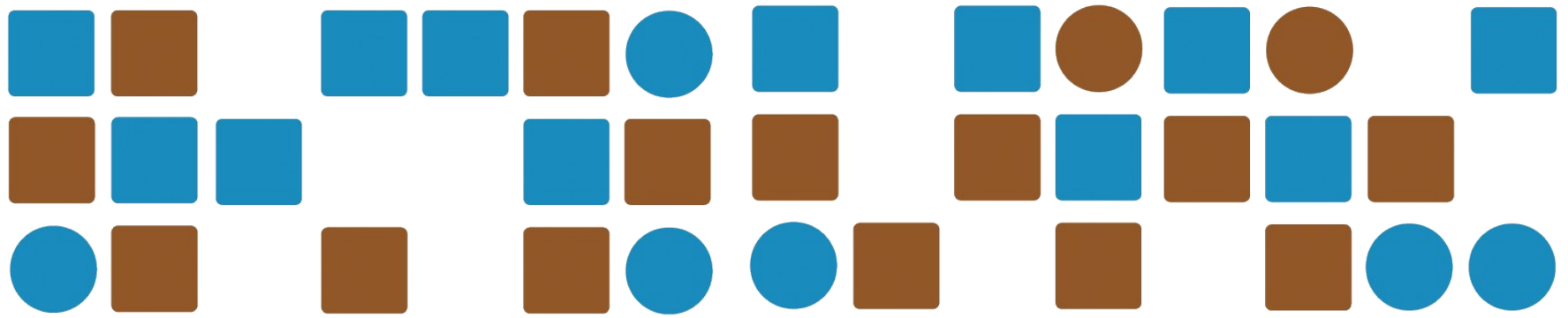




1. PLANNING

1.1. Robotic Manufacturing Systems: Analysis of the logical correctness



240IAU33 – Service and Industrial Robotics

Master's Degree in Industrial Engineering

Contents

1. Motivation examples

1. Two machines with an intermediate buffer
2. A flexible manufacturing system

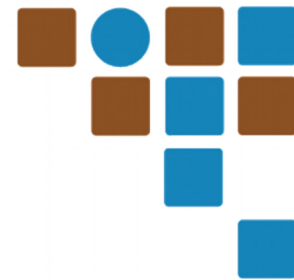
2. Introduction to Petri Nets

1. The Petri Nets formalism
2. Dynamics
3. Petri Nets for the modeling of flexible manufacturing systems

3. Qualitative analysis

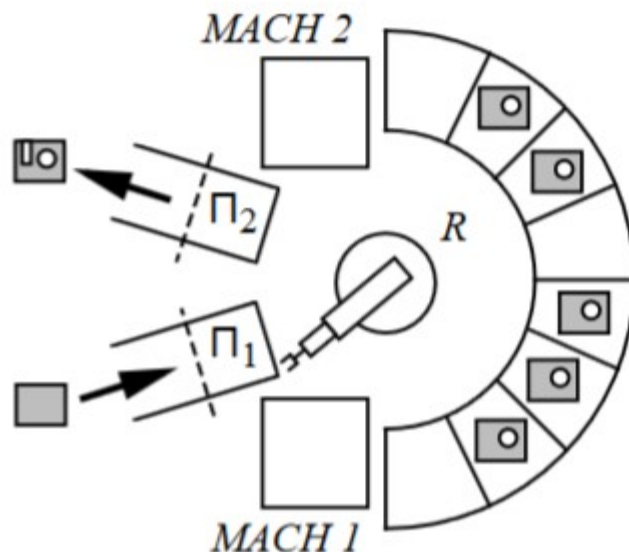
1. Qualitative properties
2. Analysis using the reachability graph
3. Analysis using reduction rules





1. Motivation examples

1.1. Two machines with an intermediate buffer



Parts enter the system through the input conveyor and exit it through the exit conveyor.

Parts are processed first at M1 and then at M2.

The capacity of the machines is one.

The robot is responsible for moving the parts:

- from the input conveyor to M1
- from M1 to the buffer
- from the buffer to M2
- from M2 to the exit conveyor

Which is the key issue in the actions of the robot in order not to block the system?

Contents

1. Motivation examples

1. Two machines with an intermediate buffer
2. A flexible manufacturing system

2. Introduction to Petri Nets

1. The Petri Nets formalism
2. Dynamics
3. Petri Nets for the modeling of flexible manufacturing systems

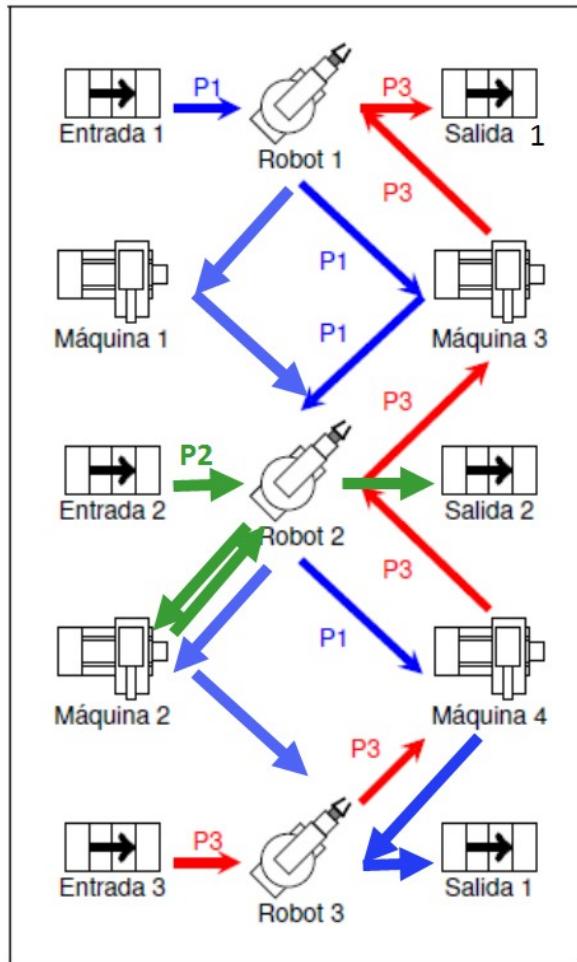
3. Qualitative analysis

1. Qualitative properties
2. Analysis using the reachability graph
3. Analysis using reduction rules



1. Motivation examples

1.2. A flexible manufacturing system



There are:

- three robots (R1, R2 and R3),
- four machines (M1, M2, M3 and M4),
- three input buffers (E1, E2 and E3),
- three output buffers (S1, S2 and S3).

R1 moves parts from E1 to M1 or M3, and from there to S1.

R2 moves parts from E2 to any machine, and from there to S2.

R3 moves parts from E3 to M2 or M4, and from there to S3.

Three types of parts are manufactured, with the following visit of machines:

P1: E1-R1-M3-R2-M4-R3-S1

E1-R1-M1-R2-M2-R3-S1

P2: E2-R2-M2-R2-S2

P3: E3-R3-M4-R2-M3-R1-S3

Contents

1. Motivation examples

2. Introduction to Petri Nets

1. The Petri Nets formalism

2. Dynamics

3. Petri Nets for the modeling of flexible manufacturing systems

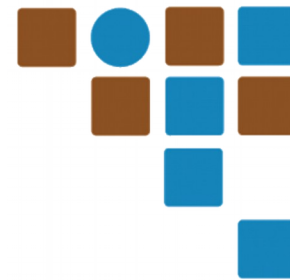
3. Qualitative analysis

1. Qualitative properties

2. Analysis using the reachability graph

3. Analysis using reduction rules



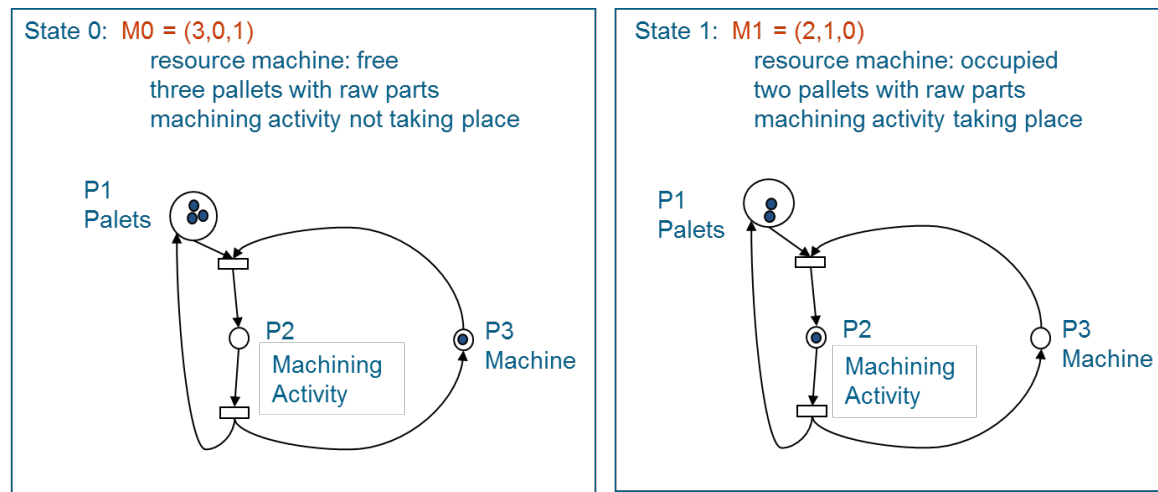


1. Introduction to Petri Nets

1.1. The Petri Nets formalism

Definition

Petri nets are a mathematic formalism based in simple object, relations and rules that allow to represent complex dynamic behaviors of discrete-event dynamic systems (characterized by parallel evolutions, synchronizations and the use of shared resources).



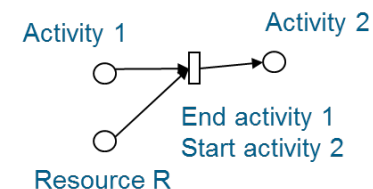
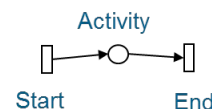
Interpretation

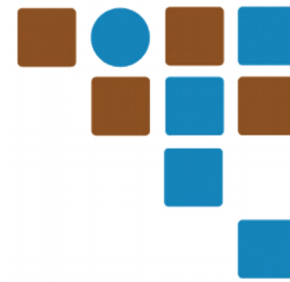
a) **Places** represent:

Activities: A token within a place indicates that the activity is taking place.

Resources: Tokens represent available resources.

b) **Transitions** represent the start and the end of an activity



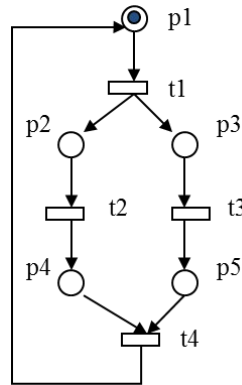


1. Introduction to Petri Nets

1.1. The Petri Nets formalism

Simple examples

Example A



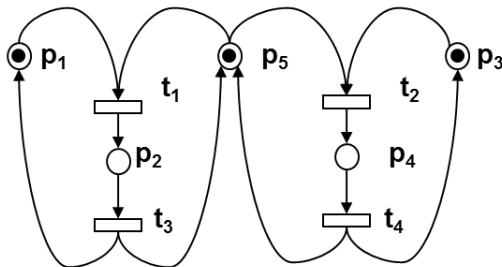
A manufacturing system composed of two lines working in **parallel**:

1. Activities p2 and p4
2. Activities p3 and p5

- Activities modeled by places p2 i p3 **start simultaneously** after an initialization activity represented by p1 is performed.
- The two lines cannot start again until activities p4 and p5 are done; i.e. there is a **synchronization** of these activities modeled by transition t4.

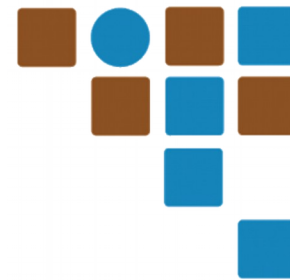
Example B

A manufacturing system consists of two machines M1 and M2 that work in parallel with the assistance of a single operator.



p₁: Machine M1
p₂: Machining activity 1
p₃: Machine M2
p₄: Machining activity 2
p₅: Operator

- The operator is a **shared resource**.
- Activities p2 and p4 are in **mutual exclusion** and compete to size the shared resource.



1. Introduction to Petri Nets

1.1. The Petri Nets formalism

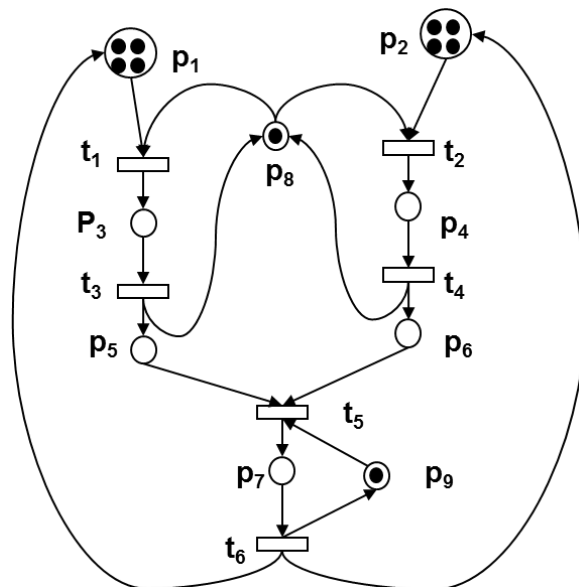
Simple examples

Example C

A manufacturing system consists of a machine M1 that machines two parts, A and B, and an assembly robot that assembles them to produce the final product AB.

The machine is a **shared resource** used in activities p3 and p4 that are in mutual exclusion.

The manufacturing fluxes are **synchronized** in transition t5 that represents the start of the assembly activity.



- p_1 : Raw parts – type A
- p_2 : Raw parts – type B
- p_3 : Machining operation – type A parts-
- p_4 : Machining operation – type A parts-
- p_5 : Buffer for type-A finished parts
- p_6 : Buffer for type-B finished parts
- p_7 : Assembly operation
- p_8 : Machine M1
- p_9 : Assembly robot

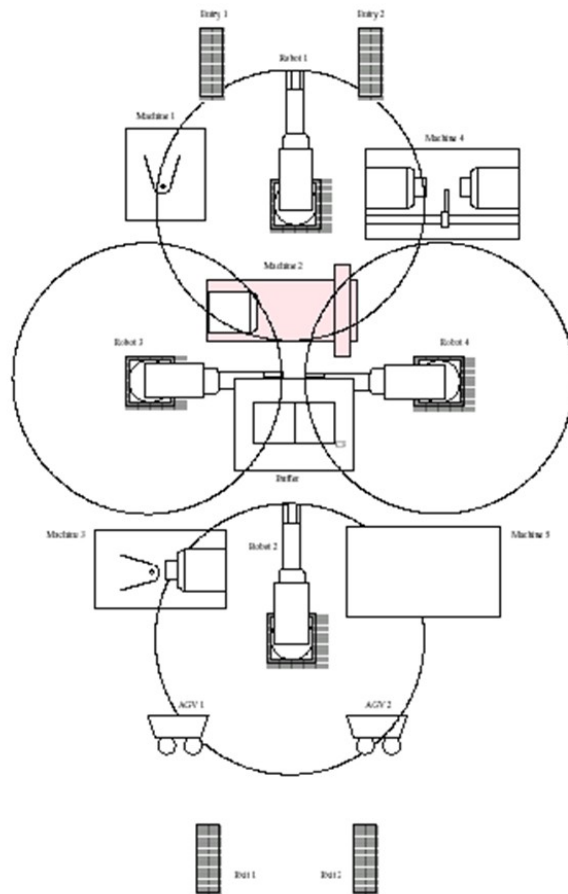


1. Introduction to Petri Nets

1.1. The Petri Nets formalism

Simple examples

Example D



Manufacturing system with the following features:

- It processes two types of parts, F and G.
- Type-F parts enter the system using input I1. Then they are loaded into either M1 or M2 machines using robot R1. After being machined they are downloaded by robot R3 that place them in buffer B. The same Robot R3 takes them from B and loads machine M3 where they are polished. Finally they are downloaded from M3 by robot R2 and placed in the AGV1 that transports them until output O1.
- Type-G parts enter the system using input I2. Then they are loaded into M4 machines using robot R1. After being machined they are downloaded by robot R4 that place them in buffer B. The same Robot R4 takes them from B and loads machine M5 where they are polished. Finally they are downloaded from M5 by robot R2 and placed in the AGV2 that transports them until output O2.
- The entrance of raw parts is unlimited; the maximum number of pallets that the system has is fixed.
- The machines capacity is one, as well as the capacity of the AGVs.
- the buffer B has a capacity for 8 parts, four for type-F parts and four for type-G parts.

1. Introduction to Petri Nets

1.1. The Petri Nets formalism

Simple examples

Example D

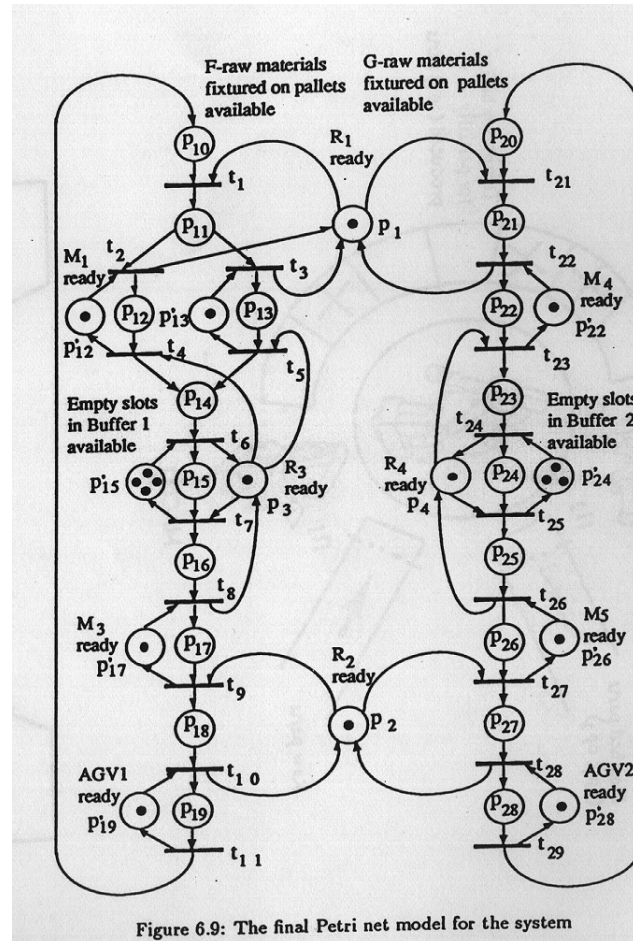
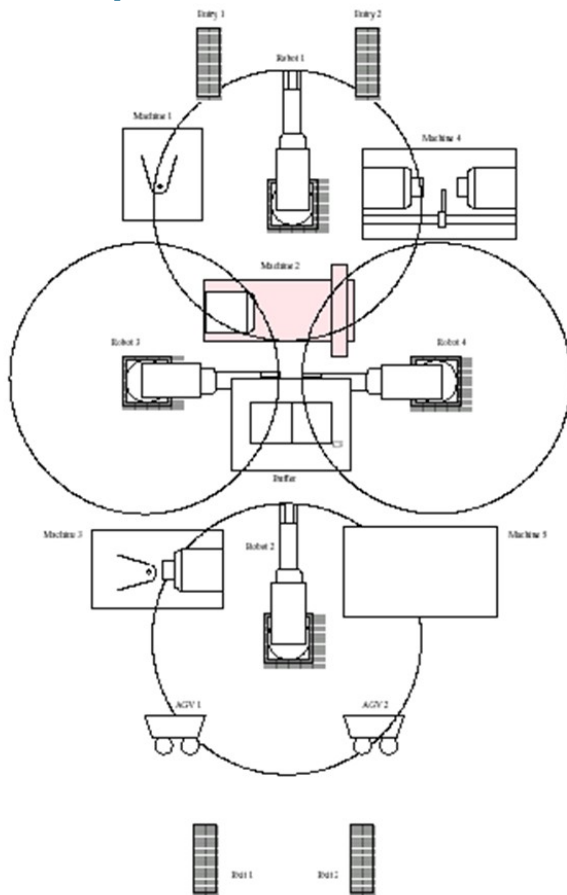


Figure 6.9: The final Petri net model for the system

Remarks:

- Resources R1 and R2: Represented as places p1 and p2, they are **shared resource** used in **parallel process**.

Loading of machines M1/M2 and of M3 (places p11 and p21); loading of AGV1 and of AGV2 (places p18 and p27).

- Resource R3 and R4: Represented as places p3 and p4, they are **shared resources** used in **sequential processes**.

Downloading of M1/M2 and loading of M3 (places p14 and p23); downloading of M4 and loading of M5 (places p23 and p25).

Contents

1. Introduction to Petri Nets

1. The Petri Nets formalism

2. Dynamics

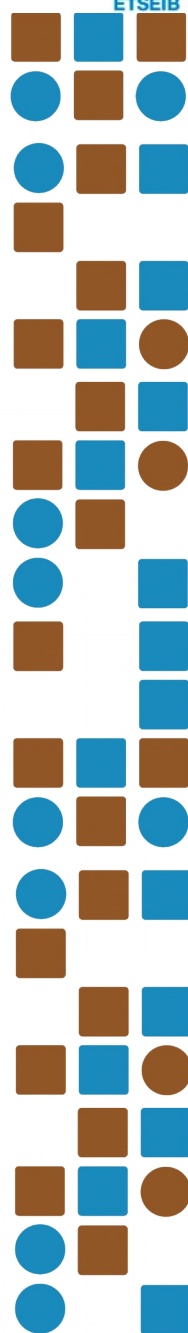
3. Petri Nets for the modeling of flexible manufacturing systems

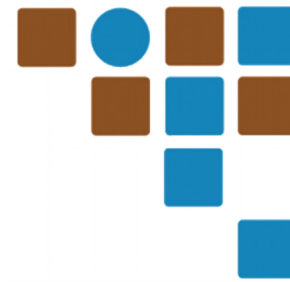
2. Qualitative analysis

1. Qualitative properties

2. Analysis using the reachability graph

3. Analysis using reduction rules





1. Introduction to Petri Nets

1.2. Dynamics

The token game

The dynamics of a PN is represented by the evolution of its marking. This evolution is defined by a rule, known as token game, that works as follows:

1.- A transition is **enabled** by a given marking if all of its input places are marked with as many tokens as indicated by the weight of the arc connecting the place with the transition.

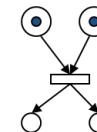
2.- The **firing** of a transition is the instantaneous operation that:

- Removes as many tokens from the input places as indicated by the weight of the arcs connecting the input place with the transition.
- Fills the output places with as many tokens as indicated by the weight of the arcs connecting the transition with the output places.

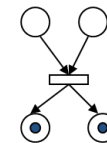
Remarks:

- Enabled transitions are not forced to fire.
- Transitions are fired once at a time.
- The number of tokens may vary after the firing of a transition.

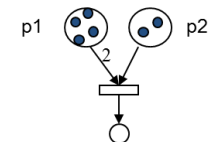
Enabled transition: the weight of all the arcs is 1 (when not shown the weights are always assumed unitary) and each input place has one token.



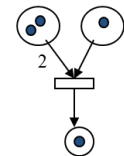
The firing of this transition removes one token from each input place and fills each output place with one (all the weights are 1). The number of tokens is maintained although this is an irrelevant point.



Enabled transition: Input places contain at least as many tokens as indicated by the input arcs.



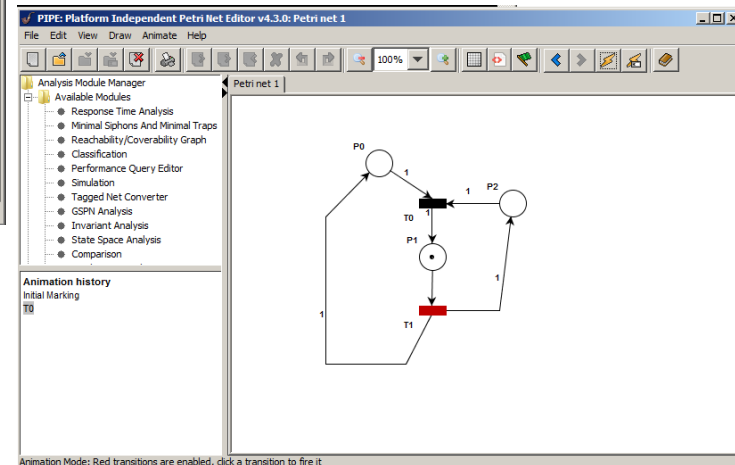
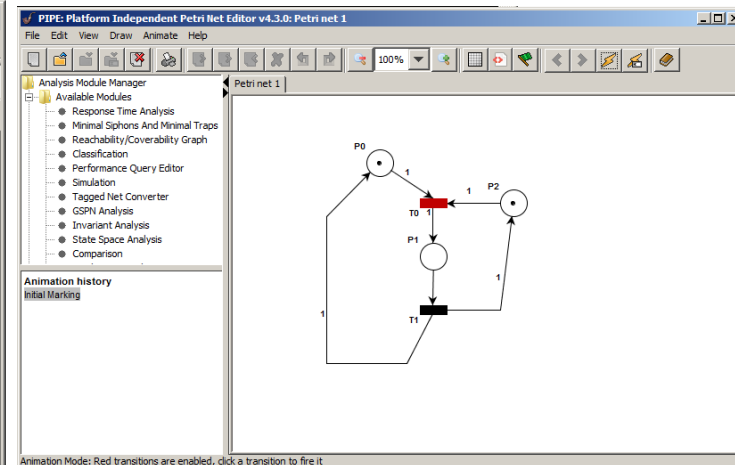
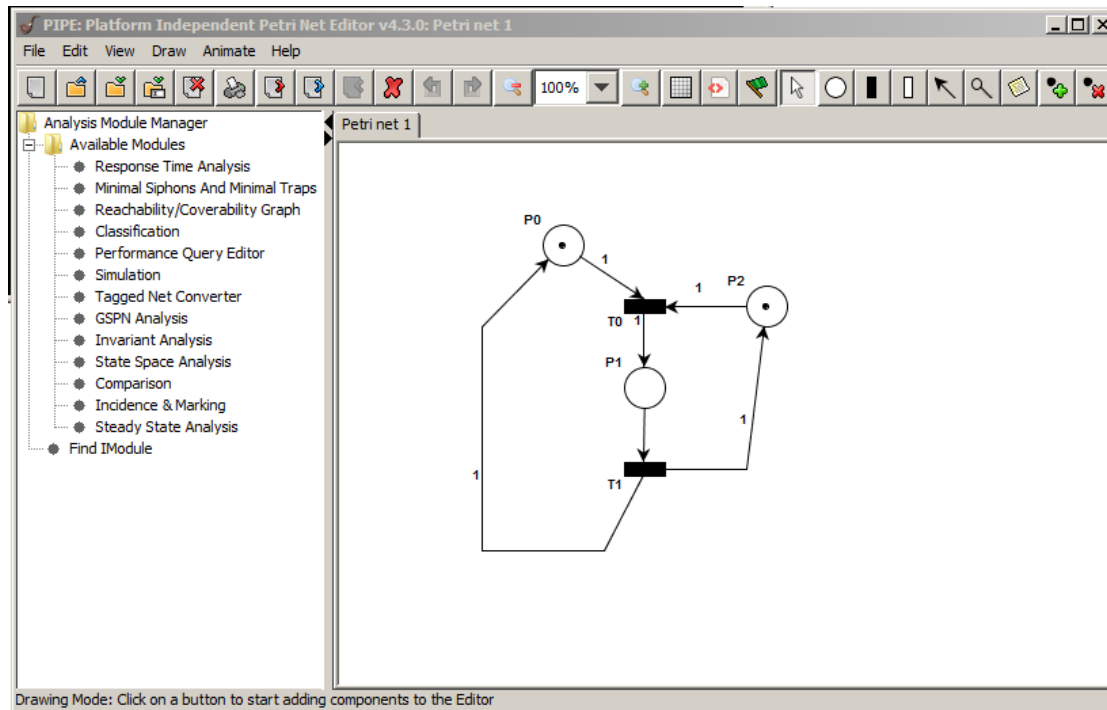
The firing of this transition removes two tokens from p1 and one from p2; and fills the output place with one token. After firing the transition is still enabled and hence prepared to be fired again.



1. Introduction to Petri Nets

1.2. Dynamics

The PIPE software



Contents

1. Introduction to Petri Nets

1. The Petri Nets formalism

2. Dynamics

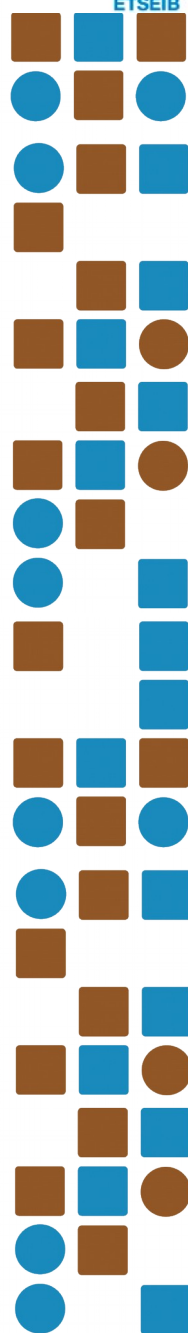
3. Petri Nets for the modeling of FMS

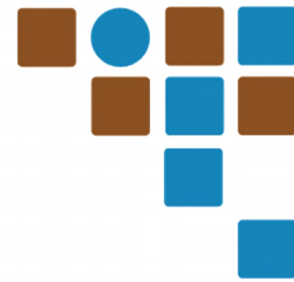
2. Qualitative analysis

1. Qualitative properties

2. Analysis using the reachability graph

3. Analysis using reduction rules





1. Introduction to Petri Nets

1.3. Petri Nets for the modeling of FMS

Features

The main features that Petri Nets provide for the modeling of flexible manufacturing systems are the following:

- They can easily represent the concurrence, causality or the synchronization of activities, as well as the conflicts and the use of shared resources.
- The design can be done either in a bottom-up or top-down manner.
- The graphic representation eases the documentation and the monitoring of the processes.
- The formal semantics allows a rigorous analysis of the qualitative properties of the system, as well as the automation of code generation for the control or the simulation of the system.
- Different meanings can be assigned to places, allowing to cover with a single tool several phases of the system design (e.g. planning, scheduling, real-time control).
- They can easily represent the flexibility in FMS.

Contents

1. Motivation examples

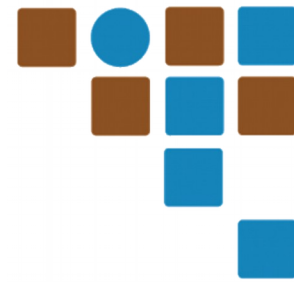
2. Introduction to Petri Nets

1. The Petri Nets formalism
2. Dynamics
3. Petri Nets for the modeling of FMS
4. Classification

3. Qualitative analysis

1. Qualitative properties
2. Analysis using the reachability graph
3. Analysis using reduction rules



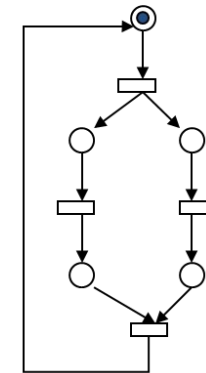


2. Introduction to Petri Nets

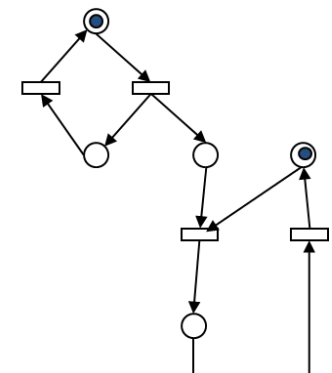
2.1. Qualitative Properties

Boundness

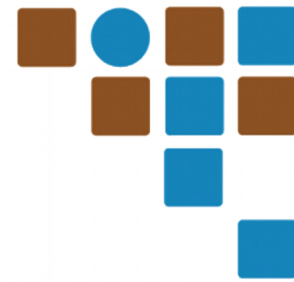
- The bound of a **place** is the maximum number of tokens that it contains, for all the reachable markings. If the bound is finite the place is called a **bounded** place.
- A Petri Net is bounded if all of its places are bounded.
- A bounded Petri Net has a finite state space.



BLR

 $\overline{\text{BLR}}$

In a manufacturing system, the fact that the system is bounded implies that the capacity of all the resources is under control (for instance, no buffer can be overflowed).

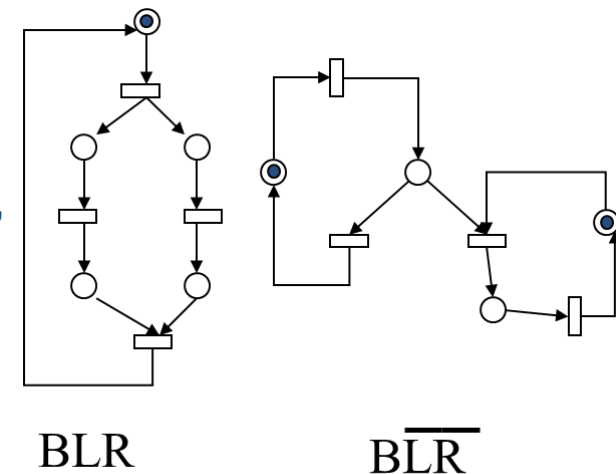


2. Introduction to Petri Nets

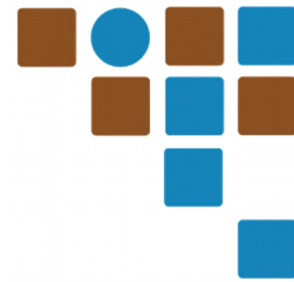
2.1. Qualitative Properties

Liveness

- A **transition** is called **live** if it does not lose the capacity of being enabled, i.e. from any reachable marking there exists a sequence of transitions that, when fired, allow to enable the transition.
- A system is called a **live system** if all of its transitions are **live**.
- A **deadlock** is a marking from where no transition can be fired, hence a system with a deadlock is a non-live system (with all the transitions non-live).



In a manufacturing system, the liveness property represents the fact that the events modeled by the transitions of the net never lose the capacity to occur.

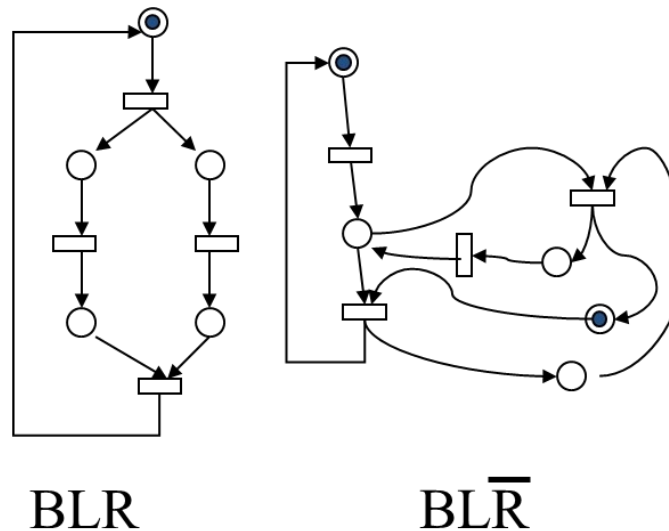


2. Introduction to Petri Nets

2.1. Qualitative Properties

Reversibility

- A system is **reversible** if the initial marking can be reached from any reachable marking.
- A system with a deadlock has not the reversibility property.



In a manufacturing system, the reversibility property represents the fact that the system can be reinitiated from any possible error state.

Contents

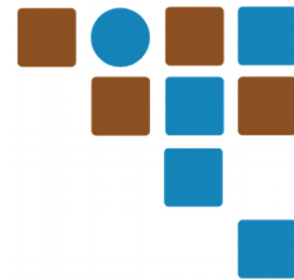
1. Introduction to Petri Nets

1. The Petri Nets formalism
2. Dynamics
3. Petri Nets for the modeling of FMS
4. Classification

2. Qualitative analysis

1. Qualitative properties
2. Analysis using the reachability graph
3. Analysis using reduction rules





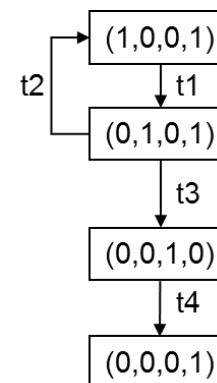
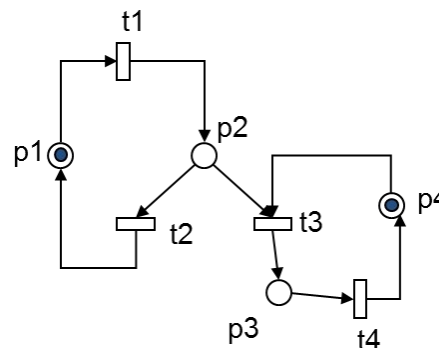
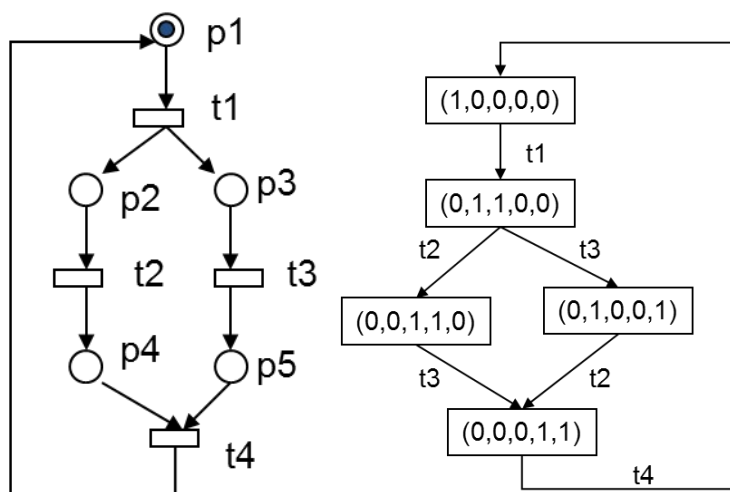
2. Introduction to Petri Nets

2.2. Analysis using the reachability graph

Reachability graph

It is based in the generation of all the possible markings of the net and their connectivity. The resulting graph is called **reachability graph**.

If the reachability graph is finite then it allows us to easily compute the qualitative properties of the net.



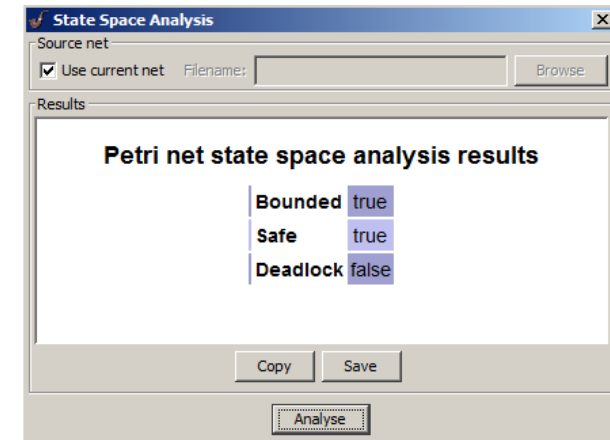
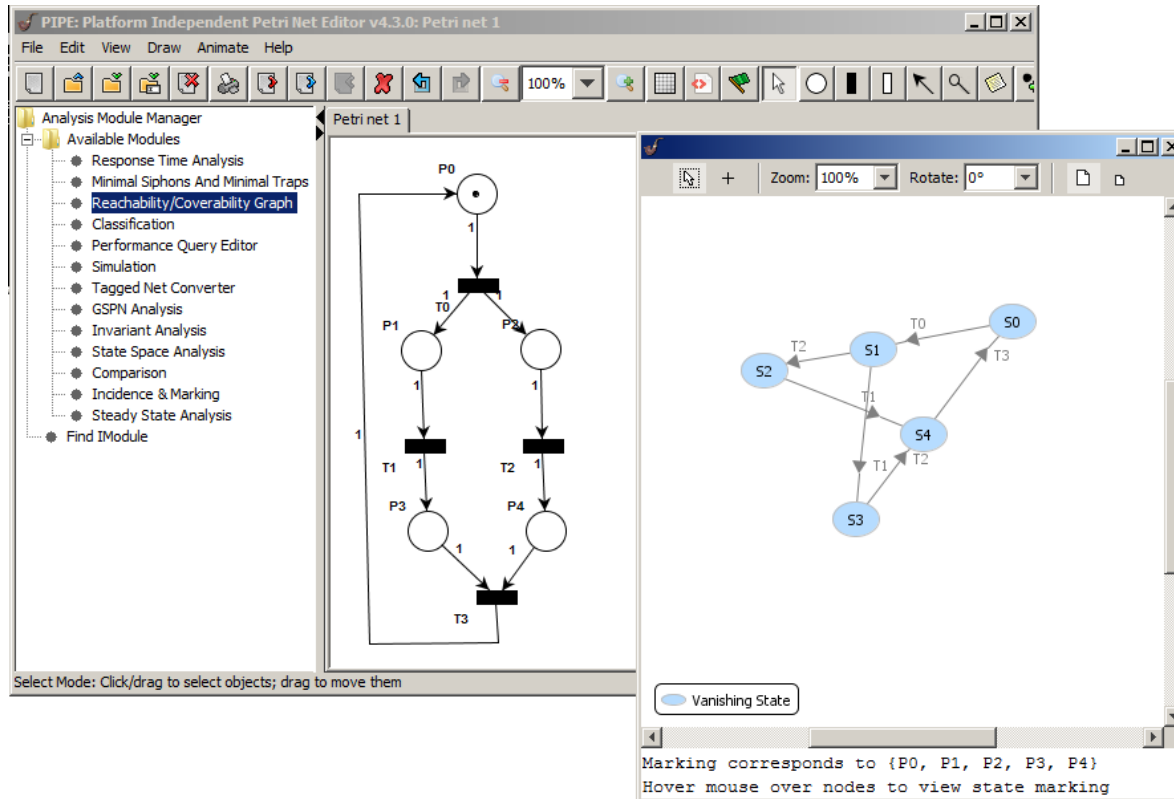
In small nets the resulting state space is usually small and therefore the reachability graph has similar number of nodes as the Petri net (or it can have even less).

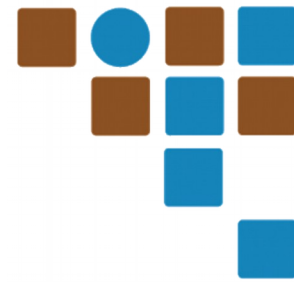
Nevertheless, this is not the usual thing to happen, since due to the great capacity of abstraction that Petri Nets have, they usually have much less nodes than the corresponding reachability graph, that may have hundreds or thousands nodes.

2. Introduction to Petri Nets

2.2. Analysis using the reachability graph

The PIPE software





2. Introduction to Petri Nets

2.2. Analysis using the reachability graph

Analysis method

A) Reversibility:

A Petri Net is reversible iff the reachability graph is strongly connected.

B) Deadlocks:

A Petri Net has a deadlock if the reachability graph has a node without descendants

C) Liveness:

A Petri Net is live iff any transition labels at least an arc of every strongly connected component of the reachability graph (when the net is reversible it has a single strongly connected component).

D) Boundness:

A Petri Net is bounded if the reachability graph is finite. A condition is evaluated at each step during the construction of the reachability graph. When this condition is met it determines that the Petri net is not bounded. The condition is the following:

There a marking M_j that can be reached from M_i such that $M_j \geq M_i$ and $M_i \neq M_j$.

Contents

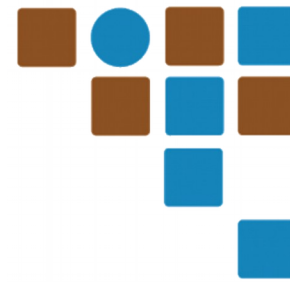
1. Introduction to Petri Nets

1. The Petri Nets formalism
2. Dynamics
3. Petri Nets for the modeling of FMS
4. Classification

2. Qualitative analysis

1. Qualitative properties
2. Analysis using the reachability graph
3. Analysis using reduction rules





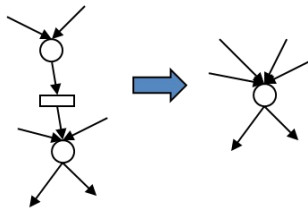
2. Introduction to Petri Nets

2.3. Analysis using reduction rules

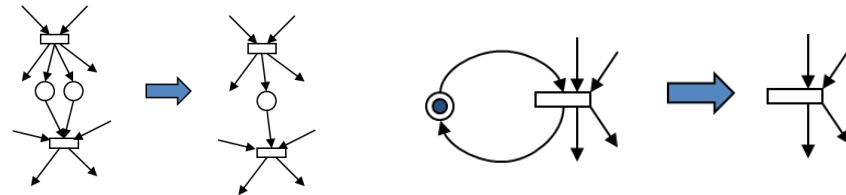
It is based in applying iteratively a set of rules such that they result in nets with less places and transitions but that preserve the qualitative properties.

The set of rules is applied until the net cannot be reduced any more. Then, the analysis of the qualitative properties is trivial or the reachability graph is build and analyzed.

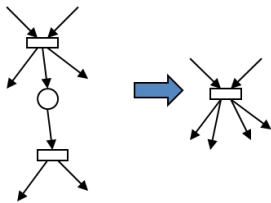
A) Fusion of places



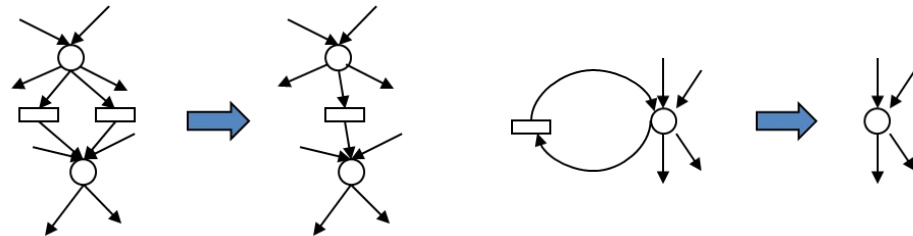
C) Identical (or implicit) places



B) Fusion of transitions



D) Identical (or implicit) transitions



2. Introduction to Petri Nets

2.3. Analysis using reduction rules

The PIPE software

