ISAE Lab class on Hybrid Systems #1
January 27th, 9:15AM-11:30AM

Download first the zip file accessible in the cloud space of the course

1. Reinitialisation of the simulink integrator

Open the simulink file *be_step1.xls*.

a) Build a system which flows with dynamics $\dot{x} = -x$ and which is reinitialized to 2 every five seconds.

Simulate the system to check its behaviour

b) Discuss how the integrator may be used to build a hybrid system defined by its quadruplet (f, $\mathcal{C}$, g, $\mathcal{D}$)

Rely the input/output of the integrator to the elements which constitute a hybrid system

2. Discover the hybrid toolbox

Read through the documentation *HyEQ_Toolbox_v203.pdf*. Do not read it in details, but do not hesitate to refer to it during the labs. This is a toolbox that you do not need to install for this lab (even if we will use several materials from this toolbox).

3. Play with a counter

Let us consider a hybrid system capturing the dynamics of a timer with reset to zero: the timer, denoted x, counts time continuously and when it reaches the value x^*, is reset to zero. For this system, the flow set is given as

$$C = [0, x^*]$$

while the jump set as

$$D = \{x \in \mathbb{R} \ : \ x = x^* \ \}$$

The flow map is simply equal to 1 while the jump map is equal to 0. A preliminary version of this system is given in the simulink file *be_counter.xls*. Run fist *be_counter_init.m*, then the simulink file.

a) Does it look as a timer?

Run the files, plot a figure (use plotflows.m and plotjumps.m) and comment.

b) Can the state x be arbitrarily initialized to generate a trajectory that represents a timer with resets to zero? Explain.

Select change the initial conditions to figure that out.

c) Define new flow and jump sets so that the issues in a) and b) are resolved. Validate it numerically and plot for 10 seconds (and submit) three representative trajectories.

Identify how are implemented continuous and discrete dynamics, how are defined flow and jump sets.

d) In the simulink model, explain the role of the *stop logic* block hidden in the IntegratorSystem.

Look under the mask to access this block

e) In the simulink model, explain the role of the *Jump logic* block hidden in the IntegratorSystem.

Look under the mask to access this block

4. The famous bouncing ball

Let us consider the hybrid simulator be_bouncingball.xls capturing the bouncing ball behavior described during the class as follows:

$$\dot{x} = f(x) = \begin{bmatrix} x_2 \\ -\gamma \end{bmatrix} \text{ if } x \in \mathcal{C} = \{x \in \Re^2; x_1 \geq 0\}$$
$$x^+ = g(x) = \begin{bmatrix} 0 \\ -\lambda x_2 \end{bmatrix} \text{ if } x \in \mathcal{D} = \{x \in \Re^2; x_1 = 0 \text{ and } x_2 \leq 0\}$$

a) Explain if there are any issues in obtaining appropriate trajectories.

First run *be_bouncingball_init.m*, then the simulink model *be_bouncingball.xls*.

b) Can the state x be arbitrarily initialized to generate a trajectory that represents a ball bouncing? Explain

c) Define a new jump set so that the issue in a) is resolved. Validate it numerically and plot for 20 seconds (and submit) a trajectory starting with unitary height and unitary velocity (both positive). Report any problem you may experience.

Modify the simulink and initialization files before to simulate

d) For the same trajectory in c), plot the energy of the ball as a function of ordinary time. Justify its shape.

Remember that the energy is the sum of the kinetic and potential energies.