Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Computer Engineering and
Networks Laboratory

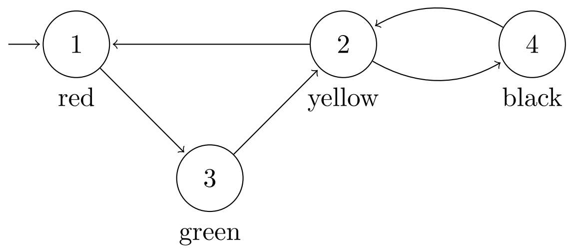
HS 2010 Dr. K. Lampka / J. Smula, T. Langner

Discrete Event Systems

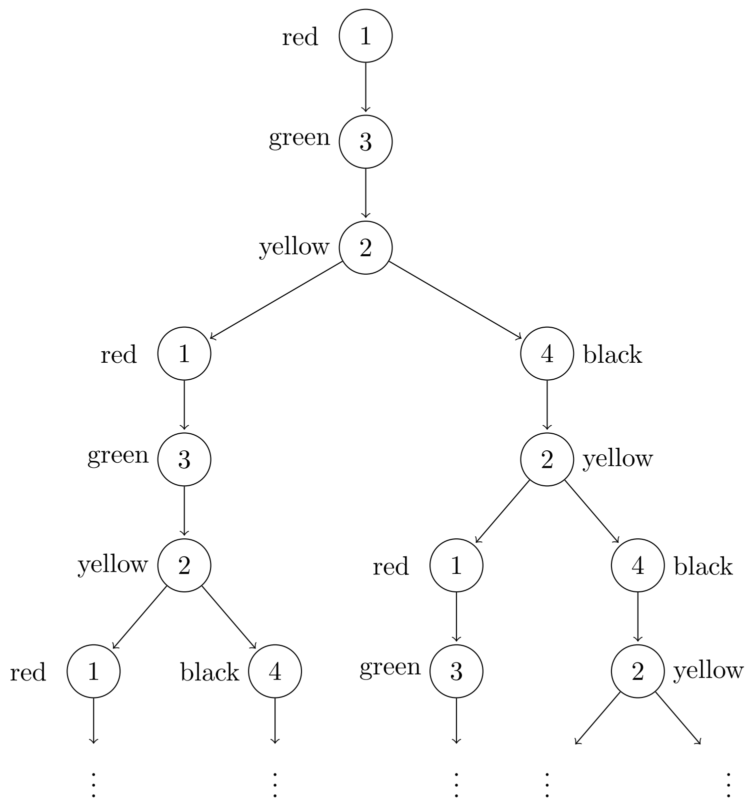
Solution to Exercise Sheet 13

1 Computation Tree Logic Model Checking

a) The graph of the Kripke structure $\mathcal{K}$ looks as follows:



b) The computation tree for the initial state s_0 upto depth 7 looks as follows:



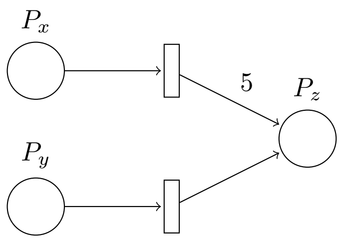
c) Ω_3 is incorrect, since the existence quantor is only defined in combination with a temporal operator $(\bigcirc, \Box, \bigcup, \Diamond)$. Here it is used with a state formula.

Ω_5 is incorrect, since the operator $\exists\bigcirc$ is only defined for a state formula. (true $\bigcup$ black), however, is a path formula.

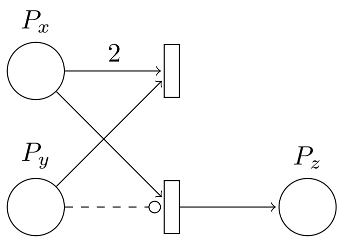
d) See solution to exercise sheet 14.

2 Petri Nets [Exam!]

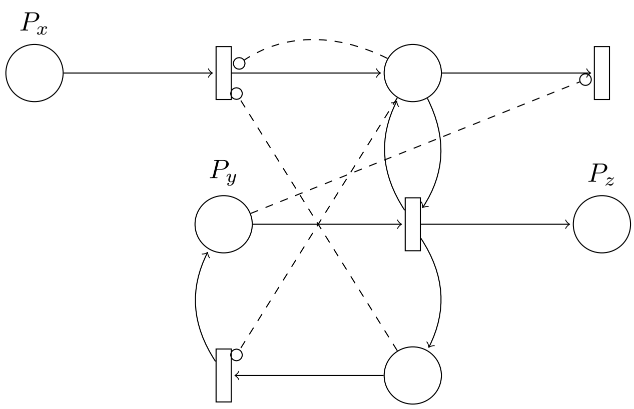
a) $f_1(x, y) = 5x + y$:



b) $f_2(x, y) = x - 2y$:



c) $f_3(x, y) = x \cdot y$:



3 Basic Properties of Petri Nets

A petri net is k-bounded, if there is no fire sequence that makes the number of tokens in one place grow larger than k. It is obvious that petri net N_2 is 1-bounded if $k \leq 1$. This holds because in the initial state there is only one token in the net, and in the case $k \leq 1$ no transition increases the number of tokens in N_2. If $k \geq 2$, the number of tokens in p_1 can grow infinitely large by repeatedly firing t_1, t_3 and t_4. So, the petri net N_2 is unbounded for $k \geq 2$.

A petri net is deadlock free if no fire sequence leads to a state in which no transition is enabled. If $k = 0$, N_2 is not deadlock-free. The fire sequence t_1, t_3, t_4 causes the only existing token to be consumed and hence, there is no enabled transition any more. For $k \geq 1$, however, no deadlock can occur.

4 Reachability Analysis for Petri Nets

a) Petri nets may possess infinite reachability graphs, e.g. N_2 with $k \geq 2$. If the state in question is actually reachable in such a petri net, the reachability algorithm will eventually terminate. If it is not reachable, the algorithm will never be able to determine this with absolute certainty (cf. halting problem).

b) We determine the incidence matrix of the petri net as explained in the lecture.

$$\mathbf{A} = \begin{pmatrix} -1 & 1 & 0 & 2 \\ 1 & -1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix}$$

We are interested in whether the state $\vec{s} = (101, 99, 4)$ is reachable from the initial state $\vec{s_0} = (1, 0, 0)$. If the equation system $\mathbf{A} \cdot \vec{f} = \vec{s} - \vec{s_0}$ has no solution, we know that the state $\vec{s}$ is not reachable from s_0. "Unfortunately",

$$\begin{pmatrix} -1 & 1 & 0 & 2 \\ 1 & -1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 100 \\ 99 \\ 4 \end{pmatrix}$$

is satisfiable. To show that $\vec{s}$ is reachable from $\vec{s_0}$, we have to give a firing sequence through which we get from $\vec{s_0}$ to $\vec{s}$. From the last equation of the above equation system, we know that $f_3 = f_4 + 4$. Hence, in the desired firing sequence, f_3 is fired four times more than f_4. However, $\vec{f}$ does not tell us about the firing order. Considering the petri net, we can see that – starting from $\vec{s_0}$ – the number of tokens in p_1 increases by one after firing t_1, t_3, and t_4 in this order. Repeating this for 203 times yields the state $(204, 0, 0)$. Firing t_1 for 103 times followed by firing t_3 for four times finally yields state $\vec{s}$.