Canonical Normalization

Solution Magnitudes

Solution for **orbits** generally involves **distances** and time periods that are quite **large**, even in the context of **Earth.**

In case we have an **inter-planetary** motion, distances and velocities can be **extremely** large quantities.

Canonical Normalization Concept

Therefore, in order to **manage** the number of **significant** digits, without losing **accuracy**, we commonly employ canonical **normalization**, based on the context.

E.g. for **earth** bound missions, radius of **earth** is the parameter used for **distance** normalization (DU), while for **heliocentric** motion, Earth's orbital **radius** is used.

Canonical Normalization Method

With regard to **velocity**, value corresponding to **circular** motion on earth's surface (**SU**) is used, while for **time** normalization, **time** taken for one radian (**TU**) is used.

The above **normalizations** keep all quantities of **similar** magnitude, which helps in their **manipulation.**

Canonical Normalization Result

A **logical** consequence of such a **normalization** is to make gravitational parameter, **μ = 1,** as shown below.

$$1\ \mathrm{SU} = \frac{1\ \mathrm{DU}}{1\ \mathrm{TU}} = \sqrt{\frac{\mu}{1\ \mathrm{DU}}}; \quad \mu = \frac{1\ \mathrm{DU}^3}{1\ \mathrm{TU}^2} = 1$$

Position / Velocity Vector Solutions

Orbital Parameter Modification

In the context of **space** missions, we generally convert **'r'** and **'v'** at the end of the ascent mission in to **orbital** parameters e.g. 'a', 'e', 'T_0', 'Ω', ω', 'θ (or M)' and 'i'.

While, this is **sufficient** for operationalizing most spacecraft, there are **many** missions in which **final** objective is **different** from end of the **ascent** mission.

Typically, this **situation** is addressed by suitably **modifying** the orbit through input of **additional energy**.

Orbital Parameter Modification

In this **regard**, it is to be noted that **orbital** parameters acquire new **values** and hence, need **new** set of values for ‘r’ and ‘v’.

An example is **return** mission, where we need **fix** ‘r’ and ‘v’ **vectors** with respect to **ECI frame**, for propagating the applicable **equations** of motion.

‘r’ and ‘v’ from Orbital Parameters

Another case arises when **we** need to shift the **spacecraft** to an entirely **new** orbit and need to **locate** it back in the **ECI** for assessing changes.

Thus, we need a **strategy** to back-calculate **‘r’ and ‘v’** vectors from the **available** orbital parameters.

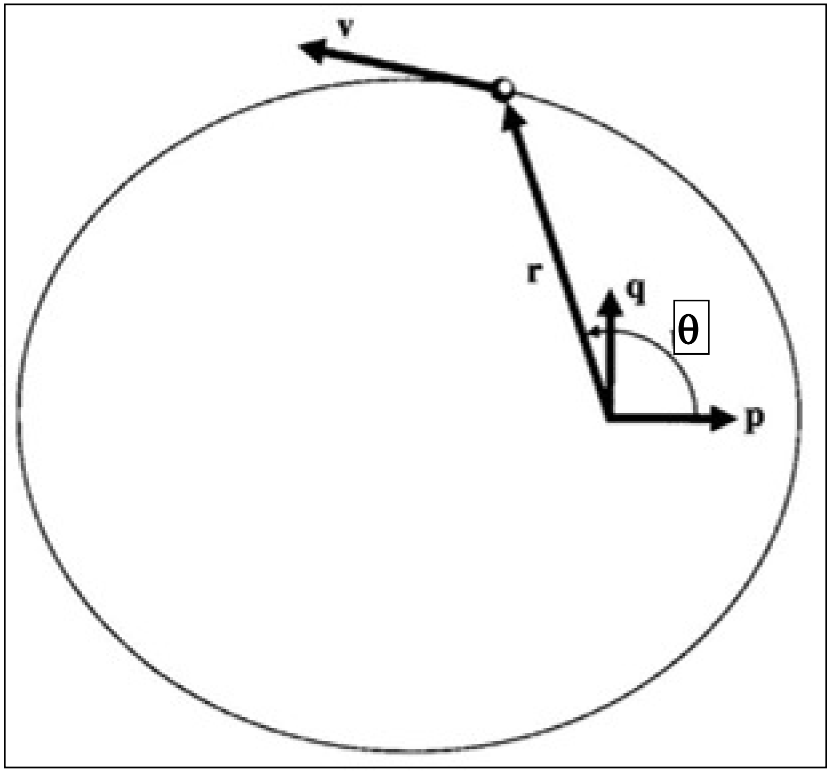
This is the **reverse** of the process **employed** for obtaining the orbital **parameters** from ‘r’ and ‘v’ and is **initiated** in the **orbital** plane, as explained next.

Orbital Velocity Solution

Consider **orbital plane** as shown alongside.

Here, '**r**', '**v**' are vectors in the **orbital plane.**

Further, '**p**' and '**q**' are unit **vectors** defining the orbital **plane.**



'r' and 'v' in Orbital Plane

We can now **obtain** position & velocity as **follows.**

$$\vec{r} = r\cos\theta\,\hat{p} + r\sin\theta\,\hat{q}; \quad h = r\left(r\dot{\theta}\right) = \sqrt{\mu p}$$

$$\vec{v} = \dot{\vec{r}} = \left(\dot{r}\cos\theta - r\dot{\theta}\sin\theta\right)\hat{p} + \left(\dot{r}\sin\theta + r\dot{\theta}\cos\theta\right)\hat{q}$$

$$r\dot{\theta} = v_t = \frac{h}{r} = \sqrt{\frac{\mu}{p}}\cdot(1+e\cos\theta), \quad \dot{r} = r\dot{\theta}\cdot\frac{e\sin\theta}{1+e\cos\theta}$$

$$\dot{r} = \sqrt{\frac{\mu}{p}}\cdot e\sin\theta; \quad \vec{v} = \sqrt{\frac{\mu}{p}}\cdot\left[-\sin\theta\,\hat{p} + (e+\cos\theta)\hat{q}\right]$$

Vectors '**r**' and '**v**' capture the **motion** of spacecraft within the **orbital** plane, with respect to **origin** (focus).

r & v in Geographical Frame

Next, **r & v** in orbital plane can be **transformed** into geographical frame (**ECI**) as follows.

$$\vec{r}_{ijk} = [R]\vec{r}_{pqw}; \quad \vec{v}_{ijk} = [R]\vec{v}_{pqw}; \quad [R] \rightarrow \text{ Rotation Matrix}$$

$$[R] = R_{\omega} \cdot R_{i} \cdot R_{\Omega}; \quad \rightarrow \text{ Elementary Rotations}$$

$$\Omega, i, \omega \rightarrow \text{ Euler angles for rigid body dynamics}$$

$$[R] = [\hat{p} \,|\, \hat{q} \,|\, \hat{w}]; \quad \hat{p} = \frac{\vec{e}}{|\vec{e}|}, \quad \hat{w} = \frac{\vec{H}}{|\vec{H}|}, \quad \hat{q} = \hat{w} \times \hat{p}$$

Thus, we can **switch** between initial **conditions** and orbital elements, as per the **need.**

Rotation Matrix Derivation

The rotation matrix **[R]** can be obtained as **follows.**

$$(x,y,z)=(p,q,w)[R]=(p,q,w)[R_\omega]\cdot[R_i]\cdot[R_\Omega]$$

$$[R_\Omega]=\begin{bmatrix}\cos\Omega & \sin\Omega & 0\\ -\sin\Omega & \cos\Omega & 0\\ 0 & 0 & 1\end{bmatrix};\quad [R_i]=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos i & \sin i\\ 0 & -\sin i & \cos i\end{bmatrix}$$

$$[R_\omega]=\begin{bmatrix}\cos\omega & \sin\omega & 0\\ -\sin\omega & \cos\omega & 0\\ 0 & 0 & 1\end{bmatrix};\quad [R]=\begin{bmatrix}l_1 & m_1 & n_1\\ l_2 & m_2 & n_2\\ l_3 & m_2 & n_3\end{bmatrix}$$

Next, we can now perform the **above** multiplications and arrive at **nine** scalar components of [R] **matrix.**

R Matrix Scalar Elements

The scalar **elements of [R]** are as follows.

$$
\begin{aligned}
l_1 &= \cos\omega\cos\Omega - \sin\omega\sin\Omega\cos i \\
m_1 &= \cos\omega\sin\Omega + \sin\omega\cos\Omega\cos i \\
n_1 &= \sin\omega\sin i \\
l_2 &= -\sin\omega\cos\Omega - \cos\omega\sin\Omega\cos i \\
m_2 &= -\sin\omega\sin\Omega + \cos\omega\cos\Omega\cos i \\
n_2 &= \cos\omega\sin i \\
l_3 &= \sin\Omega\sin i; \quad m_3 = -\cos\Omega\sin i; \quad n_3 = \cos i
\end{aligned}
$$

Position/Velocity Scalar Elements

The scalar **elements of r & v** are as follows.

$$x = l_1 r\cos\theta + l_2 r\sin\theta$$
$$y = m_1 r\cos\theta + m_2 r\sin\theta$$
$$z = n_1 r\cos\theta + n_2 r\sin\theta$$
$$\frac{dx}{dt} = \frac{\mu}{h}\left[-l_1\sin\theta + l_2(e+\cos\theta)\right]$$
$$\frac{dy}{dt} = \frac{\mu}{h}\left[-m_1\sin\theta + m_2(e+\cos\theta)\right]$$
$$\frac{dz}{dt} = \frac{\mu}{h}\left[-n_1\sin\theta + n_2(e+\cos\theta)\right]$$

(x, y, z) are defined in the **ECI** frame, as shown earlier.

Summary

Canonical normalization is a useful **tool** for ensuring the accuracy of **numerical** calculations.

We further **note** that it is possible to go back to **initial** conditions from orbital **information** through a standard form of **transformation**.