# Industrial Automation

## (Automação de Processos Industriais)

## Supervised Control of Discrete Event Systems

## *Supervision Controllers (Part 2/2)*

http://users.isr.ist.utl.pt/~jag/courses/api19b/api1920.html

Prof. Paulo Jorge Oliveira, original slides
Prof. José Gaspar, rev. 2019/2020

# Some pointers on Supervised Control of DES

Analysers & simulators

http://www.nd.edu/~isis/techreports/isis-2002-003.pdf (Users Manual)
http://www.nd.edu/~isis/techreports/spnbox/ (Software)

Bibliography:

**Supervisory Control of Discrete Event Systems using Petri Nets**, J. Moody J. and P. Antsaklis, Kluwer Academic Publishers, 1998.

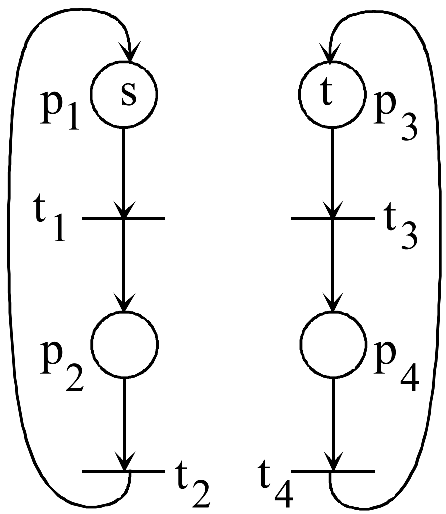
**Supervised Control of Concurrent Systems: A Petri Net Structural Approach**, M. Iordache and P. Antsaklis, Birkhauser 2006.

**Discrete Event Systems - Modeling and Performance Analysis**, Christos G. Cassandras, Aksen Associates, 1993.

**Feedback Control of Petri Nets Based on Place Invariants**, K. Yamalidou, J. Moody, M. Lemmon and P. Antsaklis, http://www.nd.edu/~lemmon/isis-94-002.pdf

# Methods of Synthesis

## Example of controller synthesis: s Producers / t Consumers




Incidence matrix

$$D_P = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Initial marking

$$\mu_{P_0} = \begin{bmatrix} s \\ 0 \\ t \\ 0 \end{bmatrix}.$$

Let p2= #machines working, t2= product produced
p3= #consumers, t3= request to consume (e.g. transport product)

Q: How to write *consume only when produced* ? What is the linear constraint?

Not possible to write it as a linear constraint on places $L\mu_p \leq b$ .
Is it impossible to solve this problem with the supervised control ?

# Methods of Synthesis

## Generalized linear constraint

Let the generalized linear constraint be

$$L\mu_P + Fq_P + Cv_P \leq b,$$
$$\mu_P \in N_0{}^n, v_P \in N_0{}^m, q_P \in N_0{}^m,$$
$$L \in Z^{n_C \times n}, F \in Z^{n_C \times m}, C \in Z^{n_C \times m}, e \quad b \in Z^{n_C},$$

where

* $\mu_P$ is the marking vector for system P;
* $q_P$ is the firing vector since $t_0$;
* $v_P$ is the number of transitions (firing) that can occur, also designated as Parikh vector.

# Methods of Synthesis

*Function LINENF of SPNBOX*

**Theorem*: Synthesis of Controllers based on Place Invariants, for Generalized Linear Constraints**

Given the generalized linear constraint $L\mu_P + Fq_P + Cv_P \leq b,$

if $b - L\mu_{P_0} \geq 0,$ then the controller with incidence matrix and initial marking, respectively

$$D_C^- = \max\left(0,\, LD_P + C,\, F\right)$$
$$D_C^+ = \max\left(0,\, F - \max\left(0, LD_P + C\right)\right) - \min\left(0,\, LD_P + C\right),$$

$$\mu_{C_0} = b - L\mu_{P_0} - Cv_{P0},$$

guarantees that constraints are verified for the states resulting from the initial marking.

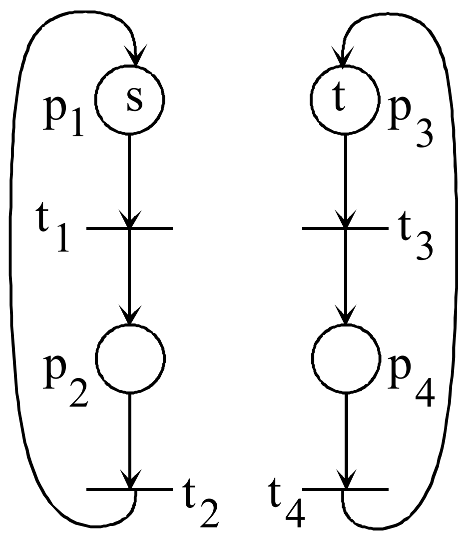
* *In the next slides this will be called the* ***LINENF theorem.***

# Methods of Synthesis

## Example of controller synthesis

$$\forall s \in N_0, \forall t \in N_0, \forall n \in N_0$$

Producer / Consumer




Linear constraint: $v_3 \leq v_2$

that can be written as:

$$Cv_P \leq b$$

$$L = 0, F = 0$$

$$\begin{bmatrix} 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \leq 0.$$

Incidence matrix

$$D_P = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Initial marking

$$\mu_{P_0} = \begin{bmatrix} s \\ 0 \\ t \\ 0 \end{bmatrix}.$$

# Methods of Synthesis

## Example of controller synthesis

Producer / Consumer

1) Test $b - L\mu_{P_0} = 0 - 0 \geq 0.$

**OK.**

2) Compute

$$D_C^- = \max\left(0,\, LD_P + C,\, F\right)$$
$$D_C^+ = \max\left(0,\, F - \max\left(0, LD_P + C\right)\right) - \min\left(0,\, LD_P + C\right),$$

$$D_C^- = \max\left(0,\ [0\ \ -1\ \ 1\ \ 0],\ \ 0\right) = [0\ \ 0\ \ 1\ \ 0]$$
$$D_C^+ = \max\left(0, -[0\ \ 0\ \ 1\ \ 0]\right) - \min\left(0, [0\ \ -1\ \ 1\ \ 0]\right)$$
$$= [0\ \ 0\ \ 0\ \ 0] - [0\ \ -1\ \ 0\ \ 0] = [0\ \ 1\ \ 0\ \ 0]$$

and

$$\mu_{C_0} = b - L\mu_{P_0} - Cv_{P0},$$
$$\mu_{C_0} = b - L\mu_{P_0} = 0 - 0 = 0.$$

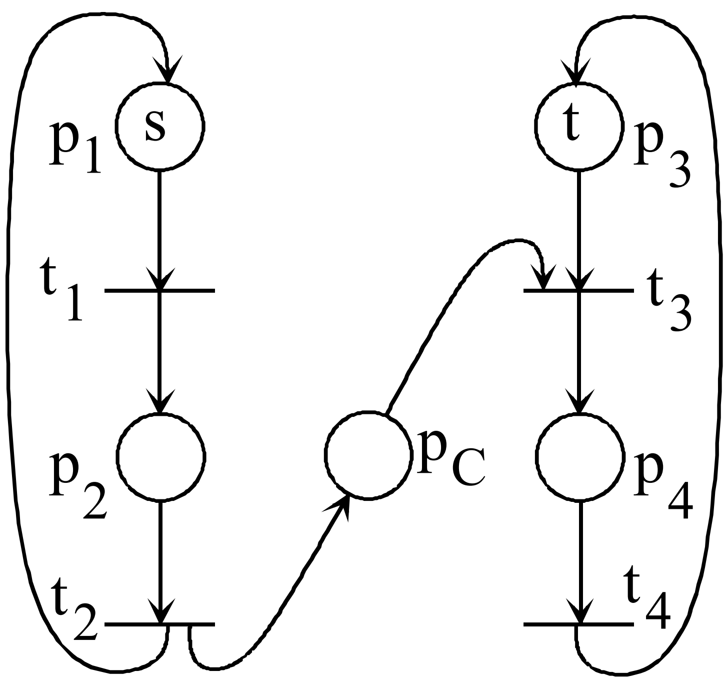
**OK.**

# Methods of Synthesis

## Example of controller synthesis

Producer / Consumer

3) Resulting in




$$D = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 1 & -1 & 0 \end{bmatrix}$$

$$\mu_0 = \begin{bmatrix} s \\ 0 \\ t \\ 0 \\ 0 \end{bmatrix}$$

**OK.**
**UAU!!!.**

# Methods of Synthesis

## Example of controller synthesis: Producer Consumer

```
% The Petri net D=Dp-Dm, and m0
%  (Dplus-Dminus= Post-Pre)

Dm= [1 0 0 0;
     0 1 0 0;
     0 0 1 0;
     0 0 0 1];

 Dp= [0 1 0 0;
     1 0 0 0;
     0 0 0 1;
     0 0 1 0];

m0= [1 0 1 0]';

% Supervisor constraint
%
L= []; F= [];  C= [0 -1 1 0];
b= 0;

% Computing the supervisor
%
[Dfm, Dfp, ms0] = linenf(Dm, Dp, L, b, m0, F, C)
Df= Dfp-Dfm
ms0
```

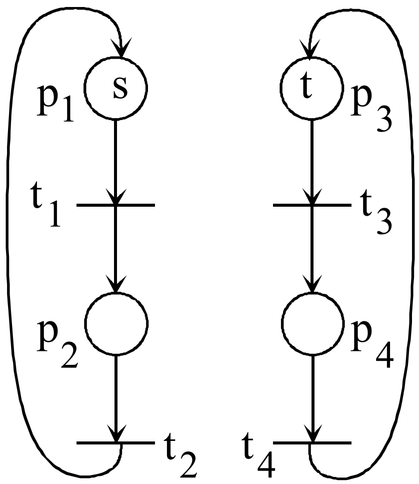
Result using the function LINENF.m of the toolbox SPNBOX:

```
Df =

    -1     1     0     0
     1    -1     0     0
     0     0    -1     1
     0     0     1    -1
     0     1    -1     0


ms0 =

     1
     0
     1
     0
     0
```

# Methods of Synthesis

## Example of controller synthesis

Bounded Producer / Consumer




Incidence matrix

$$D_P = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Initial marking

$$\mu_{P_0} = \begin{bmatrix} s \\ 0 \\ t \\ 0 \end{bmatrix}.$$

TWO linear constraints:

$$\begin{cases} v_3 \le v_2 \\ v_2 \le v_3 + n \end{cases} \Leftrightarrow \begin{cases} v_3 - v_2 \le 0 \\ v_2 - v_3 \le n \end{cases}$$

$$\forall s \in N_0, \forall t \in N_0, \forall n \in N_0$$

The two linear constraints can be written as:

$$Cv_P \le b \quad i.e.\ L = 0, F = 0 \quad \Leftrightarrow \begin{bmatrix} 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \le \begin{bmatrix} 0 \\ n \end{bmatrix}$$

# Methods of Synthesis

## Example of controller synthesis

### Bounded Producer / Consumer

1) Test

$$b - L\mu_{P_0} = b = \begin{bmatrix} 0 \\ n \end{bmatrix} \geq 0.$$

**OK.**

2) Compute

$$D_C^- = \max\left(0, \begin{bmatrix} 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \end{bmatrix}, 0\right) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

$$D_C^+ = \max\left(0, 0 - \max\left(0, \begin{bmatrix} 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \end{bmatrix}\right)\right) - \min\left(0, \begin{bmatrix} 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \end{bmatrix}\right)$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

and

$$\mu_{C_0} = b - L\mu_{P_0} = \begin{bmatrix} 0 \\ n \end{bmatrix}.$$

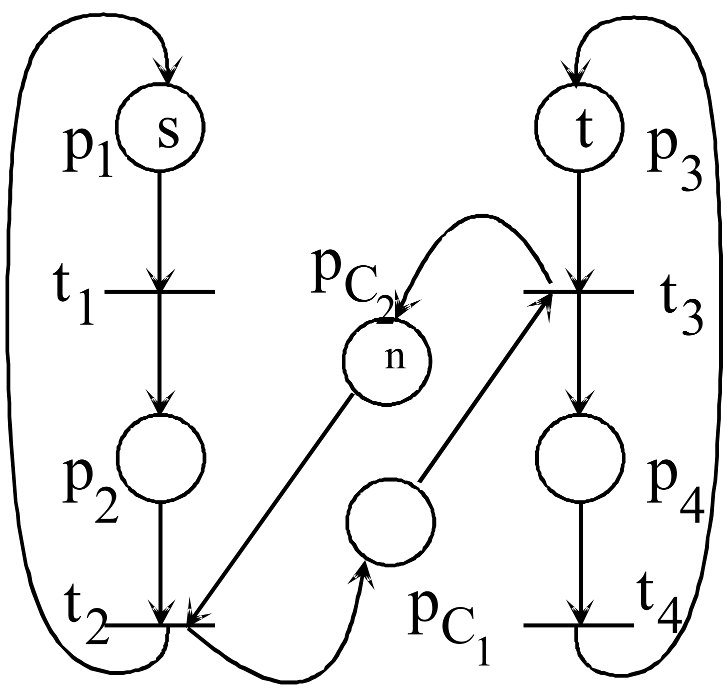
**OK.**

# Methods of Synthesis

## Example of controller synthesis

Bounded Producer / Consumer

3) Resulting in




$$D = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \end{bmatrix}$$

$$\mu_0 = \begin{bmatrix} s \\ 0 \\ t \\ 0 \\ 0 \\ n \end{bmatrix}$$

**OK.**
**UAU!!!.**

# Methods of Synthesis

*Function LINENF of SPNBOX*

## *LINENF Lemma 1:* From General Constraints to Theorem T1

Given the generalized linear constraint $L\mu_P + Fq_P + Cv_P \leq b$ and the conditions of the LINENF theorem:

If $\boxed{L \neq 0,}$ $F = 0,$ $C = 0$

then $D_C^+ = (LD_P)^-,$ $D_C^- = (LD_P)^+$ and $\boxed{D_C = -LD_P}$

$$\boxed{\mu_{C0} = b - L\mu_{P0}}$$

Notation:

$$D^+ = \max(0, D)$$
$$D^- = -\min(0, D)$$
$$D = D^+ - D^-$$

$D^+, D^- \in N_0^{n\times m}$ and $D \in Z_0^{n\times m}$

*(see proof in the next page)*

$$D_C^- = \max\left(0, LD_P + C, F\right)$$
$$D_C^+ = \max\left(0, F - \max\left(0, LD_P + C\right)\right) - \min\left(0, LD_P + C\right),$$

$$\mu_{C_0} = b - L\mu_{P_0} - Cv_{P0},$$

$$L \neq 0, \quad F = 0, \quad C = 0 \quad \Rightarrow L\mu_P \leq b$$

$$D_c^- = \max(0, LD_p + \underbrace{C}_{=0}, \underbrace{F}_{=0}) = \max(0, LD_p) = (LD_p)^+$$

$$D_c^+ = \max(0, \underbrace{F}_{=0} - \max(0, LD_p + \underbrace{C}_{=0})) - \min(0, LD_p + \underbrace{C}_{=0}) = \underbrace{\max(0, \underbrace{-(LD_p)^+}_{\leq 0})}_{=0} + (LD_p)^- = +(LD_p)^-$$

$D^- = -\min(0, D)$

$D^- \in \mathbb{N}_0^{m \times m}$

$$D_c = D_c^+ - D_c^- = (LD_p)^- - (LD_p)^+ = -\left((LD_p)^+ - (LD_p)^-\right) = -LD_p$$

$$\mu_{c_0} = b - L\mu_{p_0} - \underbrace{C}_{=0} v_{p_0} = b - L\mu_{p_0}$$

# Methods of Synthesis

*Function LINENF of SPNBOX*

## *LINENF Lemma 2:* Firing Regulation

Given the generalized linear constraint $L\mu_P + Fq_P + Cv_P \leq b$ and the conditions of the LINENF theorem:

If $L = 0$, $F \neq 0$, $C = 0$

then $D_C^+ = F^+$, $D_C^- = F^+$ and $D_C = 0$

$\mu_{C0} = b$

*(homework, prove this lemma)*

# Methods of Synthesis

*Function LINENF of SPNBOX*

## *LINENF Lemma 3:* Constraints on Counters

Given the generalized linear constraint $L\mu_P + Fq_P + Cv_P \leq b$ and the conditions of the LINENF theorem:

If $L = 0, \quad F = 0, \quad \boxed{C \neq 0}$

then $D_C^+ = C^-, \quad D_C^- = C^+$ and $\boxed{D_C = -C}$

$$\boxed{\mu_{C0} = b - Cv_{P0}}$$

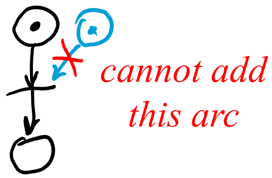
*(homework, prove this lemma)*

*(empty page)*

# Methods of Synthesis: adding **Uncontrollable** and **Unobservable** transitions

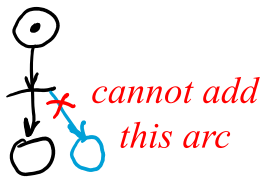
## Definition of **Uncontrollable Transition:**

A transition is uncontrollable if its firing **cannot be inhibited** by an external action (e.g. a supervisory controller).




## Definition of **Unobservable Transition:**

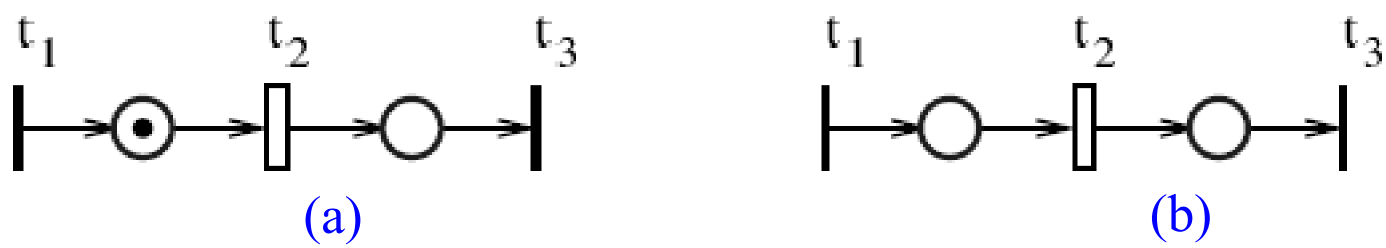
A transition is unobservable if its firing **cannot be detected or measured** (therefore the study of any supervisory controller can not depend from that firing).




**Proposition:**

In a Petri net based controller, both input and output arcs to/from plant transitions are used to trigger state changes in the controller. *Since a controller cannot have arcs connecting to unobservable transitions, then all* ***unobservable transitions are also implicitly uncontrollable****.*

# Methods of Synthesis: adding **Uncontrollable** and **Unobservable** transitions




If **t1 is controllable** and **t2 is uncontrollable**:

- case (a), then t2 cannot be directly inhibited; it will eventually fire
- case (b), then **t2 can be indirectly prevented** from firing by inhibiting t1.

*i.e. may exist indirect solution despite t2 being uncontrollable.*

If **t2 is unobservable** and **t3 is observable**, then we cannot detect when t2 fires. The state of a supervisor is not changed by firing t2. However we can **indirectly detect that t2 has fired**, by detecting the firing of t3.

*i.e. may exist indirect solution despite t2 being unobservable.*

∴ *may exist indirect solution despite t2 uncontrollable and/or unobservable.*

# Methods of Synthesis

**Definition:** A **marking** $\mu_P$ is **admissible** if

**i)** $L\mu_P \leq b$ and **ii)** $\forall \mu' \in R(C, \mu_P)$ verifies $L\mu' \leq b$

**Definition:** A **Linear Constraint** (***L***, ***b***) is **admissible** if

**i)** $L\mu_{Po} \leq b$ and

**ii)** $\forall \mu' \in R(C, \mu_{Po})$ such that $L\mu' \leq b$

$\mu'$ is an admissible marking.

Note: ii) indicates that the firing of uncontrollable transitions can never lead from a state that satisfies the constraint to a new state that does not satisfy the constraint.

# Methods of Synthesis

## Proposition: Admissibility of a constraint

A linear constraint is admissible ***iff***

- The initial markings satisfy the constraint.
- There **exists a controller** with maximal permissivity that forces the constraint and **does not inhibit any uncontrollable transition**.

## Two sufficient (not necessary) conditions:

**Corollary:** given a system with uncontrollable transitions,

$$\boxed{l^T D_{uc} \leq 0}$$ implies admissibility.

**Corollary:** given a system with unobservable transitions,

$$\boxed{l^T D_{uo} = 0}$$ implies admissibility.

# Methods of Synthesis

*Function MRO_ADM of SPNBOX*

## Lemma *: Structure of Constraint transformation

If $L'\mu_p \leq b'$ where

$L' = R_1 + R_2 L$ and $b' = R_2(b+1) - 1$

$R_1 \in Z^{nc \times n}$ and $R_1 \mu_p \geq 0$

$R_2 \in Z^{nc \times nc}$ is a matrix with positive elements in the diagonal

then $L\mu_p \leq b$ is also verified.

* Lemma 4.10 in [Moody98] pg46

# Methods of Synthesis

*Function MRO_ADM of SPNBOX*

## Lemma *: Structure of Constraint transformation

If $L'\mu_p \leq b'$ where

$L' = R_1 + R_2 L$ and $b' = R_2(b+1) - 1$

$R_1 \in Z^{nc \times n}$ and $R_1\mu_p \geq 0$

$R_2 \in Z^{nc \times nc}$ is a matrix with positive elements in the diagonal

then $L\mu_p \leq b$ is also verified.

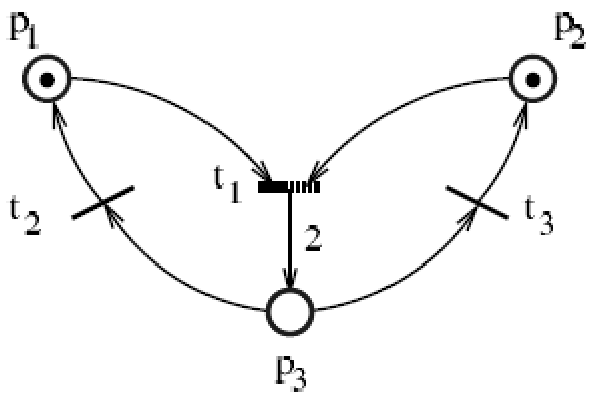
Typical usage:

- constraint L and b, and **extra constraints** → $R_1,\ R_2\ \rightarrow\ L',\ b'\ \rightarrow\ D_C^+,\ D_C^-,\ \mu_{C0}$
- see unobservable / uncontrollable extra constraints example in the next slides

* Lemma 4.10 in [Moody98] pg46

# Methods of Synthesis

## Example: design controller with t1 unobservable (1/4)




$$D = \begin{bmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \\ 2 & -1 & -1 \end{bmatrix}, \quad D_{uo} = \begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix}$$

Objectives: $\mu_1 + \mu_3 \geq 1$ and $\mu_2 + \mu_3 \geq 1$ which can be written in matrix form as

$$L\mu \leq b, \qquad L = \begin{bmatrix} -1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}, \qquad b = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

Example extracted from "Supervised Control of Concurrent Systems: A Petri Net Structural Approach", M. Iordache and P. Antsaklis, Birkhauser 2006.

# Methods of Synthesis

## Example: design controller with t1 unobservable (2/4)

```
% System and constraints

D= [-1  1  0;
    -1  0  1;
    +2 -1 -1];

Dm= -D.*(D<0);
Dp=  D.*(D>0);

m0= [1 1 0]';

L= [-1 0 -1; 0 -1 -1];
b= [-1; -1];


% Supervisor computation

[Dfp, Dfm, mf0] =
    linenf( Dp, Dm, L, b, m0 );
```

Dfp =

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 0 | 0 | 1 |
| 2 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |

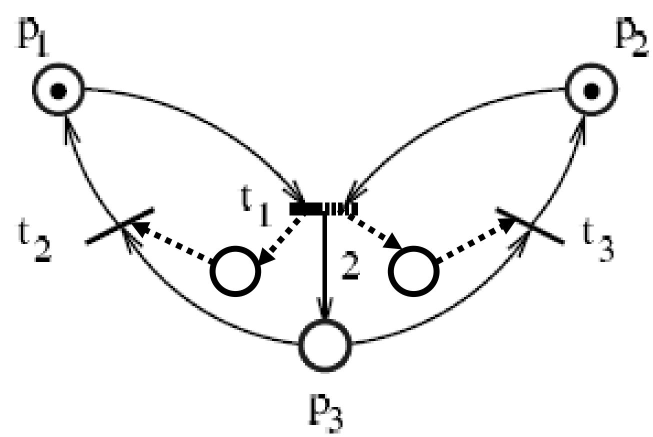
Dfm =

| | | |
|---|---|---|
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 1 | 1 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |

mf0 =

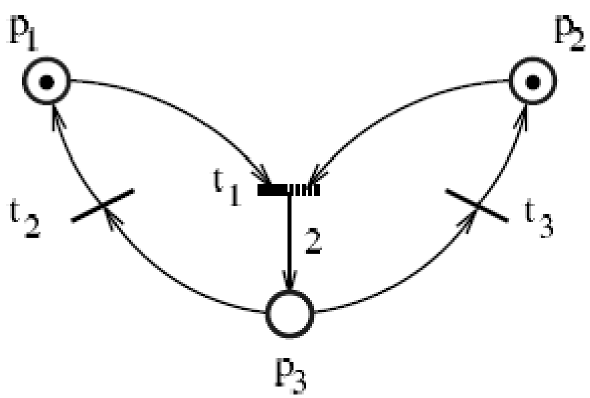
| |
|---|
| 1 |
| 1 |
| 0 |
| 0 |
| 0 |

*^ Bad news, supervisor touches $t_1$ .*



# Methods of Synthesis

## Example: design controller with t1 unobservable (3/4)




```
D= [-1  1  0;
    -1  0  1;
     2 -1 -1];

Tuo= [1]; Tuc= [];

L= [-1 0 -1; 0 -1 -1];
b= [-1 -1]';

[La, ba, R1, R2] = mro_adm( L, b, D, Tuc, Tuo );
```

Solution obtained with the function MRO_ADM.m of the SPNBOX toolbox:

$$R1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \qquad La = \begin{bmatrix} -2 & 0 & -1 \\ 0 & -2 & -1 \end{bmatrix} \qquad ba = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

$$R2 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Note: verify that $L_a \mu \leq b_a$ implies $L \mu \leq b$

# Methods of Synthesis

## Example: design controller with t1 unobservable (4/4)

Finally the supervised controller is simply obtained from $L_a$ and $b_a$:

$$D_c = -L_a D_p$$

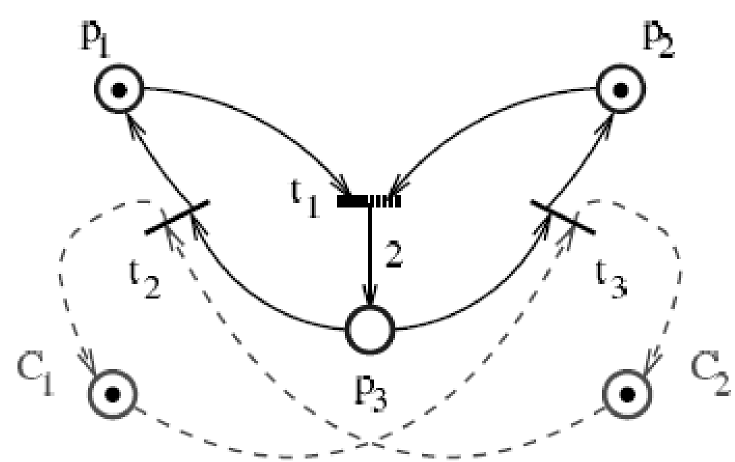
$$= \begin{bmatrix} -2 & 0 & -1 \\ 0 & -2 & -1 \end{bmatrix} \begin{bmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \\ 2 & -1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

$$\mu_{c0} = b_a - L_a \mu_{p0}$$

$$= \begin{bmatrix} -1 \\ -1 \end{bmatrix} - \begin{bmatrix} -2 & 0 & -1 \\ 0 & -2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$




*Obtained the desired result: supervisor does not touch $t_1$.*

***This course is ending. What is next?***

# *This course is ending .*

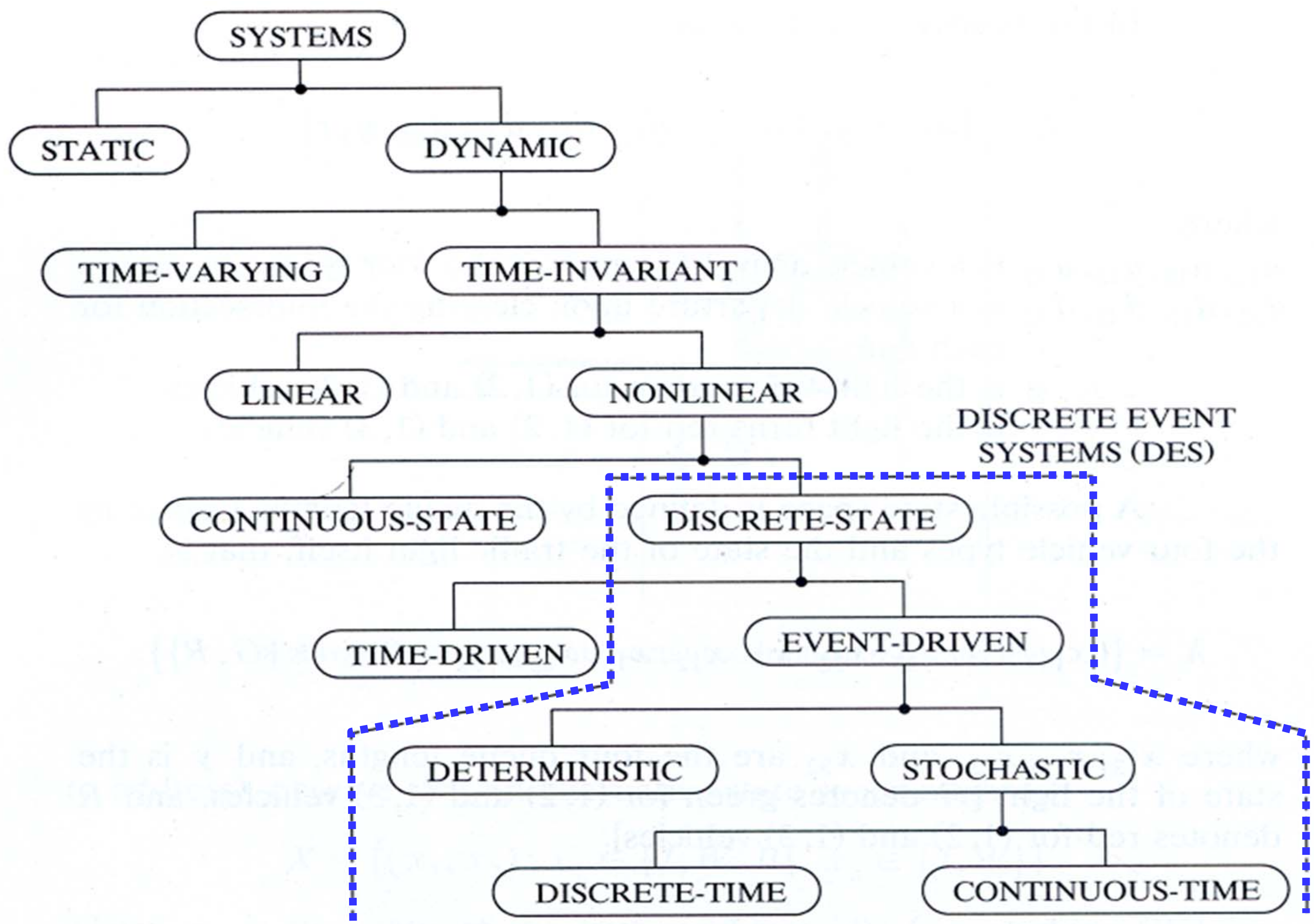



**Figure 1.29.** Major system classifications.

***What is next ?***

## *Top 10 Challenges in Logic Control for Manufacturing Systems*

*by Dawn Tilbury from University of Michigan*

| | |
|---|---|
| 10. Distributed Control | (General management of distributed control applications, Open/distributed control -- ethernet-based control) |
| 9. Theory | (No well-developed and accepted theory of discrete event control, in contrast to continuous control) |
| 8. Languages | (None of the programming languages do what we need but nobody wants a new programming language) |
| 7. Control logic synthesis | (automatically) |
| 6. Standards | (Machine-control standards -- every machine is different, Validated standards, Standardizing different types of control logic programming language) |
| 5. Verification | (Standards for validation, Simulation and verification of controllers) |
| 4. Software | (Software re-usability -- cut and paste, Sophisticated software for logic control, User-unfriendly software) |
| 3. Theory/Practice Gap | (Bridging the gap between industry and academia, Gap between commercial software and academic research) |
| 2. Education | (Educating students for various PLCs, Education and keeping current with evolution of new control technologies, Education of engineers in logic control, Lack of curriculum in discrete-event systems) |
| | And the number one challenge in logic control for manufacturing systems is... |
| 1. Diagnostics | (Integrating diagnostic tools in logic control, Standardized methodologies for design, development, and implementation of diagnostics) |

***The End .***