How the torque loop works.

September 1, 2016

The robot's motors are controlled by using PWM signals applied to the motors. Assuming that each motors affects the position of only one link via rigid transmission mechanisms, the relationship between the link's torque τ and the motor's duty cycle PWM is assumed to be:

$$PWM = PWM_{\text{control}} + k_v\dot{q} + k_c\text{sign}(\dot{q}), \tag{1}$$

with k_{ff}, k_v, k_c three constants, and $\dot{q}$ the joint's velocity. Since discontinuities may be challenging in practice, it is best to smooth the sign function. When,

$$\|\dot{q}\| \geq \text{coulombVelThr} \tag{2}$$

the coulomb friction is calculated as,

$$\text{sign}(\dot{q}). \tag{3}$$

However, for $\|\dot{q}\| < \text{coulombVelThr}$, the coulomb friction is calculated as,

$$\left(\frac{\dot{q}}{\text{coulombVelThr}}\right)^3 \tag{4}$$

For the sake of simplicity we will simply refer to this as the $\text{sign}(\dot{q})$ operation.

This model can be improved by considering possible parameters' asymmetries with respect to the joint velocity $\dot{q}$. In particular, the coulomb friction parameter k_c may depend on the sign of $\dot{q}$, and have different values depending on this sign. Then, an improved model is:

$$PWM = PWM_{\text{control}} + k_v\dot{q} + [k_{cp}s(\dot{q}) + k_{cn}s(-\dot{q})]\text{sign}(\dot{q}), \tag{5}$$

where the function $s(x)$ is the step function, i.e.

$$s(x) = 1 \quad \text{if} \quad x >= 0 \tag{6}$$
$$s(x) = 0 \quad \text{if} \quad x < 0 \tag{7}$$

As stated, the above equation constitutes the relation between the tension applied to the motor and the link torque. Then, to generate a desired torque τ_d coming from an higher control loop, it suffices to evaluate the above equation

with $\tau = \tau_d$. In practice, however, it is a best practice to add a lower loop to generate τ so that τ will converge to τ_d, i.e:

$$PWM_{control} = k_{ff}\tau_d - k_p e_\tau - k_d \frac{de_\tau}{dt} - k_i \int e_\tau \mathrm{dt}, \qquad (8)$$

where $e_\tau := \tau - \tau_d$.

The integral term is saturated using the variable $\pm$ max_int. The PWM values are determined using the motor coupling matrices. The final PWM output is saturated using $\pm$max_pwm.

This document was adapted from the documentation left over in the source code.