Aerospace Engineering Department
Faculty of Engineering
Cairo University

4th year

AER4420

Design of control systems for aeronautical and space vehicles

Task (2)

NUMERICAL SOLUTION OF ODE (RK4)
AIRPLANE SIMULATOR PART I

Note: This will be the first part of the simulator you will be using from now on in the course, you have to verify and validate your results in one of two ways:

- Compare results with the simulink simulator you will build.
- Compare results with one of your colleagues.

Task statement

Write a code that solves the Rigid body dynamics (RBD) equations for given constant values of forces & moments, using "Runge-Kutta" 4th order method.

Develop your code from scratch, and use the following data and initial conditions to solve the equations (all variables are in SI units) (Angles, Angular rates, Angular accelerations are in $rad, rad/s, rad/s^2$

$$t_{final} = 25\ sec$$

$$Forces = [2\ ;8\ ;3]N$$

$$Moments = [14;\ 20;\ 7]N.m.$$

$$mass = 11\ kg$$

$$\boldsymbol{I} = \begin{bmatrix} 1 & -2 & -1 \\ -2 & 5 & -4 \\ -1 & -4 & 0.2 \end{bmatrix} kg.m^2$$

Initial conditions ($S.I.units$):

$$[u, v, w, p, q, r, \phi, \theta, \psi, x, y, z]_{t=0} =$$

$$\left[10, 2, 0, 2 * \frac{\pi}{180}, 1 * \frac{\pi}{180}, 0 * \frac{\pi}{180}, 20 * \frac{\pi}{180}, 15 * \frac{\pi}{180}, 30 * \frac{\pi}{180}, 2, 4, 7\right]$$

Notes and hints:

RBD Equations in **scalar** form:

This screenshot is from Etkin.

$$
\begin{aligned}
X - mg \sin\theta &= m(\dot{u}^E + qw^E - rv^E) \\
Y + mg \cos\theta \sin\phi &= m(\dot{v}^E + ru^E - pw^E) \\
Z + mg \cos\theta \cos\phi &= m(\dot{w}^E + pv^E - qu^E) \\
L &= I_x\dot{p} - I_{zx}\dot{r} + qr(I_z - I_y) - I_{zx}pq + qh'_z - rh'_y \\
M &= I_y\dot{q} + rp(I_x - I_z) + I_{zx}(p^2 - r^2) + rh'_x - ph'_z. \\
N &= I_z\dot{r} - I_{zx}\dot{p} + pq(I_y - I_x) + I_{zx}qr + ph'_y - qh'_x
\end{aligned}
$$

$$
\begin{aligned}
p &= \dot{\phi} - \dot{\psi}\sin\theta \\
q &= \dot{\theta}\cos\phi + \dot{\psi}\cos\theta\sin\phi \\
r &= \dot{\psi}\cos\theta\cos\phi - \dot{\theta}\sin\phi \\
\dot{\phi} &= p + (q\sin\phi + r\cos\phi)\tan\theta \\
\dot{\theta} &= q\cos\phi - r\sin\phi \\
\dot{\psi} &= (q\sin\phi + r\cos\phi)\sec\theta
\end{aligned}
$$

$$
\begin{aligned}
\dot{x}_E = {} & u^E\cos\theta\cos\psi + v^E(\sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi) \\
& + w^E(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) \\
\dot{y}_E = {} & u^E\cos\theta\sin\psi + v^E(\sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi) \\
& + w^E(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) \\
\dot{z}_E = {} & -u^E\sin\theta + v^E\sin\phi\cos\theta + w^E\cos\phi\cos\theta
\end{aligned}
$$

RBD Equations in **vector** form:

Kinetics:

$$\sum \mathbf{F} = \frac{\mathrm{d}}{\mathrm{d}t}(m\mathbf{v})$$

$$\sum \mathbb{M} = \frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{H})$$

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = m\left(\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix}\right)$$

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = I\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I\begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Where:

$$\boldsymbol{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix}$$

Kinematics:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = [J]\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & sin\,(\phi)tan\,(\theta) & cos(\phi)tan\,(\theta) \\ 0 & cos\,(\phi) & -sin\,(\phi) \\ 0 & \dfrac{sin\,(\phi)}{cos\,(\theta)} & \dfrac{cos\,(\phi)}{cos\,(\theta)} \end{bmatrix}\begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$\begin{bmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{bmatrix} = [T]_{EB}\begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

$$= \begin{bmatrix} Cos\,(\theta)Cos\,(\psi) & -Cos\,(\varphi)Sin\,(\psi) + Sin\,(\varphi)Sin\,(\theta)Cos\,(\psi) & Sin\,(\varphi)Sin\,(\psi) + Cos\,(\varphi)Sin\,(\theta)Cos\,(\psi) \\ Cos\,(\theta)Sin\,(\psi) & Cos\,(\varphi)Cos\,(\psi) + Sin\,(\varphi)Sin\,(\theta)Sin\,(\psi) & -Sin\,(\varphi)Cos\,(\psi) + Cos\,(\varphi)Sin\,(\theta)Sin\,(\psi) \\ -Sin\,(\theta) & Sin\,(\varphi)Cos\,(\theta) & Cos\,(\varphi)Cos\,(\theta) \end{bmatrix}\begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

See Appendix for some useful functions to help you with the code.

The "Runge-Kutta 4" method for simultaneous differential equations:

Visit Wikipedia and read about RK4 algorithm.

$$y_{n+1} = y_n + \frac{1}{6} * dt * (k_1 + 2k_2 + 2k_3 + k_4)$$

$$t_{n+1} = t_n + dt$$

Where:

$$k_1 = f(t_n, y_n)$$

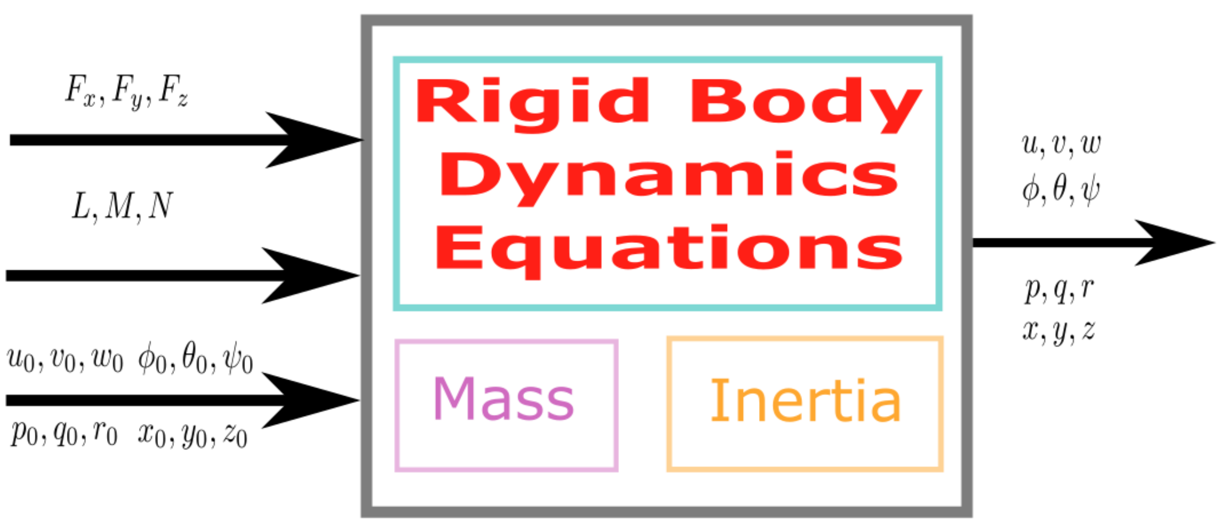
$$k_2 = f(t_n + \frac{dt}{2}, y_n + dt\frac{k_1}{2})$$

$$k_3 = f(t_n + \frac{dt}{2}, y_n + dt\frac{k_2}{2})$$

$$k_4 = f(t_n, y_n + dt)$$

This set of equations is also valid for multi-dimensional variables (vector form).

RBD Solver overview



Inputs to the RBD solver code:

- Forces and Moments $[F_x, F_y, F_z, L, M, N]$.
- Initial conditions for all 12 states $[u_0, v_0, w_0, \phi_0, \ldots]$.

Outputs from the RBD solver code:

- All 12 states at the next timestep $[u, v, w, \phi, \ldots]$.

Constants to be defined:

- Mass (m)
- Inertia matrix ($\boldsymbol{I}$)
- Time interval: $(t_{final} \& \ timestep)$

Simulink simulator

Use the 6 DOF Euler angles block in Simulink to solve the above problem and compare the results using plots.

Reminder: Adjust the solver settings for Simulink to match your Matlab solver.

Bonus

You want to compare two signals (vectors) (e.g. you want to compare the solution for RBD equations from Matlab and Simulink), what mathematical expression(s) can you use to express the error between the two vectors?

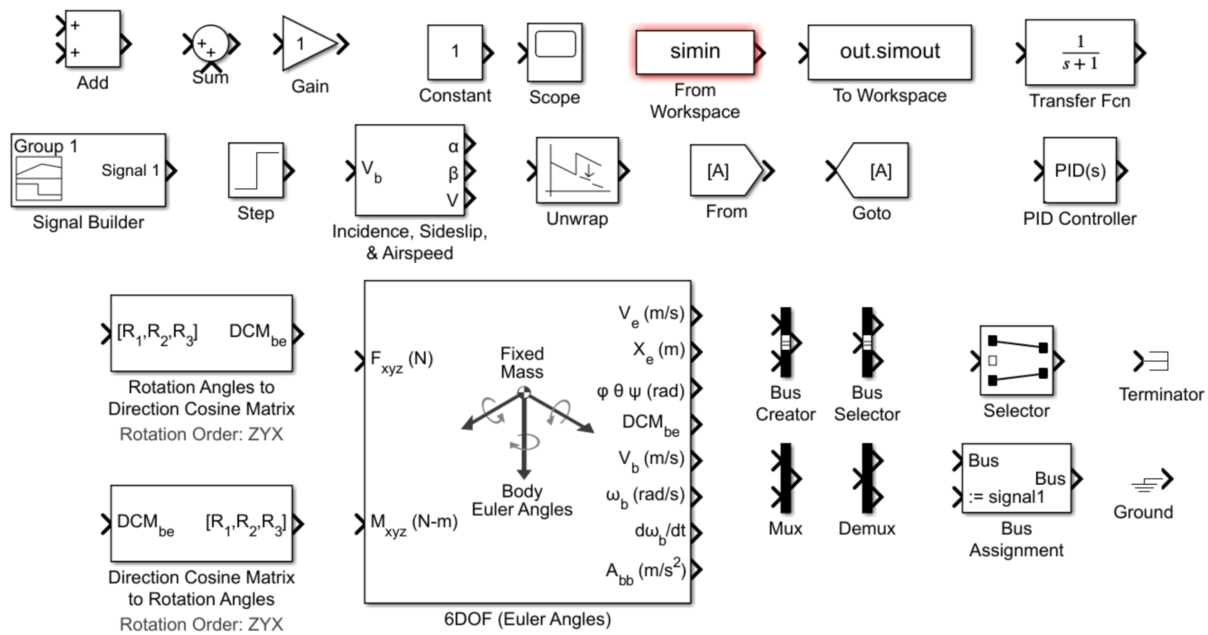
Please list your references.

Appendix

Useful Matlab functions:

- **eul2rotm**
- ode45
- sim
- simulink
- gcb

Useful Simulink blocks:



Monday, February 17th, 2025

References

1) John H. Blakelock - Automatic Control of Aircraft and Missiles-Wiley-Interscience (1991)
2) Flight Stability and Automatic Control - Robert C. Nelson
3) Etkin B., Reid L.D. - Dynamics of flight_ Stability and control-Wiley (1996)
4) NASA CR-2144--Heffley--Aircraft Handling Qualities Data