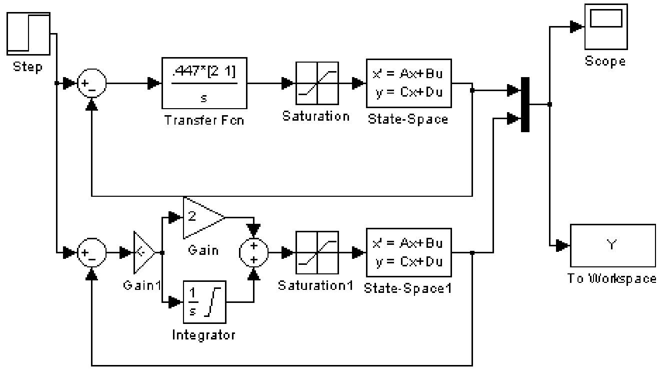
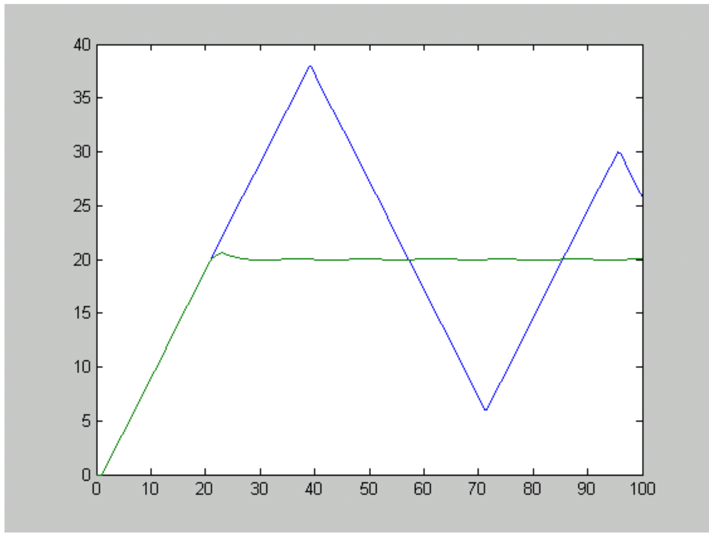
Case-Study 1: Effects of Saturating actuators

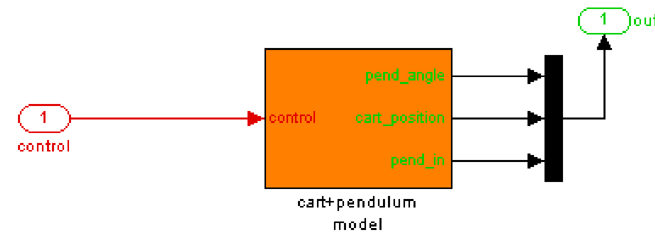
- Let $G(s) = \dfrac{1}{s}$ (approx. heating, car speed etc)
- Design a PI compensator,
  - e.g., crossover ~ 1, PM=63° => $C(s) = 0.447\dfrac{2s+1}{s}$
- Large reference inputs require large control inputs during transient (eventually, u --> 0). If the actuator saturates, the error does not decrease fast enough and the integrator "winds-up."
  - Check the "controller output" vs. time
- Remedies: Limited integrators, Use controller feedback with the difference between linear and saturated control, Bumpless transfer techniques…



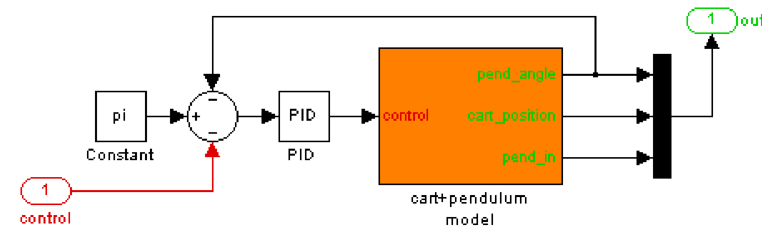


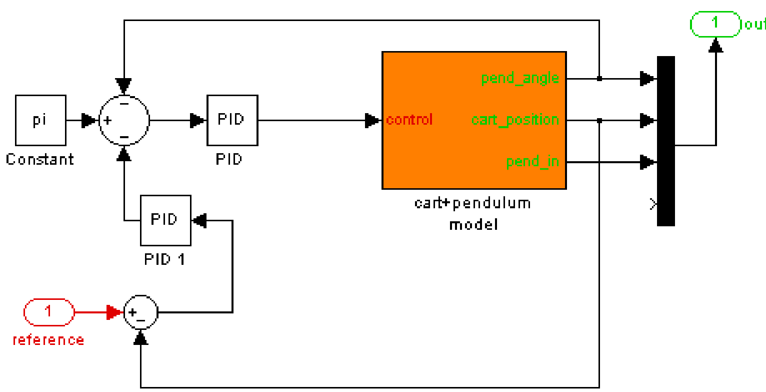
Case Study 2: Cart+Pendulum

- Model: 1-input, 2-outputs



- Controller structure:
  - 1. Stabilize the angle with an inner-loop controller





  - 2. Control the cart position with an outer-loop controller

Case Study 2: Cart+Pendulum

- Inner-loop: zero at the origin, RHP pole
- RL insight: Need a compensator with a pole in the RHP and two lead-zeros.
- General form: PID-like controller

$$C(s) = K\frac{(s+z_1)(s+z_2)}{(s-\varepsilon)(\tau_p s+1)} = K_P + \frac{K_I}{s-\varepsilon} + \frac{K_D s}{\tau_p s+1}$$

- Extract the angle model
  - [A,B,C,D]=linmod('pc_mdl');
  - [ni,di]=ss2tf(A,B,C(1,:),D(1,:))
- Tune for a sensitivity bandwidth of ~10
  - PID=pidqtune(10,ni,di,[-0.05,0.001,0,-0.05]);
  - ci=tf(PID(1,:),PID(2,:))

Case Study 2: Cart+Pendulum

- Outer-loop: two poles near/at the origin, RHP zero
- Controller: PID should do.
  - Keep bandwidth low
  - Limit derivative action
- Introduce inner-loop controller and Extract the position model
  - [A,B,C,D]=linmod('pc_mdl1');
  - [no,do]=ss2tf(A,B,C(2,:),D(2,:))
- Tune for a sensitivity bandwidth of ~0.4
  - PID=pidqtune(0.4,no,do,[0,0.01,0],0.3);
  - co=tf(PID(1,:),PID(2,:))

Case Study 2: Cart+Pendulum

- Check Nyquist + Root-Locus plots
  - Inner loop: 2 RHP poles -> 2 ccw encirclements
  - Outer loop: Essentially a 3-integrator problem
- Implement controller in Simulink
- … together with a variant of energy control to swing up the pendulum…
  - iterate on controller design parameters (bandwidth, prefilters) and, possibly, hardware parameters (mass, length, torque) to achieve reasonable behavior (required track length, motor size, etc)