MS PI controller explanation

Cassandre Contré, Simon Van Mulders

October 2024

Quick clarification of the PI controller of the gradient-based (MS) model implemented in RAPTOR, with a little help from [Haugen, 2010].

We start from eq. 2.7 of Teplukhina [2018], generalised to $\mu \in \{\mu_{ne}, \mu_{Te}\}$:

$$\mu = \mu^{\text{ff}}(t_k) + K_p \cdot e(t) + K_i \cdot \int^{\delta t} e(t)dt \tag{1}$$

We discretise eq. 1 using Euler's Backward differentiation method, namely:

$$\dot{x}(t_k) = \frac{x(t_k) - x(t_{k-1})}{\Delta t} \tag{2}$$

The key thing here (which is also the confusing one) is that we actually want to compute the incremental control value:

$$\Delta\mu(t_k) = \mu(t_k) - \mu(t_{k-1}) = \Delta t \cdot \dot{\mu}(t_k), \tag{3}$$

such that one must take the time derivative of eq. 1:

$$\begin{aligned} \dot{\mu}(t_k) &= \dot{\mu}^{\text{ff}}(t_k) + K_p \dot{e}(t_k) + K_i e(t_k) \\ &= \frac{\mu^{\text{ff}}(t_k) - \mu^{\text{ff}}(t_{k-1})}{\Delta t} + K_p \cdot \frac{e(t_k) - e(t_{k-1})}{\Delta t} + K_i \cdot e(t_k). \end{aligned} \tag{4}$$

Hence, by multiplying $\dot{\mu}(t_k)$ with the time step Δt we get the μ increment:

$$\Delta\mu = \Delta\mu^{\text{ff}}(t_k) + \Delta\mu^{\text{fb}}(t_k) \tag{5}$$

such that, for each time iteration:

$$\mu(t_k) = \mu(t_{k-1}) + \Delta\mu^{\text{ff}}(t_k) + \Delta\mu^{\text{fb}}(t_k) \tag{6}$$

with

$$\Delta\mu^{\text{ff}}(t_k) = \mu^{\text{ff}}(t_k) - \mu^{\text{ff}}(t_{k-1}) \tag{7}$$

and:

$$\Delta\mu^{\text{fb}}(t_k) = K_p \cdot [e(t_k) - e(t_{k-1})] + K_i \cdot \Delta t \cdot e(t_k) \tag{8}$$

In practice in the code, the errors are computed from the previous state estimate $x(t_{k-1})$ and the μ value computed at time iteration k is $\mu(t_{k-1})$, using the error $e(t_{k-1})$. In chi_MS.m and vpdn_MS.m, the errors are expressed respectively:

$$\begin{aligned} e^{T_e}(t_{k-1}) &= H_e^{\text{ref}}(t_{k-1}) - H_e^{\text{eff}}(t_{k-1}) \\ e^{n_e}(t_{k-1}) &= n_{e,l}^{\text{ref}}(t_{k-1}) - n_{e,l}^{\text{eff}}(t_{k-1}) \end{aligned} \tag{9}$$

Final note: in the code, the g_i gain is equivalent to $K_i \Delta t$.

References

F. Haugen. Discretization of simulator, filter, and pid controller. 2010. URL `https://www.mic-journal.no/PDF/ref/Haugen2010.pdf`.

A. A. Teplukhina. PhD thesis, EPFL, Lausanne, 2018. n° **8478**.