SimpleHelo Simulation:

After you unzip the files to a local folder on your computer. You should see the following MATLAB m-files:

SimpleHelo.m
SimpleAero.m
eqnmot.m
SimpleHeloSimInit.m
rk4.m
linearize.m
trimmer.m
table_lookup.m
test_input.m
SimpleHeloSim.slx

Setting up to run SimpleHelo in MATLAB:

Run MATLAB and set the "Current Folder" to the folder where the m-files are located. I encourage you to examine the code in each of the files and gain some understanding of them. Most of the m-files are MATAB functions. ***SimpleHeloSimInit.m***, is a MATLAB script that initializes the model and trims it at specified forward speed. When you first download the simulation, run this script. You should be prompted with "Enter trimmed forward speed in ft/sec:". Enter 0.01 (note that you cannot generally trim a helicopter simulation at exactly 0 airspeed). If everything is working correctly you should see some text on the screen similar to this:

```
>> SimpleHeloInit
Enter trimmed forward speed in ft/sec:0.01
Iteration     Max. Error
1          2329.7582
2          1207.258
3          626.3412
4          323.1628
5          165.2821
6          83.8293
7          42.2584
8          21.2222
9          10.6353
10          5.3238
11          2.6634
12          1.3321
13          0.66614
14          0.33309
15          0.16655
16          0.083275
```

17	0.041637
18	0.020818
19	0.010409
20	0.0052045
21	0.0026022
22	0.0013011
23	0.00065053
24	0.00032526

Successful trim

To run simulations over time you can use the Simulink Model: SimpleHeloSim.slx, and provide canned inputs, as was shown in class.

Summary of Model

The state vector, control vector, and force/moment vector are expected to be in the following order for all of the functions:

$$\mathbf{x} = \begin{bmatrix} u & v & w & p & q & r & \phi & \theta & \psi & x & y & z \end{bmatrix}^T$$
$$\mathbf{u} = \begin{bmatrix} \theta_{1c} & \theta_{1s} & \theta_0 & \theta_{0TR} \end{bmatrix}^T$$
$$\mathbf{FM} = \begin{bmatrix} \mathbf{F}^T & \mathbf{M}^T \end{bmatrix}^T = \begin{bmatrix} X & Y & Z & L & M & N \end{bmatrix}^T$$

The control angles in **u** should be in units of degrees. The angles and angular rates in the state vector have units of rad and rad/sec. The velocities and positions in the state vector have units of ft/sec and ft. The forces and moments have units of lbs and ft-lbs.

Summary of Files in SimpleHelo MATLAB simulation:

SimpleAero.m: Function that models basic aerodynamics of a helicopter with single main rotor, tail rotor, fuselage, and horizontal stabilizer as presented in class. The function is of the form:

```
[FM,y]=simpleaero(x,u,constants)
```

Where x is the state vector, u is the control vector, and constants is a data structure containing aircraft and simulation constants. FM is a vector consisting of the body axis forces and body axis moments acting at the aircraft CG. The vector y is an output vector, in this containing the accelerometer measurements at the helicopter CG.

eqnmot.m*:* This function models the rigid body equations of motion. The function is of the form:

```
[xdot]=eqnmot(x,FM,constants)
```

where x, constants, and FM have the same definition as above, and xdot is the state derivative.

simplehelo.m: State variable representation of the helicopter flight dynamics. The function basically just calls ***simpleaero*** and then ***eqnmot*** to calculate the state derivatives and output. The function is of the form:

```
[xdot,y]=simplehelo(x,u,~,constants)
```

Note that xdot would be the 3rd argument, but it is not in this model. The argument placeholder is kept in the function call for consistency with other sim models used in this course.

trimmer.m: Used to calculate trim (equilibrium). The function is of the form:

```
[x0,u0,itrim]=trimmer('SimpleHelo',x0_guess,u0_guess,targ_des,constants);
```

where x0, u0 are the trimmed state and control vector and itrim is a flag that is returned as 1 if the trim was successful or 0 otherwise. You need to provide initial guess for the state and control vectors x0_guess,u0_guess as well as the desired values for the "trim targets" (targ_des). This represents the desired values of the state derivatives and output. Your input argument 'SimpleHelo' is telling the algorithm to trim the dynamic function SimpleHelo (we can use the exact same trim algorithm on other dynamic models of aircraft).

The solve will find components of the trim variable vector, [x0;u0](TRIMVARS), in order to meet a subset of the trim constraints: [xdot;y](TRIMTARG)=targdes(TRIMTARG). The constants fields TRIMVARS and TRIMTARGS are index arrays that specify which of the trim variable vector will be adjusted and which of the trim target constraints will be driven to the desired values provided in targdes.

linearize.m: Generates a linearized model of a non-linear dynamic function. The function is of the form:

```
[F,G,M,A,B,C,D]=linearize('SimpleHelo',x0,u0,constants)
```

Where F,G,M describe the state space model of the form:

$$M\dot{x} = Fx + Gu$$

While A, B, C, and D are the linearized state space matrices for:

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

The linear models are linearized about the equilibrium defined by x0, u0. We will discuss this more later in the course. You need this file in your local directory because this function is used by trimmer. Note that MATLAB has another built-in function called linearize which we are NOT using. MATLAB will use this one if it is in your current working folder.

***table_lookup.m*:** A simple table lookup function.

SimpleHeloSim.slx: This is the SIMULINK model used to simulate the vehicle dynamics.