Solution

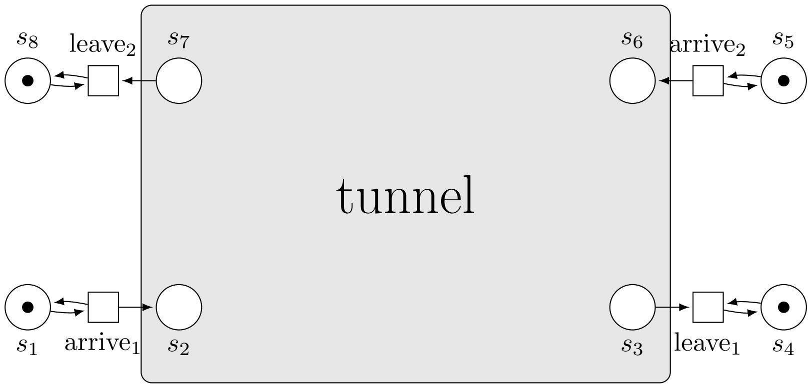
Petri nets – Homework 2

Discussed on Wednesday 18th and Thursday 19th May, 2016.

For questions regarding the exercises, please send an email to meyerphi@in.tum.de or just drop by at room 03.11.042.

Exercise 2.1 Tunnel modelling

Let us consider a tunnel connecting the TUM Campus in Garching with the Main Campus in Munich. Cars can enter the tunnel on either side and eventually exit the tunnel on the other side. A partial model of this (fictive) tunnel is given below.



(a) Extend the partial model above to a Petri net with places and transitions to keep track of the cars on each lane. Note that cars should flow from s_2 to s_3 and from s_6 to s_7. The four transitions model the environment that generates cars and absorbs cars.

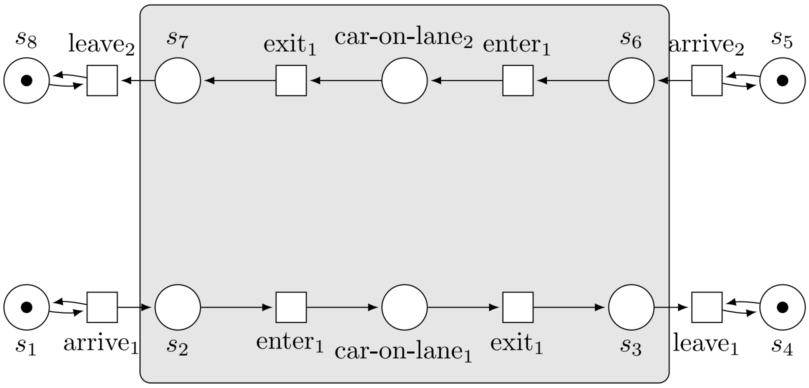
(b) The capacity of the tunnel is limited and no more than 10 cars are allowed to be in the tunnel at the same time (on both lanes together). Add this capacity constraint to your model. There should be no unnecessary waiting; for example, it cannot be the case that the tunnel remains empty while cars are queuing.

(c) As there are construction works, the tunnel is limited to one lane in one direction at time. A control system is added that can switch directions (of course only when the tunnel is empty). Modify your Petri net to incorporate this control system. Make sure that the model respects safety (no cars driving in opposite direction). However, the model does not need to be "fair".

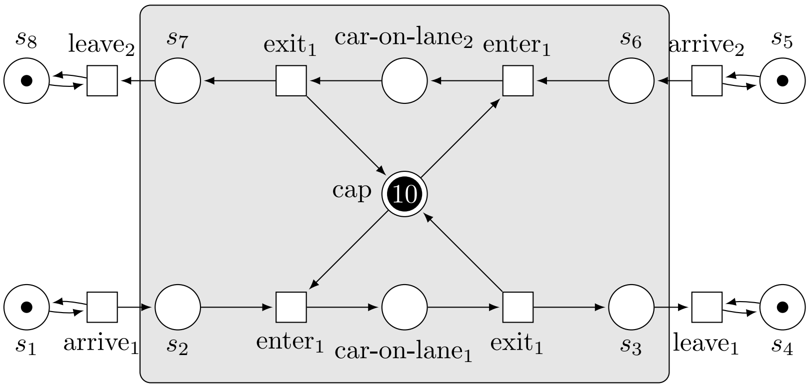
Acknowledgment: Exercise designed by Wil van der Aalst et al. (TU Eindhoven).

Solution: *(This solution is just an example. There are many possible solutions to any given modeling exercise)*

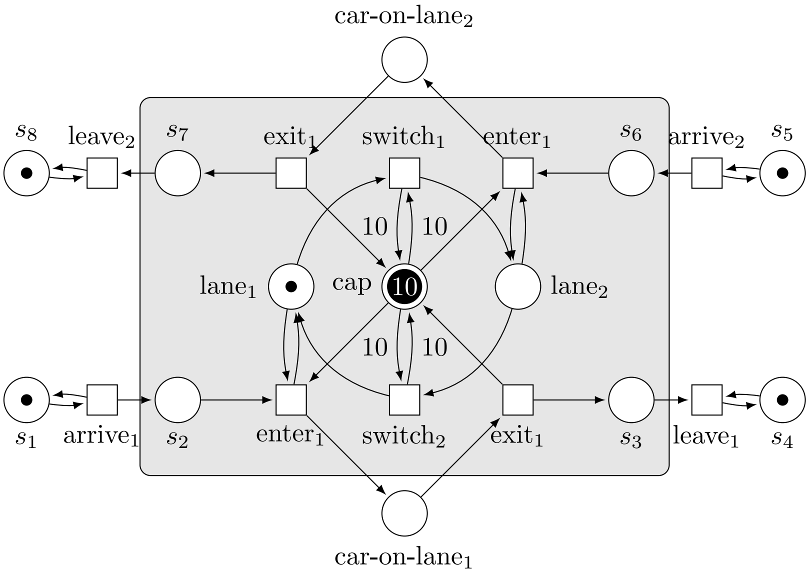
(a)



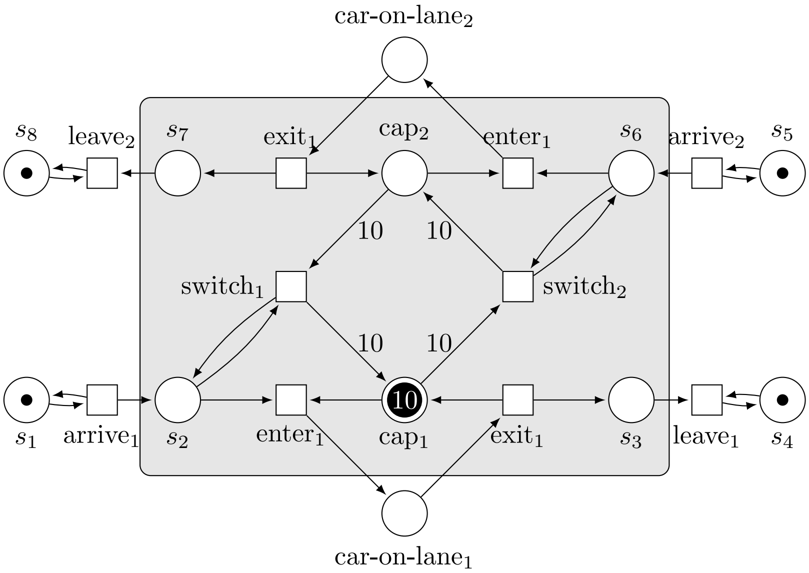
(b)



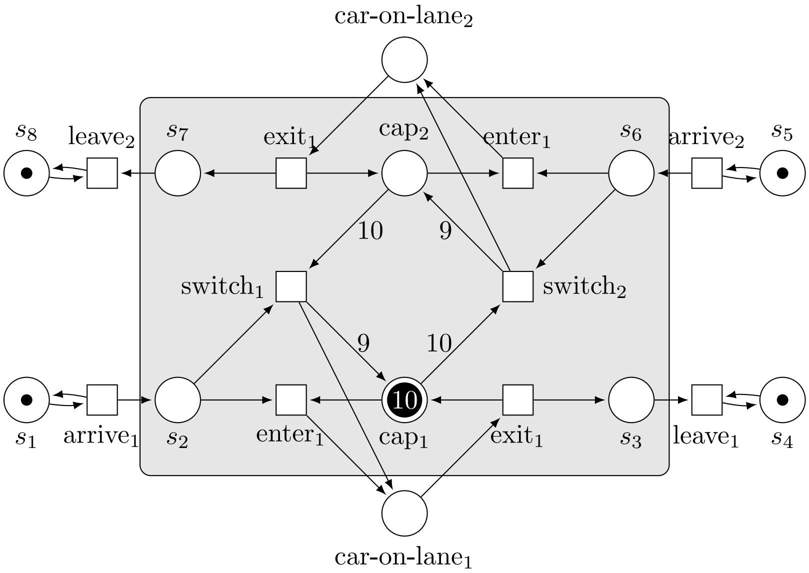
(c)



The following is an alternative solution, which only switches if there are cars waiting:

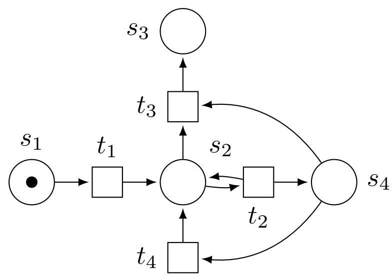


The above solution can be made to avoid unnecessary switching in the following way:

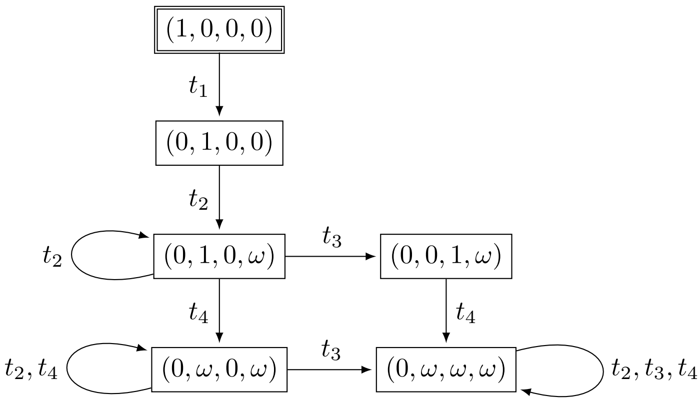


<u>Exercise 2.2</u> Coverability

Construct the coverability graph for the Petri net below.



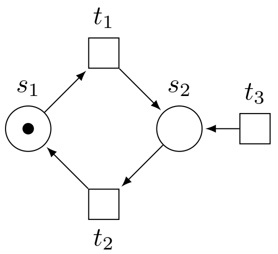
Solution: The coverability graph is as follows:



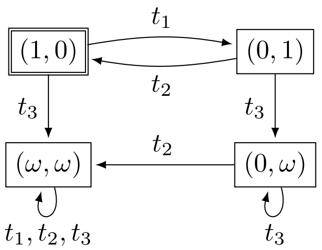
<u>Exercise 2.3</u> Uniqueness of the coverability graph

In the algorithm for the construction of the coverability graph, the search strategy (breadth-first or depth-first search and traversal order for visiting child nodes) is not specified. Show that the coverability graph obtained is not unique by exhibiting a Petri net and two different coverability graphs for this Petri net obtained by the algorithm with different search strategies.

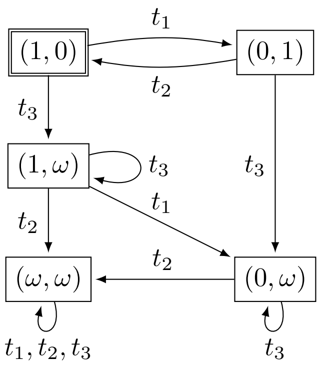
Solution: Take the following Petri net:



If, for the construction of the coverability graph, the path $M_0 = (1,0) \xrightarrow{t_1} (0,1) \xrightarrow{t_2} (1,0) \xrightarrow{t_3} (1,1)$ is first explored, then $(1,1)$ strictly covers and is reachable from $(0,1)$ and $(1,0)$, so the node (ω, ω) is added to the coverability graph. The resulting graph is:



If instead, the path $M_0 = (1,0) \xrightarrow{t_3} (1,1)$ is explored first, then $(1,1)$ only strictly covers and is reachable from $(1,0)$, so the node $(1, \omega)$ is added to the coverability graph, resulting in the following graph:

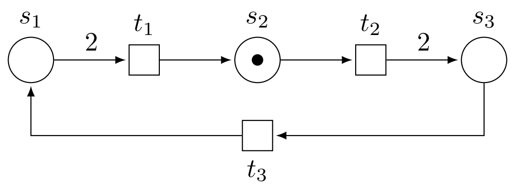


Exercise 2.4 Reachability in Petri nets with weighted arcs

Reduce the reachability problem for Petri nets with weighted arcs to the reachability problem for Petri nets without weighted arcs.

For that, describe an algorithm that, given a Petri net with weighted arcs $N = (S, T, W, M_0)$ and a marking M, constructs a Petri net $N' = (S', T', F', M_0')$ and a marking M' such that M is reachable in N if and only if M' is reachable in N'. The algorithm should run in polynomial time (you may assume unary encoding for the weights in the input, although it is also possible with a binary encoding).

Apply the algorithm to the Petri net below with the target marking $M = (2, 0, 0)$ and give the resulting Petri net N' and marking M'.



Solution: We use the following approach: We replace each place in the Petri net with weighted arcs with a ring of places in the Petri net without weighted arcs. The size of the ring is given by the maximum input or output weight, and the sum of the tokens in the ring represent the number of tokens in the original place. The tokens can move around freely in the ring, and a transition with a weighted arc that puts or takes k tokens into or out of the original place is now connected with unweighted arcs to k of the places in the ring.

Formally, let $S = \{s_1, \ldots, s_n\}$ be the places of the Petri net N. For each $s_i \in S$, define an integer k_i by

$$k_i := \max(\{W(t, s_i) \mid t \in T\} \cup \{W(s_i, t) \mid t \in T\})$$

The places in the Petri net N' are the sets of ring places for each original place. The transitions are the original transition, plus a fresh set of ring transitions.

$$S' = \bigcup_{s_i \in S} \{s_{i,j} \mid 1 \le j \le k_i\} \qquad\qquad T' = T \uplus \{t_{s_{i,j}} \mid s_{i,j} \in S'\}$$

The flow relation connects each transition t to a number of ring places $s_{i,j}$ given by the weight between t and s_i. We also connect the ring places and transitions cyclically.

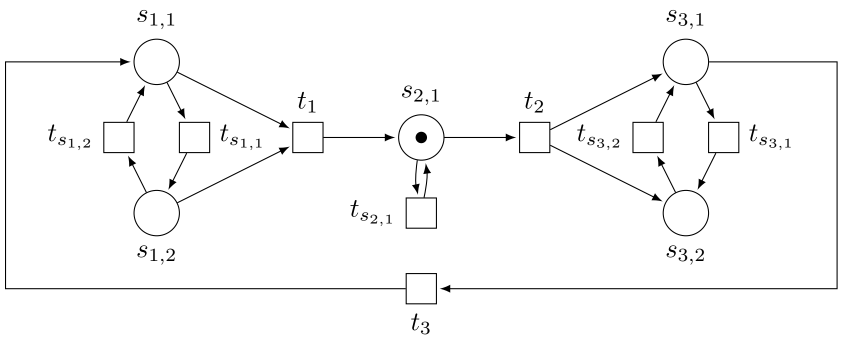
$$\begin{aligned} F' = &\{(s_{i,j}, t) \in S' \times T \mid W(s_i, t) \ge j\} \;\cup\; \{(t, s_{i,j}) \in T \times S' \mid W(t, s_i) \ge j\} \;\cup \\ &\{(s_{i,j}, t_{s_{i,j}}) \mid s_{i,j} \in S'\} \;\cup\; \{(t_{s_{i,j}}, t_{s_{i,1+(j \bmod k_i)}}) \mid s_{i,j} \in S'\} \end{aligned}$$

The initial and target marking are given by having all the tokens in the first place of each ring.

$$M_0'(s_{i,j}) = \begin{cases} M_0(s_i) & \text{if } j = 1 \\ 0 & \text{otherwise} \end{cases} \qquad\qquad M'(s_{i,j}) = \begin{cases} M(s_i) & \text{if } j = 1 \\ 0 & \text{otherwise} \end{cases}$$

By construction, if M is reachable in N, then there is a reachable marking M' in N' with $\sum_j M'(s_{i,j}) = M(s_i)$ for all $s_i \in S$. We can use the ring transitions to move all tokens in $s_{i,j}$ to $s_{i,1}$ to reach the target marking.

Applying the construction to the Petri net and marking above gives us the Petri net below, along with the marking M' given by $M'(s_{1,1}) = 2$ and $M'(s) = 0$ for all places $s \ne s_{1,1}$.



Exercise 2.5 Monotonicity of properties

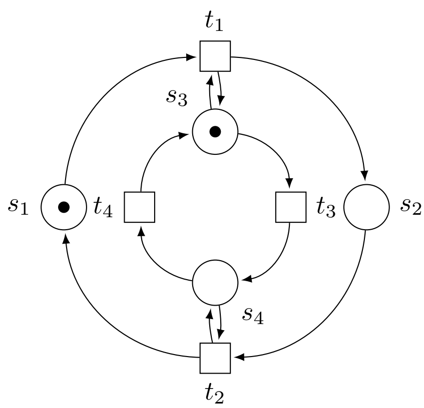
Exhibit counterexamples that disprove the following conjectures:

(a) If (N, M_0) is bounded and $M \ge M_0$, then (N, M) is bounded.

(b) If (N, M_0) is live and $M \ge M_0$, then (N, M) is live.

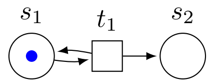
(c) If (N, M_0) is live and bounded and $M \ge M_0$, then (N, M) is bounded.

Hint: Add places and arcs to the net below to obtain a solution.

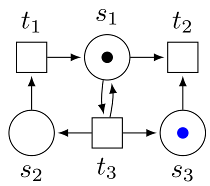


Solution:

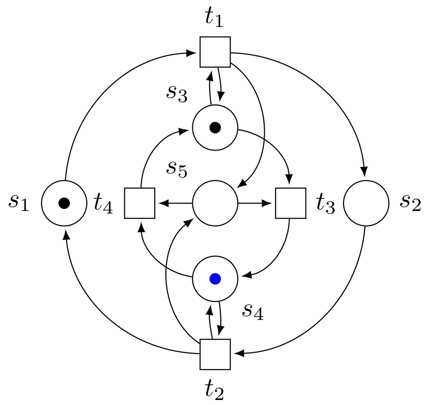
(a) The following Petri net is bounded without tokens, but not bounded with the blue token in s_1, as repeatedly firing t_1 can put an arbitrary number of tokens in s_2.



(b) The following Petri net is live with the black tokens, but not live with the additional blue token in s_3, as firing t_2 leads to a dead marking.

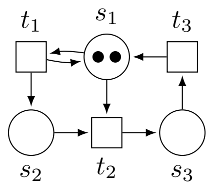


(c) The following Petri net is live and bounded with the black tokens, but not bounded with the additional blue token in s_4, as repeatedly firing $t_1 t_2$ can put an arbitrary number of tokens in s_5.



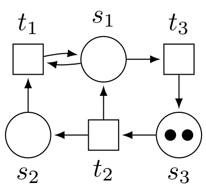
<u>Exercise 2.6</u> **Backwards reachability algorithm**

Apply the backwards reachability algorithm to the Petri net below to decide if the marking $M = (0, 0, 2)$ can be covered. Record all intermediate sets of markings with their finite representation of minimal elements.



Solution:

For the backwards reachability algorithm, it can be helpful to construct the reverse net, with all arcs inverted, to compute the predecessors of markings. Then start with the target marking M and add tokens as necessary for firing transitions.



We start with $m_0 = \{(0, 0, 2)\}$ and compute the predecessors for $(0, 0, 2)$ for each transition t, that is, the minimal marking M such that $M \xrightarrow{t} M'$ with $M' \geq (0, 0, 2)$.

$$
\begin{aligned}
pre((0,0,2), t_1) &= pre((1,1,2), t_1) = (1,0,2) \\
pre((0,0,2), t_2) &= (1,1,1) \\
pre((0,0,2), t_3) &= pre((1,0,2), t_3) = (0,0,3)
\end{aligned}
$$

After adding the new markings to m_0 and eliminating non-minimal markings, our new set is $m_1 = \{(0, 0, 2), (1, 1, 1)\}$. For the new marking $(1,1,1)$, we compute the predecessors:

$$\begin{aligned}
pre((1,1,1),t_1) &= (1,0,1)\\
pre((1,1,1),t_2) &= (2,2,0)\\
pre((1,1,1),t_3) &= (0,1,2)
\end{aligned}$$

We add the new markings, take the minimal elements and obtain $m_2 = \{(0,0,2),(1,0,1),(2,2,0)\}$. For $(1,0,1)$ and $(2,2,0)$, we compute the predecessors:

$$\begin{aligned}
pre((1,0,1),t_1) &= pre((1,1,1),t_1) = (1,0,1)\\
pre((1,0,1),t_2) &= (2,1,0)\\
pre((1,0,1),t_3) &= (0,0,2)\\
pre((2,2,0),t_1) &= (2,1,0)\\
pre((2,2,0),t_2) &= pre((2,2,1),t_2) = (3,3,0)\\
pre((2,2,0),t_3) &= (1,2,1)
\end{aligned}$$

The new minimal marking set is now $m_3 = \{(0,0,2),(1,0,1),(2,1,0)\}$. Next we compute the predecessors for $(2,1,0)$:

$$\begin{aligned}
pre((2,1,0),t_1) &= (2,0,0)\\
pre((2,1,0),t_2) &= pre((2,1,1),t_2) = (3,2,0)\\
pre((2,1,0),t_3) &= (1,1,1)
\end{aligned}$$

The new set is $m_4 = \{(0,0,2),(1,0,1),(2,0,0)\}$. For $M' = (2,0,0) \in m_4$, we have $M_0 \geq M'$, therefore we can conclude that M is coverable from M_0.

If instead we would continue, we would compute the predecessors for $(2,0,0)$:

$$\begin{aligned}
pre((2,0,0),t_1) &= pre((2,1,0),t_1) = (2,0,0)\\
pre((2,0,0),t_2) &= pre((2,0,1),t_2) = (3,1,0)\\
pre((2,0,0),t_3) &= (1,0,1)
\end{aligned}$$

No new minimal markings are obtained, so we would have reached a fixpoint.

Exercise 2.7 Suffix sequence

Give a procedure to decide the following problem:

Given a Petri net (N, M_0) and a transition sequence σ, is there a transition sequence σ' such that $\sigma'\sigma$ is enabled at M_0?

For the procedure, you may use any already known decision procedures and algorithms such as coverability graph and backwards reachability, or you may adapt those algorithms or use parts of them.

Solution: Let $\sigma = t_1 t_2 \dots t_n$. Consider the following algorithm:

$m \leftarrow \{\vec{0}\}$
for i from n to 1 **do**
  $m \leftarrow \min(pre(m, t_i))$
if $\exists M \in m : M$ is coverable from M_0 **then**
  return there is a sequence σ' s.t $M_0 \xrightarrow{\sigma'\sigma}$.
else
  return there is no sequence σ' s.t $M_0 \xrightarrow{\sigma'\sigma}$.

Basically, we fire σ backwards, starting from all markings (the upward closed set with the zero marking as the minimal element). We obtain the set m of all markings where σ is enabled (again an upward closed set). We then check if one the minimal markings of the set m is coverable from M_0, and if yes, there is a sequence σ' from which we can reach an element of m, and thus fire σ.

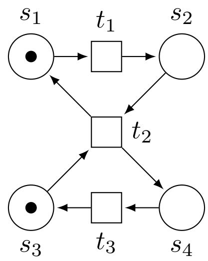
The existance of a coverable element in m can be checked with the coverability graph or the backwards reachability algorithm. With the backwards reachability algorithm, we can directly start with m as the initial set of minimal markings.

Exercise 2.8 Independence of boundedness, liveness and cyclicity

Definition 2.8.1 (Cyclic Petri nets)**.** A Petri net (N, M_0) is *cyclic* if, loosely speaking, it is always possible to return to the initial marking. Formally: $\forall M \in [M_0\rangle : M_0 \in [M\rangle$.

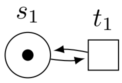
Show that the properties liveness, boundedness and cyclicity are independent of eath other by exhibiting eight Petri nets, one for each possible combination of the three properties and their negations.

Hint: A live, bounded, but not cyclic Petri net is hard to find. A possible solution can be obtained by simply adding arcs to the net below.

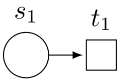


Solution: Let L,B and C be abbreviations for live, bounded and cyclic, and $\overline{L}$, $\overline{B}$ and $\overline{C}$ for their negations.

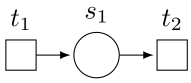
LBC



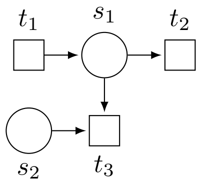
$\overline{L}BC$



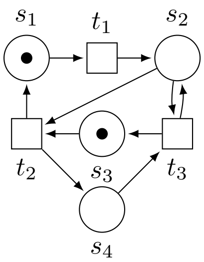
$L\overline{B}C$



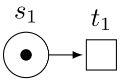
$\overline{LB}C$



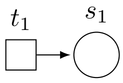
$LB\overline{C}$



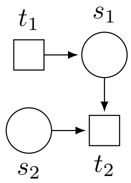
$\overline{L}B\overline{C}$



$L\overline{BC}$



$\overline{LBC}$



For most nets, it is clear that they have the desired properties. To see that the net $LB\overline{C}$ is indeed live, bounded, but not cyclic, we can construct the reachability graph, with the markings given as $M = (s_1, s_2, s_3, s_4)$: