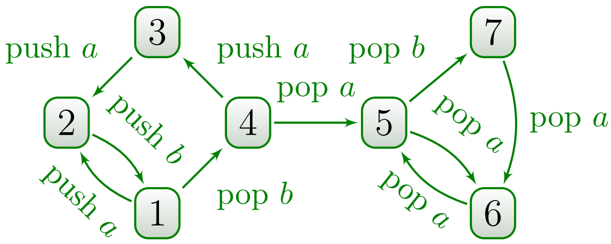
Technische Universität München (I7) **Summer Semester 2020**
J. Křetínský / P. Ashok / K. Grover / T. Meggendorfer **16.07.2019**

Model Checking – Exercise sheet 12

Exercise 12.1

Consider the pushdown system below, with stack alphabet $\Gamma = \{a, b\}$ where $1 \xrightarrow{\text{push } a} 2$, indicates the presence of transitions $1a \hookrightarrow 2aa$ and $1b \hookrightarrow 2ab$, and $4 \xrightarrow{\text{pop } a} 5$, indicates the presence of transition $4a \hookrightarrow 5$.



(a) Let $L = 7b^* = \{7, 7b, 7bb, 7bbb, \ldots\}$. Construct the $\mathcal{P}$-automaton accepting $\text{pre}^*(L)$.

(b) Give the minimal automaton accepting the language of all stacks w such that $1w \in \text{pre}^*(L)$.

Exercise 12.2

Consider the following recursive program, where `?` denotes a nondeterministic Boolean value:

```
    procedure main;
m0:   if ? then
        call a;
      else
        call b;
m1:   return;

    procedure a;
a0:   if ? then
        call b;
a1:     call b;
      else
        call a;
      end if;
a2:   return;
```

```
    procedure b;
b0:   if ? then
        call a;
b1:     if ? then
          call a;
        end if;
      end if;
b2:   return;
```

(a) Model the program with a pushdown system.

(b) Compute all configurations that can reach the program label `m1`.

Exercise 12.3

Consider the following recursive program with a global variable `g` and a local variable `l`:

```
    boolean g;

    procedure main(boolean l);
m0:   if l then
        call a;
      end if;
m1:   assert(g == l);
m2:   return;

    procedure a();
a0:   g := not g;
a1:   if not g then
        call a;
a2:     call a;
      end if;
a3:   return;
```

(a) Model the program with a pushdown system, where the values of `g` and `l` are not initialized.

(b) Compute all configurations that can reach the program label `m2`.

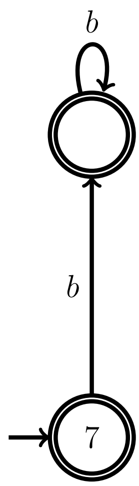
(c) ★ Compute all configurations that are reachable from the program label `m0`.

Solution 12.1

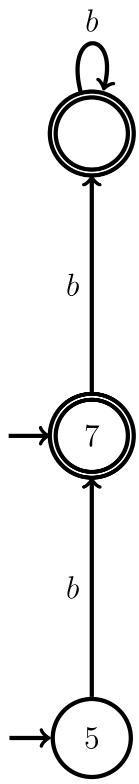
(a) First note that the transitions of the pushdown system are as follows:

$$
\begin{aligned}
1a &\to 2aa\\
1b &\to 2ab\\
1b &\to 4\\
2a &\to 1ba\\
2b &\to 1bb\\
3a &\to 2aa\\
3b &\to 2ab\\
4a &\to 3aa\\
4b &\to 3ab\\
4a &\to 5\\
5b &\to 7\\
5a &\to 6\\
6a &\to 5\\
7a &\to 6.
\end{aligned}
$$

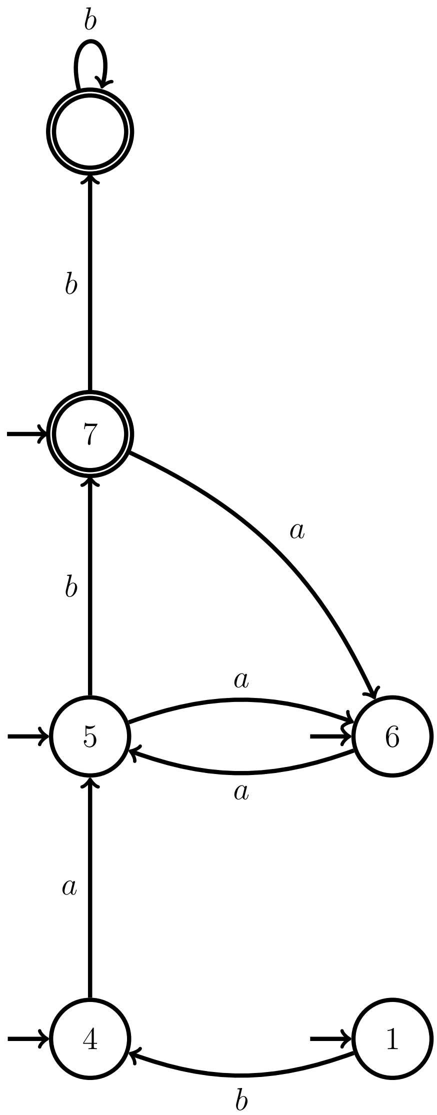
We are looking for $\mathrm{pre}^*(L)$ where $L = 7b^*$. We construct the following $\mathcal{P}$-automaton for L:



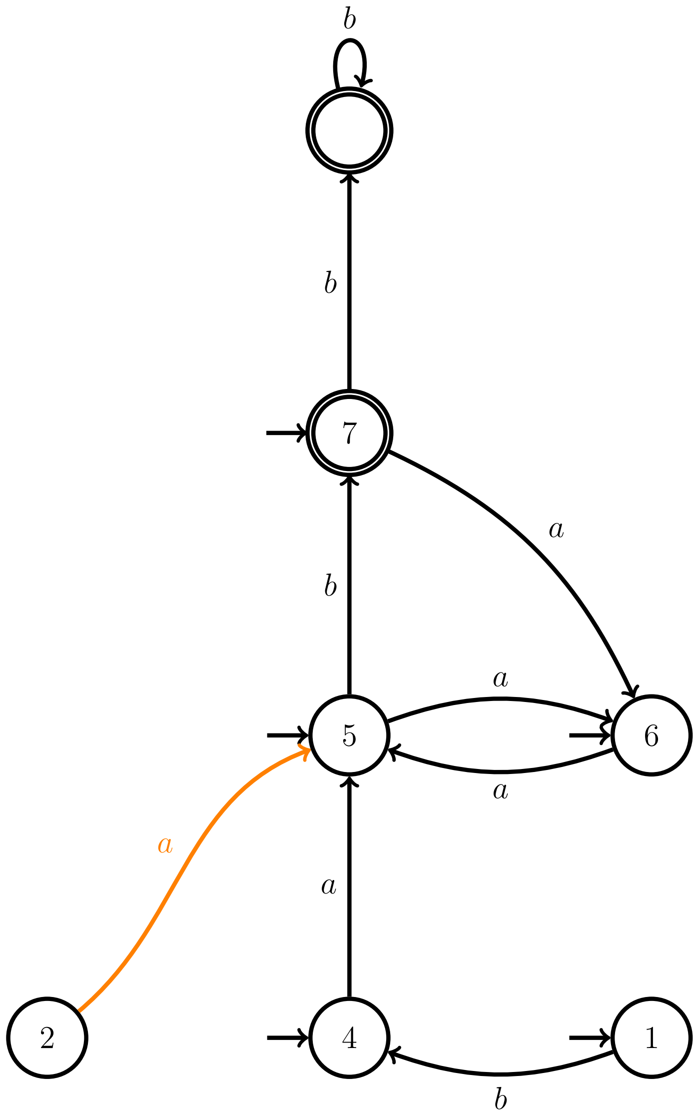
We apply the algorithm to compute $\mathrm{pre}^*(L)$ on the above automaton $\mathcal{A}$. More precisely, if $q \xrightarrow{w} r$ in $\mathcal{A}$ and if the pushdown system contains a rule $pA \to qw$, then we add a transition $p \xrightarrow{A} r$ to $\mathcal{A}$. For example, this is the case for $7 \xrightarrow{\varepsilon} 7$ and $5b \to 7$, so we add the transition $5 \xrightarrow{b} 7$:



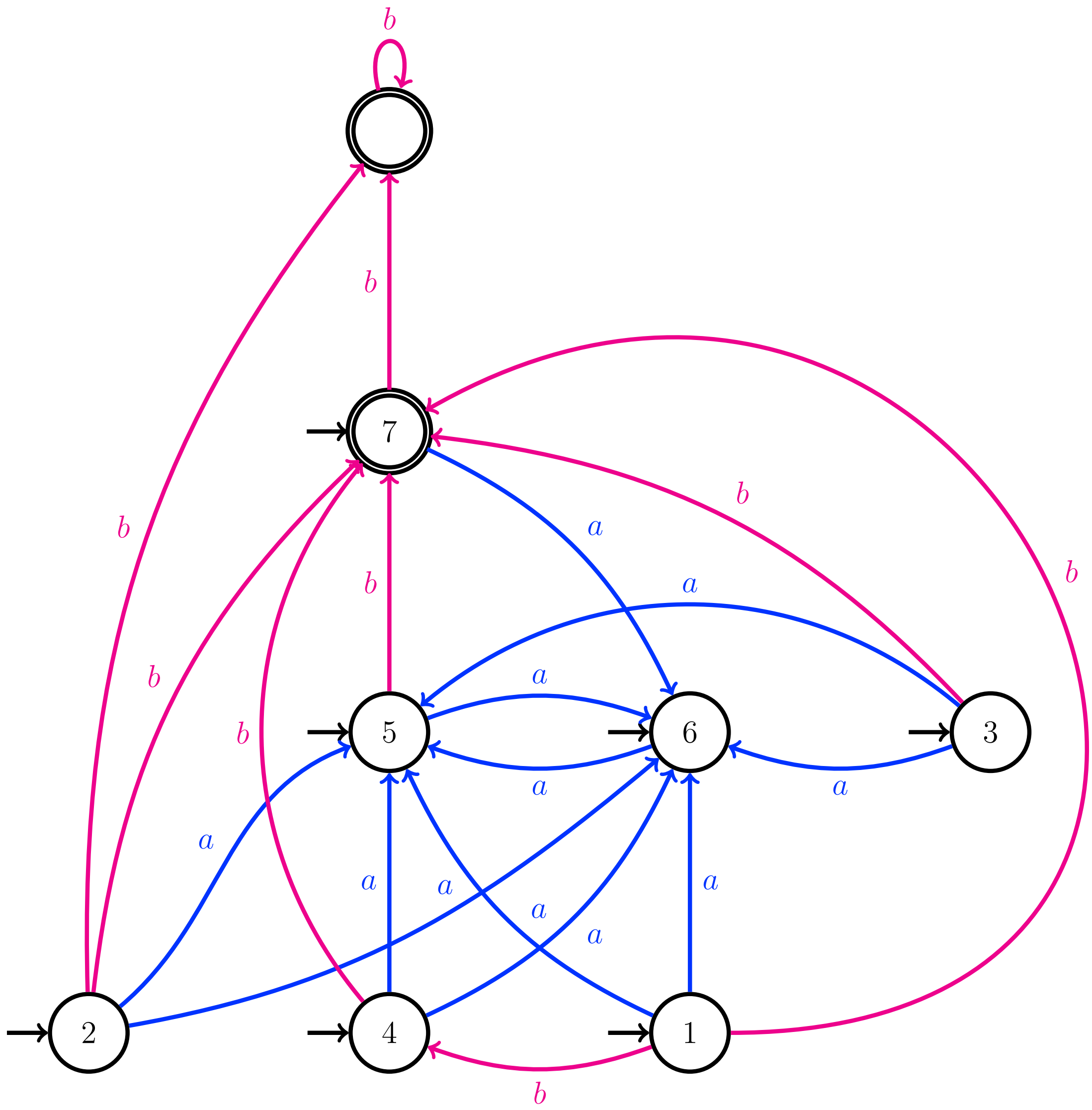
By repeatedly doing the same with rules $4a \to 5$, $6a \to 5$, $5a \to 6$, $7a \to 6$ and $1b \to 4$, we obtain:



From rule $2a \to 1ba$, we add the following (orange) transition:

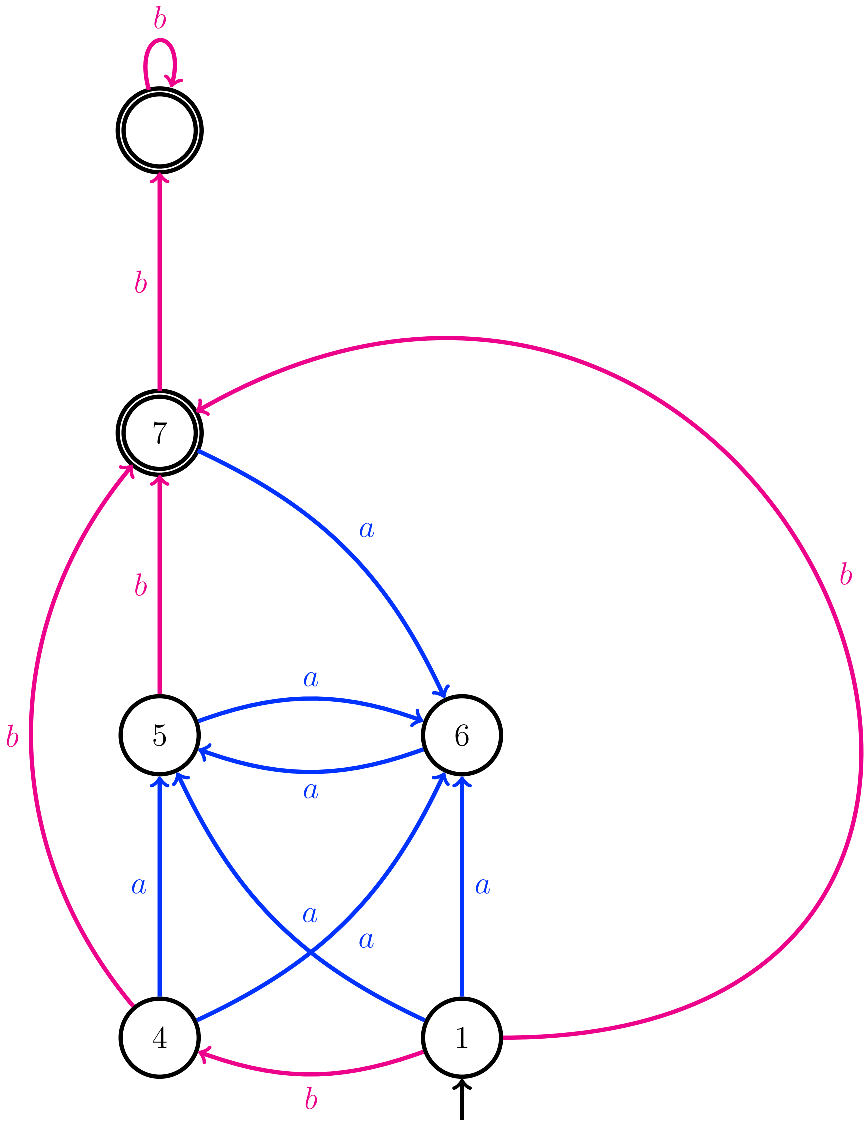


We repeat the process until we derive the following $\mathcal{P}$-automaton[1]:

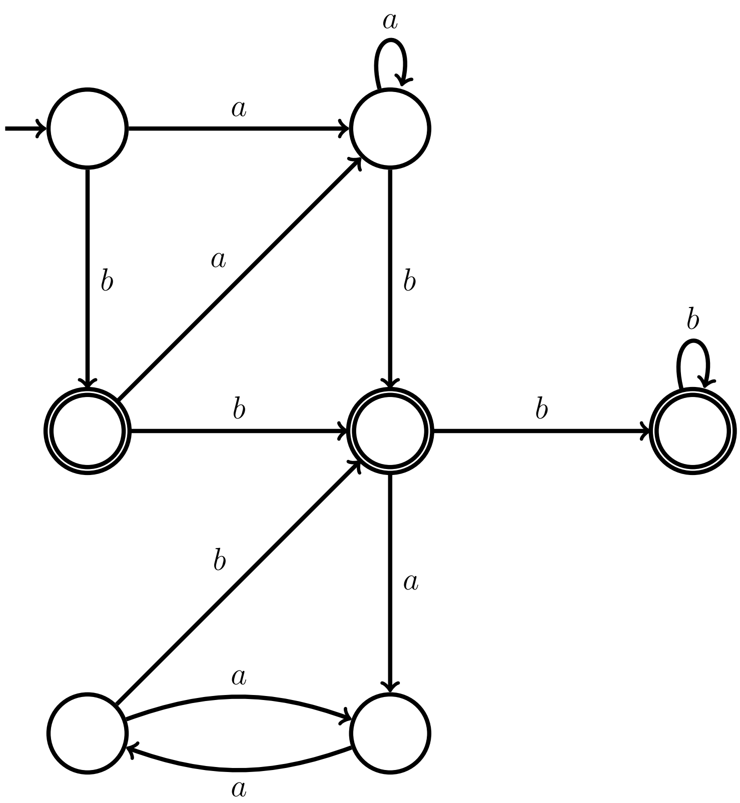


[1]Blue and magenta are only used to help distinguishing a and b-transitions.

(b) We are interested in the language accepted by the $\mathcal{P}$-automaton obtained in (a) starting from control-state 1. By removing the control-states which are non reachable, we obtain the following automaton:



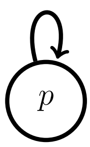
By determinizing and minimizing the above automaton, we derive:



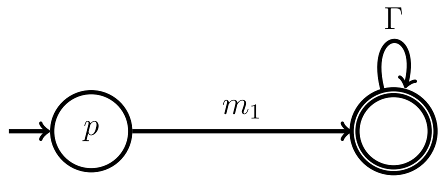
Solution 12.2

(a) Since the program has no global variable, the pushdown system has a single control-state, say p. The stack alphabet is $\Gamma = \{m_0, m_1, a_0, a_1, a_2, b_0, b_1, b_2\}$. The resulting pushdown system is:

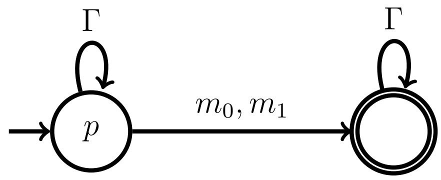
$$
\begin{array}{lcl}
m_0 & \to & a_0 m_1 \mid b_0 m_1 \\
m_1 & \to & \varepsilon \\
a_0 & \to & b_0 a_1 \mid a_0 a_2 \\
a_1 & \to & b_0 a_2 \\
a_2 & \to & \varepsilon \\
b_0 & \to & a_0 b_1 \mid b_2 \\
b_1 & \to & a_0 b_2 \mid b_2 \\
b_2 & \to & \varepsilon
\end{array}
$$



(b) We are looking for $\text{pre}^*(L)$ where $L = p\, m_1 \Gamma^*$. We construct the following $\mathcal{P}$-automaton for L:



By applying the algorithm to compute $\text{pre}^*(L)$, we derive the following $\mathcal{P}$-automaton:

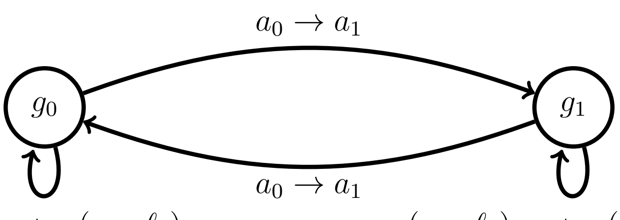


Solution 12.3

(a) Since the program has a global boolean variable g, the pushdown system has two control-states g_0 and g_1 representing respectively $g = \texttt{false}$ and $g = \texttt{true}$. The stack alphabet is

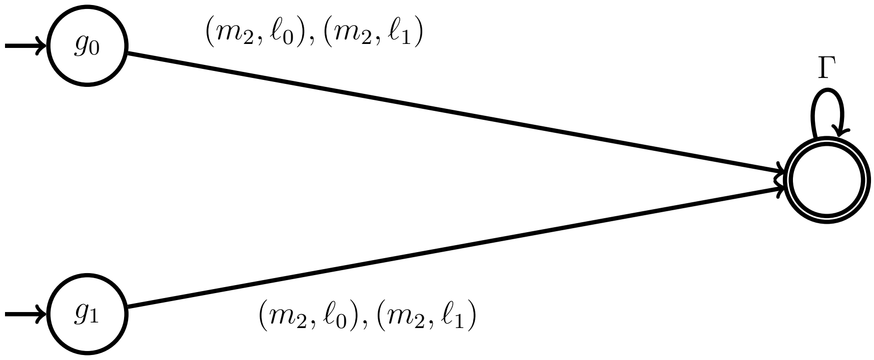
$$\Gamma = \{(m_0, \ell_0), (m_0, \ell_1), (m_1, \ell_0), (m_1, \ell_1), (m_2, \ell_0), (m_2, \ell_1), a_0, a_1, a_2, a_3, \bot\}$$

where $\bot$ stands for an error, and (m_i, ℓ_j) stands for location m_i of `main` with $\texttt{l} = \texttt{true}$ if $j = 1$, and $\texttt{l} = \texttt{false}$ if $j = 0$. The resulting pushdown system is:

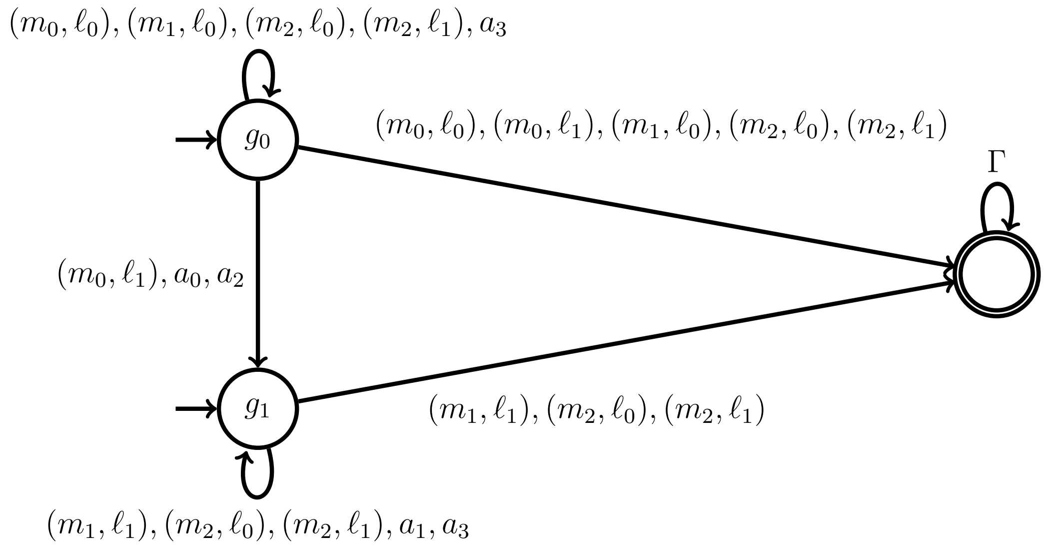


(m_0, ℓ_0)	$\to$	(m_1, ℓ_0)	(m_0, ℓ_0)	$\to$	(m_1, ℓ_0)
(m_0, ℓ_1)	$\to$	$a_0(m_1, \ell_1)$	(m_0, ℓ_1)	$\to$	$a_0(m_1, \ell_1)$
(m_1, ℓ_0)	$\to$	(m_2, ℓ_0)	(m_1, ℓ_0)	$\to$	$\bot$
(m_1, ℓ_1)	$\to$	$\bot$	(m_1, ℓ_1)	$\to$	(m_2, ℓ_1)
(m_2, ℓ_0)	$\to$	ε	(m_2, ℓ_0)	$\to$	ε
(m_2, ℓ_1)	$\to$	ε	(m_2, ℓ_1)	$\to$	ε
a_1	$\to$	$a_0 a_2$	a_1	$\to$	a_3
a_2	$\to$	$a_0 a_3$	a_2	$\to$	$a_0 a_3$
a_3	$\to$	ε	a_3	$\to$	ε

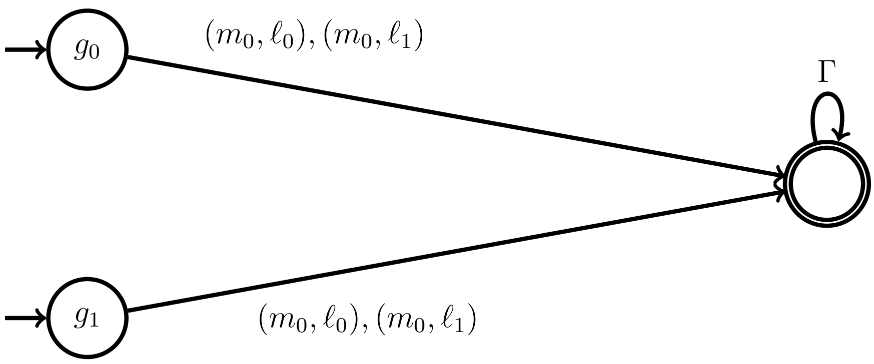
(b) We are looking for $\text{pre}^*(L)$ where $L = (g_0 + g_1)\ ((m_2, \ell_0) + (m_2, \ell_1))\Gamma^*$. We construct the following $\mathcal{P}$-automaton for L:



By applying the algorithm to compute $\text{pre}^*(L)$, we derive the following $\mathcal{P}$-automaton:



(c) We are looking for $\text{post}^*(L)$ where $L = (g_0 + g_1)\ ((m_0, \ell_0) + (m_0, \ell_1))\Gamma^*$. We construct the following $\mathcal{P}$-automaton for L:



By applying the algorithm to compute $\mathrm{post}^*(L)$, we derive the following $\mathcal{P}$-automaton: