Crane Model

Kenneth Kuchera

October 23, 2012

1 Model

This example is located in *svn/software/examples/simulink/crane/Branch/v0.5*. The model used is described in deliverable 8.2. We will start by restating the model equations here. We will use these equations which describe a non-linear continuous model to derive a linear discrete model. The matrices that are used in the simulation as well as the process of linearizing, discretizing, can be found in the file *craneInformation.m*.

1.1 Non-Linear Continuous Model

The non linear model is given by the following equations.

$$\begin{aligned}
\dot{x}_C &= v_C \\
\dot{x}_L &= v_L \\
\dot{v}_C &= \frac{-1}{\tau_C} v_C + \frac{A_C}{\tau_C} u_C \\
\dot{v}_L &= \frac{-1}{\tau_L} v_L + \frac{A_L}{\tau_L} u_L \\
\dot{\theta} &= \omega \\
\dot{\omega}_T &= -\frac{1}{x_L}\left(g sin(\theta) + \left(\frac{-1}{\tau_C} v_C + \frac{A_C}{\tau_C} u_C\right) cos(\theta) + 2 v_L \omega + \frac{c\omega}{m x_L}\right) \\
\dot{u}_C &= u_{CR} \\
\dot{u}_L &= u_{LR}
\end{aligned} \tag{1}$$

In a first step we will linearize this. In a second step we will discretize it. This is because we want to use a linear discrete model for the MPC instead of a non-linear continuous model.

1.1.1 Linear Continuous Model

Linearizing the model from equation 1 around a steady state yields an equation of the form $\dot{x} = \mathbf{A_c}(x - x^*) + \mathbf{B_c}(u - u^*)$. Expanding this equation gives

$$
\underbrace{\begin{bmatrix} \dot{x}_C \\ \dot{x}_L \\ \dot{v}_C \\ \dot{v}_L \\ \dot{\theta} \\ \dot{\omega} \\ \dot{u}_C \\ \dot{u}_L \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{-1}{\tau_C} & 0 & 0 & 0 & \frac{A_C}{\tau_C} & 0 \\ 0 & 0 & 0 & \frac{-1}{\tau_L} & 0 & 0 & 0 & \frac{A_L}{\tau_L} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{\partial \dot{\omega}}{\partial x_L} & \frac{\partial \dot{\omega}}{\partial v_C} & \frac{\partial \dot{\omega}}{v_L} & \frac{\partial \dot{\omega}}{\partial \theta} & \frac{\partial \dot{\omega}}{\partial \omega} & \frac{\partial \dot{\omega}}{\partial u_C} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{A_c}} \underbrace{\begin{bmatrix} x_C - x_C^* \\ x_L - x_L^* \\ v_C - v_C^* \\ v_L - v_L^* \\ \theta - \theta^* \\ \omega - \omega^* \\ u_C - u_C^* \\ u_L - u_L^* \end{bmatrix}}_{(x-x^*)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\mathbf{B_c}} \underbrace{\begin{bmatrix} u_{CR} - u_{CR}^* \\ u_{LR} - u_{LR}^* \end{bmatrix}}_{(u-u^*)} \qquad (2)
$$

with

$$
\begin{aligned}
\frac{\partial \dot{\omega}}{\partial x_L} &= \frac{1}{x_L^2}\left(gsin(\theta) + \left(\frac{-1}{\tau_C}v_C + \frac{A_C}{\tau_C}u_C\right)cos(\theta)) + 2v_L\omega + 2\frac{c\omega}{mx_L}\right) \\
\frac{\partial \dot{\omega}}{\partial v_C} &= \frac{cos(\theta)}{\tau_C x_L} \\
\frac{\partial \dot{\omega}}{\partial v_L} &= \frac{-2\omega}{x_L} \\
\frac{\partial \dot{\omega}}{\partial \theta} &= \frac{1}{x_L}\left(-gcos(\theta) + \left(-\frac{v_C}{\tau_C} + \frac{A_C}{\tau_C}u_C\right)sin(\theta)\right) \\
\frac{\partial \dot{\omega}}{\partial \omega} &= \frac{-2v_L}{x_L} \\
\frac{\partial \dot{\omega}}{\partial u_C} &= \frac{-A_C cos(\theta)}{\tau_C x_L}
\end{aligned}
$$

1.2 Linear Discrete Model

The discretization takes place by constructing a matrix M, taking the exponent and finally extracting the matrices for the discrete model. This is done as follows.

$$
\begin{bmatrix} \mathbf{A_d} & \mathbf{B_d} \\ \mathbf{0_{n_u \times n_x}} & \mathbf{I_{n_u \times n_u}} \end{bmatrix} = exp\left(\begin{bmatrix} \mathbf{A_c} t_s & \mathbf{B_c} t_s \\ \mathbf{0_{n_u \times n_x}} & \mathbf{0_{n_u \times n_u}} \end{bmatrix}.\right)
$$

Where t_s is the sampling rate, $\mathbf{A_d}$ and $\mathbf{B_d}$ the matrices $\mathbf{A}$ and $\mathbf{B}$ respectively for the discrete system.

1.3 Implementation

The values used in the implementation are listed in the tables below. They can also be found in the file *craneInformation.m* which is located in the repository.

Note: The point around which we have linearized (working point) is also the setpoint.

Constants		
Constant	Value	Description
m	1318	mass (g)
g	9.81	gravity ($\frac{m}{s^2}$)
c	0	damping ($\frac{kgm^2}{s}$)
A_C	0.047418203070092	Gain of cart vel. ctrl. ($\frac{m}{s}V$)
τ_C	0.012790605943772	time constant cart ctrl. (s)
A_L	0.034087337273386	Gain of winch vel. ctrl. ($\frac{m}{s}V$)
τ_L	0.024695192379264	time constant winch ctrl. (s)

Working point		
Working point	Value	Description
x_C^*	1	cart position (m)
x_L^*	1.5	cable length (m)
v_C^*	0	cart velocity ($\frac{m}{s}$)
v_L^*	0	cable velocity ($\frac{m}{s}$)
θ^*	0	cable deflection (rad)
ω^*	0	cable deflection rate ($\frac{rad}{s}$)
u_C^*	0	cart controller voltage (V)
u_L^*	0	cable controller voltage (V)

Initial point		
Initial point	Value	Description
x_C	0	cart position (m)
x_L	0.75	cable length (m)
v_C	0	cart velocity ($\frac{m}{s}$)
v_L	0	cable velocity ($\frac{m}{s}$)
θ	0	cable deflection (rad)
ω	0	cable deflection rate ($\frac{rad}{s}$)
u_C	0	cart controller voltage (V)
u_L*	0	cable controller voltage (V)

2 Model Predictive Control

A standard MPC algorithm is used with the following costs.

$$\begin{aligned}
&Q = diag(100, 100, 100, 100, 100, 100, 1, 1) \\
&R = diag(1, 1) \\
&P = \text{solution to the Ricatti equation for the discrete system, see craneInformation.m}
\end{aligned} \tag{3}$$

Here Q is the cost on the states, R is the cost on the inputs and P is the terminal cost on the states.