(* Notebook to model IBEX repointing with simple model *)

(* INPUTS:

uFrom	unit vector pointing along spin axis at start of maneuver
uTgt	unit vector pointing where ADC is commanded to drive to
spinRate	(rad/s) spin rate of S/C
ctlCycle	(s) delay from true to pulse-time-0.5 s nominally
pulseSz	(rad) angle spin axis is moved by a pulse
deadBand	(rad) angle between uNow and uTgt at which ADC halts maneuver

*)

```
vecFrom = {0.383699, 0.842720, 0.377623};
vecTgt = {0.367310, 0.634537, 0.680034};

spinRate = 4 * 360. / 60 Degree (* per second *) ;
ctlCycle = 1.0(* seconds *);
pulseSz = 0.13 Degree;
deadBand = 0.8 Degree;
```

(* VARIABLES USED IN CALCULATIONS:

uNow	unit vector pointing in direction of current spin axis
vecDirect	(non-unit) vector pointing from uNow to uTgt
uStep	unit vector pointing in direction next pulse will take spin axis
vecStep	direction of uStep,magnitude of pulseSz
uSide	unit vector pointing in direction of side-step
delayang	(rad) angle by which pulse is delayed
raDecList	List of RA, Dec pairs to plot

*)

(* FUNCTIONS:
unitVec[vector]=unit vector pointing in same direction as vector
raDec[3-vector]=Right Ascension, Declination where vector is pointing. SINGULARITY at Poles.
*)

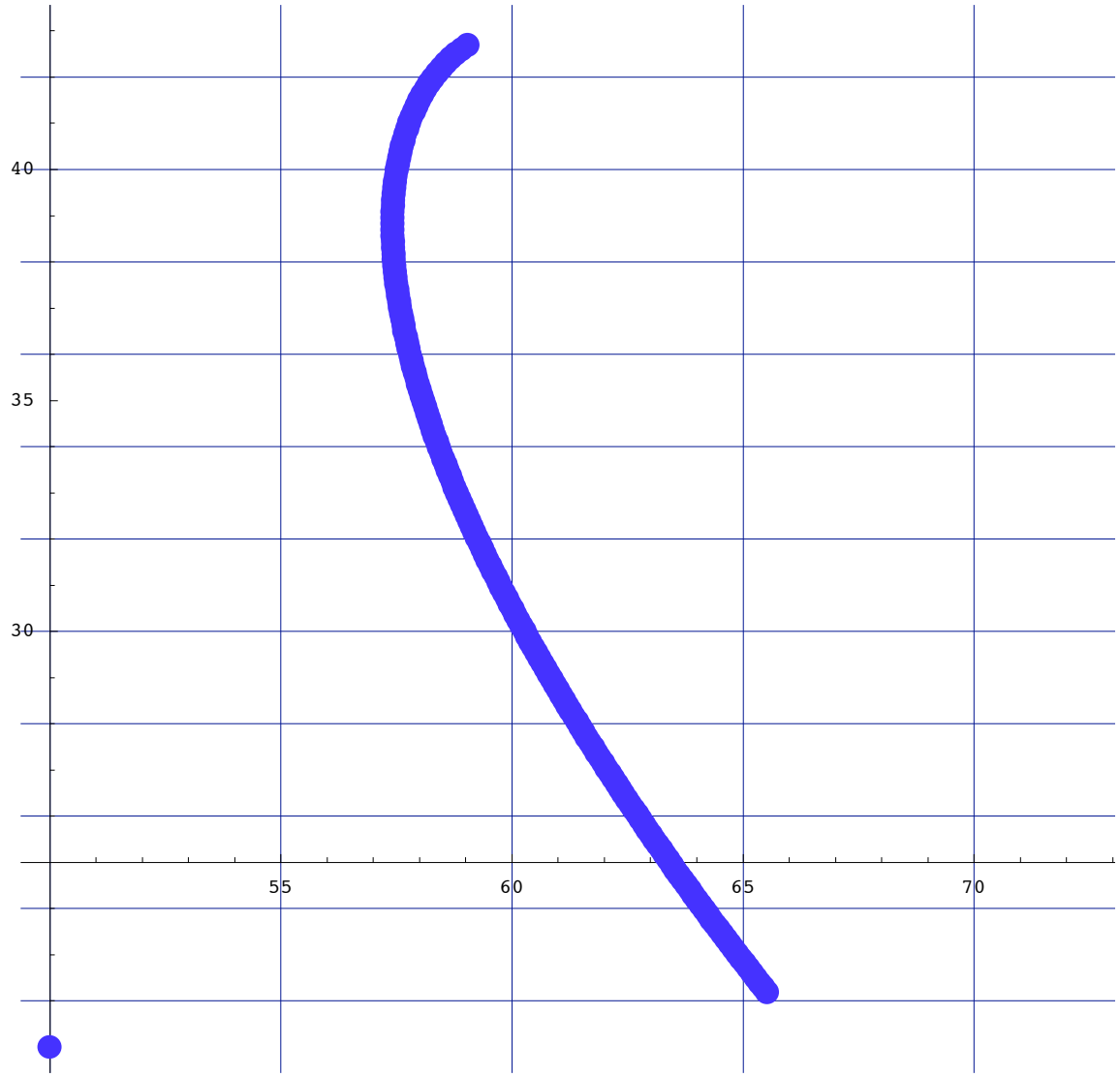
```
unitVec[vec_] := vec / Norm[vec];
raDec[{x_, y_, z_}] := {ArcTan[x, y], ArcTan[z / Sqrt[x^2 + y^2]]}
```

(* SETUP CALCULATIONS *)

```
uTgt = unitVec[vecTgt]  (* unit vector *);
uNow = unitVec[vecFrom]  (* unit vector *);
vecDirect = uTgt - uNow  (* vector *);
delayAng = spinRate * ctlCycle (* radians *);
raDecList = {{50., 21.} Degree, {75., 43.} Degree, raDec[vecFrom] }
 (* radians - List of RA, Dec pairs *);
```

```
(* LOOP TO GENERATE SUCCESSIVE POINTING VECTORS *)

While[
 Norm[vecDirect] > deadBand   (* vectors on unit sphere ~ radians *)
 ,
 vecDirect = uTgt - uNow;
 uSide = unitVec[ Cross [uNow , uTgt] ];
 uDirect = unitVec[vecDirect];

 uStep = uDirect * Cos[delayAng] + uSide * Sin[delayAng];
 vecStep = uStep * pulseSz;

 uNow = unitVec[uNow + vecStep];

 radecNow = raDec[uNow];

 raDecList = Join [raDecList, {radecNow}];
]
```

```
(* Plot out list of RA, Dec pairs *)

ListPlot[raDecList / Degree,
  GridLines → {{50, 55, 60, 65, 70, 75}, {22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42}},
  AspectRatio → Automatic,
  PlotStyle → {PointSize[.02], Hue[0.7]}
 ];
```



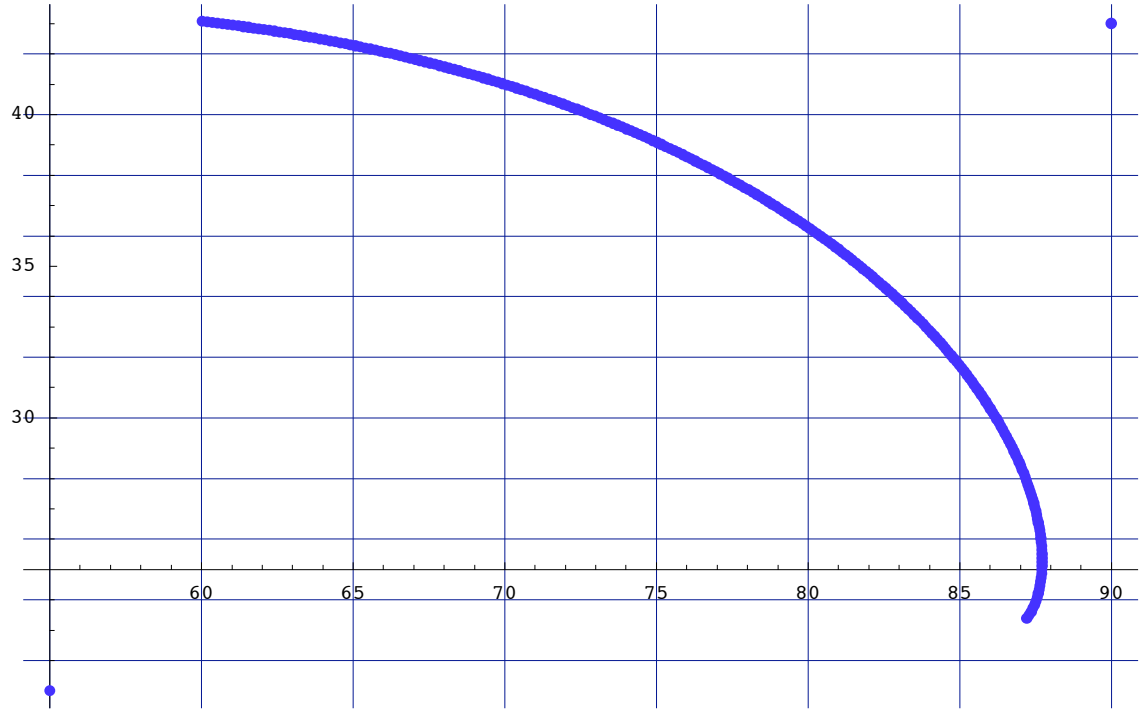

```
(* Calculations to choose biased repoint vector for June 20 *)

(* Where we are now, during Delta-V burns *)
vecFrom = {0.3650257, 0.6328483, 0.6828318};

(* Where we want to be *)
vecDesired = {0.044536, 0.916826, 0.396795};
```

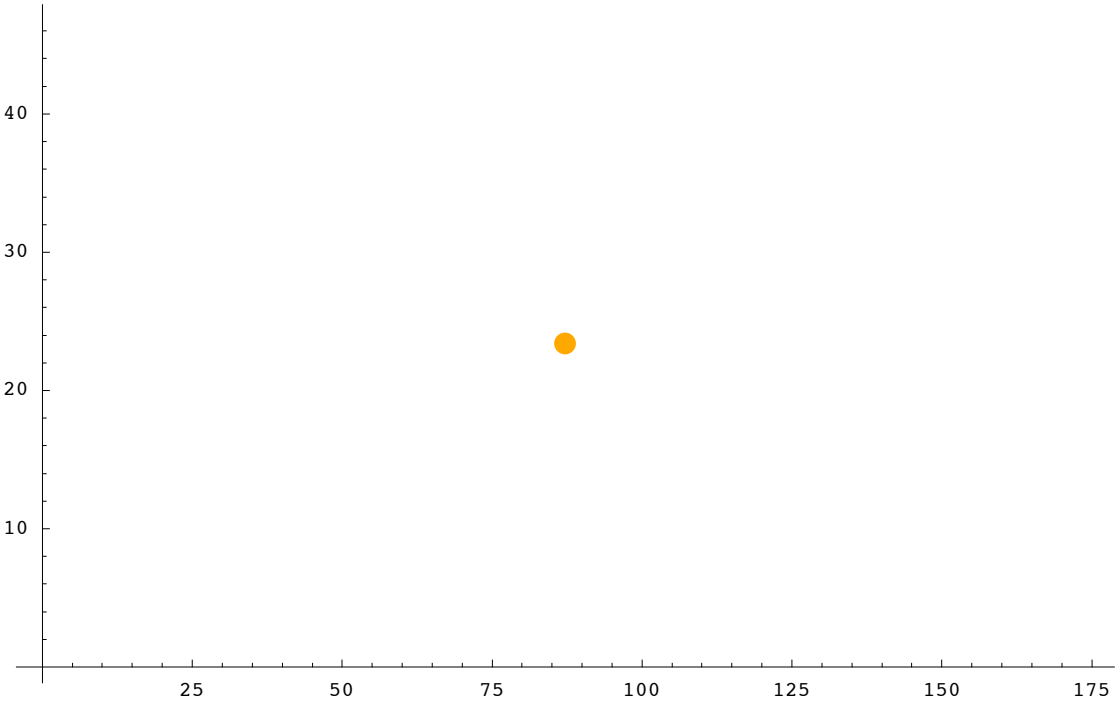
```
(* Where we tell ADC to go to get us to vecDesired *)
raDecTgt = {(86.6211699309974) Degree, (23.076131027330046) Degree}
vecTgt = unitVec[ {Cos[raDecTgt[[1]] ] * Cos[raDecTgt[[2]] ],
   Sin[raDecTgt[[1]] ] * Cos [raDecTgt[[2]] ], Sin[raDecTgt[[2]] ]}
 ]
```

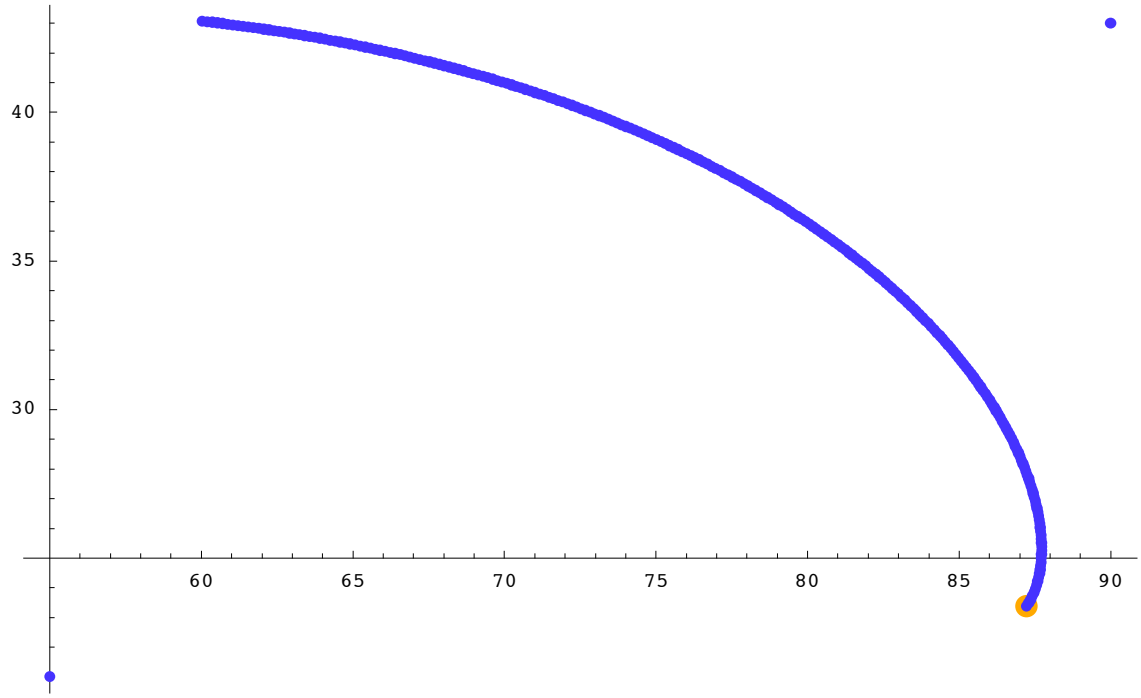
```
{1.51182, 0.402754}
```

```
{0.0542216, 0.918386, 0.391954}
```

```
(* ADC parameters, all same as above *)
spinRate = 4 * 360. / 60 Degree (* per second *) ;
ctlCycle = 1.0(* seconds *);
pulseSz = 0.13 Degree;
deadBand = 0.8 Degree;

uTgt = unitVec[vecTgt]  (* unit vector *);
uNow = unitVec[vecFrom]  (* unit vector *);
vecDirect = uTgt - uNow  (* vector *);
delayAng = spinRate * ctlCycle (* radians *);
raDecList = {{55., 21.} Degree, {90., 43.} Degree, raDec[vecFrom] }
 (* radians - List of RA, Dec pairs *);


(* LOOP TO GENERATE SUCCESSIVE POINTING VECTORS *)

While[
 Norm[vecDirect] > deadBand   (* vectors on unit sphere ~ radians *)
 ,
 vecDirect = uTgt - uNow;
 uSide = unitVec[ Cross [uNow , uTgt] ];
 uDirect = unitVec[vecDirect];

 uStep = uDirect * Cos[delayAng] + uSide * Sin[delayAng];
 vecStep = uStep * pulseSz;

 uNow = unitVec[uNow + vecStep];

 radecNow = raDec[uNow];

 raDecList = Join [raDecList, {radecNow}];
]


(* Plot out list of RA, Dec pairs *)

spinAxPlt = ListPlot[raDecList / Degree,
   GridLines → {{50, 55, 60, 65, 70, 75, 80, 85, 90, 95},
     {22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42}},
   AspectRatio → Automatic,
   PlotStyle → {PointSize[.01], Hue[0.7]}
  ];
desiredDot = ListPlot[ {raDec[vecDesired] / Degree },
   PlotStyle → {PointSize[.02], Hue[0.1]}
  ];
Show[desiredDot, spinAxPlt]
```


40
35
30
60
65
70
75
80
85
90


40
30
20
10
25
50
75
100
125
150
175



- Graphics -

```
error = raDec[vecDesired] / Degree - Last[raDecList] / Degree
```

```
{0.00034539, 0.000399199}
```