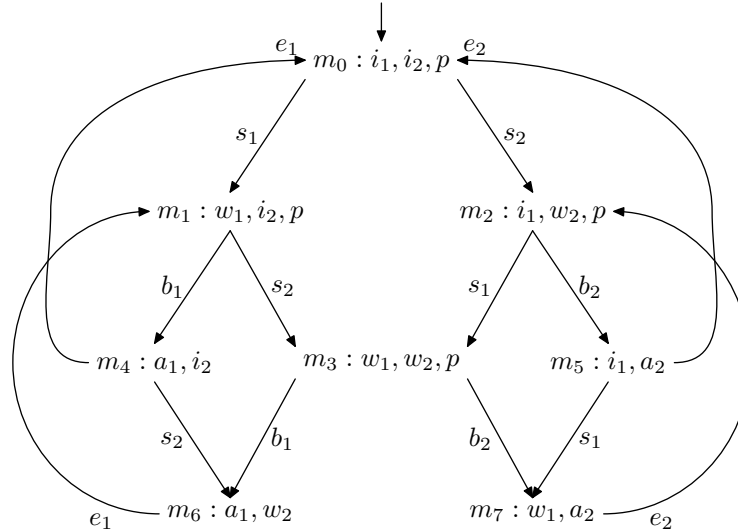


1. Generate the reachability graph of the system:



The transition rate matrix \mathbf{Q} can be generated from the reachability graph. The firing rates of transitions s_1 and s_2 are represented by constants s_1 and s_2 in the matrix.

$$\mathbf{Q} = \begin{pmatrix} -s_1 - s_2 & s_1 & s_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & -20 - s_2 & 0 & s_2 & 20 & 0 & 0 & 0 \\ 0 & 0 & -15 - s_1 & s_1 & 0 & 15 & 0 & 0 \\ 0 & 0 & 0 & -35 & 0 & 0 & 20 & 15 \\ 30 & 0 & 0 & 0 & -30 - s_2 & 0 & s_2 & 0 \\ 15 & 0 & 0 & 0 & 0 & -15 - s_1 & 0 & s_1 \\ 0 & 30 & 0 & 0 & 0 & 0 & -30 & 0 \\ 0 & 0 & 15 & 0 & 0 & 0 & 0 & -15 \end{pmatrix}$$

Now the steady state solution can be computed:

$$\begin{aligned} \pi \cdot \mathbf{Q} &= \mathbf{0} \\ \pi \cdot \mathbf{E} &= \mathbf{e} \end{aligned}$$

The equations become $\pi = \mathbf{e} \cdot (\mathbf{Q} + \mathbf{E})^{-1}$. This equation can be solved for example in GNU Octave with command `ones(1,8)*inverse(Q+ones(8))`, when \mathbf{Q} has been defined.

We get the average marking by multiplying π with a column vector representing the marking of a place in each reachable marking. For example, the average marking of the place a_1 can be computed with $\pi \cdot (00001010)^T$.

In the homepage of the course there is a function for Octave computing the steady state and the average marking of the places w_1, w_2, a_1, a_2 , when given the firing rates of transitions s_1, s_2 as arguments.

2. The CCS description of the protocol:

$$\begin{aligned}
S & ::= m_1 \cdot S_1 + m_2 \cdot S_2 & D & ::= send_1 \cdot (rec_1 \cdot D + lose \cdot D) + \\
S_1 & ::= send_1 \cdot S_3 & & send_2 \cdot (rec_2 \cdot D + lose \cdot D) \\
S_2 & ::= send_2 \cdot S_4 & A & ::= a_1 \cdot ack_1 \cdot A + a_2 \cdot ack_2 \cdot A \\
S_3 & ::= ack_1 \cdot S + tout \cdot S_1 & R & ::= rec_1 \cdot a_1 \cdot r_1 \cdot R + \\
S_4 & ::= ack_2 \cdot S + tout \cdot S_2 & & rec_2 \cdot a_2 \cdot r_2 \cdot R
\end{aligned}$$

$$P ::= S \mid R \mid D \mid A$$

m_i The message is given to the protocol for sending

$send_i$ The message is passed to channel

rec_i The message arrives to the receiving end

r_i The protocol delivers the message to the receiver

a_i The receiver sends an acknowledgement

ack_i The sender gets the acknowledgement

$tout$ The sender's timer goes off

$lose$ The data channel loses a message

The agents can be transformed to place/transition systems by interpreting them as parallel labelled transition systems (lecture 1, p. 1–20) which can then be transformed to place/transition nets. For every state in LTS there will be on place in P/T net and for every arc in LTS there will be one net transition. There is a token in a place if the LTS is in a corresponding state. We thus get the initial marking from the initial states of the LTSs.

In this case we will first make separate nets for the agents S , R , D and A . The nets representing the different parts of the system can then be connected in a way corresponding to the parallel composition by combining the transitions with same name.

When combining the transitions one must take in the account the used parallel composition operator. The basic parallel composition \parallel used in CCS requires also other transitions that the parallel composition \mid defined in this tutorial.

The protocol resembles the protocol in tutorial 1 with reliable acknowledgement channel. The protocol does not work correctly in this case, either. The errors can be found by looking the strongly connected components of the net. Commands **strong** and **terminal** help in finding the errors.

There is a place/transition system corresponding to the CCS description in the homepage of the course.