Automation – EIEN50 (former EIEF45)

Exam Thursday March 17, 2022

You may bring the course book and the hardcopy of the chapter on HMI, but *not* the solutions to problems nor your own solutions to simulation tasks, lecture slides, home works or other personal notes. A calculator (memory cleared), TEFYMA, formula sheet from the Automatic Control department and a dictionary are permitted. You may answer in **Swedish** or in **English**.

Grading: There are 30 points all together. The following grades will apply:
Grade 3: at least 15 points
Grade 4: at least 20 points
Grade 5: at least 25 points

Note that all answers <u>should be complete</u> and <u>well motivated</u>. Your line of thought should be easy to follow and hand calculations should be provided for all mathematical problems. Corrections will be completed not later than *Friday April **8**, 2022*.

Good Luck!

Problem 1 (5 points)
Four completely mixed tanks with constant volumes are connected according to Figure 1 (initially filled with pure water). The external inputs to the first tank are the flow rate (Q_1) and the concentration of a specific dissolved compound in that flow (C_{in1}). The external inputs to the second tank are the flow rate (Q_2) and a concentration of the *same* compound in that flow (C_{in2}). The compound does not react with anything. The volumes of the tanks are V_1, V_2, V_3 and V_4. An automatic flow splitter directs 40% of the output flow rate from tank 1 to tank 2, 40% of the output flow rate from tank 1 to tank 4 and 20% exits the system from tank 1. V_2 and V_3 are the same sizes and 4 times as large as V_1 and V_4 is 10 times larger than V_1. The output flow from tank 2 goes to tank 3 and the outputs from tank 3 and 4 simply exits the process (note: constant liquid volume in all tanks at all times and at time = 0 only water and no compound is in the reactors).

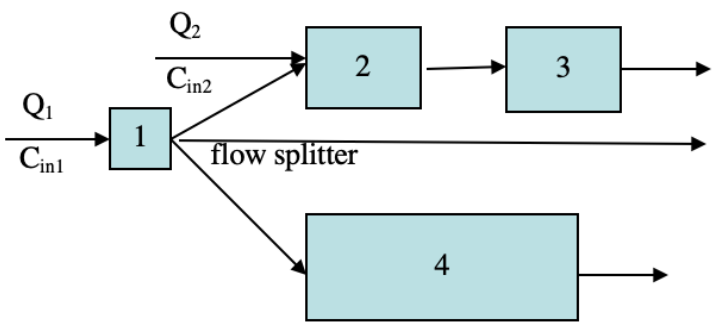


Figure 1: Tank system for problem 1.

a) Set up the differential equations for the concentrations C_1, C_2, C_3 and C_4 to represent the above process (define all variables you use). (**1 p**)

b) Transform the equations into state-space form (assume you can measure three things: $C_{in1}+C_1$, C_2 and C_4). Cleary provide the A, B, C and D matrices. (**1 p**)

c) Draw in *one* diagram (concentration as a function of time) as detailed as you can and also explain in words the behaviour of how the four concentrations will change over time in the four tanks, if all are initially filled with pure water (i.e. concentration = 0) and the input concentration C_{in1} at time 0 is set to 50 g/l and the input concentration C_{in2} at time 0 is set to 25 g/l. Assume both the influent flow rates and concentrations to remain constant from time = 0 and onwards. Use Q_1 = 5 l/h, Q_2 = 10 l/h, V_1 = 1 l, V_2 = V_3 = 4 l, V_4 = 10 l and discuss the details of the dynamic response. Also clearly provide the steady state concentrations in all four reactors and discuss the time constants of the system. (**2 p**)

d) When the system in c) has reached its steady state a pump is started which continuously pumps a flow rate Q_{rec} (= 5 l/h) from tank 3 back into tank 1. Neglect any time delays and observe that all volumes are still constant. Define the differential equations for the new system and calculate its new steady state values. (**1 p**)

Solution

a) Using the variables defined in the task the mass balances give us the dynamic system, i.e.

$$\begin{cases} \dfrac{dc_1}{dt} = \dfrac{Q_1}{V_1}(c_{in1} - c_1) \\ \dfrac{dc_2}{dt} = \dfrac{0.4 \cdot Q_1}{V_2} c_1 + \dfrac{Q_2}{V_2} c_{in2} - \dfrac{0.4 \cdot Q_1 + Q_2}{V_2} c_2 \\ \dfrac{dc_3}{dt} = \dfrac{0.4 \cdot Q_1 + Q_2}{V_3}(c_2 - c_3) \\ \dfrac{dc_4}{dt} = \dfrac{0.4 \cdot Q_1}{V_4}(c_1 - c_4) \end{cases}$$

b) The state space representation (**A** and **B**) can be directly identified from the above equations as:

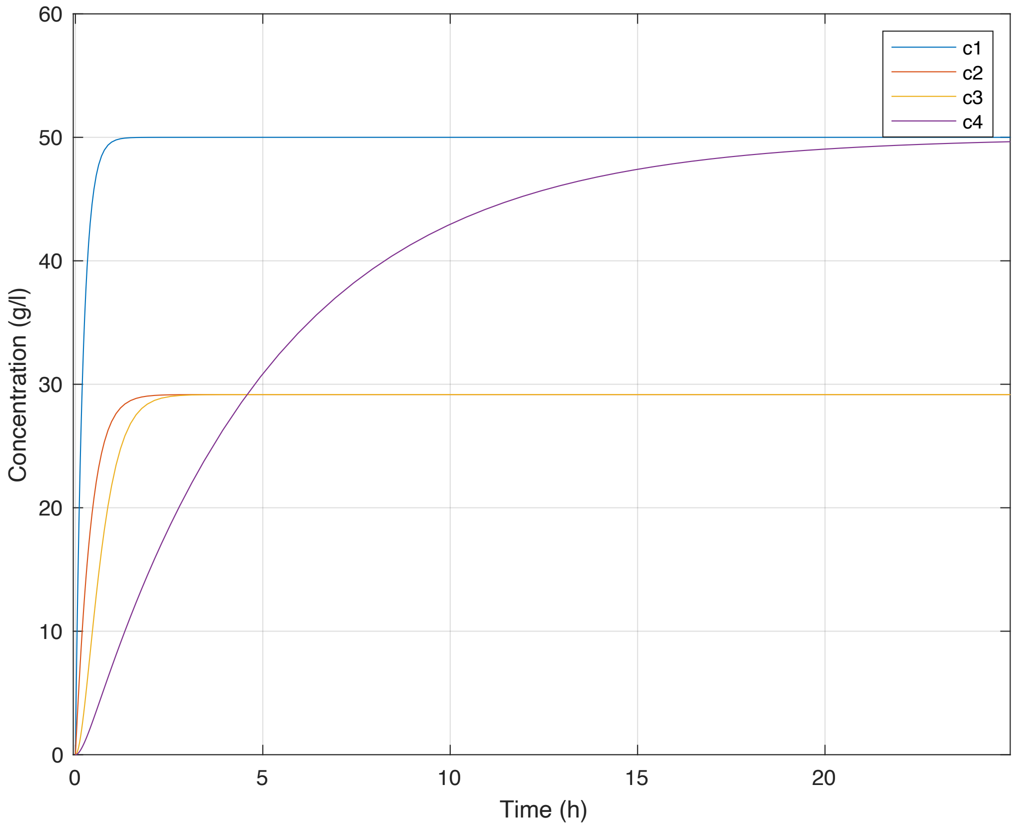
$$\left\{ \frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{Ax} + \boldsymbol{Bu} = \begin{bmatrix} -\dfrac{Q_1}{V_1} & 0 & 0 & 0 \\ \dfrac{0.4 \cdot Q_1}{V_2} & -\dfrac{0.4 \cdot Q_1 + Q_2}{V_2} & 0 & 0 \\ 0 & \dfrac{0.4 \cdot Q_1 + Q_2}{V_3} & -\dfrac{0.4 \cdot Q_1 + Q_2}{V_3} & 0 \\ \dfrac{0.4 \cdot Q_1}{V_2} & 0 & 0 & -\dfrac{0.4 \cdot Q_1}{V_2} \end{bmatrix} \cdot \boldsymbol{x} + \right.$$

$$\begin{bmatrix} \dfrac{Q_1}{V_1} & 0 \\ 0 & \dfrac{Q_2}{V_2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \boldsymbol{u}$$

where $\boldsymbol{x} = [c_1 \quad c_2 \quad c_3 \quad c_4]^T$ and $\boldsymbol{u} = [c_{in1} \;\; c_{in2}]^T$ and the **C** and **D** matrices are defined by the given available measurements as:

$$
\boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \boldsymbol{x} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \boldsymbol{u}
$$

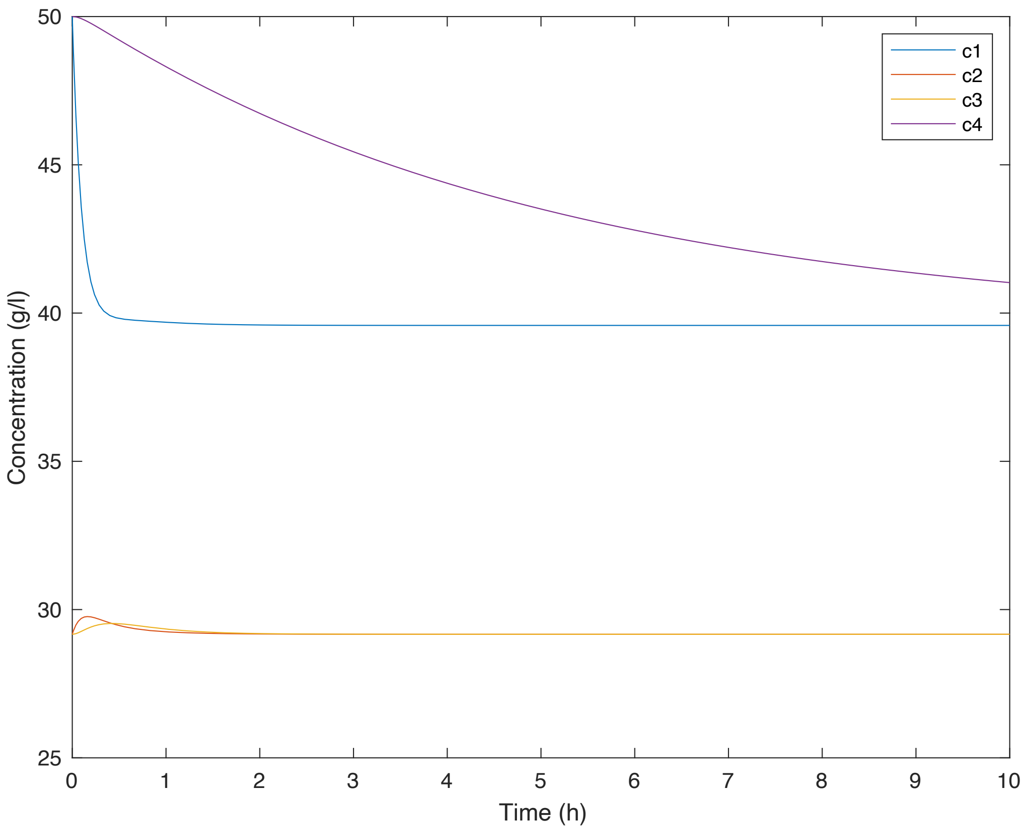
c) The initial concentrations in all four tanks are equal to 0. Tank 1 is a normal first-order system with constant input and a time constant of V_1/Q_1 (= 0.2 h) and a steady state value equal to C_{in1} (= 50 g/l). Tank 2 has inputs from tank 1 and via Q_2. A simple calculation yields that the steady state value must be $(0.4 \cdot Q_1 {*} C_{in1} + Q_2 {*} C_{in2})/(0.4 \cdot Q_1 + Q_2)$ = 29.17 g/l (based on the values given in the task). The same steady state concentration must also hold for tank 3, since all the input to that tank comes from tank 2. Tank 4 receives all input from tank 1 and must therefore also have a steady state value = 50 g/l. The steady state values can also be obtained by simply setting the derivatives in (a) equal to 0 and solving the resulting equation system. The small size of tank 1 compared to tank 2 and tank 4 means that the dynamic response of tanks 2 and 4 will be almost as if tank 1 did not exist, i.e. a time constant of $V_2/(0.4 \cdot Q_1 + Q_2)$ = 0.33 h for tank 2 and $V_4/0.4 \cdot Q_1$ = 5 h for tank 4. In reality, a little bit slower response due to tank 1 (especially for tank 2). Tank 3 is the same size as tank 2 and will therefore follow the dynamic behaviour of tank 2, only somewhat later in time. Tank 4 will be the last tank to reach steady state. Based on this discussion it is now possible to plot the dynamic behaviour of the tanks fairly accurately. A detailed plot of the simulated system is shown below.



d) The initial concentrations in the four tanks are equal to the steady states from c), i.e. 50, 29.17, 29.17 and 50 g/l. Since all volumes are constant the start of the recirculation does affect the flow rate *leaving* the system from tank 3 (as $0.6 \cdot Q_{rec}$ does not reach tanks 2 and 3), this output is now $0.4 \cdot Q_1 + Q_2 - 0.6 \cdot Q_{rec}$. So Q_{rec} can never be larger than $(0.4 \cdot Q_1 + Q_2)/0.6$ = 20 l/h, otherwise the volume in tank 3 could not remain constant but would start to decrease. We are assuming no time delays so oscillations in the system seem unlikely. The system is still linear. The new set of differential equations becomes:

$$\begin{cases} \dfrac{dc_1}{dt} = \dfrac{Q_1}{V_1} c_{in1} + \dfrac{Q_{rec}}{V_1} c_3 - \dfrac{Q_1 + Q_{rec}}{V_1} c_1 \\ \dfrac{dc_2}{dt} = \dfrac{0.4 \cdot (Q_1 + Q_{rec})}{V_2} c_1 + \dfrac{Q_2}{V_2} c_{in2} - \dfrac{0.4 \cdot (Q_1 + Q_{rec}) + Q_2}{V_2} c_2 \\ \dfrac{dc_3}{dt} = \dfrac{0.4 \cdot (Q_1 + Q_{rec}) + Q_2}{V_3} (c_2 - c_3) \\ \dfrac{dc_4}{dt} = \dfrac{0.4 \cdot (Q_1 + Q_{rec})}{V_4} (c_1 - c_4) \end{cases}$$

Setting all derivatives to zero and inserting the values from the task gives us an equation system with four unknowns that we can solve.
We get the steady state results $c_1 = c_4 = 39.58$ g/l and $c_2 = c_3 = 29.17$ g/l. So the recirculation will only affect the steady state value of c_1 and c_4 and since we are starting from the previous steady state the dynamics will be very limited. Apart from the change in c_1 and c_4 we could also expect to see a small initial increase in c_2 and c_3 because of the increased flow rate from tank 1 to tank 2, which initially has a concentration of 50 g/l. A rough estimate of the time constant for the change in tank 1 would be $V_1/Q_{rec}=0.2$ h, since it is the 'new' inflow to tank 1 that is causing the system to change. The time constant for the change in tank 4 will be roughly $V_4/(0.4 \cdot Q_{rec}) = 5$ h. Based on this discussion it is now possible to plot the dynamic behaviour of the tanks fairly accurately (NOT required in the task). A detailed plot of the simulated system is shown below.



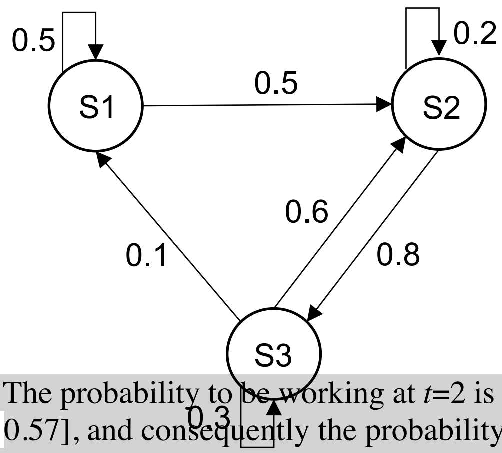
Problem 2 (4 points)
A machine can be described as being in three different states: (1) under repair, (2) waiting for a new job, (3) working. While the machine is working the probability to break is 0.1 and the probability to get finished (go to waiting) is 0.6. If the machine is under repair there is a 0.5 probability to get repaired, and then the machine will become waiting. A broken machine is never brought directly (in one step) to working mode. If the machine is waiting there is a 0.8 probability to become working. A waiting machine does not break.

a) Describe the system as a Markov chain, define the transition probability matrix and make a state diagram. (**1 p**)
b) Assume that the machine is working at time=0. What is the probability
   a. to be working at time=2? (**0.5 p**)
   b. to be under repair at time=3? (**0.5 p**)
c) Calculate the eigenvalues of the system. Based on the result motivate if the system is ergodic or not. (**1 p**)
d) What is the probability for the machine to be in waiting mode after a long time (steady state)? Complete solution required (no calculator results accepted). (**1 p**)

Solution:

a) S1 = under repair; S2 = waiting mode; S3 = working mode

The transition probability matrix is $\mathbf{P} = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0 & 0.2 & 0.8 \\ 0.1 & 0.6 & 0.3 \end{bmatrix}$ and the state graph is:



b) The probability to be working at t=2 is $p_3(t=2)$ given by $\mathbf{p}(2) = \mathbf{p}(0)*\mathbf{P}^2 = \; = [0.08 \;\; 0.35 \;\; 0.57]$, and consequently the probability to be operating at time = 2 is equal to 0.57.
The probability to be under repair at t=3 is $p_1(t=3)$ given by $\mathbf{p}(3) = \mathbf{p}(2)*\mathbf{P} = \; = [0.097 \;\; 0.452 \;\; 0.451]$, and consequently the probability to be under repair at time = 3 is equal to 0.097.

c) Calculate the eigenvalues as $\det(\lambda \cdot \mathbf{I} - \mathbf{P}) = 0$ which gives the three eigenvalues 1, 0.4123, –0.4123. As the two eigenvalues that are not equal to 1 are strictly within the unit circle this means that the system must be ergodic. One eigenvalue must be equal to 1 for a Markov chain system. The knowledge that one eigenvalue is equal to 1 simplifies the calculation of the eigenvalues as it is then obvious that the third order equation can be written as $(\lambda - 1)(\dots)$ and only a second order equation needs to be solved.

d) Use Gauss elimination or similar to solve the equation system $(p1 \;\; p2 \;\; p3)*(\mathbf{I} - \mathbf{P}) = 0$ together with the extra condition p1 + p2 + p3 = 1. This gives after some calculations: p1 = 0.0964, p2 = 0.4217 and p3 = 0.4819. Therefore $\mathbf{p}(\infty) = [0.0964 \;\; 0.4217 \;\; 0.4819]$, and the stationary probability to be in waiting mode is $p_2(\infty) = 0.4217$.

Problem 3 (1 point)
Assume that an input buffer is empty at time $t = 0$. The arrival of new jobs is Poisson distributed, with an arrival rate of $\lambda = 2$ per hour. What is the probability that there will be more than 3 jobs in the buffer after 1 hour?

Solution
This is a traditional birth process. We define $p_k(t)$ as the probability that there are k jobs in the system at time t. Here $\lambda = 2$ per hour. At least four parts means the same as (1 – no parts – one part – two parts – three parts). The equation from the book, page 198, gives:
The probability that there are exactly 0 jobs in the system at time $t = 1$ is:

$p_0(1) = e^{-2\cdot 1} = e^{-2} \approx 0.1353$, i.e. about 13.53%.

The probability that there are exactly 1 job in the system at time $t = 1$ is:

$p_1(1) = (2\cdot 1)\cdot e^{-2\cdot 1} = 2\cdot e^{-2} \approx 0.2707$, i.e. about 27.07%.

The probability that there are exactly 2 jobs in the system at time $t = 1$ is:

$p_2(1) = (2\cdot 1)^2/(2!)\cdot e^{-2\cdot 1} = 2\cdot e^{-2} \approx 0.2707$, i.e. about 27.07%.

The probability that there are exactly 3 jobs in the system at time $t = 1$ is:

$p_3(1) = (2\cdot 1)^3/(3!)\cdot e^{-2\cdot 1} = 1.333\cdot e^{-2} \approx 0.1804$, i.e. about 18.04%.

So the probability of there being at least 4 jobs in the buffer at time t=1 is $1 - p_0(1) - p_1(1) - p_2(1) - p_3(1) \approx 0.1429$, i.e. about 14.3%.

Problem 4 (3 points)
A transfer line consists of two machines (M1 and M2) with a buffer in between described as a Markov process. The machines can be either operating or broken and the buffer can hold 0, 1 or 2 items but some stationary probabilities have been lost due to a computer break-down. You have information from company notes based on historic data that the system efficiency (E1) for machine 1 is 0.491 and the system efficiency for machine 2 (E2) is 0.3507. Moreover, the probability that M2 is starved is 0.5065 and the average buffer size is 0.6489 items. Based on this information help the company to back-calculate the missing stationary probabilities in the table below, i.e. values for *a*, *b*, *c*, *d* and *e*.

State name	Machine 1 (0=broken,1=operating)	Machine 2 (0=broken,1=operating)	Buffer size (0, 1 or 2)	Stationary probability
S0	0	0	0	*a*
S1	0	0	1	0.0080
S2	0	0	2	0.0118
S3	0	1	0	*b*
S4	0	1	1	*c*
S5	0	1	2	0.0169
S6	1	0	0	0.0212
S7	1	0	1	0.0233
S8	1	0	2	*d*
S9	1	1	0	0.2711
S10	1	1	1	*e*
S11	1	1	2	0.1250

Solution
E1 = all states where machine 1 is producing = p(1,0,0) + p(1,1,0) + p(1,0,1) + p(1,1,1) = p(S6) + p(S9) + p(S7) + p(S10) = 0.0212 + 0.2711 + 0.0233 + e = 0.491. Therefore e = 0.1754.

E2 = all states where machine 2 is producing = p(0,1,1) + p(1,1,1) + p(0,1,2) + p(1,1,2) = p(S4) + p(S10) + p(S5) + p(S11) = *c* + *e* + 0.0169 + 0.1250 = 0.3507. Therefore *c* = 0.0334.
M2 is starved when M2 is operating but the buffer is empty = p(S3) + p(S9) = *b* + 0.2711 = 0.5065. Therefore *b* = 0.2354.
Average buffer size = 0 * (p(S0) + p(S3) + p(S6) + p(S9)) + 1 * (p(S1) + p(S4) + p(S7) + p(S10)) + 2 * (p(S2) + p(S5) + p(S8) + p(S11)) = 1 * (0.0080 + *c* + 0.0233 + *e*) + 2 * (0.0118 + 0.0169 + *d* + 0.1250) = 0.6489. Therefore *d* = 0.0507.
Normality tells us that the sum of ALL the probabilities must always equal 1. Therefore *a* = 0.0278.
To summarize:
a = 0.0278; *b* = 0.2354; *c* = 0.0334; *d* = 0.0507; *e* = 0.1754.

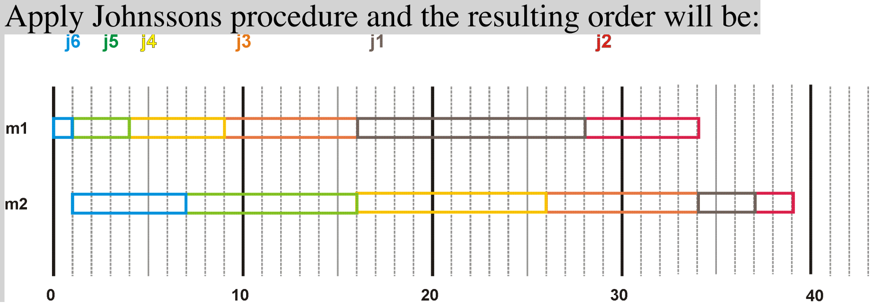
Problem 5 (4 points)
Consider a transfer line. Every job has to be machined first in machine m_1 and then in machine m_2. (You can assume that there is a large buffer between the machines).
Six jobs ($j_1 - j_6$) have to be processed where the processing times are:

	j_1	j_2	j_3	j_4	j_5	j_6
m_1	12	6	7	5	3	1
m_2	3	2	8	10	9	6

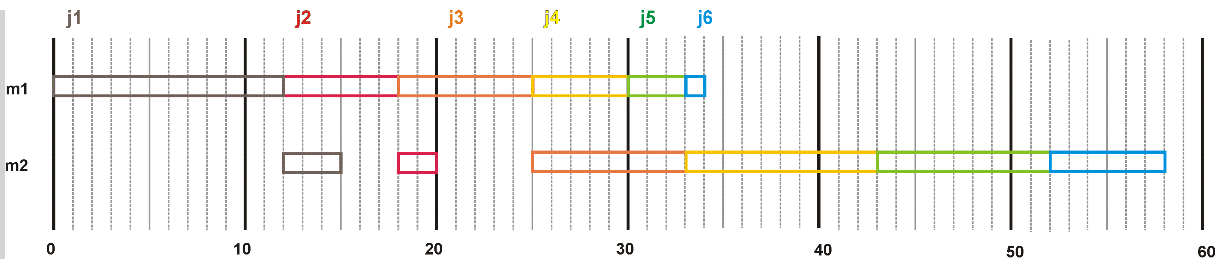
a) Calculate the order between the jobs that will minimize the time to finish all jobs. **(1p)**
b) How long is the total processing time in this case? **(0.5 p)**
c) What is the idle time for each one of the machines in this case? **(0.5 p)**
d) How long is the total processing time if the job order is according to the table, i.e. j1, j2 … j6 ? **(0.5 p)**
e) What is now the idle time for each one of the machines? **(0.5 p)**
f) Calculate the mean flow time for the jobs in both sequence orders. **(1 p)**

Solution

a) Apply Johnssons procedure and the resulting order will be:



b) From the Gantt chart we see that the total processing time is 39 time units.
c) Machine m1 is idle during 5 and m2 during 1 time unit, as can be seen in the chart.



d)

The total processing time is now 58 time units.

e) Machine m1 is idle during 24 and m2 during 20 time unit,
f) MFT can simply be calculated by reading the finish time for each job. It is: 7, 16, 26, 34, 37, 39 in the diagram in b). The average (MFT) is 26.5. In d) the times are: 15, 20, 33, 43, 52, 58. The average is 36.8.

Problem 6 (4 points)
In a foundry aluminum is melted for casting various products. A linear control (Pulse Width Modulation) of an electric heater rated 50 kW is used. The temperature is measured by the infrared radiation directly from the melted aluminum. To control the temperature a PID controller is considered

a) When the process is started with cold solid aluminum there will probably be a bad control sequence if the basic structure of a PID controller is used. Why? What in the scenario is causing this? (More than one reason?) **(2 p)**
b) What function should you ask for in the PID-controller and explain in your own words how this can be implemented in program code. **(2 p)**

Solution

a) The actuator amplitude is limited and since we start with solid cold material the heating will probably go on for a long time and during this the integrator will wind up. Furthermore, the actuator is only heating which means that we don't have a symmetrical actuator control. The process of melting is in itself an unlinear function since the melting requires a certain amount of energy at the same temperature and the heat capacity for the solid material and the liquid can be different.
b) If we still should use a PID controller the largest problem is probably the windup. An antiwindup function is often implemented in controllers. There are several ways to implement this function. A common way is to calculate the I-part backwards when the output goes to a limit. Then the I-part is umax-P-D. In this way the I-part corresponds only to the contribution that could be implemented. Some filtering functions to make this smooth could be needed.

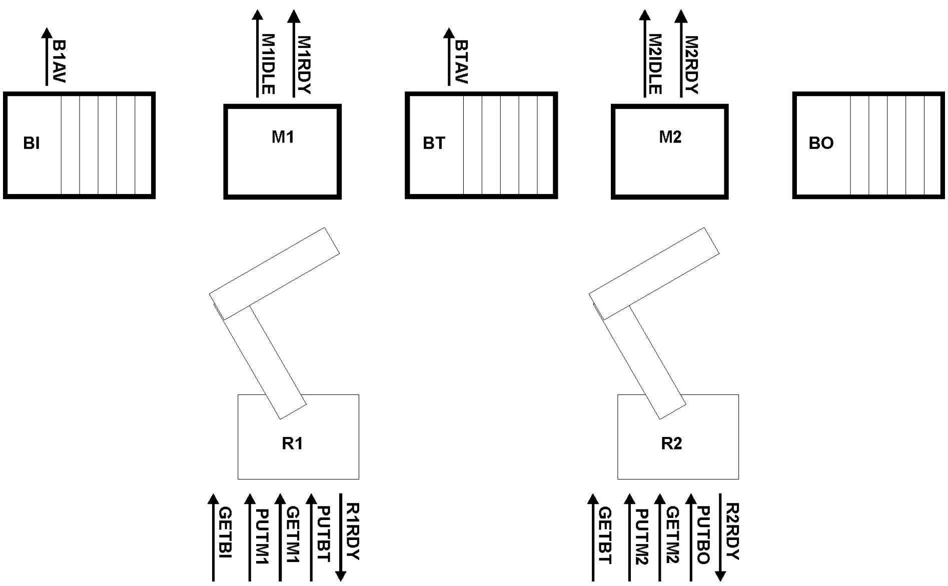
Problem 7 (1 point)
A measurement signal of 600 Hz is sampled with 3 kHz. Suddenly there is an equipment failure and a sinusoidal disturbance signal of 16 kHz is added to the input measurement signal. No anti-alias filters have been included in the measurement system. How does the disturbance signal affect the sampled signal (at what frequency does the disturbance appear)?

Solution
Use for example the graph in Figure 10.3 (page 361) in the book: the value $f/f_s = 16/3 = 5.33$, which will intersect the graph at the y-value 0.33, i.e. when $f_o/f_s = 0.33$. This means that the observed alias frequency will appear at $f_o = 0.33 \cdot 3 = 1$ kHz = 1000 Hz.

Problem 8 (1 point)
The Shannon sampling theorem says that the sampling frequency must be at least 2 times the highest frequency component of the input signal. In practice one needs to sample more often than that. Why?

Solution
The Shannon sampling theorem (or Nyquist theorem) is based on the assumption that the signal that is to be reconstructed is perfectly periodical and that we have infinite time to make the reconstruction in a perfect way. As this is never the case in reality we need to sample faster than what the theoretical lower limit says. The answer can also be based on a discussion of bandwidth of the signal.

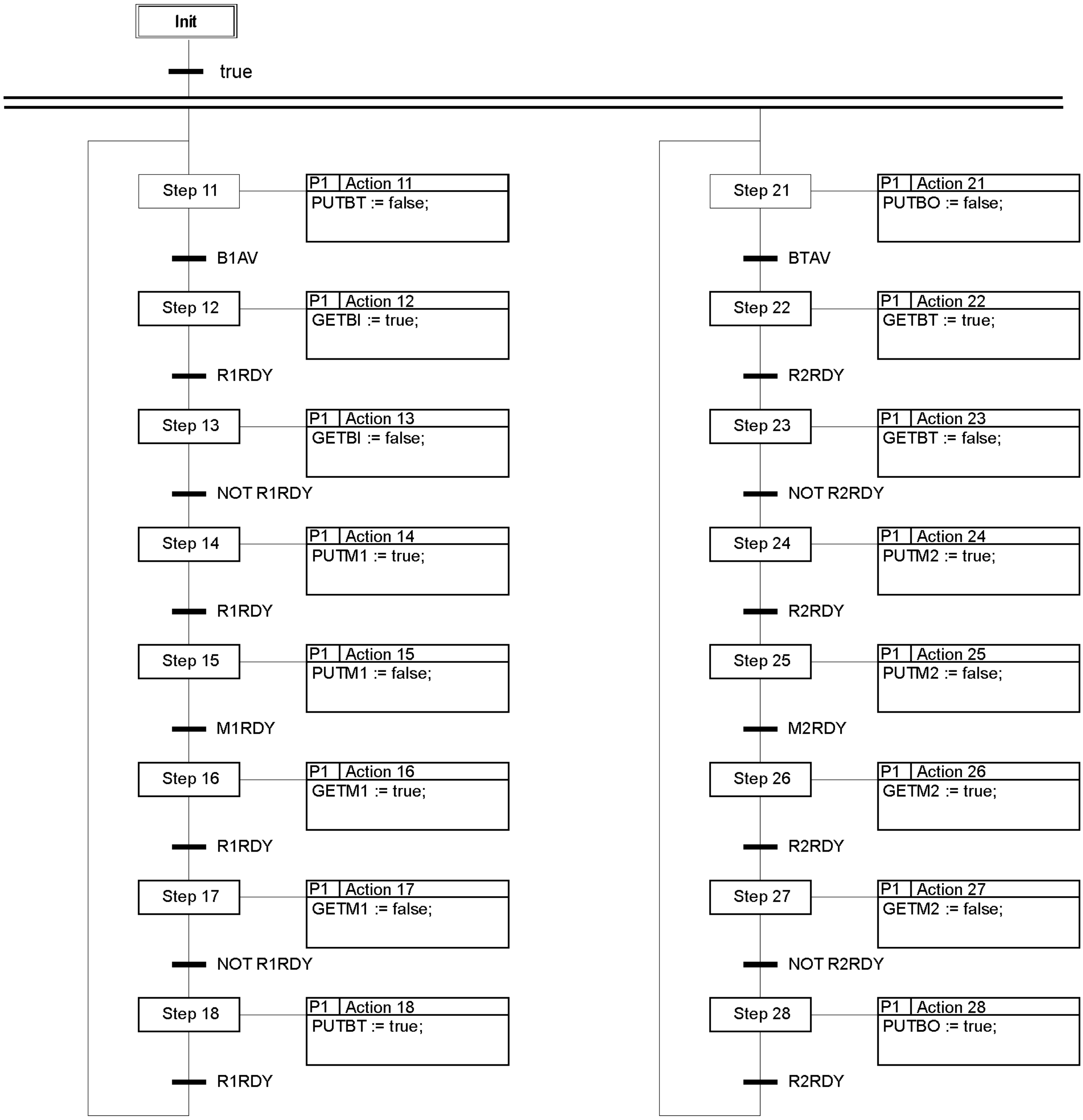
Problem 9 (5 points)
A production line consists of two machines, two robots for job transportation and buffers as shown in the figure below. All jobs should first be processed in M1 and then in M2. The buffer size is unlimited.



a) The buffers BI and BT indicate if a job is available by BIAV and BTAV respectively. The machines will start automatically when a job is loaded. They indicate with the signal MxIDLE (x is 1 or 2) that no job is loaded and with MxRDY that a processed job can be picked up. Robot R1 can execute the commands GETB1, PUTM1, GETM1 and PUTBT. When a command has been executed it is acknowledged by the robot status signal R1RDY.
Robot R2 can execute the commands GETBT, PUTM2, GETM2 and PUTBO. When a command has been executed it is acknowledged by the robot status signal R2RDY.
When a robot command is set false again RxRDY returns to false (handshaking).
Write an SFC program that will control the robots so that the jobs are fed through the production line in an efficient way. You may assume that all machines are empty at program start. (**3 p**)

Solution

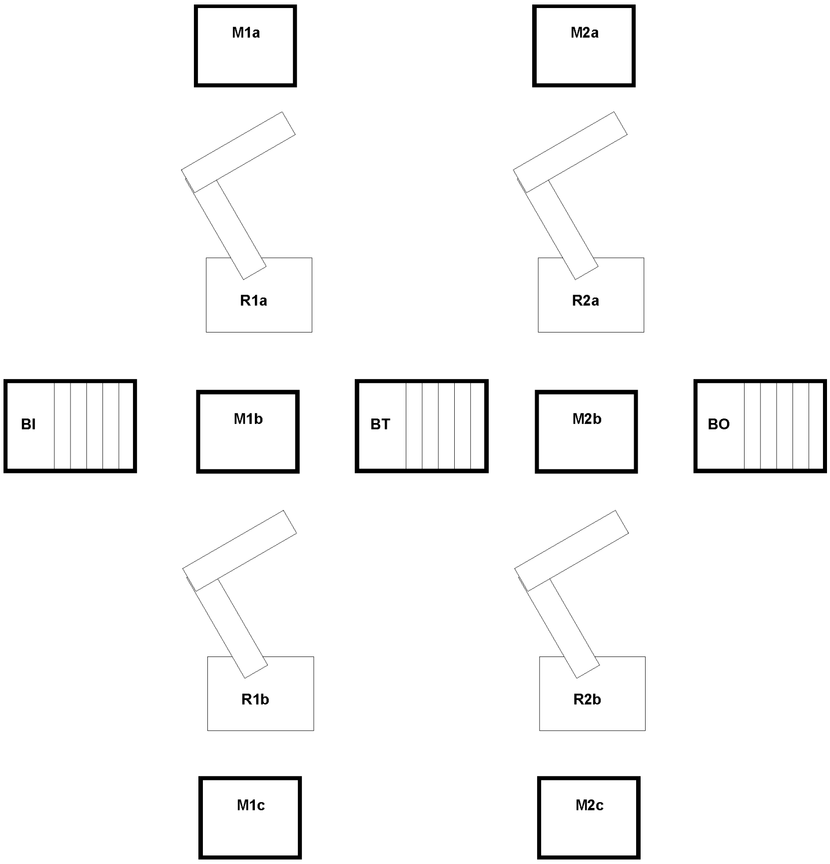
The two robots can work independently. Furthermore, only one job can be handled at a time in the cell since there is only one machine between the buffers. This gives a solution with two parallel execution paths that each form a simple single loop.



The checking of that RxRDY becomes false before a new command is given is maybe not necessary. It depends upon the design and timing of the robot control.

Note that MxIDLE is not used since the robot has got full control of the job and the machines are empty at start.

b) To increase the production the line has been extended. However, the robots are the most costly part of the line and because of that the following structure is used.

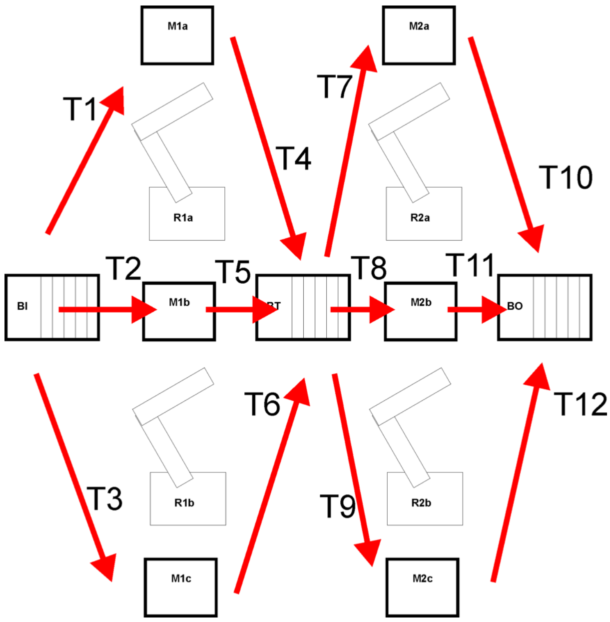


R1a can operate on BI, M1a, M1b and BT while R1b can operate on BI, M1b, M1c and BT. The corresponding is valid for the R2-robots. How should the robot control be structured? You may write the complete SFC program **(not required)** but a graph with program modules and explained communication signals is sufficient. Handling of the BxAV-signals should be clearly described. You should also refer to necessary modifications of your program in the previous task. (**2 p**)

Solution

The situation here is much more complicated and we can immediately see the risk of race conditions and the need for resource protection. Since the IEC61131-3 don't provide tools like semaphores the solution is to introduce a scheduler that assigns tasks to the robot programs.

In the Figure we can see the possible tasks that the robots can get. Not all robots can do all tasks but some tasks can be done by one of two robots. Consequently, we have a lot to keep track of.



It could probably be a good idea to introduce Boolean variable list that indicates if critical resources are allocated to tasks.

R1a can get: T1, T2, T4 and T5

R2b can get: T2, T3, T5 and T6

et cetera.

Actually, all machines, buffers and robots will be critical resources but BT can probably be one input resource and one output depending on the physical design.

When the scheduler looks for possible tasks it will have to check job source available (e.g. B1AV, M1aRDY), destination available (e.g. M1aIDLE) and robot available. Furthermore it will have to check the allocation list so that all resources are free. E.g. even if M1b is idle and R1b is free there might be task T2 started in R1a.

When a task is clear to start the scheduler allocates all necessary resources in the list and then sends the task command to the robot program. As the robot program proceeds it resets the allocations in the list. (E.g. when T2 is executed and the job has been picked from BI the allocation of BI can be released.) It is also probably a good idea to handle the robots in the same way. When a task is assigned the robot is allocated but when the robot has finished the task it reset the allocation just before going to wait for a new task.

The previous program must be extended. For each of the robots we have 4 possible tasks and each will have their own sequence so 16 programs under the double line will be the result. Note that in T2, T5, T8 and T11 the command also must specify which robot. The sequences are now started by job available signals. This will be replaced by a task command. The reset code for the allocation list must also be included.

Problem 10 (2 points)
In program systems with parallel execution of many activities there is sometimes a need for mutual exclusion when operations on variables are made. Take the three cases:

- setting a bit variable,
- incrementing an integer,
- updating a set of PID-controller parameters.

Where is it necessary to implement mutual exclusion? (Additional assumptions might be needed to answer.) What will happen if a necessary protection is not implemented? How can the protection be implemented?

Solution
The need for mutual exclusion depends on the machine language instruction that will make the data manipulation. If more than one instruction have to be executed there is a risk for an interrupt in the middle of the manipulation sequence, but if the complete operation takes place in one indivisible machine language instruction there is no risk.

- Setting a bit variable is always a single indivisible operation so no protection is needed.
- Incrementing an integer could be of both types depending on the data length and the processor. If the integer resides in one word of storage memory there is a good chance that the operation is indivisible. More about the system has to be known. Consequently, the answer has to be that "it depends on…"
- Finally the set of control parameters are probably related and should be changed simultaneously but they are stored in different memory words so protection is needed.

A common way of creating mutual exclusion is the use of semaphores. The provide a "test and set"-operation so that it is possible to in an indivisible operation test a value and depending on the outcome modify the variable or leave it unchanged. In this way the wait()-signal() primitives can guarantee a mutual exclusion.