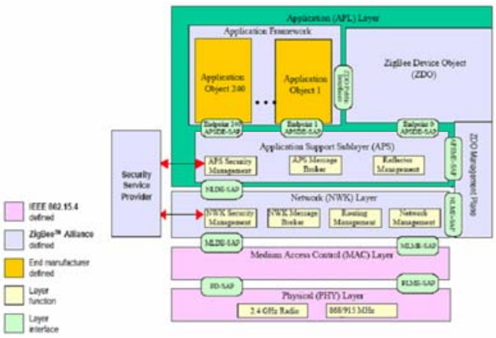
Collapsed Model

•Ex. **Single-Hop Domain**: Factory Automation (**Field Bus**).

•Ex. **Multi-Hop Domain**: Wireless Sensor Networks.



ZigBee Stack



QoS definition

- **QoS** requirements are **application dependent**
- **Main QoS metrics:**
  - **Throughput**: (AvailBand - OverheadBand)/ AvailBand
  - **Maximum Delay**: time-bounded transmission (**real-time**)
  - **Jitter**: variability on message transmission/receiving time
  - **Reliability**: Integrity of messages. Guarantee that all messages will be delivered correctly
- **Other Performance Metrics:**
  - **Energy Dissipation**: Energy wasted should be limited, either to achieve a predefined **system lifetime** or to **maximize** the **system lifetime**
  - **Fairness**: assignment of **network resources** in a **balanced** fashion among the nodes
  - **Stability**: the network is a dynamic system. The protocols performance should be **stable under any working condition**
  - **Robustness**: normal **network** (protocols) **operation** should be **guaranteed** even under some **control packet losses** or **node failure** (e.g. coordinator node failure)

Examples on QoS requirements

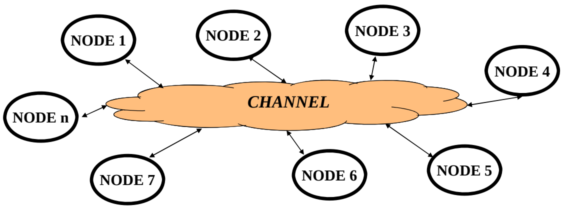
- **Multimedia streaming**:
  - High packet delivery ratio
  - Low Delay
  - Low Jitter
- **Control Applications**:
  - Low Jitter
  - Periodic Message Delivery
- **Distributed Information Systems (data base)**:
  - Integrity of exchanged messages -> the system should guarantee the integrity of retrieved data (data without errors)

Data Link Layer

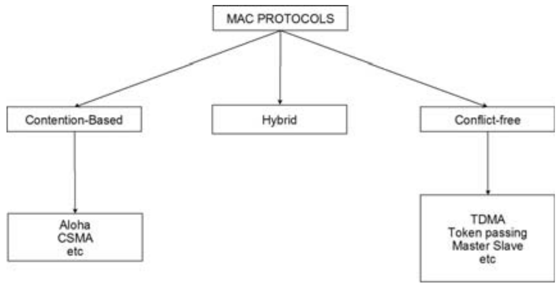
- **Data Link Layer** (It is of paramount importance for Real-time Communication)
  - Logic Link Control (LLC)
  - Medium Access Mechanism (MAC)
- **LLC**:
  - Formation and maintenance of links between one-hop neighbors nodes
    - Link discovery, setup, maintenance, link quality estimation
  - Reliable and Efficient information (packets) transmission over the established links
    - Addressing and Flow Control
    - Admission Control
    - Error Control:
      - ACKnowledgement (ACK)
      - Automatic Repeat reQuest (ARQ)
      - Forward Error Correction (FEC) (Preferred for Real-Time Comm.)
- **MAC**:
  - Management of medium (channel/link) access

MAC protocols

- The task of a MAC protocol is to manage the channel access by network nodes.
- From the standpoint of a MAC sub-layer, the network is composed by n nodes sharing a common channel (medium).
- It determines the order of the channel access by contending nodes. Hence it determines the network access delay
- The MAC protocol is fundamental for the real-time performance of a network that shares a common medium



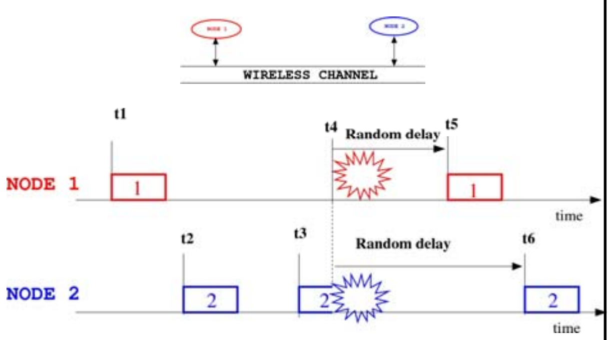
Wireless MAC protocols



Aloha

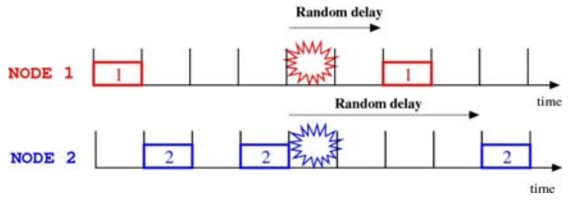
- Designed by Norman Abramson at University of Hawaii in 1970s
- The base algorithm is simple:
  - Whenever you have a packet to transmit, send the packet
  - If the packet collides with an other transmission, wait for a random interval and then try to send the packet
- It is assumed that a node can be aware of a collision either by listening to the channel while transmitting, or by some feedback mechanism (e.g. ACK)

Aloha



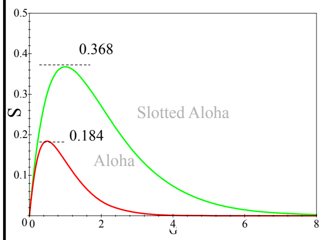
Slotted-Aloha

- Slotted-Aloha improves the basic version
  - The time is divided in time-slots
  - A node can try to send a packet only at the beginning of each slot. A node cannot try to send a packet in the middle of a slot
  - The number of collisions is reduced
  - Nodes must be synchronized.



Aloha

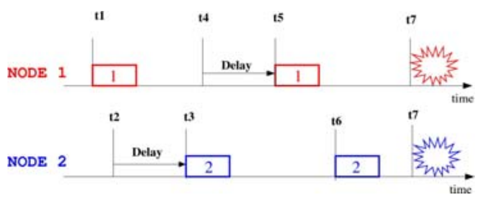
- Maximum Throughput of Aloha: $S_{\max} = 0.5e^{-1} = 0.184$
- Maximum Throughput of Slotted-Aloha: $S_{\max} = e^{-1} = 0.368$



- G = Offered Load.
- G is the average number of transmission attempts per packet (slot) time
- With Aloha S_{max} is for G =0.5
- With Slotted-Aloha S_{max} is achieved for G=1

CSMA

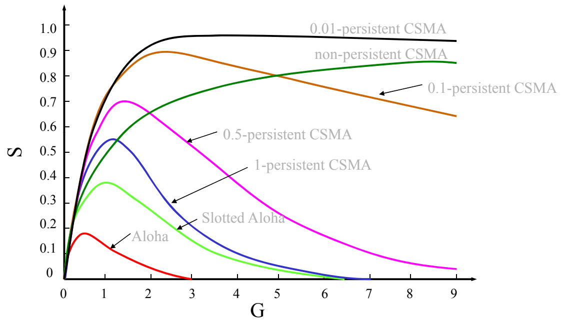
- Carrier Sense Multiple Access (CSMA):
  - Sense the channel every time you have a packet to send. If the channel is free (idle) then send your packet
  - If the channel is busy then retry



CSMA non persistent
CSMA x-persistent

- CSMA 1-persistent
  - Step 1: if the channel is free, transmit the packet
  - Step 2: if the channel is busy, continue to listen the channel until it is free then transmit the packet
    - If two nodes are listening the channel when a third node is transmitting, when this last finished the two nodes start transmitting causing a collision
- CSMA non persistent
  - Step 1: if the channel is free, transmit the packet
  - Step 2: if the channel is busy, wait for a random time and repeat Step1
    - Random backoff reduces collisions probability
    - Too long backoff reduces the throughput
- CSMA p-persistent
  - This algorithm is usually used when the time is divided in slots
  - Step 1: if the channel is free, transmit with probability p and defer to next time slot with probability $1-p$
  - Step 2: if the channel is busy, continue to sense the channel. When the channel is free repeat Step 1
  - Step 3: if transmission is deferred by a time slot repeat Step 1
    - A tradeoff between non-persistent and 1-persistent

Throughput (S) vs Offered Load (G)



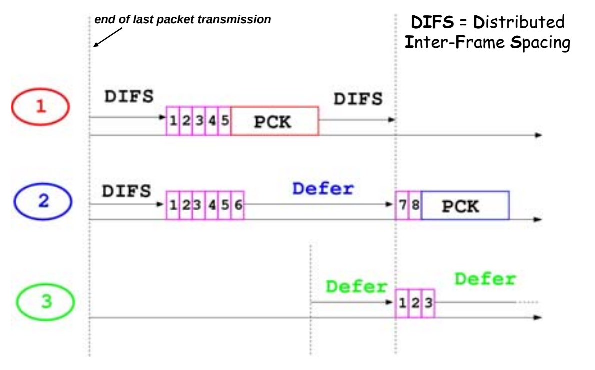
CSMA/CA

- To reduce the wasted time due to collisions, if two or more nodes transmit at the same time, hence there is a collision, it would be better that the nodes stop transmitting
- Possible with wired networks, because a node can transmit and listen the channel at the same time (e.g *CSMA/CD-Ethernet*)
- With a wireless channel, to transmit and to listen at the same time is difficult or even impossible
- Solution: *CSMA/CA* (*Collision Avoidance*)
  - The worst situation: when the medium is busy and two or more nodes are sensing the medium waiting to transmit
  - *CSMA/CA* tries to reduce the collision probability by a random backoff procedure:
    - if the channel is free then backoff for a random time, after that, if the channel is still free transmit

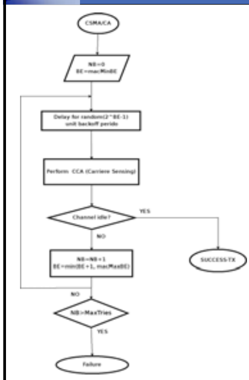
IEEE 802.11 CSMA/CA

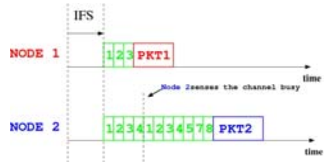
- **Nodes ready** to transmit **sense** the medium
- If the **channel is busy**, wait until the **end of current transmission**
- Then **wait** for an **additional** predetermined time period **DIFS** (Distributed Inter Frame Spacing)
- Then pick up a **random number of slots** (the initial value of **backoff counter**) within a **Contention Window** to wait before transmitting the **frame** (**packet**)
  - Contention Window is defined by [0,CW], where $CW_{min} \leq CW \leq CW_{max}$
- If there are **transmissions** by **other nodes** during this time period (backoff time), the node **stops its counter**
- It **resumes count down** after nodes **finish transmission** plus **DIFS**. The node can start its transmission when the **counter** value is **zero**
- If the **channel access fails** (e.g. there is a **collision**), then increment the **CW** value. (CW = 2*CW)
- The **initial backoff makes CSMA/CA** similar to **p-persistent CSMA**

IEEE 802.11 CSMA/CA



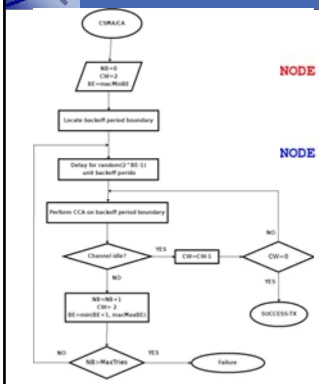
IEEE 802.15.4/ZigBee un-slotted CSMA/CA





- NB: **number of performed tries**
- BE: **backoff exponent**
  - **Backoff interval**: (0, 2^{BE} -1)
- The **initial backoff** makes **CSMA/CA** similar to **p-persistent CSMA**

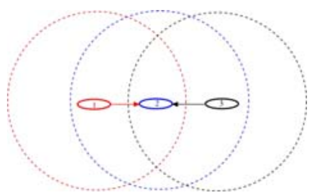
IEEE 802.15.4/ZigBee slotted CSMA/CA



- NB: number of tries performed
- BE: backoff exponent
  - Backoff interval: (0, 2^{BE} -1)
- CW (Contention Window). CW=2
- The initial backoff makes CSMA/CA similar to p-persistent CSMA

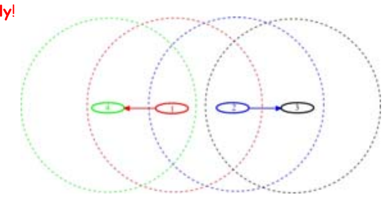
Hidden Node problems

- CSMA protocols suffer the Hidden Node problem:
  - Node 1 wants to transmit to Node 2 , it finds the channel free and starts transmitting
  - Node 3 wants to transmit to Node 2, since Node 3 is out of range with respect to Node 1, it finds the channel free and start transmitting to Node 2
  - There is a collision between the transmissions of Node 1 and Node 3



Exposed Node problem

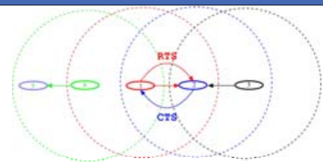
- CSMA protocols suffers the Exposed Node problem:
  - Node 1 wants to transmit to Node 4, it finds the channel free and starts transmitting
  - Node 2 wants to transmit to Node 3, if finds the channel busy, then it blocks waiting for the channel to be free
  - Transmission from Node 1 cannot reach Node 3, transmission from Node 2 cannot reach Node 4, therefore Node 1 and Node 2 could **transmit simultaneously**!



Mitigating Hidden/Exposed Node Problem

- The **Hidden**/**Exposed** Node Problem can be **mitigated** by a **handshaking mechanism:**
  - A node that **wants to transmit sends** a Request To Send (**RTS**) packet to receiver node
  - The receiver replies with a Clear To Send packet (**CTS**)
  - A node that **ears** a **CTS** packet **keeps silent** for duration of incoming transmission
  - A node that ears a **RTS** packet but not a **CTS**, assumes to be an **Exposed** node, then it can transmit also whether it finds the channel busy for the duration of the incoming transmission
  - Both **RTS** and **CTS** report the length of the packet being to be transmitted
- This **mechanism** is used, for instance, in **IEEE 802.11**, **MACA**, **MACAW** protocols

Mitigating Hidden/Exposed Node Problem



- Both Node 1 and Node 3 want to send a packet to Node 2
- Node 1 senses the channel free and sends a **RTS** packet
- Node 2 receives the **RTS** and responds with a **CTS** packet
- Node 3 receives the **CTS** then keeps silent
- Node 4 receives a **RTS** but not the **CTS**, then it assumes to be an **exposed node**
- Node 1 transmit its packet
- Node 4 being an **exposed node** might transmit a packet even if it senses the channel busy

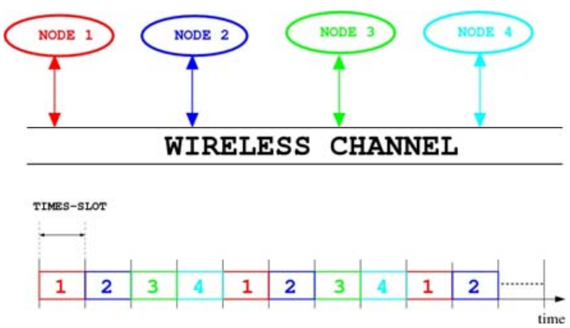
Scheduling approaches

- The **time** is divided in **slots**
- Each **time slot** is **reserved**/**dedicated** to a **node**
- Each node has an **exclusive access** to its **time slots**: **no collisions**
- **Different scheduling policy** can be used to assign the **time-slots**:
  - Round Robin (**RR**)
  - Weighted Round Robin (**WRR**)
  - Rate Monotonic (**RM**)
  - Earliest Deadline First (**EDF**)
  - Etc.
- Any **algorithm** from **resource scheduling theory** might be **applied**
- **Time slot dimension** is an **important parameter**:
  - All packets have the **same dimension**: time-slot=packet time
  - Packets have **different dimension**: an important **portion of bandwidth** can be **lost**: **a bandwidth reclaiming mechanism is desirable**
- **Nodes must be synchronized**: **synchronization mechanisms** are needed

Fully distributed Scheduling Approaches

- In case of fully distributed approaches:
  - Each **node** must **know/build** the **schedule**
  - In order to build a **common schedule** either each node must **know** the **traffic parameters** of other nodes, or at least some **common information** should be **shared** by the nodes
- Example of such approaches: **Implicit EDF**
- We will see some details of **Implicit EDF** later

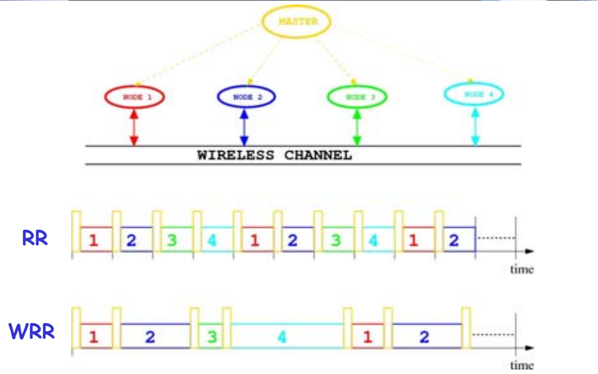
Round Robin Scheduling



Coordinated (Centralized) Scheduling Approaches

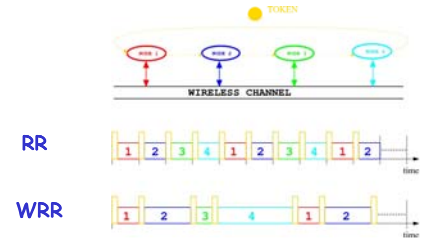
- There is a central **Coordinator** node (a.k.a. **Master**)
- The **Coordinator** decides when a node can **access** the **channel**
  - **Polling** (Master/Slave):
    - **Coordinator** polls the nodes for packet transmission using a scheduling policy (**Bluetooth**)
  - **Access Window** approach:
    - **Coordinator** defines a channel **Access Window** by means of a **periodic beacon** transmission. The **Access Window** is defined by two consecutive **beacons** (Ex: IEEE 802.15.4 beaconed-mode)
    - The **Access Window** is divided in **time slots**
    - **Coordinator** communicates the **Access Window** scheduling in the **beacon** packet (for instance)
- Both **poll** and **beacon** mechanisms **synchronize** the nodes

Polling approach (MASTER SLAVE):



Token Passing

- There is a **token traveling** among the nodes
  - Each node has a **time budget**
  - Every time a node **receives** the **token**, it can transmit its traffic for a **time no greater** then its **budget**
  - It needs a **policy** to exchange the **token** among the nodes (e.g. **circular fashion**)
  - It needs a **policy to assign the budgets** (e.g. **Weighted RR**)

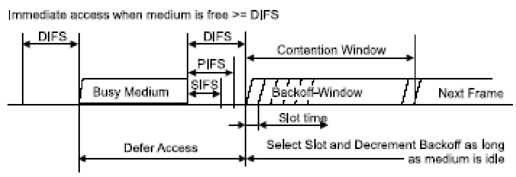


Mixed Approaches (Hybrid)

- **Mixed approaches** exploit both CSMA techniques and **scheduling based** (**collision-free**) techniques
- Several **standard protocols** use a **mixed approach**: e.g. **IEEE 802.11** and **IEEE 802.15.4**
- **IEEE 802.11 (Wi-Fi)**
  - **D**istributed **C**oordination **F**unction (**DCF**)
    - **RTS/CTS** + **CSMA/CA** + **NAV** (**N**etwork **A**llocation **V**ector ->**Virtual Carrier Sensing**)
    - **Virtual Carrier Sensing**: a node extracts the length of the incoming transmission from either **RTS** or **CTS**, then keeps silent for the entire packet length
    - A **positive ACK** is used to confirm the packet has been received correctly;
  - **P**oint **C**oordination **F**unction (**PCF**) (Polling approach)
    - **Central Coordinator** (**Access Point**)
- IEEE 802.15.4: Access Window with CSMA/CA and **reserved time slots** (more details later)

802.11-DCF and PCF coexistence

- **I**nter **F**rame **S**pacing (**IFS**): minimum space between two consecutive packets
- **D**istributed **IFS** (**DIFS**): between consecutive packets under **DCF**
- **P**oint **IFS** (**PIFS**): **PCF** traffic
- **S**hort **IFS**: **ACK** or **CTS**



Real-Time MAC protocols

- ***Contention Based Protocols***
  - ***Differentiation Mechanisms*** for IEEE 802.11 CSMA/CA
  - Black Burst
- ***Scheduling Based Protocols***
  - Implicit EDF
  - WBuST
  - …
- ***Mixed Contention and Scheduling Protocols***
  - IEEE 802.15.4/ZigBee
  - …

Differentiation Mechanism IEEE 802.11

- **IEEE 802.11 DCF** fairness: each node has the **same probability** to access the channel
- For a **timely communication** (**QoS** in general), a node (**network traffic source**) should receive:
  - **precedence (probability) on channel access** based on its **traffic priority**
  - a portion of **bandwidth proportional** to its **priority/traffic parameters**
- **Priority traffic differentiation mechanisms**:
  - *Scaling Contention Window* (CW) according to the **priority** of each traffic source (**node**)
  - Assigning **different DIFSs** based on the **priority** of traffic sources

Contention Window Scaling

$$CW_i = \left\lceil CW\left(2 + \frac{priority_i - 1}{\max\_priority}\right)\right\rceil$$

- **CW** is expressed in **time slots**, e.g. $CW=4$ backoff slots
  - CW is the base value for the **Contention Window**
  - CW_i is the **Contention Window** of node i
- **Example:**
  - Network composed by n nodes
  - Each node has a **periodic stream** $S_i=(m_i,T_i,D_i=T_i)$
  - Node **priority** assigned by **Rate Monotonic**
  - $priority_i$ proportional to T_i (**RM**)
  - $max\_priority$ proportional to $\max(T_i)$ (**RM**)
  - The **higher the priority number**, the **lower the priority**
  - The **higher the priority**, the **lower CW**

DIFS differentiation

$$DIFS_i = BASE\_DIFS * priority_i$$

- **Example:**
  - Network composed by n nodes
  - Each node has a **periodic stream** $S_i=(m_i,T_i,D_i=T_i)$
  - Node **priority** assigned by **Rate Monotonic**
  - $priority_i$ proportional to T_i (**RM**)
  - The **higher the priority number**, the **lower the priority**
  - The **higher the priority** the **lower DIFS**

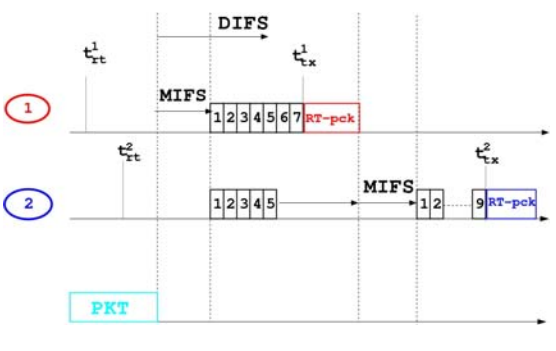
IEEE 802.11e

- **IEEE 802.11e** is the **standard version** supporting **QoS requirements**
- It defines Enhanced **DCF** (**EDCF**) which provides service **differentiation mechanisms**
- It defines also a **new polling based access mechanism** called Hybrid Coordination Function (**HCF**) Controlled Access Channel (**HCCA**) (an enhanced **PCF**).
- **EDCF** defines **four classes** of channel Access Categories (**ACs**)
- Each **AC** has a **different priority**
- **Service differentiation** is achieved by:
  - Contention Window differentiation: it assigns to each **AC** a different CW_{min}, CW_{max}
  - DIFS differentiation: Instead of using an unique **DIFS**, **EDCF** uses a different Arbitration **IFS** (**AIFS**) value for each **AC**. The **higher** the **AC** priority the **shorter** the **AIFS**.

Black Burst

- **Black Burst** is a technique to guarantee a **better performance for real-time traffic** under **IEEE 802.11**
- A Real-Time (**RT**) node is one that has **real-time traffic** to deliver
- **RT** nodes contend to access the channel after a Medium **IFS** (**MIFS**<**DIFS**)
- **RT** nodes sort the **access right** by **jamming** the channel **sending pulses of energy** (**BB**)
- The **node** that sends the **longest BB wins** the contention and **it can transmit its real-time packet**

Black Burst



Black Burst

$$t_{rt}^i = t_{tx}^i + t_{sch} \qquad BB(t_{rt}^i) = (1 + t - t_{rt}^i)\,t_{bbslot}$$

$t \rightarrow$ Current time instant

$t_{rt}^i \rightarrow$ Time instant at which node *i* attempts to access the channel for transmitting

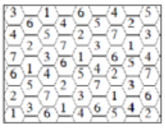
$t_{tx}^i \rightarrow$ Time instant at which node *i* transmits its real-time packet

$t_{sch} \rightarrow$ Minimum interval between two consecutive real-time packet transmission attempts (equal for all nodes)

$t_{bbslot} \rightarrow$ BB slot dimension

Implicit EDF (IEDF) [Caccamo et al. 2002]

- Nodes are grouped into **hexagonal cells**: __cellular structure__
- Each cell contains a **router** and a **set of nodes**
  - Each **router** has **two radios**
- A **radio channel** is assigned to **each cell**
- **Intra-cell** and **Infra-cell** communications



Implicit EDF (IEDF)

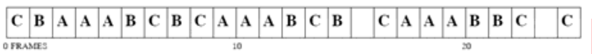
- It uses a **scheduling base** channel access mechanism
- It uses the Earliest Deadline First (**EDF)** algorithm to compute the **transmission schedule**
- Each node must know the traffic parameters of each other node:
  - S_i=(m_i,T_i,D_i=T_i) Traffic Parameters
    - T_i message period
    - D_i message relative deadline
    - m_i message length (#packets)
- Each node computes the schedule. The schedule is replicated at each node:
  - __Each node will know which one has the shortest deadline hence the right to access the channel to transmit__.
- Each nodes has an exclusive access to the channel

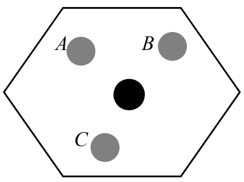
IEDF Intra-cell communication

Message table

nodes rank

Transmitter node	Message length	Message period
A	3	8
B	1	6
	1	8
C	1	4



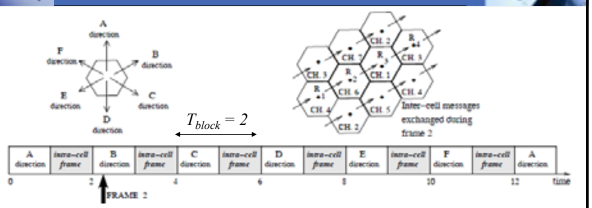


IEDF

- Nodes must by **synchronized**
- **Unused bandwidth problem**: a **reclaiming bandwidth mechanism** is necessary: **FRASH**
- **Dynamic schedule update** mechanism is needed when a node wants to **join** the network or a node **leaves** the network
- It is possible to manage both **periodic traffic** and **sporadic traffic** (through **Aperiodic Servers**)
- Consider having a message stream set M =(S_1,S_2...S_n), a set of s **servers** with:
  - Q_j **Server Capacity**
  - Ts_j **Server Period**
- **Stream set Feasibility Test** (__classic EDF+Servers test__):

$$\sum_{i=1}^{n}\frac{m_i}{T_i}+\sum_{j=1}^{s}\frac{Q_j}{Ts_j}\leq 1$$

IEDF Infra-cell communication



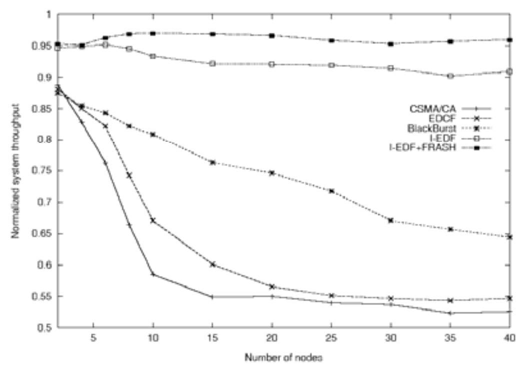
Schedulability Test

$$\forall i,\quad i=1,\ldots,n \quad \sum_{k=1}^{n}\frac{m_k}{T_k}+\frac{B_i}{T_i}\leq 1$$

$$B_i=\left\lceil\frac{T_{block}}{T_i}\right\rceil$$

A inter-cell frame is inserted every $T_{block} \geq 2$

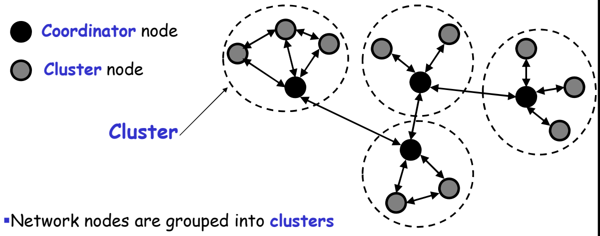
IEDF



WBuST

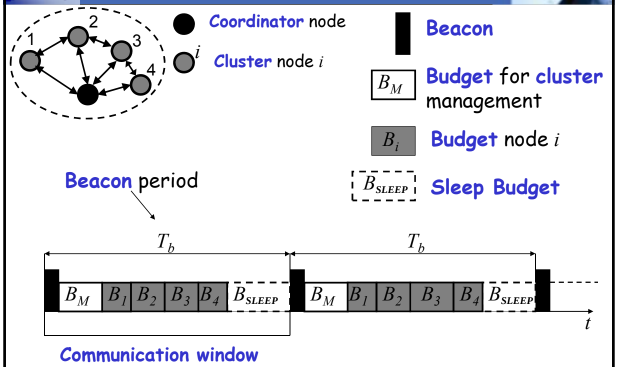
- The **W**ireless **B**udget **S**haring **T**oken (**WBuST**) protocol is a **MAC layer** protocol designed for **real-time communication** in wireless networks
- **WBuST** supports both **single-hop** and **multi-hop** communication
- **WBuST** manages both **real-time** and **best-effort** (**non real-time**) traffic

Network structure



- Network nodes are grouped into **clusters**
- A **cluster** is formed by a **coordinator** node and a set of **cluster** nodes
- Each **cluster** has a different **communication channel**
  - Communications inside different **clusters** can proceed in parallel without interfere

Intra-cluster communication example

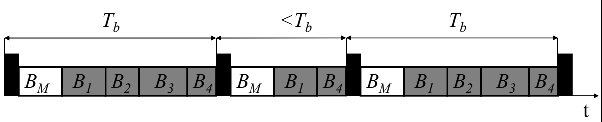


Traffic model

- Each **cluster** node i can have a **periodic stream** of messages $S_i(m_i, T_i, D_i)$:
  - m_i: maximum message length
  - T_i: message period
  - D_i: relative deadline
- It is also possible to represent **sporadic** streams with the **same model**
- Each node can also have **non real-time** traffic to deliver
- A node i uses its **time budget** B_i to transmit both **real-time** (**periodic**) and **non real-time traffic**
  - Node budget is shared between real-time and non real-time traffic
  - Real-time traffic has precedence over non real-time traffic

Implicit bandwidth reclaiming

- **Bandwidth reclaiming** under **WBuST**:
  - If node i does not use its **time budget** entirely, it sends a **TX-END packet** to node i+1
  - Node i+1 can so start using its **budget** before than expected
  - When node i+1 finishes its transmission sends a **TX-END packet** to node i+2, and so on until the last node
  - If the last node (node 4 in the example) sends a **TX-END packet**, then the **coordinator** transmits the new **beacon**



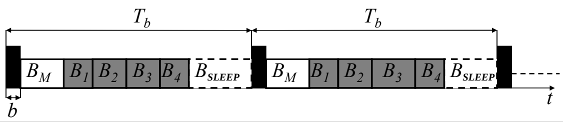
Energy saving

- **Energy model**:

$$P_i = \left[P_i^{TX} \cdot U_i^{TX} + P_i^{RX} \cdot U_i^{RX} + P_i^{SL} \cdot U_{SL} \right] \cdot \frac{t}{T_b}$$

$$\begin{cases} U_i^{TX} = \dfrac{B_i}{T_b} \\ U_i^{RX} = \dfrac{b}{T_b} + \displaystyle\sum_{j=1, j \neq i}^{n} \dfrac{B_j}{T_b} \\ U_{SL} = \dfrac{B_{SL}}{T_b} \end{cases}$$

- P_i is the **average power** wasted after t time units
- Given a **desired network life time**, by means of the **energy model**, both **node budgets** and **sleep budget** are calculated to guarantee both the desired **network lifetime** and the **message deadlines**



Budget Allocation

- Time budgets B_i are assigned by **B**udget **A**llocation **S**chemes (**BAS**)

- **Example**. **M**odified **L**ocal **A**llocation (**MLA**):

$$B_i = \frac{C_i}{\left\lfloor \frac{D_i}{D_{\min}} \right\rfloor} \qquad D_{\min} = \min(D_i)$$

- Stream node i: $S_i(C_i, T_i, D_i)$
- $D_i \le T_i$

Message deadlines guarantee

- Consider a cluster of n nodes that generate a **periodic** traffic described by a stream set $\mathbf{M} = \{S_1, S_2, \ldots, S_n\}$
- Message deadlines guarantee is based on U^*
- U^* is the **least upper bound** on the **stream set utilization** U^{RT}, which guarantees the message deadlines

$$U^{RT} = \sum_{i=1}^{n} \frac{C_i}{T_i}$$

Sufficient Condition for stream set feasibility:
given a stream set $\mathbf{M} = \{S_1, S_2, \ldots, S_n\}$, if $U^{RT}(\mathbf{M}) \le U^*$ then $\mathbf{M}$ is feasible.

- A stream set $\mathbf{M}$ is said to be feasible when all message deadlines are met.

Message deadlines guarantee

- Under **WBuS**, U^* depends on both the choice of **beacon** period T_b and D_{min}.

- **Example**. **M**odified **L**ocal **A**llocation (**MLA**):

$$U^* = \frac{\left\lfloor \frac{D_{\min}}{T_b} \right\rfloor}{\left\lfloor \frac{D_{\min}}{T_b} + 1 \right\rfloor}(1-\alpha)$$

$1-\alpha$ **Available bandwidth**

$\alpha = \frac{H_M + t_{bc}}{T_b}$ **Bandwidth lost due to protocol overhead**

Hardware/Software tools

Hardware:

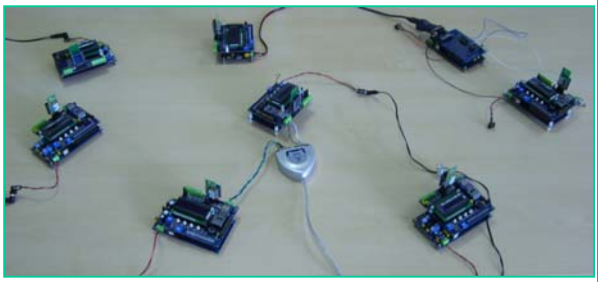
- **FLEX**
  - Base board featuring Microchip dsPIC33FJ256MC710
  - Daughter board

- ***CC2420***
  - IEEE 802.15.4 compliant transceiver

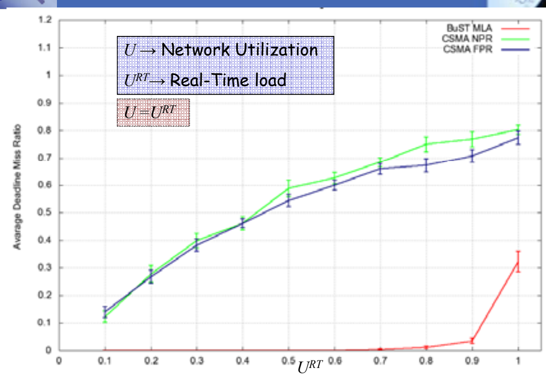
Software:

- RTOS **Erika Enterprise**

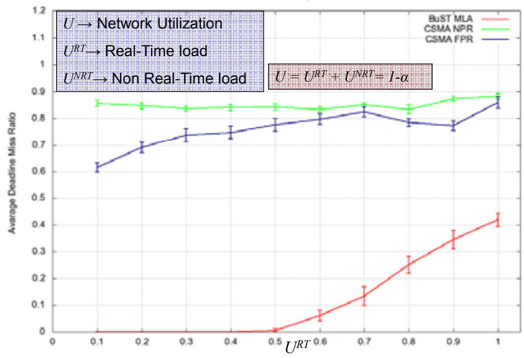
Experimental Analysis



6 nodes, only RT traffic



6 nodes, RT and Best Effort traffic



RETIS LAB

Real-Time Systems Laboratory

Retis Real-Time Systems Laboratory

uWireless and IEEE 802.15.4

Gianluca Franchino

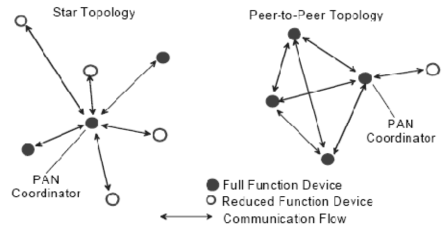
g.franchino@sssup.it

IEEE 802.15.4/ZigBee

- **IEEE 802.15.4** defines **MAC** and **PHY** layers
- **ZigBee** defines the **Network layer** and the **Application layer**
- Nodes are **grouped** into Personal Area Network (**PAN**):
  - A **PAN** is a **cluster of nodes** managed by a **Coordinator** node
- **PAN** Topologies: **Star** and **Mesh**
- **IEEE 802.15.4** defines **two working modes**:
  - **Non-beaconed mode** operation
  - **Beaconed mode** operation
- **Non-beaconed mode** (used by **ZigBee**):
  - **whenever** a node in the **PAN** wants to transmit, it uses the **un-slotted CSMA/CA** algorithm
  - The task of the **Coordinator** is to **manage** the **PAN**:
    - **Es.** Node **association/disassociation**

IEEE 802.15.4 topologies

- **Two kinds of device:**
  1. Full Function Device (**FFD**);
  2. Reduced Function Devices (**RFD**).
- **FFD** can communicate to both **FFD** and **RFD** devices;
- **RFD** can communicate only to **FFD** devices.



IEEE 802.15.4/ZigBee

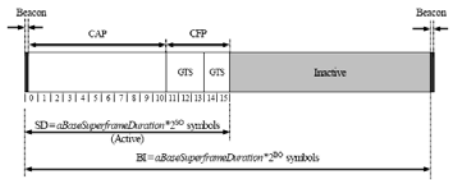
▪**Beaconed mode:**

–**Coordinator** defines a **superframe structure** by sending a **periodic beacon**

–The **superframe** contains 16 **time slots**

–The **superframe** contains a Contention Access Period (**CAP**) and a Contention Free Period (**CFP**) (optional)

–The **superframe** can contain also an **Incative** portion to **save energy** (optional)



IEEE 802.15.4/ZigBee

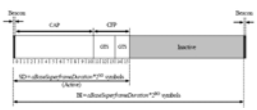
▪During the **CAP** nodes use **slotted-CSMA/CA** to:

–Send data packets

–Send GTS allocation requests

–To join the PAN

–Etc.



▪Nodes can have Guaranteed Time Slots (**GTS**) **allocated** in the **CFP**

▪**GTS** is composed by **one or more** superframe **slots**

▪Max 7 **GTSs**

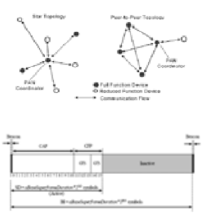
▪During the **CFP**, there are **no collisions**: **Exclusive channel access** by nodes **holding** a **GTS**

▪**Scheduling** of **GTSs** is contained in the **beacon**

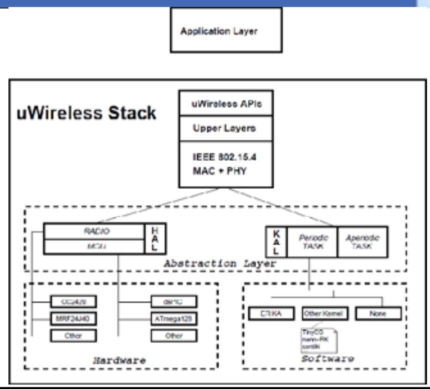
µWireless

- µWireless: an implementation of the IEEE802.15.4 standard for WPAN
- Fast prototyping of IEEE 802.15.4 based protocols
  - To decrease the time for developing new protocols
  - Example: BACCARAT a protocol for bandwidth management in WSN
- µWireless is developed by a software architecture that makes code writing independent from both the hardware and the RTOS

IEEE 802.15.4



µWireless architecture



RETIS LAB

Real-Time Systems Laboratory

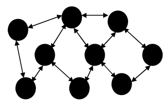
Real-Time Capacity of Multihop Wireless Networks

Gianluca Franchino

g.franchino@sssup.it

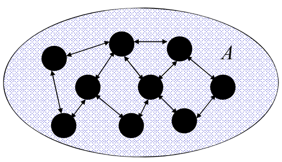
Introduction

- **Assumptions**:
  - Nodes communicate without any centralized control: Ad Hoc networks
  - Multihop networks
  - A proper scheduling scheme is used
- **Transport Capacity of a Wireless Network**:
  - sum of products of bits and the distances over which they are carried in an unit of time ([bit-meters/sec])
  - A capacity of k bit-meters: k bits have been transported over a distance of one meter toward its destination



Arbitrary Networks

- **Arbitrary networks**:
  - Each node can transmit at W bps (bits per second) over a wireless channel
  - n nodes located arbitrarly in a disk of area A
  - Nodes send traffic at an arbitrary rate to an arbitrary destination → Arbitrary Traffic Pattern



Arbitrary Networks [Gupta and Kumar 2000]

Network Transport Capacity $\quad C = \Theta\left(W\sqrt{An}\right)$ [bit-meters/sec]

Per node Transport Capacity $\quad C = \Theta\left(\frac{W\sqrt{A}}{\sqrt{n}}\right)$ [bit-meters/sec]

Per node Troughput $\quad C = \Theta\left(\frac{W}{\sqrt{n}}\right)$ [bit/sec]

Knuth's notation:

- $f = \Theta(g(n))$ *iff* $f(n) = O(g(n))$ and $g(n) = O(f(n))$
- f is bounded both above and below by g asymptotically

Real-Time Capacity

- **Capacity bounds** give an inside on the **network throughput** as a function of **network parameters**:
  - Bandwidth (W), size (A) and average density (# nodes n) etc.
- **Real-Time Capacity**: it concerns capacity limits on real-time information transfer in mutlihop wireless networks
- **Schedulability** in distributed system is **NP-Hard**, hence, no closed-formula for **Real-Time Capacity**
- **Real-Time Capacity** depends on the **packet scheduling** protocol
- **Sufficient** (**not necessary**) closed-formula for **fixed-priority** packet scheduling protocols [Abdelzaher et al. 2004].
- **Real-time capacity** C_{RT} of a network (sufficient bound)
- The **capacity requirement** U of a wireless multihop network is given by the **bit-meters product** of messages normalized by their **relative deadlines**

Real-Time Capacity

- **Adopting the usual notation**:
  - Considering a set M(S_1, …, S_i, S_n) of **message streams** $S_i(C_i, D_i)$, with $i=1,\dots,n$
  - C_i is the product of the **message lenght** (bits) and the **destination distance** (meters)
  - D_i is the **relative deadline**

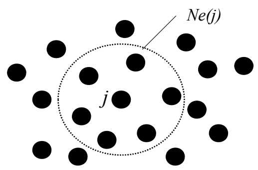
Capacity Requirement

$$U = \sum_i \frac{C_i}{D_i}$$

Sufficient Condition for stream set feasibility: given a multihop wireless network with a capacity requirement U, if $U \le C_{RT}$ then all message deadlines will be met

Capacity Requirement

- All nodes have on average m neighbour nodes
  - Distances measured in *hops*
  - m is called node density ([nodes/hops])
- *Ne(j)*: neighbourhood of node j, which is the set of nodes whose transmissions can be heard by node j
- H_j is the traffic generated into *Ne(j)*



$$U = \frac{W}{m}\sum_j H_j$$

$$H_j = \sum_{i \in Ne(j)} \frac{C_i}{D_i}$$

Load-balanced Networks

- N is the **communication diameter** ([*hops*])
  - N is the longest communication path a node can be part of
- α is the **minimum relative deadline ratio** across all priority-sorted packet
  - It describes the degree of **urgency inversion**: a packet with a shorted deadline receive a priority lower than a packet with a larger deadline
- **Deadline Monotonic** (**DM**) algorithm: the lower the deadline the greater the priority
  - With Deadline Monotonic $\alpha=1$ (**it maximizes** C_{RT})
  - It happens beacuse for any possible couple of packets $D_{lo} = D_{hi}$
- Assigning priorities **randomly** (**RAND**)

$$\alpha = \min_{pck^{hi},pck^{lo}} \left(\frac{D_{lo}}{D_{hi}}\right)$$

RAND

$$\alpha = \frac{D_{\min}}{D_{\max}} \le 1$$

DM

$$\alpha = 1$$

C_{RT} for load-balanced networks

$$C_{RT} = \frac{n}{m}\left(1 + \frac{\alpha}{N} - \sqrt{1 + \left(\frac{\alpha}{N}\right)^2}\right)W$$

For large networks, N is large $\left(\frac{\alpha}{N}\right)^2 << 1 \Rightarrow C_{RT} = \frac{n\alpha}{mN}W$

If N is bounded and does not depends on n $\Rightarrow C_{RT} = O(n)\frac{\alpha}{m}W$

If the routing paths are randomly chosen:

$$\frac{n}{N} = O(\sqrt{n}) \Rightarrow C_{RT} = O(\sqrt{n})\frac{\alpha}{m}W$$

Realistic MAC protocols

- Before we considered **ideal MAC protocols** with **no time overhead** (mainly due to channel access arbitration)
- Considering the **overhead**, let d the **maximum delay** that a packet can experience for **channel arbitration** at each hop

$$C_{RT} = \frac{n}{m}\left(1 + \frac{\alpha'}{N} - \sqrt{1 + \left(\frac{\alpha'}{N}\right)^2}\right)W$$

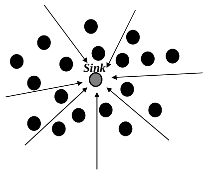
$$\alpha' = \alpha\left(1 - N\frac{d}{D_{\min}}\right) \le \alpha$$

$$D_{\min} = \min_i(D_i)$$

Un-balanced networks: WSN example

- **Realistic load patterns** are **difficult to characterize**
- **WSN (Wireless Sensors Network) example**:
  - K sink nodes
  - N_k maximim number of *hops* between a sensor and a sink

$$C_{RT} = \frac{\alpha K N_k}{1 + 0.5 \ln N_k} W$$



How to use Real-Time Capacity

- C_{RT} **can be used to**:
  - Given a set of message streams, **verify the schedulability** ($U \le C_{RT}$)
  - **Find network parameters** that guarantee **packets schedulability**

WSN dimensioning

- **Consider a WSN composed by**:
  - $n = 1000$ nodes, $K = 5$ sink nodes
  - $N_k = 5$ is the **average hops** between sensors and sinks
  - $W = 250\ kbps$
  - n periodic streams $S_i(C_i, T_i, D_i)$: for any i, $C_i = C = 25$ bytes, $D_i = 1$ sec, $T_i = T$
  - **Deadline Monotonic** $\rightarrow \alpha = 1$

Real-Time Capacity

$$C_{RT} = \frac{\alpha K N_k}{1 + 0.5 \ln N_k} W = 434027 \left[\frac{byte-hops}{\sec}\right]$$

Capacity Requirement

$$U = nC \left\lceil \frac{D}{T} \right\rceil \frac{N_K}{D} = 125000 \left\lceil \frac{D}{T} \right\rceil \left[\frac{byte-hops}{\sec}\right]$$

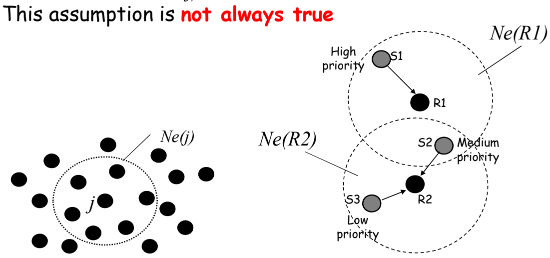
\# messages generated be each source when the system is schedulabe

Schedulabilty Condition

$$U \le C_{RT} \rightarrow T \ge 330 \quad [m\sec]$$

Pseudo Priority Inversion

- So far we assumed that a packet transmission in a neighbourhood *Ne(j)* contends only with other transmissions in *Ne(j)*
- This assumption is **not always true**



Real-Time Capacity and Pseudo-Priority inversion

- Considering both Pseudo-Priority Inversion and channel arbitration delay:

Real-Time Capacity for load balanced networks

$$C_{RT} = \frac{n\alpha'}{2mN} W$$

Real-Time Capacity for networks with K sink nodes

$$C_{RT} = \frac{\alpha' K N_k}{2 + \ln N_k} W$$

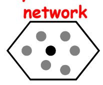
$$\alpha' = \alpha\left(1 - N\frac{d}{D_{\min}}\right)$$

$$D_{\min} = \min_i (D_i)$$

Real-Time Capacity of IEDF

- Caccamo and Zhang (2003) derived the Real-Time Capacity bound (throughput) of Implicit EDF

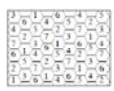
Fully connected network



$$C_{RT} = \Theta\left(\frac{W}{N}\right)$$

N is the number of nodes

Cellular structure



$$C_{RT} = \Theta\left(\frac{W}{nc}\right)$$

- n is the number of nodes per cell
- c number of channels

Literature

- **Distributed Systems**
  - Paulo Verissimo, Luìs Rodrigues. “*Distributed Systems for System Architects*”. Kluwer Academic Publisher
- **A general overview on networking**
  - Andrew Tanembaum.” *Computer Networks* ”. Prentice Hall
- **Wireless Sensor Networks (protocols and energy aware issues)**
  - Holger Carl, Andreas Willing. “*Protocols and Architecture for Wireless Sensor Networks*”. Wiley-Interscience
- **Real-time, QoS and resource management on Wireless Communication**
  - Mihaela Cardei, Ionut Cardei, Ding-Zhu Du. “*Resource Management in Wireless Networking*”. Springer
  - Bulent Tavli, Wendi Heinzelman. “*Mobile Ad Hoc Networks, Energy-Efficient Real-Time Data Communications*”. Springer
- **CSMA mechanisms**
  - L. Kleinrock and F. Tobagi. “*Packet switching in radio channels: Part I –carrrier sense multiple-access and modes and their throughput delay characteristics*”, IEEE Transaction on Communication, vol 23, no.12, Dec 1975
  - G.Bianchi. “*Performance analysis of the of the IEEE 802.11 distributed coordination function* ”
- **CSMA traffic Differentiation Mechanisms**
  - Imad Aad, Claude Castelluccia. “*Differentiation Mechanisms for IEEE 802.11*”, INFOCOM 2001
  - Yang Xiao. “*Performance Analysis of Priority Schemes for IEEE 802.11 and IEEE 802.11e Wireless LANs*”, IEEE Transaction on wireless communications, vol. 4, no. 4, July 2005

Literature

- **Black Burst**
  - Joào L. Sobrinho, A. S. Krishnakumar.”*Quality-of-Service in Ad Hoc Carrier Sense Multiple Access Wireless Networks* ”. IEEE Journal on selected areas in communication, vol. 17, NO. 8, August 1999
- **Scheduling Protocols (I-EDF)**
  - M. Caccamo, L. Y. Zhang, L. Sha, and G. Buttazzo. “*An Implicit Prioritized Access Protocol for Wireless Sensor Networks”. Proceedings of the IEEE Real-Time Systems Symposium*, Dec. 2002
  - T. L. Crenshaw, A. Tirumala, S. Hoke, and M. Caccamo, "*A Robust Implicit Access Protocol for Real-Time Wireless Collaboration*", Proceedings of the IEEE Euromicro Conference on Real-Time Systems, Palma de Mallorca, Spain, July 2005
- **IEEE 802.15.4 analysis**
  - J. Mišic, S. Shafi, and V. B. Mišic, “*The Impact of MAC Parameters on the Performance of 802.15.4 PAN*”, Elsevier Ad hoc Networks Journal, 3(5):509–528, 2005.
  - Anis Koubaa, Mário Alves, Eduardo Tovar. “*A Comprehensive Simulation Study of Slotted CSMA/CA for IEEE 802.15.4 Wireless Sensor Networks*” Proceedings of the 5th IEEE International Workshop on Factory Communication Systems (WFCS’06), Torino, Italy, JUN, 2006
  - Anis Koubaa, Mário Alves, Eduardo Tovar. **“***An implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks: theory and practice*” Real-Time Systems Journal, Volume 39, Numbers 1-3, pp 169 - 204, Springer, August 2008
    - http://www.springerlink.com/content/u203825646vv811r/
  - At the following link you can find several papers on real-time communication over IEEE 802.15.4, and other similar topics
    - http://www.cister.isep.ipp.pt/asp/list_docs2.asp

Literature

- **Real-Time Capacity**
  - P. Gupta, P. R. Kumar, “*The Capacity of Wireless Networks*”, IEEE Trans. on Information Theory, Vol. 46, No. 2, March 2000
  - T. F. Abdelzaher, S. Prabh, R. Kiran, “*On Real-Time Capacity Limits of Multihop Wireless Sensor Networks*”, Proc. Of the 25th IEEE Real-Time Systems Symposium (RTSS2004), Lisbon portugal, Dec. 2004
  - M. Caccamo and L. Y. Zhang, "*The Capacity of Implicit EDF in Wireless Sensor Networks*", IEEE Proceedings of the 15th Euromicro Conference on Real-Time Systems, Porto, Portugal, July 2003