SSM/I and SSMIS Stewardship Code Geolocation Algorithm Theoretical Basis

CSU Technical Report

Mathew R P Sapiano, Stephen Bilanow, Wesley Berg

November 2010

http://rain.atmos.colostate.edu/FCDR/

TABLE of CONTENTS

LIST of FIGURES

LIST of TABLES

ACRONYMS AND ABBREVIATIONS

Acronym or Abbreviation	**Meaning**
CATBD	Climate Algorithm Theoretical Basis Document
CDR	Climate Data Record
NCDC	National Climatic Data Center
NOAA	National Oceanic and Atmospheres Administration

1. Introduction

The following document was prepared for the CSU NOAA/NCDC SSMI/SSMIS Satellite Data Stewardship project and describes how to obtain geolocation information for a conically scanning sensor such as SSM/I or SSMIS. It is assumed that the following are known for each pixel:

1. Spacecraft position vector
2. Spacecraft velocity vector
3. Time
4. Spacecraft attitude and sensor mount angles

In the case of the SSMI/SSMIS Satellite Data Stewardship project, the spacecraft position/velocity vectors were calculated for the first and last pixels for each scan using the NORAD Two Line Elements (TLEs) with the SGP4 code and these values were added to the Basefiles. The spacecraft position and velocity vectors were linearly interpolated across the scan to obtain values at each scan position. The adequacy of the interpolation technique was tested by calculating values from the SGP4 code for the center of the scan and comparing with the interpolated value. This test showed errors of a few centimeters, which is more than adequate for the purpose of geolocation for SSMI/SSMIS. Spacecraft attitude was estimated from a coastline analysis, with slightly different sensor mount angles being used for each feedhorn of SSMIS.

The following describes in detail the calculation of:

1. Pixel geodetic longitude and latitude
2. Satellite zenith angle (Earth Incidence Angle) and satellite azimuth angle
3. Solar beta angle, sun glint angle, solar zenith and azimuth angles and the time since eclipse
4. Spacecraft ephemeris (spacecraft longitude, latitude and altitude)

Some of the more detailed working and explanation can be found in the appendices.

2. Calculation of Pixel Geodetic Latitude and Longitude

The process for calculating the pixel latitude and longitude starts with the calculation of the Instantaneous Field-Of-View (IFOV) matrix in sensor coordinates. Several rotations are required to obtain the IFOV in Geocentric Inertial (GCI) coordinates. First there is the sensor-to-spacecraft rotation that obtains the IFOV relative to the spacecraft. Next there is the spacecraft-to-orbital (geodetic nadir pointing) rotation that obtains the IFOV relative to the path of the spacecraft. Finally, there is the orbital-to-GCI rotation that obtains the IFOV in GCI coordinates. With the IFOV in GCI coordinates, the intersection of the IFOV with the oblate spheroid Earth is calculated and this is then used to get geocentric, then geodetic latitude and longitude.

2.1 Specification of the IFOV

The Instantaneous Field-of-View (IFOV) vector **D** is calculated in coordinates relative to the sensor. Several aspects of the instruments design must be known in order to construct the IFOV including the number of pixels per scan, the scan start angle, the angle between each scan position and the direction of pointing relative to the satellite (forward or backward).

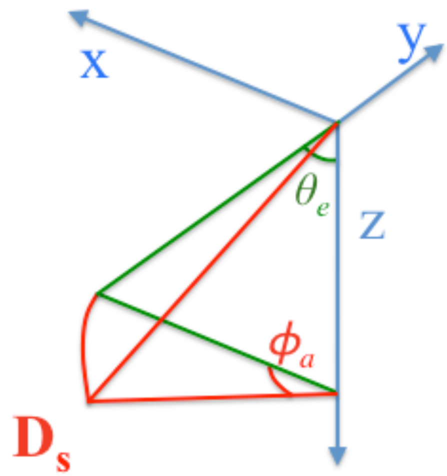

Figure 1. Simple scan geometry for IFOV in sensor coordinates.

Figure 1 shows the scan geometry for a conically scanning instrument. The IFOV vector in sensor coordinates $\mathbf{D_s}$ is calculated as

$$\mathbf{D}_s = \begin{bmatrix} \sin(\theta_e + \Delta_e)\cos(\phi_a + \Delta_a) \\ \sin(\theta_e + \Delta_e)\sin(\phi_a + \Delta_a) \\ \cos(\theta_e + \Delta_e) \end{bmatrix} \quad (1)$$

where θ_e is the nominal elevation angle (half cone angle), ϕ_a is the scan angle,), ϕ_a is the and Δ_a are additive offsets to the nominal elevation and scan angles. The scan angle is calculated from

$$\phi_a = \phi_0 + k_a \Delta_{\text{scan}} \quad (2)$$

where ϕ_0 is the scan start angle, k_a is the scan number (ranges from one to the number of scans) and Δ_{scan} is the angle between scan positions.

2.2 Calculation of the spacecraft attitude matrix A

The spacecraft attitude rotation matrix **A** gives the mapping to go from spacecraft coordinates to nadir pointing coordinates. The spacecraft attitude rotation matrix is calculated using an Euler rotation based on some known (or estimated) attitude offset angles in the directions of pitch, roll or yaw, denoted respectively by θ_p, θ_r and θ_y. For convention, a positive pitch is in the "nose-up" direction, positive roll is in the "bank-left" direction and positive yaw is to the right, with the z-axis pointed towards the Earth as shown in Figure 2. Each of these offsets represents a different rotation matrix denoted as $\mathbf{A}_p$, $\mathbf{A}_r$ and $\mathbf{A}_y$ respectively and give by

$$\mathbf{A}_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_p & -\sin\theta_p \\ 0 & \sin\theta_p & \cos\theta_p \end{bmatrix}$$
$$\mathbf{A}_r = \begin{bmatrix} \cos\theta_r & 0 & \sin\theta_r \\ 0 & 1 & 0 \\ -\sin\theta_r & 0 & \cos\theta_r \end{bmatrix}. \tag{3}$$
$$\mathbf{A}_y = \begin{bmatrix} \cos\theta_y & -\sin\theta_y & 0 \\ \sin\theta_y & \cos\theta_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

These three matrix rotations are combined to form **A**. Since rotations in three dimensions do not commute, the rotation sequence must be specified by the user with the first rotation (preferably) being the largest. For instance, if the yaw is the largest correction, then a 3-2-1 rotation order would be employed. Assuming a 3-2-1 rotation order, the spacecraft attitude rotation matrix **A** is given by

$$\mathbf{A} = \mathbf{A}_p\mathbf{A}_r\mathbf{A}_y. \tag{4}$$

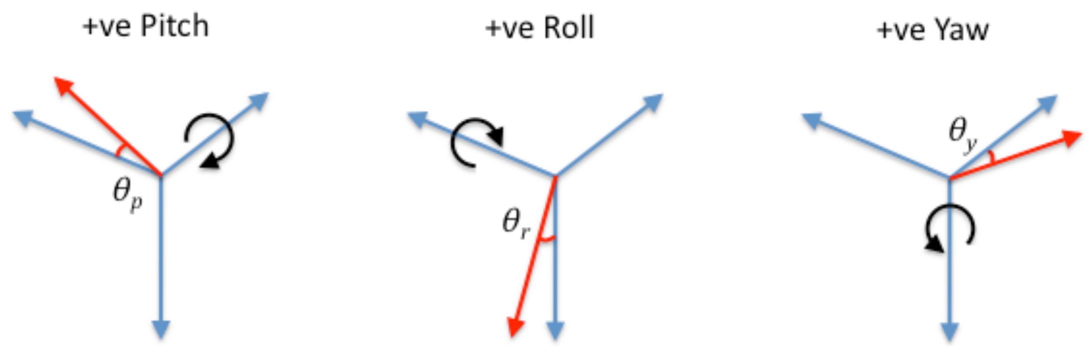

Figure 2. Directions of positive pitch, roll and yaw.

2.3 Calculation of the sensor alignment matrix S

The sensor alignment rotation matrix **S** gives the mapping to go from sensor coordinates to spacecraft coordinates. The sensor alignment rotation matrix is calculated using an Euler rotation based on some known (or estimated) sensor alignment angles in the directions of pitch, roll or yaw. These offsets are relative to the sensor and correspond directly to the pitch, roll and yaw directions if the attitude of the spacecraft is perfectly aligned (ie: spacecraft pitch, roll and yaw adjustments are all zero).

The sensor alignment rotation matrix **S** is calculated in exactly the same way as the spacecraft attitude rotation matrix, but with the sensor offsets rather than the attitude offsets. Additionally, the spacecraft attitude rotation matrix does not change by channel, but the sensor alignment rotation matrix can change based on separate feedhorn alignment (such as with SSMIS). The sensor alignment rotation matrix must therefore be calculated for each feedhorn.

2.4 Interpolation of the spacecraft position and velocity vectors

The basefiles contain the spacecraft position and velocity vectors for the start and end of the active scan. These values are then linearly interpolated to each pixel position across the scan. Each dimension is interpolated separately. Interpolation weights are calculated using time.

2.5 Calculation of the nadir to GCI rotation matrix N

The nadir-to-GCI rotation matrix **M** gives the mapping to go from nadir pointing coordinates to geocentric inertial (GCI) coordinates. Section C2 gives the details of how to obtain the z-axis of the nadir-to-GCI rotation matrix, which does not have a closed form solution. The calculations used here follow those from Patt and Gregg (1994) that have accuracy of around 0.3 arcseconds for a 700km orbit.

The components of the nadir vector $\mathbf{M}_z$ come from (C17), (C20) and (C21)

$$\mathbf{M}_z = \begin{bmatrix} -P_x f' \Big/ \sqrt{P_z^2 + f'^2\left(P_x^2 + P_y^2\right)} \\ -P_y f' \Big/ \sqrt{P_z^2 + f'^2\left(P_x^2 + P_y^2\right)} \\ -P_z \Big/ \sqrt{P_z^2 + f'^2\left(P_x^2 + P_y^2\right)} \end{bmatrix} \tag{5}$$

where

$$f' = \left(1 - f_p\right)^2 = \frac{R_m\left(1 - f\right)^2 + |\mathbf{P}| - R_m}{|\mathbf{P}|} \tag{6}$$

such that P_x, P_y and P_z are the components of the spacecraft position vector, f is the flattening factor for the Earth ellipse, f_p is the flattening factor for the ellipse described in section C2, R_m is the mean Earth radius (R_m≈6371) and |**P**| is the magnitude of the vector **P**.

The *y*-axis component of **M** is estimated using the spacecraft velocity vector **V**. The vector **T** is defined as normal to the orbit plane (the *z*-*x* plane) such that

$$\mathbf{T} = \mathbf{M}_z \times \mathbf{V} \tag{7}$$

The *y*-axis component of **M** is then given by

$$\mathbf{M}_y = \begin{bmatrix} T_x / |\mathbf{T}| \\ T_y / |\mathbf{T}| \\ T_z / |\mathbf{T}| \end{bmatrix} \tag{8}$$

where T_x, T_y and T_z are the components of the vector **T**.

Finally, the *x*-axis component of **M** is estimated from the *z* and *y*-axis components

$$\mathbf{M}_x = \mathbf{M}_y \times \mathbf{M}_z \tag{9}$$

The nadir-to-GCI rotation matrix is then constructed by combining these

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_x & \mathbf{M}_y & \mathbf{M}_z \end{bmatrix} \tag{10}$$

2.6 Calculation of the IFOV vector in GCI coordinates

The sensor alignment, spacecraft attitude and nadir to GCI rotation matrices are used to convert the IFOV in sensor coordinates $\mathbf{D}_s$ to the IFOV in GCI coordinates. The sensor alignment rotation matrix can be used to convert a vector from spacecraft to sensor coordinates, so the transpose of this matrix is required to go from sensor to spacecraft coordinates. In a similar way, the spacecraft attitude rotation matrix can be used to convert a vector from nadir to spacecraft coordinates, so the transpose of this matrix is required to go from spacecraft to nadir coordinates. Finally, the nadir to GCI rotation matrix can be used to convert from nadir to GCI coordinates. The IFOV in GCI coordinates $\mathbf{D}_i$ is therefore obtained by pre-multiplying $\mathbf{D}_s$ by the three rotation matrices

$$\mathbf{D}_i = \mathbf{M}\mathbf{A}^T\mathbf{S}^T\mathbf{D}_s \tag{11}$$

2.7 Intersection with Oblate Earth

Next, the intersection of the IFOV in GCI coordinates $\mathbf{D}_i$ with the Earth is calculated to find the target position vector **G** in GCI coordinates (see Figure 3) and the distance between the satellite and the ground target *d*.

The vector for the target position can be found directly from the spacecraft position vector **P** and the IFOV vector $\mathbf{D}_i$ (both already in GCI coordinates)

$$\mathbf{G} = \begin{bmatrix} P_x + dD_{ix} \\ P_y + dD_{iy} \\ P_z + dD_{iz} \end{bmatrix} \tag{12}$$

The equation for an ellipsoid in standard Cartesian coordinates is

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{13}$$

For an oblate spheriod Earth a=b=R_E which is the equatorial radius of the Earth and c=R_P which is the polar radius of the Earth. Using the definition of the Earth flattening factor R_P =(1-f)R_E, substituting these into the above equation and rearranging gives

$$x^2 + y^2 + z^2 \big/ \left(1 - f\right)^2 = R_E^2 \tag{14}$$

A quadratic equation is found by substituting the components of **G** into (14), expanding out all brackets and rearranging in terms of *d*

$$\left[\frac{D_{ix}^2 + D_{iy}^2}{R_E^2} + \frac{D_{iz}^2}{R_P^2}\right]d^2 + \left[\frac{2P_xD_{ix} + 2P_yD_{iy}}{R_E^2} + \frac{2P_zD_{iz}}{R_P^2}\right]d + \left[\frac{P_x^2 + P_y^2}{R_E^2} + \frac{P_z^2}{R_P^2} - 1\right] = 0 \qquad (15)$$

(15) can be solved using the factorization of $ad^2+bd+c=0$ that is given by

$$d = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \qquad (16)$$

If there are two real solutions for *d*, the smaller one is the required value of *d*. If there is only one solution, the line of sight is tangent to the ellipsoid. If there is no real solution, the line of sight is missing the ellipsoid.

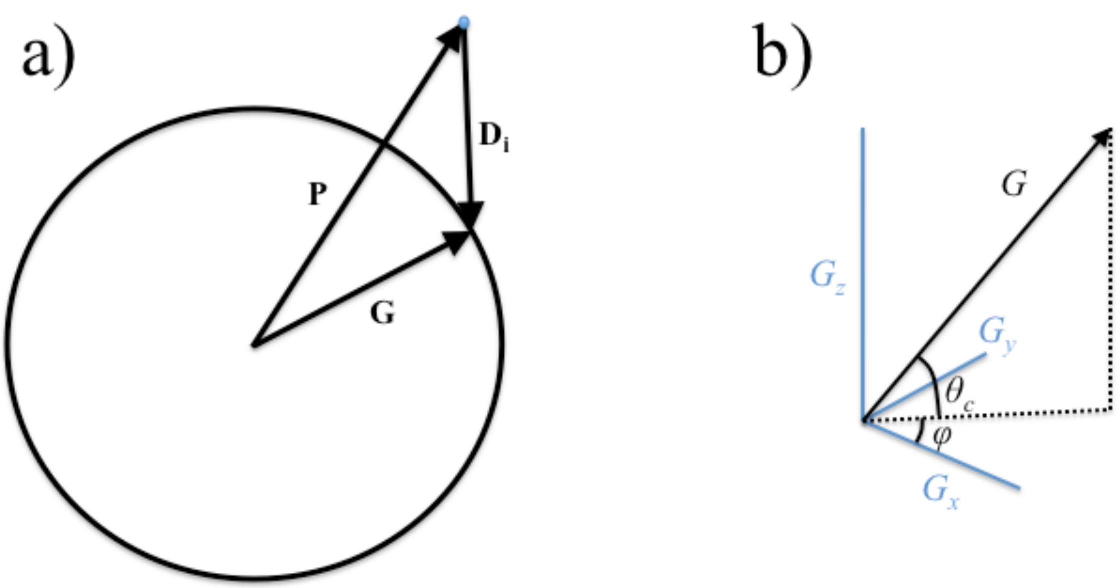

Figure 3. Diagram showing the spacecraft position vector **P**, the IFOV vector $\mathbf{D}_i$ and the target position vector **G**.

2.8 Calculation of Greenwich Hour Angle

The Greenwich Hour Angle (GHA) serves to rotate the inertial coordinates to an Earth-fixed coordinate system. The geodetic longitude as calculated in the inertial frame (GCI coordinates) would coincide with the vernal equinox, so the GHA is required to adjust the geodetic longitude to the current orientation of the Earth. The details of the calculation of the GHA are given in section C3.

2.9 Calculation of geodetic latitude and longitude

The calculation of the geodetic latitude and longitude is given in section A1. The geodetic latitude is given by (C3) as

$$\theta_d = \tan^{-1}\left(\frac{G_z}{\left(1-f\right)^2\sqrt{G_x^2+G_y^2}}\right) \tag{17}$$

It is recommended that the atan2 function be used to calculate the arctan so as to ensure the answer is in the correct quadrant. The longitude in inertial coordinates is given by (C1) as

$$\phi_d = \tan^{-1}\left(\frac{G_y}{G_x}\right) \tag{18}$$

although this must be adjusted to Earth fixed coordinates. This adjustment is achieved using the following

$$\phi = \tan^{-1}\left(\frac{G_y}{G_x}\right) - \text{GHA} \tag{19}$$

where ϕ is the geodetic longitude of the pixel in Earth fixed coordinates and GHA is the Greenwich hour angle for the current pixel time.

2.10 Calculation of satellite zenith and azimuth

In order to calculate the satellite zenith (aka Earth Incidence Angle) and azimuth angles, it is necessary to first determine the local zenith (up), north and east vectors (Figure 4a). The satellite zenith angle is the angle between the local up vector and the pointing vector and the azimuth angle is the angle between the projection of the pointing vector on the surface and the local north vector (fig. 4b) where positive azimuth angle is clockwise when viewed from above.

The local up vector represents the vector normal to the Earth's surface at a given point. It is analogous to the nadir $\mathbf{M}_z$, which was calculated as part of the nadir-to-GCI rotation matrix, but with **M** replaced by $\mathbf{L_u}$ and **P** replaced by **G**

$$\mathbf{L}_U = \begin{bmatrix} -G_x f' \Big/ \sqrt{G_z^2 + f'^2\left(G_x^2 + G_y^2\right)} \\ -G_y f' \Big/ \sqrt{G_z^2 + f'^2\left(G_x^2 + G_y^2\right)} \\ -G_z \Big/ \sqrt{G_z^2 + f'^2\left(G_x^2 + G_y^2\right)} \end{bmatrix} \tag{20}$$

where *f'* is now simply

$$f' = \left(1 - f\right)^2 \tag{21}$$

The local east vector is normal to both the local up vector and the z-axis. It can therefore be found as

$$\mathbf{L}_E = \frac{\mathbf{Z} \times \mathbf{L}_U}{\left|\mathbf{Z} \times \mathbf{L}_U\right|} \tag{22}$$

The local north can then be found as

$$\mathbf{L}_N = \mathbf{L}_U \times \mathbf{L}_E \tag{23}$$

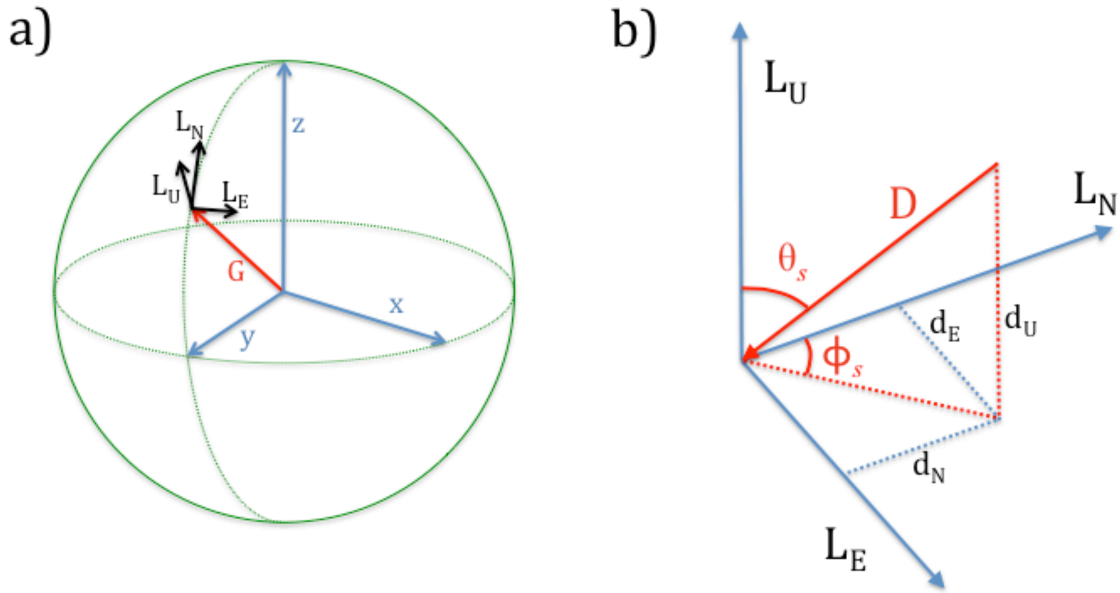

Figure 4. Diagram showing (a) the local zenith (up), east and west vectors on Earth's surface relative to the GCI frame and (b) the zenith and azimuth angles relative to the pointing vector **D**.

The zenith and azimuth angles can now be calculated using the local vectors and a vector from the pixel to the satellite, which is the negative of the pointing vector (fig. 4) $\mathbf{D}_n=-\mathbf{D}$. The components of this vector can be found as the dot product (ie: the scalar projection) of $\mathbf{D}_n$ on the local up, north and east vectors

$$\begin{aligned} d_U &= \mathbf{D}_n \bullet \mathbf{L}_U \\ d_N &= \mathbf{D}_n \bullet \mathbf{L}_N \\ d_E &= \mathbf{D}_n \bullet \mathbf{L}_E \end{aligned} \tag{24}$$

Following the diagram in fig 4b, the zenith and azimuth angles can be expressed as

$$\theta = \tan^{-1}\left(\frac{\sqrt{d_N^2 + d_E^2}}{d_U}\right) \tag{25}$$

and

$$\phi = \tan^{-1}\left(d_E / d_N\right) \tag{26}$$

In both cases, the atan2 function should be used for calculation so as to ensure that the answer is in the right quadrant.

3. Calculation of Solar angles

3.1 Computation of the Sun vector

The computation of the Sun vector in GCI coordinates is given in section C4.

3.2 Solar Beta Angle

The solar beta angle is the angle of elevation of the Sun in the orbit plane as shown in Figure 5.

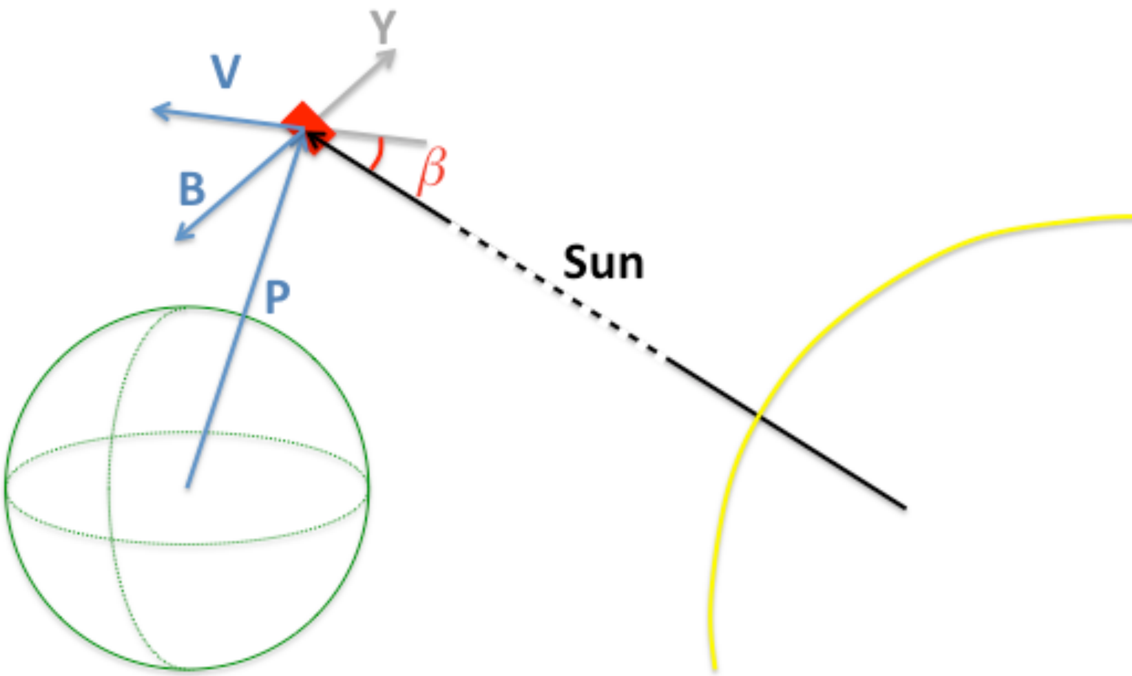

Figure 5. Diagram of angles associated with the solar beta angle.

To calculate the solar beta angle, first calculate the unit orbit normal in GCI coordinates

$$\mathbf{B} = \frac{\mathbf{P} \times \mathbf{V}}{|\mathbf{P} \times \mathbf{V}|} \quad (27)$$

Next, calculate the projection of the unit orbit normal on the Sun vector

$$b = \mathbf{B} \bullet \mathbf{S}' \quad (28)$$

where **S**' is the normalized Sun vector in GCI coordinates. Finally, the solar beta angle is found as

$$\beta_s = 90 - \cos^{-1} b \tag{29}$$

3.3 Solar zenith and azimuth

The solar zenith and azimuth are calculated in the same way as the satellite zenith and azimuth but with the pointing vector $\mathbf{D}_i$ replaced by the **Sun** vector.

3.4 Sun Glint angle

The Sun glint angle at any pixel location on the Earth is the angle between the satellite direction and the surface reflected Sunlight direction.

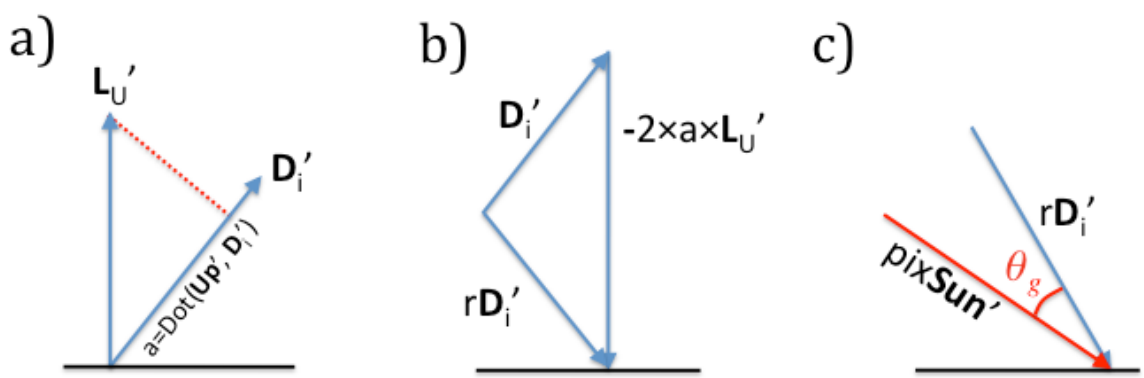

Figure 6. Angles required to obtain the Sun glint angle.

In order to obtain the Sun glint angle several other angles must first be known in a common coordinate system (GCI in this case) including the Sun vector **Sun** (section 3.1), the pixel location vector **G** (12), the pointing vector $\mathbf{D}_i$ (11) and the local vertical vector $\mathbf{L}_U$ (20). Figure 6 shows some of the angles required to get the Sun glint angle. All of the vectors must be unit vectors, which is denoted by a dash.

First, the projection of the local vertical unit vector onto the pointing unit vector is calculated as

$$a = \mathbf{L}_U' \bullet \mathbf{D}_i' \tag{30}$$

Conceptually, this used to construct a right angle triangle as shown in fig 6b. The reflected pointing vector is then

$$r\mathbf{D}_i' = \mathbf{D}_i' - 2a\mathbf{L}_U' \tag{31}$$

After normalizing the reflected pointing vector, the Sun glint angle θ_g is found (as shown in fig 6c) from the Sun vector at the pixel location (pix**Sun**=**Sun**-**G**) as

$$\theta_g = \text{pix}\mathbf{Sun} \bullet r\mathbf{D}_i' \tag{32}$$

3.5 Time since eclipse

The time since eclipse is useful for assessing solar heating issues with a sensor where information on how long a spacecraft has been in direct sunlight or shadow can be used to derive corrections.

The first step is to calculate the GCI-to-Orbital rotation matrix **O**. The orbital coordinate system is explained in section B3 and is geocentric nadir pointing. The z-axis is aligned with the spacecraft position vector **P**

$$\mathbf{O}_z = -\mathbf{P} / |\mathbf{P}| \tag{33}$$

The y-axis is normal to the z-axis and the velocity vector **V**

$$\mathbf{O}_y = \frac{\mathbf{O}_z \times \mathbf{V}}{|\mathbf{O}_z \times \mathbf{V}|} \tag{34}$$

The x-axis is then simply

$$\mathbf{O}_x = \mathbf{O}_y \times \mathbf{O}_z \tag{35}$$

Finally, the GCI-to-Orbital rotation matrix is

$$\mathbf{O} = \begin{bmatrix} \mathbf{O}_x & \mathbf{O}_y & \mathbf{O}_z \end{bmatrix} \tag{36}$$

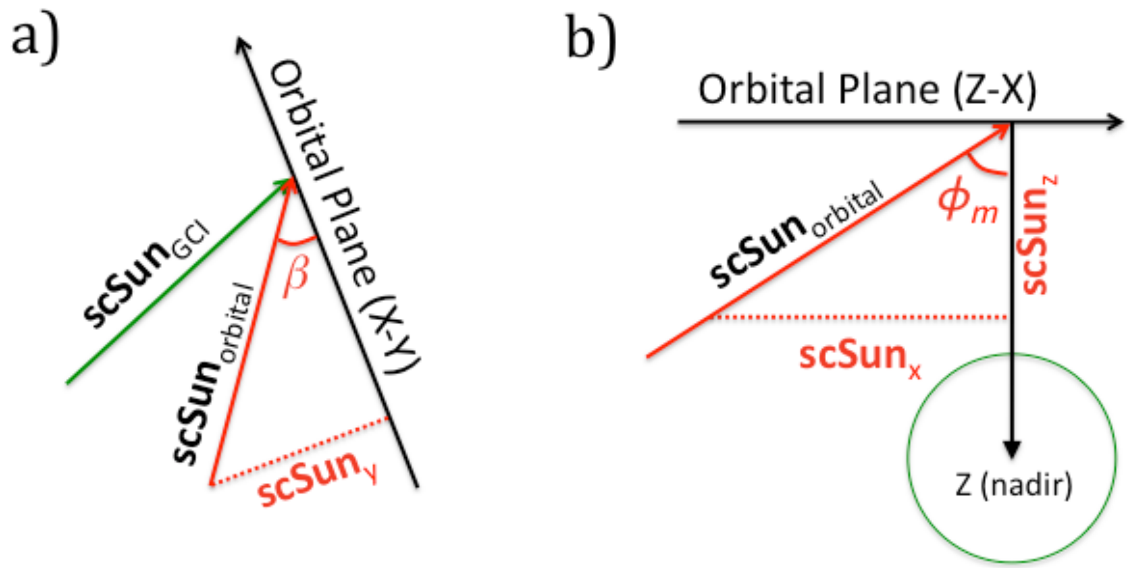

Figure 7. Diagrams of (a) alternate method for obtaining solar beta angle and (b) method for obtaining phase of orbit midnight.

The solar beta angle can be calculated using an alternate method to that shown in section 3.2. This method is more consistent with the calculations in this section, but requires the GCI-to-Orbital rotation matrix. The Sun vector in GCI coordinates **Sun** is used to calculate the Sun to spacecraft vector as

$$\mathrm{sc}\mathbf{Sun}_i = \mathbf{Sun} - \mathbf{P} \tag{37}$$

This is then rotated to the orbital frame using the GCI-to-Orbital rotation matrix as shown in Figure 7a. The solar beta angle β can be calculated as

$$\beta = -\sin^{-1}\left(\mathrm{sc}\mathbf{Sun}_y\right) \tag{38}$$

Next, the phase since orbit midnight (the point at which the Sun, Earth and Satellite are aligned along a straight line) which is the phase of the Sun in the orbital plane as shown in Figure 7b

$$\phi_\mathrm{m} = \tan^{-1}\left(\frac{\mathrm{sc}\mathbf{Sun}_x}{\mathrm{sc}\mathbf{Sun}_z}\right) \tag{39}$$

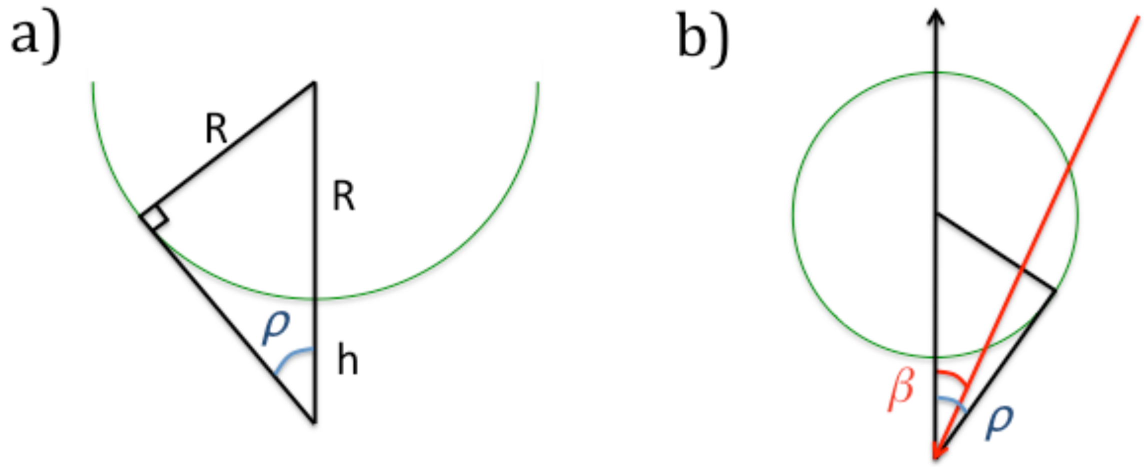

Figure 8. Diagrams showing angles used to calculate (a) Earth angular radius and (b) orbit phase for eclipse entry/exit.

Figure 8a shows a diagram of the Earth angular radius ρ as seen from the spacecraft. This model assumes a spherical Earth with Earth radius R that must be known at the current point. Wertz (1978; Eq. 4.14) gave a simple expression for the Earth radius at a given geodetic latitude θ_d (the spacecraft latitude in this case)

$$R = R_E\left(1 - f \sin^2 \theta_d\right) \tag{40}$$

The Earth angular radius ρ as seen from the spacecraft is then a function of the Earth Radius and the spacecraft altitude h

$$\rho = \sin^{-1}\left(\frac{R}{R+h}\right) \tag{41}$$

Next, the orbit phase for eclipse entry/exit is calculated as shown in fig 8b. This is essentially the difference between the Earth angular radius and the solar beta angle and gives a measure of the angular distance between eclipse entry/exit based on orbit midnight. The orbit phase for eclipse entry/exit is

$$\phi_e = \cos^{-1}\left(\frac{\cos\rho}{\cos\beta}\right) \tag{42}$$

Next, the angular orbit rate is calculated. The angular orbit rate requires is calculated from the velocity vector perpendicular to the position vector, which is slightly different to the velocity vector. The velocity vector therefore undergoes a correction outlined in Figure 9 so as to obtain the correct orbit rate. First, the position and velocity vectors are unitized. Second, the angular separation between these two vectors is calculated as

$$\theta = \cos^{-1}(\mathbf{P}' \bullet \mathbf{V}') \quad (43)$$

Third, the magnitude of the velocity perpendicular to the position vector is found as

$$|\mathbf{V}_p| = |\mathbf{V}|\sin\theta \quad (44)$$

Finally, the orbit rate is given by

$$\omega_{\mathrm{sc}} = \frac{|\mathbf{V}_p|}{|P|} \quad (45)$$

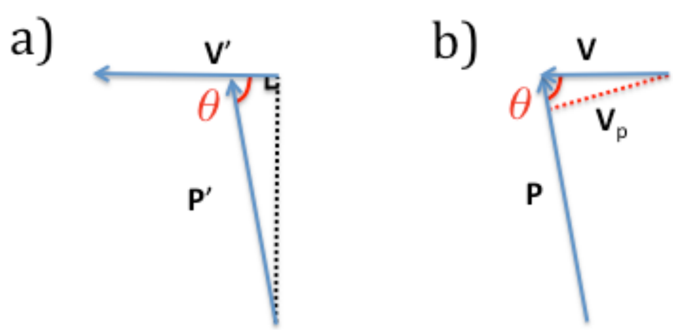

Figure 9. Diagrams showing the process used to correct the velocity vector and calculate the orbit rate.

The angle since eclipse entry can then be estimated as the phase from orbit midnight (i.e. how far is the spacecraft from the mid-point of Earth shadow) plus the phase of eclipse exit (i.e. how long since orbit midnight). If this angle is negative, then it should have 360° added to it. The time since eclipse entry is therefore

$$t_E = \frac{\phi_m + \phi_e}{\omega_{\mathrm{sc}}} \quad (46)$$

4. Calculation of spacecraft ephemeris

The spacecraft ephemeris for SSM/I consists of spacecraft latitude, longitude and altitude at each scan time. These are calculated using the same technique as was used to calculate the pixel position, but with a simplified pointing vector.

The spacecraft latitude and longitude are defined at the point on Earth closest to the satellite, which is found by considering a vector normal to the surface that extends to the satellite. The pointing vector in sensor coordinates is thus

$$\mathbf{D}_S = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \quad (47)$$

The nadir to GCI rotation matrix is required to rotate $\mathbf{D}_S$ to GCI coordinates. This is calculated using the technique in section 2.5 with the position and velocity vectors for the scan time. The pointing vector in GCI coordinates is then found as

$$\mathbf{D}_i = \mathbf{M}\mathbf{D}_S \quad (48)$$

The intersection with the oblate spheroid is done using the technique in section 2.7 and the distance between the satellite and the ground point given in (16) is the satellite altitude.

The Greenwich Hour Angle is calculated as in section 2.8 and is used to calculate the spacecraft latitude and longitude using the same method outlined in section 2.9

Appendix A: Constants and satellite values

A1. Constants used in calculations

Earth equatorial radius:	R_E = 6378.137 km
Earth polar radius:	R_P = 6356.755 km
Earth mean radius:	R_M = 6371 km
Earth flattening factor:	f = 1/298.257 (dimensionless)
Earth rotation rate:	$\omega_E = 360/86400 = 7.29211585494\times10^{-5}\ s^{-1}$

Table A-1: Earth Constants

A2. Standard SSM/I parameters

Number of pixels per scan:	128 high-res; 64 low-res
Angle between scans:	0.8°
Time between scans:	1.899 s high-res; 3.798 s low res
Scan cone angular radius:	45°
Start scan angle:	-51.0°
Total rotation spanned for scan:	102.4°
Active scan duration:	1.899*0.8*127/360 = 0.5359 s
Sampling frequency:	4.22 ms

Table A-2: SSM/I Scan Parameters

Note: SSM/I was generally mounted so that it viewed behind with the exception of F8, which viewed forward.

A3. Standard SSMIS parameters

Number of pixels per scan:	180 high-res; 90 low-res
Angle between scans:	0.8°
Time between scans:	1.899 s high-res; 3.798 s low res
Scan cone angular radius:	45°
Start scan angle:	
Total rotation spanned for scan:	143.2°
Active scan duration:	1.899*0.8*179/360 = 0.7554 s
Sampling frequency:	4.22 ms
Time bias for first pixel:	

Table A-3: SSMIS Scan Parameters

Appendix B: Reference frames and time scales

B1. GCI

Geocentric Inertial (GCI; alternatively known as Earth Centered Inertial or ECI) coordinates have their center at the center of the Earth. The z axis is in line with the rotation axis of the Earth. These are celestial coordinates, so the *x*-axis is fixed as the point where the plane of Earth's orbit around the Sun crosses the prime meridian, known as the vernal equinox. The *y*-axis is then normal to the *z* and *x* axes. One issue with this coordinate system is that it is not truly inertial because the coordinate system is not fixed relative to the position of the stars (this is the phenomenon of the *precession of the equinoxes*). For this reason, a reference time is needed to define the position of the vernal equinox, which is the approach taken here, but this can also be handled by using true-of-date coordinates where corrections are made for each time. This problem must be incorporated into Greenwich Hour Angle calculations that are not discussed here.

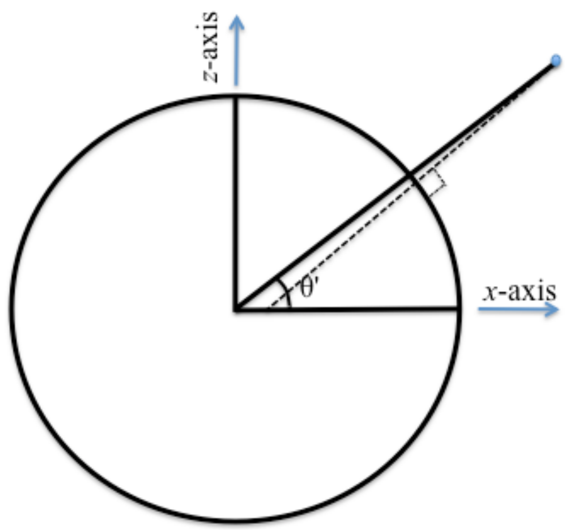

Figure B-1: Diagram of geocentric inertial coordinates

B2. Geodetic

The Geodetic coordinate system is the most commonly used system for describing positions on the Earth (including latitude and longitude). They have the same basic directions as GCI coordinates, but the center is no longer the Earth's center. Rather, the center is the point at which a vector normal to the surface intersects the *x*-*y* plane.

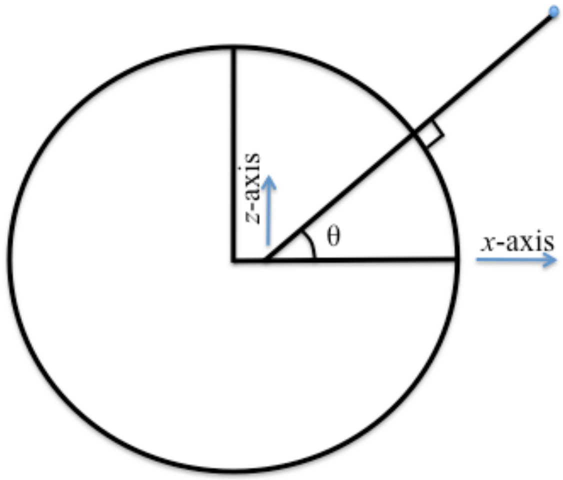

Figure B-2: Diagram of geodetic coordinates

B3. Orbital (geocentric nadir)

The *z*-axis points in the nadir direction, but is centered at the geocentric origin. The z-axis is therefore aligned with the position vector. The *x*-axis is approximately in the direction of travel of the spacecraft and the *y*-axis is normal to the orbit plane.

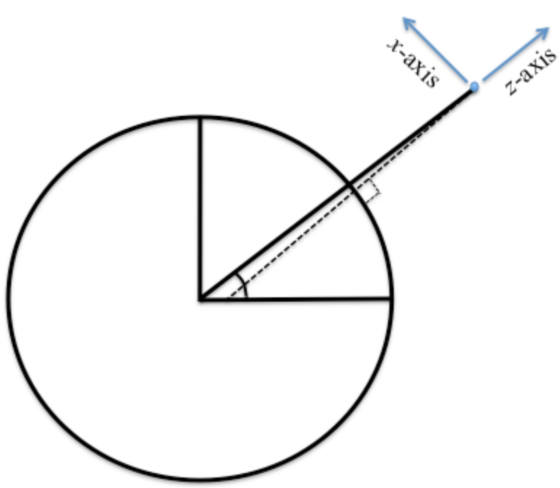

Figure B-3: Diagram of orbital coordinates

B4. Local horizontal

Local horizontal coordinates have their center at a given point on the Earth's surface. The N-axis points towards the direction of increasing latitude (North) and the E-axis points towards the direction of increasing longitude (East). The *z*-axis points opposite the geodetic nadir direction (ie: outward from Earth and normal to the surface).

B5. Time

Several time definitions are required in order to calculate the geolocation. In particular, the calculation of the Greenwhich Hour Angle (GHA) and the Sun vector are sensitive to errors in the input time. The GHA affects the spacecraft and pixel longitudes and the Sun vector affects the sun angles (solar zenith, azimuth, solar beta, Sun glint angle, time since eclipse). It is important to note that relatively small errors in D_U can lead to large errors; for example: a 10 second error in D_U might lead to an error in the longitude of around 4km.

International Atomic Time (**TAI**; Temps Atomique International): measured as seconds since January 1st 1958, *with no leap seconds*. The unit of TAI is seconds on the geiod.

Terrestrial Time (**TT**): idealized form of TAI. TAI can be estimated as TT=TAI+32.184 seconds.

Universal Time 1 (**UT1**): mean solar time. UT1 is counted from 0h (midnight) and is affected by irregularities in Earth's rotation rate.

Coordinated Universal time (**UTC**): equivalent to UT1 to within 0.9 seconds. Leap seconds are irregularly added to correct UTC to match mean solar time (UT1). UTC can be estimated as UTC=TAI+ΔUT where ΔUT is the number of leap seconds.

Julian Day: number of days since 4713 BC January 1 12z (noon). Julian Days can be measured on either the UTC or TT timescales.

J2000: a commonly used epoch for calculations. J2000 is equivalent to Julian Day 2451545.

If the date is known in year, month, day form, then it must be converted to Julian day. This can be done using a simple form given by van Flandern and Pulkkinen (1979). The application of this formula requires that division by an integer yield an integer, and it is thus somewhat ambiguous. Fernie (1983) gave a simple computer program (in Basic) for the same simplified formula. The function denoted "floor" implies truncation of the subject to an integer. The Julian day (on UTC timescale) as an integer is given by

$$d_j = 367y - \mathrm{floor}\left\{7\left[y + \mathrm{floor}\left\{(m+9)/12\right\}\right]/4\right\} + \mathrm{floor}\left\{275m/9\right\} + d + 1721014 \tag{B1}$$

with the fractional part given by

$$d_f = \frac{h}{24} + \frac{m}{1440} + \frac{s}{86400} \tag{B2}$$

In order to calculate the geolocation, the time in Julian centuries in TT time and Julian days since J2000 in UTC time are required. These are calculated using the Julian day on the TT and UTC timescales. SSM/I and SSMIS measure time as TAI seconds since 1987 January 1st 00z (xtime). This is easily converted to Julian days on the TT timescale using

$$\mathrm{JD}_{TT} = 2446796.5 + (\mathrm{xtime} + 32.184)/86400 \tag{B3}$$

where 2446796.5 is the Julian Day for 1987 January 1st 00z. Julian Day on the UTC scale is then calculated as

$$\mathrm{JD}_{UTC} = \mathrm{JD}_{TAI} + \Delta\mathrm{UT}/86400 \tag{B4}$$

where ΔUT is the number of leap seconds.

Appendix C: Calculations required for geolocation

The following describes the basis for the calculation of pixel geolocation, satellite zenith and azimuth angles and solar beta angle, zenith and azimuth angles. The transformation from GCI to geodetic nadir is often found using an iterative technique, however, the approach described here is based on that suggested by Patt and Gregg (1994) who give a closed form solution based on some minor assumptions, the accuracy of which are discussed in the text.

C1. Calculation of latitude and longitude from a pixel location vector

Figure 3 shows the (currently unknown) geocentric pixel location vector **G** that can be used to calculate pixel geodetic latitude and longitude. The geodetic longitude ϕ_d in inertial coordinates is equal to the geocentric longitude ϕ_c that is simply

$$\phi_d = \phi_c = \tan^{-1}\left(\frac{G_y}{G_x}\right) \tag{C1}$$

where G_x and G_y are the *x* and *y* components of **G**. Note that (C1) must be rotated to Earth-fixed coordinates. The geocentric latitude θ_c is

$$\tan\theta_c = \frac{G_z}{\sqrt{G_x^2 + G_y^2}} \tag{C2}$$

The geodetic latitude θ_d can be obtained from the geocentric latitude θ_c using

$$\tan\theta_d = \frac{\tan\theta_c}{(1-f)^2} = \frac{G_z}{(1-f)^2\sqrt{G_x^2 + G_y^2}} \tag{C3}$$

where f is the Earth flattening factor (Bate et al. 1971; P 94).

C2. Calculation of z-axis component of the nadir to GCI rotation matrix

The nadir to GCI rotation matrix **N** rotates a vector from geodetic nadir coordinates to GCI coordinates. The z-axis component of this rotation matrix does not have a closed form solution so that iterative solutions have been used in the past. Patt and Gregg (1994) suggested an approximation that produces a very high accuracy and was used for the NASA SeaWiFS and TRMM missions. They state that the accuracy at altitude 705 km is 0.3 arcseconds.

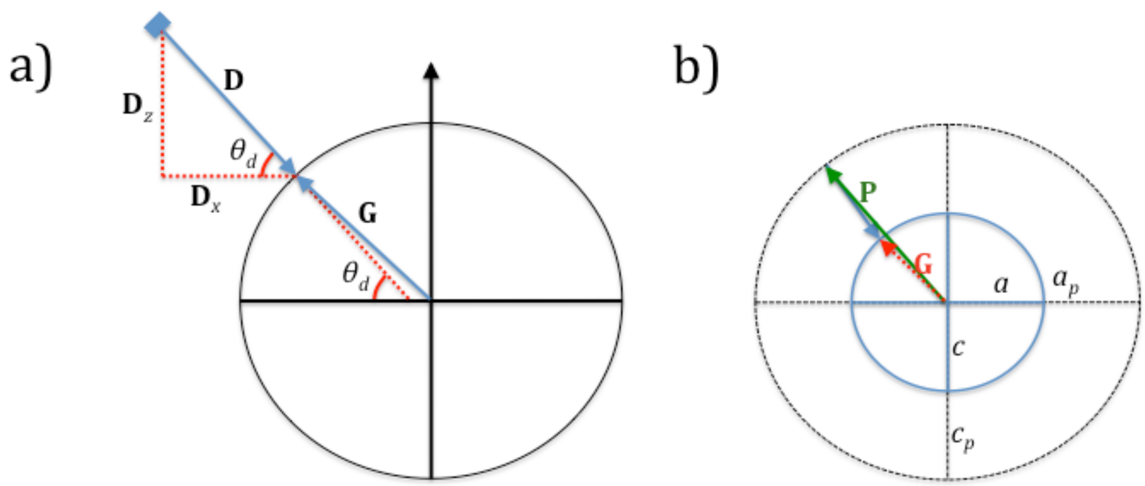

Figure C-1. Diagram showing (a) nadir vector **D**, pixel location vector **G** along with some other relevant quantities and angles used to find the z-axis component of the nadir-to-GCI rotation matrix, and (b) the ellipsoid spheroid Earth with flattening factor f and the ellipsoid with flattening factor f_p which denotes the approximate orbit of the satellite.

First, the nadir vector along the z-axis is calculated. Figure C1 shows the nadir vector in the x-z plane. The point of interest is the point nearest the spacecraft where the ellipsoid is normal to the vector **D**. From C2, it can be seen that

$$\mathbf{D}_z = \tan\theta_d \mathbf{D}_x \tag{C4}$$

Since this is in the x-z plane, (C3) becomes

$$\tan\theta_d = \frac{G_z}{(1-f)^2 G_x} \tag{C5}$$

The position vector **P** is shown in Figure 3 as the difference between **G** and **Z**, so that

$$\mathbf{P}_z = \mathbf{G}_z - \mathbf{D}_z = G_z - \frac{G_z D_x}{G_x(1-f)^2} = \frac{G_z\left[G_x(1-f)^2 - D_x\right]}{G_x(1-f)^2} \tag{C6}$$

Now, a second ellipsoid can be defined at the spacecraft that is also normal to the zenith vector **G**, as shown in fig. C1b. The flattening factor of the Earth ellipsoid is $f=(a-c)/a$, whereas the flattening factor of this second ellipsoid is $f_p=(a_p-c_p)/a_p$, which is different from *f*. Analogous to (C5)

$$\frac{P_z}{(1-f_p)^2 P_x} = \frac{G_z}{(1-f)^2 G_x} \tag{C7}$$

Noting from fig C1a that $P_x= G_x-D_x$, and using (C6), (C7) becomes

$$\frac{G_z\left[G_x(1-f)^2 - D_x\right]}{G_x(1-f)^2(1-f_p)^2(G_x - D_x)} = \frac{G_z}{(1-f)^2 G_x} \tag{C8}$$

that cancels out to give

$$(1-f_p)^2 = \frac{G_x(1-f)^2 - D_x}{G_x - D_x} \tag{C9}$$

A simplification can be made by assuming that P_x, G_x and D_x have the same relative magnitudes as |**P**|, |**G**| and |**D**|.

$$(1-f_p)^2 = \frac{|\mathbf{G}|(1-f)^2 - |\mathbf{D}|}{|\mathbf{G}| - |\mathbf{D}|} \tag{C10}$$

This now gives a method for determining f_p, but **G** is still unknown. A good approximation can be made by assuming |**G**| is the mean Earth radius R_m. This can be further simplified by noting that, in this case, |**D**|=|**G**|-|**P**| so that (C10) becomes

$$\left(1 - f_p\right)^2 = \frac{R_m\left(1 - f\right)^2 + |\mathbf{P}| - R_m}{|\mathbf{P}|} \tag{C11}$$

Patt and Gregg (1994) note that an iterative procedure can be used to improve the accuracy is necessary, by calculating f_p and recalculating (C9).

Figure C3 shows a diagram of the vectors **P**, **G** and **N** and gives a schematic for how to obtain N_z. Starting with **N**

$$\tan\theta_d = \frac{-N_z}{\sqrt{N_x^2 + N_y^2}} \tag{C12}$$

and **P**

$$\tan\theta_c = \frac{P_z}{\sqrt{P_x^2 + P_y^2}} \tag{C13}$$

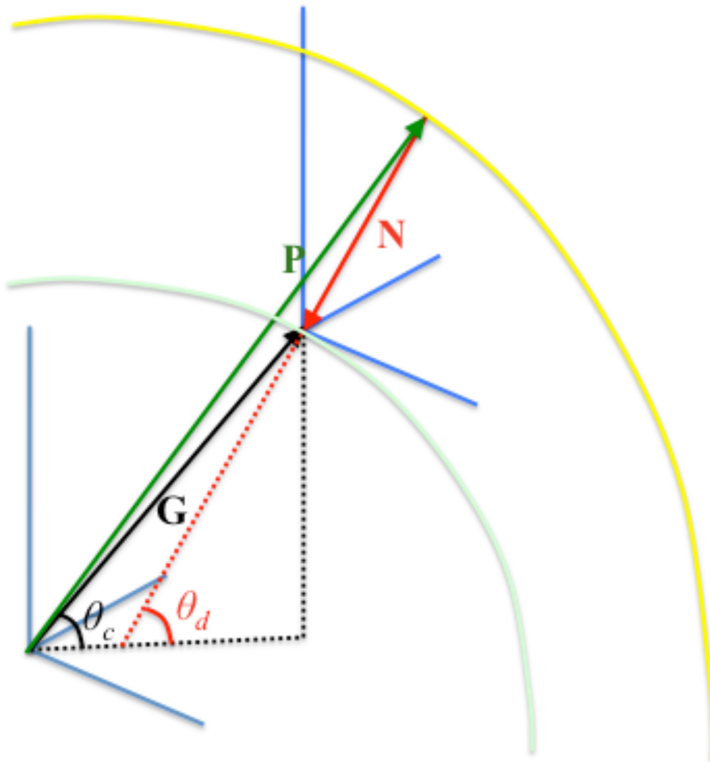

Figure C-2. Diagram showing angles and ellipsoids. The light green ellipsoid represents Earth and has flattening factor *f*, whereas the yellow ellipsoid has flattening factor f_p.

Following the form of (C3) but for the yellow ellipse in fig C3 gives

$$\tan\theta_d = \frac{\tan\theta_c}{\left(1-f_p\right)^2} = \frac{P_z}{\left(1-f_p\right)^2\sqrt{P_x^2+P_y^2}} \tag{C14}$$

and substituting (C12) gives

$$\frac{-N_z}{\sqrt{N_x^2+N_y^2}} = \frac{P_z}{\left(1-f_p\right)^2\sqrt{P_x^2+P_y^2}} \tag{C15}$$

If **N** is a unit vector, then

$$N_x^2+N_y^2 = 1-N_z^2 \tag{C16}$$

so that (C15) becomes

$$N_z^2 = \frac{P_z^2 - P_z^2 N_z^2}{\left(1 - f_p\right)^4 \left(P_x^2 + P_y^2\right)} = \frac{P_z^2 - P_z^2 N_z^2}{\alpha} = \frac{P_z^2}{\alpha} - \frac{P_z^2 N_z^2}{\alpha}$$

that gives

$$N_z = \frac{P_z}{\sqrt{\left(1 - f_p\right)^4 \left(P_x^2 + P_y^2\right) + P_z^2}} \tag{C17}$$

In a similar way, (C16) can be used to eliminate N_z from (C15)

$$\sqrt{N_x^2 + N_y^2} = \frac{\left(1 - f_p\right)^2 \sqrt{P_x^2 + P_y^2}}{\sqrt{P_z^2 + \left(1 - f_p\right)^4 \left(P_x^2 + P_y^2\right)}} \tag{C18}$$

Since geodetic and geocentric longitudes are equal, then

$$\tan\phi = \frac{N_x}{N_y} = \frac{P_x}{P_y} \tag{C19}$$

which can be used to eliminate N_y and N_x respectively from (C18) to get

$$N_x = \frac{P_x\left(1 - f_p\right)^2}{\sqrt{P_z^2 + \left(1 - f_p\right)^4 \left(P_x^2 + P_y^2\right)}} \tag{C20}$$

and

$$N_y = \frac{P_y\left(1 - f_p\right)^2}{\sqrt{P_z^2 + \left(1 - f_p\right)^4 \left(P_x^2 + P_y^2\right)}} \tag{C21}$$

C3. Calculation of Greenwich Hour Angle

The first step for calculation of the Greenwich Hour Angle (GHA) is to obtain the correctly formatted time. The Julian Date must be found on both TT and UTC timescales (see Appendix B). Once these are known, the Julian Day since the J2000 epoch in UT1 time can be defined as

$$D_U = \mathrm{JD}_{UTC} + 2451545.0 \tag{C24}$$

where JD_{UTC} is the Julian day (including fractional part) on the UTC timescale. In addition to the definition of time in (C24), the number of Julian centuries (of length 36525 days) since J2000 in terrestrial time is required. This can be found as

$$T = \left(\mathrm{JD}_{TT} + 2451545.0\right)/36525 \tag{C25}$$

In order to calculate the GHA, several quantities must first be calculated (Astronomical Almanac, 2010; page B11). The Greenwich Mean Sidereal Time (GMST) is calculated as (Astronomical Almanac, 2010; page B8)

$$\begin{aligned}\mathrm{GMST}(D_U,T) = 360\Big\{&86400\times\left[0.7790572732640 + 0.00273781191135448 D_U + \mathrm{mod}(D_U,1)\right]\\ &+0.00096707 + (307.47710227\times T) + (0.092772113\times T^2)\\ &-(0.0000000293\times T^3) - (0.00000199707\times T^4) - (2.453\times 10^{-9}\times T^5)\Big\}/86400\end{aligned} \tag{C26}$$

which has units of seconds.

Next, the Equation of the Equinoxes must be found using the equation given in the Astronomical Almanac (2010; page B10) but by

$$E_e(T) = \Delta\psi\cos(\varepsilon) \tag{C27}$$

where $\Delta\psi$ is the total nutation in longitude and ε is the mean obliquity of the ecliptic. The latter is given by Seidelmann (2006; Eq. 3.222-1) as

$$\varepsilon = \varepsilon_0 + \Delta\varepsilon \tag{C28}$$

where $\Delta\varepsilon$ is the nutation in obliquity and ε_0 is given by

$$\begin{aligned}e_0 = 84381.448 - (46.8150\times T) -\\ (0.00059\times T^2) + (0.001813\times T^3)\end{aligned} \tag{C29}$$

which is in units of arcseconds.

Seidelmann (2006) gave details of how to calculate the nutation terms (Δψ and Δε) based on the International Astronomical Union (IAU) 2000B standard that is a simplified version of the original IAU 2000 standard. The original standard has over 1000 terms, whereas the simplified version uses the largest 78 terms (McCarthy and Luzum, 2003). This reduced set still greatly exceeds the required accuracy for our purpose, so several assumptions were made to simplify the calculation. The coefficients given by Seidelmann (2006) are used here, but only terms with coefficients exceeding one arcsecond are kept. Additionally, only the main term for each coefficient is considered with the smaller time term neglected. The nutation in obliquity Δε is now given b

$$\Delta\varepsilon = 9.2025\sin(\Omega_m) + 0.5736\sin(2F_m - 2D_m + 2\Omega_m) \quad \text{(C30)}$$

and the nutation in longitude $\Delta\psi$ is given by

$$\Delta\psi = \Big[-17.1996\sin(\Omega_m) + 0.2062\sin(2\Omega_m) - 1.3187\sin(2F_m - 2D_m + 2\Omega_m) + 0.1426\sin(l_p) - 0.2274\sin(2F + 2\Omega_m)\Big]\Big/3600 \quad \text{(C31)}$$

which has units of degrees and where Ω_m is the longitude of the mean ascending node of the lunar orbit

$$\Omega_m = \Big[450160.28 - (6962890.539T) + (7.455T^2) + (0.008T^3)\Big]\Big/3600 \quad \text{(C32)}$$

F_m is the mean longitude of the Moon minus mean longitude of the Moon's node

$$F_m = \Big[335778.877 + (1739527263.137T) - (13.257T^2) + (0.011T^3)\Big]\Big/3600 \quad \text{(C33)}$$

D_m is the mean longitude of the Moon minus mean longitude of the Sun

$$D_m = \Big[1072261.307 + (1602961601.328T) - (6.891T^2) + (0.019T^3)\Big]\Big/3600 \quad \text{(C34)}$$

and l_p is the mean longitude of the Moon minus mean longitude of the Moon's perigee

$$l_p = \Big[1287099.804 + (129596581.224T) - (0.577T^2) - (0.012T^3)\Big]\Big/3600 \quad \text{(C35)}$$

(C32)-(C35) are taken from Seidelmann (2006; eq 3.222-6) and have units of degrees.

Greenwich Hour Angle is then given by

$$\text{GHA}(D_U, T) = \text{GMST}(D_U, T) + E_e(T) \quad \text{(C36)}$$

C4. Calculation of Sun position vector

The following Sun model is based on the model of van Flandern and Pulkkinen (1979). The Sun position vector in GCI coordinates is given by

$$S_i = d_s \begin{bmatrix} \cos l_{sa} \\ \sin l_{sa}\cos\varepsilon \\ \sin l_{sa}\sin\varepsilon \end{bmatrix} \quad \text{(C37)}$$

where l_{sa} is the apparent solar longitude in the ecliptic and ε is the true obliquity of the ecliptic as defined in (C28). The Earth to Sun distance r_s in astronomical units (au) is calculated from van Flandern and Pulkkinen (1979; Table 4, RP) as

$$r_s = 1.00014 - 0.01675 \cos g_s - 0.00014 \cos(2g_s) \quad \text{(C38)}$$

where g_s is the Sun mean anomaly in degrees

$$g_s = 360 \times [0.993126 + 0.00273777850 D_U] \quad \text{(C39)}$$

The Earth to Sun distance d_s in km is therefore

$$d_s = 149597870.660 r_s \quad \text{(C40)}$$

The apparent solar longitude in the ecliptic l_{sa} can be calculated as

$$l_{sa} = l_s + \Delta l_s + \Delta\psi - \frac{k}{r_s} \quad \text{(C41)}$$

where l_s is the mean solar longitude, Δl_s is the geometric correction to the mean solar longitude, $\Delta\psi$ is the nutation in longitude and k is constant of aberration where k=0.0056932. The mean solar longitude l_s is given by

$$l_s = 360 \times [0.779072 + 0.00273790931 D_U] \quad \text{(C42)}$$

The geometric correction to the mean solar longitude Δl_s is a combination of several factors: the mean longitude of the Moon

$$l_m = 360 \times [0.606434 + 0.03660110129 D_U] \quad \text{(C43)}$$

the mean anomaly for the Sun

$$g_s = 360 \times [0.993126 + 0.00273777850 D_U] \quad \text{(C44)}$$

the mean anomaly for Venus

$$g_2 = 360 \times [0.140023 + 0.00445036173 D_U] \quad \text{(C45)}$$

the mean anomaly for Mars

$$g_4 = 360 \times [0.053856 + 0.00145561327 D_U] \quad \text{(C46)}$$

and the mean anomaly for Jupiter

$$g_5 = 360 \times \left[0.056531 + 0.00023080893 D_U\right] \tag{C47}$$

Finally, the geometric correction to the mean solar longitude Δl_s is calculated from van Flandern and Pulkkinen (1979; Table 4, PLON) as

$$\begin{aligned} \Delta l_s = \Big[& 6910 \sin g_s + 72 \sin(2g_s) - 17T \sin g_s - 7\cos(g_s - g_5) \\ & + 6\sin(l_m - l_s) + 5\sin(4g_s - 8g_4 + 3g_5) - 5\cos(2g_s - 2g_2) \\ & - 4\sin(g_s - g_2) + 4\cos(4g_s - 8g_4 + 3g_5) + 3\sin(2g_s - 2g_2) \\ & - 3\sin g_5 - 3\sin(2g_s - 2g_5)\Big] \Big/ 3600 \end{aligned} \tag{C48}$$

5. References

Astronomical Almanac for the Year 2010, 2009. US Government Printing Office, Washington D.C.

Bate, R. R, Mueller, D. D. and White, J. E., 1971: *Fundamentals of Astrodynamics*. Dover Publications, New York. 455pp.

van Flandern, T. C. & Pulkkinen, K. F., 1979: Low-precision formulae for planetary positions. *Astrophysical Journal Supplement Series*, (**41**), 391-411.

Fernie, J. D., 1983: A one-line algorithm for Julian date. *International Amateur-Professional Photoelectric Photometry Communications*. **13**, 16-18.

Patt, F. S. and Gregg, W. W., 1994: Exact closed-form geolocation algorithm for Earth survey sensors. *International Journal of Remote Sensing*, **15**(18), 3719-3734. doi:10.1080/01431169408954354.

Seidelmann, P. K. 2006: *Explanatory Supplement to the Astronomical Almanac*. University Science Books, Sausalito, CA. 752pp.