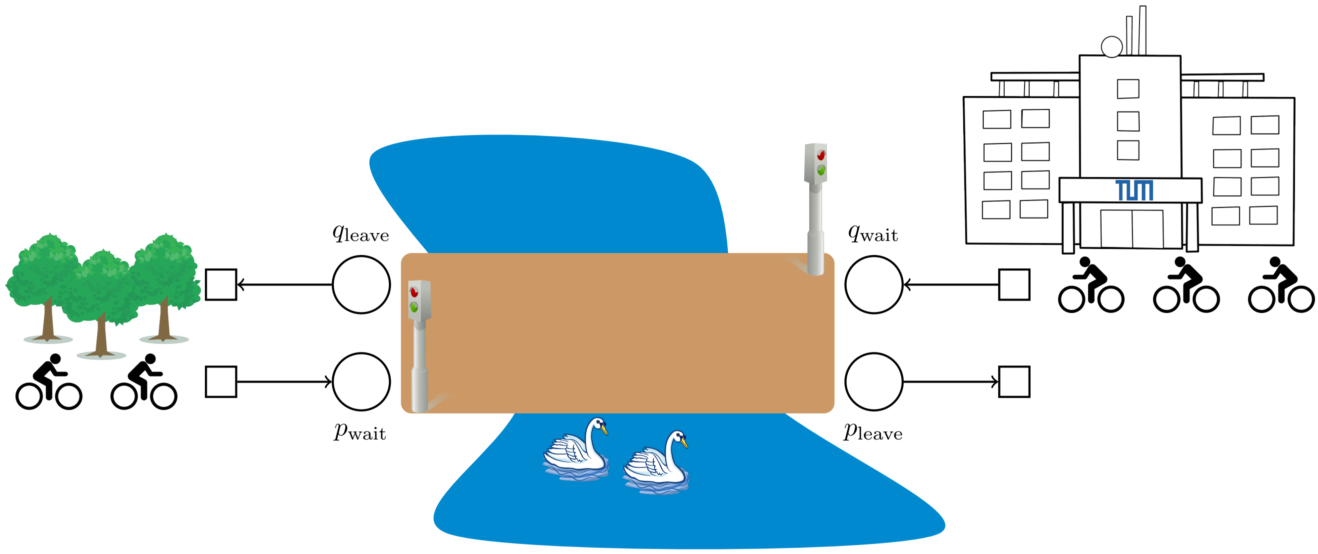
Technische Universität München, I7
Prof. J. Esparza / Dr. M. Blondin

Summer term 2018
10.04.2018

Petri nets — Exercise sheet 1

Due 18.04.2018

Exercise 1.1 (adapted from [1, ex. 2.22])
Consider a new (fictive) bridge connecting TUM to the other side of the Isar. Since this bridge is narrow, it can only be used in one direction at a time. Moreover, for safety reasons, there should not be more than six cyclists at a time on the bridge. The university wants the bridge to be equipped with a system controlling green and red lights on both ends of the bridge. For each direction, when the green light is on, cyclists are allowed to get onto the bridge; and when the red light is on, cyclists are *not* allowed to get onto the bridge.



Model the bridge as a Petri net (with weighted arcs) by extending the partial model shown above. Cyclists should flow from p_{wait} to p_{leave}, and from q_{wait} to q_{leave}. Assume that, initially, the left green light is on, the right red light is on, and the bridge is empty. Make sure that the model respects safety, i.e. that bikes are not allowed to go in opposite directions simultaneously, and that the bridge cannot hold more than six bikes.

Exercise 1.2
Consider Lamport's 1-bit mutual exclusion algorithm:

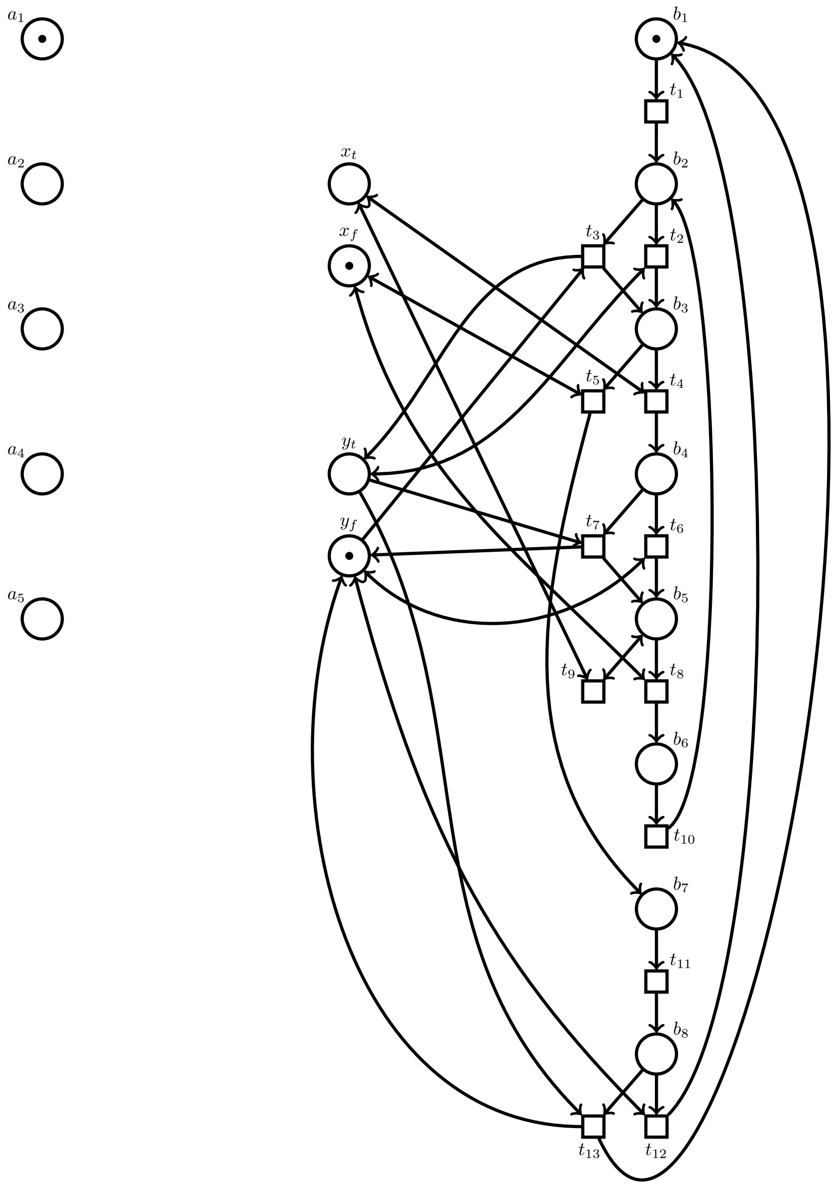
First process

```
while True:
  x = True
  while y: pass
  # critical section
  x = False
```

Second process

```
while True:
  y = True
  if x then:
    y = False
    while x: pass
    goto 2
  # critical section
  y = False
```

The algorithm can be modeled by a Petri net $\mathcal{N}$ where each program location (i.e. line of code of a process) is associated to a place, and where the shared binary variables `x` and `y` are associated to two places each. In more details, $\mathcal{N} = (P, T, F)$ where $P = \{a_1, \ldots, a_5, b_1, \ldots, b_8, x_t, x_f, y_t, y_f\}$. A token in a_i (resp. b_i) indicates that the first (resp. second) process is at line i; a token in x_t (resp. y_t) indicates that `x` (resp. `y`) has value `True`; and a token in x_f (resp. y_f) indicates that `x` (resp. `y`) has value `False`. The initial marking of $\mathcal{N}$ is $M_0 = \{a_1, b_1, x_f, y_f\}$. We give a partial Petri net that only models the second process:



(a) Complete the above Petri net $\mathcal{N}$ so that it also models the first process. You should not add new places, only transitions and arcs. Note that `pass` is a "no operation", i.e. an operation without any effect.

(b) Complete the given `APT` file for $\mathcal{N}$ accordingly, and verify whether

   (i) $(\mathcal{N}, M_0)$ is bounded;

   (ii) $(\mathcal{N}, M_0)$ is live.

(c) Complete the given `LoLA` file for $\mathcal{N}$ accordingly, and verify whether

   (i) $(\mathcal{N}, M_0)$ is deadlock-free;

   (ii) a process can be at multiple program locations at the same time;

   (iii) whether both processes can reach their critical sections simultaneously.

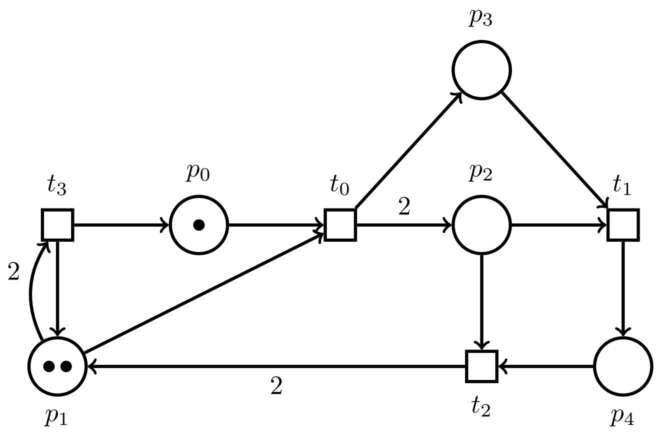
Exercise 1.3

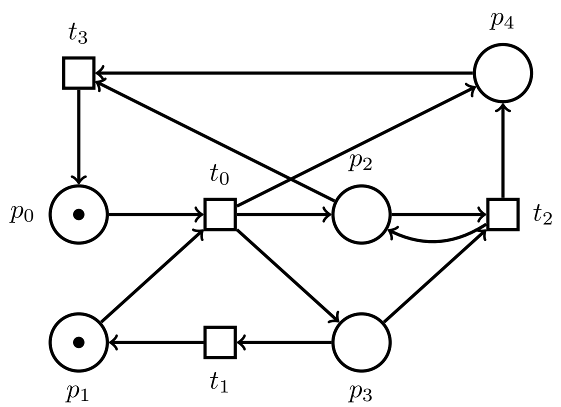
For each Petri net $(\mathcal{N}, M_0)$ below:

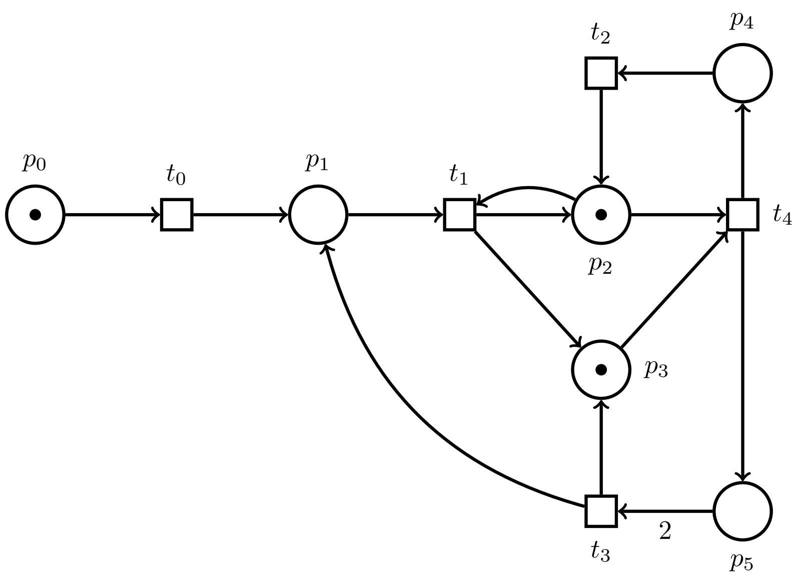
(a) construct the reachability graph of $(\mathcal{N}, M_0)$.

(b) say whether $(\mathcal{N}, M_0)$ is bounded, deadlock-free and/or live. If it is bounded, give the smallest k such that it is k-bounded. Justify your answers.

(c) give the subnet $\mathcal{N}' = (P', T', F')$ of $\mathcal{N}$ such that $P' = \{p_0, p_1, p_2, p_4\}$ and $|T'|$ is maximal.







References

[1] Wil van der Aalst, Massimiliano de Leoni, Boudewijn van Dongen, and Christian Stahl. Course business information systems: exercises, 2015. Available at `http://wwwis.win.tue.nl/~wvdaalst/old/courses/BIScourse/exercise-bundle-BIS-2015.pdf`.