Assignment 5: Satellite orbits

Questions

Dr. Robert Heinkelmann

Assignment 5

The orbits of six satellites are given in terms of their two line elements (attachment 1) in the Earth Centered Inertial (ECI) system.

Attachment

1. Two line elements of six Earth satellites (source: Heavens Above)

Satellite name	i (deg)	Ω (deg)	e	ω (deg)	M (deg)	n (d^{-1})	t (date, time) UT1
ISS	51.7	19.1	0.00	17.6	10.4	15.50	2022-01-14, 04:49
Sentinel 1B	98.2	23.5	0.00	80.6	279.5	14.59	2022-01-13, 22:15
Molniya-1T	63.6	198.3	0.62	299.8	13.0	3.19	2022-01-12, 18:29
Galileo 9	55.7	266.4	0.00	14.0	346.1	1.71	2022-01-09, 02:07
Beidou IGSO 3	60.1	55.8	0.00	192.4	355.6	1.00	2022-01-13, 01:41
Meteosat 11	0.3	325.8	0.00	117.3	338.5	1.00	2022-01-13, 20:35

a (semi-major axis) can be computed from mean motion **n** via the orbital period **T**.

Thereafter **a**, **e** (ellipse size and shape), **Ω**, **ω** and **i** (orientation angles to ECI) and **M** the mean anomaly (position in the orbit) are given.

Two line elements

A commonly used conventional format for specification of orbital elements [1]

```
1 25544U 98067A   20167.53669693  .00016717  00000-0  10270-3 0  9033
2 25544  51.6417 356.6431 0002136  45.6438 314.4887 15.49487269 31752
```

Interesting elements for this assignment:

The date (year yy) and time (UT1 day.decimal) (line 1, item 3).
Date and time are important
1) to align the orbital position (anomaly) with the Earth rotation phase (GAST / ERA) and
2) because in reality the orbital elements change with time („orbit perturbation“).

Inclination (degrees): 51.6417 (line 2, item 2)
Right Ascension of the Ascending Node (degrees): 356.6431 (line 2, item 3)
Eccentricity (decimal point assumed): 0.0002136 (line 2, item 4)
Argument of Perigee (degrees): 45.6438 (line 2, item 5)
Mean Anomaly (degrees): 314.4887 (line 2, item 6)
Mean Motion (revolutions per day): 15.49487269 (line 2, item 7)

1) http://celestrak.com/columns/v04n03/

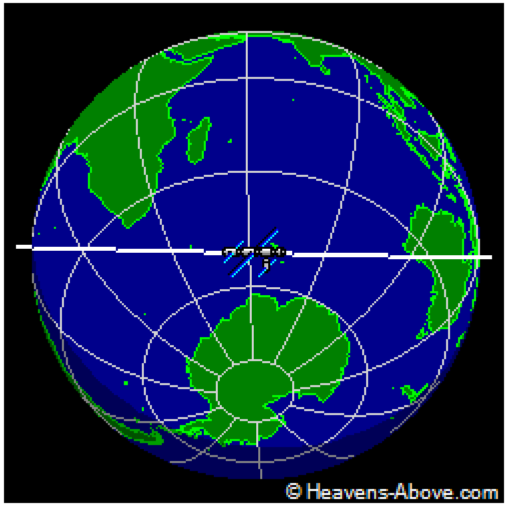
Credits: Heavens above

The most popular source for orbital elements [1]

Heavens-Above GmbH, Munich, Germany

a non-profit website developed and maintained by Chris Peat

Hosted by DLR/GSOC



Current position of ISS

1) https://www.heavens-above.com/

Exercise 1

Specify the orbital periods T of the satellites in UT1 (min precision).

Starting with the mean motion of the satellite in turns per day n_{sat} $[\mathrm{d}^{-1}]$ from the two line elements.

We calculate **the orbital period** of the Earth satellite in an equivalent circular orbit:
From 3rd Keplerian law:

$$T_{sat} = \frac{1}{n_{sat}}$$

If the mean motion is given in d^{-1} we will get T in d: $T_{sat}[d]$.

T in hours: $T_{sat}[h] = T_{sat}[d] \cdot 24$

Now convert $T_{sat}[h]$ to [hour, minute, second] using your code from assignment#2 and specify the orbital period rounded to integer minutes.
! For the further calculations, continue with the exact orbital period. !

Exercise 2

What semi major axis a (m precision), perigee and apogee heights do the satellite orbits have?

Starting with the orbital period T_{sat} from exercise#1. (! Dont use the rounded value !)

The semi-major axis of an Earth satellite knowing the orbital period and the geocentric gravitational constant $GM_{\oplus} = 398600.44 \text{ km}^3\text{s}^{-2}$:

From 3rd Keplerian law:

$$T_{sat}{}^2 = \frac{4\pi^2}{GM_{\oplus}} \cdot a_{sat}{}^3 \Leftrightarrow a_{sat} = \sqrt[3]{\frac{T_{sat}{}^2 \cdot GM_{\oplus}}{4\pi^2}}$$

where T_{sat} [s] is the orbital period (see last slide but in seconds) compatible with $GM_{\oplus}$ $[\text{km}^3\text{s}^{-2}]$ results in semi-major axis a_{sat}[km]. Then display as a_{sat} [m]=a_{sat} [km]*1000 and round to integer m. (For the remaining calculations, continue with the semi-major axis.) Perigee height: $a_{sat}(1 - e_{sat})$[m] Apogee height: $a_{sat}(1 + e_{sat})$[m]

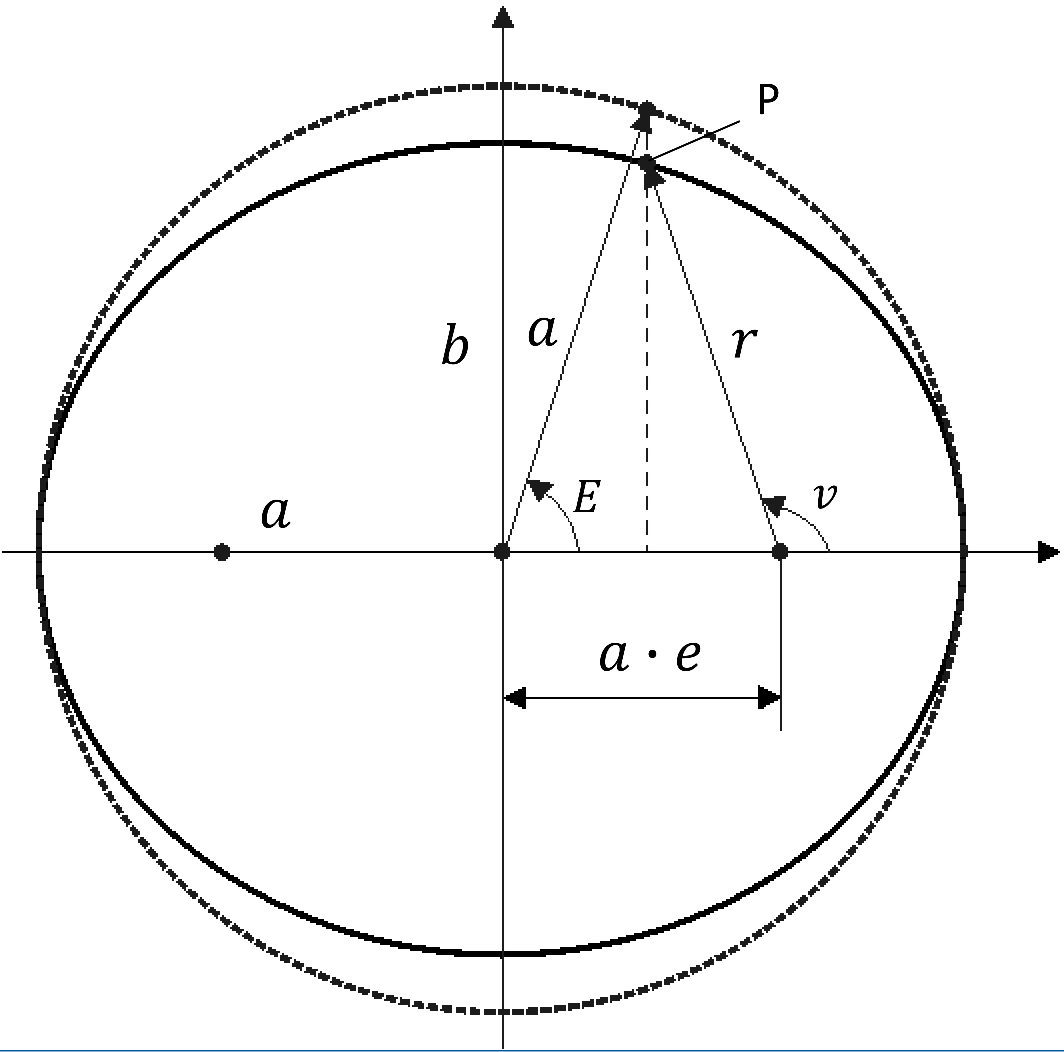
! For the further calculations, continue with the exact semi-major axis. !

Exercise 3

Compute the position and velocity vectors of the satellites (attachment 1) during one day (t = start epoch, time increment 1 minute) in the ECI system.

We need to compute the ECI-compatible position vector from the two line elements, T_{sat}, and a_{sat}.

Exercise 3: position vector

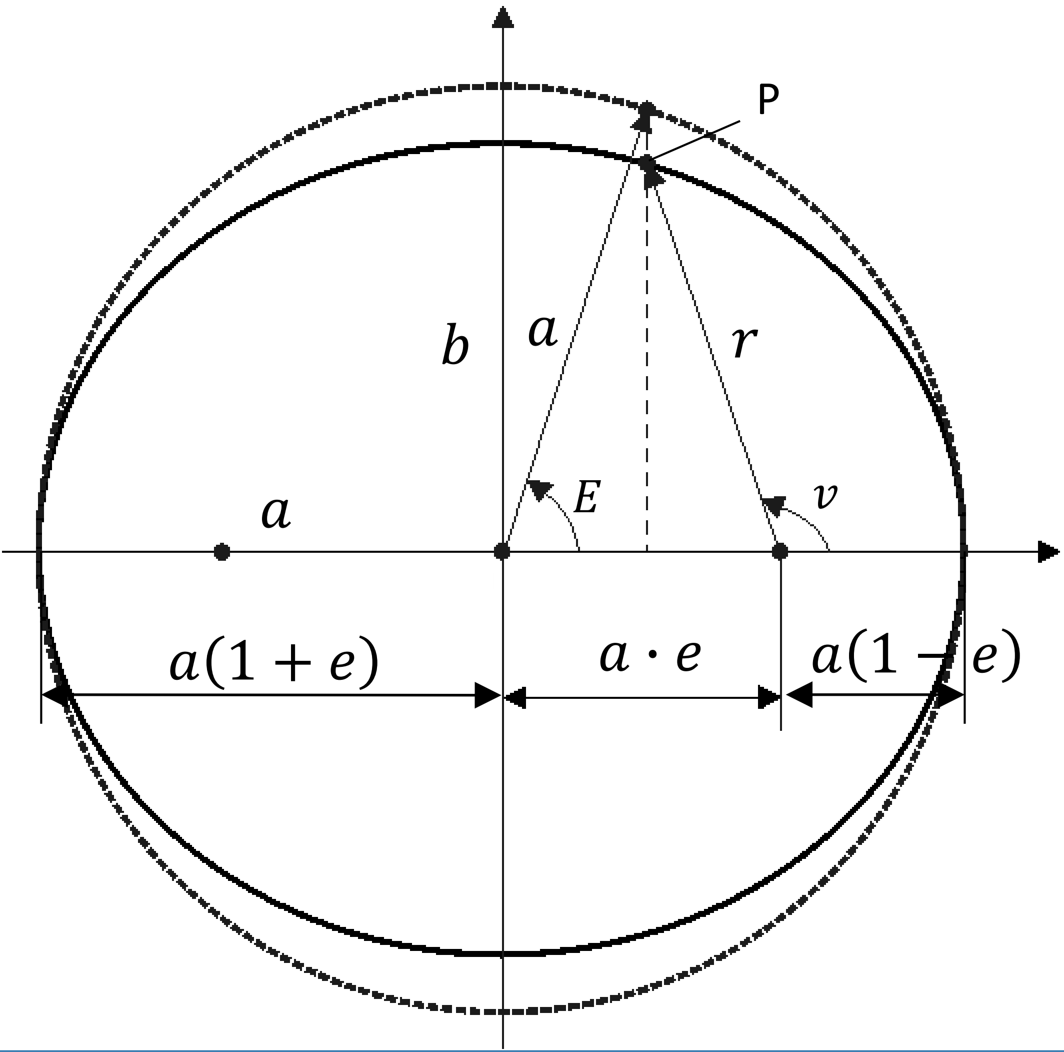


Given: Two line elements, T_{sat}, a_{sat}
Goal: position vector in orbital system

Mean anomaly at time t_i?
Start anomaly M_{sat} (see table)
Time increment $1[\mathrm{min}]$
i = 1 : 1440
$t_i = (i-1) \cdot 1[\mathrm{min}]$
$t_i[\mathrm{d}] = t_i[\mathrm{min}]/1440$
$M_i = t_i[\mathrm{d}] \cdot n_{sat}[1/\mathrm{d}] \cdot 2\pi[\mathrm{rad}]$
$M_{sat,i} = M_{sat}[\mathrm{rad}] + M_i[\mathrm{rad}]$

Eccentric anomaly $E_{sat,i}$ from mean anomaly $M_{sat,i}$ via inverse KEPLER equation $M_{sat,i} = E_{sat,i} - e_{sat} \sin E_{sat,i}$ iteration with break criterion = 1e-9
$\Rightarrow E_{sat,i}$

Exercise 3: position vector



Given: Two line elements, T_{sat}, a_{sat}, $E_{sat,i}$

Goal: position vector in orbital system

Distance of the satellite to Earth:

$$r_{sat,i} = a_{sat} \cdot \left(1 - e_{sat} \cdot \cos E_{sat,i}\right)$$

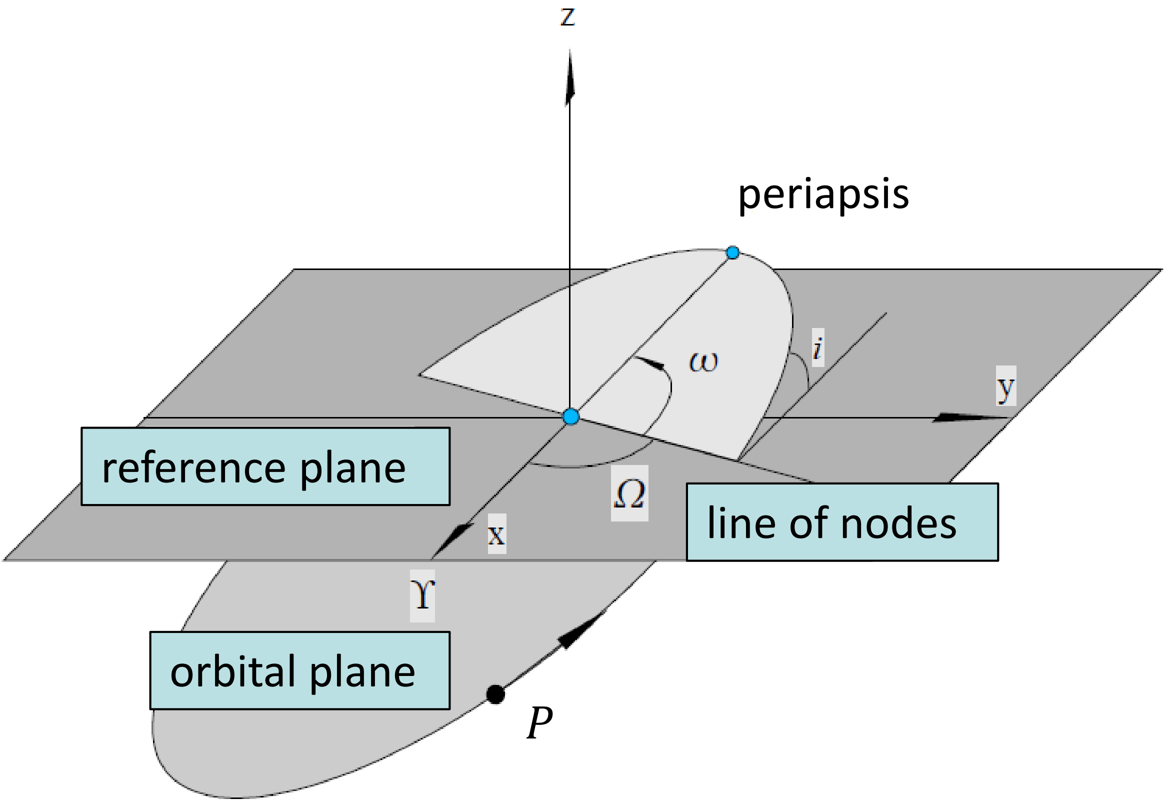
True anomaly v from eccentric anomaly E:

$$v_{sat,i} = \operatorname{atan2}\left(\frac{\sqrt{1 - {e_{sat}}^2} \sin E_{sat,i}}{\cos E_{sat,i} - e_{sat}}\right)$$

Position vector in orbital system:

$$\boldsymbol{r_{orb,sat,i}} = r_{sat,i} \cdot \begin{pmatrix} \cos v_{sat,i} \\ \sin v_{sat,i} \\ 0 \end{pmatrix}$$

Exercise 3: position vector in ECI



Given: position vector in orbital system $\boldsymbol{r_{orb}}$
Goal: position vector in ECI $\boldsymbol{r_{ECI}}$

Rotation from the orbital system to the reference system (ECI):

$$\boldsymbol{r_{ECI}} = R_3(-\Omega)R_1(-i)R_3(-\omega)\boldsymbol{r_{orb}}$$

Since the orbital elements are undisturbed, this rotation is time-independent (needs to be computed only once per satellite)!

$$\boldsymbol{r_{ECI}} = \begin{bmatrix} \cos\Omega\cos\omega - \sin\Omega\sin\omega\cos i & -\cos\Omega\sin\omega - \sin\Omega\cos\omega\cos i & \sin\Omega\sin i \\ \sin\Omega\cos\omega + \cos\Omega\sin\omega\cos i & -\sin\Omega\cos\omega + \cos\Omega\cos\omega\cos i & -\cos\Omega\sin i \\ \sin\omega\sin i & \cos\omega\sin i & \cos i \end{bmatrix}\boldsymbol{r_{orb}}$$

Note: Ω, i, ω depend on the satellite (only general solution is shown on this slide).

Exercise 3: ECI position vector plot

Plot the 3d-orbits together with a sphere of radius $R = 6371$ km in the same figure.

We have positions in ECI here => no information about the Earth surface, no plot with Earth surface image or coastlines! Purpose of orbital plot together with a spherical Earth is just to display the distance-scales.

To create a sphere, try `help sphere`
Example for sphere with radius 6371[km]:

```
[x,y,z] = sphere(180); x=6371.*x; y=6371.*y; z=6371.*z;
```

provides the spherical surface in Cartesian coordinates

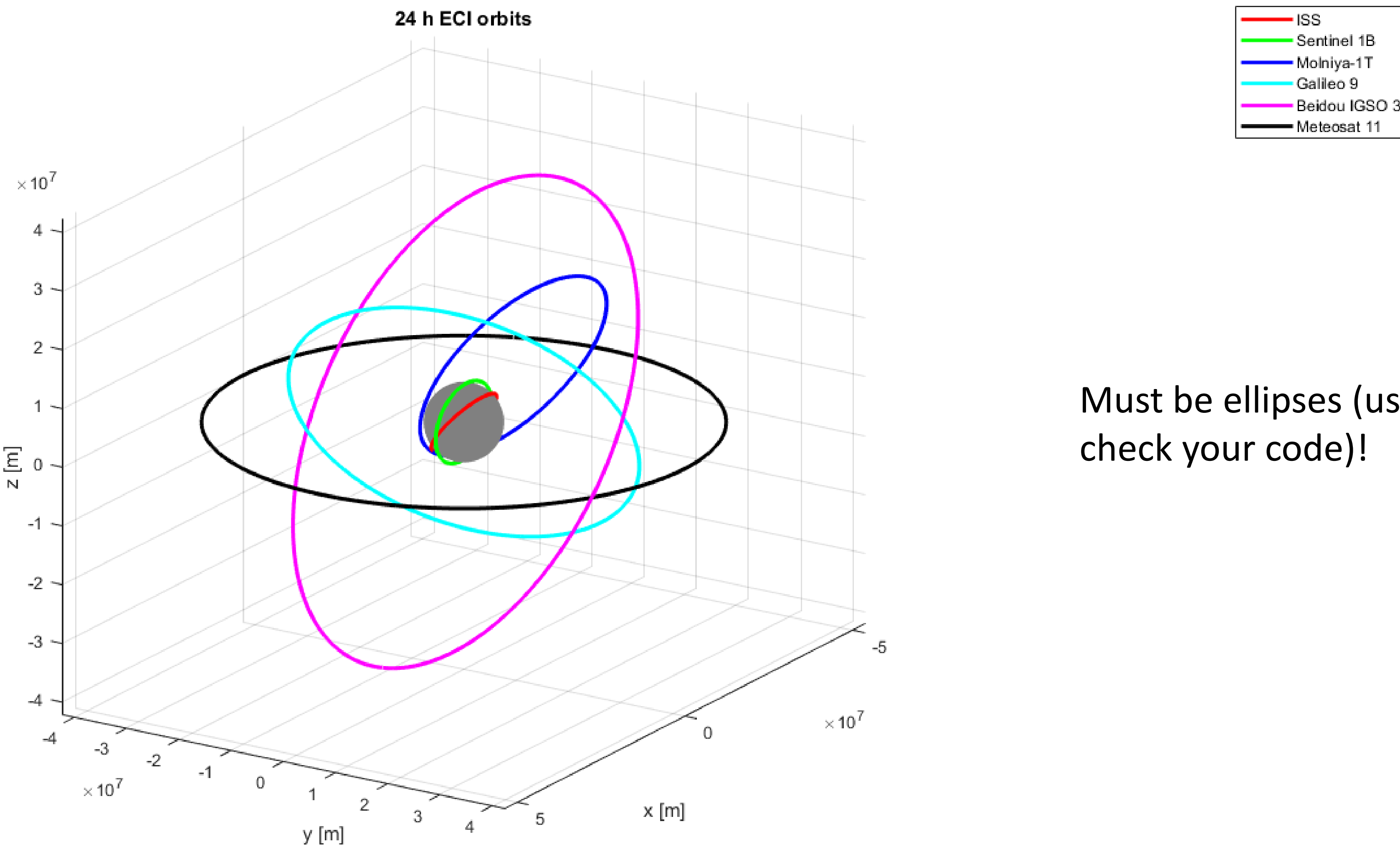
```
surf(x,y,z, 'FaceColor', 'none', 'EdgeColor', 0.5*[1 1 1]);
```

plots the spherical surface in grey color

then `plot3` the orbits after `hold on;`

Test: all orbits should be ellipses in ECI with the grey sphere at rest in a focal point!

Exercise 3: orbits in ECI



Must be ellipses (useful to check your code)!

Exercise 3: ECI velocity vector

Parameter of the ellipse: $p = r \cdot (1 + e \cdot \cos v)$

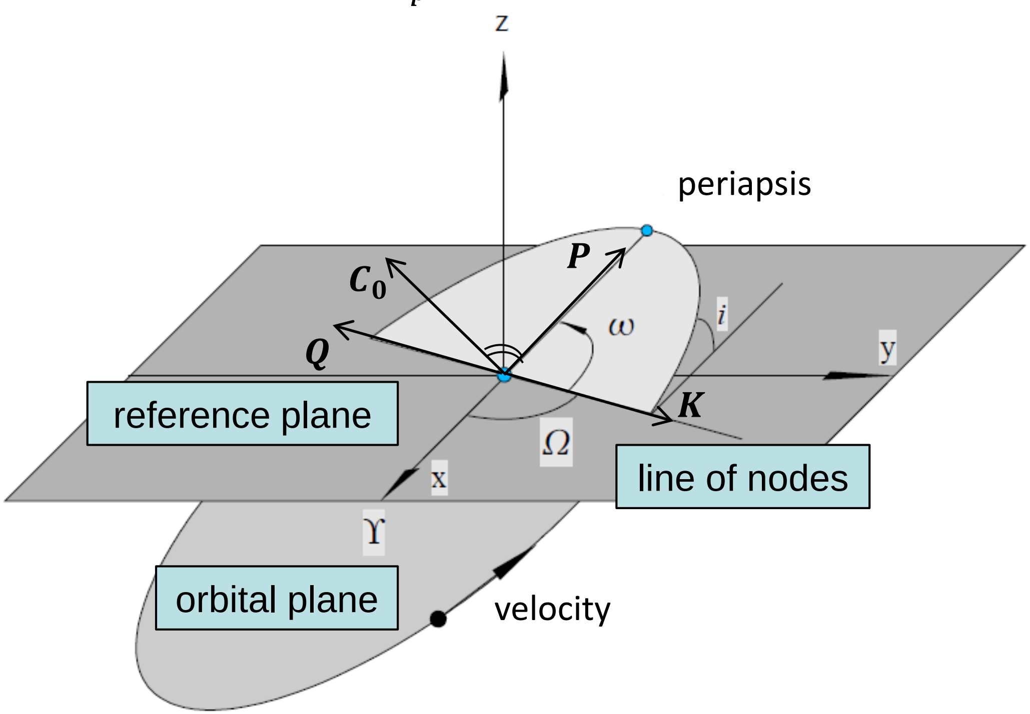
Magnitude of the vector normal to the orbital plane: $C = \sqrt{p \cdot GM}$

Unit vector normal to the orbital plane: $\widehat{\boldsymbol{C}} = (\sin\Omega \sin i \quad -\cos\Omega \sin i \quad \cos i)^T$

Unit vector of the line of nodes: $\widehat{\boldsymbol{K}} = (\cos\Omega \quad \sin\Omega \quad 0)^T$

Orbit system vectors: $\boldsymbol{P} = \cos\omega\, \widehat{\boldsymbol{K}} + \sin\omega \left(\widehat{\boldsymbol{C}} \times \widehat{\boldsymbol{K}}\right)$, $\quad \boldsymbol{Q} = -\sin\omega\, \widehat{\boldsymbol{K}} + \cos\omega\, (\widehat{\boldsymbol{C}} \times \widehat{\boldsymbol{K}})$

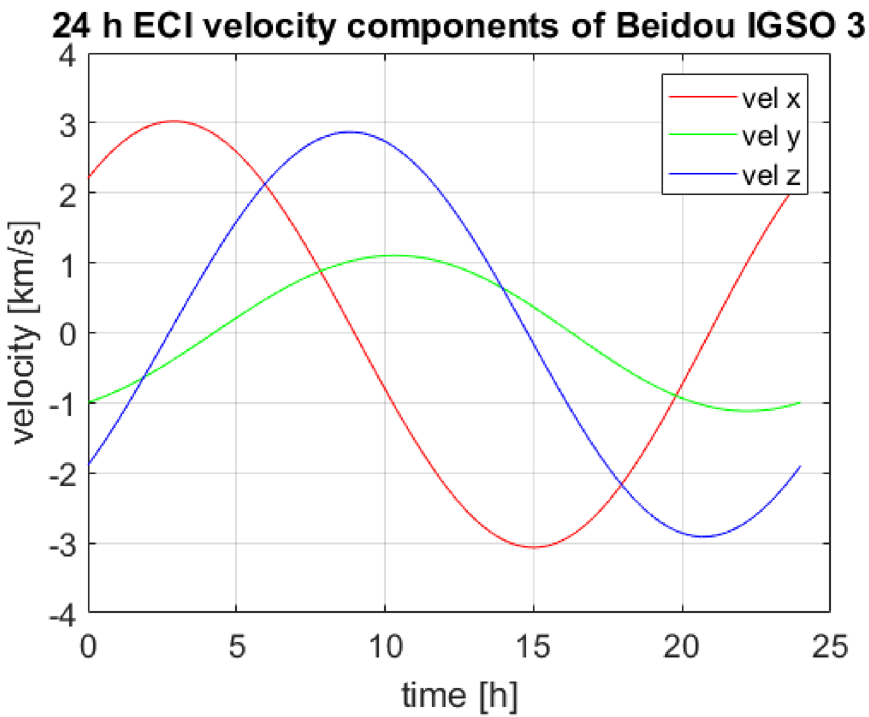
Velocity: $\boldsymbol{v_{orb}} = \frac{C}{p} \cdot (-\sin v \cdot \boldsymbol{P} + (e + \cos v) \cdot \boldsymbol{Q})$

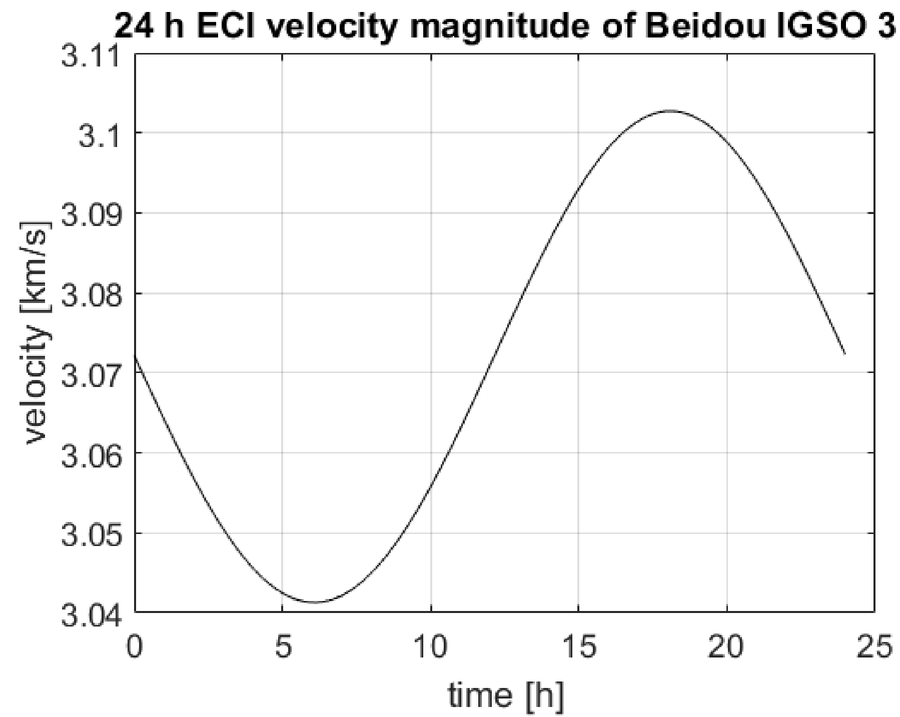


Note: all quantities depend on the satellite and eventually on time (t_i), what is just not indicated on this slide.

Exercise 3: ECI velocity vector plots

Plot the three velocity components and the magnitude of the velocity of each satellite each into an own figure.





These two velocity plots are to be drawn for each of the six satellite. The plots are just a qualitative example. Correct plots will look differently.

Exercise 4

Transform the ECI positions to the Earth Centered Earth Fixed (ECEF) system by considering the Earth phase of rotation in terms of $GMST$ only, i.e. neglecting precession, nutations, and polar motion.

$\boldsymbol{r}_{\boldsymbol{ECEF}} =$ W R N P $\boldsymbol{r}_{\boldsymbol{ECI}}$
here simplified to
$\boldsymbol{r}_{\boldsymbol{ECEF}} =$ R $\boldsymbol{r}_{\boldsymbol{ECI}}$
with
R = $R_3\big(GMST(t_j)\big)$

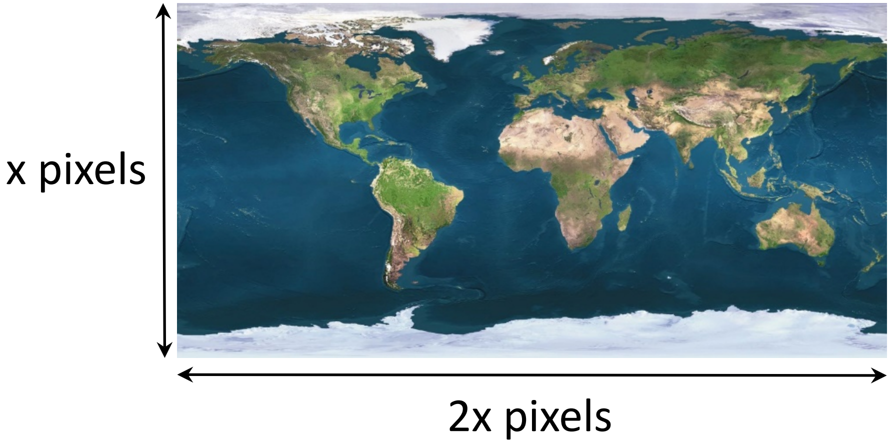
To compute $GMST(t_j)$ use your code from assignment#2 and the UT1 time starting at t (table), then increase t in step size 1 minute for 24h: $t_j = t + (j-1) \cdot 1\ [\mathrm{min}]$, where $j = 1, \ldots, 1440$. Make sure the time arguments have corresponding units.
Compute $GMST$ and make sure it is converted to the correct unit (depends on your function): `R = rot3d(GMST,3)`

Finally matrix multiply the 3x3-matrix R from the left with the 3-vector in ECI:
$\boldsymbol{r}_{\boldsymbol{ECEF}} =$ R $\boldsymbol{r}_{\boldsymbol{ECI}}$

Exercise 4: spherical Earth 3d-plot

Earth surface image projected on a sphere of radius $R = 6371$ km

Get an Earth image from the internet, e.g. in *.jpg format, horiz : vert #pixel has to be 2 : 1



Provide the spherical surface in Cartesian coordinates

```
[x,y,z] = sphere(180); x=6371.*x; y=6371.*y; z=6371.*z;
```

Plot the sphere with z-coordinate upside-down to match with the image

```
earth=surf(x,y,-z,'FaceColor','none','EdgeColor',[1 1 1])
```

Read in the image

```
cdata = imread('my_Earth_image.jpg')
```

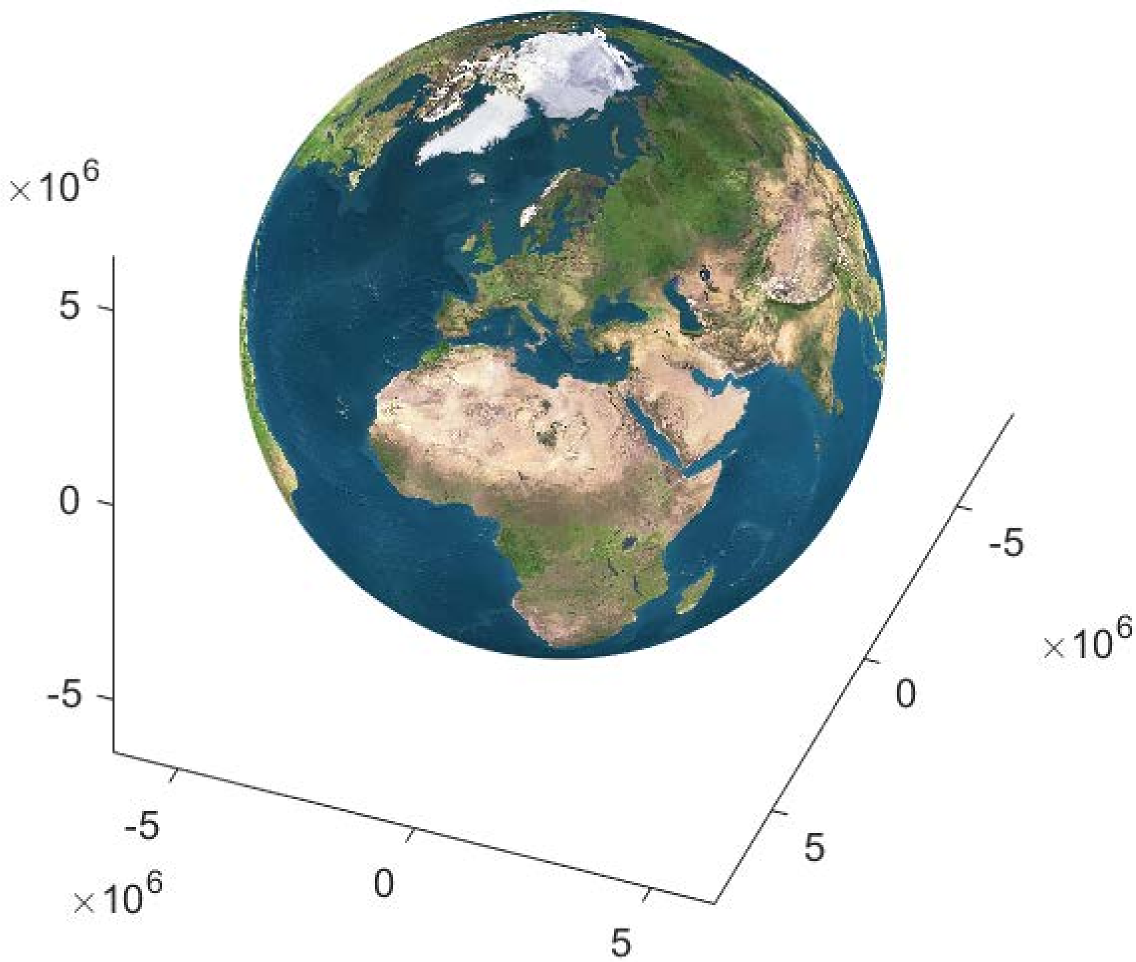
Display the image on the sphere

```
set(earth,'FaceColor','texturemap','CData',cdata,...
'FaceAlpha',1,'EdgeColor','none');
```

Exercise 4: Earth 3d-plot

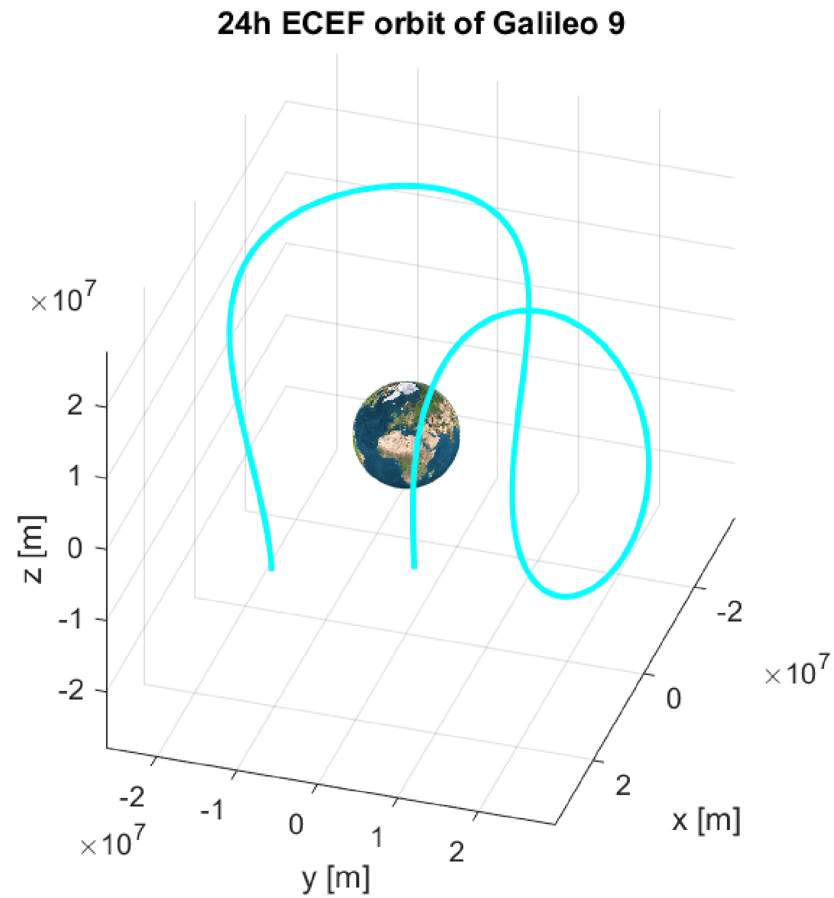
For a better look try

```
axis equal
axis auto
axis vis3d
```



Exercise 4: orbits in ECEF

then `plot3` the orbits after `hold on;`



This ECEF plot is to be drawn for each of the six satellite. The plot is just a qualitative example. Correct plots will look differently.

Exercise 5

Compute the ground tracks of the satellites and plot them on top of an Earth surface image or a coastline plot.

If you want to work with coastlines, use the file provided in ISIS. It is a matlab *.mat file that contains the coastlines in terms of lambda and phi pairs. Read it in with load ...

Exercise 5: ground tracks

Repeat exercise 4. Then continue with converting ECEF Cartesian to spherical coordinates, e.g.

```
[phi,lam,~]=xyz2plr(r_ECEF(1),r_ECEF(2),r_ECEF(3))
```

Example (with an image taken from internet)

Read in the image

```
cdata = imread('my_Earth_image.jpg')
```

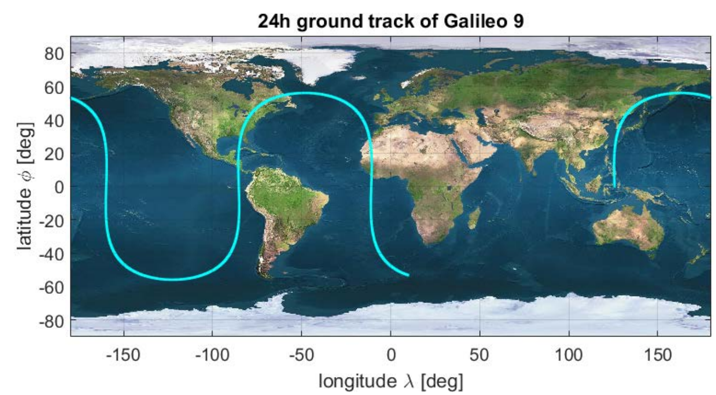
Assign the range and plot the image

```
imagesc([-180 180], [90 -90], cdata);
```

Then flip it in north (up)-south (down) direction

```
set(gca,'ydir','normal');
```

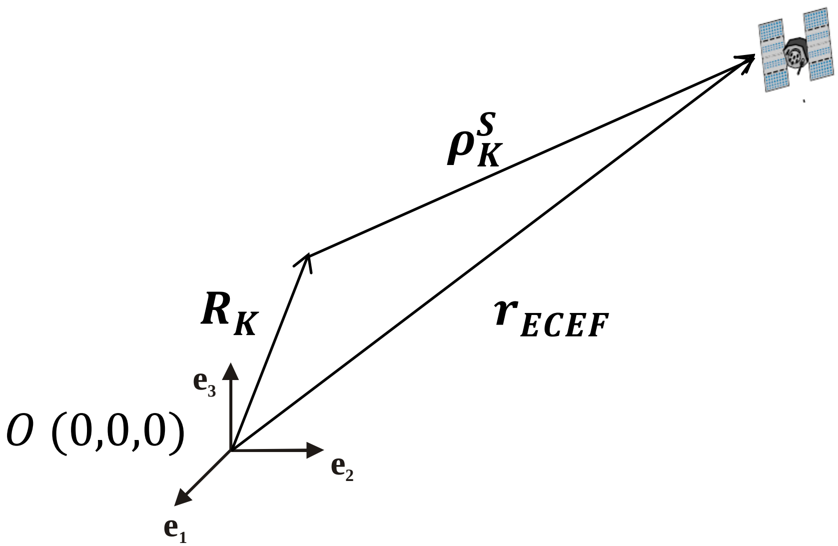
Now with `hold on;` you can continue plotting the orbits on top: `plot(lam,phi,‘.‘)`, where `lam` and `phi` must be in degree and the dot `‘.‘` is for omitting horizontal lines when `lam` jumps from longitude -180° to +180° or vice versa.



Exercise 6

Compute the minima of the 3d-distance (to m precision) and of the zenith distance (to arc min precision) to an observer near Berlin (attachment 2) and the corresponding epochs (minute precision) of these events. Which satellite approaches the observer with the smallest 3d-distance and which satellite comes closest to the local zenith of the observer? Why do some of the satellites not approach Berlin or Berlin's zenith considerably?

Exercise 6



Earth centered Earth fixed (ECEF) frame

Vector addition (commutative):

Vector:

$$\boldsymbol{r}_{ECEF} = \boldsymbol{R}_K + \boldsymbol{\rho}_K^S$$

$$\boldsymbol{\rho}_K^S = \boldsymbol{r}_{ECEF} - \boldsymbol{R}_K$$

Distance = magnitude:

$$\|\boldsymbol{\rho}_K^S\| = \|\boldsymbol{r}_{ECEF} - \boldsymbol{R}_K\|$$

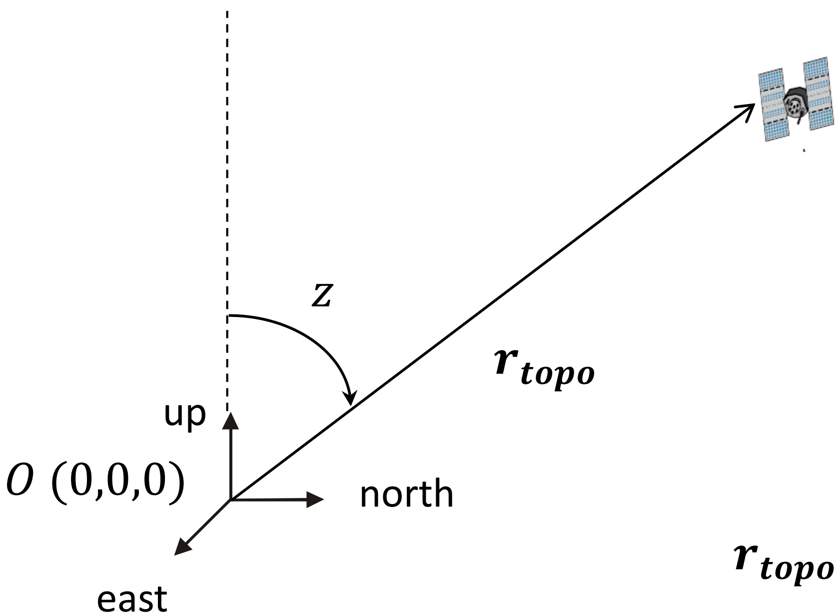
<u>Given/known:</u>

$\boldsymbol{r}_{ECEF}, \boldsymbol{R}_K$

<u>Unknown:</u>

$$\|\boldsymbol{\rho}_K^S\|$$

Exercise 6



Topocentric horizontal system

<u>Given/known:</u>

$\boldsymbol{r_{ECEF}}$

<u>Unknown:</u>

z

$$\boldsymbol{r_{topo}} = M_1 R_2\left(\frac{\pi}{2} - \phi\right) R_3(\Lambda) \boldsymbol{r_{ECEF}}$$

$$z = \frac{\pi}{2} - \arctan\left(r_{topo}(3) / \sqrt{r_{topo}(1)^2 + r_{topo}(2)^2}\right)$$

Exercise 6

Repeat exercise 4. Then continue with the transformation ECEF → local horizontal system

$\boldsymbol{r_{topo}} = M_1\, R_2\left(\frac{\pi}{2} - \phi\right) R_3(\Lambda)\, \boldsymbol{r_{ECEF}}$

where ϕ and Λ can be obtained from coordinate conversion of $\boldsymbol{R_{Ber}}$ attachment 2

```
[ϕ , Λ ,~]=xyz2plr(R_Ber(1),R_Ber(2),R_Ber(3))
```

then compute the zenith distance

$$z = \frac{\pi}{2} - \arctan\left(r_{topo}(3) / \sqrt{r_{topo}(1)^2 + r_{topo}(2)^2}\right)$$

for computation of minima MatLab provides in-built commands

```
min(...)
```

and to find an entry with a certain value try

```
find(...)
```

Round to the desired precision only for screen output at last. (Do not work with rounded quantities!)