Motor simulation and implimentation in Matlab.

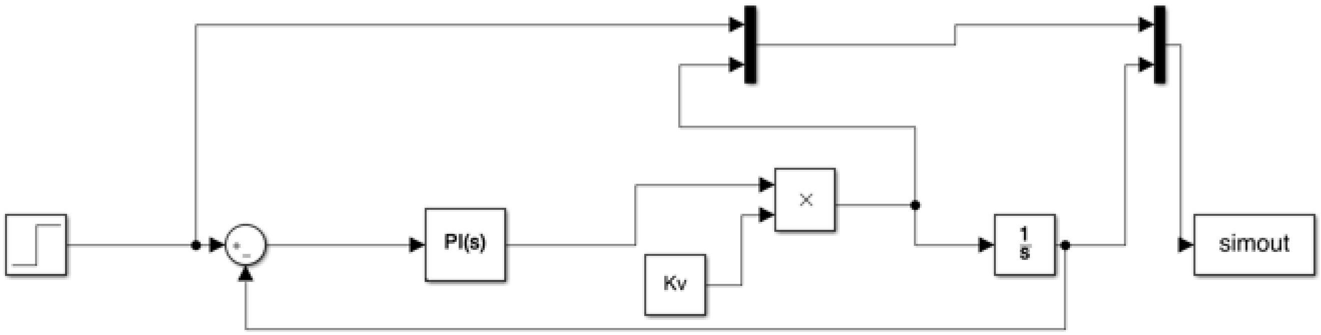
simulations where done for 5 linear feet and 180 degrees to find ki and kp values. A PI controller was used as given the large time allowance it was not neccessary to utilize a PID controller as Ki sould be adjusted accordingly

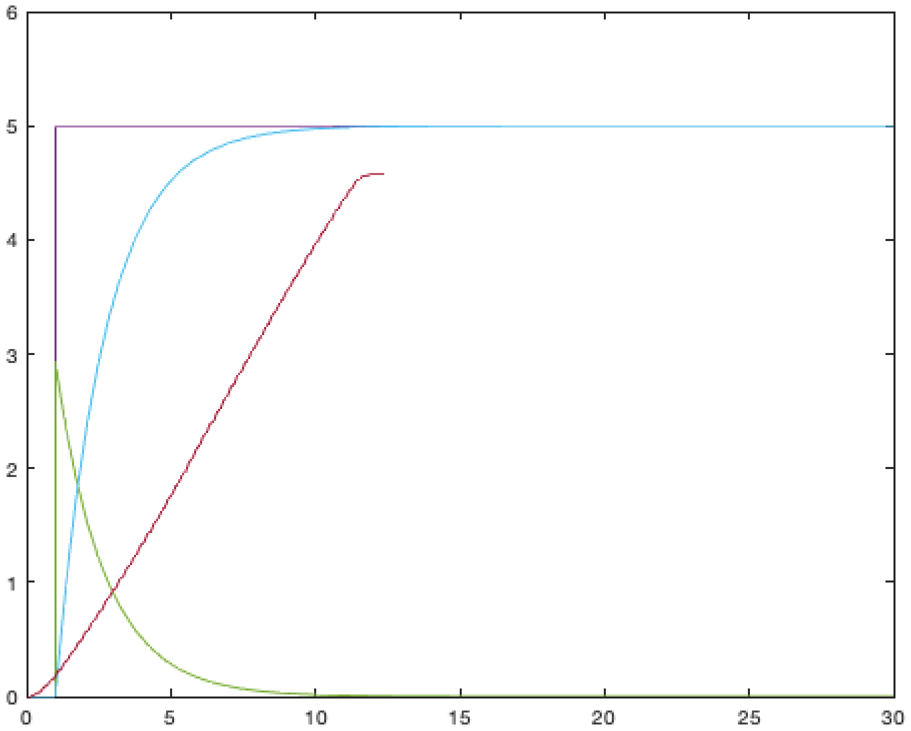
Contents

Distance Control and simulation of robot using PI controler

Purple is distance desired. Green is output voltage. Blue is modeled distance. Red is the actual output. This is the results from the final tuned motor controller. Clearly it is evdient that matlab is helpful for getting an idea of modeling the output, however final testing should be done. The results are most likely off as ther is small wheel splippaged and modifications to the wheel size in arduino code where made during the tuning. As such the robot final position reads slightly lower. However this is satifactory as the final positiononing of the robot is correct.

```
Distance = 5;
Kv = 13; %kv of the motor with loaded robot
Kp = 0.045;
Ki = 0.000001;
out=sim('simulinkturn');
plot(out.simout)
hold on
plot(out.simout)
A = importdata('arduinoposition.txt');
time = [0:0.12:12.4];
plot(time, A)
hold off
open('simulinkturn');
```

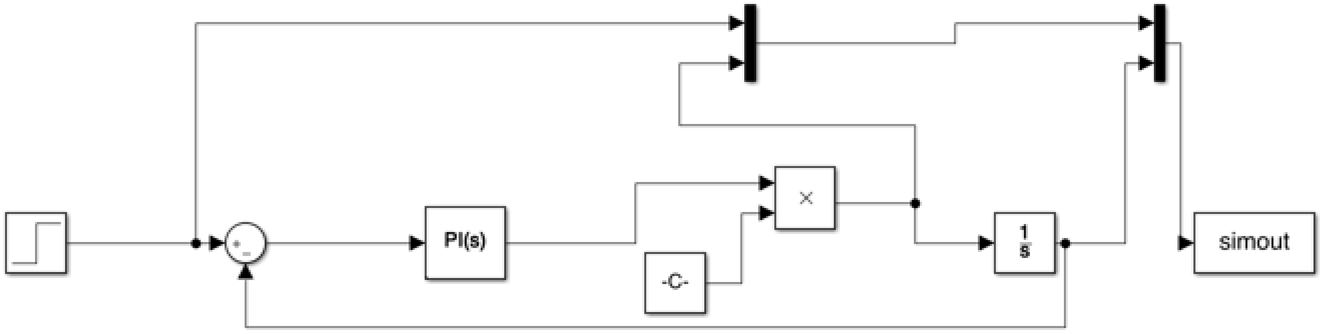


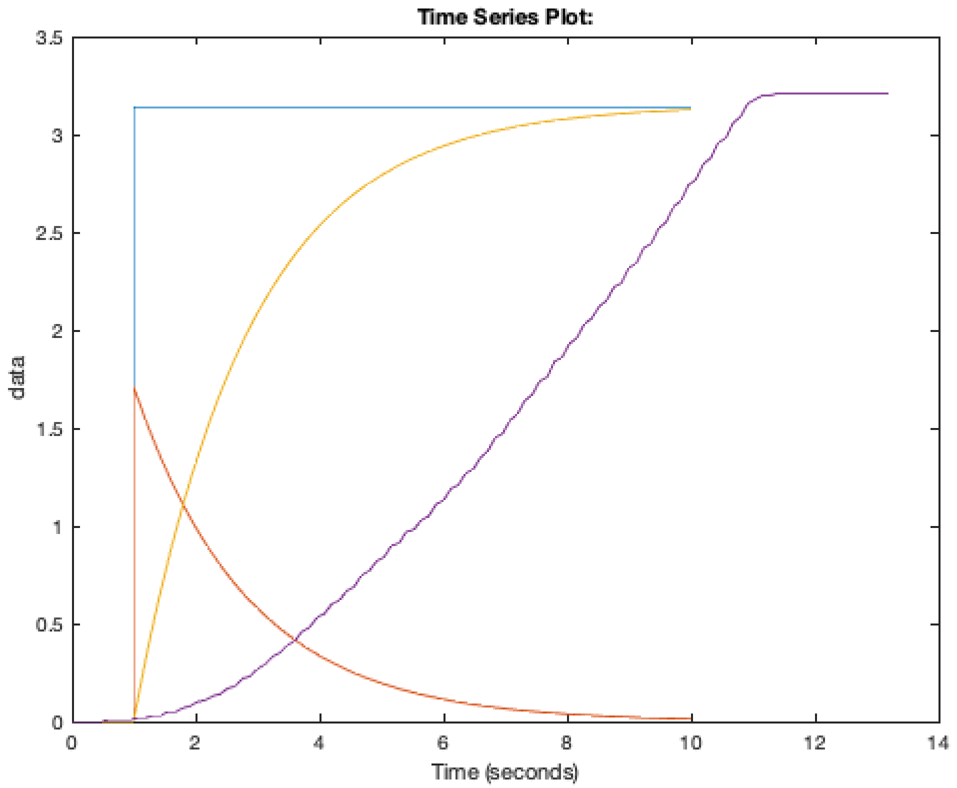


Spin Control Model vs Real simulation.

This output is fairly similar to the linear position output. This is due to the fact that although matlab is helpful for getting a rough idea of the final output it is still necessary to conduct many test runs with real life scenarios. Blue is the desired angle in radians. Orange is the modeled output in radians. Red is the voltage. Purple is the actual recorded output. The ki and kp values had to be tuned extensivly in real life simulations in order tp prevent skidding overshoot due to angualr momentum of of the robot.

```
AngleRobotGo = 3.14159;
Kpspin = 0.07;
Kispin = 0.0001;
out=sim('simulinktturn1');
Kvspineqiv = 7.78399;
plot(out.simout)
hold on
B = importdata('arduinospinposition.txt');
time = [0:0.12:13.3];
plot(time, B)
hold off
open('simulinktturn1');
```





Published with MATLAB® R2019a