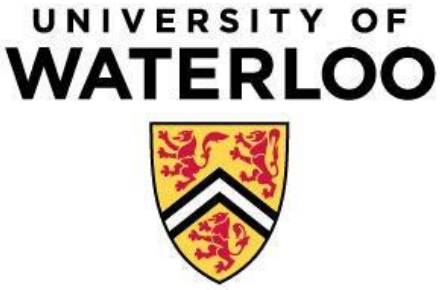
X`Analysis and Simulation of a Human Transporter



Richard Ding, Richard Wang, Armaan Sengupta, Kean Yeung

Group 1

April 4th, 2025

SYDE 351: Systems Models

Dr. Charbel Azzi

Abstract

This report presents the modeling and simulation of a two-wheeled human transporter, which is modeled as an inverted pendulum. The primary objective is to analyze the system's dynamic behavior in forward motion and determine the optimal values for parameters such as user lean angle and thrust force to achieve the maximum forward translation distance. The approach begins by constructing free-body and kinematic diagrams (FBDs and KDs) to define forces and constraints. This is followed by the derivation of nonlinear equations of motion. These equations are then linearized with small-angle approximations and rewritten to form a state-space model. Two simulation platforms, Simulink and ESI SimulationX, were utilized to analyze the system's characteristics. Simulink was used to create a block diagram and evaluate the control response and sensitivity of the system to thrust force and lean angle inputs, while SimulationX provided more of a physically grounded model, including mass distribution and joint dynamics. Extensive parameter testing/variation and input optimization were conducted using both platforms to assess the system's behavior under different conditions. Python scripts were used to optimize the workflow and plot the data for visualization, ultimately contributing to isolating the optimal values which allow the system to achieve the maximum possible distance. The optimal values obtained were a thrust force of 170 N and lean angle of -0.2 radians (-11.4592 degrees) in Simulink, resulting in a maximum distance of 12.3 meters. In contrast, SimulationX yielded a maximum distance of 9.15 meters with a thrust force of 110 N and an angle of 11 degrees. The study concludes with a comparison of the different outcomes of Simulink and SimulationX simulations and offers design recommendations to improve the mass distribution of the system for more optimal results and possible controller integration to allow the system to compensate for any imbalances.

Table of Contents

Abstract 2

Table of Contents 3

List of Figures 4

List of Tables 5

1 Introduction (1 page) 6

2 Background (1 page) 7

3 Methodology 8

3.1 FBD and KD of the Subsystems 8

3.2 Mathematical Model of the Transporter 10

3.3 Linearization and State Models 13

3.4 Numerical Simulation with Simulink 15

3.5 Design Methodology of Transporter Using SimulationX 17

3.6 Iterations of Parameters in SimulationX 20

4 Results and discussion 23

4.1 Simulink Results 23

4.2 SimulationX Iterated Results 25

5 Conclusions and recommendations (1-2 pages) 27

6 Acknowledgements 28

7 References 29

Appendix A: MATLAB Init Code 30

List of Figures

Figure 1: FBD of the System 9

Figure 2: KD of the System 10

Figure 3: Simulink Model of the Transporter System 16

Figure 4: SimulationX Human Model 17

Figure 5: SimulationX Handlebar Setup 18

Figure 6: SimulationX Full Human Transporter Setup 19

Figure 7: Cleanup.py Python Script to Clean Output Data 22

Figure 8: Python File to Plot Cleaned Data 22

Figure 9: Initial Simulation with Untuned Parameters 23

Figure 10: Distance vs Thrust Force and Lean Angle 24

Figure 11: Final Tuned Result 24

Figure 12: Broad Test for Large Range of Possible Inputs 26

Figure 13: Narrow Test with Smaller Window Beta and Force 26

List of Tables

Table 1: Parameters Used 8

Table 2: Human Parameters 18

Table 3: Transporter Parameters 20

Table 4: Final Optimized Values 25

1 Introduction

The development and use of personal transportation devices have significantly increased in recent years, particularly in urban environments and campuses. Among these devices, the two-wheel human transporter, exemplified by the Ninebot S Max, has become notably popular at the University of Waterloo campus and within city environments due to its compact form and user-friendly operation. They are efficient and convenient for short-distance travel.

However, the inherently unstable configuration of a two-wheeled transporter, which mimics an inverted pendulum, requires advanced control mechanisms to maintain stability and balance. This instability is primarily due to the elevated center of mass and the need for coordination between the wheels when maneuvering. Without proper control, the transporter and its user risk falling during operation.

The objective of this project is to analyze, solve, and experimentally model the human transporter system. This study focuses exclusively on the forward and backward translational motion, deliberately excluding turning motions and other advanced control systems. By modeling the transporter as a simplified two-wheel inverted pendulum, the aim is to derive and solve the equations of motion before experimentally validating these results using the SimulationX software.

Ultimately, the project's goal is to optimize the model to maintain balance and maximize the distance traveled without falling. Results from this study provide insights into dynamic stability and control considerations, aiding in the potential improvement and safer design of similar personal transportation devices.

2 Background

Personal mobility devices have seen significant improvements in recent years. Initially, mobility was solely reliant on human power, with innovations such as manual wheelchairs, crutches, and basic scooters improving independence for those facing impairments. In recent years, personal mobility devices have seen a transition from simple manual tools to sophisticated electrically driven systems designed to enhance individual transportation and accessibility.

One notable innovation in personal mobility is the introduction of self-balancing transporters. These transporters utilize gyroscopic sensors and accelerometers to maintain balance, enabling users to control their movement by shifting body weight. Introduced in the early 2000s, devices like the Segway marked a critical point in the evolution of personal transport, blending portability with advanced stabilization technology to facilitate efficient short-distance travel, particularly in urban spaces.

Subsequent innovations include electric scooters, hoverboards, and electric unicycles, all reflecting a broader societal shift toward sustainable and efficient transportation solutions. These compact, battery-powered devices have gained popularity not only as recreational tools but also as practical alternatives to conventional vehicles, significantly reducing congestion and environmental impact within urban centers (PBIC, 2024).

Today, the personal transporter industry continues to benefit from advancements in battery technologies, materials science, and intelligent control systems. Features like regenerative braking, real-time connectivity, and enhanced safety functionalities are becoming standard in contemporary transporters, continually improving user experience and operational efficiency. Looking ahead, further integration of autonomous capabilities and smarter human-machine interfaces is expected, potentially transforming personal transporters into even more intuitive, safe, and accessible mobility solutions (PBIC, 2024).

3 Methodology

3.1 FBD and KD of the Subsystems

Both the FBD and KD of the Human Transporter System are shown in Figure 1 and Figure 2, the list of parameters is given below in Table 1:

Table 1: Parameters Used

F_T	Thrust Force	[N]	Variable
F_N	Normal Force to Wheel	[N]	Variable
F_f	Friction Force	[N]	Variable
O_x	Reaction Force at pin	[N]	Variable
O_y	Reaction force at pin	[N]	Variable
g	Gravity Constant	m/s^2	9.81
θ	Tilt of Transporter	[rad]	Variable
β	Lean Angle	[rad]	Variable
m	Mass of Wheel	[kg]	35
m_u	Mass of User	[kg]	50.5
R	Radius of Wheel	[m]	12.7
L_g	Distance from O to G	[m]	1
$(I_G)_u$	Moment of Inertia of User	$[kg \cdot m^2]$	20
$(I_G)_w$	Moment of Inertia of Wheel	$[kg \cdot m^2]$	0.02
$\ddot{x}$	Acceleration in x Direction	$[m/s^2]$	Variable
$\ddot{\theta}$	Angular Acceleration	$[rad/s^2]$	Variable

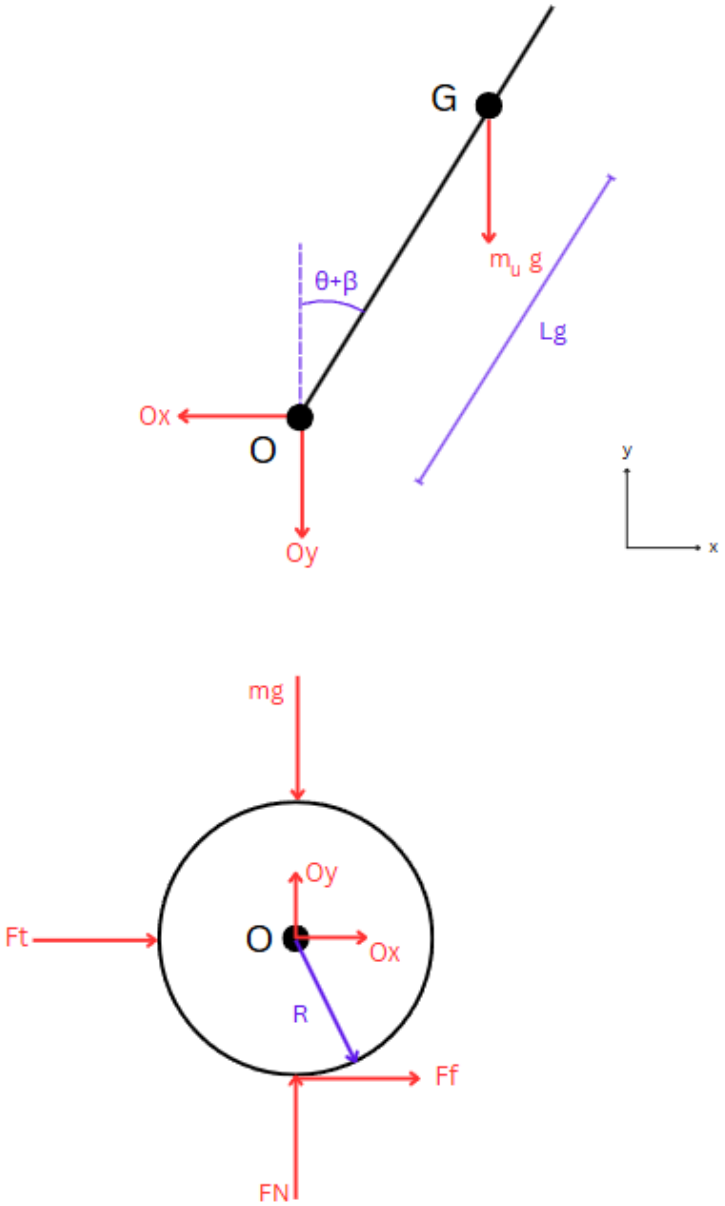


Figure 1: FBD of the System

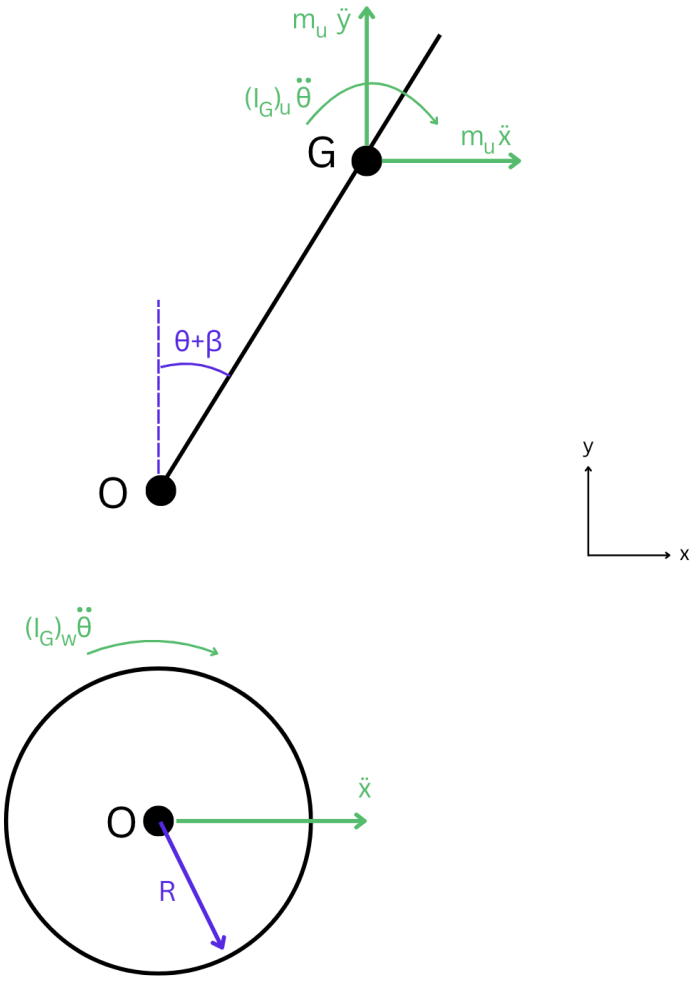


Figure 2: KD of the System

3.2 Mathematical Model of the Transporter

To develop the mathematical model of the transporter, several assumptions were made to simplify the scope of the problem and reduce number of parameters to solve for.

1. The system consists of two identical wheels plus the base.
2. The total mass of these elements is lumped at the wheel center of mass with mass M.
3. Each wheel has a radius of R and moments of inertia $(I_G)_w$.

4. Rolling without slipping implies a relationship between wheel angular motion and translational motion:

$$r\,\theta = x, \quad r\,\dot{\theta} = \dot{x}, \quad r\,\ddot{\theta} = \ddot{x}$$

5. We assume small angles, so

$$\sin(\theta) \approx \theta \; and \cos(\theta) \approx 1$$

Starting by solving for the wheel-base subsystem. First is to write all newton's second law equations, by taking the force balance in the x, y and torque about the wheel center.

In the y-direction:

$$\sum F_y = m\ddot{y} \quad \Rightarrow \quad F_N - mg + a_y = 0 \quad \Rightarrow \quad F_N = mg - a_y$$

Moment about wheel center:

$$\sum M_O = -(I_G)_w\,\alpha_w, \quad \alpha_w\,R = \ddot{x} \quad \Longrightarrow \quad \alpha_w = \frac{\ddot{x}}{R}$$

$$F_F\,R = -(I_G)_w\,\frac{\ddot{x}}{R} \quad \Longrightarrow \quad F_F = -\frac{(I_G)_w}{R^2}\,\ddot{x}$$

In the x-direction:

$$\sum F_x = m\,\ddot{x} \quad \Longrightarrow \quad F_T + a_x + F_F = m\,\ddot{x}$$

Substituting F_F :

$$F_T + a_x - \frac{(I_G)_w}{R^2}\,\ddot{x} = m\,\ddot{x}$$

Combine terms to get:

$$\left(m + \frac{(I_G)_w}{R^2}\right)\ddot{x} - a_x = F_T$$

Next is to solve the user-handlebar system, which has the assumption that:

1. The mass of the user is combined into m_u, mass of the handlebar is neglected.

2. A pin joint at O (the wheel center) allows the user and handlebar to pivot with respect to the wheels.
3. Inertia of the motors is neglected as the motors will only be modeled via the thrust force applied at the middle of the base in-line with the wheel's axle.
4. L_G is the distance from O to the user's center of mass.
5. Let β be the fixed lean angle of the user relative to the transporter, and θ the tilt of the transporter.

Starting with summing the force in the y-direction:

$$\sum F_y = m_u \ddot{y}$$

This gives:

$$a_y = m_u L_G\left[\sin(\theta + \beta)\ \ddot{\theta} + \cos(\theta + \beta)\ \dot{\theta}^2\right] - m_u g$$

Now take the sum of forces in x:

$$\sum F_x = m_u \ddot{x}$$

$$-a_x = m_u\left[\ddot{x} + L_G \cos(\theta + \beta)\ \ddot{\theta} - L_G \sin(\theta + \beta)\ \dot{\theta}^2\right]$$

Take the sum of moments about G:

$$\sum M_G = -(I_G)_u\, \alpha_u$$

$$-a_x\, L_G \cos(\theta + \beta) + a_y\, L_G \sin(\theta + \beta) = -(I_G)_u\, \ddot{\theta}$$

Substituting a_x and a_z yields:

$$(m_u\, L_G^2(\cos^2(\theta + \beta) + \sin^2(\theta + \beta)) + (I_G)_u)\, \ddot{\theta} + m_u\, L_G \cos(\theta + \beta)\ \ddot{x} - m_u\, g\, L_G \sin(\theta + \beta) = 0$$

Using the trig identity $\cos^2\theta + \sin^2\theta = 1$:

$$(m_u\, L_G^2 + (I_G)_u)\, \ddot{\theta} + m_u\, L_G \cos(\theta + \beta)\ \ddot{x} - m_u\, g\, L_G \sin(\theta + \beta) = 0$$

Combining the user and wheel system results in two coupled differential equations for angular and x acceleration. The equations are in the form of:

$$C_1\,\ddot{x} + C_2(\theta)\,\ddot{\theta} - C_3(\theta)\,\dot{\theta}^2 = C_7\,u(t)$$

$$C_4\,\ddot{\theta} + C_5(\theta)\,\ddot{x} + C_6(\theta) = 0$$

Where:

$$C_1 = m + \frac{(I_G)_w}{R^2} + m_u$$

$$C_2(\theta) = m_u\,L_G\,\cos(\theta + \beta)$$

$$C_3(\theta) = m_u\,L_G\,\sin(\theta + \beta)$$

$$C_4 = m_u\,L_G^2 + (I_G)_u$$

$$C_5(\theta) = m_u\,L_G\,\cos(\theta + \beta)$$

$$C_6(\theta) = -m_u\,g\,L_G\,\sin(\theta + \beta)$$

$$C_7 = 1$$

$$u(t) = F_T$$

The Wheel-Base Subsystem contributes the first differential equation relating $\ddot{x}$ to thrust F_T and the coupling terms involving $\ddot{\theta}$. Handlebar-User Subsystem contributes the second differential equation that governs $\ddot{\theta}$ and couples back into $\ddot{x}$.

3.3 Linearization and State Models

Now to linearize the equations and convert to state variable system. A simplification was made to the coupled ODEs. Again, starting with small angle approximation, it was recognized that through linearization, the system is approximated to be around a nominal or point, and the system is nearly stationary at equilibrium, hence the velocity term squared can be approximated to be zero:

$$\cos(\theta + \beta) \approx 1, \quad \sin(\theta + \beta) \approx \theta + \beta$$

$$\dot{\theta}^2 \to 0$$

Rewriting the equation after eliminating the term:

$$C_1\ddot{x} + C_2\ddot{\theta} = u(t)$$

$$C_4\ddot{\theta} + C_5\ddot{x} - C_6(\theta + \beta) = 0$$

Input variables are given as u(t) as the input control force, and β as the small fixed lean angle.

To acquire the linearized equations, rearranging for $\ddot{\theta}$ and $\ddot{x}$:

$$\ddot{x} = \frac{1}{C_1}\left(u(t) - C_2\ddot{\theta}\right)$$

$$\ddot{\theta} = \frac{1}{C_4}\left(C_6(\theta + \beta) - C_5\ddot{x}\right)$$

The system vector is defined as:

$$X = \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix}, \quad \dot{X} = \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix}$$

To convert into state variable form, first rearrange into matrix form:

$$\begin{bmatrix} C_1 & C_2 \\ C_5 & C_4 \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} u(t) \\ C_6(\theta + \beta) \end{bmatrix}$$

Calculating the determinant of the A matrix:

$$Det \; = \; C_1C_4 - C_2C_5$$

Using the rule of matrix inversion:

$$\begin{bmatrix} C_1 & C_2 \\ C_5 & C_4 \end{bmatrix}^{-1} = \frac{1}{C_1C_4 - C_2C_5} \begin{bmatrix} C_4 & -C_2 \\ -C_5 & C_1 \end{bmatrix}$$

Multiplying the inverse matrix with the RHS vector yields:

$$\begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \frac{1}{C_1C_4 - C_2C_5} \begin{bmatrix} C_4 & -C_2 \\ -C_5 & C_1 \end{bmatrix} \begin{bmatrix} u(t) \\ C_6(\theta + \beta) \end{bmatrix}$$

Carrying matrix multiplication gives:

$$\ddot{x} = \frac{C_4u(t) \; - \; C_2C_6(\theta + \beta)}{C_1C_4 - C_2C_5}$$

$$\ddot{\theta} = \frac{-C_5 u(t) + C_1 C_6(\theta + \beta)}{C_1 C_4 - C_2 C_5}$$

Building the state-variable matrix:

$$\dot{x} = \dot{x}$$

$$\ddot{x} = \frac{-C_2 C_6}{C_1 C_4 - C_2 C_5} \cdot \theta + \frac{C_4}{C_1 C_4 - C_2 C_5} \cdot u(t) - \frac{C_2 C_6}{C_1 C_4 - C_2 C_5} \cdot \beta$$

$$\dot{\theta} = \dot{\theta}$$

$$\ddot{\theta} = \frac{C_1 C_6}{C_1 C_4 - C_2 C_5} \cdot \theta - \frac{C_5}{C_1 C_4 - C_2 C_5} \cdot u(t) + \frac{C_1 C_6}{C_1 C_4 - C_2 C_5} \cdot \beta$$

Combining into the final state vector form gives

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-C_2 C_6}{C_1 C_4 - C_2 C_5} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{C_1 C_6}{C_1 C_4 - C_2 C_5} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{C_4}{C_1 C_4 - C_2 C_5} & \frac{-C_2 C_6}{C_1 C_4 - C_2 C_5} \\ 0 & 0 \\ \frac{-C_5}{C_1 C_4 - C_2 C_5} & \frac{C_1 C_6}{C_1 C_4 - C_2 C_5} \end{bmatrix} \begin{bmatrix} u(t) \\ \beta \end{bmatrix}$$

Where the constants after small approximation equal to:

$$C_1 = M + M_u + \frac{(I_G)_w}{R^2}$$

$$C_2 = M_u L_G$$

$$C_4 = M_u L_G^2 + (I_G)_u$$

$$C_5 = M_u L_G$$

$$C_6 = M_u g L_G$$

3.4 Numerical Simulation with Simulink

The linearized system derived in Task 3 was implemented in Simulink by combining integrator, gain, and summation blocks to represent each term in the linearized ordinary differential equations. All equations were arranged according to the small-angle approximations for θ and the fixed lean angle β,

and the key parameters were applied directly in the block diagram. The model was configured to accept the thrust force $\boldsymbol{F_T}$ as an external input so that various values could be tested without altering the underlying structure. Simulation time was chosen to be sufficiently long to capture the initial transient behavior and any subsequent steady-state tendencies. A stop criterion was set up to end the simulation at when the angle θ reaches 90 degrees, indicative of the transporter tipping over and hitting the ground. The following figure shows the full simulation setup.

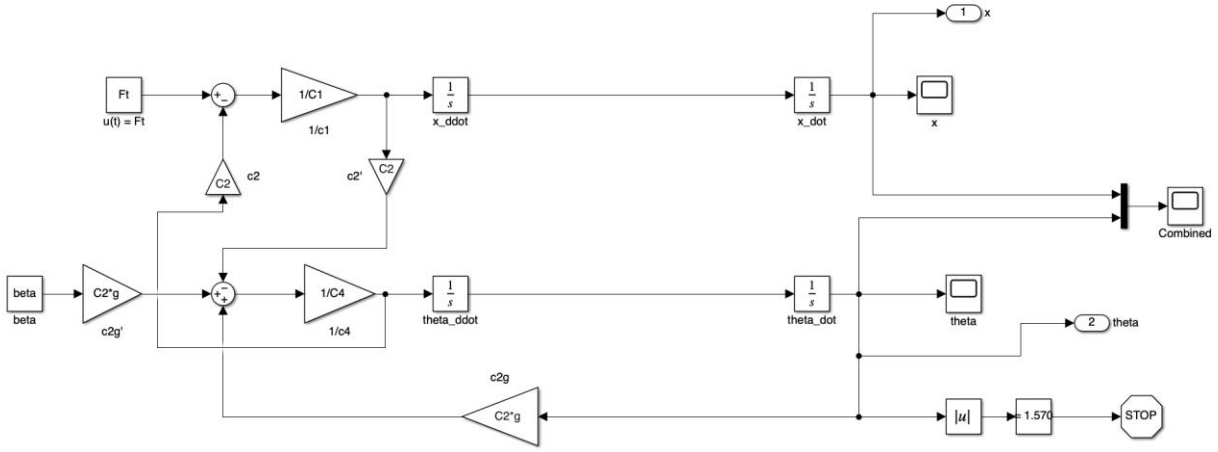


Figure 3: Simulink Model of the Transporter System

With the model accepting β and F_T as input parameters, the model was simulated by sweeping over a range of input combinations. A large range was first simulated with a low resolution, which narrowed the range of best input combination to a force of 160N to 180N in 0.5N discrete steps and angle of 0.2 to 0.8 radians in 0.05 radians increments.

3D graphs were plotted by taking x being input force, y being lean angle and z being the distance travelled by the transporter. Point of global maxima was identified, which symbolizes the combination of input values the yields the farthest distance travelled by the transporter. From this point on, hand tuning was implored to further extend the distance travelled by the human transporter, this can be found in the results section. Appendix A lists the MATLAB code used for this part of the project.

3.5 Design Methodology of Transporter Using SimulationX

ESI SimulationX was used to create a rough model of the human transporter for analysis. The system was broken down into two large components: the transporter and the human model. The coordinate system was chosen with the x-axis being the direction of travel, the y-axis pointing to the left of the human, and the z-axis being the vertical axis. The road was modelled using a 0-thickness cuboid.

The human model was created using connected massless cuboid rigid body elements, along with a sphere to model the correct mass, Center of Gravity (COG), and rotational inertia. By modeling the human this way we did not need to consider the effect of all the various geometries and masses of the other objects and could successfully replicate the given parameters. The dimensions given to the human model were defined as constants (constParams1) and were chosen based on average human body dimensions. The setup and model are shown in Figure 4, and the values used are shown in Table 2. Note that values highlighted in green are constant values set by project guidelines.

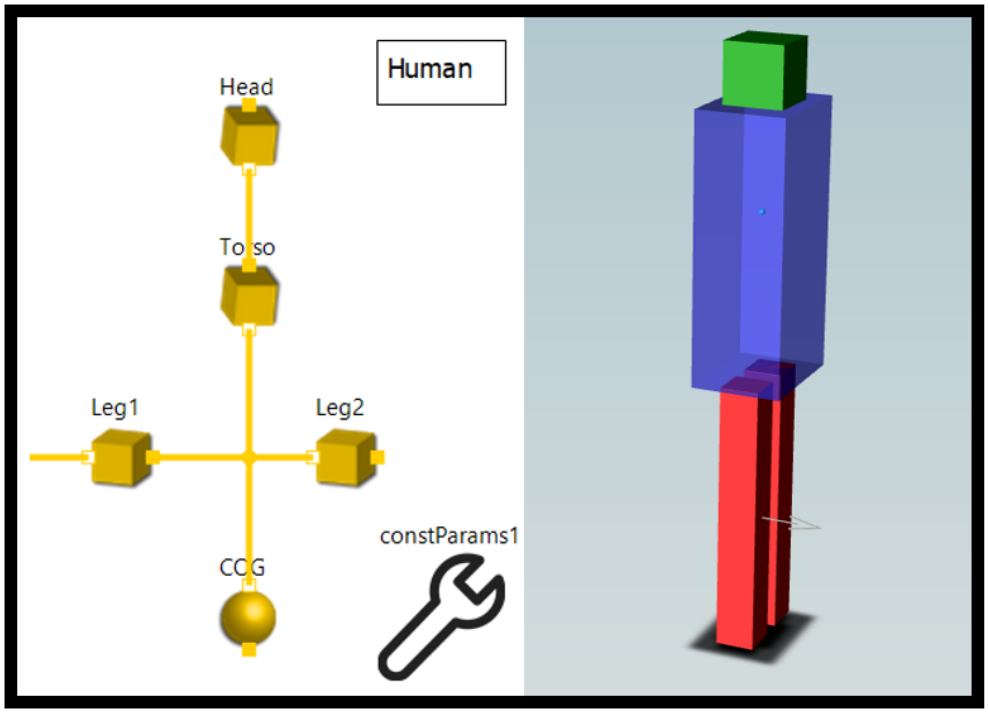


Figure 4: SimulationX Human Model

Table 2: Human Parameters

Parameter	Value	Unit
LegLength	0.9	m
LegGap	0.2	m
LegThickness	0.13	m
TorsoHeight	0.9	m
TorsoWidth	0.4	m
TorsoThickness	0.3	m
HeadSize	0.2	m
UserMass	50.5	kg
Length from Base to COG	1	m

Next, the transporter was modelled. This process was different and more complex as the dynamic capabilities of the transporter had to be considered. Two key components were used: prismatic and revolute joints. These joint elements connect rigid bodies but allow 1 degree of freedom. In this case, the joints allow for the wheels to rotate and for the entire transporter to translate along the x-axis.

The transporter itself is split into two main sections: the drive unit and the handlebars. The handlebars are modelled using cylinders connected with a rigid link, and the entire handlebar system is then connected to the base with another rigid link as shown in Figure 5.

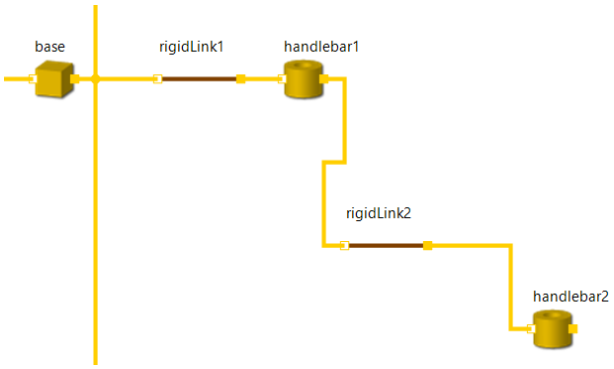


Figure 5: SimulationX Handlebar Setup

The values for mass and length assigned to these were specified by the project guidelines and listed in Table 3. Next, the drive unit was created. It is comprised of two wheels and a base. The base was created using a simple cuboid. The wheels are modelled with cylinders and attached to the base

with a revolute joint, which allows the wheels to rotate with respect to the base. The interaction between the wheels and the road is defined using a circle-line contact, which forces the wheel to always make tangent contact with the road. Finally, this is all linked together with a revolute joint at the base allowing the base to rotate with respect to the road, a prismatic joint which will allow the entire system to translate, and a global force simulates the thrust force. The full system setup is shown in Figure 6.

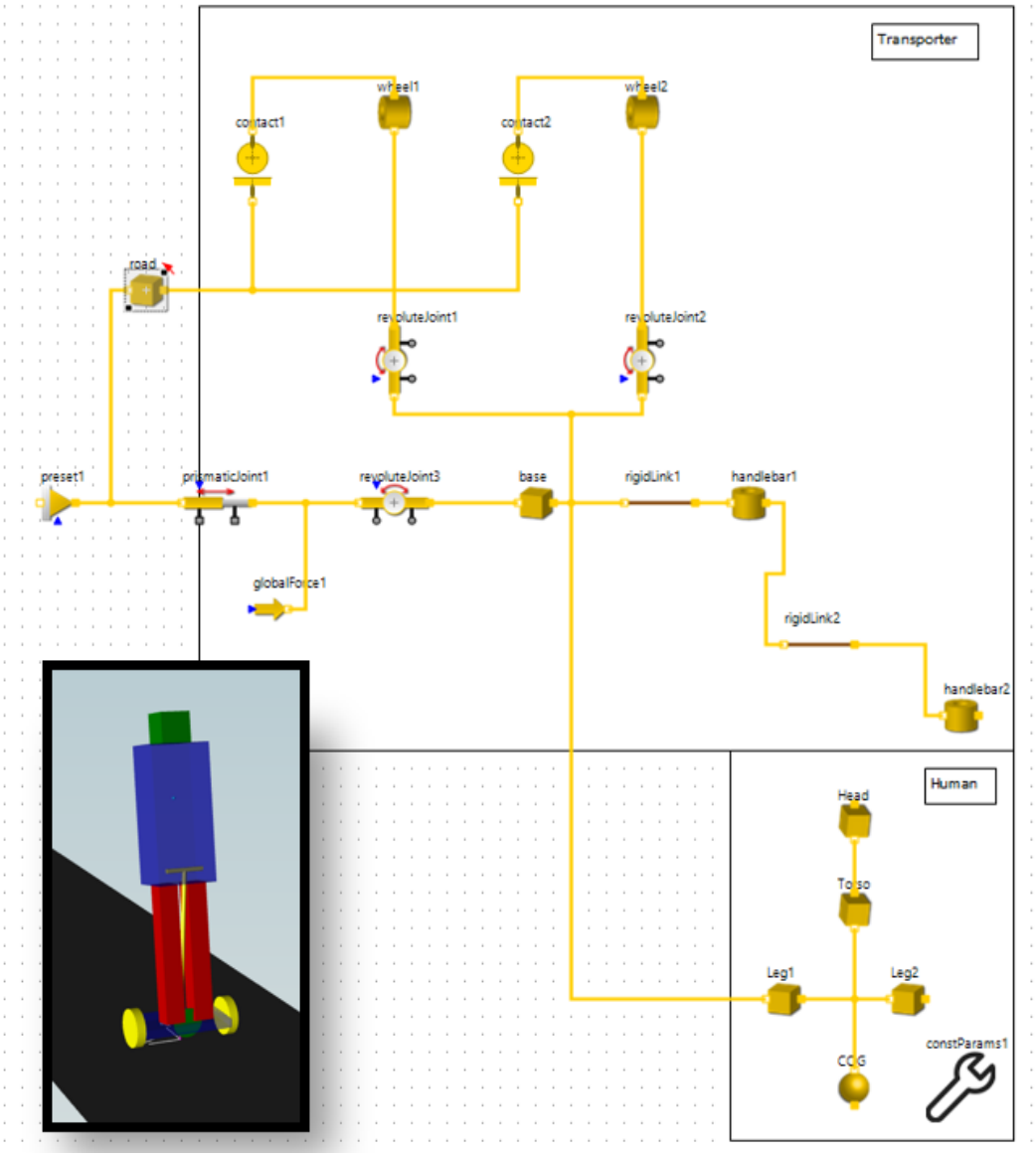


Figure 6: SimulationX Full Human Transporter Setup

All parameters used to model the human transporter are listed in Table 3. The values highlighted in green are constant values set by project guidelines.

Table 3: Transporter Parameters

Component	Parameter	Value	Unit
Upper Handlebar	Inner Radius	0	m
	Outer Radius	0.015	m
	Height	0.25	m
	Mass	0.25	kg
Handlebar Vertical Shaft	Inner Radius	0	m
	Outer Radius	0.015	m
	Height	1	m
	Mass	1.25	kg
Transporter Base	Dimension in X	0.30	m
	Dimension in Y	0.60	m
	Dimension in Z	0.06	m
	Mass	30	kg
Wheels (2x)	Inner Radius	0	m
	Outer Radius	0.127	m
	Height	0.06	m
	Mass	2.5	kg
	Axis of Rotation Symmetry	Y-axis	-

Note: the masses of the upper handlebar and handlebar shaft sum to a total of 1.5 kg, and the masses of the base and two wheels sum to 35 kg, meeting the constraints set by the project guidelines.

3.6 Iterations of Parameters in SimulationX

It is not expected that the optimized parameters found using Simulink would match the ones in SimulationX. It was decided that the best way to handle searching for the optimal values would be to conduct a "broad" test and then a "narrow" test. What is meant by a broad test is to establish a general idea of what domain of inputs (both β and F_t) will yield the maximum distance. To accomplish this, the variants wizard was used to iterate through a large combination of those input values across a 2D input space, thus the compute time was $O(n^2)$. It is important to note that for the variants wizard to work, an "auto stop" was implemented for when the modelled transporter tips more than 90°. This was done through the termination condition in SimulationX: $\text{termCond} = \text{abs}(\text{revoluteJoint3.phiRel}) > 1.5708$

This allowed the software to capture the last distance value in the variants wizard which corresponded to the maximum distance the transporter travelled for this given set of β and F_t values before tipping past 90°. The first solution space covered 520 simulations, where β was varied from 7 degrees to 20 degrees in steps of 0.5 degrees while F_t was varied from 100 to 200 Newtons in steps of 5 Newtons. This was accomplished using the Modelica syntax below:

$$\beta = 7{:}\,0.5{:}\,20$$

$$F_t = \{\{i, 0, 0\}\ for\ i\ in\ 100{:}\,5{:}\,200\}$$

Once this simulation was completed, a separate Python script was developed to parse the CSV file since the array of F_t contained commas that were incorrectly being interpreted as separate columns. Using regular expressions to just get the first element of the array as F_t (all others are 0 and irrelevant), this was written to a new CSV file. The code is shown in Figure 7. Another Python script was developed to create a 2D surface plot of distance traveled vs β and F_t to improve data visualization (code shown in Figure 8). The result of this will help show if there is any relationship between the ideal values of β and F_t, guiding what test should be run next. However, it should be noted that lumped parameter grouping where F_t has a dependance on β is not supported by SimulationX. Thus, the team decided to implement a narrow test, where a focus was placed on the specific input region where the maximum distance was achieved, and iterations were conducted with smaller step sizes to achieve more accurate results.

cleanup.py > ...

```python
import re

# File paths
input_path = 'Finer.csv'  # Input file name
output_path = 'Cleaned_Finer.csv'  # Output file name

def clean_force_value(value):
    """Extract the first number from the problematic value and remove braces."""
    # Match patterns like {100,0,0}.0 or {90.25,0,0} and extract the first number
    match = re.search(r'\{([\d\.]+),[^\}]+\}\.?0?', value)
    if match:
        return match.group(1)  # Return the first number as a string
    return value

with open(input_path, 'r') as infile, open(output_path, 'w') as outfile:
    header = infile.readline()
    outfile.write(header)

    for line in infile:
        # Fix the problematic part using regex
        cleaned_line = re.sub(r'\{([\d\.]+),[^\}]+\}\.?0?', r'\1', line)
        outfile.write(cleaned_line)

print(f"Cleaned CSV file saved at: {output_path}")
```

Figure 7: Cleanup.py Python Script to Clean Output Data

```python
# Create a grid for the surface plot
xi = np.linspace(min(x), max(x), 100)
yi = np.linspace(min(y), max(y), 100)
xi, yi = np.meshgrid(xi, yi)

# Interpolate the data for a smooth surface
zi = griddata((x, y), z, (xi, yi), method='cubic')

# Create the 3D surface plot
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

# Plot the surface
surface = ax.plot_surface(xi, yi, zi, cmap='viridis', edgecolor='none', alpha=0.8)

# Plot the line y = 10x
line_x = np.linspace(min(x), max(x), 100)
line_y = 10 * line_x
line_z = np.full_like(line_x, min(z))  # Use the minimum z value for line height
ax.plot(line_x, line_y, line_z, color='red', label='y = 10x', linewidth=2)

# Plot the maximum point with adjusted z-coordinate
ax.scatter(max_x, max_y, max_z, color='red', s=100, label='Max Point')

# Add the original data points to compare
ax.scatter(x, y, z, color='black', s=20, alpha=0.5, label='Original Data Points')

# Explicitly set axis limits to cover the data range
ax.set_xlim([min(x), max(x)])
ax.set_ylim([min(y), max(y)])
ax.set_zlim([min(z), max(z)])

# Annotate the maximum point with accurate coordinates
annotation = f'Max: Beta={max_x}, Force={max_y}, Position={max_z:.2f}'
ax.text(max_x, max_y, max_z, annotation, color='red', fontsize=10)
```

Figure 8: Python File to Plot Cleaned Data

4 Results and discussion

4.1 Simulink Results

Observations from preliminary runs revealed that the linear approximation remains valid for small tilts and moderate thrust forces, whereas excessively large or small thrust values resulted in rapid tipping of the transporter. Specifically, the θ angle is an exponential curve that suggest a rapid destabilization of the transporter after a certain threshold have been reached. This behavior confirms that the model can be employed for further tuning of thrust force and other parameters in subsequent tasks aimed at maximizing travel distance before tipping.

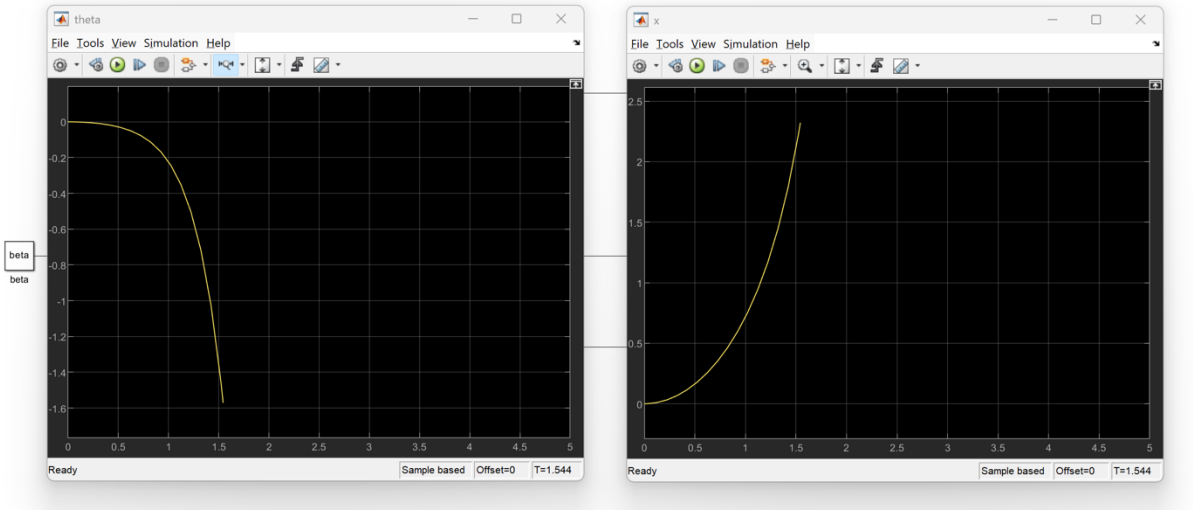


Figure 9: Initial Simulation with Untuned Parameters

Iterating through all possible combinations within the domain of thrust force and lean angle resulted in 480 unique and discrete distances. A 3D plot was constructed, and the maximum distance were identified as 6.55m at 170N thrust force and -0.2 rad (-11.46°) lean angle, as shown in the following figure.

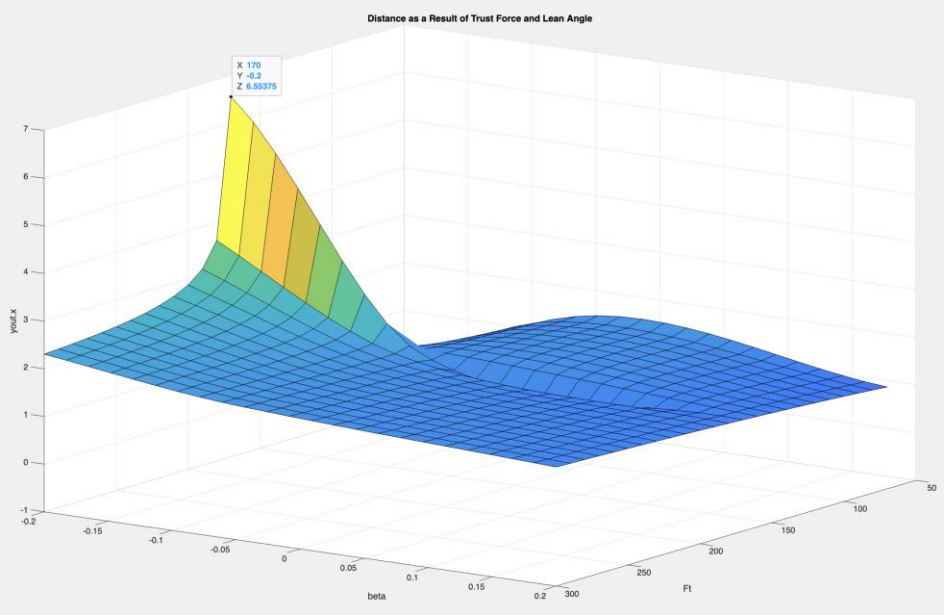


Figure 10: Distance vs Thrust Force and Lean Angle

At this point, the model was tuned by hand centered around the 170N force, and a final distance of 12.3m before θ reaches 90° at 170.203N force.

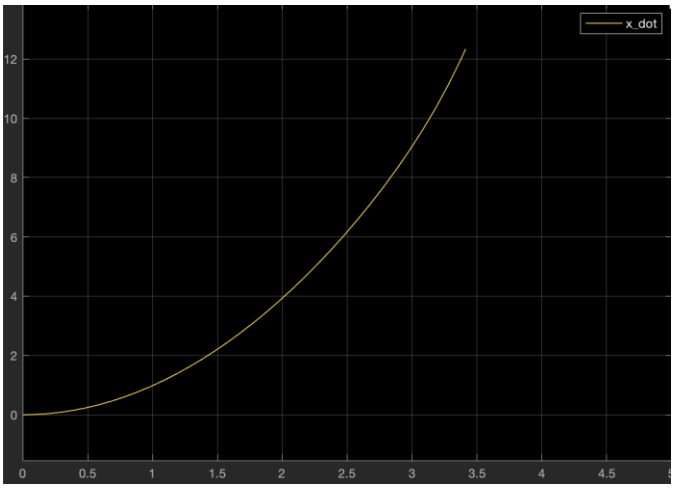


Figure 11: Final Tuned Result

Increasing the mass of the base exerts a notable influence on the transporter's tipping dynamics by raising the overall inertia of the system. A heavier base generally becomes more resistant to small tilts, thereby postponing or slowing the onset of tipping. However, the increased mass also necessitates a larger thrust force to maintain stability and achieve the same forward motion observed with a lighter

system. Thus, while adding mass to the base can mitigate rapid tipping, it simultaneously introduces higher force requirements to sustain upright travel.

4.2 SimulationX Iterated Results

From the broad test, the results in Figure 12 showed that the maximum distance is achieved when F_t is roughly 10 times greater than β. Specifically, it was observed that the maximum distance occurred when $F_t = \beta \times 10 \pm 5N$. Now, more experiments more closely clustered around that relation must be run.

The narrow test was conducted by rerunning the simulation with a smaller window for F_t at a higher step resolution. Specifically, β was narrowed down to range from 10° to 12° in steps of 0.5° (extra resolution was not wanted for β, only for F_t), while F_t ranged from 90 to 130 Newtons in steps of 0.25N (much finer resolution than β). Running the simulation and performing the same post processing and visualization as shown in Figure 13 show that the previous force was in fact still the maximum, now to one decimal place of confidence. The final optimized input values are listed in Table 4.

Table 4: Final Optimized Values

Parameter	Value	Unit
Beta Angle (β)	11	deg
Thrust Force (F_T)	110.0	N
Final Maximum Displacement	**9.15**	**m**

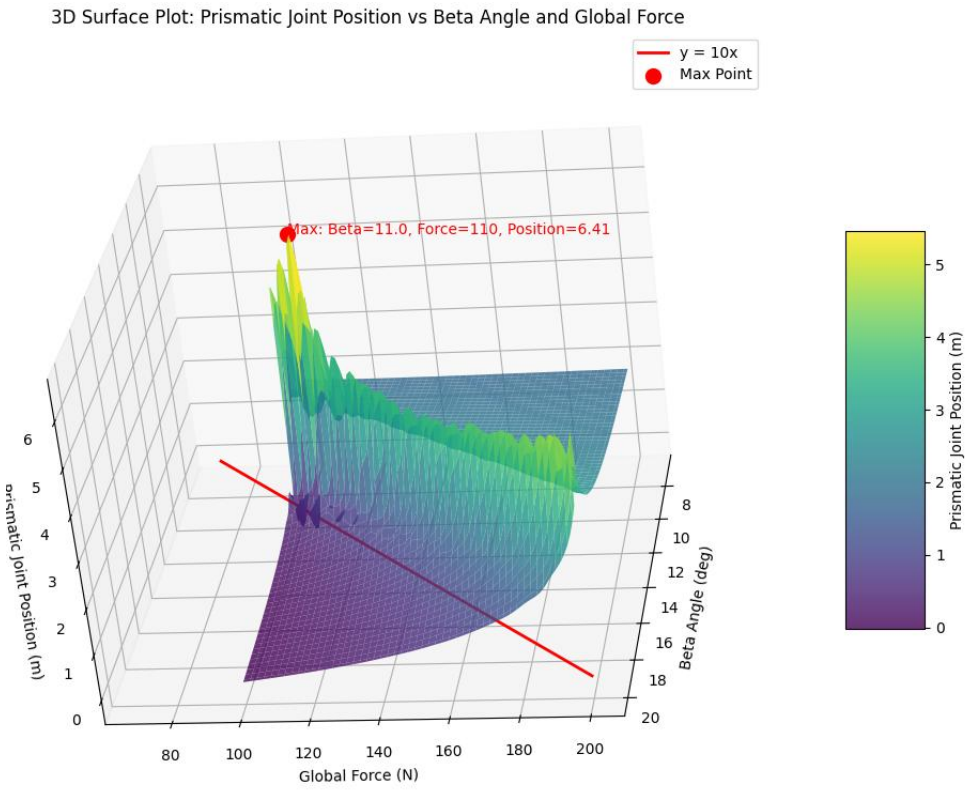


Figure 12: Broad Test for Large Range of Possible Inputs

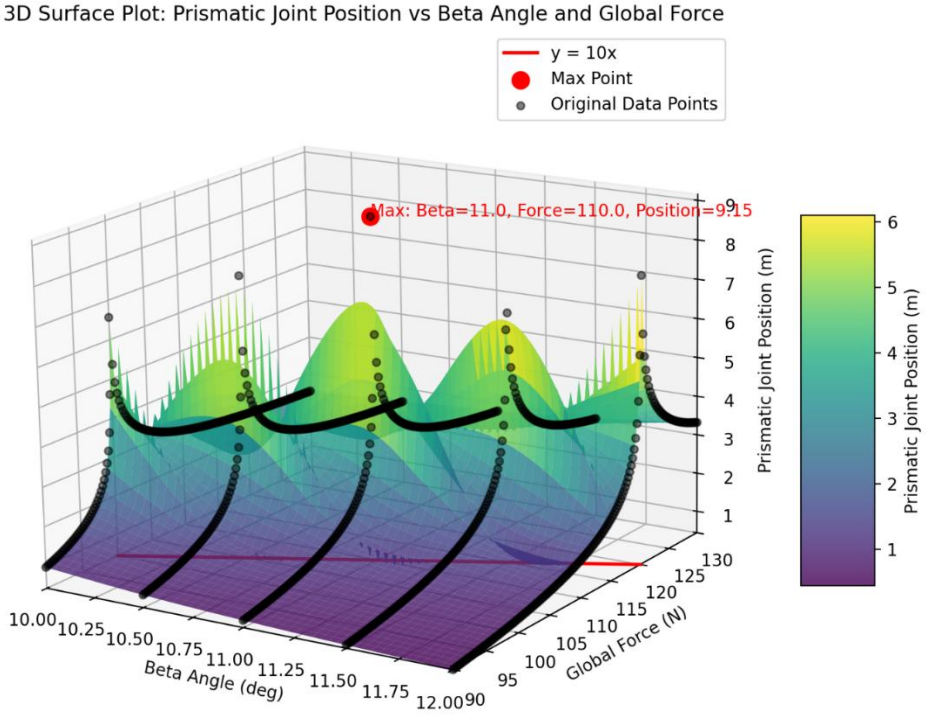


Figure 13: Narrow Test with Smaller Window Beta and Force

5 Conclusions and recommendations

This project successfully modeled, simulated, and analyzed the forward translational dynamics of a two-wheeled human transporter using both analytical techniques and numerical simulation tools. By representing the human transporter as a simplified two-wheel inverted pendulum system, the team derived a set of coupled nonlinear differential equations, which were linearized using small-angle assumptions to allow for simulation using Simulink. An approximate model was also constructed in SimulationX to validate the team's findings from Simulink with a more physics-based simulation environment.

The results of the simulations showed that the system is incredibly sensitive to changes in both thrust force and user lean angle. In Simulink, it was found that the optimal conditions for maximum distance travelled before tipping were at a thrust force of 170 N and a lean angle of -0.2 radians, resulting in a maximum distance of 12.3 m. In SimulationX, the maximum distance was achieved at a thrust force of 110 N and a beta angle of 11 degrees, yielding a maximum displacement of only 9.15 m. The difference between the two platforms highlights a key topic: while Simulink is great for theoretical models and analysis, SimulationX is more accurate as it accounts for physical interactions and constraints.

The results showed a clear exponential instability trend when the thrust or lean angle passed certain thresholds. Additionally, increasing the mass of the base improved the system's resistance to tipping, but in turn demanded higher thrust force, which suggests a vital balance between stability and energy efficiency.

A recommendation to improve the human transporter's stability and performance is to implement a Linear Quadratic Regulator (LQR) controller applied to the linearized state-space model. By assigning weighting factors that penalize deviations in both the tilt angle (and angular velocity) and the forward speed, the LQR controller continuously computes an optimal thrust force to minimize a cost

function. In doing so, it effectively counteracts tipping forces, thus extending the distance traveled under stable conditions. Compared to other feedback schemes, an LQR can balance factors such as limiting energy consumption (thrust force) and maintaining rider comfort, while correcting any deviations from an upright orientation. Moreover, it accommodates varying operating conditions, such as changes in user mass or slight slopes in the terrain, by adjusting the thrust on the fly to preserve stability.

A final recommendation is adjustments with the mass distributions of various sub-systems in SimulationX. While sub-systems like the handlebars and the entire base (user platform with both wheels) had fixed lumped overall masses specified by the project requirements, the distribution of these masses between sub-system components was at the discretion of the team (e.g. the wheels were given a combined mass of 5kg, while the platform was given a mass of 30kg. These can be varied to analyze the results of having heavier wheels but a lighter platform). The masses were held constant across all experiments, but with more time, the team would attempt to vary the mass distribution to determine if the maximum distance could be further improved.

6 Acknowledgements

The authors would like to extend their sincere gratitude to Dr. Charbel Azzi and his teaching assistants for their invaluable support and guidance throughout the course of this project. His expert advice, timely feedback, and willingness to address questions and concerns at every stage greatly facilitated the successful completion of the modeling, simulation, and analyses presented in this report.

7 References

[1] PBIC. (2024). Micromobility Typology. Pedestrian and Bicycle Information Center. Retrieved from: https://www.pedbikeinfo.org/cms/downloads/PBIC_Brief_MicromobilityTypology.pdf

Appendix A: MATLAB Init Code

```
clear all;
clc;
close all;

M = 35;
Mu = 60 + 1/2;
I_Gw = 0.02;
R = (12.7)/100;
L_g = 1;
I_Gu = 20;
g = 9.81;
beta = 0.2;
Ft = 170.203;


C1 = M + Mu + (I_Gw / R^2);
C2 = L_g * Mu;                      % cos(beta) ≈ 1    % sin(beta) ≈ (theta + beta)
C4 = (Mu * L_g^2 + I_Gu);
C7 = 1; % Given
```