Table of Contents

Clean

```
close all; clear; clc;
```

Problem 1.1

```
pos = [500; 300; -1655];
euler_angles = [6; 9; -75] .* pi/180;
vel = [21; -1; 4];
omega = [0.2; -1.4; 4.8] .* pi/180;
wtilde = [0, -omega(3), omega(2); omega(3), 0, -omega(1); -omega(2),
omega(1), 0];

full_state = [pos; euler_angles; vel; omega];

vdot = [0.1;-0.06; -0.4];
yaccel = [9.75; 1.62; 0.67];

% Calculate what the accelerometer should see
sb_true = vdot + wtilde*vel - TransformFromInertialToBody([0; 0; 9.81],
euler_angles);
```

Problem 1.2

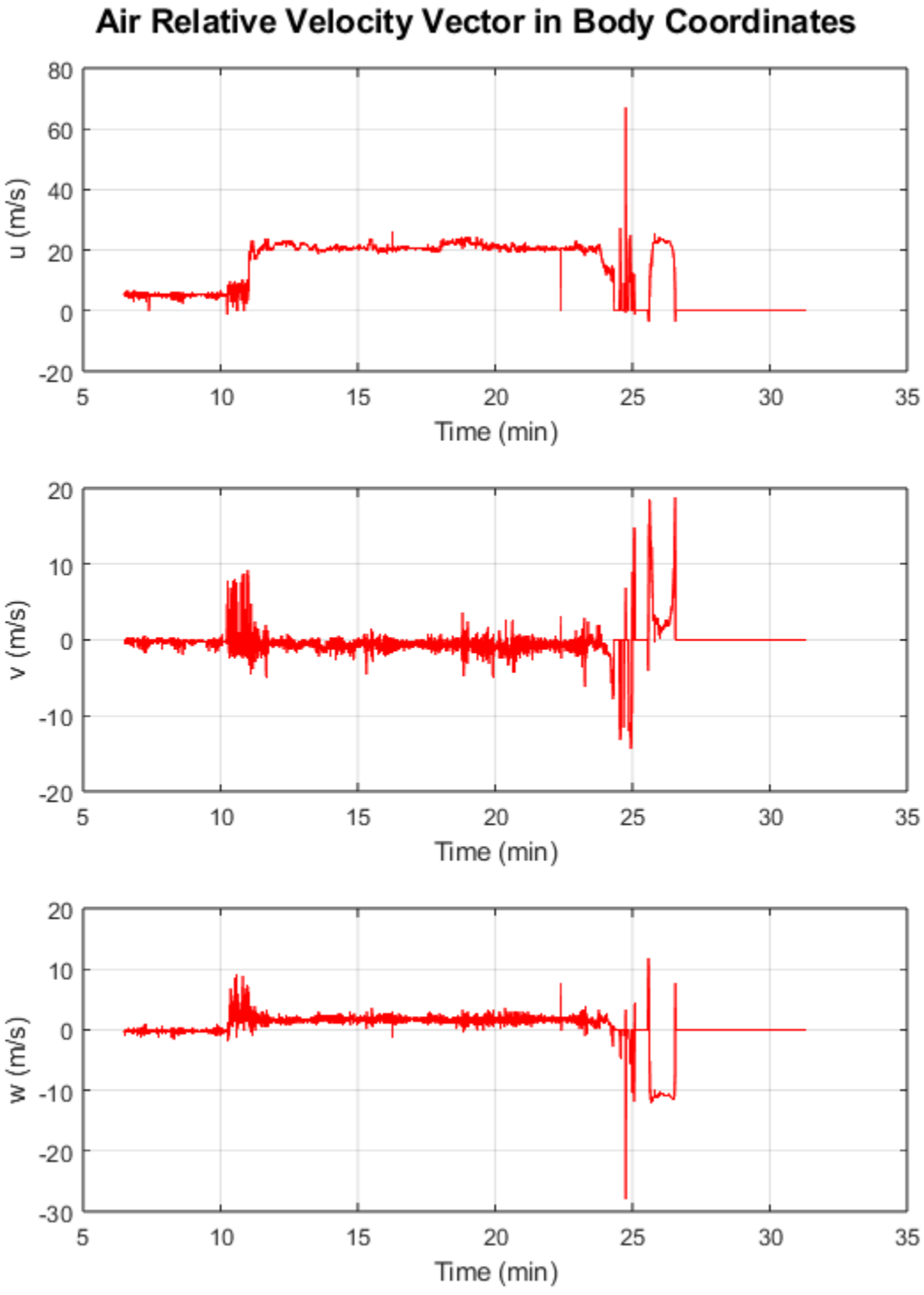
```
wind_hat = [12.2; -0.2; 2.3];
wind_true = [0; 6; 0];

% Check airspeed
air_rel_hat = vel - wind_hat;
air_rel_true = vel - TransformFromInertialToBody(wind_true, euler_angles);

airspeed_hat = norm(air_rel_hat);
airspeed_true = norm(air_rel_true);

% Check sideslip
beta_hat = asin(air_rel_hat(2) / airspeed_hat) * 180.0/pi;
beta_true = asin(air_rel_true(2) / airspeed_true) * 180/pi;

% Check angle of attack
aoa_hat = atan(air_rel_hat(3) / air_rel_hat(1)) * 180/pi;
aoa_true = atan(air_rel_true(3) / air_rel_true(1)) * 180/pi;
```

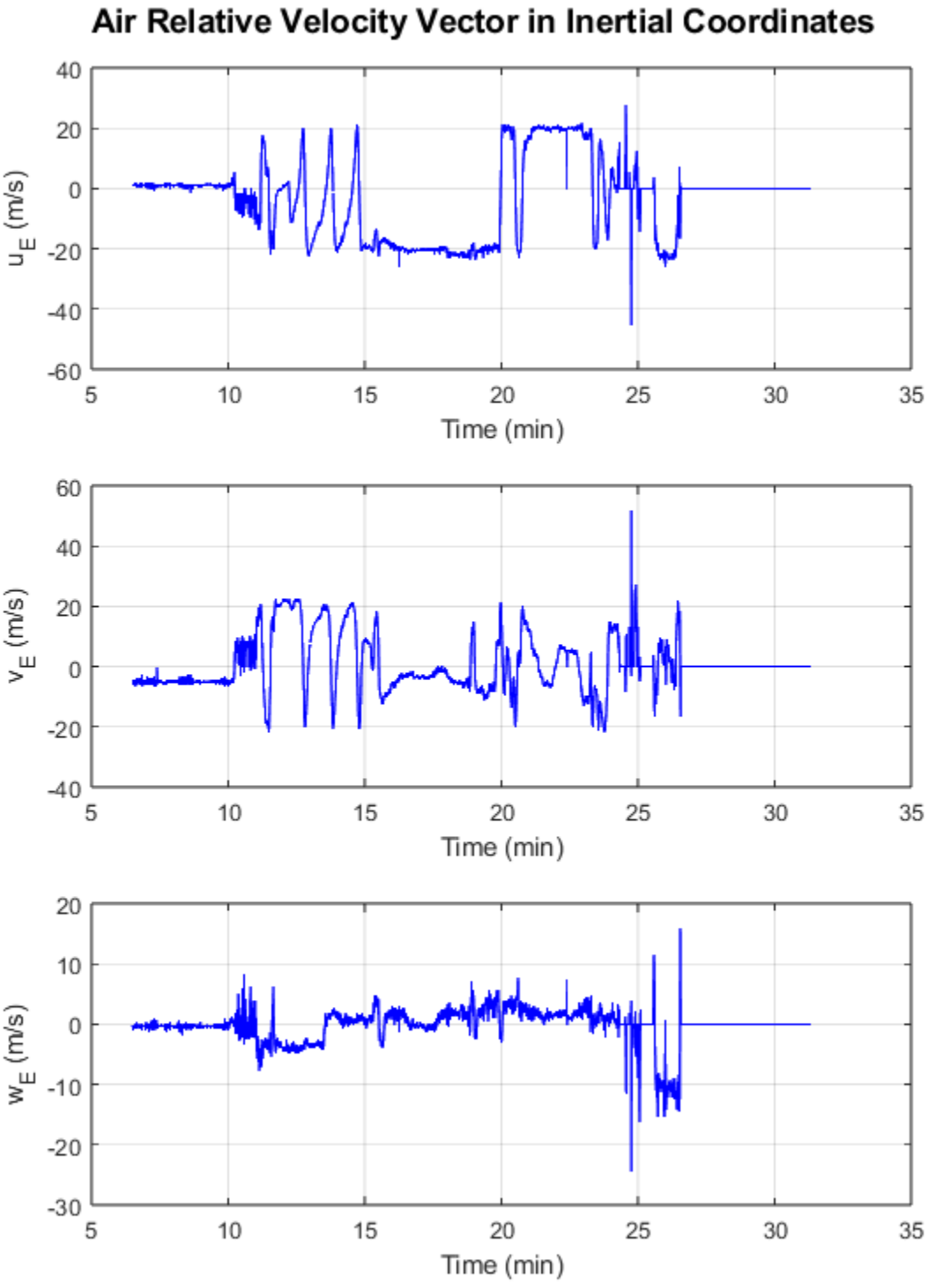
Problem 2

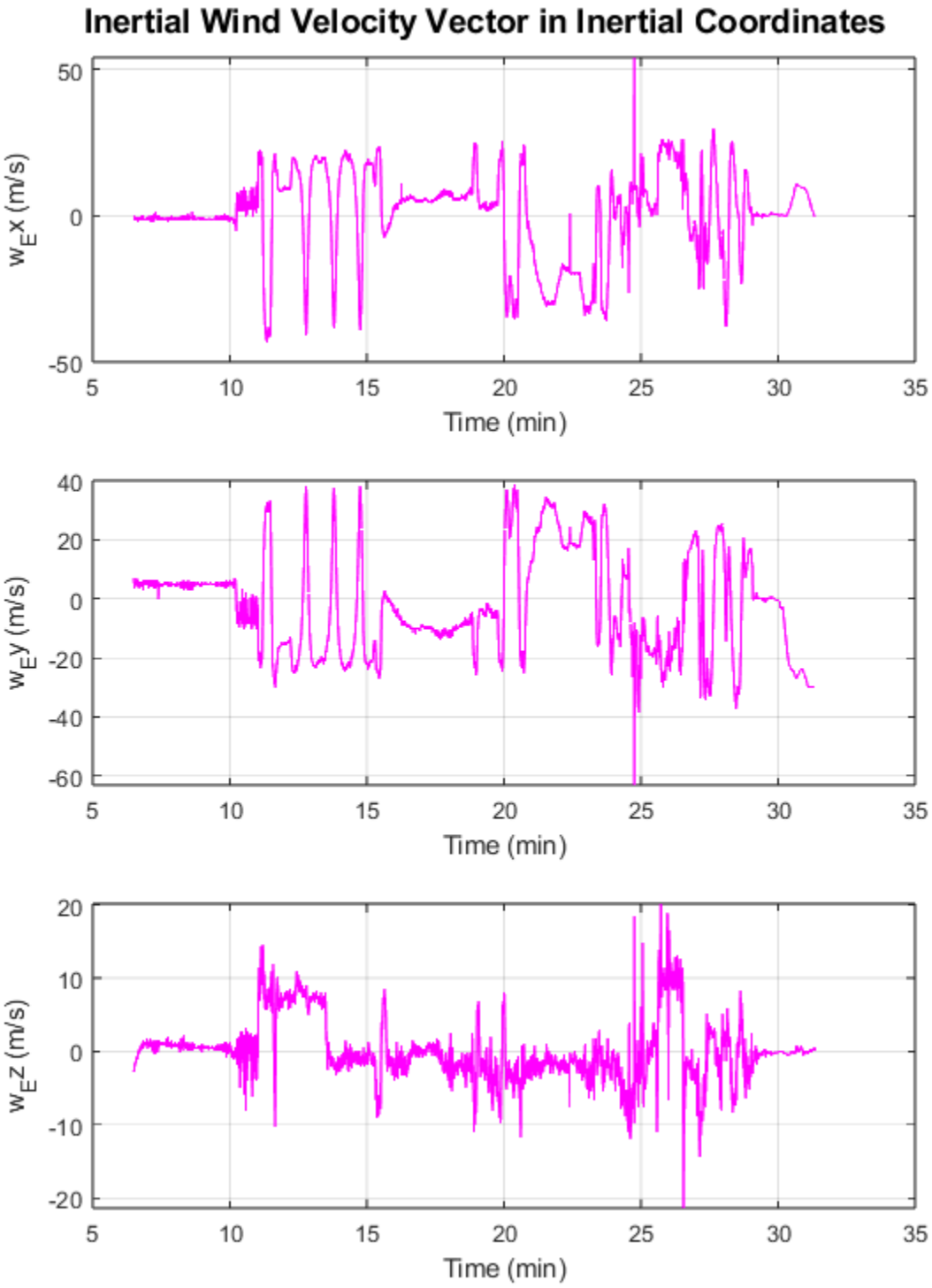
```
load('RaavenWindData.mat');

% Calculate the air relative velocity vector in body, in inertial, and the
inertial
% wind in inertial coordinates
wind_matrix = zeros(length(Time), 3);
air_rel_body_matrix = zeros(length(Time), 3);
air_rel_inertial_matrix = zeros(length(Time), 3);
for i = 1:length(Time)
    wind_angles = [Va(i); beta(i)*pi/180; alpha(i)*pi/180];
    euler_angles = [roll(i)*pi/180; pitch(i)*pi/180; yaw(i)*pi/180];
    wind_matrix(i,:) = CalculateInertialWind(aircraft_velocity_e_e(i,:)',
euler_angles, wind_angles(1), wind_angles(2), wind_angles(3))';
    air_rel_body_matrix(i,:) =
WindAnglesToAirRelativeVelocityVector(wind_angles)';
    air_rel_inertial_matrix(i,:) =
TransformFromBodyToInertial(air_rel_body_matrix(i,:)', euler_angles);
end


%%%% Plotting
% Air relative velocity vector in body coordinates
% Enhanced plot for Air Relative Velocity Vector in Body Coordinates
figure();
sgtitle('Air Relative Velocity Vector in Body Coordinates', 'FontSize', 14,
'FontWeight', 'bold')

subplot(3,1,1);
plot(Time, air_rel_body_matrix(:,1), 'r', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('u (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

subplot(3,1,2);
plot(Time, air_rel_body_matrix(:,2), 'r', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('v (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

subplot(3,1,3);
plot(Time, air_rel_body_matrix(:,3), 'r', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('w (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

% Adjust spacing between subplots for a cleaner look
set(gcf, 'Position', [100, 100, 600, 800]);
```

```
% Air relative velocity vector in inertial coordinates
figure();
sgtitle('Air Relative Velocity Vector in Inertial Coordinates', 'FontSize',
14, 'FontWeight', 'bold')

subplot(3,1,1);
plot(Time, air_rel_inertial_matrix(:,1), 'b', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('u_E (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

subplot(3,1,2);
plot(Time, air_rel_inertial_matrix(:,2), 'b', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('v_E (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

subplot(3,1,3);
plot(Time, air_rel_inertial_matrix(:,3), 'b', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('w_E (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

% Adjust spacing between subplots for a cleaner look
set(gcf, 'Position', [100, 100, 600, 800]);



% Inertial wind velocity in inertial coordinates
figure();
sgtitle('Inertial Wind Velocity Vector in Inertial Coordinates', 'FontSize',
14, 'FontWeight', 'bold')

subplot(3,1,1);
plot(Time, wind_matrix(:,1), 'm', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('w_Ex (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

subplot(3,1,2);
plot(Time, wind_matrix(:,2), 'm', 'LineWidth', 1)
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('w_Ey (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

subplot(3,1,3);
plot(Time, wind_matrix(:,3), 'm', 'LineWidth', 1)
```

```
grid on;
xlabel('Time (min)', 'FontSize', 12);
ylabel('w_Ez (m/s)', 'FontSize', 12)
set(gca, 'FontSize', 10)

% Adjust spacing between subplots for a cleaner look
set(gcf, 'Position', [100, 100, 600, 800]);
```


Air Relative Velocity Vector in Body Coordinates
u (m/s)
v (m/s)
w (m/s)
Time (min)
Time (min)
Time (min)

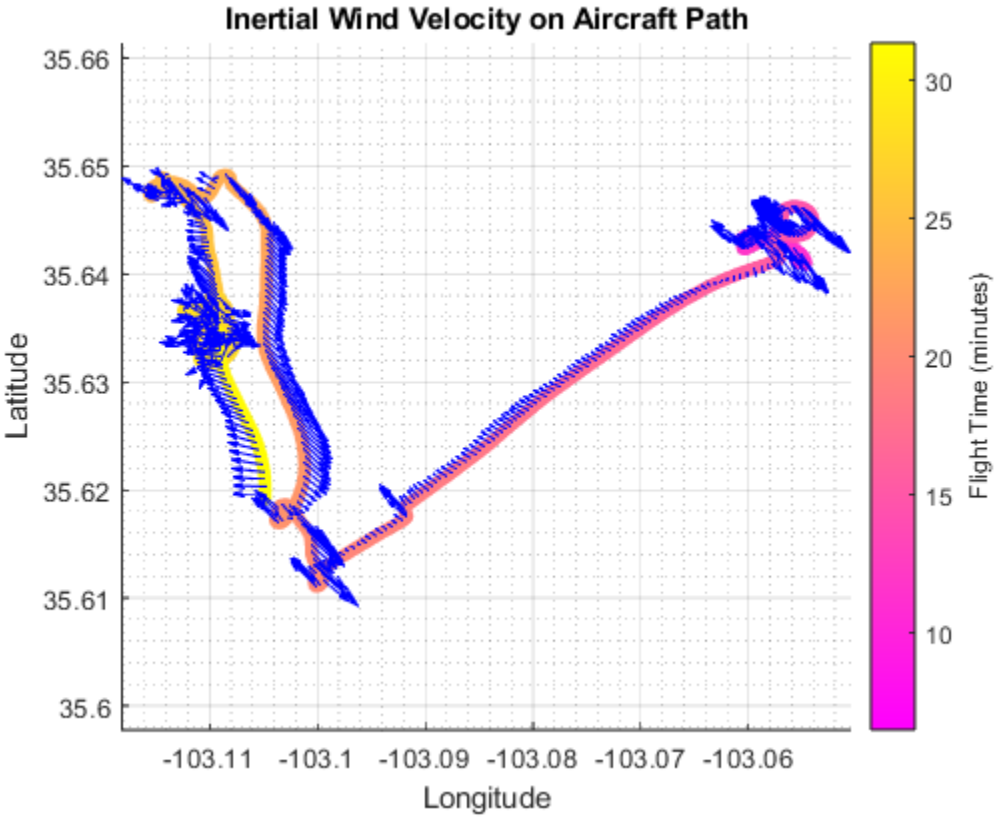

Air Relative Velocity Vector in Inertial Coordinates
u_E (m/s)
40
20
0
-20
-40
-60
5
10
15
20
25
30
35
Time (min)
v_E (m/s)
60
40
20
0
-20
-40
5
10
15
20
25
30
35
Time (min)
w_E (m/s)
20
10
0
-10
-20
-30
5
10
15
20
25
30
35
Time (min)


Inertial Wind Velocity Vector in Inertial Coordinates
$w_E x$ (m/s)
50
0
-50
5
10
15
20
25
30
35
Time (min)
$w_E y$ (m/s)
40
20
0
-20
-40
-60
5
10
15
20
25
30
35
Time (min)
$w_E z$ (m/s)
20
10
0
-10
-20
5
10
15
20
25
30
35
Time (min)

Recreate Figure 1

```
figure();
% Create the scatter plot with time as the color
scatter(lon, lat, 25, Time, 'filled');
hold on

% Add vectors using quiver
plot_indices = 1:250:length(lat);
wind_vec_x = 0.0001 * wind_matrix(:,1);
wind_vec_y = 0.0001 * wind_matrix(:,2);
quiver(lon(plot_indices), lat(plot_indices), wind_vec_y(plot_indices),
wind_vec_x(plot_indices), 0, 'b', 'MaxHeadSize', 0.1, 'LineWidth', 0.5);


% Customize the color bar and labels
colormap(spring);                % Color map (choose any you like: parula,
jet, etc.)
c = colorbar;                  % Create color bar and assign to variable c
c.Label.String = 'Flight Time (minutes)';  % Add label to color bar
caxis([min(Time) max(Time)]); % Set color axis limits
xlabel('Longitude');
ylabel('Latitude');
title('Inertial Wind Velocity on Aircraft Path');
axis equal
grid on
grid minor

function inertial_wind = CalculateInertialWind(Ve, euler_angles, Va, beta,
alpha)
% Calculates the inertial wind vector in inertial coordinates
% Inputs: (n is the number of measurements)
%   Ve = inertial velocity
%   euler_angles = vector of euler angles
%   Va = airspeed
%   beta = sideslip
%   alpha = angle of attack
% Outputs:
%   inertial_wind = inertial wind vector in inertial coordinates
%
% Author: Thomas Dunnington
% Modified: 11/6/2024

% Get the air relative velocity vector
wind_angles = [Va; beta; alpha];
air_rel_velocity = WindAnglesToAirRelativeVelocityVector(wind_angles);

% Convert to inertial coordinates
air_rel_velocity_inertial = TransformFromBodyToInertial(air_rel_velocity,
euler_angles);

% Calculate the wind
```

```
inertial_wind = Ve - air_rel_velocity_inertial;
end
```



Published with MATLAB® R2023b