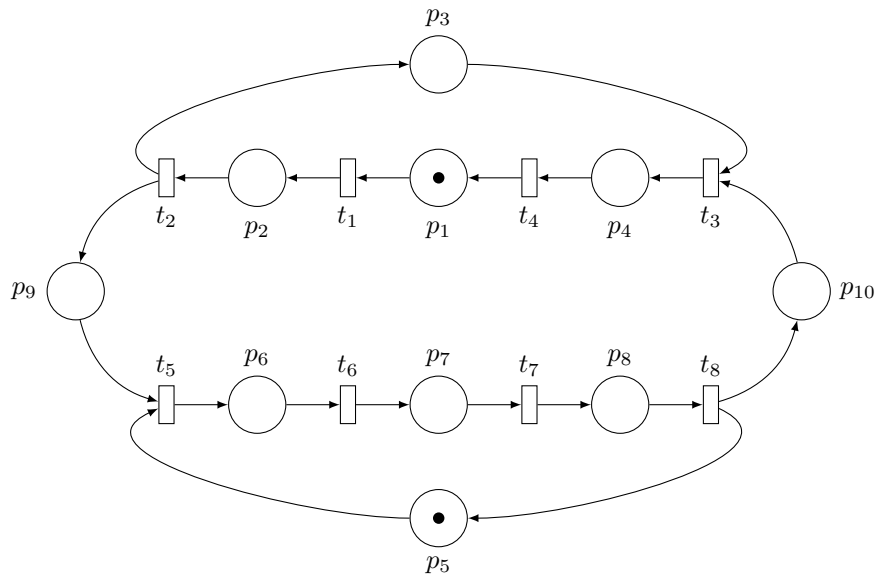


# Discrete Event Systems

## Exercise Sheet 12

### 1 Structural Properties of Petri Nets and Token Game

Given is the following Petri net  $N_1$ :

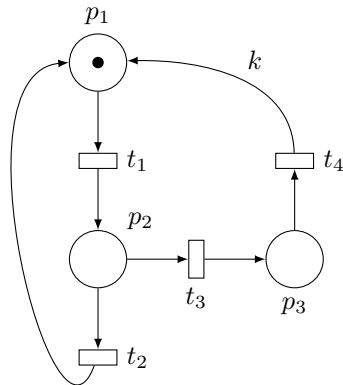


- What are the input ( $Pre$ ) and output ( $Post$ ) sets of transitions  $t_5$  and  $t_8$ ? Of place  $p_3$ ?
- Starting from the initial marking, which transitions are enabled after  $t_1$  and  $t_2$  have fired?
- What is the total number of tokens in  $N_1$  before and after  $t_2$  fired?
- Play the token game for  $N_1$  and construct the reachability graph.

**Hint** You may denote the states in such a way that the index indicates the places that hold a token in this state, for example  $\vec{s}_0 = (1, 0, 0, 0, 1, 0, 0, 0, 0, 0) \triangleq s_{1,5}$ .

## 2 Basic Properties of Petri Nets

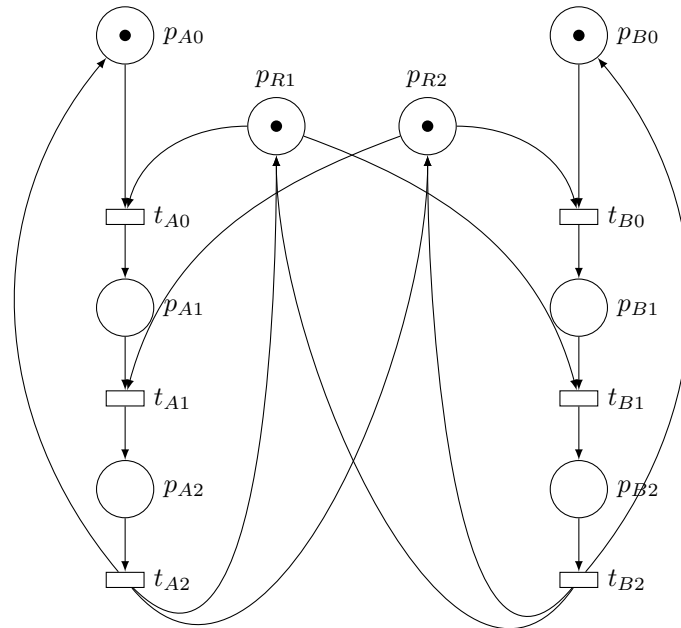
Given is the following Petri net  $N_2$ :



Explain the terms *boundedness* and *deadlock-freeness* using this example, i.e. for which values of  $k \in \mathbb{N}$  is the Petri net  $N_2$  bounded/unbounded and not deadlock-free?

### 3 Identifying a deadlock

The following Petri net  $N_3$  describes two linear processes ( $P_{A0/A1/A2}$  and  $P_{B0/B1/B2}$ ) sharing resources  $R_1$  and  $R_2$ .



In the following, use  $M = (P_{A0}, P_{A1}, P_{A2}, P_{R1}, P_{R2}, P_{B0}, P_{B1}, P_{B2})$  as marking vector and  $T = (t_{A0}, t_{A1}, t_{A2}, t_{B0}, t_{B1}, t_{B2})$  as firing vector.

- Construct the reachability graph of net  $N_3$  for the given initial marking. Explicit one firing sequence leading to a blocking marking (i.e., to a deadlock). What is the blocking marking you obtain?
- Express the upstream ( $W^-$ ) and downstream ( $W^+$ ) incidence matrices and deduce the incidence matrix  $A$ . Using  $A$ , compute the marking reached after firing  $t_{A0}t_{B0}$  from the initial marking.
- Using the upstream incidence matrix  $W^-$ , how can you **prove** that the marking obtained in the previous question represents a deadlock state?
- Suggest a modification to this Petri net which allows the two linear processes  $P_A$  and  $P_B$  to run "as intended in the first place" (i.e. without blocking).

**Hint** You don't need more than one place and a few arcs.

## 4 From mutual exclusion to starvation

In this exercise, your task is to design a Petri net that models a system in which two processes ( $A$  and  $B$ ) want to access a common exclusive resource, similarly as in Exercise 3. This means that the two processes have to exclude each other mutually from the concurrent access to the resource (e.g. a critical program section). In computer science, this concept is often called a *mutex*. Here is the desired behavior of the system:

1. One process executes its program.
2. In order to enter the critical section, the mutex value must be 1 (i.e. the mutex is available).
3. If this is the case, the process sets the mutex to 0 and executes its critical section.
4. When it is done, it resets the mutex to 1 and enters an uncritical section.
5. It loops back to start.

a) Propose a Petri net representing the desired behavior.

**Hint** Use 5 places and 4 transitions.

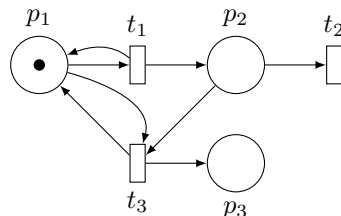
b) In this setting, it may happen that one process starves the other. That means one process always uses the resource and the other never enters its critical section (e.g. this may happen when using fixed-priority scheduling). Adapt your model such that process  $A$  cannot execute more than two times more than  $B$ . This may result in having only one process able to start running from the initial marking.

c) What is the problem of such approach? What would you suggest as a specification for the scheduling of processes  $A$  and  $B$ ?

d) **Bonus** You can try to express your specification using a Petri Net.

## 5 Coverability tree and graph

Given is the following Petri net  $N_5$ , compute its coverability tree and coverability graph. Deduce which are the unbounded places of this net given the initial marking.



## 6 Reachability Analysis for Petri Nets

In the lecture we presented an algorithm to perform a reachability analysis on Petri nets.

a) Why is it not possible with a reachability algorithm to determine *in general*, whether a given state in a Petri net is reachable or not? To which extend the notion of *coverability* helps?

b) Consider the Petri net  $N_2$  from exercise 2. Is the state  $s = (p_1 = 101, p_2 = 99, p_3 = 4)$  reachable from the initial state  $s_0 = (1, 0, 0)$  if  $k = 2$ ? Prove your answer, either by giving a suitable firing sequence, or a justification for non-reachability.

**Hint** Start with incidence matrix of  $N_2$  and express the necessary condition (presented in the lecture) for the reachability of the marking  $s$  from  $s_0$ . Eventually, discuss whether or not the marking is reachable.