Design and Implementation of an Adaptive Controller for Differential-Drive Robots in Artificial Gravity Environments

Jadyn Chowdhury[1]

DP1-AE 353: Aerospace Control Systems,, Urbana, IL, 61820, United States of America

This paper details an extensive investigation into the design and implementation of a closed-loop control system for a differential-drive robot operating in a rotating space station influenced by artificial gravity. The research adopts a holistic approach, integrating theoretical analysis, symbolic computations, and pragmatic application to master the robot's dynamics. The aim is to ensure stable, efficient navigation and maneuverability of the robot, overcoming the challenges posed by the unique environment of artificial gravity. The project encompasses developing a user-centric controller, conducting rigorous simulations, and validating performance under diverse conditions, thereby achieving effective control and stabilization of the robot's movement in space station settings.

I. Nomenclature

$e_{lateral}$	=	lateral error (m)
$e_{heading}$	=	heading error (rad)
v	=	forward speed (ms^{-1})
ω	=	turning rate ($rads^{-1}$)
θ	=	pitch angle (rad)
$\dot{\theta}$	=	pitch rate ($rads^{-1}$)
τ_R	=	right wheel torque (Nm)
τ_L	=	left wheel torque (Nm)
dx	=	length (m)
dy	=	width (m)
dz	=	height (m)
m_b	=	chassis mass (kg)
m_w	=	wheel mass (kg)
r	=	radius of each wheel (m)
h	=	distance between axle and COM of chassis (m)
h_w	=	width of each wheel (m)

II. Introduction

Navigating the complexities of artificial gravity in space stations poses significant challenges for robotic systems. This paper addresses the development of a specialized closed-loop control system for a differential-drive robot, tailored to operate effectively in such environments. Our focus is on overcoming the distinct dynamics introduced by artificial gravity, ensuring both stability and manoeuvrability. The research blends theoretical analysis of robot dynamics with practical control design, utilizing simulation for validation. The ultimate goal is to demonstrate that differential-drive robots, with an appropriately designed control system, can function efficiently in space station settings. This work paves the way for their expanded use in space exploration and habitation.

[1] Aerospace Engineering and Computer Science Double Major, Grainger School of Engineering at UIUC

III. Theory and Model

A. Introduction to Control Theory

Control theory is integral to robotic systems, particularly in dynamic environments. It involves managing system behaviours through feedback mechanisms, crucial for a differential-drive robot operating in a space station's artificial gravity. Operating under artificial gravity affects traction and stability. The control system must effectively address these unique dynamics.

B. Differential-Drive Robot Dynamics

Differential-drive robots are a fundamental design in robotics, primarily due to their simple yet effective mechanism for movement and turning. The model robot will utilise two parallel wheels mounted on a common axis, with each wheel driven independently. The ability to vary wheel speeds allows for a wide range of movements. The robot can move forward, backward, and rotate by varying the relative speeds of the two wheels. The distance between the wheels $\left(a = \frac{0.7}{2} m\right)$ plays a critical role in determining the turning radius and manoeuvrability.

The linear velocity V of the robot is the average of the velocities of the two wheels,

$$V = \frac{V_1 + V_2}{2}$$

The angular velocity ω is determined by the difference in wheel velocities divided by the distance between the wheels,

$$\omega = \frac{V_R - V_L}{2a}$$

The robot's position in a plane can be described by coordinates (x, y) and its orientation by an angle θ. The kinematic model can be represented by a set of differential equations that describe how the robot's position and orientation change over time,

$$\dot{x} = V\cos(\theta)$$
$$\dot{y} = V\sin(\theta)$$
$$\dot{\theta} = \omega$$

By integrating these differential equations over time, one can determine the robot's trajectory and orientation based on the wheel velocities.

Understanding these dynamics informs the design of the control system, ensuring it compensates for the unique challenges posed by the artificial gravity environment.

C. Mathematical Modelling of the Robot's Dynamics

Firstly, the robots physical parameters are defined. These being its length, width, height, axle COM distance, and mass specifications. Relative moments of inertia are calculated defined as follows,
MOI around the x-axis,

$$J_{b_x} = \frac{m_b}{12}(dy^2 + dz^2)$$

MOI around the y-axis,

$$J_{b_y} = \frac{m_b}{12}(dx^2 + dz^2)$$

MOI around the z-axis,

$$J_{b_z} = \frac{m_b}{12}(dx^2 + dy^2)$$

Rotational inertia about the wheel's axis,

$$J_w = \frac{m_w}{2} r^2$$

Translational inertia representing the wheel's resistance to motion in the plane perpendicular to the wheel's rotation,

$$J_{w_t} = \frac{m_w}{12}(3r^2 + h_w^2)$$

MOI about the x-axis, including the chassis and the wheels' translational inertia,

$$J_x = J_{b_x} + 2J_{w_t}$$

MOI about the y-axis, solely from the chassis as the wheels do not contribute to this axis,

$$J_y = J_{b_y}$$

MOI about the z-axis, including the chassis and the wheels' translational inertia,

$$J_z = J_{b_z} + 2J_{w_t}$$

Next the acceleration due to the rotating station that acts as local gravity is calculated as,

$$g_{station} = v_{station}^2 r_{station}$$

The inertia matrix (M) represents how the robot's mass is distributed and its resistance to changes in motion and is given by,

$$M = \begin{bmatrix} \frac{2J_w}{r^2} + m & 0 & hm_b\cos(\theta) \\ 0 & \frac{2J_w a^2}{r^2} + J_z\cos^2(\theta) + 2a^2mw + (J_x + h^2m_b)\sin^2(\theta) & 0 \\ hm_b\cos(\theta) & 0 & J_y h^2 m_b \end{bmatrix}$$

The external forces/torques matrix (N) accounts for external forces acting on the robot, such as gravity and interaction forces and is given by,

$$N = \begin{bmatrix} hm_b\left(\dot{\theta}^2 + \omega^2\right)\sin(\theta) \\ -hm_b v\omega\sin(\theta) + \dot{\theta}\omega(-2J_x + 2J_z - 2h^2m)\sin(\theta)\cos(\theta)\ ghm_b\sin(\theta) + \omega^2(J_x - J_z + h^2m_b)\sin(\theta)\cos(\theta) \end{bmatrix}$$

The control input matrix (R) relates the control inputs (wheel torques) to their effect on the robot's motion,

$$R = \begin{bmatrix} \frac{1}{r} & \frac{1}{r} \\ -\frac{a}{r} & \frac{a}{r} \\ -1 & -1 \end{bmatrix}$$

The dynamics equation (f): Derived from the robot's physical parameters, representing its motion in response to external forces and control inputs,

$$\begin{bmatrix} \frac{r\left(Jyh\left(hmbr\left(\dot{\theta}^2 + w^2\right)\sin(\theta) + \tau_L + \tau_R\right) - r\left(ghmb\sin(\theta) - \tau_L - \tau_R + \frac{w^2(Jx - Jz + h^2mb)\sin(2\theta)}{2}\right)\cos(\theta)\right)}{h(2JwJy + Jymr^2 - mbr^2\cos^2(\theta))} \\ -\frac{r\left(a\tau_L - a\tau_R + rw\left(hmbv + 2\dot{\theta}(Jx - Jz + h^2m)\cos(\theta)\right)\sin(\theta)\right)}{2Jwa^2 + Jxr^2\sin^2(\theta) + Jzr^2\cos^2(\theta) + 2a^2mwr^2 + h^2mbr^2\sin^2(\theta)} \frac{-hmbr\left(hmbr\left(\dot{\theta}^2 + w^2\right)\sin(\theta) + \tau_L + \tau_R\right)\cos(\theta) + (2Jw + mr^2)\left(ghmb\sin(\theta) - \tau_L - \tau_R + \frac{w^2(Jx - Jz + h^2mb)\sin(2\theta)}{2}\right)}{h^2mb(2JwJy + Jymr^2 - mbr^2\cos^2(\theta))} \end{bmatrix}$$

Now the system control can be numerically solved using LQR[2] control theory.

The system matrix, A, represents the linearized system dynamics calculated as the Jacobian of the dynamics function, f, with respect to the state variables,

$$[A = \frac{\partial f}{\partial[e_l, e_h, v, w, \theta, z]}]$$

Evaluated at the equilibrium point.

[2] Linear Quadratic Regulator

The input matrix, B, represents the effect of control inputs on the system's state. It is defined as the Jacobian of f with respect to the control inputs,

$$[B = \frac{\partial f}{\partial[\tau_R, \tau_L]}]$$

Evaluated at the same equilibrium point as A.

The state cost matrix, Q, is a positive-definite matrix in LQR for weighing state errors in the cost function. It is predefined.

$$Q = \begin{bmatrix} q_1 & 0 & \cdots & 0 \\ 0 & q_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q_n \end{bmatrix}$$

Larger values in Q increase the penalty for deviations in corresponding state variables.

The control effort cost matrix, R, is a positive-definite matrix in LQR for weighing control efforts and is also predefined.

$$R = \begin{bmatrix} r_1 & 0 \\ 0 & r_2 \end{bmatrix}$$

Larger values in R increase the penalty for using control inputs.

The feedback gain matrix, K, determines the control law. Calculated from the CARE solution where,

$$[K = R^{-1}B^T P]$$

P being the solution to the continuous-time algebraic Riccati equation (CARE) given by,

$$[A^T P + PA - PBR^{-1}B^T P + Q = 0]$$

The Controllability Matrix, W, is used to check system controllability. Constructed by concatenating products of powers of A and B,

$$[W = [B, AB, A^2B, \dots, A^{n-1}B]]$$

The system is controllable if W has full rank, equal to the number of state variables.

These matrices are fundamental in the analysis and design of the control system.

IV. Experimental Methods

A. Requirement and Verification 1

The robot should stabilise to $\pm 0.1m$ around its centre line within 13 seconds with initial angle defined as $-\frac{\pi}{14} \leq \vartheta \leq \frac{\pi}{14}$.

PyBullet will be used to simulate the robot,

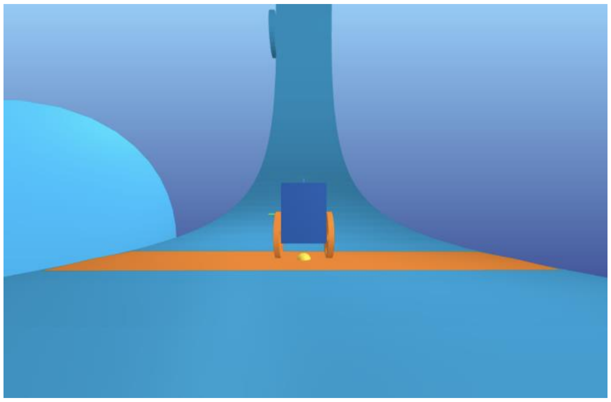


Fig. 1 PyBullet simulation

The data generated by this simulation will be imported into a Jupyter Notebook for analysis with Python. From an initial facing angle, the lateral error will be found at each time step. The time for lateral error over all time steps will be reported and graphed. If the time for the error line to become linear is less than 5, the requirement is met for that ϑ. The simulation will be repeated over a range of ϑ that can confidently be used to verify the requirement.

This is a necessary requirement to deduce the effectiveness of control system to stabilise in a reasonable time frame, given that changes should be adapted to quickly in real-word environments where time could be critical.

B. Requirement and Verification 2

The robot should not oscillate significantly while trying to reach the stabilisation point with initial angle defined as $-\frac{\pi}{14} \leq \vartheta \leq \frac{\pi}{14}$ where a significant oscillation is defined as the robot deviating from the centre line in alternating deviations exceeding 1m.

PyBullet will be used to simulate the robot. The data generated by this simulation will be imported into a Jupyter Notebook for analysis with Python. From an initial facing angle, a (x,y) coordinate will be found at each time step using the negative of the lateral error as the x coordinate and the product of the velocity and time for the y coordinate. These will be reported and graphed to produce the trajectory of the robot. If the robot does not deviate around the centre line by more than 1m after intersecting it, the requirement is met for that ϑ. The simulation will be repeated over a range of ϑ that can confidently be used to verify the requirement.

This is a necessary requirement as the control system should be precise and use minimal significant corrections since real-world environments have unpredictable landscapes that could be more sensitive to even small deviations from expected paths.

C. Limitations

Different initial conditions will be applied to examine how certain environmental conditions impact the performance of the control system. These include station gravity, robot speed, ground pitch, control effort cost, state cost and masses. Using the verification processes mentioned above, the impact these changes have can be compared.

These requirements will be experimented in an environment that does not include obstacles as they hinder the actual verification since the robot may hit one and fall. In addition, each experiment will be ran 3 times to confirm if variation exists inherently within the system. However, verifying these requirements prove that the robot can be controlled to avoid obstacles by moving the centre line albeit with specific limitations.

V. Results and Discussion

Assuming initial conditions,

Variable	**Value**
$\boldsymbol{dx}$	$0.4m$
$\boldsymbol{dy}$	$0.6m$
$\boldsymbol{dz}$	$0.8m$
$\boldsymbol{h}$	$0.3m$
$\boldsymbol{a}$	$0.35m$
$\boldsymbol{m_b}$	$12kg$
$\boldsymbol{r}$	$0.325m$
$\boldsymbol{h_w}$	$0.075m$
$\boldsymbol{m_w}$	$1.2kg$
$\boldsymbol{v_{station}}$	$-0.5ms^{-1}$
$\boldsymbol{r_{station}}$	$20m$
$\boldsymbol{v_{inital}}$	$3ms^{-1}$
$\boldsymbol{Q}$	$\begin{bmatrix} 10.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 4.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 2.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 5.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 4.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 2.00 \end{bmatrix}$
$\boldsymbol{R}$	$\begin{bmatrix} 15.00 & 0.00 \\ 0.00 & 15.00 \end{bmatrix}$

Table 1: Initial assumed conditions

At the following equilibrium values,

Variable	**Value**
$\mathrm{e}_{\mathrm{l}_{\mathrm{eq}}}$	$0\ m$
$e_{h_{eq}}$	$0\ rad$
v_{eq}	$5\ ms^{-1}$
ω_{eq}	$0\ rads^{-1}$
θ_{eq}	$0\ rad$
z_{eq}	$0\ rads^{-1}$
$\tau_{R_{eq}}$	$0\ Nm$
$\tau_{L_{eq}}$	$0\ Nm$

Table 2: Equilibrium values

Which yields the following calculated factors,

Control Element	**Matrix**
K	$\left[\begin{matrix} 0.57735027 & 3.24950345 & -0.25819889 & 1.80582566 & -10.59783396 & -0.58619606 \\ -0.57735027 & -3.24950345 & -0.25819889 & -1.80582566 & -10.59783396 & -0.58619606 \end{matrix}\right.$
W	$\left[\begin{matrix} 0.00 & 0.00 & 0.00 & 0.00 & 5.25 & -5.25 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 1.05 & -1.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 12.07 & 12.07 & 0.00 & 0.00 & 6432.51 & 6432.51 & 0.00 & 0.00 & 3484278.55 & 3484278.55 & 0.00 & 0.00 \\ 1.05 & -1.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & -51.46 & -51.46 & 0.00 & 0.00 & -27874.23 & -27874.23 & 0.00 & 0.00 & -15098540.40 & -15098540.4 \\ -51.46 & -51.46 & 0.00 & 0.00 & -27874.23 & -27874.23 & 0.00 & 0.00 & -15098540.40 & -15098540.40 & 0.00 & 0.00 \end{matrix}\right.$

Table 3: Calculated factors

The system is therefore controllable since the rank of W is 6 which is equal to the number of variables.

A. Stabilisation Time

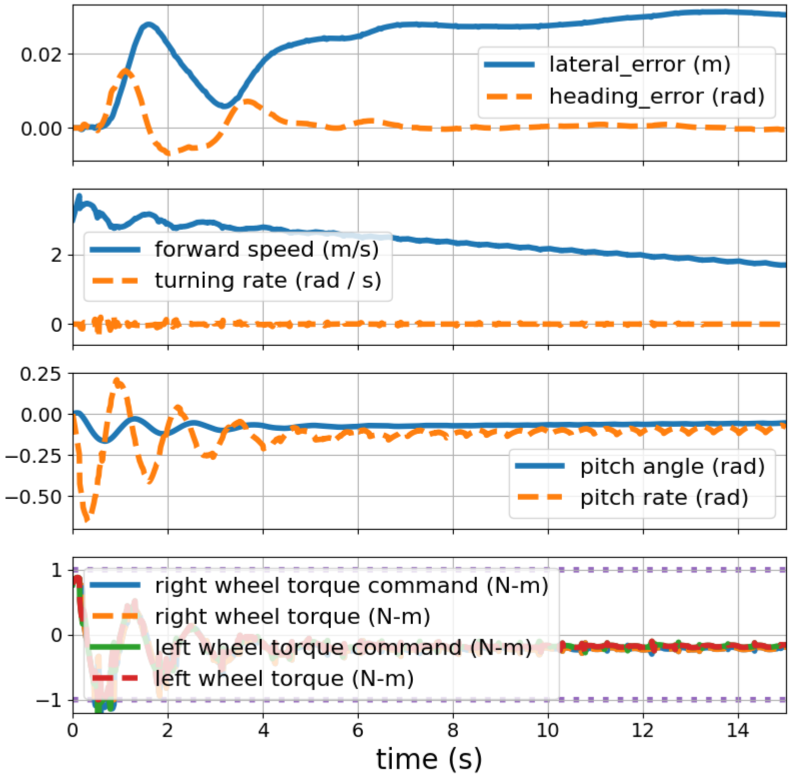
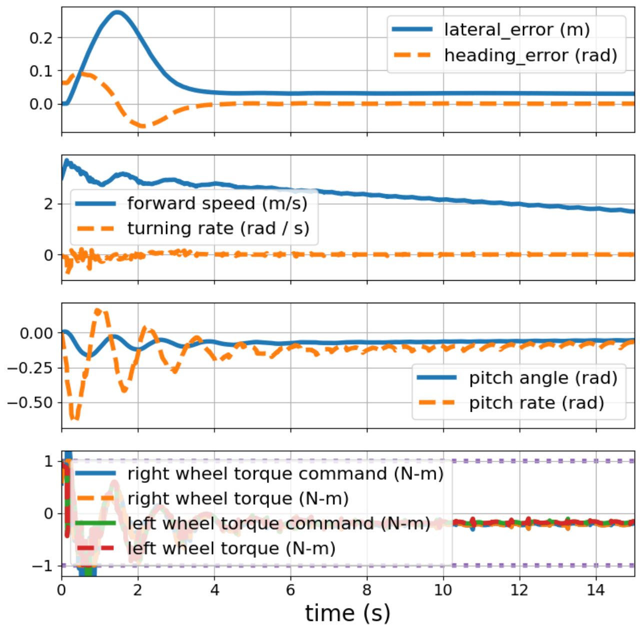


Fig. 2 Variable plot at ϑ = 0

The system stabilises at 7 seconds around 0.02m from the centre line.



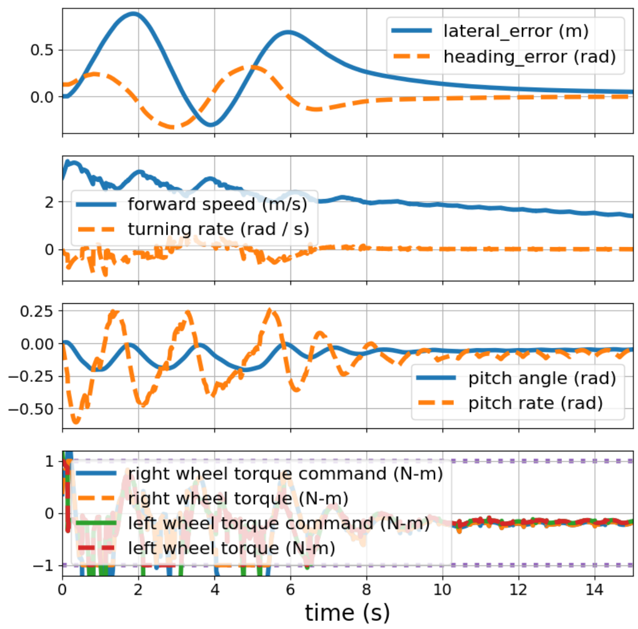


Fig. 4 Variable plot at $\boldsymbol{\vartheta} = \frac{\boldsymbol{\pi}}{\mathbf{50}}$

Fig. 3 Variable plot at $\boldsymbol{\vartheta} = \frac{\boldsymbol{\pi}}{\mathbf{25}}$

At $\boldsymbol{\vartheta} = \frac{\boldsymbol{\pi}}{\mathbf{50}}$, the system stabilises at 0.33m from the centre line after 4 seconds. At $\boldsymbol{\vartheta} = \frac{\pi}{25}$ the system stabilizes after 13 seconds at 0.1m from the centre line.

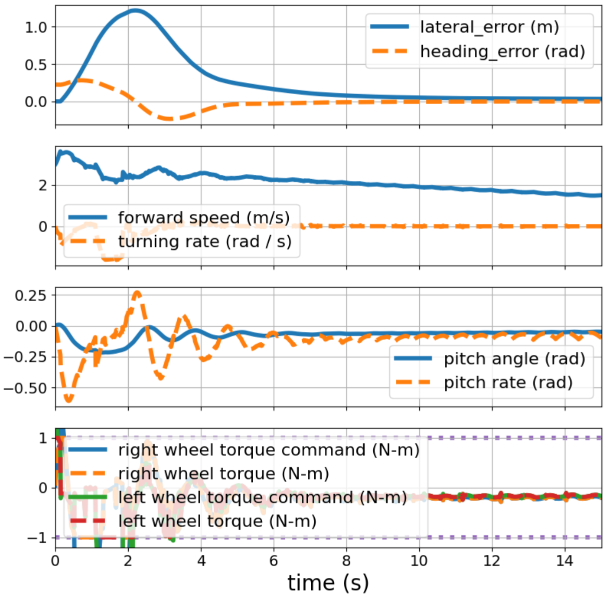


Fig. 6 Variable plot at $\boldsymbol{\vartheta} = \frac{\pi}{14}$

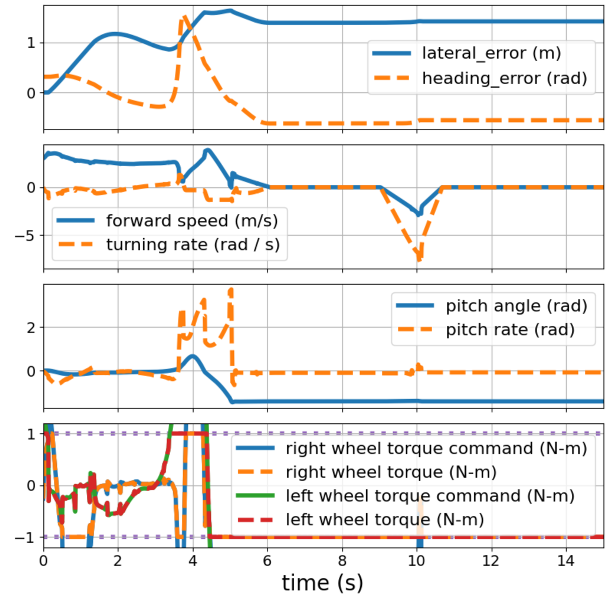


Fig. 5 Variable plot at $\boldsymbol{\vartheta} = \frac{\pi}{10}$

At $\boldsymbol{\vartheta} = \frac{\pi}{14}$ the system stabilizes at 0.05m from the centre line after 10 seconds. At $\boldsymbol{\vartheta} = \frac{\pi}{10}$ the system stabilizes at 1.5m from the centre line after 6 seconds, however a snapshot from the simulation makes it evident that robot fell over.

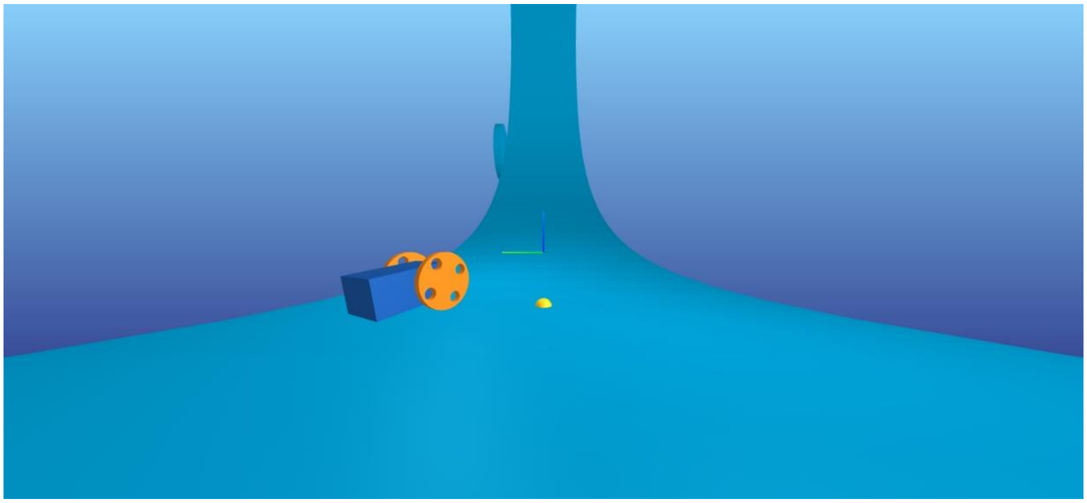


Fig. 7 Capture of end of simulation at $\boldsymbol{\vartheta} = \frac{\pi}{10}$

The system is shown to be symmetric,

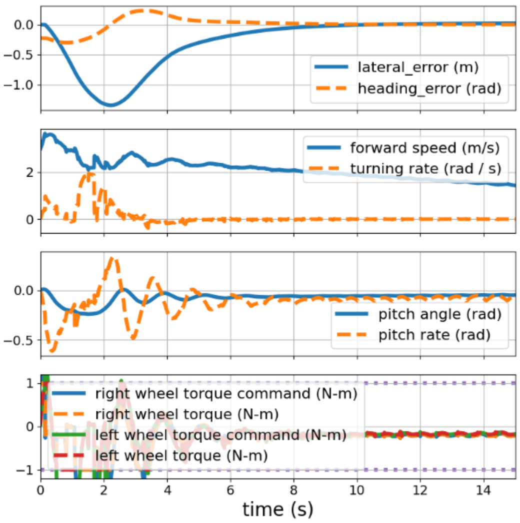


Fig. 8 Variable plot at $\boldsymbol{\vartheta = -\frac{\pi}{14}}$

This is just a reflection of $\vartheta = \frac{\pi}{14}$, so for any angle, the reflection would be seen for its conjugate.

Based on these results and the verification process outlined above, the requirement has been satisfied. The system stabilised within 13 seconds and did not deviate by more than 0.1m from the centre line once stabilised for all angles $-\frac{\pi}{14} \leq \vartheta \leq \frac{\pi}{14}$. It should be noted that the case of $\vartheta = \frac{\pi}{25}$, the simulation produced irregular readings that did not fit the pattern observed in the other examples as oscillation was evident from the lateral error. It also took significantly longer to stabilize than the other experiments. This was confirmed across all 3 trials. It brings into question the reliability of the model given a lack of consistency.

B. Oscillations

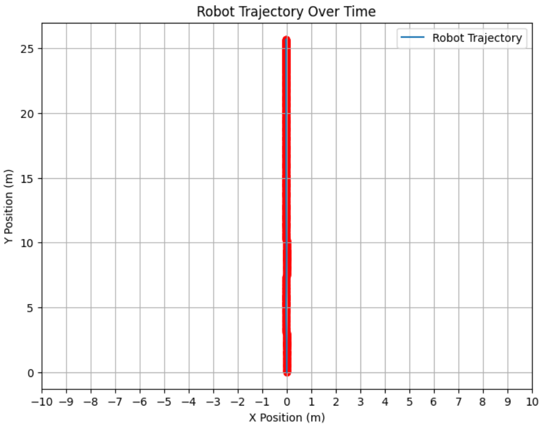


Fig. 9 Trajectory at $\boldsymbol{\vartheta = 0}$

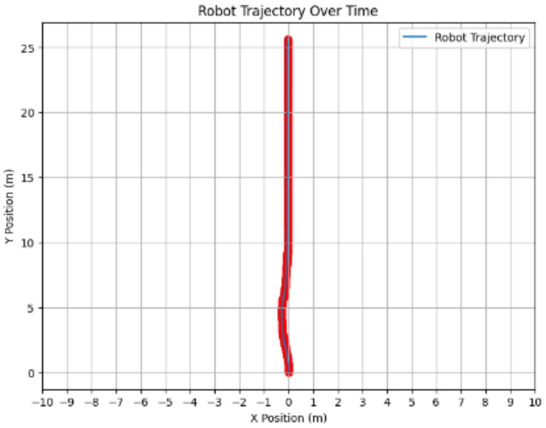


Fig. 10 Trajectory at $\boldsymbol{\vartheta = \frac{\pi}{50}}$

At $\vartheta = 0$ the robot moved in essentially a straight line and did not deviate any significant amount as expected. At $\vartheta = \frac{\pi}{50}$ the robot is observed to correct its course almost immediately to the centre line, again significant oscillations were not recorded.

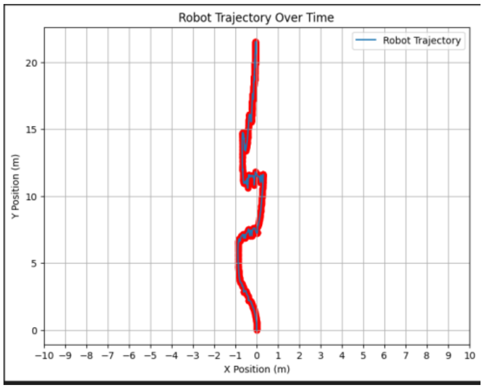


Fig. 12 Trajectory at $\boldsymbol{\vartheta = \frac{\pi}{25}}$

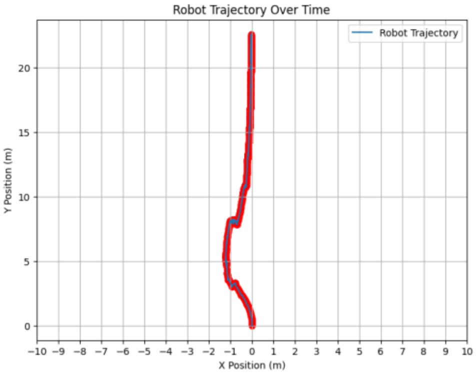


Fig. 11 Trajectory at $\boldsymbol{\vartheta = \frac{\pi}{14}}$

At $\boldsymbol{\vartheta = \frac{\pi}{15}}$, oscillations were observed, however they did not exceed 1m from the centre line. At $\vartheta = \frac{\pi}{14}$ the system behaved ideally, no significant oscillations were observed, the system corrected and smoothed into the centre line.

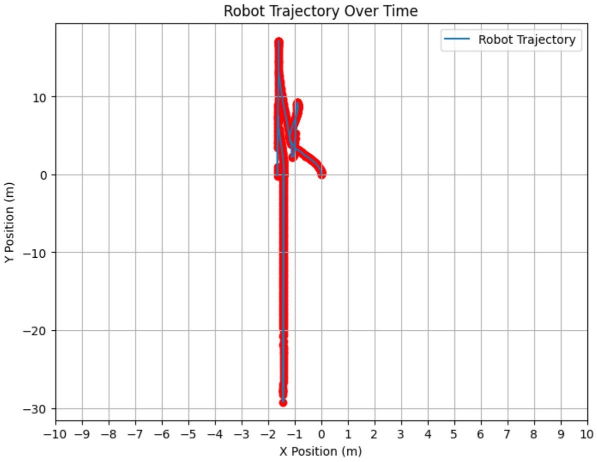


Fig. 13 Trajectory at $\boldsymbol{\vartheta = \frac{\pi}{10}}$

As noted above, the robot loses control and then falls, evident from the erratic path shown.

Based on these results and the verification process outlined above, the requirement has been satisfied. The system stabilised without oscillations exceeding 1m from the centre line for all angles $-\frac{\pi}{14} \leq \vartheta \leq \frac{\pi}{14}$. Again, as mentioned above, $\vartheta = \frac{\pi}{25}$ showed irregular results being the only observed angle to oscillate at all.

C. Limitations

Certain inputs will be changed, and their resulting trajectories mapped to observe differences. This will shed light on how sensitive the system is.

A much larger of ϑ angles were tried to see if the robot would fall of the ends of the surface. This was not observed through all angles testing including $\frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{6}$ it was noted that at $\vartheta = \frac{\pi}{8}$ the system was able to stabilize as described by the requirements,

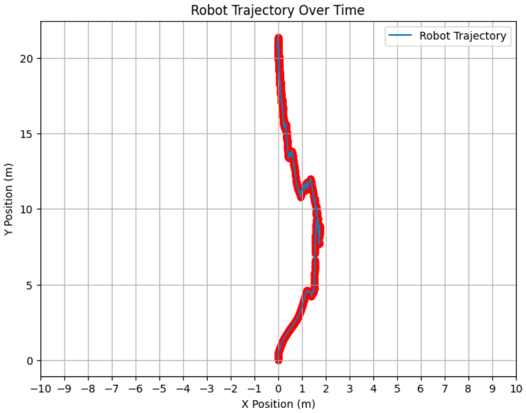
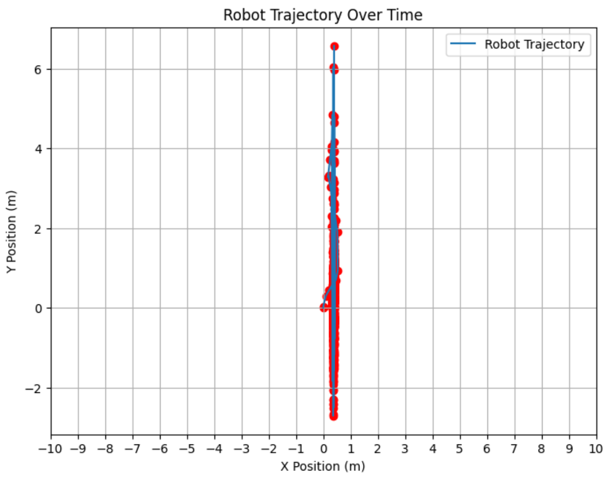


Fig. 14 Trajectory at $\boldsymbol{\vartheta = \frac{\pi}{8}}$

For all other experiments, ϑ is chosen to be $\frac{\pi}{14}$ since it produced a stable control to equilibrium.



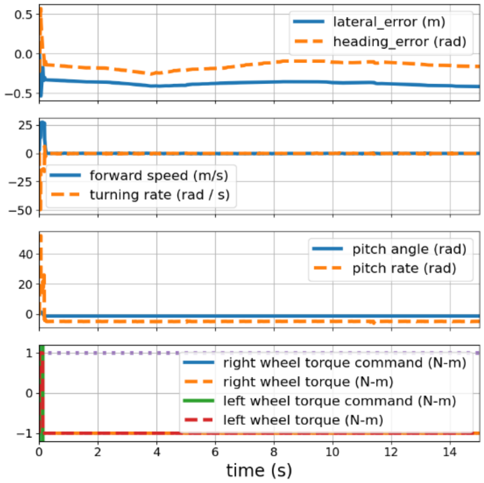


Fig. 15 Trajectory at $\boldsymbol{v_{station} = -10}$

Fig. 16 Variable plot at $\boldsymbol{v_{station} = -10}$

The robot immediately collapses. The gravitational force experienced is too great for the torque produced by the wheels to overcome.

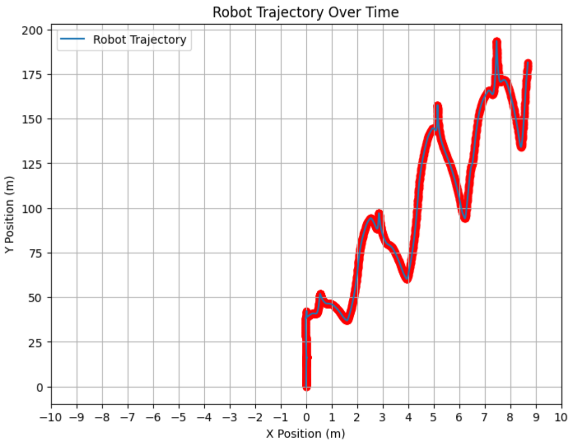


Fig. 17 Trajectory at initial speed of 10m/s

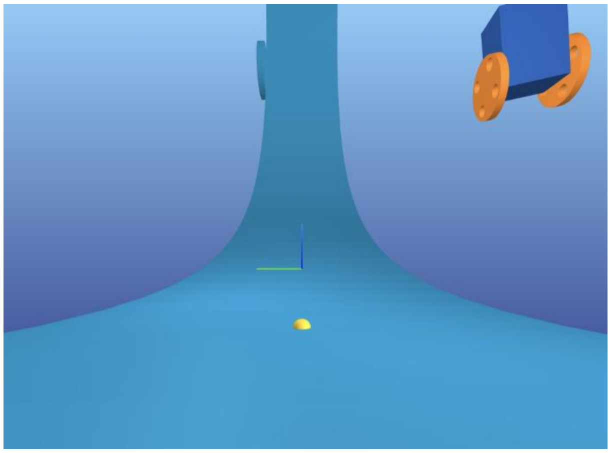


Fig. 18 Capture at initial speed of 10m/s

When the initial speed is increased to 10m/s the robot in fact floats away from the ring. This suggests that it reaches escape velocity relative to the ring's gravity and therefore is no longer bound by it. Using trial and error, the fastest speed observed that the robot could travel around the track without falling off and still being able to stablilise is 4m/s.

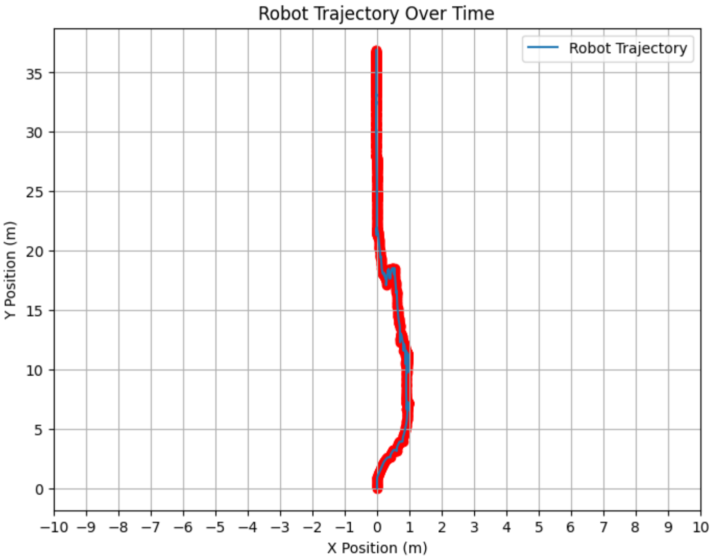


Fig. 19 Fastest observable speed around track, 4m/s

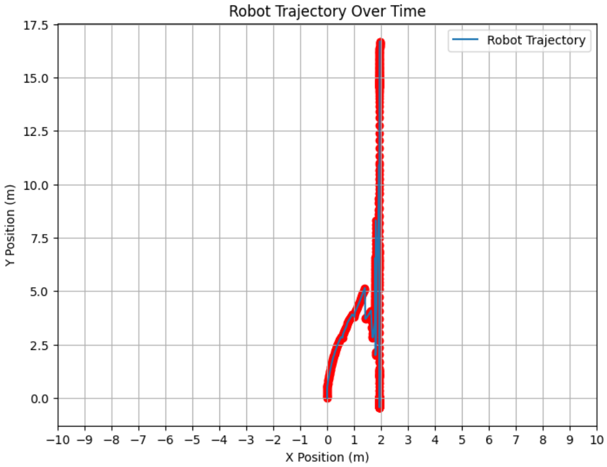


Fig. 21 Trajectory at Q having 10 times original values

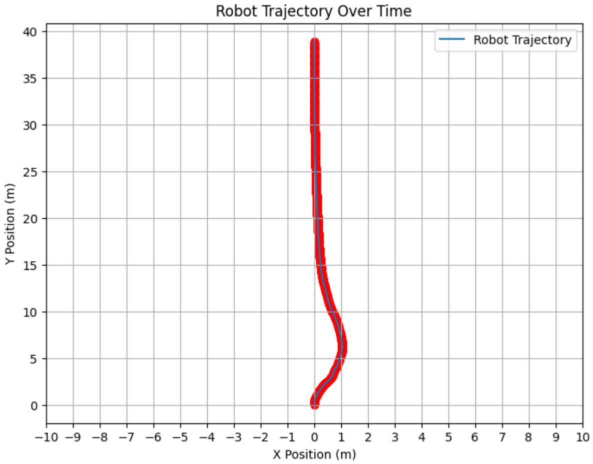


Fig. 20 Trajectory at Q having a tenth of the original values

At ten times the original values, the robot deviates rapidly from the centre line and collapses, however at a tenth of the original values, the path is much smoother and the robot stabilises, indeed, the smoothing appears more refined and stable than at the original values. This is consistent with the functionality of the state cost matrix. Increasing the values makes the control response more aggressive, but at the same time for volatile. Comparatively, reducing the values results in smoother changes but can make the system less precise.

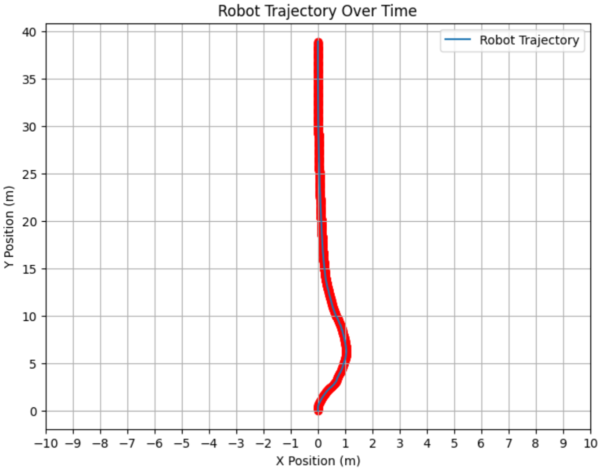


Fig. 22 Trajectory at R having 10 times original values

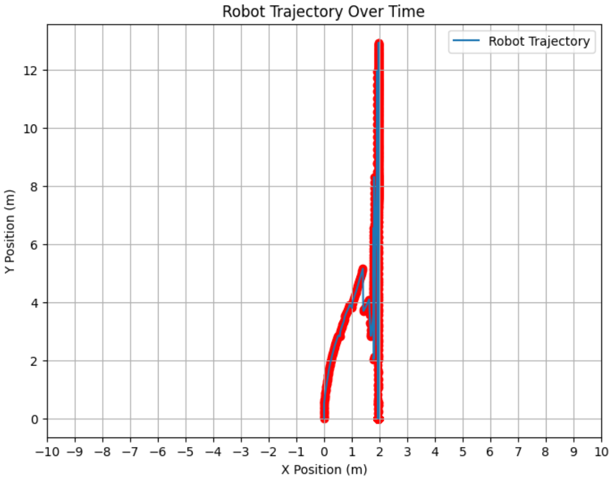


Fig. 23 Trajectory at Q having a tenth of the original values

The inverse results are seen for the case where the cost effort matrix is changed in the same way as Q. This is expected as the changing the R matrix has the exact opposite effects of those on Q.

For all experiments, the system was always controllable, as per the rank check.

VI. Conclusion

The exploratory research presented in this study lays a significant foundation for further advancement in the control mechanisms of differential-drive robots, particularly in artificial gravity environments. The development and implementation of an adaptive controller, as illustrated, highlights both the challenges and potential of robotic mobility in complex gravitational conditions.

From the observations and findings in this report, several avenues for future work emerge:

Trajectory Analysis: The study hints at discrepancies between linear model predictions and nonlinear simulation results as the model was shown to behave irregularly at certain inputs. Future research should focus on quantifying these differences, enhancing model fidelity, and refining control algorithms to better align theoretical predictions with empirical results.

Influence of Initial Conditions: The initial conditions' impact on the resulting motion of the robot remains a critical aspect. Increasing velocity of the ring increases the acceleration due to gravity felt by the robot. It was shown to be unable to function at certain values. Changing the initial angle showed significant differences. Although not observed the inconsistency seen at $\frac{\pi}{25}$ indicated that there likely exists an angle within that range that the robot collapses and fails to stabilize. Not all angles could be tested due to processing limitations. That being said, the verifications of the requirements from this study should be approached as initial assumptions and not empirical fact. Adjustments of the state cost and cost effort matrices behaved as expected. Subsequent experiments should systematically alter these conditions to evaluate their influence on stability, maneuverability, and control efficiency to therefore optimize these conditions.

Optimizing Speed and Stability: Determining the maximum speed at which the robot can reliably navigate without compromising stability is crucial. This study found it at 4m/s. Future studies should balance the trade-off between speed and stability, potentially integrating real-time adaptive control strategies to dynamically adjust to varying speeds.

Performance Metrics: Understanding how performance, particularly lateral error, varies with speed is vital. It was noted that larger speeds caused larger oscillations and, in some cases, resulted in the robot reaching escape velocity and ‘floating’ away from the platform. Future work should include a comprehensive analysis of performance metrics across a range of speeds to establish optimal operating conditions.

Starting Conditions: The impact of the robot's initial motion state (rest vs. moving) on its subsequent performance showed that at an initial velocity of 0, the robot remained stationary and at higher velocities the robot behaved as above. Again, further studies would attempt to optimize this factor and design a model that could move from rest. An expansion of this could be the ability of the robot to self-rectify itself if it falls.

In summary, while the current study has made commendable strides in understanding and controlling differential-drive robots in artificial gravity environments, considerable work remains. The control system laid out can effectively be used to navigate the robot manually through obstacles. However, the potential improvements and expansions outlined above not only promise enhanced performance and reliability but also pave the way for practical applications in extraterrestrial exploration, automated transportation, and robotic systems in varying gravitational conditions. The integration of advanced computational models, robust control strategies, and real-world testing will be key in pushing the frontiers of this exciting field.

VII. Acknowledgements

The author wishes to express their sincere gratitude to Professor Melkior Ornik for their invaluable guidance and insights throughout the course of this research and by imparting the required knowledge to complete it. Special thanks are also extended to Gokul Puthumanaillam for their significant contributions and tireless efforts to clarify and walk the author through the experimental and analytical aspects of this study.

VIII. References

"Design Project 2," AE353 - Introduction to Control System Design, Fall 2023, accessed [12/11/2023], <https://github.com/uiuc-ae353/ae353-fa23/wiki/Design-Project-2>