ODTK Applied to GNSS Satellite Orbits and Clocks

1 Foreword

ODTK is specifically designed to support operations of the GNSS satellite system through orbit and clock estimation. ODTK also allows an analyst to simulate, solve for, and otherwise perform analysis on the GNSS satellites, their clocks, and the monitor stations that support GNSS operations. There are several ways to use ODTK to these ends, both with simulated data and with live data. ODTK supports the following GNSS satellite constellations: GPS, QZSS, GLONASS, Galileo, and BeiDou.

The purpose of this document is to provide an overview of these capabilities and to provide a “how-to” guide for various tasks. Since this is perhaps the most complicated application of ODTK, this guide is written assuming that you have worked through various ODTK tutorials, are familiar with the ODTK GUI, and are more interested in workflow than basic product orientation. This guide also assumes that you are familiar with GNSS satellite systems, monitor station systems, satellite Block designations, clock types, the need to steer clock ensembles, international standard formats for GNSS data, and standard measurement types such as pseudorange and phase.

“GNSS Orbit Determination” in this case is a misnomer, because the task requires estimation of orbit and clock states. The orbit is actually “easy” while clock estimation is more difficult. The difficulty arises because every clock is different, with different stochastic model parameters (process noise settings). Additional complexity arises because parameters change unpredictably as the clock ages, and change significantly due to clock events such as resets in phase and frequency or even clock replacement.

In an operational setting, old clocks are replaced by backup clocks, old satellites are replaced, and certain events can cause a satellite to be deemed “unusable” for a period of time.

The following discussion will address which of these issues ODTK handles.

2 Publicly available information

High-precision post-fit ephemerides and clock estimates are available from several sources on the internet. The “standard” format for such data is the SP3 format, which is defined in several places on the internet. ODTK supports SP3-a, SP3-c, and SP3-d formats. You can download SP3 files from the National Geospatial-Intelligence Agency (NGA) website in either of two forms, one as the ephemeris of the center of mass of the vehicle and another where the ephemeris is defined at the antenna phase center for the space vehicle (SV). Other sources of SP3 files include Jet Propulsion Laboratory (JPL), International GNSS Service (IGS), and the European Space Agency (ESA).

ODTK provides tools to read, manipulate, and exploit SP3 data. AGI has found several common abuses of the standard SP3 format, and ODTK traps those. However, AGI does not claim to trap all abuses.

Except for NGA, most internet sources also provide Monitor Station data, usually in a standard format called RINEX. ODTK provides standard readers for standard RINEX formats. ODTK does not supply readers for Compact RINEX, but format conversion software is available on the internet to convert Compact RINEX to standard RINEX. Some monitor stations abuse the RINEX standard; if that happens, ODTK may not read the data. Usually this results in an error message, but the message is only that “file xxx cannot be read” without diagnostics on the exact nature of the format error, since there are a number of creative ways to abuse the format.

You can find long-term clock stability data on the internet for individual Navstar clocks as well as AF and NGA Monitor Station clocks. The primary source of this data is the proceedings of the Precise Time and Time Interval (PTTI) annual meetings. These are on the /ptti page of tycho.usno.navy.mil or archived by search engines from the /ptti page. AGI has not identified a source for clock performance for other monitor stations and does not expect to find one, since most of the civilian monitor stations steer their clocks to GPS Time.

While these data are readily available, other critical data required to set up ODTK for GNSS analyses are not generally available, particularly clock phase and frequency “biases” for monitor stations, solar pressure coefficients for the SVs, and solar pressure process noise settings. You need to derive these to properly define an ODTK Scenario.

The following discussion includes tips on ways to define these data.

3 Populate the GNSS constellation

GNSS Satellite objects appear in the Object Browser as children of a GNSS Constellation object. AGI has shortened the GNSS Constellation object name to the constellation type, such as “GPS” and “QZSS.” This saves space in reports and graphs. To populate a GNSS Constellation with satellites, you need one or more files. You must have a catalog TXT file, which the ODTK install provides in C:\ProgramData\ AGI\ODTK 7\Databases\GNSS\. When ODTK is not estimating GNSS satellite orbits, you must also supply a reference file, which can come from one of the following types: AUX, RINEX, SEM, SP3, or YUMA. The reference file provides timely data describing the position and velocity of the satellites along with the clock offsets for each satellite clock relative to the GNSS system master clock.

A catalog TXT file contains a set of default values for each SV, including dates when the satellite became active and when it was retired, the Block type, the clock type, the PRN assigned, and various default values such as satellite mass. The reference file contains the actual orbit states and clock states for each GNSS SV. It is important that the content of the catalog file be compatible with the content of the reference file to achieve the most accurate solutions. For this reason, both NGA- and IGS-compatible catalog files are provided for GPS.

The ODTK-provided GNSS catalog files (e.g., GPSCatalog.txt) are current a month before the release date for ODTK. You can download updated catalog files using the Data Update Utility; see the Utility tab on the top menu bar.

Note whether the reference file provides SV position and velocity at the SV antenna phase center or at the SV body center of gravity. Set the corresponding option in the GNSS Constellation properties (SVReference.SVEphemerisReference.ReferencePoint) once the reference file finishes loading.

Once you assign the GNSS Catalog TXT file and the reference file in the GNSS Constellation Object Properties window, highlight the GNSS Constellation object in the Object Browser. This action will cause a *GNSS Constellation* top menu option to appear, with options to Add GNSS Satellites and Remove GNSS Satellites. Use the *Add GNSS Satellites* option to populate the constellation.

NOTES

- The *Add GNSS Satellites* option selects solar pressure models based on GNSS system and GPS Block type (II, IIA, IIR, IIRM, etc.) if applicable. The default models are the corresponding JPL models, which are currently used in most high-accuracy GPS analyses. The older II, IIA ROCK42 model, and tabular IIR model are available as options under each satellite's solar pressure force model attributes, but AGI does not recommend these for precision work.

- Default values for satellite masses are only approximate, using a single value for each Block of vehicles. AGI has not found an open source for actual satellite masses on the internet. Since SV mass is used in solar pressure and in maneuver modeling, small variations in mass will result in small differences in the computed forces, and if correction states are estimated, small differences in the estimated coefficients.

- Default transmit antenna offsets are defined based on the information available on the NGA and IGS websites at the time ODTK was released. You can obtain updated values from NGA, JPL, IGS and other agencies that publish on the internet.

- Solid Earth Tides are used by default for each GNSS SV. AGI recommends keeping this setting. The Earth gravity model is formally defined with a tidal model, and a Mean Tide component is subtracted from the gravity field, which you can only recover by applying solid earth tide perturbations to the orbit. This Mean Tide component is important at GNSS altitudes and GNSS accuracies. This setting is not obvious, but arises because AGI astrodynamics and Earth models carefully follow the international force model standards and corresponding recommendations for implementation.

- ODTK automatically sets Solar pressure EclipsingBodies to "Moon" when you add GNSS satellites to the constellation. GNSS satellites will occasionally pass through the Moon's shadow, with a measurable impact on solar pressure coefficients and consequential impacts on orbit and clock solutions.

- If the *Add GNSS Satellite* option seems to omit a satellite, it may be that the satellite is omitted from the reference file at the desired epoch. Most agencies will omit an SV from the reference over intervals where the satellite is designated as unusable by the Master Control Station.

- The clock frequency correction provided in the reference file by some agencies may not be the instantaneous rate of phase employed by ODTK. Sometimes a longer-term average value is provided. Rapid Filter convergence may require refining the initial clock frequency using the ODTK filter and smoother.

- The *Remove GNSS Satellites* option is also under the *GNSS Constellation* menu. Use great caution when selecting *Remove GNSS Satellites*, as this option wipes out all GNSS Constellation subobjects and all data entered in GNSS Satellite objects will be lost. ODTK provides a confirmation prompt to remind you of the consequences of removing satellites.

4 Monitor Stations

GNSS Monitor Stations (MS) are represented in the Object Browser as Facilities with GNSS Receivers attached.

4.1 Antenna switching

GNSS Receivers can have more than one antenna, such as a primary and a backup antenna. If the antenna ID is included with the GPS MS data, and is recognized by the measurement data provider, then ODTK can switch between antennas automatically. The locations provided in the Antenna objects are ECF Cartesian offsets to the facility location (sometimes called “eccentricities”). The antenna IDs in the tracking data must agree with the antenna IDs in the antenna object.

The international standard RINEX format does not conveniently provide for two or more antenna IDs for a single MS.

4.2 Multipath

ODTK has implemented an optional local horizon mask for each monitor to enable you to block out local features leading to multipath errors. As an alternative, you could simply use a large minimum elevation.

4.3 Monitor Station motion

Monitor station models in ODTK follow the international standards for station motion, as documented in IERS conventions. These effects include Tectonic Plate Motion, Ocean Tide Loading, Solid Earth Tide, and the Pole Tide. These are all important effects in the context of GPS satellite and clock estimation.

By default, these options are all set to false. Review the definitions for the MS coordinates and set these flags appropriately.

NOTES

- Some agencies frequently update station coordinates to account for tectonic plate motion.
- You cannot turn on the Ocean Tide Loading effect until you provide a site-specific file to set the local values. See ODTK Help for instructions for retrieving the proper file from the internet.

5 Optional modes of estimating GNSS states

Building up an ODTK scenario for an entire GNSS constellation from scratch requires the setting of many parameters. To get the best performance for ODTK, provide inputs that are reasonably “close” to the correct values. Use the following modes of operation to define these parameters.

- Solve for MS clock states while using the SP3 as a reference for ephemeris and clocks.
- Solve for orbits while using the SP3 as a reference for SV clocks.
- Solve for SV clocks while using the SP3 as a reference for SV orbits.
- Solve for all parameters simultaneously.
- Eliminate any or all MS clocks from the solution using single differencing.
- Eliminate all clocks from the solution using double differencing.
- Treat SP3 ECF positions as measurements input to ODTK.

6 GNSS solar pressure coefficients

The JPL solar pressure models for various GNSS Blocks of satellites all have two solve-for parameters used to adjust the model to the physical behavior of the satellite. These parameters are called K1 and K2 in the Object Properties window for the satellite, and are also called Scale and Y-Bias in various reports and graphs. You can find both designations in the open literature.

Initial values for these quantities are not available to the public and must be derived from the tracking data or from publicly available precision ephemerides. The following process uses the SP3 files and a few tricks to generate reasonable values for K1 and K2. It also provides some insight into how you should set the process noise for these parameters.

Normally it takes a week or more of data for solar pressure coefficient solutions to converge. SP3 files typically contain data covering a single day. To support a longer time period, you can add two or more SP3 files to the Source.Files list (see Section 12.1) or several contiguous SP3 files into one long SP3 file. Before running any utilities, be sure that the EOP files are up-to-date through the SP3 file end-time. Then run the utility SP3_To_Ephem.htm (from LaunchPad) to convert the concatenated SP3 to STK External ephemeris files (*.e). The name of the generated (*.e) file needs to contain a Tracking ID in the form of TrkID(nnnn)*.e, where "nnnn" represents the integer ID that will be assigned to a Satellite object in the steps below to allow the trajectory information to be processed as measurements.

To analyze each GNSS satellite one at a time, follow these steps:

1. Clear the Scenario Measurements.Files list and add the (*.e) file generated above to the list.
2. Create a satellite object at the scenario level (not under the GNSS Constellation). Set the Tracking ID for this satellite to match what was specified in the name of the (*.e) file.
3. Add a Tracking Instrument to the Satellite object. The Tracking ID of the Tracking Instrument should be set to be the same as Tracking ID on the Satellite.
4. Make sure the Tracking Instrument measurement statistics include "Eph Pos" and "Eph Vel" statistics.
5. Use the InitialStateTool.htm utility (from LaunchPad) to set the initial orbit position and velocity, using the corresponding *.e file as a source.
6. Force acceptance of all tracking data (set satellite MeasurementProcessing.ResidualEditing.NominalSigma = 1000).
7. Then run the filter over the entire data span and graph or report SRP and YBias estimates. The converged values add to the default values that are set in the Object Properties window.

The variability in the converged solution provides an indication of the settings for process noise for these parameters (sigma and half-life). Typically, every GNSS SV exhibits a periodic solution for Scale and YBias, with a period of one revolution. The amplitude of variations in YBias differs significantly between satellites. This indicates that you can make some satellite-dependent modeling improvements to the solar pressure models, if necessary.

NOTE:

- If a satellite is in eclipse season, this process can lead to some very erratic coefficient solutions. Initial values are most easily derived when a satellite is not in its eclipse season. If there is any doubt about the behavior, use STK to check for eclipsing by the Earth and by the Moon.

7 SV clock state selection and tuning

Clock process noise statistics A_0 and A_{-2} represent white noise of frequency and random walk of frequency over a time interval of interest. The values are defined by the random behavior of clock phase after the systematic trends due to phase bias, frequency bias, and aging bias are removed. You can obtain clock process noise statistics from a few internet sources or through analysis of live data. The following discussions address both methods.

7.1 Determine Satellite Clock Process Noise Statistics

There are reports on the internet that present some historical performance information for SV clocks. Most of these reports are from various PTTI meetings and are on http://tycho.usno.navy.mil/**ptti**. Look for Allan Deviation statistics, usually in graphical form, providing $\sigma_y(\tau)$ at various times. Typically, AGI computes clock statistics using $\tau = 1$ day.

PTTI values are not current, but should be representative of SV clocks. You can use the values to compute clock process noise statistics A_0 and A_{-2} using:

$$A_0 = \tau\sigma_y^2(\tau)$$
$$A_{-2} = \frac{\sigma_y^2(\tau)}{\tau}$$

There are two steps to determining good clock process noise statistics. The first step is an approximation that you can derive from the SP3 files. This is only an “approximation” due to smoothing inherent in SP3 generation. The second step is to process MS data with the filter and to recompute refined process noise statistics from the filter estimates.

Given the phase history in an SP3 file, you must first remove the systematic drift and aging effects. A simple quadratic fit over a month of phase histories should remove the systematic effects. AGI recommends a long fit to eliminate the short-term periodic behavior of some clocks. The phase residuals for such a fit characterize the random clock phase behavior approximately. For example, you can use the phase residuals nominally reported in the SP3 on 15-minute epochs to compute an Allan Variance at 60 minutes as follows:

1. Define the phase residual at each 15-minute epoch as $\Delta\phi(t_k)$.

2. Assume that τ is a time of interest for Allan Variance calculation, for this example 60 minutes (a multiple of 15 minutes).

3. The sample random change in phase over a specific τ = 60 minute interval (t_i,t_{i+1}) due to stochastic behavior would be the sum of all clock phase residuals over that 60-minute interval.

$$\Delta y_i = \sum_{k>1}^{i+1} \Delta\phi(t_k)$$

4. The time average over a single 60-minute interval is

$$\Delta\bar{y}_i(\tau) = \frac{1}{\tau}\sum_{k>1}^{i+1} \Delta\phi(t_k)$$

5. You can construct the next sample interval by sliding the start time and stop time by 15 minutes.

6. The Allan Variance for the interval τ = 60 minutes is then the average (< >) change over all pairs of successive 60-minute intervals, defined as:

$$\sigma_y^2 = \frac{1}{2} < \Delta\bar{y}_{i+1} - \Delta\bar{y}_i >$$

You can then derive the clock noise statistics for the interval τ = 60 minutes by using

$$A_0 = \tau\sigma_y^2(\tau)$$
$$A_{-2} = \frac{\sigma_y^2(\tau)}{\tau}$$

You can compute the Allan Variance for different multiples of 15 minutes based on the SP3 clock phase estimates, but you must extrapolate for other times.

Once you compute the approximate values for A_0 and A_{-2}, you can use the filter to process MS data and generate refined values from the filter residuals. The method of calculation is similar to that discussed in Section 8.2 below. AGI recommends setting the MS clock statistics first before running the full filter to refine SV clock process noise values.

NOTE:

- The filter residuals will be a function of both the MS clock and the SV clock behavior, which will cloud the statistics somewhat. Good separation of clock effects is necessary in order to set SV clock process noise statistics correctly. For this reason, AGI recommends that you conduct the analysis over many days and if possible over a month. Then you can discard residuals over the filter convergence period.

7.2 Estimate GPS satellite clock aging

Clock aging is synonymous with frequency drift, which causes a systematic change in clock phase that is generally quadratic. Some GPS SV clocks exhibit significant aging. You can determine aging values from publicly available SP3 files by taking the second difference of the reported phase values over a long time. For example, using NGA SP3 files for May 2006, you can determine that the aging estimate for PRN30 was approximately -0.7 ns/day^2.

It is possible and advisable to estimate the aging values for SV clocks using ODTK, but if the tracking data are some version of pseudorange, it will take a long time for the filter to converge to the correct value. You can achieve better accuracy immediately after initialization if you seed the filter with a starting estimate derived from publicly available data. In this case, AGI recommends determining the Aging values for the month immediately preceding the filter epoch. For those clocks with negligible Aging trends, you can estimate them without Aging states, but clocks exhibiting significant aging require continual re-estimation.

There are three settings in the Satellite object related to satellite clock aging: AgingBias, AgingWN, and AgingSigma. If you set AgingBias to a nonzero value, then ODTK uses it in the clock prediction model, regardless of whether or not AgingBias is a state in the filter state. Adding aging bias to filter state space requires nonzero entries in both AgingWN and AgingSigma.

Clock process noise is very sensitive to aging process noise, and you should use very small numbers whether simulating clocks or processing live clock data. A useful value to remember for AgingWN is 1e-56 sec^{-3}. This is the value for Aging process noise, where aging contributes negligibly to clock phase sigma growth. Larger values quickly cause substantial growth in the clock phase error covariance.

NOTES:

- There are rules of thumb that AGI has collected from various clock experts during the development of the clock estimation capability. These include the following:
    - Cesium clocks do not exhibit aging behavior.
    - Rubidium clocks exhibit aging behavior.
- These rules-of-thumb are not always consistent with the empirical evidence. The Block IIR rubidium clocks should not exhibit aging behavior, but if you compute second differences on NGA SP3 ephemerides across months at a time, this appears to be questionable.

8 Monitor station initial values and tuning

You can generate initial values for MS clocks iteratively by using the current SP3 file as a reference, filtering pseudorange data from these monitor stations, and solving for clock states. Then you can use statistical analysis on the resultant clock estimates to set clock parameters in ODTK.

8.1 Establish initial phase and frequency estimates

Set all SVEstimatedStates to “false” in the GNSS Constellation object, and filter the tracking data for a short period of time. MS Clock phase estimates will converge very rapidly to the proper values, while frequency takes a little longer to converge. Running the filter followed by the smoother will provide the best estimates of the clock phase and clock frequency at the starting epoch for the scenario. You can then copy the smooth estimates for phase and frequency into the initial conditions for MS clocks.

If the filter clock covariance does not stabilize during the interval selected, then choose a longer interval.

8.2 MS clock data availability on the internet

There are reports on the internet that present some historical performance information for AF and NGA MS clocks. Most of these reports are from various PTTI meetings and are on tycho.usno.navy.mil/ptti**.** Most of the PTTI reports focus on the performance of satellite clocks, but occasionally a report will specify performance statistics for the monitor stations. For example, "Performance of GPS On-Orbit NAVSTAR Frequency Standards and Monitor Station Time References" by Oaks, Reid, and Buisson in 1998 provides the following detail:

> "The performance of the ground reference clock at the Colorado Springs Monitor Station is superior. It is the Alternate Master Clock #1 which is a hydrogen maser steered to UTC (USNO) by two-way satellite transfer …The performance of the ground reference clocks at the remaining four Air Force stations, which are equipped with HP5061 cesium beam tubes, has more noise than that of the ground reference clocks at the NIMA MS, which are equipped with HP5071 high performance cesium-beam tubes."

This reference provides graphs that show a distinct grouping of AF MS clocks, NGA MS clocks, and the Colorado Springs hydrogen maser as a unique high-quality clock. The graphs are of very poor quality and cannot be reproduced here, but are sufficient to discern that the Allan Deviation at one day for AF MS was approximately 4.2 parts per 10^{14}, while NGA MS value at one day is 3 parts per 10^{14}, and finally the hydrogen maser was much better at 1.5 parts per 10^{14}.

These values are not current, but should be representative of these clocks. You can use the values to compute clock process noise statistics A_0 and A_{-2} using:

$$A_0 = \tau\sigma_y^2(\tau)$$
$$A_{-2} = \frac{\sigma_y^2(\tau)}{\tau}$$

8.3 Determine MS clock process noise statistics

The following discussion assumes that there are no clock events during the analysis span. The presence of clock events will require additional logic.

The task of calibrating A_0 and A_{-2} becomes the task of estimating phase and frequency over a long period of time and performing a statistical analysis on the clock residuals after the prediction is removed. This is approximately the same as performing a statistical analysis on the cumulative state corrections to phase over a time interval of interest.

As in Section 8.1 above, use the SP3 as a reference for both orbit and clock and solve for MS clock states. Run the filter for many days at a convenient measurement update rate. Generate an Export report for each MS in the solution, giving the clock estimates for phase and drift, and save as a CSV file. It will be convenient for subsequent analysis to provide time in MJD and seconds as an output. You can write a simple script to collect the cumulative change in clock phase over an interval of interest, for example, over one (1) day. You can then use the set of all cumulative phase changes to define an Allan Variance. Finally, you can use the Allan Variance to derive clock noise statistics (A_0 and A_{-2}) for the interval of interest. An example may clarify this explanation:

1. Assume that the measurement update rate for all MS is 1.5 seconds.
2. Define the phase correction at each 1.5 second epoch to be $\Delta\varphi(t_k)$.
3. Assume that τ is a time of interest for Allan Variance calculation, for example 15 minutes.
4. The sample random change in phase over a specific τ = 15 minute interval (t_i,t_{i+1}) due to stochastic behavior would be the sum of all clock phase corrections over that 15 minute interval.

$$\Delta y_i = \sum_{k>1}^{i+1} \Delta\phi(t_k)$$

5. The time average over a single interval is.

$$\Delta\bar{y}_i(\tau) = \frac{1}{\tau}\sum_{k>1}^{i+1} \Delta\phi(t_k)$$

6. You can construct the next sample interval by sliding the start time and stop time by 1.5 seconds. Does this imply that the samples are statistically correlated?
7. The Allan Variance for the interval τ = 15 minutes is then the average (< >) change over all pairs of successive 15 minute intervals, defined as:

$$\sigma_y^2 = \frac{1}{2} < \Delta\bar{y}_{i+1} - \Delta\bar{y}_i >$$

You can then derive the clock noise statistics for the interval τ = 15 minutes by

$$A_0 = \tau\sigma_y^2(\tau)$$
$$A_{-2} = \frac{\sigma_y^2(\tau)}{\tau}$$

NOTES:

- The definition of Allan Variance (above) involves two consecutive intervals of length = τ. If the time span of interest is one (1) day, a filter run of two days will have only one sample pair and will be statistically uncertain, while a filter run of three days at a 1.5 second measurement rate will yield thousands of pairs, which may be more significant.
- Clock process noise statistics perform two functions: one to constrain the filter state correction process, and the other to model the probable change in the measurement due to clock behavior. Optimal settings are not necessarily the same for each function, simply because the spectrum of clock behavior is not globally characterized by two pa-

rameters. If the mission objective of the filter is to provide long term stability to the orbit and clock predictions, then the process noise would be set by choosing A_0 and A_{-2} values to reflect long term clock behavior using the Allan Variance at $\tau = 1$ to 10 days. On the other extreme, a requirement for a local best fit and short-term prediction accuracy might require setting A_0 and A_{-2} at $\tau = 1$ to 10 hours. Optimal selection of clock process noise interval τ may require a parametric analysis versus mission requirements.

- Some analysts prefer to use a Hadamard Variance. Extracting white noise of frequency and random walk of frequency statistics from a Hadamard Variance is left to the user.

8.4 Choose a tropospheric model and set parameters

If meteorological data ("met data") are available from the MS, then AGI advises using the Saastamoinen tropospheric correction model and including the optional state parameter to estimate bias. This will improve the initial clock estimates by removing a small error. If local met data are not available, then there is still value in using the SCF model, but only if you use EstimateBias. Using the model is better than ignoring the troposphere. If met data are not reported reliably, then you cannot use them reliably; ODTK does not have a mechanism to switch between various model options when a gap in met data appears.

9 Clock Management

9.1 Clock Ensemble and Ensemble Steering

9.1.1 Background

Classically, you can make any closed ensemble of clocks to be self-consistent, such as everyone in a room synchronizing their wristwatches. However, the "time" that they agree on may be different from some time standard (UTC, TAI, GPS, etc.). Drift rates depend on the stability of the clocks and the whole ensemble can drift with respect to the standard. Without some outside reference, the ensemble can drift to very large offsets and the holders of the watches would be oblivious to the error. When you compare the ensemble to a standard, you can determine and minimize the error.

In a Kalman Filter simultaneously estimating an ensemble of clocks, the previous discussion would be characterized by high cross-correlations in the covariance between all of the clocks and a continuously growing covariance on each clock component (e.g., the relative error is small and the absolute error is unobservable).

The GNSS systems have this problem. As a closed system of clocks, any GNSS Time system can drift with respect to UTC while it is supposed to be running in parallel to UTC (offset by an integral number of seconds). The formal requirement levied on a GNSS system is to maintain time compatibility to within 0.1 μs; however, this is actually unacceptable for certain GNSS applications and therefore requires that an ensemble of GNSS clocks be *steered* to maintain an integer offset to UTC at the few-nanoseconds level.

The GNSS clock steering problem is further complicated because the clocks in a GNSS constellation have different stability characteristics, and some have serious issues that make them poor candidates for determining the ensemble mean, and therefore GNSS Time. For example, in May 2006, PRN01 exhibited large periodic variation in phase and frequency while PRN25 randomly changed clock frequency a few times.

9.1.2 Process controls

Consider the GPS constellation. The GPS MCS uses two algorithms to define and steer GPS Time to be parallel to UTC. First it defines a Composite Ensemble, composed of the most reliable clocks. Then, it applies a Clock Steering Algorithm, a damped bang-bang controller, to steer the Composite Ensemble to be parallel with UTC. AGI has implemented these algorithms in ODTK and provides the control parameters for these algorithms under the GNSS Constellation object.

The attributes under SVEstimatedStates, available when the GNSS Constellation is populated with GNSS satellites, enable selection of a Composite Clock (ClockEstimationMode) and selection of clocks to make up the Composite Clock (ReferenceClocks). For highly stable MS clocks, it is recommended that MS clocks be included in the Composite Clock. Any highly stable SV clocks should also be in the ensemble. The number of clocks in the Composite Clock is not restricted; however too few clocks make GPS Time vulnerable to a single clock event. Nominally, 18-20 of the best clocks in the system should be used to define the Composite Clock. The value for ClockCovReductionSigma specifies the sigma on a pseudophase measurement applied to the ReferenceClocks whenever there is a time update.

Clock Steering is controlled under GPSUTCTimeSteering. The InitialBias and InitialDrift (in sec/sec) parameters tell the damped bang-bang controller the size of the error that has to be removed by steering, while MaxSteeringRate provides the damping, limiting the magnitude of the correction, and ClockCovReductionUpdateInterval specifies the minimum frequency for applying steering in the absence of measurements.

NOTES:

- It is impossible to set the InitialDrift without running the filter for several days and comparing GNSS Time to the filter ensemble mean.
- The GPS MCS documentation recommends ClockCovReductionSigma = 31.8 ns and MaxSteeringRate = 1e-19 sec^{-1}. These values reflect considerable experimentation using a steady state Kalman Filter, the current AF monitor station configuration, the current Kalman Filter, and the corresponding one-smoothed observation per minute update rate. They also reflect an operational desire to make any changes to GPS Time transparent to the users. This does not necessarily apply to a brand-new initialization from scratch and data rates up to and including 1.5 seconds. In this case, AGI has used MaxSteeringRate = 1e-17 sec^{-1} successfully on a few test cases.

9.2 Recover from clock events

Clock events take many forms. NGA monitor stations will reset their clocks, creating step functions in the phase correction at the millisecond level. The NGA clock resets appear to be deliberate, occurring when GPS Time according to the receiver has drifted beyond some limit from GPS Time according to the constellation. You can swap out satellite clocks, which creates instantaneous offsets in phase, frequency, and drift. Anomalous clock events, such as random frequency changes by PRN 25, are particularly nefarious.

Clock events are detected in the pseudorange residuals, and that detection also provides the data that enables ODTK to recover from such an event. A millisecond step function in clock phase will appear as a pseudorange residual of ≈300 km. A change in clock frequency will result in an increasing error in range residuals.

ODTK treats clock events similarly to impulsive maneuvers. The user enters an approximation for the magnitude of the event (gleaned from residuals) and an uncertainty. The user can control phase, frequency, and aging separately. The mechanism for entering clock events is in the attributes for each GNSS satellite and in each GNSS receiver. For the satellite, enter events under ClockControls.ClockResets. For MS receiver clock events, use Clock.ClockResets.

NOTES:

- The PhaseChange attribute will add the quantity provided to the state estimate at the ResetTime specified.
- The PhaseDeweightingSigma specifies the additive process noise for clock phase to be applied instantaneously at the ResetTime
- Similar controls are provided for clock frequency and aging.

You can use pseudorange residuals to define the magnitude and sign of the Reset. The sign is determined by recognizing that ODTK computes the residual as (Time of Receipt) minus (Time of Transmission), so that a positive residual can be the result of a positive time step at the receiver or a negative time step at the transmitter (GPS SV). The sign of the slope can be similarly interpreted.

NOTES:

- After some sensitivity analysis, if you can approximate PhaseChange to within ~50 ns and PhaseDeweightingSigma is no larger than ~100 ns, then the clock reset appears to be reliable. Attempts to push PhaseDeweightingSigma to 10 or 100 μs resulted in poor performance.
- Even though you can reset phase without resetting frequency, it is prudent to provide some small deweighting value for FreqDeweightingSigma. AGI has had good results with 1 μs/day for MS clock Resets.

- When the true event time is not easily determined, it is better to choose an estimated event time after the true event time than to choose one before the true event time. This is because the filter will converge to a new clock estimate based on very few measurements after the estimated event time, and if those measurements reflect the clock behavior before the event, then the filter will not model the event properly.

10 Manipulate state space

State space partitioning was included in the original Kalman Filter for the GNSS operational control station and served two purposes:

- manage the small memory found in older mainframe computers
- separate out "sick" satellites, so that they could benefit from the full solution derived from "healthy" satellites without disturbing that solution

ODTK processes all GNSS satellite states and all MS states in a single *partition*, taking advantage of modern memory sizes. However there still are requirements to add and drop satellites and to partition off sick satellites in support of normal operations. ODTK has options to support these requirements.

10.1 A separate partition for "sick" satellites

Each GNSS satellite object has an attribute called PartitionControls.SeparatePartition. When you set this to true, ODTK isolates the satellite from the other satellites in the following sense:

- The orbit and clock for the partitioned satellite remain in the filter state space.

- The orbit solution for the partitioned satellite uses the full state solution from the main partition in calculating the state correction.

- Through manipulation of measurement partials, the partitioned satellite makes no contribution to corrections to states in the main partition.

- Each partitioned satellite is in a partition by itself. If two or more satellites are partitioned, they are each in a separate partition.

10.2 Add new GNSS satellites to the scenario

The capability exists to simply add a GNSS satellite to the scenario at a restart time. The simplest method may be to copy an existing GNSS Satellite object, highlight the GNSS Constellation object, and then paste the copy into the constellation. Then you have to set all of the satellite-specific parameters. In particular, the satellite OrbitState.Epoch should be exactly the same as that of the remaining satellites.

AGI recommends that you process new satellites as partitioned satellites until the state estimates stabilize.

10.3 Remove GNSS satellites from the scenario

The capability exists to simply delete a GNSS satellite from the scenario at a restart time; however, the next filter restart will probably complain that the covariance has negative eigenvalues (ODTK performs a check on restart). This is because the clock solutions become highly cross-correlated between all clocks (recall Section 9.1.1), and deleting any state while it is highly correlated to other states can drive a covariance negative.

AGI recommends that deleted satellites remain in the filter state space until the cross-correlations to other satellites become negligible. You can accomplish this by declaring the satellite a partitioned satellite. The covariance for all parameters will grow in the absence of tracking data to process, and eventually they will become uncorrelated from other states.

NOTES:

- To examine cross-correlations to other states, remember to enable the Filter attribute Output.DataArchive.SaveCrossCorrelations. You can generate cross-correlations to other states manually with ODTK reports and graphs. You can check cross-correlation automatically by examining the filrun file. Use the StateFileDumperTool.htm on LaunchPad or in a script use the DumpFile functional attribute and parse the output for the desired cross-correlations.
- If you partition a satellite, then do not include it in the Composite Clock (should not be a Reference Clock).

10.4 Change SV PRN assignments

You can change SV PRN assignments through the process of adding and deleting satellites. This approach assumes that the PRN slot will be vacant for some time before reassigning it. During this time, you could partition off a PRN and it would become decorrelated. Then you could delete it, all before assigning a new satellite the same PRN.

10.5 Replace clocks

GNSS systems will swap out SV clocks on short notice. When that happens, there will be insufficient time to use the Partition approach in replacing a satellite. There are two possible ways to handle replacement clocks in the filter:

- If there is a long interval with the clock inoperable, then the Add and Drop method of handling satellite replacement will suffice.

- If the clock replacement takes place over a few hours, then the clock replacement is functionally similar to a clock event, where phase, frequency, and aging will all change at an epoch. See Section 9.2 for methods of handling clock events.

11 Measurement Models

ODTK supports many of the GNSS measurement types. These include single- and dual-frequency pseudorange and phase measurement processing, as well as single-differencing and double-differencing of pseudorange and phase.

ODTK does not support Phase-as-Range, the use of an “unknown” range bias to treat phase cycle counts as range. Phase tracking data are treated as integrated Doppler.

The ODTK Simulator generates the basic single-frequency measurements, for example P1, P2, and CA pseudorange, and L1, L2, and LA phase, as they would be collected by a GNSS receiver. ODTK does not simulate the dual-frequency corrected nor differenced measurement types directly since they would not be directly output by a GNSS receiver.

The filter constructs dual-frequency (DF) ionosphere free measurement combinations and single-differenced (SD) measurements at the time it reads the tracking data, provided that you request DF, SD, or both measurement types.

You can generate doubly differenced tracking data using the utility GenerateGPSDDObs.htm, which creates a *.gobs file containing the double differences. The double-differencing algorithm establishes independent pairs of dual-frequency corrected data, with the consequence that any odd SV or MS may be dropped. This algorithm contrasts with double-differencing algorithms used elsewhere that form all permutations and combinations of differences. The ODTK approach avoids the issue of high correlations between differenced observations.

12 Manipulate SP3s to support analyses

ODTK can produce an SP3 as a standard interface to other agencies and can convert an externally defined SP3 to an internal ODTK runfile, which then enables you to perform accuracy comparisons using ODTK reports and graphs.

12.1 Concatenate SP3s

A concatenated SP3 file provides a “truth” against which you can measure ODTK performance. You can also use a concatenated SP3 as a reference ephemeris for the entire GNSS constellation. This enables estimation of MS and SV clock biases and derivation of clock process noise.

ODTK provides a utility called ConcatenateSp3Files.htm (in LaunchPad) to concatenate SP3 files. This utility enables you to select two or more SP3 files. It will then order the SP3 files before concatenating, and it will create one long SP3 file from the assembly of inputs.

NOTES:

- The utility assumes that the reporting grids in all SP3s are the same, and that points are not duplicated between successive files. The utility will not interpolate if data are missing or are of differing densities.
- The legacy utility SP3_To_Ephem.htm generates an STK *.e file for each GNSS SV. By combining this utility with ConcatenateSP3Files.htm, you can generate very long *.e files for GNSS SVs.

12.2 Create a Simrun file from SP3

You can convert an SP3 file to a *.simrun file. This conversion enables you to difference and compare ODTK-created filter and smoother runfiles with the SP3 data, and it enables you to display results with ODTK reports and graphs.

The utility that does this conversion is called SP3_To_Simrun.htm.

12.3 Create an SP3 file from ODTK

You can use any ODTK runfile (filrun, smtrun, or simrun) to create an SP3 file. Use the utility StateFile_To_SP3.htm to dump all data points or extract values on a fixed grid. This tool does not interpolate, so the desired grid must be explicitly present in the runfile. You can enforce a specific grid in the filter and smoother by setting Filter.ProcessControl.ProcessNoiseUpdateInterval to the desired time interval. Similarly, in the simulator, use the Simulator.ProcessControl.TimeStep. You can also set these controls to integer divisors of the desired time step; for example, 1.5 seconds will allow for an SP3 on a 15-minute grid.

13 Graphs and reports

Graphs and reports are available for Satellite clocks, GNSS solar pressure coefficients, and tropospheric scale parameter estimation. These should be sufficient for getting started in GNSS orbit and clock estimation.

The reported times on graphs and reports will respond automatically to the time selection at the scenario level, under the attribute Units.DateFormat. The choices are either GPSG or UTCG. If you do not load a scenario, then Tools → Options → Units controls the default.

One specialized style is “Differenced Orbit and Clock in URE.” This style uses orbit and clock differences between two runfiles and presents the results in a single statistic called URE. The equation given in the ODTK Help system documentation is also in the Air Force document “Standard GPS Performance Metrics”, dated 21 Feb 2006, where it is called “ERD”. If the runfiles are a filrun generated by the ODTK filter and a simrun created from an SP3 file, then ERD or URE is a measure of the ODTK performance.

You will need customized graphs and reports to support any long-term application of ODTK to the GNSS mission.

14 Processor loading and disk utilization

ODTK provides “Scalar” and “Simultaneous” modes of measurement processing in the filter. In “Scalar” mode, N simultaneously recorded measurements require N separate (Scalar) measurement updates. For GPS orbit and clock estimation, with all MSs reporting on fixed GPS Z-count epochs, “Simultaneous” mode is preferred due to the computational burden imposed by the “Scalar” mode. The control for this feature is in the filter properties, under ProcessControl.MeasurementProcessingMode.

Long term GNSS analyses can require terabytes of disk space. For example, AGI ran a single filter-smoother analysis across many days for the entire constellation and 11 MSs (six AF and five NGA MSs) with all options turned on, processing all tracking data at the full reporting rate. The *.rough file reached 500 GB after 20 days. Likewise, runfiles can exceed 250 GB for a 20-day run.

You can set the filter flag Output.DataArchive.SaveCrossCorrelations to false to eliminate all SV-to-SV and MS-to-MS cross-correlations and thereby reduce runfile storage by an order of magnitude. Cross-correlations within a set of SV or MS states are still preserved when you set the SaveCrossCorrelations attribute to false.

There is no corresponding compression option for the *.rough file short of thinning tracking data.

15 Computer selection considerations

Several factors may influence selection of a computer for running ODTK:

- The size of the L2 cache can be more important than processor clock speed when seeking better throughput for very large scenarios.
- Large-scale GNSS OD can use terabytes of disk space. AGI has found it useful to have several large (500GB or larger) drives and to direct different outputs to different drives. AGI has not experimented with throughput for a RAID disk array.

16 Notes for new users

While the following tips are obvious to the experienced orbit analyst, they bear repeating for the new user:

- There are literally hundreds of parameters to set in ODTK for the GNSS orbit and clock estimation problem, and plenty of opportunities for typos. AGI recommends using scriptlets to set values. You can save scriptlets and number versions to enable recovery of intermediate analyses. You can use the ScriptingTool.htm for this purpose.

- Use short names for ODTK objects like scenarios, tracking systems, satellites, monitor stations, and receivers, otherwise the long names will overflow their fields in reports, and graph legends will become so large that you cannot view legend and graph simultaneously.

- Thinning data enables preliminary “tuning” analyses to proceed more quickly. The option to thin data is at the filter level. You can execute this by adding an entry to the CustomDataEditing Schedule with the Action set to “Thin”.

17 An example scenario

AGI has provided a sample scenario for May 1, 2006. The settings for orbits, clocks, antenna offsets, and solar pressure coefficients are approximately correct, as derived from NGA SP3 and AF and NGA MS data. Satellite masses are strictly nominal. The MS station numbering and naming conventions follow two SOPS conventions. AGI does not provide the MS data since it is the property of the U.S. Air Force. As an alternative, AGI provides a Simulator object, which will enable you to exercise the scenario. You can find the scenario in:

<install directory>\ODTK\UserData\DemoScenarios\Global Positioning System Operations

The scenario is set to simulate ten minutes of tracking data. There is no computational reason for such a short scenario, but it ensures that the scenario would not use too much disk space if exercised as is.