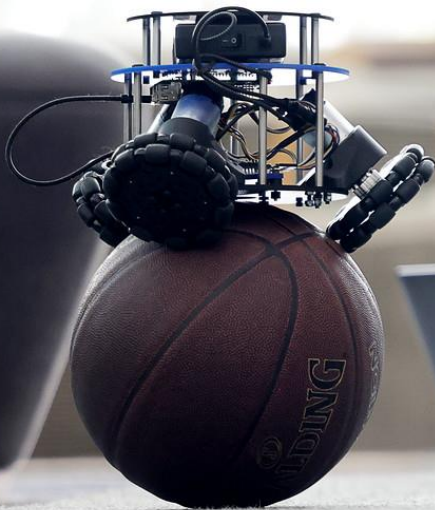


Robotics 311 : How to build robots and make them move

Prof. Elliott Rouse

GSI Yves Nazon MS

Fall 2022



ROB 311 – Lab 9

- Today we will
 - Modify Python program for ball rolling
 - Collect data of ball rolling with upside down ball-bot
 - Confirm kinematics relationships

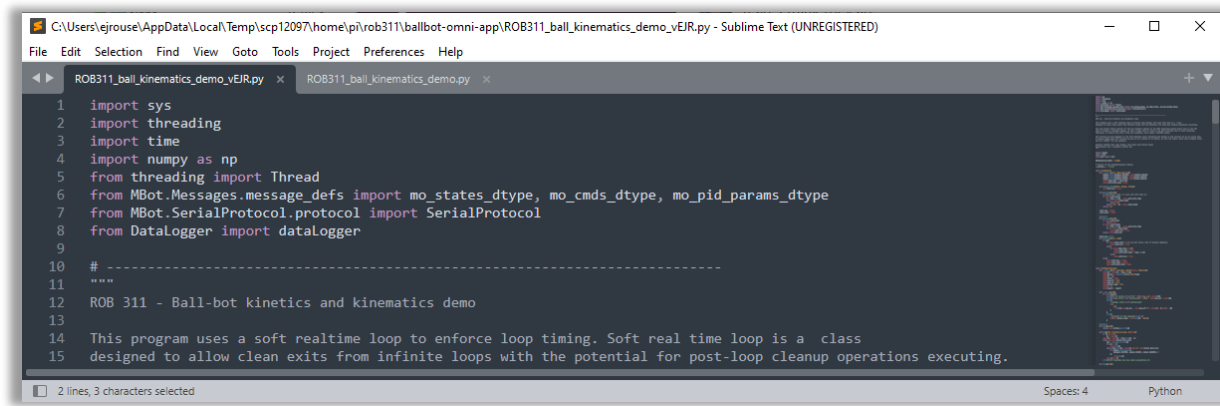
Kinematics

- One thing that is important is understanding the motion of your ball-bot
- To facilitate this understanding, today we will verify our ability to roll the basketball in a desired direction
- You will edit a provided Python script
- Collect data of the ball-bot rolling the ball (upside down)
- Analyze the data in MATLAB to investigate kinematics
- Confirm your layout and wiring are done correctly



Files Provided

- You will need two software files from Canvas under Lab 9
- Python script to run your ball-bot / collect data
 - ROB311_ball-bot_kinematics_demo.py



A screenshot of a Sublime Text editor window showing a Python script named `ROB311_ball_kinematics_demo_vEJR.py`. The script includes imports for `sys`, `threading`, `time`, `numpy`, and various MBot-related modules. It contains a comment block describing the program as a demo for ball-bot kinetics and kinematics, and a note about a soft realtime loop.

```
1 import sys
2 import threading
3 import time
4 import numpy as np
5 from threading import Thread
6 from MBot.Messages.message_defs import mo_states_dtype, mo_cmds_dtype, mo_pid_params_dtype
7 from MBot.SerialProtocol.protocol import SerialProtocol
8 from DataLogger import dataLogger
9
10 # -----
11 """
12 ROB 311 - Ball-bot kinetics and kinematics demo
13
14 This program uses a soft realtime loop to enforce loop timing. Soft real time loop is a class
15 designed to allow clean exits from infinite loops with the potential for post-loop cleanup operations executing.
```

- MATLAB script to process data



A screenshot of a MATLAB Editor window showing a script named `ROB311_ball_bot_data_analysis.m`. The script includes comments about the data being processed (from a ball-bot) and the kinematic analysis. It also includes a list of parameters to be used in the analysis, such as `alpha`, `beta`, `Rk`, and `Rw`.

```
1 % ROB 311, Fall 2022
2 % Data importing / plotting from ball-bot
3 % Kinematic analysis of an upside down ball-bot
4 %
5 % Prof. Elliott Rouse
6 % University of Michigan
7 %
8 % =====
9
10 clear
11 clc
12 %close all
13
14 alpha = pi/4;
15 beta = pi/2;
16 Rk = 0.11925;
17 Rw = 0.04778;
18
19 % Input file name - this will have to be updated with the name of the trial
```

Ball-Bot Code Overview

- The Python file applies a specified torque and sends power to the motors
- It also records many relevant variables for analysis
- There are pre-written functions in the top of the script
- These function can be used for the code you write
 - Computation of motor torques from x , y , and z torques (`compute_motor_torques`)
 - Computation of ball rotation from wheel rotation (`compute_phi`)
- Overview of script

Ball-Bot Code Overview

- In this exercise, we will do some conversions of motor rotation (psi) from radians to counts
- This isn't necessary, but gives you a feel for doing the conversions—future versions of our code will stay in radians
- You need to make three changes to the Python script
- 1. add your code convert psi into encoder counts
 - You will need to know the encoder conversion
- 2. Change torque components to change ball rolling direction

```
273 # -----
274
275 def rad_2_counts(psi_rad):
276     """
277     Parameters:
278     -----
279     psi_rad: Wheel rotation in radians
280
281     Returns:
282     -----
283     psi_counts: Wheel rotation in counts
284     """
285     # YOUR
286     # CODE
287     # GOES
288     # HERE
289
290     psi_counts = psi_rad
291
292     return psi_counts
293
```

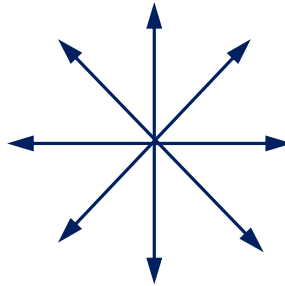
```
360 # -----
361 # Compute motor torques (T1, T2, and T3) with Tx, Ty, and Tz
362
363 # Beginning with torque rolling toward positive y-axis
364 # CHANGE THESE TO ADJUST THE ROLLING DIRECTION OF YOUR BALL-BOT
365
366 Tx = 2
367 Ty = 0
368 Tz = 0
369
370 # -----
371
```


Ball-Bot Code Overview

- Finally, you need to convert wheel rotation into ball motion
- The function `compute_phi` is expecting psi in counts (not rad)
- Add a line to compute phi, adding any conversions to psi as needed

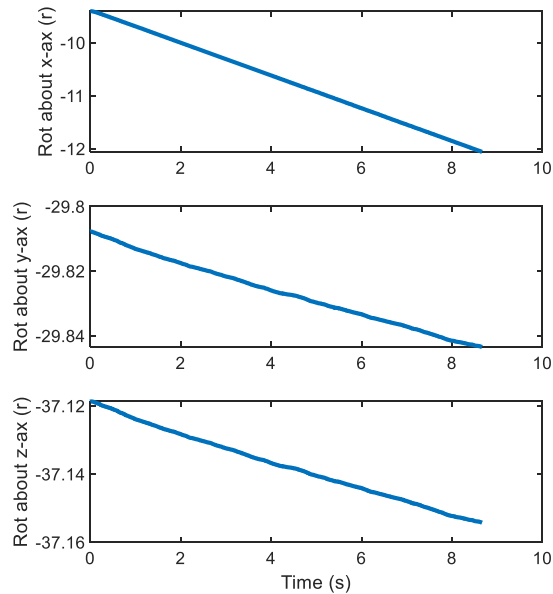
```
377 # -----  
378 # Compute ball rotation (phi) with psi_1, psi_2, and psi_3  
379  
380 # YOUR  
381 # CODE  
382 # GOES  
383 # HERE  
384  
385 # -----  
386
```

- Once this is completed, you should collect data with many torque directions around a circle

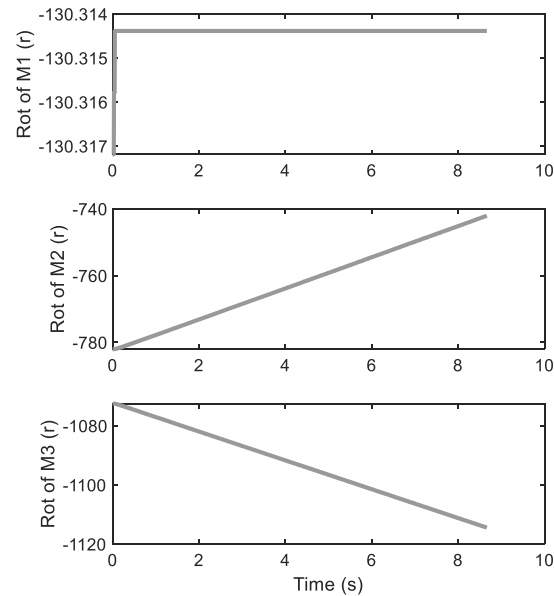


MATLAB Analysis Overview

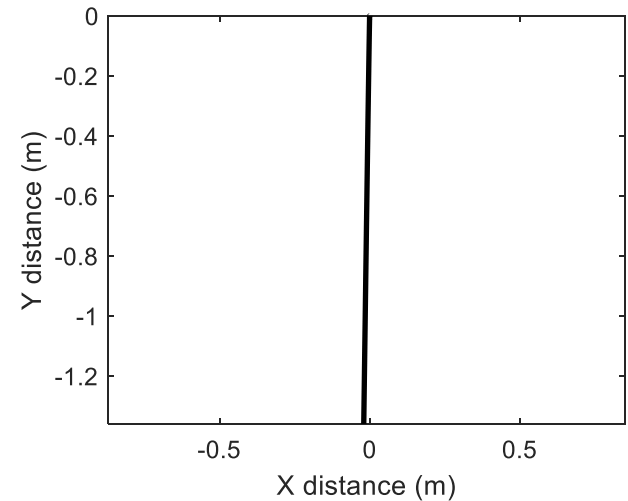
- The MATLAB file provided is similar to last week
- It reads in the saved data and plots it for you
- It creates three plots



Phi / ball rotation
shown across all
planes



Psi / wheel
rotation shown
across all planes



x-y planar motion
of the ball

Your Goal

- Your goal is to collect trials for torque around a circle
- Plot the x-y motion of the ball on a single plot
- On the same plot, show vectors that describe the torque directions (components of T_x and T_y)
- How similar / different are the torque directions and motion directions?