# Technical aspect for the paper: A new model for control of systems with friction[1]

Auralius Manurung - ME - Universitas Pertamina

Given a LuGre friction model:

$$\dot{z} = \frac{dz}{dt} = v - \frac{|v|}{g(v)} z \tag{1}$$

and:

$$g(v) = \frac{F_C + (F_S - F_C) e^{-(v/v_s)^2}}{\sigma_0} \tag{2}$$

Substituting Eq. (2) to Eq. (1) gives us:

$$\dot{z} = v - \frac{z|v|\sigma_0}{F_C + (F_S - F_C) e^{-(v/v_s)^2}} \tag{3}$$

As for a system of mass:

$$\ddot{x} = \frac{u - F}{M} \tag{4}$$

where

$$F = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v \tag{5}$$

and

$$v = \dot{x} \tag{6}$$

Substituting Eq. (5) and Eq. (6) to Eq. (4) gives us:

$$\ddot{x} = \frac{u - \sigma_0 z - \sigma_1 \dot{z} - \sigma_2 v}{M} \tag{7}$$

where:

$$\dot{z} = \dot{x} - \frac{z|\dot{x}|\sigma_0}{F_C + (F_S - F_C) e^{-(\dot{x}/v_s)^2}} \tag{8}$$

Now, let us take:

$$\begin{aligned} q_1 = x &\rightarrow \dot{q}_1 = \dot{x} \\ q_2 = \dot{x} &\rightarrow \dot{q}_2 = \ddot{x} \\ q_3 = z &\rightarrow \dot{q}_3 = \dot{z} \end{aligned} \tag{9}$$

Applying Eq. (9) to Eq. (7) and Eq. (8) then gives us:

$$\begin{aligned} \dot{q}_1 &= q_2 \\ \dot{q}_2 &= \frac{(u - \sigma_0 q_3 - \sigma_1 \dot{z} - \sigma_2 q_2)}{M} \\ \dot{q}_3 &= \dot{z} \end{aligned} \tag{10}$$

where:

$$\dot{z} = q_2 - \frac{z|q_2|\sigma_0}{F_C + (F_S - F_C) e^{-(q_2/v_s)^2}} \tag{11}$$

We need to be aware that in the right-hand-side of Eq. (10), we do not replace $\dot{z}$ with $\dot{q}_3$ because we will compute $\dot{z}$ independently by using Eq. (11). Eq. (11) merely acts as an additional equation for Eq. (10). Eq. (10) now follows the formal form: $\dot{q} = f(q)$. Thus, we can implement it into MATLAB built-in ODE solver.

Next, we will address $u(t)$. In the selected paper, only two kinds of force input are found: a ramp force input as in the varying break-away force simulation and a piecewise linear force input as in the presliding simulation.

[1] C. Canudas de Wit, H. Olsson, K. J. Astrom, and P. Lischinsky, “A new model for control of systems with friction,” IEEE Trans. Automat. Contr., vol. 40, no. 3, pp. 419–425, Mar. 1995.

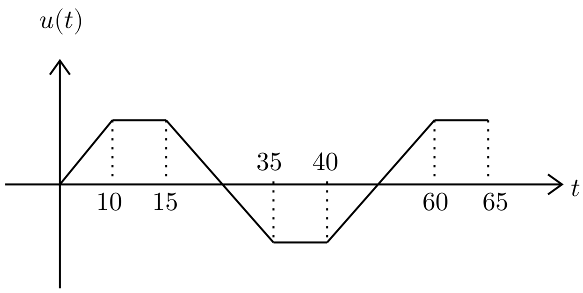
Ramp Force Input

```
function xdot = sim_mass_with_ramp_force_input(t, q, M, Fs, Fc, sigma_0, sigma_1, sigma_2, vs, force_rate)
    u = force_rate*t;     % ramped-up force input

    zdot = q(2) - ( (q(3)*abs(q(2))*sigma_0) / (Fc+(Fs-Fc)*exp(-(q(2)/vs)^2)) );
    F = sigma_0*q(3) + sigma_1 * zdot + sigma_2*q(2);

    qdot_1 = q(2);
    qdot_2 = (u - F) / M;
    qdot_3 = zdot;
    xdot = [qdot_1 ; qdot_2; qdot_3 ];
end
```

Piecewise Linear Force Input




```
function xdot = sim_presliding(t, q, M, Fs, Fc, sigma_0, sigma_1, sigma_2, vs)
    rate = 0.1425;

    if t <= 10
        u = rate*t;
    elseif t > 10 && t <= 15
        u = 1.425;
    elseif t > 15 && t <= 35
        u = 1.425 - rate*(t-15);
    elseif t > 35 && t <= 40
        u = -1.425;
    elseif t > 40 && t <= 60
        u = -1.425 + rate*(t-40);
    elseif t > 60
        u = 1.425;
    end

    zdot = q(2) - ( (q(3)*abs(q(2))*sigma_0) / (Fc+(Fs-Fc)*exp(-(q(2)/vs)^2)) );
    F = sigma_0*q(3) + sigma_1 * zdot + sigma_2*q(2);

    qdot_1 = q(2);
    qdot_2 = (u - F) / M;
    qdot_3 = zdot;
    xdot = [qdot_1 ; qdot_2; qdot_3 ];
end
```

A System with PID

For a system with a PID control, we need to create a new system which is also in a form of $\dot{x} = f(x)$. Here, we need to introduce a new variable to accommodate the integral term.

Given a system:

$$\ddot{x} = \frac{u - F}{M} \tag{12}$$

where:

$$u = -K_v v - K_p e - K_i \int e \ dt \tag{13}$$

and

$$e = x - x_d \tag{14}$$

By substituting Eq. (13) to Eq. (12), we get the following equation:

$$\ddot{x} = \frac{\overbrace{-K_v v - K_p e - K_i \int e \ dt}^{u} - \overbrace{(\sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v)}^{F}}{M} \tag{15}$$

We then introduce a new variables $q_4$ as:

$$q_4 = \int e \rightarrow \dot{q}_4 = e \tag{16}$$

Substituting Eq. (16) to Eq. (15) and replacing $v$ with $q_2$ and $\ddot{x}$ with $\dot{q}_2$ give us:

$$\dot{q}_2 = \frac{\overbrace{-K_v q_2 - K_p e - K_i \int e \ dt}^{u} - \overbrace{(\sigma_0 z + \sigma_1 \dot{z} + \sigma_2 q_2)}^{F}}{M} \tag{17}$$

Hence, we now have a new system as follows:

$$\begin{gathered} \dot{q}_1 = q_2 \\ \dot{q}_2 = \frac{\left(-K_v q_2 - K_p q_4 - K_i \int e \ dt - \sigma_0 q_3 - \sigma_1 \dot{z} - \sigma_2 q_2\right)}{M} \\ \dot{q}_3 = \dot{z} \\ \dot{q}_4 = e \end{gathered} \tag{18}$$

The MATLAB implementation is as follows:

```
function xdot = sim_pid(t, q, M, Fs, Fc, sigma_0, sigma_1, sigma_2, vs, xd)
    Kp = 3;
    Ki = 4;
    Kv = 6;

    e = q(1) - xd;

    u = -Kv*q(2)-Kp*e-Ki*q(4);

    zdot = q(2) - ( (q(3)*abs(q(2))*sigma_0) / (Fc+(Fs-Fc)*exp(-(q(2)/vs)^2)) );
    F = sigma_0*q(3) + sigma_1 * zdot + sigma_2*q(2);

    qdot_1 = q(2);
    qdot_2 = (u - F) / M;
    qdot_3 = zdot;
    qdot_4 = e;
    xdot = [qdot_1 ; qdot_2; qdot_3; qdot_4];
end
```