

Embedded Systems 08/09 – Problem Set 5

Problem 1 (Kahn Process Networks – Part One) (30 pts.)

Alice owns seven flower shops in different cities. At the end of each month she wants to know the total profit of all stores. So, each shop sends its individual profit in a letter to Alice by mail and she computes the sum as soon as all seven individual profits have arrived.

1. Model the depicted system as a Kahn process network, where Alice and the seven stores are respectively represented by the processes A , S_1 , S_2 , S_3 , S_4 , S_5 , S_6 , and S_7 . The individual profit of store i is represented by the local variable $\text{profit}(i)$ which is only accessible by process S_i . (10 pts.)
2. Alice is tired of doing all the summation by herself. Luckily for her, each store can not only send but also receive mails by other stores. Therefore, she decides to distribute the summation work to all the individual stores such that she doesn't have to do any addition at all and each store has to perform at most two addition operations¹. Assuming that sending a letter always takes one day, Alice wants to receive the total profit within three days.
Model an alternative Kahn process network that exploits the fact that different processes can send letters concurrently. (15 pts.)
3. Suppose Alice owns n shops where each shop performs at most two addition operations. In this case, what is the minimal number of days until Alice gets the total profit of all shops? Justify your answer! (5 pts.)

For subproblems 1 and 2 please specify the process network graphically and the implementation of each process in a C-like syntax using the primitives `<var> = wait(<in>)` and `send(<expr>, <out>)`.

Problem 2 (Kahn Process Networks – Part Two) (20 pts.)

Specify a Kahn process network that computes a sequence of natural numbers:

$$\begin{aligned}f(0) &= 1 \\f(n) &= nf(n-1)\end{aligned}$$

Please specify the process network graphically and give C-like code for each process. Use atomic processes that can add or multiply two numbers. You may also use processes that initially generate a constant, and simply forward their input afterwards. Finally, the sequence f

¹here, $x + y$ is one addition operation, $x + y + z$ are two operations, etc.

should be sent to a dedicated process RES that acts as a sink. For the process implementations use the following primitives:

- `<var> = wait(<in>)`
- `send(<expression>, <out>)`
- `while(true) { ... }`
- Expressions may be `<var>+<var>`, `<var>+<const>`, or `<var>*<var>`.

Problem 3 (SDF)

(30 pts.)

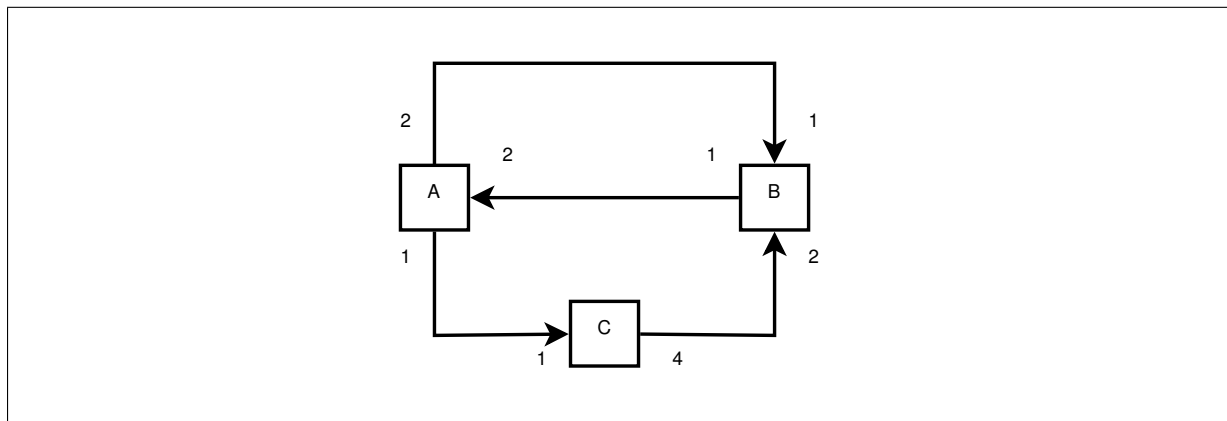


Figure 1: SDF network for problem 3.

For the SDF system that is given in Figure 1 assume $d(BA) = 4$.

1. Setup the topology matrix according to Lee/Messerschmitt. (5 pts.)
2. Compute the relative execution rates. (15 pts.)
3. Determine a periodic schedule. (10 pts.)

Problem 4 (Message Sequence Charts)

(20 pts.)

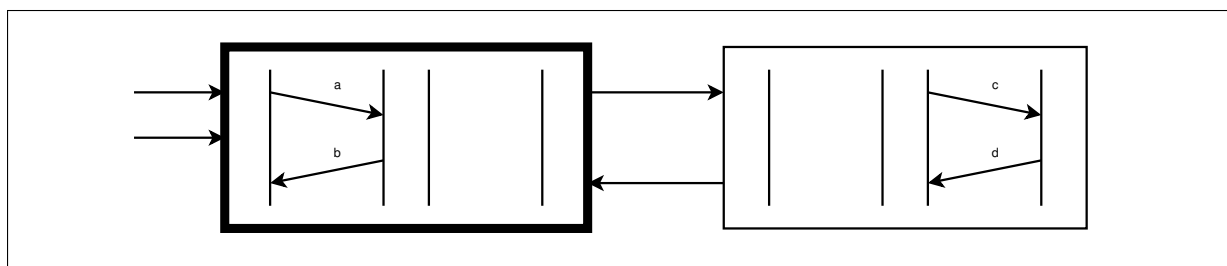


Figure 2: MSG for problem 4.

For the MSG that is given in Figure 2 assume asynchronous synchronisation. The left MSC is both initial (incoming double arrow) and final (bold outline). Is the given MSG regular? Justify your answer!