Flight Simulation

Elisa Capello
Politecnico di Torino

02IIQMT / 01PETMT - Simulazione del volo
Laurea Magistrale in Ingegneria Aerospaziale
a.a. 2021-2022

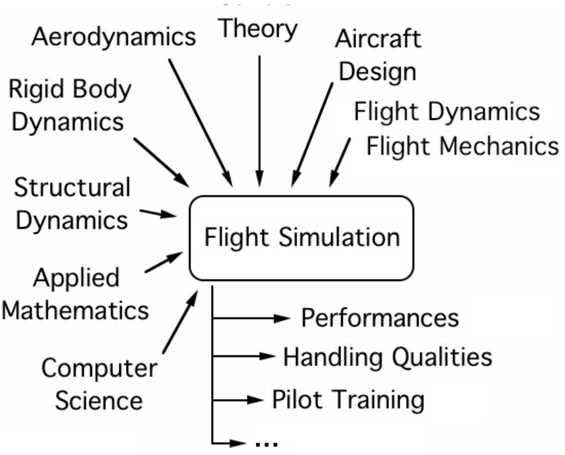
Outline (I)

- Reference Frames:
  1. Inertial Frame
  2. North, East, Down (NED) Frame
  3. Body Frame
  4. Wind Frame
- Aircraft Model:
  1. General Equations of motion
  2. Aerodinamic Derivatives
  3. Linear Model
- Detailed Examples: Fixed Wing UAV:
  1. Nonlinear Model
  2. Linear Model
  3. Example of implementation on Matlab/Simulink

Outline (II)

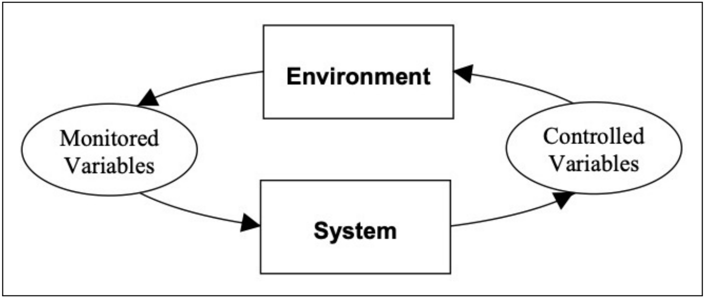
- Detailed Examples: Multi Rotor UAV:
  1. Nonlinear Model
  2. Linear Model
  3. Example of implementation on Matlab/Simulink
- Identification of Model Parameters:
  1. Literature Review of Identification Methods
  2. Thrust/Torque Evaluation
  3. Moments of Inertia Evaluation

Flight Simulation



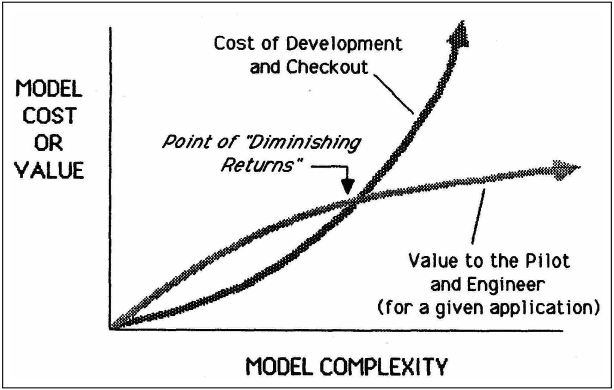
One of the most important activities for the definition of aircraft simulation models is the outline of **functional** and **software requirements**, assuming that the boundaries between the systems being modeled and their environment are clearly defined. This provides a sound understanding of what lies within each system to be modeled and what lies within a larger environment. Every system is embedded within the environment in which it operates, and this environment is often a larger collection of systems. Without a clear definition of the system boundary, it is very easy to write requirements that duplicate or conflict with those being defined at a higher level or to miss requirements because they are assumed to be provided by the environment. This is particularly important when a simulation model is being developed by multiple entities.

Flight Simulation

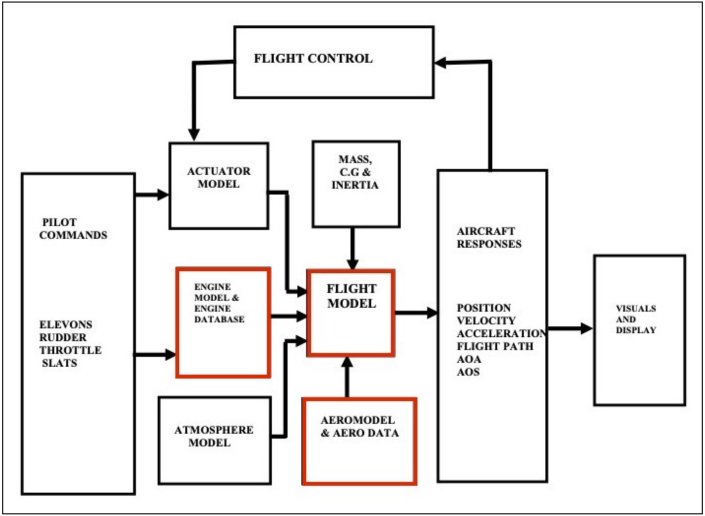


One way to define the boundary of a system (DOT/FAA/AR-08/32) is to view the system as an element that interacts with its environment through a set of **monitored** (the actual altitude of an aircraft or its airspeed) and **controlled variables** (the position of a control surface or the displayed value of the altitude on the primary flight display).

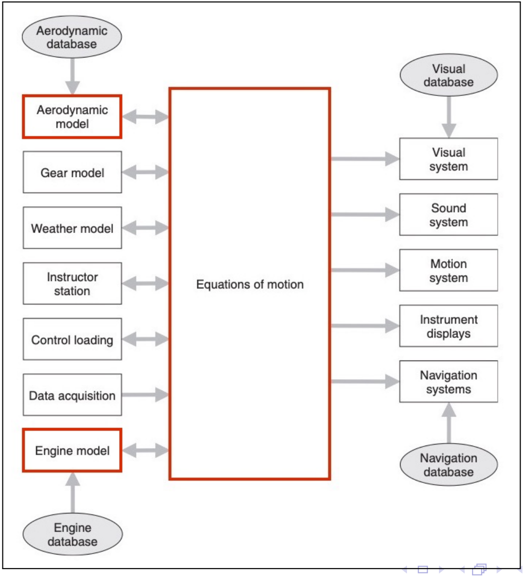
Flight Simulation



Flight Simulation



Flight Simulation

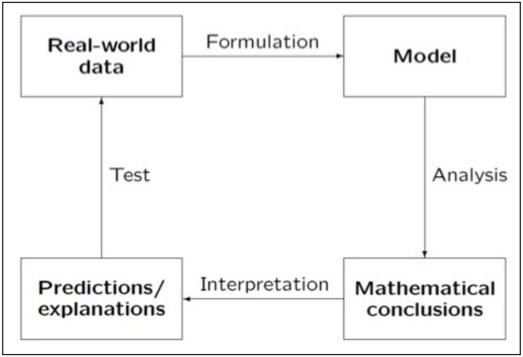


Flight Simulation

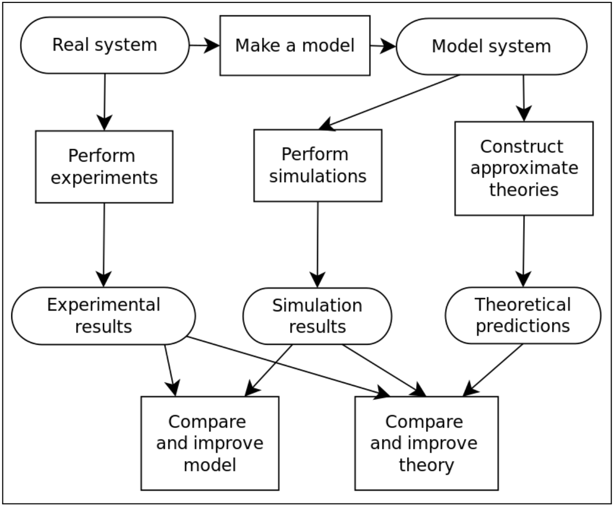
Flight simulators are based on computer models implementing a mathematical representation for a specific flying vehicle. The process of building a computer model is complex, and it exists an interplay between experiment, simulation, and theory:

- 1. Problem identification
- 2. Model formulation
- 3. Analysis
- 4. Computation
- 5. Model validation

Flight Simulation



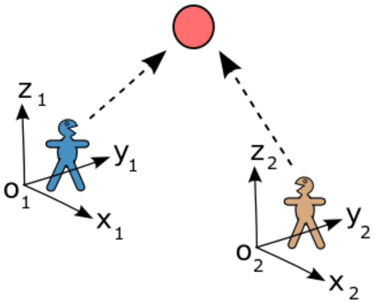
Flight Simulation



The analysis of the equations of motion of the aircraft is not easy considering a reference attached to the Earth. Usually, a moving reference system centered in the CG (barycentric) can be defined and profitably used. Such reference was first introduced by Leonhard Euler (Basel 1707 - St. Petersburg 1783), better known as Euler. The equations describing the unsteady motion of an aircraft have been developed and studied more recently by B. Melville Jones, L. Bairstow, H. B. Glauert, W. L. Cowley, S. B. Gates and H. C. Garner.

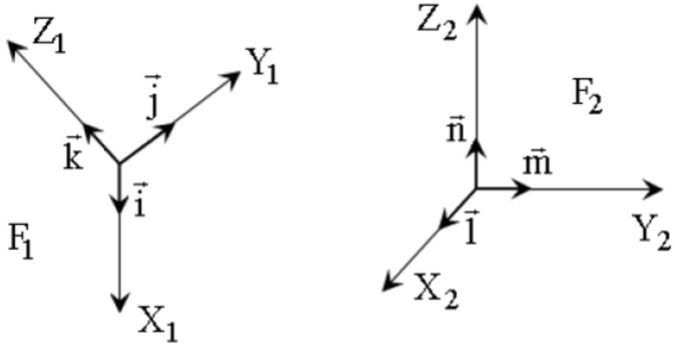
Earth-Centered Earth-Fixed Frame (ECEF) is used as inertial reference frame, considering the Earth as a flat and still surface. Three moving reference frames are typically considered in the analysis of flight dynamics: a local vertical reference, a wind reference and a body reference.

Reference Frames



Reference Frames (I)

Euler Angles are three independent quantities that define the orientation of a generic reference frame to another reference frame. Note that the sequence of rotations is fixed as (ψ, θ, ϕ) is not a base for a linear space.
The components of a vector relative to the frame $F_1 = (X_1, Y_1, Z_1)$ with unit vectors (i, j, k) are converted into a second frame $F_2 = (X_2, Y_2, Z_2)$ with unit vectors (l, m, n).



The three rotations (with anti-clockwise direction considered positive) are applied to the frame F_2 so that it will be aligned to F_1:

- 1^{st} rot. (ψ): positive rotation about Z_2 (anti-clockwise), so that an intermediate frame is defined $F_2^{'} = (X_2^{'}, Y_2^{'}, Z_2^{'})$ with unit vectors $(l^{'}, m^{'}, n^{'})$. —> $Z_2^{'} \equiv Z_2$
- 2^{nd} rot. (θ): rotation about $Y_2^{'}$. Another intermediate frame is defined as $F_2^{''} = (X_2^{''}, Y_2^{''}, Z_2^{''})$ with unit vectors $(l^{''}, m^{''}, n^{''})$. —> $Y_2^{''} \equiv Y_2^{'}$
- 3^{rd} rot. (ϕ): rotation about $X_2^{''}$. Finally, the vector in the desired frame is obtained, $F_1 = (X_1, Y_1, Z_1)$ with unit vectors (i, j, k). —> $X_1 \equiv X_2^{''}$

Right handed reference frames are assumed.

Reference Frames (III)

The properties of elemental rotation matrices can be summarized as:

- The diagonal of the matrix is populated by the cosines of the generic angle of rotation and by a single unitary element 1 for the line that refers to the axis of rotation.
- The unit element belonging to the diagonal identifies a row and a column of zero entries.
- Other terms of the matrix are sines of the angle of rotation (with opposite signs and the negative sign falls for positive rotation in the line under the unitary diagonal element).
- The elementary rotation matrices are orthogonal or, in an equivalent manner, inverse and transpose matrices coincide.

Reference Frames (IV)

Therefore, for each Euler angle is possible to associate a matrix of elementary rotation:

$$\left[\ \Psi\ \right] = \begin{bmatrix} \cos\Psi & -\sin\Psi & 0 \\ \sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \rightarrow \left[\ \Psi\ \right]^{-1} = \begin{bmatrix} \cos\Psi & \sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\left[\ \Theta\ \right] = \begin{bmatrix} \cos\Theta & 0 & \sin\Theta \\ 0 & 1 & 0 \\ -\sin\Theta & 0 & \cos\Theta \end{bmatrix} \rightarrow \left[\ \Theta\ \right]^{-1} = \begin{bmatrix} \cos\Theta & 0 & -\sin\Theta \\ 0 & 1 & 0 \\ \sin\Theta & 0 & \cos\Theta \end{bmatrix}$$

$$\left[\ \Phi\ \right] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \sin\Phi & \cos\Phi \end{bmatrix} \rightarrow \left[\ \Phi\ \right]^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{bmatrix}$$

Reference Frames (V)

Now, the unit vectors of the intermediate reference systems (introduced in the definition of the three Euler angles) can be expressed by means of arrays of elemental rotations:

$$\begin{bmatrix} R_{x2} \\ R_{y2} \\ R_{z2} \end{bmatrix} = [\Psi] \cdot \begin{bmatrix} R'_x \\ R'_y \\ R'_z \end{bmatrix} \rightarrow \begin{bmatrix} R'_x \\ R'_y \\ R'_z \end{bmatrix} = [\Theta] \cdot \begin{bmatrix} R''_x \\ R''_y \\ R''_z \end{bmatrix} \rightarrow \begin{bmatrix} R''_x \\ R''_y \\ R''_z \end{bmatrix} = [\Phi] \cdot \begin{bmatrix} R_{x1} \\ R_{y1} \\ R_{z1} \end{bmatrix}$$

from which:

$$\begin{bmatrix} R_{x2} \\ R_{y2} \\ R_{z2} \end{bmatrix} = [\Psi] \cdot [\Theta] \cdot [\Phi] \cdot \begin{bmatrix} R_{x1} \\ R_{y1} \\ R_{z1} \end{bmatrix} = [T_{21}] \cdot \begin{bmatrix} R_{x1} \\ R_{y1} \\ R_{z1} \end{bmatrix}$$

Transformation Matrix (I)

The matrix $[L_{21}] = [\Psi][\Theta][\Phi]$ is the rotation matrix that transforms the overall components of a vector from the first reference system F_1 to the second set of three axes F_2:

$$\begin{bmatrix} R_{x2} \\ R_{y2} \\ R_{z2} \end{bmatrix} = [L_{21}] \cdot \begin{bmatrix} R_{x1} \\ R_{y1} \\ R_{z1} \end{bmatrix} \rightarrow \begin{bmatrix} R_{x1} \\ R_{y1} \\ R_{z1} \end{bmatrix} = [L_{21}]^{-1} \cdot \begin{bmatrix} R_{x2} \\ R_{y2} \\ R_{z2} \end{bmatrix}$$

Transformation Matrix (II)

$$[L_{21}] =$$

$$\begin{bmatrix} \cos\Psi\cos\Theta & \cos\Psi\sin\Theta\sin\Phi - \sin\Psi\cos\Phi & \cos\Psi\sin\Theta\cos\Phi + \sin\Psi\sin\Phi \\ \sin\Psi\cos\Theta & \sin\Psi\sin\Theta\sin\Phi + \cos\Psi\cos\Phi & \sin\Psi\sin\Theta\cos\Phi - \cos\Psi\sin\Phi \\ -\sin\Theta & \cos\Theta\sin\Phi & \cos\Theta\cos\Phi \end{bmatrix}$$

$$[L_{21}]^{-1} =$$

$$\begin{bmatrix} \cos\Psi\cos\Theta & \sin\Psi\cos\Theta & -\sin\Theta \\ \cos\Psi\sin\Theta\sin\Phi - \sin\Psi\cos\Phi & \sin\Psi\sin\Theta\sin\Phi + \cos\Psi\cos\Phi & \cos\Theta\sin\Phi \\ \cos\Psi\sin\Theta\cos\Phi + \sin\Psi\sin\Phi & \sin\Psi\sin\Theta\cos\Phi - \cos\Psi\sin\Phi & \cos\Theta\cos\Phi \end{bmatrix}$$

$$[L_{21}]^{-1} = [L_{12}] = [L_{21}]^T.$$

Kinematic Equations (I)

For the kinematic equations,
$(\dot{\psi}, \dot{\vartheta}, \dot{\phi}) = f(\omega_{X1}, \omega_{Y1}, \omega_{Z1}, \phi, \theta, \psi) \Rightarrow$

$$\omega_r = \dot{\psi} n + \dot{\vartheta} m' + \dot{\phi} i$$

$\dot{\psi}$ = angular velocity of F_2' referred to F_2
$\dot{\vartheta}$ = angular velocity of F_2'' referred to F_2'
$\dot{\phi}$ = angular velocity of F_1 referred to F_2''

$$\dot{\phi} = p + q \sin\phi \tan\vartheta + r \cos\phi \tan\vartheta \quad (1)$$

$$\dot{\vartheta} = q \cos\phi - r \sin\phi \quad (2)$$

$$\dot{\psi} = \frac{q \sin\phi}{\cos\vartheta} + \frac{r \cos\phi}{\cos\vartheta} \quad (3)$$

Kinematic Equations (II)

$$\begin{bmatrix} \omega_{x1} \\ \omega_{y1} \\ \omega_{z1} \end{bmatrix} = \begin{bmatrix} \dot{\Phi} \\ 0 \\ 0 \end{bmatrix} + [\Phi]^{-1} \cdot \begin{bmatrix} 0 \\ \dot{\Theta} \\ 0 \end{bmatrix} + [L_{21}]^{-1}_{\psi=0} \cdot \begin{bmatrix} 0 \\ 0 \\ \dot{\Psi} \end{bmatrix}$$

$$= \begin{bmatrix} \dot{\Phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\Theta} \cdot \cos\Phi \\ -\dot{\Theta} \cdot \sin\Phi \end{bmatrix} + \begin{bmatrix} -\dot{\Psi}\sin\Theta \\ \dot{\Psi}\cos\Theta\sin\Phi \\ \dot{\Psi}\cos\Theta\cos\Phi \end{bmatrix}$$

Kinematic Equations (III)

These equations provide the relationship between the time derivatives of the Euler angles and the components of the relative angular speed between the two reference frames. Note that the matrix $[L]$ does not depend on the angle Ψ. This matrix is singular for $\Theta = \pm 90^o$.

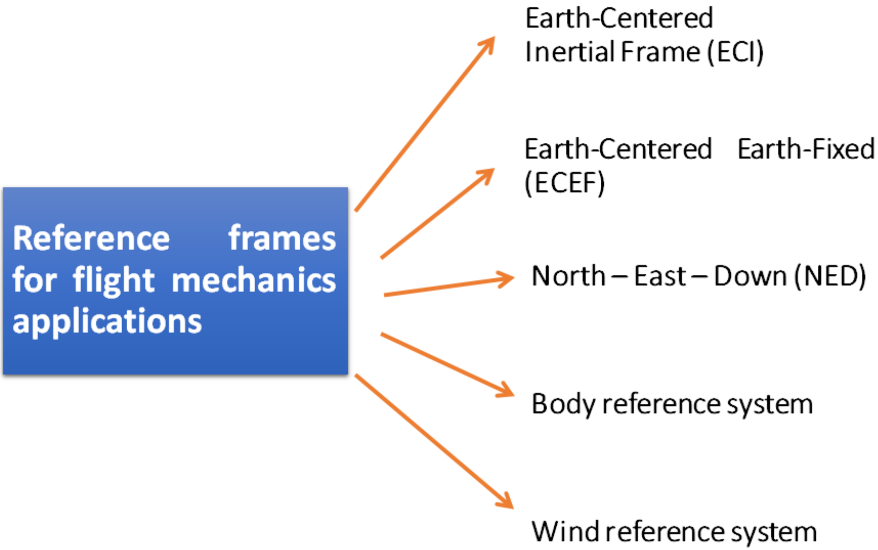
Assuming that F_1 is coincident with the body axes reference frame (aircraft fixed axes) and F_2 is the local vertical frame ('NED' axes), the following equivalences apply: $\omega_{x1} = p$, $\omega_{y1} = q$ and $\omega_{z1} = r$ while $\Phi = \varphi$, $\Theta = \theta$ and $\Psi = \psi$.

Hence:

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = [L] \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \approx \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

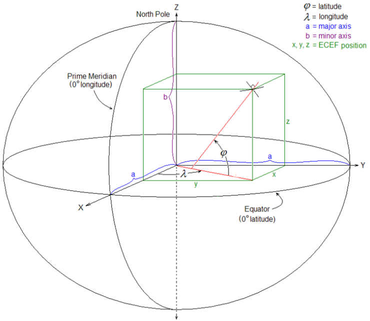
where the last approximation applies in case of linearization of the equations of motion, as for the small perturbation theory.

Reference Frames for UAVs



Earth Frames (I)

The origin of the system is the Earth Centre of Mass. The Z-axis Z_E points towards the North Pole. The direction of X-axis X_E is determined by the intersection of the plane defined by the Greenwich meridian and the equatorial plane. The Y-axis Y_E completes the right handed reference frame: it lies in the equatorial plane and points 90 deg East of X-axis direction.

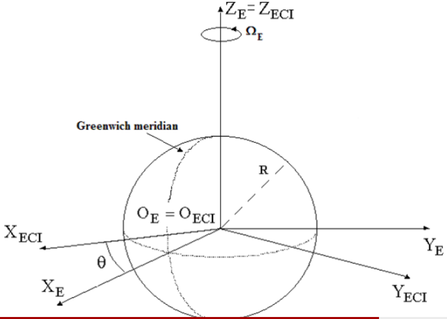


ECEF rotates with the Earth.
Reference frame for latitude and longitude definition

Earth Frames (II)

The ECI is the so called Inertial Geocentric Reference Frame. It is commonly used to study the motion of a body orbiting around the Earth (for example a satellite) referred to a pseudo geocentric inertial frame, with the axes oriented to the fixed stars.

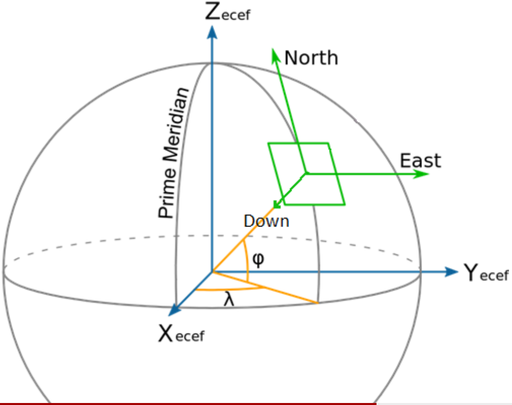
The origin of this reference frame is the Centre of the Earth. The X-axis X_{ECI} is directed to Aries (spring equinox), the Z-axis Z_{ECI} points toward the North Pole with Y-axis Y_{ECI} completing the right-handed frame.



ECI: FIXED FRAME!

North, East, Down Frame (I)

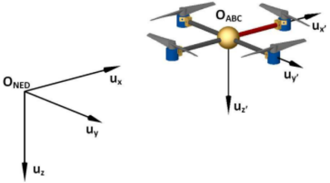
The origin of this ref. frame points on the surface of the geoid below the aircraft CoG. The vertical axis Z_V is directed along the local gravity acceleration vector. X_V and Y_V are in a plane parallel to the one tangent to the surface of the Earth for $h = 0$. X_V aims to the North, while Y_V is oriented Eastwards (right-handed frame).

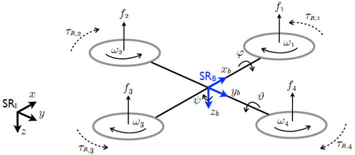


NED origin is fixed in ECEF coordinates!

North, East, Down Frame (II)

It is usually used for quadrotor as FIXED frame.



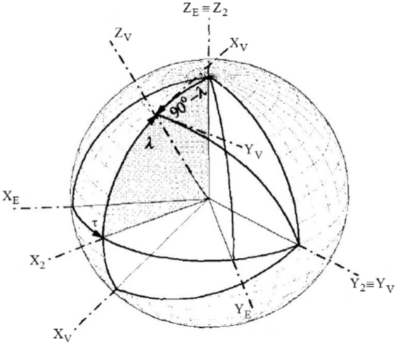


Optitrack, Vision sensors measure in this reference frame!

North, East, Down Frame (III)

To align ECEF with NED, two rotations are needed:

1. $\psi = \tau$, about Z_E axis aligning Y_E with Y_V (East axis)
2. $\theta = -(90deg + \lambda)$ about Y_E aligning the two frames



$$L_{EV} = [\Psi][\Theta] = \begin{bmatrix} -\sin\lambda\cos\tau & -\sin\tau & -\cos\lambda\cos\tau \\ -\sin\lambda\sin\tau & \cos\tau & -\cos\lambda\sin\tau \\ \cos\lambda & 0 & -\sin\lambda \end{bmatrix}$$

Body Reference Frame (I)

The body reference frame has its origin in the CG of the aircraft. Differently from the wind reference, this frame is rigidly attached to the aircraft, and changes its orientation with it. The first axis x_b may be defined parallel to the longitudinal axis, and the frame defined as body frame. If the first axis coincides with the principal barycentric axis the reference is defined principal body frame. The first axis points to the front of the aircraft, the z_b axis lies in the longitudinal plane, it is normal to x_b, and points downwards. The second axis y_b is normal to the other two, and oriented to the right in order to obtain a right-handed frame.

Body Reference Frame (II)

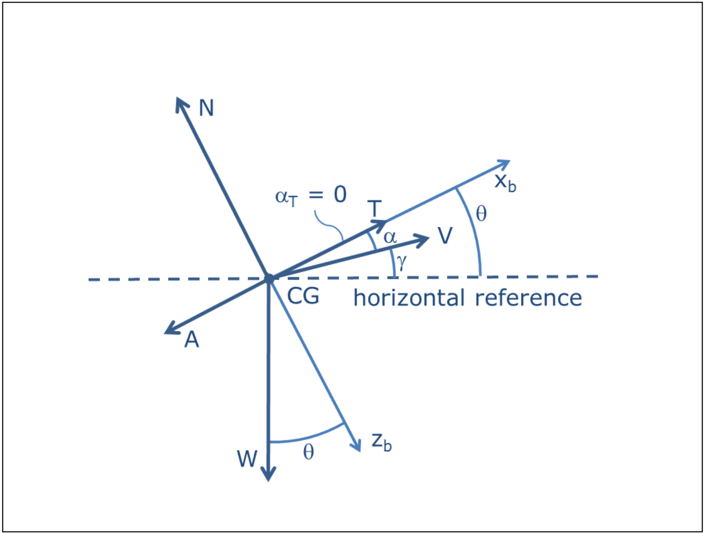
The body reference frame is commonly adopted for expressing the equations of the dynamics of the aircraft, and it is very convenient for the inertial characteristics of the aircraft are constant if measured in this reference. Furthermore, on real aircraft the accelerometers provide measurements defined with respect to this frame. The body reference for stability (stability axes) is a particular body reference, often adopted to analyze the dynamic stability of the aircraft. This frame features an x_b axis aligned with the direction of the projection of the speed vector on the longitudinal plane at the beginning of the considered time frame, i.e. typically for $t = 0$. The frame then follows the aircraft, as can be expected from any body reference frame.

Body Reference Frame (III)

To align NED with a Body frame, three rotations are needed:

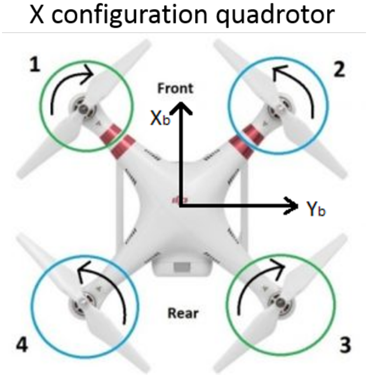
- 1^{st} rot. ($\psi_B =$ *Heading*): positive rotation about Z_V. The axis X'_V of the intermediate frame F'_V lies in the local horizontal plane. Neglecting the effects of magnetic deviation, ψ_B is the magnetic course indicated by the compass.
- 2^{nd} rot. ($\theta_B =$ *pitch*): rotation about Y'_V, aligns X'_V with X_B. The pitch angle is a measure of aircraft inclination referred to the horizon.
- 3^{rd} rot. ($\phi_B =$ *roll*): rotation about X_B aligns the two reference frame. The roll angle is the measure of lateral bank.

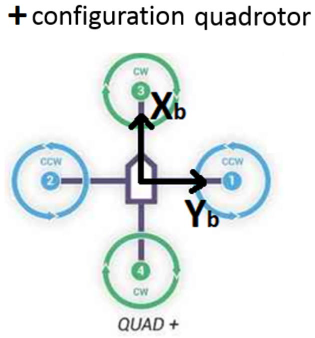
Body Reference Frame (IV)



Body Reference Frame (V)

The origin is the CoG of the UAV.
X_B and Z_B lie in the aircraft plane of symmetry, with X_B generally parallel to the fuselage reference line and Z_B directed from upper to lower surface of wing airfoil. Y_B axis is selected so that the frame is right-handed.





Body Reference Frame (VI)

For the quadrotor UAV, the Body reference frame is a Principal Axes of Inertia. $\Rightarrow$ **THE TENSOR OF INERTIA IS DIAGONAL!!**

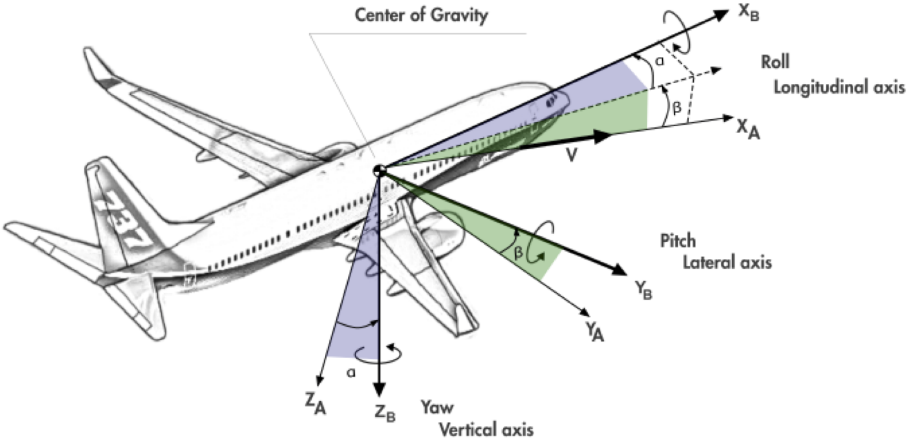
The P.I. reference frame has the axes directed along the aircraft principal axes of the inertia.

The Z_B direction is considered positive from upper to lower surface of wing airfoil. X_B is orthogonal to Z_B, pointing in forward direction, whereas Y_B completes the right handed reference frame.

Wind Axes (I)

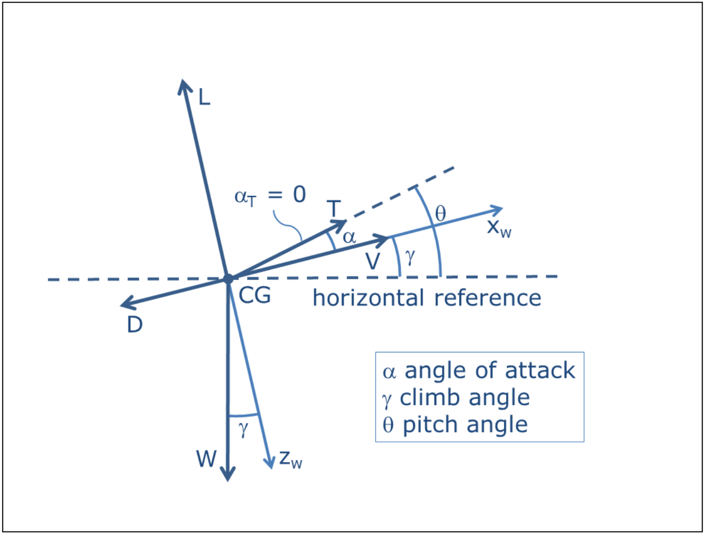
The wind reference system is of use for studying the conditions of static equilibrium, cause the usual components of the aerodynamic force exerted on the aircraft will be parallel to z_w (lift) and x_w (drag). On the other hand, the axes of the wind frame rotate with respect to the aircraft body, hence the moments of inertia are not constant in this reference. Mainly for this reason the wind reference system is not commonly used for the analysis of flight dynamics, except when the flight trajectory lies in the longitudinal plane of symmetry of the aircraft (the inertia J_y would be constant in that case).

Wind Axes (II)



Aerodynamic coordinate system (wind axes)

Wind Axes (III)



Wind Axes (IV)

The reference frame has its origin fixed at the aircraft CoG.
The longitudinal axis X_W is aligned with the airspeed direction
$V = V_E + w$
(aircraft velocity relative to the atmosphere).
The Z-axis Z_W lies in the plane of symmetry of the aircraft, directed from upper to lower surface of wing airfoil.
Y_W is orthogonal to X_W, oriented from left to right with respect to CoG trajectory.
This reference frame is usually used for FIXED-WING UAV.

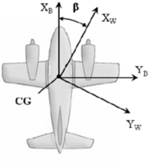
Wind Axes (V)

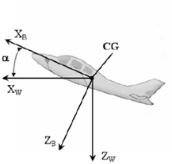
To align a Body frame with the Wind frame, two rotations are needed:

1st rot. ($\psi = -\beta$): positive rotation about Z_W aligning Y_W with Y_V.

2nd rot. ($\theta = \alpha$): rotation about Y_W, axis aligning the wind reference frame.

$$L_{WB} = [\Psi][\Theta] = \begin{bmatrix} \cos\alpha\cos\beta & \sin\beta & \sin\alpha\cos\beta \\ -\cos\alpha\sin\beta & \cos\beta & -\sin\alpha\,\lambda\sin \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix}$$





Application of Transformation Matrix (I)

An interesting example of application is the conversion of velocity components from body-fixed axes to local vertical axes ($\Phi = \varphi$, $\Theta = \theta$ and $\Psi = \psi$), a transformation of components conventionally used for the numerical integration of aircraft trajectory in 'NED' axes:

$$\begin{bmatrix} \dot{x}_v \\ \dot{y}_v \\ \dot{z}_v \end{bmatrix} = [L_{21}] \cdot \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

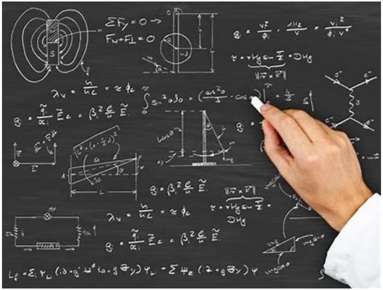
Application of Transformation Matrix (II)

Another relevant application is the conversion of velocity components from wind axes to body-fixed axes ($\Phi = 0$, $\Theta = \alpha$ and $\Psi = -\beta$):

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{bmatrix} \cdot \begin{bmatrix} V \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} V\cdot\cos\alpha\cos\beta \\ V\cdot\sin\beta \\ V\cdot\sin\alpha\cos\beta \end{bmatrix}$$

that implies $\tan\alpha = w/u$ and $\sin\beta = v/V$.

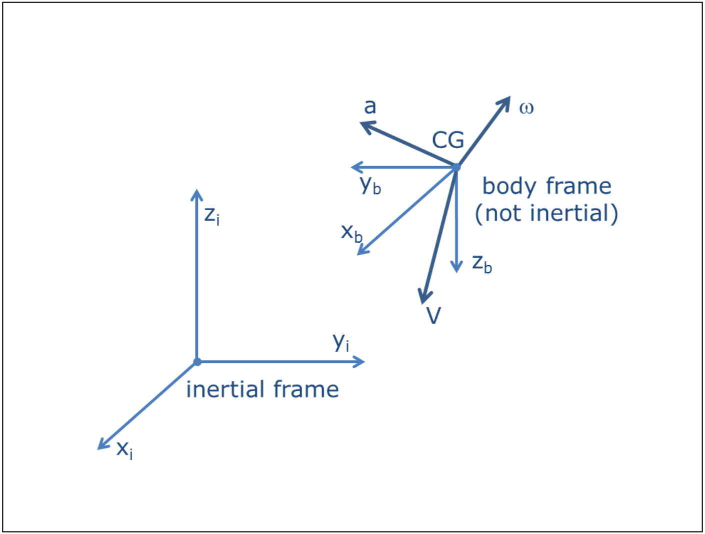
Section II: Mathematical Models



Outline

- Aircraft Model:
    1. General Equations of motion
    2. Aerodinamic Derivatives
    3. Linear Model
- Fixed Wing UAV:
    1. Nonlinear Model
    2. Linear Model
    3. Example of implementation on Matlab/Simulink
- Multi Rotor UAV:
    1. Nonlinear Model
    2. Linear Model
    3. Example of implementation on Matlab/Simulink

Aircraft Model



Aircraft Model

The motion of an airplane is affected by external **forces** (F) and **moments** (M) resulting from flight through the atmosphere, engine thrust and landing gear forces during take-off and landing. The airplane motions are calculated using the equations of rigid body dynamics as derived from Newton's laws:

$$d\mathrm{F} = \frac{d}{dt}(dm \cdot \mathrm{V})$$

where point mass dm moves with the varying velocity **V** under the influence of a force d**F**, and:

$$d\mathrm{M} = \frac{d}{dt}(dm \cdot \mathrm{r} \times \mathrm{V})$$

as the moment d**M** is equal to the time derivative of the angular momentum of the body.

Aircraft Model

The equations of motions of an airplane are derived on the basis of the above expressions, by adding up the forces as well as the moments about the center of mass of the aircraft.

The resulting **general equations of motion** in the body axes reference frame (scalar form where $I_{xy} = I_{yz} = 0$) are:

$$
\begin{aligned}
F_x &= X - W \cdot \sin\theta = m \cdot (\dot{u} + qw - rv) \\
F_y &= Y + W \cdot \cos\theta \cdot \sin\varphi = m \cdot (\dot{v} + ru - pw) \\
F_z &= Z + W \cdot \cos\theta \cdot \cos\varphi = m \cdot (\dot{w} + pv - qu) \\
M_x &= L = I_x \dot{p} + (I_z - I_y) \cdot qr - I_{xz} \cdot (\dot{r} + pq) \\
M_y &= M = I_y \dot{q} + (I_x - I_z) \cdot rp - I_{xz} \cdot (p^2 - r^2) \\
M_z &= N = I_z \dot{r} + (I_y - I_x) \cdot pq - I_{xz} \cdot (\dot{p} - rq)
\end{aligned}
$$

Aircraft Model

The above equations describe the most general motions an aircraft can perform and are based on the following assumptions:

1. the airplane **mass** is **constant** in the time interval during which the motions of the airplane are considered;
2. the airplane is a **rigid body**;
3. the **mass-distribution** of the airplane is **symmetric** relative to the XOZ-plane ($I_{xy} = I_{yz} = 0$);
4. the **rotation** of the Earth in space, as well as its **curvature** are negligible.

See **NASA SP-3070** for a more general formulation removing these limitations.

Aircraft Model

The equations of motion in the body axes reference frame can be decoupled with the application of the small perturbation theory for an initial equilibrium condition $F_{eq} = M_{eq} = 0$, $V_{eq} = (u_{eq}, v_{eq}, w_{eq})$ and $\omega_{eq} = 0$.

For the longitudinal plane (u, w, q, θ):

$$\begin{cases} \Delta F_x & = & m \cdot (\dot{u} + q\, w_{eq}) \\ \Delta F_z & = & m \cdot (\dot{w} - q\, u_{eq}) \\ \Delta M_y & = & I_y \dot{q} \end{cases}$$

For the lateral-directional plane (v, p, r, φ, ψ):

$$\begin{cases} \Delta F_y & = & m \cdot (\dot{v} + r\, u_{eq} - p\, w_{eq}) \\ \Delta M_x & = & I_x \dot{p} - I_{xz} \dot{r} \\ \Delta M_z & = & I_z \dot{r} - I_{xz} \dot{p} \end{cases}$$

This revised formulation gives the offset dynamics for the **increments** Δx with respect to the equilibrium condition i.e. the states are defined as $x = x_{eq} + \Delta x$.

Aircraft Model

The equations for the body axes velocities are:

$$\begin{cases} u = V \cdot \cos\alpha \cos\beta \\ v = V \cdot \sin\beta \\ w = V \cdot \sin\alpha \cos\beta \end{cases}$$

and

$$V = \sqrt{u^2 + v^2 + w^2}$$

$$\tan\alpha = w/u$$

$$\sin\beta = v/V$$

.

The linear accelerations are usually obtained as well (reference inertial accelerations):

$$
\begin{aligned}
a_x &= \dot{u} + qw - rv + g\sin\theta \\
a_y &= \dot{v} + ru - pw - g\cos\theta\sin\varphi \\
a_z &= \dot{w} + pv - qu - g\cos\theta\cos\varphi
\end{aligned}
$$

Aircraft Model

The equations for the Earth-relative NED velocities are:

$$
\begin{aligned}
\dot{h} &= V \cdot [\cos\alpha \, \cos\beta \, \sin\theta - \sin\beta \, \sin\varphi \, \cos\theta \\
&\quad -\sin\alpha \, \cos\beta \, \cos\varphi \, \cos\theta]
\end{aligned}
$$

$$
\begin{aligned}
\dot{x} &= V \cdot [\cos\alpha \, \cos\beta \, \cos\theta \, \cos\psi + \sin\beta(\sin\varphi \, \sin\theta \, \cos\psi - \cos\varphi \, \sin\psi) \\
&\quad +\sin\alpha\cos\beta(\cos\varphi \, \sin\theta \, \cos\psi + \sin\varphi \, \sin\psi)]
\end{aligned}
$$

$$
\begin{aligned}
\dot{y} &= V \cdot [\cos\alpha \, \cos\beta \, \cos\theta \, \sin\psi + \sin\beta \, (\cos\varphi \, \cos\psi \, + \\
&\quad \sin\varphi \, \sin\theta \, \sin\psi \,) + \sin\alpha \, \cos\beta \, (\cos\varphi \, \sin\theta \, \sin\psi - \\
&\quad \sin\varphi \, \cos\psi)]
\end{aligned}
$$

The **kinematic equations** should be added, expressing the relations between the rates of change of the attitude angles and the angular velocities about the aircraft body axes:

$$\begin{cases} \dot{\varphi} &= p + q \cdot \sin\varphi \cdot \tan\theta + r \cdot \cos\varphi \cdot \tan\theta \\ \dot{\theta} &= q \cdot \cos\varphi - r \cdot \sin\varphi \\ \dot{\psi} &= q \cdot \dfrac{\sin\varphi}{cos\theta} + r \cdot \dfrac{\cos\varphi}{cos\theta} \end{cases}$$

Aircraft Model

The three Euler's equations have singularities at $\theta = \pm 90^\circ$. Furthermore, the Euler's angles may integrate up to values outside the normal $\pm 90^\circ$ range of pitch, and the normal $\pm 180^\circ$ range of the roll and yaw angles. This wraparound problem may make it difficult to determine the attitudes uniquely. Finally, observe that the equations are linear in p, q, r but nonlinear in terms of the desired Euler's angles.

A possible solution is the use of **quaternions** to represent the orientation of the airplane body-fixed frame with respect to the Earth fixed reference frame.

Euler's Theorem shows that any two orientations can be related by a single rotation about some axis (not necessarily a principle axis). This means that we can represent an arbitrary orientation as a rotation about some unit axis by some angle (4 numbers: q_0, q_1, q_2, q_3) (axis/angle form). Alternately, we can scale the axis by the angle and compact it down to a single 3D vector (rotation vector).

Quaternions are an interesting mathematical concept with a deep relationship with the foundations of algebra and number theory. Quaternions are actually an extension to complex numbers. Invented by W.R. Hamilton in 1843. In practice, they are most useful to us as a means of representing orientations.

Storing an orientation as an axis and an angle uses 4 numbers, but Euler's theorem says that we only need 3 numbers to represent an orientation.

Mathematically, this means that we are using 4 degrees of freedom to represent a 3 degrees of freedom value.

This implies that there is possibly extra or redundant information in the axis/angle format.

The redundancy manifests itself in the magnitude of the axis vector. The magnitude carries no information, and so it is redundant. To remove the redundancy, we choose to normalize the axis, thus constraining the extra degree of freedom.

Aircraft Model

The rates of change of the quaternion parameters in term of the three-body-axes rotational rates (p, q, r) are given by:

$$\frac{d}{dt}\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = -\frac{1}{2}\begin{bmatrix} 0 & p & q & r \\ -p & 0 & -r & q \\ -q & r & 0 & -p \\ -r & -q & p & 0 \end{bmatrix}\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}$$

These four equations replace in the numerical model the three conventional Euler's equation.

Aircraft Model

The Euler's angles are computed (when required) only for feedback control and aircraft attitude visualization:

$$\begin{cases} \theta = sin^{-1}\left[-2(q_1 q_3 - q_0 q_2)\right] \\ \\ \psi = tan^{-1}\left[\dfrac{2(q_1 q_2 + q_0 q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2}\right] \\ \\ \varphi = tan^{-1}\left[\dfrac{2(q_2 q_3 + q_0 q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2}\right] \end{cases}$$

Because the quaternion differential equations are linear in both the three rotational rates and the four quaternion parameters, the computer programming is simplified. The singularities and the wraparound problem are removed. Finally, none of the quaternion elements can exceed unit magnitude.

Aircraft Model

The quaternion parameters are related to the Euler's angles by the following equations:

$$
\begin{aligned}
q_0 &= \pm(\cos(\varphi/2)\cos(\theta/2)\cos(\psi/2) + \sin(\varphi/2)\sin(\theta/2)\sin(\psi/2)) \\
q_1 &= \pm(\sin(\varphi/2)\cos(\theta/2)\cos(\psi/2) - \cos(\varphi/2)\sin(\theta/2)\sin(\psi/2)) \\
q_2 &= \pm(\cos(\varphi/2)\sin(\theta/2)\cos(\psi/2) + \sin(\varphi/2)\cos(\theta/2)\sin(\psi/2)) \\
q_3 &= \pm(\cos(\varphi/2)\cos(\theta/2)\sin(\psi/2) - \sin(\varphi/2)\sin(\theta/2)\cos(\psi/2))
\end{aligned}
$$

Aircraft Model

An expression of the change of the active force and moment components in terms of problem variables is needed. The author G. H. Bryan (in *Stability in Aviation*, McMillan, London, 1911) introduced a linear representation of these changes ($F = F_x, F_z, M_y$ due to weight W, propulsion T and aerodynamics X, Z, M) with respect to the characteristic variables of the longitudinal motion, and their time derivatives, formally yielding (for both force and moment):

$$\Delta F = \Delta F(V, \alpha, \theta, \dot{V}, \dot{\alpha}, \dot{\theta} = q, \delta_e)$$

where

$$\begin{cases} F_x = F_{x_{eq}} + \Delta F_x = \Delta F_x \\ F_z = F_{z_{eq}} + \Delta F_z = \Delta F_z \\ M_y = M_{eq} + \Delta M_y = \Delta M \end{cases}$$

Assuming a small change (first order) ΔF, the dependence with respect to all such variables can be expressed as in the following equation for the longitudinal plane, making use of the dimensional and nondimensional stability (or control) derivatives,

$$\frac{\partial F}{\partial x} \cdot \Delta x = F_x \cdot \Delta x \rightarrow \frac{\partial C_F}{\partial \hat{x}} \cdot \Delta \hat{x} = C_{F_X} \cdot \Delta \hat{x}$$

that are applicable within the linearity assumptions of small perturbation theory:

$$\Delta F = \frac{\partial F}{\partial V}\Delta V + \frac{\partial F}{\partial \alpha}\Delta \alpha + \frac{\partial F}{\partial \theta}\Delta \theta + \frac{\partial F}{\partial \dot{V}}\dot{V} + \frac{\partial F}{\partial \dot{\alpha}}\dot{\alpha} + \frac{\partial F}{\partial q}q + \frac{\partial F}{\partial \delta_e}\delta_e$$

Aircraft Model

The introduction of $\dot{V}$ and $\dot{\alpha}$ in the first order Taylor's linearized polynomial expression is rather atypical, and can be justified on the basis of empirical considerations. Usually, the stability derivatives are defined with respect to the unknowns of the differential problem (V, α, θ, q), and not with respect to their derivatives ($\dot{V}$, $\dot{\alpha}$, which depend on V and α respectively). On the other hand, a better adherence to the real data was discovered through experimentation on the model extended including the effect of $\dot{V}$ and $\dot{\alpha}$.

An equivalent formulation for the lateral-directional plane ($F = F_y, M_x, M_z$) can be derived, formally yielding for both force and moment the following result:

$$\Delta F = \Delta F(\beta, \varphi, \dot{\beta}, \dot{\varphi} = p, \dot{\psi} = r, \delta_a, \delta_r)$$

and

$$\Delta F = \frac{\partial F}{\partial \beta}\Delta\beta + \frac{\partial F}{\partial \varphi}\Delta\varphi + \frac{\partial F}{\partial \dot{\beta}}\dot{\beta} + \frac{\partial F}{\partial p}p + \frac{\partial F}{\partial r}r + \frac{\partial F}{\partial \delta_a}\delta_a + \frac{\partial F}{\partial \delta_r}\delta_r$$

where $\dot{\beta}$ depend on β (similar to $\dot{\alpha}$ for the longitudinal plane).

The stability and control derivatives can be obtained with analytical, numerical and experimental methods (wind tunnel tests and flight experiments).

The major advantage of this approach is that the equations of aircraft motion are reformulated in a linearized form that is compatible with the classical methods of stability analysis (based on eigenvalues) and linear control theory:

$$\{\dot{x}\} = [A] \cdot \{x\} + [B] \cdot \{u\}$$

The major limitation is the restricted validity for local linearity of aerodynamic forces and moments and bounded perturbations, precluding the analysis of aircraft response for large amplitude maneuvers and large control inputs.

Aircraft Model

The availability of computational tools allows a more general implementation of aerodynamic force/moment changes, based on the multi-variable interpolation (linear or polynomial vs α, β, M, ..., δ) of a database, applicable for a wider set of flight conditions:

$$X = T(\tau) + \frac{\rho V^2 S}{2} \cdot C_X(\alpha, \beta, \hat{q}, \delta_e) \qquad L = \frac{\rho V^2 Sb}{2} \cdot C_l(\alpha, \beta, \hat{p}, \hat{r}, \delta_a, \delta_r)$$

$$Y = \frac{\rho V^2 S}{2} \cdot C_Y(\alpha, \beta, \hat{p}, \hat{r}, \delta_a, \delta_r) \qquad M = \frac{\rho V^2 Sc}{2} \cdot C_m(\alpha, \beta, \hat{q}, \delta_e)$$

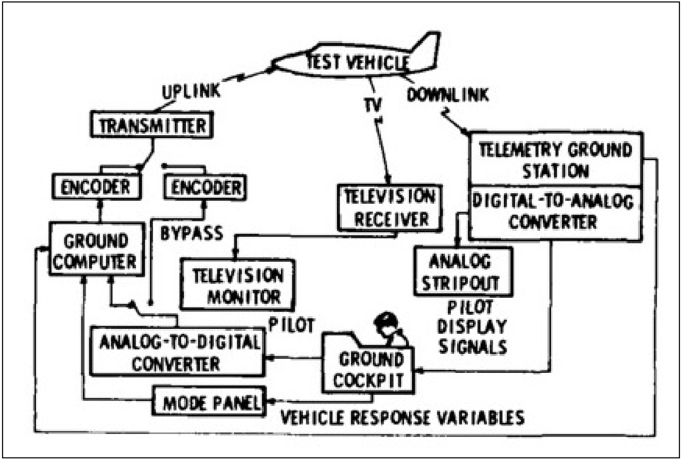
$$Z = \frac{\rho V^2 S}{2} \cdot C_Z(\alpha, \beta, \hat{q}, \delta_e) \qquad N = \frac{\rho V^2 Sb}{2} \cdot C_n(\alpha, \beta, \hat{p}, \hat{r}, \delta_a, \delta_r)$$

where[1]

$$\hat{p} = pb2V \qquad \hat{q} = qc2V \qquad \hat{r} = rb2V$$

[1] c: wing chord - b: wing span - $\delta_e/\delta_a/\delta_r$: elevator/aileron/rudder deflection - τ: throttle setting (%)

Aircraft Model



Flight test experiments

Aircraft Model

Flight test experiments (scaled vehicle)

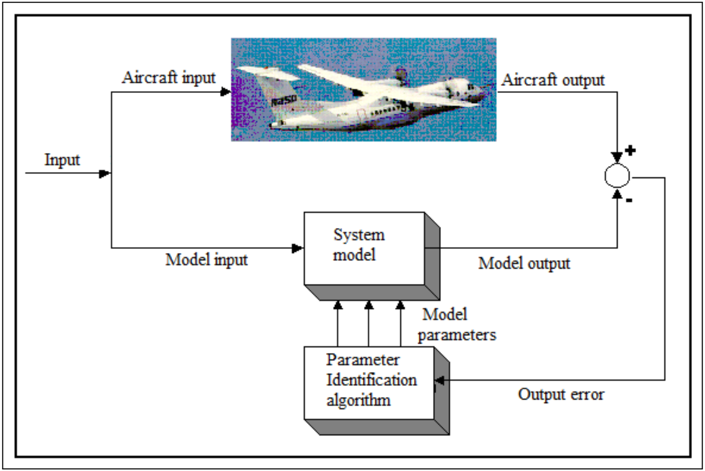
Aircraft Model

Flight test experiments (scaled vehicle)

Aircraft Model

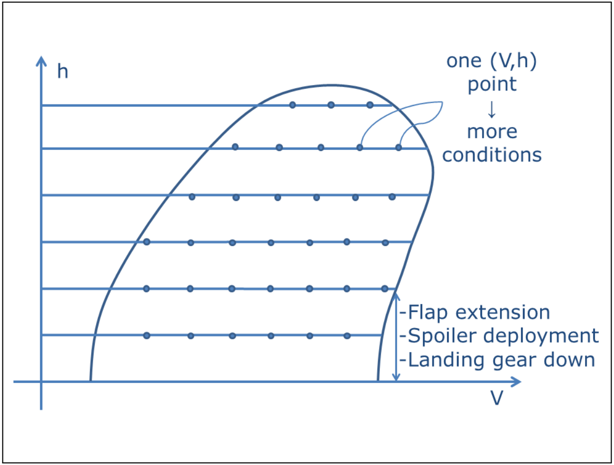
Flight test experiments (full size vehicle)

Aircraft Model



Flight test experiments (parameter identification)

Aircraft Model



Flight test diagram (parameter identification)

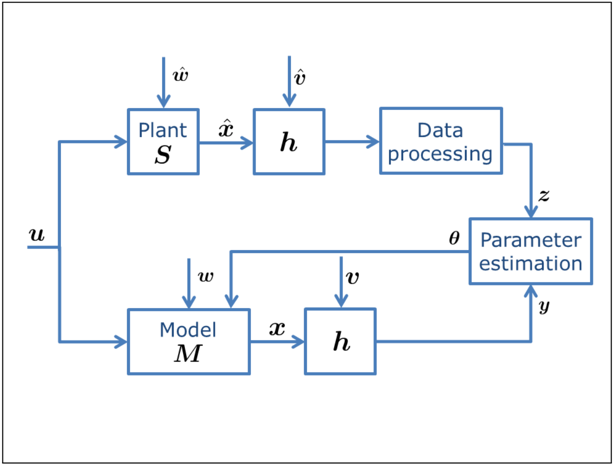
A problem of parameter identification is that of finding the values of the parameters of a model with assigned structure that best match experimental observations in a statistical sense.

Parameter identification is a sub-problem of model identification, which is first focused on the formulation of a model structure suitable for mathematically describing a certain physical effect. In the case of aircraft dynamics, it is possible to formulate two complementary models for the longitudinal and for the lateral-directional dynamics, which considered together provide for a detailed description of the behavior of the aircraft in most configurations of practical interest. For this reason, the structure of the model in the case of aircraft can be considered assigned.

Aircraft Model

What is generally intended with parameter identification is the process through which the results of experiments from flight testing (or in a previous design stage also from testing in the wind tunnel) are treated to find the actual values of the coefficients of the dynamic model of an aircraft. The result of this phase can be the finalized values of these coefficients, which can be used for instance for setting up a sophisticated, high-fidelity virtual model of the aircraft to be used as a simulator for crew training or for designing a control or guidance system (autopilot). As the most accurate and high-fidelity information is required in this final stage, flight testing on an assembled flying prototype of the machine is typically involved, so this kind of analysis is usually performed in the final stages of aircraft design.

Aircraft Model



Conceptual scheme of the parameter identification process

Aircraft Model

The basic concept of parameter identification can be explained considering a model of a plant $M(p)$ function of an array of parameters p. Let the model be constituted by the following equations:

$$\dot{x} = f(x, u, p) + Fw$$
$$y = h(x, u, p) + Gv$$
$$x(0) = x_0$$

where x is the state array of the system, u the input array, y the output array, x_0 is the initial condition of the state, F and G are process and measurement noise distribution matrices, and finally w and v are arrays of process and measurement noise.

Aircraft Model

The real plant S is characterized by state $\hat{x}$ and is disturbed by a process noise $\hat{w}$. The measurement function h is interested by a measurement noise $\hat{v}$, and produces the output z.

The synthetic model M, as previously noted, is characterized by the state array x, a noise v insists on the measurement, and the array of measurements is y.

The parameter estimation block in the figure performs the estimation of the parameters θ, based on the knowledge of the error $e = y - z$. The estimation block produces the output array θ of the parameters which is fed to the model $M(p)$, and which is such that the error e be minimized.

Aircraft Model

In real practice, test flights are carried out with prescribed time histories of input u, such that some relevant behavior of the aircraft be excited. Large amounts of data are collected from these flights, during which the input u and the resulting output z are simultaneously recorded. The parameter identification phase can be carried out working off-line (i.e. after flying testing has been completed), running (i.e. integrating) the synthetic model M based on some parameters, given a time history of the input and generating a time history of the output y. The parameters θ are tuned based on some logic (i.e. on a parameter identification technique) in order to reduce the error e between the output of the plant z and of the synthetic system y.

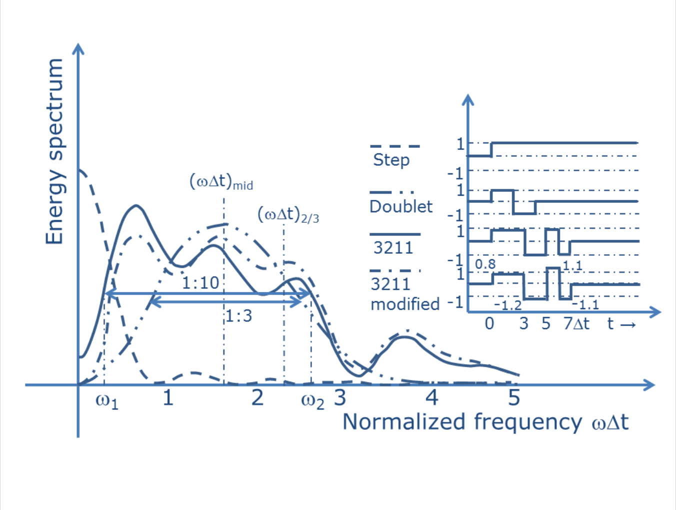
Aircraft Model

Starting from an equilibrium condition, the process of parameter identification will involve performing a series of maneuvers aimed at the excitation of the various modes of the aircraft. These maneuvers should start from and return to an equilibrium condition. Typical maneuvers are:

- short period excitation
- phugoid excitation
- level turn maneuver
- bank-to-bank maneuver
- dutch roll excitation
- thrust change maneuver

For an increased statistical relevance of the testing phase, it is common practice to perform every maneuver at least three times.

Aircraft Model



Power spectrum of notable signals

Aircraft Model

The system model M can be linearized in proximity of an equilibrium condition, yielding

$$\begin{aligned} \dot{x} &= A(p)x + B(p)(u - \Delta u) + Fw \\ y &= C(p)x + D(p)(u - \Delta u) + Gv + \Delta z \\ x(0) &= x_0 \end{aligned} \tag{4}$$

which accounts for biases in the input and output. As previously stated, the model parameters may be defined as $\theta = \{p, x_0, \Delta u, \Delta z\}$.

It is clear that among the target parameters of the identification process we can also include the coefficients (not necessarily all of them) in the system and control matrices A and B.

Aircraft Model

In general, several methods have been studied to cope with the majority of identification problems. They can be often basically categorized into time domain methods and frequency domain methods. What changes between these two categories is the type of signal to be analyzed, and the target of the identification process.

Methods in time domain are used with signals defined as time sequences (continuous or discrete). The aim of such methods is the identification of the coefficients of the state equation of the system, namely the coefficients in matrices A and B for a linear system.

Two techniques that can be profitably used for the identification of the parameters of the dynamic model of an aircraft are namely the **least squares method** and the **maximum likelihood method**.

Methods in frequency domain treat signals defined through their frequency content, and aim at finding the coefficients of a frequency representation of the considered system, i.e. its transfer function. All methods in frequency domain can be applied to linear systems only, as the whole theory of the frequency response only applies to such systems. Furthermore, in order to make possible the identification of the frequency response over a sufficiently broad frequency spectrum, the input signal must be suitably rich in frequency. This poses some constraints on the design of the input which are not easy to meet if the system is controlled through relatively slow control devices. As a consequence, these techniques are not widely applicable in general.

Numerical methods compute approximate solutions to mathematical problems by discretizing continuum processes. Numerical methods are used when analytical or symbolic approaches to solving math problems are computationally difficult. Among these methods, **numerical integration** is the most relevant application in the domain of flight simulation.

The equations of flight motion are approximated (in most cases) by finite difference equations of at least the same order of the equations of motion for consistency.

Anyway, numerous high order methods[2] have been proposed for the solution of differential equations implemented in flight simulators.

[2] F.M. Cardullo, B. Kaczmarck, B.J. Waycechowsky, A comparison of several numerical integration algorithms employed in real-time simulation, AIAA Flight Simulation Technologies Conference, New Orleans, USA, 1991.

Aircraft Model

Any finite-difference approximation of order greater than the order of the equations of motion introduces extraneous roots. In addition to the extraneous roots are other roots which approach the true roots of the differential equation in the limit as the interval approaches zero (that is, $h \to 0$ where h is the integration interval size) so that nh (n is positive integer) is finite and equal to t. If the extraneous roots are unstable, the approximate solution is sensitive to small errors (for example, starting error or roundoff error) introduced in the calculation. This sensitivity (called **numerical instability**) usually occurs when the sample rate and the systems frequencies are the same order of magnitude.

Aircraft Model

Generally we will prefer constant time step explicit numerical integration methods for real time solutions. These are explicit because they are based on previous values, whereas in implicit methods, the predicted value is used in the solution (it occurs on both sides of the equations). On the other hand, the advantage of implicit solvers is that they may not suffer instability issues even if computationally slower or less accurate than equivalent explicit schemes.

Note that variable step/variable order or recursive numerical methods **may not fit** real time flight simulation applications, where the frame rate is held constant.

Aircraft Model

Many linear numerical integration techniques with single and multistep are available which can also be classified into implicit and explicit numerical integration techniques. With respect to the stability and accuracy, each of these numerical integration techniques has advantages and disadvantages. Depending on the performance, these methods can be suitably used for stiff and non-stiff systems. Methods not designed for stiff problems (i.e. prone to numerical instability) must use time steps small enough to resolve the fastest possible changes, which makes them rather ineffective on intervals where the solution changes slowly.

Aircraft Model

Conceptually, a numerical method starts from an initial point and then takes a short step forward in time to find the next solution point. The process continues with subsequent steps to map out the solution.

Single-step methods (such as Euler's method) refer to only one previous point and its derivative to determine the current value.

Methods such as Runge-Kutta take some intermediate steps (for example, a half-step) to obtain a higher order method, but then discard all previous information before taking a second step.

Linear Multistep Methods (LMMs) differently attempt to gain efficiency by keeping and using the information from previous steps rather than discarding it. Consequently, multistep methods refer to several previous points and derivative values. In the case of linear multistep methods (consider Adams-ashforth method as an example), a linear combination of the previous points and derivative values is used. Hence, the linear multistep methods require past values of the state and they are **not self-starting** and they do not directly solve the initial-value problem.

Aircraft Model

The simplest Runge-Kutta (RK) method is Euler integration (Euler's method is also a one-step first order method), which merely truncates the Taylor series after the first derivative and is nevertheless very accurate.

A RK method (e.g., Euler) could be used to generate the starting values for LMMs. Higher order RK algorithms are an extension Taylor series expansion to higher orders. An important feature of the RK methods is that the only value of the state vector that is needed is the value at the beginning of the time step; this makes them **well suited** to the Ordinary Differential Equations initial value problem.

Aircraft Model

The ever increasing complexity of the math models used as a basis for real time flight simulation has continued to apply pressure on digital processor speed requirements for such simulations. More effective numerical integration algorithms can help relieve some of this pressure.

One of the most popular solutions currently in use for flight simulation is the Adams-Bashforth second order predictor method, usually referred to as AB2. Its advantages include second order accuracy with respect to integration step size, only one required pass through the state equations per integration step, and compatibility with real-time inputs. Disadvantages include stability problems associated with extraneous roots and response delays of one or two frames following transient inputs. Note that the AB2 method is explicit and hence only conditionally stable.

Aircraft Model

With the enhancement in required computing speeds, higher order linear multistep predictor-corrector methods (consider Adams-Bashforth-Moulton ABp-AMp method as an example) may also provide a solution with strong stability and significant improvements in accuracy without affecting the real-time computational rate.

In general, Adams-Bashforth (AB), Adams-Moulton (AM) and Adams predictor-corrector methods are widely used multistep methods for approximating solutions to first order differential equations. These methods maintain reasonably good accuracy and stability properties and have lower computational costs than equivalent order Runge-Kutta methods.

Euler's method

Consider the function $x(t)$. If the solution is computed at a fixed rate, the interval between t_n and t_{n+1} is h (for all n), then the value at x_n at time t_n is known and the value x_{n+1} at time t_{n+1} is required.

Let f_n be the gradient at t_n, then

$$\boxed{f_n = \frac{x_{n+1} - x_n}{h}}$$

where h is the step length, then

$$\boxed{x_{n+1} = x_n + h \cdot f_n}$$

For the series of inputs to the integrator at time t_n we can compute the next output of the integrator (at time t_{n+1}), simply by applying the formula at every step. In other words, the solution is executed frame by frame at the frame rate determined by the step length h.

Clearly, this is an approximation: there is an error ϵ between the true solution and the computed value at t_{n+1} . It can be shown that for the forward Euler's method, this error is of the order

$$\boxed{\epsilon_{n+1} \approx \frac{h^2}{2}}$$

This is an inherent problem with these methods. The error introduced by the numerical integration algorithm will continue to grow and can only be reduced by decreasing the step length or by moving to a higher order method. Although various improvements have been proposed to Euler's first order method, if numerical accuracy is an issue, higher order integration methods must be used.

Runge-Kutta-Gill method

The Runge-Kutta-Gill method is the most widely used single step method for approximating the solution of the differential equation x'(t) = f(t,x). When t = t_n the method numerically evaluates f(t,x) four times per step:

$$\boxed{x_{n+1} = x_n + 1/6 \cdot (k_1 + (2-\sqrt{2}) \cdot k_2 + (2+\sqrt{2}) \cdot k_3 + k_4)}$$

where

$$k_1 = h \cdot f(t_n, x_n)$$
$$k_2 = h \cdot f(t_n + h/2, x_n + k_1/2)$$
$$k_3 = h \cdot f(t_n + h/2, x_n + (-1+\sqrt{2})/2 \cdot k_1 + (2-\sqrt{2})/2 \cdot k_2)$$
$$k_4 = h \cdot f(t_n + h, x_n - \sqrt{2}/2 \cdot k_2 + (2+\sqrt{2})/2 \cdot k_3)$$
$$t_{n+1} = t_n + h$$

This version of the Runge-Kutta method is fourth order:

$$\boxed{\epsilon_{n+1} \approx O(h^5)}$$

Adams-Bashforth method (AB2)

The second order Adams-Bashforth (AB2) method is a linear multistep predictor method given by

$$\boxed{x_{n+1} = x_n + \frac{h}{2} \cdot (3 \cdot f(t_n, x_n) - f(t_{n-1}, x_{n-1}))}$$

Note that the AB2 method is explicit and hence only conditionally stable. Moreover, the AB2 method requires the solution from the n-1th and the nth steps to find the solution at the n+1th step.

This version of Adams-Bashforth predictor method is second order:

$$\boxed{\epsilon_{n+1} \approx O(h^3)}$$

Adams-Moulton method (AM2)

The second order Adams-Moulton (AM2) is an implicit technique (linear multistep corrector method), sometimes referred to as the trapezoidal rule. The time-stepping equation for AM2 is given by

$$\boxed{x_{n+1} = x_n + \frac{h}{2} \cdot \left(f\left(t_{n+1}, x_{n+1}\right) + f\left(t_n, x_n\right)\right)}$$

The implicit nature of the method is evident. For a non-linear initial value problem, we have to solve a non-linear algebraic equation at every time step. This is much more expensive as compared to the explicit AB2 method. However, being an implicit technique, AM2 does not suffer from the numerical instability of the AB2 for relatively large values of the time step. Once again, it is a trade-off between stability and computational cost, since both AM2 and AB2 are second order accurate.

One way to implement an implicit scheme is to couple it with a corresponding explicit scheme of the same order (predictor-corrector approach).

The errors of numerical computation are of two classes: errors due to representing numbers by a finite word size (that is, **roundoff**) and errors due to neglecting higher order terms in approximate techniques (that is, **truncation**). Although the effects due to truncation can, more or less, be determined, the propagated error associated with round-off can seldom be rigorously studied for practical problems. Rather, the propagated roundoff error is studied through model building and experimentation.

Aircraft Model

The solution of the equations is computed to a known resolution, in terms of floating point accuracy, as specified by IEEE 754. Rounding errors in the computation of equations can introduce noise into the solution. Generally, rounding errors can be ignored as their effects can be reduced by using extended precision arithmetic, when it is anticipated to be a problem. In ordinary calculations, the roundoff error is so small (on the order of decimal 10^{-15}) that under normal circumstances you won't notice. However, we may encounter numerous situations where the roundoff error matters (large cumulative effect).

Aircraft Model

Many numerical integration methods are based on truncation of Taylor series, which defines the order of the method. Hence, generally higher order methods are more accurate.

Truncation errors generated in a previous step are included in the computation of the current step and will accumulate. Although this limitation is unacceptable in the solution of an equation over a long period of time, in systems with feedback, numerical errors are attenuated as a result of the feedback mechanism.

Aircraft Model

The step length may affect the truncation error and the accuracy of the solution. Clearly, a very coarse integration step can lead to a very inaccurate solution and intuitively, it would seem that the smaller the step, the more accurate is the solution. However, in real-time systems, the solution is usually implemented at an iteration rate which is matched to the time constants of the system, bounding the selection of the step length.

As a conclusion, the system designer is faced with three compromises in selecting a numerical integration method:

- The speed of the method (if several thousand integrations are executed per second, the impact on processor throughput can be significant)
- The accuracy of the method
- The stability of the method.

The effect of speed and accuracy can be determined from knowledge of the step length, the type of integration method, the truncation error term (given by the order of the method), the system structure and the number of steps to be executed.

Aircraft Model

Stability is much harder to assess. Considerable care is needed to ensure that the numerical method chosen to implement a particular model, does not excite instabilities which are not in the model dynamics (numerical artifact). It depends on the system dynamics, the type of integration method, the order of the method and the step length. One way to assess the impact of the numerical method in terms of stability is to apply it to a well-conditioned equation and then investigate the limits of the onset of instability. The use of implicit solvers is an option to avoid numerical instability as they may be unconditionally stable, which however does not mean that they are faster or very accurate.

Among numerical methods, **root-finding** is another relevant application for flight simulation code development.

One of the most basic techniques is Newton's method, a root-finding algorithm which produces successively better approximations to the roots (or zeroes) of a real-valued function. The method can also be extended to complex functions (eigenvalues) and to systems of equations (aircraft trim problems and/or initialization of numerical integration schemes). If the function is not continuously differentiable in a neighborhood of the root then it is possible that Newton's method will always diverge and fail, unless the solution is guessed on the first try.

The most basic version starts with a single-variable function f defined for a real variable x, the function's derivative f', and an initial guess x_0 for a root of f. If the function satisfies sufficient assumptions and the initial guess is close, then

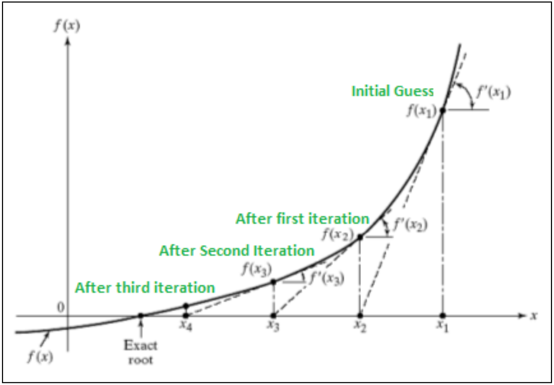
$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

is a better approximation of the root than x_0. Geometrically, (x1, 0) is the intersection of the x-axis and the tangent of the graph of f at (x0, f'(x0)): that is, the improved guess is the unique root of the linear approximation at the initial point. The process is repeated as

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

until a sufficiently precise value is reached.

Aircraft Model



Aircraft Model

An example: to evaluate the equilibrium conditions (aircraft trim in the longitudinal plane) it is necessary to impose the residual functions $f_1 \ldots f_4$ equal to zero:

$$\begin{aligned} f_1 &= M \\ f_2 &= X - m \cdot g \cdot \sin\theta \\ f_3 &= Z + m \cdot g \cdot \cos\theta \\ f_4 &= \theta - \gamma - \alpha \end{aligned}$$

This function is performed by the numerical solver, which varies iteratively state variables in the vector

$$\mathsf{x} = [\alpha \, \mathrm{V} \, \theta \, \gamma]^{\mathrm{T}}$$

so that the final residual functions are less than a given tolerance ϵ.

Aircraft Model

All flight simulation models need to access databases structured in tabular format (**table look up process**) for many different reasons such as interpolation of aerodynamic coefficients and propulsion system mapping just as an example.

Generally, an index search or look-up process will be performed first to locate the data and this is followed by **linear interpolation**. The following steps need to be performed for table look-up process:

- First we should decide between which pair of values in the table the current input value of independent variable (X) lies
- Next, calculate the local slope
- Finally, apply the linear interpolation formula.

For real-time simulation, it is always important to save the processing time. One of the techniques to save the processing time is to remember the index of the lower pair the interpolation range used in the previous iteration. The value of the independent variable (X) is unlikely to have changed substantially from one time step to the next, and hence it is a good first try to use the same interval as before and thus save time in searching from one end of the table each time.

Aircraft Model

Huge and complex aerodynamic and engine database has to be handled in such a way that it can be easily read and interpolated for a given set of input conditions. One way of ensuring the speed required for real-time simulation, is to have uniformly spaced database. For this, the normal practice is to convert the supplied database with non-uniform break points for independent variables to equi-spaced format. It is necessary to choose an appropriate step size for independent variables such as angle of attack, Mach number, elevator deflection, angle of sideslip, converting this non-uniform database to equi-spaced format.

Generally, the total aerodynamic forces and moments depend not only on the present values of variables, such as control surface deflections, angle of attack and sideslip angle, but also on the past trajectory with respect to the surrounding airflow. This leads to aerodynamic models consisting of integrals of **indicial functions**[3][4].

A more practical and well proven alternative is to expand each of the above mentioned variables as a **truncated Taylor series** backwards in time. This results in aerodynamic models in the form of nonlinear algebraic functions (eventually reduced to a linear form) of the above mentioned variables and their derivatives with respect to time.

[3] M. Tobak, On the use of the indicial function concept in the analysis of unsteady motions of wings and wing tail combination, NACA-TR-1188, 1954

[4] M. Tobak, L.B. Schiff, Aerodynamic Mathematical Modeling - Basic Concepts, AGARD-LS-114, 1981

Aircraft Model

For the **linearized form** of the aircraft mathematical model, in the whole system of the equations of motion there are neither terms of **order higher than the first one** nor **coupled terms**.

Aerodynamic models describe the aerodynamic forces and moments which act on the aircraft during the dynamic flight maneuver to be analyzed. One can choose to **linearize** only the expressions of **aerodynamic forces and moments**: then the linearized aerodynamic model contains a set of parameters called **stability** and **control derivatives**.

Aircraft Model

Linear models of aircraft are **widely used**, not only for computer applications but also for quick approximations and desk calculations. Aerodynamics can be represented by homogeneous polynomials of the first degree in the state and control input variables of the linearized equations of motion. Such polynomials are used as linear approximations of aerodynamic forces and moments acting on the aircraft in dynamic flight conditions, avoiding the interpolation of the aerodynamic database, as the coefficients of these polynomials (aerodynamic derivatives) are normally held constant.

In general, the domain in which **linear models** are valid is restricted to **small deviations** from a nominal flight condition which is stationary.

Aircraft Model

On the other side, the advantage of using **nonlinear models** is that such models should be valid for a **larger range of flight conditions**.

In addition, the simulations of flight test maneuvers are much less constrained with respect to the amplitudes of angle of attack and airspeed excursions. One specific form of representing nonlinear aerodynamic models is by using higher order polynomials in state and control input variables. The coefficients of these polynomials are normally updated as a function of angle of attack changes during the simulation.

In principle, the domain of nonlinear models covers **larger deviations** from a given nominal flight condition, as compared to linear models.

Aircraft Model

To correctly analyze the dynamic characteristics of an aircraft, we should use a proper model, but in its formulation it is common to encounter the following typical problems:

- definition of the mathematical model with **adequate accuracy** in order to represent all the aspects that need to be studied (avoiding an overweighted model for simple analysis)
- **validation** of the model by means of correlation with experimental data, flight tests, wind tunnel tests, real behavior of the system in general
- **limitations** of the model itself due to inaccuracy or inability in understanding the system (lack of engineering insight).

Aircraft Model

In **many practical applications** the **linear model** of an aircraft is completely **sufficient**. The linear equations[5] can be formulated as the longitudinal equations including only the short-period mode and expressing perturbations with respect to a horizontal steady flight:

$$
\begin{aligned}
\dot{\alpha} &= q + \frac{\rho V_0 S}{2m}\left(C_{Z_\alpha}\alpha + C_{Z_q}\cdot\frac{q\bar{c}}{2V_0} + C_{Z_{\delta_e}}\delta_e\right)\\
\dot{q} &= \frac{\rho V_0^2 S\bar{c}}{2I_y}\left(C^*_{m_\alpha}\alpha + C^*_{m_q}\cdot\frac{q\bar{c}}{2V_0} + C^*_{m_{\delta_e}}\delta_e\right)\\
\dot{\theta} &= q\\
a_z &= V_0(\dot{\alpha} - q)
\end{aligned}
$$

[5] V. Klein, Estimation of Aircraft Aerodynamic Parameters from Flight Data, Progress in Aerospace Science, Vol. 26, 1989, pp. 1-77.

Aircraft Model

In the pitching moment equation the $\dot{\alpha}$ term is not explicitly included for it is contained in the C_m-derivatives:

$$\begin{aligned}
C^*_{m_\alpha} &= C_{m_\alpha} + \frac{\rho S\bar{c}}{4m} C_{m_{\dot{\alpha}}} C_{Z_\alpha} \\
C^*_{m_q} &= C_{m_q} + C_{m_{\dot{\alpha}}} \left(1 + \frac{\rho S\bar{c}}{4m} C_{Z_q}\right) \\
C^*_{m_{\delta_e}} &= C_{m_{\delta_e}} + \frac{\rho S\bar{c}}{4m} C_{m_{\dot{\alpha}}} C_{Z_{\delta_e}}
\end{aligned}$$

with

$$C_{m_\alpha} = \frac{\partial C_m}{\partial \alpha} \quad C_{m_{\dot{\alpha}}} = \frac{\partial C_m}{\partial \frac{\dot{\alpha}\bar{c}}{2V}} \quad C_{m_q} = \frac{\partial C_m}{\partial \frac{q\bar{c}}{2V}} \quad C_{m_{\delta_e}} = \frac{\partial C_m}{\partial \delta_e}$$

In a similar way, the lateral-directional equations for the perturbations with respect the steady flight are given as:

$$
\begin{aligned}
\dot{\beta} + r - \frac{g}{V_0}\cos\theta_0 \cdot \varphi &= \frac{\rho V_0^2 S}{2m}\left(C_{Y_\beta}\beta + C_{Y_p}\frac{pb}{2V_0} + C_{Y_r}\frac{rb}{2V_0} + C_{Y_{\delta_r}}\delta_r \right) \\
\dot{p} - \frac{I_{xz}}{I_x}\dot{r} &= \frac{\rho V_0^2 Sb}{2I_x}\left(C_{l_\beta}\beta + C_{l_p}\frac{pb}{2V_0} + C_{l_r}\frac{rb}{2V_0} \right. \\
&\quad \left. + C_{l_{\delta_a}}\delta_a + C_{l_{\delta_r}}\delta_r \right) \\
\dot{r} - \frac{I_{xz}}{I_z}\dot{p} &= \frac{\rho V_0^2 Sb}{2I_y}\left(C_{n_\beta}\beta + C_{n_p}\frac{pb}{2V_0} + C_{n_r}\frac{rb}{2V_0} \right. \\
&\quad \left. + C_{n_{\delta_a}}\delta_a + C_{n_{\delta_r}}\delta_r \right) \\
\dot{\varphi} &= p + r\tan\theta_0 \\
a_y &= V_0(\dot{\beta} + r) - (g\cos\theta_0) \cdot \varphi
\end{aligned}
$$

In high-fidelity real-time flight simulations, the nonlinear equations of motion covering the **complete flight envelope** are used, starting from the total force and moment equations. To present non-linear aerodynamic data, two possible methods are available: the **table look-up method** and the **polynomial fit to data**.

Using the polynomial fit to data, an example[6] of non-linear aerodynamic model (this aerodynamic model includes control nonlinearity and unsteady aerodynamics effects ($\dot{\alpha}$, $\dot{\beta}$) still keeping motion planes separated) in terms of dimensionless force and moment coefficients is given hereafter.

[6] M. Baarspul, A Review of Flight Simulation Techniques, Progress in Aerospace Science, Vol. 27, 1990, pp. 1-120.

Aircraft Model

$$C_X = C_{X_0} + C_{X_\alpha}\alpha + C_{X_{\alpha^2}}\alpha^2 + C_{X_{\alpha^3}}\alpha^3 + C_{X_q}\frac{q\bar{c}}{2V}$$

$$+C_{X_{\delta_r}}\delta_r + C_{X_{\delta_f}}\delta_f + C_{X_{\alpha\delta_f}}\alpha\delta_f$$

$$C_Y = C_{Y_0} + C_{Y_\beta}\beta + C_{Y_p}\frac{pb}{2V} + C_{Y_r}\frac{rb}{2V}$$

$$+C_{Y_{\delta_a}}\delta_a + C_{Y_{\delta_r}}\delta_r + C_{Y_{\delta_r\alpha}}\delta_r\alpha + C_{Y_{\dot\beta}}\frac{\dot\beta b}{2V}$$

$$C_Z = C_{Z_0} + C_{Z_\alpha}\alpha + C_{Z_{\alpha^3}}\alpha^3 + C_{Z_q}\frac{q\bar{c}}{2V}$$

$$+C_{Z_{\delta_e}}\delta_e + C_{Z_{\delta_e\beta^2}}\delta_e\beta^2 + C_{Z_{\delta_f}}\delta_f + C_{Z_{\alpha\delta_f}}\alpha\delta_f$$

Aircraft Model

$$C_l = C_{l_0} + C_{l_\beta}\beta + C_{l_p}\frac{pb}{V} + C_{l_r}\frac{rb}{2V}$$

$$+C_{l_{\delta_a}}\delta_a + C_{l_{\delta_r}}\delta_r + C_{l_{\delta_r\alpha}}\delta_r\alpha + C_{Y_{\dot\beta}}\frac{\dot\beta b}{2V}$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_{\alpha^2}}\alpha^2 + C_{m_q}\frac{q\bar{c}}{2V}$$

$$+C_{m_{\delta_e}}\delta_e + C_{m_{\beta^2}}\beta^2 + C_{m_r}\frac{rb}{2V} + C_{m_{\delta_f}}\delta_f$$

$$C_n = C_{n_0} + C_{n_\beta}\beta + C_{n_p}\frac{pb}{2V} + C_{n_r}\frac{rb}{2V}$$

$$+C_{n_{\delta_a}}\delta_a + C_{n_{\delta_r}}\delta_r + C_{n_q}\frac{q\bar{c}}{2V} + C_{n_{\beta^3}}\beta^3$$

Aircraft Model

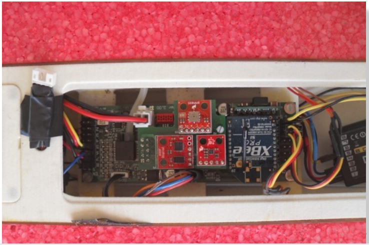
The derivation and definition of a **linear model** for a rigid aircraft of constant mass flying over a flat, nonrotating Earth can also be given with a **mixed approach** using wind and body axes,[7] retaining the aerodynamic loads in the original wind-oriented frame but giving the rotational dynamics with respect to body frame i.e. keeping the inertial properties constant with time. This formulation also provides (after some analytical manipulation of equations) an explicit formulation for angular accelerations. The thrust vector T is defined through its components along body axes (X_T, Y_T, Z_T), suitable for modelling thrust-vectoring effects. No assumption for reference trajectory or vehicle inertial symmetry is introduced.

[7] E.L. Duke, R.F. Antoniewicz, K.D. Krambeer, Derivation and Definition of a Linear Aircraft Model, NASA-RP-1207, 1988.

Other Simulation Models: Fixed-Wing UAV and Multicopters

Fixed wing UAV (I)

- Tailless Configuration
- Tractor propeller
- Wingspan = 85 cm
- Mass = 1 kg
- Endurance of about 30 minutes
- Able to perform autonomous flight



Fixed wing UAV (I)

The following assumptions are considered for the evaluation of the equations of motion:

- Aircraft is a rigid body.
- The plane of symmetry is X-Z body frame.
- Decoupling of longitudinal and lateral-directional plane.
- The equilibrium point of linearization is the level flight.
- The linearized model is used for waypoint-following path.

Model Setup

- A complete nonlinear model is considered.
- For the navigation, NED position $[x, y, h]^T$ are included.
- For controller implementation, a linearized model is also included.

- The longitudinal variables include: the longitudinal and vertical components of the airspeed (u, w), the pitch angle (θ) and rate (q).
- The lateral-directional states are: the lateral component of the airspeed (v), the roll rate (p) and angle (ϕ) and the yaw rate (r).
- The control inputs are trailing edge elevons (symmetric deflections for the elevator δ_e and antisymmetric for the aileron δ_a).

Nonlinear Model (I)

A complete nonlinear model (as defined in [1]) is a set of nine equations describing the forces, moments, angles and angular speeds which characterize the flight condition of the aircraft.

These equations of motion are referred to a body reference frame, that is fixed in the aircraft. Classical assumptions of rigid body and flat non-rotating Earth are made. These assumptions are supported by the application to a Mini-UAV.

Nonlinear Model (II)

The components of the total speed V can be expressed as follows

$$\dot{u} = \frac{F_X}{m} + qw - rv + g\sin\vartheta,$$
$$\dot{v} = \frac{F_Y}{m} - pw + ru - g\cos\vartheta\sin\phi,$$
$$\dot{w} = \frac{F_Z}{m} + pv - qu - g\cos\vartheta\cos\phi,$$

where m is the aircraft mass, $[u\ v\ w]^T$ are respectively the longitudinal, lateral and vertical components of the speed V, $[F_X\ F_Y\ F_Z]^T$ are the forces acting on the three reference axes, $[p\ q\ r]^T$ are the angular speeds along the three axes, ϑ is the pitch angle and ϕ is the roll angle.

Nonlinear Model (III)

The variation of the angular speeds $[p\ q\ r]^T$ is expressed by the following equations

$$\dot{p} = \frac{L}{J_x} + [J_{xz}(\dot{r} + pq) + qr(J_y - J_z)]J_x,$$
$$\dot{q} = \frac{M}{J_y} + \frac{[J_{xz}(r^2 - p^2) + pr(J_z - J_x)]}{J_y},$$
$$\dot{r} = \frac{N}{J_z} + \frac{[J_{xz}(\dot{p} - pq) + pq(J_x - J_y)]}{J_z},$$

where $[L\ M\ N]^T$ are the roll, pitch and yaw moments respectively, J_i are the moments of inertia with $i = x, y, z, xz$.

Nonlinear Model (IV)

The variation of the Euler angles $[\phi \ \vartheta \ \psi]^T$ is defined by the kinematic equations

$$\dot{\phi} = p + q \sin\phi \tan\vartheta + r \cos\phi \tan\vartheta,$$
$$\dot{\vartheta} = q\cos\phi - r\sin\phi,$$
$$\dot{\psi} = \frac{q\sin\phi}{\cos\vartheta} + \frac{r\cos\phi}{\cos\vartheta}.$$

For the evaluation of the aircraft navigation, the position vector $[x \ y \ h]^T$ is considered

$$\begin{aligned} V_N &= u\cos\vartheta\cos\psi + v(\sin\phi\sin\vartheta\cos\psi - \cos\phi\sin\psi) + \\ &\quad w(\cos\phi\sin\vartheta\cos\psi + \sin\phi\sin\psi), \\ V_E &= u\cos\vartheta\sin\psi + v(\sin\phi\sin\vartheta\sin\psi + \cos\phi\cos\psi) + \\ &\quad + w(\cos\phi\sin\vartheta\sin\psi - \sin\phi\cos\psi), \\ V_D &= u\sin\vartheta + v\cos\vartheta\sin\phi + w\cos\phi\cos\vartheta, \end{aligned}$$

where $[V_N \ V_E \ V_D]^T$ are the components of the total airspeed along the three axes in the vehicle-carried vertical reference frame.

Linear Model (I)

Reference flight conditions for the model are speed $U_0 = 13.5$ m/s, altitude $h_0 = 100$ m, angle of attack $\alpha_0 = 5.18$ deg and $\theta_0 = 5.18$ deg.
The equations of motion linearization procedure results in the decoupling of the longitudinal and lateral-directional planes. Each of them is modeled with standard continuous time-invariant state space representation

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \tag{5}$$

where $x(t)$ is the state vector, $u(t)$ the control signal, $y(t)$ the controlled output, A the state matrix, B the input matrix and C the output matrix.

Linear Model (II)

- The state variables in the longitudinal plane are the longitudinal component of the total airspeed u, the vertical component of the total airspeed w (and, as a consequence, the angle of attack $\alpha \simeq \frac{w}{V}$), the pitch angle θ and the pitch rate q.
- The lateral-directional states are the lateral component of the total airspeed v, the roll rate p, the yaw rate r and the roll angle ϕ.
- The aircraft control input is based on trailing edge elevon (symmetric deflection for elevator δ_e and antisymmetric for aileron δ_a). Moreover, the throttle is considered as input in the longitudinal plane.

Linear Model (III)

DIMENSIONAL DERIVATIVES

The derivatives are defined in wind axes.

$$A_{long} = \begin{bmatrix} X_u & \boxed{X_\alpha} & -g\cos\theta_0 & 0 \\ \frac{Z_u}{U_0 - Z_{\dot\alpha}} & \frac{\boxed{Z_\alpha}}{U_0 - Z_{\dot\alpha}} & \frac{-g\sin\theta_0}{U_0 - Z_{\dot\alpha}} & \frac{Z_q + U_0}{U_0 - Z_{\dot\alpha}} \\ 0 & 0 & 1 & 0 \\ M_u + \frac{M_{\dot\alpha} Z_u}{U_0 - Z_{\dot\alpha}} & M_\alpha + \frac{M_{\dot\alpha} Z_\alpha}{U_0 - Z_{\dot\alpha}} & \frac{-g\sin\theta_0 M_{\dot\alpha}}{U_0 - Z_{\dot\alpha}} & M_q + \frac{M_{\dot\alpha}(Z_q + U_0)}{U_0 - Z_{\dot\alpha}} \end{bmatrix}$$

$X_\alpha = X_w U_0$, $Z_\alpha = Z_w U_0$

Usually, $Z_{\dot\alpha} = 0$ and $M_{\dot\alpha} = 0$.

$A_{lat} = \begin{bmatrix} Y_v & Y_p & Y_r - U_0 & 0 \\ L_v & L_p & L_r & 0 \\ N_v & N_p & N_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$, where $U_0 = 13.5$ m/s is the cruise speed and $\theta_0 = 5.18$ deg.

Linear Model (IV)

The control matrices can be written in the following way.

$$B_{long} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{Z_{\delta_e}}{U_0 - Z_{\dot{\alpha}}} \\ 0 & 0 \\ 0 & M_{\delta_e} + \frac{M_{\dot{\alpha}} Z_{\delta_e}}{U_0 - Z_{\dot{\alpha}}} \end{bmatrix}$$

$$B_{lat} = \begin{bmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \end{bmatrix}$$

No rudder for the selected UAV!

Control variables: Throttle, Elevator and Aileron.

Linear Model (IV)

The aerodynamic derivatives are defined as function of the flight conditions and of the aircraft mass, of the airfoil parameters, of the propeller.
In the analyzed case, dimensional aerodynamic derivatives are considered, so we have (for example)

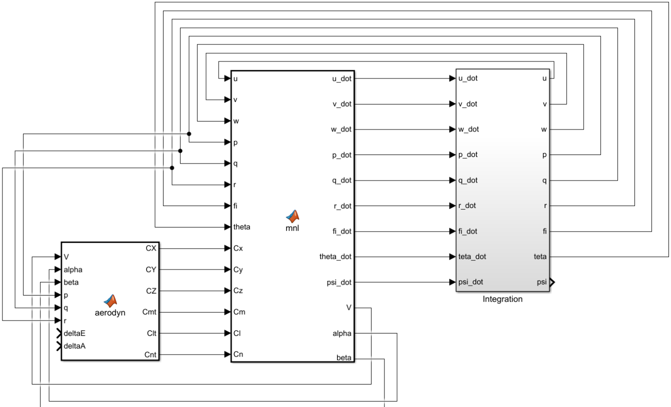
$$
\begin{aligned}
X_u &= \frac{\rho S U_0}{2m}(-3C_{D_e} - C_{D_u}) \\
X_q &= \frac{\rho S U_0 c}{4m}(-C_{L_q}) \\
Z_{\delta e} &= \frac{\rho S U0^2}{2m}(-C_{L_{\delta_e}}) \\
M_{\delta e} &= \frac{\rho S U0^2 c}{2Iy}(Cm_{\delta_e})
\end{aligned}
$$

where $C_{X_u} = C_{T_u} = -3C_{D_e} - C_{D_u}$ and $C_{D_u} = 0$ for "slow" aircraft.

Matlab/Simulink Example: FW-UAV (I)

The nonlinear model is designed as:

1. definition of the UAV parameters on a Matlab script, including the aerodynamic derivatives (both nondimensional and dimensional)
2. definition of the Equations of motion on a Matlab function, included in a Simulink file



Matlab/Simulink Example: FW-UAV (II)

Database Example in a text file, defined with ACI program

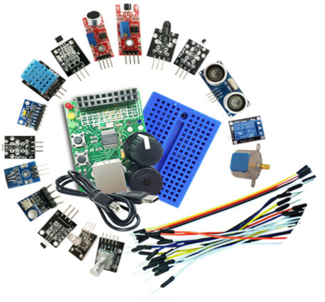
```
*      S T A N D A R D    I N P U T    F I L E
*      DATA IN S.I. UNITS            BODY AXES REFERENCE
*  AIRCRAFT DESCRIPTION  / MH850-3_tf_100m
 0.972600000000000              MASS
 0.872000000000000              WING SPAN
 0.247038336582196              WING AREA
 0.295856197079013              CHORD
0.6            MACH DRAG RISE
0.0   THRUST AXIS OFFSET (REF. TO XB ALONG X)
0.0   THRUST AXIS OFFSET (REF. TO XB ALONG Y)
0.0   THRUST AXIS OFFSET (REF. TO XB ALONG Z)
0.0   THRUST AXIS ANGULAR OFFSET (REF. TO XB / X-Y PLANE / POSITIVE RIGHT)
0.0   THRUST AXIS ANGULAR OFFSET (REF. TO XB / X-Z PLANE / POSITIVE DOWN)
       110         NUMBER OF ANGLES OF ATTACK
1          ROTARY DERIVATIVES (0/MISSING 1/PRESENT)
 0.134284878291602          CENTER OF GRAVITY REFERENCE LOCATION REF. TO CMAER
 1.088831471831585E-002         JX
 1.192188547073001E-002         JY
 2.232461745527658E-002         JZ
 4.590493189389271E-005         JXZ
0          OPTION FOR C.G. UPDATE (0/NO 1/YES)
 0.134284878291602          CENTER OF GRAVITY REFERENCE LOCATION (UPDATED)
0.0   PILOT POSITION (REF. TO CG ALONG XB)
0.0   PILOT POSITION (REF. TO CG ALONG YB)
0.0   PILOT POSITION (REF. TO CG ALONG ZB)
*
*      DEFLECTION LIMITS
*
20.       ELEVATOR (max)
-20.       ELEVATOR (min)
20.       AILERONS (symmetrical)
20.       RUDDER   (symmetrical)
0.0       FLAP (up)
30.       FLAP (down)
```

```
*
*
*     STEADY STATE COEFFICIENTS
```

ALPHA					C_{mb}
-5.0000	-0.002401	0.000000	0.253031	0.000000	0.020384
-4.8000	-0.003124	0.000000	0.241121	0.000000	0.019476
-4.6000	-0.003846	0.000000	0.229211	0.000000	0.018569
-4.4000	-0.004569	0.000000	0.217302	0.000000	0.017661
-4.2000	-0.005263	0.000000	0.205933	0.000000	0.016857
-4.0000	-0.005909	0.000000	0.195468	0.000000	0.016225
-3.8000	-0.006555	0.000000	0.185002	0.000000	0.015594
-3.6000	-0.007201	0.000000	0.174536	0.000000	0.014962
-3.4000	-0.007839	0.000000	0.164208	0.000000	0.014363
-3.2000	-0.008474	0.000000	0.153926	0.000000	0.013775
-3.0000	-0.009109	0.000000	0.143644	0.000000	0.013187
-2.8000	-0.009729	0.000000	0.133410	0.000000	0.012603
-2.6000	-0.010242	0.000000	0.123510	0.000000	0.012052
-2.4000	-0.010755	0.000000	0.113610	0.000000	0.011500
-2.2000	-0.011268	0.000000	0.103710	0.000000	0.010948
-2.0000	-0.011744	0.000000	0.093159	0.000000	0.010327
-1.8000	-0.012181	0.000000	0.081957	0.000000	0.009639
-1.6000	-0.012619	0.000000	0.070755	0.000000	0.008950
-1.4000	-0.013056	0.000000	0.059553	0.000000	0.008261
-1.2000	-0.013315	0.000000	0.048880	0.000000	0.007528
-1.0000	-0.013548	0.000000	0.038283	0.000000	0.006790
-0.8000	-0.013781	0.000000	0.027686	0.000000	0.006051
-0.6000	-0.013972	0.000000	0.016934	0.000000	0.005274
-0.4000	-0.014036	0.000000	0.005716	0.000000	0.004384
-0.2000	-0.014100	0.000000	-0.005501	0.000000	0.003494
0.0000	-0.014164	0.000000	-0.016718	0.000000	0.002604
0.2000	-0.014119	0.000000	-0.028202	0.000000	0.001604
0.4000	-0.014009	0.000000	-0.039845	0.000000	0.000540
0.6000	-0.013898	0.000000	-0.051487	0.000000	-0.000525
0.8000	-0.013788	0.000000	-0.063130	0.000000	-0.001590
1.0000	-0.013499	0.000000	-0.075053	0.000000	-0.002812
1.2000	-0.013209	0.000000	-0.086976	0.000000	-0.004035
1.4000	-0.012920	0.000000	-0.098899	0.000000	-0.005258
1.6000	-0.012559	0.000000	-0.110553	0.000000	-0.006495
1.8000	-0.012079	0.000000	-0.121760	0.000000	-0.007756
2.0000	-0.011599	0.000000	-0.132966	0.000000	-0.009018
2.2000	-0.011119	0.000000	-0.144173	0.000000	-0.010280
2.4000	-0.010464	0.000000	-0.157487	0.000000	-0.011674
2.6000	-0.009750	0.000000	-0.171505	0.000000	-0.013113
2.8000	-0.009037	0.000000	-0.185522	0.000000	-0.014552
3.0000	-0.008302	0.000000	-0.199237	0.000000	-0.015963
3.2000	-0.007419	0.000000	-0.210834	0.000000	-0.017178
3.4000	-0.006535	0.000000	-0.222430	0.000000	-0.018392
3.6000	-0.005652	0.000000	-0.234027	0.000000	-0.019607
3.8000	-0.004672	0.000000	-0.245604	0.000000	-0.020818

Quadrotor UAV

- Brushless Motors
- Arm length = 15 cm
- Mass = about 1.5 kg
- Endurance of about 20 minutes
- Able to perform autonomous flight



Quadrotor UAV: Assumptions (I)

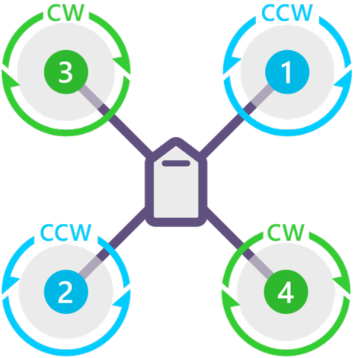
The following assumptions are considered for the evaluation of the equations of motion:

- Aircraft is a rigid body.
- Propellers are rigid.
- Flat Earth approximation.
- Quadrotor frame is symmetrical.
- Mass center and geometric center coincide.
- Motor inertia small and neglected.
- The ground effect is ignored.

Quadrotor UAV: Assumptions (II)

The following assumptions are made for the modeled 4Rotor:

- X Quadrotor frame
- Geometric and inertial symmetry.
- Propellers are rigid. and (ii) Barometer



Quadrotor UAV: Modeling (I)

The attitude and position of the quadrotor can be controlled to desired values by changing the speeds of the four motors.

The following forces and moments can be performed on the quadrotor:

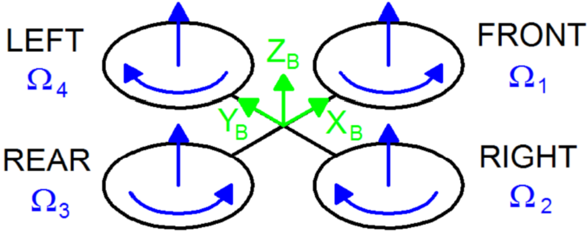
1. the thrust caused by rotors rotation,
2. the pitching moment and rolling moment caused by the difference of four rotors thrust,
3. the gravity, the gyroscopic effect, and the yawing moment.

1. The gyroscopic effect only appears in the lightweight construction quadrotor or when aggressive maneuvers should be performed.
2. The yawing moment is caused by the unbalanced of the four rotors rotational speeds.

The yawing moment can be cancelled out when two rotors rotate in the opposite direction.

Quadrotor UAV: Modeling (II)

So, the propellers are divided in two groups.
In each group there are two diametrically opposite motors that we can easily observe thanks to their direction of rotation.



The control for six degrees of freedom motions can be implemented by adjusting the rotational speeds of different motors.
The motions include forward and backward movements, lateral movement, vertical motion, roll motion, and pitch and yaw motions.

Quadrotor UAV: Hovering

See .gif...

Quadrotor UAV: Modeling (III)

The yaw motion of the quadrotor can be realised by a reactive torque produced by the rotor. The size of the reactive torque is relative to the rotor speed.

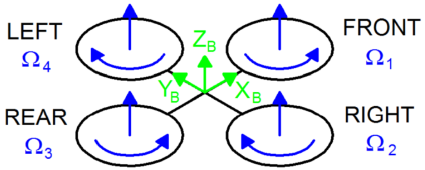
When the four rotor speeds are the same, the reactive torques will balance each other and quadrotor will not rotates, whereas if the four rotor speeds are not absolutely same, the reactive torques will not be balanced, and the quadrotor will start to rotate.

When the four rotor speeds synchronously increase and decrease is also required in the vertical movement.

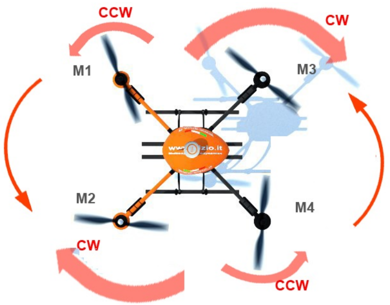
Quadrotor UAV: Basic Movements (I)

Depending on the speed rotation of each propeller it is possible to identify the four basic movements of the quadrotor.

THRUST!

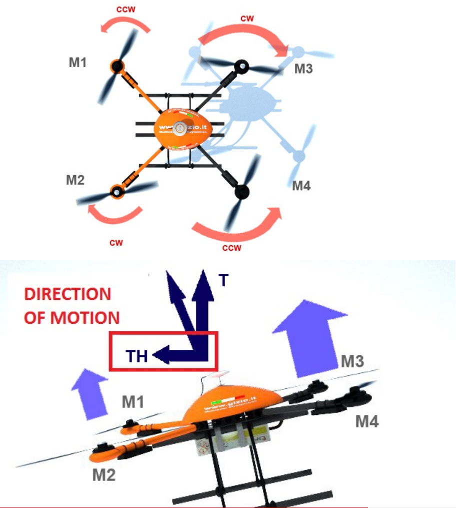


YAW!

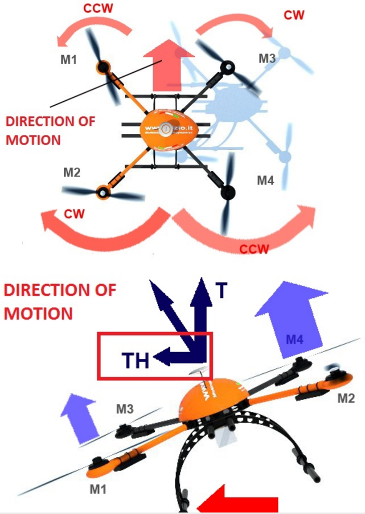


Quadrotor UAV: Basic Movements (II)

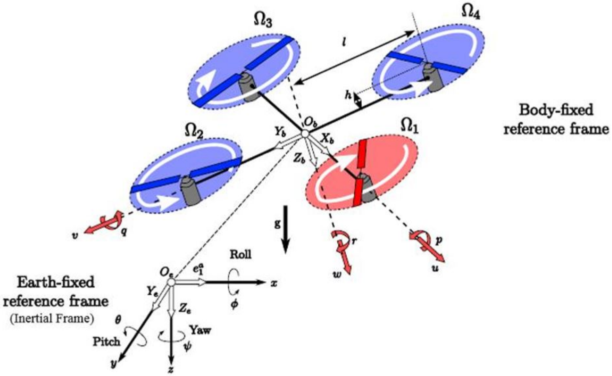
PITCH!

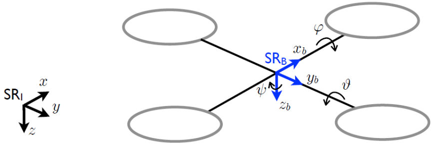


ROLL!



Quadrotor UAV: Basic Movements (III)





$$ {}^{\mathrm{I}}\mathrm{R}_{\mathrm{B}} = \left(\begin{array}{lll} c_\psi c_\vartheta & c_\psi s_\vartheta s_\varphi - s_\psi c_\varphi & c_\psi s_\vartheta c_\varphi + s \\ s_\psi c_\vartheta & s_\psi s_\vartheta s_\varphi + c_\psi c_\varphi & s_\psi s_\vartheta c_\varphi - s \\ -s_\vartheta & c_\vartheta s_\varphi & c_\vartheta c_\varphi \end{array} \right. $$

Quadrotor UAV: Model Setup (I)

Here a mathematical model of the quadrotor is provided, exploiting Newton and Euler equations for the 3D motion of a rigid body.

The goal of this section is to obtain a deeper understanding of the dynamics of the quadrotor and to provide a model that is sufficiently reliable for simulating and controlling its behavior.

As for the FW-UAV, three different models are analyzed:

1. a complete nonlinear model
2. a simplified nonlinear model (for control purposes)
3. a linear model

Quadrotor UAV: Model Setup (II)

The variables of the model are:

- three components of linear speed along Body axes $(u, v, w)^T$)
- three components of the angular velocities along Body axes $(p, q, r)^T$
- three angles that identify the attitude of the 4Rotor $(\phi, \vartheta, \psi)^T$
- three positions in NED axes

Because of four inputs and six (at least, the vector containing the linear and angular velocities in the body frame) outputs in a quadrotor, the quadrotor is considered an underactuated nonlinear complex system.

Quadrotor UAV: Nonlinear Model (I)

Let us call $(x, y, z, \phi, \vartheta, \psi)^T$ the vector containing the linear and angular position of the quadrotor in the NED frame and $(u, v, w, p, q, r)^T$ the vector containing the linear and angular velocities in the body frame.

From 3D body dynamics, it follows that the two reference frames are linked by the following relations:

$$\begin{aligned} v &= R_B^I v_B \\ \omega &= T \omega_B, \end{aligned}$$

where $v = (\dot{x}, \dot{y}, \dot{z})^T$ and $v_B = (u, v, w)^T$, R_B^I is the transformation matrix (see slide before), $\omega = (\dot{\phi}, \dot{\vartheta}, \dot{\psi})^T$ and $\omega_B = (p, q, r)^T$. The matrix T is defined starting from the kinematic equations (as we saw before!).

$$T = \begin{bmatrix} 1 & s(\phi)t(\vartheta) & c(\phi)t(\vartheta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & \frac{s(\phi)}{(\phi)} & \frac{c(\phi)}{(\phi)} \end{bmatrix}$$

Quadrotor UAV: Nonlinear Model (II)

Starting from the Newton-Euler formalism, we have

translational dynamics

$\sum F_I = m\dot{v}$,

with F_I external forces applied to the CoG ($\equiv$ CoM).

rotational dynamics

$\sum M_B = I_B \dot{\omega}_B + \omega_B \times (I_B \omega_B)$,

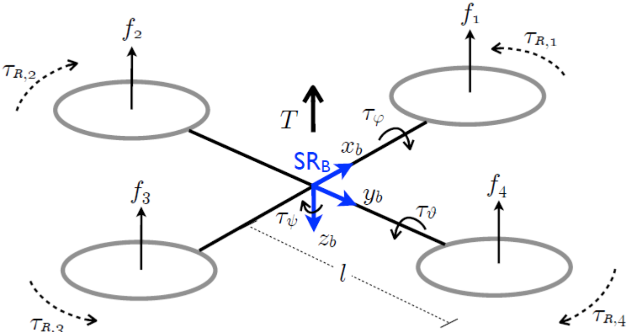
with M_B external moment around CoG and I_B tensor of inertia. $I_B =$

$\begin{bmatrix} I_Y & 0 & 0 \\ 0 & I_Y & 0 \\ 0 & 0 & I_Z \end{bmatrix}$ (DIAGONAL!)

Just as reminder, $\omega_B = (p, q, r)^T$

CONTROL INPUTS

$$
\begin{aligned}
T &= f_1 + f_2 + f_3 + f_4 \\
\tau_\phi &= l(f_2 - f_4) \\
\tau_\vartheta &= l(f_1 - f_3) \\
\tau_\psi &= -\tau_{R,1} + \tau_{R,2} - \tau_{R,3} + \tau_{R,4}
\end{aligned}
$$



APPLIED FORCES AND MOMENTS

$$\sum F_I = \underbrace{\begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix}}_{\text{weight}} + \underbrace{{}^{\mathbf{I}}\mathbf{R}_{\mathbf{B}}(\varphi, \vartheta, \psi) \begin{pmatrix} 0 \\ 0 \\ -T \end{pmatrix}}_{\text{actuation}} + \underbrace{F_A}_{\text{aerodynamics}} + \underbrace{F_D}_{\text{disturbances}}$$

weight

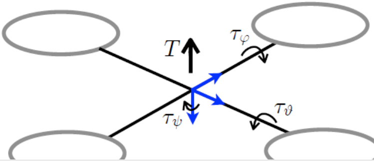
actuation

aerodynamics

disturbances

$$\sum M_B = \underbrace{\begin{pmatrix} L_\varphi \\ L_\vartheta \\ L_\psi \end{pmatrix}}_{\text{gyroscopic effects}} + \begin{pmatrix} \tau_\varphi \\ \tau_\vartheta \\ \tau_\psi \end{pmatrix} + \tau_A + \tau_D$$

gyroscopic effects



Quadrotor UAV: Nonlinear Model (IV)

EQUATIONS OF MOTION

$$
\begin{aligned}
\dot{x} &= v_x \\
\dot{y} &= v_y \\
\dot{z} &= v_z \\
\dot{v}_x &= F_{A,x} - (\cos(\psi)\sin(\vartheta)\cos(\varphi) + \sin(\psi)\sin(\varphi))\frac{T}{m} \\
\dot{v}_y &= F_{A,y} - (\sin(\psi)\sin(\vartheta)\cos(\varphi) - \sin(\varphi)\cos(\psi))\frac{T}{m} \\
\dot{v}_z &= F_{A,z} + g - \cos(\vartheta)\cos(\varphi)\frac{T}{m} \\
\dot{\varphi} &= p + \sin(\varphi)\tan(\vartheta)q + \cos(\varphi)\tan(\vartheta)r \\
\dot{\vartheta} &= \cos(\varphi)q - \sin(\varphi)r \\
\dot{\psi} &= \sin(\varphi)\sec(\vartheta)q + \cos(\varphi)\sec(\vartheta)r \\
\dot{p} &= \tau_{A,x} + \frac{I_r}{I_x}q\Omega_r + \frac{I_y - I_z}{I_x}qr + \frac{\tau_\varphi}{I_x} \\
\dot{q} &= \tau_{A,y} + \frac{I_r}{I_y}p\Omega_r + \frac{I_z - I_x}{I_y}pr + \frac{\tau_\vartheta}{I_y} \\
\dot{r} &= \tau_{A,z} + \frac{I_x - I_y}{I_z}pq + \frac{\tau_\psi}{I_z}
\end{aligned}
$$

Quadrotor UAV: Nonlinear Model (V)

EQUATIONS OF MOTION

$\dot{\xi} = f(\xi) + g(\xi)u,$

where $\xi = (x, y, x, v_x, v_y, v_z, \phi, \vartheta, \psi, p, q, r)^T$

$u = (T, \tau_\phi, \tau_\vartheta, \tau_\psi)^T$

Ω_r = average blade angular velocity

I_r = propeller inertia.

NOTE THAT THE GYROSCOPIC EFFECTS DUE TO PROPELLER (i.e., Ω_r, I_r) ARE CONSIDERED FOR AGGRESSIVE MANEUVERS OR LIGHT STRUCTURES!

Quadrotor UAV: Simplified Nonlinear Model (I)

negligible

- aerodynamic effects
- gyroscopic effects
- disturbances

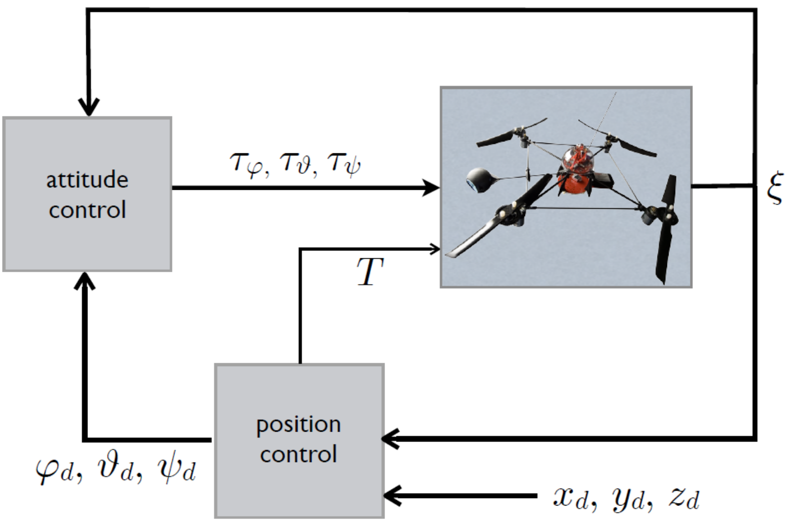
assuming

- small $(\phi, \vartheta) \Rightarrow$ $(\dot{\phi}, \dot{\vartheta}, \dot{\psi})^T = (p, q, r)^T$
- symmetric shape

The model is written in NED and Body frame.

$$\boxed{\begin{aligned}
\ddot{x} &= -(\cos(\psi)\sin(\vartheta)\cos(\varphi) + \sin(\psi)\sin(\varphi))\frac{T}{m} \\
\ddot{y} &= -(\sin(\psi)\sin(\vartheta)\cos(\varphi) - \sin(\varphi)\cos(\psi))\frac{T}{m} \\
\ddot{z} &= -\cos(\vartheta)\cos(\varphi)\frac{T}{m} + g \\
\ddot{\varphi} &= \frac{\tau_\varphi}{I_x} \\
\ddot{\vartheta} &= \frac{\tau_\vartheta}{I_y} \\
\ddot{\psi} &= \frac{\tau_\psi}{I_z}
\end{aligned}}$$

Quadrotor UAV: Simplified Nonlinear Model (II)



Quadrotor UAV: Simplified Nonlinear Model (III)

The variation of the vector of velocities should be re-written, as before in Body frame.

$$
\begin{aligned}
\dot{u} &= qw - rv + g\sin\vartheta, \\
\dot{v} &= -pw + ru - g\cos\vartheta\sin\phi, \\
\dot{w} &= pv - qu - g\cos\vartheta\cos\phi - \frac{T}{m},
\end{aligned}
$$

The only force is the THRUST, acting on the vertical component of the velocity.

Quadrotor UAV: Linear Model (I)

As for FW-UV, a standard continuous time-invariant state space representation is proposed

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \tag{6}$$

where $x(t)$ is the state vector, $u(t)$ the control signal, $y(t)$ the controlled output, A the state matrix, B the input matrix and C the output matrix.
Two different options can be found in literature:

1. definition of the aerodynamic derivatives
2. linearization of the simplified nonlinear model

The first option is usually used to have a deep knowledge of the platform dynamics to implement a robust controller for a stable flight.
The second option is proposed for simplified solution and simple maneuver (i.e. altitude hold, attitude hold,...).

The derivatives are calculated using the method explained in Prouty 's book [3], considering the platform characteristics.

Example: direct force damping X_u

$$\frac{\partial X}{\partial u} = -\rho A_b (\Omega R)^2 \frac{\partial C_H/\sigma}{\partial a_{1S}} \cdot \frac{\partial a_{1S}}{\partial \mu} \cdot \frac{\partial \mu}{\partial \dot{x}}$$

helicopter $$\frac{\partial a_{1S}}{\partial \mu} = \frac{8}{3}\vartheta_0 - 2\frac{v_1}{\Omega R} + 2\vartheta_{tw}$$

quadrotor $$\frac{\partial a_{1S}}{\partial \mu} = 4\vartheta_{0.75} - 2\lambda$$

For the second option, the linearization's procedure is developed around an equilibrium point $\hat{x}$, which for fixed input $\hat{u}$ is the solution of the algebraic system:

$$\hat{f}(\hat{x}, \hat{u}) = 0.$$

If the complete nonlinear system is considered, the solution is difficult to find in closed form because of trigonometric functions related each other in no-elementary way.
So, the linearization is performed starting from the simplified system.

Quadrotor UAV: Linear Model (IV)

The following state vector is considered
$x = (\phi, \vartheta, \psi, p, q, r, u, v, w, x, y, z)^T \in R^{12}$ and
$u = (T, \tau_\phi, \tau_\vartheta, \tau_\psi)^T \in R^4$

The linearized vectors are:
$\hat{x} = (0, 0, 0, 0, 0, 0, 0, 0, 0, \hat{x}, \hat{y}, \hat{z})^T \in R^{12}$
$\hat{u} = (mg, 0, 0, 0)^T \in R^4$

Note that this input vector represents the force necessary to delete the quadrotor weight and it consents its hovering.

Quadrotor UAV: Linear Model (V)

$$
\mathbf{A} = \left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}}\right|_{\substack{\mathbf{x}=\bar{\mathbf{x}}\\ \mathbf{u}=\bar{\mathbf{u}}}} = \begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & -g & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
g & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix}
\qquad
\mathbf{B} = \left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{u})}{\partial \mathbf{u}}\right|_{\substack{\mathbf{x}=\bar{\mathbf{x}}\\ \mathbf{u}=\bar{\mathbf{u}}}} = \begin{bmatrix}
0 & 0 & 0 & 0\\
0 & 0 & 0 & 0\\
0 & 0 & 0 & 0\\
0 & \frac{1}{I_x} & 0 & 0\\
0 & 0 & \frac{1}{I_y} & 0\\
0 & 0 & 0 & \frac{1}{I_z}\\
0 & 0 & 0 & 0\\
0 & 0 & 0 & 0\\
\frac{1}{m} & 0 & 0 & 0\\
0 & 0 & 0 & 0\\
0 & 0 & 0 & 0\\
0 & 0 & 0 & 0
\end{bmatrix}
$$

Quadrotor UAV: Linear Model (V)

Finally, we should include an actuator dynamics: the values of the input forces and torques are proportional to the squared speeds of the rotors.
So, we have

$$\begin{aligned} T = &\quad k_T(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ \tau_\phi = &\quad k_T(\Omega_3^2 - \Omega_1^2) \\ \tau_\vartheta = &\quad k_T(\Omega_4^2 - \Omega_2^2) \\ \tau_\psi = &\quad k_Q(\Omega_2^2 + \Omega_4^2 - \Omega_3^2 - \Omega_1^2) \end{aligned}$$

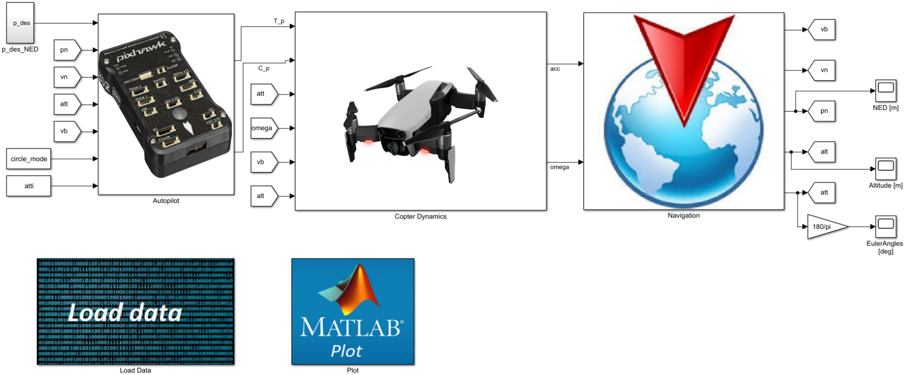
with k_T thrust factor and k_Q torque factor.

k_T and k_Q with identification methods!!

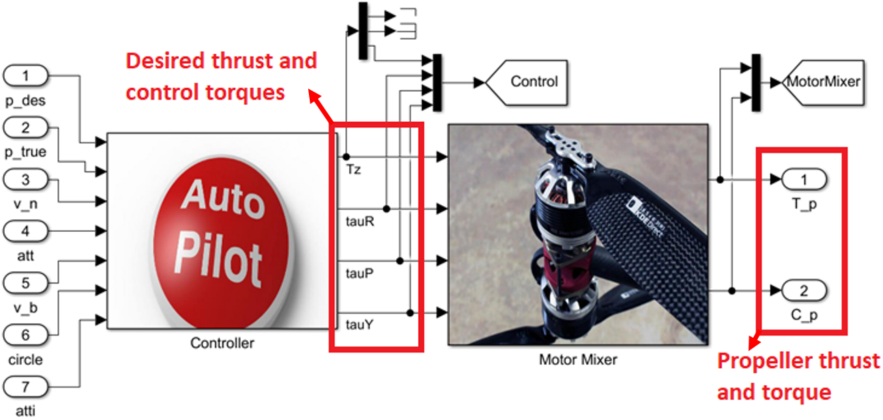
Matlab/Simulink Example: 4Rotor UAV (I)

The nonlinear model is designed as:

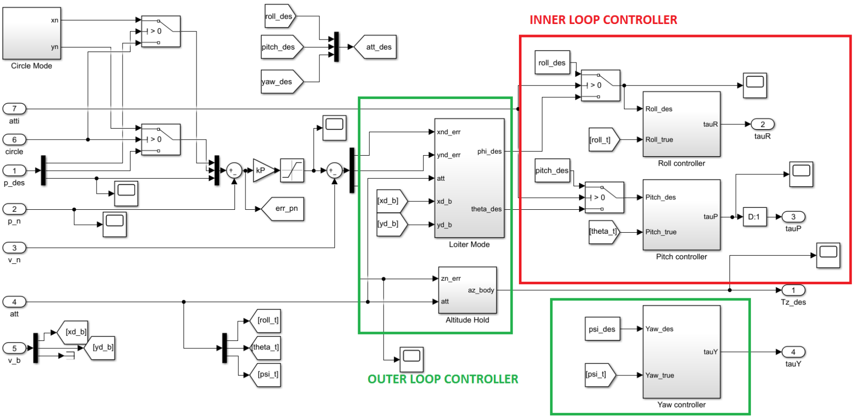
1. definition of the UAV parameters on a Matlab script
2. definition of the Equations of motion on a Matlab function,included in a Simulink file



Matlab/Simulink Example: 4Rotor UAV (II)



Matlab/Simulink Example: 4Rotor UAV (III)



How to design a multicopter? (I)

Why not a quadrotor?

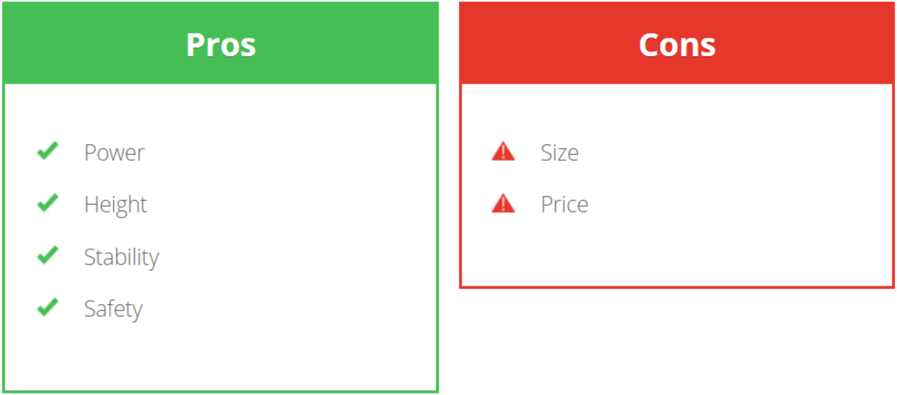
Pros	Cons
✔ Fast	▲ No Backup Motors
✔ Highly Maneuverable	▲ No Heavy Payloads
✔ Cheap	
✔ Easy to Build	

These copters utilize four propellers to ensure that the aircraft is able to lift up into the air. Essentially, there is a 4 propeller layout in the design of a square or rectangle around the body.

How to design a multicopter? (II)

How to design a multicopter? (III)

A hexacopter is the next step up from a quadcopter. These models have six motors and corresponding propellers. This adds to the capability of the aircraft and really makes this the optimal choice for anyone flying with expensive cameras attached.

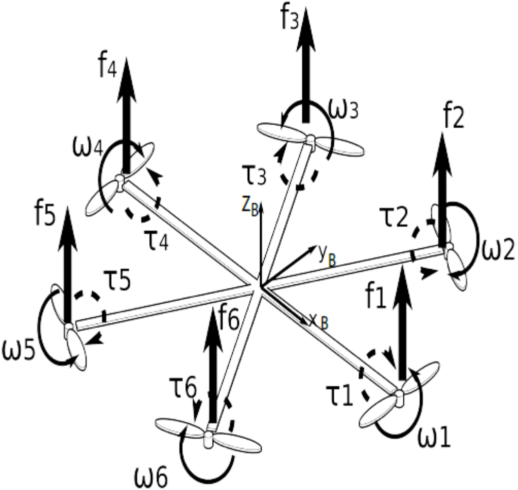


How to design a multicopter? (IV)

- The octocopter has all of the benefits seen with the hexacopters, but with even more power.
- These models are not cheap by any means and are often seen capturing the best aerial footage available.
- Price and battery life are a concern. Newer models will have higher flight times, which is ideal. However, first generation models would have their batteries drained within 10 minutes and would require immense charging times.

Example: Hexacopter Model (I)

The starting point are the same equations of motion of the QUADCOPTER.



Example: Hexacopter Model (II)

$$\mathbf{F} = \mathbf{d}(\mathbf{m}\,\mathbf{v_B})/\mathbf{dt} + \nu \times (\mathbf{m}\,\mathbf{v_B})$$

$$\Downarrow$$

$$\mathbf{F} = \mathbf{Q}^T \mathbf{F}_g + \mathbf{T}_B$$

$$\Downarrow$$

$$m\,\dot{\mathbf{v}}_B + \nu \times (\mathbf{m}\,\mathbf{v_B}) = \mathbf{Q^T F_g} + \mathbf{T_B}$$

$$\mathbf{T_B} = [\mathbf{0}\ \mathbf{0}\ \mathbf{T}]^{\mathbf{T}}$$

$$\Downarrow \quad \mathbf{f}_i = [0\ 0\ \omega_i^2]$$

$$T = \sum_{i=1}^{6} f_i = k \sum_{i=1}^{6} \omega_i^2$$

Example: Hexacopter Model (III)

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})\, q\, r / I_{xx} \\ (I_{zz} - I_{xx})\, p\, r / I_{yy} \\ (I_{xx} - I_{yy})\, p\, q / I_{zz} \end{bmatrix} - I_r \begin{bmatrix} q / I_{xx} \\ -p / I_{yy} \\ 0 \end{bmatrix} \omega_\Gamma + \begin{bmatrix} \tau_\phi / I_{xx} \\ \tau_\theta / I_{yy} \\ \tau_\psi / I_{zz} \end{bmatrix}$$

$$\omega_\Gamma = \omega_1 - \omega_2 + \omega_3 - \omega_4 + \omega_5 - \omega_6$$

$$\tau_{M_i} = b\, \omega_i^2 + I_{M_i}\, \dot{\omega}_i$$

$$\begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} \frac{3}{4} kl\, (\omega_2^2 + \omega_3^2 - \omega_5^2 - \omega_6^2) \\ kl \left(-\omega_1^2 - \frac{\omega_2^2}{4} + \frac{\omega_3^2}{4} + \omega_4^2 + \frac{\omega_5^2}{4} - \frac{\omega_6^2}{4} \right) \\ b\, (-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2 - \omega_5^2 + \omega_6^2) + \\ + I_M\, (\dot{\omega}_1 + \dot{\omega}_2 + \dot{\omega}_3 + \dot{\omega}_4 + \dot{\omega}_5 + \dot{\omega}_6) \end{bmatrix}$$

Summary of the Section (I)

So, what's the right model for you?
This is a rather difficult question!!

The following points should be kept in mind:

- Quad : These models are inexpensive and less durable. Perfect for beginners and non-professionals. EASY TO USE!!
- Hexa : The ideal choice for semi-professionals want a very durable, steady-flying copter that can carry heavier weights.
- Octo : **The top of the line.** These are copters that have a high price tag, but they are powerful, stable and fast. Advanced functionality and stabilization features make an octocopter a must-have for professionals.

Summary of the Section (II)

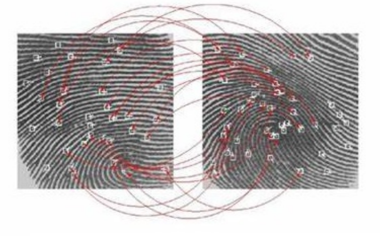
A QUAD-ROTOR IS CHOSEN—>BEGINNERS!!

- Various applications: rescue missions, inspection of structures, environmental monitoring
- Advantages of multicopters on mechanical simplicity and high maneuverability enable to conduct advanced missions in academic research and industrial applications with advanced control algorithms.
- It is important to understand and obtain a precise model of a multicopter for advanced controllers in order to execute more "sophisticate" missions.

Summary of the Section (III)

- The linearization of the nonlinear model around hover fight regime is conducted and used to construct controller to stabilize the quadrotor's attitude under small roll and pitch angles.
- In the case of hovering and forward flight with slow velocity, those assumptions are approximate reasonable.

Section 3: Identification Methods



Outline

Quadrotor: Nonlinear, multivariable, underactuated, and unstable system, even if they are used for different applications.
It is important to understand and obtain a precise model of a multicopter for advanced controllers $\Rightarrow$ **Main challenge:** parameter identification of a multicopter.

- Overview of literature methods [6, 7, 8]
- Identification of Thrust/Torque
- Identification of Moment of Inertia

Overview (I)

WHY AN IDENTIFICATION METHOD?

- As the quadrotor research shifts to new research areas (i.e. Mobile manipulation, Aerobatic moves, etc.), the need for an elaborate mathematical model arises, ans the simplification assumptions are no more suitable.
- When aggressive maneuvers such as fast forward and heave flight actions, VTOL and the ground effect appear, the dynamics of quadrotors could be influenced significantly under these aerodynamic force and moment.
- It is shown in [11] that existing techniques of modeling and control are inadequate for accurate trajectory tracking at higher speed and in uncertain environments if aerodynamic influence is ignored.

Overview (I)

In literature, different methodologies can be found to identify parameters of a multicopter.

Some of these methods are:

- Direct computation of geometry [7, 9]
- Compound pendulum method [12, 13] and Optical position tracking system (Vicon) [14]
- Analysis from flight data [8, 10, 7]

Overview (II)

The first two methods are **model-dependent approaches**, even if once performed on one system it can easily be repeated on other systems. For these classes of method, the most tested approach is the identification of physical parameters such as moments of inertia and the relation between propeller thrust/torque and propeller angular speed.

The last method is based on the variation of control inputs (called **direct approach**). This approach is a black box identification between the virtual control input and the angular rate. The drawback of this approach is that to perform the flight experiments to acquire identification data, the multirotor has to have a working controller. This means that the input signal will not be the one actually controlling the multirotor, but a reference signal to the control system or an overlay on the controller output.

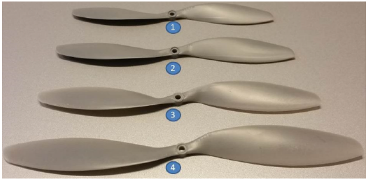
Direct Computation (I)

- Masses and spatial distributions of components are measured: arm length, total mass of the quadrotor, inertia matrix, friction coefficients, thrust coefficient and drag coefficient.
- CAD models can be used.

Disadvantages

- Time consuming (mathematical and physical calculations)
- Obtained result might be inaccurate

Direct Computation (II)



One problem of the geometric evaluation of the moments of inertia is that the body has to be assembled or disassembled in the laboratory.
If a commercial multirotor is considered, disassembling the overall system should be time consuming.

The moments of inertia can be identified using available CAD software, where all parts of the quadrotor have to be modeled: using model software like SOLIDWORKS to model all the parts of the quadrotor.
Another option is to decompose the quadrotor to spatial objects and calculate their moments of inertia using the following equations

$$I_0 = \int x^2 dm$$

The inertia matrix of the quadrotor can then be calculated applying the Huygens-Steiner theorem

$$I_{q_i} = I_0 + md^2,$$

for i =x,y,z, where d is the perpendicular distance between the axis of rotation and the axis that would pass through the centre of gravity of the quadrotor.

Compound Pendulum and Vicon (I)

The method proposed in [14] use two "tools" definition:

1. Compound pendulum
   - Evaluation of principal moments of inertia
   - Evaluation of engine thrust
2. Vicon system
   - Infrared marker-tracking system
   - Millimeter resolution of 3D spatial displacement
   - Position and attitude of the multirotor (i.e. hexacopter)

With this combination: avoiding complicated computations of geometry data and the risk of flight tests. This approach should be chosen for reducing the time of collecting data and to avoid the risk of accidents during flight tests, if a robust controller is not yet implemented on-board.

SEE "*SECONDLESSONCOMPOUND.PPTX*"

Analysis from flight data

- Identify and estimate parameters from flight data
- Estimate unknown parameters using flying data near to hovering point
- Implement Unscented Kalman Filter, prediction error method, etc ..
- Accurate and computationally simpler

Disadvantages

Possibility to occur damage on multicopter from flight without exact understanding physical characteristics.

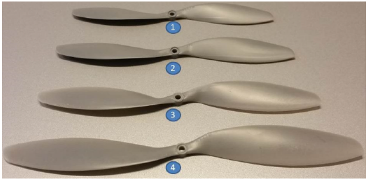
Identification of Elements(I)

Open the file *quadcopter.xlsx* configured for a specific rotary wing configuration (Q4T as an example).


Identification of Elements(I)

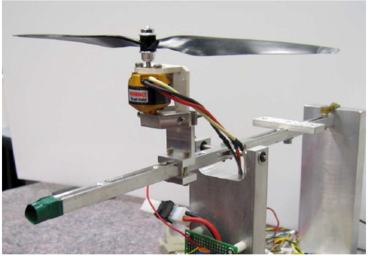
The Excel file compares the experimental data available for 15''x4.5'' and 15''x5'' propellers (T-Motor) with the total quadcopter weight estimated from mass breakdown of vehicle's components.
This approach provides a high fidelity estimation of endurance and propulsion system performances.
On the other hand, experimental data must be measured with a specific test rig (RC Benchmark Thrust Stand Series 1520 as an example).

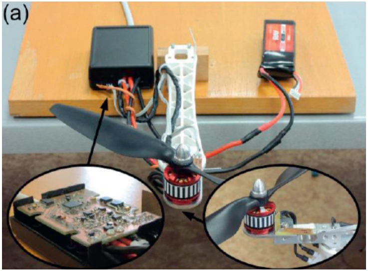
The UAV design is an optimization process between required thrust and available power. The definition of the propulsion system involves the choice of the number of rotors, as well as motor and propeller selection.
Before assembling the UAV, test engineers check the power consumption and generated thrust to be sure to meet the needs.
Thrust-to-speed as well as power-to-speed curves are fundamental for the dynamic behaviour simulation.



Identification of Thrust (II)

A test bench tool is needed for design and simulation purposes.
Different systems can be found in literature.

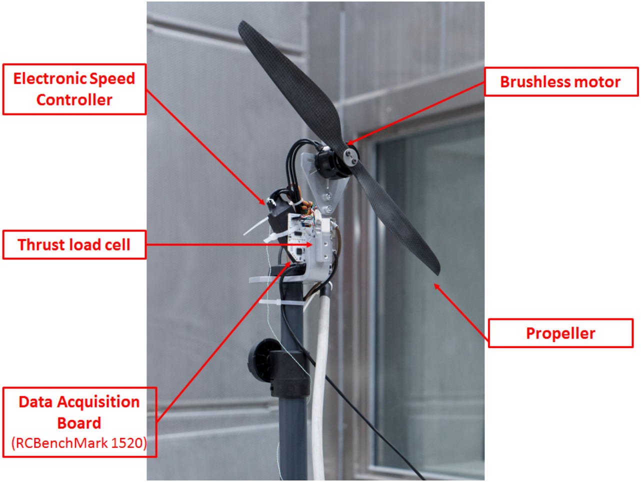




For example (picture on the left), a load cell measures the force and torque exerted on the mounting point. Battery monitoring circuitry measures motor voltage and current. Data is captured to the computer using an Atmel processor for A/D at 400 Hz.

Our example (I)

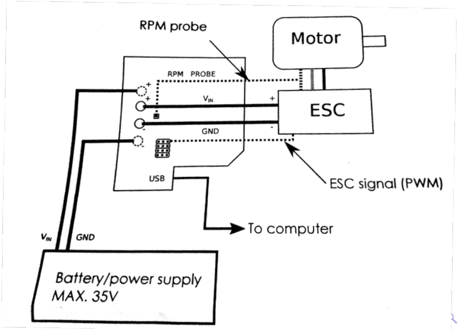
We perform identification tests with RC Benchmark Thrust Stand Series 1520.



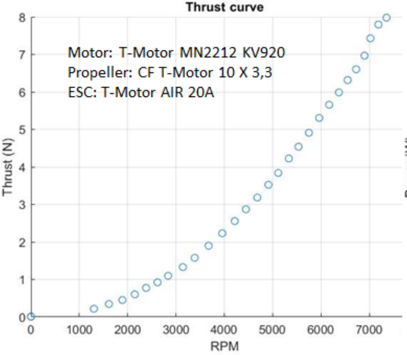
Our example (II)

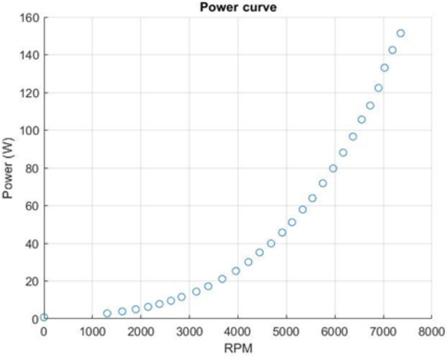
Sensor included:

- Thrust: one axis load cell
- Torque: not available
- Motor speed: electric probe
- Current: precision shunt resistor
- Mechanical power: not available
- Electric power: computed given voltage and current



Our example (III)





Item No.	Voltage (V)	Prop	Throttle	Current (A)	Power (W)	Thrust (G)	RPM
MN2212 KV920 V2.0	14.8	T-MOTOR 9545	50%	2.1	31.08	293	4260
			65%	4	59.2	476	5300
			75%	5.6	82.88	605	5960
			85%	7.4	109.52	742	6000
			100%	10.3	152.44	918	7350

Our example (IV)

The propeller thrust is proportional to the square of the motor speed. The least square method can be applied to estimate the thrust coefficient from experimental data

$$T = k_T \omega^2$$

The thrust coefficient is affected by:

- Airfoil type
- Propeller geometry (diameter, chord and pitch distribution along the radius)
- Environmental conditions (air density is influenced by temperature, pressure and humidity)

Our example (V)

The electrical energy behaves as a cubic polynomial due to propeller torque is proportional to the square of the motor speed

$$P_e = k_p \omega^3$$

The propeller torque coefficient can be estimated given the power coefficient

$$P_e = 2\pi\omega$$

Since $Q = k_Q \omega^2$, we have

$$k_Q = \frac{k_p}{2\pi\eta}$$

Remark 1

Results can be affected by ground effect if a vertical mount is used and the distance between the propeller plane and the floor is too small [16, 17, 18]. The test bench arm affects the measured thrust. The propeller mount and therefore the airflow direction must be considered when preparing the test bench [19].

A complete propeller database performance can be found at: https://m-selig.ae.illinois.edu/props/propDB.html [20].

The actual rotor thrust and power consumption is affected by neighborhood rotors in a nonlinear way due to complex aerodynamic interactions. A comparison between single rotor and complete UAV thrust production is given in [21].

Ground effect (I)

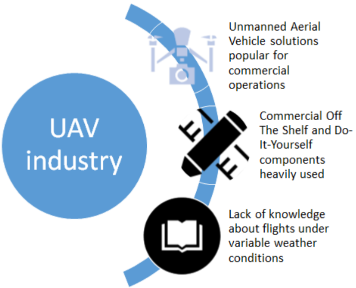
The experimental apparatus allows for incremental adjustment of three fundamental parameters:

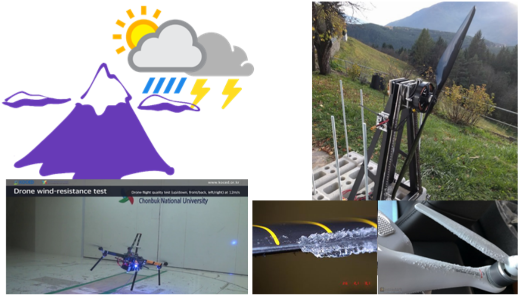
1. the distance between the propeller plane and the ground (Z),
2. the spacing between the propellers on a typical X-shaped quadrotor frame, and
3. the speed of the motors.

The experimental setup is focused on eliminating uncontrolled and undesirable influences as much as possible to isolate only the desired test parameters.

Ground effect (II)

Thrust vs Temperature (I)





Thrust vs Temperature (II)

Temperature

- From -40°C to +60°C, 20°C step

Pressure

- From sea level to 4000m , 500m step
- From 4000m to 9000m, 1000m step

Combined effect of temperature and pressure based on ISA model

Combined effect of temperature and humidity

Thrust vs Temperature (III)

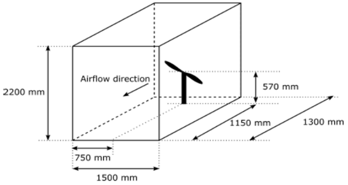
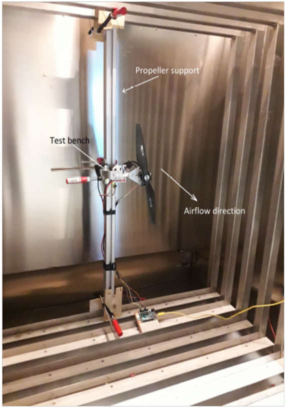
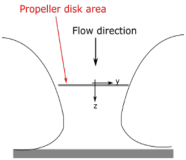
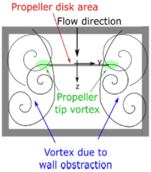


Fig. 7: Climatic chamber dimensions

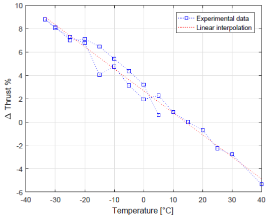


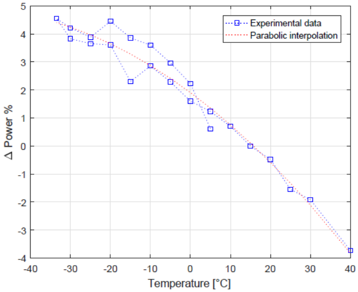


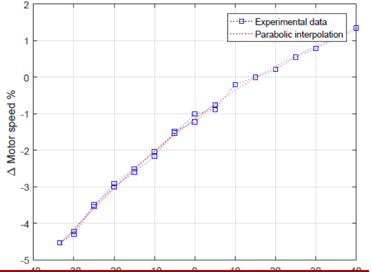
Airflow inside the climatic chamber



Thrust vs Temperature (IV)







Linear interpolation: for thrust data. Second order polynomial regression model: to fit speed and power measurements.

As an example, at 0 deg, thrust has experienced a mean increase of 2.58% while the electric power increase is only 1.91%. This is related to the motor speed that shows a reduction of 1.12%.

Remark 2

Hover condition is the main status of the quadrotor, as quite a few tasks, such as surveillance, search and rescue, are implemented in the condition.
Therefore, linear model is simplified on the complex nonlinear one derived from first principle model, in which the feasibility is proved by application result.
However, aggressive maneuver shows the obvious nonlinear characteristics, so that the nonlinear model is needed, in which acquisition of data with advanced methods (i.e. neural networks) is a optional scheme despite of a large amount of tests is indispensable. The data-based approaches have shown the distinctive advantages in other application areas, the use on a quadrotor will be a commendable attempt.

QUESTIONS?

Aircraft Simulation Models: Theoretical Background

References

Allerton, D., "Principles of Flight Simulation", AIAA, 2009.

Gainer, T.G., Hoffman, S., "Summary of transformation equations and equations of motion used in free-flight and wind tunnel data reduction and analysis", NASA SP-3070, NASA, 1972.

Guglieri, G., Riboldi, C.E., "Introduction to Flight Dynamics", CELID, 2014.

References I

- B. Etkin and L. Reid, “Dynamics of Flight: Stability and Control," New York: John Wiley and Sons, 1996.
- B. L. Stevens, F. L. Lewis, “Aircraft Control and Simulation, New York: John Wiley and Sons, 2004.
- Prouty, R.W., “Helicopter Performance, Stability and Control," PWS Engineering, 1986.
- Q. Quan, “Introduction to Multicopter Design and Control," Springer Singapore, 2017.
- R. Mahony, V. Kumar, P. Corke, “Multirotor Aerial Vehicles, Modeling, Estimation, and Control of Quadrotor," *IEEE Robotics Automation Magazine*, September 2012.

X. Zhang, X. Li, K. Wang, and Y. Lu, "A Survey of Modelling and Identification of Quadrotor Robot," *Abstract and Applied Analysis*, vol. 2014, 2014.

A. Chovancová, T. Fico, L. Chovanec, and P. Hubinsk, "Mathematical Modelling and Parameter Identification of Quadrotor (a survey)," *Procedia Engineering*, vol. 96, pp. 172-181, 2014.

N. Abas, A. Legowo, and R. Akmeliawati, "Parameter identification of an autonomous quadrotor," *4th International Conference On Mechatronics (ICOM)*, pp. 1-8, 2011.

M. Elsamanty, A. Khalifa, M. Fanni, A. Ramadan, and A. Abo-Ismail, "Methodology for identifying quadrotor parameters, attitude estimation and control," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp.1343-1348, 2013.

I. Stanculeanu and T. Borangiu, "Quadrotor black-box system identification," *World Academy of Science, Engineering and Technology*, vol. 78, 2011.

H. M. Huang, G.M. Hoffmann, S.L. Waslander, and C.J. Tomlin, "Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering,â *Proceedings to IEEE International Conference on Robotics and Automation*, Kobe, Japan, pp. 3277â3282, 2009.

W. Gracey, "The experimental determination of the moments of inertia of airplanes by a simplified compound-pendulum method," *NACA-TN-1629*, NASA, 1948.

S. Patankar, D. Schinstock, and R. Caplinger, "Application of pendulum method to UAV momental ellipsoid estimation," *6th AIAA Aviation Technology, Integration and Operations Conference (ATIO)*, 2006.

References IV

E. Capello, H. Park, B. Tavora, G. Gugleri, and M. Romano, "Modeling and Experimental Parameter Identification of a Multicopter via a Compound Pendulum Test Rig," *2015 International Workshop on Research, Education, and Development on Unmanned Aerial Systems (RED-UAS 2015)*,

D. Sartori, and W Yu, "Experimental Characterization of a Propulsion System for Multi-rotor UAVs," *J Intell Robot Syst (2019)*

Conyers, S. A., Rutherford, M. J., and Valavanis, K. P., "An empirical evaluation of ground effect for small-scale rotorcraft," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1244-1250, May 2018.

References V

- Conyers, S. A., Rutherford, M. J., and Valavanis, K. P., "An empirical evaluation of ceiling effect for small-scale rotorcraft., *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 243-249, June 2018.
- Kocer, B. B., Kumtepeli, V., Tjahjowidodo, T., Pratama, M., Tripathi, A., Lee, G. S. G., and Wang, Y., "UAV Control in Close Proximities-Ceiling Effect on Battery Lifetime, " arXiv preprint arXiv:1812.11707, 2018.
- Theys, B., Dimitriadis, G., Hendrick, P., and De Schutter, J., "Influence of propeller configuration on propulsion system efficiency of multi-rotor unmanned aerial vehicles, *2016 international conference on unmanned aircraft systems (ICUAS)*, pp. 195-201, June 2016.

References VI

Brandt, J. B., and Selig, M. S., “Small-scale propeller performance at low speedsâOnline Database, " *URL* : *www.ae.illinois.edu/m − selig/props/propDB.html*, [cited June 2012].

Russell, C. R., Jung, J., Willink, G., and Glasner, B., “Wind tunnel and hover performance test results for multicopter UAS vehicles", 2016.