Theoretical Foundations of the UML WS 17/18

— Exercise Sheet 5 —

Hand in until November 28th before the exercise class.

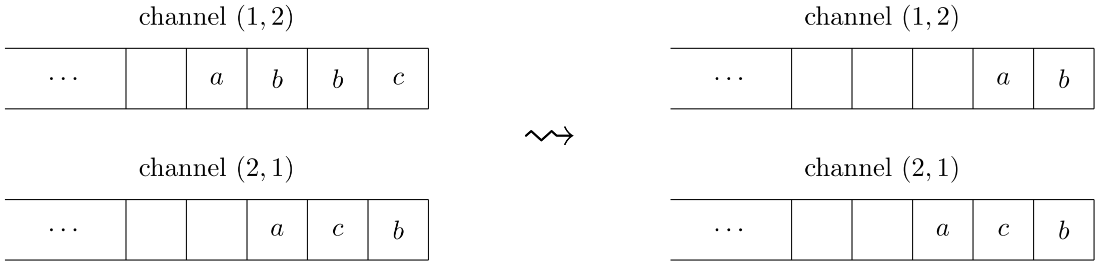
General Remarks

- The exercises should be solved in groups of *three* students.
- You may hand in your solutions for the exercises just before the exercise class starts at 15:30 or by dropping them into the “TFUML” box at our chair. Do *not* hand in your solutions via L2P.

Exercise 1 (4 Points)

In the lecture, we have introduced the CFM (cf. Lecture 9 slide 9) with *perfect* channels (i.e. messages stored in the channel will never get lost). In this exercise, we consider a modified CFM with *lossy* channels. We assume the channels between processes are unreliable, hence can lose messages at any time point.

For example, consider the channel contents depicted on the left side of the following figure. The first message c and the third message b stored in channel $(1, 2)$ can be lost which results in a configuration as shown on the right.


channel $(1, 2)$
$\cdots$ | | a | b | b | c
channel $(2, 1)$
$\cdots$ | | | a | c | b
$\rightsquigarrow$
channel $(1, 2)$
$\cdots$ | | | | a | b
channel $(2, 1)$
$\cdots$ | | | a | c | b

Task: Give a formal semantics and the word language of a *lossy channel* CFM as mentioned above.

Exercise 2 (6 Points)

Given the following specification $\mathcal{S}$ where a producer p and a consumer q are the acting units:

- The producer p starts sending messages with content 0 (one bit) to the consumer q until he receives an acknowledgement message a from the consumer. In that case the bit is swapped to 1 and p starts sending messages with content 1 to q until the next a is received. Then again the bit is inverted and the procedure can continue as before.
- The consumer process q, however, starts by receiving at least one 0. After that he may receive more 0s until finally he sends an a to p. After this acknowledgement the remaining 0s in the buffer (p, c) have to be received. Then process q starts receiving 1s (if p sent at least one). Having received at least one message with content 1, q may send an a after any of the successing 1s. Having sent the a, the remaining 1s have to be processed before another round of receiving 0s may start.
- The system may accept directly after any a that is received by process p (as long as the empty-buffer condition is fulfilled).

Tasks:

a) Give three example MSCs that conform to the specification $\mathcal{S}$.

b) Give a CFM implementation for $\mathcal{S}$.

Exercise 3 (4 Points)

a) Prove or disprove whether the following decision problem is decidable:

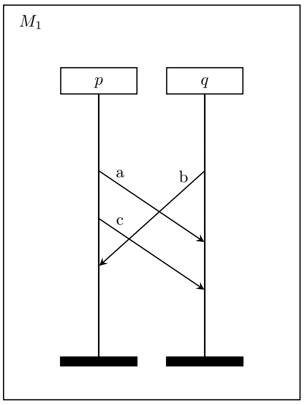
> **Problem 5.1:**
> *Given a CMSG determine whether a particular message content can be received in at least one accepting path. The corresponding receive event can be unmatched.*

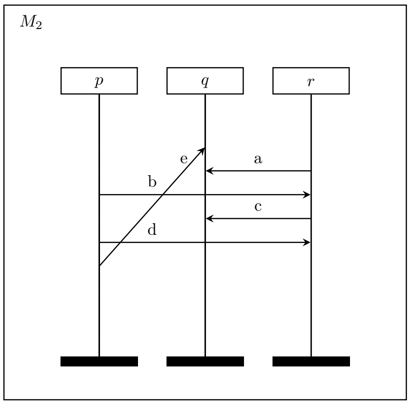
b) Prove or disprove whether the following decision problem is decidable:

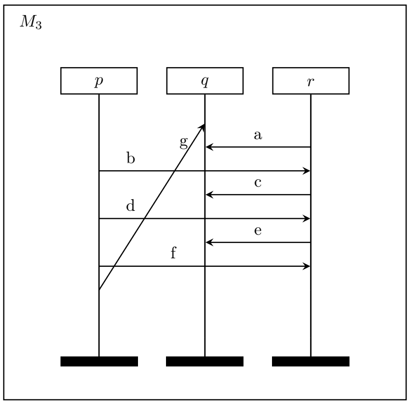
> **Problem 5.2:**
> *Given a CMSG determine whether a particular message content can be received in at least one accepting* and safe *path.*

Exercise 4 (6 Points)

Determine for each of the following MSCs (M_1, M_2, M_3) if they are existentially ($\exists-$) or universally ($\forall-$) bounded. In case an MSC is $\exists/\forall-$bounded, determine the smallest B such that the MSC is $\exists/\forall - B-$bounded and justify why it cannot be $\exists/\forall - (B-1)-$bounded.




M_2
p
q
r
e
a
b
c
d


M_3
p
q
r
g
a
b
c
d
e
f