Computational Foundations of Cyber Physical Systems (CS61063)

Tutorial 4

on Delay-aware Control Strategies using Jittertime/Truetime

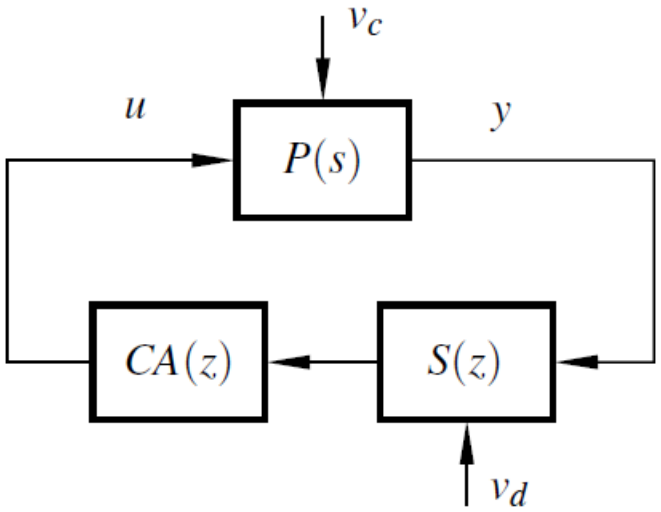
Objective and About the Jittertime

Jittertime be used for calculating the performance of a controller under non-ideal timing conditions.

PreRequisite Softwares*:*

Matlab-Simulink Tool

- JitterTime Source Code - https://www.control.lth.se/research/tools-and-software/jittertime/



Pre Defined Function from Jittertime

jtAddContSys : Add a continuous-time system to a JITTERTIME model.

jtAddDiscSys : Add a discrete-time system to a JITTERTIME model.

jtCalcDynamics : Calculate the total dynamics of a JITTERTIME system before simulation can start.

jtExecSys : Simulate the execution of a discrete-time system.

jtInit : Initialize a new JITTERTIME model.

jtPassTime, jtPassTimeUntil : Simulate the passing of time, integrating the dynamics of all continuous-time systems.

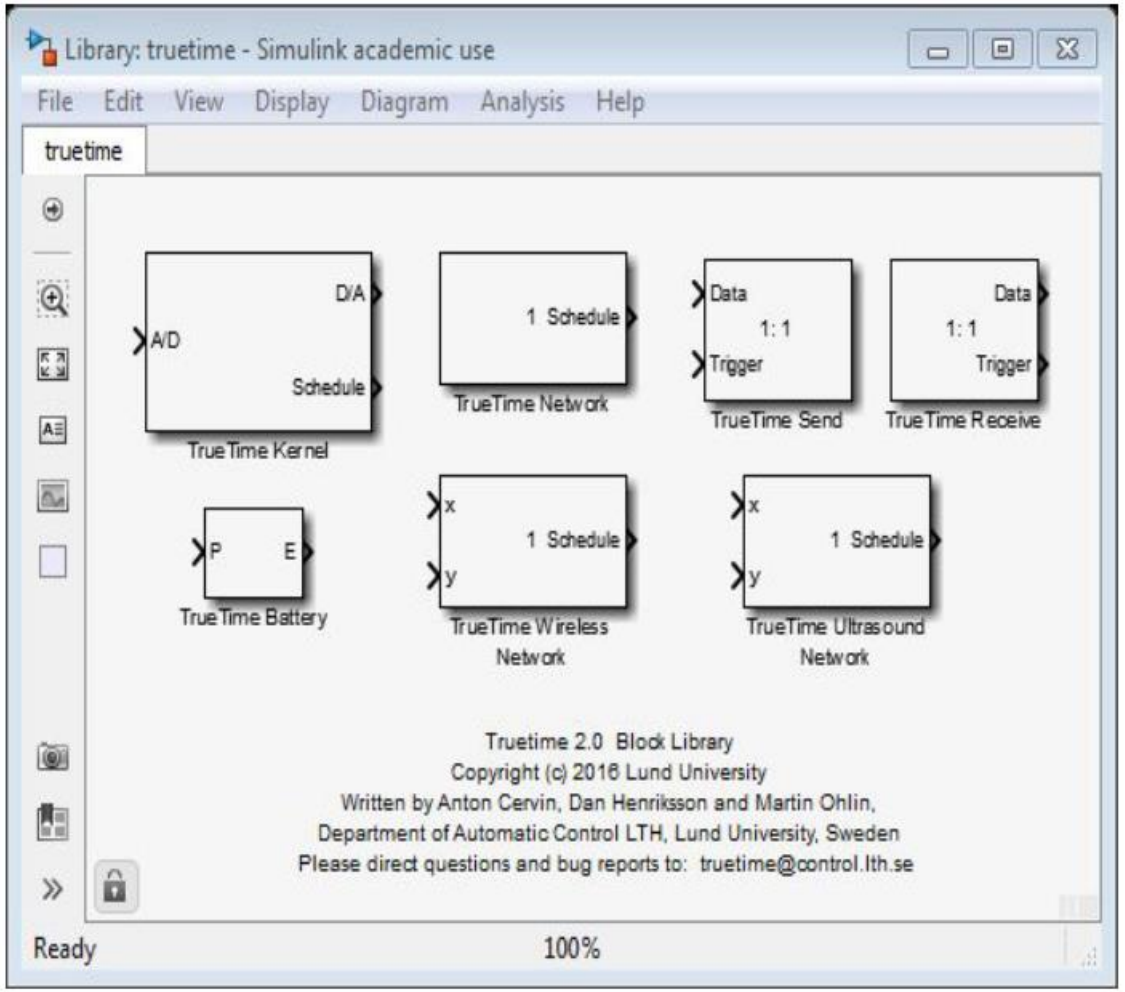
Objective and About the Truetime

Objective of this tutorial is to explore tools for modelling and design of control systems over network.

PreRequisite Softwares*:*

Matlab-Simulink Tool

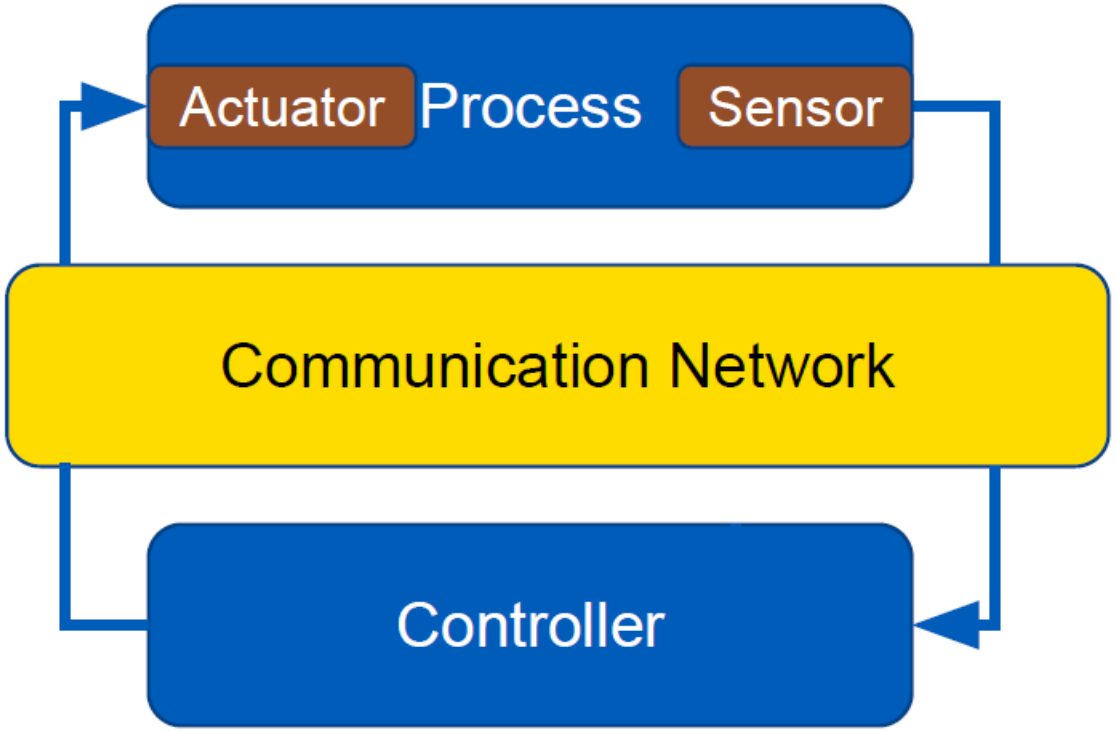
- TrueTime Source Code - http://www.control.lth.se/truetime/

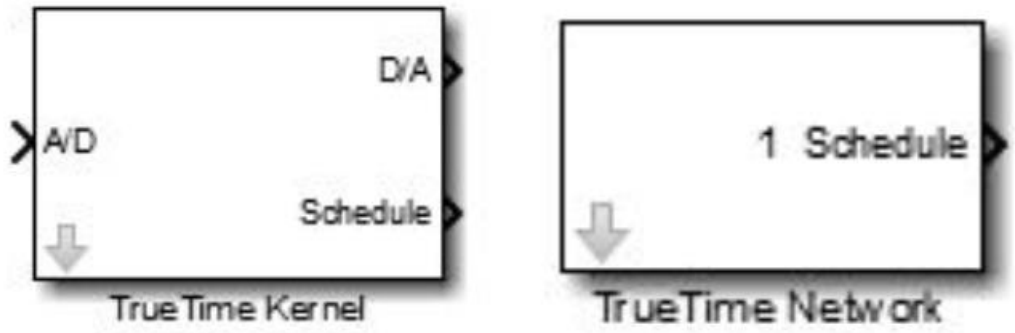


True Time

TrueTime is a Matlab/Simulink
Event-based simulator for real-time control system.

- It is used for the simulation of Networked Control System (NCS).
- In NCS, actuators, sensors and controllers are interconnected by a communication network.





Pre Defined Function from Truetime

ttInitKernel(priofcn): Initialize the kernel, specifying the scheduling policy. There are three predefined scheduling policy are as follow: **prioFP** (fixed-priority scheduling) ,**prioDM** (deadline-monotonic scheduling) & **prioEDF** (earliest-deadline-first)

- **ttAnalogIn(Input channel):** Read a value from an analog input channel
- **ttAnalogOut(output channel):**Write a value to an analog output channel
- **ttSendMsg(receiver, data, length):**Send a message over a TRUETIME network
- **ttGetMsg:**Get a message that has been received over a TRUETIME network

Contd.

ttCreatePeriodicTask(name, starttime, period, codeFcn):
This function is used to create a periodic task to run in the TRUETIME kernel.

- **ttCreateTask(name, deadline, codeFcn):**This function is used to create a task to run in the TRUETIME kernel
- **ttAttachNetworkHandler(networkNbr, taskname):**This function is used to associate an aperiodic task or interrupt handler with a network interface. The task/handler will be invoked every time a message arrives over the network

Truetime initialization and Truetime kernel block

The purposes of the initialization script are:

- ❖ Specify the number of I/O
- ❖ Define a scheduling policy
- ❖ Creating tasks
- ❖ Creating interrupts handlers

The name of the initialization script

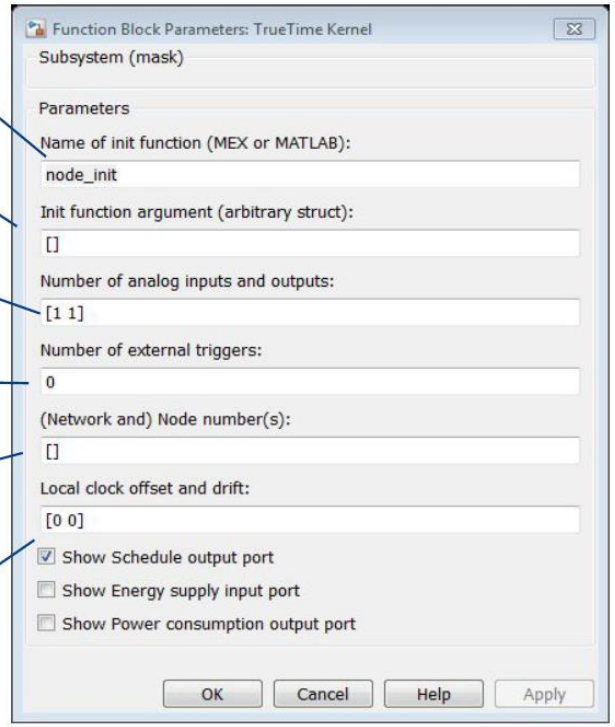
An optional argument to the initialization script

Number of local Inputs and Outputs

Number of external triggers for the kernel task

Number of the node

A constant time offset and time drift from the nominal time.



Sample Problem

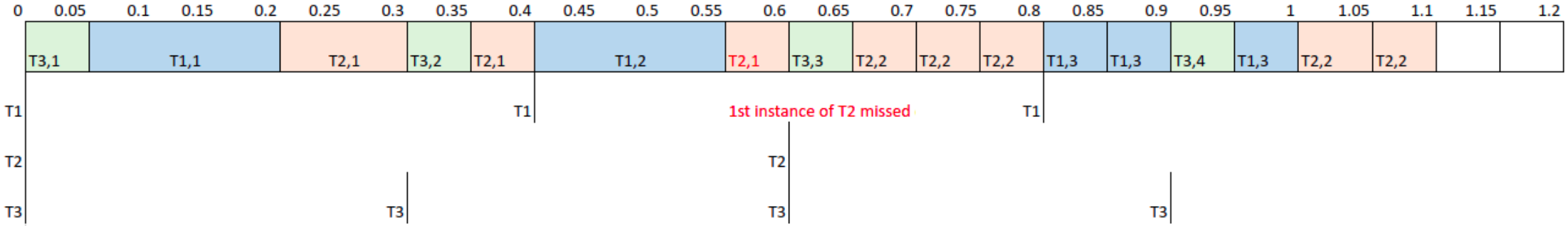
Consider two control tasks T1 and T2 are running on a single processor with periodicity *p1= 0.4* and *p2= 0.6*. They are RM schedulable. Now an authentication task T3 is also introduced with periodicity p3 = 0.3 in order to secure the system against unauthorized access.

Task	Time for Control Task Execution (s)	Deadline (s) (same as their periodicities)	Priority
Control Task T1	0.15	0.4	2
Control Task T2	0.25	0.6	3
Security Task T3	0.05	0.3	1

Scheduling with Delay/Deadline Hit-Miss or Control Execution-Skip Pattern

Solution 1:

- ***Not** RM schedulable since the **1st instance of T2 (T2,1) miss its deadline** in **every hyperperiod** to accommodate others.*
- ***So we kill the task (T2,1)***
- ***Less idle time :: Less reduction in Processor util***



Deadline Hit/Miss Pattern for T1 : **1 1 1** [Arrives 3 times in a hyperperiod]

Deadline Hit/Miss Pattern for T2 : **0 1** [Arrives 2 times in a hyperperiod]

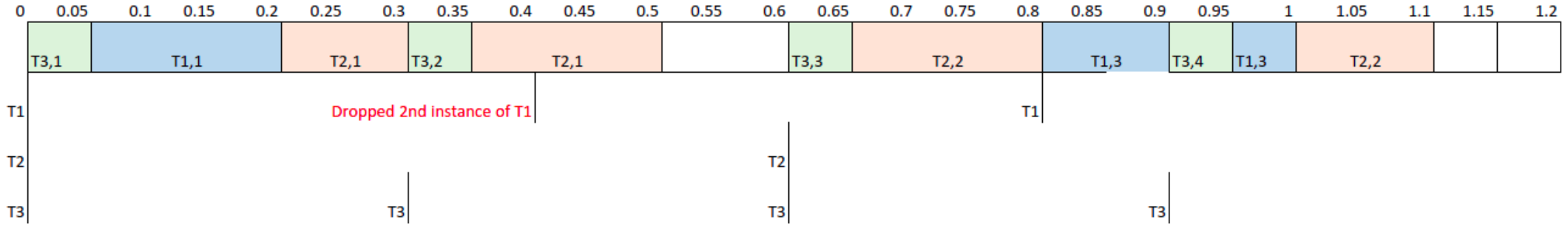
Deadline Hit/Miss Pattern for T3 : **1 1 1 1** [Arrives 4 times in a hyperperiod]

[Deadline Miss is denoted using 0 and Deadline Hit is denoted using 1]

Scheduling with Delay/Deadline Hit/Miss or Control Execution/Skip Pattern

Solution 2:

- *(Considering the robustness/criticality of control)* ***Execution of the 2nd instance of T1 (T1,2) can be skipped*** *in* ***every hyperperiod*** *to accommodate others.*
- ***So we kill the task (T1,2)***
- ***Little more idle time :: Little more reduction in Processor util***



Control Execution/Skip Pattern for T1 : **1 0 1** [Arrives 3 times in a hyperperiod]

Control Execution/Skip Pattern for T2 : **1 1** [Arrives 2 times in a hyperperiod]

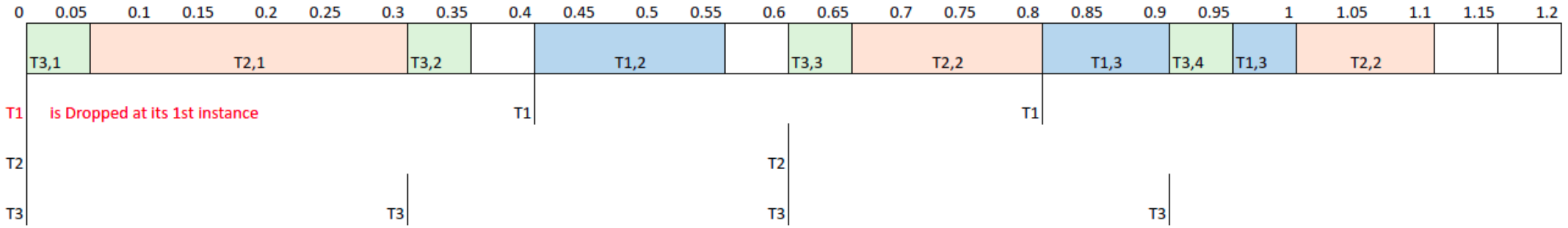
Control Execution/Skip Pattern for T3 : **1 1 1 1** [Arrives 4 times in a hyperperiod]

Control Execution denoted using 1 *and* ***Control Execution Skip denoted using 0***

Scheduling with Control Execution/Skip Pattern

Solution 3:

- *(Considering the robustness/criticality of control)* ***Execution of the 1st instance of T1 (T1,1) can be skipped*** *in* ***every hyperperiod*** *to accommodate others.*
- ***So we kill the task (T1,1)***
- ***Little more idle time :: Little more reduction in Processor util***



Control Execution/Skip Pattern for T1 : **0 11** [Arrives 3 times in a hyperperiod]

Control Execution/Skip Pattern for T2 : **1 1** [Arrives 2 times in a hyperperiod]

Control Execution/Skip Pattern for T3 : **1 1 1 1** [Arrives 4 times in a hyperperiod]

[Control Execution denoted using 1 *and* ***Control Execution Skip denoted using 0]***

Tool Implementation

Tool Implementation

Tool Implementation

Tool Implementation