Overview:

Hardware:

Software:

This is a list of all the files that might be important. The more important files have short descriptions.

* 3dGrapher.py
* acrobot\_math.py
* acrobot\_state.py - A helper data structure for storing the position of an acrobat. (lists of size 4 or 5 are also used in older sections of the code)
* balance\_curve.csv
* build\_cython.sh - compiles lookup\_table\_hopper\_helper.pyx and tests it against the pure python version
* dampening.py
* hardware\_listener.py
* hardware\_lookup\_table\_generator.py
* hardware.py - Sends and receives data from the acrobat hardware. It should be interchangeable with ode\_interface.py (in theory)
* hardware\_simulation.py - Acts as a main class for all acrobat activities, **simulated or real**. A lot of parameters (such as the target state and whether ODE or the acrobat hardware should be used) can be controlled by changing the default values of the arguments of run().
* hopper2.py
* input\_box.py - Used in the graphical simulation
* interpolation\_consistancy\_checker.py - Checks lookup\_table\_hopper\_helper.pyx against the python version (for debugging).
* logger.py - Utility for writing csv files
* lookup\_table\_checker.py - Checks the accuracy of a lookup table against ODE
* lookup\_table\_generator\_interpolated.py - Creates lookup tables from logs generated by hardware\_lookup\_table\_generator.py
* lookup\_table\_generator.py - Creates lookup tables from ODE
* lookup\_table\_hopper\_helper.pyx - Cython
* lookup\_table\_hopper.py - Actually searches the lookup table for possible paths. This file has much better comments than the rest, so reading through it is probably better than a summary here.
* ode\_interface.py - A wrapper for ODE (the physics library used). It lets ODE and the actual hardware be used interchangeably
* parameter\_optimizer.py -
* parent\_log\_trimmed.csv
* pid.py
* setup.py - Compiles lookup\_table\_hopper\_helper.pyx
* simulation.py
* start\_parallel.bat
* stop, stop.py - An emergency stop for the hardware

The following files are previously generated lookup tables (both from the hardware and simulation). The parameters following table and number are the dimensions of the table and maximum/minimum angles stored. Table names are parsed in the function get\_table\_name in lookup\_table\_hopper.py

* table114\_[1.2, 1.8, -0.5, 0.5, -0.8, 0.8, -1.25, 1.25, -1, 1, 0.025, 0.025, 0.25, 10, 0, 0].npy
* table115\_[1.45, 1.65, -0.2, 0.2, -0.8, 0.8, -1.25, 1.25, -1, 1, 0.01, 0.025, 0.25, 10, 0, 0].npy
* table116\_[1.45, 1.65, -0.2, 0.2, -0.8, 0.8, -1.25, 1.25, -1, 1, 0.01, 0.025, 0.25, 10, 0, 0].npy
* table117\_[0.8726646259971648, 2.0943951023931953, -0.8726646259971648, 0.8726646259971648, -2.0, 2.0, -4.0, 4.0, -100, 100, 0.1, 0.2, 10, 10, 0, 0].npy
* table200\_[0.8726646259971648, 2.0943951023931953, -0.8726646259971648, 0.8726646259971648, -2.0, 2.0, -4.0, 4.0, -80, 80, 0.15, 0.25, 20, 10, 0, 0].npy
* table27\_[0, 6.283185307179586, -3.141592653589793, 3.141592653589793, -2.0, 2.0, -4.0, 4.0, -40, 40, 0.2, 0.2, 2].npy
* table28\_[0, 6.2831853071795862, -3.1415926535897931, 3.1415926535897931, -1.5, 1.5, -4.0, 4.0, -60, 60, 0.29999999999999999, 0.29999999999999999, 4].npy
* table300\_[0, 6.283185307179586, -3.141592653589793, 3.141592653589793, -2.0, 2.0, -3.0, 3.0, -1.0, 1.0, 0.1, 0.1, 0.5, 10, 0, 0].npy

Somewhat important log files

* parent\_log\_balance\_curve.csv - A list of balanced points
* parent\_log\_balance\_curve\_left.csv - A list of points with an elbow angle slightly smaller than balanced
* parent\_log\_balance\_curve\_right.csv - A list of points with an elbow angle slightly greater than balanced