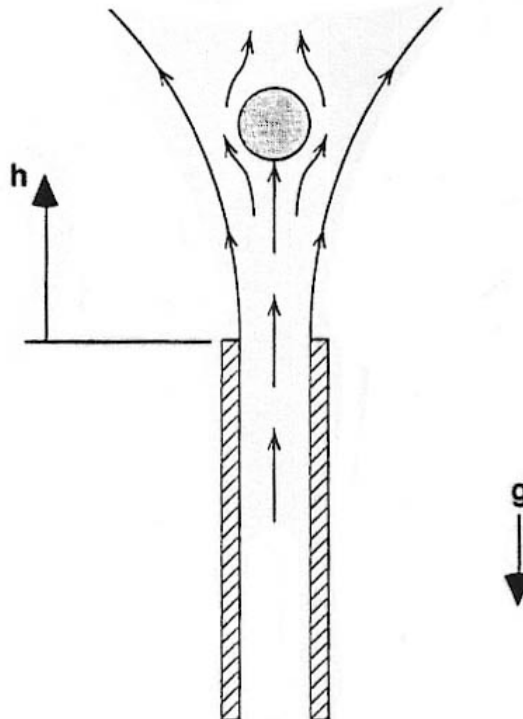


The Floating Ball Problem
AAE 421 Homework Assignments 4, 5, and 6

A ball is balanced on a spreading jet of air. The speed of the jet decreases linearly with height, h , given by $V = V_0 - V_1 h$, where V_0 and V_1 are positive constants. The air density in the jet equals ρ , and the drag coefficient, C_d , of the ball is based on projected area, S . The ball has mass m . When the ball is placed in the air jet, the transient motion of the ball will eventually settle out leaving the ball at an equilibrium or reference height, h_R , where the weight and drag are in balance.

PART 1 Homework #4, Due Wednesday 9/12/07

- Write a nonlinear differential equation governing the vertical motion of the ball in the jet of air.
- Linearize this nonlinear differential equation about the reference height, h_R .
- Find the period of the vertical motion of the ball about its reference height.



PART 2 Homework #5, Due Friday 9/14/07

Create your own Simulink model (from scratch) to simulate the nonlinear dynamics of the Floating ball

Write a Simulink model called "floatingball.mdl." Use an S-function block from the Simulink Block Library for nonlinear elements. Name the underlying S-function floatball. Use the Simulink model we developed in the pendulum problem as a guide (see pendulum.mdl of the course web site

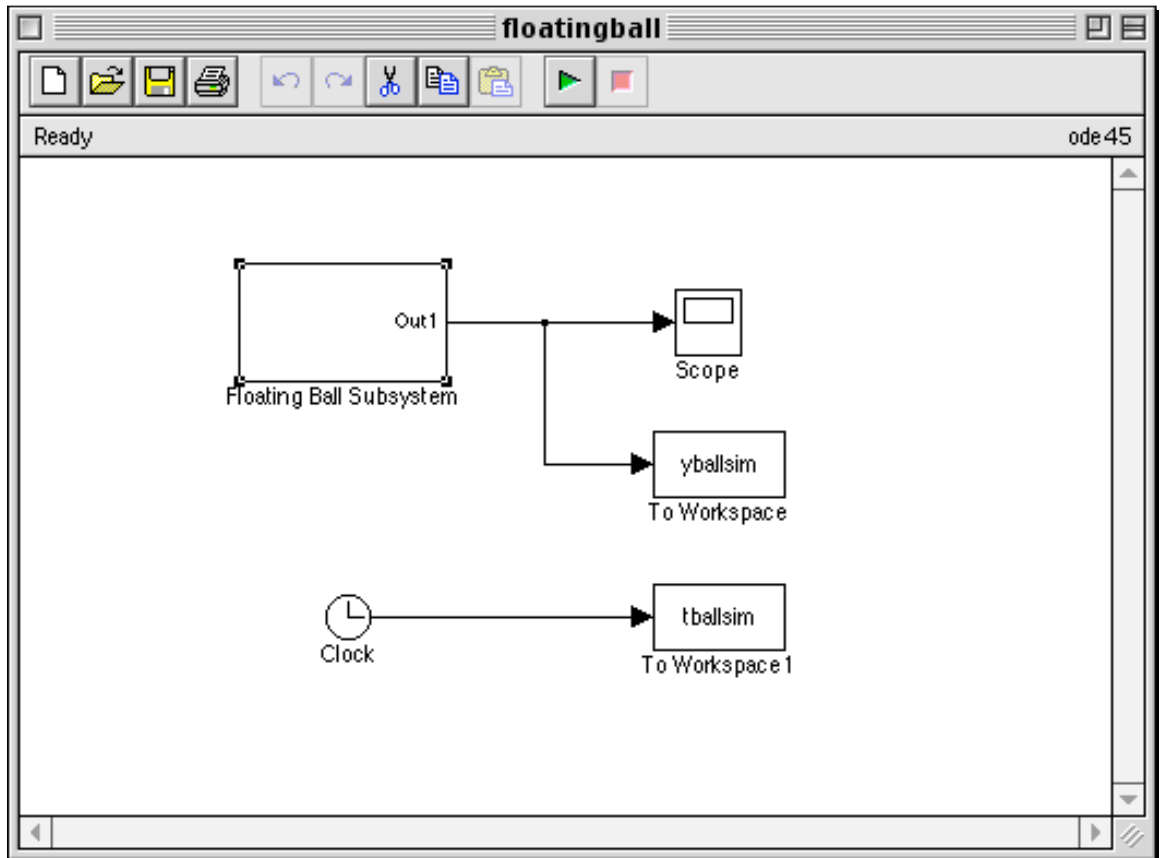
http://cobweb.ecn.purdue.edu/~andrigan/Courses/AAE421_Fall_2007/AAE421_Buffer_F07/Pendulum/

Write an m-file called floatball.m that contains the differential equation model of the foating ball in Simulink protocol. Use as a starting point the m-file called pendeom.m from the pendulum problem (see course web site).

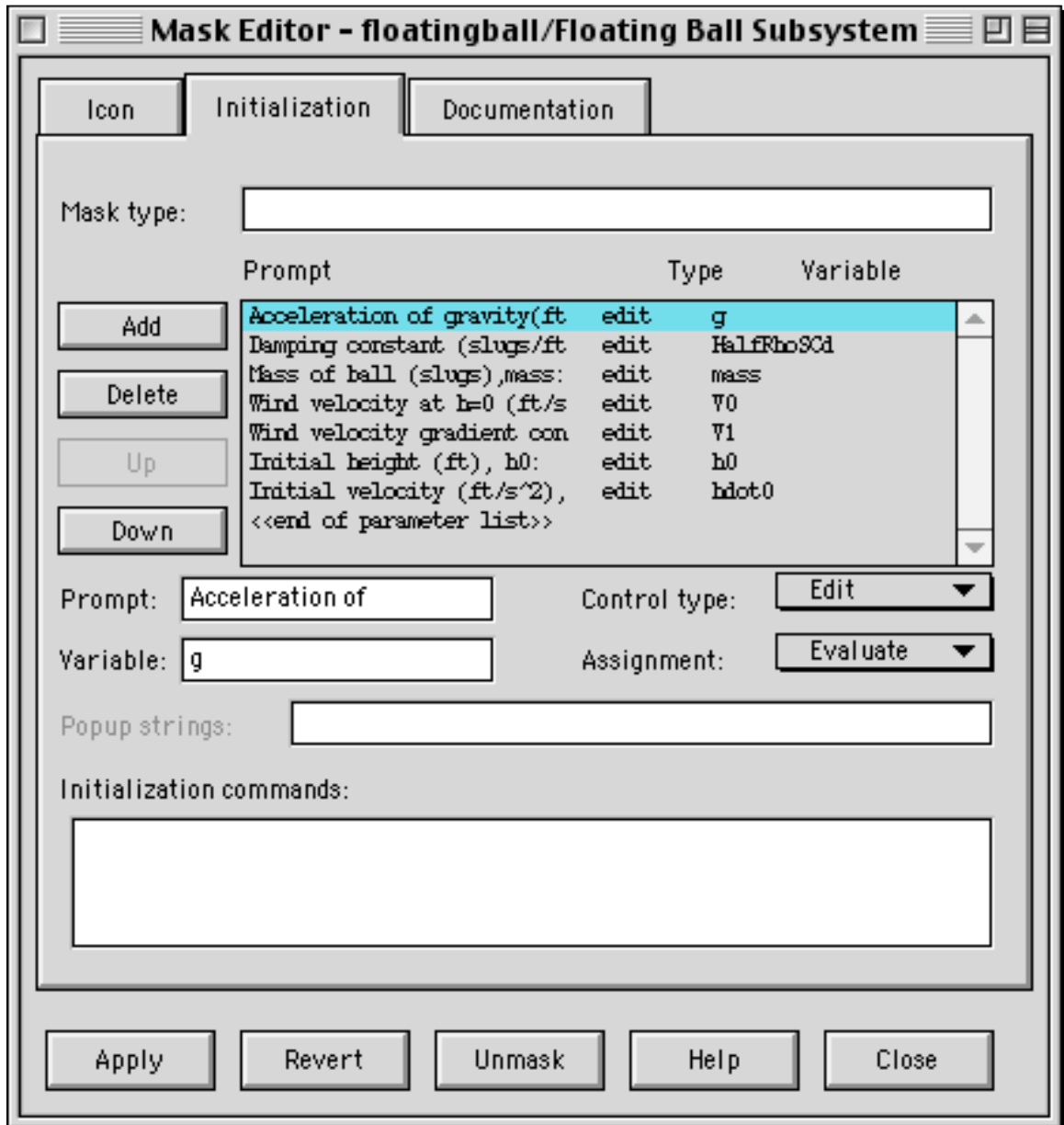
Select the S-function block. From the edit menu Create a Subsystem of the the S-function block and call the resulting block Floating Ball Subsystem. Click on the subsystem. Delete the input port and the input arrow. The block should have no inputs and one output.

Add a Scope block from the Sinks window of the Simulink Block Library on the output. Connect up the output from the Floating Ball Subsystem to the Scope. And add a To Workspace block from the Sinks window of the Simulink Block Library. . Connect up the output from the Floating Ball Subsystem to the To Workspace block. Name the To Workspace block yballsim. Add a clock from the Sources window of the Simulink Block Library. Connect the clock to the input of another To Workspace block. Name the To Workspace block tballsim.

When you are done, your Simulink window should look like this.



Click on the Floating Ball Subsystem and select Mask Subsystem from the edit menu. Select Edit Mask from the Edit menu. Click the Initialization tab and add Prompts and Variables until you get the following window in the mask Editor.



Masking gives a convenient place in the Simulink workspace to define parameters and give numerical values of those parameters.

If you double click on the Floating Ball Subsystem the following window should pop up.

Block Parameters: Floating Ball Subsystem

Subsystem (mask)

Parameters

Acceleration of gravity (ft/s²), g:
32.2

Damping constant (slugs/ft), HalfRhoSCd:
.5*0.00213*(pi*(1.5/12)²/4)*0.445

Mass of ball (slugs), mass:
0.000152

Wind velocity at h=0 (ft/s), Y0:
39.165

Wind velocity gradient constant (1/s), Y1:
51.816

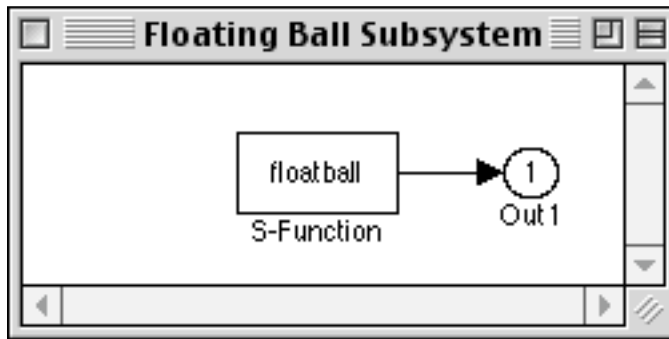
Initial height (ft), h0:
0.5

Initial velocity (ft/s²), hdot0:
0

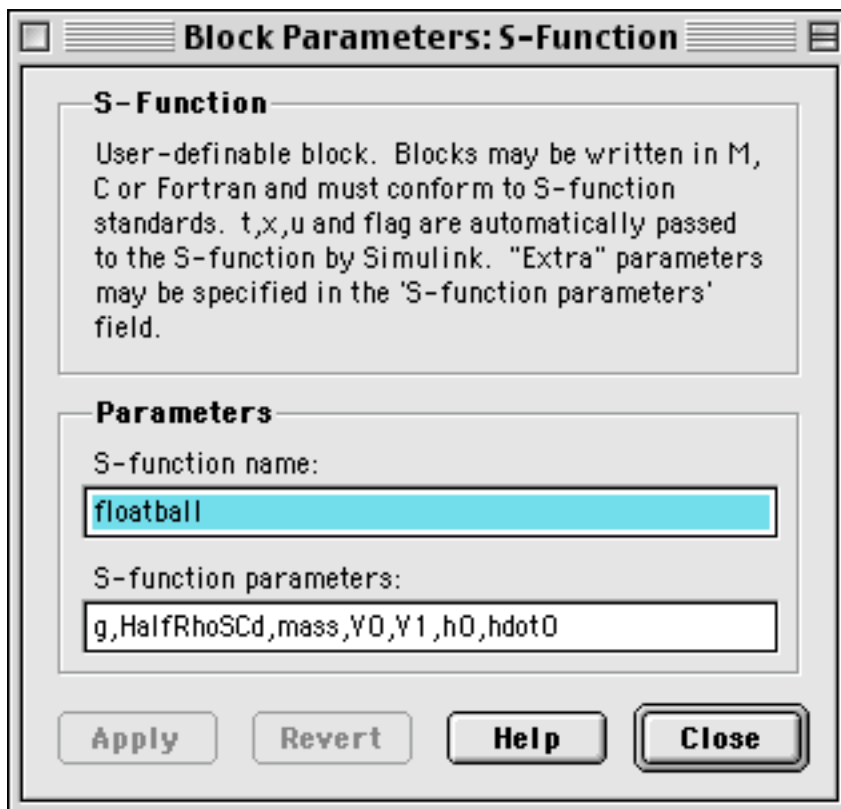
Apply Revert Help Close

The figure above shows all the numerical constants needed for the nonlinear simulation. If you don't get this, edit the mask for the Floating ball Subsystem until you do.

Select the Floating Ball Subsystem. If you “Look Under the Mask” of the Floating ball Subsystem you should see the following window pop up.



If you double click on the floatball S-function the following window should pop up.



The m-file called floatball.m must be written to accept the parameters that are required to form the nonlinear differential equations. I suggest that the m-file floatball.m start as follows.

```
function [sys,x0,str,ts] = floatball(t,x,u,flag,g,HalfRhoSCd,mass,V0,V1,h0,hdot0)
% S-file for floating ball problem.
% This is an S-file subsystem that models a ball floating over a hair dryer.
% g is the acceleration of gravity.
% mass is the mass of the ball that forms the massive part of the pendulum.
% h0, hdot0 are the initial condition on h and hdot.
```

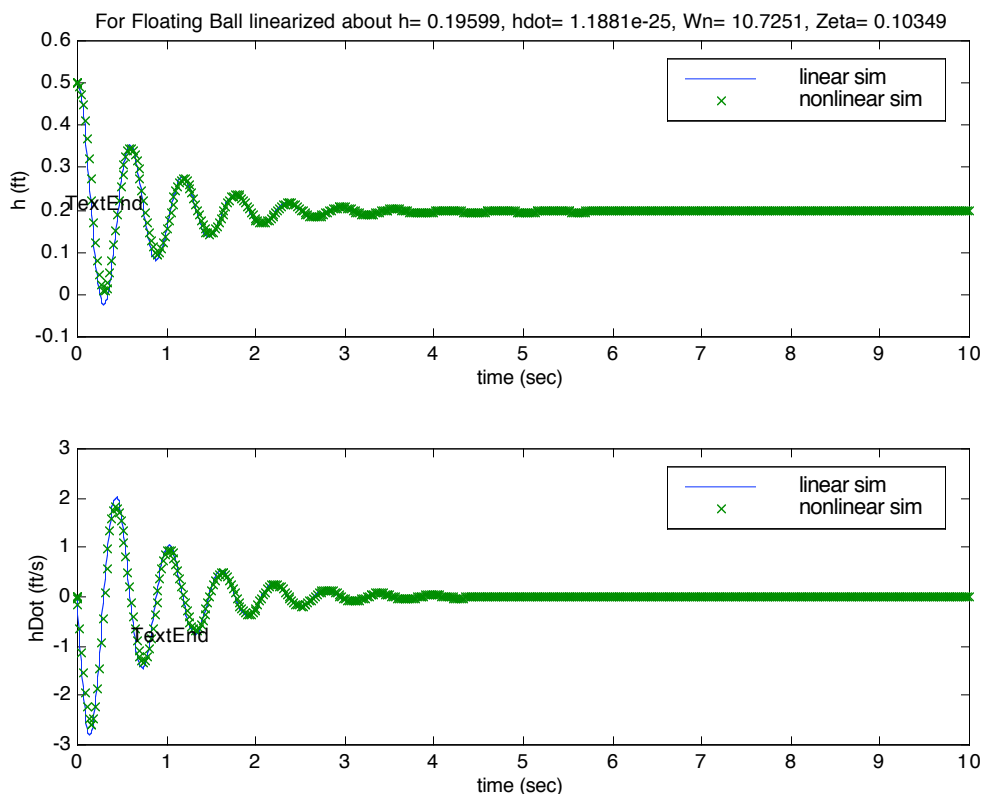
PART 3 Homework #6, Due Monday 9/17/07

Write a MATLAB script called FloatBallAnal.m to analyze the linearized behavior of the floating ball problem. Pattern it after the script we used in the pendulum problem (PendAnal.m from the class web site).

Be careful to use the small perturbation initial condition for the linearized simulation. You must subtract off \mathbf{h}_R from initial height $h(0)=0.5$ to get the small perturbation initial condition. Also remember to compute the total approximate solution by adding the reference \mathbf{h}_R to the small perturbation linear solution for the height. We did not have to do this in the pendulum problem because the reference solution was zero.

Overplot the total approximate solution and the output from the nonlinear simulation (stored matrix yballsim which exists in the MATLAB workspace after the Simulink simulation is run).

FloatBallAnal.m should plot both the nonlinear and linearized simulation results as in a figure that looks like this.



How do you're the results from linearized computer model from PART 3 agree with what you found in PART 1c?