Ideas/To-do/thoughts/responses

4/11/19

* Went over code briefly
* Talked about the policy optimisation sometimes getting stuck
  + Could try a combinatiokn of policies
  + Implement random actions still to discover new information

9/11/19

* Set-up github and code on my laptop and desktop
* Read through cart pole example code, and simulated some runs

10/11/19

* Read through unicycle simulation to understand the key processes in the code, and how I would change them to experiment on roll angle and other parameters
* Initiated tests onmy laptop and desktop to test speed and output data form
  + 4hrs for a test limited to 17 degrees on my desktop
  + Output data is all the simulation variables after each policy iteration, currently set to 40 policies
    - The variable all\_data contains the trajectories and actions taken during these rollouts

12-13/11/19

* Read through unicycle code in far greater detail
  + Understood the form of the cost function
    - This is apparently a messy part of the code and cold do with reviewing
    - The exponential form comes from the desire to want a reward that decays, if you were just using a quadratic function then rollouts could have massive losses. These losses would originate from the distribution of state space traversed in the rollout, and the fact its expected loss that is computed.
    - Noticed an error in the function compared to the desired function in the reports and pdf (1/2pi not 1/4pi)
    - Drew diagrams of the variables in question ig a unicycle to better understand the wording and augmentation equations
    - Is there a specific reason all elecments of the cost function aren’t seperable? Hats the benefit of using d(x) which combines the effect of roll yaw and position?
      * Why not use three separate terms to better select their scaling? Or is this providing too much information?
      * Using separate items gives the unicycle model information about what is a good thing to keep an eye on
      * This is why the original term was fully geometric (d(x)) so should be kept, as we don’t want to mould this function with prior knowledge
      * The second term was to ensure the unicycle didn’t learn to balance by spinning really fast ( the 4pi signifies two rotations being bad not just one)
  + Noticed where the roll limit is called
    - This roll limit determines the new additional cost function term and when the simulation stops
      * Added in simulations the trajectory cuts out if violated
      * The new roll limit is there to make the unicycle know this is a bad position to be in
    - Could these terms separate help to improve learning?
      * When its at the roll limit the exponential is exp(…. -0.5) <0.6060, this means the instantaneous cost could still be ~0.4 which is relatively low
      * The cost then jumps to 1 the next step, however no action was taken to get here so does the process really learn that being at this limit was bad?
      * The limit of 17 degrees is often reaced in two timeseps
        + This means only one action per rollout is recorded
      * Propose that the cost function should approach 1 at the limit, possibly making the cost limit half the real limit, or in the case of a higher limit (eg 42 or 50 degrees) make the cost limit where the motors are unable to help any more (~38 degrees)
        + Research some benefits of cost functions
    - Want to test the success of changing variable parameters if a new unicycle were to be built
      * Test the performance on (d0 3-5 tests of each)
        + average(+sd) of time upright vs policy iteration number
        + time upright vs simulation time, as the larger roll limit will enable more simulation time per rollour
        + loss vs policy iteration
        + loss vs simulation time
      * Test factors
        + Change roll limit with additional cost term
        + Change roll limit without additional cost term

One mechanical redesign involved changing wheel size…. Incorporate this

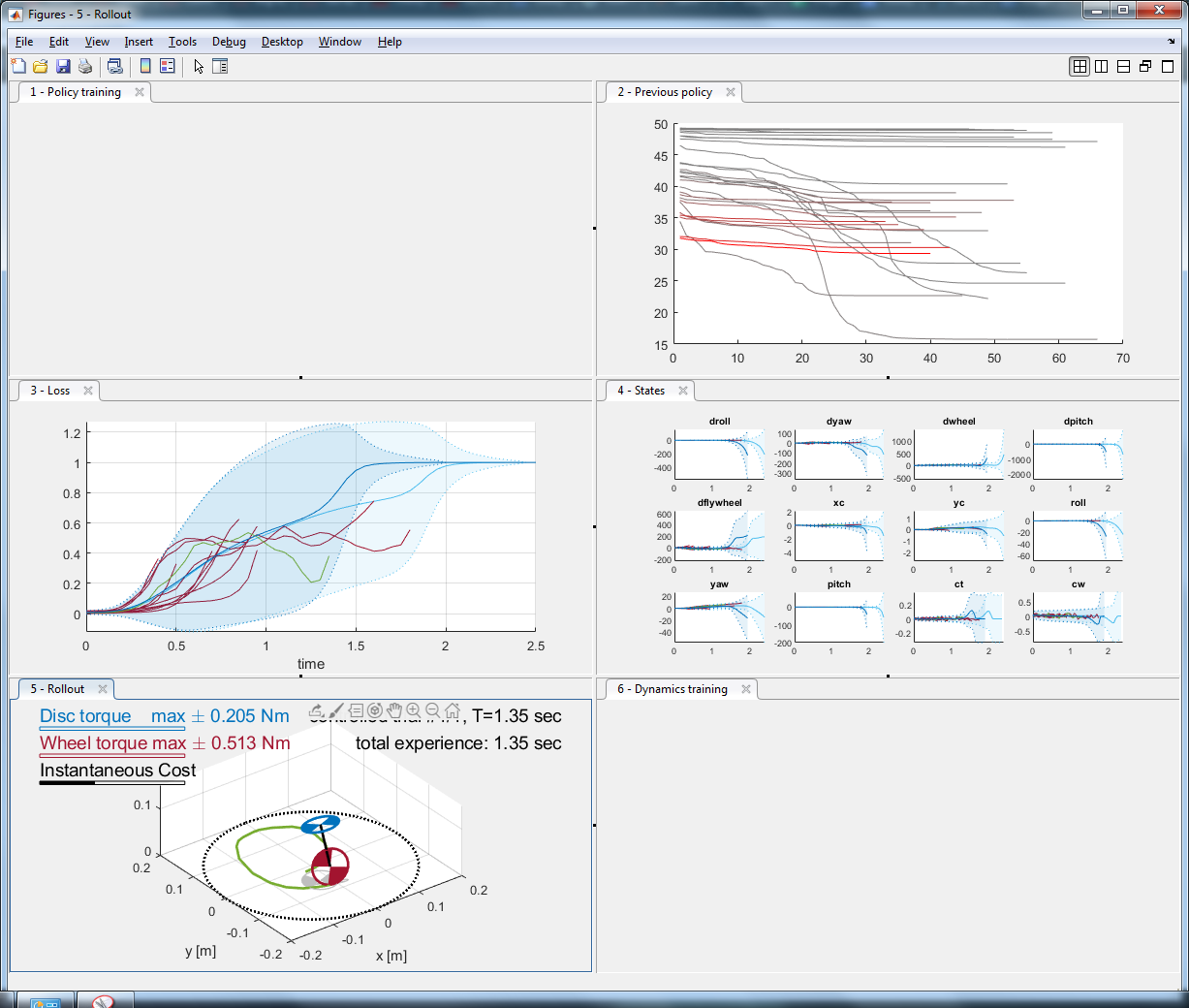
* + - * + Separate cost limit and roll limit
        + If successful for a redesign test motor maximum torques
      * Write script to extract these variables from each test run and store them in a easiler location (Or do this across a simulation and output a smaller variable at the end??)
      * Write script to combine multiple of these files to produce the desired graphs with uncertainties
  + Fixed the view of the rollout model to be 3d
  + Noticed the docked windows aren’t all there
  + Error pops up when hovering over certain graph areas
  + The unicycle model puts the unicycle coordinate system at the turntable centre
    - This is not an accurate representation of the phones sensors coordinates in real rollouts
      * This means that real unicycle has greater change in velocity and position for given angular accelerations
        + Thus changing the cost associated with each angle

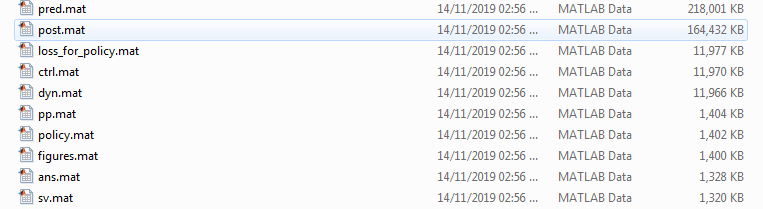
How is the global position calculated????

* + - * + Possible correct this to be above the true area
        + Does the non centralised location of the phone sensors make a difference?
  + Add button to replay the trajectory video
    - This was mentioned in erics report as done however it appears broken (context menu entry)
    - This has been found as right clicking a specific line in the states or loss
    - Could add time slider to slow down animation
* Look into applycontroller , dyn.train and showposteria
  + How many rollouts get run after each iteration? I thought it was one , but then why are there many lines on each graph
  + There are 10 rollouts plotted but only one is ever used for training the policy
  + However it plots 10 so that you can assess if it was a good rollout to pick or an anomoly
  + The animation is plotted using the latenent states, these latent states are the real states
  + The states used in training however are the accessible states and are the latent states plus measurement noise. We don’t want to plot these noisy functions tho
  + Plotting is done in this order
    - Prior distribution
    - Then 10 plots of trajectory (find this in code somewhere??)
    - Then animate direction
    - Then posteria
    - The dymanics training is not plotted because this does 1000’s of iterations and the graphing is slower then the optimisation meaning its not worthwhile
      * This is controlled by a verbosity parameter in the optimisation which defines how much output should be given for the optimisation.
* Possible change the state plots to be zoomed in more on the lines instead of the exploding uncertainties
* Display the itation number and current policy evaluation on figures
* Make the policy evaluation connected lines not a scatter plot as this is hard to read
* Is the policy model retrained on one rollout, could this number be changed to see its effect?
* Non linear controller
  + It was a surprise that the linear control with about 20 parameters worked,
  + Using a quadratic or rbf one was suggested and is implemented in pilco but using just 5 rbf with 20 parameters greatly increases the search space of optimisation to be very large and therefore expensive
  + Also possible to use a gaussian process to model a group of functions that fit pretend data points, these data points being the policy parameters. This is like flipping a GP on its head
* Plotting features break after certain amount of runtime
  + Closing down the gui and restarting can bring these features back as the GUI can die in matlab after a while

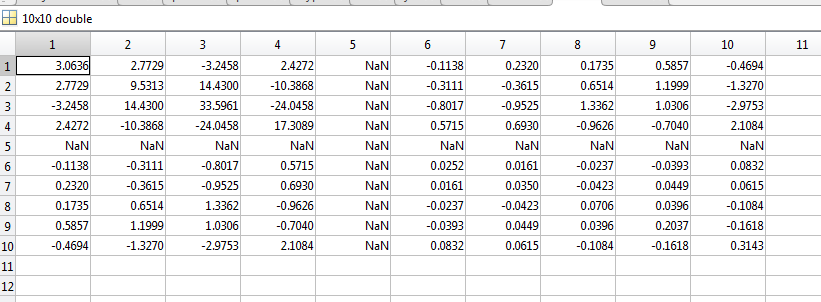
Could compare dynamic modoels from both simlations and real runs to compare the accuracy

14/11/2019

* The three simulations that ran all came to a solution that didnt fall over but kept moving away from the origin
* Corrected the 1/2pi term in the cost fn to be 1/4pi
* Looked at the variables
  + The useful information for a test run is all stroed in the last generated file
  + Trajectories contains the 50 runs with loss and actions
  + The file size ends up very big with about 5-6GB for a run
    - Each iteration generates a new file which gets progressivly larger up to 500mb , which makes sense because the runs are getting longer
    - Two variables proved to be much larger hen the rewst pred and post were about 200mb each
    - These both store the distributions (gaussians) of state, action and loss for each rollout, at each timestep. So approximatly 8000 gaussians, with each gaussian being a 10x10



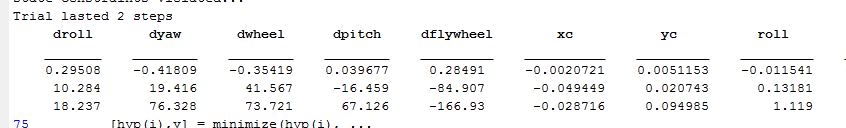
18/11/19

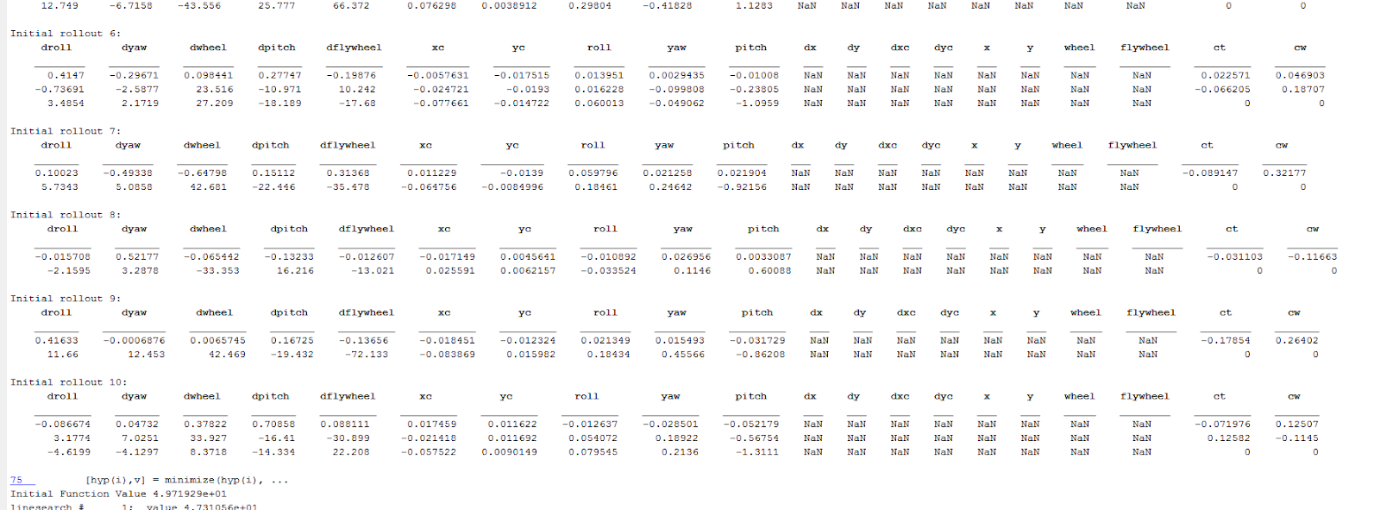
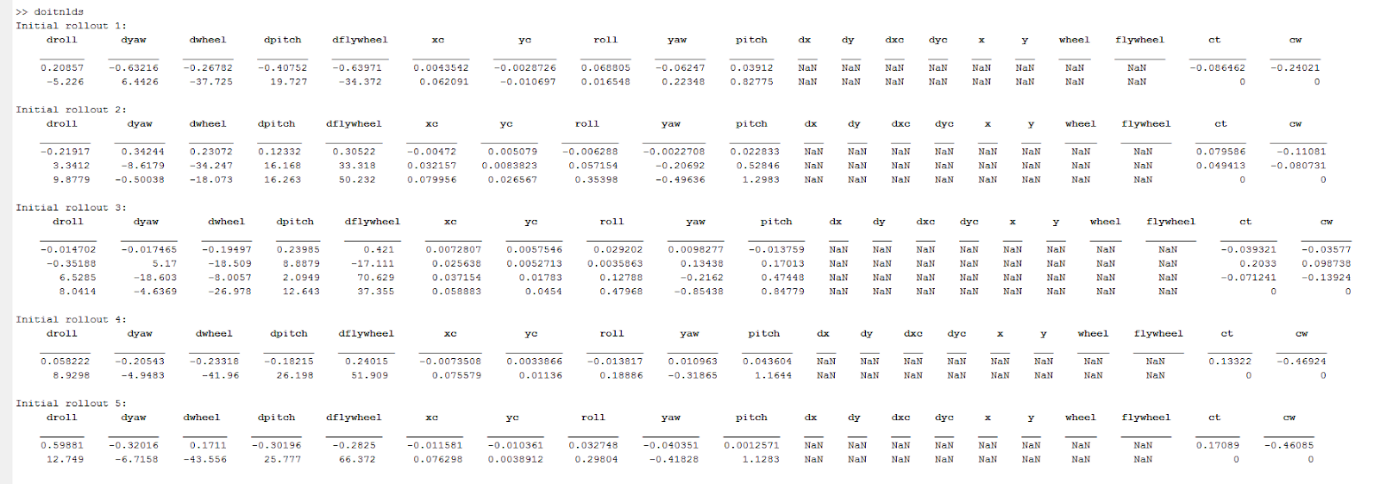


19/11/19

Noticed that rollouts are the same for the first run after initial 10?

Are the first 10 the same…..





4/12/19

Meeting notes

Interface in the cost function is a bit of a mess could reqrite to just be for the unicycle

Effect of lowering the timestep

Easier to learn dynamics

Each time the policy is called you inject a bit of noise into the latent states, as the policy was based off a measured state which had uncertainty

Smaller timestep means that the zero order hold control approach is wrong for less time, averages

Getting the phone to connect by usb

Wifi could be slower and unreliable due to very small cutouts in signal that are normally unnoticed

Itial orientation check in both testing and training

Could be with a gravity check, is this reliable enough

Or you incorporate additional hyper prameters for an itial off set which the policy will learn

How do you do this in testing on the fly?

Sensor error not independent , is this worth addressing

Change simulation to calculate angles by numeric integration

Is the drift significant? Apparently 1 deg per min

Low cost value at failure

Test out on linux/mac

16-20 dec

* Ran some more angle tests

21/12/19

* Read through the code that runs on the rasberyy pi and communicates with the phone
  + Why is the noise level of an observation 0?
* Currently a PC is connected to the phone by wifi, this will provide a log to the PC of the angular velocities
* The raspberry pi is then connected by ehernet in order to control the motors

Alternative method

* No cables attached to unicycle during motion as this could add extra noise
  + Disadvantage is possibly the need to reconnect after each rollout
* Attach the phone to raspberry pi by usb
  + Run the policy on the phone and communicate to the raspberry pi the required torques
* Transfer the trajectory back to the computer after a test
  + - How big is the traj file?
    - Is this too large for wifi?
      * Wifi connection loss is not an issue when transferring this
    - How much computing can the phone do?
      * Will applying a large policy require too much ram?
      * Will the phone be as fast as a computer?
    - How big are the variables during a rollout test
      * Policy, controller, H, plant, cost, verbosity (start can be determined by a gravity check at the start of a run)
      * Can these be transferred easily?
      * Can these be stored on device easily
* Run policy optimisation on the computer and then re initate a rollout on the unicycle

Turns out that matlab mmobile only has limited functionality

This means its now able to run a callable script

So need to change the wifi connection between pc and phone to a usb

* This means it is entirely wireless but will have two attached cables….

Todo

* Try and communicate between the phone and the raspberry pi
* Test file sizes and transfer mechanism
* Check computation speed
* Add gravity check
* Add a way to initiate a rollout with a button or command line thing… from pc
* When cabled method is working
  + Run lots of measurements using both methods and the phone held still to measure the noise levels associated with wifi and with usb
  + Check the timinging with the usb to ensure each measurement is made correctly

22/12/19

Trying to use

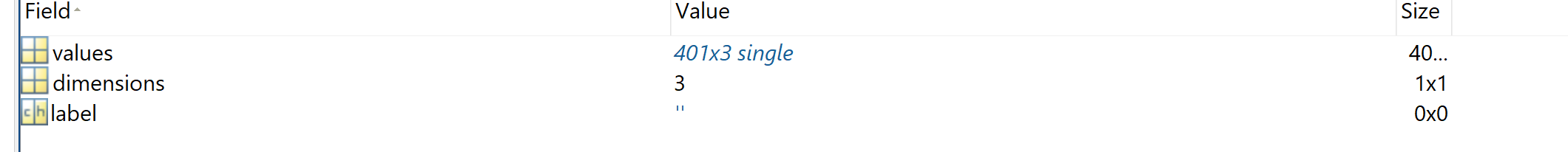
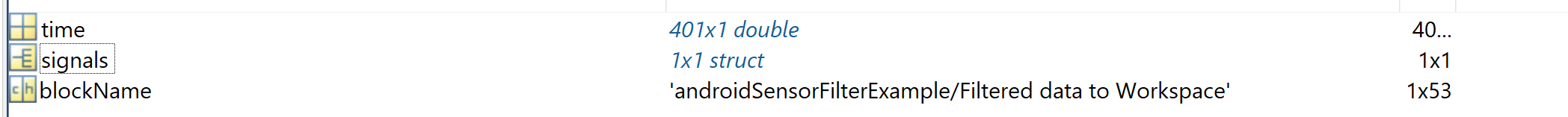
Simulink support package for android device :

<https://uk.mathworks.com/hardware-support/android-programming-simulink.html>

<https://uk.mathworks.com/help/supportpkg/android/examples/sensor-application.html>

With android studio (version 3.4.2 later version didn’t work on install) <https://developer.android.com/studio/archive>

* Setup takes a long time
* Usb connection with cabe I had was very wobbly and insecure
* Device must be in developer mode
* Device and computer must be connected to the same wifi network for some reason
  + The network in my room didn’t work
  + But the loft did
* Can then build a model in Simulink which can be run
* Make sure the location of the Simulink file is in a short path name directory and the directory has no special characters
* Need to figure out how to collect data in real time
  + Might need to figure out how to use external mode
  + Need to figure ut how to call this model from a matlab scrpt
* Examples files stored here C:\ProgramData\MATLAB\SupportPackages\R2019b\toolbox\target\supportpackages\android\androidexamples



Returns this ^

23/12/19

* Started work on calculating orientation from acceleration data
* Realised that mobiledev is missing from matlab 2019b which makes the steaming over wifi no longer possible in this version
  + Could try and backdate it to fix this issue
  + Or will need to get the usb connection to work

24/12/19

* The Simulink approach doesn’t allow live data transmission to a matlab workspace
  + As far as I can tell its only transmitted after the simulation stops running
  + Workaround would be to run the simulation for a very short time
  + However this would lose track of overall time information
    - This means that delays in opening the model aren’t accounted for and mean that the time used for readings are incorrect
* To access the data by usb directly I would need to develop an app to transmit this data sensibly
  + Not the main purpose of android apps
  + So far been unable to find one that uses usb not wifi
* Some premade apps seem to transmit over wifi again, if I used this I would need to figure out how to interface this with the matlab workspace as its running?
* Will try downgrading matlab and testing the old function first
  + Need to check noise, signal distuption etc

27/12/19

Method 1 – Android to pi to pc

* Could try and communicate between the pi and phone and then use the ethernet cable to get the phone data
  + <https://www.hackster.io/jm4rc0/android-sensors-as-imu-for-raspberry-pi-2-via-usb-and-udp-452bad>
  + This uses usb tethering
  + Not sure if the time data is recorded in the app provided
* Communicate with the pi in a few ways
  + Connect to serial device on the pi
    - <https://uk.mathworks.com/help/supportpkg/raspberrypiio/ref/serialdev.html>
    - <https://uk.mathworks.com/help/supportpkg/raspberrypiio/ref/raspi.serialdev.read.html>
  + Some kind of script on the device which gets data and then passes it on
    - <https://uk.mathworks.com/help/supportpkg/raspberrypiio/ref/raspi.system.html>

<https://uk.mathworks.com/matlabcentral/answers/278605-read-accelerometer-from-raspberry-pi>

<https://uk.mathworks.com/matlabcentral/answers/385237-send-data-from-raspberry-pi-to-matlab>

<https://www.quora.com/How-can-I-send-sensor-data-from-Raspberry-Pi-to-MATLAB>

Method 2 – Phone to pc via usb

* Cant seem to find any premade apps that send the data by usb protocols
* Would need to make one

Method 3 – phone to pc via Bluetooth

* Possibly more reliable then wifi?

Method 4 – phone to pc via wifi

<https://uk.mathworks.com/matlabcentral/answers/143073-how-to-receive-udp-packets-in-matlab>

All the methods don’t require a whole log of thr data to be sent, this will slow down the response. The time of data acquisition should be recorded

Serial communication is becoming non existent on pi,pc and probs android so it ends up being very complicated

Method 1 works

* Use usb tethering in order to ensure the pc and phone are on the samre network
  + On phone its settings, wireless and network, thetering &personal hotspots , usb tethering
  + Check the pcs ipv4 adresss in cmd with ipconfig
  + Use an app that transmits using udp
  + And a script that reads by udp
  + This doesn’t require internet as it works with both devices in airplane mode
* Need to choose a suitable app that gives the required info
  + Acceleration, gyroscope , time , adjustable refresh rate?
  + Unpack the data into matlab
  + HyperIMU seems the most suitable
* Need to try and automate the ipv4 connection
  + Add a user input to the real roll out, either requesting this or just reuiring hitting enter or something to ensure the user is ready for a rollout to begin
* Change real\_roll out to work with live data not in blocks
* Test refresh rate

See if sensor calibration will be required sometimes?

30/12/19

* Started writing the packet conversion to arrays and implementing it into real rollout
* Started implementing initial orientation into the real roll rout
* Tested the connection of the usb
  + Time differences between pc and ohoje proved to be a problem when measuring delay as the clocks were originally 5 seconds out, then 0.5 after a restart.
  + The variation in time difference appeard to be about 2ms at most
  + Worked up to a 5ms sample time

Problems with implementing the initial orientation not sure if its correct

Problems with the first quaternion update, it jumps a lot from the itial position, this could be due to an initial position being incorrect or the quat to euler conversion being wrong as it comes out as a non zero pitch which it shouldn’t.

Appears to be a problem with the initial starting quaternion

First time step appears to take about 50ms, likely due to the initialisation of variables? Either initialise them first or skip the first step

Whats the point of the quaternions…..

* Got it working much easier by just using new\_pos = omega\*dt +old position
* <https://folk.uio.no/jeanra/Informatics/QuaternionsAndIMUs.html>
* Is drift a bigger problem doing it this way
* Does doing it this way mess up 3d rotations

3/1/20

* Realised that yaw is actually about the permatnent Z axis not the robots moving zaxis, this means the angular velocities don’t directly correspond to the euler angles
* Roll is down after pitch

shsjdhjhafj

24/1/20

Squared error loss cost fn most commonly used as it often leads to linear equations

Because of how the cost is predictied to the future it all goes wrong very quickly and when taking deriavatives then the tiotal loss is mainly caused by a far out but unlikely point

Not yet verified

1. Gausiian. Is locally quadratic but saturates

Currently doesn’t use dynamic variables to not impose too much knowledge on the system, giving the opportunity to figure out the effect of velocitys

Measure exact delay by doing two way communication

Create a test procedure document

29/1/20

* Processed the results from a 53 degree roll limit and 53 degrees with larger wheel,
  + Unfortunately the whell size was defined twice and I only changed it once so the extra results were the same…
* At 53 degrees time upright was actually worse then other angles, this was due to the robot in lots of trials learning to balance on the spot and not wander

A picture containing text

Description automatically generated

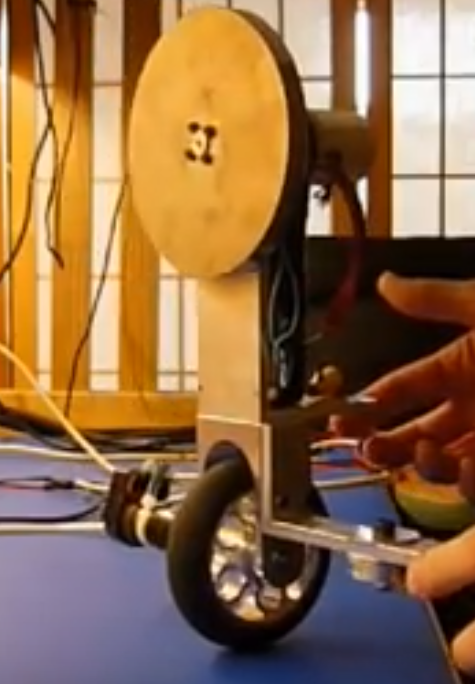
* + This technique shows gradual improvement across trials but its very slow, only really getting more time as the uncertainty is gradulally reduced with more trials and predictions become slightly more accurate
  + Does the extra roll allow the unicycle to try and reverse the direction of tilt, instead of just trying to keep up with it (wheel below com) as the reversing is what is wanted for the unicycle end position
  + Can we fix this to have a mix of both techniques? As the other method of going off into the distance doesn’t learn to slow down at the origin within the time horizon
  + Add exploration with a random controller into the process?
    - Can the chance of this random action be proportional to rate of change in loss across policy iterations?
  + Is it possible to train multiple policices and combine these?
    - gygg
    - <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6170616/>
    - Possible train individually and then apply them with some probability in the model runs. Then train them based on this new data. Would this allow a policy to explore in a way that another policy has deemed useful.
    - Can training different policcyies be done in parallel as to not increase computational time.
    - How would separately trained policies differ?
      * Different controller?
      * Different loss function? Would this be imparting too much prior knowledge on the system, as the loss functions could be made to “force” the system to value both types of balancing.
      * Train on multiple trials at once? Instead of updating the current policy every trial, update it every couple. This would extend training time in this instance.
      * Train each policy on the rollout controlled by the other policy? With each policy initiialed from a different 10 rollouts.
      * Train across two trials, and then combine them after maybe 20 iterations. Then continue from there?
* Is a quadratic controller required?
  + To fix pitch it’s a simple linear controller (tilt forward = +ve torque)
  + However for roll in order to get in a position where roll= 0 and its just a pitch problem then the direction matters when the uniccle is both in roll and pitch. (see arsland report) however im not sure he’s correct. After having fiddled with the unicycle it doesn’t seem quadrtic. You can correct the problem going in either direction, one of them is just easier to do. So while a quadratic controller may be ideal, it should be possible with a linear controller.
  + Is this a caused by the horizonal wheel being present and not a vertical one

Alternative design ideas? Some of these control using a flywheel the opposite way (this separated the roll and pitch problems) and some control using control theory

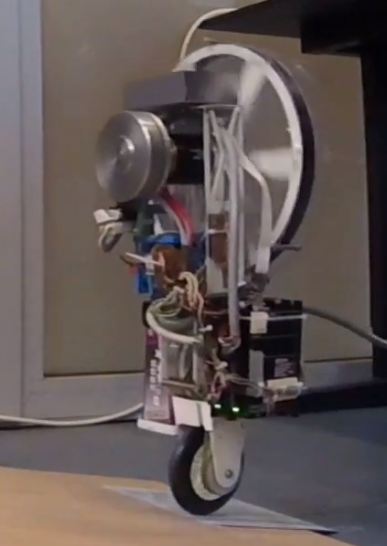
[https://www.youtube.com/watch?v=W5OyoFX0DhI&feature=share](https://www.youtube.com/watch?v=W5OyoFX0DhI&feature=share&fbclid=IwAR2e8RncC3XH9HHQQW4VjuYmizu2gXNNcW3ODunff0w7YPlUlWxE9PGGwIk)



[https://www.youtube.com/watch?v=ANUFWvtGFgU&feature=share](https://www.youtube.com/watch?v=ANUFWvtGFgU&feature=share&fbclid=IwAR0lfzrzopKclO8OKzraOJMe--_Oqs4VkEK18forQzn8HCqeaKhHaiw0XlA)



<https://www.youtube.com/watch?v=RqOYLWgq2QA>



All have large flywheels? Maybe increase the current one to be a higher proportion of the mass? Making a top heavy uniccle may be beneficial, as it improves balance. Rotation is slower as MoI is higher, allowing more time for correction. But this also makes it harder to correct from distance angles?

Plan

* Create rought sketch of whole robot
  + Ask the workshop their opinions, any imporvmenets, and how I would go about getting it made
* Test the affect of exploration
* Test the affect of the wheel size again
* Test quadratic loss function and decaying one
* Test strength of loss parameters

Do we need to measure what the motor is actually doing?

The motor sends 256 pulses oer iteration, which can we be counted on the raspberry pi? Might need additional hardware to allow counting down.

Complicated translation of current to torues.

Could assume current was near rest as acccelerations aren’t too small.

Or fast accelerations will only occur in similar states so do we need tohe extra measurements to know were here

4/2/20

Researched inner tubes for designs

6x1 ¼

<https://www.amazon.co.uk/gp/product/B0068I47LQ/ref=ask_ql_qh_dp_hza>

<https://www.ebay.co.uk/itm/Inner-Tube-6-x-1-1-4-with-a-Bent-Valve-/161890106945>

<https://www.dmehub.net/6-x-1-1-4-Standard-Inner-Tube-p/tag-162300.htm>

<https://www.aliexpress.com/item/32716034329.html>

<https://www.tyre-choice.co.uk/6-Inch-Inner-Tubes-With-TR87-Valve/>

10x2 (250-6)

<https://www.ebay.co.uk/c/1446336415>

<https://www.blackwoods.com.au/manual-handling-and-storage/wheels-tubes-tyres/pneumatic-tube-and-tyre-ehi/tube-inner-suit-250x6/p/00963670>

5inch

<https://www.tyre-choice.co.uk/5-Inch--Inner-Tubes-With-TR87-Valve/>

200x50 (too big)

<https://www.amazon.co.uk/Inner-Tube-Bent-Valve-Petrolscooter/dp/B007FPPKPS>

2.5-4

<https://www.amazon.com/dp/B002Y289W8/ref=psdc_13789601_t1_B00DT2X04Y>

<https://www.amazon.co.uk/Wingsmoto-Inner-2-50-4-250-4-2-80-4/dp/B06XKQW4BD/ref=pd_rhf_dp_p_img_1?_encoding=UTF8&psc=1&refRID=KEZ3PHF5TA79GDMQ5Z4F>

<https://www.amazon.co.uk/Martin-Wheel-250-4-Inner-Mower/dp/B00DT2X04Y>

<https://www.tyre-and-tube.co.uk/250-4-inner-tube-for-sack-trucks-garden-carts-mobility-scooters--trolleys-250-x-4-340-p.asp>

<https://www.cokertire.com/250-4-tr-87-tube.html>

8 x 1 ¼

<https://www.amazon.co.uk/gp/product/B00AHRGX0W/ref=ask_ql_qh_dp_hza>

Would work really well at the top of the unicycle, hoiwever this would shift mass upwards (probs, might not if electrons are heavy) this increases moment of inertia and would slow down rotattions. This helps in balancing near the equilbrium point as its slower. However as the unicycle doesnt know how to stay there it will probs inhibit learning as the uniclce will fall faster and reuquire larger torques to correct (less sensitive -  good?)

14x6,145/70-6

<https://www.cokertire.com/accessories/tubes/14x6-145-70-6-tr87-tube-bms.html>

600x6

<https://www.cokertire.com/accessories/tubes/a-6-600x6-tube-tr-13.html>

Phone mount

<https://www.halfords.com/technology/mobile-phone-accessories/car-phone-holders/halfords-one-touch-universal-car-mount-holder?request_type=bestseller>

Bearing

<https://www.vxb.com/6x10x3-Shielded-Bearing-Miniature-p/mr106zz-1.htm>

Need flanged

<http://www.rcbearings.com/products/mf106zz-6-x-10-x-3-flanged-bearing.html>

Questions

* Can the hole for the bearing mount be a screw hole?
* Would two holes here be enough

What sizes of top bar are possible in the range 26mm -32mm

* Stregth/ redgitity of set-up with all the bolts present
* Could the roll angle be increased by flippingthe lower gear around, shuffling the bottom motor right slightly?

24/2/20

Had to order new charger, and screws

Finished building the robot

Managed to get the rasberry pi to communicate with matlab and successfully tested the motors

This invloved reinstallling the operating system on the pi and using my desktop and a ethernet to usb converter appeared to not work

One of the motor cables broke and required ordering.

Tried running rollouts

Flywheel made loud screeching noise which needs investigating

The phone setup with rasberyy pi worked well

Needed to remeasure the anglue limits and put these into the file

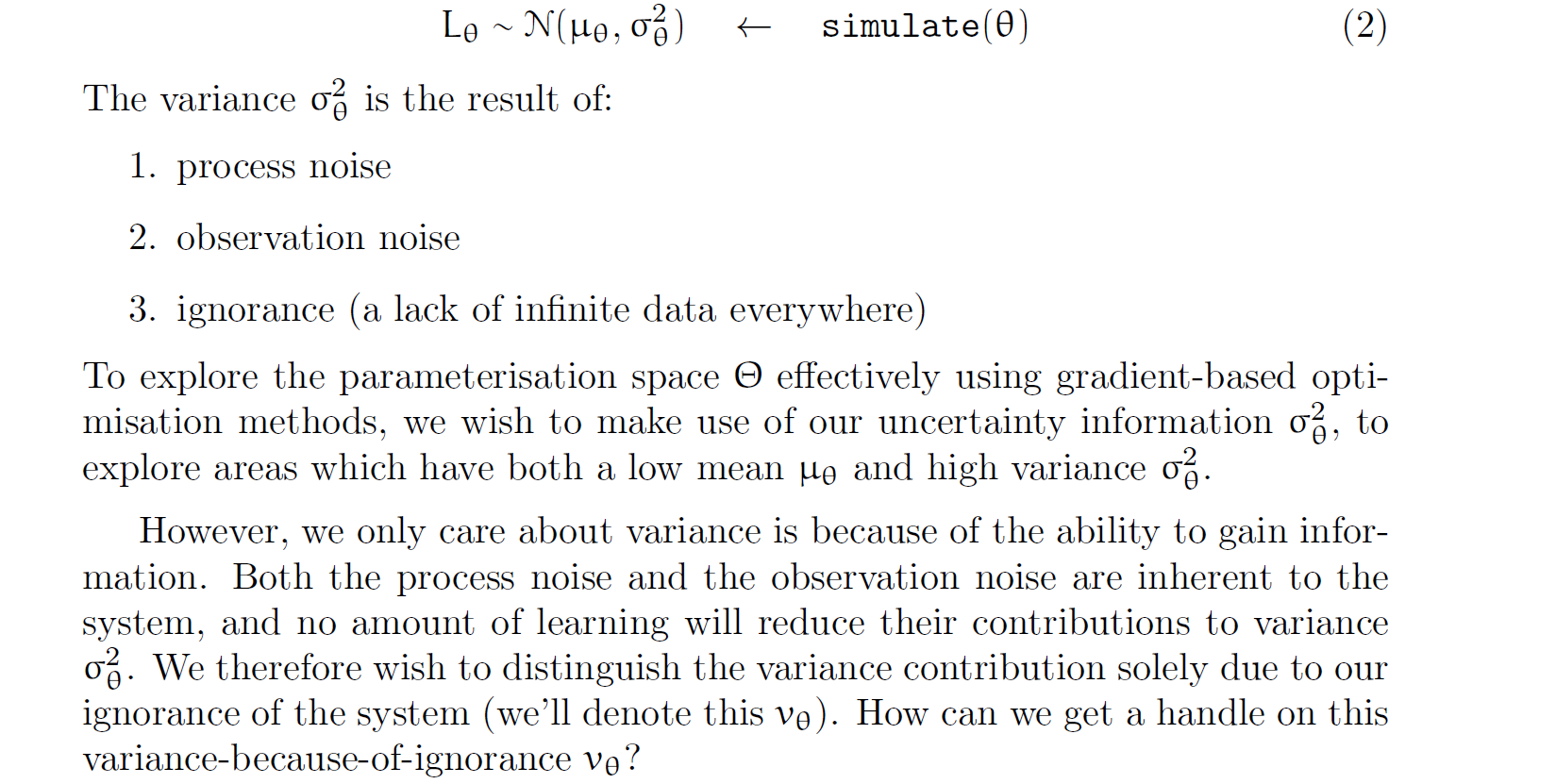
Problems were encountered on t=0.05 as very few measurments were recorded before it crashed, also if a measurment occured that was deemed ok, but the next one was out of bounds, due to the high accelerations in the crash the calcualtions thought the robot had bounced back up and had not hit the floor

* Meant that for the remainder of the time it assumed itwas balancing ok,
* This meant spikes in x/y position occured
* Could fix by projecting a step forward when working out if the constanit had been hit, in order to terminate the run correctly
* Tried on faster recoring speed,
  + Seemed to overt the problem as more steps could occur, however this wont have fixed it and will still need fixing to avoid errors
  + Noticed that 0.01 seconds from the phone wasnt possible with the extra calculations, a time of around 0.015 was the minimum
  + However the real dt was not recorded properly for plotting and storing in trajectories, it always just took the constant time set in the doit file
  + This isnt the biggest problem if its repeatable but should be fixed

Started writing a plan for my report

1/3/2020

Read estimated uncertainty in loss.

Could we fantiize about what data the policy will generate in the future and predict if this will help to improve the expected loss 

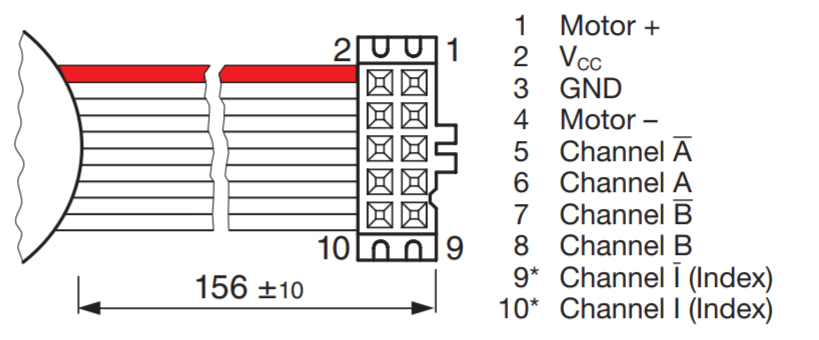
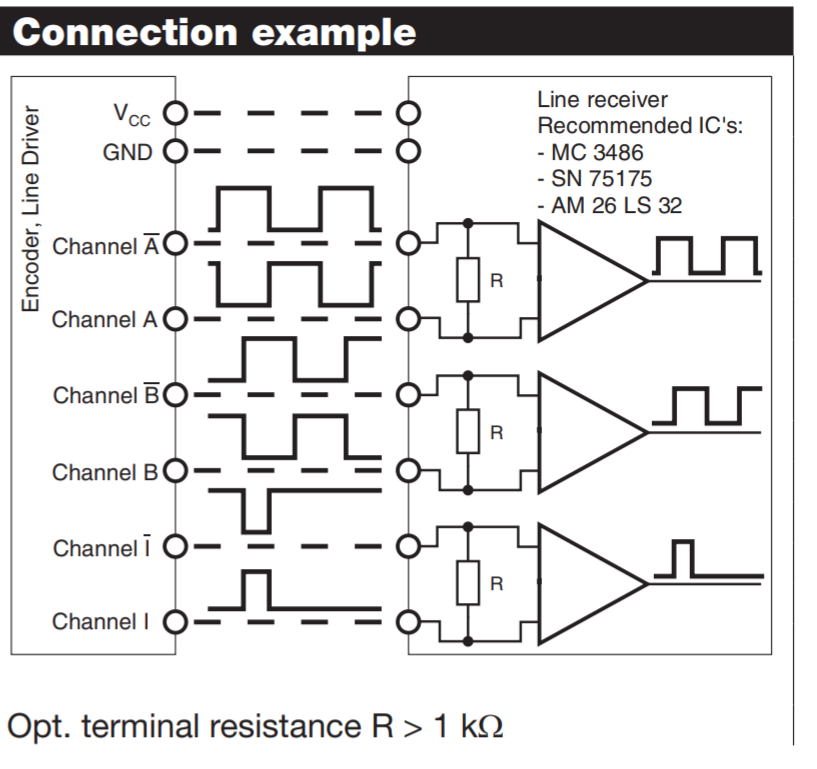
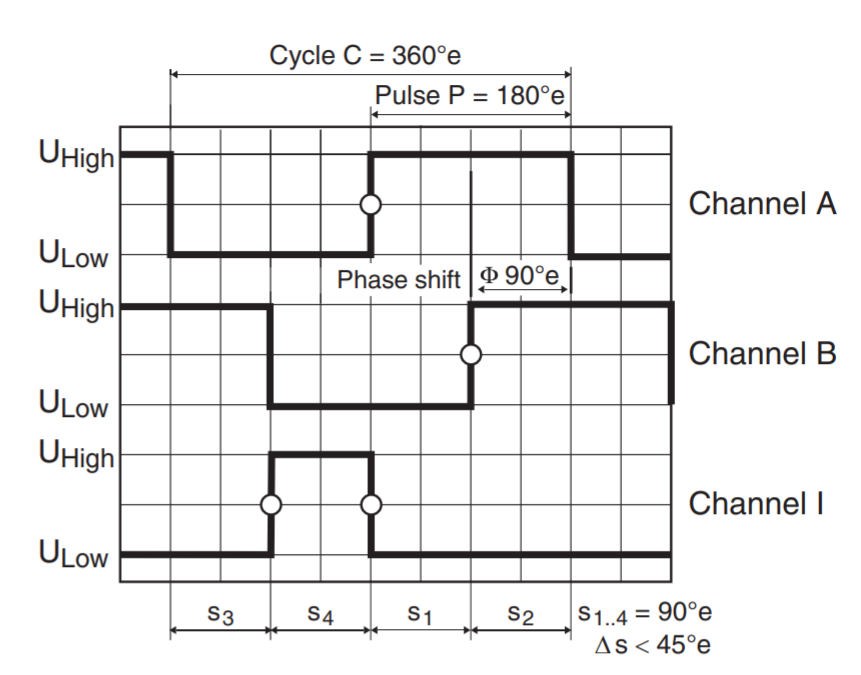
6/3/20

Whats the purpose of the latent state in a real rollout? Since we are only measuring the observed state. This was privo=iously used as the dynamics model on the oc had no noise so latent states had it added on

9/3/20

Looking at the enconder

512 counts per turn



To use the enconder to count directions both the raspberry pi and the phone would need to supply measuments.

This would have to be done in the following way

* At time 0
  + Position read from the phone
  + Action computed and send to phone
* At time t
  + Motor position read by rasbery pi after controlling the motors, possibly with a small pause
  + Motor position/count transferred back to computer
  + Phone position transferred back to computer
  + Incorporate both bits of information in position calculation

Why should we measure the positon?

* Because the unicycle position changes the torque required and therefore the current required for a given movement, the actual torque applied may not be what was desired
  + Surely this wouldn’t make a difference as the position will still be read correctly and the dynamic model can model this behaviour
* The greatest change and differiculty modelling will occur at fast angular accelerations, where the unicycle is about to hit the ground and the moment is large
  + This is a known location that doesn’t matter too much as it will probs fail here
* The rest of the motion should have smaller accerations and therefore easier to model behaviour. Meaning the motor current would be near to its resting value

What you can do is install an interrupt service routine to trigger on the gpio rising or falling. The wiringPi library supports doing it and will run up to a maximum of about 10000 triggers per second. If you need more speed you could use my fast kernel interrupt sample ([http://www.raspberrypi.org/phpBB3/viewt ... 72&t=33880](http://www.raspberrypi.org/phpBB3/viewtopic.php?f=72&t=33880)) which will run up to about 100000 triggers per second.

<https://www.raspberrypi.org/forums/viewtopic.php?f=72&t=33880>

No load speed of 6240rpm (104hz) = 53,248Hz . which is could be countable

However this will be smaller for the wheel in normal motion

This has a 1:14 reduction followed by a 16:40 reduction = reduction in speed of 35, wheel could be estimated to have a max speed of about 60rpm =2100rpm at the motor , so could expect about half the no load speed

Might require an external counter at these speeds

<https://www.youtube.com/watch?v=41ogu0UlwCc&t=35s>

Can the software count posive and negative pulses?

15/3/20

* Dismanteled the oldunicycle
* Was build in a different way to expected, with each disk etc being in two parts, required breaking all the welds in order to dismantle
* Was very weak at points
* Very hard to remove flywheel

18-19/3/20

* Having picked up the parts and shopped I started to assemble the robot
* Problems encountered
  + Waterjet parts were fit a lot tighter then designed
    - Required filing down slots to get a very snug fit
  + Motor bracket had slot the wrong size
    - Drilled to be correct
  + Motor bracket didn’t need slot in main sheet, not a problem due to the bolts
  + Assembly was quite fiddly but ended up being very ridged when complete
  + wheel was stuck on old axel, wd40 didn’t help
    - decided to just cut down the old axel
    - it ended up coming off after I finished sawing…
  + wheel was about 1mm further to one side then expected due to bolts not considered
    - cut off part of the main sheet
  + main sheet looks slightly warped
    - corrected using the bolts attaching the disks
  + some bolts were too long for the bottom disk
    - sawed them shorter
* end result
  + was very sturdy
  + looks good
  + heavier then expected… 1354g + 156g batter/pi = about 1.5kg the turntable is only about 200g
    - may be a problem for speed of falling and ability to correct tilts
    - could reduce weight by getting the handle bar made thinner
      * or maybe move this to be lower to lower com
    - cut out holes in disks and plate where not needed
    - increase mass of flywheel if found insuffient
      * or remove gearbox?
  + phone mount sturdy and allows variation
  + foam padding is very good
  + raspberry pi will need to be attached differently due to different mounting of bar then considered
    - some more holes need drilling and then it can be mounted sideways
      * the screws for this are arriving in post as they were quite specialised
  + extra hole needed for securing ends of foam with cable tie

tested out the raspberry pi/ motors

* discovered I was missing the dc cable and battery meter for the battery
  + unfortunately due to covid-19 Im unable to get these from the department
  + have ordered a new dc cable#
  + battery lukily had some charge in it
* got the motors turning when running directy from the raspberry pi
* had difficulty connecting it via matlab and ethernet cable
  + the usb – ethernet adapter used on my laptop didn’t allow reconisition of the raspberry pi
  + worked on my desktop as this doesn’t need an adapter
  + still unable to setup the co nnection
    - the support package fails when copying filels
    - currently looking for a solution

16/4/20

dynamics model has no notion of time

may not be worth it as it can open a whole set of possible problems

zeroing of the angles?

Measuring the initial orientation

Add one more variable to the learning to learn the initial offset in roll and yaw, set to be free variables with one value per trial

As a separate hyper parameter per hyperparameter

Worked very well

Doesn’t react to that in the current trial, as no online inference is done on the fly

So controller will still suffer in actual trials, as its assuming the inatial is 0.

Use the dynamics to infer what the offset was

Simple predictor using 2 or three timesteps could be effective

Compare measured offset

1.3 degrees of deviation, reasonable values

Fix the problems if theryre not needed

A wrapper to intercept the GP and add the constants

**12-27/5/2020**

Write report and conducted rollout tests

Reasarch and consider future work to be conducted

**20/5/20**

Calculate fall equations

T = mgl sin a

T = I a..

Mgl/I sina = a..

Mgl/I sina t + c\_1 = a.

Mgl/2I sina t^2 + c\_1 t + c\_2 = a

At t= 0 theta theta dot = 0

So c12 = 0

**Draft feedback**

* reference style – change to author year
* two column he found interesting but not bad
* technical descriptions are good, be more explicit about the motivationand be very clear what is preexisiting
* in conclusion maybe add in the achievements I did
* Figure captions should explain everything needed to unstand what the figure shows, the text that references the figure should show how it blends into the main story
  + so that searching less text searching needs to occur
* be more clear about which parts are simulation, and pertain to the experiment but not in the experiment itself
* sampling time and obervavtion noise maybe a combination is the pain point
  + empasise the fact that all the issues are complicated and connected, and very hard to separate the issues
* comparson between timestep is very clearly different
* more clear about why wheel size is being compared
* more about combining learning with control
* lab books
* empasise all the little steps
  + maybe present a timeline of work conducted
  + to get the flow and all the aspects brought out
* think of it more of a design project, and getting alignment between software and hardware
* maybe send over the conclusion or introduction , if wanted

test summary

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test name | Location | roll limit | seed | additional cost term | seperate cost term | runtime |  | T | wheel | cause of failure |  |  |  |  |  |  |
| 1 | desktop | 17 | 4 | TRUE | FALSE | 4hr | had 2pi | 0.05 |  |  |  |  |  |  |  |  |
| 14 | desktop | 17 | 4 | TRUE | FALSE | 4hr | had 2pi | 0.05 |  | balanced for 2.5 but ran away into the distance |  |  |  |  |  |  |
| 15 | desktop | 17 | 4 | TRUE | FALSE |  | had 2pi | 0.05 |  | balanced for 2.5 but ran away into the distance |  |  |  |  |  |  |
| 16 | desktop | 17 | 4 | TRUE | FALSE |  | 4pi | 0.05 |  | went in circle but hit roll limit eventually |  |  |  |  |  |  |
| 17 |  | 17 | 4 | TRUE | FALSE |  | 4pi | 0.05 |  | went in circle but hit roll limit eventually |  |  |  |  |  |  |
| 18 |  | 17 | 4 | TRUE | FALSE |  | 4pi | 0.05 |  | went in circle but hit roll limit eventually |  |  |  |  |  |  |
| 24 |  | 17 | 4 | TRUE | FALSE |  | 4pi | 0.05 |  | went in circle but hit roll limit eventually |  |  |  |  |  |  |
| 52 |  | 17 | 5 | TRUE | FALSE |  | 4pi | 0.05 |  | a lot worse , never really got past inital badess |  |  |  |  |  |  |
| 54 |  | 17 | 7 (6 broekn) | TRUE | FALSE |  | 4pi | 0.05 |  |  |  |  |  |  |  |  |
| 55 |  | 17 | 8 | TRUE | FALSE |  | 4pi | 0.05 |  | much better graphs then the rest, I could see this working given a few more iterations |  |  |  |  |  |  |
| 56 |  | 17 | 9 | TRUE | FALSE |  | 4pi | 0.05 |  | " | annoyingly i stopped it at trial no 39 |  |  |  |  |  |
| 57 |  |  | 10 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 60 |  | 42 | 6 |  |  | 7hr15 |  | 0.05 |  | (seed 6 didnt work on t=0.4) |  |  |  |  |  |  |
| 61 |  | 42 | 5 |  |  | 1.5hr |  | 0.04 |  | seemed to be trying to learn to stay on the spot rater then wandering offf, had a time of 0.4s not 0.5s . different graphs to all other trieals |  |  |  |  |  |  |
|  |  | 42 | 7 |  |  |  |  |  |  | didnt work on 0.04 |  |  |  |  |  |  |
| 64 |  | 42 | 8 |  |  | 31.5hrs |  | 0.04 |  | on 0.04 | still not that great at the end, didnt show much improvment for the last policy iterating but ewach one took 2-4hrs as there was so much data availabele |  |  |  |  |  |
| 66 |  | 42 | 8 |  |  | 1hr |  | 0.05 |  | was still pretty shit after 40 iterations |  |  |  |  |  |  |
| 67 |  | 42 | 9 |  |  | 40min |  | 0.05 |  | once again learnt noting |  |  |  |  |  |  |
|  |  | 42 | 10 |  |  |  |  | 0.05 |  | didnt work failed at 18 iterations |  |  |  |  |  |  |
| 69 |  | 42 | 11 |  |  | 4hr20 |  | 0.05 |  |  |  |  |  |  |  |  |
| 70 |  | 42 | 12 |  |  | 9hr20 |  | 0.05 |  | decent |  |  |  |  |  |  |
| 71 |  | 42 | 13 |  |  | 8hr |  | 0.05 |  | trailed off |  |  |  |  |  |  |
|  |  | 47 | 4 |  |  |  |  | 0.05 |  | didit run |  |  |  |  |  |  |
|  |  |  | 5 |  |  |  |  |  |  | failed instantly |  |  |  |  |  |  |
| 74 |  | 47 | 6 |  |  | 8hr |  | 0.05 |  | ok |  |  |  |  |  |  |
| 75 |  | 47 | 7 |  |  |  |  | 0.05 |  | different error | Error using chol Matrix must be positive definite. Error in gpa/pre\_full (line 187) L = chol(K + exp(2\*self.hyp(i).n)\*eye(n))'; Error in gpa/pre (line 168) pre\_full(self); % full GP initialisation Error in gpa/train (line 128) self.pre(); % do possible pre-computations Error in doitnlds (line 204) dyn.train(to\_data\_struct(trajectories),1:D,dyno); |  |  |  |  |  |
| 76 |  | **47** | **8** |  |  |  |  | 0.05 |  | was still very bad at the end, but seemd to be trying to learn how to stay on the spot instead of just stay upright, the loss prediction of the final run had no variance after 1second? is this a fault... very interesting state predictions tho (right) |  |  |  |  |  |  |
| 77 |  | 47 | 9 |  |  |  |  | 0.05 |  | alright |  |  |  |  |  |  |
| 78 |  | 47 | 10 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 79 |  | 47 | 11 |  |  |  |  | 0.05 |  | trial 26 was pretty interesting and it did a lot of loops |  |  |  |  |  |  |
| 80 |  | 47 | 12 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 81 |  | 53 | 4 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
|  |  | 53 | 5 |  |  |  |  |  |  | didnt worek |  |  |  |  |  |  |
| 83 |  | 53 | 6 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 84 |  | 53 | 7 |  |  |  |  | 0.05 |  | another one that learns to balance not move |  |  |  |  |  |  |
|  |  | 53 | 8 |  |  |  |  | 0.05 |  | failed quickly |  |  |  |  |  |  |
|  |  | 53 | 9 |  |  |  |  | 0.05 |  | failed quickly |  |  |  |  |  |  |
| 87 |  | 53 | 10 |  |  |  |  | 0.05 |  | another one that learns to balance not move |  |  |  |  |  |  |
| 88 |  | 53 | 11 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 89 |  | 53 | 12 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 90 |  | 53 | 4 |  |  |  |  | 0.05 | 0.04 |  |  |  |  |  |  |  |
| 92 |  | 53 | 6 |  |  |  |  | 0.05 | 0.04 |  |  |  |  |  |  |  |
| 93 |  | 53 | 7 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
|  |  |  | 8 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
|  |  |  | 9 |  |  |  |  | 0.05 |  | failed in a different way |  |  |  |  |  |  |
| 96 |  | 53 | 10 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 98 |  | 53 | 11 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 99 |  |  | 12 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 100 |  |  | 13 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 101 |  |  | 14 |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 103 |  |  | 5 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 104 |  |  | 6 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 105 |  |  | 7 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 106 |  |  | 8 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 107 |  |  | 9 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 108 |  |  | 10 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 109 |  |  | 11 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 110 |  |  | 12 |  |  |  |  | 0.05 | 0.4 |  |  |  |  |  |  |  |
| 111 |  | 42 | 10 |  |  |  |  | 0.04 | 0.4 |  |  |  |  |  |  |  |
| 112 |  | 42 | 11 |  |  |  |  | 0.04 |  | stopped at 34 runs because it was taking ages |  |  |  |  |  |  |
| 118 | real accidentally had DT as 10ms tho and I didn't realise, which could explain the shit constraint detection. also crashes the program if no sensor data found... could be good to fix this to just pause or somethibg |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 119 |  |  |  |  |  |  |  | 0.05 |  |  |  |  |  |  |  |  |
| 120 |  |  |  |  |  |  |  | 0.045 |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |