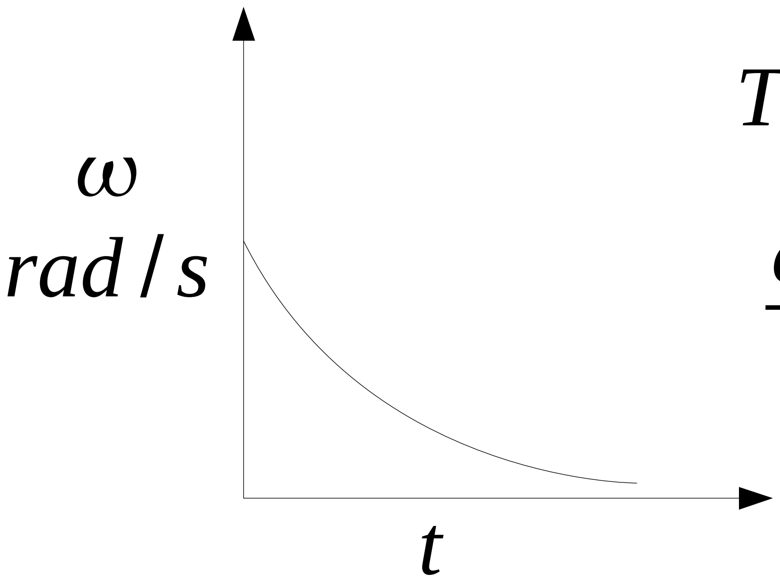
FILTERS

Take an fan which has been just turned off....

Sensor:
Tachometer measuring the angular velocity

What we know about it:



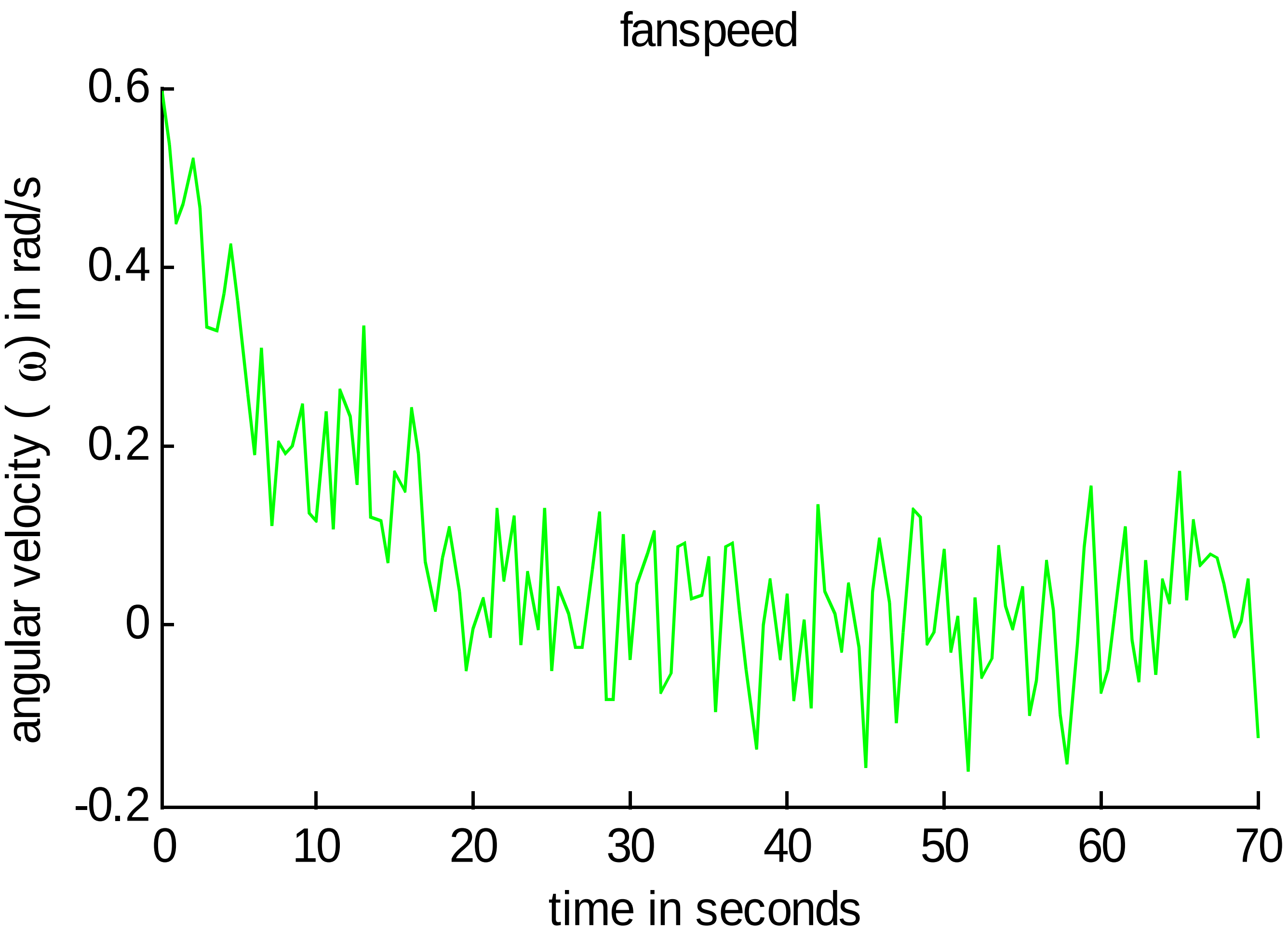
$$T = I\frac{d\omega}{dt} = -k\omega \quad (+random T)$$

$$\frac{d\omega}{dt} = -c\omega \quad (+random \;\; accl.)$$

Due to viscous friction the velocity slowly decrease to 0.

There might be some random torques (say due to air currents)

Here is the data mesured by Tachometer....



Pretty noisy!

Solution:
Filter of the noise

Finite Impulse reponse filters (window filter)

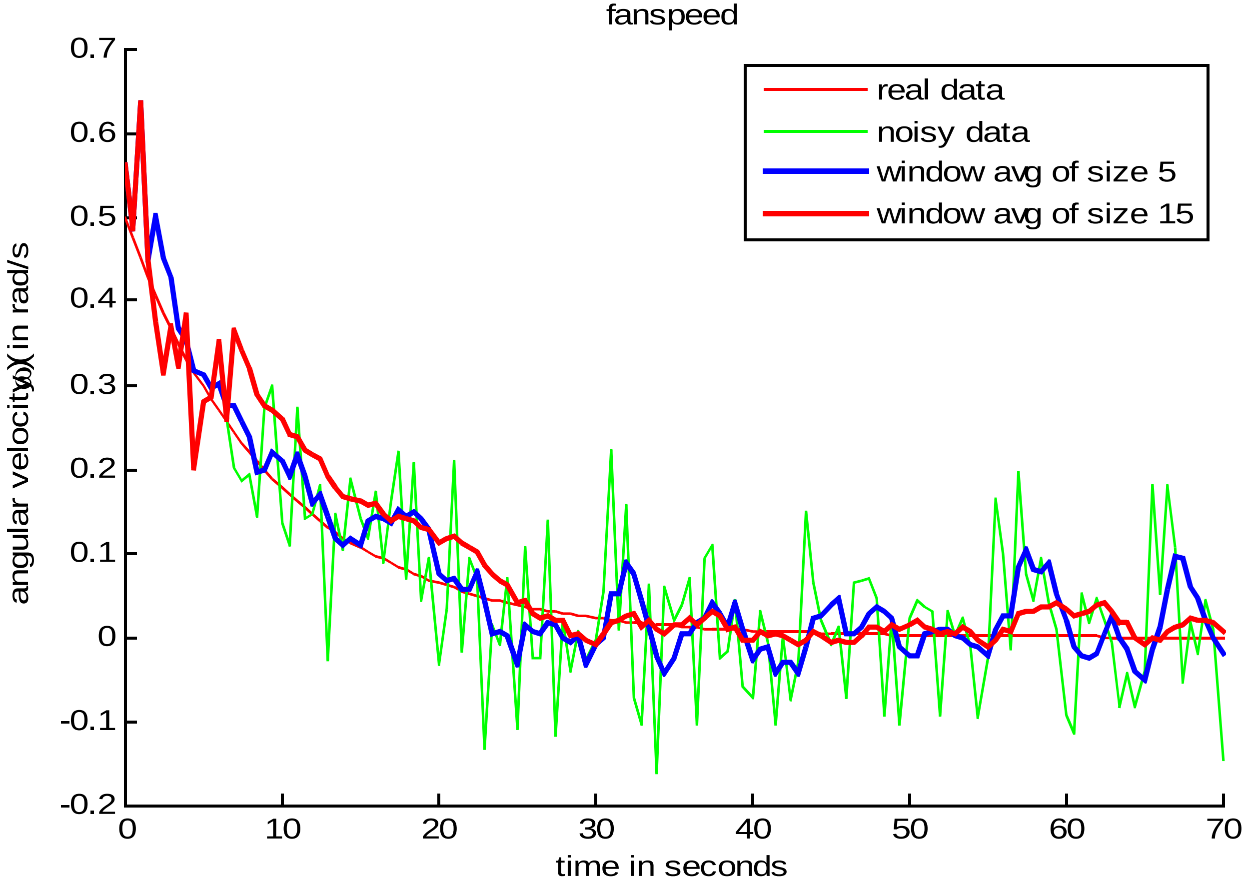
- Idea:

  current data = linear function of a few previous data points (including the new one).

$$y_n^{filt} = a_n y_n + a_{n-1} y_{n-1} + a_{n-2} y_{n-2}$$
$$\text{where:} \quad a_n + a_{n-1} + a_{n-2} = 1$$

- If all the *'a'*s are 1/3. We call it a window average.

Lets try window averages



Problems:

- Delay
- Initialization

INFinite Impulse reponse filters (exponential filter)

- Idea:

current data = linear function of a few previous data points (including new one)

+

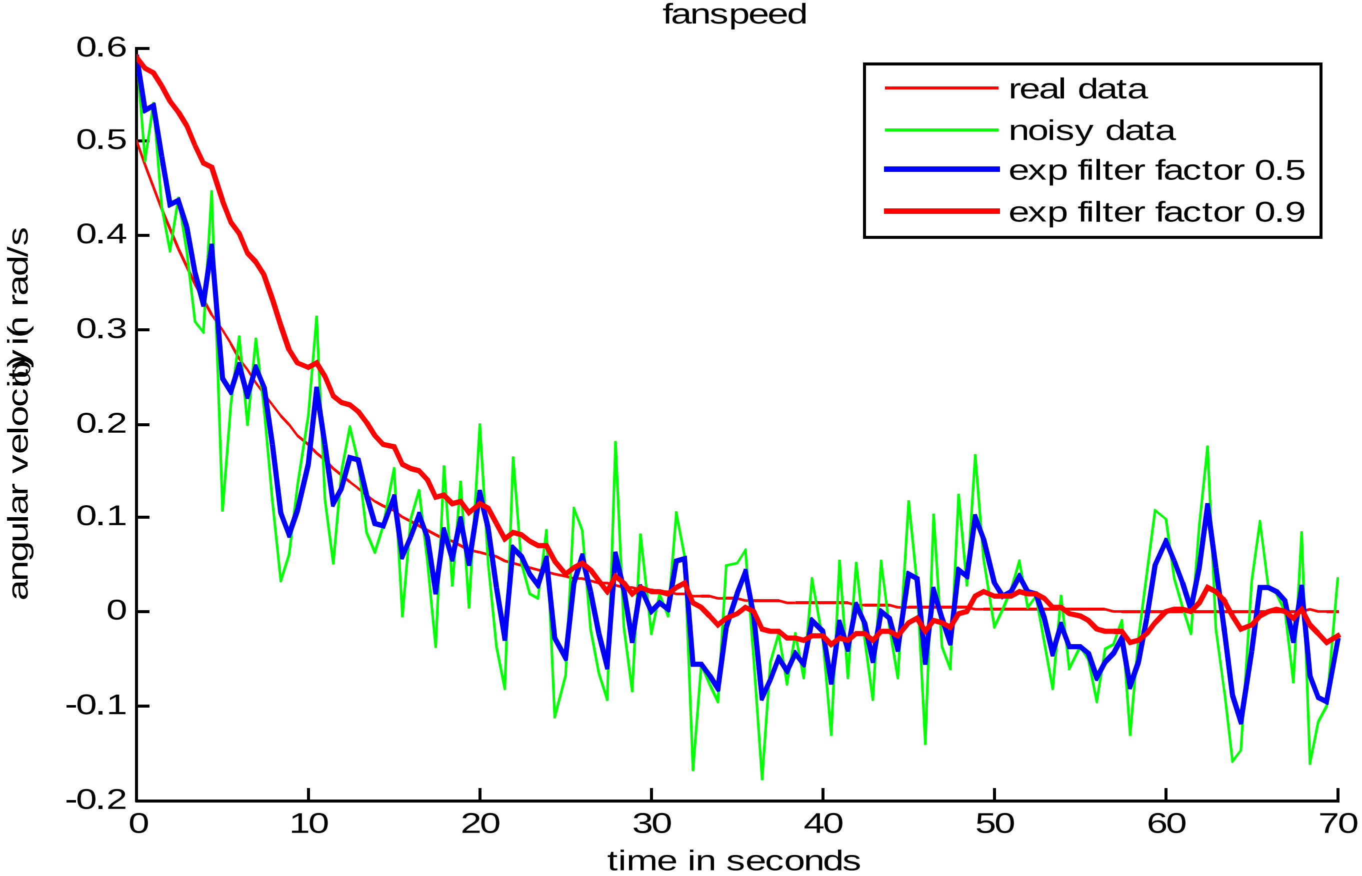
few previous filtered data points

- Example: exponential filter

$$y_n^{filt} = \alpha \, y_{n-1}^{filt} + (1-\alpha) \, y_n$$

where: α is the filter factor

Exponential filter



Issues:

- Delay
- Not much problem in Initialization
- Easier to code higher order filters

But these are not very good!

Even our eye can do much better

We are using our intuition about how the line should be, to filter in our mind

We are using

A MODEL

of

the *process* and the *noise*

Using more information

- If we had 2 **<u>independent</u>** sensors (y1 and y2) & we **<u>know their errors variance</u>** (V1 and V2) then we can do better:
- Variance of combination:

$$y_{total} = \beta\, y_1 + (1-\beta)\, y_2$$
$$V_{total} = \beta^2 V_1 + (1-\beta)^2 V_2$$

- Minimum variance:

$$V_{total}^{min\,var} = \frac{V_2 V_1}{V_1 + V_2} \qquad \text{at: } \beta = \frac{V_2}{V_1 + V_2}$$

Kalman filter

- 1st sensor = tachometer, 2nd sensor = model!
- **<u>Step1:</u> Predict**

$$\omega^{-}_{n+1}=(1-c\,\delta t)\,\omega_n$$

$$V^{-}_{n+1}=(1-c\,\delta t)^2 V_n+Q$$

Process noise (variance of random forces)

- **<u>Step2:</u> Update**

$$\omega_{n+1}=\beta\,\omega^{-}_{n+1}+(1-\beta)\,y_{n+1}$$

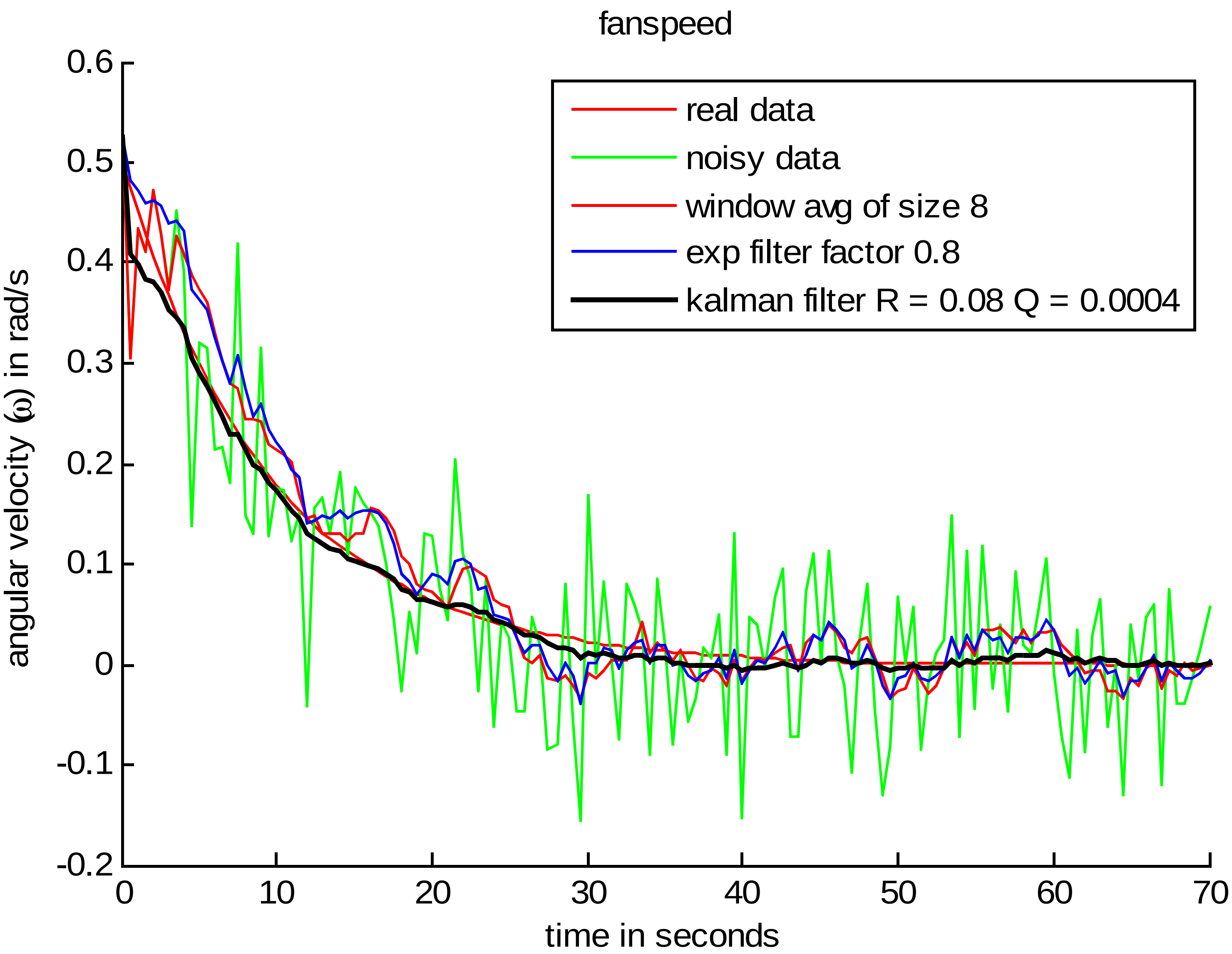
Sensor noise variance.

$$\beta=\frac{R}{(V^{-}_{n+1}+R)}\ \text{and}\ V_{n+1}=\frac{V^{-}_{n+1}R}{(V^{-}_{n+1}+R)}$$

- People prefer to call (1-beta) as Kalman gain (K) and they write it as:

$$\omega_{n+1}=\omega^{-}_{n+1}+K\,(y_{n+1}-\omega^{-}_{n+1})$$

Kalman filter



Issues

- Computationaly more expensive
- Is linear only,
- works best for uncorrelated noise.
- Sensitive to unmodelled dynamics and process noise.