Topic #23

16.30/31 Feedback Control Systems

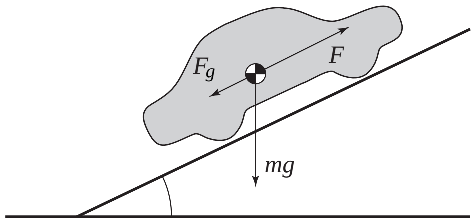
Analysis of Nonlinear Systems

- Anti-windup

- Notes developed in part from 16.30 Estimation and Control of Aerospace Systems, Lecture 21: Lyapunov Stability Theory by Prof. Frazzoli

Nonlinear System Analysis

- Example: Car cruise control[1]



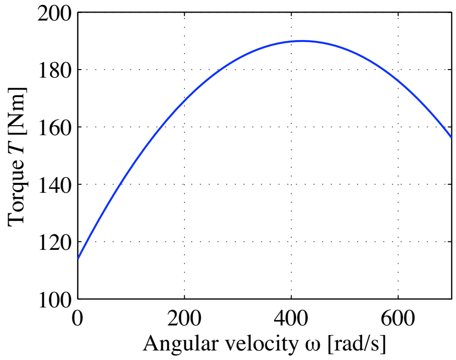
- Equation of motion in the direction parallel to the road surface:

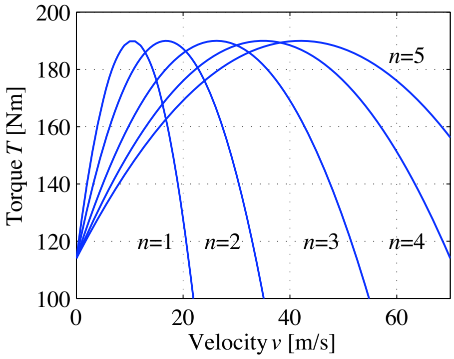
$$m\frac{dv}{dt} = F_{\text{eng}} + F_{\text{aero}} + F_{\text{frict}} + F_{\text{g}}.$$

where

$$\begin{aligned} F_{\text{aero}} &= -\frac{1}{2}\rho C_{\text{d}} A v \cdot |v|, \\ F_{\text{g}} &= -mg\sin(\theta), \\ F_{\text{frict}} &= -mgC_{\text{r}}\cos(\theta)\texttt{sgn}(\texttt{v}). \end{aligned}$$

- Engine model





- Engine torque (at full throttle): $T_\omega = T_m\left(1 - \beta\left(\frac{\omega}{\omega_m} - 1\right)^2\right)$, where $\omega = \frac{n}{r}v = \alpha_n v$, n is gear ratio, and r wheel radius.
- The engine driving force can hence be written as

$$F_{\text{eng}} = \alpha_n T(\alpha_n v)u, \qquad 0 \le u \le 1.$$

[1]The example is taken from Åström and Murray: Feedback Systems, 2008

Jacobian Linearization

- Any (feasible) speed corresponds to an equilibrium point.
- Choose a reference speed $v_{ref} > 0$, and solve for $dv/dt = 0$ with respect to u, assuming a horizontal road ($\theta = 0$).

$$0 = \alpha_n T(\alpha_n \bar{v})\bar{u} - \frac{1}{2}\rho C_\mathrm{d} A \bar{v}^2 - mgC_\mathrm{r}$$

i.e.,

$$\bar{u} = \frac{\frac{1}{2}\rho C_\mathrm{d} A \bar{v}^2 + mgC_\mathrm{r}}{\alpha_n T(\alpha_n \bar{v})}.$$

- Linearized system ($\xi = v - \bar{v}$, $\eta = u - \bar{u}$):

$$\frac{d}{dt}\xi = \underbrace{\frac{1}{m}\left(\alpha_n \left.\frac{\partial T(\alpha_n v)}{\partial v}\right|_{\bar{v}} \bar{u} - \rho C_\mathrm{d} A \bar{v}\right)}_{A_{dyn}} \xi + \underbrace{\frac{1}{m}\alpha_n T(\alpha_n \bar{v})}_{B_{dyn}} \eta$$

- Example: numerical values
  - Let us use the following numerical values (all units in SI):

$$T_m = 190, \beta = 0.4, \omega_m = 420, \alpha_5 = 10, C_\mathrm{r} = 0.01,$$

$$m = 1500, g = 9.81, \rho = 1.2, C_\mathrm{d} A = 0.79.$$

  - For $\bar{v} = 25$ (90 km/h, or 55 mph), we get $\bar{u} = 0.2497$.
  - The linearization yields:

$$A_{dyn} = -0.0134, \qquad B_{dyn} = 1.1837$$

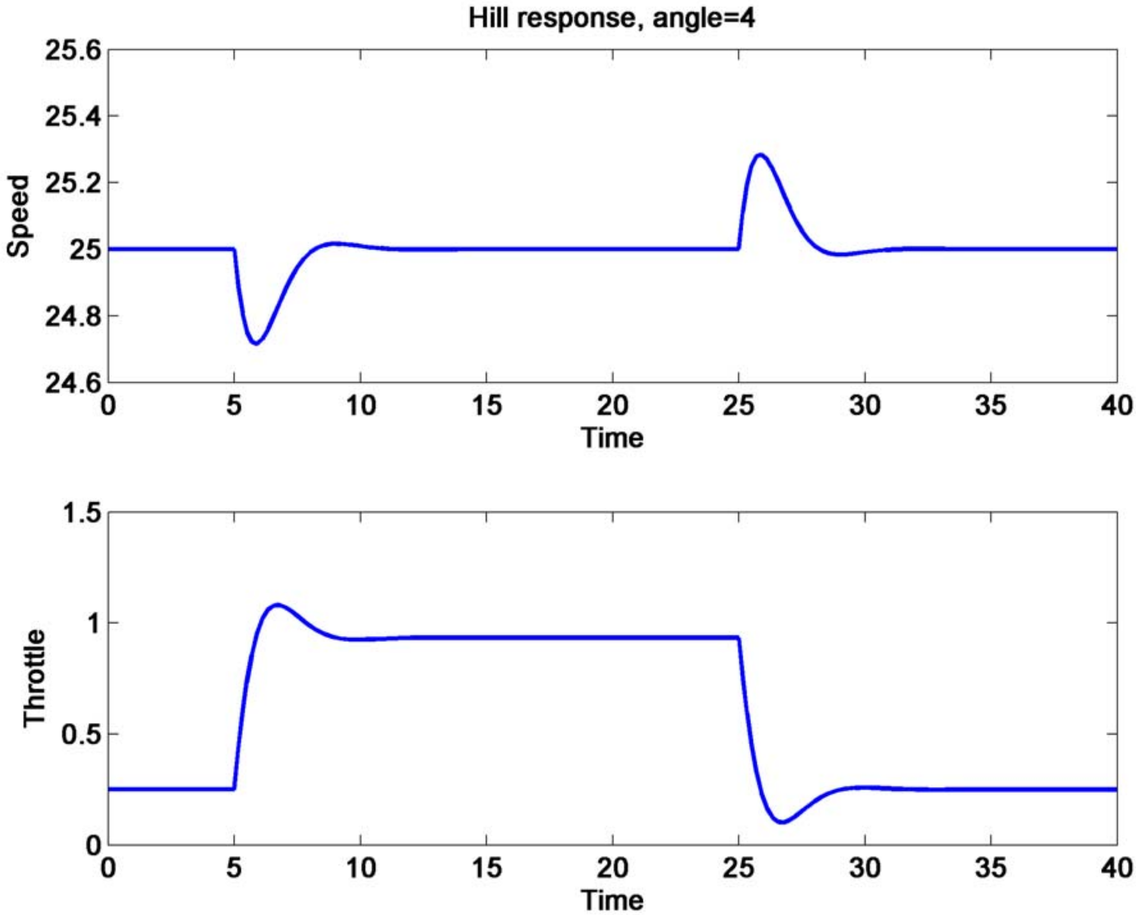
$$\Rightarrow \quad G(s) = \frac{1.1837}{s + 0.0134}$$

Cruise control design

- A proportional controller would stabilize the closed-loop system.
- Assume we want to maintain the commanded speed (cruise control): we need to add an integrator.
- A PI controller will work, e.g.,

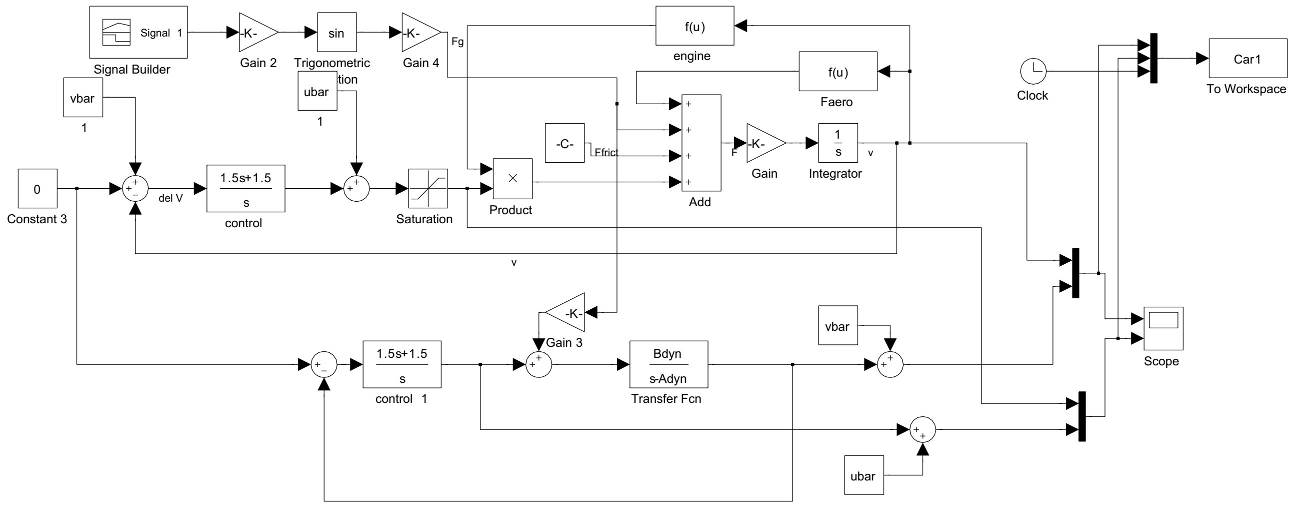
$$C(s) = 1.5\frac{s+1}{s}$$

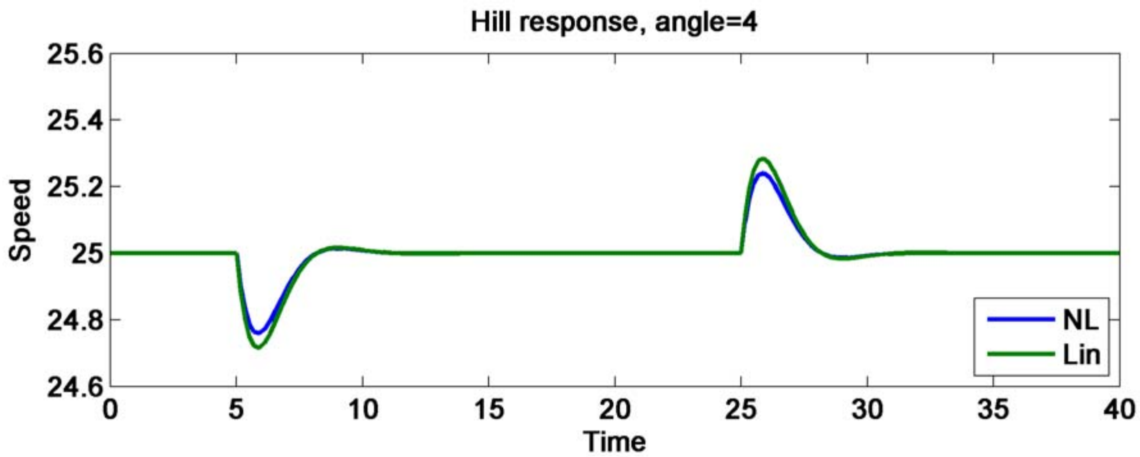
- Linear control/model response to hill at specified angle between 5 and 25 sec

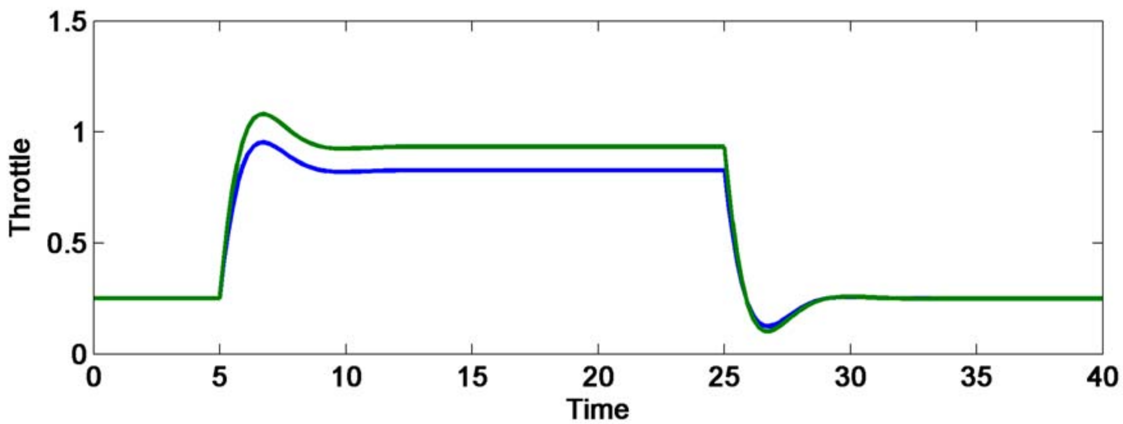


Nonlinear Simulation

- Check with BOTH linear AND nonlinear simulation







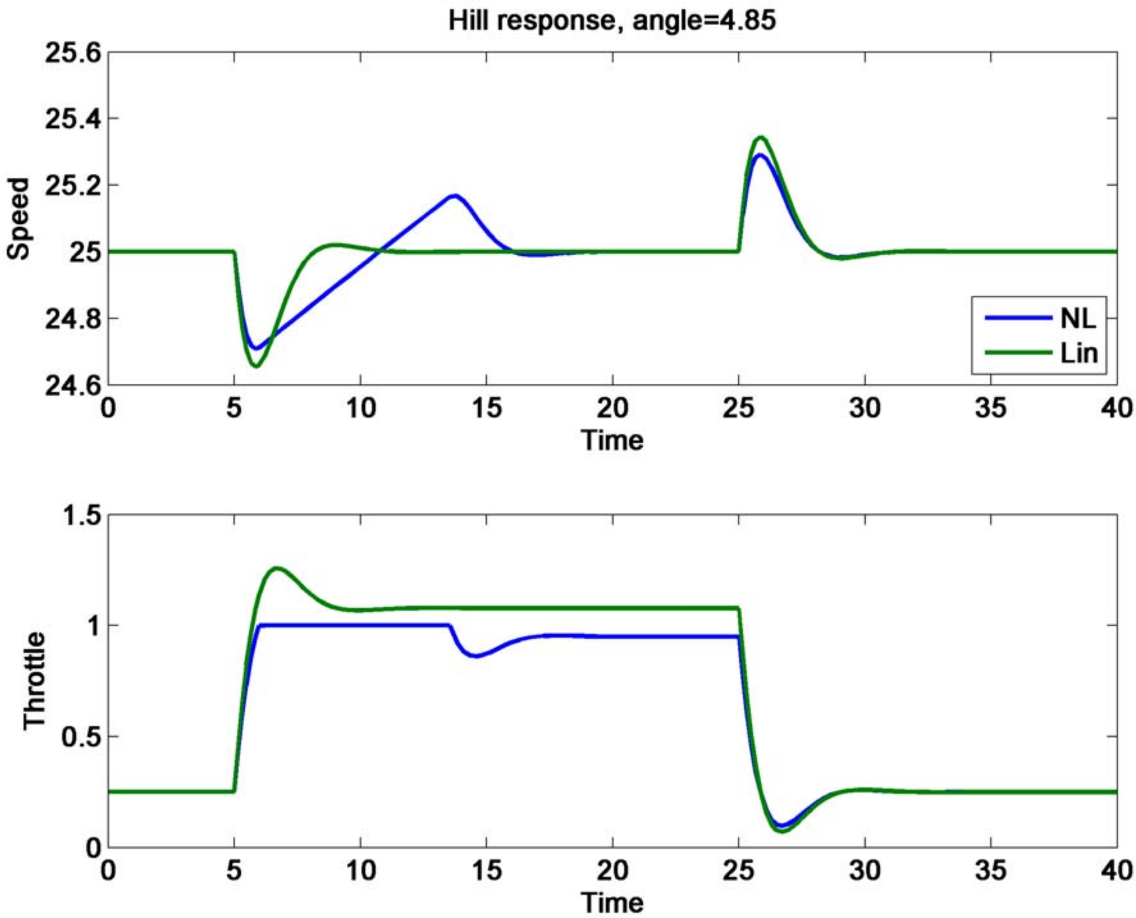
Review

- (Jacobian) linearization:
  - Find the desired equilibrium condition (state and control).
  - Linearize the non-linear model around the equilibrium.

- Control design:
  - Design a linear compensator for the linear model.
  - If the linear system is closed-loop stable, so will be the nonlinear system—in a neighborhood of the equilibrium.
  - Check in a (nonlinear) simulation the robustness of your design with respect to "typical" deviations.

Effects of the saturation

- What if the slope is a little steeper (say 4.85 degrees)?



- What is wrong?

- Systems experiencing **Integrator wind-up**
  - Once the input saturates, the integral of the error keeps increasing.
  - When the error decreases, the large integral value prevents the controller from resuming "normal operations" quickly (the integral error must decrease first!) – so the response is delayed

- **Idea**: once the input saturates, stop integrating the error (can't do much about it anyway!)

Anti-windup Logic

- One option is the following logic for the integral gain:

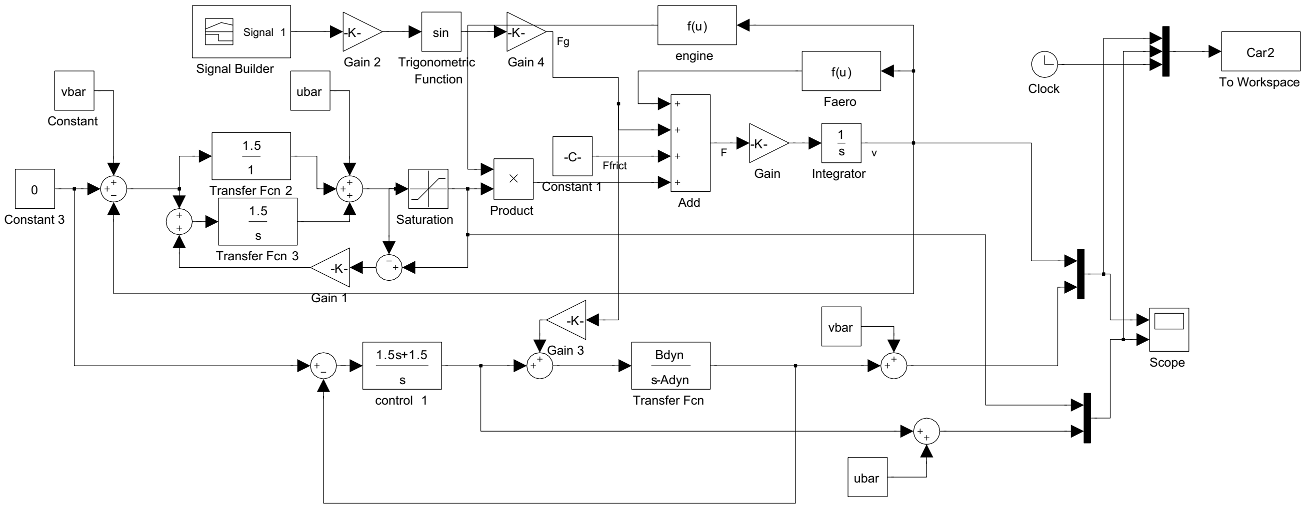
$$K_I' = \begin{cases} K_I & \text{if the input does not saturate;} \\ 0 & \text{if the input saturates} \end{cases}$$

- Another option is the following:
  - Compare the actual input and the commanded input.
  - If they are the same, the saturation is not in effect.
  - Otherwise, reduce the integral error by a constant times the difference.

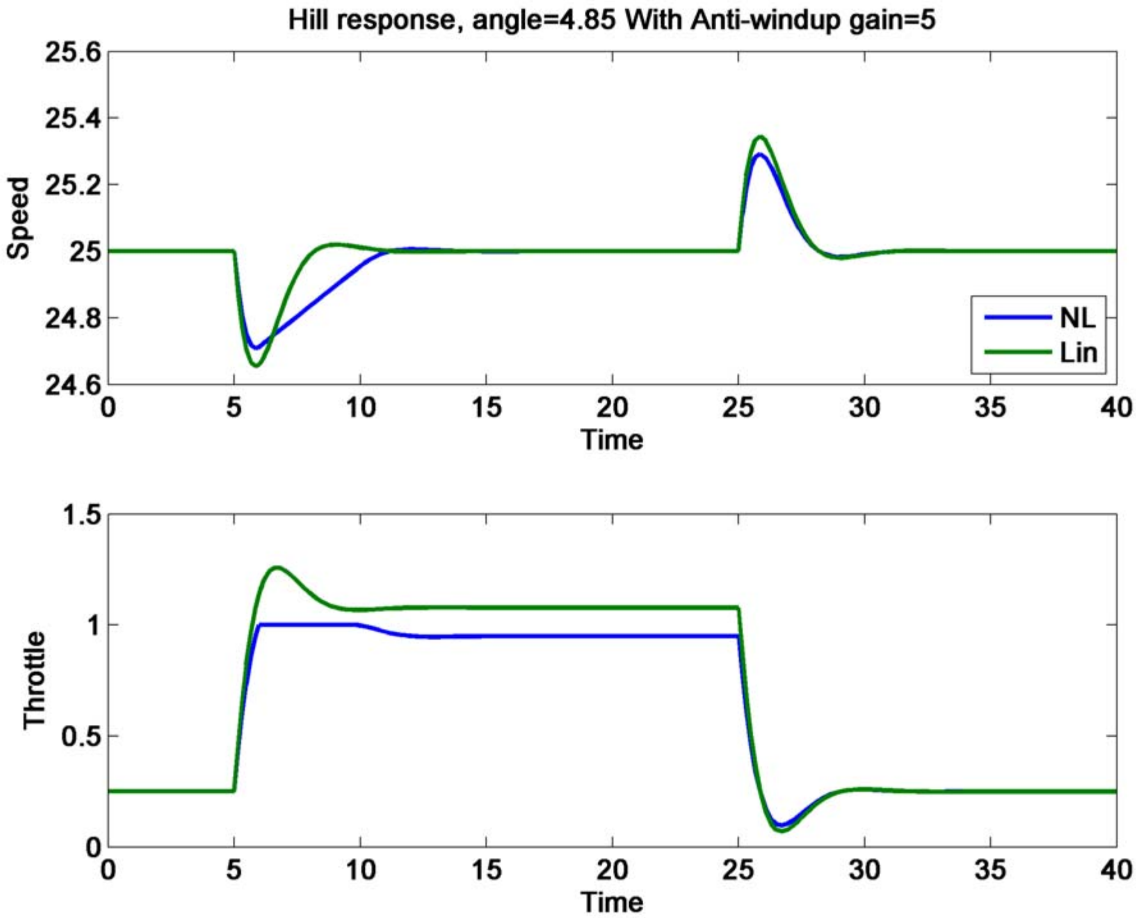
$$u_{int} = \int K_i \left(e + K_{aw}(\mathtt{sat}(u) - u)\right) dt$$

- With this choice, under saturation, the actuator/commanded difference is fedback to the integrator to so that $e_{act} = \mathtt{sat}(u) - u$ tends to zero
  - Implies that the controller output is kept close to saturation limit.
  - The controller output will then change as soon as the error changes sign, thus avoiding the integral windup.
  - If there is no saturation, the anti-windup scheme has no effect.

Anti-windup Scheme



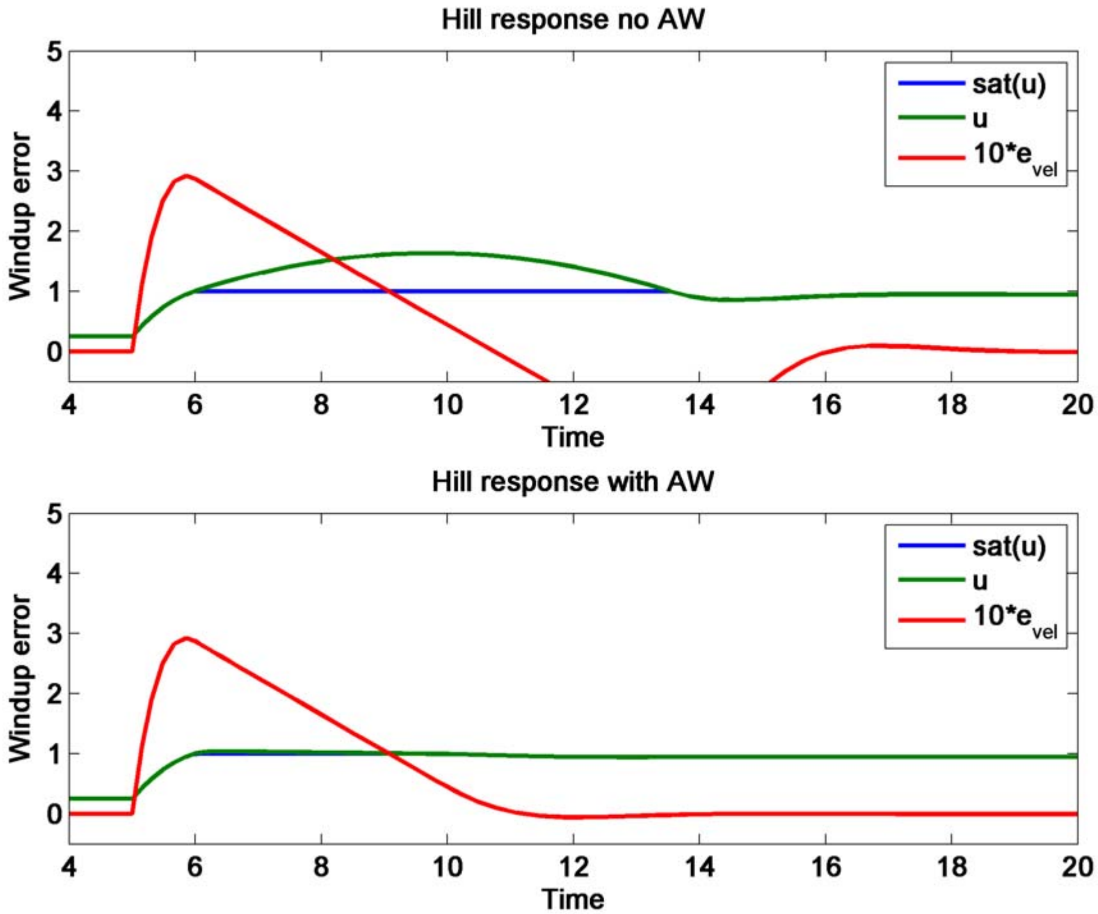
- Response to a 4.85 degree slope



- Anti-windup compensator avoids the velocity overshoot by preventing the error buildup in the integral term

Anti-windup Summary

- Anti-wind up schemes guarantee the **stability of the compensator** when the (original) feedback loop is effectively opened by the saturation.
- Prevent divergence of the integral error when the control cannot keep up with the reference.
- Maintain the integral errors "small."



Code: Car Setup

```
set(0, 'DefaultAxesFontSize', 12, 'DefaultAxesFontWeight','demi')
set(0, 'DefaultTextFontSize', 12, 'DefaultTextFontWeight','demi')
set(0,'DefaultAxesFontName','arial')
set(0,'DefaultAxesFontSize',12)
set(0,'DefaultTextFontName','arial')

clear all
%close all
% Parameters for defining the system dynamics
theta=0;
gear=5;
alpha = [40, 25, 16, 12, 10];       % gear ratios
Tm = 190;                 % engine torque constant, Nm
wm = 420;                 % peak torque rate, rad/sec
beta = 0.4;               % torque coefficient
Cr = 0.01;                % coefficient of rolling friction
rho = 1.2;                % density of air, kg/m^3
A = 2.4;                  % car area, m^2
Cd = 0.79/A;                  % drag coefficient
g = 9.8;                  % gravitational constant
m=1500;                   % mass
v=25;

% Compute the torque produced by the engine, Tm
omega = alpha(gear) * v;        % engine speed
torque = Tm * ( 1 - beta * (omega/wm - 1)^2 );
F = alpha(gear) * torque;

% Compute the external forces on the vehicle
Fr = m * g * Cr;              % Rolling friction
Fa = 0.5 * rho * Cd * A * v^2;        % Aerodynamic drag
Fg = m * g * sin(theta);          % Road slope force
Fd = Fr + Fa + Fg;            % total deceleration

ubar=(Fa+Fr)/(F)
vbar=v;

dTdv=Tm*(-2*beta*(alpha(gear)*vbar/wm-1)*(alpha(gear)/wm))
Adyn=(alpha(gear)*dTdv*ubar-rho*Cd*A*vbar)/m;
Bdyn=F/m;

Hillangle=4; % non-saturating angle
sim('cruise_control')
figure(4);
subplot(211);
plot(Car1(:,5),Car1(:,[2]),'LineWidth',2);xlabel('Time');ylabel('Speed')
title(['Hill response, angle=',num2str(Hillangle)])
subplot(212);
plot(Car1(:,5),Car1(:,[4]),'LineWidth',2);xlabel('Time');ylabel('Throttle')

figure(1);
subplot(211);
plot(Car1(:,5),Car1(:,[1 2]),'LineWidth',2);xlabel('Time');ylabel('Speed')
legend('NL','Lin','Location','SouthEast');
title(['Hill response, angle=',num2str(Hillangle)])
subplot(212);
plot(Car1(:,5),Car1(:,[3 4]),'LineWidth',2);xlabel('Time');ylabel('Throttle')

Antiwindup_gain=0; % us esame code with and without AWup
Hillangle=4.85;
sim('cruise_control_awup')
figure(2);
subplot(211);
plot(Car2(:,5),Car2(:,[1 2]),'LineWidth',2);xlabel('Time');ylabel('Speed')
legend('NL','Lin','Location','SouthEast');
title(['Hill response, angle=',num2str(Hillangle)])
subplot(212);
plot(Car2(:,5),Car2(:,[3 4]),'LineWidth',2);xlabel('Time');ylabel('Throttle')

Car2_no_AW=Car2;

Antiwindup_gain=5;
sim('cruise_control_awup')
figure(3);
```

```
subplot(211);
plot(Car2(:,5),Car2(:,[1 2]),'LineWidth',2);xlabel('Time');ylabel('Speed')
legend('NL','Lin','Location','SouthEast');
title(['Hill response, angle=',num2str(Hillangle),' With Anti-windup gain=',num2str(Antiwindup_gain)])
subplot(212);
plot(Car2(:,5),Car2(:,[3 4]),'LineWidth',2);xlabel('Time');ylabel('Throttle')

figure(5);
subplot(211);
plot(Car2_no_AW(:,5),Car2_no_AW(:,[6 7]),Car2_no_AW(:,5),10*Car2_no_AW(:,[8]),'LineWidth',2);xlabel('Time');yla
legend('sat(u)','u','10*e_{vel}','Location','NorthEast');
axis([4 20 -0.5 5])
title(['Hill response no AW'])
subplot(212);
plot(Car2(:,5),Car2(:,[6 7]),Car2(:,5),10*Car2(:,[8]),'LineWidth',2);xlabel('Time');ylabel('Windup error')
legend('sat(u)','u','10*e_{vel}','Location','NorthEast');
title(['Hill response with AW'])
axis([4 20 -0.5 5])

print -dpng -r300 -f1 AW1.png
print -dpng -r300 -f2 AW2.png
print -dpng -r300 -f3 AW3.png
print -dpng -r300 -f4 AW4.png
print -dpng -r300 -f5 AW5.png
```

16.30 / 16.31 Feedback Control Systems
Fall 2010