# Lecture 2

## Linear Matrix Inequalities. Control Networks

Giancarlo Ferrari Trecate[1]

[1] Dependable Control and Decision Group
École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
`giancarlo.ferraritrecate@epfl.ch`

# Recap from last lecture

## LTV DT model

$$x(k+1) = A(k)x(k) + B(k)u(k)$$

Stability of the system = stability of the equilibrium $(\bar{x}, \bar{u}) = (0,0)$

## Lyapunov theorems with candidate Lyapunov function $V(x) = x^T P x$

- For the LTI system $x(k+1) = Ax(k)$

$$\text{AS/ES} \iff \exists P = P^T > 0 \text{ verifying } A^T P A - P < 0 \qquad (1)$$

- For the DT linear switched system $x(k+1) = A_{\sigma(k)}x(k)$, $\sigma(k) \in \mathcal{I} = \{1, \ldots, M\}$

$$\exists P = P^T > 0 \text{ verifying } A_i^T P A_i - P < 0, \qquad \forall i \in \mathcal{I} \Rightarrow \text{ES} \qquad (2)$$

## Problem

How to check the existence of $P$ verifying the inequalities in (1) and (2)?

# Outline

- Introduction to Linear Matrix Inequalities (LMIs)
- Control networks: basics and performance analysis
  - Physical properties of communication links
  - Delays in control networks
  - Packet collisions and MAC protocol
  - Wireless control networks

# Very short introduction to linear matrix inequalities

## Definition

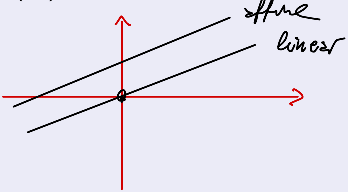
A Linear Matrix Inequality (LMI) is an inequality $F(X) > 0$ where

$$F : V \rightarrow S^n, \quad S^n = \text{set of symmetric } n \times n \text{ matrices}$$

is an affine function and $V$ is a finite dimensional vector space

## Remarks

- $F(X) > 0$ means the matrix $F(X)$ is positive-definite



# Very short introduction to linear matrix inequalities

## Definition

A Linear Matrix Inequality (LMI) is an inequality $F(X) > 0$ where

$$F : V \to S^n, \quad S^n = \text{set of symmetric } n \times n \text{ matrices}$$

is an affine function and $V$ is a finite dimensional vector space

## Remarks

- $F(X) > 0$ means the matrix $F(X)$ is positive-definite
- $F(X)$ is an affine function if $F(X) = F_0 + T(X)$ where $F_0 \in S^n$ and $T(X) : V \to S^n$ is a linear function

# Very short introduction to linear matrix inequalities

## Definition

A Linear Matrix Inequality (LMI) is an inequality $F(X) > 0$ where

$$F : V \to S^n, \quad S^n = \text{set of symmetric } n \times n \text{ matrices}$$

is an affine function and $V$ is a finite dimensional vector space

## Remarks

- $F(X) > 0$ means the matrix $F(X)$ is positive-definite
- $F(X)$ is an affine function if $F(X) = F_0 + T(X)$ where $F_0 \in S^n$ and $T(X) : V \to S^n$ is a linear function
- Let $e_1 \ldots, e_m$ be a basis for $V$ and $X = \sum_{i=1}^m \theta_i e_i$, $\theta_i \in \mathbb{R}$, $i = 1, \ldots, m$. Then, $T(X) = \sum_{i=1}^m \theta_i T(e_i)$, i.e. $T$ is a linear combination of symmetric matrices

# Very short introduction to linear matrix inequalities

## LMI and control theory

Case of interest for control theory: $F : \underbrace{\mathbb{R}^{m_1 \times m_2}}_{V} \to S^n$, i.e. the variable $X$ of $F(X)$ is a matrix

# Very short introduction to linear matrix inequalities

## LMI and control theory

Case of interest for control theory: $F : \mathbb{R}^{m_1 \times m_2} \rightarrow S^n$, i.e. the variable $X$ of $F(X)$ is a matrix

## Example - ~~Stability~~ AS test for LTI systems

The discrete-time system $x(k+1) = Ax(k)$, $x(k) \in \mathbb{R}^n$ is AS iff $\exists P \in S^n$ such that

$$P > 0 \tag{3}$$

$$A^T PA - P < 0 \tag{4}$$

(3) and (4) are matrix inequalities. Are they LMI ? Yes because

- (3) is $F_1(P) > 0$ with $F_1(P) = P$, which is affine in the unknown $P$. Moreover $F_1(P) = F_1(P)^T$

  $$\rightarrow F_2(P)^T = -A^T \underbrace{P^T}_{=P} A + P = -A^T P A + P = F_2(P)$$

- (4) is $F_2(P) > 0$ with $F_2(P) = -A^T PA + P$, which is affine in $P$. Moreover it is easy to show that $F_2(P) = F_2(P)^T$.

1h 22 ↓

# Very short introduction to linear matrix inequalities

## LMI systems

**Proposition.** The system of LMIs

$$\begin{cases} F_1(X) > 0 \\ \vdots \\ F_p(X) > 0 \end{cases}$$

$$A_1 > 0 \quad A_2 > 0$$

$$\Updownarrow$$

$$\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} > 0$$

is equivalent to the single LMI $\text{diag}(F_1(X), \dots, F_p(X)) > 0$

# Very short introduction to linear matrix inequalities

## LMI systems

**Proposition.** The system of LMIs

$$\begin{cases} F_1(X) > 0 \\ \vdots \\ F_p(X) > 0 \end{cases}$$

is equivalent to the single LMI $\mathrm{diag}(F_1(X), \ldots, F_p(X)) > 0$

## Example - (ctd.)

The system $x(k+1) = Ax(k)$, $x(k) \in \mathbb{R}^n$ is asymptotically stable iff $\exists P \in S^n$ such that

$$\begin{bmatrix} P & 0 \\ 0 & -A^T PA + P \end{bmatrix} > 0$$

# Very short introduction to linear matrix inequalities

## LMI optimization problem

$$\min_X c(X) \qquad c(X) : V \to \mathbb{R}$$

subject to

$$\begin{cases} F_1(X) > 0 \\ \vdots \\ F_p(X) > 0 \end{cases}$$

where $c(X)$ is a linear function and $F_i(X) > 0$ are LMIs

## LMI feasibility problem

Check if there is $X$ verifying the constraints

$$\begin{cases} F_1(X) > 0 \\ \vdots \\ F_p(X) > 0 \end{cases}$$

# Very short introduction to linear matrix inequalities

## Remarks

- LMI feasibility and optimization problems are *convex programming problems* for which there are efficient (i.e. polynomial-time) algorithms. Free software in MatLab:
  - LMI control toolbox
  - SDPT3 toolbox
  - SeDuMi toolbox
  - ... and many others

# Very short introduction to linear matrix inequalities

## Remarks

- LMI feasibility and optimization problems are *convex programming problems* for which there are efficient (i.e. polynomial-time) algorithms. Free software in MatLab:
    - LMI control toolbox
    - SDPT3 toolbox
    - SeDuMi toolbox
    - ... and many others
- Tons of interesting problems in control and engineering can be cast into LMIs. See, e.g. the book
    - Boyd, S. and Vandenberghe, L., V. *Convex optimization*, Cambridge University Press, 2004.
- In this course: LMIs for analyzing stability of NCSs

# Example: From LMI to MatLab Code

- Quadratic Lyapunov Function: LMI's

$$\begin{cases} A^T P A - P < -Q \\ P > 0 \end{cases}$$

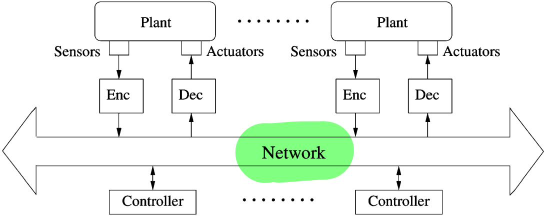
- MatLab + Yalmip code

```
A = 0.1*[-1 2 0;-3 -4 1;0 0 -2];
P = sdpvar(3,3); %Unknown 3x3 symmetric matrix
Q = 1/100 * eye(3,3);
L1 = [A'*P*A - P + Q < 0]; %Constr. 1
L2 = [P > 0]; %Constr. 2
L = L1 + L2; %Combine all constraints
solvesdp(L); %Solving for P (matlab workspace)
P = double(P); %Converts to standard format
```

More in the exercise session !

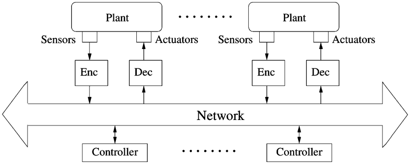
# Control Networks: Basics and Performance Analysis

# Networked Control System (NCS)



- Today we focus on the communication network
    - Goals: understand how it works and sources of delays and packet drop
    - Disclaimer: simplified description!
- NCSs use *control networks.* Why are they needed ?

# Networked Control System (NCS)



## Control networks vs Internet

Control networks

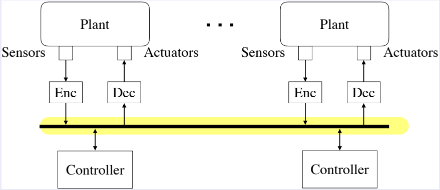
- have simpler topologies (no need of sophisticated routing)
- devices simpler than computers (e.g. a microcontroller does not run several applications in parallel requiring the network)
- shuttle small but frequent packets
- aim at meeting time-critical requirement $\Rightarrow$ support real-time or time-critical applications !

Ideal goal of control nets: transmit a message within a bounded and small time-delay !

# Networked Control System

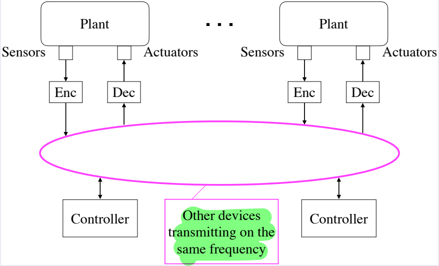
## Reference topologies

### Bus network



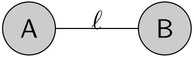
The most frequent topology for a control network

### Wireless network (broadcast)



- Shared medium: how to access it minimizing conflicts ?
- In the sequel: focus on a single link

# Nodes and links



## Physical properties of the link $\ell$

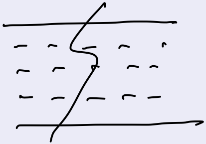
1. $\ell$ = low-pass filter with bandwidth $B$ [Hz]
2. Signal-to-noise (S/N) ratio → the bigger, the better

   → **Shannon's theorem:** every link has a maximal transmission rate $\tilde{B}$ = max n° of bits/sec= $B\log_2(1+\mathrm{S/N})$

   $\tilde{B}$ measured in bits per second (bps). Also called "Bandwidth" in computer science

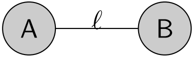
   → Remark: if S/N is not constant, $\tilde{B}$ changes as well!

- $\frac{1}{\tilde{B}}$ : the time it takes for 1 bit to enter the network



- $\tilde{B}$ is independent of the length of $\ell$

# Nodes and links



## Physical properties of the link $\ell$

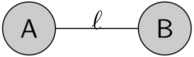
1. $\ell$ = low-pass filter with bandwidth $B$ [Hz]
2. Signal-to-noise (S/N) ratio
   → **Shannon's theorem:** every link has a maximal transmission rate
   $\tilde{B}$ = max n° of bits/sec= $B\log_2(1 + \mathrm{S/N})$
   $\tilde{B}$ measured in bits per second (bps). Also called "Bandwidth" in computer science
   → Remark: if S/N is not constant, $\tilde{B}$ changes as well!

## Example - Telephone line (ADSL)

Link bandwidth: 1 MHz, S/N: 10000⇒ max n° of bps
$= 10^6\log_2(1 + 10000) \simeq 13$ Mbps

Simplifying assumption
↳ We assume S/N ratio is constant. Not true, especially in wireless communication

# Nodes and links



## Physical properties of the link $\ell$

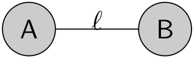
1. $\ell =$ low-pass filter with bandwidth $B$ [Hz]
2. Signal-to-noise (S/N) ratio
   → **Shannon's theorem:** every link has a maximal transmission rate
   $\tilde{B}$ = max n° of bits/sec$= B\log_2(1+\mathrm{S/N})$
   $\tilde{B}$ measured in bits per second (bps). Also called "Bandwidth" in computer science
   → Remark: if S/N is not constant, $\tilde{B}$ changes as well!

### Example - Telephone line (ADSL)

Link bandwidth: 1 MHz, S/N: 10000⇒ max n° of bps
$= 10^6\log_2(1+10000) \simeq 13$ Mbps

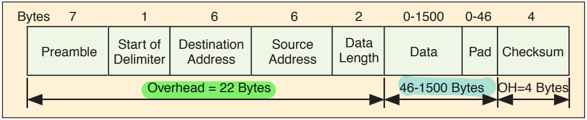
3. Latency (delay): propagation time [s] for 1 bit to travel along the link
   ⇒ usually proportional to the length of $\ell$

# Packet networks



- Data is transmitted in atomic units called *packets*[1]

## Ethernet packet



| Bytes | 7 | 1 | 6 | 6 | 2 | 0-1500 | 0-46 | 4 |
|---|---|---|---|---|---|---|---|---|
| | Preamble | Start of Delimiter | Destination Address | Source Address | Data Length | Data | Pad | Checksum |

[Lian et al., '01]

- Roughly, a packet is composed of a header and a data field
- Packets can have different sizes, depending on the data field
- Transmitting 1 bit of data or several bytes always costs 1 packet

---

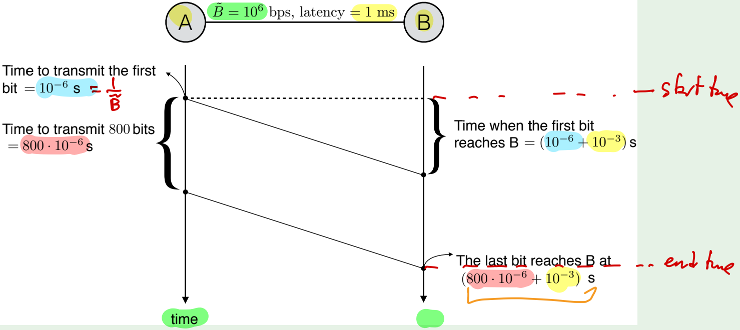
[1] At the link level, packets are more correctly called "frames"

# Delays in Control Networks

# Packet delay

The whole packet must be transmitted ⇒ additional delay source, on top of latency

## Sending a 800 bit packet from A to B



A

$\tilde{B} = 10^6$ bps, latency $= 1$ ms

B

Time to transmit the first bit $= 10^{-6}$ s $= \frac{1}{\tilde{B}}$

Time to transmit $800$ bits $= 800 \cdot 10^{-6}$ s

Time when the first bit reaches B $= (10^{-6} + 10^{-3})$ s

start time

The last bit reaches B at $(800 \cdot 10^{-6} + 10^{-3})$ s

end time

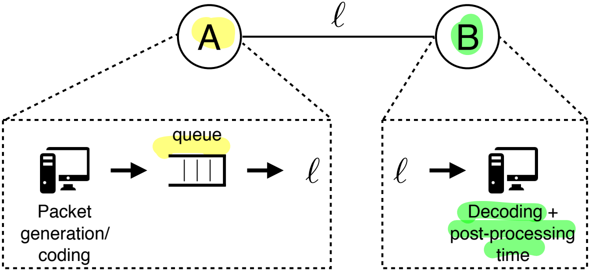
time

$$\text{Packet delay} = \frac{\text{Packet size}}{\tilde{B}} + \text{link latency}$$

Deterministic delay component if the S/N is constant (not true for wireless...)

# Other sources of delays

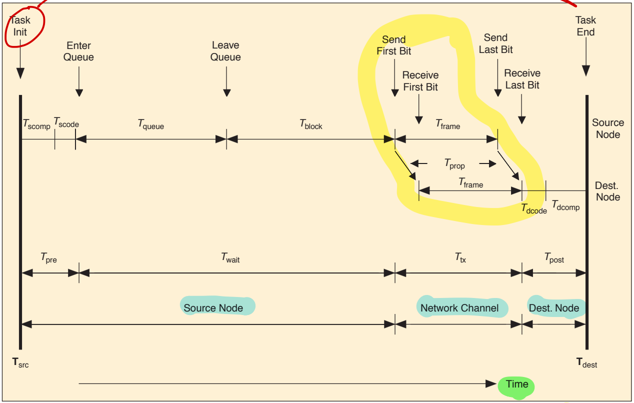
- Source nodes are equipped with queues needed for resolving conflicts
  - Delay due to queuing time = time a message waits in the queue while previous messages in the queue are sent
    - Depends on the network load and protocol (see next) → stochastic delay component
- Destination nodes need to decode and post-process packets before the data can be used → additional delay



# Other sources of delays

Summarizing

time restart



[Lian et al., '01]

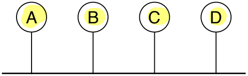
Several sources, three main categories (source node, network channel, destination node)

2h L2

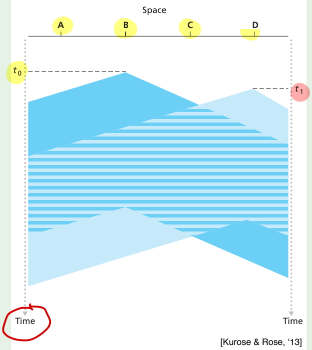
# Packet collisions and the MAC protocol

# Packet collision

- Premise: nodes can sense if the bus is free all the time
- If they follow the rule of transmitting only when the bus is free (Carrier Sense Multiple Access (CSMA) rule), why collisions happen ?



## Space-time diagram: B and D transmit



[Kurose & Rose, '13]

- At time $t_0$, B senses the bus is free and starts transmitting
- At time $t_1$, D senses the bus is free and starts transmitting
  → Collisions !
- The longer the bus, the higher the probability of collision

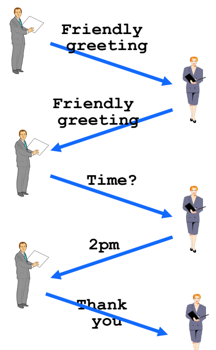
# Collision management

- Nodes can detect collision (sensed $\neq$ transmitted)
- Retransmit the packet ? Who retransmits ?

  $\Rightarrow$ Need of a Medium Access Control (MAC) protocol !

## MAC protocol

**What is a protocol?**



- Agreement between different devices about network access
- The MAC protocol influences a lot delays and packet losses (see next)

  $\Rightarrow$ it is a "non physical" source of packet loss and delays

# Next: compare 3 popular types of control networks

Bus topology: 3 different MAC protocols

- Ethernet with "Carrier Sense Multiple Access with Collision Detection (CSMA/CD)"
- Token-passing (e.g. ControlNet)
- Controller Area Network (CAN) (e.g. DeviceNet)

Bonus: wireless control networks

# Ethernet CSMA/CD (simplified description)

- When a node wants to transmit, it listens to the network (busy = wait)
- Two nodes transmit at the same time → messages collide and get corrupted
  ⇒...but nodes listen while transmitting and detect collision
- Collision detected: the transmission node stops, waits a random time and retransmits
  ⇒ after 16 collisions, the node drops the packet and tells it to the microprocessor (the "packet generator")

**Pros**

Simple MAC protocol → almost no delay at low network loads

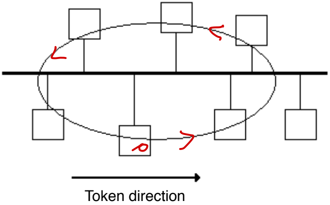
**Cons**

Nondeterministic protocol. At high network loads delays may be unbounded

# Token-passing bus (e.g. ControlNet)

Nodes arranged logically in a ring

- The node with the token transmits until
  - it has no more data or
  - the max time for holding a token is reached
- The token is passed to the successor



## Pros

- Data frames never collide
- Transmission delay bounded by the token rotation time !
- Easy to add nodes
- Excellent throughput at high network loads

## Cons

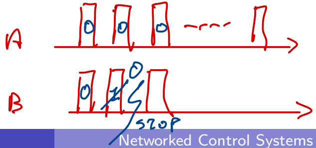
- Limited n° of nodes (1, . . . , 99) [needed for implementing implicit token passing through addresses] ⇒ each node must know which is the next one (unique MAC ID)
- Less efficient then CSMA/CD at low traffic, because token-passing introduces overhead

# CAN bus with CSMA/Arbitration Message Priority (e.g. DeviceNet)

- Each message has a priority, used to arbitrate access to the bus in case of simultaneous transmissions
- A node that wants to transmit waits until the bus is free. Then:
    - starts sending the message identifier (11 bits) bit-by-bit (a logic 0 is dominant on a logic 1)
    - All nodes have synchronized clocks for detecting the start of a bit-period

  ⇓

  In this phase, arbitration is performed and as soon as a node receives a bit different from the one it sent, it stops sending his message ⇒ An ongoing transmission is NEVER corrupted !



# CAN bus with CSMA/Arbitration Message Priority (e.g. DeviceNet)

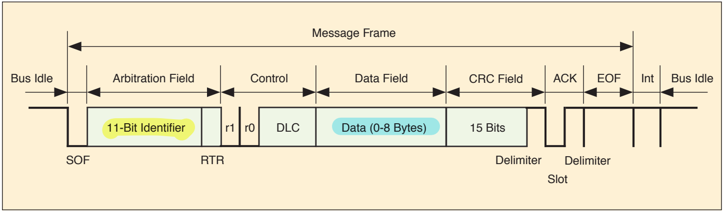
- Each message has a priority, used to arbitrate access to the bus in case of simultaneous transmissions
- A node that wants to transmit waits until the bus is free. Then:
  - starts sending the message identifier (11 bits) bit-by-bit (a logic 0 is dominant on a logic 1)
  - All nodes have synchronized clocks for detecting the start of a bit-period

    $\Downarrow$

    In this phase, arbitration is performed and as soon as a node receives a bit different from the one it sent, it stops sending his message ⇒ An ongoing transmission is NEVER corrupted !

The destination/source unit might not even be specified, but the message identifier is unique in the network. All units listen and discard messages they are not interested in. This is called *multicast*.

# CAN bus with CSMA/Arbitration Message Priority (e.g. DeviceNet)

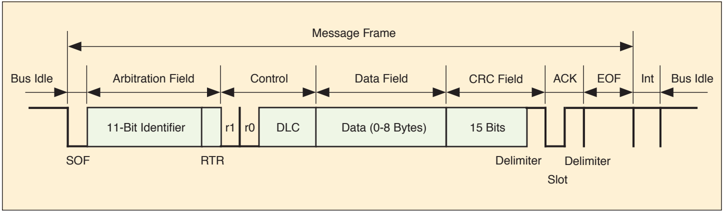


[Lian et al., '01]

## Pros

- Deterministic protocol, optimized for short messages
- Transmission of high-priority messages is guaranteed with a given maximal delay
- An ongoing transmission is never corrupted

# CAN bus with CSMA/Arbitration Message Priority (e.g. DeviceNet)



[Lian et al., '01]

## Cons

- Keeping precise clock synchronization requires
  - slow transmission rate (max 500 kb/s)
  - short cable length
- Variable delay for low-priority messages (that must be promoted to high-priority for increasing chances to be transmitted)

# Typical parameters of control networks

(token-passing) (CAN)

Bandwith →

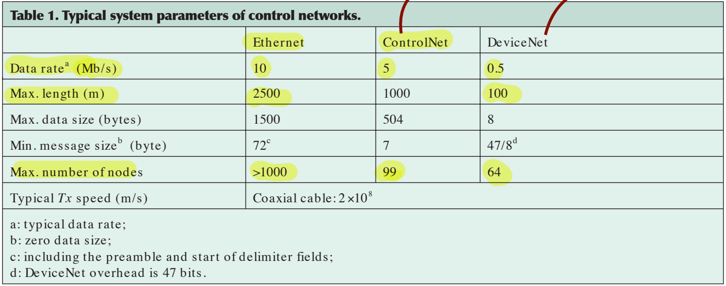


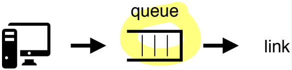
**Table 1. Typical system parameters of control networks.**

| | Ethernet | ControlNet | DeviceNet |
|---|---|---|---|
| Data rate[a] (Mb/s) | 10 | 5 | 0.5 |
| Max. length (m) | 2500 | 1000 | 100 |
| Max. data size (bytes) | 1500 | 504 | 8 |
| Min. message size[b] (byte) | 72[c] | 7 | 47/8[d] |
| Max. number of nodes | >1000 | 99 | 64 |
| Typical *Tx* speed (m/s) | Coaxial cable: $2\times10^8$ | | |

a: typical data rate;
b: zero data size;
c: including the preamble and start of delimiter fields;
d: DeviceNet overhead is 47 bits.

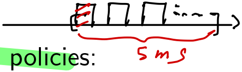
## General remark

Retransmission, clock synchronization and token passing require to implement a queue at the source node, in order to decouple transmission from the functioning of the microprocessor



# Case study on network-induced delays: 10 nodes network

- Each node uses a sampling time (aka "message period") of 5000 $\mu$s (chosen so that network is not saturated)



- Each node sends 8 bytes in every period. Three release policies:
  1. "Zero": all nodes start transmitting at the beginning of the period
  2. "Random": the beginning of transmission is chosen randomly within each period
  3. "Scheduled": pre-specified beginning-of-transmission time for each node within each period

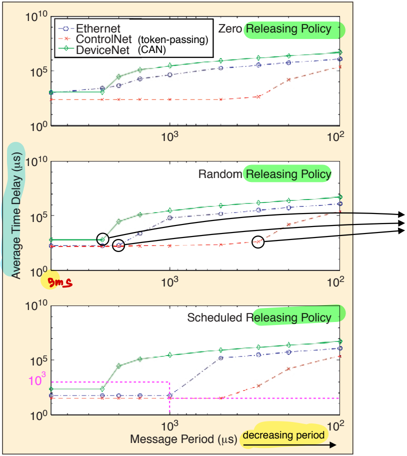
**Table 2. Simulation result of three releasing policies with message period of 5000 $\mu$s (ten-node case).**

| Releasing Policies | Zero | Random | Scheduled |
|---|---|---|---|
| **Average time delay ($\mu$s)** | | | |
| Ethernet | 1081 | 172 | 58 |
| ControlNet (token-passing) | 241 | 151 | 32 |
| DeviceNet (CAN) | 1221 | 620 | 222 |

## Main message

Delays also depend on how packets are released (on top of the sources of delays previously analyzed)

# Average delay as a function of the sampling time



Different saturation points due to different data rates-releasing policy

[Lian et al., '01]

- New experiments where the sampling time is varying (5000 $\mu s$ is the origin of the horizontal axis)
- Total delays from the packet generation to the packet post-processing
- Packets arrived after the end of the sampling interval are discarded, all networks suffer from packet drops (time-varying and random, as the delays)

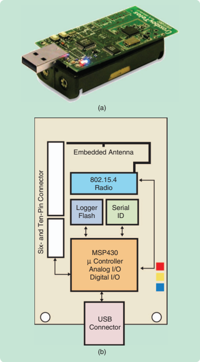
## Main message

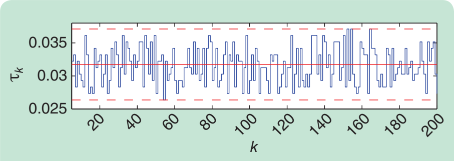
Delays also depend on the sampling time (on top of the sources of delays previously analyzed)

# Wireless control networks

[Bauer et al., '14]

- Experiment: output-feedback control of an inverse pendulum on a cart
- Sensors transmit position and angle to the controller
- Telos B motes communicating in the 2.4 GHz band implement the wireless link from the sensors to the controller
- MAC protocol: Token-passing-like ⇒ avoids packet losses if NO other device is using the 2.4 GHz band (e.g. Bluetooth, WiFi, etc.)





### Main message

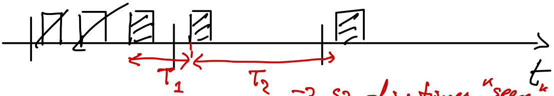
Delays also depend on other devices using the same band and vary in a stochastic fashion

# Time-varying sampling intervals in control networks

Why sampling intervals experienced by the controller might be time-varying?

- Retransmission after conflict detection causes fluctuations around a nominal duration of the sampling time
  - Packet dropouts are caused only by *multiple consecutive* conflicts
- Some MAC protocols can modify the sampling intervals for reducing the network load

sensor with multiple sampling within an interval



→ sampling times "seen" by the controller

# Take-home messages

- Control networks aim at supporting real-time operations (small and frequent packets)
- Delays are induced by
  - the physical layer
  - the MAC protocol

  ...and are time-varying, often stochastic
- Packet dropouts due to
  - collisions + no retransmission of old packets
- Sampling intervals can be time-varying

1h23