Complementary Filter

Or

A Special Instance of an

Alpha-Beta Filter

Or

How I Came to Love Weighted Averages

Overview of Averages

- What is the average of 6 and 10?
  - Avg = (6 + 10) / 2
- What about the average of 6, 10, and 12?
  - Avg = (6 + 10 + 12) / 3
- In general form:
  - Avg = sum(values) / number of values

A Different Way to Look at Averages

- Avg = (6 + 10) / 2
- Avg = 6/2 + 10/2   (Distribute)
- Avg = (6 * 0.5) + (10 * 0.5)   (Inverse)
- Avg = (6 * 50%) + (10 * 50%) (Definition of percentage)

- Avg = (6 * 33.3%) + (10 * 33.3%) + (12 * 33.3%)

- All numbers have an equal contribution/weight in the average

Weighted Averages

- What if I told you the third voltage measurement in a series is always more accurate than the first two?
- You could discount the first two, or you could take more of a share of the third value when computing the average.

- Avg = (6 * 20%) + (10 * 20%) + (12 * 60%)
- Avg = (6 * 15%) + (10 * 20%) + (12 * 65%)
- Avg = (6 * 05%) + (10 * 10%) + (12 * 85%)

- In all cases the percentages add up to 100%.

Complementary

- We are not talking about free nuts at the bar. (That is complimentary)
- Complementary is a math term meaning "to complete a whole"
- In our case the 'whole' is 100%.

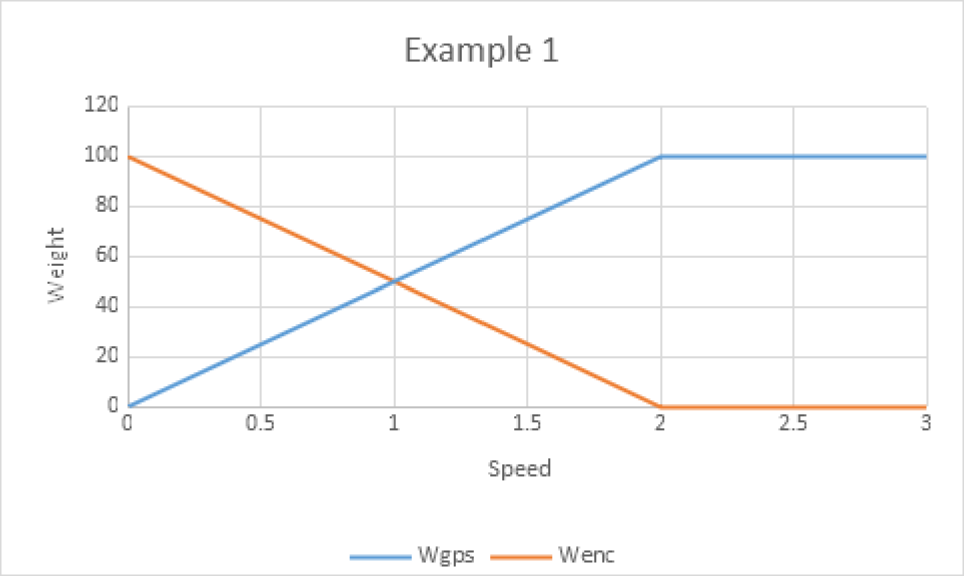
- Avg = (6 * 15%) + (10 * 20%) + (12 * 65%)
- Avg = (6 * W1) + (10 * W2) + (12 * W3)
- Avg = (V1 * W1) + (V2 * W2) + … + (Vn * Wn)
  - Where W1 + W2 + … + Wn = 100%

Example 1

- Robot with wheel encoders and GPS sensor each providing speed.
- $V_{avg} = (V_{enc} * 50\%) + (V_{gps} * 50\%)$
- But GPS can't provide a velocity when traveling slowly.
- Encoders start to slip when traveling fast.
- These two sensors complement each other!
  - GPS is good going fast, and encoders good when going slow.
- $V_{avg} = (V_{enc} * W_{enc}) + (V_{gps} * W_{gps})$

Example 1

- GPS speed is assumed perfect when going above 2 m/s (100%)
- Encoders are assumed worthless when going above 2 m/s (0%)



Example 1

- Ratio = Vavg/2.0;   // 2 is from 2 m/s
- If (Ratio > 1.0) Ratio = 1.0;
- Wgps = Ratio;
- Wenc = 1.0 – Wgps; // Wenc is the complement of Wgps
- Vavg = (Vgps * Wgps) + (Venc * Wenc); // Weighted average

Example 2

- Compute heading of a robot with magnetometer, GPS, and gyro.
- Magnetometer is useless if motors are turned on.
- GPS heading is useless unless moving at 2 m/s
- Gyro heading degrades over 120 seconds since last discipline.

- This is not an easy split of weights as last time!

Gyro Discipline

- A gyro provides an angular rate output not an angle.
- You can integrate the rate to get an angle.
- That angle is not absolute but relative.
- Somehow you need to set the gyro what the real heading is.
- This is called disciplining a gyro.
- It can be done lots of ways including:
  - Pointing the robot north than pressing a button.
  - Transferring the heading from another sensor (like a compass)

Example 2

Weight for the magnetometer

```
If (ESC == 0.0) {     // Electronic Speed Control
      Wmag = 1.0;
} else {
     Wmag = 0.0;
}
```

Example 2

Weight for the GPS heading

```
Ratio = speed/2.0;      // 2 m/s
If (Ratio > 1.0) Ratio = 1.0;
Wgps = Ratio;
```

Example 2

Weight for the Gyro

```
tRatio = 1.0 - timeSinceLastDiciplineInSeconds / (2*60);
If (tRatio > 1.0) tRatio = 1.0;
If (tRatio < 0.0) tRatio = 0.0;
Wgyro = tRatio;
```

Not Complementary

- The weights as just computed are not guaranteed to be complementary.
- If the robot has been stationary for a while, then
  - Wmag = 1.0
  - Wgps = 0.0
  - Wgyro = 1.0; // Constant disciplining.
- If the robot is creeping along
  - Wmag = 0.0
  - Wgps = 0.0
  - Wgyro = Depends on last time the gyro was disciplined
- If the robot is moving quickly
  - Wmag = 0.0
  - Wgps = 1.0
  - Wgyro = Depends on last time the gyro was disciplined

Normalizing the weights

- sum = Wgps + Wmag + Wgyro
- Wgps = Wgps/sum
- Wmag = Wmag/sum
- Wgyro = Wgyro/sum

- Watch out should sum be zero!

- Other ways might be more appropriate for your situation

Sidebar on Averaging Headings

- Angles are a pain because they are ‘circular’.
- There is a discontinuity when you go from 359 to 0 degrees.
- Or at +- 180 for ‘half circle angles’.
- Average of 359 and 1 degree should be 0/360 but gives you 180!
- It is very difficult to write exception code to take this into account.
- It is even more difficult to do this with differing weights.
- But there is a trick!

Averaging Headings

```
sumSines    = sin(H1)*W1  + sin(H2)*W2   + … + sin(Hn)*Wn;
sumCosines = cos(H1)*W1 + cos (H2)*W2 + … + cos(Hn)*Wn;
avg = atan2(sumSines, sumCosines);
```

Computationally a bit expensive, but it solves the logic issues.

Just be careful when both sums are zero!

Example 3

Drone with an accelerometer and a gyro trying to determine pitch.

The accelerometer works very well over long periods of time.

The gyro works well over short periods of time.

Sounds like a job for “Complementary Man”!

Example 3

$W_{Accel} = 0.8$

$W_{Gyro} = 1.0 – W_{accel}$

$Pitch_{Avg} = Pitch_{Accel} * W_{accel} + Pitch_{Gyro} * W_{gyro}$

“Lets take most of the accelerometer and only mix in a bit of the gyro.”

Time Filter For Noisy Data

A small variant of this filter is very useful for time averaging noisy inputs. (exponential averaging)

$$Volt_{Avg} = Volt_{Avg} * (1-w) \quad + \quad Volt_{Raw} * (w)$$

$w = 1 - e^{-\Delta t/\tau}$ Where Δt is the sample time and τ is the time constant.

"Take a chunk of the old average and add in a bit of the new value."

Kalman Filter

- Kalman filter is not much more than a glorified complementary filter
- Differences
  - The weights are computed using statistics.
  - One of the inputs is a math model of how the system should work.

Questions?