Algorithm Theoretical Basis Document (ATBD)

NASA TRMM TMI Level 1B Algorithm

Version 1

Prepared for:

Tropical Rainfall Measuring Mission (TRMM)
National Aeronautics and Space Administration (NASA)

Prepared by:

Yimin Ji
Steve Bilanow
Erich Stocker

TRMM Science and Data Information System (TSDIS)
Goddard Space Flight Center Code 610.2, NASA
Greenbelt MD 20771

March, 2008

TABLE OF CONTENTS

Page

1. INTRODUCTION

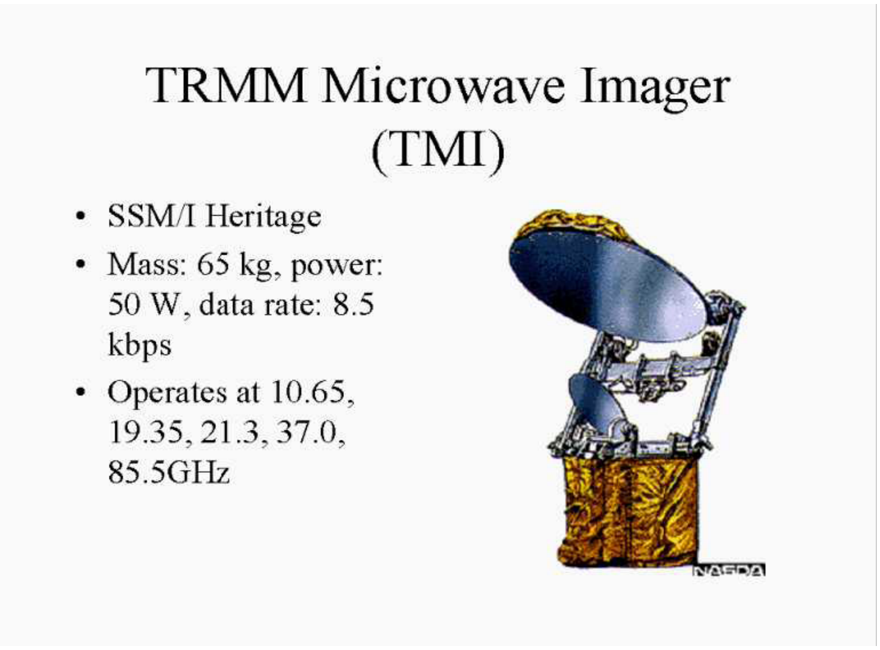
1.1 OBJECTIVE AND RELEVANT DOCUMENTS

This document describes the Tropical Rainfall Measuring Mission (TRMM) Microwave Imager (TMI) level 1B algorithm. It consists of physical bases and mathematical equations for TMI geolocation and calibration as well as validation and evolution of the algorithm since the launch of the satellite. This document also describes software design for the level 1B algorithm. However, the detailed software architecture and code design of TMI L1B algorithm are presented in a separate document: the TSDIS *Level 1 Software Design Specification (L1 SDS)*

1.2 INSTRUMENT DESCRIPTION

The TRMM satellite was launched on November 27, 1997. After launch, the satellite (Simpson *et al*. 1996) flew in a 350-km circular orbit with a 35^{o} inclination angle and carried five sensors: TRMM Microwave Imager (TMI), the precipitation radar (PR), the Visible and Infrared Radiometer System (VIRS), the Clouds and Earth's Radiant Energy System (CERES), and the Lightning Imaging System (LIS). The TRMM orbit was boosted to 402-km in August 2001 to extend the life of the TRMM mission. Since the launch of the TRMM, all the three major sensors (PR, TMI, and VIRS) have been worked normally while the TRMM algorithms and products have been advanced and reprocessed several times.

TMI (Fig. 1.1) is a heritage passive microwave sensor of the highly successful Special Sensor Microwave Imager (SSM/I, Hollinger 1998; Colton and Poe 1998) which has been flying continuously on Defense Meteorological Satellites since 1987.



TRMM Microwave Imager
(TMI)

- SSM/I Heritage
- Mass: 65 kg, power: 50 W, data rate: 8.5 kbps
- Operates at 10.65, 19.35, 21.3, 37.0, 85.5GHz

Figure 1.1. TRMM Microwave Imager

The TMI antenna is an offset paraboloid, with an aperture size of 61 cm (projected along the propagation direction) and a focal length of 50.8 cm. The antenna beam views the earth surface with the ‘‘nadir’’ angle of 49° between the beam viewing direction and the local nadir which is perpendicular to the Earth ellipsoid surface. This results in an incidence angle of about 52.8° at the earth’s surface prior to the boost and about 53.4° after the boost. The TMI antenna rotates about a nadir axis at a constant speed of 31.6 rpm. The rotation draws a ‘‘circle’’ on the earth’s surface. Only 130° of the +X spacecraft body direction of the complete circle is used for taking earth view data. The rest is used for calibrations and housekeeping purposes. From the TRMM orbit, the 130° scanned sector yields a swath width of TMI about 760 km prior to the boost, and about 880 km after the boost. During each complete revolution (i.e., a scan period of about 1.9 s), the sub-satellite point advances a distance of 13.9 km prior to the boost, and 13.7 km after the boost due to the slightly longer orbit period at the higher altitude. (The average period from one orbit ascending node to the next is 91.32 minutes pre-boost and 92.39 minutes post-boost.) Since the smallest footprint (85.5-GHz channels) size pre-boost is only 6.9 km (down-track direction) by 4.6 km (cross-track direction), there was a ‘‘gap’’ of 7.0 km between successive scans. After the boost the footprint sizes grew by about 15 percent due to the higher altitude, and the gap between successive scans shrank to about 5.5 km. For all other lower-frequency channels, footprints from successive scans overlap the previous scans.

The performance characteristics of the nine TMI channels are summarized by Kumara *et al.* (1998) in Table1.1, which gives Instantaneous Field-of-View (IFOV) and Effective Field-of-View (EFOV) sizes Along-Track (AT) and Cross-Track (CT) for the pre-boost altitude. The post-boost sizes in km on the surface are about 15 percent larger.

TABLE 1.1. TMI characteristics (From Kummerow et al. 1998)

Channel number	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
Center freq (GHz)	10.65	10.65	19.35	19.35	21.3	37.0	37.0	85.5	85.5
Polarization	V	H	V	H	V	V	H	V	H
Bandwidth (MHz)	100	100	500	500	200	2000	2000	3000	3000
Stability (MHz)	10	10	20	20	20	50	50	100	100
Beamwidth (deg)	3.68	3.75	1.90	1.88	1.70	1.0	1.0	0.42	0.23
IFOV-DT (km)	59.0	60.1	30.5	30.1	27.2	16.0	16.0	6.7	6.9
IFOV-CT (km)	35.7	36.4	18.4	18.2	16.5	9.7	9.7	4.1	4.2
Integration time per sample (ms)	6.6	6.6	6.6	6.6	6.6	6.6	6.6	3.3	3.3
EFOV-CT (km)	9.1	9.1	9.1	9.1	9.1	9.1	9.1	4.6	4.6
EFOV-DT (km)	63.2	63.2	30.4	30.4	22.6	16.0	16.0	7.2	7.2
EFOVs per scan	104	104	104	104	104	104	104	208	208
Samples (*N*) per beamwidth	4	4	2	2	2	1	1	1	1
Beam EFOV (km^2)	63 x 37	63 x 37	30 x 18	30 x 18	23 x 18	16 x 9	16 x 9	7 x 5	7 x 5
Beam EFOVs per scan	26	26	52	52	52	104	104	208	208
Temperature sensitivity, NEDT (K)	0.63	0.54	0.50	0.47	0.71	0.36	0.31	0.52	0.93
Beam temperature sensitivity NEDT (K)	0.32	0.27	0.35	0.33	0.50	0.36	0.31	0.52	0.93
Beam efficiency/Xpol(%)	93/0.4	93/0.5	96/0.4	96/0.5	98/0.6	91/2.2	92/2.1	82/2.0	85/3.0

Between each of the 130° active earth-viewing parts of the scan, the TMI goes through a two-point calibration process. During the hot calibration, the feed horns are moved to a position viewing a ‘‘hot load’’ whose brightness temperature is known accurately. Next, the feed horns are moved to a second position where their beams are reflected by a ‘‘cold space sub-reflector’’ to view the cosmic background radiation. The two-point calibration is repeated every scan to establish the instantaneous calibration of the TMI and remove any gain and baseline fluctuation effects. The frequent onboard calibration is a key to preserving the stability of the TMI.

1.3 ALGORITHM DESCRIPTION

The level 1B algorithm and software transform Level 1A granules into Level 1B granules. The main functions include radiometric calibration and geolocation. The Level 1A products hold the original housekeeping and science data telemetry packets. The Level 1B products are calibrated and geolocated brightness temperatures (Tb). Additional processing, such as QA (phase B, C, and D - including instrument checks) and quick look processing, is performed at this level as well. The regular Level 1B process is composed of the following components:

- Satellite Maneuvers
- Combined Geolocation Algorithm
- L1B Common Processing
- L1B TMI Calibration and Processing
- Post-L1B Processing
- Quality Assurance

A simplified version of the algorithm is used for realtime data processing.

1.4 L1B DATA DESCRIPTION

The standard level 1B TMI data are the ‘‘calibrated’’ microwave brightness temperatures for the EFOV and corrected for the cross-polarization effect. Each scan line contains the time of the scan line, along with data quality indicators and a TRMM direction indicator. The earth science data in each scan line consist of latitude and longitude values along with brightness temperatures for the 208 EFOVs at 85 GHz and 104 EFOVs at all the remaining lower-frequency channels. Data are granularized into one-orbit elements with orbits beginning at the southernmost point of the satellite orbit. Each granule of TMI is padded with 50 scan lines before and after the actual orbit. This is done to facilitate the merging of TMI data with the PR and VIRS instrument data. The product is in HDF format.

Section 2 and 3 describe the physics and math of geolocation/calibration algorithms. The L1B data processing will be discussed in section 4.

2. COMBINED GEOLOCATION ALGORITHM

This section discusses algorithms related to the TRMM TMI geolocation and provides details of the mathematical computations. This includes the handling of spacecraft ephemeris and attitude data, and various miscellaneous computations to provide metadata related to the navigation. The geolocation sections describe the computations for pixel latitude and longitude assuming that the onboard attitude (i.e. the orientation or pointing) of the TRMM spacecraft is correct as provided by the onboard Attitude Control System (ACS). The pointing accuracy of the ACS is described in papers published since launch (Robertson et. al. 1999; Andrews and Bilanow, 2002; and Bilanow and Slojkowski 2006). Generally the pointing knowledge in normal mission mode has been much better than the 0.2 degree requirement, and usually better than 0.1 degrees, except for occasional minor anomalies such that the error can be as large as 0.5 degree for very brief periods. The overall instrument pointing accuracy is also affected by the instrument alignment accuracy, and this has only been verified only to about a half-pixel accuracy (as defined by separations between highest resolution pixel observations rather than by field-of-view 1/2 power widths) by using spot checks of images with a coastline database. This 1/2 pixel accuracy represents about 0.4 degrees for TMI for the worst case direction along track, and about 0.2 degrees in the along scan direction. The effects of systematic alignment errors would cause geolocation errors and incidence angle errors which would vary systematically across each scan. The discussion is provided in the following areas:

- Direct Referencing Method
- Inverse Referencing Method
- Ancillary Data Calculations

Direct referencing refers to the process of calculating the latitude and longitude of a pixel, the point on the Earth's surface at which the sensor is looking when a particular radiance measurement is taken. This direct referencing calculation is executed for an entire granule, pixel by pixel. This process includes handling ephemeris and spacecraft attitude data. The algorithm also computes and stores key information per scan line which is referred to as navigation data. This includes the spacecraft position and the sensor orientation. The navigation data can be used to re-compute the geolocation including bias effects if needed, and can be used for the inverse referencing algorithm.

Inverse referencing is the computation of scan and pixel numbers in an image where a given latitude/longitude location is found. This computation can be used in various image mapping applications. It will be applied in QA processing to overlay coastline data on a full resolution image.

In addition, various ancillary data items will be calculated and stored in the granule metadata. The list of ancillary data items includes Keplerian orbital elements, mean motion, attitude and orbit adjust flags, solar beta angles, and minimum and maximum of pitch, roll, and yaw.

2.1 DIRECT REFERENCING METHOD

The goal of the Direct Referencing Method is to compute the Earth location for each image pixel. This method is based on the formula for the intersection of a line with an ellipse. Oblateness of the Earth and the effects of Earth's rotation are also taken into consideration, as well as motion of the satellite during the scan. This method is suitable for geolocation computations for the three TRMM rain instruments, PR, VIRS, and TMI.

Since the TRMM direct referencing method uses a basic Earth ellipsoid shape model for geolocation, it does NOT include effects of topography or cloud heights in computing the exact geodetic latitude and longitude of observed location. These effects must be considered by the data users for cases where they might be important in viewing cloud tops or mountain tops at high slant paths—but most of the time very little error is introduced by this simplification. It should also be noted that geolocation accuracy is degraded around thruster burns for orbit adjusts and during yaw turns and control anomalies.

The relationship between the spacecraft position vector from Earth's center to spacecraft (P), the look vector from sensor to pixel (D), and the target vector from Earth center to pixel (G) on an oblate Earth is shown in Figure 2.1.

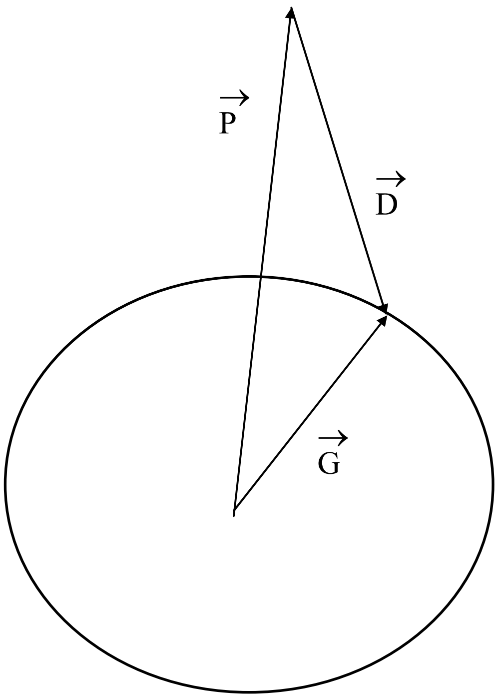


Figure 2.1 Relationships Between Vectors Involved in the Geolocation Calculations

The calculation of geodetic latitude and longitude involves the following:

- Derive Spacecraft Position and Velocity
- Calculate Sensor Alignment Matrix
- Compute the Nadir-to-GCI Rotation Matrix
- Compute the GCI-to-Orbital Rotation Matrix
- Calculate Attitude Matrix
- Calculate Sensor Orientation Matrix
- Calculate Instantaneous Field of View (IFOV) Vector
- Intersect With Oblate Earth
- Calculate Greenwich Hour Angle
- Derive Geodetic Latitude and Longitude

Derive Spacecraft Position and Velocity

The position and velocity vectors of a spacecraft at a specified time are obtained by reading position and velocity data from the platform ephemeris file and interpolating between the obtained data points. An ephemeris is a table that gives the position and velocity components of a heavenly body at regular time intervals. In the case of the TRMM satellite ephemeris, these are one minute intervals. TSDIS receives (via SDPF) a definitive orbit ephemeris file from the GSFC Flight Dynamics Facility (FDF). The FDF ephemeris product, called EPHEM, is delivered daily. It is based on tracking data and covers about 34 hours including the given day and about 10 hours of the next day. A Hermite polynomial interpolation, which takes into account both position and velocity components, is applied to obtain the position and velocity at a specified time. The EPHEM file is read and interpolated using legacy software provided by the FDF to obtain the spacecraft position at the mid-time of each scan. The spacecraft velocity is used to extrapolate the exact position during each pixel observation during the scan.

Calculate Sensor Alignment Matrix

Instrument alignment was measured prior to launch and can be incorporated in the geolocation algorithm as constant parameters. The alignment may change by the launch shock or due to thermal distortions in orbit which would require a change to the alignment parameters to avoid systematic errors in the geolocation. Sensor alignment angles are converted into a sensor alignment matrix, [M], using a standard Euler rotation sequence. An existing subroutine will be used which will allow any choice of alignment sequence and alignment angles. Specific sensor alignment angles for the VIRS instruments were derived prior to launch and were incorporated in our sensor model parameters file—angles -.06, 0.08, -.14 in degrees for a 3-2-1 Euler sequence. For TMI, end-to-end alignments could not be derived from available instrument manufacturer data, so nominal alignments, 0.0, 0.0, 0.0 error angles were used. For PR, a special input table giving the line of sight vectors of each of the 49 beam angles was supplied by the Japanese radar team for incorporation with the geolocation toolkit routines. The nominal alignments worked well enough to meet mission geolocation accuracy requirements within a pixel or less.

Compute the Nadir-to-GCI Rotation Matrix

The Nadir-to-GCI rotation matrix rotates a vector from geodetic nadir frame coordinates to GCI coordinates. This is computed based on the position and velocity vectors at the required time, according to the following steps.

Let **P** and **V** be the position and velocity vectors.

Step 1. Calculate the Local Geodetic Nadir Vector

An algorithm implemented for SeaWiFS will be used to calculate the Z-axis as the local geodetic nadir vector.

Let $\varepsilon' = ((1-\varepsilon)^2 R + p - R) / p$ be the eccentricity of the earth

where:

p = magnitude of the position vector
R = Mean radius of the Earth

$$N_Z = \begin{bmatrix} -\varepsilon * P_1 / (P_3^2 + \varepsilon^2 (P_1^2 + P_2^2)) \\ -\varepsilon * P_2 / (P_3^2 + \varepsilon^2 (P_1^2 + P_2^2)) \\ -\varepsilon * P_3 / (P_3^2 + \varepsilon^2 (P_1^2 + P_2^2)) \end{bmatrix}$$

Step 2. Calculate the Y - axis

Compute the Y- axis along negative orbit normal:

$$\mathbf{T} = N_Z \times \mathbf{V}$$

$$N_y = \begin{bmatrix} T_1 / t \\ T_2 / t \\ T_3 / t \end{bmatrix}$$

where:

T = negative orbit normal
t = magnitude of **T**

Step 3. Calculate the X - axis

Compute the X - axis to complete the orthonormal triad:

$$N_X = N_y \times N_Z$$

Step 4. Assemble the Nadir-to-GCI matrix
Finally the Nadir-to-GCI matrix is assembled from the three component vectors in the rows,

$$[N] = \begin{bmatrix} N_X \\ N_Y \\ N_Z \end{bmatrix}$$

Compute the GCI-to-Orbital Rotation Matrix

The GCI-to-Orbital rotation matrix transforms vectors from Geocentric Inertial Coordinates to orbital coordinates. The orbital coordinates are defined as follows :

Z is along the geocentric nadir (from the spacecraft)
Y is opposite the orbit normal direction, and,
X which completes the orthogonal triad is approximately in the velocity direction for a near circular orbit.

The orbital coordinates now provide the reference frame relative to which pitch, roll and yaw rotation angles will be provided in the TSDIS navigation data.

The GCI-to-Orbital rotation matrix is calculated as explained below, given the position vector **P** and the velocity vector **V**.

Step 1: Calculate the Nadir unit vector (i.e. Z axis)

$$[O_z] = \begin{bmatrix} -1 * P_1 / p_{mag} \\ -1 * P_2 / p_{mag} \\ -1 * P_3 / p_{mag} \end{bmatrix}$$

Where p_{mag} is the magnitude of the position vector **P**.

Step 2: Compute the unit Y-axis along the negative orbit normal

The Y-axis is the vector cross product of the geocentric nadir and the velocity vector.

$$[T] = [O_z] \text{ x } \mathbf{V}$$

$$[O_y] = \begin{bmatrix} T_1 / t_{mag} \\ T_2 / t_{mag} \\ T_3 / t_{mag} \end{bmatrix}$$

Where t_{mag} is the magnitude of [T].

Step 3: Calculate the unit X-axis

X axis is computed as the cross product of the Y and X axes.

$$[O_x] = [O_y] \text{ x } [O_z]$$

Step 4: Assemble the GCI-to-Orbital Rotation Matrix

Finally the GCI-to-Orbital matrix is assembled using the 3 axes calculated above.

$$[O] = \begin{bmatrix} O_x \\ O_y \\ O_z \end{bmatrix}$$

Calculate Attitude Matrix

The satellite attitude is the orientation of the spacecraft relative to a well-defined reference frame. In this case the reference frame chosen is the Geocentric Inertial (GCI) coordinate system. Therefore the Attitude matrix rotates a vector from Spacecraft coordinates to GCI coordinates.

Often, the attitude is also expressed in other coordinate systems. Typically for Earth-pointing spacecraft, the attitude is described by pitch, roll, and yaw angles. Generally, pitch is a rotation about the orbit axis, roll is a rotation about the velocity direction, and yaw is a rotation about the nadir direction. The exact definition requires a specific coordinate frame and a sequence of rotations. The specific coordinate frame to be used for pitch, roll, and yaw definitions for the TSDIS products is described in Section 1.1.4. The Euler rotation sequence used is a 3-2-1 rotation sequence.

For the TRMM satellite, onboard attitude data are available from the ACS ancillary data packets (APID 45). The ACS packet size is 70 bytes and it is recorded every 0.5 seconds. TSDIS use of this attitude information is described as follows.

The calculation of Attitude matrix involves the following steps.

- extract onboard attitude and apply corrections
- compute the Attitude Error rotation matrix
- compute the S/C-to-ACS rotation matrix
- compute the Nadir to ACS rotation matrix
- compute S/C-to-Nadir rotation matrix
- compute GCI-to-S/C rotation matrix

Extract Onboard Attitude and Apply Corrections

The TRMM onboard attitude is taken from the ACS ancillary data packet, and, starting with Version 6 processing, corrections are applied to the onboard attitude for selected data spans. The corrections were deemed necessary due to onboard attitude errors encountered after the change in the TRMM average operating altitude from 350 to 402.5 kilometers in August 2001. These corrections are taken from two new data files that are described further below: geo_ptch_corr.dat and geo_roll_corr.dat.

The ACS ancillary packet is unpacked to get the onboard attitude and various attitude control system status flags. The onboard attitude angles for roll, r_o, pitch, p_o, and yaw, y_o (referred to as "position error" in the telemetry handbook) give rotations about the TRMM X, Y and Z spacecraft axes respectively. These rotations are relative to a "tip frame" defined onboard which is very close to a geodetic frame (see section A.1.1.5.4). Bytes 22 - 25 represent the X position or roll error, r_o, bytes 26 - 29 the Y position or pitch error, p_o, and, bytes 30 - 33 the Z position or yaw error, y_o. A key flag, the spacecraft orientation flag (defining +X, -X, or +Y forward—see section A.1.1.5.3) is obtained from the 47th byte. Also unpacked are the spacecraft control mode, the target orientation used for yaw maneuvers, the yaw update status flag, and the contingency mode flag. TRMM switched to the contingency or "Kalman Filter" mode during the orbit boost to 402.5 kilometers altitude in August of 2001.

The pitch corrections file, geo_ptch_corr.dat, was developed because of errors encountered in the onboard ephemeris that affect the onboard attitude in the Kalman filter control mode after August 2001. Since the onboard ephemeris affects the reference frame for earth pointing in this control mode, occasional along track errors in the onboard orbit would affect pitch and cause errors up to about 0.7 degrees. Only cases where errors in pitch were larger than 0.2 degrees due to this problem were selected for correction. This resulted in corrections for about 30 data spans, mostly less than one day in duration, with the earliest on August 25th 2001, and the last on September 8th 2002. Since additional safeguards were added for the onboard ephemeris loads, it is not expected that further errors of significant magnitude will occur. Each correction is based on a 2nd order polynomial fit to the accumulating orbit error for the period until a new ephemeris vector is loaded onboard. Since the errors grow basically as a quadratic with time, a 2nd order fit is adequate. The formula for the pitch correction, δ_p, in degrees is

$$\delta_p = c_0 + c_1(t-t_0)/86400. + *c_2((t-t0)/86400.)^2$$

where t is the TAI time for the data (in seconds since O hours UT on Jan. 1, 1993), and t_0, c_0, c_1, and c_2 are read from the file, geo_ptch_corr.dat and are defined as follows. t_0 is the start TAI time in seconds for the period of pitch correction (this period ends at the start time given in the next line in the file), c_0 is the start value for the correction (and y-intercept of the polynomial fit), c_1 is the linear correction coefficient, in degrees per day, and c_2 is the quadratic correction coefficient, in degrees per day squared. Before, after, and between the periods of applied corrections, spans with zero values for c_0, c_1, and c_2 are inserted in the geo_ptch_corr.dat file to apply a null correction.

One longer term pitch correction, from 9/21/2001 to 9/27/2001, was included for expected coupling of pitch errors to a period of fairly large roll and yaw errors. This particular correction period allowed for a period of linear error growth and decline.

The roll corrections file, geo_roll_corr.dat, was developed due to orbit period systematic errors detected for the first three months following the switch to the Kalman filter control. These errors were reduced after an adjustment for the onboard Three Axis Magnetometer (TAM) alignment matrices was uploaded on Nov 28 2001. The errors followed an orbit period sinusoidal pattern in roll and yaw with varied amplitude occasionally nearly 0.5 degrees. The errors were estimated by fitting an independent measure of the spacecraft roll using the PR data. The correction applied is an orbit period sinusoid with the proper phases in roll and yaw to correct for the expected onboard error. Corrections are applied for each orbit from August 22nd 2001 through December 2001. The formula for the roll and yaw corrections, δ_r and δ_y respectively, in degrees are

$$\delta_r = c_a \cos(\alpha + c_p) + c_b$$

$$\delta_y = c_s\, c_a \sin(\alpha + c_p)$$

where α is the mean anomaly orbit angle from the granule starting point (southernmost latitude) for each orbit, and the correction terms c_a, c_p, and c_b are read from the file geo_roll_corr.dat and are defined as follows: c_a is the amplitude of the orbit period correction in degrees, c_p, gives the phase angle of the orbit period correction in degrees, and c_b is a bias term allowed for the roll correction. This bias is always zero for version 6 reprocessing—values were computed from the PR data and showed small systematic variation with the Sun elevation changes in the orbit plane, however the values were considered too small to be worth including without further evaluation. Finally, c_s is a sign factor applied for the yaw correction to account for the effect of changes in flying mode with yaw maneuvers. It is –1.0 for TRMM flying +X forward, and 1.0 for TRMM flying -X forward, and is switched in the middle of all yaw turns (see section A.1.1.5.3 for discussion of the flight mode handling). The angle around the orbit, α, is simply computed from the time, t, relative to the orbit start and orbit end times (t_s and t_e respectively):

$$\alpha = (t-t_s) / (t_e-t_s)$$

The orbit starting and ending times for TSDIS data granules are at the southernmost point in each orbit. The orbit angle parameter, α, is essentially an orbit mean anomaly angle relative to the southernmost reference direction over a nodal orbit period (t_e-t_s). This is essentially the same x-axis parameter used in getting fits to the orbit period errors from the PR roll data, so it is appropriate to apply.

The correctons for pitch, roll and yaw are simply added to the onboard estimates for these values to get the pitch, p, roll, r, and yaw, y, used for TSDIS geolocation computations.

$$p = p_o + \delta_p\, (\pi/180.)$$

$$r = r_o + \delta_r\, (\pi/180.)$$

$$y = y_o + \delta_y\, (\pi/180.)$$

where $\pi = 3.14159$ and the degrees-to-radians conversion factor (π/180.) is used because δ_p, δ_r, and δ_y are defined in degrees while the onboard values (p_o, r_o, y_o) and p, r, and y are defined in radians for subsequent usage.

Compute the Attitude Error Rotation Matrix

The Attitude Error rotation matrix gives the onboard attitude error relative to the coordinate frame to which the Attitude Control System (ACS) is presumed to be controlling at any given time. Since attitude errors are nominally expected to stay quite small in normal control modes, we will use a small angle rotation matrix using the ACS Ancillary Data Packet information. Thus the Attitude Error matrix is given by

$$[A'] = \begin{bmatrix} 1.0 & y & -p \\ -y & 1.0 & r \\ p & -r & 1.0 \end{bmatrix}$$

The small angle rotation matrix assumes small rotation errors (expressed in radians). Although it is not strictly a normalized rotation matrix, it is expected that for the range of attitude angles in the normal control mode, it will not introduce significant computational errors. This is the most efficient way, computationally, to use the attitude data provided. During yaw maneuvers, the interpretation of the attitude angles is as Euler angles in a 3-2-1 sequence.

Compute the S/C-to-ACS Rotation Matrix

The S/C-to-ACS rotation matrix is the product of the Attitude Error matrix, which is obtained from the position errors in X, Y, and Z directions, and the Axis Shuffling matrix, which is determined based on the spacecraft orientation. The “Axis Transformation” Matrix does not mix the axes randomly, but simply accounts for the basic control modes: +X forward, -X forward, or Y forward.

The Axis Shuffling Matrix will depend on the spacecraft orientation flag.

If the flag has a value of 0 (+x forward or forward) then

$$[X] = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

If the flag has a value of 1 (-x forward or backward)

$$[X] = \begin{bmatrix} -1.0 & 0.0 & 0.0 \\ 0.0 & -1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

If the flag has a value of 2 (-y forward or 90 deg. yaw mode)

$$[X] = \begin{bmatrix} 0.0 & 1.0 & 0.0 \\ -1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

Now the S/C-to-ACS matrix is given by

$$[S] = [X]\,[A']$$

Compute ACS-to-Nadir Rotation Matrix

The ACS-to-Nadir Rotation Matrix, N', provides an approximation of the rotation matrix between the ACS coordinate frame used onboard (relative to which yaw, pitch, and roll are defined for the onboard ACS ancillary data packet), and the nadir/reference frame (which is the "orbital" frame in which yaw, pitch, and roll are reported in the TSDIS L1B navigation structure). It is given by:

$$[N'] = [O]\,[N],$$

where [O] is the GCI to geocentric/inertial nadir frame rotation matrix and [N] gives the rotation from geodetic/inertial nadir frame and GCI as defined earlier.

The approximation lies in the assumption that [N] represents the onboard ACS frame rotation from GCI, whereas the actual ACS frame for onboard analysis uses a specially computed "tip angle" to define this frame relative to the orbital frame.

Compute the S/C-to-Nadir Rotation Matrix

The S/C-to-Nadir rotation matrix is given by

$$[N''] = [N']\,[S]$$

This is the rotation matix from which the pitch, roll, and yaw attitude angles are computed. These will be calculated with standard trigonometric formulas using an existing subroutine that allows use of any selected sequence of Euler angles.

Compute the GCI-to-S/C Rotation Matrix

The GCI-to-S/C rotation matrix, which is also called the Attitude matrix, is computed as follows.

As per the traditional definition of the attitude matrix in many applications, it is the product of the following:

$[O]$ = rotation from GCI to the orbital frame, and
$[N']^T$ = rotation from the orbital frame to the spacecraft. This is the transpose of theACS-to-Nadir [N'] matrix computed above

Now the GCI-to-S/C rotation (or Attitude) matrix,

$$[A] = [N']^T\,[O]$$

<u>Calculate Sensor Orientation Matrix</u>

The sensor orientation matrix rotates a vector from the sensor coordinate system to GCI coordinate system. This matrix is stored with the navigation data in TSDIS products.

Compute the product of the spacecraft attitude and the sensor alignment matrices :

$$[P] = [A][M]$$

where [A] and [M] were defined above .

Now the Sensor orientation matrix is the transpose of the above matrix:

$$[C] = [P]^T$$

This matrix [C] is stored with the navigation data for each scan.

Calculate Instantaneous Field of View (IFOV) Vector

The look vector D, shown in the Figure 2.1, is called the Instantaneous Field of View (IFOV) vector. Let D_S be the unit look vector in the sensor coordinates and D_i be the unit look vector in the GCI coordinates. Vectors D_S and D_i are computed as follows for TMI and VIRS. The exact start and end rotation angles, φ, for TMI and VIRS were computed from data supplied by the instrument manufacturer, in coordination with analysis by the respective instrument scientists. These values are all stored in sensor model parameter files for use by the Level 1 algorithms. For PR a table of the 49 beam directions in the sensor coordinates was provided by the Japanese algorithm developers and incorporated with the geolocation toolkit routines in the 1B21 algorithms.

Computation of Ds

The basic sensor scan model is a cone rotating about its axis at a uniform scan rate. Given a unit sphere and two reference vectors F and S based at the sphere's center, we compute the unit vector based at the sphere's center defined by an arc length θ from F and a rotation angle φ from S about F.

Let F and S be two unit vectors based at the center of a unit sphere. The unit vector D_S based at the sphere's center, defined at an arc length θ from F and a rotation angle φ from S about F, is obtained as follows:

Let I denote a plane containing the center of the unit sphere, with normal F and let PROJ_I^S be the projection of S on I. Then

$$\mathrm{PROJ}_I^S = S - \mathrm{PROJ}_F^S = S - (F \bullet S)\, F \qquad (2\text{-}1)$$

The desired vector D_S is given by:

$$D_S = F\cos(\theta) + [\,(\mathrm{PROJ}_I^S)\cos(\varphi) + (F \times \mathrm{PROJ}_I^S)\sin(\varphi)\,]\sin(\theta) \qquad (2\text{-}2)$$

Table 2.1 specifies the values for the above parameters for each of the instruments.

Table 2.1 Values for Equation (2-2)

Instrument	**F (sensor coordinates)**	**S (sensor coordinates)**	**θ**	**φ**
TMI	(0,0,1) - Z-axis	(1,0,0) - X-axis	49°	-64.4024° to +65.0967° in 208 equal steps
VIRS	(1,0,0) - X-axis	(0,0,1) - Z-axis	90°	-44.86° to +44.87° in 261 equal steps
PR	(1,0,0) - X-axis	(0,0,1) - Z-axis	Apprx. 90°*	Apprx. -17° to +17°*

- Vectors for PR defined by table supplied by NASDA

In addition, a correction is made to each D_S, to account for the nominal rotation rate of TRMM in inertial space during each scan. Since it is inefficient to re-compute the Sensor Orientation Matrix for the time of each pixel observation, this calculation is done for the mid –scan time. Since the orbit period rotation is always at a nominal rate about the pitch (Y) axis, a small adjustment of the D_S vectors in the body frame include a rotation about the Y body axis according to the orbit rate and the pixel observation offset from the mid-scan time. The D_S can normal be computed once for each L1 product, but they are recomputed in the case that there is a yaw maneuver which changes the direction of the body rotation. The change in the nominal orbit rate (the period is about 1% longer) after the boost to 402.5 kilometer operating altitude introduced a slight error in this correction for post-boost scans, but this does not introduce a significant error.

Computation of Di:

D_i, the IFOV vector in GCI coordinates, is computed by rotating D_S using the instrument rotation matrix. The instrument rotation matrix [C] is a product of the sensor alignment matrix [M] and spacecraft attitude matrix [A]:

$$D_i = [A]\,[M]\,D_S = [C]\,D_S \qquad (2\text{-}3)$$

Intersect With Oblate Earth

Let G be the target position vector in GCI coordinates and P the satellite position vector in GCI coordinates. P is derived from information in the ephemeris file using a first order adjustment for the exact satellite position at the time each pixel is recorded.

The distance, d_x, from the satellite to a target is calculated using an approach that assumes a straight line intersecting an ellipsoidal Earth. The equation for the ellipsoidal Earth in x, y, z coordinates can be written as:

$$x^2 + y^2 + z^2 / (1-f)^2 = r^2 \qquad (2\text{-}4)$$

where

f = flattening factor = $1 - (r_p / r)$

r = Earth's equatorial radius

r_p = Earth's polar radius

The parameterized vector equation for the target position is expressed as:

$$G = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} P_X + d_X D_{iX} \\ P_Y + d_X D_{iY} \\ P_Z + d_X D_{iZ} \end{bmatrix} \tag{2-5}$$

We get a quadratic equation by substituting into equation (2-4) the components x, y, and z as a function of d_X and regrouping terms. By solving the quadratic equation we get the distance to the intersection point. The solution for the quadratic equation is:

$$d_X = [- B \pm (B^2 - 4AC)^{1/2}] / 2A \tag{2-6}$$

where

$$A = (D_{iX}^2 + D_{iY}^2) / r^2 + (D_{iZ}^2 / r_P^2) \tag{2-7}$$

$$B = (D_{iX} P_X + D_{iY} P_Y) / r^2 + (D_{iZ} P_Z / r_P^2) \tag{2-8}$$

$$C = (P_X^2 + P_Y^2) / r^2 + (P_Z^2 / r_P^2) - 1 \tag{2-9}$$

If we get two real solutions for d_X, then the smaller one will be the required value for d_X. If we get only one solution, the line-of-sight is tangent to the ellipsoid. If we get no real solution, then the line of sight misses the ellipsoid. This would occur during a space view calibration maneuver.

Calculate Greenwich Hour Angle

The Flight Dynamics Facility (FDF) supplies the Greenwich Hour Angle (GHA) at the reference time of the daily Ephemeris, 0h Greenwich Mean Time at the starting date of the Ephemeris. This is usually the same as the Epoch time. This GHA is derived through a polynomial fit to Earth rotation data obtained from the U.S. Naval Observatory. The FDF routinely uses these earth rotation data to update their earth rotation coefficients.

A standard constant Earth rotation rate is used to extrapolate from the reference time of the GHA in the FDF file to the time of each pixel observation.

If the GHA calculated by the FDF is unavailable, the following procedure, which does not account for the impact of leap seconds, can be used as a backup or fallback

The Greenwich Hour Angle, h, is calculated with a polynomial approximation:

$$h = 100.4606184 + 0.9856473663\ d + 2.908 \times 10^{-13}\ d^2 \tag{2-10}$$

where

d = days since epoch J2000 (January 1 12:00:00, 2000) , with time-of-day expressed as fractional days and

$$d = jd - 2451545.50 + fd \tag{2-11a}$$

$$jd = 367 * iy - 7 * (iy + (im+9)/12) / 4 + 275 * im / 9 + id + 1721014 \tag{2-11b}$$

where

jd	=	the Julian day of the calendar date
iy, im, id	=	the integer year, month, and day respectively
fd	=	the fractional day

Nutation correction for the mean sidereal time can be applied as follows:

$$gha = h + d\psi * \cos(\varepsilon) + fd * 360.0 \tag{2-12}$$

Other parameters are calculated as follows:

$$l_s = 280.46592 + 0.9856473516\ d \tag{2-13}$$

$$g_s = 357.52772 + 0.9856002831\ d \tag{2-14}$$

$$l_m = 218.31643 + 13.17639648\ d \tag{2-15}$$

where

l_s = mean solar longitude (degrees)
g_s = Sun mean anomaly (degrees)
l_m = Moon mean longitude (degrees)

The nutation in longitude is computed as follows:

$$d\psi = [-17.1996 \sin(\Omega_m) + 0.2062 \sin(2\Omega_m) - 1.3187 \sin(2l_s) + 0.1426 \sin(g_s) - 0.2274 \sin(2l_m)] / 3600 \tag{2-16}$$

and

$$\Omega_m = 125.04452 - 0.0529537648\ d \tag{2-17}$$

where

Ω_m = ascending node of the Moon’s orbit

The obliquity of the ecliptic ε is given by:

$$\varepsilon = \varepsilon_M + d_\varepsilon \tag{2-18}$$

$$\varepsilon_M = 23.439291 - 3.560 \times 10^{-7}\ d \tag{2-19}$$

$$d_\varepsilon = [9.2025 \cos(\Omega) + 0.5736 \cos(2l_s)] / 3600 \tag{2-20}$$

where

ε_M = mean obliquity

d_ε = nutation in obliquity

Derive Geodetic Latitude and Longitude

The target position vector, G in GCI coordinates, from the Earth center at the line of intersection, is computed using equation (2-5). The coordinates of the geodetic system, i.e., geodetic latitude and longitude, are computed as follows:

$$\tan(lat_c) = G_z / (G_x^2 + G_y^2)^{1/2} \tag{2-21}$$

$$\tan(lat_d) = \tan(lat_c) / (1 - f)^2 \tag{2-22}$$

$$= G_z / (1 - f)^2 (G_x^2 + G_y^2)^{1/2} \tag{2-23}$$

where

lat_c = geocentric latitude
lat_d = geodetic latitude

Geocentric and geodetic longitudes are the same and they are calculated as follows:

$$\tan(lon_d) = \tan(lon_c) = G_y / G_x - (gha + \Omega * \Delta t) \tag{2-24}$$

$$\tan(gra) = G_y / G_x \tag{2-25}$$

$$lon_d = lon_c = gra - (gha + \Omega * \Delta t) \tag{2-26}$$

where

lon_c = geocentric longitude
lon_d = geodetic longitude
gha = Greenwich Hour Angle at the scan reference time
Ω = Spin rate of Earth (0.004178074622 deg/sec)
Δt = time offset from the scan reference time
gra = geocentric right ascension

2.2 INVERSE REFERENCING METHOD

The inverse referencing algorithm computes the fractional line and pixel number that look toward a selected geographic location. It is used for drawing coastlines on full resolution images. This is applied during geolocation QA phase D.

The inverse referencing computations are described in the following subsections.

- Get Navigation Data
- Calculate Earth Location Vector
- Calculate Dot Product of Line of Sight and Scan Axis
- Iterate to the Nearest Scan Lines
- Compute Scan Phase

- Interpolate Vectors

Get Navigation Data

The navigation data which are available in the Level 1B and higher products must be available so that this information can be used in the inverse referencing calculations. The following parameters are needed per scan line.

Spacecraft Position Vector, **P**, in GCI coordinates

Sensor Orientation Matrix, [**M**], for the Sensor to GCI rotation.

The following parameters are needed for the data span.

number of scan lines used for inverse referencing mapping
Start Time; time of first scan line at mid-scan
Bias time offset between Start Time and time of first pixel
time step between scan lines
number of pixels in the scan
time step between pixels along a scan
angle step between pixels along a scan
start rotation angle of first pixel relative to reference axis
Greenwich Hour Angle at start time
scan axis indicator
reference axis indicator
cosine of scan cone angle

Calculate Earth Location Vector

Given geodetic latitude and longitude coordinates, we want to calculate the Earth location vector in GCI coordinates. An existing routine can be applied for this calculation.

Calculate Dot Product or Line of Sight and Scan Axis

Given the Earth position vector and the Satellite position
vector, both in GCI coordinates, the line of sight vector from the satellite to the Earth, Q, is given by:

$$Q = G - P$$

The instrument spin axis, S, in GCI coordinates, is taken from one column of the Sensor Orientation Matrix, M. For VIRS this is the first column since the scan rotates about the X-axis, and for TMI this is the third column since the sensor rotates about the Z axis.

The dot product of Q and S is

$$\mathbf{d} = Q \, . \, S = Q(1)*S(1) + Q(2)*S(2) + Q(3)*S(3)$$

Iterate to the Nearest Scan Lines

Initially, the dot product, $\mathbf{d}_1$, will be calculated for the first scan line, and the dot product, $\mathbf{d}_n$, will be calculated for the last scan line. If these span the desired dot product, $\mathbf{d}_d$, for the scan line to intersect the selected Earth location, i.e. if:

$$\mathbf{d}_1 < \mathbf{d}_d < \mathbf{d}_n$$

or if

$$\mathbf{d}_1 > \mathbf{d}_d > \mathbf{d}_n$$

then one of the scan lines in the interval will scan close to the selected Earth location. A first estimate of the scan line number can be made by linear interpolation.

$$i1 = \text{int}\ \{ n * [(\mathbf{d}_d - \mathbf{d}_1) / (\mathbf{d}_n - \mathbf{d}_1)] \}$$

Then the dot product for this new scan line, $\mathbf{d}_{i1}$, is calculated, and it is determined which side of $\mathbf{d}_d$ this new dot product lies. This $\mathbf{d}_{i1}$ can substituted for either $\mathbf{d}_1$ or $\mathbf{d}_n$ in the above formula, whichever is on the same side of $\mathbf{d}_d$, and the above formula can be used to get a new line number estimate. This procedure can be repeated until the line number estimated does not change. At that point, this unchanged integer and the next one in the direction that still spans $\mathbf{d}_d$ are the adjacent scans passing closest to the selected Earth location.

Compute Scan Phase

The scan phase angle is given by the position of the line of sight vector Q relative to the scan axis, S, and a reference axis, R, indicating the zero rotation angle. This reference axis, like the scan axis, is given in GCI coordinates by an appropriate column of the sensor orientation matrix [M]. This computation will be based on an existing routine in the GSC attpac library, PHASED.

Interpolate Vectors

To account for the spacecraft motion and the time varying attitude in inertial space, we can interpolate all of the following vectors to the time of the pixel observation;

- P, the spacecraft position
- S, the instrument scan axis
- R, the instrument scan reference axis

Since the change in all of these vectors from one scan line to the next is quite small, we can justify applying a simple linear interpolation to all the components, and normalize the result if appropriate (as for unit vectors). This interpolation is done as follows. Let d be the fraction between zero and one that gives the amount to interpolate between the vectors V1 and V2.

$$v1 = \begin{bmatrix} x1 \\ y1 \\ z1 \end{bmatrix}$$

$$\lceil \quad x2 \rceil$$

$$v2 = \begin{vmatrix} y2 \\ z2 \end{vmatrix}$$

The vector V, at fraction d from V1 to V2, is given by

$$v = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x1 + d\,(x2 - x1) \\ y1 + d\,(y2 - y1) \\ z1 + d\,(z2 - z1) \end{bmatrix}$$

2.3 ANCILLARY DATA CALCULATIONS

This section deals with the computations of the several parameters needed for both metadata and Level 1B data. The main computations are zenith and azimuth angles of the satellite, zenith and azimuth angles of the Sun, solar beta angle, and, the orbital (or Keplarian) elements.

Satellite Zenith and Azimuth Angles

The satellite zenith angle is defined as the angle between the local vertical at a given Earth location and a vector from that location to the position of the satellite. The computation method is obtained from and Patt and Gregg (1994).

Let S be the local Position vector and S_N, S_E, and S_V be the north, east and vertical components of S respectively.

The Zenith angle, theta = $\arctan\{(S_N^2 + S_E^2)^{1/2}/S_V\}$

The Azimuth angle, phi = $\arctan(S_E / S_N)$

Solar Zenith and Azimuth Angles

The determination of the solar zenith and azimuth angles is essentially the same as the satellite zenith and azimuth, with the Earth-to-Sun vector in GCI coordinates replacing the pixel-to-spacecraft vectors. Thus we require the Earth-to-Sun vector, which will be computed using a model developed by Patt and Gregg (1994) and implemented in Pathfinder project.

Solar Beta Angle

The solar beta Angle gives the elevation of the Sun from the satellite orbit plane, with positive being toward the positive orbit normal direction. The solar beta angle calculation involves the computation of Sun vector, which will be carried out using Pathfinder code, and the orbit normal vector as follows from the orbit parameters.

Let i be the inclination of the orbit and o be the right ascension of ascending node.

$$\begin{bmatrix} \sin(i)\sin(o) \end{bmatrix}$$

The unit orbit normal, **N** = $\begin{matrix} \vert & -\sin(i)\cos(o) & \vert \\ \lfloor & \cos(i) & \rfloor \end{matrix}$

Let **S** be the unit sun vector and p is the dot product of **N** and **S**.

$$p = \mathbf{N} \bullet \mathbf{S}$$

Now the Solar Beta angle, B, is just the complement of the arc-cosine of this dot product,

$$B = 90 - \cos^{-1}(p)$$

Since the solar beta angle for TRMM will vary between +/- 59 degrees (this is the sum of the 35 degree TRMM orbit inclination and the 23.5 degree tilt of the ecliptic plane from the Earth equatorial plane, which defines the total possible variation of the Sun from the orbit plane), there is no need for special checks for p being near plus or minus one.

Orbital Elements

The Keplerian elements (also known as “orbital” or “tracking” elements) are a set of six quantities that completely describe the shape and orientation of an elliptical orbit around the Earth, and the position of a satellite in that orbit at a given epoch. These six quantities are:

- The semi-major axis .
- The eccentricity. These first two parameters specify the size and shape of the ellipse.
- The inclination of the plane of the orbit measured from the equator. An inclination of 0^o corresponds to an orbit that coincides with the orbital plane, an inclination of 90^o corresponds to an orbit that passes over the poles, and an inclination of 180^o corresponds to an orbit that coincides with the equatorial plane but in which the satellite moves in a direction opposite the planet’s rotation (that is, a retrograde orbit).
- The right ascension of the ascending node. That is, the celestial longitude at which the satellite crosses the equator going north.
- The argument of perigee. This is the angular separation (argument) between the ascending node and perigee, where the satellite makes its closest approach to the gravitational center of the planet.

The true anomaly. The angle in the plane of the satellite orbit between perigee and the satellite position, measured in the direction of the satellites position.

A full description of the Keplerian elements can be found on the World Wide Web at the URL http://www.jpl.nasa.gov/basics.

The elements can be calculated given the position and velocity vectors of the spacecraft and a simple two body ideal orbit model, and are carried out by using a reusable routine RECEL obtained from the ORBPACK library. But these elements cannot be used with a simple orbit model to compute the TRMM spacecraft position, because it is in a low-Earth orbit. Such an orbit, generally considered as one at an altitude less than 2,000 km above the Earth’s surface and having an orbital period of less than two hours, is subject to the effects of drag in the Earth's atmosphere and to the gravitational effects of the Earth's oblateness. Consequently, the Keplerian elements vary with time. In the context of low-Earth orbit, these time variable elements are also called osculating elements.

3. L1B TMI CALIBRATION ALGORITHM

3.1 RADIOMETRIC CALIBRATION

The TMI Earth view data are calibrated on a scan-by-scan basis. Each scan begins with 130° of Earth view followed by a cold load reference measurement, and then a hot load reference measurement. These reference measurements, along with the known temperatures of the calibration loads, serve to calibrate Earth view data.

Both Earth view data and reference data are provided for each scan in the same science data telemetry packet (TMI Science Packet, APID 52). The Earth view data consist of 104 pixels for low-resolution channels (channels 1 through 7), and 208 pixels for high-resolution channels (channels 8 and 9). Each hot load and cold load reference measurement consists of 8 samples for low-resolution channels, and 16 samples for high-resolution channels. The hot load temperatures, measured by three thermistors, are also telemetered for each scan. The temperature of cold space ("cold load") is assumed as 2.7 Kelvin for channels 1-7, and 3.2 Kelvins for channels 8-9 (Hollinger 1990; Fixsen *et al*. 2004). Telemetry of each scan also includes hot load bridge reference measurements and channel gain values, which are not used in the current radiometric calibration algorithm.

Radiometric Calibration

The object of radiometric calibration is to obtain antenna temperature for each pixel using a first order 2-point linear calibration equation of the radiometer transfer function. The slope (A_n) and intercept (B_n) of this linear equation is determined by the hot load and cold load reference measurements, as well as the hot load and cold load temperatures.

The temperature of the hot load, measured by three precision thermistors, is provided in the science telemetry. It is received as digital counts (P_m). The counts are converted to Kelvin (T_m) by

$$T_m = a_{1m} + a_{2m}P_m + a_{3m}P^2_m + a_{4m}P^3_m + a_{5m}P^4_m + a_{6m}P^5_m \qquad (3\text{-}1)$$

where

T_m = thermistor temperature in Kelvins for thermistor m
m = thermistor number (1, 2, or 3)
$a_{1m} \ldots a_{6m}$ = conversion coefficients for thermistor m ((algorithm tuning parameters, see Table 3.1)
P_m = thermistor temperature telemetry in digital counts for thermistor m

Table 3.1 Conversion coefficients for the three thermistors

m	a_{1m}	a_{2m}	a_{3m}	a_{4m}	a_{5m}	a_{6m}
1	2.5415e+2	1.6631e-2	8.1994e-7	-8.5383e-11	3.0207e-14	-2.7487e-18
2	2.54457e+2	1.6793e-2	5.4255e-7	1.0260e-10	-2.4391e-14	2.9235e-18

3	2.54266e+2	1.6655e-2	7.7749e-7	-6.2308e-11	2.4515e-14	-2.2180e-18

The calculation of the arithmetic average of the three thermistors (Th) is performed:

$$Th = (T_{m=1} X + T_{m=2} Y + T_{m=3} Z) / (X + Y + Z) \qquad (3\text{-}2)$$

where

Th = average temperature of all thermistors in Kelvins
$T_{m=1,2,3}$ = three thermistor temperatures in Kelvin, from Eq. 3-1
X, Y, Z = thermistor on/off flag, nominally equal to 1, but can be set to zero to exclude a non-working thermistor

The hot load temperature from thermistors is corrected to account for radiative coupling between the hot load and the top plate of rotating drum assembly that faces the hot load. The top radiator telemetry comes from the TMI housekeeping file. Its temperature is calculated by converting the digital counts to temperature in Celsius:

$$Trp = b_1 + b_2Prp + b_3Prp^2 + b_4Prp^3 + b_5Prp^4 + b_6Prp^5 \qquad (3\text{-}3)$$

where

Trp = top radiator temperature in Celsius
Prp = top radiator temperature telemetry in digital counts (byte # 66 in APID 47)
b_1 ... b_6 = conversion coefficients (algorithm tuning parameters, see Table 3.2)

Table 3.2 Top plate of rotating drum assembly conversion coefficients

b_1	b_2	b_3	b_4	b_5	b_6
-2.397e+2	1.588	6.346e-4	0	0	0

Then, the hot load temperature is corrected:

$$Tch = Th + \alpha(Trp + Kos - Th) \qquad (3\text{-}4)$$

where

Tch = corrected hot load temperature in Kelvins
Th = hot load temperature in Kelvins, from Eq. 3-2
Trp = top radiator temperature in Celsius, from Eq. 3-3
α = correction coefficient (algorithm tuning parameter s, 0.0 at the current version of the algorithm)
Kos = offset for Kelvin temperature (= 273.15 degree)

The average voltage of the hot load reference measurements is calculated:

$$VH_n = (\Sigma^8 \ Vh_{k,n}) / 8 \qquad \text{for } n = 1 \text{ to } 7 \qquad (3\text{-}5a)$$

$$VH_n = (\Sigma^{16} \ Vh_{k,n}) / 16 \qquad \text{for } n = 8 \text{ to } 9 \qquad (3\text{-}5b)$$

where

VH_n = average of hot load voltages for channel n (over 8 samples for low resolution channels, and over 16 samples for high resolution channels)
$Vh_{k,n}$ = voltage of hot load for channel n, sample k
n = channel number (1 to 9)
k = sample number (1-8 for channel 1-7 and 1-16 for channel 8-9)

The average voltage of the cold load reference measurements is calculated:

$$VC_n = (\sum^{8} Vc_{k,n}) / 8 \qquad \text{for } n = 1 \text{ to } 7 \qquad (3\text{-}6a)$$

$$VC_n = (\sum^{16} Vc_{k,n}) / 16 \qquad \text{for } n = 8 \text{ to } 9 \qquad (3\text{-}6b)$$

where

VC_n = average of cold load voltages for channel n (over 8 samples for low resolution channels, and over 16 samples for high resolution channels)
$Vc_{k,n}$ = voltage of cold load for channel n, sample k
n = channel number
k = sample number

The Earth view temperature is calibrated by the radiometer transfer equation. Based on above reference parameters, the slope (A_n) and intercept (B_n) of the radiometer transfer function are calculated:

$$A_n = (Tc - Tch) / (VC_n - VH_n) \qquad (3\text{-}7a)$$

$$B_n = Tch - A_n VH_n \qquad (3\text{-}7b)$$

where

A_n = slope calibration coefficient for channel n
B_n = intercept calibration coefficient for channel n
Tc = cold load temperature, 2.7°K for channels 1 to 7, and 3.2°K for channels 8 and 9.
Tch = hot load temperature, from Eq. 3-4

Finally, the antenna temperature of each Earth view pixel is calibrated:

$$Ta_{n,i} = A_n V_{n,i} + B_n \qquad (3\text{-}8)$$

where

$Ta_{n,i}$ = calibrated antenna temperature in Kelvins, for channel n, pixel i
$V_{n,i}$ = raw voltage of Earth view in digital counts, for channel n, pixel i
i = pixel number (1 to 104 for channels 1-7, and 1 to 208 for channels 8-9)

Reference Data Analysis

The hot load and cold load reference measurements and some other reference telemetry data are analyzed during the calibration. These results of the analyses, however, are not incorporated into the calibration. They are provided to the TMI instrument scientist for the purpose of health and safety monitoring of the TMI instrument.

The standard deviations of the voltages of hot load measurements are calculated:

$$VHSD_n = \{[\Sigma^8 (Vh_{k,n} - VH_n)^2] / 7\}^{1/2} \qquad \text{for } n = 1 \text{ to } 7 \qquad (3\text{-}9a)$$

$$VHSD_n = \{[\Sigma^{16} (Vh_{k,n} - VH_n)^2] / 15\}^{1/2} \qquad \text{for } n = 8 \text{ to } 9 \qquad (3\text{-}9b)$$

where

$VHSD_n$ = standard deviation of hot load measurements for channel n
$Vh_{k,n}$ = voltage of hot load for channel n, sample k
VH_n = average of hot load samples, from Eqs. 3-5a and 3-5b
n = channel number (1 to 9)
k = sample number (8 or 16)

The standard deviations of the voltages of cold load measurements are calculated:

$$VCSD_n = \{[\Sigma^8 (Vc_{k,n} - VC_n)^2] / 7\}^{1/2} \qquad \text{for } n = 1 \text{ to } 7 \qquad (3\text{-}10a)$$

$$VCSD_n = \{[\Sigma^{16} (Vc_{k,n} - VC_n)^2] / 15\}^{1/2} \qquad \text{for } n = 8 \text{ to } 9 \qquad (3\text{-}10b)$$

where

$VCSD_n$ = standard deviation of cold load measurements for channel n
$Vc_{k,n}$ = voltage of cold load for channel n, sample k
VC_n = average of cold load samples, from Eqs. 3-6a and 3-6b
n = channel number (1 to 9)
k = sample number (8 or 16)

Based on above standard deviations, the noise equivalent delta temperatures are calculated:

$$NEDTH_n = VHSD_n A_n \qquad \text{for hot load measurement} \qquad (3\text{-}11a)$$

$$NEDTC_n = VCSD_n A_n \qquad \text{for cold load measurement} \qquad (3\text{-}11b)$$

where

$NEDTH_n$ = noise equivalent delta temperature of hot load measurement for channel n

$NEDTC_n$ = noise equivalent delta temperature of cold load measurement for channel n

$VHSD_n$ = standard deviation of hot load measurements for channel n, from Eqs. 3-9a and 3-9b
$VCSD_n$ = standard deviation of cold load measurements for channel n, from Eqs. 3-10a and 3-10b
A_n = calibration coefficient for channel n, from Eq. 3-7a

n = channel number (1 to 9)

85.5 GHz receiver temperature (Tmix), which is the sensor's average thermal temperature, is calculated by converting from digital counts to Celsius:

$$Tmix = c_1 + c_2Pmix + c_3Pmix^2 + c_4Pmix^3 + c_5Pmix^4 + c_6Pmix^5 \quad (3\text{-}12)$$

where

Tmix = 85.5 GHz receiver temperature in Celsius
Pmix = 85.5 GHz receiver temperature in digital counts (byte # 62 in APID 47)
$c_1 \ldots c_6$ = conversion coefficients (algorithm tuning parameters supplied by the TMI instrument scientist)

3.2 ANTENNA CORRECTION

Corrections of the calibrated antenna temperatures are performed following the radiometric calibration in order to transform calibrated antenna temperature to brightness temperature. The antenna temperature of each pixel is corrected for the co-polarization, cross polarization, and spillover effects of the antenna.

$$Tb_{n,i} = C_nTa_{n,i} + D_nTa^*_{n,i} + E_n \quad (3\text{-}13)$$

where

$Tb_{n,i}$ = brightness temperature in Kelvins, for channel n, pixel i
$Ta_{n,i}$ = TMI antenna temperature in Kelvins, for channel n, pixel I, from Eq. 3-8
$Ta^*_{n,i}$ = antenna temperature of cross polarized channel of the $Tar_{n,i}$ (for example, if Tar of channel 19V is calculated, then Tar* is the antenna temperature of channel 19H)
C_n, D_n, E_n =APC coefficients for channel n (algorithm tuning parameters see Table 3.3)

The 21V channel (n=5) does not have a cross-polarized counterpart (there is no 21H channel), the cross-polarized term in the Eq. 3-13 is equal to zero (coefficient $D_{n=5}$ is zero).

Table 3.3 TMI APC coefficients for 9 channels used in version 4 (and earlier) L1B algorithm

n	1	2	3	4	5	6	7	8	9
C_n	1.0200	1.0209	1.0267	1.0277	1.0205	1.0372	1.0317	1.0331	1.0412
D_n	0.0037	0.0047	0.0044	0.0046	0.0000	0.0246	0.0192	0.0208	0.0304
E_n	0.0163	0.0163	0.0223	0.0230	0.0248	0.0127	0.0124	0.0123	0.0109

3.3 POST LAUNCH VALIDATION AND CALIBRATION

The post-launch validation and calibration employ the following aspects:

- Special limit checks
- Statistical analysis

- Inter-comparison
- Deep Space Satellite Maneuver

Special Limit Checks

Quality assurance takes place following the radiometric calibration and the corrections for each scan processing. The limit checks are performed on following calibration parameters:

- thermistor temperatures T_m and Tch
- hot load bridge reference measurements
- average voltages of hot load and cold load measurements VH_n and VC_n
- slope coefficient of the radiometer transfer equation A_n

The limit values are provided by the TMI instrument scientist as the algorithm tuning parameters. The results of those that are out of limits are recorded and reported to the TMI instrument scientist via the intermediate output file.

Statistical Analysis of Observations

Statistical analysis is used to find systematic errors. This method only requires TMI observations. Typically, TMI is a conically scanning radiometer, maintaining a nearly constant earth incidence angle through out the scan. In principle, the measurement characteristics of the sensor should be completely independent of scan position. However in practice, problems such as spacecraft attitude errors, obstructions in field of view, and sidelobes seeing the spacecraft may result in systematic errors that are a function of scan position. Given enough observations, we can determine the along-scan error by making the assumption that the effect of weather and surface features is, on the average, the same at all scan positions.

Inter-comparison of TMI with SSM/I.

The inter-comparison compares results between the TMI observations and collocated observations taken by the Special Sensor Mircowave Imager (SSM/I) and other instruments. SSM/I has proven to be a very stable sensor, and its absolute calibration as compared to radiative transfer models is estimated to be between 1 and 2 K. Thus it can be used as a calibration source for TMI except for the 10.7 GHz channels which are missing from SSMI. The TMI instrument is similar to SSM/I, with some notable exceptions. TMI has an additional feedhorn to accommodate the 10.7 GHz channel, and the frequency of the water vapor channel for TMI is 21.3 GHz as compared to 22.235 for SSM/I. TMI also has a wider active scan (130° versus 102° for SSM/I), and the fabrication of TMI's main reflector is slightly different. Radiative transfer models can be used to normalize the SSM/I T_a to a constant incidence angle of 52.75° (the TMI value) and to convert the 22.235 GHz observations to a frequency of 21.3 GHz (the TMI value).

Deep-Space Observations

Analyzes of the TMI observations during orbit maneuvers in which the TMI main reflector observed cold space may provide good indicators of TMI calibration quality.

As part of the Clouds and the Earth's Radiant Energy System (CERES) calibration process, the TRMM platform was pitched to view deep space rather than the Earth during several orbits in January 1998 and again in September 1998. Although not intended for the benefit of the TMI instrument, these maneuvers proved to be extremely useful for calibrating TMI and for confirming the other calibration methods. When the spacecraft is pitched 180° relative to the Earth nadir, the TMI main reflector has a clear view of deep space which is at a nearly uniform temperature of 2.7 K. Given a homogenous scene at a known temperature, calibration errors are readily detectable.

The above validation analysis was performed by Wentz et al. (2001). Following is a summary of their findings and the results applied to the version 5 TMI L1B algorithm.

Validation Results

Wentz et al. (2001) detected two types of calibration errors using statistical regression, TMI-SSM/I inter-comparison, and deep-space maneuver data: 1) a small (~ 1 K) systematic error that is correlated with the scan position, and 2) a warm bias in the TMI T_A that is related to the temperature of the scene being viewed by the main reflector. The along-scan error is determined from two completely independent methods: monthly averages of ocean observations and the analysis of the deep-space observations. The two approaches yield similar along-scan errors.

The comparison of the TMI and SSM/I antenna temperatures for all 7 channels shows nearly the same characteristic. For very warm Earth scenes (295 K), the TMI and SSM/I values are in close agreement, but as the scene temperature decreases, the TMI T_A shows a warm bias relative to SSM/I. For cold ocean observations, the TMI-SSM/I offset reaches values as high as 5 K.

The deep-space observations provide an independent assessment of the TMI warm bias. Extrapolating the TMI-SSM/ I comparisons to a scene temperature of 2.7 K predicts a TMI warm bias near 10 K. The maneuver was done three times over a period of nine hours. At the start of the maneuver, the main reflector is viewing the Earth and the cold mirror sees deep space. As the spacecraft is pitched, the footprint of the main reflector moves towards the Earth's limb and then views deep space. This transition is shown by a steep reduction in counts. Meanwhile, the cold mirror is going through an opposite transition, going from deep space to viewing the Earth. During a brief period of time (~100 scans), both the main reflector and the cold mirror have a clear view of deep space. The results obtained from the deep space observations are remarkably consistent with those coming from statistical analyses involving Earth observations.

Following the TRMM TMI after launch validation and calibration results, the TSDIS version 6 L1B algorithm made the following revisions.

Along-Scan Correction

The along-scan correction of the antenna temperature is performed for version 5 TMI L1B algorithm following the Wentz's (2001) post launch validation results:

$$Taa_{n,i} = Ta_{n,i} - Ca_{n,i} \qquad (3\text{-}14)$$

Where

$Taa_{n,i}$ = along-scan corrected antenna temperature in Kelvin, for channel n, pixel i
$Ta_{n,i}$ = antenna temperature in Kelvin, for channel n, pixel i, from Eq. 3-8
$Ca_{n,i}$ = along-scan correction coefficients (algorithm tuning parameters see TMI tuning data file in version 6 TMI L1B algorithm)

Warm Bias Correction

The TMI warm bias is also corrected in version 5 & 6 TMI L1B algorithm using Wentz' s results:

$$Tar_{n,i} = Ar_n \, Taa_{n,i} + Br_n \qquad (3\text{-}15)$$

where

$Tar_{n,i}$ = TRMM rollover maneuver corrected antenna temperature in Kelvin, for channel n, pixel i
$Taa_{n,i}$ = along-scan corrected antenna temperature in Kelvin, for channel n, pixel i, from Eq. 3-14
Ar_n = slope coefficient of TRMM rollover maneuver correction, for channel n (algorithm tuning parameters see Table 3.4)
Br_n = intercept coefficient of TRMM rollover maneuver correction, for channel n (algorithm tuning parameters see Table 3.4)

Table 3.4 Coefficient of TMI rollover maneuver correction used in version 6 L1B algorithm

n	10V	10H	19V	19H	21V	37V	37H	85V	85H
Ar_n	1.0395	1.0351	1.0443	1.0439	1.0484	1.0473	1.0436	1.0492	1.0465
Br_n	-11.64	-10.34	-13.07	-12.96	-14.28	-13.95	-12.85	-14.51	-13.73

APC Correction

The procedures of Version 5 APC corrections are similar to those described in Section 3.2. However, the SSM/I APC coefficients (Table 3.5) were used to replace the TMI APC coefficients (Table 3.3) and the antenna temperatures are corrected using both along scan and rollover corrections.

$$Tb_{n,i} = C_n Tar_{n,i} + D_n Tar^*_{n,i} + E_n \qquad (3\text{-}16)$$

$Tb_{n,i}$=brightness temperature in Kelvins, for channel n, pixel i
$Tar_{n,i}$=TMI rollover maneuver corrected antenna temperature in Kelvins, for channel n, pixel I, from Eq. 3-15
$Tar^*_{n,i}$= MI rollover maneuver corrected antenna temperature of cross polarized channel of the $Tar_{n,i}$
C_n, D_n, E_n = SSM/I APC coefficients for channel n (algorithm tuning parameters see Table 3.5)

Table 3.5 SSM/I APC coefficients for 9 channels used in version 5 (and later) L1B algorithm

n	1	2	3	4	5	6	7	8	9
C_n	1.02001	1.02094	1.03698	1.0277	1.02151	1.03681	1.04217	1.02632	1.03219
D_n	0.00375	0.00468	0.00394	0.00544	0.0000	0.02226	0.02762	0.01432	0.02019
E_n	0.0163	0.0163	0.0223	0.0230	0.0248	0.0127	0.0124	0.0123	0.0109

4. L1B DATA PRODUCT

4.1 LEVEL 1B PROCESSING

Satellite Maneuver

The TRMM satellite occasionally performs special maneuvers. These maneuvers take place to support the calibration and maintenance of the various instruments on board. The TRMM instruments will continue collecting data during satellite maneuvers. The collected data may not always have scientific value. It is left up to higher level processing to decide whether or not the collected, geolocated, and calibrated data are useful. Annotations are made in the data QA fields to indicate that maneuvers were in progress while the associated data were collected. The geolocation quality flag (see Volume 2 - Section 3.2.2.3) will record when special maneuvers are taking place.

TSDIS recognizes satellite maneuvers by reading the Attitude Control System (ACS) packet (APID 45) fields “ACS Spacecraft Orientation,” “ACS Control Mode,” and “Yaw Update Status” (bytes 46, 47, and 48 respectively). The TRMM satellite is *not* undergoing maneuvers when all of the following statuses are as follows:

- Current Spacecraft Orientation is one of the following:
  - +x forward
  - −x forward
- and Current ACS Mode is Nominal

The ACS Control Mode information will be kept in the L1B products and subsequent users will be expected to check these flags as needed. Otherwise, the L1B processing proceeds normally. If the sensor line-of-sight misses the Earth, flag values will be entered for latitude and longitude in the geolocation results. The geolocation quality flag will include an indication if maneuvers are in progress (making geolocation less reliable).

Common Processing

L1B Common Processing consists of adjusting the UTCF, determining if a science or HK packet is valid, check the time of a packet, checking for MDULs, getting the time from a packet or header, and writing a warning message to a log file.

TMI L1B Processing

L1B TMI Processing processes the L1B11 algorithm. This processing includes geolocation and calibrations of TMI data, and QA processing. L1B TMI Processing is composed of the following sections:

- Activation
- Execution

- Termination

Activation

The scheduler spawns L1BTMImain as an autonomous process upon the available of a TMI Level 1A granule. The scheduler passes it several parameters as follows:

- Granule ID of the L1A granule that is the input data to the L1B TMI Processing
- Granule ID of the L1B granule that is to contain the output data
- Granule ID of the ephemeris data file which is used for geolocation
- Granule ID of the limit checks log files which are used for QA

Execution

The flow chart for executing the main program is shown in Figure 4.1.

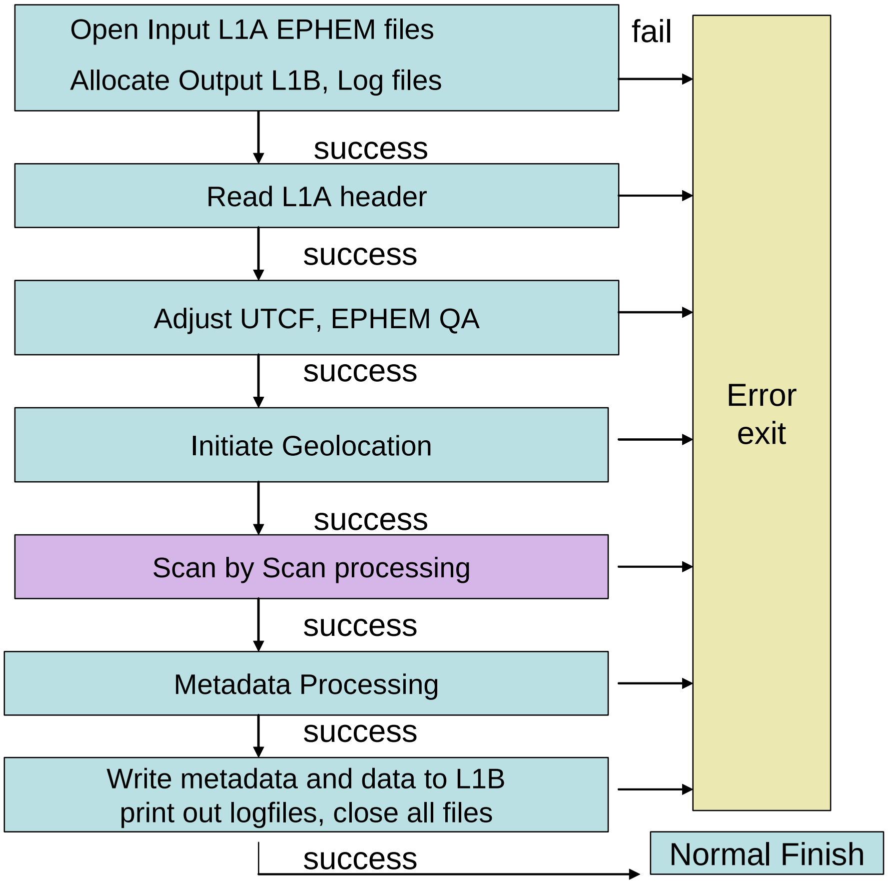


Fig. 4.1 Flow char for executing the L1B main program

First, module L1BTMImain opens the input L1A granule by calling the utility UTopen, the output L1B granule by calling the toolkit function TKopen, and the ephemeris file by calling GEOopenEphem. The log files is also opened.

Next, L1BTMImain calls the utility UTreadHeader to obtain the granule header from the L1A granule. L1BTMImain then calls L1BadjustUTCF to update the UTCF time if it was corrected, and carry out QA on the update. After UTCF is updated, L1BTMImain calls GEOinitGeolocation which reads the model parameters file and carries out various initial calculations required for GEOprocessGeolocation.

Next, L1BTMImain calls the function L1BdoScans, the driver routine which manages the scan-by-scan processing. It determines the number of science packets in the granule and loops through this number. For each packet, routines are called to unpack, geolocate, calibrate, perform QA, collect metadata and output the scan. L1BdoScans also reads MDUL packets and call geolocation, metadata collection and outputting routines for each missing scan.

Upon completion of L1BdoScans, L1BTMImain calls the function L1BfinalizeMetadata which takes the L1Bmetadata structure and calculates the final values for all metadata items.

Next, L1BTMImain calls L1BputMetadata which outputs the metadata to the Metadata database and outputs the PS and ECS metadata to the L1B granule. As the final step, the module L1BTMImain closes the input L1A granule by calling the utility UTclose, the output L1B granule by calling the toolkit function TKclose, and the ephemeris file by calling GEOcloseEphem. The limit checks log file is also closed.

The flow chart for executing scan-by-scan process is shown in Figure 4.2. If not success, the program exit with an error report.

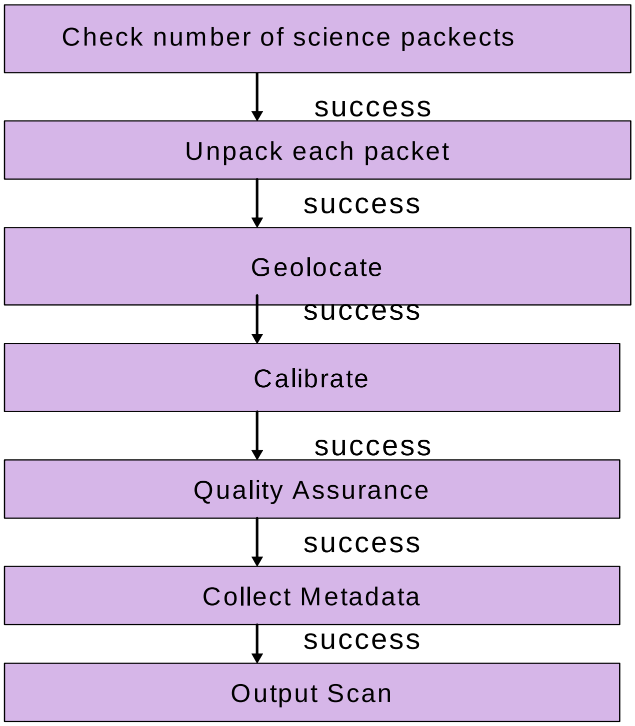


Fig. 4.2 Flow chart for executing the L1B scan-by-scan processing

Termination

When L1BTMImain finishes execution, successfully or otherwise, it passes a return code to the Scheduler and stops (ceasing to exist as a spawned process). The return code tells the Scheduler the reason for termination. The following return states are defined:

- problem reading input (i.e., no L1B created)
- problem creating output (i.e., no L1B created)
- fatal anomaly interrupted processing (i.e., partial L1B created)
- normal termination (i.e., L1B created)
-

4.2 DATA STRUCTURE

The format of TMI L1B product is HDF format. TSDIS has developed IDL tools to view the data and both C and Fortran programs to help user to read the data. The details of the Metadata and data structure should be found in the TSDIS L1 SDS document.

Metadata

The TMI L1B software generates much information that eventually becomes metadata. Throughout the software execution, the information is kept in local data structures. Near the conclusion of each executable's life, calls are made to the Toolkit function and information from the local data structures is recorded as metadata.

Structure

Key level 1B structures will be composed of the following:

- Common Structures
- Orbit Definition Structures
- Split L0 Housekeeping Structures
- Granularization Structures
- Geolocation Structures
- VIRS Structures
- TMI Structure

4.3 TUNING PARAMETERS

Various algorithms within the level 1 processing require a handful of parameters that are called "tuning parameters." The values are set manually to some initial values (during pre-launch software development) and possibly changed by scientists throughout the mission depending on observation of the algorithm performance and external physical changes. For the most part, these parameters are constants that may, but seldom, change. This section describes the data files associated with the level 1 processing software. These files will reside in the same directory structure as their associated algorithm.

Table 4-1 summarizes the current list of files and cross references them to the module that creates each.

Table 4.1 List of Data Files

File Name	Created By/Updated By
geolocationTuningParameters	(manually)
TMIcalibrationTuningParameters	(manually)
VIRScalibrationTuningParameters	(manually)
EarthAndSensorModelParameters	(manually)
granularizationAlgConstants	(manually)
VIRSAlgConstants	(manually)
orbitStartTimes	L1AgetOrbitStartTimes

Geolocation QA Tuning Parameters

File Name:	geo_qa_params.dat
Contents:	Parameters that modify the function of the associated algorithm.
Used By:	geolocation algorithm

TMI Calibration Tuning Parameters

File Name: tmi_tuning.dat

Contents: Parameters that modify the function of the associated algorithm. The following parameters are defined:

$gl_1 \ldots gl_9$ = 9 gain low limits, in digital counts (type integer)
$gh_1 \ldots gh_9$ = 9 gain high limits, in digital counts (type integer)
$tl_1 \ldots tl_3$ = 3 thermistor low limits, in digital counts (type integer)
$th_1 \ldots th_3$ = 3 thermistor high limits, in digital counts (type integer)
$a_{1m} \ldots a_{6m}$ = 6 conversion coefficients for thermistor m, m = 1...3 (type real)
$b_1 \ldots b_6$ = 6 conversion coefficients for top radiator temperature (type real)
$c_1 \ldots c_6$ = 6 conversion coefficients for 85.5 GHz receiver temperature (type real)
X, Y, Z = nominally equal to 1, but can be set to zero to exclude a non-working thermistor (type integer)
α = correction coefficient (approximately 0.01, type real)
NumScans = number of scans over which to average (10 scans, type integer)
Tc_{1to7} = cold space temperature (2.7°K for channels 1 to 7, type real)
Tc_{8to9} = cold space temperature (3.2°K for channels 8 and 9, type real)

mr = number of regions r along scan (type integer)
C_n, D_n, E_n = 3 weighting coefficients for antenna correction for channel n (type real)
maxGainDev$_n$ = maximum deviation in gain of channel n expressed as a fraction of the gain value from the previous scan (type real)
maxTherm$_q$ = maximum thermistor value for thermistor q (q = 1, 2, 3), digital counts (type integer)
minTherm$_q$ = minimum thermistor value for thermistor q (q = 1, 2, 3), digital counts (type integer)
maxHLbrRefPos = maximum hot load bridge reference value for positive bridge voltage, digital counts (type integer)
minHLbrRefPos = minimum hot load bridge reference value for positive bridge voltage, digital counts (type integer)
maxHLbrRefZer = maximum hot load bridge reference value for near zero bridge voltage, digital counts (type integer)
minHLbrRefZer = minimum hot load bridge reference value for near zero bridge voltage, digital counts (type integer)

Used By: algorithm TMI L1B11. Parameters added or revised in version 6 are described in Section 3.3.

<u>Geolocation Sensor Model Parameters</u>

File Name: geo_model_params_TMI.dat

Contents: Parameters that describe the Earth and Sensor for the geolocation algorithm. The following parameters are defined and approximate default values are noted.

Axis of Scan (+/-1,2,or 3 for +/-X,Y or Z) :3
Ref. Axis of Rotation (+/-1,2,or 3 for +/-X,Y or Z) :1
Scan Cone Angular Radius (deg) :49.0
Starting Rot. Angle relative to Ref. Axis (deg) :-64.4024
Total Rotation Spanned for Scan (deg) :129.4991
Active Scan Duration (sec) :0.6831
Time Bias for the First Pixel :0.127194
Alignment Angle - First Rotation (deg) :0.0
Alignment Angle - Second Rotation (deg) :0.0
Alignment Angle - Third Rotation (deg) :0.0
Euler Rotation Sequence - First Rotation Axis :1
Euler Rotation Sequence - Second Rotation Axis :2
Euler Rotation Sequence - Third Rotation Axis :3

Other geo parameter files used in the V6 L1B algorithm are: geo_qa_params.dat, geo_ptch_corr.dat (post-boost correction), geo_roll_corr.dat (post-boost correction).

5. REFERENCES

Andrews, S.F. and S. Bilanow, 2002: Recent Flight Results of the TRMM Kalman Filter, AIAA Guidance Navigation and Control Conference, AIAA 2002-5047, Monterey, CA, USA, August 2002.

Bilanow, S., S. Slojkowski, 2006: TRMM On-orbit Performance Reassessed After Control Change, 19th International Symposium on Space Flight Dynamics, ISTS 2006-d-35, Kanazawa, Japan, June 2006.

Colton, M., and Poe, G., 1998: Inter-Sensor Calibration of DMSP SSM/I F-8 to F-14 1987-1997, IEEE trans. On Geoscience, Feb. 1998

Fixsen, D. J., A. Kogut, S. Levin, M. Limon, P. Lubin, P. Mirel, M. Seiffert, and E. Wollack, 2004: The Temperature of the Cosmic Microwave Background at 10 GHz, *The Astrophysical Journal*, 612, 86–95.

Hollinger, J., 1998: "DMSP Special Sensor Microwave / Imager Calibration /Validation", Naval Research Laboratory, Final Report, Vol. I, 20 July 1989 WGS84: : DMA TR 8350.2-A, Defense Napping Agency (DMA) Technical

Hollinger, J. P., J. L. Pierce, G. A. Poe. 1990. SSM/I instrument evaluation. IEEE Transactions on Geoscience and Remote Sensing 28(5):781-790

Kummerow,, C., W. Barnes, T. Kozu, J. Shiue, and J. Simpson, 1998: The Tropical Rainfall Measuring Mission (TRMM) sensor package. J. Atmos. Oceanic Technol., 15, 809–816.

Patt, F., Gregg, W., 1994: "Exact closed- form geolocation algorithm for Earth Survey Sensors", Int. J. Remote Sensing, Vol. 15, No. 18, 3719-3734

Robertson, B., S. Placanica, W. Morgenstern, J. Hashmall, J. Glickman, G. Natanson, 1999: On-orbit Performance of the TRMM Mission Mode, 14th International Symposium on Space Flight Dynamics, Foz do Iguaçu, Brazil, February 1999.

Simpson, J., C. Kummerow, W.-K. Tao, and R. F. Adler, 1996: On the Tropical Rainfall Measuring Mission (TRMM). Meteor. Atmos. Phys., 60, 19–36.

Wentz,F.J., P.D.Ashcroft, and C.L.Gentemann, 2001: Post-launch calibration of the TMI microwave radiometer, IEEE trans. On Geoscience, 415-422

6. ACRONYMS

ACS	Attitude Control System
ECS	EOSDIS Core System
EOS	Earth Observing System
EOSDIS	Earth Observing System Data and Information System
GCI	Geocentric Coordinates Inertial
GHA	Greenwich Hour Angle
GHz	GigaHertz
GSFC	Goddard Space Flight Center (Greenbelt, MD)
HK	Housekeeping
HDF	Hierarchical Data Format
IAD	Instrument Analysis Data
IAR	Instrument Analysis Report
IDL	Interactive Data Language
IEEE	International Association of Electrical and Electronic Engineering
IFOV	Instantaneous Field of View
L1A	Level 1A
L1B	Level 1B
MOC	Mission Operations Center
MUL	Missing Unit List
NASA	National Aeronautics and Space Administration
NASDA	National Space Development Agency (of Japan)
NOAA	National Oceanic Atmospheric Administration
PDL	Program Design Language
PR	Precipitation Radar
QA	Quality Assurance
QL	Quick Look
SeaWiFS	Sea Viewing Wide Field of View Sensor
SRS	Software Requirements Specification
SSM/I	Special Sensor Microwave/ Imager
TMI	TRMM Microwave Imager
TRMM	Tropical Rainfall Measuring Mission
TSDIS	TRMM Science Data and Information System
UTC	Universal Time Coordinates
UTCF	Universal Time Correlation Factor
VIRS	Visible and Infrared Sensor