Abstract about Signal filtering: Why and how

Introduction

Today, I want to share a paper. Signal filtering: Why and how *Prevent over-filtering by simultaneously optimizing loop tuning and filter parameters*, written by Mark Darby, Greg McMillan July 24, 2019. The main reason to filter a signal is to reduce and smooth out high-frequency noise associated with a measurement such as flow, pressure, level or temperature. A common example is the noise associated with the differential pressure (DP) across an orifice plate used to infer flow rate. High-frequency noise is normally considered to be random and additive to a measured signal, and is usually uncorrelated in time; i.e., the value of the noise at any time τ does not depend on previous values of the noise. Ideally, we want to estimate the underlying signal without noise, introducing as little distortion as possible.

When a noisy signal is used in control, filtering is important for effective derivative action and for avoiding excessive movement in the controller output that causes valve wear or disturbs other control loops. A complicating factor for the control case is that both filtering and controller tuning can be used to reduce movement of the controller output.

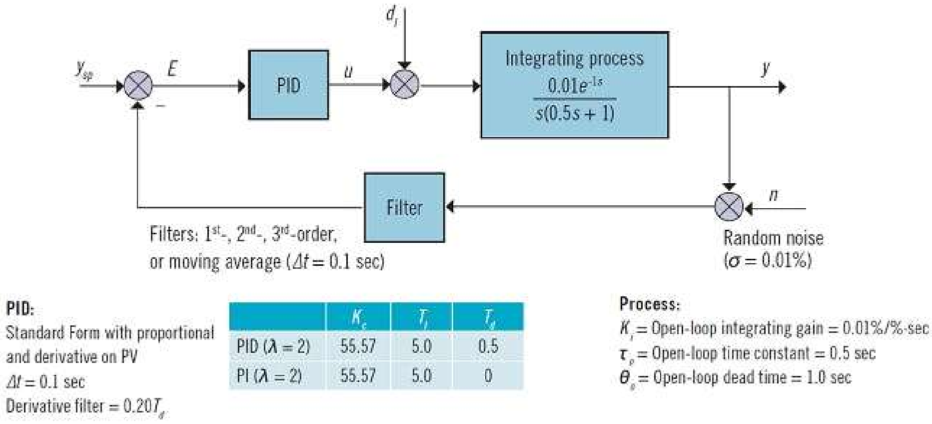
What is the problem of lag?

The downside of filtering is the lag introduced, especially with heavier filtering, which can have a detrimental effect on timely detection of changes in the underlying signal. When used for feedback, the filtered value can result in control that is sluggish or, in the worst case, becomes oscillatory or even unstable. In the authors' experience, the more typical problem encountered is excessive filtering of a signal, as opposed to under-filtering.

Reason

I find that Heavier filtering can introduce lag, which can be detrimental to the timely detection of changes in the underlying signal. This can result in sluggish or oscillatory control, or even instability.

Besides, the importance of effective derivative action in control systems, particularly for integrating and near-integrating systems with additional lags and dead time. Filtering the feedback variable (PV) is crucial for ensuring that the derivative action is not affected by noisy signals, which can cause erratic movement of the controller output. Many industrial PID controllers include a filter on just the derivative term, expressed as a fraction of the specified derivative time, to address this issue.



Simulated closed-loop system Figure 1:

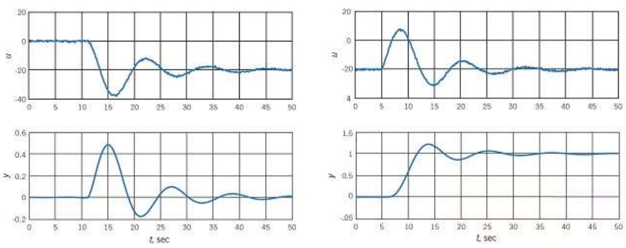
Finally, it's important to note that filtering should not be used as a band-aid solution for underlying problems such as unmeasured disturbances or issues with equipment or installation. In many cases, excessive filtering can actually make the problems worse. It's important to regularly review and tune control loops to ensure optimal performance, rather than relying solely on filtering to address issues as they arise.

Table 1: McMillan Lambda tuning formulas

Algorithm	K_c	T_i	T_d
PID	$\frac{2\lambda+\theta_o}{K_i(\lambda+\theta_o)^2}$	$2\lambda+\theta_o$	$\min\left(4T_i, \max(\frac{1}{2}\theta_o, \tau_o)\right)$
PI	$\frac{2\lambda+\theta_o}{K_i(\lambda+\theta_o)^2}$	$\max\left(4\theta_o, 2\lambda+\theta_o\right)$	

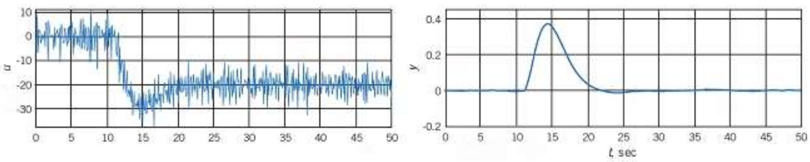
Simulation leads to optimization

To evaluate filters in a closed loop, we simulate an integrating system with a noisy disturbance signal and an aggressively tuned PID controller. The loop and simulation details are shown in **Figure 1.** Lambda tuning rules modified by McMillan to provide better load disturbance rejection for a Standard Form PID (shows in Table 1) were used for an integrating process. A value of λ = 2 sec (twice the loop dead time) was used as the arrest time (time to stop excursion) for the tuning, as it



produced a non-oscillatory response in both the PV and CO, provided the system is linear and the dynamics are fixed and well known. The following loop criteria are evaluated for each filter:

Peak error: Maximum error following an unmeasured step in load. Peak error is important to prevent relief, alarm and SIS activation, and environmental violation.



IAE: Integrated absolute error over time is a common criterion for measuring loop performance. It directly relates to economics as it provides a measure of the amount of process material that is off-setpoint.

Setpoint overshoot: The maximum error overshoot for a setpoint change. Excessive overshoot can have economic or safety implications. **IAD:** Integrated absolute difference of the controller output. For the case that the PID directly manipulates a valve, this is total valve travel distance and directly relates to valve wear.

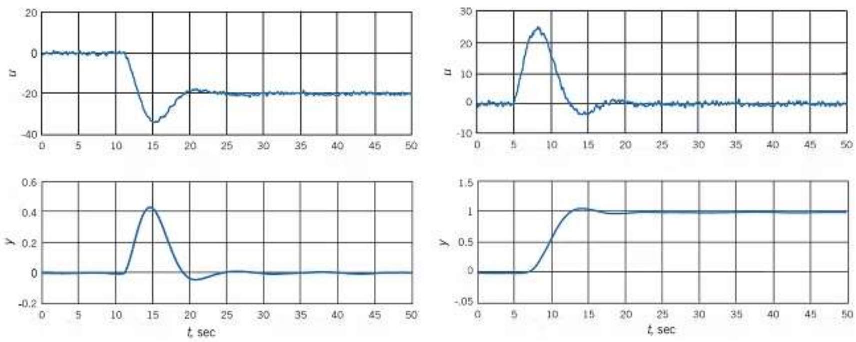
	Peak error (%PV)	IAE (%PV-sec)	IAD (%CO-sec)
PID	0.373	1.974	280.7
PI	0.444	2.436	32.538

What have we learned, what are the lessons?

Decreased attenuation of noise and disturbances comes at a cost: The more attenuation, the slower the filtered signal approaches a new average value—i.e., the more time lag. For the same amount of noise attenuation, higher-order filters (2nd- and 3rd-order) and the moving average filter approach a new average value quicker than the often-used exponential or 1st-order filter. An advantage of the moving average filter is that it can eliminate a periodic cycle if the filter Δτ and N are selected appropriately.

	Load response			Setpoint response		
PID with filter	Peak error (%PV)	IAE (%PV-sec)	IAD (%CO-sec)	Peak error (%PV)	IAE (%PV-sec)	IAD (%CO-sec)
1st order, $\tau_f = 0.8161$	0.488	3.597	23.886	0.222	6.372	24.174
2nd order, $\tau_f = 0.2177$	0.434	2.167	23.886	0.064	5.045	23.676
3rd order, $\tau_f = 0.1388$	0.443	2.235	23.886	0.073	5.066	23.748
Moving average, N=11	0.464	2.555	23.886	0.114	5.285	23.892

Table 4: PID results with filtering, 91.5% reduction in load IAD



Before returning a loop, make sure to note what the filter parameters are: time constant or filter factor and Δτ. Only use filtering to temper movement in the manipulated variable (controller output) caused by the noise. If aggressive control is required (higher gain and derivative action), higher-order and moving average filters allow tighter control with improved metrics (peak error and IAE) compared to the exponential or 1st-order filter. One should be careful to not cause loop instability with filtering. Although not extensive, these simulation results, plus others we have performed, generally show that of the filters considered, the 2nd-order filter provides the best metrics when targeting reduced levels of IAD.

We took the simple approach of keeping the tuning the same to show the improvement possible with more sophisticated filters. Ideally, the tuning and the selection of filter parameters should be jointly optimized, e.g., using a loop tuning optimizer.