Rapid Controller Prototyping

Closed-Loop Helicopter Controller
[Software Version 4]

Overview

During this exercise you will take the parameters for the controller design you have developed in MATLAB and load then into a pre-build controller developed in LabVIEW.

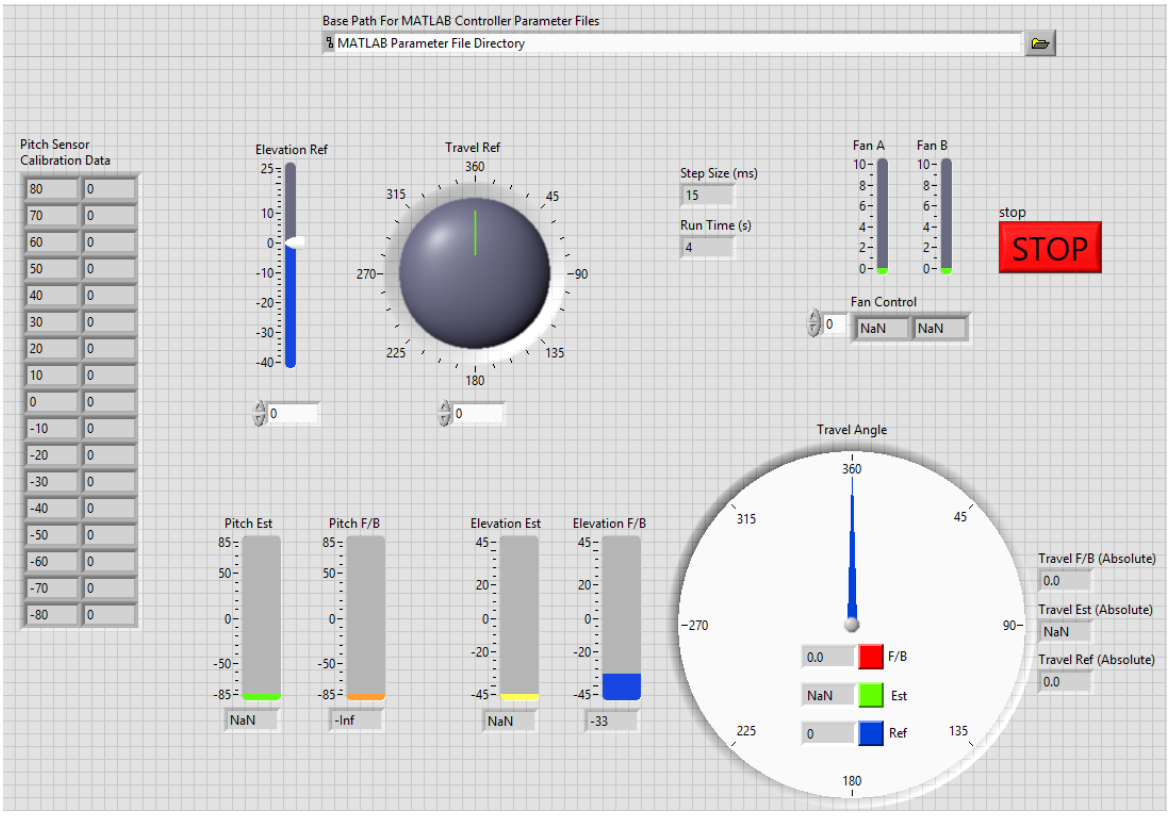
You may modify this VI to provide you with any results you may need for your final assessment, if you feel that this would be useful.

Closed-Loop Helicopter Controller

Description:
A closed loop controller has been developed in LabVIEW, using the same control structure you have been simulating in MATLAB/Simulink.

The MATLAB script you have been using to develop your controller; produces a set of parameter files for your controller and observer designs. These parameter files are read by the LabVIEW controller VI, to implement the controller values in the prebuilt controller prototype.

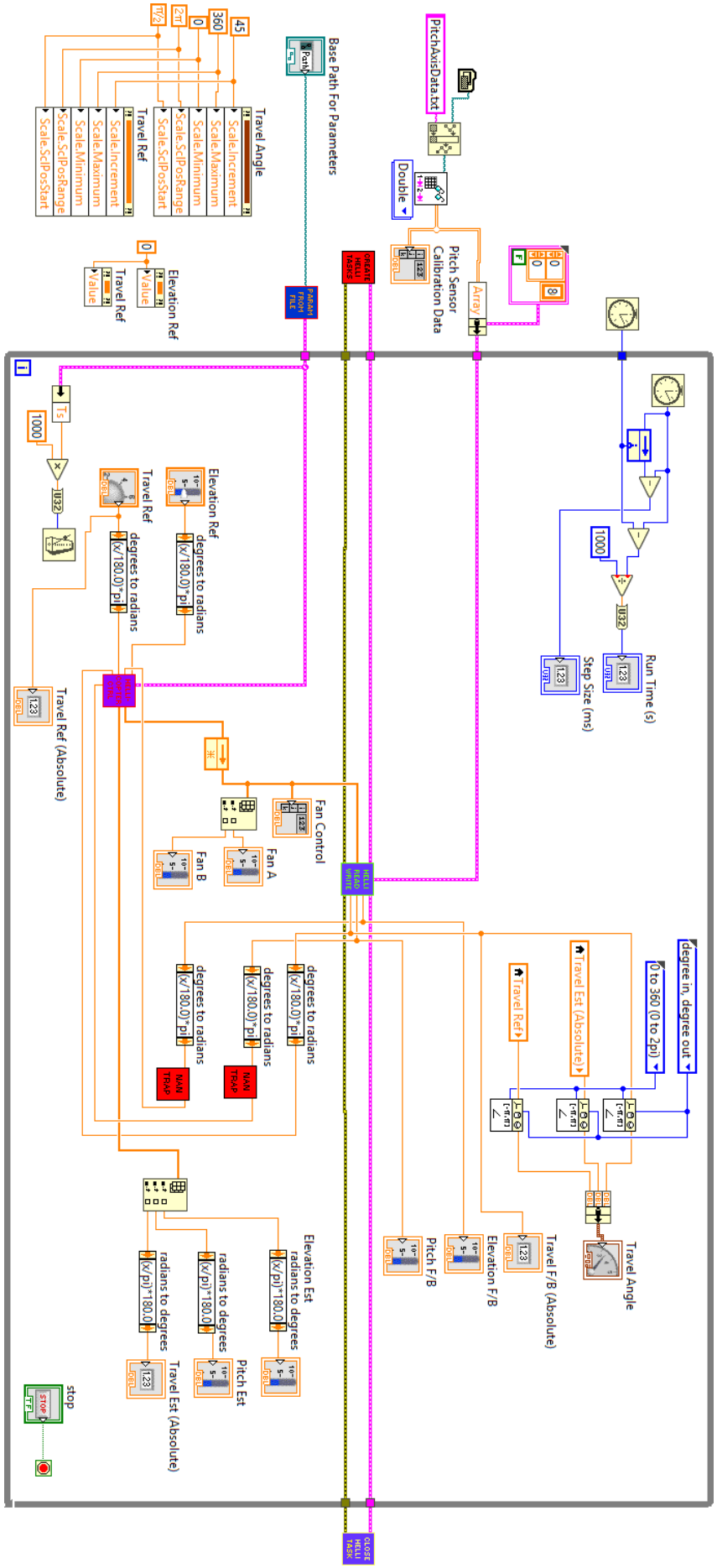
The closed-loop controller VI is implemented using static reference inputs for the elevation angle and the travel angles, entered through the 'Elevation Ref' slider control and Travel Ref' dial control, in the upper left quadrant of the front panel shown below.



The front panel also provides indicators for the actual measured values, from the helicopter system, and the estimated values, from the Kalman filter you have designed.

The actual and estimated travel angles are displayed in the gauge indicator, in red and green respectively, along with the travel reference in blue. The actual and estimated elevation angles are displayed on the blue and yellow tank indicators, respectively. The green tank indicator provides an estimation of the pitch angle, provided by the Kalman filter and the orange tank indicator provides the pitch angle measurement. The two green tank indicators, above the travel gauge indicator, provide an indication of the control output to the two fans.

The table in the left of the front panel contains the pitch sensor calibration data, read from the Pitch Calibration data file - PitchAxisData.txt.


Base Path For Parameters
PitchAxisData.txt
Double
Pitch Sensor Calibration Data
Array
Travel Angle
Travel Ref
Elevation Ref
Value
Scale.SclPosStart
Scale.SclPosRange
Scale.Minimum
Scale.Maximum
Scale.Increment
Ts
1000
Step Size (ms)
Run Time (s)
degrees to radians
Travel Ref (Absolute)
Fan Control
Fan A
Fan B
NAN TRAP
Travel Est (Absolute)
Travel F/B (Absolute)
Elevation F/B
Pitch F/B
Travel F/B
degree in, degree out
0 to 360 (0 to 2pi)
radians to degrees
Elevation Est
Pitch Est
stop

The block diagram for the closed loop helicopter model is provided on the previous page. The implementation of the VI is complete, and nothing needs to be edited on the block diagram.

When you run the closed-loop VI, the system will automatically load the controller parameter files and the pitch sensor calibration data, which you have created in MATLAB and the open loop VI, respectively, then the helicopter should start to move to the reference point you have entered.

There are three new sub-Vis implemented in this program:

1. **Get Model Parameters From File.vi**

   This sub-VI imports the controller and Kalman filter parameters from the files you have generated at the end of your MATLAB design script.

   Look inside the sub-VI to see how this has been achieved.

2. **Helicopter Controller.vi**

   This sub-VI implements the same controller that you have designed in MATLAB, using the parameters imported using **Get Model Parameters From File.vi**.

   Look inside the sub-VI to see how this has been achieved.

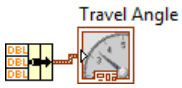
3. **NaN Trap.vi**

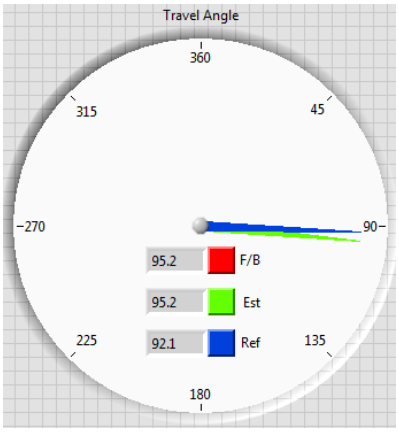
   Occasionally a minor error occurs in the system that causes a NaN, (Not a Number) value to read from the myDAQ module, but does not cause an error to be flagged on the DAQmx error line. This NaN value can cause problems with the controller; therefore, this block is needed to prevent the NaN value from reaching the controller.

   Look inside the sub-VI to see how this has been achieved.

4. **Other elements of interest**

   a. **Modified Dial Gauge**



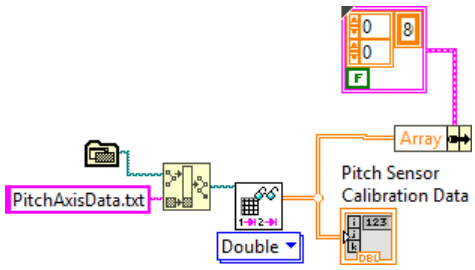


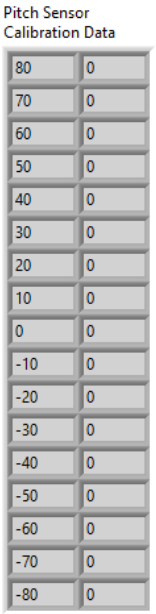
The dial gauge for the closed loop helicopter controller VI has been modified to allow 3 travel angle variables to be monitored: feedback value, (red), estimated value, (green), and the travel reference, (blue), as shown in the right-hand figure, above.

The data written to the multi-needle gauge is a cluster of double precision numbers, shown above left-hand figure, as opposed to a single value, as used in the open loop helicopter controller VI

b. Pitch Calibration

The calibration of the pitch sensor is taken from the pitch calibration data file - PitchAxisData.txt, written using the open loop control VI. The pitch calibration is read into the VI, using the code segment shown below left, and displayed on the front panel in a table, similar to that shown below right.





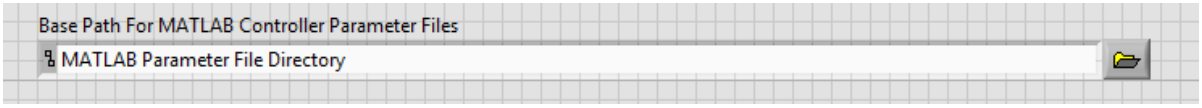
The pitch calibration data, measured in the open loop controller exercise, is specific for each helicopter kit, so you must use the data that YOU have measured on your own piece of equipment. Furthermore, this calibration is sensitive to the temperature of the pitch angle sensor, and may drift over time if the temperature of the device changes. (For example, you may notice differences in the pitch angle calibration at the start of working in a warm computer room, after walking into the University on a cold day, compared to later in that session, when

the helicopter has warmed to room temperature.) It is, therefore, recommended that you recalibrate the pitch angle sensor from time to time, using the methods in the open loop control document.

c. Base Path for the MATLAB controller files

To run the closed loop VI, you must first enter the base path for the controller parameter files that were created by your heli.m script. This must be done using the "Base Path For MATLAB Controller Parameter Files" control located at the top of the front panel.

To enter the base folder path where these .txt files are located, you can either type the folder path in to the text input of the control, or click on the folder icon, to the right of the text input, and navigate to the base folder.



To save this value for the folder base location, you must first set the input value for the file dialogue to default. To achieve this, you must right click on the control and select **Data Operations» Make Current Values Default**, before saving your closed loop VI.

Procedure:

1. Download the close-loop Controller .ZIP file from the MOLE site.
2. Save the contained VIs into the same directory that you have saved the VIs for the open loop control program.

   NOTE: The open loop software and the closed loop software both use the PitchAxisData.txt calibration file. To ensure both the open loop software and closed loop software is sharing the same PitchAxisData.txt, the VIS for both programs should be located in the same directory.

3. Open the **Main - Hardware and Controller 3.vi**
4. Modify the Base Path For MATLAB Controller Parameter Files, as described above, and if you wish, set the folder path to default.
5. Save the VI.
6. Run the VI, and assess the performance of your controller.

   Your helicopter should now be flying in a controller manner.

7. At any point, you may modify the controller design parameters and rerun your MATLAB script to update the LabVIEW controller model.

At any point you can modify the LabVIEW VI to provide you with any results you may need for your final assessment, if you feel that this would be useful.