# Modeling GUI

## Purpose:

This GUI accepts user input of physical measurements from their quadcopter vehicle. A primary purpose of this interface is to calculate the moment of inertia matrix for the vehicle based on specific dimensions and measurements of the quadcopter. The interface also accepts the performance coefficients and other parameters obtained through motor testing. Once all of the fields are populated, the interface saves a MATLAB "structure" that contains all the parameters required to run the simulation. A structure is a data type that gives us a way to combine multiple pieces of information into a single variable that can easily be passed around the MATLAB environment. The modeling GUI can be seen below in Figure 1.

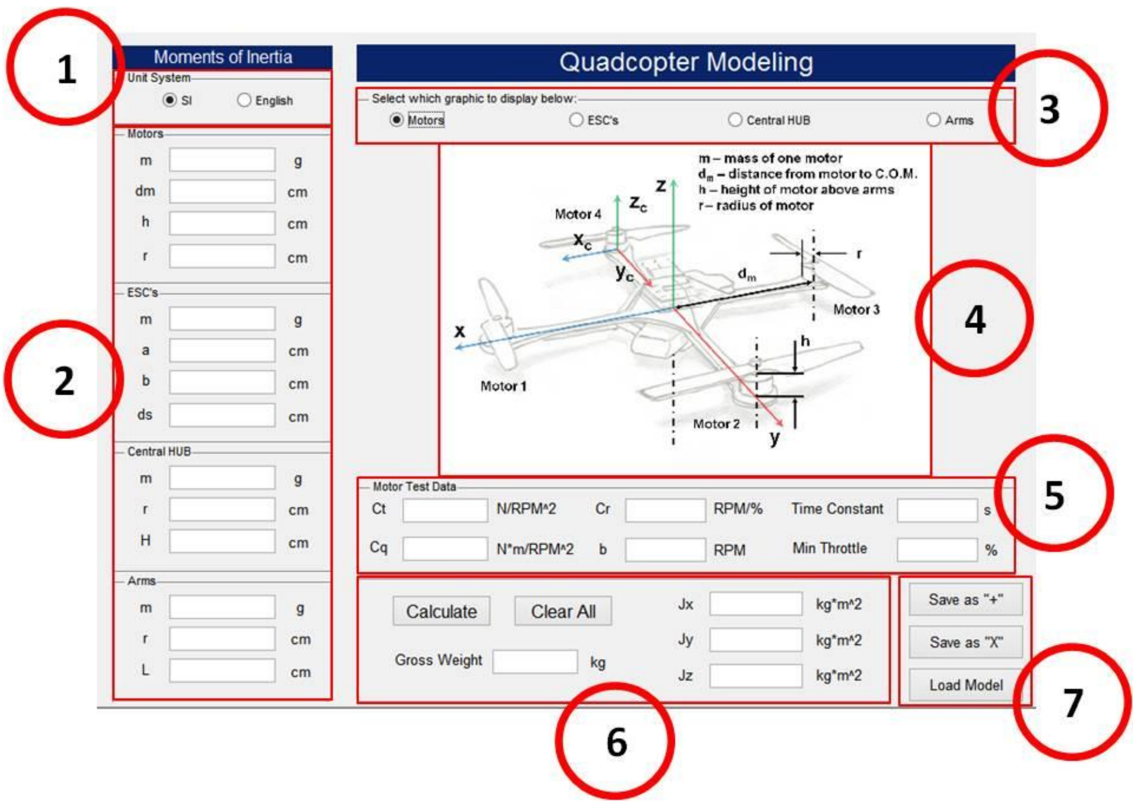



*Figure 1. Modeling GUI*

## Description:

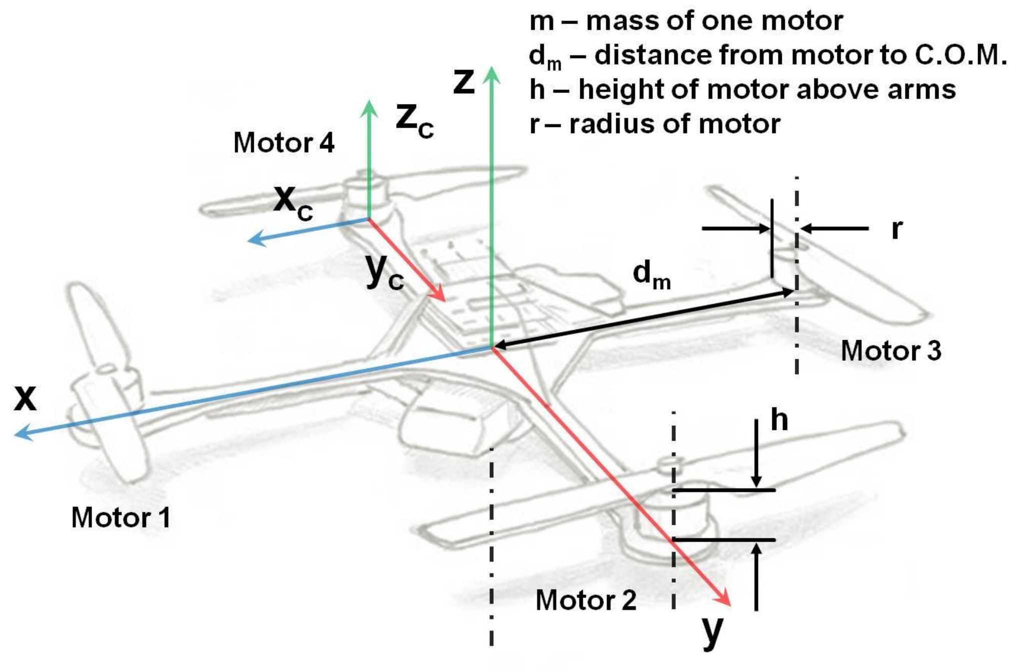
1. Unit System toggle: toggle units between SI and English based on the measurement tools being utilized. IMPORTANT: Whether inputs are in grams/centimeters or ounces/inches, the program will convert everything into the SI unit system (kilograms/meters) for use in the simulation.
2. Inputs for Motor, ESC, HUB, and Arm dimensions (either in g/cm or oz./in.).
    - Motors:
        - m: mass of 1 motor (with propeller) using scale
        - dm: distance from the center of mass (COM) of the motor to the COM of the quad vehicle (COM is assumed to coincide with geometric center!)
        - h: height of the motor above the arm (not including prop axel)
        - r: radius of a motor
    - ESC's:
        - m: mass of 1 ESC
        - a: width of an ESC
        - b: length of an ESC
        - ds: distance from the COM of the ESC to the COM the quad vehicle
    - Central HUB:
        - m: total mass of the central HUB (including battery, controller, power distribution board, etc.). This value might be most easily obtained by subtracting the mass of the individual components mentioned herein by the total weight of the quad in a "ready-to-fly" state (i.e. Hub mass = Total mass – (Motors + ESC's + Arms).
        - r: radius of the central HUB (modeled as a cylinder, estimate as needed)
        - H: height of the central hub (total height, modeled as a cylinder, see Hub diagram)
    - Arms:
        - m: mass of (1) arm (motor, ESC, etc. excluded)
        - r: radius of an arm (modeled as a cylindrical rod)

    - L: length of an arm (arm only up to attachment to HUB)
3. Toggle between graphics to display in center of GUI to assist in measuring parameters.
4. Motor, ESC, HUB, and Arm graphics displayed in center of GUI
5. Motor Test Data inputs: Coefficients obtained from motor test data are entered here. Also, “Min Throttle” (the minimum throttle setting for which actual motor rotation is achieved) can be entered. <u>If this value is not known, simply enter “0”</u>.
6. Select the “Calculate” button to calculate the Gross Weight of the vehicle, and the Jx, Jy, and Jz values of the inertia matrix. Or, select “Clear All” to clear all of the GUI fields.
7. Once the fields are populated, the user can select “Save as +” or “Save as X” to save the model in a “plus” configuration or in an “X” configuration. The parameters are saved inside a MATLAB “structure” from which the Simulink simulation extracts them during simulation. If a model is already saved, the user can select “Load Model” to select a desired structure and the GUI will populate the fields with these saved parameters. This is useful because if the user wishes to adjust 1 parameter of their model they don’t need to retype everything in again.

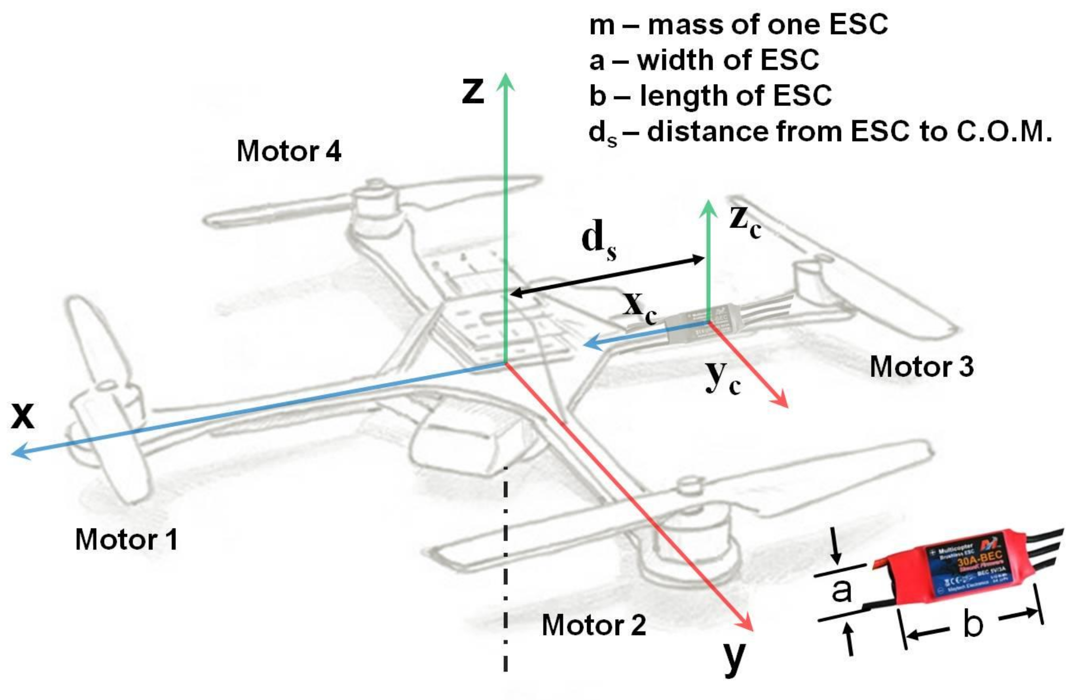
NOTE: While this GUI provides a useful interface, a user can modify the quadcopter model structure directly from the command line. However, note that this procedure must be done carefully, since some of the parameters are used within the GUI to calculate other parameters, and these operations are actually performed within the GUI itself and NOT by the structure or the simulation. In other words, changing the quadcopter model from outside the GUI is okay, but should be done only after making certain the simulation will interact with the structure in the manner you intend. For more details, you will need to look at the code of this GUI and the Simulink diagram and quadcopter dynamics S-Function “quadcopterDynamicsSFunction.m” to see how your needs can best be met.

## Graphics:

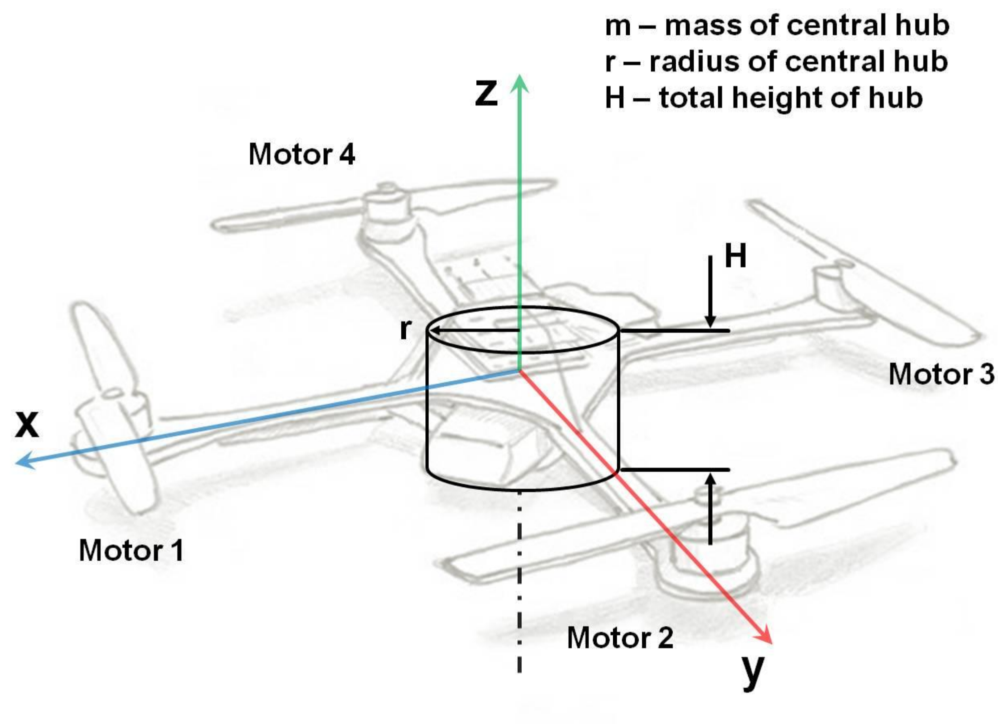
**Motors**



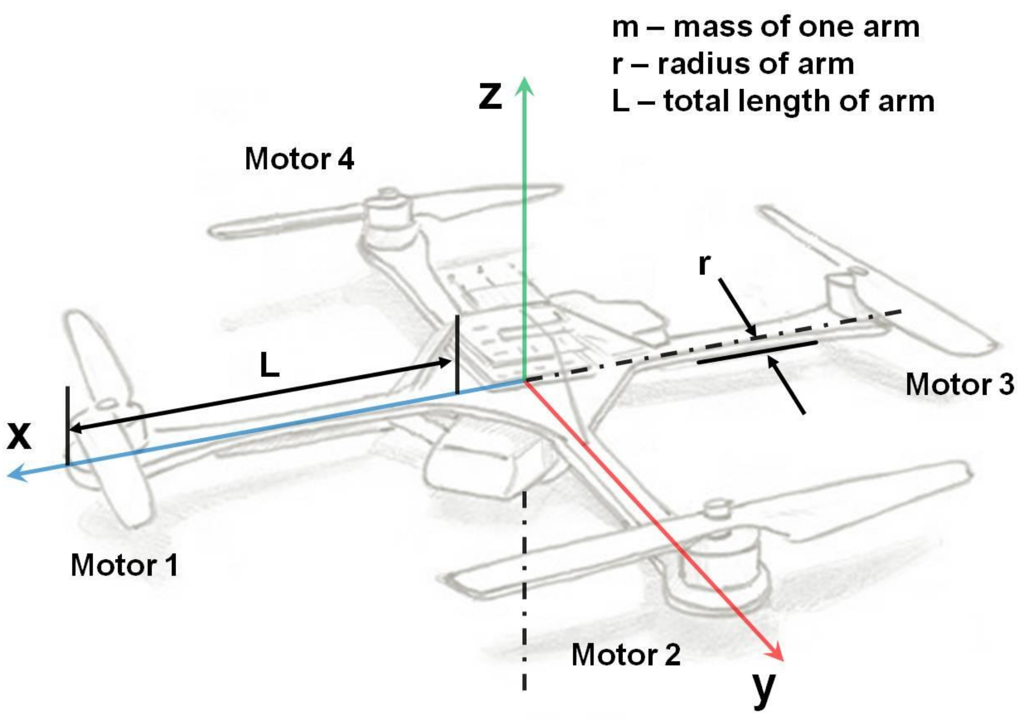

**ESC's**




**Central HUB**


m – mass of central hub
r – radius of central hub
H – total height of hub
z
Motor 4
H
r
Motor 3
x
Motor 1
Motor 2
y

<u>**Arms**</u>


m – mass of one arm
r – radius of arm
L – total length of arm
z
Motor 4
r
L
Motor 3
x
Motor 1
Motor 2
y