MINOTTO GABRIELE - INGENERIA MECCATRONICA - A.A. 2015 - 2016
Exercise: monocycle model in chained form, using piecewise functions

```
> restart: with(plots):
> # Data Inputs.
  data := [T = 1, n=3];
```

$$data := [T = 1, n = 3] \tag{1}$$

Model definition

```
> eq_tr := <x__1(t) = -delta(t), x__2(t) = x(t)*cos(delta(t)) + y
  (t)*sin(delta(t)), x__3(t) = -x(t)*sin(delta(t)) + y(t)*cos(delta
  (t))>;
```

$$eq\_tr := \begin{bmatrix} x_1(t) = -\delta(t) \\ x_2(t) = x(t)\cos(\delta(t)) + y(t)\sin(\delta(t)) \\ x_3(t) = -x(t)\sin(\delta(t)) + y(t)\cos(\delta(t)) \end{bmatrix} \tag{2}$$

```
> eq_v := [v__1(t) = x__3(t)*u__1(t) + u__2(t), v__2(t) = -u__1(t)]
  ;
```

$$eq\_v := [v_1(t) = x_3(t)\,u_1(t) + u_2(t), v_2(t) = -u_1(t)] \tag{3}$$

```
> in_cond := [0, 0, 0];
  fi_cond := [1, 1, Pi/2];
```

$$in\_cond := [0, 0, 0]$$

$$fi\_cond := \left[1, 1, \frac{1}{2}\,\pi\right] \tag{4}$$

1) Initial transformed conditions

```
> in_tr_cond := convert(evalf(subs(x(t)=in_cond[1],y(t)=in_cond[2],
  delta(t)=in_cond[3],eq_tr)),list);
  fi_tr_cond := convert(evalf(subs(x(t)=fi_cond[1],y(t)=fi_cond[2],
  delta(t)=fi_cond[3],eq_tr)),list);
```

$$in\_tr\_cond := [x_1(t) = 0., x_2(t) = 0., x_3(t) = 0.]$$

$$fi\_tr\_cond := [x_1(t) = -1.570796327, x_2(t) = 0.9999999998, x_3(t) = -1.000000000] \tag{5}$$

2) Divide time int.

```
> n_i := subs(data,n-1);
```

$$n\_i := 2 \tag{6}$$

```
> delta_T := subs(data,T/n_i);
```

$$delta\_T := \frac{1}{2} \tag{7}$$

3)Input u__1 e u__2 in symbolic form (convert operation is useful to extract informations from piecewise functions)

```
> u__1s(t) := (x__f1 - x__i1)/T; #symbolic form
  u__1(t) := subs(data,x__f1=rhs(fi_tr_cond[1]), x__i1=rhs
  (in_tr_cond[1]),u__1s(t));
```

$$u_{1s}(t) := \frac{x_{f1} - x_{i1}}{T}$$

$$u_1(t) := -1.570796327 \tag{8}$$

```
> u__2(t) := piecewise(0 <= t and t <= delta_T, u__21, delta_T <= t
```

```
  and t <= n_i*delta_T, u__22);
```

$$u_2(t) := \begin{cases} u_{21} & 0 \le t \text{ and } t \le \frac{1}{2} \\ u_{22} & \frac{1}{2} \le t \text{ and } t \le 1 \end{cases} \quad (9)$$

```
> x__i(t) := convert(<x__1(t) = int(rhs(diff_ui(t)[1]),t) + rhs
  (in_tr_cond[1]), x__2(t) = int(rhs(diff_ui(t)[2]),t) + rhs
  (in_tr_cond[2])>,list);
Error, invalid input: rhs received diff_ui(t)[1], which is not
valid for its 1st argument, expr
> convert(<diff(x__3(t),t) = x__2(t)*u__1(t)>,list);
  diff_u3(t) := subs(x__i(t),%);
```

$$\left[\frac{d}{dt} x_3(t) = -1.570796327\, x_2(t)\right]$$

```
Error, invalid input: subs received x__i(t), which is not valid
for its 1st argument
> # Generalized velocities and positions.
  vel_gen := [op(diff_ui(t)), op(diff_u3(t))];
  pos_gen := [op(x__i(t)), simplify(x__i3)];
```

$$vel\_gen := [t, t]$$

$$pos\_gen := [t, x_{i3}] \quad (10)$$

4 -5) Constraint initial and final conditions - linear system. Then solve for the u__2i unknowns

```
> sys__u2 := op(solve(subs(subs(data,t=T),[convert(rhs(pos_gen[2]),
  list)[6] = rhs(fi_tr_cond[2]), convert(rhs(pos_gen[3]),list)[6] =
  rhs(fi_tr_cond[3])]),[u__21, u__22]));

Error, invalid input: rhs received x__i3, which is not valid for
its 1st argument, expr
```

6) Obtain x__i(t) from u__2

7) Obtain generalized coordinate x, y, delta

8) Find the two inputs

```
> the_controls := subs(x__i_final,sol__u2,eq_v);
Error, invalid input: subs received x__i_final, which is not
valid for its 1st argument
```

9) Plot trajectories & controls

```
> P1 := plot([subs(gen_coord[1],x(t)), subs(gen_coord[2],y(t)), t =
  0 .. 1], color="DarkOrange",gridlines=true,labels=["x(t) [m]","y
  (t) [m]"],title="Trajectory in time",axes=boxed):
  P2 := plot([subs(the_controls[1],v__1(t)), subs(the_controls[2],
  v__2(t))], t = 0 .. 1,color=["DarkOrange","DarkMagenta"],
  gridlines=true,labels=["t [s]","v(t)"],title="Controls in time",
  axes=boxed,legend=["v__1(t) [m/s]","v__2(t) [rad/s]"]):
  display(array(1..2,[P1,P2]));

Error, invalid input: subs received gen_coord[1], which is not
valid for its 1st argument
```

```
Error, invalid input: subs received the_controls[1], which is
not valid for its 1st argument
Error, (in plots:-display) element 1 of the rtable is not a
valid plot structure
```

NOTES: third initial and final conditions must not be the same, because equation related to x3(t) is always equal to zero and another setting of piecewise equations must be performed.