WORKSHEET

U4.3 The low pass filter

Noise and filtering · University · about 30 min

Name _____ Class _____ Date _____

What this lesson is about

A moving average, then the exponential filter that does the same job in one line and no memory.

- The moving average
- The exponential filter
- It is the same idea as the Kalman filter
- Starting it

Questions

7 marks in all

1. An exponential filter starts at 50 and sees three readings of 60. What does it print? [1 mark]

```
ALPHA = 0.2
filtered = 50.0
for raw in (60, 60, 60):
    filtered = ALPHA * raw + (1 - ALPHA) * filtered
    print(round(filtered, 2))
```

2. Using the lesson's rule of thumb, an exponential filter with alpha = 0.1 behaves roughly like a moving average of how many readings? [1 mark]

3. Why should an exponential filter be started at the first reading rather than at 0? [1 mark]

- ☐ **A** Starting at 0 makes the output climb from zero for the first second, which looks like a real signal
- ☐ **B** Starting at 0 makes alpha wrong for the whole run
- ☐ **C** Starting at 0 makes the filter unstable
- ☐ **D** The filter needs a nonzero value to avoid dividing by zero

4. Compared with a moving average giving similar smoothing, what is the main practical advantage of the exponential filter? [1 mark]

- ☐ **A** It stores only its previous output, instead of the last n readings
- ☐ **B** It has no delay
- ☐ **C** It rejects outliers
- ☐ **D** Its output reaches a step's new value in exactly one step

5. Rearranged, the filter is filtered = filtered + alpha * (raw - filtered). In the Kalman filter of U6.3 the update has the same shape. What is different there? [1 mark]

- ☐ **A** The weighting is computed each tick from how uncertain the belief is compared with the measurement
- ☐ **B** The weighting is always 0.5
- ☐ **C** The measurement is filtered twice
- ☐ **D** The difference raw - filtered is squared

6. In signal processing terms, the exponential filter is a first order what kind of filter? [1 mark]

7. Put these lines in order to run an exponential filter while driving. [1 mark]

Number the lines 1 to 5 to put them in the right order.

___ `filtered = ALPHA * raw + (1 - ALPHA) * filtered`
___ `for i in range(60):`
___ `raw = distance()`
___ `plot("filtered", filtered)`
___ `filtered = distance()`

The task: a low pass filter

Drive towards the wall with an exponential filter running, plotting `raw` and `filtered` as you go, and print the `alpha:` you chose.

```
from bugbot import *
connect()

ALPHA = 0.25
filtered = distance()
```

Plan your program here, then type it in and press Run.

Do it on the robot

www.bugbotlab.com/learn/u4-3-the-low-pass-filter/

The simulator checks it and tells you when it passes. Nothing to install, no account.

Challenges

1. Run the same drive at alpha 0.05, 0.2 and 0.6 and compare the charts.
2. Implement the moving average as well and plot all three. Which is smoother for the same delay?
3. Filter `imu()[1]` instead. What is a sensible alpha for a signal you want to use in a controller?