
```

%*****
% function hpol = plotAzEl(az,el,svs,varargin)
%
% DESCRIPTION:
%
%   Creates an az-el plot of satellites
%
% ARGUMENTS:
%
%   az - vector of azimuth angles, in degrees
%   el - vector of elevation angles, in degrees
%   svs - vector of satellite PRN numbers
%   NOTE: To avoid printing PRN numbers on the plot, make 'svs' a vector
%         of zeros.
%   varargin - axes handle for plot on previous
%
% OUTPUT:
%
%   hpol - handle to polar plot axes
%
% CALLED BY:
%
%   createAzElMap
%
% FUNCTIONS CALLED:
%
%   None
%
% MODIFICATIONS:
%
%           ?? : P. Axelrad - Original
%   02-05-02 : Lisa Reeh
%   05-17-04 : Stephen Russell - minor modifications to allow plot
%                   overlaying using new code (i.e. varargin with axes handle)
%
%
% Colorado Center for Astrodynamics Research
% Copyright 2004 University of Colorado, Boulder
%*****
function hpol = plotAzEl(az,el,svs,varargin)

line_style = 'auto';

if nargin < 1
    error('Requires 3 input arguments.')
end

if isstr(az) | isstr(el)
    error('Input arguments must be numeric.');
```

```

end

% get hold state
if(nargin > 3)
    axes(varargin{1});
end
cax = newplot;
next = lower(get(cax, 'NextPlot'));
hold_state = ishold;

% get x-axis text color so grid is in same color
tc = get(cax, 'xcolor');

% Hold on to current Text defaults, reset them to the
% Axes' font attributes so tick marks use them.
fAngle = get(cax, 'DefaultTextFontAngle');
fName = get(cax, 'DefaultTextFontName');
fSize = get(cax, 'DefaultTextFontSize');
fWeight = get(cax, 'DefaultTextFontWeight');
set(cax, 'DefaultTextFontAngle', get(cax, 'FontAngle'), ...
    'DefaultTextFontName', get(cax, 'FontName'), ...
    'DefaultTextFontSize', get(cax, 'FontSize'), ...
    'DefaultTextFontWeight', get(cax, 'FontWeight') )

% only do grids if hold is off
if ~hold_state

    % make a radial grid
    hold on;
    hhh=plot([0 2*pi],[0 90], '-','linewidth',0.5);
    v = [get(cax, 'xlim') get(cax, 'ylim')];
    ticks = length(get(cax, 'ytick'));
    delete(hhh);

    % check radial limits and ticks
    rmin = 0; rmax = v(4); rticks = ticks-1;

    if rticks > 5 % see if we can reduce the number
        if rem(rticks,2) == 0
            rticks = rticks/2;
        elseif rem(rticks,3) == 0
            rticks = rticks/3;
        end
    end

    % define a circle
    th = 0:pi/50:2*pi;
    xunit = cos(th);
    yunit = sin(th);

    % now really force points on x/y axes to lie on them exactly
    inds = [1:(length(th)-1)/4:length(th)];
    xunits(inds(2:2:4)) = zeros(2,1);
    yunits(inds(1:2:5)) = zeros(3,1);

```

```

rinc = (rmax-rmin)/rticks;
for i=(rmin+rinc):rinc:rmax
    plot(yunit*i,xunit*i,'-', 'color',tc, 'linewidth',0.5);
    text(0,i+rinc/20,[' ' num2str(90-i)], 'verticalalignment', 'bottom' );
end

    % plot spokes
th = (1:6)*2*pi/12;
cst = cos(th); snt = sin(th);
cs = [cst; -cst];
sn = [snt; -snt];
plot(rmax*sn,rmax*cs,'-', 'color',tc, 'linewidth',0.5);

    % annotate spokes in degrees
rt = 1.1*rmax;
for i = 1:max(size(th))
    text(rt*snt(i),rt*cst(i),int2str(i*30), 'horizontalalignment', 'center' );
    if i == max(size(th))
        loc = int2str(0);
    else
        loc = int2str(180+i*30);
    end
    text(-rt*snt(i),-rt*cst(i),loc, 'horizontalalignment', 'center' );
end

    % set viewto 2-D
view(0,90);
    % set axis limits
axis(rmax*[-1 1 -1.1 1.1]);
end

% Reset defaults.
set(cax, 'DefaultTextFontAngle', fAngle , ...
    'DefaultTextFontName', fName , ...
    'DefaultTextFontSize', fSize, ...
    'DefaultTextFontWeight', fWeight );

set(gcf, 'color', 'white');

% transform data to Cartesian coordinates.
yy = (90-el).*cos(az*pi/180);
xx = (90-el).*sin(az*pi/180);

% plot data on top of grid
q = plot(xx,yy,'ok', 'MarkerSize',2);

% Place satellite PRN numbers with satellite position
for i = 1:length(svs)
    if(svs(i)~=0)
        text(xx(i)+3,yy(i),int2str(svs(i)));
    end
end
end

```

```
if nargout > 0
    eval(['hpol = gca;']);
end

if ~hold_state
    axis('equal');axis('off');
end

% set hold state
if ~hold_state
    hold on;
end
```

Published with MATLAB® R2013b