Industrial Automation

(Automação de Processos Industriais)

Analysis of Discrete Event Systems

http://www.isr.ist.utl.pt/~pjcro/courses/api1011/api1011.html

Prof. Paulo Jorge Oliveira
pjcro @ isr.ist.utl.pt
Tel: 21 8418053 or 2053 (internal)

Syllabus:

Chap. 6 – Discrete Event Systems [2 weeks]

…

Chap. 7 – Analysis of Discrete Event Systems [2 weeks]

Properties of DESs.

Methodologies to analyze DESs:

* The Reachability tree.
* The Method of Matrix Equations.

…

Chap. 8 – DESs and Industrial Automation [1 week]

Some pointers to Sistemas de Eventos Discretos

History: http://prosys.changwon.ac.kr/docs/petrinet/1.htm

Tutorial: http://vita.bu.edu/cgc/MIDEDS/
http://www.daimi.au.dk/PetriNets/

Analyzers, and Simulators: http://www.ppgia.pucpr.br/~maziero/petri/arp.html (in Portuguese)
http://wiki.daimi.au.dk:8000/cpntools/cpntools.wiki
http://www.informatik.hu-berlin.de/top/pnk/download.html

Bibliography:

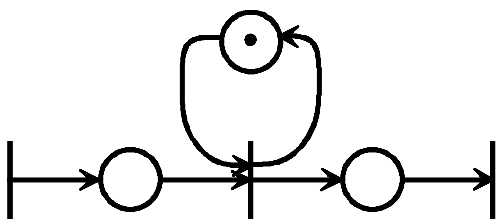
* Cassandras, Christos G., "Discrete Event Systems - Modeling and PerformanceAnalysis", Aksen Associates, 1993.
* Peterson, James L., "Petri Net Theory and the Modeling of Systems", Prentice-Hall,1981. Online em http://prosys.changwon.ac.kr/docs/petrinet/
* Petri Nets and GRAFCET: Tools for Modelling Discrete Event Systems R. DAVID, H. ALLA, New York : PRENTICE HALL Editions, 1992

Properties of Discrete Event Systems

Reachability

Given a Petri net $C=(P, T, I, O, \mu_0)$ with initial marking μ_0 , the set of all markings that can be obtained is the reachable set $\mu' \in R(C,\mu)$.

Note: in general is infinite!



How to compute $R(C,\mu)$?

How to describe $R(C,\mu)$?

Properties of Discrete Event Systems

Coverability

Given a Petri net C=(P, T, I, O, μ_0) with initial marking μ_0, the state $\mu' \in R(C, \mu)$ is covered if $\mu'(i) \leq \mu(i)$, for all places $p_i \in P$.

Is it possible to use this property to help on the search for the reachable set?

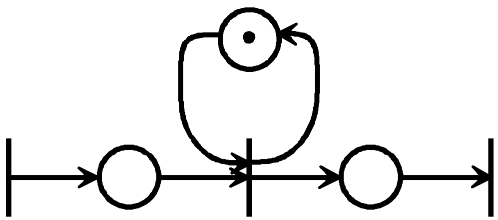
Yes!, see next...

Properties of Discrete Event Systems

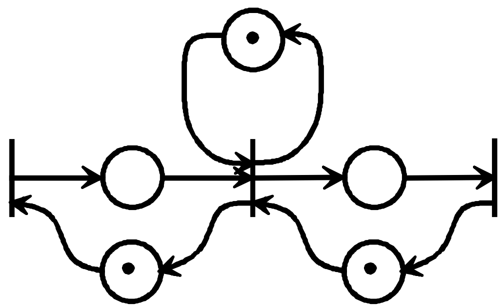
Safeness

A place $p_i \in P$ of the Petri net $C=(P, T, I, O, \mu_0)$ is safe if for all $\mu' \in R(C,\mu_0)$: $\mu_i' \leq 1$.

A Petri net is safe if all its places are safe.



Petri net not safe



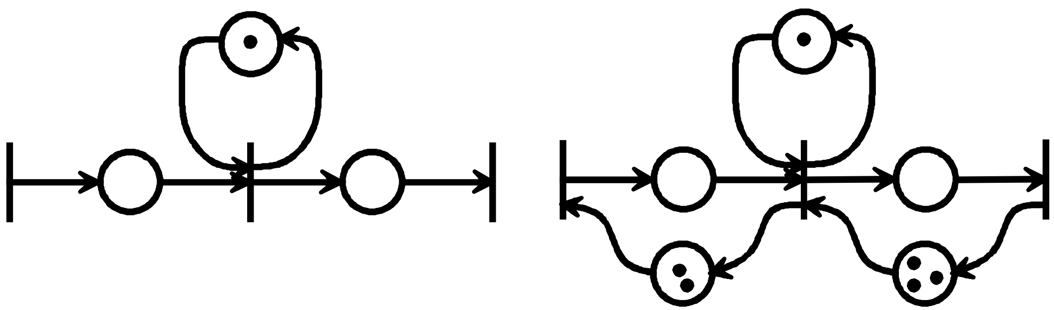
Petri net safe

Properties of Discrete Event Systems

Boundness

A place $p_i \in P$ of the Petri net $C=(P, T, I, O, \mu_0)$ is k-bounded if for all $\mu' \in R(C,\mu_0)$: $\mu_i' \leq k$.

A Petri net is k-bounded if all places are k-bounded.



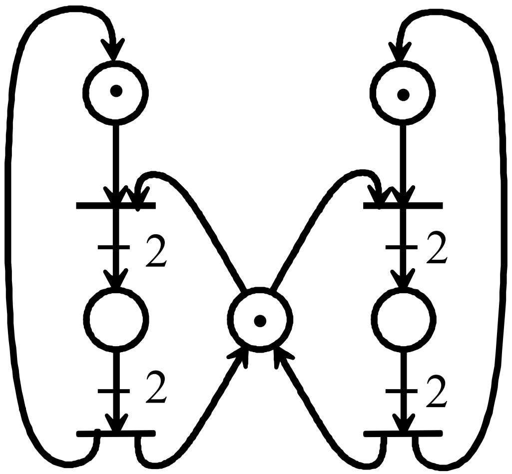
Petri net not bounded   Petri net 3-bounded

Properties of Discrete Event Systems

Conservation

A Petri net $C=(P, T, I, O, \mu_0)$ is **strictly conservative** if for all $\mu' \in R(C,\mu)$

$$\sum_{p_i \in P} \mu'(p_i) = \sum_{p_i \in P} \mu(p_i)$$



Petri net strictly conservative

Properties of Discrete Event Systems

Liveness

A transition t_j is live of

Level 0 - if it can never be fired.

Level 1 - if it is potentially firable, that is if there exists $\mu' \in R(C,\mu)$ **such that** t_j **is enabled in** μ'.

Level 2 - if for each integer *n*, there exists a firing sequence such that t_j **occurs** n **times.**

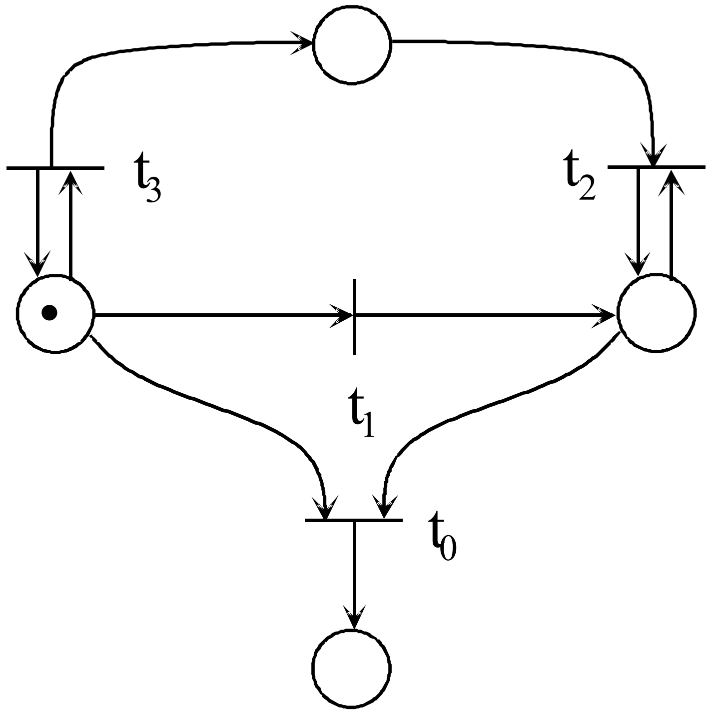
Level 3 - if there exists an infinite firing sequence such that t_j **occurs infinite times.**

Level 4 - if for each $\mu' \in R(C,\mu)$ **there exist a sequence** s **such that the transition** t_j **is enabled.**

Properties of Discrete Event Systems

Example of liveness of transitions

- t_0 is of level 0.
- t_1 is of level 1.
- t_2 is of level 2.
- t_3 is of level 3.



Properties of Discrete Event Systems

Reachability

Given a Petri net $C=(P, T, I, O, \mu_0)$ with initial marking μ_0 and a marking μ', is $\mu' \in R(C, \mu_0)$ reachable?

Analysis methods:

- Brute force...
- Reachability tree
- Matrix Equations

Analysis Methods

Reachability Tree

Build the tree of reachable markings;
Constituted by three types of nodes:

- terminals
- interiors
- duplicated

This method can also be used to study the other properties previously introduced.

See examples...

The infinity marking symbol (ω) is introduced whenever a marking covers other. Used to allow to obtain finite trees.

Analysis Methods

Reachability Tree

Algebra of the infinity symbol (ω):

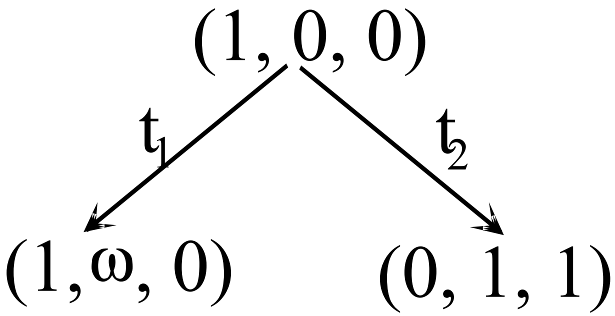
For every positive integer *a* the following relations are verified:

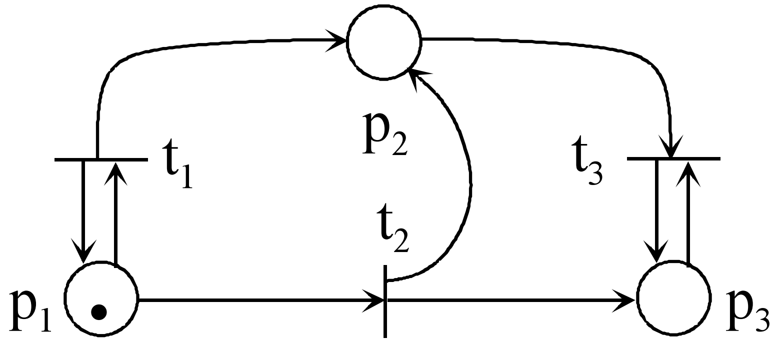
1. $\omega + a = \omega$
2. $\omega - a = \omega$
3. $a < \omega$
4. $\omega \leq \omega$

Theorem - If there exist terminal nodes in the reachability tree then the corresponding Petri net has *deadlocks*.

Analysis Methods

Example of reachability tree:



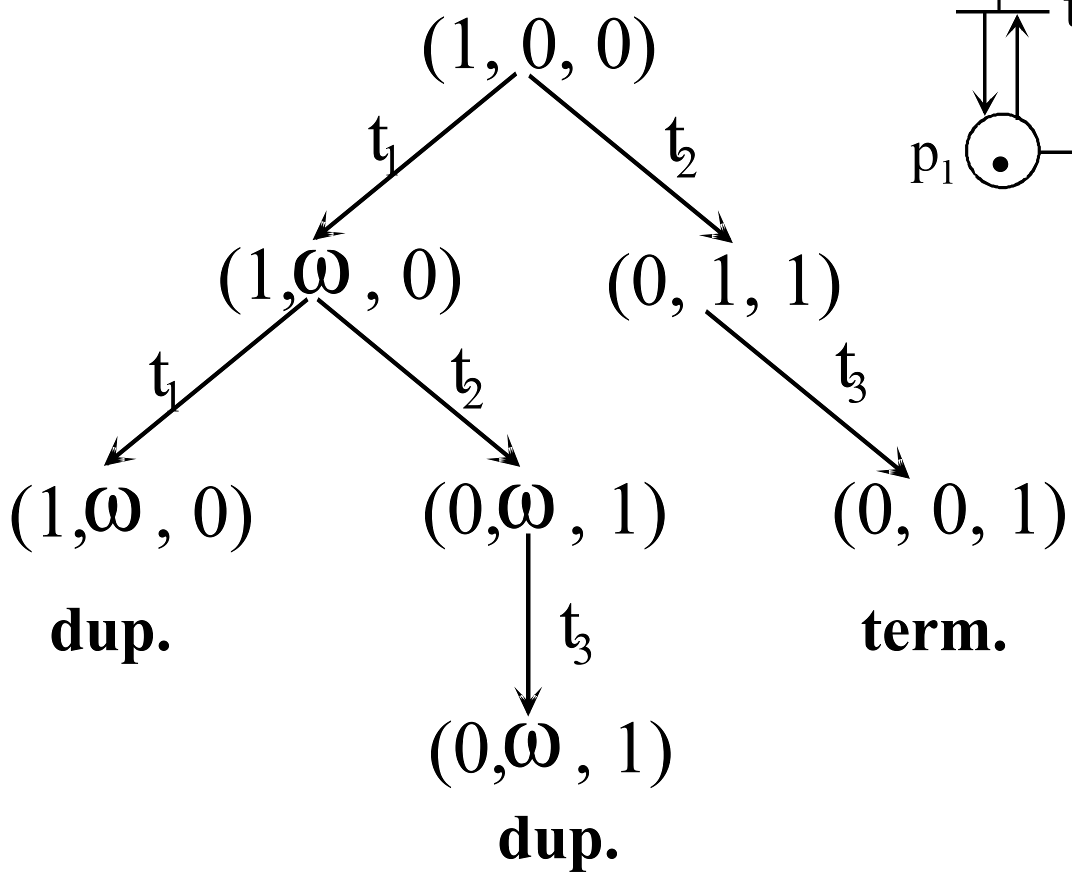


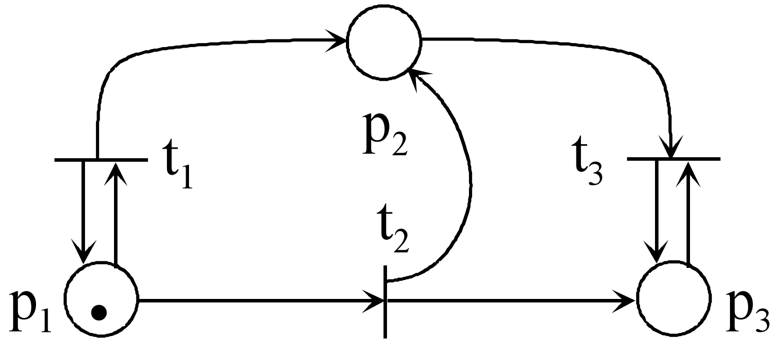
... but (1, 1, 0) covers (1, 0, 0)!

Then the infinity symbol ω can be introduced.

Analysis Methods

Example of reachability tree:



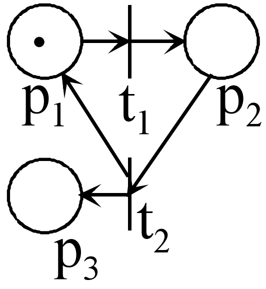


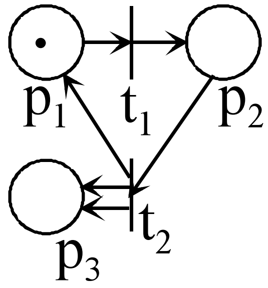
We can conclude imediately that there are

DEADLOCKS!

Other example:

(or a couter-example)





(1, 0, 0)	(1, 0, 0)
↓ t_1	↓ t_1
(0, 1, 0)	(0, 1, 0)
↓ t_2	↓ t_2
(1, 0, ω)	(1, 0, ω)
↓ t_1	↓ t_1
(0, 1, ω)	(0, 1, ω)
↓ t_2	↓ t_2
(1, 0, ω)	(1, 0, ω)

Different reachable sets with the same reachability tree!!!

Decidibility Problem

Example of a Petri net

(P, T, A, w, x_0)

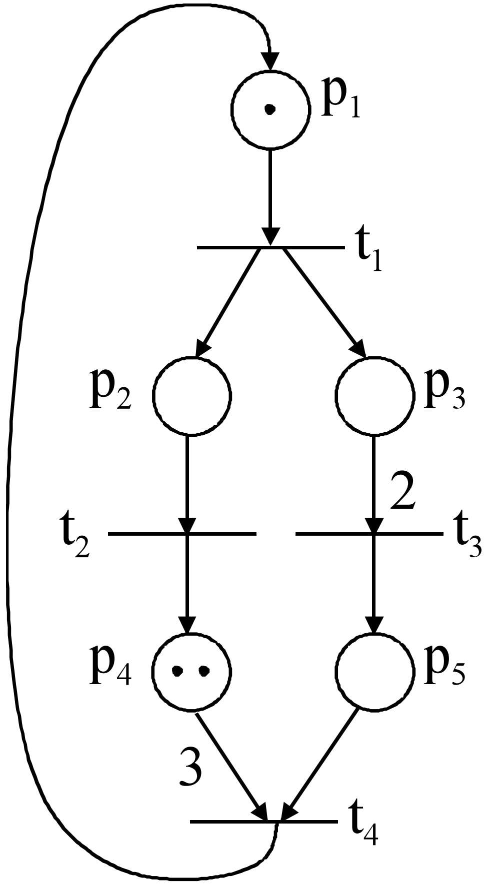
$P=\{p_1, p_2, p_3, p_4, p_5\}$

$T=\{t_1, t_2, t_3, t_4\}$

$A=\{(p_1, t_1), (t_1, p_2), (t_1, p_3), (p_2, t_2), (p_3, t_3),$
$(t_2, p_4), (t_3, p_5), (p_4, t_4), (p_5, t_4), (t_4, p_1)\}$

$w(p_1, t_1)=1, w(t_1, p_2)=1, w(t_1, p_3)=1, w(p_2, t_2)=1$
$w(p_3, t_3)=2, w(t_2, p_4)=1, w(t_3, p_5)=1, w(p_4, t_4)=3$
$w(p_5, t_4)=1, w(t_4, p_1)=1$

$x_0 = \{1, 0, 0, 2, 0\}$

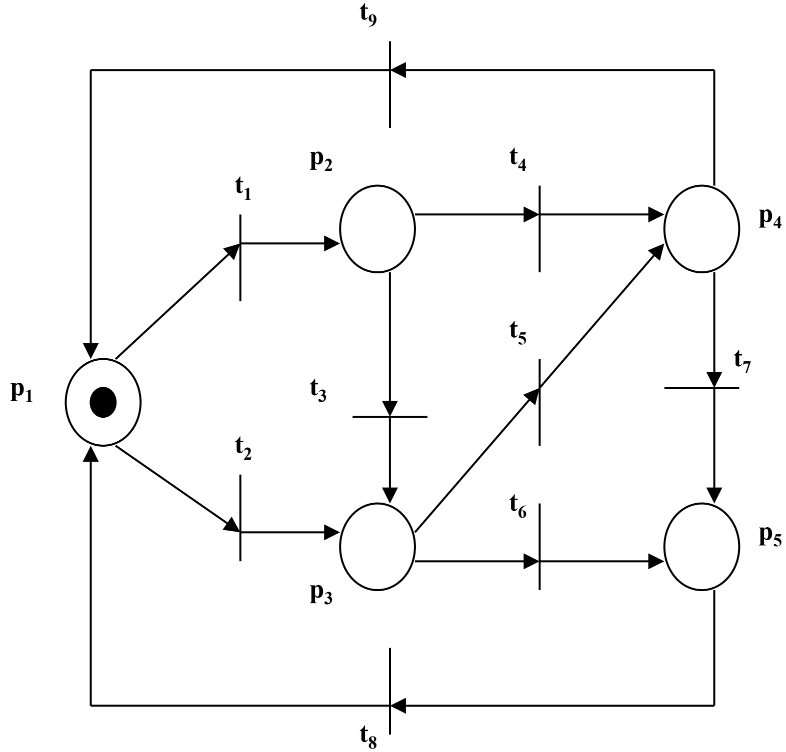


Discrete Event Systems

Example of a simple automation system modelled using PNs

An automatic soda selling machine accepts 50 c and \$1 coins and sells 2 types of products: SODA A, that costs \$1.50 and SODA B, that costs \$2.00.

Assume that the money return operation is omitted.



p_1: machine with \$0.00;
t_1: coin of 50 c introduced;
t_8: SODA B sold.

Analysis Methods

Method of the Matrix Equations (of State Evolution)

The dynamics of the Petri net state can be written in compact form as:

$$\mu(k+1) = \mu(k) + Dq(k)$$

This method can also be used to study the other properties previously introduced.

Requires some thought... ;)

where:

- $\mu(k+1)$ - marking to be reached
- $\mu(k)$ - initial marking
- $q(k)$ - firing vector (transitions)
- D - incidence matrix. Accounts the balance of tokens, giving the transitions fired.

Analysis Methods

How to build the Incidence Matrix?

For a Petri net with *n* places and *m* transitions

$$\mu \in N_0^{\ n}$$

$$q \in N_0^{\ m}$$

$$D = D^+ - D^- \qquad \in \mathrm{Z}^{n\times m}$$

The enabling firing rule is $\mu \geq D^- q$.

Can also be writen in compact form as the inequality

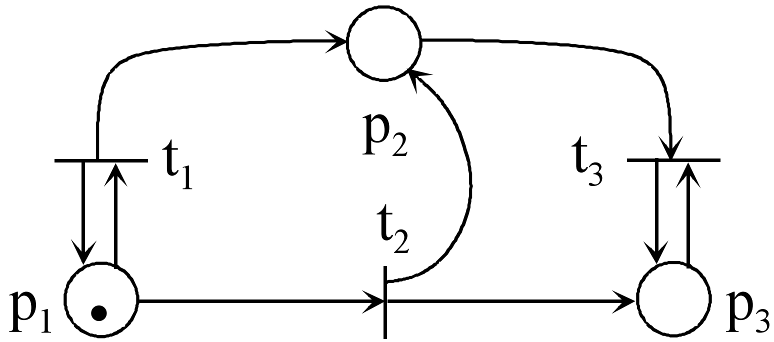
$$\mu + Dq \geq 0,$$

interpreted element by element.

Analysis Methods

Example on the use of the method of matrix equations

$$\mu(k+1) = \mu(k) + Dq(k)$$



$$\mu(k+1) = \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix}, \ \mu(k) = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 1 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad q(k) = \begin{bmatrix} \sigma_{t1} \\ \sigma_{t2} \\ \sigma_{t3} \end{bmatrix} \quad \begin{cases} 1 = 1 - \sigma_{t2} \\ 3 = \sigma_{t1} + \sigma_{t2} - \sigma_{t3} \\ 0 = \sigma_{t2} \end{cases} \quad \begin{cases} \sigma_{t2} = 0 \\ \sigma_{t1} - \sigma_{t3} = 3 \end{cases}$$

Verify!

Analysis Methods

Properties that can be studied immediately with the Method of Matrix Equations

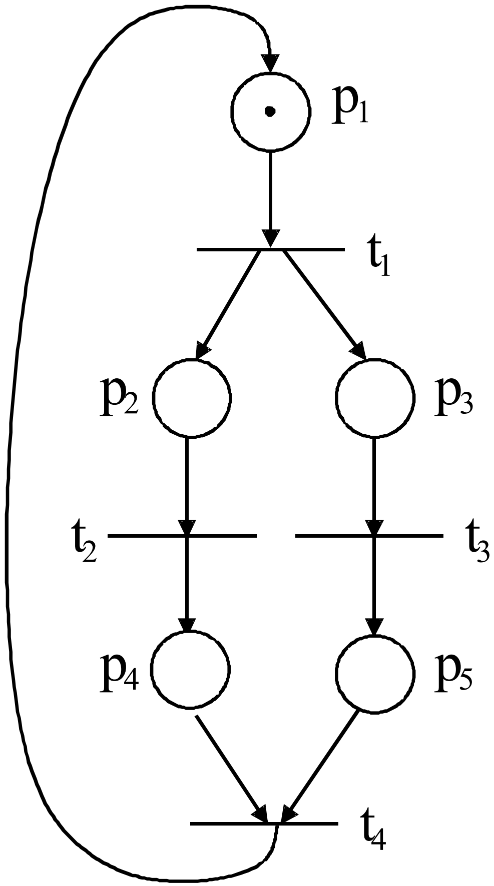
- Reachability (sufficient condition)

Theorem – if the problem of finding the transition firing vector that drives the state of a Petri net from μ to state μ' has no solution, resorting to the method of matrix equations, then the problem of reachability of μ' does not have solution.

- Conservation – the firing vector is a by-product of the MME.

- Temporal invariance – cycles of operation can be found.

Example of a Petri net



Conservation

For the number of tokens (weighted) to be preserved :

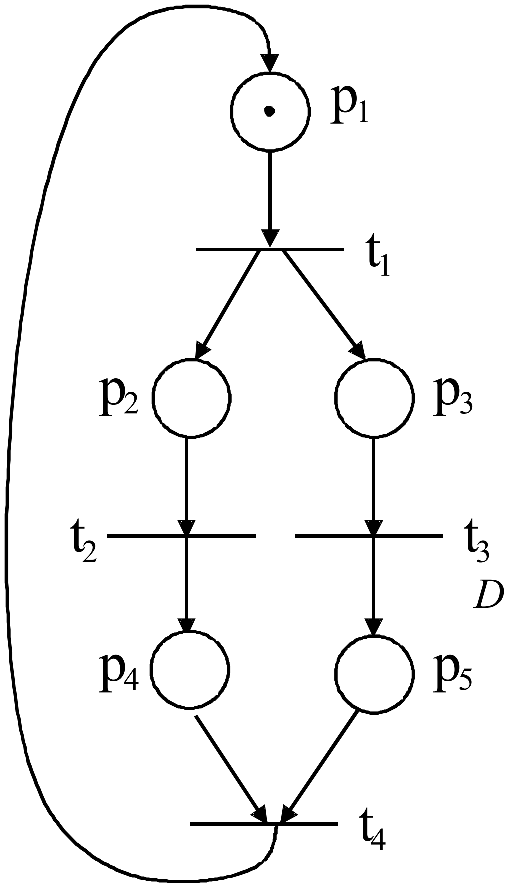
$$x^T \mu' = x^T \mu + x^T D q$$

$$x^T D = 0$$

$$D = \begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \quad \begin{cases} -x_1 + x_2 + x_3 = 0 \\ -x_2 + x_4 = 0 \\ -x_3 + x_5 = 0 \\ x_1 - x_4 - x_5 = 0 \end{cases} \quad \begin{cases} x_1 = x_2 + x_3 \\ x_2 = x_4 \\ x_3 = x_5 \end{cases}$$

Solution: undetermined system of equations $x^T = [2 \quad 1 \quad 1 \quad 1 \quad 1].$

Example of a Petri net



Temporal invariance

To determine the transition firing vectors that make the Petri net return to the same state(s)

$$Dq = 0$$

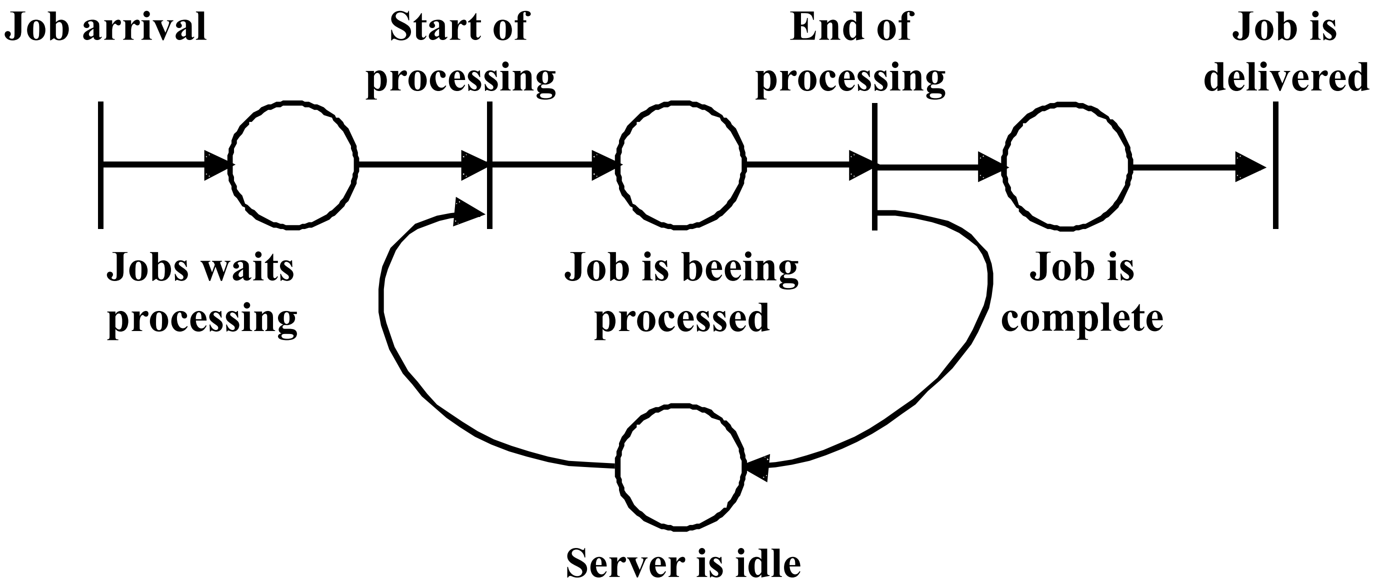
$$D = \begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}, \quad q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \qquad \begin{cases} -q_1 + q_4 = 0 \\ q_1 - q_2 = 0 \\ q_1 - q_3 = 0 \\ q_2 - q_4 = 0 \\ q_3 - q_4 = 0 \end{cases}$$

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

Solution: undetermined system of equations

Example for the analysis of properties:

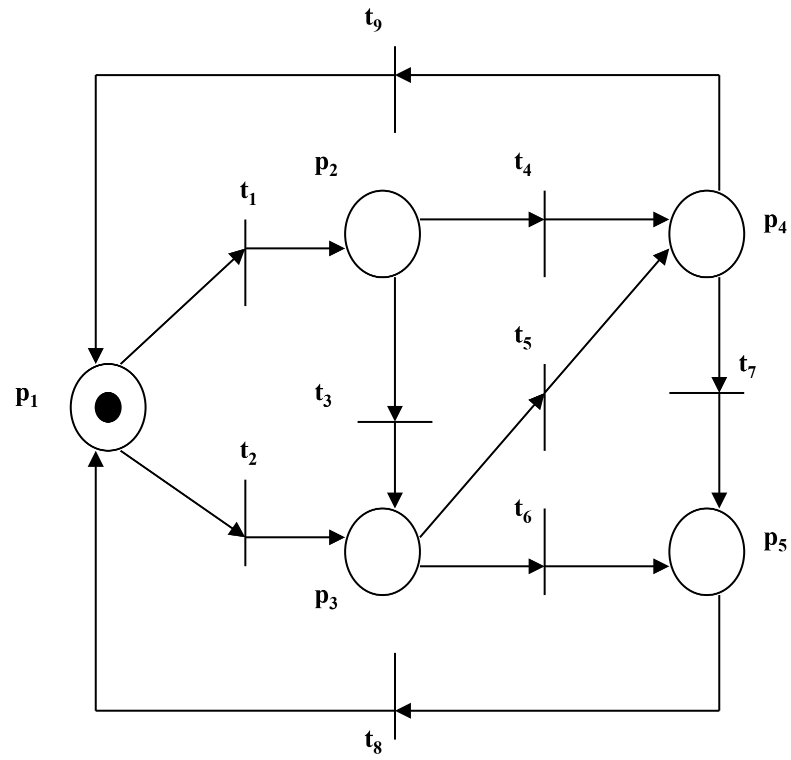
Event	Pre-conditions	Pos-conditions
1	-	b
2	a,b	c
3	c	d,a
4	d	-



Example for the analysis of properties:

An automatic soda selling machine accepts 50 c and \$1 coins and sells 2 types of products: SODA A, that costs \$1.50 and SODA B, that costs \$2.00.

Assume that the money return operation is omitted.

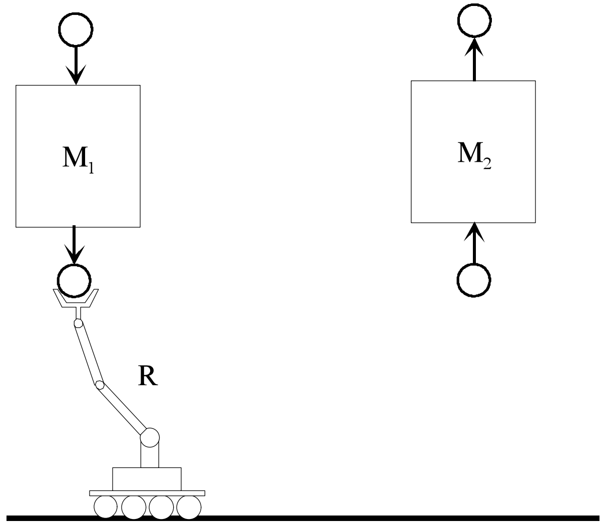


p_1: machine with \$0.00;
t_1: coin of 50 c introduced;
t_8: SODA B sold.

Example for the analysis of properties:

Manufacturing cell with robotic parts handling

Manufacturing system composed by 2 machines (M_1 and M_2) and a robotic manipulator (R). This takes the finished parts from machine M_1 and transports them to M_2.



No buffers available on the machines. If R arrives near M_1 and the machine is busy, the part is rejected.

If R arrives near M_2 and the machine is busy, the manipulator must wait.

Machinning time: M_1=0.5s; M_2=1.5s; $R_{M1 \text{®} M2}$=0.2s; $R_{M2 \text{®} M1}$=0.1s;

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)

Complexity and Decidibility

• A problem is *undecidable* if it is proven that no algorithm to solve it exists.

An example of a undecidable problem is the stop of a Turing machine (MT): "Will the TM stops for the code n after using the number m?".

• For *decidable* problems, the complexiy of the solutions have to be taken into account, that is, the computational cost in terms of memory and time.

Basic example: multiplication of number in the arabic and latin civilizations...

Reducibility

When to solve a given problem it is possible to reduce it to other problems with known solution

Theorem: Assume that the problem *A* is reducable to problem B:

Then an instance of A can be transformed in an instance of B:

- If *B* is decidable then *A* is decidable.
- If *A* is undecidable then *B* is undecidable.

Reducibility

Equality Problem: Given two marked Petri nets

$C_1 = (P_1, T_1, I_1, O_1)$ and $C_2 = (P_2, T_2, I_2, O_2)$, with markings

m$_1$ e m$_2$, respectively, is $R(C_1, \mu_1) = R(C_2, \mu_2)$?

Subset Problem: Given two marked Petri nets

$C_1 = (P_1, T_1, I_1, O_1)$ and $C_2 = (P_2, T_2, I_2, O_2)$, with markings

m$_1$ e m$_2$, respectively, is $R(C_1, \mu_1) \subseteq R(C_2, \mu 2)$?

> The equality problem is reducable to the subset problem
>
> (Sugg: proove that each set is a subsets of the other)

Decidibility

If a proble in undecidable does it means that it is not solvable?
NO, it means that it was not yet solved!

Classical example: (Fermat Last Theorem)
$x^n + y^n = z^n$ has solution for n>2 and nontrivial integers x, y e z?
Now it is known that the problem is impossible. The problem remained undecidable for more than 2 centuries (solution proven in 1998).

The MT problem is undecidable.

If it were decidable, for instance the Format last theorem would have been proven long time ago, i.e. there would be an algorithm (MT with code n) that computing all combinations of x,y,z and n>2 (number m) to find a solution verifying $x^n + y^n = z^n$.

Reachability Problems

(Given a Petri net C=(P,T,I,O) with initial marking m)

Reachability Problem: For the marking μ', is $\mu' \in R(C,\mu)$?

Sub-marking Reachability Problem:

Given the marking μ' and a subset $P' \subseteq P$, exist $\mu'' \in R(C,\mu)$ such that $\mu''(p_i) = \mu' \; \forall p_i \in P'$?

Zero Reachability Problem:

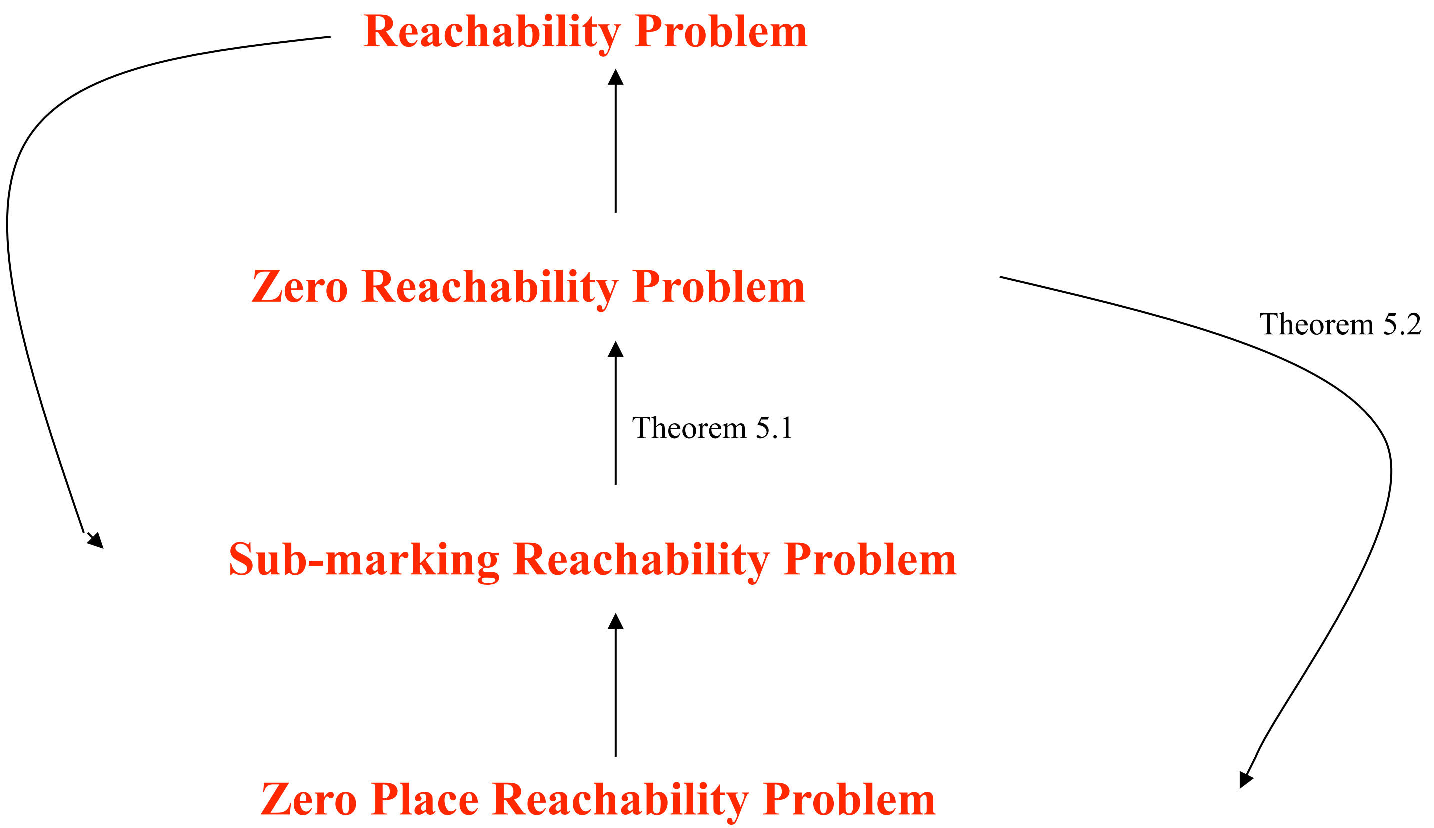
Given the marking μ'=(0 0 … 0), is $\mu' \in R(C,\mu)$?

Zero Place Reachability Problem:

Given the place $p_i \in P$, is $\mu' \in R(C,\mu)$ with $\mu'(p_i) = 0$?

Reachability Problems

A → B : A reducable to B



Reachability Problems

Theorem 5.3: The following reachability are equivalent:

- **Reachability Problem**;
- **Zero Reachability Problem**;
- **Sub-marking Reachability Problem**;
- **Zero Place Reachability Problem**.

Liveness and Reachability

(Given a Petri net C=(P,T,I,O) with initial marking m)

Liveness Problem

Are all transitions t_j of T live?

Transition Liveness Problem

For the transition t_j of T, is t_j live?

The liveness problem is reducable to the transition liveness problem. To solve the first it remains only to solve the second for the *m* Petri net transitions (#T = *m*).

Liveness and Reachability

(Given a Petri net C=(P,T,I,O) with initial marking μ)

Theorem 5.5: The problem of reachability is reducable to the liveness problem.

Theorem 5.6: The problem of liveness is reducable to the reachability problem.

Theorem 5.7: The following problems are equivalent:

- Reachability problem
- Liveness problem

Decidibility results

Theorem 5.10: The sub-marking reachability problem is reducable to the reachable subsets of a Petri net.

Theorem 5.11: The following problem is undecidable:

- Subset problem for reachable sets of a Petri net

They are all reducable to the famous Hilbert's 10th problem:

The solution of the Diophantine equation of n variables, with integer coefficients $P(x_1, x_2, \ldots, x_n)=0$ is undecidable.

(proof by Matijasevic that it is undecidable in the late 1970s).