WORKSHEET

U6.3 The Kalman gain

State estimation · University · about 35 min

Name ____________ Class ____________ Date ____________

What this lesson is about

Where the weighting comes from: two variances, one line of arithmetic, no taste involved.

- The scalar filter, in full
- Reading the gain
- The steady state gain

Questions

6 marks in all

1. A scalar Kalman filter has p = 100 and R = 25. What is the Kalman gain k? [1 mark]

2. One full predict and correct of the scalar filter. What does it print? [1 mark]

```
x, p = 0.0, 100.0
v, dt, Q, R = 20.0, 0.1, 0.5, 25.0
measured = 5.0
x = x + v * dt
p = p + Q
k = p / (p + R)
x = x + k * (measured - x)
p = (1 - k) * p
print(round(k, 3), round(x, 3), round(p, 3))
```

3. The steady state gain from the lesson's formula, checked by running the filter. What does it print? [1 mark]

```
Q, R = 0.5, 25.0
p = (Q + (Q ** 2 + 4 * Q * R) ** 0.5) / 2
print(round(p / (p + R), 3))

p = 100.0
for i in range(200):
    p += Q
    k = p / (p + R)
    p = (1 - k) * p
print(round(k, 3))
```

4. Q stays at 0.5 and R is doubled from 25 to 50. What happens to the steady state gain? [1 mark]

- ☐ **A** It falls, because the measurement is now trusted less
- ☐ **B** It rises, because there is more noise to correct
- ☐ **C** It halves exactly
- ☐ **D** It is unchanged

5. The filter is started with p = 0.01 while the estimate is in fact badly wrong. What happens for the first second or so? [1 mark]

- ☐ **A** The gain is near 0, so it largely ignores the measurements
- ☐ **B** The gain is near 1, so it jumps to each measurement
- ☐ **C** It diverges immediately
- ☐ **D** Nothing different from starting at p = 100

6. Once the gain has settled, which statements are true? [1 mark]

Tick every answer that is true.

- ☐ **A** The filter behaves like an exponential filter with alpha equal to the settled gain
- ☐ **B** The gain can be computed once and hard-coded, skipping the variance arithmetic
- ☐ **C** The filter has stopped using the measurements
- ☐ **D** The gain is always 0.5 at steady state

The task: a Kalman filter in one dimension

Run the scalar filter while driving at the wall. Plot `measured`, `estimate` and `gain`, with the gain as a percentage so that it shows. Print `my y:` and `final gain:`, the gain itself as a number between 0 and 1. No `position()`.

```
from bugbot import *
connect()

DT, START_GAP = 0.1, 110.0
R = 25.0
Q = 0.5
est, p = 0.0, 100.0
```

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

Do it on the robot

www.bugbotlab.com/learn/u6-3-the-kalman-gain/

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

Challenges

1. Start `est` at 30, which is wrong, and `p` at 0.01, which says you are sure of it. How long does the filter take to believe the measurements, and how does that compare with `p` at 100?
2. Work out the steady state gain from the formula and check it against the one you measured.
3. Double `R` and halve it. Which way does the gain go, and does that match the sentence "R is how noisy the measurement is"?