Department of the Interior
U.S. Geological Survey

LANDSAT 8
OPERATIONAL LAND IMAGER (OLI) – THERMAL INFRARED SENSOR (TIRS) SOLAR AND VIEW ANGLE GENERATION ALGORITHM DESCRIPTION DOCUMENT (ADD)

Version 2.0

October 2018

science for a changing world

LANDSAT 8
OPERATIONAL LAND IMAGER (OLI) –
THERMAL INFRARED SENSOR (TIRS)
SOLAR AND VIEW ANGLE GENERATION
ALGORITHM DESCRIPTION DOCUMENT (ADD)

October 2018

Approved By:

CHRISTOPHER ENGEBRETSON

Digitally signed by CHRISTOPHER ENGEBRETSON
Date: 2018.10.30 15:45:06 -05'00'

C. Engebretson Date
Landsat DPAS CCB Chair
USGS

EROS
Sioux Falls, South Dakota

Executive Summary

The Landsat 8 Operational Land Imager (OLI) / Thermal Infrared Sensor (TIRS) Solar and View Angle Generation Algorithm Description Document (ADD) defines the algorithm used for the generation of solar and view angle bands, which are contained within the Landsat Collection 1 Level 1 (L1) data products created at the U.S. Geological Survey (USGS) Earth Resource Observation and Science (EROS) Center.

This document is under Landsat Data Processing and Archive System (DPAS) Configuration Control Board (CCB) control. Please submit changes to this document, as well as supportive material justifying the proposed changes, via Change Request (CR) to the Process and Change Management Tool.

Document History

Document Number	Document Version	Publication Date	Change Number
LSDS-1928	Version 1.0	May 2018	CR 13997
LSDS-1928	Version 2.0	October 2018	CR 14534

Contents

List of Figures

List of Tables

Section 1 Introduction

1.1 Background

Landsat 8 Level 1 Terrain Precision (Corrected) (L1TP) (and Level 1 Systemic Terrain (Corrected) (L1Gt)) products (henceforth referred to as L1X) provide radiometrically and geometrically corrected geolocated image samples for each spectral band. These samples are 16-bit fixed point numbers that can be related to either at-sensor radiance or reflectance using parameters provided in the product metadata. The L1X samples are also precisely registered to a Universal Transverse Mercator (UTM) (or polar stereographic) map projection grid, which makes it straightforward to construct pixel ground coordinates from the product corners.

For some applications, additional information about the scene geometry is needed, including elevation, slope/aspect, sensor viewing angles (zenith and azimuth), and/or solar illumination angles. This algorithm provides a method for generating per-pixel sensor viewing and solar illumination angles for L1X products by providing a new angle coefficient file, containing selected information from the geometric model and resampling grid, and associated logic for using the new file to compute the required angles.

The Operational Land Imager (OLI) and Thermal InfraRed Sensor (TIRS) payloads on the Landsat 8 mission are both push broom imagers with focal planes that span the full Landsat swath width. Full swath coverage is achieved by using multiple sensor chip assemblies (SCAs) across-track, with sufficient overlap between adjacent SCAs to avoid coverage gaps. This SCA-to-SCA overlap is achieved by displacing alternate SCAs along-track so that adjacent SCAs can cover overlapping portions of the across-track field of view. For the OLI, which uses 14 SCAs to cover the full swath, the 7 odd SCAs (1 through 13) are arranged to point slightly forward of nadir, and the 7 even SCAs (2 through 14) are arranged to point slightly aft. Similarly for TIRS, the central SCA-C points forward while the outboard SCAs (A and B) point aft. The layout of the OLI focal plane is shown in Figure 1-1 below.

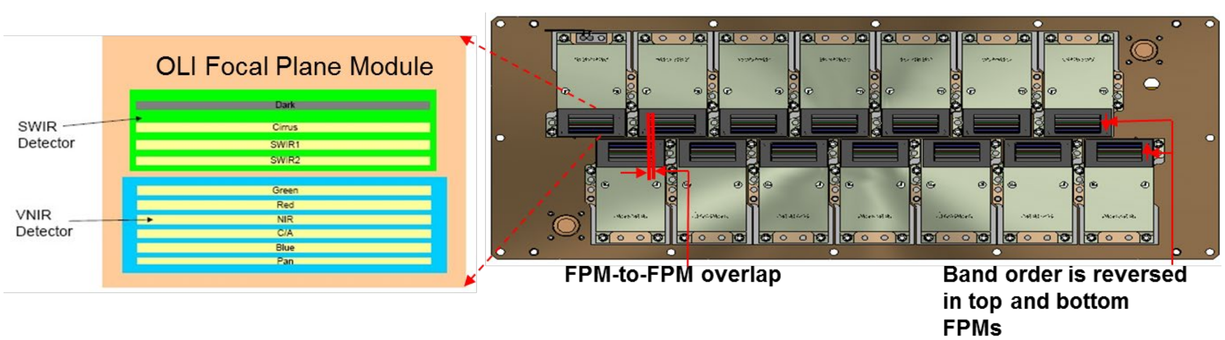

Figure 1-1. OLI Focal Plane Layout

Figure 1-1 raises a problem of terminology. Ball Aerospace uses the term SCA for just the detector and Read-Out Integrated Circuit (ROIC) chips (without filters) and refers to the complete unit as a focal plane module (FPM). The TIRS developers used SCA to refer to the entire assembly, while the FPM acronym had an entirely different meaning. The Calibration/Validation (Cal/Val) team decided to adopt SCA as the standard terminology for both instruments.

A key challenge in analyzing the viewing geometry for both the OLI and TIRS sensors is the along-track offset between adjacent SCAs, as this focal plane geometry leads to discontinuities in the viewing geometry at SCA boundaries. The view angle changes, occasioned by the alternating even/odd SCA geometry, would make it difficult to fit a simple function to the not-very-smooth angle patterns. This argues for generating and storing the view angles for each pixel. On the other hand, the along-track distribution of the spectral bands, also shown in Figure 1-1, ensures that the viewing angles will be different for each spectral band. Explicitly representing the angles for each pixel in each band would be space prohibitive as the angle file would be larger than the L1X product. These considerations led to a compromise solution, described herein, that uses multiple rational polynomial functions to model the viewing geometry for each band on each SCA. These functions are implemented in an exploitation tool that uses scene-specific parameters, stored in an angle coefficient file provided with each L1X product, to generate viewing angles on demand.

The algorithm is implemented in two parts. The first part, intended to run in the Image Assessment System (IAS) / Landsat Product Generation System (LPGS) environment at product generation time, uses the geometric model and grid files used to create the L1X product to build an additional angle coefficient file that accompanies the product. This new file captures the elements of the scene geometry needed to subsequently calculate the solar illumination and sensor viewing angles for each active product pixel (i.e., those that contain OLI or TIRS image data). The second part of the algorithm uses the new angle coefficient file to compute these angles. This part is implemented as a standalone software tool that is available to the user community (see https://landsat.usgs.gov/solar-illumination-and-sensor-viewing-angle-coefficient-file for more details). Two versions of this tool were developed: an experimental version that supports multiple processing options and a simpler operational prototype that provides only basic angle generation capability.

1.2 Purpose

The primary purpose of this document is to provide technical details and information on the Landsat 8 OLI/TIRS Solar and View Angle Generation Algorithm.

1.3 Document Organization

This document contains the following sections:

- Section 1 introduces the Landsat 8 OLI/TIRS Solar and View Angle Generation Algorithm.
- Section 2 provides technical details on the dependencies, inputs, and outputs.
- Section 3 describes the algorithm procedure.
- Appendix A comprises a list of acronyms

- The References section contains a list of reference documents and supporting webpages.

Section 2 Dependencies, Inputs, Outputs

2.1 Dependencies

The angle generation algorithm assumes that the standard L1X geometric modeling algorithms have run successfully and that the geometric model, geometric grid, and calibration parameter files used to create the L1X product are available. The angle computation algorithm can optionally use an input elevation model for improved accuracy. If provided, this model must match the scene frame (corners, projection, pixel size) of the L1X product multispectral bands.

2.2 Algorithm Inputs

Table 2-1 lists the Solar and View Angle Generation Algorithm and its component sub-algorithms.

Algorithm Inputs – Angle Coefficient File Generation
Geometric Grid File
Scene Framing Information:
Scene corner coordinates
Scene map projection information:
Projection General Cartographic Transformation Package (GCTP) code: 1 = Universal Transverse Mercator (UTM), 6 = Polar Stereographic
UTM zone number (1-60)
GCTP map projection parameters
Datum and spheroid codes (World Geodetic System 1984 (WGS84))
Geometric Model File
Worldwide Reference System (WRS) path and row (orbital, for use in output file name construction)
Image times
Spacecraft ephemeris (position vs. time)
Calibration Parameter File
Earth model parameters
WGS84 ellipsoid parameters
Earth orientation parameters (UTC Corrected (UT1) – Universal Time Code (UTC) offset, pole wander)
Leap second table
Naval Definition Vector Astronomy Software (NOVAS) solar ephemeris (sun Earth Centered Inertial True of Date (ECITOD) direction vs. time)
Algorithm Inputs – Angle Computation
Angle Coefficient File – see output table below for contents
Digital Elevation Model (DEM) File (optional)
WGS84 ellipsoid height (in meters) for each 30-meter pixel in the L1X product
Subsampling Factor (optional)

Table 2-1. Solar and View Angle Generation Algorithm Inputs

2.3 Algorithm Outputs

Table 2-2 shows the Solar and View Angle Generation Algorithm outputs. See Table 3-1 below for more detail contents of the output angle coefficient file.

Algorithm Outputs – Angle Coefficient File Generation
Angle Coefficient File
File Header
Angle coefficient file name
Satellite ID
WRS path and row (orbital)
List of bands included
Projection Information
Ellipsoid parameters
Projection type/code
Projection units (meters)
Projection spheroid and datum (WGS84)
UTM zone number
GCTP projection parameters
L1X product projection corners
Ephemeris Data
UTC epoch (year, day of year, seconds of day)
Number of ephemeris points
Time from epoch for each point
Earth-Centered, Earth-Fixed (ECEF) X, Y, and Z position for each point
Solar Vector Data
UTC epoch (year, day of year, seconds of day)
Number of solar vectors provided
Time from epoch for each vector
ECEF X, Y, and Z directions for each vector
Rational Polynomial Coefficient Data for each Band
Number of SCAs
Number of lines and samples in L1X product
Number of lines and samples in Level 1 Radiometric (Corrected) (L1R) input (full scene)
L1X pixel size (in meters)
Image start time relative to ephemeris epoch
Image line time (time between lines)
Mean height in scene
Mean L1R line/sample coordinates in scene
Mean L1X line/sample coordinates in scene
Mean satellite viewing vector components (local east-north-vertical coordinates)
Rational polynomial numerator coefficients for the viewing vector X component
Rational polynomial denominator coefficients for the viewing vector X component
Rational polynomial numerator coefficients for the viewing vector Y component
Rational polynomial denominator coefficients for the viewing vector Y component
Rational polynomial numerator coefficients for the viewing vector Z component
Rational polynomial denominator coefficients for the viewing vector Z component
Mean solar illumination vector components (local east-north-vertical coordinates)
Rational polynomial numerator coefficients for the solar vector X component
Rational polynomial denominator coefficients for the solar vector X component

Rational polynomial numerator coefficients for the solar vector Y component
Rational polynomial denominator coefficients for the solar vector Y component
Rational polynomial numerator coefficients for the solar vector Z component
Rational polynomial denominator coefficients for the solar vector Z component
List of SCAs in current band
Rational Polynomial Coefficient Data for each SCA
Mean height in SCA
Mean L1R line/sample coordinates in SCA
Mean L1X line/sample coordinates in SCA
Rational polynomial numerator coefficients for L1R line coordinate
Rational polynomial denominator coefficients for L1R line coordinate
Rational polynomial numerator coefficients for L1R sample coordinate
Rational polynomial denominator coefficients for L1R sample coordinate
Algorithm Outputs – Angle Computation
Satellite Viewing Angle File for each Band
Viewing zenith angle for each L1X pixel (unless subsampled)
Viewing azimuth angle for each L1X pixel (unless subsampled)
Zenith and azimuth “bands” are sequential
Zenith and azimuth angles are stored as 16-bit integers scaled to units of 0.01 degrees
Satellite Viewing Angle (Environment for Visualizing Images (ENVI)) Header File (one per angle file)
Number of lines and samples in angle file
Number of bands in angle file (2)
Data type (signed 16-bit integer)
Interleaving type (Band Sequential (BSQ))
Projection information
Projection type (UTM or Polar Stereographic (PS))
UTM zone/PS projection parameters
Output angle file pixel size (in meters) = L1X pixel size * subsampling factor
Upper Left (UL) corner coordinates
Solar Angle File for each Band
Solar zenith angle for each L1X pixel (unless subsampled)
Solar azimuth angle for each L1X pixel (unless subsampled)
Zenith and azimuth “bands” are sequential
Zenith and azimuth angles are stored as 16-bit integers scaled to units of 0.01 degrees
Solar Angle (ENVI) Header File (one per angle file)
Number of lines and samples in angle file
Number of bands in angle file (2)
Data type (signed 16-bit integer)
Interleaving type (BSQ)
Projection information
Projection type (UTM or PS)
UTM zone/PS projection parameters
Output angle file pixel size (in meters) = L1X pixel size * subsampling factor
UL corner coordinates

Table 2-2. Solar and View Angle Generation Algorithm Outputs

2.4 Angle Computation Options

Angle computation can work with or without elevation data input.

The output angle “bands” can be optionally subsampled.

Section 3 Procedure

The primary tasks performed by the solar and view angle generation algorithm are to:

1. At product generation time: create an angle coefficient file that contains all of the information a user needs to calculate per-pixel solar illumination and sensor viewing angles for each band in the L1X data product.
2. On demand for the user: use the angle coefficient file to generate solar illumination and sensor viewing angles that correspond to the L1X product pixels.

3.1 Phase 1: Generate Angle Coefficient File

Central to the ability to compute the satellite viewing or solar illumination geometry for a particular L1X image pixel is the ability to associate that pixel with its time of observation. Once the time is known, it can be used to calculate the spacecraft position, from which the sensor viewing geometry is derived, and the solar direction, from which we calculate sun angles. The key to mapping output product image pixels to imaging time is to reconstruct the relationship between the resampled L1X product pixels and the un-resampled L1R calibrated detector samples from which they are derived, since there is a simple linear relationship between L1R line number and time. The L1R to L1X mapping can be calculated from the geometric model. To facilitate efficient L1X product generation, this relationship is stored in the geometric grid file for an array of points spanning the image bounds. The goal here is to formulate a set of equations that represent, in a compact form, the input space (Level 1R) line/sample to output space (Level 1T) line/sample mappings contained in the geometric grid file. Experimentation has shown that sub-pixel accuracy in the L1X line/sample to L1R line/sample mapping can be achieved using rational polynomial functions of the following form:

$$L1R_{Line} = L1R_{MeanLine} + \frac{(a_0 + a_1 * L1T_L + a_2 * L1T_S + a_3 * Hgt + a_4 * L1T_L * L1T_S)}{(1 + b_1 * L1T_L + b_2 * L1T_S + b_3 * Hgt + b_4 * L1T_L * L1T_S)} \tag{1}$$

$$L1R_{Sample} = L1R_{MeanSample} + \frac{(c_0 + c_1 * L1T_L + c_2 * L1T_S + c_3 * Hgt + c_4 * L1T_L * L1T_S)}{(1 + d_1 * L1T_L + d_2 * L1T_S + d_3 * Hgt + d_4 * L1T_L * L1T_S)} \tag{2}$$

Where:

$L1X_L = L1X_{Line} - L1X_{MeanLine}$

$L1X_S = L1X_{Sample} - L1X_{MeanSample}$

$Hgt = Height - Height_{Mean}$

a0 to a4, b1 to b4, c0 to c4, and d1 to d4 are model coefficients.

One set of rational polynomial coefficients (RPCs) is computed for each band on each SCA using the information in the geometric grid file. For OLI, there are 23 model parameters (18 polynomial coefficients plus 5 mean offsets) per band/SCA. With 14 SCAs and 9 bands, this results in a total of 2898 model constants per scene. For TIRS, there are 23 model parameters/band/SCA * 3 SCAs * 2 bands for a total of 138 additional model constants. Note that five model parameters, which are the mean values of the input and output coordinates, are added to reference the rational polynomial formulation to the center of the band/SCA area covered by the functions. This helps provide numerical stability in the least squares solution for the model coefficients.

3.1.1 Calculate the Model Coefficients

The geometric grid file contains a set of three-dimensional arrays of L1R to L1X pixel mappings, one for each band on each SCA. The array axes are L1R line, L1R sample, and (ellipsoid) height, with each array point corresponding to one L1R line / L1R sample / height triplet. This is depicted in Figure 3-1. The L1X line/sample location corresponding to each triplet is computed using the line-of-sight projection model and the selected output L1X scene frame. The results are stored in the grid structure for subsequent use during image resampling. The grid provides all of the information required to solve for the rational polynomial model coefficients.

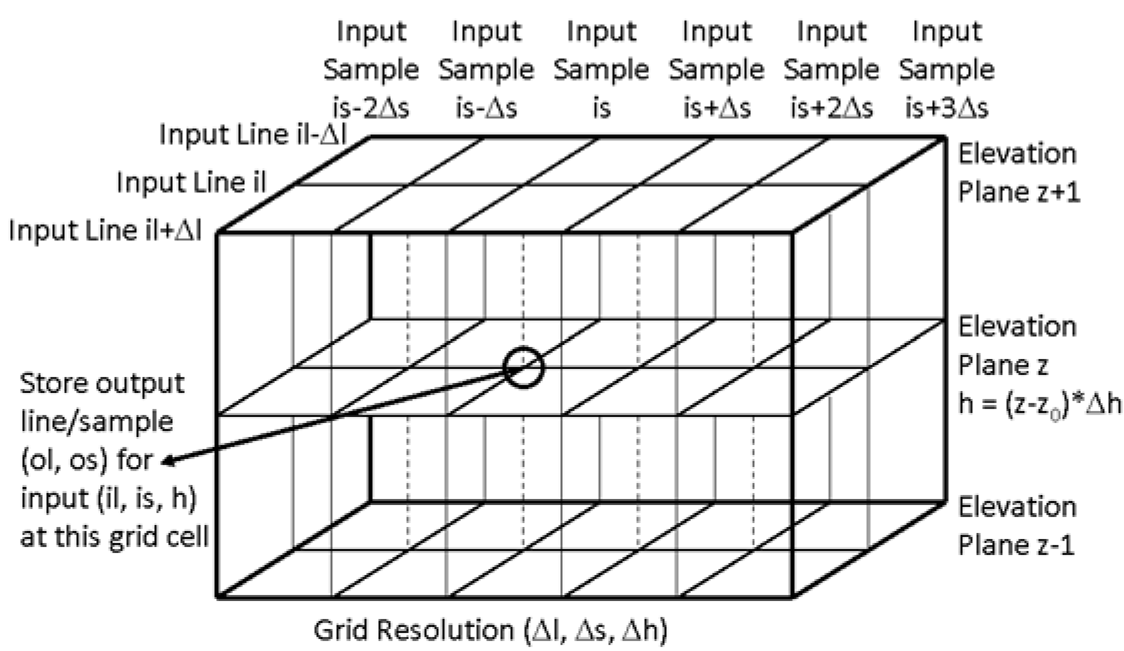

Figure 3-1. Geometric Grid Structure

All of the grid points for a given band/SCA are used to solve for the model coefficients for that band/SCA. Calculating the mean L1R line, L1R sample, L1X line, L1X sample, and height values is straightforward. The model coefficients are determined by a least squares solution. To accomplish this, the rational functions are linearized by multiplying the denominator by the left-hand side and rearranging terms as follows:

$$L1R_L = (a_0 + a_1 * L1T_L + a_2 * L1T_S + a_3 * Hgt + a_4 * L1T_L * L1T_S) - (b_1 * L1T_L * L1R_L + b_2 * L1T_S * L1R_L + b_3 * Hgt * L1R_L + b_4 * L1T_L * L1T_S * L1R_L) \quad (3)$$

$$L1R_S = (c_0 + c_1 * L1T_L + c_2 * L1T_S + c_3 * Hgt + c_4 * L1T_L * L1T_S) - (d_1 * L1T_L * L1R_S + d_2 * L1T_S * L1R_S + d_3 * Hgt * L1R_S + d_4 * L1T_L * L1T_S * L1R_S) \quad (4)$$

Where:

$L1R_L = L1R_{Line} - L1R_{MeanLine}$

$L1R_S = L1R_{Sample} - L1R_{MeanSample}$

One pair of equations of this form can be constructed for each grid point. Standard least squares techniques are used to solve for the nine coefficients in each equation.

3.1.2 Construct the Angle Coefficient File

The sequence of activities required to assemble the information required to build the angle coefficient file is as follows:

1. Open and read the input data files:
    a. Load the geometric model from the Line of Sight (LOS) model file.
    b. Load the geometric grid from the grid file.
    c. Load the Earth model parameters from the Calibration Parameter File (CPF).
    d. These operations are accomplished using standard IAS library input/output modules.
2. Initialize the Earth model:
    a. Initialize the IAS time conversion library functions using the leap second table read from the CPF.
    b. Store the WGS84 ellipsoid semi-major and semi-minor axes from the CPF in the angle coefficient (ANGCF) structure.
3. Get path/row, ephemeris, and sun vector information from the geometric model.
    a. Store the WRS path and row from the model in the ANGCF structure.

b. Extract ephemeris data covering the current scene from the geometric model.

    i. Using the ephemeris start time (UTC epoch) as a reference, calculate the time offsets to the first and last line in each band by invoking the ias_math_get_time_difference utility.

    ii. Determine the earliest band start time (this will normally be the TIRS bands) and latest band end time.

    iii. Subtract 8 seconds from the earliest start time and add 8 seconds to the latest end time to get the target time bounds for extracting ephemeris data to cover the scene.

    iv. Find the index of the first ephemeris point with a time after the target start time and the index of the last ephemeris point with a time before the target end time.

    v. Establish a new ephemeris epoch at the time of the first sample to be extracted.

    vi. Load the time (adjusted for the new epoch) and ECEF position fields for the selected ephemeris points from the model into the ANGCF structure.

c. Use NOVAS to compute ECEF sun vectors at the ephemeris sample times.

    i. Initialize the NOVAS solar ephemeris package.

    ii. For each ephemeris point:

        1. Construct the full UTC time by adding the point's time offset to the ephemeris UTC epoch.

        2. Convert the UTC year and day of year to month and day.

        3. Use the year, month, day, and seconds of day to compute the Julian day required by NOVAS.

        4. Invoke NOVAS to compute the Earth Centered Inertial (ECI) true-of-date solar direction vector at the specified Julian day.

        5. Use IAS library routines to convert the ECITOD sun vector to ECI of epoch J2000 (by applying nutation and precession

models). Note that the IAS library ECEF/ECI coordinate transformation routines also invoke NOVAS.

6. Use IAS library routines to convert the ECIJ2000 sun vector to ECEF, including the pole wander and UT1-UTC corrections from the geometric model.

7. Load the time and ECEF solar unit vector into the ANGCF structure.

iii. Shut down the NOVAS package.

4. Get map projection and scene corner information from the geometric grid.

    a. Load the projection code, units, zone, spheroid, datum, and GCTP map projection parameter fields from the grid into the ANGCF structure. These parameters are needed to convert map X/Y to geodetic latitude/longitude.

    b. Load the scene corner map projection coordinates from the grid into the ANGCF structure. The corners are needed to convert L1X line/sample to map projection X/Y.

5. Initialize the IAS library map projection logic.

    a. Construct a map projection structure using the parameters loaded in the ANGCF structure, by invoking the ias_geo_set_projection module.

    b. Construct a geodetic projection structure to produce latitude/longitude coordinates in radians, using the ias_geo_set_projection module.

    c. Construct a projection transformation that converts map X/Y to geodetic latitude/longitude using the structures from a. and b. above, and the ias_geo_create_proj_transformation module.

    d. Pre-establishing this transformation will make subsequent map projection conversion computations easier.

6. Assemble the band-specific angle coefficient structure fields for each band.

    a. Load the band number, number of SCAs, number of L1X lines/samples, and pixel size from the grid into the ANGCF structure.

    b. Load the number of L1R lines/samples, band start time, and line increment time (sampling time) from the geometric model into the ANGCF structure.

c. For each SCA, record the SCA number in the ANGCF structure and calculate the coefficients of the L1X-to-L1R RPC model described above.

    i. Compute the mean height of the grid points as:

       Mean_hgt = (num_Zplanes – 1 – 2*zeroplane)*Zspacing/2

    ii. Compute the mean L1R line and sample values by cycling through the grid in_lines and in_samps arrays.

    iii. Compute the mean L1X line and sample values by cycling through the grid point out_lines and out_samps arrays.

    iv. Loop through all the points in the grid for this band/SCA to construct the normal equations:

        1. The form of the observations is shown in equations (3) and (4) above. Each grid point yields one line and one sample observation, expressed in matrix/vector notation:

           $X_L^T \theta_L = Y_L \qquad\qquad X_S^T \theta_S = Y_S$

           Where:

$$X_L = \begin{bmatrix} 1 \\ L1T_L \\ L1T_S \\ Hgt \\ L1T_L * L1T_S \\ -L1R_L * L1T_L \\ -L1R_L * L1T_S \\ -L1R_L * Hgt \\ -L1R_L * L1T_L * L1T_S \end{bmatrix} \qquad \theta_L = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \qquad Y_L = [L1R_L]$$

$$X_S = \begin{bmatrix} 1 \\ L1T_L \\ L1T_S \\ Hgt \\ L1T_L * L1T_S \\ -L1R_S * L1T_L \\ -L1R_S * L1T_S \\ -L1R_S * Hgt \\ -L1R_S * L1T_L * L1T_S \end{bmatrix} \qquad \theta_S = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} \qquad Y_S = [L1R_S]$$

        2. Each observation of each type (line or sample) contributes to the normal equations:

$N_L\theta_L = L_L$ and $N_S\theta_S = L_S$

Where we have accumulated the N and L matrices as:

$N_L \mathrel{+}= X_L X_L^T$ $\qquad$ $L_L \mathrel{+}= X_L Y_L$

$N_S \mathrel{+}= X_S X_S^T$ $\qquad$ $L_S \mathrel{+}= X_S Y_S$

v. Once the observation contributions from all of the grid points are collected into the normal equation matrices, we solve for the unknown rational polynomial coefficient vectors θ_L and θ_S:

$\theta_L = N_L^{-1} L_L$ $\qquad$ $\theta_S = N_S^{-1} L_S$

vi. This procedure generates a set of L1X-to-L1R RPCs for each SCA in the band.

The procedure described thus far provides everything necessary to support the computation of the required view and sun angles: a mechanism for relating L1X product line/sample to L1R line/sample, which yields time of observation; the ECEF position of the spacecraft as a function of time; the ECEF direction to the sun as a function of time; and the scene framing and projection information needed to convert L1X line/sample to map X/Y, then to geodetic (optionally including height from an input DEM), and finally to ECEF. Though feasible, this approach requires the application of multiple complex coordinate transformations for every pixel in the L1X product. Experiments with this approach to generating view and sun angles demonstrated that, while angle accuracies of 1 arc-minute or better can be achieved if terrain data are included, the required computations are rather time consuming. On the IAS development platform, generating satellite viewing and solar illumination zenith and azimuth angles for every imaged pixel in all bands required 40-45 minutes of processing time. In an effort to reduce this processing time, a more computationally efficient alternative was developed.

3.1.3 Rapid Angle Computation

In the alternate approach, a second-tier rational polynomial model is fitted directly to the satellite and sun unit viewing vectors making it possible to compute them directly. This circumvents the need for complex map projection and geodetic computations involving trigonometric functions. Unit vector components, rather than the angles themselves, are fitted to avoid the +/-180-degree azimuth discontinuity.

The second-tier "angle" rational polynomial functions, one set per band, are more complicated than the first-tier per-SCA L1X-to-L1R rational functions, because they must account for the SCA-to-SCA discontinuities. This is achieved by including both L1X and L1R input terms in the formulation. The RPC model equation for the satellite viewing unit vector X component is:

$$Sat_X = Sat_{XMean} + \frac{Num_x(L1T_L, L1T_S, Hgt, L1R_L, L1R_S)}{Den_x(L1T_L, L1T_S, Hgt, L1R_L, L1R_S)} \qquad (5a)$$

$$\begin{aligned} Num_x(L1T_L, L1T_S, Hgt, L1R_L, L1R_S) \\ &= a_0 + a_1 * L1T_L + a_2 * L1T_S + a_3 * Hgt + a_4 * L1R_L + a_5 * L1T_L \\ &* L1T_L + a_6 * L1T_L * L1T_S + a_7 * L1T_S * L1T_S + a_8 * L1R_S * L1R_L \\ &* L1R_L + a_9 * L1R_L * L1R_L * L1R_L \qquad (5b) \end{aligned}$$

$$\begin{aligned} Den_x(L1T_L, L1T_S, Hgt, L1R_L, L1R_S) \\ &= 1 + b_1 * L1T_L + b_2 * L1T_S + b_3 * Hgt + b_4 * L1R_L + b_5 * L1T_L * L1T_L \\ &+ b_6 * L1T_L * L1T_S + b_7 * L1T_S * L1T_S + b_8 * L1R_S * L1R_L * L1R_L + b_9 \\ &* L1R_L * L1R_L * L1R_L \qquad (5c) \end{aligned}$$

Where:

$L1X_L = L1X_{Line} - L1X_{MeanLine}$

$L1X_S = L1X_{Sample} - L1X_{MeanSample}$

$Hgt = Height - Height_{Mean}$

$L1R_L = L1R_{Line} - L1R_{MeanLine}$

$L1R_S = L1R_{Sample} - L1R_{MeanSample}$

a_0 to a_9, and b_1 to b_9, are the RPC model coefficients.

There are similar models for Sat_Y, Sat_Z, Sun_X, Sun_Y, and Sun_Z.

The terms included in these equations were determined by experimentation to minimize the rational polynomial model fit residuals. Note that, in order to use these models, it is necessary to first evaluate the L1X-to-L1R RPC model to determine the values for $L1R_L$ and $L1R_S$.

The final steps in the assembly of the angle coefficient file are to compute these “angle” rational polynomial model coefficients, using a procedure much like that described in step #6 above, and to write out the angle coefficient Object Description Language (ODL) file:

7. Calculate the direct angle RPCs for each band.
    a. Calculate the satellite viewing vector and the solar illumination vector in the local vertical coordinate system at each point in the geometric grid.
        i. Extract the height (from the grid Z-plane), L1R line/sample, and L1X line/sample for the current point from the grid structure.
        ii. Use the L1X corners and pixel size to convert L1X line/sample to map X/Y:
        X = upleft_X + $L1X_S$ * pixel_size

        Y = upleft_Y – $L1X_L$ * pixel_size

Note that this assumes projection north-up products, which all L8 L1X products currently are. This could be made more elaborate to support path-oriented products if necessary, since all four scene corners are included in the angle coefficient file.

iii. Use the (already initialized) map projection transformation to convert map X/Y to latitude/longitude using IAS library functions. These functions implement the map projection algorithms documented in, “Map Projections – A Working Manual” by John P. Snyder, USGS Professional Paper 1395, U.S. Government Printing Office, Washington, DC, 1987.
iv. Convert geodetic latitude, longitude, and height (from the grid) to ECEF X, Y, Z using IAS library functions. The details of the geodetic to ECEF transformation are described in the OLI Line-of-Sight Projection/Grid Generation Algorithm Description Document. This yields the ground point ECEF vector G_{ECEF}.
v. Calculate the local vertical coordinate system basis vectors from the latitude and longitude:

$$H_x = \begin{bmatrix} -\sin(lon) \\ \cos(lon) \\ 0 \end{bmatrix} \qquad \text{East in ECEF}$$

$$H_y = \begin{bmatrix} -\sin(lat) * \cos(lon) \\ -\sin(lat) * \sin(lon) \\ \cos(lat) \end{bmatrix} \qquad \text{North in ECEF}$$

$$H_z = \begin{bmatrix} \cos(lat) * cos(lon) \\ \cos(lat) * \sin(lon) \\ \sin(lat) \end{bmatrix} \qquad \text{Up in ECEF}$$

vi. Calculate the time of observation from the L1R line:
Time = Band Start Time + $L1R_L$ * Line Time

vii. Interpolate the spacecraft ECEF X, Y, Z position at the time of observation using 4-point Lagrange interpolation. This is implemented using IAS library functions, and it yields the spacecraft ECEF vector S_{ECEF}.
viii. Calculate the ground-to-space viewing vector:
$V_{ECEF} = S_{ECEF} - G_{ECEF}$

ix. Project the ECEF viewing vector into the local vertical coordinate system:

$$V_{LV} = \begin{bmatrix} H_x^T \\ H_y^T \\ H_z^T \end{bmatrix} V_{ECEF}$$

This is equivalent to taking the dot product of the ECEF viewing vector with each of the local vertical system basis vectors.

x. Interpolate the ECEF sun direction vector at the time of observation using 4-point Lagrange interpolation. This is the same functionality used for the ephemeris data.

xi. Project the sun direction ECEF vector into the local vertical coordinate system, as was done in step ix above, to yield the local vertical sun direction vector S_{LV}.

b. As each grid point is processed, accumulate the sums of and then calculate the average values for the height, L1R line, L1R sample, L1X line, L1X sample, view vector X, Y, Z coordinates, and sun vector X, Y, Z coordinates.

c. For each component of the viewing vector V_{LV} and the sun vector S_{LV}, compute the coefficients of a RPC model of the form shown above in equations 5a, 5b, and 5c.

   i. Loop through all the vectors (each corresponding to a point in the grid) for this band to construct the normal equations:

      1. The form of the observations is shown in equations (5a), (5b), and (5c) above. Each grid point yields one observation, expressed in matrix/vector notation:

         $$X_{Satx}^T \theta_{Satx} = Y_{Satx}$$

         Where:

$$X_{Satx} = \begin{bmatrix} 1 \\ L1T_L \\ L1T_S \\ Hgt \\ L1R_L \\ L1T_L * L1T_L \\ L1T_L * L1T_S \\ L1T_S * L1T_S \\ L1R_S * L1R_L * L1R_L \\ L1R_L * L1R_L * L1R_L \\ -Sat_X * L1T_L \\ -Sat_X * L1T_S \\ -Sat_X * Hgt \\ -Sat_X * L1R_L \\ -Sat_X * L1T_L * L1T_L \\ -Sat_X * L1T_L * L1T_S \\ -Sat_X * L1T_S * L1T_S \\ -Sat_X * L1R_S * L1R_L * L1R_L \\ -Sat_X * L1R_L * L1R_L * L1R_L \end{bmatrix} \qquad \theta_{Satx} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \\ b_9 \end{bmatrix}$$

$$Y_{Satx} = [Sat_X]$$

Note that all of the L1X, L1R, height, and vector component inputs above are offset by the means as shown in equation (5).

2. Each observation contributes to the normal equations:

$$N_{Satx}\theta_{Satx} = L_{Satx}$$

Where we have accumulated the N and L matrices as:

$$N_{Satx} += X_{Satx}X_{Satx}^T \qquad L_{Satx} += X_{Satx}Y_{Satx}$$

ii. Once the observation contributions from all of the grid points are collected into the normal equation matrices, we solve for the unknown rational polynomial coefficient vectors θ_L and θ_S:

$$\theta_{Satx} = N_{Satx}^{-1}L_{Satx}$$

d. This procedure is run six times, on the X, Y, and Z components of the view vector (Satx, Saty, and Satz) and on the X, Y, and Z components of the sun vector (Sunx, Suny, and Sunz). This generates a set of L1X-to-angle RPCs for the band.

e. Load the angle RPC model coefficients into the ANGCF structure.

8. Write the ANGCF structure to an output ODL formatted angle coefficient file.
    a. Construct the output file name from the path, row, and date. This will need to be enhanced to include ground station ID and version number for operational use.
    b. Write the FILE_HEADER group containing the file name, satellite ID, path, row, number of bands, and band list.
    c. Write the PROJECTION group containing the ellipsoid parameters, projection, units, datum, spheroid, and zone codes, the GCTP projection parameters, and the scene corner coordinates.
    d. Write the EPHEMERIS group containing the ephemeris data start UTC epoch, the number of points, and the time from epoch, ECEF X, Y, and Z coordinates (in meters) for each point.
    e. Write the SOLAR_VECTOR group containing the start UTC epoch, the number of points, and the time from epoch, ECEF X, Y, and Z directions for each point. The times will match the ephemeris data, so those values are somewhat redundant.
    f. Write an RPC_BAND group for each band containing the number of SCAs, SCA list, number of L1X lines and samples, number of L1R lines and samples, pixel size, band start UTC epoch and line time increment, mean height, mean L1X line/sample, mean L1R line/sample, mean view vector components, view vector RPC model coefficients, mean sun vector components, and mean sun vector RPC model coefficients.
        i. For each SCA in the band, write the mean height, mean L1X line/sample, mean L1R line/sample, and the L1X-to-L1R RPC model coefficients.

Calculate the satellite viewing vector and the solar illumination vector in the local vertical coordinate system at each point in the geometric grThe output angle coefficient file contains all of the information needed by the phase 2 portion of the algorithm to generate satellite viewing and solar illumination angles for each L1X product pixel.

3.2 Phase 2 – Compute Satellite Viewing and Solar Illumination Angles

The phase 2 experimental angle generation portion of the algorithm (simple_view) provides two options for performing the angle computations. Both methods use the L1X-to-L1R RPC models to calculate the L1R coordinates that correspond to a given L1X pixel. Both methods will also retrieve the pixel height from an input DEM, if provided. Otherwise, the mean elevation for the band, from the angle coefficient file, is used. Having been given the L1X line/sample and determined the corresponding L1R line/sample and height, the first, "rigorous", method follows the procedure described above in step 7a of the phase 1 algorithm. The second, “RPC”, method applies equation (5) above using the parameters of the angle RPC model for the current band, stored in the angle coefficient file. The simpler operational prototype version (l8_angles) supports only the RPC and non-DEM options.

There are some subtleties to the use of the L1X-to-L1R RPCs that deserve some elaboration. Although the rational polynomial functions will generate L1R line and sample coordinates for any given L1X product pixel (and height), each SCA has a separate set of RPCs, so it is necessary to know which SCA the pixel falls inside to select the correct model parameters. Having only the L1X image to work with, we will know which band number to use, but not which SCA. This was the original reason for including an L1R sample RPC model. We can evaluate the L1R sample coordinates for each SCA in the current band to decide which SCA, or SCAs, the L1X pixel came from. At most 2 SCAs will return L1R sample values that fall within the actual range of samples on that SCA, identifying the set of rational polynomial coefficients to use to evaluate the L1R line coordinate. In SCA overlap areas, two SCAs will be valid. Depending upon the accuracy required, either could be used to compute the time and angles. In practice, both are evaluated and averaged, since overlapping pixels are averaged when the L1X products are generated.

3.2.1 Calculate Viewing and Solar Angles

The sequence of activities required to generate satellite viewing angles and solar illumination angles for each L1X pixel using the angle coefficients is as follows:

1. Capture the input command line parameters to determine which processing options to apply: RPC or rigorous computation, DEM input or mean height, subsampling factor.

2. Initialize the angle coefficient interface using the inputs provided:

    a. Open the input angle coefficient file and load the contents into an ANGCF data structure.

    b. Initialize the map projection logic:

        i. Construct a map projection structure using the parameters loaded in the ANGCF structure, by invoking the ias_geo_set_projection module.

        i. Construct a geodetic projection structure to produce latitude/longitude coordinates in radians, using the ias_geo_set_projection module.

        ii. Construct a projection transformation that converts map X/Y to geodetic latitude/longitude using the structures from a. and b. above, and the ias_geo_create_proj_transformation module.

    iii. Pre-establishing this transformation will make subsequent map projection conversion computations easier.

c. Load the Hierarchical Data Format (HDF) formatted DEM, if one is provided.

    i. Read the header information and store in a data structure.

    ii. Load the elevation array.

    iii. If no DEM is provided or the DEM does not match the image dimensions specified in the angle coefficient file, set the elevation array to NULL.

d. Get the number of bands from the ANGCF structure and return this value to the calling procedure.

3. For each band:

    a. Extract the scene framing information from the ANGCF structure.

        i. Extract the spectral band number, scene dimensions, map projection information (code, zone), pixel size, and upper-left corner coordinates from the ANGCF structure. The projection information will be used to generate the output angle image header files.

    b. Calculate the size of the output angle images using the size of the L1X image and the subsampling factor:

    Angle nlines = (L1X nlines – 1) / subsample + 1

    Angle nsamps = (L1X nsamps – 1) / subsample + 1

    c. Step through the L1X image pixels using the subsampling factor as a loop increment. Calculate the view and sun angles at each L1X line/sample location:

        i. Select the angle coefficients for the current band.

        ii. Calculate the subsampling relationship between the L1X image and the DEM:

        DEM subsample = (L1X nlines – 1) / (DEM nlines – 1)

        This is needed to properly index the DEM elevations when processing the panchromatic band.

iii. Calculate the DEM indices that correspond to the current L1X indices by dividing by the DEM subsample factor, noting that the indices are zero-relative.

iv. Extract the height from the DEM at the specified indices. If no DEM was provided, set the height to NULL.

v. Calculate the angles using the selected method:

1. Rigorous method – see below for details.

2. RPC method – see below for details.

vi. Quantize the computed angles to units of 0.01 degrees.

d. Write the angles to output band files:

i. Calculate the angle band pixel size by multiplying the L1X pixel size by the subsampling factor.

ii. Construct the output file names using the angle coefficient input file root name and the band number.

iii. Write the satellite zenith and azimuth angle values, band sequentially, to the satellite angle file.

iv. Write an ENVI-format header file for the satellite angles using the framing information extracted previously.

v. Write the solar zenith and azimuth angles, band sequentially, to the solar angle file.

vi. Write an ENVI-format header file for the solar angles using the framing information extracted previously.

4. Shut down the angle coefficient logic by releasing the allocated ephemeris data memory in the ANGCF structure and in the map projection transformation structure.

3.2.2 Compute Angles Using the Rigorous Method

To compute the satellite viewing and solar illumination angles at a specified L1X line/sample location, given the corresponding elevation and angle coefficients, use the rigorous method:

1. If the input height is NULL, replace it with the mean height from the band RPC parameters in the ANGCF structure.

2. Determine which SCA, or SCAs, viewed the L1X pixel:
    a. If the last_sca flag is invalid (e.g., for the first point calculated), start with the central SCA (isca = num_sca /2), otherwise use isca = last_sca.
    b. If the current SCA number (isca) is not valid (< 0 or >= num_SCA), return the number of valid SCAs found so far. Otherwise, compute the L1X-to-L1R RPCs for the current SCA:
        i. Offset the input L1X line, L1X sample, and height by the mean values for this band/SCA.
        ii. Evaluate L1R line and L1R sample using equations (1) and (2) above with the RPC coefficients for this band/SCA.
    c. If the computed L1R sample coordinate is between 0 and the number of L1R samples per SCA for this band.
        i. Increment the number of successful searches, ntry.
        ii. If the L1R line number is between 0 and the number of L1R lines in the image:
            1. Store the calculated L1R line coordinate.
            2. Convert the L1R sample SCA coordinate to a L1R file coordinate, and store that also:

            L1R_file_samp = L1R_SCA_samp + isca*num_samp_per_SCA

            3. Increment the number of SCAs found.
        iii. Set last_sca = isca
        iv. If we've found more than one SCA (ntry > 1) return the number found.
        v. If the L1R SCA sample number is below the SCA overlap threshold, decrement the current SCA index (isca) and go back to step b to test for a second overlapping SCA.
        vi. If the L1R SCA sample number is within the SCA overlap threshold of the number of samples per SCA, increment the current SCA index (isca) and go back to step b to test for a second overlapping SCA.
        vii. If the L1R SCA sample number is not within the potential overlap regions, return the number of SCAs found.
    d. If the L1R sample is out of range for the current SCA (i.e., the test in step c. above fails):
        i. If at least one SCA has already been found (ntry > 0), return the number found.
        ii. If the L1R sample number is outside the image (< 0 for the first SCA or > number of samples for the last SCA), return the number of SCAs found.
        iii. Calculate the L1R file sample number:

L1R_file_samp = L1R_SCA_samp + isca * num_samp_per_SCA

iv. Calculate the predicted SCA index:
isca = L1R_file_samp / num_samp_per_SCA

isca = MAX (isca, 0)

isca = MIN (isca, num_SCA – 1)

v. Go back to step b.

e. This sub-algorithm returns the number of valid SCAs found and the corresponding L1R line and sample coordinates for each.

3. For each SCA found to contain the point, calculate the satellite and solar vectors:
   a. This procedure is described in step 7.a. of the phase 1 algorithm above with the exception of the first sub-step (height retrieval). Here, the height is provided as an input.
4. Calculate the satellite and sun zenith angles corresponding to the vectors, clipping the zenith angles at the horizon (90 degrees):

   if satvector.z > 0 then sat_zenith = acos(satvector.z)

   else sat_zenith = 0

   if sunvector.z > 0 then sun_zenith = acos(sunvector.z)

   else sun_zenith = 0

5. Calculate the satellite and sun azimuth angles, setting the azimuth equal to zero if the vector is vertical:

   hdist = sqrt (satvector.x*satvector.x + satvector.y*satvector.y)

   if hdist > 0 then sat_azimuth = atan2(satvector.x, satvector.y)

   else sat_azimuth = 0

   hdist = sqrt (sunvector.x*sunvector.x + sunvector.y*sunvector.y)

   if hdist > 0 then sun_azimuth = atan2 (sunvector.x, sunvector.y)

   else sun_azimuth = 0

6. Average the angles computed from the individual SCAs.

3.2.3 Compute Angles Using Rational Polynomial Coefficient (RPC) Method

To compute the satellite viewing and solar illumination angles at a specified L1X line/sample location, given the corresponding elevation and angle coefficients, using the RPC method:

1. If the input height is NULL, replace it with the mean height from the band RPC parameters in the ANGCF structure.
2. Determine which SCA, or SCAs, viewed the L1X pixel. This procedure is the same as for the rigorous method and is described above. Note, however, that this sub-algorithm returns the L1R line and L1R file sample coordinates for each valid SCA. The L1R sample coordinate was not used in the rigorous method, but will be here.
3. Offset the L1X line, L1X sample, and height values by the mean values for the current band.
4. For each valid SCA:
    a. Offset the L1R line and L1R sample coordinates by the mean values for the current band.
    b. Use the offset values and the angle RPC model parameters for this band to evaluate equations (5a), (5b), and (5c) above for each component of the satellite viewing vector and each component of the solar illumination vector.
    c. Calculate the angles corresponding to the resulting vectors using the methods described in steps 4 and 5 of the rigorous method, above.
5. Average the angles computed from the individual SCAs.

3.3 Angle Coefficient Output File

The detailed contents of the angle coefficient (ANG) file are shown in Table 3-1 below. Note that although some of the fields in the ANG file duplicate information found in the standard L1X product metadata (MTL) file, this was done intentionally to make the ANG file self- contained. In some cases, different parameter names are used in the ANG file. In a production implementation, it may be desirable to harmonize the parameter names or even combine the files into one. Such decisions are beyond the scope of this algorithm.

The ANG file is ODL structure text and consists of 15 parameter groups: a file header group, a projection group, an ephemeris group, a solar vector group, and one group of RPC model parameters for each of the 11 spectral bands.

Group	Parameter	Type	Size	Contents
FILE_HEADER	FILE_NAME	char	29	The ANG file name mimics the MTA file name with the extension “MTA” replaced by “ANG”.
FILE_HEADER	SATELLITE	char	9	Satellite identifier = LANDSAT_9.
FILE_HEADER	WRS_PATH	int	1	Scene Worldwide Reference System 2 (WRS-2) orbit-based path (1-233).
FILE_HEADER	WRS_ROW	int	1	Scene WRS-2 orbit-based row (1-248).
FILE_HEADER	NUMBER_OF_BANDS	int	1	Number of bands contained in this file, normally 11.
FILE_HEADER	BAND_LIST	int	11	List of the spectral band numbers contained in this file, normally (1,2,3,4,5,6,7,8,9,10,11).
PROJECTION	ELLIPSOID_AXES	double	2	WGS84 ellipsoid semi-major and semi-minor axes in meters.
PROJECTION	PROJECTION_CODE	int	1	Code for map projection type: 1 = UTM, 6 = PS.
PROJECTION	PROJECTION_UNITS	char	6	Map projection units will always be METERS.
PROJECTION	PROJECTION_DATUM	char	5	Datum will always be WGS84.
PROJECTION	PROJECTION_SPHEROID	int	1	The projection spheroid code will always be 12.
PROJECTION	PROJECTION_ZONE	int	1	UTM zone number (1-60). Note that only northern hemisphere zones are used, so this number will always be positive.
PROJECTION	PROJECTION_PARAMETERS	double	15	GCTP map projection parameters. All zeros for UTM. For polar stereographic, this contains the ellipsoid axes, false easting and northing (both 0), latitude of true scale (+/-71 degrees), and the vertical axis longitude (0).
PROJECTION	UL_CORNER	double	2	L1X upper left corner map projection coordinates (meters).
PROJECTION	UR_CORNER	double	2	L1X upper right corner map projection coordinates (meters).
PROJECTION	LL_CORNER	double	2	L1X lower left corner map projection coordinates (meters).
PROJECTION	LR_CORNER	double	2	L1X lower right corner map projection coordinates (meters).
EPHEMERIS	EPHEMERIS_EPOCH_YEAR	int	1	Year of ephemeris epoch (start time).
EPHEMERIS	EPHEMERIS_EPOCH_DAY	int	1	Epoch day of year.
EPHEMERIS	EPHEMERIS_EPOCH_SECOND	double	1	Epoch seconds of day.
EPHEMERIS	NUMBER_OF_POINTS	int	1	Number of ephemeris points provided in following four parameter fields.
EPHEMERIS	EPHEMERIS_TIME	double	variable	Ephemeris sample time offsets (from epoch) in seconds.

Group	Parameter	Type	Size	Contents
EPHEMERIS	EPHEMERIS_ECEF_X	double	variable	Ephemeris sample Earth Centered Earth Fixed X coordinate in meters.
EPHEMERIS	EPHEMERIS_ECEF_Y	double	variable	Ephemeris sample Earth Centered Earth Fixed Y coordinate in meters.
EPHEMERIS	EPHEMERIS_ECEF_Z	double	variable	Ephemeris sample Earth Centered Earth Fixed Z coordinate in meters.
SOLAR_VECTOR	SOLAR_EPOCH_YEAR	int	1	Year of solar vector epoch (start time). This is the same as the ephemeris epoch.
SOLAR_VECTOR	SOLAR_EPOCH_DAY	int	1	Epoch day of year.
SOLAR_VECTOR	SOLAR_EPOCH_SECOND	double	1	Epoch seconds of day.
SOLAR_VECTOR	NUMBER_OF_POINTS	int	1	Number of solar vectors provided in following four parameter fields.
SOLAR_VECTOR	SAMPLE_TIME	double	variable	Vector sample time offsets (from epoch) in seconds.
SOLAR_VECTOR	SOLAR_ECEF_X	double	variable	Solar vector sample Earth Centered Earth Fixed X direction.
SOLAR_VECTOR	SOLAR_ECEF_Y	double	variable	Solar vector sample Earth Centered Earth Fixed Y direction.
SOLAR_VECTOR	SOLAR_ECEF_Z	double	variable	Solar vector sample Earth Centered Earth Fixed Z direction.
RPC_BAND01	NUMBER_OF_SCAS	int	1	Number of SCAs: 14 for OLI bands.
RPC_BAND01	NUM_L1X_LINES	int	1	Number of lines in the L1X product.
RPC_BAND01	NUM_L1X_SAMPS	int	1	Number of samples in the L1X product.
RPC_BAND01	NUM_L1R_LINES	int	1	Number of lines in the L1R product.
RPC_BAND01	NUM_L1R_SAMPS	int	1	Number of samples per SCA in the L1R product.
RPC_BAND01	PIXEL_SIZE	double	1	L1X pixel size, in meters.
RPC_BAND01	START_TIME	double	1	L1R image start time, in seconds, from the ephemeris epoch.
RPC_BAND01	LINE_TIME	double	1	L1R image line time increment in seconds.
RPC_BAND01	BAND01_MEAN_HEIGHT	double	1	Mean height offset for the RPC angle model.
RPC_BAND01	BAND01_MEAN_L1R_LINE_SAMP	double	2	Mean L1R line and (file) sample offsets for the RPC angle model.
RPC_BAND01	BAND01_MEAN_L1X_LINE_SAMP	double	2	Mean L1X line and sample offsets for the RPC angle model.
RPC_BAND01	BAND01_MEAN_SAT_VECTOR	double	3	Mean satellite view vector for the RPC angle model.
RPC_BAND01	BAND01_SAT_X_NUM_COEF	double	10	Numerator polynomial coefficients for the satellite view vector X coordinate.
RPC_BAND01	BAND01_SAT_X_DEN_COEF	double	9	Denominator polynomial coefficients for the satellite view vector X coordinate.

Group	Parameter	Type	Size	Contents
RPC_BAND01	BAND01_SAT_Y_NUM_COEF	double	10	Numerator polynomial coefficients for the satellite view vector Y coordinate.
RPC_BAND01	BAND01_SAT_Y_DEN_COEF	double	9	Denominator polynomial coefficients for the satellite view vector Y coordinate.
RPC_BAND01	BAND01_SAT_Z_NUM_COEF	double	10	Numerator polynomial coefficients for the satellite view vector Z coordinate.
RPC_BAND01	BAND01_SAT_Z_DEN_COEF	double	9	Denominator polynomial coefficients for the satellite view vector Z coordinate.
RPC_BAND01	BAND01_MEAN_SUN_VECTOR	double	3	Mean sun vector for the RPC angle model.
RPC_BAND01	BAND01_SUN_X_NUM_COEF	double	10	Numerator polynomial coefficients for the sun vector X coordinate.
RPC_BAND01	BAND01_SUN_X_DEN_COEF	double	9	Denominator polynomial coefficients for the sun vector X coordinate.
RPC_BAND01	BAND01_SUN_Y_NUM_COEF	double	10	Numerator polynomial coefficients for the sun vector Y coordinate.
RPC_BAND01	BAND01_SUN_Y_DEN_COEF	double	9	Denominator polynomial coefficients for the sun vector Y coordinate.
RPC_BAND01	BAND01_SUN_Z_NUM_COEF	double	10	Numerator polynomial coefficients for the sun vector Z coordinate.
RPC_BAND01	BAND01_SUN_Z_DEN_COEF	double	9	Denominator polynomial coefficients for the sun vector Z coordinate.
RPC_BAND01	BAND01_SCA_LIST	int	14	List of SCAs in this band. For OLI bands, this is: (1,2,3,4,5,6,7,8,9,10,11,12,13,14).
RPC_BAND01	BAND01_SCA01_MEAN_HEIGHT	double	1	Mean height offset for the SCA01 L1X-to-L1R RPC model.
RPC_BAND01	BAND01_SCA01_MEAN_L1R_LINE_SAMP	double	2	Mean L1R line and (SCA) sample offsets for the SCA01 L1X-to-L1R RPC model.
RPC_BAND01	BAND01_SCA01_MEAN_L1X_LINE_SAMP	double	2	Mean L1X line and sample offsets for the SCA01 L1X-to-L1R RPC model.
RPC_BAND01	BAND01_SCA01_LINE_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA01 L1R line RPC model.
RPC_BAND01	BAND01_SCA01_LINE_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA01 L1R line RPC model.
RPC_BAND01	BAND01_SCA01_SAMP_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA01 L1R sample RPC model.
RPC_BAND01	BAND01_SCA01_SAMP_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA01 L1R sample RPC model.

Group	Parameter	Type	Size	Contents
RPC_BAND01	BAND01_SCA02_MEAN_HEIGHT	double	1	Mean height offset for the SCA02 L1X-to-L1R RPC model.
RPC_BAND01	BAND01_SCA02_MEAN_L1R_LINE_SAMP	double	2	Mean L1R line and (SCA) sample offsets for the SCA02 L1X-to-L1R RPC model.
RPC_BAND01	BAND01_SCA02_MEAN_L1X_LINE_SAMP	double	2	Mean L1X line and sample offsets for the SCA02 L1X-to-L1R RPC model.
RPC_BAND01	BAND01_SCA02_LINE_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA02 L1R line RPC model.
RPC_BAND01	BAND01_SCA02_LINE_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA02 L1R line RPC model.
RPC_BAND01	BAND01_SCA02_SAMP_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA02 L1R sample RPC model.
RPC_BAND01	BAND01_SCA02_SAMP_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA02 L1R sample RPC model.
RPC_BAND01	BAND01_SCAnn_...			The previous seven parameters repeat for SCAs 03 through 14.
…				
RPC_BANDmm				The RPC_BAND01 group parameters are repeated for bands 2 through 9.
…				
RPC_BAND10	NUMBER_OF_SCAS	int	1	Number of SCAs: 3 for TIRS bands.
RPC_BAND10	NUM_L1X_LINES	int	1	Number of lines in the L1X product.
RPC_BAND10	NUM_L1X_SAMPS	int	1	Number of samples in the L1X product.
RPC_BAND10	NUM_L1R_LINES	int	1	Number of lines in the L1R product.
RPC_BAND10	NUM_L1R_SAMPS	int	1	Number of samples per SCA in the L1R product.
RPC_BAND10	PIXEL_SIZE	double	1	L1X pixel size, in meters.
RPC_BAND10	START_TIME	double	1	L1R image start time in seconds from the ephemeris epoch.
RPC_BAND10	LINE_TIME	double	1	L1R image line time increment in seconds.
RPC_BAND10	BAND10_MEAN_HEIGHT	double	1	Mean height offset for the RPC angle model.
RPC_BAND10	BAND10_MEAN_L1R_LINE_SAMP	double	2	Mean L1R line and (file) sample offsets for the RPC angle model.
RPC_BAND10	BAND10_MEAN_L1X_LINE_SAMP	double	2	Mean L1X line and sample offsets for the RPC angle model.
RPC_BAND10	BAND10_MEAN_SAT_VECTOR	double	3	Mean satellite view vector for the RPC angle model.
RPC_BAND10	BAND10_SAT_X_NUM_COEF	double	10	Numerator polynomial coefficients for the satellite view vector X coordinate.

Group	Parameter	Type	Size	Contents
RPC_BAND10	BAND10_SAT_X_DEN_COEF	double	9	Denominator polynomial coefficients for the satellite view vector X coordinate.
RPC_BAND10	BAND10_SAT_Y_NUM_COEF	double	10	Numerator polynomial coefficients for the satellite view vector Y coordinate.
RPC_BAND10	BAND10_SAT_Y_DEN_COEF	double	9	Denominator polynomial coefficients for the satellite view vector Y coordinate.
RPC_BAND10	BAND10_SAT_Z_NUM_COEF	double	10	Numerator polynomial coefficients for the satellite view vector Z coordinate.
RPC_BAND10	BAND10_SAT_Z_DEN_COEF	double	9	Denominator polynomial coefficients for the satellite view vector Z coordinate.
RPC_BAND10	BAND10_MEAN_SUN_VECTOR	double	3	Mean sun vector for the RPC angle model.
RPC_BAND10	BAND10_SUN_X_NUM_COEF	double	10	Numerator polynomial coefficients for the sun vector X coordinate.
RPC_BAND10	BAND10_SUN_X_DEN_COEF	double	9	Denominator polynomial coefficients for the sun vector X coordinate.
RPC_BAND10	BAND10_SUN_Y_NUM_COEF	double	10	Numerator polynomial coefficients for the sun vector Y coordinate.
RPC_BAND10	BAND10_SUN_Y_DEN_COEF	double	9	Denominator polynomial coefficients for the sun vector Y coordinate.
RPC_BAND10	BAND10_SUN_Z_NUM_COEF	double	10	Numerator polynomial coefficients for the sun vector Z coordinate.
RPC_BAND10	BAND10_SUN_Z_DEN_COEF	double	9	Denominator polynomial coefficients for the sun vector Z coordinate.
RPC_BAND10	BAND10_SCA_LIST	int	3	List of SCAs in this band. For TIRS bands this is: (1,2,3).
RPC_BAND10	BAND10_SCA01_MEAN_HEIGHT	double	1	Mean height offset for the SCA01 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA01_MEAN_L1R_LINE_SAMP	double	2	Mean L1R line and (SCA) sample offsets for the SCA01 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA01_MEAN_L1X_LINE_SAMP	double	2	Mean L1X line and sample offsets for the SCA01 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA01_LINE_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA01 L1R line RPC model.
RPC_BAND10	BAND10_SCA01_LINE_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA01 L1R line RPC model.
RPC_BAND10	BAND10_SCA01_SAMP_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA01 L1R sample RPC model.
RPC_BAND10	BAND10_SCA01_SAMP_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA01 L1R sample RPC model.

Group	Parameter	Type	Size	Contents
RPC_BAND10	BAND10_SCA02_MEAN_HEIGHT	double	1	Mean height offset for the SCA02 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA02_MEAN_L1R_LINE_SAMP	double	2	Mean L1R line and (SCA) sample offsets for the SCA02 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA02_MEAN_L1X_LINE_SAMP	double	2	Mean L1X line and sample offsets for the SCA02 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA02_LINE_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA02 L1R line RPC model.
RPC_BAND10	BAND10_SCA02_LINE_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA02 L1R line RPC model.
RPC_BAND10	BAND10_SCA02_SAMP_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA02 L1R sample RPC model.
RPC_BAND10	BAND10_SCA02_SAMP_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA02 L1R sample RPC model.
RPC_BAND10	BAND10_SCA03_MEAN_HEIGHT	double	1	Mean height offset for the SCA03 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA03_MEAN_L1R_LINE_SAMP	double	2	Mean L1R line and (SCA) sample offsets for the SCA03 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA03_MEAN_L1X_LINE_SAMP	double	2	Mean L1X line and sample offsets for the SCA03 L1X-to-L1R RPC model.
RPC_BAND10	BAND10_SCA03_LINE_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA03 L1R line RPC model.
RPC_BAND10	BAND10_SCA03_LINE_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA03 L1R line RPC model.
RPC_BAND10	BAND10_SCA03_SAMP_NUM_COEF	double	5	Numerator polynomial coefficients for the SCA03 L1R sample RPC model.
RPC_BAND10	BAND10_SCA03_SAMP_DEN_COEF	double	4	Denominator polynomial coefficients for the SCA03 L1R sample RPC model.
RPC_BAND11				The RPC_BAND10 group parameters are repeated for band 11.

Table 3-1. Angle Coefficient File Detailed Content

Appendix A Acronyms

ADD	Algorithm Description Document
ANGCF	Angle Coefficient
BSQ	Band Sequential
Cal/Val	Calibration/Validation
CCB	Configuration Control Board
CPF	Calibration Parameter File
CR	Change Request
DEM	Digital Elevation Model
DPAS	Data Processing and Archive System
ECEF	Earth-Centered, Earth-Fixed
ECI	Earth Centered Inertial
ECITOD	Earth Centered Inertial True of Date
ENVI	Environment for Visualizing Images
EROS	Earth Resource Observation and Science
FPM	Focal Plane Module
GCTP	General Cartographic Transformation Package
HDF	Hierarchical Data Format
IAS	Image Assessment System
L1	Level 1
L1Gt	Level 1 Systemic Terrain (Corrected)
L1R	Level 1 Radiometric (Corrected)
L1TP	Level 1 Terrain Precision (Corrected)
LOS	Line of Sight
LPGS	Landsat Product Generation System
NOVAS	Naval Definition Vector Astronomy Software
ODL	Object Description Language
OLI	Operational Land Imager
PS	Polar Stereographic
ROIC	Read-Out Integrated Circuit
RPC	Rational Polynomial Coefficients
SCA	Sensor Chip Assembly
TIRS	Thermal Infrared Sensor
UL	Upper Left
USGS	U.S. Geological Survey
UT1	UTC Corrected
UTC	Universal Time Code
UTM	Universal Transverse Mercator
WGS84	World Geodetic System 1984
WRS	Worldwide Reference System
WRS-2	Worldwide Reference System 2

References

Please see https://landsat.usgs.gov/glossary-and-acronyms for a list of acronyms.

Visit https://landsat.usgs.gov/solar-illumination-and-sensor-viewing-angle-coefficient-file for more information.