Connor McGarty, cmcgarty - HW01: P1.7 from Chapra Text

File: cmcgarty_EE254_HW01_P1_7.m

Submission Date: Tuesday, January 28, 2020 - 1/28/20

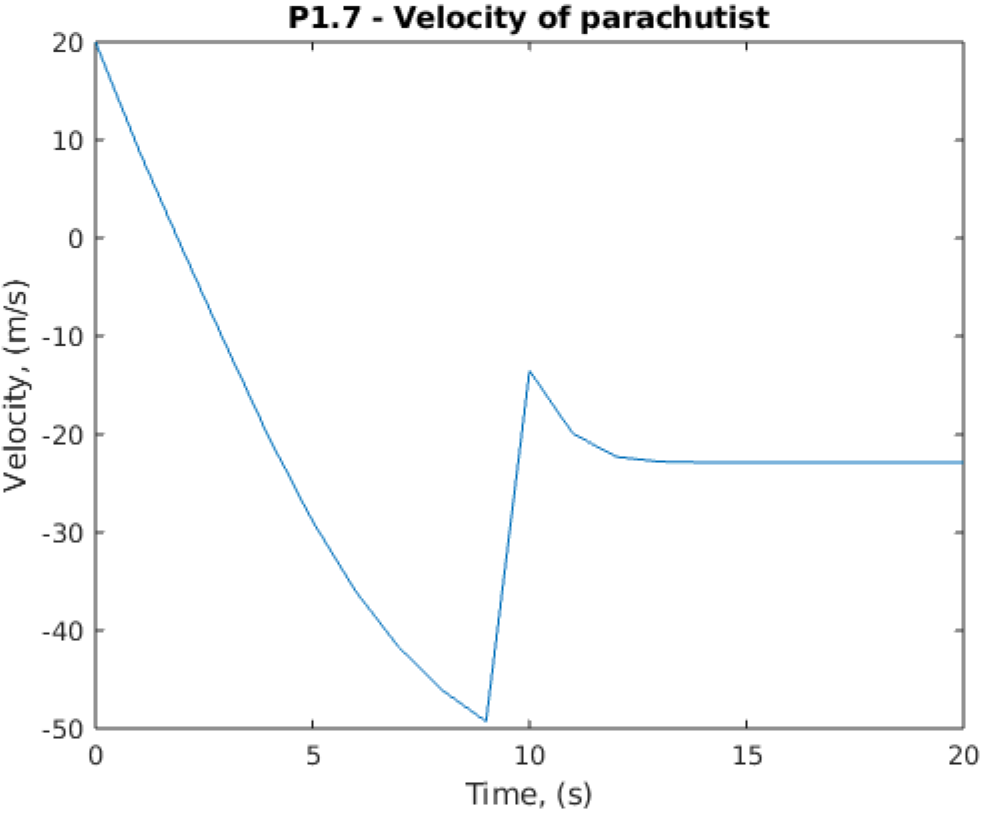
Problem, P1.7: For the second-order drag model* (Eq. 1.8), compute the velocity of a free-falling parachutist using Euler's method for the case where m = 80kg and c_d = 0.25 kg/m. Perform the calculation from t = 0 to 20 seconds with a step size of 1 s. Use an initial condition that the parachutist has an upward velocity of 20 m/s at t = 0. At t = 10s, assume that the chute is instantaneously deployed so that the drag coefficient jumps to 1.5 kg/m.

- Eq 1.8, second-order drag model:

$$\frac{dv}{dt} = g - \frac{c_d}{m} * v^2$$

Solution

```
clear;clc;close all;

v_init = +20; % m/s - define +v as upwards direction and -v as
 downwards

t_start = 0; % s
t_stop = 20; % s
t_delta = 1; % s
t_vector = zeros(1, ((t_stop - t_start) / t_delta) + 1); % s - pre-
init for speed
v_vector = zeros(1, ((t_stop - t_start) / t_delta) + 1); % m/s
v_vector(1) = v_init;

mass_jumper = 80; % kg
c_d_free_fall = 0.25; % kg/m
c_d_parachute = 1.5; % kg/m
g = -9.8; % m/s^2 - acceleration due to gravity

done = false; % loop flag
t_current = t_start;
v_previous = v_init;
iterator = 1;

while (done == false)
    iterator = iterator + 1;
    t_current = t_current + t_delta;
    t_vector(iterator) = t_current;
    if (t_current < 10) % jumper deploys parachute a t = 10 seconds
        c_d = c_d_free_fall;
    else
        c_d = c_d_parachute;
    end
    % Euler's method to estimate velocity of jumper
    slope = g - ((c_d / mass_jumper) * (abs(v_previous) .*
 v_previous));
    v_current = v_previous + (slope * t_delta);
    v_vector(iterator) = v_current;
    v_previous = v_current;
    if t_current >= t_stop
        done = true;
    end
end

plot(t_vector, v_vector);
xlim([0, 20]);
xlabel('Time, (s)');
ylabel('Velocity, (m/s)');
title('P1.7 - Velocity of parachutist');
```



Published with MATLAB® R2019b