GRAV-D General Airborne Gravity Data User Manual

GRAV-D Airborne Data Release User Manual v2.1

Applies to Data Releases starting in 09/2017

Edited by Dr. Theresa M. Damiani, Monica Youngman, and Jeffery Johnson

Introduction to Data User Manuals

This manual describes details of the Gravity for the Redefinition of the American Vertical Datum (GRAV-D) project's data naming schemes and distribution, nominal airborne field operations, data post-processing software specifics, data formats, and how to calculate other commonly used gravity values from the released data.

For details specific to a block of data, see the "GRAV-D Airborne Gravity Data Release User Manual" version that is most current for that block of data.

GRAV-D uses some specialized terminology and acronyms (e.g. "block" for a geographic area with enough flown data and tie lines to provide accurate error statistics, and "survey" for an occupation by the field team at a particular airport with a particular aircraft and instrument suite). For a full list of terminology, refer to the Glossary in the Appendices of this manual.

Contents

Figure List

Table List

1. General GRAV-D Project Description

NOAA's National Geodetic Survey (NGS) has initiated an ambitious program to re-define the vertical datum of the United States and its holdings by the year 2022. The project is titled "Gravity for the Redefinition of the American Vertical Datum," or GRAV-D.

The new gravity-based vertical datum will be accurate at the 1 to 2-cm level (where possible) throughout most of the United States. The current vertical datum, NAVD 88, is known to be biased by approximately 50 cm, with a 1 m tilt across the conterminous United States. Orthometric heights—unlike ellipsoidal heights—may be used to accurately predict water-flow patterns, land slide/slump risk, and other factors affected by Earth's gravity field. Currently, the Global Positioning System (GPS) provides only ellipsoidal heights, but a new GRAV-D-derived datum will facilitate rapid access to orthometric heights. The GRAV-D Project will provide $4.8 billion in socio-economic benefits to the United States including costs avoided by improved floodplain mapping and management.

GRAV-D consists of two major goals:

- **First: A high-resolution "snapshot" of gravity in the United States.**
  The entire United States and its holdings will be flown in the following order of priority: Puerto Rico and the Virgin Islands, Alaska, the Gulf Coast, the Great Lakes, the East and West Coasts of the continental United States, Hawaii, the American Pacific island holdings, and the interior of the continental United States.
- **Second: A low-resolution "movie" of gravity changes.**
  Primarily a terrestrial campaign, the second phase will involve periodic visits to Absolute Gravity sites to monitor changes in gravity over time. This phase will allow time-dependent geoid modeling—and thus time-dependent orthometric height monitoring—through Global Navigation Satellite System (GNSS) technology. A steering committee comprised of members of the scientific community will advise NGS on this project phase.

For more information and project materials, visit NGS on the Web or contact us by email:
GRAV-D Homepage: http://www.ngs.noaa.gov/GRAV-D
Project Manager, Monica Youngman: Monica.Youngman@noaa.gov

1.1 Data Distribution (How to find what you're looking for)

1.1.1 Survey & Block Name Scheme

GRAV-D uses some specialized terminology to differentiate between geographic data areas and field team occupations for data collection. The term "block" refers to a pre-defined geographic area with enough planned data lines and cross lines to provide accurate error statistics. The term "survey" refers to an occupation of the field team at a particular airport with a particular aircraft and instrument suite. This naming scheme allows a block of data to be completed by multiple surveys spanning several years, or to be completed as only part of the work done during a large survey. Since their purposes are different, block and survey names are generated uniquely.

Block names arise from the geographic area of the U.S. in which the data were primarily collected, according to time zone. The time zones are then split into North and South sections by the 40° N latitude line, allowing for further distinction of the blocks (Figure 1). In Alaska, the 63° N latitude line serves the same purpose (Figure 2). From there, blocks are numbered chronologically by when they were planned (not by when data collection is started or finished within them). For instance, the first block planned in the Central time zone, north of the 40° N latitude line would be CN01. The next planned in the same area would be CN02. CN02 could be completed either before or after CN01. Similarly, the first block planned in the Pacific time zone, south of the 40° N latitude line would be PS01; And the first block planned in the Alaska time zone, south of the 63° N latitude line would be AS01.

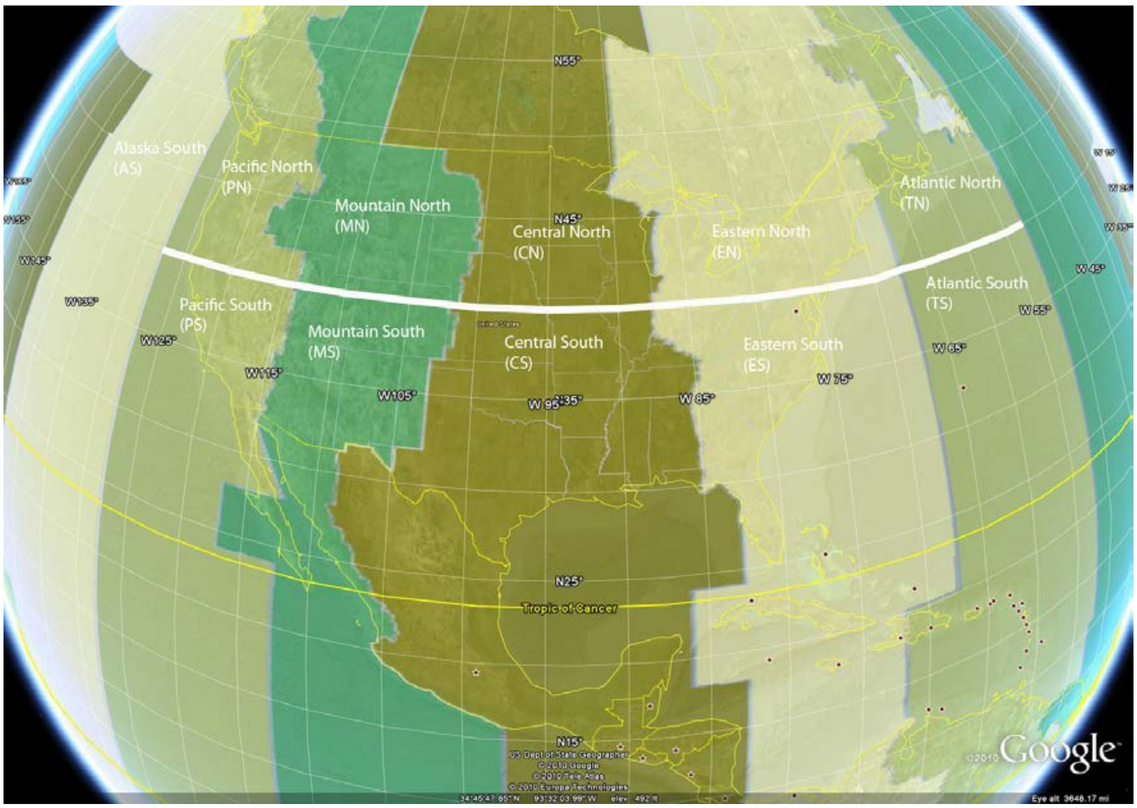

Figure 1: Time Zones of Continental United States overlaid with the block name for each North or South section of the time zone (delimited by the 40° N latitude, thick white line).

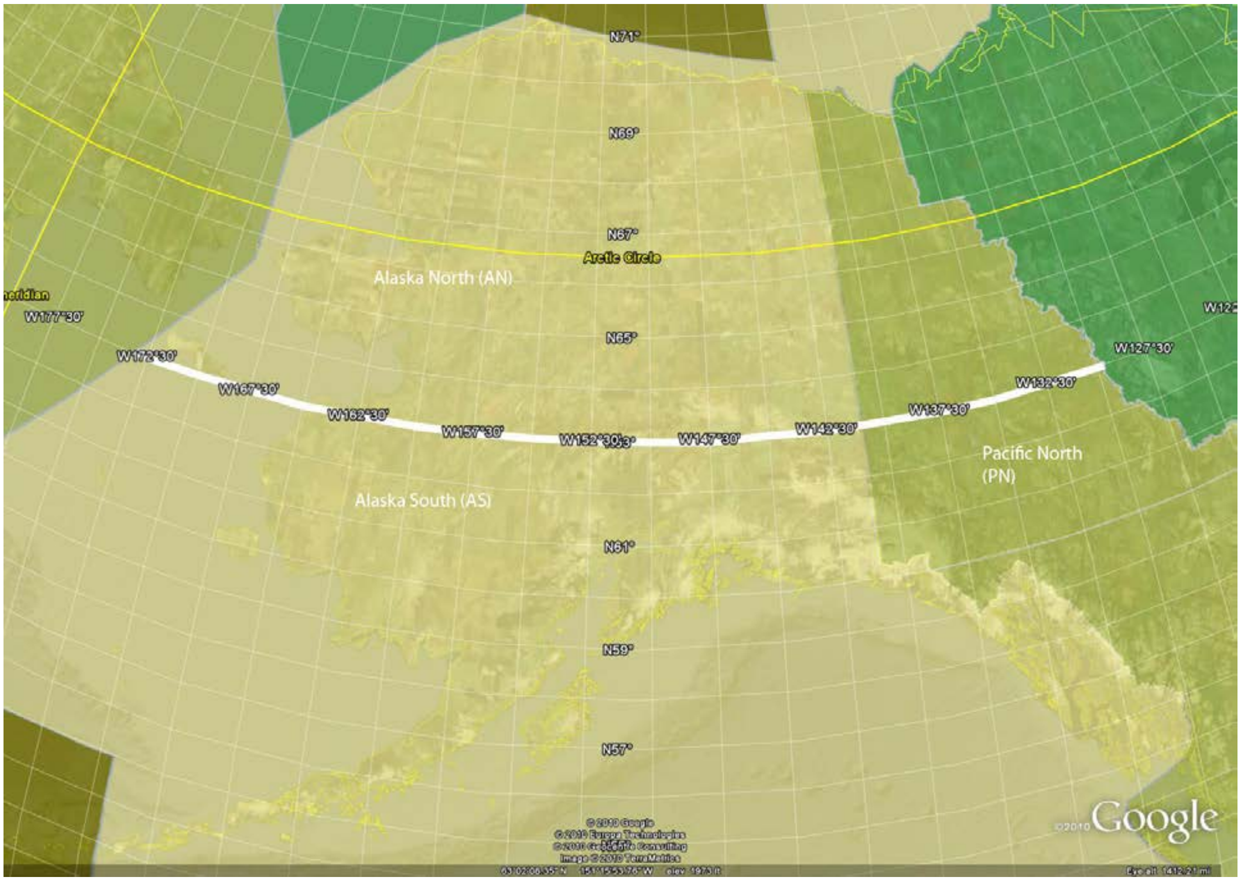

Figure 2: Time Zones of Alaska overlaid with the block name for each North or South section of the time zone (delimited by the 63° N latitude, thick white line).

Surveys are named for the 2 letter state abbreviation where the airport of operations is located and the year in which the survey was started. For example, the first time that the field team takes a particular plane and equipment suite to Buffalo, New York in 2011, the survey will be named NY11-1. If the plane or equipment changes while at Buffalo, a new survey is started and it would be called NY11-2. If the plane and equipment are then transferred to Anchorage, Alaska for a survey and then transferred again to Miami, Florida later in the 2011 season, the surveys for each would be called AK11-1 and FL11-1, respectively.

1.1.2 Block Line Naming Conventions

Within each block, data lines and cross lines are named according to the following rules:

100s- Primary data lines (closely spaced)
200s- First reflight of a data line
300s- Second reflight of a data line
400s- Third reflight of a data line
500s- Primary cross lines (widely spaced)
600s- First reflight of a cross line
700s- Second reflight of a cross line
800s- Third reflight of a cross line
900s- Lines of opportunity (specially-planned e.g. cross-grid or different altitude; or transits)

Examples of line naming are:

- A primary data line: 115
- First reflight of that line: 215
- Second reflight of that line: 315
- Third reflight of that line: 415
- A primary cross line: 507
- First reflight of that line: 607
- Second reflight of that line: 707
- Third reflight of that line: 807
- Third transit collected in a survey with no pre-planned special lines: 903

During a survey, full line names are used. These names designate which data block and line is being flown. For example, if the NY11-1 survey is meant to accomplish two blocks (EN01 and EN02) during their occupation, the log books and data files will show the line 103 in each block as EN01103 and EN02103.

1.1.3 Where to find data

GRAV-D airborne data products and metadata are available on the data distribution website: https://www.ngs.noaa.gov/grav-d/data_products.shtml. This Google Maps data portal will always display the most up-to-date information about GRAV-D blocks that are planned, being executed, being processed, and publicly released. When data are available for public release, clicking on that block will produce a pop-up link to that block's data site.

1.1.4 What data are available

On each block site, a new Google Maps data portal displays the extent of the block and block lines that are available for download. The block lines are clickable for information about when they were collected and during which survey. All lines must be downloaded together in a .zip file. In the download section of that webpage, the .zip file will contain the most up-to-date versions of data, metadata, and block documentation. Usually, the .zip file will contain: a ReadMe file, a gravity data file of the official products, a supplementary data file (containing cross line data, transit line data, or other experimental data), a .txt file containing .xml metadata written to Federal Geographic Data Committee (FGDC) standards, a .kml file of the block extent, and a .kml file of the line locations for that block. The gravity data file format is found in Table 6: Publicly released data, By Block (a compilation of all the flight files): NGS_GRAVD_Block_BB##_Gravity_Data_v#.dat. Also on the block webpage, there will be a link to the FTP site where older data release versions are stored.

Finally, the webpage for each block will provide specific citation information that should be used when GRAV-D data, methods, or information are incorporated into research, presentations, or papers. The same citation information can also be found in the block documentation.

1.1.5 Point of Contact for more info

On every GRAV-D block data webpage, a point of contact (POC) will be provided for technical questions about the data, methods, and project. Usually, this POC will be one of the GRAV-D staff scientists who was involved in the data collection and/or data processing. Please direct all question about the data to this person, but direct any website concerns or download problems to the NGS webmaster: ngs.webmaster@noaa.gov.

1.2 Field Operation Details

1.2.1 Block Layout and Survey Flight Execution

GRAV-D airborne gravimetry data blocks are generally planned to meet certain standard requirements. However, every GRAV-D block and survey is unique and may depart from these requirements in some way. Refer to the block-specific documentation that accompanies each data set for details on the surveys that were conducted inside that data block. Blocks are laid out by taking into account the:

- trends of the gravity field in the collection area;
- locations of nearby airports;
- practical altitude for aircraft and adequate terrain clearance;
- practical line lengths for the duration of the aircraft and gravity signal capture;
- adequate data sampling along line and between lines;
- minimum cross line spacing needed for error statistics; and
- adequate overlap with previous surveys or extension into neighboring nations.

Table 1: Nominal Block and Survey Characteristics

Characteristic	**Nominal Value**
Altitude	20,000 ft (~6.3 km)
Ground speed	250 knots (250 nautical miles/hr)
Along-track gravimeter sampling	1 Hz = 128.6 m (TAGS) 20 Hz = 6.43 m (TAGS7) (at nominal ground speed)
Data Line Spacing	10 km
Data Line Length	~400 km
Cross Line Spacing	40-80 km
Cross Line Length	~500 km
Data Minimum Resolution	~20 km (dependent on altitude and line spacing)

The nominal block characteristics listed in Table 1 represent the ideal layout for a block to be flown with a medium duration, double-engine aircraft like a King Air 200. A real example of this kind of survey layout is shown in Figure 3. The nominal altitude represents the best tradeoff between recovery of the mid-wavelengths of the gravity field, suppression of noise for downward continuation in geoid models, and logistical concerns such as density of commercial air traffic and length of time to complete long lines at this altitude. In a nominal survey such as described in Table 1, a 500 km line is flown in just over 60 minutes.

A main limitation of the resolution of high-altitude gravimetry is the flight altitude. Childers, et al. (1999, Equation 8), approximately defines the “geologic wavelength” (λ_g) of the minimum recoverable feature as:

$$\lambda_g = 2w_{1/2} = 1.54z_c \quad (1)$$

where $w_{1/2}$ is the half-width of the anomaly at the half-amplitude point and z_c is the height of the measurement point above the center of the spherical body causing the anomaly. For the general case of airborne surveying above flat terrain, the height is defined as the flight altitude. The Fourier wavelength of the smallest recoverable feature is defined in relation to the "geologic wavelength" and flight altitude:

$$\lambda_f = 2\lambda_g = 4w_{1/2} = 3.08z_c \quad (2)$$

For GRAV-D surveys, our nominal altitude of 20,000 ft (6096 m), yields a minimum recoverable geologic wavelength of 9.4 km and Fourier wavelength of 18.8 km. Accordingly, no matter the line spacing flown at this altitude, no feature smaller than almost 20 km wide will be resolved for an area with flat terrain. If there is significant terrain, then the Fourier wavelength will be smaller and result in better small feature recovery.

Based on Nyquist theory, the minimum resolution (or the expected minimum size of a well-sampled feature) for the block is 20 km, as a result of the 10 km data line spacing. Along-track resolution is a function of aircraft speed, altitude and the amount of low-pass filtering required, and is usually better than 20 km. It is possible that some data would require heavy filtering, which would become the limiting factor on feature resolution. However, this is unlikely because quality control processing in the field is in place to ensure that noisy data are flagged and the line reflown. Each block manual will contain the survey plans and data resolution for that area.

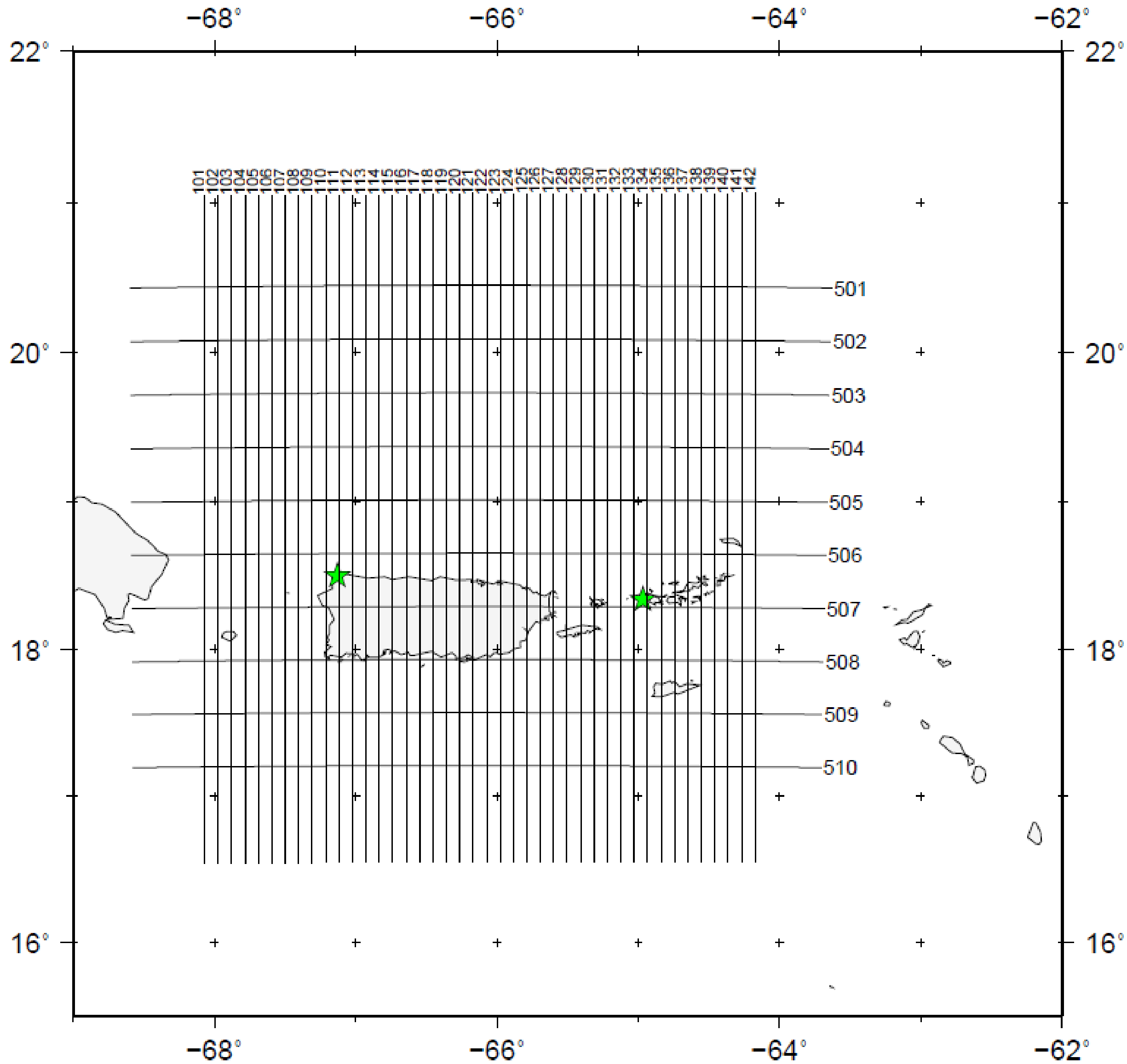

Figure 3: Layout used for the TS01 data block (Atlantic time zone, south #01). The block was completed during the PV09 survey. Data line spacing 10 km; cross line spacing 40 km.

GRAV-D surveys may purposefully refly good data lines a second time. This practice is critical for calibrating the final filter used in post-processing the data set, given that the noise properties of each survey are different. Since the location of the lines is the same, the gravity field is also the same for both lines. The best filter is found by applying various filters to the reflown data until a correlation of at least 99% and an RMS nominally below 1 mGal is attained between the two recordings. A close match between the reflown lines provides a guide with which to filter the rest of the data and cross lines.

1.2.2 Aircraft

GRAV-D uses a variety of aircraft, depending on their availability, location of the survey, and capabilities (Table 2). When GRAV-D is using a shorter-duration aircraft (~4-5 hours per flight), two flights per day are accomplished as weather permits. When using longer duration aircraft (6 hrs or greater), only one flight per day is logistically possible. Flights are normally done during the daytime, while airfields are regularly open for business. Weather forecasts are consulted the day of the flights to find the smoothest, least turbulent, air in the survey area. Aviation weather forecasts are available online (http://aviationweather.gov/).

Table 2: Aircraft Used by GRAV-D as of 09/2017

Aircraft Operator	Aircraft Type	Engine Type	Number of Engines	Approximate Maximum Duration (hrs)
NOAA	Cessna Citation II CE-550	Jet	Double	4.5
Naval Research Lab	Hawker Beechcraft King Air RC-12 (Non-military: 200)	Turboprop	Double	4
NOAA	Turbo Commander 1000 695A	Turboprop	Double	4
Bureau of Land Management	Pilatus PC-12	Turboprop	Single	5
NOAA	Lockheed P-3 Orion	Turboprop	Four	9.5
Fugro Earth Data/ Airborne Services	Hawker Beechcraft King Air E90A	Turboprop	Double	5.5
Fugro Earth Data/ Airborne Servives	Cessna 441 Conquest II	Turboprop	Double	6.5
Aurora	Centaur	Turboprop	Double	4
Dynamic Aviation	King Air 200T	Turboprop	Double	4
NOAA	Gulfstream IV	Jet	Double	8.5

1.2.3 Airborne Instrumentation and Data Products

GRAV-D uses a Micro-g LaCoste TAGS (Turn-key Airborne Gravimetry System), which is a beam-type, zero-length spring gravity sensor on a gyro-stabilized platform (Micro-g LaCoste, 2010). The TAGS is a modern gravimeter that measures to 0.01 mGal precision in the laboratory, has horizontal accelerometer output recorded to the data files, is controlled by a computerized user-friendly system, and provides digital data files. The TAGS unit is started each morning and allowed to settle to take a still reading screenshot. The TAGS is run continuously throughout the day until the last flight is completed and an end-of-day still reading is taken. If two flights are accomplished, a third still reading is taken between the flights to more closely monitor for gravimeter drift or tares.

The TAGS (depending on the unit serial number) comes with either a NovAtel DL4 Plus or NovAtel DLV3 GPS unit integrated into its timing unit module. The timing unit provides real-time location information to the TAGS, as well as accurate timing via the GPS and rubidium clock inside the module. Nearly all GRAV-D flights have a second GPS on the aircraft that is integrated with an inertial measurement unit (IMU) into a positioning system. Post-mission, this positioning system provides attitude information about the aircraft (pitch, roll, yaw, and true heading) in addition to the GPS positions. In case GPS+IMU data is not available for a flight, a spare TAGS timing unit is installed in the aircraft to provide GPS redundancy. A list of all airborne instrumentation available to GRAV-D is in Table 3.

Table 3: NGS Owned Airborne Instrumentation and Data Types

Type of Instrument	Manufacturer and Model	Data Files
Relative Gravity	Micro-g LaCoste (MGL) TAGS S-137, S-161, or S-211	.dat, .env, .txt .bmp
GPS-only	NovAtel DL-4 Plus or NovAtel DLV3	.pdc
GPS+IMU Positioning	Applanix POS AV 510 (stopped using in September 2011)	DEFAULT.###
GPS+IMU Positioning	NovAtel SPAN (began using in January 2011)	SPAN.log

1.2.4 Ground Instrumentation and Data Products

One or two GPS base stations are set up at temporary points around the airport of operations, usually near the parking space of the aircraft. These base stations allow for GPS processing in differential mode for QC in the field and also are backups, in case other GPS processing options (i.e. Precise Point Positioning, PPP) do not meet the necessary accuracy specifications. The GPS base stations are put in locations with the best view of sky and fewest possibilities for multipath. Usually, GRAV-D uses GPS units that have a separate antenna and receiver, though- on rare occasion- integrated receiver/antenna units have been used. The antennas are always set up on 2 m fixed-height poles, attached to a receiver in a rugged case and running off of battery power for the whole day. A few types of GPS base stations have been used by GRAV-D, including: Ashtech Z Surveyor, NovAtel DL4 Plus, and Trimble R8 (Table 4).

A gravity tie is necessary for any relative gravity instrument, such as the TAGS, in order to convert the meter's counter units (an arbitrary measurement unit) to m/s^2 or mGal. a field-tested absolute gravity meter (an MGL A-10) takes measurements on the exact spot on the airport tarmac where the TAGS will be parked in the plane, but at a lower height. A gradient is measured at the spot with the hand-carried relative meters and is used to calculate gravity at the height of the TAGS' sensor center.

Table 4: NGS Owned Ground Instrumentation and Data Types

Type of Instrument	Manufacturer and Model	Data Files
GPS-only	Trimble R8	.DAT, .T01
GPS-only	Ashtech Z-surveyor	B,E, &S files
GPS-only	NovAtel DL-4 Plus	.pdc
Absolute Gravity Lab Instrument	Micro-g LaCoste FG-5 102	N/A
Absolute Gravity Field Instrument	Micro-g LaCoste A-10	N/A
Relative Gravity	LaCoste and Romberg G and D meters (NGS owns 5), Sintrex CG-6	N/A

1.2.5 Quality Control Data Processing

During the survey, it is critical that data be processed not long after flying to ensure that all instruments are working and data quality is acceptable. Field personnel perform data checks at least once per day to ensure all GPS receivers, the gravimeter, and IMUs are working up to standard. The field personnel also process GPS and gravity data, mapping the preliminary gravity product, to check quality. Additional quality control plots are built into the GPS and gravity processing software packages for advanced data analysis in the field.

2. Post-Processing and Software Specifics

2.1 GPS Processing

2.1.1 GPS Processing Overview

Kinematic GPS processing is done for every GRAV-D flight, either differential or PPP. If a GPS+IMU (inertial measurement unit) instrument was working on the aircraft, a combined solution is obtained. If IMU data are not available, a GPS-only position file is produced.

2.1.2 GPS Processing Specifics

The preferred GPS processing method for GRAV-D data sets is precise point positioning (PPP). Occasionally, if there is a problem that results in a poor PPP solution, the data is processed as a kinematic differential solution using the base station and the remote data collected during the flight.

The choice of rover GPS is straightforward, since coupled GPS + IMU solutions are the preferred product. Since September 2011, GRAV-D retired their use of the Applanix POS AV system and moved entirely to operations with the SPAN system.

No matter the system used, GPS solutions are created for a variety of satellite elevation masks, usually between 5 and 10 degrees. The qualities of these solutions are compared by checking six plots within the processing software: combined separation (i.e. the difference between the forward and reverse solutions); position dilution of precision (PDOP); number of satellites shared between rover and base; estimated position accuracy; carrier phase residual RMS and standard deviation; and a software-estimated quality factor. Based on these plots, a final GPS+IMU position solution is chosen.

NGS has created a "quality grade" (QG) measure for the GPS-only solutions coming from Inertial Explorer. The combined separation (CS), and estimated position accuracy (EPA) information are exported from the GPS-processing software and fed into a Matlab code to assign a quantitative QG to the flight's position solution. The QG is calculated as:

$$QG = [(0.6 * CSgrade) + (0.4 * EPAgrade)] \tag{3}$$

where *QG* is the flight's quality grade, *CSgrade* is the combined separation grade, and *EPAgrade* is the estimated position accuracy grade. The criteria-specific grades are calculated as follows:

$$CSgrade = (0.1 * [\% \; of \; CS < 0.4 \; meters]) + (0.15 * [\% \; of \; CS < 0.3 \; meters]) + (0.35 * [\% \; of \; CS < 0.2 \; meters]) + (0.4 * [\% \; of \; CS < 0.1 \; meters]) \tag{4}$$

$$EPAgrade = (0.1 * [\% \; of \; EPA < 0.4 \; meters]) + (0.15 * [\% \; of \; EPA < 0.3 \; meters]) + (0.35 * [\% \; of \; EPA < 0.2 \; meters]) + (0.4 * [\% \; of \; EPA < 0.1 \; meters]) \tag{5}$$

weights in the QG calculation (including criteria-specific grades) were optimized to produce a realistic range of final grades and to more-heavily weight the best solutions. The user-noted, qualitative range of GPS position qualities for the 2008-2010 GRAV-D surveys are reflected in the quantitative range of QGs produced by the program. The QG is calculated as a guide so that users understand that quality of GPS products used in the gravity product calculations.

In previous GRAV-D user manuals, the QG formula included a ratio based on the number of epochs that had either a float or a fixed ambiguity status. We have left this ratio out of the formula because our GPS processing software no longer provides a variable printout that resolves the ambiguity status sufficiently. Instead, the GPS processing personnel verify that the ambiguity status is either a fixed integer or stable float for the entirety of the flight by checking a graphical output provided by the processing software.

Another way GRAV-D communicates GPS solution quality is by including a horizontal and a vertical position accuracy for the composite of all the data lines included in the block. The vertical position accuracy is simply a 1.96 scale on the vertical standard deviation to achieve a 95% confidence level. The horizontal position accuracy is an estimate of the Circular Error Probable (CEP) at the 95% confidence interval (equation 6). CEP_{95} is calculated using the standard deviation of the northing, σ_N, and the standard deviation of the easting, σ_E .

$$CEP_{95} = 2.4477 \frac{\sigma_N + \sigma_E}{2} \tag{6}$$

Since the manual provides a single horizontal and vertical position accuracy for a whole block, the northing, easting, and height standard deviations from each epoch during gravity data collection are averaged before scaling up to the 95% confidence level. An additional scale factor is then applied to make the final estimated horizontal and vertical position accuracy values more realistic based on results of the GPS Challenge (Damiani, et al. 2013). In short, we expect the horizontal position accuracy to be close to 1 dm, and the vertical position accuracy to be close to 2 dm.

Finally, for every GRAV-D aircraft installation, lever arms are measured by the field team. A lever arm is the distance along the body of the aircraft between instruments on board and the aircraft center of gravity. Measuring the distances relative to the center of gravity for the aircraft is important because the center of gravity is the “origin” of the aircraft body frame. For instance, the GPS antenna for the aircraft is always mounted on the outside of the aircraft fuselage, but can be positioned anywhere between the fore and aft of the aircraft. So, GPS solutions yield the location of the GPS antenna, but not that of the gravimeter sensor unless the distance between the two is accounted for in vertical, along the fuselage, and along the wings (i.e. in the body frame of the aircraft). This is an important and non-trivial correction during flight because the aircraft is often oriented so that it is not pointed in the direction of flight. Also, if the GPS antenna is on one side of the aircraft’s center of gravity and the gravimeter is on the other, their senses of motion will be opposite. E.g. A GPS fore of the center of gravity will move up when the plane pitches up, but a gravimeter aft of the center of gravity will move down for the same maneuver. Using the lever arm while taking IMU aircraft orientation data into account will yield the best estimate of the gravimeter’s position while in the air. However, when IMU orientation data are not available, only a vertical rigid distance correction is applied during the gravity processing, which is likely insufficient to account for orientation errors in the positioning.

2.2 Gravity Processing

2.2.1 Gravity Processing Overview

Newton, the NGS-developed airborne gravity processing software, is used to compute full field gravity at flight altitude from raw gravimeter data and GPS solutions. In the processing, first the raw gravimeter reading is re-computed as described in the TAGS manual. Next, the timing of the gravity and GPS data are synchronized to within 0.01 seconds by adjusting the gravity’s timing. Corrections for aircraft motion (Eötvös and vertical acceleration corrections), meter off-level, and drift are calculated and applied with an absolute gravity tie. The results are then filtered with a simple time-domain Gaussian filter applied three times. Tracks are trimmed to remove any remaining effects of the turns. These data are then checked for internal consistency by examining crossing points and the correlation of adjacent lines. Finally, the processed gravity profile is compared to both satellite gravity and geoid models.

2.2.2 Gravity Processing Theory

The TAGS gravimeter is a highly-damped, beam-type gravity sensor mounted on a gyro-stabilized platform. The raw gravity measurement can be computed:

$$g_{raw} = CalScale * (ST + k * \dot{B} + CC) \quad (7)$$

where *CalScale* is the calibration scale converting counter units to mGal, *ST* is spring tension, *k* is the scaling factor for converting beam velocity to counter units, $\dot{B}$ is the beam velocity and *CC* is the cross coupling correction (Micro-g LaCoste, 2010). For more detail on calculating the raw gravity, see the TAGS Hardware Manual version 2.0, 13 April 2010 pages 1-2 to 1-6.

The raw gravity is a measure of specific force in the direction of the local vertical averaged over four minutes by the stabilized platform. The raw gravimeter output must be differenced with the aircraft's acceleration as measured by GPS to yield an uncorrected gravity measurement. The raw gravity is a relative measurement and must be tied to an absolute measurement of gravity. Instrument drift and any instantaneous off-level must also be corrected. So, g can be calculated as follows:

$$g = g_{raw} - a_{vertical} + offlevel - drift + tie \tag{8}$$

The vertical acceleration due to the motion of the aircraft in a rotating reference system, $a_{vertical}$, can be computed from the GPS-derived positions throughout the flight by taking the vertical component of the following equation of motion,

$$\vec{a} = \frac{d^2\vec{r}}{dt^2} + 2\vec{\omega} \times \frac{d\vec{r}}{dt} + \frac{d\vec{\omega}}{dt} \times \vec{r} + \vec{\omega} \times \vec{\omega} \times \vec{r} \tag{9}$$

Where *t* is time, *r* is the distance from the point of interest to the center of the Earth and *ω* is the rotation rate of Earth. Following Harlan (1968), the vertical component of terms two and four together comprise the Eötvös correction. Term three is the acceleration of the coordinate system which is zero if we assume the rotation rate of the Earth is constant. The first term is the acceleration of the aircraft within the coordinate system.

Off-level errors, where the meter sensitive axis is slightly misaligned with the instantaneous vertical vector, are a problem for two reasons. The first is that only a portion of the gravitational acceleration, which by definition is along the local vertical vector, is measured. The second and more challenging is that a component of the aircraft's horizontal acceleration is mistakenly included in the raw gravity. To correct this we apply a variation of the direct method from Peters & Brozena (1995).

$$oe = \sqrt{g^2 + (xacc_{meter}^2 + lacc_{meter}^2) - (xacc_{gps}^2 + lacc_{gps}^2)} - g - \widetilde{oe} \tag{10}$$

The offlevel error is *oe,* and $\widetilde{oe}$ is the median of the offlevel error. The nominal value of gravity is *g* and the cross (*xacc*) and along track (*lacc*) accelerations from the meter accelerometers and derived from GPS complete this equation.

Meter drift is found by linearly interpolating between the pre-flight and post-flight still readings.

The absolute gravity tie is measured for each survey as described in the Field Operations section 1.2.4 Ground Instrumentation and Data Products of this manual.

2.2.3 Gravity Processing Software

Newton is a Matlab-based post processing packaging for airborne gravimetry. It was developed specifically to handle a high-altitude, high-speed survey configuration while delivering the precision and accuracy required by NGS's GRAV-D project. The software has been under development since 2008 and continues to be refined. Several operational versions have been used as of 2017. There are three basics steps to processing with Newton. First the unfiltered gravity is computed following the basic theory presented above. Next the best length for a time domain Gaussian filter is found by comparing data from repeat tracks. That filter is then applied three times and data are edited to remove periods that are unrecoverable due to large aircraft motion or maneuvers. Finally, data quality is assessed by comparing cross-over points, correlation of adjacent tracks, and comparison with satellite data and geoid models. It is important to note that although GRAV-D data presents the crossover statistics and comparisons with EGM08 in each data manual, no leveling (also called crossover adjustments) or bias corrections of any kind are applied to the airborne data that is released. The released data are simply the full-field values as calculated above, with no other post-processing adjustment. If the user would like to apply any adjustments or corrections of their own to the data, that is at their discretion.

2.2.4 Processed Gravity Resolution and Step Function Appearance with 20 Hz data

Theoretically, full field gravity data measured along a GRAV-D flight line are continuous everywhere and when modeled will graph as a "smooth" line. However, we have noticed a discontinuous, "step function" phenomenon that we believe is related to the limitations of the TAGS meter to completely resolve the gravity signal. TAGS measures to approximately 1 mGal accuracy, but we give full field gravity values to 0.01 mGal. Due to rounding, it is possible for several adjacent gravity values to be the same, causing a "step function" appearance.

This "step function" appearance is more noticable with our new TAGS7 unit which has a 20 Hz collection rate (the older units are only 1 Hz). This high frequency collection rate combined with a slower moving aircraft, such as the Aurora pilot optional Centaur, and a region with very little change in the gravity field can result in significant stepping in the data. See Figure 4 for an example of what this looks like. If continuous data is necessary, we suggest sampling this data set at a certain interval (i.e. every 20th index value) or interpolating a polynomial.

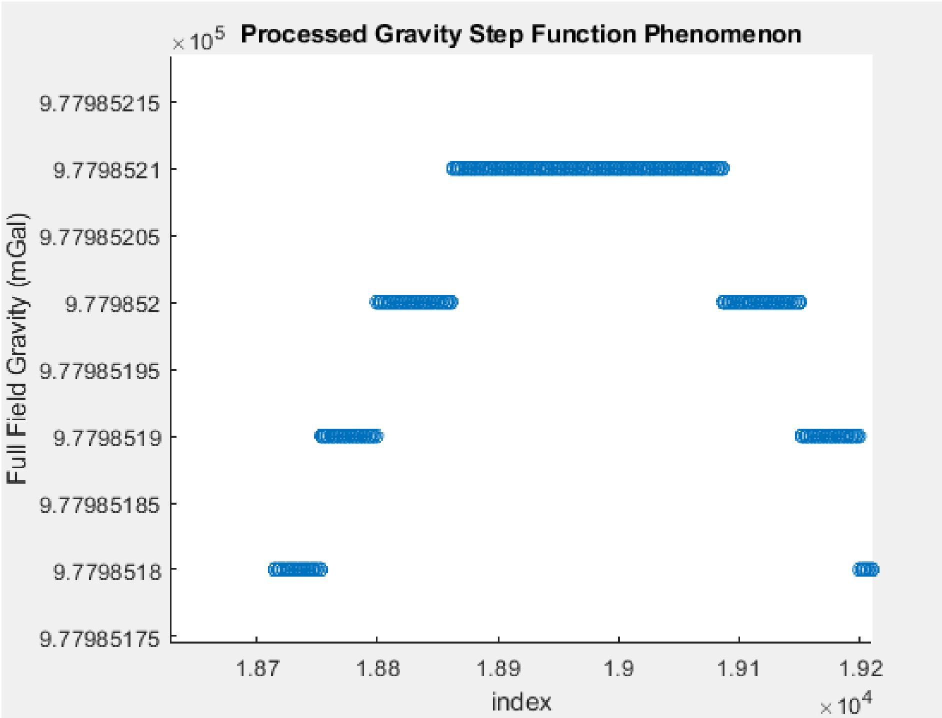

Figure 4: Discontinuous Step Function Phenomenon with 20 Hz Gravity Data.

3. Calculating Other Commonly-Used Gravity Values

Two scalar gravity quantities are commonly calculated from the full-field gravity values: gravity disturbance and gravity anomaly. These quantities are more useful in the interpretation of gravity signals than the full-field gravity values. In the scientific literature and other texts, the definitions of these terms can be confusing and vary from source to source. The definitions also tend to vary somewhat between the geodesy and geophysics communities; two good discussions of this are Li and Götze (2001) and Hackney and Featherstone (2003). For a traditional geodesy perspective on working with both gravity potentials and gravity, see Hofmann-Wellenhof and Moritz (2006).

The gravity disturbance is the difference in the observed gravity with the normal gravity calculated based on an ellipsoid and the gravity anomaly is the difference in the observed gravity with the normal gravity calculated based on a geoid (these are defined in detail below). In this text we discuss the various ways to calculate the gravity disturbance and gravity anomaly. The generic terms are used here because a variety of methods are discussed. Some texts and communities will refer to the free-air disturbance (FAD) and/or free-air anomaly (FAA), which refers to values calculated specifically using the free-air correction method (this is not our preferred method, see Section 3.2).

3.1 Ellipsoidal Height and Orthometric Height for Free-Air Corrections

Since the choice of reference surface is so important to the calculation of free-air gravity products, a brief discussion of their differences is necessary (Figure 5). If the datum for the free-air correction is an ellipsoid, ellipsoidal heights (h, height along a line normal to the ellipsoid) are used in the calculation. If the datum is a geoid model instead (i.e. a model of the equipotential surface that best fits mean global sea level), orthometric heights (H, height along the curved plumb line above the geoid) are used in the free-air correction calculation.

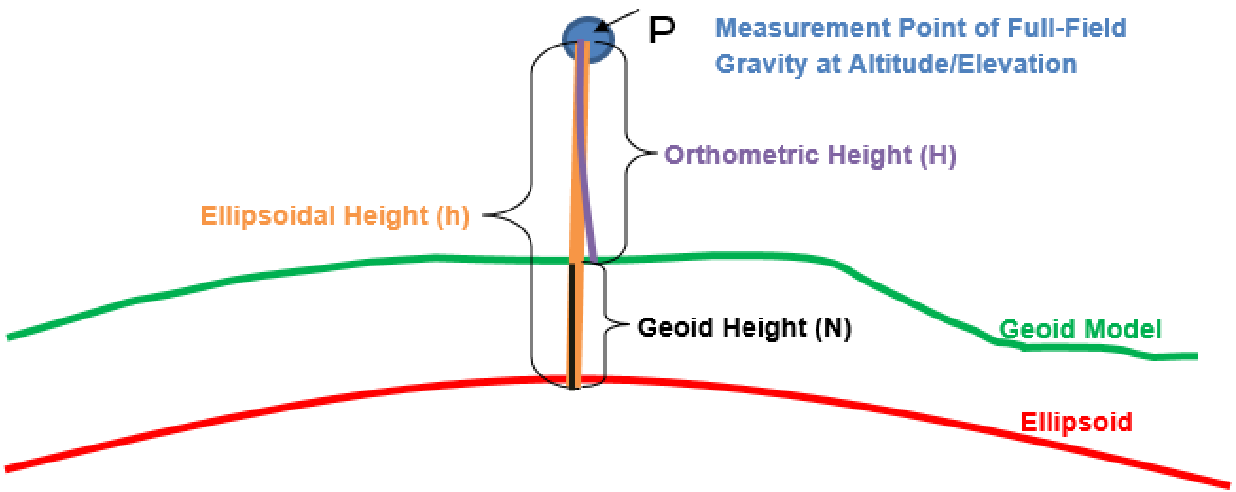

Figure 5: Sketch (not to scale) of the difference between orthometric height (height above the geoid), ellipsoidal height (height above the ellipsoid), and geoid height (height of geoid above ellipsoid).

Because of the curvature of the plumb line,

$$H \approx h - N \tag{11}$$

where H is orthometric height, h is ellipsoidal height, and N is geoid height. Although this relation is accurately stated as approximately equal, the error attributed to H when assuming that H runs along ellipsoidal normal instead of along the plumb line is usually considered negligible: "…the directions of [normal gravity along the plumb line] and [normal gravity along the ellipsoidal normal] coincide virtually." (Hofmann-Wellenhof and Moritz, 2006; just below equation 2-232). So that, for the purposes of defining gravity quantities:

$$H = h - N \tag{12}$$

3.2 Gravity Disturbance (GD or FAD)

There are numerous ways to calculate a gravity disturbance and they are listed below from most accurate to least accurate. It is important to define that any gravity value calculated based upon an ellipsoid is called a "normal gravity" value.

NOTE: Gravity data users are strongly urged to utilize the 1-step confocal ellipsoid computation for normal gravity at any height and location, which provides 1μGal-accuracy disturbances, instead of the traditionally-used, 2-step free-air and surface normal gravity corrections.

Gravity disturbance values should always be reported with a statement about which ellipsoid and calculation methods were used. Two examples would be: "Gravity disturbance values were calculated using the confocal ellipsoid method (Damiani, 2013) with respect to the GRS-80 ellipsoid (Moritz, 2000)" or "Free-air gravity disturbance values were calculated using the 2nd order free air correction (Featherstone, 1995) and Somigliana-Pizetti normal gravity equation (Hofmann-Wellenhof and Moritz, 2006) with respect to the WGS-84 ellipsoid (NIMA, 2000)." Note that gravity disturbance values are relative to an ellipsoid that is (by definition) not representative of global sea level.

Table 5: Parameter values for two common reference ellipsoids, GRS-80 (Moritz, 2000; Hofmann-Wellenhof and Moritz, 2006) and WGS-84 (NIMA, 2000; Hofmann-Wellenhof and Moritz, 2006)

Parameter	Description	GRS-80 value	WGS-84 value
γ_a	Equatorial normal gravity	9.780 326 7715 m/s^2	9.780 325 3359 m/s^2
γ_b	Polar normal gravity	9.832 186 3685 m/s^2	9.832 184 9378 m/s^2
e^2	First eccentricity squared	0.006 694 380 022 90	0.006 694 379 990 14
a	Semi-major axis	6 378 137 m	6 378 137.0 m
b	Semi-minor axis	6 356 752.3141 m	6 356 752.3142 m
f	Flattening	0.003 352 810 681 18	0.003 352 810 664 747
ω	Angular velocity of Earth	7 292 115 x10^{11} rad/s	7 292 115.0 x10^{11} rad/s
GM	Earth's gravitational constant	3 986 005 x 10^8 m^3/s^2	3 986 004.418 x 10^8 m^3/s^2

3.2.1 1-step High-Accuracy Gravity Disturbance (GD) Calculation Confocal Ellipsoids

The work presented in this section is summarized in a poster presentation from the American Geophysical Union conference (Damiani, 2013) and is available online. The results show that the most accurate gravity disturbances for all gravity data (not just airborne) are computed with the "confocal ellipsoid" method.

The tradition of calculating a free-air disturbance has been persistent because explicit equations exist for both $g_{FAC\_ellipsoid}$ and $\gamma_{normal\_ellipsoid\_surface}$, and the accuracy of the 2nd order equation is sufficient for mGal-level near-surface gravity applications (see Section 3.2.2). However, those traditional equations lose accuracy with increasing altitude and latitude. The 2nd order free-air correction produces errors of 20 to 100 µGal most latitudes and heights (Figure 5). Even less accurate, a linear "free-air" approximation (which assumes that the gravity gradient is 0.3086 mGal/m everywhere, see Section 3.2.3), produces significant errors (-0.5 to 10 mGal) for gravity data in most places and at most altitudes (Figure 6, Figure 7, and Figure 8).

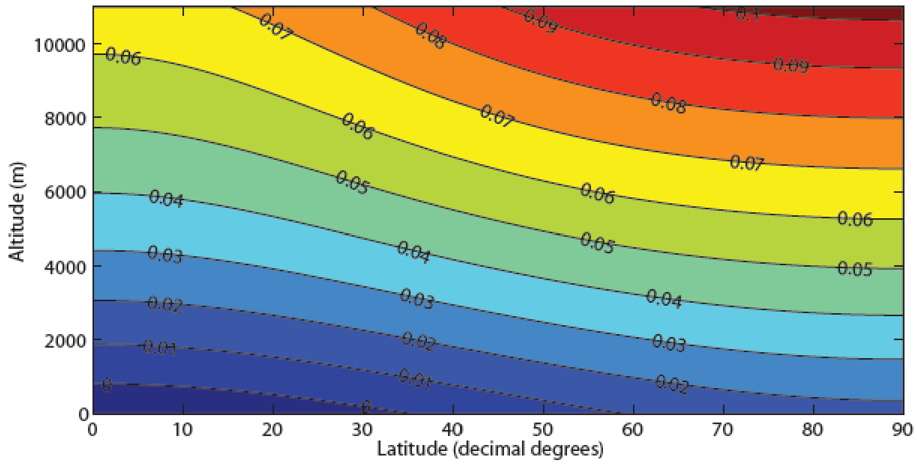

Figure 6: Difference (mGal) between the 2nd order free-air correction and a comparable correction derived from the confocal ellipsoid computation of normal gravity at any height with WGS84.

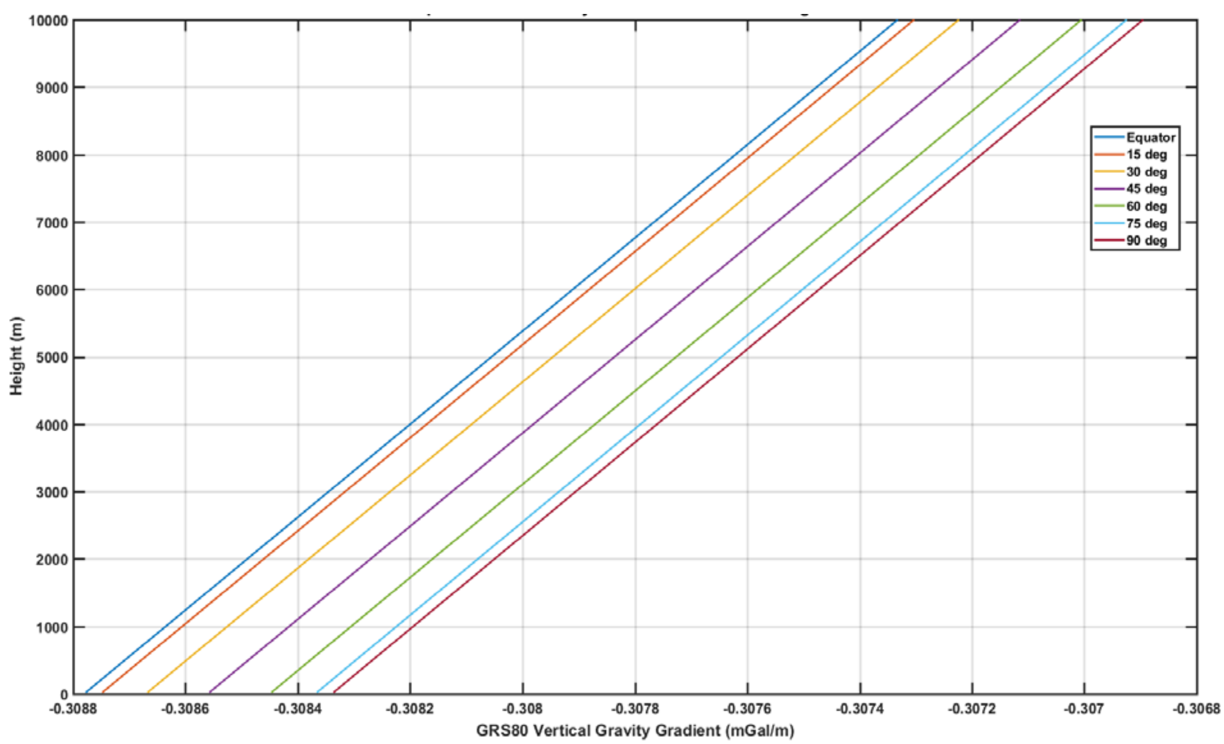

Figure 7: GRS80 ellipsoid vertical gravity gradient variation with respect to latitude and height above the ellipsoid surface.

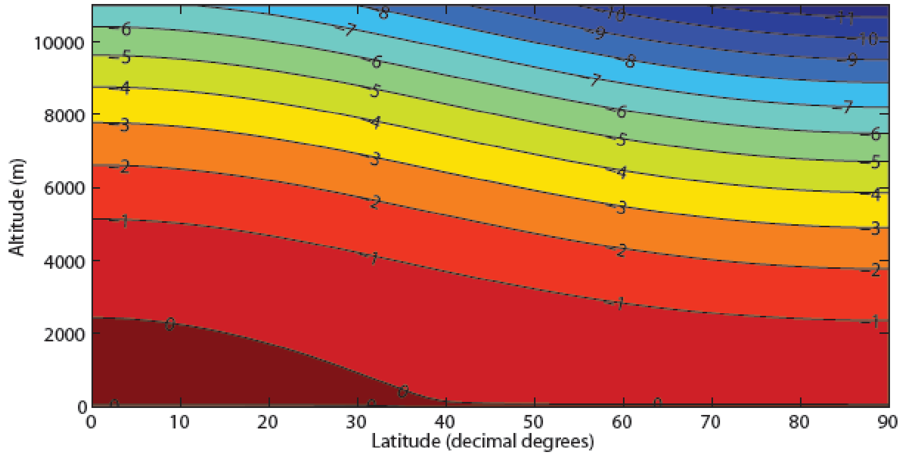

Figure 8: Difference (mGal) between the 1st order (linear) free-air correction and a comparable correction derived from the confocal ellipsoid computation of normal gravity at any height with WGS84.

However, the traditional 2-step corrections (free-air correction plus normal gravity at the surface) can be easily-replaced by a precise computation for normal gravity *on or at any height above the ellipsoid*. Accurate to 1 µGal, it is fast to compute on any standard laptop or desktop computer. The computation uses the concept of confocal ellipsoids and has been implemented in code from the National Geospatial-Intelligence Agency (NGA) for over a decade. It was publicly-released as part of the harmonic synthesis code companion to the Earth Gravity Model 2008 (EGM2008) called hsynth_WGS84.f (NGA, 2010; Pavlis, et al., 2012).

Succinctly, for any point (P) in space, an ellipsoid can be drawn through that point that has the same foci as a chosen reference ellipsoid. This “confocal” ellipsoid has an associated Somigliana-Pizetti equation that will precisely calculate normal gravity on a confocal ellipsoid’s surface. Inside NGA’s Fortran routines of the hsynth_WGS84.f program (NGA, 2010; Pavlis, et al., 2012), is a subroutine called “radgrav” that calculates: 1. a confocal ellipsoid that passes through any point (P) and 2. normal gravity at P. NGS has translated the Fortran code into Matlab and both the Fortran and Matlab versions are available in Section 3.5.

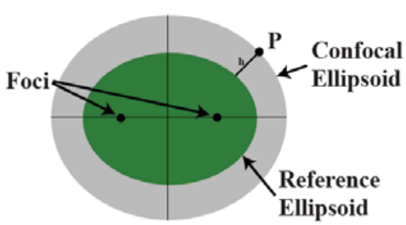

Figure 8: Illustration of the concept of two gravitationally-equivalent confocal ellipsoids.

For this confocal ellipsoid normal gravity correction, the following formula applies:

$$g_D = g_{observedFFG} - \gamma_{normal\_ellipsoid} \tag{13}$$

where g_D is the gravity disturbance, $g_{observedFFG}$ is the observed full-field gravity at altitude, and $\gamma_{normal\_ellipsoid}$ is normal gravity computed with the confocal ellipsoid method at the measurement's exact height above the ellipsoid.

Thus, this correction replaces both the free-air correction and normal gravity at Earth's surface correction, in one step.

3.2.2 2nd order Free-Air Gravity Disturbance (FAD)

The FAD is the observed full-field gravity at altitude corrected for 1. the free-air effect using ellipsoidal height and 2. normal gravity on the ellipsoid.

$$g_{FAD} = g_{observedFFG} - g_{FAC\_ellipsoid} - \gamma_{normal\_ellipsoid\_surface} \tag{14}$$

where g_{FAD} is the free-air gravity disturbance, $g_{observedFFG}$ is the observed full-field gravity at altitude, $g_{FAC\_ellipsoid}$ is the 2nd order free-air correction using ellipsoidal height, and $\gamma_{normal\_ellipsoid\_surface}$ is normal gravity computed on the ellipsoid.

There are two 2nd order free-air corrections currently in use, which are of similar accuracy. The difference is that the Hofmann-Wellenhof and Moritz (2006) equation includes an additional approximation of Earth's surface normal gravity, likely included for faster computation back when it was developed in the 1960s and computers were slow. The more accurate of the two 2nd order corrections in use (Damiani, 2013; Featherstone and Dentith, 1997; Featherstone, 1995) does not include an approximation of surface normal gravity.

The slightly MORE accurate 2nd order free-air correction ($g_{FAC\_ellipsoid}$) equation (from Damiani, 2013; Featherstone and Dentith, 1997; Featherstone, 1995) is:

$$g_{FAC\_ellipsoid} = \frac{\partial \gamma}{\partial h} h + \frac{1}{2}\frac{\partial^2 \gamma}{\partial h^2} h^2 \approx -\frac{2\gamma_0}{a}(1 + f + m - 2f\sin^2\phi)h + \frac{3\gamma_0}{a^2}h^2 \tag{15}$$

where γ_0 is the normal gravity on the ellipsoid at a given latitude, a is the semi-major axis for the ellipsoid, b is the semi-minor axis for the ellipsoid, f is the flattening, ϕ is geodetic latitude, and h is ellipsoidal height at the measurement point. All additional parameters are defined by the chosen reference ellipsoid with the following additional equations:

$$f = \frac{(a - b)}{a} \tag{16}$$

$$m = \frac{\omega^2 a^2 b}{GM} \tag{17}$$

The Somigliana-Pizetti normal gravity ($\gamma_0 = \gamma_{normal\_ellipsoid\_surface}$) formula for numerical calculations (Hofmann-Wellenhof and Moritz, 2006 (2-146)) is:

$$\gamma_0 = \gamma_{normal\_ellipsoid\_surface} = \frac{a\gamma_a \cos^2\phi + b\gamma_b \sin^2\phi}{\sqrt{a^2\cos^2\phi + b^2\sin^2\phi}} \tag{18}$$

where γ_a is equatorial normal gravity on the ellipsoid, γ_b is polar normal gravity on the ellipsoid, a is the semi-major axis, b is the semi-minor axis, e^2 is the first eccentricity squared, and ϕ is geodetic latitude.

The slightly LESS accurate 2nd order free-air correction ($g_{FAC\_ellipsoid}$) equation (from Hofmann-Wellenhof and Moritz, 2006 (2-215)) is:

$$g_{FAC\_ellipsoid} = \frac{\partial\gamma}{\partial h}h + \frac{1}{2}\frac{\partial^2\gamma}{\partial h^2}h^2 = -\frac{2\gamma_a}{a}\left(1 + f + m + \left(-3f + \frac{5}{2}m\right)\sin^2\phi\right)h + \frac{3\gamma_a}{a^2}h^2 \tag{19}$$

where γ_a is the equatorial normal gravity on the ellipsoid and all additional parameters are defined in the same way as Equation 16.

3.2.3 1st Order (Linear) Free-Air Gravity Disturbance (FAD)

Not recommended for any precision gravity application, a 1st order free-air correction exists. It approximates the surface vertical gravity gradient around the world due to the ellipsoid as an average:

$$g_{FAC\_ellipsoid} = \frac{\partial\gamma}{\partial h}h = 0.3086\ h \tag{20}$$

where h is ellipsoidal height. However, for nearly all gravity measurements, the error in using the 1st order free-air correction is prohibitively large (Figures 6 and 7).

3.3 Free-air Gravity Anomaly (FAA)

FAA values should always be reported with a statement of which geoid and ellipsoid were used for the calculation of the anomaly. An example would be to report “Free-air gravity anomaly values were calculated with respect to the EGM2008 geoid (NGA, 2008) and WGS-84 ellipsoid (NIMA, 2000) for the free-air correction and with respect to the WGS-84 (NIMA, 2000) for the Somigliana-Pizetti normal gravity correction.”

The FAA is the observed full-field gravity at altitude corrected for 1. the free-air effect using orthometric height and 2. normal gravity on the ellipsoid.

$$g_{FAA} = g_{observedFFG} - g_{FAC\_geoid} - \gamma_{normal\_ellipsoid\_surface} \tag{21}$$

where g_{FAA} is the free-air gravity anomaly, $g_{observedFFG}$ is the observed full-field gravity at altitude, g_{FAC_geoid} is the 2nd order free-air correction using orthometric height, and $\gamma_{normal\_ellipsoid\_surface}$ is normal gravity computed on the ellipsoid. The $\gamma_{normal\_ellipsoid\_surface}$ equation is the same for the FAA as for the FAD (Equation 19).

Given that we can consider the orthometric height to be along the ellipsoidal normal with some small error (see Equation 11), the free-air correction applied for a FAA is similar to the 2nd order FAD (Equation 16):

$$g_{FAC\_geoid} = \frac{\partial\gamma}{\partial h}H + \frac{1}{2}\frac{\partial^2\gamma}{\partial h^2}H^2 \approx -\frac{2\gamma_0}{a}(1+f+m-2f\sin^2\phi)H + \frac{3\gamma_0}{a^2}H^2 \quad (22)$$

Thus, the relationship between the FAD and FAA can be summarized as this (Hofmann-Wellenhof and Moritz, 2006):

$$g_{FAD} = g_{FAA} - \frac{\partial\gamma}{\partial h}N = g_{FAA} - \frac{\partial\gamma}{\partial h}(h-H) \quad (23)$$

where $\frac{\partial\gamma}{\partial h}$ is the vertical gravity gradient along ellipsoidal normal, N is the geoid height, H is the orthometric height, and h is the ellipsoidal height.

Note that the accuracy of the FAA is dependent on the accuracy of the orthometric height value used in the free-air correction. Choose a geoid that is well-defined in your study area to avoid propagating orthometric height errors into the FAA. As of 2017, the best global geoid model freely available is EGM2008 (NGA, 2008). The official gravimetric geoid model for the United States is USGG2012 (http:/www.ngs.noaa.gov/GEOID/USGG2012/) and is roughly equivalent to EGM2008. NGS also has a number of experimental geoids (xGEOIDs) available from (https://beta.ngs.noaa.gov/GEOID/xGEOID/). In 2022 NGS will be publishing GEOID2022 from the airborne gravity data that is expected to be more accurate in the United States.

3.5 Code

The functions below can be copied into separate Matlab .m files and the "calc_Disturbance.m" file will call the others in order to calculate a gravity disturbance or a free-air disturbance.

3.5.1 Confocal Ellipsoid Method for Normal Gravity at any Height and Latitude

3.5.1.1 Mathworks Matlab Code from the National Geodetic Survey

Content of the file "normal_gravity_confocal.m"

```
----------------------------------------------------------------------------
function [gamma] = normal_gravity_confocal(ityp,Ellipsoid,lat,height)
%
%Translated from NGA's Fortran by Theresa Damiani, NGS, November 2013
%Simplified for geodetic coordinate input only.
%
%From the FORTRAN subroutine "radgrav" in the hsynth_WGS84.f
%online codes. Written originally by Nikos K. Pavlis and Simon Holmes.
%Available:
%http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm2008/egm08_wgs84.html
%
%Documentation as follows is directly from the original subroutine,
%modified only to remove mentions of non-geodetic coordinates:
```

```
%"C  Subroutine to perform coordinate transformations on the meridian
% C  plane and to compute the components of normal gravity/gravitation
% C  vector (in ellipsoidal, geocentric, and geodetic coordinates),
% C  and the magnitude of the normal gravity/gravitation vector.
% C
% C  An equipotential ellipsoid of revolution and its associated
% C  Somigliana-Pizzetti normal gravity field underlie all computations.
% C
% C  The location of the computation point is input through its
% C  geodetic latitude and geodetic (ellipsoidal) height.
% C  Due to rotational symmetry, no longitude information is necessary.
% C
% C  This subroutine uses closed expressions that yield precise results
% C  regardless of the point's geodetic height."
%
%NOTE: GPS provides geodetic coordinates
%
%INPUT
%ityp  output type: 0 for normal gravitation, 1 for normal gravity
%Ellipsoid  structure  ellipsoidal parameters
%lat  latitude in degrees (geodetic)
%height  ellipsoidal height in meters
%
%OUTPUT
%gamma  full-field gravity/gravitation of ellipsoid at location (mGal)
%
%FUNCTION CALL
%For gravity (i.e. Mass and Rotational effects of Earth):
%[gamma]=normal_gravity(1,'wgs84',lat,h);
%
%For gravitation (i.e. Mass effect of Earth only):
%[gamma]=normal_gravity(0,'wgs84',lat,h);
%
%A disturbance at alitude is g_disturbance=ffg_measured-gamma;
%where ffg_measured is the output of NGS' Newton software and is the
%gravity value of our GRAV-D products released on the web.
%
%REFERENCES
%Heiskanen, W. and H. Mortiz (1967). Physical Geodesy. W.H. Freedman and
%Company: San Francisco. 354pp.
%
%Moritz, H. (2000). Geodetic Reference System 1980 (vol 66, pg 2, 1992).
%[Correction]. Journal of Geodesy, 74(1), 128-133.
%
%NIMA (2004). Department of Defense World Geodetic System 1984: Its
%Definition and Relationshops with Local Geodetic Systems. NIMA Technical
%Report (Vol. Stock No. DMATR83502WGS84; NSN 7643-01-402-0347, pp. 175).

%% Ellipsoidal parameters
a=Ellipsoid.a;
b=Ellipsoid.b;
e=Ellipsoid.e;
e2=Ellipsoid.e2;
GM=Ellipsoid.GM;
omega=Ellipsoid.omega;
```

```
%% Other parameters
e21=1-(e2);

dtr=pi./180;
asqr=a.^2;
omega2=omega.^2;
Esqr=asqr.*e2;
E=sqrt(Esqr); %linear eccentricity, H&M p. 74
e_prime=E./b; %second eccentricity, H&M p. 74
ep2=E.^2./b.^2;

q0=0.5.*((1+3./(ep2)).*atan(e_prime)-(3./(e_prime))); %H&M p.66, 2-58

%Convert to radians
phi=dtr.*lat;

%% Calculate Coordinates psi and beta (geodetic coordinates)
sphi  = sin(phi);
cphi  = cos(phi);
sphi2 = sphi.*sphi;
dn=a./sqrt(1-e2.*sphi2);
p=(dn+height).*cphi;
z=(dn.*e21+height).*sphi;
z2=z.^2;
%Compute geocentric distance (m)
r=sqrt(p.*p+z2);

%Test for critical locations (poles/equator)
beta=atan(sqrt(e21).*sin(phi)./cos(phi));

%% Compute semi-minor axis of confocal ellipsoid (m)
dummy   = r.*r-Esqr;
usqr    = .5.*dummy.*(1+sqrt(1+4*Esqr.*z2./(dummy.*dummy)));
u       = sqrt(usqr);

uE     = usqr + Esqr;
uEi    = 1./uE;
uErt   = sqrt(uE);
uErti  = 1./uErt;
uEb    = usqr + Esqr.*sin(beta).^2;
uEbrt  = sqrt(uEb);

Ei     = 1./E;
uoE    = u./E;
uoE2   = uoE.*uoE;
Eou    = E./u;
atEu   = atan(Eou);
const  = omega2.*a.*a.*E./q0;
const2 = const.*Ei;
fnb    = (0.5.*sin(beta).^2 - 1./6);

q    = 0.5.*((1 + 3.*uoE2).*atEu - 3.*uoE);
q1   = 3*(1 + uoE2).*(1 - uoE.*atEu) - 1;
w = uEbrt./uErt;
wi = 1./w;
```

```
q1uEi = q1.*uEi;

fu    = GM.*uEi + const.*fnb.*q1uEi - ityp.*omega2.*cos(beta).^2.*u;
gamu  = 1E5.*(-wi.*fu);

fb   = (-const2.*q.*uErti + ityp.*omega2.*uErt);
gamb = 1E5.*(-wi.*fb.*cos(beta).*sin(beta));

%% Compute normal gravity/gravitation (mGal)
%Ellipsoidal System
gamma     = sqrt(gamu.*gamu + gamb.*gamb);
end
----------------------------------------------------------------------------
```

3.5.1.2 Fortran Code from the National Geospatial-Intelligence Agency (NGA, 2010)

```
SUBROUTINE RADGRAV(IFORM,ISKIP,IREP,ITYP,A,E2,E21,Q0,GM,OMEGA,
     &                    PHI,H,PSI,R,BETA,U,GAMMAE,GAMMAG,GAMMAD,GAMMA,
     &                    upot,gam_sh,gam_sr)
C-----------------------------------------------------------------------
C
C  Subroutine to perform coordinate transformations on the meridian
C  plane and to compute the components of normal gravity/gravitation
C  vector (in ellipsoidal, geocentric, and geodetic coordinates),
C  and the magnitude of the normal gravity/gravitation vector.
C
C  An equipotential ellipsoid of revolution and its associated
C  Somigliana-Pizzetti normal gravity field underlie all computations.
C
C  The location of the computation point is input through either its
C  geodetic latitude and geodetic (ellipsoidal) height if IFORM=1, or
C  its geocentric latitude and geocentric distance if IFORM=2.
C  Due to rotational symmetry, no longitude information is necessary.
C
C  This subroutine uses closed expressions that yield precise results
C  regardless of the point's geodetic height.
C
C  Note: The components gamma_u, gamma_r, gamma_h are positive
C        outwards, and the components gamma_beta, gamma_psi, gamma_phi
C        are positive northwards.
C
C
C  Input
C  =====
C
C  IFORM: Flag such that: IFORM=1 ==> input are geodetic coordinates
C                         IFORM=2 ==> input are geocentric coordinates
C
C  ISKIP: If ISKIP=0 then GAMMAG and GAMMAD computations are skipped
C
C   IREP: Flag that should equal 0 the first time this s/r is called
C
C   ITYP: Flag such that:  ITYP=0 ==> normal gravitation computations
C                          ITYP=1 ==> normal gravity computations
```

```
C
C      A: Semi-major axis (m)
C     E2: First eccentricity squared
C    E21: 1-E2
C     Q0: Term defined in [Heiskanen & Moritz, 1967, p. 66, eq. (2-58)]
C     GM: Geocentric gravitational constant (m^3/s^2)*1.d5
C  OMEGA: Rotational rate (rad/s)
C
C    PHI: Geodetic latitude (rad)                      if IFORM=1
C      H: Geodetic (ellipsoidal) height (m)            if IFORM=1
C
C    PSI: Geocentric latitude (rad)                    if IFORM=2
C      R: Geocentric distance (m)                      if IFORM=2
C
C
C  Output
C  ======
C
C        PHI: Geodetic latitude (rad)                  if IFORM=2
C          H: Geodetic (ellipsoidal) height (m)        if IFORM=2
C
C        PSI: Geocentric latitude (rad)                if IFORM=1
C          R: Geocentric distance (m)                  if IFORM=1
C
C       BETA: Reduced latitude (rad)
C          U: Semi-minor axis of confocal ellipsoid (m)
C
C  GAMMAE(1): gamma_u    (mGal)
C  GAMMAE(2): gamma_beta (mGal)
C  GAMMAG(1): gamma_r    (mGal)                        if ISKIP .ne. 0
C  GAMMAG(2): gamma_psi  (mGal)                        if ISKIP .ne. 0
C  GAMMAD(1): gamma_h    (mGal)                        if ISKIP .ne. 0
C  GAMMAD(2): gamma_phi  (mGal)                        if ISKIP .ne. 0
C
C      GAMMA: magnitude of normal gravity vector (mGal)
c
c-----------------------------------------------------------------------
c     CLOSED EXPR. 4 pot'l,d(gam)/dh,d(gam)/dr:   SIMON HOLMES, JUL 2004
c-----------------------------------------------------------------------
c
      implicit real*8(a-h,o-z)
      dimension gammae(2),gammag(2),gammad(2)
      save
      data pi/3.14159265358979323846d+00/,eps/1.d-10/,iter_max/2/
      data ix/0/
c
c-----------------------------------------------------------------------
c
c     if(irep.eq.0) then
c
      if (ix.eq.0) then
        ix     = 1
        dtr    = pi/180.d0
        asqr   = a*a
```

```
        omega2 = omega*omega
        Esqr   = asqr*e2
        E      = sqrt(Esqr)
        cst    = sqrt(e21)
        b      = sqrt(asqr-Esqr)
        f13 = 1.d0/3.d0
      endif  !  ix
c
C-----------------------------------------------------------------------
c
      if(iform.eq.1) then  ! (input are geodetic coordinates)
        sphi  = sin(phi)
        cphi  = cos(phi)
        sphi2 = sphi*sphi
        dn    = a/sqrt(1.d0-e2*sphi2)
        p     = (dn    +h)*cphi
        z     = (dn*e21+h)*sphi
        z2    = z*z
C
C  Compute geocentric distance (m).
C
        r = sqrt(p*p+z2)
C
C  Test for critical locations (Poles and Equator).
C
        test = dabs(phi)-90.d0*dtr
        if(dabs(test).lt.eps.or.dabs(phi).lt.eps) then
          psi  = phi
          beta = phi
C
        else  ! Point not at Poles or Equator
C
C  Compute geocentric latitude (rad).
C
          psi  = atan(z/p)
C
C  Compute reduced latitude (rad).
C
          beta = atan(cst*sphi/cphi)
        endif  ! Poles and Equator when iform=1
C
C  Compute terms needed for normal gravity/gravitation computation.
C
        spsi   = sin(psi)
        cpsi   = cos(psi)
        sbeta  = sin(beta)
        sbeta2 = sbeta*sbeta
        cbeta  = cos(beta)
C
      else  ! iform=2 (input are geocentric coordinates)
C
C  Test for critical locations (Poles and Equator).
C
        test = dabs(psi)-90.d0*dtr
```

```
        if(dabs(test).lt.eps.or.dabs(psi).lt.eps) then
c
          phi    = psi
          beta   = psi
          spsi   = sin(psi)
          cpsi   = cos(psi)
          p      = r*cpsi
          z      = r*spsi
          z2     = z*z
          sbeta  = spsi
          sbeta2 = sbeta*sbeta
          cbeta  = cpsi
          cbeta2 = cbeta*cbeta
          sphi   = spsi
          cphi   = cpsi
          h      = (p-a*cbeta)*cphi+(z-b*sbeta)*sphi
C
        else  ! Point not at Poles or Equator
C
          spsi     = sin(psi)
          cpsi     = cos(psi)
          p        = r*cpsi
          z        = r*spsi
          z2       = z*z
          ap       = a*p
          bp       = b*p
          az       = a*z
          bz       = b*z
          rprim    = sqrt(ap*ap+bz*bz)
          cmega    = atan2(bz,ap)
          c        = Esqr/rprim
          beta0    = atan2(az,bp)
          iter_num = 0
C
C  Compute iteratively reduced latitude (rad).
C
    1     twobeta0 = 2.d0*beta0
          s2beta0  = sin(twobeta0)
          c2beta0  = cos(twobeta0)
          diff     = beta0-cmega
          sdiff    = sin(diff)
          cdiff    = cos(diff)
          beta     = beta0-(sdiff-0.5d0*c*s2beta0)/(cdiff-c*c2beta0)
          iter_num = iter_num + 1
          if(iter_num.lt.iter_max) then
            beta0    = beta
            goto 1
          endif  ! Finish iterative calculation of beta
C
C  Compute auxilliary terms and geodetic latitude (rad).
C
          sbeta  = sin(beta)
          sbeta2 = sbeta*sbeta
          cbeta  = cos(beta)
```

```
          cbeta2 = cbeta*cbeta
          csbeta = cbeta*sbeta
          phi    = atan(a*sbeta/(b*cbeta))
          sphi   = sin(phi)
          cphi   = cos(phi)
C
C  Compute geodetic (ellipsoidal) height (m).
C
          h      = (p-a*cbeta)*cphi+(z-b*sbeta)*sphi
C
        endif  ! Poles and Equator when iform=2
C
      endif  ! All iform cases considered
C
C  Compute semi-minor axis of confocal ellipsoid (m).
C
      dummy   = r*r-Esqr
      usqr    = .5d0*dummy*(1.d0+sqrt(1.d0+4.d0*Esqr*z2/(dummy*dummy)))
      u       = sqrt(usqr)
c
c-----------------------------------------------------------------------
c
      uE     = usqr + Esqr
      uEi    = 1.d0/uE
      uE2i   = 1.d0/(uE*uE)
      uErt   = dsqrt(uE)
      uErti  = 1.d0/uErt
      uEb    = usqr + Esqr*sbeta2
      uEbrt  = dsqrt(uEb)
      uEbrti = 1.d0/dsqrt(uEb)
c
      ui     = 1.d0/u
      u3     = u*u*u
      Ei     = 1.d0/E
      E3     = E*E*E
      uoE    = u/E
      uoE2   = uoE*uoE
      Eou    = E/u
      Eou2   = Eou*Eou
      atEu   = datan(Eou)
      frac   = 1.d0/(1.d0 + Eou2)
      const  = omega2*a*a*E/q0
      const2 = const*Ei
      fnb    = (0.5d0*sbeta2 - 1.d0/6.d0)
c
      q    = 0.5d0*((1.d0 + 3.d0*uoE2)*atEu - 3.d0*uoE)
      q1   = 3.d0*(1.d0 + uoE2)*(1.d0 - uoE*atEu) - 1.d0
      w = uEbrt/uErt
      wi = 1.d0/w
      q1uEi = q1*uEi
c
c-----------------------------------------------------------------------
c
      fu    = 1.d-5*gm*uEi + const*fnb*q1uEi - ityp*omega2*cbeta2*u
```

```
      gamu  = 1.d5*(-wi*fu)
c
      fb   = (-const2*q*uErti + ityp*omega2*uErt)
      gamb = 1.d5*(-wi*fb*csbeta)
c
c-----------------------------------------------------------------------
c
      dwi   = u*(1.d0/(uErt*uEbrt) - uErt/(uEbrt**3))
      dq1u  = 3.d0*(ui*frac - Ei*atEu + (2.d0*u/Esqr)
     &                      - 3.d0*uoE2*Ei*atEu + (u/Esqr)*frac)
      duEi   = -2.d0*u*uE2i
      dq1uEi = dq1u*uEi + q1*duEi
      dfu    = 1.d-5*gm*duEi +const*fnb*dq1uEi - ityp*omega2*cbeta2
      dq = -0.5d0*((Eou*ui + 3.d0*Ei)*frac -6.d0*uoE*Ei*atEu + 3.d0*Ei)
      dfb = (-const2*(dq*uErti - q*u*uErti**3) + ityp*omega2*u*uErti)
      dfudb = const*q1uEi*csbeta + 2.d0*csbeta*ityp*omega2*u
      dwidb = -wi*Esqr*csbeta/uEb
c
c-----------------------------------------------------------------------
c
      dgudu  = -wi*dfu -dwi*fu
      dgbdu  = (-wi*dfb - dwi*fb)*csbeta
      dgudb  = -wi*dfudb - dwidb*fu
      dgbdb  = fb*(-dwidb*csbeta - wi*(cbeta2 - sbeta2))
c
      dgudu_s  =     wi *dgudu *1.d5
      dgbdu_s  =     wi *dgbdu *1.d5
      dgudb_s  = uEbrti *dgudb *1.d5
      dgbdb_s  = uEbrti *dgbdb *1.d5
c
      cent = 5.d4*ityp*omega2*uE*cbeta2
      upot = (GM/E)*atEu +5.d4*const2*q*(sbeta2-f13) + cent
c
c-----------------------------------------------------------------------
c
C  Compute normal gravity/gravitation components and magnitude (mGal).
C
C-- Ellipsoidal System -------------------------------------------------
C
      gammae(1) = gamu
      gammae(2) = gamb
      gamma     = sqrt(gammae(1)*gammae(1) + gammae(2)*gammae(2))
c
      gam_su = (gamu*dgudu_s + gamb*dgbdu_s)/gamma
      gam_sb = (gamu*dgudb_s + gamb*dgbdb_s)/gamma
c
      if(iskip.eq.0) goto 99
C
C-- Geocentric System --------------------------------------------------
C
      dummy     = u/uErt
      gammag(1) = wi*((dummy*cbeta*cpsi + sbeta*spsi)*gammae(1) +
     $                (dummy*cbeta*spsi - sbeta*cpsi)*gammae(2))
      gammag(2) = wi*((sbeta*cpsi - dummy*cbeta*spsi)*gammae(1) +
```

```
     $                 (sbeta*spsi + dummy*cbeta*cpsi)*gammae(2))
c
      gam_sr = wi*((dummy*cbeta*cpsi + sbeta*spsi)*gam_su +
     &                 (dummy*cbeta*spsi - sbeta*cpsi)*gam_sb)
      gam_sy = wi*((sbeta*cpsi - dummy*cbeta*spsi)*gam_su +
     &                 (sbeta*spsi + dummy*cbeta*cpsi)*gam_sb)
C
C-- Geodetic System ---------------------------------------------------
C
      alfa    = phi - psi
      salfa   = sin(alfa)
      calfa   = cos(alfa)
      gammad(1) =  gammag(1)*calfa + gammag(2)*salfa
      gammad(2) = -gammag(1)*salfa + gammag(2)*calfa
c
      gam_sh =  gam_sr*calfa + gam_sy*salfa
      gam_st = -gam_sr*salfa + gam_sy*calfa
c
   99 return
      end
```

3.5.2 Gravity Disturbance or Free-Air Disturbance

Content of file named “calc_Disturbance.m”.

```
----------------------------------------------------------------------------
function Disturbance = calc_Disturbance(lat, height, ffg, ellipType, method)
% calc_Disturbance(lat, height, ffg, method) calculates either a gravity
% disturbance or a free air disturbance given a certain latitude, height,
% and full field gravity value. The user must select with method they wish
% to use: confocal (1 step) disturbance, or Free Air Correction (2 step)
% Free Air Disturbance.
%
% Inputs
%   lat (double)        Geodetic latitude (decimal degrees)
%   height (double)     ELLIPSOIDAL height (m)
%   ffg (double)        Full-Field Gravity at Altitude (mGal)
%   ellipType (string)  ellipsoid: accepts 'WGS84' or 'GRS80'
%   method (int)        1 for confocal method
%                       2 for 2nd order free air correction method
%
%   Output
%       Disturbance     Gravity disturbance or Free Air Disturbance
%                       depending on method choice (mGal)
%
%% Define ellipsoid parameters
Ellipsoid = make_Ellipsoid(ellipType);

%% Calculate Gravity Disturbance if method 1 is selected
if method == 1
    gnorm_at_altitude = normal_gravity_confocal(1, Ellipsoid, lat, height);
    Disturbance = ffg - gnorm_at_altitude;

% Calculate Free Air Disturbance if method 2 is selected
elseif method == 2
```

```
    fac = calc_FAC(lat, Ellipsoid, height); % free air correction
    gnorm = calc_gnorm_ellip_surface(lat, Ellipsoid); % normal gravity
    Disturbance = ffg - fac - gnorm;

% Give error and return control to invoking function if neither method 1 nor
% 2 was chosen.
else
    fprintf('Method must either be a 1 or 2');
    return
end
end
-----------------------------------------------------------------------------
```

3.5.3 2nd order Free-Air Correction

Content of file named “calc_FAC.m”.

```
-----------------------------------------------------------------------------
function fac=calc_FAC(lat, Ellipsoid, height)
% calc_FAC(lat, ellipType, height) calculates a 2nd order free air
% correction to a reference surface given a latitude or height (ellipsoidal
% or orthometric)
%
%   Inputs:
%       lat (double)        Geodetic latitude (decimal degrees)
%       Ellipsoid (struct)  ellipsoid parameters
%       height (double)     ellipsoid height or orthometric height (m)
%
%   Output:
%       fac (double)        free air correction (mGal)
%
%% Convert latitude from decimal degrees to radians
latr = lat*pi/180;
sinlatr2 = (sin(latr)).^2;

% Calculate normal gravity on the ellipsoid at the given latitude
normGrav = calc_gnorm_ellip_surface(lat, Ellipsoid);

% Calculate ellipsoid parameter m
m = Ellipsoid.omega^2*Ellipsoid.a^2*Ellipsoid.b/Ellipsoid.GM;

% 2nd order free air correction: formula 16 for ellipsoid reference surface
% and equation 23 for geoid reference surface from the GRAV-D General User
% Manual
c1 = (-2*normGrav)/Ellipsoid.a;
term1 = (1 + Ellipsoid.f + m - 2*Ellipsoid.f.*sinlatr2).*height;
c2 = (3.*normGrav./(Ellipsoid.a^2));
term2 =height.^2;

fac = (c1.*term1) + (c2.*term2);
end
-----------------------------------------------------------------------------
```

3.5.4 Normal Gravity Correction

Content of file named “calc_gnorm_ellip_surface.m”.

```
----------------------------------------------------------------------------
function gnorm=calc_gnorm_ellip_surface(lat, Ellipsoid)
% calc_gnorm_ellip_surface(lat, ellipType) calculates normal gravity values
on the surface
% of the ellipsoid at the given latitude.
%
%   Inputs:
%       lat (double)        geodetic latitude (decimal degrees)
%       ellipsoid (struct)  ellipsoid parameters
%
%   Output:
%       gnorm (double)  normal gravity on the surface of the ellipsoid at
%       given latitude (mGal)
%
%% Convert from decimal degrees to radians
latr=lat*pi()/180;

% Somigliana-Pizetti normal gravity formula. Equation 19 from GRAV-D
% General User manual
n1 = Ellipsoid.a .* Ellipsoid.gammaEquator .* (cos(latr)).^2 + Ellipsoid.b .*
...
    Ellipsoid.gammaPole .* (sin(latr)).^2;
d1 = sqrt(Ellipsoid.a^2.*(cos(latr)).^2 + Ellipsoid.b^2.*(sin(latr)).^2);

gnorm = n1./d1;
end
---------------------------------------------------------------------------
```

3.5.5 Normal Gravity Correction

Content of file named “make_Ellipsoid.m”.

```
---------------------------------------------------------------------------
function [ellipsoid] = make_Ellipsoid(ellipType)

% get_Ellipsoid(ellipType) returns a structure containing ellipsoid
% parameters for the 'WGS84' and the 'GRS80' ellipsoids.
%
%   Input:
%       ellipType (string) Accepts 'WGS84' or 'GRS80'
%
%   Output: ellipsoid (Struct)
%       fields: GM (double)             Earth's Gravitational Constant
(m^3/s^2)
%               omega (double)          Angular Velocity of Earth (rad/s)
%               gammaEquator (double)   Equatorial Normal Gravity (mGal)
%               gammaPole (double)      Polar Normal Gravity (mGal)
%               a (double)              Semi-major Axis (m)
%               b (double)              Semi-minor Axis (m)
%               e (double)              first eccentricity (unitless)
%               e2 (double)             first ecentricity squared (unitless)
%               f (double)              flattening (unitless)
```

```
%
switch ellipType
    case 'WGS84'
        ellipsoid.GM = 3986004.418E8;
        ellipsoid.omega = 7.292115e-5;
        ellipsoid.gammaEquator = 978032.53359;
        ellipsoid.gammaPole = 983218.49378;
        ellipsoid.a = 6378137.0;
        ellipsoid.b = 6356752.3142;
        ellipsoid.e = 8.1819190842622.*1E-2; %first eccentricity
        ellipsoid.e2 = 6.69437999014.*1E-3; %first eccentricity squared
        ellipsoid.f = 1/298.257223563; %flattening
        %WGS84 defining parameters from NIMA (2004).
    case 'GRS80'
        ellipsoid.GM=3986005E8;
        ellipsoid.omega=7.292115e-5;
        ellipsoid.gammaEquator=978032.67715;
        ellipsoid.gammaPole = 983218.63685;
        ellipsoid.a = 6378137;
        ellipsoid.b = 6356752.3142;
        ellipsoid.e = 0.081819190842622; %first eccentricity
        ellipsoid.e2 = 0.00669437999014; %first eccentricity squared
        ellipsoid.f = 0.00335281066474; %flattening
        %GRS80 defining parameters from Moritz (2000).
    otherwise
        warning('This ellipsoid is not supported')
        return
end
end
----------------------------------------------------------------------------
```

Appendix A: Data Formats

Table 6: Publicly released data, By Block (a compilation of all the data from every flight): NGS_GRAVD_Block_BB##_Gravity_Data_BETA1.txt and NGS_GRAVD_Block_BB##_Supplement#_BETA1.txt

Column #	Data	Format	Units
1	Block + Line Number	%7s	None
1	Time	%5d or %17d	Seconds since start of day or UTC Time: yyyymmddHHMMSSFFF*
2	Latitude	%3.8f	Signed dec. degrees
3	Longitude	%3.8f	Signed dec. degrees
4	Ellipsoidal Height (at altitude)	%5.3f	Meters
5	Filtered Full-Field Gravity (at altitude)	%7.2f	mGals

*yyyy = 4 digit year, mm = 2 digit month, dd = 2 digit day, HH = 2 digit 24 hours, MM = 2 digit minutes, SS = 2 digit seconds, FFF = 3 digit miliseconds.

Appendix B: Glossary

Block

A geographically-defined section of the country that is planned with airborne gravity data lines and cross lines to provide a self-consistent data set, with meaningful crossover-error statistics. The name for a block is generated from the time zone in which it is located, north or south portion, and number of blocks that have been planned in that time zone.

Cross Line

A widely-spaced (40-100 km) line of gravity data that was collected for the purpose of calculating the error of the data lines. The cross line gravity data are considered supplemental and may not be included in any NGS geoid products.

Data Line

A closely-spaced (10 km or less) line of gravity data that was collected for inclusion into NGS gravimetric geoid models, toward a new vertical datum.

Equipotential Surface

A surface along which the gravitational potential is constant.

Flight

The time period that includes pre-flight preparation, take-off, data collection, landing, and post-flight completion.

Geoid

The equipotential surface that most closely approximates mean sea level.

GRAV-D

Gravity for the Redefinition of the American Vertical Datum program, started by the National Geodetic Survey in 2007 for the purpose of collecting airborne gravity data toward adopting a 1-2 cm accuracy (where possible) gravimetric geoid in 2022.

NGS

The National Geodetic Survey, a Program Office within NOAA's National Ocean Service, has a mission "To define, maintain, and provide access to the National Spatial Reference System (NSRS) to meet our nation's economic, social, and environmental needs." The GRAV-D project us a part of the NGS Ten Year Plan to modernize the National Spatial Reference System by 2022.

Newton

The airborne gravity processing software developed in-house by NGS, to meet the needs of high-speed, high-altitude gravity data processing for geodesy purposes.

Survey

An occupation of the field team with a particular aircraft and instrument suite at a specific airport. The name for a survey is generated by the state in which the airport of operations is located, the calendar year operations commenced, and the number of times they've operated out of that state in the current calendar year.

TAGS

Turn-key Airborne Gravimetry System, manufactured by Micro-g LaCoste, Inc. to measure the acceleration due to gravity from aircraft, using a spring-type gravity sensor in a gyro-stabilized platform.

Appendix C: References

Childers, V. A., Robin E. Bell, and John M. Brozena. "Airborne Gravimetry: An Investigation of Filtering." *Geophysics* 64, no. 1 (1999): 61-69.

Damiani, T.M., Bilich, A., Mader, G.L., "Evaluating Aircraft Positioning Methods for Airborne Gravimetry: Results from GRAV-D's "Kinematic GPS Processing Challenge"," *Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2013)*, Nashville, TN, September 2013, pp. 3489-3507.

Federal Geographic Data Committee, 1998. *Geospatial Positioning Accuracy Standards*, FGDC-STD-007.2-1998, Federal Geographic Data Committee, Reston, Virginia, USA, 128 pp., www.fgdc.gov/standards/projects/FGDC-standards-projects/accuracy/.

Hackney, R.I., Featherstone, W.E., 2003. "Geodetic versus geophysical perspectives of the 'gravity anomaly'." *Geophysical Journal International,* **154**, 35-43.

Harlan, R.B., 1968. "Eötvös Corrections for Airborne Gravimetry." *J Geophys Res,* **73**, 4675-4679.

Hofmann-Wellenhof, B. and Moritz, H., 2006. Physical Geodesy. Springer-Verlag Wien: Austria. 403 pp.

Li, Xiong and Hans-Jürgen Götze (2001). "Tutorial: Ellipsoid, geoid, gravity, geodesy, and geophysics." *Geophys*, **66**(6), 1660-1668.

Micro-g LaCoste, 2010. "TAGS Turnkey Airborne Gravity System AIR III Hardware & Operations Manual."

Moritz, Helmut (2000). "Geodetic Reference System 1980." *Journal of Geodesy*, **74**(1), 128-162. DOI: 10.1007/s001900050278. Available online as of 06 September 2011: http://www.springerlink.com/content/0bgccvjj5bedgdfu/

Peters, M.F., Brozena, J.M., 1995. "Methods to improve existing shipboard gravimeters for airborne gravimetry," in: K.-P. Schwartz, J.M. Brozena, G.W. Hein, (Eds), IAG Symposium on Airborne Gravity Field Determination, IUGG XXI General Assembly, Calgary, Boulder, CO, pp. 39-45.

National Geospatial-Intelligence Agency- NGA (2008). "Earth Gravitational Model 2008 (EGM2008)." Available Online as of 06 September 2011: http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm2008/index.html

National Imagery and Mapping Agency- NIMA (2000). "Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems." Third Edition, Amendment 1. NIMA Stock No. DMATR83502WGS84, NSN 7643-01-402-0347. Available online as of 06 September 2011: http://earth-info.nga.mil/GandG/publications/tr8350.2/tr8350_2.html