Comp2403 Robot Kinematics - Wheel Encoders (2)

C.B.Price October 2021

Purpose	(i) To learn how wheel encoders work with a PID algorithm (ii) To get the robot to move on a straight line (iii) To get the robot to move on an arc
Files Required	Arduino Sketchbook and Octave scripts on the web-pages.
ILO Contribution	LOs 2
Send to Me	nix
Homework	Read chapter 1 (updated 09-10-21)

Activities

---

1 Moving on a Straight Line using a PID controller

Here we shall review the straight-line approach, but using a PID controller. The approach is similar to the FSM algorithm, first we calculate the desired steps **nL** and **nR** and then we measure the actual encoder trigger counts **countL** and **countR**. The difference produces an error. There's not much to do:

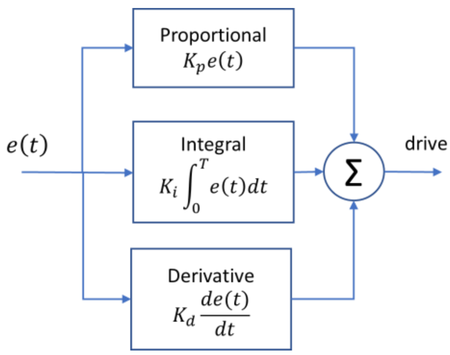
(a) Calculate the error between **countL** and **countR** which should be the same when we are moving on a straight line

```
error = ( (float)countL – (float)countR ) / ( (float)countL + (float)countR  );
```

Two comments (i) **countL** etc are type 'unsigned long' so we must cast them to floats before we divide to the the float **error**. (ii) we divide the difference of the counts by their sum; this guarantees that **error** will be in the range 0.0 – 1.0. So, we know what it can be.

(b) Calculate the drives

```
driveL = drive - Kp*error - Kd*deriv;
driveR = drive + Kp*error + Kd*deriv;
```



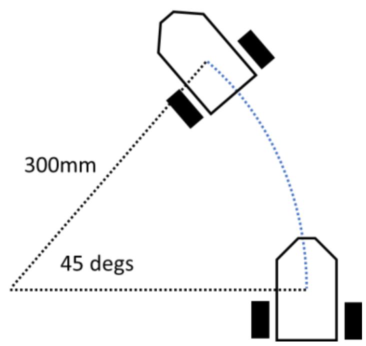
Two comments here. First, we *add* the proportional and derivative terms, since that's how the PID algorithm works, here's a diagram to remind you. Second, we subtract the error from the left drive and add it to the right drive. This gets the wheel speeds increasing and decreasing to keep the robot moving straight.

(c) Some values for **Kp** and **Kd** have been suggested. Run with these, then perhaps investigate a bit changing these, based on your deep understanding of PID controllers.

---

2 Moving along an arc using a PID controller

The approach is similar to moving along a straight line, but here one wheel (we shall take the right) moves further, and since both wheels take the same time to complete their run, the right wheel must move faster. See chapter 1 for the associated theory. Here's an example of our task



(a) Choose values **desRadius** and **desDegrees** for your arc. Specify the angle in degrees!

(b) Calculate the number of steps for the left motor

```
nLf = (desRadius - axleLen/2.0)*desTheta/dx;
```

(c) Repeat for the right motor

(d) The forward speed of the left motor has been set to 20. Calculate the larger speed of the right motor

```
fwdR = fwdL*nRf/nLf;
```

(e) Calculate the error as

```
error = (float)(countL)/nLf - (float)(countR)/nRf;
```

(f) Some starting values for **Kp** and **Kd** have been provided. You might like to experiment with these.