Center for Control, Dynamical-systems, and Computation University of California at Santa Barbara

# Delay Impulsive Systems: A Model For NCSs

Payam Naghshtabrizi
Joao Hespanha

44th Allerton Conference on Communication, Control, and Computing Sep. 27-29, 2006

## Motivation

Largest sampling interval that system remains stable?

Less comm. → more users

Important for high cost comm. e.g.
Wireless comm. → longer battery life

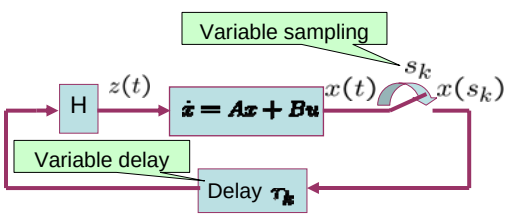
Example: $\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & -0.1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ 0.1\end{bmatrix}u, \qquad u(t_k) = \begin{bmatrix}3.75 & 11.5\end{bmatrix} x(t_k)$

| | Max. sampling interval | # of plants in CAN based NCS | Battery life |
|---|---|---|---|
| Walsh ACC 99 | $2.7 \times 10^{-4}$ | 0 | |
| Zhang Allerton 01 | 0.0593 | 12 | 1 month |
| Naghshtabrizi | 1.6 | 400 | 30 month |

# Outline

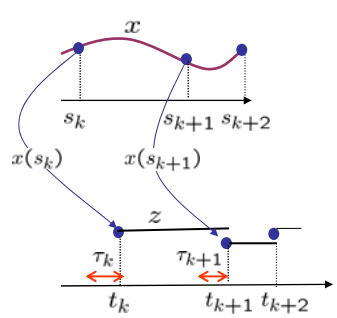
- Sampled-data systems (SDSs) with variable sampling & delay
- Different Network Control Systems (NCSs) can be presented by SDSs
- SDSs/NCSs as impulsive systems
- Stability of impulsive systems
- NCSs protocols
- Conclusions and future work

# SDSs with variable sampling & delay

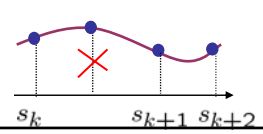



$$u(t) = z(t) = x(s_k), \quad t_k \le t < t_{k+1},$$
$$z(t_{k+1}^+) = x(s_{k+1}), \quad k \in \mathbb{N}$$

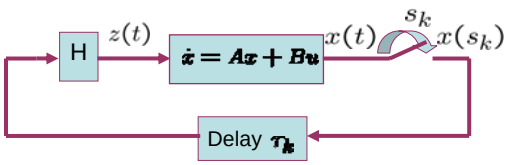
*K-th*
sampling time $s_k$
update time $t_k := s_k + \tau_k$



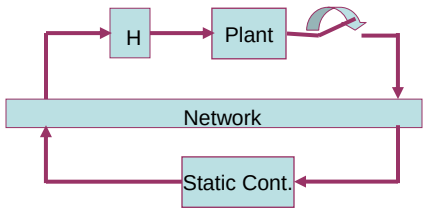

Missing samples:
If we only index samples that get to destination model can capture missing samples.



## NCSs (Network Control Systems) v.s. SDSs (Sampled-Data Systems)




Network: Variable sampling, delays, packet dropouts.



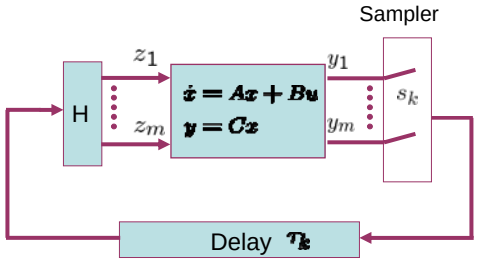

Plant: $\dot{x}_p = A_p x_p + B_p u_p$

Cont.: $y_c = K x_p$

$A := A_p, \quad B := B_p K$

All states are measurable.
Measurements/control command can be sent in a single packet. → No distributed sensors and actuators.

## Multi-Input, Multi-Output (MIMO) SDSs



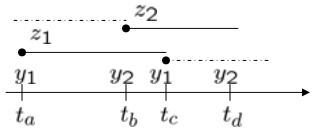

*K-th*
sampling time $s_k$
update time $t_k := s_k + \tau_k$

$$z := \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} \qquad y := \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

$$\mathcal{K}_i := \{k : y_i(t) \text{ is sampled at } s_k\}, \quad 1 \le i \le m$$

$$k, j : \text{consecutive sampling time indexes in } \mathcal{K}_i, \ 1 \le i \le m,$$

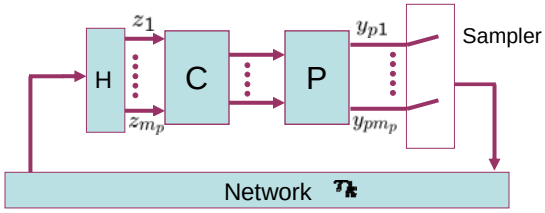
$$z_i(t) := y_i(s_k), \quad t_k \le t < t_j,$$




$a, c \in \mathcal{K}_1$
$b, d \in \mathcal{K}_2$

## NCSs configurations modeled by MIMO SDSs

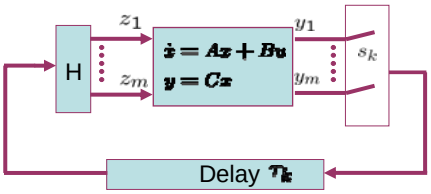
1. One-channel NCS with dynamic feedback controller




Plant: $\begin{cases} \dot{x}_p = A_p x_p + B_p u_p \\ y_p = C_p x_p \end{cases}$

Cont: $\begin{cases} \dot{x}_c = A_c x_c + B_c u_c \\ y_c = C_c x_c + D_c u_c \end{cases}$

*K-th*
sampling time $s_k$
update time $t_k := s_k + \tau_k$



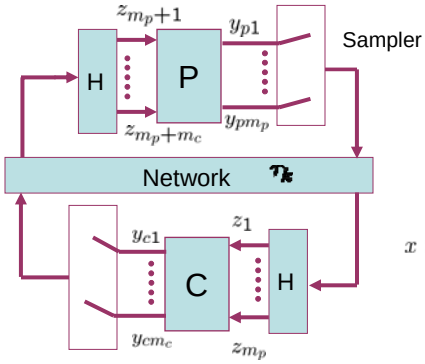

$y_i(t) := y_{pi}(t), \quad x := \begin{bmatrix} x_p \\ x_c \end{bmatrix},$

$A := \begin{bmatrix} A_p & B_p C_c \\ 0 & A_c \end{bmatrix}, \quad B := \begin{bmatrix} B_p D_c \\ B_c \end{bmatrix},$

$C := \begin{bmatrix} C_p & 0 \end{bmatrix}.$

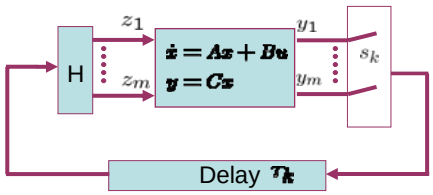
## NCSs configurations Modeled by MIMO SDSs

2. Two-channel NCS with dynamic feedback controller (assume $D_c = 0$ )




*K-th*
sampling time $s_k$
update time $t_k := s_k + \tau_k$

$x := \begin{bmatrix} x_p \\ x_c \end{bmatrix}, \quad y_i := \begin{cases} y_{pi}, & 1 \le i \le m_p \\ y_{ci}, & m_p + 1 \le i \le m_p + m_c \end{cases}$



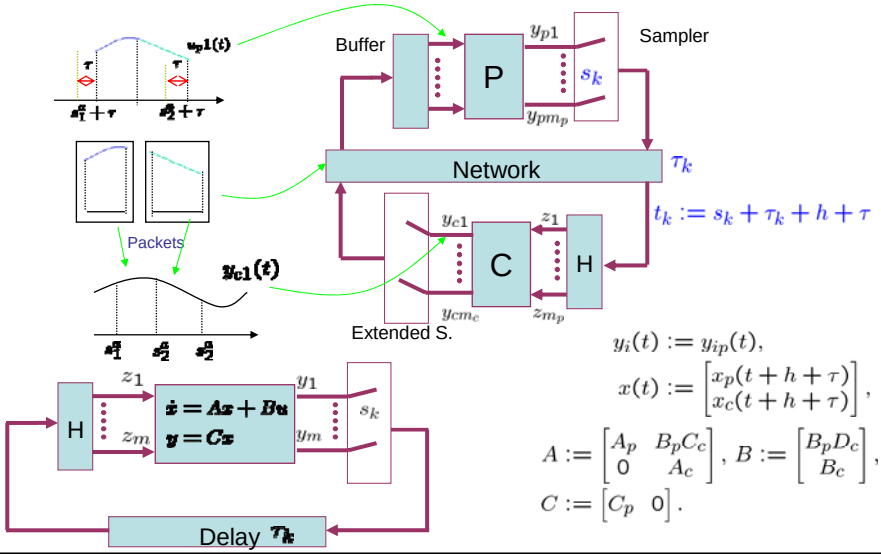

$A := \begin{bmatrix} A_p & 0 \\ 0 & A_c \end{bmatrix}, \quad B := \begin{bmatrix} 0 & B_p \\ B_c & 0 \end{bmatrix},$

$C := \begin{bmatrix} C_p & 0 \\ 0 & C_c \end{bmatrix}.$

# NCSs configurations Modeled by MIMO SDSs

3. Two-channel NCS with anticipative feedback controller

For simplicity, sampling intervals and delays are constant in control channel equal to $h, \tau$




$$t_k := s_k + \tau_k + h + \tau$$

$$y_i(t) := y_{ip}(t),$$

$$x(t) := \begin{bmatrix} x_p(t+h+\tau) \\ x_c(t+h+\tau) \end{bmatrix},$$

$$A := \begin{bmatrix} A_p & B_pC_c \\ 0 & A_c \end{bmatrix}, \; B := \begin{bmatrix} B_pD_c \\ B_c \end{bmatrix},$$

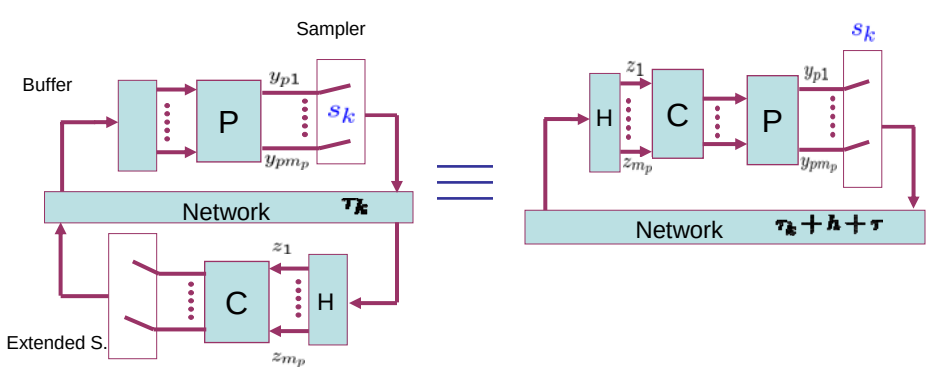
$$C := \begin{bmatrix} C_p & 0 \end{bmatrix}.$$

# Two-channel and One-channel NCSs

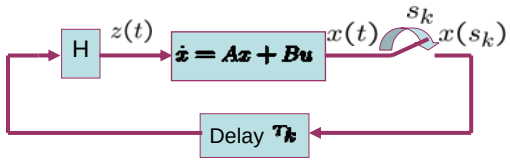
For analysis purposes:

Two-ch NCS with anticipative controller

One-ch NCS with dynamic feedback



## SDSs (with delay) as infinite-dim. impulsive systems




*K-th*
sampling time $s_k$
update time $t_k := s_k + \tau_k$

$$u(t) = z(t) = x(s_k), \quad t_k \le t \le t_{k+1},$$

$$z(t_{k+1}^+) = x(s_{k+1}), \quad k \in \mathbb{N}$$

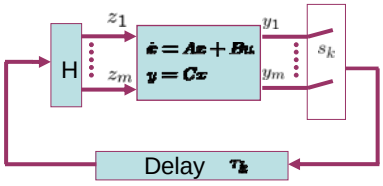
$$\xi := \begin{bmatrix} x \\ z \end{bmatrix}, \quad F := \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix},$$

Flow: $\dot{\xi}(t) = F\xi(t), \qquad t_k \le t < t_{k+1},$

Jumps or impulses: $\xi(t_{k+1}^+) = \begin{bmatrix} x(t_{k+1}) \\ x(t_{k+1} - \tau_{k+1}) \end{bmatrix}, \quad k \in \mathbb{N}$

$s_{k+1}$

## MIMO case




$$u := \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} \quad y := \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

$\mathcal{K}_i := \{k : y_i(t) \text{ is sampled at } s_k\}, \quad 1 \le i \le m$

$k, j$ : consecutive sampling time indexes in $\mathcal{K}_i$,

$$z(t) := \begin{bmatrix} z_1(t) \\ \vdots \\ z_m(t) \end{bmatrix}$$

$$z_i(t) := y_i(s_k), \quad t_k \le t < t_j,$$

$$\xi := \begin{bmatrix} x \\ z \end{bmatrix}, \quad F := \left[\begin{array}{c:c} A & B \\ \hdashline 0 & 0 \end{array}\right]$$

Flow: $\dot{\xi}(t) = F\xi(t), \qquad t_k \le t < t_{k+1},$

Jumps or impulses: $\xi(t_{k+1}^+) = \begin{bmatrix} x(t_{k+1}) \\ \hdashline y_1(t_{k+1}) \\ \vdots \\ z_i(t_{k+1} - \tau_{k+1}) \\ \vdots \\ z_m(t_{k+1}) \end{bmatrix}, \qquad k+1 \in \mathcal{K}_i, \; 1 \le i \le m,$

## Stability of (finite dimensional) impulsive systems

Consider impulsive system (finite dimensional)

$$\dot{x} = f_k(x,t), \qquad s_k \le t < s_{k+1},$$
$$x(s_{k+1}^+) = g_k(s_{t_{k+1}}) \qquad k \in \mathbb{N}.$$

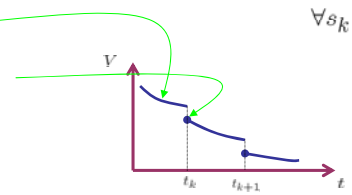
and a class $\mathcal{S}$ of impulse sequences. System is GUES over the class $\mathcal{S}$ if for every impulse sequence in $\mathcal{S}$, $\exists V : \mathbb{R}^n \times \mathbb{R}^+ \to \mathbb{R}^+$ s.t.

(a) $c_1|x|^b \le V(x,t) \le c_2|x|^b \qquad \forall x \in \mathbb{R}^n$

(b) $\dfrac{dV(x,t)}{dt} \le -c_3|x(t)|^b, \qquad \forall s_k \le t < s_{k+1},\ k \in \mathbb{N}$

(c) $V(x,t_k) \le \lim_{t \uparrow t_k} V(x,t), \qquad k \in \mathbb{N}$

for $c_1, c_2, c_3, b > 0$.




Adopted from Decarlo-Branicky ITAC 00, Liberzon (book) 03

## Stability of infinite-dimensional impulsive systems

Consider delay impulsive system

$$\dot{x} = f(x,t), \qquad t_k \le t < t_{k+1},$$
$$x(t_{k+1}^+) = g(x(t_{k+1} - \tau_k)) \qquad k \in \mathbb{N}.$$

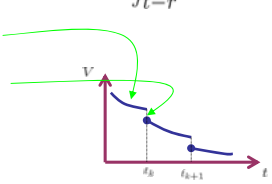
and a class $\mathcal{S}$ of impulse-delay sequences. System is GUES over the class $\mathcal{S}$ if for every sequence in $\mathcal{S}$, $\exists V : C\big([-r,0],\mathbb{R}^n\big) \times \mathbb{R}^+ \to \mathbb{R}^+$ s.t.

(a) $d_1|\phi(0)|^b \le V(\phi,t) \le d_2|\phi(0)|^b + \bar{d}_2 \int_{t-r}^{t} |\phi(s)|^b ds \quad \forall \phi \in C\big([-r,0]\big), t \in \mathbb{R}^+$

(b) $\dfrac{dV(x_t,t)}{dt} \le -d_3|x(t)|^b, \qquad \forall t_k \le t < t_{k+1},\ k \in \mathbb{N}$

(c) $V(x_{t_k},t_k) \le \lim_{t \uparrow t_k} V(x_t,t), \qquad k \in \mathbb{N}$

for $d_1, d_2, \bar{d}_2, d_3, b > 0$,

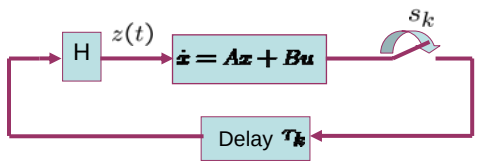



- Extended version of L-K Theorem for infinite dimensional (delay) systems with jumps.
- Results by *Liu ITAC 01*, *Sun-Michel ITAC 05*, didn't lead to LMI cond. for linear case.

# Stability of NCSs with delay

Assume that

$$s_{k+1} + \tau_{k+1} - s_k \le \tau_{MATI},$$
$$\tau_{min} \le \tau_k \le \tau_{max}, \forall k \in \mathbb{N},$$




The system is exponentially stable if $\exists P, X, Z, R_i > 0, N_i, 1 \le i \le 4$ s.t

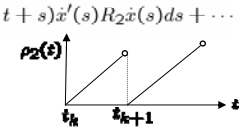
$$\begin{bmatrix} M_1 + \beta_{\max}(M_2 + M_3) & \tau_{\max} N_1 & \tau_{\min} N_3 \\ * & -\tau_{\max} R_1 & 0 \\ * & * & -\tau_{\min} R_3 \end{bmatrix} < 0,$$

$$\begin{bmatrix} M_1 + \beta_{\max} M_2 & \tau_{\max} N_1 & \tau_{\min} N_3 & \beta_{\max}(N_1 + N_2) & \beta_{\max} N_4 \\ * & -\tau_{\max} R_1 & 0 & 0 & 0 \\ * & * & -\tau_{\min} R_3 & 0 & 0 \\ * & * & * & -\beta_{\max}(R_1 + R_2) & 0 \\ * & * & * & * & -\beta_{\max} R_4 \end{bmatrix} < 0,$$

where $\beta_{max} = \tau_{MATI} - \tau_{\min}$, $M_1 := \cdots$

$$V := x'Px + \int_{t-\rho_1}^{t} (\rho_{1\max} - t + s)\dot{x}'(s)R_1\dot{x}(s)ds + \int_{t-\rho_2}^{t} (\rho_{2\max} - t + s)\dot{x}'(s)R_2\dot{x}(s)ds + \cdots$$

$$\rho_1(t) := t - s_k,\ \rho_2(t) := t - t_k \quad t_k \le t < t_{k+1}$$
$$\rho_{1\max} := \sup_{t \ge 0} \rho_1(t), \rho_{2\max} := \sup_{t \ge 0} \rho_2(t), \ldots$$




# Benchmark problem

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.1 \end{bmatrix} u, \qquad u(t_k) = \begin{bmatrix} 3.75 & 11.5 \end{bmatrix} x(t_k)$$

Variable sampling:

$$s_{k+1} - s_k \le \tau_{MATI}, \forall k \in \mathbb{N}$$

| | |
|---|---|
| Fridman Auto 04, Yue ITAC 04 | 0.8696 |
| Yue Automatica 05 | 0.8871 |
| This approach | 1.1137 |

Variable sampling+delay:

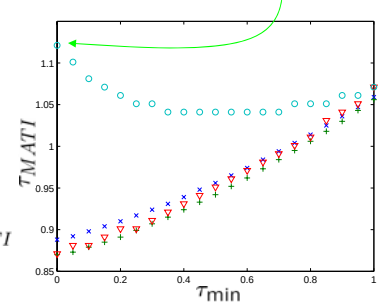
$$s_{k+1} + \tau_{k+1} - s_k \le \tau_{MATI},$$
$$\tau_{min} \le \tau_k \le \tau_{max}, \forall k \in \mathbb{N},$$

○ This approach constant delay $\tau_{\max} = \tau_{\min}$

▽ This approach largest delay upper bound $\tau_{\max} = \tau_{MATI}$

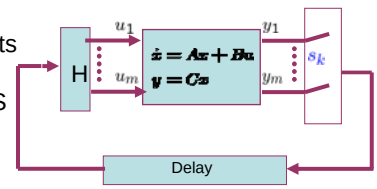
× Yue et al. Automatica 05

\+ Naghshtabrizi et al. CDC 05



## Stability of NCSs, distributed sensors/actuators ($\tau_{MATI}$ VS $\rho_{max}^i$)

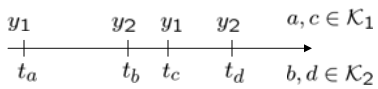
- Based on previous slide we can extend the results to MIMO case.
- For simplicity no delay. By solving stability LMIS one gets constants $\rho_{max}^1, \cdots, \rho_{max}^m$.




$\forall i$ Interval between consecutive sampling of $y_i$ $\leq \rho_{max}^i$ ⟹ Exp. Stability

- Previous results:

interval between any consecutive samplings $\leq \tau_{MATI}$ ⟹ Exp. Stability




$\tau_{MATI}$ : $t_b - t_a, t_c - t_b, t_d - t_c, \cdots \leq \tau_{MATI}$

$\rho$ : $t_c - t_a < \rho_{\max}^1, t_d - t_b < \rho_{\max}^2,$

## Benchmark problem: batch reactor

Linearized model of a batch reactor controlled by a PI controller through one-ch. NCS

$$\begin{cases} \dot{x}_p = A_p x_p + B_p u \\ y_p = C_p x_p \end{cases}, A_p := \begin{bmatrix} 1.38 & -0.2077 & 6.715 & -5.676 \\ -0.5814 & -4.29 & 0 & 0.675 \\ 1.067 & 4.273 & -6.654 & 5.893 \\ 0.048 & 4.273 & 1.343 & -2.104 \end{bmatrix}, B_p := \begin{bmatrix} 0 & 0 \\ 5.679 & 0 \\ 1.136 & -3.146 \\ 1.136 & 0 \end{bmatrix}, C_p := \begin{bmatrix} 1 & 0 & 1 & -1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\begin{cases} \dot{x}_c = A_c x_c + B_c u_c \\ y_c = C_c x_c + D_c u_c \end{cases}, A_c := \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad B_c := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad C_c := \begin{bmatrix} -2 & 0 \\ 0 & 8 \end{bmatrix}, \quad D_C := \begin{bmatrix} 0 & -2 \\ 5 & 0 \end{bmatrix}.$$

| No delay, Policy: output1, output2 periodically | $\tau_{MATI}$ |
|---|---|
| Walsh et al. ITAC 02 (deterministic) | $10^{-5}$ |
| Nesic et al. ITAC 04 (deterministic) | 0.0082 |
| Tabbara et al. CDC05 (deterministic) | 0.0123 |
| Hespanha et al. MTNS 06 (stochastic arbitrary dist. ) | 0.0279 |
| Naghshtabrizi et al. | 0.0405 |

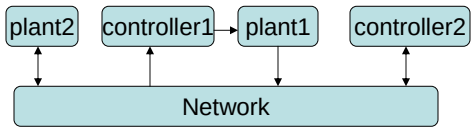
$\rho_{\max}^1 = 0.081,$
$\rho_{\max}^2 = 0.113$

$$\tau_{MATI} = \frac{1}{2} \min\left(\rho_{\max}^1, \rho_{\max}^2\right)$$

Max of delay *0.05*, $\rho_{\max}^1 = 0.047, \rho_{\max}^2 = 0.076$

# Round-Robin (RR) protocol

- Protocol determines how the access to network is granted.



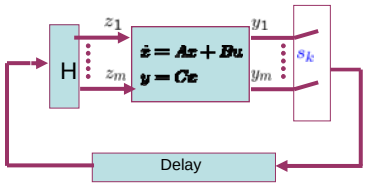

E.g. plant 2, controller 1, controller 2, plant 1, …….

- RR is a static protocol, i.e., assigns access to network in a predetermined and in a cyclic manner.
- Simple, implementable by Token passing based network
- Sufficient condition for stability can be found in Literature. ($\tau_{MATI}$ analysis)
- Our analysis also provides a sufficient condition for stability. ( $\rho_{\max}$ analysis)
- Not robust

Ref: Lian-Talibury IEEE Cont. Sys. Magazine, Walsh ITAC 02, Nesic-Teel ITAC 04 ……..

# TOD (try once discard) protocol

- TOD is a dynamic protocol, i.e, assigns access to network based on the *current error* in the network.




$$e := \begin{bmatrix} y_1 - z_1 \\ \vdots \\ \vdots \\ y_m - z_m \end{bmatrix}$$

The node i∈{1,….m} with the largest error will be granted the access to network.

- TOD is efficient (based on the current situation of network).
- TOD is robust.
- Sufficient condition for stability can be found in Literature. ($\tau_{MATI}$ analysis)
- TOD is not distributed: there is a centeral entity that has access to & compares all errors. (relaxed in Tabbara et al. cdc06 by introducing hybrid TOD)

Ref: Walsh ITAC 02, Nesic-Teel ITAC 04 ……..

## Priority based protocol

- Each node has a sending priority based on a monotonically decreasing function of $\rho^i_{\max} - (t - T_i)$

$T_i$ : Last time that node i send a packet
$t$ : Current time
$\rho^i_{max}$ : Deadline

- Inspired by Earliest Deadline First algorithm (Liu and Layland 73):

Given $\tau^i_{max}$ and $\rho^i_{max}$ such that the stability LMIs are feasible and

$$\sum_{i=1}^{m} \frac{\tau^i_{\max}}{\rho^i_{\max}} \leq 1$$

Then the algorithm is able to generate sampling sequence $\{s_k\}$ for which system is exponentially stable.

- Robust
- Distributed/scalable
- Implementable on CAN based networks
  - CAN is designed for short (8 byte), time critical messages.
  - 11 bit identifier (version 2.0A) is used to prioritization

## Conclusions and future work

Conclusions:

- Sampled-data systems (SDSs) with variable sampling and delay
- We show different Network Control Systems can be presented by SDSs
- Stability of SDSs/impulsive systems
- We introduce priority based protocol

Future work

- Sensor failure
- Ethernet or wireless networks, higher probability of transmission is assigned if deadline is close.
- Controller design.