1 Examples

The examples below illustrate the use of the Simulink implementation above.

Example 1.3 (bouncing ball with input) For the simulation of the bouncing ball system with a constant input and regular data given by

$$f(x,u) := \begin{bmatrix} x_2 \\ -\gamma \end{bmatrix}, C := \left\{(x,u) \in \mathbb{R}^2 \times \mathbb{R} \mid x_1 \geq u\right\} \tag{1}$$

$$g(x,u) := \begin{bmatrix} u \\ -\lambda x_2 \end{bmatrix}, D := \left\{(x,u) \in \mathbb{R}^2 \times \mathbb{R} \mid x_1 \leq u \ , \ x_2 \leq 0\right\} \tag{2}$$

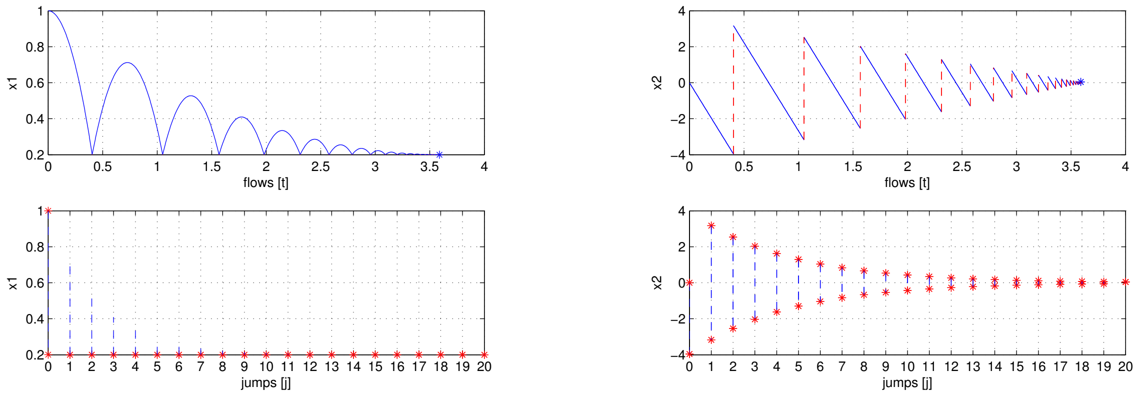
where $\gamma > 0$ is the gravity constant, u is the input constant, and $\lambda \in [0,1)$ is the restitution coefficient. The MATLAB scripts in each of the function blocks of the implementation above are given as follows. An input was chosen to be $u(t,j) = 0.2$ for all (t,j). The constants for the bouncing ball system are $\gamma = 9.81$ and $\lambda = 0.8$.

The following procedure is used to simulate this example using the model in the file `Example_1_2a.slx`:

- `Example_1_2a.slx` is opened in MATLAB/Simulink.
- The *Embedded MATLAB function blocks f, C, g, D* are edited by double-clicking on the block and editing the script. In each embedded function block, parameters must be added as inputs and defined as parameters by selecting `Tools>Edit Data/Ports`, and setting the scope to `Parameter`. For this example, *gamma* and *lambda* are defined in this way.
- The initialization script `initialization.m` is edited by opening the file and editing the script. The flow time and jump horizons, T and J are defined as well as the initial conditions for the state vector, x_0, and input vector, u_0, and a rule for jumps, *rule*.
- The postprocessing script `postprocessing.m` is edited by opening the file and editing the script. Flows and jumps may be plotted by calling the functions *plotflows* and *plotjumps*, respectively. The hybrid arc may be plotted by calling the function *plotHybridArc*.
- The simulation stop time and other simulation parameters are set to the values defined in `initialization.m` by selecting `Simulation>Configuration Parameters>Solver` and inputting T, $RelTol$, $MaxStep$, etc..
- The masked integrator system is double-clicked and the simulation horizons and initial conditions are set as desired.
- The block labeled *Double Click to Initialize* is double-clicked to initialize variables.
- The simulation is run by clicking the run button or selecting `Simulation>Start`.
- The block labeled *Double Click to Plot Solutions* is double-clicked to plot the desired solutions.

Flow map

```
function xdot = f_ex1_2a(x)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
% http://hybridsimulator.wordpress.com/
% Filename: f_ex1_2a.m
%--------------------------------------------------------------------------
% Project: Simulation of a hybrid system (bouncing ball)
% Description: Flow map
```



(a) Height

(b) Velocity

Figure 1: Solution of Example 1.3

```
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00

% state
x1 = x(1);
x2 = x(2);

global gamma

% differential equations
xdot = [x2 ; gamma];
end
```

Flow set

```
function [value] = C_ex1_2a(x)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
% http://hybridsimulator.wordpress.com/
% Filename: C_ex1_2a.m
%--------------------------------------------------------------------------
% Description: Flow set
% Return 0 if outside of C, and 1 if inside C
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00

```

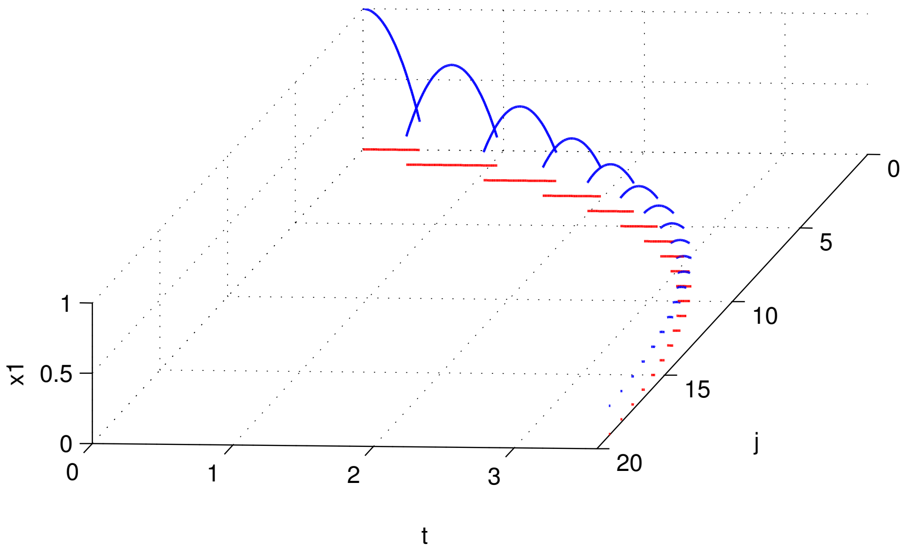


Figure 2: Hybrid arc corresponding to a solution of Example 1.3: height

```

x1 = x(1);

if x1 >= 0
    value = 1;
else
    value = 0;
end
end
```

Jump map

```
function xplus = g_ex1_2a(x)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
% http://hybridsimulator.wordpress.com/
% Filename: g_ex1_2a.m
%--------------------------------------------------------------------------
% Project: Simulation of a hybrid system (bouncing ball)
% Description: Jump map
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00
```

```
% state
x1 = x(1);
x2 = x(2);

global lambda

xplus = [-x1 ; -lambda*x2];
end
```

Jump set

```
function inside = D_ex1_2a(x)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
% http://hybridsimulator.wordpress.com/
% Filename: D_ex1_2a.m
%--------------------------------------------------------------------------
% Description: Jump set
% Return 0 if outside of D, and 1 if inside D
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00

x1 = x(1);
x2 = x(2);
if (x1 <= 0 && x2 <= 0)
    inside = 1;
else
    inside = 0;
end
end
```

A solution to the bouncing ball system from $x(0,0) = [1,0]^\top$ and with $T = 10, J = 20$, $rule = 1$, is depicted in Figure 1(a) (height) and Figure 1(b) (velocity). Both the projection onto t and j are shown. Figure 2 depicts the corresponding hybrid arc for the position state.

These simulations reflect the expected behavior of the bouncing ball model. Note the only difference between this example and the example of a bouncing ball without a constant input is that, in this example, the ball bounces on a platform at a height of the chosen input value 0.2 rather than the ground at a value of 0.

For MATLAB files of this example, see Examples/Example_1.2a.

□