Spacecraft Dynamics

Lesson 4: Attitude Kinematics

Rafael Vázquez Valenzuela

Aerospace Engineering Department
Escuela Superior de Ingenieros, Universidad de Sevilla rvazquez1@us.es

June 30, 2022

Attitude Differential Kinematic Equations

- Remember that, when talking about displacements, the differential kinematic equations (for short:kinematics) relate the position and velocity vectors whereas the differential dynamic equations (dynamics) relate the velocity and force vectors.
- For attitude, the kinematics relate the chosen representation of attitude (DCM, Euler angles, quaternions,...) with the angular velocity $\vec{\omega}$ (normally, expressed in body axes). Typically these equations are non-linear.
- In attitude estimation (which is a part of inertial navigation), gyros measure $\vec{\omega}$ and one uses kinematics (integrating the equation) to compute attitude (Lesson 6).
- Thus, it is important to know the kinematics for the different representations, to see the possible computational advantages (hint: quaternions win).

DCM kinematics I

- Suppose we want to compute the attitude of a frame B w.r.t. to A, using the DCM $C_A^B(t)$, knowing B is *rotating* w.r.t. A at an angular velocity $\vec{\omega}_{B/A}^B$.
- By definition $\frac{d}{dt} C_A^B = \frac{C_A^B(t+dt) - C_A^B(t)}{dt}$ (if someone prefers limits the reasoning is analogous)
- Fixing A, we can imagine that B is moving, so in fact $B = B(t)$ and, formally, we can write $C_A^B(t) = C_A^{B(t)}$.
- Using this reasoning, $C_A^B(t+dt) = C_A^{B(t+dt)} = C_{B(t)}^{B(t+dt)} C_A^{B(t)}$. Then:

$$A \longrightarrow B(t) \longrightarrow B(t+dt)$$

- During a time dt, the reference frame B has rotated w.r.t to itself just a small angle; remembering Lesson 3: $C_{B(t)}^{B(t+dt)} = \mathrm{Id} - \left(d\vec{\theta}^B\right)^{\times}$, where $d\vec{\theta}^B$ is a small angles vector.

DCM kinematics II

- Then: $\frac{d}{dt}C_A^B = \frac{C_A^B(t+dt)-C_A^B(t)}{dt} = \frac{C_{B(t)}^{B(t+dt)}C_A^B(t)-C_A^B(t)}{dt} = \frac{(\mathrm{Id}-\left(d\vec{\theta}^B\right)^\times)C_A^B(t)-C_A^B(t)}{dt} = -\frac{\left(d\vec{\theta}^B\right)^\times}{dt}C_A^B(t)$
- The matrix $\frac{\left(d\vec{\theta}^B\right)^\times}{dt}$ is written

$$\frac{\left(d\vec{\theta}^B\right)^\times}{dt} = \begin{bmatrix} 0 & -\frac{d\theta_3}{dt} & \frac{d\theta_2}{dt} \\ \frac{d\theta_3}{dt} & 0 & -\frac{d\theta_1}{dt} \\ -\frac{d\theta_2}{dt} & \frac{d\theta_1}{dt} & 0 \end{bmatrix} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix},$$

where $\vec{\omega}_{B/A}^B = [\omega_1\ \omega_2\ \omega_3]^T$ since $d\vec{\theta}^B$ is the angle the body rotates in a dt seen from its own frame, w.r.t. reference system A: by definition this is the angular velocity. Then

$$\left(\vec{\omega}_{B/A}^B\right)^\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix},$$

- Thus: $\frac{d}{dt}C_A^B = \dot{C}_A^B = -\left(\vec{\omega}_{B/A}^B\right)^\times C_A^B$.

DCM kinematics III

- A variation: transposing both sides of $\dot{C}_A^B = -\left(\vec{\omega}_{B/A}^B\right)^\times C_A^B$ we reach $\dot{C}_B^A. = C_B^A \left(\vec{\omega}_{B/A}^B\right)^\times$
- DCM kinematics: matrix differential equation, solved component-wise (system of 9 coupled scalar ODEs).
- Main difficulty in numerical resolution: conservation of orthogonality. Notice that, since $I = (C_A^B)(C_A^B)^T$, taking derivative:

$$\left[\frac{d}{dt}(C_A^B)\right](C_A^B)^T + C_A^B \frac{d}{dt}(C_A^B)^T$$

$$= -\left(\vec{\omega}_{B/A}^B\right)^\times C_A^B (C_A^B)^T + C_A^B C_B^A \left(\vec{\omega}_{B/A}^B\right)^\times$$

$$= -\left(\vec{\omega}_{B/A}^B\right)^\times + \left(\vec{\omega}_{B/A}^B\right)^\times = 0$$

- Thus kinematics preserve orthogonality. But numerical schemes will not.

DCM kinematics IV

- There exists algorithms to find, given a certain matrix, another orthogonal matrix "closest" to the starting one in some sense.
- For instance, given M, one can compute

$$Q = M(M^T M)^{-1/2}$$

which is orthogonal (and equal to M if it was orthogonal to start with).

- Problem: computing the square root of a matrix is not simple. An iterative method that avoids the computation is the following.
- Start: $Q_0 = M$; iterate $Q_{k+1} = 2M(Q_k^{-1}M + M^T Q_k)^{-1}$, and it's easy to see that this converges to Q when $k \to \infty$, with the condition that M is close to some orthogonal matrix (and therefore invertible).
- If M is very close to being orthogonal to start with, convergence is quite fast!

Euler angles kinematics I

- Example: aircraft set of Euler angles (yaw,pitch,roll). Start from the definition:

$$n \xrightarrow[z^n]{\psi} S \xrightarrow[y^S]{\theta} S' \xrightarrow[x^{S'}]{\varphi} b$$

- Angular velocity can be decomposed between frames as $\vec{\omega}_{b/n} = \vec{\omega}_{b/S'} + \vec{\omega}_{S'/S} + \vec{\omega}_{S/n}$.
- Writing the equation in b: $\vec{\omega}^b_{b/n} = \vec{\omega}^b_{b/S'} + \vec{\omega}^b_{S'/S} + \vec{\omega}^b_{S/n}$
- On the other hand: $\vec{\omega}^b_{b/S'} = [\dot{\varphi}\ 0\ 0]^T$, $\vec{\omega}^{S'}_{S'/S} = [0\ \dot{\theta}\ 0]^T$, $\vec{\omega}^S_{S/n} = [0\ 0\ \dot{\psi}]^T$.
- Then: $\vec{\omega}^b_{b/n} = \vec{\omega}^b_{b/S'} + C^b_{S'}\vec{\omega}^{S'}_{S'/S} + C^b_S\vec{\omega}^S_{S/n}$ and since $C^b_S = C^b_{S'}C^{S'}_S$, we reach: $\vec{\omega}^b_{b/n} = \vec{\omega}^b_{b/S'} + C^b_{S'}\vec{\omega}^{S'}_{S'/S} + C^b_{S'}C^{S'}_S\vec{\omega}^S_{S/n}$

Euler angles kinematics II

- Developing:

$$
\begin{aligned}
\vec{\omega}_{b/n}^{b} &= \begin{bmatrix} \dot{\varphi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}\varphi & \mathrm{s}\varphi \\ 0 & -\mathrm{s}\varphi & \mathrm{c}\varphi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} \\
&+ \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}\varphi & \mathrm{s}\varphi \\ 0 & -\mathrm{s}\varphi & \mathrm{c}\varphi \end{bmatrix} \begin{bmatrix} \mathrm{c}\theta & 0 & -\mathrm{s}\theta \\ 0 & 1 & 0 \\ \mathrm{s}\theta & 0 & \mathrm{c}\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\
&= \begin{bmatrix} \dot{\varphi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \mathrm{c}\varphi\dot{\theta} \\ -\mathrm{s}\varphi\dot{\theta} \end{bmatrix} + \begin{bmatrix} -\mathrm{s}\theta\dot{\psi} \\ \mathrm{s}\varphi\mathrm{c}\theta\dot{\psi} \\ \mathrm{c}\varphi\mathrm{c}\theta\dot{\psi} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & -\mathrm{s}\theta \\ 0 & \mathrm{c}\varphi & \mathrm{s}\varphi\mathrm{c}\theta \\ 0 & -\mathrm{s}\varphi & \mathrm{c}\varphi\mathrm{c}\theta \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}
\end{aligned}
$$

Euler angles kinematics III

- What we actually need is an expression for the time derivatives of angles as a function of $\vec{\omega}^b_{b/n} = [\omega_1\ \omega_2\ \omega_3]^T$, therefore, inverting the matrix we reach

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\mathrm{s}\theta \\ 0 & \mathrm{c}\varphi & \mathrm{s}\varphi\mathrm{c}\theta \\ 0 & -\mathrm{s}\varphi & \mathrm{c}\varphi\mathrm{c}\theta \end{bmatrix}^{-1} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{\mathrm{c}\theta} \begin{bmatrix} \mathrm{c}\theta & \mathrm{s}\theta\mathrm{s}\varphi & \mathrm{s}\theta\mathrm{c}\varphi \\ 0 & \mathrm{c}\varphi\mathrm{c}\theta & -\mathrm{s}\varphi\mathrm{c}\theta \\ 0 & \mathrm{s}\varphi & \mathrm{c}\varphi \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$

- Notice these are 3 non-linear ODEs, with several trig functions.
- There is a singularity at $\theta = \pm 90^{\mathrm{o}}$. In fact Euler angles are not well defined for this attitude. This singularity is the reason why Euler angles are frequently avoided in inertial navigation (for aircraft or spacecraft).
- All other sets of Euler angles also exhibit singularities; there is no combination of angles free of them.

Euler's axis and angle kinematics

- Representation as Euler's axis and angle, namely $(\vec{e}^{b}_{b/n}, \theta)$, has the following kinematics:
- For Euler's angle: $\dot{\theta} = (\vec{e}^{b}_{b/n})^T \vec{\omega}^{b}_{b/n}$
- For Euler's axis:

$$\dot{\vec{e}}^{b}_{b/n} = \frac{1}{2}\left[\left(\vec{e}^{b}_{b/n}\right)^{\times} + \frac{1}{\tan\theta/2}\left(\mathrm{Id} - \vec{e}^{b}_{b/n}(\vec{e}^{b}_{b/n})^T\right)\right]\vec{\omega}^{b}_{b/n}$$

- These are 4 ODEs, non-linear.
- They exhibit a singularity at $\theta = 0$.
- If $\vec{\omega}$ has a constant direction equal to the initial axis $\vec{e}$, then kinematics simplify to $\dot{\vec{e}} = \vec{0}$ (this is, $\vec{e}(t) = \vec{e}(0)$) and $\dot{\theta} = \|\vec{\omega}\|$ (important case!).
- In practice these are seldom used; we just apply them as an intermediate step towards quaternion kinematics.

Quaternion kinematics I

- Remember the attitude quaternion defined from Euler's angle and axis:
  $q_0 = \cos\theta/2,\ \vec{q} = \sin\theta/2\vec{e}^{\,b}_{b/n}$.
- Taking derivative in the q_0 definition and substituting the kinematics for θ, one gets
  $\dot{q}_0 = -\frac{1}{2}\sin\theta/2\dot{\theta} = -\frac{1}{2}\sin\theta/2(\vec{e}^{\,b}_{b/n})^T\vec{\omega}^{\,b}_{b/n} = -\frac{1}{2}\vec{q}^{\,T}\vec{\omega}^{\,b}_{b/n}$
- Taking derivative now in the $\vec{q}$ definition:

$$\dot{\vec{q}} = \frac{1}{2}\cos\theta/2\vec{e}^{\,b}_{b/n}\dot{\theta} + \sin\theta/2\dot{\vec{e}}^{\,b}_{b/n}$$

- Substituting Euler's axis and angle kinematics:

$$\begin{aligned}\dot{\vec{q}} &= \frac{1}{2}\cos\theta/2\vec{e}^{\,b}_{b/n}(\vec{e}^{\,b}_{b/n})^T\vec{\omega}^{\,b}_{b/n} \\ &\quad + \frac{1}{2}\sin\theta/2\left[\left(\vec{e}^{\,b}_{b/n}\right)^{\times} + \frac{1}{\tan\theta/2}\left(\mathrm{Id} - \vec{e}^{\,b}_{b/n}(\vec{e}^{\,b}_{b/n})^T\right)\right]\vec{\omega}^{\,b}_{b/n} \\ &= \frac{1}{2}\left[q^{\times} + q_0\mathrm{Id}\right]\vec{\omega}^{\,b}_{b/n}\end{aligned}$$

Quaternion kinematics II

- Quaternion kinematics in matrix form:

$$\frac{d}{dt}\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix}\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

  where $\vec{\omega}^b_{b/n} = [\omega_x\ \omega_y\ \omega_z]^T$.
- These are 4 bilinear ODEs, without singularities.
- Notice the absence of trig functions, which helps precision.
- These properties of quaternion kinematics are perhaps the most important reasons why its use is wide among the aerospace community to represent spacecraft (and aircraft!) attitude. All computations can be done (internally) with quaternions, and if necessary one can transform them to other representations for visualization or other purposes, depending on the application.

Quaternion kinematics III

- Remembering the definition of quaternion product as a matrix, one can notice some similarities with the differential kinematic equation. In fact, defining a "quaternion" q_ω with zero scalar part and whose vector part is equal to the components of the angular velocity, namely:

$$q_\omega = \left[\begin{array}{cccc} 0 & \omega_x & \omega_y & \omega_z \end{array} \right]^T$$

  kinematics can be expressed very simply as

$$\dot{q} = \frac{1}{2} q \star q_\omega$$

- The only drawback of using quaternion kinematics is that numerical errors can creep in and make the quaternion modulus different from 1. However, unlike the DCM, making the quaternions verify its constraint is easy; just normalizing the quaternion (dividing by its modulus) we can make its modulus stay at one.

Other kinematics

- RP:

$$\dot{\vec{g}} = \frac{1}{2}\left[\mathrm{Id} + \vec{g}^{\times} + \vec{g}\vec{g}^{T}\right]\vec{\omega}$$

- MRP:

$$\dot{\vec{p}} = \frac{1 + \|\vec{p}\|^2}{4}\left[\mathrm{Id} + 2\frac{\vec{p}^{\times} + \vec{p}^{\times}\vec{p}^{\times}}{1 + \|\vec{p}\|^2}\right]\vec{\omega}$$

- Rotation vector:

$$\dot{\vec{\theta}} = \vec{\omega} + \frac{1}{2}\vec{\theta} \times \vec{\omega} + \frac{1}{\theta}\left(1 - \frac{\theta}{2\tan\theta/2}\right)\vec{\theta} \times (\vec{\theta} \times \vec{\omega})$$

Slew maneuvers

- Given two different attitudes expressed as quaternions, q_0 and q_1 and some time interval T, can we *construct* a continuous angular velocity $\vec{\omega}(t)$ such that $q(t=0)=q_0$ and $q(t=T)=q_1$?
- The key to do it is, as in interpolation, to find the so-called rotation quaternion q_2 representing the attitude between q_0 and q_1: $q_2 = \frac{1}{q_0} \star q_1 = q_0^* q_1$. From this quaternion extract Euler's angle θ_1 and axis $\vec{e}$ which verify $q_2 = \begin{bmatrix} \cos\theta_1/2 \\ \vec{e}\sin\theta_1/2 \end{bmatrix}$, this is, $\theta_1 = 2\arccos(q_{20})$ and $\vec{e} = \frac{\vec{q}_2}{\sin\theta_1/2}$
- The solution angular speed $\vec{\omega}(t)$ goes in the direction of $\vec{e}$ and represents the shortest rotation. Call its modulus $\omega(t)$. Then $\theta(t) = \int_0^t \omega(\tau)d\tau$ and the attitude evolves as

$$q(t) = q_0 \star \begin{bmatrix} \cos(\theta(t)/2) \\ \sin(\theta(t)/2)\vec{e} \end{bmatrix}$$

- Any $\omega(t)$ such that $\int_0^T \omega(\tau)d\tau = \theta_1$ is a solution.

Linearizing quaternion kinematics I

- Linearizing is crucial in many aerospace guidance and control applications. Asume we have a reference angular speed $\vec{\omega}_r$ that generates a reference quaternion $\bar{q}$ according to kinematics. If $\vec{\omega} = \vec{\omega}_r + \delta\vec{\omega}$, where $\delta\vec{\omega}$ is "small," what is the new resulting quaternion due to this small change?
- Use the error quaternion as $q = \bar{q} \star \delta q$, and let us determine δq. Taking derivative:

$$\dot{q} = \dot{\bar{q}} \star \delta q + \bar{q} \star \dot{\delta q} = \frac{1}{2} q \star q_\omega$$

- Using $\dot{\bar{q}} = \frac{1}{2}\bar{q} \star q_{\omega_r}$:

$$\frac{1}{2}\bar{q} \star q_{\omega_r} \star \delta q + \bar{q} \star \dot{\delta q} = \frac{1}{2}\bar{q} \star \delta q \star q_\omega$$

- Left-multiplying by $\bar{q}^*$ and solving for $\dot{\delta q}$, one gets:

$$\dot{\delta q} = \frac{1}{2}\delta q \star q_\omega - \frac{1}{2} q_{\omega_r} \star \delta q$$

Linearizing quaternion kinematics II

- Express now $\vec{\omega} = \vec{\omega}_r + \delta\vec{\omega}$ and remember the linearization of δq as a function of the parameter $\vec{a}$:

$$\frac{d}{dt}\left[\begin{array}{c} 1 \\ \vec{a}/2 \end{array}\right] \approx \frac{1}{2}\left[\begin{array}{c} 1 \\ \vec{a}/2 \end{array}\right] \star \left[\begin{array}{c} 0 \\ \vec{\omega}_r + \delta\vec{\omega} \end{array}\right] - \frac{1}{2}\left[\begin{array}{c} 0 \\ \vec{\omega}_r \end{array}\right] \star \left[\begin{array}{c} 1 \\ \vec{a}/2 \end{array}\right]$$

- Remebering: $\left[\begin{array}{c} q_0' \\ \vec{q}' \end{array}\right] \star \left[\begin{array}{c} q_0 \\ \vec{q} \end{array}\right] = \left[\begin{array}{c} q_0' q_0 - \vec{q}'^T \vec{q} \\ q_0 \vec{q}' + q_0' \vec{q} + \vec{q}' \times \vec{q} \end{array}\right]$, one has:

$$\frac{d}{dt}\left[\begin{array}{c} 1 \\ \vec{a}/2 \end{array}\right] \approx \frac{1}{2}\left[\begin{array}{c} -\vec{a}^T/2(\vec{\omega}_r + \delta\vec{\omega}) + \vec{\omega}_r^T \vec{a}/2 \\ \vec{\omega}_r + \delta\vec{\omega} + \vec{a}/2 \times (\vec{\omega}_r + \delta\vec{\omega}) - \vec{\omega}_r - \vec{\omega}_r \times \vec{a}/2 \end{array}\right]$$

- Since we are linearizing $\|\vec{a}\|\|\delta\vec{\omega}\| \approx 0$ because it is a double product of small terms. Operating:

$$\frac{d}{dt}\left[\begin{array}{c} 1 \\ \vec{a}/2 \end{array}\right] \approx \frac{1}{2}\left[\begin{array}{c} 0 \\ \delta\vec{\omega} + \vec{a} \times \vec{\omega}_r \end{array}\right]$$

- This is: $\dot{\vec{a}} \approx \delta\vec{\omega} + \vec{a} \times \vec{\omega}_r$. A quite simple expression. Thus the reference angular velocity also influences $\vec{a}$.