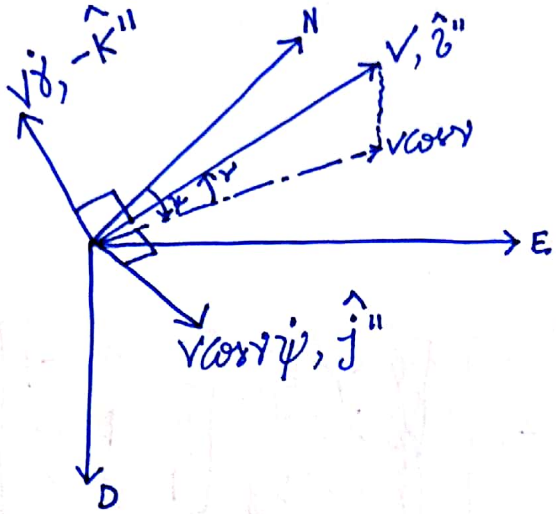

the unit vector of the Navigational axes are shown in the picture.

$$\vec{F}^{Nav}_{ext} = m\,\vec{a}^{Nav}$$

$$\vec{F}^{Nav}_{ext} = R^{Nav/bod}\,\vec{F}^{bod}_{Thrust} + R^{Nav/NED}\,\vec{F}^{NED}_{gravity} + R^{Nav/wind}\,\vec{F}^{wind}_{Aerodynamic}$$

$$\vec{F}^{bod}_{Thrust} = \begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix};\quad \vec{F}^{NED}_{Gravity} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix};\quad \vec{F}^{wind}_{Aerodynamic} = \begin{bmatrix} -D \\ 0 \\ -L \end{bmatrix}$$

Based on the Equation for the Rotation matrices given

$$R^{Nav/bod} = R_1(-\phi)\,R_2(-\alpha)$$

$$R^{Nav/NED} = R_2(\gamma)\,R_3(\psi)$$

$$R^{wind/NED} = R_1(\phi)\,R_2(\gamma)\,R_3(\psi)$$

So we write $R^{Nav/wind}$ as

$$R^{Nav/wind} = R^{Nav/NED} \; R^{NED/wind} = R^{Nav/NED}\left[R^{wind/NED}\right]^T$$

$$= R_2(\gamma) R_3(\psi) \left[R_1(\phi) R_2(\gamma) R_3(\psi)\right]^T$$

$$= R_2(\gamma) R_3(\psi) \; R_3(\psi)^T \; R_2(\gamma)^T \; R_1(\phi)^T$$

using the fact the $R_3(\psi)^T$ is inverse of $R_3(\psi)$ and $R_2(\gamma)^T$ is for $R_2(\gamma)$

$$R^{Nav/wind} = R_1(\phi)^T$$

also $R_1(\phi)^T$ is undoing the Rotation, so this is Equivalent to $R_1(-\phi)$

$$\boxed{\therefore\; R^{Nav/wind} = R_1(-\phi)}$$

So we substitute these to $\vec{F}^{\,Nav}_{ext}$

$$\vec{F}^{\,Nav}_{ext} = R_1(-\phi) R_2(-\alpha)\begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix} + R_1(-\phi)\begin{bmatrix} -D \\ 0 \\ -L \end{bmatrix} + R_2(\gamma) R_3(\psi)\begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$

$$\vec{F}^{\,Nav}_{ext} = R_1(-\phi)\left\{R_2(-\alpha)\begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -D \\ 0 \\ -L \end{bmatrix}\right\} + R_2(\gamma) R_3(\psi)\begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}$$

$$R_2(-\alpha)\begin{bmatrix}T\\0\\0\end{bmatrix} = \begin{bmatrix}\cos\alpha & 0 & \sin\alpha\\ 0 & 1 & 0\\ -\sin\alpha & 0 & \cos\alpha\end{bmatrix}\begin{bmatrix}T\\0\\0\end{bmatrix} = \begin{bmatrix}T\cos\alpha\\0\\-T\sin\alpha\end{bmatrix}$$

$$R_1(\phi)\left\{R_2(-\alpha)\begin{bmatrix}T\\0\\0\end{bmatrix} + \begin{bmatrix}-D\\0\\-L\end{bmatrix}\right\} = \begin{bmatrix}1 & 0 & 0\\ 0 & \cos\phi & -\sin\phi\\ 0 & \sin\phi & \cos\phi\end{bmatrix}\begin{bmatrix}T\cos\alpha - D\\0\\-T\sin\alpha - L\end{bmatrix}$$

$$= \begin{bmatrix}T\cos\alpha - D\\ (T\sin\alpha + L)\sin\phi\\ -(T\sin\alpha + L)\cos\phi\end{bmatrix}$$

$$\vec{F}^{\,Nav}_{ext} = \begin{bmatrix}T\cos\alpha - D\\ (T\sin\alpha + L)\sin\phi\\ -(T\sin\alpha + L)\cos\phi\end{bmatrix} + R_2(\gamma)R_3(\psi)\begin{bmatrix}0\\0\\mg\end{bmatrix}$$

$$R_3(\psi)\begin{bmatrix}0\\0\\mg\end{bmatrix} = \begin{bmatrix}\cos\psi & \sin\psi & 0\\ -\sin\psi & \cos\psi & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}0\\0\\mg\end{bmatrix} = \begin{bmatrix}0\\0\\mg\end{bmatrix}$$

$$R_2(\gamma)R_3(\psi)\begin{bmatrix}0\\0\\mg\end{bmatrix} = \begin{bmatrix}\cos\gamma & 0 & -\sin\gamma\\ 0 & 1 & 0\\ \sin\gamma & 0 & \cos\gamma\end{bmatrix}\begin{bmatrix}0\\0\\mg\end{bmatrix} = \begin{bmatrix}-mg\sin\gamma\\0\\mg\cos\gamma\end{bmatrix}$$

$$\boxed{\vec{F}^{\,Nav}_{ext} = \begin{bmatrix}T\cos\alpha - D - mg\sin\gamma\\ (T\sin\alpha + L)\sin\phi\\ -(T\sin\alpha + L)\cos\phi + mg\cos\gamma\end{bmatrix}}$$

based of the definitions of $\hat{i}''$, $\hat{j}''$, $\hat{k}''$ and

$\dot{V}$, $V\dot{\gamma}$, $V\cos\gamma\dot{\psi}$

$$m\vec{a}^{\,NAV}_{Ext} = m\begin{bmatrix}\dot{V}\\ V\cos\gamma\dot{\psi}\\ -V\dot{\gamma}\end{bmatrix} = \vec{F}^{\,NAV}_{Ext}$$

So we show that

$$\boxed{\begin{aligned} m\dot{V} &= T\cos\alpha - D - mg\sin\gamma \\ mV\dot{\gamma} &= [T\sin\alpha + L]\cos\phi - mg\cos\gamma \\ mV\cos\gamma\dot{\psi} &= (T\sin\alpha + L)\sin\phi \end{aligned}}$$

Table of Contents

Question 2

```
clear;clc;
close all;
format long

load rotmateuler123_data01

R01 = rotmateuler123(psi,theta,phi);

disp('Checking Result for Euler angles data set 01')
disp('Error is:')
disp(norm(R01-R_true));

clear
load rotmateuler123_data02

R_true = rotmateuler123(psi,theta,phi);
disp('R_true from Euler angles for data set 02 is:')
disp(R_true);
```

```
Checking Result for Euler angles data set 01
Error is:
     0

R_true from Euler angles for data set 02 is:
   0.053726755769759  -0.968751794569192  -0.242143338197493
   0.296835946406159  -0.216037226294276   0.930170058524911
  -0.953415888600361  -0.121851866518510   0.275953376479323
```

Question 3

```
clear;

load rotmatquaternion_data05

R01 = rotmatquaternion(q);

disp('Checking Result for quaternion data set 01')
disp('Error is:')
disp(norm(R01-R_true));

clear
```

```
load rotmatquaternion_data06

R_true = rotmatquaternion(q);
disp('R_true from quaternion for data set 02 is:')
disp(R_true);
format short
```

```
Checking Result for quaternion data set 01
Error is:
     0

R_true from quaternion for data set 02 is:
  -0.337238027363993   0.634586043613785  -0.695392742376742
  -0.874691034865763   0.061932657860731   0.480707748445733
   0.348117949013122   0.770366730292079   0.534180675833550
```

Question 4

$$R_a = R_1(\frac{\pi}{2})$$

$$R_b = R_2(\frac{\pi}{2})$$

$$R_a R_b = R_1(\frac{\pi}{2}) R_2(\frac{\pi}{2})$$

$$R_b R_a = R_2(\frac{\pi}{2}) R_1(\frac{\pi}{2})$$

```
Ra = rotmateuler123(pi/2,0,0);
disp('Ra is:')
disp(Ra)

Rb = rotmateuler123(0,pi/2,0);
disp('Rb is:')
disp(Rb)

RaRb = Ra*Rb;
RbRa = Rb*Ra;

disp('Ra*Rb is:')
disp(RaRb)
disp('Rb*Ra is:')
disp(RbRa)

disp('These examples of Ra and Rb show that Ra*Rb is not always equal
 to Rb*Ra')
```

```
Ra is:
    0.0000    1.0000         0
   -1.0000    0.0000         0
         0         0    1.0000
```

```
Rb is:
    0.0000         0   -1.0000
         0    1.0000         0
    1.0000         0    0.0000

Ra*Rb is:
    0.0000    1.0000   -0.0000
   -0.0000    0.0000    1.0000
    1.0000         0    0.0000

Rb*Ra is:
    0.0000    0.0000   -1.0000
   -1.0000    0.0000         0
    0.0000    1.0000    0.0000

These examples of Ra and Rb show that Ra*Rb is not always equal to
 Rb*Ra
```

Published with MATLAB® R2017a