Orbital Elements

• Defining an orbit in the plane required 2 elements, typically \((h, e)\) or \((a, e)\).

• Adding a third element, \(\theta\), identified spacecraft location in the orbit.

• Everything else needed could be found from these 3 parameters (3DOF).

• Determining an orbit in 3D will require us to identify 5 parameters, with a total of 6 parameters needed for position information (6DOF).

The remaining information can be found from the process needed to transform GEF \((G)\) into the perifocal frame \((P)\)
Orbital Elements

- Defining an orbit in the plane required 2 elements, typically \((h, e)\) or \((a, e)\).
- Adding a third element, \(\theta\), identified spacecraft location in the orbit.
- Everything else needed could be found from these 3 parameters (3DOF).
- Determining an orbit in 3D will require us to identify 5 parameters, with a total of 6 parameters needed for position information (6DOF).

The remaining information can be found from the process needed to transform GEF \((G)\) into the perifocal frame \((P)\).

\[
G \rightarrow P \text{ is a 3-1-3 Euler angle transformation}
\]
• Two planes of importance in these frames, the equatorial plane (\(XY\)) and the orbit plane (\(x\bar{y}\))

• Any two planes that share a point must intersect, and these share the center of the Earth (the focus of the orbit)

• Unless the planes are the same, an intersection among two planes takes place along a line in space.

• We call this line the **node line**.

• The orbit crosses the node line at two points, the **ascending node** and the **descending node**.
• Let \( \mathbf{N} \) be a vector in the direction of the ascending node.
• \( \mathbf{N} \) is in the equatorial plane, so \( \mathbf{N} \perp \mathbf{Z} \)
• We can compute a vector in this direction using \( \mathbf{N} = \mathbf{Z} \times \mathbf{h} \)

\[
\Rightarrow \mathbf{N} = v_Z \mathbf{r} - r_Z \mathbf{v}
\]

where \( \{\mathbf{r}\}_G = [r_X, r_Y, r_Z]^T \)

\( \{\mathbf{v}\}_G = [v_X, v_Y, v_Z]^T \)

The right-handedness of the cross product will automatically give us the correct (ascending) direction.
• The first Euler angle of the $G \rightarrow P$ transformation is the **right ascension of the ascending node**, $\Omega$, often abbreviated RAAN (pronounced “Ran”)

• This angle is a rotation about the 3 axis of the $G$ coordinate frame ($\mathbb{Z}$)

• $\Omega$ can be computed as the angular displacement from the old 1 axis ($\mathbb{X}/\mathbb{Y}$) to the new 1 axis ($\mathbb{N}$):

$$\cos \Omega = \frac{\mathbb{N} \cdot \mathbb{X}}{|\mathbb{N}|} = \frac{N_X}{N}$$

where $\{\mathbb{N}\}^G = [N_X \ N_Y \ N_Z]^T$; $N = |\mathbb{N}|$

$\Omega$ has a value from [0,2\pi) radians ([0,360) deg)
Most implementations of $\cos^{-1}()$ (ACOS) return values between $[0, \pi]$ radians, so the ACOS function is multi-valued on the domain of $\Omega$.

Could use the $Y$ component of the node vector to check if we are in the range $(\pi, 2\pi)$ radians:

$$\sin \Omega = \frac{N \cdot Y}{|N|} = \frac{N_Y}{N}$$

So, a hand algorithm could be:

$$\Omega^* = \cos^{-1}(N_X/N)$$

if $(N_Y > 0)$; $\Omega = \Omega^*$

otherwise; $\Omega = 2\pi - \Omega^*$
Most implementations of $\cos^{-1}()$ \textbf{(ACOS)} return values between $[0,\pi]$ radians, so the ACOS function is multi-valued on the domain of $\Omega$.

Could use the $Y$ component of the node vector to check if we are in the range $(\pi,2\pi)$ radians:

$$\sin \Omega = \frac{\mathbf{N} \cdot \mathbf{Y}}{|\mathbf{N}|} = \frac{N_Y}{N}$$

So, a hand algorithm could be:

$$\Omega^* = \cos^{-1}(N_X/N)$$

if $(N_Y > 0)$; $\Omega = \Omega^*$

otherwise; $\Omega = 2\pi - \Omega^*$

or

$$\Omega = \tan^{-1}\left(\frac{\sin \Omega}{\cos \Omega}\right) = \text{ATAN2}(N_Y, N_X)$$
The dihedral between the orbital plane and the equatorial plane forms the second Euler angle of the $G \rightarrow P$ transformation.

This angle is called the inclination of the orbit, $i$.

The second rotation (around the new $1$ axis, $N$) moves the $3$ axis from $Z$ to normal to the orbit plane (parallel to $h$).

The value of inclination runs from $[0, \pi]$ radians ($[0, 180]$ deg) with angles less than $\pi/2$ radians ($90^\circ$) being prograde and angles larger than $\pi/2$ being retrograde.
• The dihedral between the orbital plane and the equatorial plane forms the second Euler angle of the $G \rightarrow P$ transformation.
• This angle is called the **inclination** of the orbit, $i$
• The second rotation (around the new 1 axis, $N$) moves the 3 axis from $Z$ to normal to the orbit plane (parallel to $h$)

The value of inclination runs from $[0, \pi]$ radians ($[0,180]$ deg) with angles less than $\pi/2$ radians ($90^\circ$) being **prograde** and angles larger than $\pi/2$ being **retrograde**

Orbits with inclinations exactly $\pi/2$ radians ($90^\circ$) are called **polar**

Orbital Elements

**Diagram of the 3 Euler Angles for the Perifocal Frame**
Orbital Elements

• The third Euler angle is the angular separation from the ascending node to periapsis, known as the argument of periapsis, $\omega$.

• $\omega$ can range from $[0,2\pi]$ radians ($[0,360)$ deg).

• Since $\mathbf{e}$ points towards periapsis, it satisfies:

$$\cos \omega = \frac{\mathbf{N} \cdot \mathbf{e}}{Ne}$$

• Due to ambiguity of ACOS, this is not sufficient to compute $\omega$.
We begin to measure $\omega$ at the ascending node and sweep through the orbital plane.

Thus, if $\omega \in [0, \pi)$ radians we reach periapsis before we reach the descending node, and $\mathbf{e}$ lies above the equatorial plane ($e_Z > 0$). Otherwise, it is below the equatorial plane and $\omega > \pi$ radians.

So, a hand algorithm could be:

$$\omega^* = \cos^{-1}\left[\mathbf{N} \cdot \mathbf{e} / (N e)\right]$$

if $(e_Z > 0);$ $\omega = \omega^*$

otherwise; $\omega = 2\pi - \omega^*$
• Some computer algorithms prefer to reduce branching (no “if-then”s).
• To do this for $\omega$ requires an expression for its sine. Let’s define a vector advanced $\pi/2$ radians from $\mathbf{N}$ in the orbit plane as $\mathbf{s}$.
• Since $\mathbf{s} \perp \mathbf{N}$ and $\mathbf{s} \perp \mathbf{h}$ we can compute a unit vector in the correct direction as

$$\mathbf{s} = \left( \mathbf{h} / h \right) \times \left( \mathbf{N} / N \right) = \frac{1}{hN} \mathbf{h} \times \mathbf{N}$$

and then $\sin \omega = \frac{\mathbf{e} \cdot \mathbf{s}}{e}$
Some computer algorithms prefer to reduce branching (no “if-then”s).

To do this for $\omega$ requires an expression for its sine. Let’s define a vector advanced $\pi/2$ radians from $\mathbf{N}$ in the orbit plane as $\mathbf{s}$.

Since $\mathbf{s} \perp \mathbf{N}$ and $\mathbf{s} \perp \mathbf{h}$ we can compute a unit vector in the correct direction as

$$
\mathbf{s} = \left( \frac{\mathbf{h}}{h} \right) \times \left( \frac{\mathbf{N}}{N} \right) = \frac{1}{hN} \mathbf{h} \times \mathbf{N}
$$

and then

$$
\sin \omega = \frac{\mathbf{e} \cdot \mathbf{s}}{\mathbf{e}}
$$

$$
\Rightarrow \omega = \tan^{-1}\left( \frac{\sin \omega}{\cos \omega} \right) = \text{ATAN2} \left[ \mathbf{e} \cdot \mathbf{s}, (\mathbf{N} \cdot \mathbf{e})/N \right]
$$
Orbital Elements

Diagram of the 3 Euler Angles for the Perifocal Frame
• The 3 Euler angles define the orientation of the orbit with respect to inertial space
• With the addition of 2 parameters (for an elliptical or hyperbolic orbit) defining the orbit size/shape, an orbit can be specified completely in 3 dimensions.
• To specify a position along the orbit, we typically add an angular measure from periapsis (such as $\theta$) though time since periapsis would work as well.
The 3 Euler angles define the orientation of the orbit with respect to inertial space.

With the addition of 2 parameters (for an elliptical or hyperbolic orbit) defining the orbit size/shape, an orbit can be specified completely in 3 dimensions.

To specify a position along the orbit, we typically add an angular measure from periapsis (such as $\theta$) though time since periapsis would work as well.

The position information is meaningless for most applications unless we specify the time an object was in that position, known as the *epoch*. 
• Since $\theta$ is measured from periapsis, it can be computed from inertial state vectors using its relationship to $\mathbf{e}$:

$$\cos \theta = \frac{\mathbf{r} \cdot \mathbf{e}}{re}$$

• Due to the ambiguity with ACOS this isn’t sufficient. However, we know that for $\theta \in (0, \pi)$ radians we are moving away from the body while for $\theta \in (\pi, 2\pi)$ radians we are moving inward, which can be expressed using $v_r$.

• If $\theta > \pi$ we have $\mathbf{r} \cdot \mathbf{v} < 0$
A suitable hand algorithm could be:

\[
\theta^* = \cos^{-1}\left[\frac{\mathbf{r} \cdot \mathbf{e}}{re}\right]
\]

if \((\mathbf{r} \cdot \mathbf{v}) > 0\); \(\theta = \theta^*\)

otherwise; \(\theta = 2\pi - \theta^*\)
A suitable hand algorithm could be:

\[
\theta^* = \cos^{-1}\left[\mathbf{r} \cdot \mathbf{e} / (re)\right]
\]

if \((\mathbf{r} \cdot \mathbf{v} > 0)\); \(\theta = \theta^*\)

otherwise; \(\theta = 2\pi - \theta^*\)

- Again, if we want to avoid branching we will want the sine.
- \(\sin \theta\) would give us a projection along the \(\hat{q}\) direction in the \(P\) frame, so one method is to get a vector along that direction (\(\hat{q} \perp \mathbf{e}\) and \(\hat{q} \perp \mathbf{h}\))
• A unit vector in the correct direction (but not necessarily associated with the $P$ frame) would be:

$$q = \left(\frac{\mathbf{h}}{h}\right) \times \left(\frac{\mathbf{e}}{e}\right) = \frac{1}{he} \mathbf{h} \times \mathbf{e}$$

and then

$$\sin \theta = \frac{r \cdot q}{r}$$

• So

$$\theta = \tan^{-1} \left(\frac{\sin \theta}{\cos \theta}\right) = \text{ATAN2}[r \cdot q, (r \cdot e)/e]$$
6 Classical Orbital Elements:

- \( h \), specific angular momentum
- \( i \), inclination [0°, 180°). (>90° is retrograde)
- \( \Omega \), RAAN, [0°, 360°)
- \( e \), eccentricity
- \( \omega \), argument of periapsis, [0°, 360°)
- \( \theta \), true anomaly, [0°, 360°)

First 5 fix the orientation and shape of orbit, true anomaly fixes our position in the orbit.

Some common substitutions are \((a, p) \rightarrow h\) or \((M_e, \Delta t_p, t_0) \rightarrow \theta\)
Algorithm: Compute the Classical Orbital Elements (COE) set given state vectors, \( r \) and \( v \), and \( \mu \):

1. Compute the position magnitude \( r = |r| \)
2. Compute the position magnitude \( v = |v| \)
3. Compute the radial velocity \( v_r = \frac{v \cdot r}{r} \)
4. Compute the angular momentum vector: \( h = r \times v \)
5. Compute the momentum magnitude \( h = |h| \)
6. Compute the radial velocity \( v_r = \frac{v \cdot r}{r} \)
7. Compute the angular momentum vector: \( h = r \times v \)
8. Using Eq. (4.3) compute the eccentricity vector:

\[
\mathbf{e} = \left( \frac{v^2}{\mu} - \frac{1}{r} \right) \mathbf{r} - \left( \frac{r v_r}{\mu} \right) \mathbf{v}
\]
9. Compute the eccentricity magnitude \( e = |e| \)
10. (OPTIONAL) Compute \( a \) if required \((e \neq 1)\):

\[
a = \frac{h^2}{\mu(1 - e^2)}
\]
11. (OPTIONAL) Compute \( p \) if desired:

\[
p = \frac{h^2}{\mu}
\]
12. Compute the ascending node vector from Eq. (4.22):

\[
\mathbf{N} = \mathbf{Z} \times \mathbf{h} = v_2 \mathbf{r} - r_2 \mathbf{v}
\]

Note that \( \mathbf{N} = 0 \) for an equatorial orbit \((i = 0 \text{ or } i = \pi)\).
13. Compute the magnitude \( N = |\mathbf{N}| \)
14. Compute RAAN using the hand procedure from Eq. (4.23) or

\[
\Omega = \text{ATAN2}(N_Y, N_X)
\]

If \((N \approx 0)\), define \( \Omega = 0 \)
15. Compute inclination using Eq. (4.24):

\[
i = \cos^{-1}(h_Z/h)
\]
16. Compute argument of periapsis using the hand procedure from Eq. (4.25) or

\[
\omega = \text{ATAN2}[\mathbf{e} \cdot \mathbf{q}, (\mathbf{r} \cdot \mathbf{e})/e]
\]

If \((e \approx 0), \text{ use } \mathbf{N} \text{ instead of } \mathbf{e}.\)
If \((N \approx 0)\), use \( \mathbf{X} \) instead of \( \mathbf{N} \).
If additional anomalies or time since periapsis are required, these can be computed using the equations of Chapter 6.
17. Compute true anomaly using the hand procedure from Eq. (4.27) or

\[
\theta = \text{ATAN2}[(\mathbf{r} \cdot \mathbf{q}, (\mathbf{r} \cdot \mathbf{e})/e)]
\]

If \((e \approx 0), \text{ use } \mathbf{N} \text{ instead of } \mathbf{e}.\)
If \((N \approx 0), \text{ use } \mathbf{X} \text{ instead of } \mathbf{N} \).
If additional anomalies or time since periapsis are required, these can be computed using the equations of Chapter 6.
• As mentioned, 3 of the orbital elements describe a 3-1-3 Euler angle transformation from Geocentric Equatorial coordinates \((G)\) to perifocal coordinates \((P)\).

• We can use elementary transformations to both visualize the rotations that these angles specify, and to transform state vectors from one system to another.
Orbital Elements

- First stage of transformation: Rotate \( P \) relative to \( G \) around the 3-axis (\( Z \)) by angle \( \Omega \).

\[
C_3(\Omega) = \begin{bmatrix}
\cos\Omega & \sin\Omega & 0 \\
-\sin\Omega & \cos\Omega & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

Rotation about the Z-axis.
• Second stage of transformation:
  Rotate $P$ relative to $G$ around the new 1-axis ($N$) by angle $i$

\[
C_1(i) = \begin{bmatrix}
  1 & 0 & 0 \\
  0 & \cos i & \sin i \\
  0 & -\sin i & \cos i
\end{bmatrix}
\]

Rotation about the X-axis.
• Last stage of transformation: Rotate $P$ relative to $G$ around the new 3-axis ($\mathbf{h}$) by angle $\omega$.

$$C_3(\omega) = \begin{bmatrix} \cos \omega & \sin \omega & 0 \\ -\sin \omega & \cos \omega & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation about the Z-axis.
Orbital Elements

• To convert orbital elements to state vectors, we first compute the positions and velocity relative to the $P$ frame, then transform into the inertial frame.

• Recall: $\mathbf{r} = \bar{x}\hat{\mathbf{p}} + \bar{y}\hat{\mathbf{q}}$

$$\mathbf{v} = \bar{x}\hat{\mathbf{p}} + \bar{y}\hat{\mathbf{q}}$$

From the Euler angle transformation (313) $G \rightarrow P$:

$$C_{P/G} = C_3(\omega)C_1(i)C_3(\Omega)$$

$$C_{G/P} = \left[ C_{P/G} \right]^T$$

$$\{\mathbf{r}\}_G = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = C_{G/P} \{\mathbf{r}\}_P$$

$$\{\mathbf{v}\}_G = \begin{bmatrix} v_X \\ v_Y \\ v_Z \end{bmatrix} = C_{G/P} \{\mathbf{v}\}_P$$

SO

$$\{\mathbf{r}\}_P = \frac{h^2 / \mu}{1 + e \cos \theta} \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \end{bmatrix}$$

$$\{\mathbf{v}\}_P = \frac{\mu}{h} \begin{bmatrix} -\sin \theta \\ e + \cos \theta \\ 0 \end{bmatrix}$$