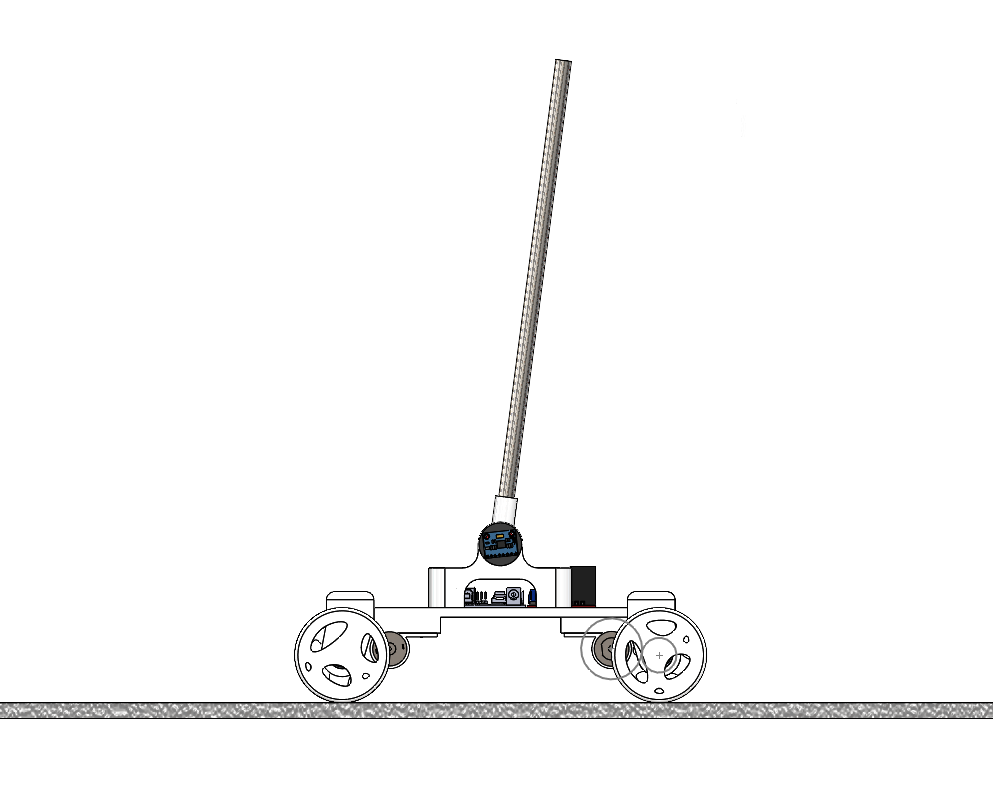
**ME 4012 Final Project Theory Formulation**

**SECTION A. System Dynamics Modelling**

Part 1. Cart Dynamics ()

**[Schematic]**



**[Known]**

1. World frame definition. Note that clockwise, rather than counterclockwise, is defined as the positive direction for angular quantities.
2. All relevant geometric & inertial quantities of the cart:

**[Find]**

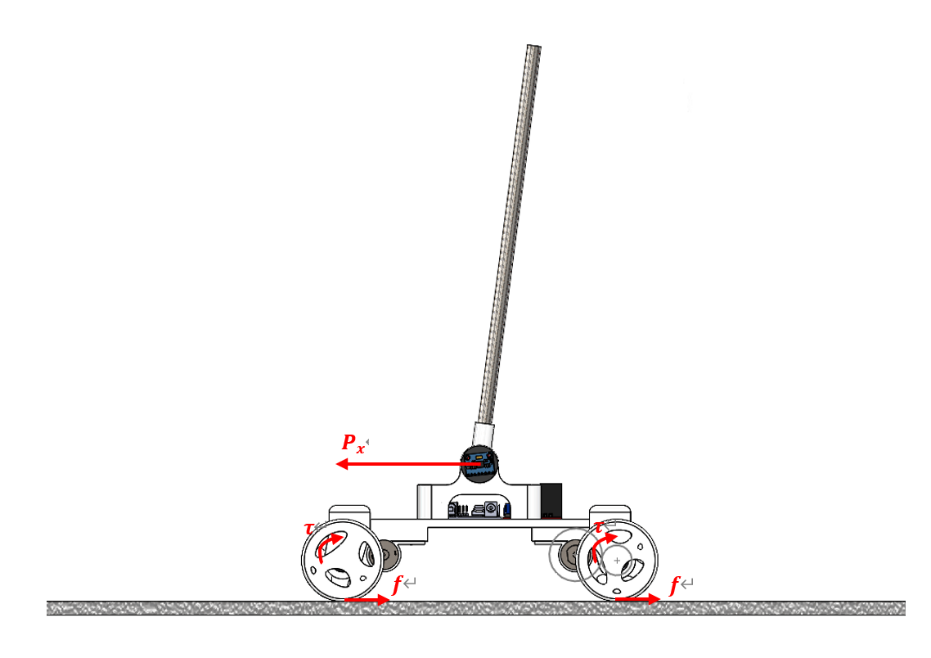
The transfer function from wheel torque to the angular displacement of the inverted pendulum from its equilibrium position .

**[Assumptions]**

1. The cart body, excluding the pendulum, is perfectly symmetrical front-to-back and left-to-right.
2. The input torques at four wheels are of identical value at all times.
3. The torque will never break the traction of the tire (non-slip condition at the wheels).

**[Analysis]**

Free Body Diagram of the Cart Body (Excluding the Pendulum) in the x direction:

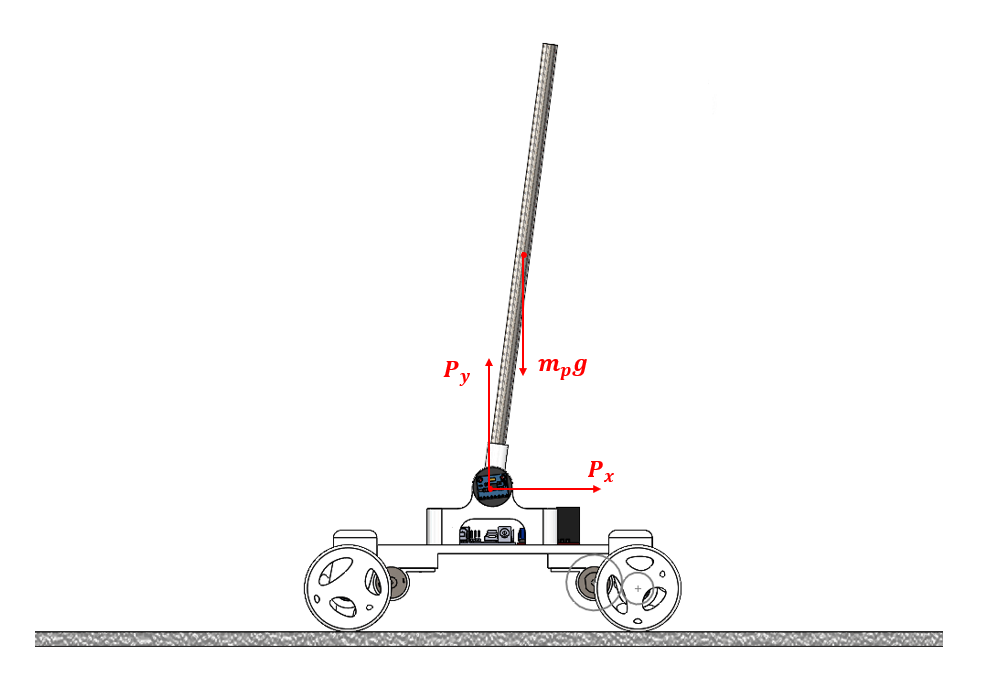


Dynamics at the wheel:

Dynamics of the cart body:

Kinematics Constraint:

Free Body Diagram of the Pendulum:



Linear Dynamics of the Pendulum:

Angular Dynamics of the Pendulum:

Kinematics Constraint:

Take second order derivative of equation (7) and (8) to transform kinematics relationship at displacement level to acceleration level:

(First Order)

(Second Order)

From equation (1) and (3), we have

Multiply both sides by :

Add the above equation with (2) to eliminate friction term:

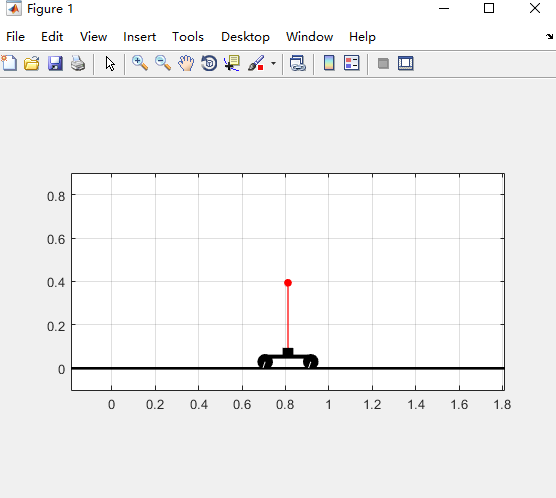
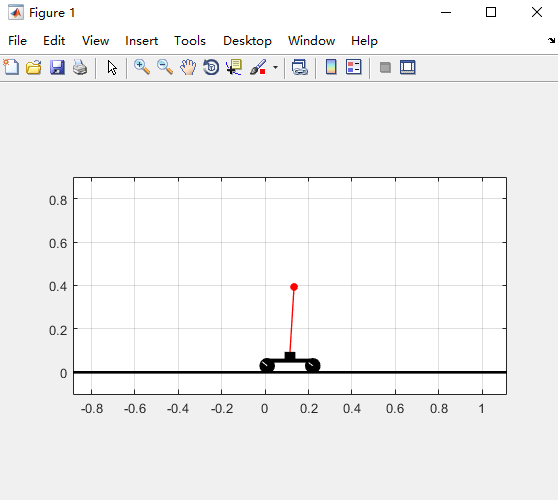
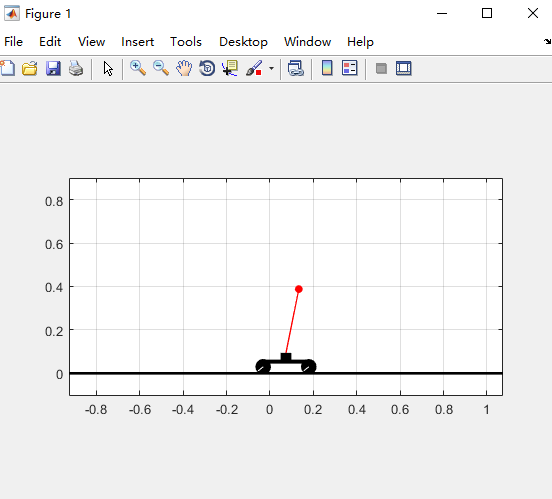
From equation (3), (4) and (9), we can represent using a bunch of accelerations:

Therefore, the first equation of motion relating input torque to the two kinematics variables and can be obtained

Combine equation (5) and (10), we can represent with a bunch of accelerations:

Plug equation (11) and (12) into equation (6), we can obtain the second equation of motion relating the two kinematics variables and .

Using the above two equations of motion, MATLAB is able to solve for the exact solutions of the dynamics when provided with initial conditions and control torque. The “nonlinear\_simulation.m” file utilizes this approach and creates a simulation environment for testing and tuning our controllers.



t = 0.03s t = 0.08s t = 0.5s

To obtain transfer function, however, the system needs to be linearized. Linearize around the operating point

Linearizing EOM1:

First, reorganize terms:

Linearize using the following approximations:

Linearize EOM2 using the same approximations:

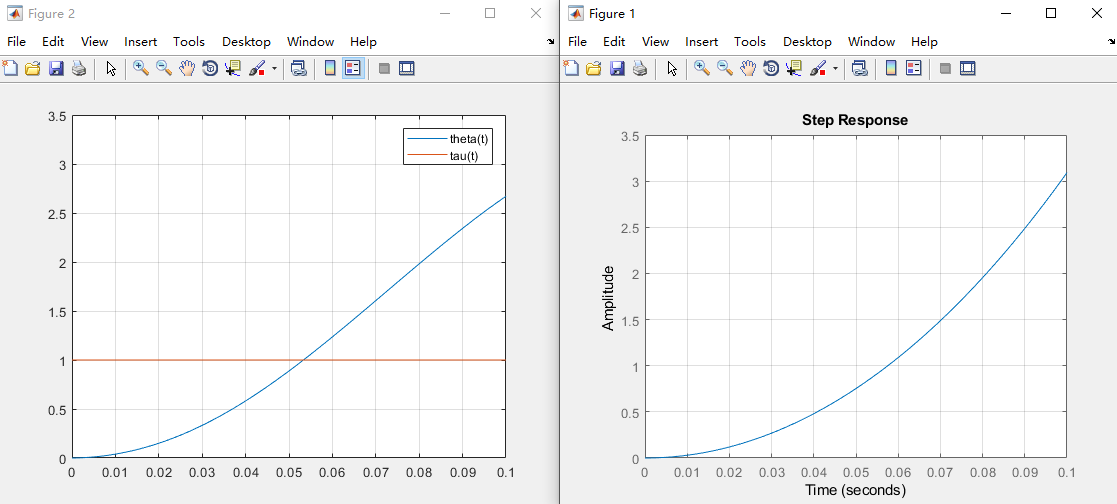
Reorganize terms:

Take Laplace transform on both linearized equations:

Plug (14) back into (13) yields the ultimate transfer function :

Overall, the system appears to be second order with zero damping, which agrees with our intuition since zero-friction condition was assumed.

Comparing the step response of the linearized system with that of the original nonlinear system:



The result is pretty satisfactory.