Discrete and Continuous Dynamical Systems – tutorial
Petri net models

1 Formal description of Petri net models

Let us given the following Petri net with its graphical description

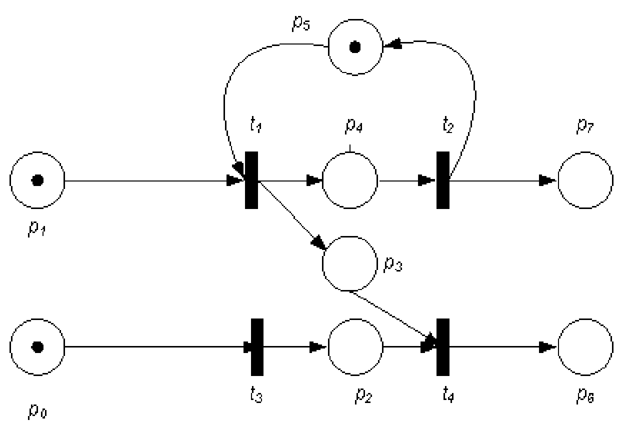


Fig. 1:

The following tasks should be carried out

1. Construct the formal description
2. Give the marking vector μ_0 shown in the figure.
3. Construct the marking sequence starting from μ_0 that describes the operation of the Petri net.

1.1 Solution

1. Construct the formal description

   - The formal definition of a Petri net is $PN = (P, T, I, O)$.
   - In order to give the set of places P, we simply list the 8 places of the Petri net: $P = \{p_0, p_1, p_2, p_3, p_4, p_5, p_6, p_7\}$
   - In order to give the set of transitions T we simply list the 4 transitions of the Petri net: $T = \{t_1, t_2, t_3, t_4\}$
   - In order to define the input function I of the transitions, we need to list the input places for all transitions. Look at Figure 1 and find the input places of the transitions.

   $$\begin{aligned} I(t_1) &= \{p_1, p_5\} \\ I(t_2) &= \{p_4\} \\ I(t_3) &= \{p_0\} \\ I(t_4) &= \{p_2, p_3\} \end{aligned}$$

   - The output function can be defined similarly to the input function.

   $$\begin{aligned} O(t_1) &= \{p_3, p_4\} \\ O(t_2) &= \{p_5, p_7\} \\ O(t_3) &= \{p_2\} \\ O(t_4) &= \{p_6\} \end{aligned}$$

2. Give the marking vector μ_0 shown in the figure.

   - In the marking vector we list the number of tokens in the order of the places. We have 1 token on places p_0, p_1 and p_5 and 0 token on the other places. You can see the number of tokens in a tabular form below.

place	p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7
# tokens	1	1	0	0	0	1	0	0

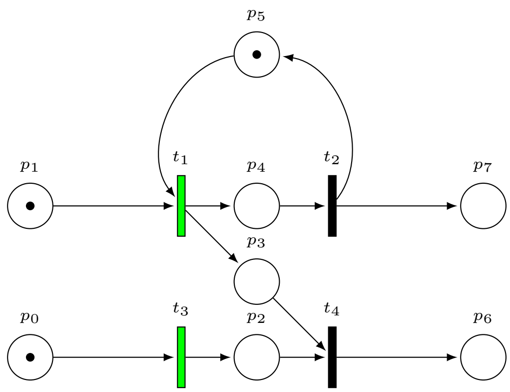
   The marking vector is $\mu_0 = [1, 1, 0, 0, 0, 1, 0, 0]^T$

3. Construct the marking sequence starting from μ_0 that describes the operation of the Petri net.

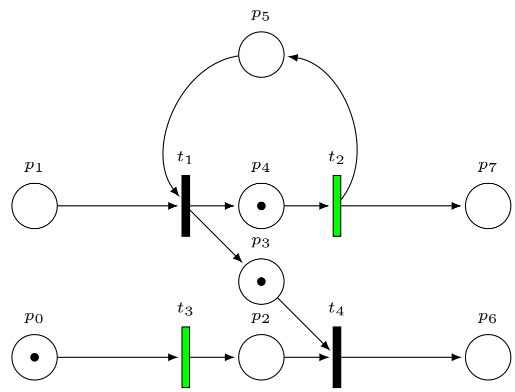
- For the sake of simplicity, only one possible marking sequence from all potential marking sequences is given here.
- The initial marking is $\mu_0 = [1, 1, 0, 0, 0, 1, 0, 0]^T$.
- Initially t_1 and t_3 are both enabled. It can be seen that they do not affect each other, therefore they are concurrent transitions. Let's start the firing with t_1. The resulted marking is $\mu_1 = [1, 0, 0, 1, 1, 0, 0, 0]^T$
- Now t_2 and t_3 are enabled. Let's continue with the firing of t_2. The resulted marking is $\mu_2 = [1, 0, 0, 1, 0, 1, 0, 1]^T$.
- Now only t_3 is enabled and after its firing we get the following marking: $\mu_3 = [0, 0, 1, 1, 0, 1, 0, 1]^T$.
- Now only $t4$ is enabled and after its firing we get the following marking: $\mu_4 = [0, 0, 0, 0, 0, 1, 1, 1]^T$. This is the final marking as there is no more enabled transition in the Petri net.
- The number of tokens on the places after each firing step is summarized in a tabular form below.

marking	p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7	transition to fire
μ_0	1	1	0	0	0	1	0	0	t_1
μ_1	1	0	0	1	1	0	0	0	t_2
μ_2	1	0	0	1	0	1	0	1	t_3
μ_3	0	0	1	1	0	1	0	1	t_4
μ_4	0	0	0	0	0	1	1	1	-

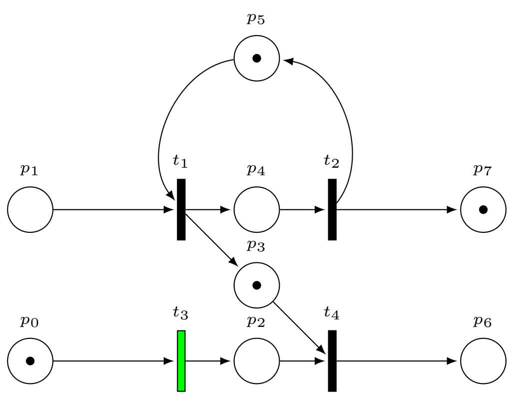
- The marking sequence can be written in the form $\mu_0[t_1 > \mu_1[t_2 > \mu_2[t_3 > \mu_3[t_4 > \mu_4$.
- The current state of the Petri net can be seen in the figures below. Enabled transitions are highlighted with green color.
- NOTE that this is only one possible marking sequence. We could start the firing with t_3 too, or choose t_3 at the second step. You can write the other marking sequences if you want to practice.


p_5
p_1 t_1 p_4 t_2 p_7
p_3
p_0 t_3 p_2 t_4 p_6

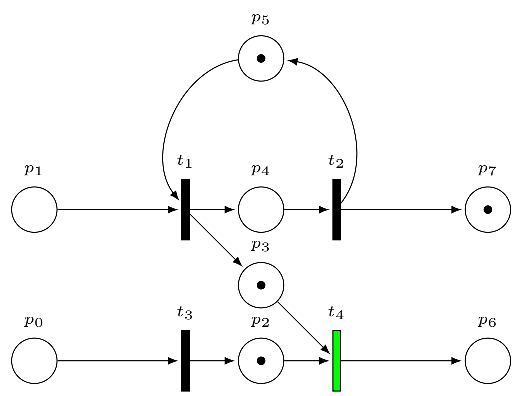
1.


p_5
p_1 t_1 p_4 t_2 p_7
p_3
p_0 t_3 p_2 t_4 p_6

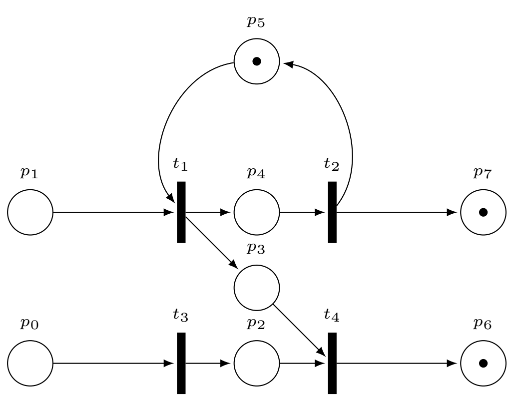
2.


p_5
p_1 t_1 p_4 t_2 p_7
p_3
p_0 t_3 p_2 t_4 p_6

3.


p_5
p_1 t_1 p_4 t_2 p_7
p_3
p_0 t_3 p_2 t_4 p_6

4.


p_5
p_1 t_1 p_4 t_2 p_7
p_3
p_0 t_3 p_2 t_4 p_6

5.

2 Construction of Petri net models

Consider a simple coffee making automaton that prepares a cup of coffee for a given coin.

Its operation steps are as follows.

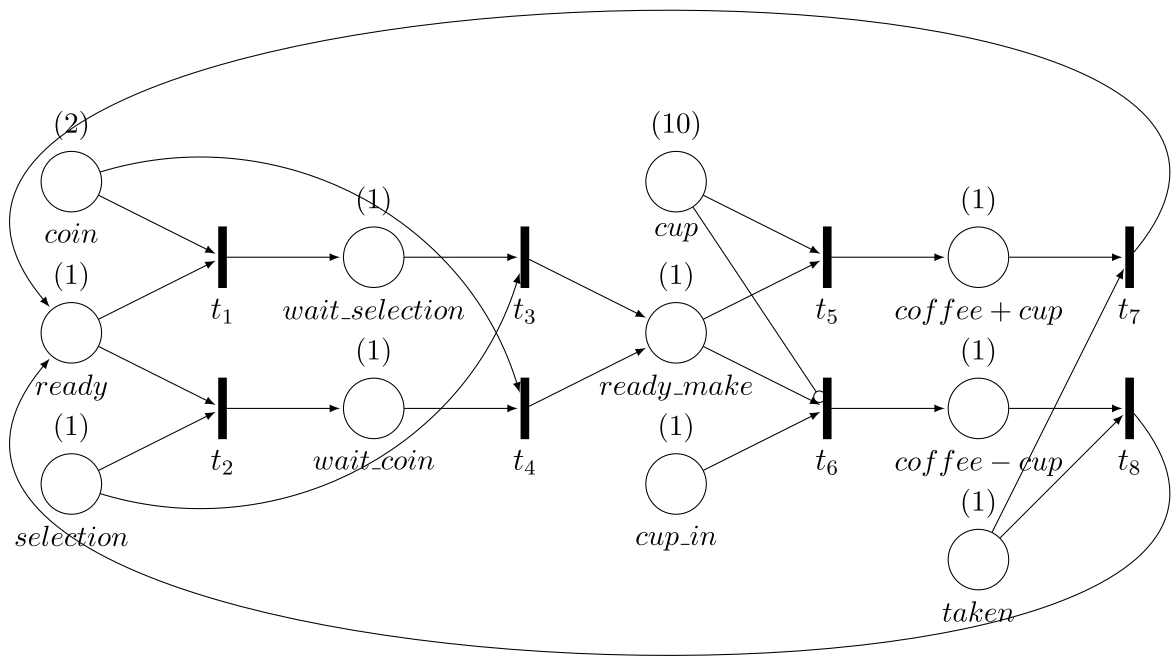
1. Make the selection of the coffee type and insert the coin (in arbitrary order).
2. If the automaton does not have plastic cup available, put your own cup.
3. Take your coffee.

TASK: Construct the graphical Petri net model of the automaton.

2.1 Solution

A possible solution of the task is the following. We need to determine the **states** and the *actions* of the coffee machine, and give the conditions and consequences for each action.

- To start the operation the following conditions must be true: the machine is **ready**, a **coin is inserted**, and the **selection button is pressed**. The order of the coin insertion and the press of the button is arbitrary.
- If the coin is inserted first then the machine *goes into an internal state* **waiting for the button**.
- If the selection button is pressed first then the machine *goes into an other internal state* **waiting for the coin**.
- If the coin is inserted and the button is pressed, then the machine **goes into the next state** which is **ready to make coffee**. Here two scenarios may occur.
  - The machine **has cups**. In this case the machine *gives* a **coffee and a cup**
  - The machine does not have cups therefore we need to **insert a cup**. In this case the machine *makes* **coffee without a cup**.



- Important that the machine use its own cups by default.

- The machine *returns* to the **ready** state after the coffee is **taken** from its output.

A graphical Petri net of the coffee automaton van be seen below. Note that the places have capacities given between round brackets.

The places and their meanings are given here:

- *ready*: machine is ready, capacity: 1
- *coin*: the coin inserted, capacity: 2 - at most 2 coins can be inserted at the same time
- *selection*: the selection button is pressed, capacity: 1
- *wait_selection*: the machine is waiting for the press of the selection button when the coin is already inserted, capacity: 1

- *wait_coin*: the machine is waiting for the insertion of the coin when the selection button is already pressed, capacity: 1
- *ready_make*: the coin is inserted and the button is pressed therefore the machine is ready to make coffee
- *cup*: available cups in the coffee automaton, capacity: 10 - at most 10 cups are in the machine
- *cup_in*: insertion of your own cup, capacity: 1
- $coffee + cup$: the machine gave you the coffee and a cup, capacity: 1
- $coffee - cup$: the machine gave you the coffee in your own cup, capacity: 1
- *taken*: the coffee made by the machine is taken from the machine, capacity: 1

Some remarks on the structure of the Petri net:

- As you can see t_1 and t_2 are conflicted transitions. It means that the coin is inserted at the same time as the selection button is pressed (which is very unlikely to happen in reality), the machine decides randomly which action is executed first.
- *cup* is connected to t_6 with an *inhibitor edge* to prevent the use of your own cup, when cups are available in the machine.

3 Homework:

(a) Consider the graphical description of your Petri net given by the png file named after your Neptun ID.

1. Construct the formal description
2. Give the marking vector μ_0 shown in the figure.
3. Construct the marking sequence starting from μ_0 that describes the operation of the Petri net.

(*) **(Supplementary)**
Give the action sequence which is necessary to operate a lift in a two-storied building (ground floor - first floor)!

1. Design the Petri net of the action sequence!
2. Give the initial state(s)!
3. What are the possible final states?

Deadline of submission: 2021.05.08. 10am