

## Coverage Characteristics of Earth Satellites

This document describes two MATLAB scripts that can be used to determine coverage characteristics of single satellites, and Walker and user-defined satellite constellations. A third script is also included that determines the view periods of satellites in circular orbits.

### **coverage1.m – geometry of satellite coverage**

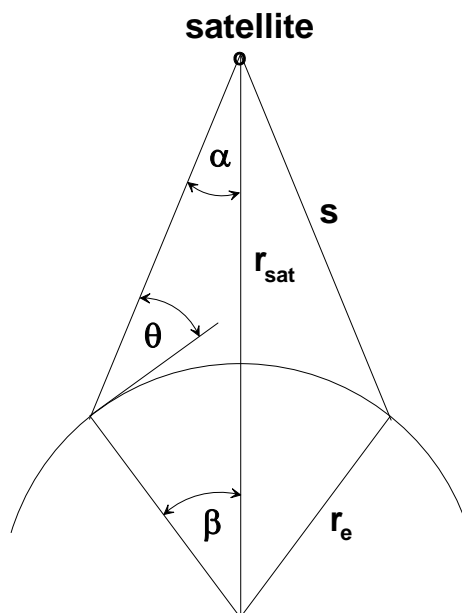
This MATLAB application calculates a variety of coverage information for satellites in circular and elliptic Earth orbits. The user can elect to compute coverage characteristics from one of the following satellite orbital positions:

- perigee
- apogee
- extreme northern latitude
- extreme southern latitude
- user-defined true anomaly
- user-defined satellite latitude

The user can also select one of the following coverage constraints and perform calculations for one or two values:

- nadir angle
- Earth central angle
- elevation angle
- slant range

The following diagram illustrates the coverage geometry of a satellite in orbit above a spherical planet with a nadir-pointing conical sensor or field-of-view:



## Orbital Mechanics with MATLAB

where

$\alpha$  = nadir angle

$\beta$  = central angle

$\theta$  = elevation angle

$s$  = slant range

$r_{sat}$  = geocentric radius of satellite =  $r_e + h_{sat}$

$h_{sat}$  = altitude of satellite

$r_e$  = radius of the Earth

The fundamental relationship between these angles and distances is given by

$$\theta + \alpha + \beta = 90^\circ$$

$$s \cos \theta = r_{sat} \sin \beta$$

$$s \sin \alpha = r_e \sin \beta$$

Furthermore, the nadir angle from the satellite's location to the Earth horizon is

$$\alpha_h = \sin^{-1} \left( \frac{r_e}{r_{sat}} \right)$$

The equations that define these angles as a function of slant range  $s$  are as follows:

$$\alpha = \cos^{-1} \left( \frac{r_{sat}^2 - r_e^2}{2sr_{sat}} + \frac{s}{2r_{sat}} \right)$$

$$\beta = \cos^{-1} \left( \frac{r_{sat}^2 + r_e^2 - s^2}{2r_{sat}r_e} \right)$$

$$\theta = \sin^{-1} \left( \frac{r_{sat}^2 - r_e^2}{2sr_e} - \frac{s}{2r_e} \right)$$

The relationships between the nadir angle, central angle and the slant range as functions of the elevation angle  $\theta$  are defined by the following expressions:

$$\alpha = \sin^{-1} \left( \frac{r_e}{r_{sat}} \cos \theta \right)$$

## Orbital Mechanics with MATLAB

$$\beta = \cos^{-1} \left( \frac{r_e}{r_{sat}} \cos \theta \right) - \theta$$

$$s = \sqrt{r_{sat}^2 - r_e^2 \cos^2 \theta} - r_e \sin \theta$$

The central angle, elevation angle and slant range as functions of the nadir angle  $\alpha$  can be determined from

$$\beta = \sin^{-1} \left( \frac{r_{sat}}{r_e} \sin \alpha \right) - \alpha$$

$$\theta = \cos^{-1} \left( \frac{r_{sat}}{r_e} \sin \alpha \right)$$

$$s = r_{sat} \cos \alpha - \sqrt{r_e^2 - r_{sat}^2 \sin^2 \alpha}$$

The relationship between the nadir angle, elevation angle and slant range and the central angle  $\beta$  is given by the following expressions:

$$\alpha = \tan^{-1} \left( \frac{r_e \sin \beta}{r_{sat} - r_{sat} \cos \beta} \right)$$

$$\theta = \tan^{-1} \left( \frac{r_{sat} \cos \beta - r_e}{r_{sat} \sin \beta} \right)$$

$$s = \sqrt{r_{sat}^2 + r_e^2 - 2r_{sat} r_e \cos \beta}$$

The width of the swath is given by

$$w = 2r_e \beta$$

The percentage of the Earth's surface viewed by this configuration is  $50(1 - \cos \beta)$  and the coverage surface area is given by

$$A = 2\pi r_e^2 (1 - \cos \beta)$$

The following is a typical user interaction with this program.

```
program coverage1
coverage characteristics of Earth satellites
please input the semimajor axis (kilometers)
(semimajor axis > 0)
? 8000
```

## *Orbital Mechanics with MATLAB*

```
please input the orbital eccentricity (non-dimensional)
(0 <= eccentricity < 1)
? 0

please input the orbital inclination (degrees)
(0 <= inclination <= 180)
? 28.5

please select the satellite's position

<1> perigee
<2> apogee
<3> extreme northern latitude
<4> extreme southern latitude
<5> user-defined true anomaly
<6> user-defined satellite latitude

selection (1, 2, 3, 4, 5 or 6)
? 3

please select the coverage constraint

<1> nadir angle
<2> earth central angle
<3> elevation angle
<4> slant range

selection (1, 2, 3 or 4)
? 3

please input the number of coverage constraints (1 or 2)
? 1

coverage constraint number 1

please input the elevation angle (degrees)
? 5
```

The following is the program output for this example.

```
satellite altitude      1626.7427 kilometers
true anomaly            90.0000 degrees
slant range             4305.0081 kilometers
```

## Orbital Mechanics with MATLAB

```
nadir angle                52.5829 degrees
earth central angle        32.4171 degrees
elevation angle            5.0000 degrees

earth coverage area        39831241.9936 square kilometers
earth coverage area        7.7916 percent

arc distance                3608.6532 kilometers

view latitude  1            -3.9171 degrees
view latitude  2            60.9171 degrees
```

### **coverage2.m – coverage characteristics of satellite constellations**

This MATLAB application can be used to assess the partial coverage performance of Walker and user-defined satellite constellations. The user can specify the total simulation time and a single geographic target, and the software will determine such coverage statistics as minimum, average, and maximum coverage time, etc. The user can also specify a minimum elevation angle constraint or “mask” at the target. During the simulation satellite orbits are propagated using Kozai’s method and the Earth is modeled as an oblate spheroid.

The following is a typical data file (`constell.dat`) that simulates a user-defined constellation and ground site. This constellation consists of six satellites.

```
number of satellites
6

initial orbital elements - satellite #1
6865.85585
0.0
39.0
0.0
348.17
0.0

initial orbital elements - satellite #2
6865.85585
0.0
39.0
0.0
22.83
201.21

initial orbital elements - satellite #3
6865.85585
0.0
39.0
0.0
57.5
42.42

initial orbital elements - satellite #4
```

## Orbital Mechanics with MATLAB

```
6865.85585
0.0
39.0
0.0
168.57
180.0

initial orbital elements - satellite #5
6865.85585
0.0
39.0
0.0
203.23
21.21

initial orbital elements - satellite #6
6865.85585
0.0
39.0
0.0
237.9
222.42

ground site latitude (degrees)
30.0

ground site longitude (degrees)
0.0

ground site altitude (meters)
0.0

minimum elevation angle constraint (degrees)
5.0

simulation duration (days)
1.0
```

The following is a typical Walker constellation data file (walker1.dat). Notice the use of the T/P/F (T is the total number of satellites, P is the total number of orbit planes, F is the phasing unit) method to define the constellation.

```
Walker T/P/F configuration
7,7,4

constellation semimajor axis (kilometers)
6865.222

constellation inclination (degrees)
38.0

ground site latitude (degrees)
30.0

ground site longitude (degrees)
240.0
```

## Orbital Mechanics with MATLAB

```
ground site altitude (meters)
100.0

minimum elevation angle constraint (degrees)
5.0

simulation duration (days)
1.0
```

The following is a typical user interaction and program output created with this application. It illustrates the performance of a Walker 7/7/4 constellation relative to a ground site at 30° north latitude and 240° east longitude. For Walker constellations the software determines the east longitude of the ascending node and mean anomaly of each satellite.

```
program coverage2

analysis of partial coverage constellations

<1> evaluate Walker constellation
<2> evaluate user-defined constellation

selection (1 or 2)
? 1

please input the name of the Walker data file
(be sure to include the filename extension)
? walker1.dat

constellation mean orbital elements

semimajor axis          6865.2220 kilometers
inclination             38.0000 degrees

satellite               east longitude of          mean anomaly
number                 ascending node (deg)        (degrees)
  1                     0.0000                      0.0000
  2                     51.4286                     205.7143
  3                     102.8571                    51.4286
  4                     154.2857                    257.1429
  5                     205.7143                    102.8571
  6                     257.1429                    308.5714
  7                     308.5714                    154.2857

coverage statistics

target latitude         30.0000 degrees
target east longitude   240.0000 degrees
target altitude        100.0000 meters

minimum elevation angle 5.0000 degrees
```

## Orbital Mechanics with MATLAB

total number of accesses	45	
minimum coverage time	3.297437	minutes
average coverage time	8.487978	minutes
maximum coverage time	9.586444	minutes
total coverage time	381.959005	minutes
total number of gaps	46	
minimum gap time	1.563970	minutes
average gap time	23.000891	minutes
maximum gap time	29.841843	minutes
total gap time	1058.040995	minutes
total simulation time	1.000000	days

### Long-Term Prediction of View Periods

This section describes an algorithm and MATLAB script called `vperiod` that can be used to determine the long-term view periods between a satellite in a circular Earth orbit and a ground site on a spherical Earth. The mathematics of this algorithm are described in “The Long-Term Forecast of Station View Periods”, by Martin W. Lo, JPL TDA Progress Report 42-118, August 15, 1994. This report also contains an error analysis comparing this closed-form solution and the classical method of perturbed orbit propagation.

This technique is valid for orbits and ground sites that satisfy the following conditions:

- Non-equatorial, circular Earth orbits
- Orbits perturbed by the secular effect of  $J_2$  only
- Non-repeating ground track
- Ground stations not at the north or south pole

The *view period ratio* is defined by the following expression:

$$\rho = \lim_{T \rightarrow \infty} \frac{P(T)}{T}$$

where  $T$  is the total elapsed or simulation time and  $P(T)$  is the total station view period of a satellite during the simulation time  $T$ .

The view period function is given by the following expression:

## Orbital Mechanics with MATLAB

$$f(\phi) = \frac{\cos \phi \cos^{-1} \left( \frac{\cos \theta - \sin \phi \sin \phi_s}{\cos \phi_s \cos \phi} \right)}{\pi^2 \sqrt{\sin^2 i - \sin^2 \phi}}$$

where  $\phi_s$  is the latitude of the ground site and  $\theta$  is the Earth central angle. The Earth central angle is a function of the satellite's altitude  $h_{sat}$  and the minimum elevation angle  $e_{min}$  as follows:

$$\theta = \cos^{-1} \left( \frac{r_e}{r_{sat}} \cos e_{min} \right) - e_{min}$$

where  $r_e$  is the radius of the Earth and  $r_{sat} = r_e + h_{sat}$ .

The maximum ground site latitude which can be viewed by this satellite is equal to the sum of the orbital inclination  $i$  and the Earth central angle  $\theta$ .

The view period ratio for any ground site is calculated using the following definite integral:

$$\rho = \int_{\phi_1}^{\phi_2} f(\phi) d\phi$$

where the finite limits of integration are given by

$$\phi_1 = \max(\phi_s - \theta, -L_i)$$

$$\phi_2 = \min(\phi_s + \theta, L_i)$$

These limits depend upon the orbital inclination  $i$ , the ground site latitude of interest  $\phi_s$ , the Earth central angle  $\theta$  and the value of  $L_i$  which is given by

$$L_i = \begin{cases} i & \text{if } i \leq 90 \text{ degrees} \\ 180 - i & \text{if } i > 90 \text{ degrees} \end{cases}$$

From the view period ratio, we can determine the total view period during any time interval of interest using

$$\rho T = \text{total view period during elapsed time } T$$

Furthermore,

$$\rho 24 \text{ hours} = \text{average daily total view period}$$

$$\rho 7 \text{ days} = \text{average weekly total view period}$$

Another application is the maximum average contact time possible with the  $K$  ground stations and a single satellite during a 24 hour interval given by:

## Orbital Mechanics with MATLAB

$$C = (\rho_1 + \rho_2 + \dots + \rho_k) \text{ 24 hours}$$

The extension of this technique to constellations of satellites is straight-forward.

### Numerical Solution

The view period integral is solved numerically using a Gauss-Kronrod quadrature algorithm due to T. Patterson with enhancements by F. Krogh and W. Snyder. The syntax of this MATLAB function is as follows:

```
function [result, k, icheck] = kquad(usrfunc, a, b, tol)

% one-dimensional quadrature

% method due to T. Patterson with modifications
% by F. Krogh and W. Snyder (ACM Algorithm 699)

% input

% usrfunc = name of user-defined function
% a       = lower integration limit
% b       = upper integration limit
% tol     = convergence tolerance

% output

% result = array of solution(s)
% k      = index of result array which best satisfies tol
% icheck = 0 ==> convergence
%        = 1 ==> non-convergence
```

The following is a typical user interaction with this script.

```
program vperiod

< long-term prediction of view periods >

please input the altitude (kilometers)
? 200

please input the orbital inclination (degrees)
(0 <= inclination <= 180)
? 28.5

please input the minimum elevation angle (degrees)
(abs(elevation) >= 0 degrees)
? 0

please select the type of calculation
```

## Orbital Mechanics with MATLAB

```
<1> single ground site  
<2> range of ground sites
```

```
? 2
```

The script will ask the user if he or she would like to create a plot of the view period ratio as a function of the ground site latitude with the following request:

```
would you like to create and display graphics (y = yes, n = no)  
? y
```

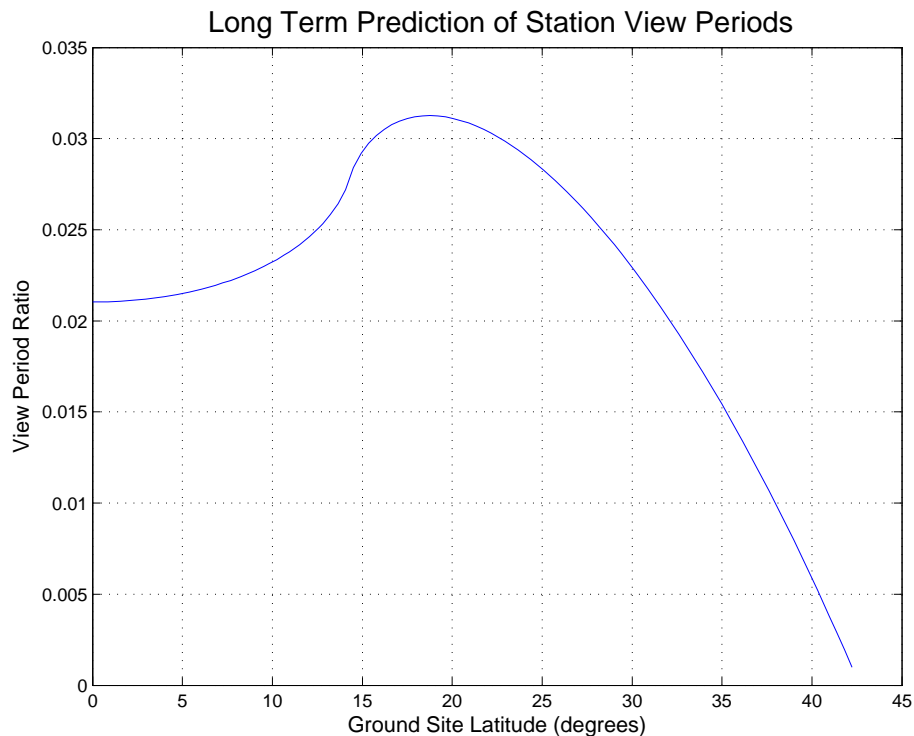
The user can also create a data file of these conditions by responding with *y* for yes to the following prompt:

```
would you like to create a data file (y = yes, n = no)  
? y
```

Finally, the user can specify the number of data points in both the graphics and data file via the following response:

```
please input the number of data points to use  
? 100
```

The following is the companion graphics display for this example.



## *Orbital Mechanics with MATLAB*

This plot is symmetric about the equator.

The following are the first few lines of the output data file. Column 1 is the ground site latitude in degrees, and column 2 is the view period ratio which is non-dimensional.

0.0000	0.02102956
0.4266	0.02103287
0.8533	0.02104284
1.2799	0.02105949
1.7066	0.02108287
2.1332	0.02111308
2.5599	0.02115022
2.9865	0.02119442
3.4132	0.02124584
3.8398	0.02130467
4.2665	0.02137115
4.6931	0.02144552
5.1198	0.02152810
5.5464	0.02161923
5.9731	0.02171931