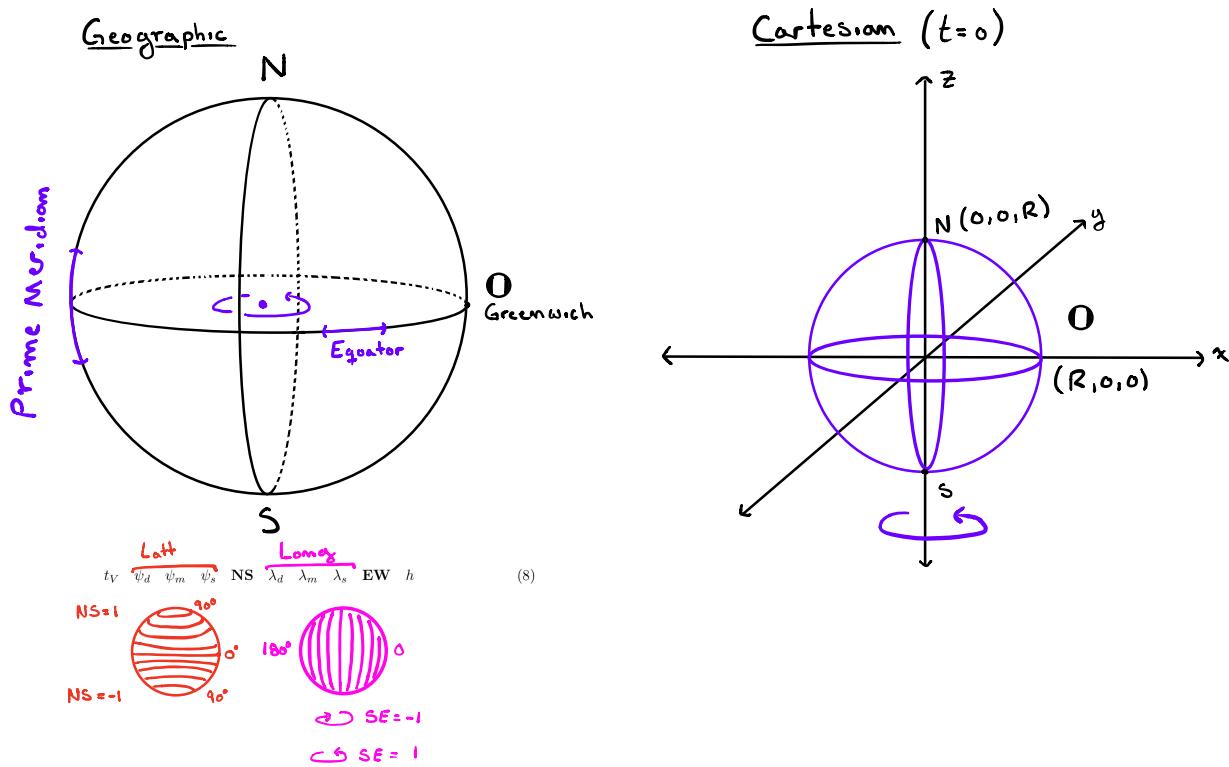


Model



t_V is a real number given to an accuracy of 10^{-2} seconds ranging from 0 to 10^6 . This is the time at which the vehicle is at the specified position.

ψ_d is an integer ranging from 0° (i.e., the Equator) to $+90^\circ$ (i.e., the North or South Pole).

ψ_m is an integer ranging from 0 to 59 minutes of degree.

ψ_s is a real number ranging from 0 to 59.9999 seconds of degree. It should be given to an accuracy of 10^{-2} (which corresponds to an accuracy of about a foot).

NS is an integer that is +1 North of the equator and -1 South of the equator.

λ_d is an integer ranging from 0° (i.e., the meridian of Greenwich) to 180° (i.e., 180 degrees east, or west, the date line).

λ_m is an integer ranging from 0 to 59 minutes of degree.

λ_s is a real number ranging from 0 to 59.9999 seconds of degree, given to the same accuracy as ψ_s .

EW is an integer that is +1 east of Greenwich and -1 west of Greenwich.

h is a real number giving the altitude in meters, to an accuracy of 1cm.

Exercise 1: Find a formula that describes the trajectory of the point **O** in Cartesian coordinates as a function of time.

$$\mathbf{O}_{\text{car}}(t) = \begin{bmatrix} R \cos(\frac{2\pi}{S} t) \\ R \sin(\frac{2\pi}{S} t) \\ 0 \end{bmatrix}$$

$$\theta = \omega \cdot t = \frac{2\pi}{S} \cdot t$$

Exercise 2: Write a program that converts angles from degrees, minutes, and seconds to radians, and vice versa. Make sure your program does what it's supposed to do.

For the following four exercises assume that t_V equals true Universal time and denote it by t .

* Convert Program

```
import math
pi = math.pi

def deg_to_rad(deg, min, sec):
    rad = (deg + min / 60 + sec / 3600) / 180 * pi
    return rad

#I need to find a way to seperate the whole number
def rad_to_deg(rad):
    dummy = rad*180/math.pi
    deg = int(dummy)          #Just took the whole number
    min = int((dummy-int(dummy))*60)      #Take the left over decimals of the oringnal number and mult by 60
    sec = int(((dummy - int(dummy))*60-min)*60)
    return [deg, min, sec]
```

We can change this of course.

Python math.* returns radians.

Exercise 3: Find a formula that converts position as given in (8) at time $t = 0$ into Cartesian coordinates.

$$\begin{array}{ccccccc} t_V & \psi_d & \psi_m & \psi_s & \text{NS} & \lambda_d & \lambda_m \\ = 0 & \in [0, \frac{\pi}{2}] & \in [-1, 1] & \in [0, \pi] & \in [-1, 1] & \in [0, \pi] & \in [-1, 1] \\ & & & & & & \\ & & & & & & \xrightarrow{=} h \end{array} \quad (8)$$

Need x (eqn 8), y (eqn 8), z (eqn 8)

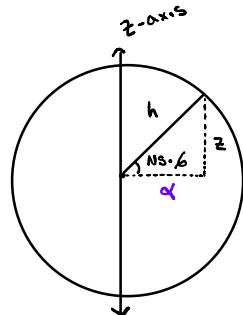
① $\psi_d, \psi_m, \psi_s \rightsquigarrow \beta \in [0, \frac{\pi}{2}]$ (convert to radians using * Convert program)

$\lambda_d, \lambda_m, \lambda_s \rightsquigarrow \theta \in [0, \pi]$ (convert to radians using * Convert program)

② z $\sin(\text{NS} \cdot \beta) = \frac{z}{h}$



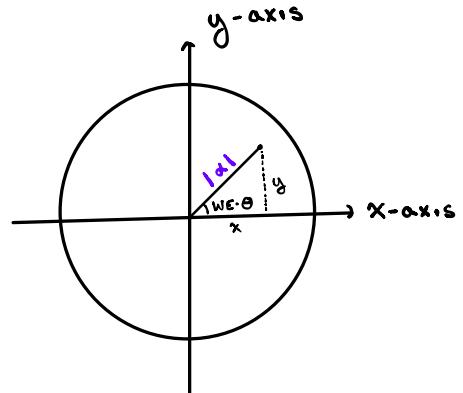
$z(\beta, \text{NS}, h) = (R + h) \sin(\text{NS} \cdot \beta)$



$\alpha(\beta, \text{NS}, h) = (R + h) \cos(\text{NS} \cdot \beta)$

$x \notin y$ we don't need this

$$\begin{aligned} x(\theta, \text{WE}, \alpha) &= |\alpha| \cdot \cos(\text{WE} \cdot \theta) \\ &= |(R+h) \cos(\text{NS} \cdot \beta)| \cos(\text{WE} \cdot \theta) \end{aligned}$$



$$\begin{aligned} y(\theta, \text{WE}, \alpha) &= |\alpha| \cdot \sin(\text{WE} \cdot \theta) \\ &= |(R+h) \cos(\text{NS} \cdot \beta)| \sin(\text{WE} \cdot \theta) \end{aligned}$$

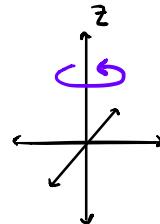
Thus Given we know $\beta \notin \theta$ (using program from exercise 2)
we have at $t=0$

$$\begin{aligned} \text{Eqn (8)} : = \vec{x}_0 &= \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} |\alpha| \cdot \cos(\text{WE} \cdot \theta) \\ |\alpha| \cdot \sin(\text{WE} \cdot \theta) \\ (R + h) \cdot \sin(\text{NS} \cdot \beta) \end{bmatrix} \end{aligned}$$

Exercise 4: Find a formula that converts position and general time t as given in (8) into Cartesian coordinates.

We just use our formula from exercise 3 to find the cartesian coordinates at $t=0$ & then use a rotation matrix to find position at time t .

Rotation Matrix: $R(t) = \underbrace{\begin{bmatrix} \cos\left(\frac{2\pi}{s}t\right) & -\sin\left(\frac{2\pi}{s}t\right) & 0 \\ \sin\left(\frac{2\pi}{s}t\right) & \cos\left(\frac{2\pi}{s}t\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{Rotates around } z\text{-axis}}$



$$\Rightarrow \vec{x}(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix} = R(t) \vec{x}_0$$

$$= \begin{bmatrix} \cos\left(\frac{2\pi}{s}t\right) & -\sin\left(\frac{2\pi}{s}t\right) & 0 \\ \sin\left(\frac{2\pi}{s}t\right) & \cos\left(\frac{2\pi}{s}t\right) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} |x| \cdot \cos(WE \cdot \theta) \\ |x| \cdot \sin(WE \cdot \theta) \\ h \cdot \sin(NS \cdot b) \end{bmatrix}$$

$$\vec{x}(t) = \begin{bmatrix} |x| \cos\left(\frac{2\pi}{s}t\right) \cos(WE \cdot \theta) - |x| \sin\left(\frac{2\pi}{s}t\right) \sin(WE \cdot \theta) \\ |x| \sin\left(\frac{2\pi}{s}t\right) \cos(WE \cdot \theta) + |x| \cos\left(\frac{2\pi}{s}t\right) \sin(WE \cdot \theta) \\ h \cdot \sin(NS \cdot b) \end{bmatrix}$$



Exercise 5: Find a formula that converts a position given in Cartesian coordinates at time $t = 0$ into a position of the form (8).

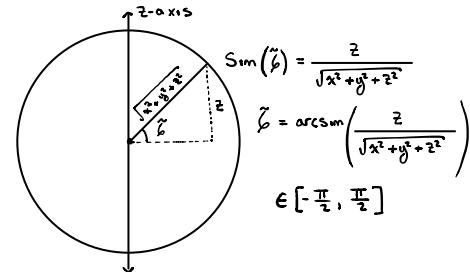
Cartesian \rightsquigarrow Geographical

Given $[x, y, z]^T$ need to find: $t_V \psi_d \psi_m \psi_s \text{ NS } \lambda_d \lambda_m \lambda_s \text{ EW } h$

• $t_V = 0$

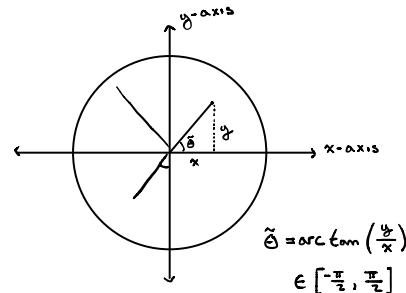
• $h = \sqrt{x^2 + y^2 + z^2} - R$

• $\text{NS} = \begin{cases} 1 & z \geq 0 \\ -1 & z < 0 \end{cases}$



• $\psi_d, \psi_m, \psi_s = \left| \arcsin\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right) \right| \xrightarrow[\text{* Convert program}]{\text{function}} [\psi_d, \psi_m, \psi_s]$

• $\text{WE} = \begin{cases} 1 & y \geq 0 \\ -1 & y < 0 \end{cases}$



• $\lambda_d, \lambda_m, \lambda_s = \begin{cases} \frac{\pi}{2} & x=0 \\ 0 & y=0 \wedge x>0 \\ \pi & y=0 \wedge x<0 \\ \arctan\left(\frac{y}{x}\right) & y>0 \wedge x>0 \\ |\arctan\left(\frac{y}{x}\right)| & y<0 \wedge x>0 \\ \pi + \arctan\left(\frac{y}{x}\right) & y>0 \wedge x<0 \\ \pi - \arctan\left(\frac{y}{x}\right) & y<0 \wedge x<0 \end{cases} \xrightarrow[\text{* Convert program}]{\text{function}} [\lambda_d, \lambda_m, \lambda_s]$

Not sure entirely how to deal w/ boundary cases.

Exercise 6: Find a formula that converts general time t and a position given in Cartesian coordinates into a position of the form (8).

Need to unrotate for t seconds

$$\bar{R}(t) = R(t)^T = \begin{bmatrix} \cos\left(\frac{2\pi}{5}t\right) & \sin\left(\frac{2\pi}{5}t\right) & 0 \\ -\sin\left(\frac{2\pi}{5}t\right) & \cos\left(\frac{2\pi}{5}t\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Given $\vec{x}(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix}$ we have $\vec{x}_o = \underline{R^T(t)x(t)}$

$$\vec{x}_o = \begin{bmatrix} \cos\left(\frac{2\pi}{5}t\right) & \sin\left(\frac{2\pi}{5}t\right) & 0 \\ -\sin\left(\frac{2\pi}{5}t\right) & \cos\left(\frac{2\pi}{5}t\right) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix}$$

$$\underline{\vec{x}_o} = \begin{bmatrix} x(t)\cos\left(\frac{2\pi}{5}t\right) + y(t)\sin\left(\frac{2\pi}{5}t\right) \\ -x(t)\sin\left(\frac{2\pi}{5}t\right) + y(t)\cos\left(\frac{2\pi}{5}t\right) \\ z(t) \end{bmatrix} = \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix}$$

Then we just apply formula from exercise 5 to get geographic coordinates.

Exercise 7: Find a formula that describes the trajectory of lamp post B12 in Cartesian coordinates as a function of time.

The `vehicle` should not produce impossible positions (like a latitude of $90^\circ 59' 59''$). For testing purposes we will use `vehicles` that produce a stream of data corresponding, for example, to a walk from JWB to the Marriott Library, a hike up to Mount Olympus, or a flight from Salt Lake City to the North Pole. The data sets should be spaced at least 1 second (of time) apart.

In addition `vehicle` writes a copy of the standard input and the standard output into the file

`vehicle.log`.