winter term 2021/22
heinkelmann@gfz-potsdam.de

Introduction to Satellite Geodesy - Exercise

Assignment 4: Reference systems and transformations

Exercises

1. Write code that computes the precession matrix based on the NEWCOMB precession angles $P = R_3(-z_A)\, R_2(\theta_A)\, R_3(-\zeta_A)$, where R_i denotes a 3D-rotation matrix. Evaluate your code by calculating the variation of the coordinates of a fictitious celestial object at the vernal equinox due to precession during the years 1990 to 2030 (annual resolution). Plot the coordinates vs. time.
2. Write code that computes the nutation matrix $N = R_1(-\varepsilon_A - \Delta\varepsilon) \cdot R_3(-\Delta\psi) \cdot R_1(\varepsilon_A)$. Evaluate your code by calculating the variation of the fictitious celestial object at the vernal equinox due to nutation during the years 1990 to 2030 (monthly resolution). Plot the coordinates vs. time.
3. Download the polar motion time series compatible with the IAU1980 theory[1] during 2021. Write a function that outputs the polar motion matrix $W = R_2(-x_p)R_1(-y_p)$ for a given time. Use linear interpolation between the diurnal polar motion values.
4. Given are the coordinates α and δ of four ICRF3 objects (attachment 1) and the astronomical coordinates of an observatory near Berlin ($\Lambda = 13°24'$, $\phi = 52°36'$) in the body-fixed system. Compute the Azimuth A and elevation E to the celestial objects in the local horizon system of the observatory during one day (2021-06-11, entire day with resolution of one minute). Use the classical transformation $x_{\mathrm{g}} = M_1\, R_2\left(\frac{\pi}{2} - \phi\right) R_3(\Lambda)\, W\, R_3(GMST)\, N\, P\, x_{i0}$, based on the equator of date, where M_i denotes a reflection matrix. $GAST$ shall be approximated with $GMST$. Use linear interpolation for polar motion, if indicated. Plot the apparent paths of the four celestial objects as time series of azimuth and of elevation and by a "*skyplot*" (attachment 2). In the skyplot, label the zenith, the North and East axes and the horizon and plot and label the celestial pole. Label the time series plots appropriately as well.

Attachment

1) Excerpt from ICRF3 catalog (please mind the different units and the sign of the coordinates):

Source name	**α [h min sec.dec.]**	**δ [deg min sec.dec.]**
0454+844	05 08 42.36351222	84 32 04.5441733
1101−536	11 03 52.22168463	- 53 57 00.6966389
1111+149	11 13 58.69508613	14 42 26.9526507
1738+499	17 39 27.39049431	49 55 03.3683385

[1] https://www.iers.org/IERS/EN/DataProducts/EarthOrientationData/eop.html

2) *MatLab* code drawing a "*skyplot*"

```
function hsky = skyplot(azim,elev,line_style)
if nargin == 2 , line_style = '*'; end
cax = newplot;
next = lower(get(cax,'NextPlot'));
hold_state = ishold;
tc = get(cax,'xcolor');
if ~hold_state
   hold on;
   zenmax = max(90-elev(:)); zenmax = 15*ceil(zenmax/15);
   elmax  = 90;
   az     = 0:pi/50:2*pi;
   xunit  = sin(az);
   yunit  = cos(az);
   for i=[30 60]
      plot(xunit*i,yunit*i,'-','color',tc,'linewidth',1);
   end
   i=90; plot(xunit*i,yunit*i,'-','color',tc,'linewidth',2);
   for i=[15:30:75 105:15:zenmax]
      plot(xunit*i,yunit*i,':','color',tc,'linewidth',1);
   end
   for i=30:30:zenmax
      text(0,i,['  ' num2str(90-i)],'verticalalignment','bottom');
   end
   az = (1:6)*2*pi/12; caz = cos(az); saz = sin(az);
   ca = [-caz; caz]; sa = [-saz; saz];
   plot(elmax*ca,elmax*sa,'-','color',tc,'linewidth',1);
   if zenmax > elmax
      plot(zenmax*ca,zenmax*sa,':','color',tc,'linewidth',1);
   end
   rt = 1.1*elmax;
   for i = 1:length(az)
      loc1 = int2str(i*30);
      if i == length(az)
         loc2 = int2str(0);
      else
         loc2 = int2str(180+i*30);
      end
      text( rt*saz(i), rt*caz(i),loc1,'horizontalalignment','center');
      text(-rt*saz(i),-rt*caz(i),loc2,'horizontalalignment','center');
   end
   view(0,90);
   axis(max(zenmax,elmax)*[-1 1 -1.1 1.1]);
   set(cax,'position',[.05 .05 .9 .9])
end
yy = (90-elev).*cos(azim/180*pi);
xx = (90-elev).*sin(azim/180*pi);
q = plot(xx,yy,line_style);
if nargout > 0, hsky = q; end
if ~hold_state, axis('equal'); axis('off'); end
if ~hold_state, set(cax,'NextPlot',next); end
```

Due date

January 17, 2022