ORBIT DETERMINATION USING ODTK VERSION 6

EUROPEAN SPACE ASTRONOMY CENTRE (ESA/ESAC), MADRID, SPAIN
3 – 6 MAY 2010

David A. Vallado [1], Richard S. Hujsak [2], Thomas M. Johnson [3], John H. Seago [4], James W. Woodburn [5]

[1] Center for Space Standards and Innovation, 7150 Campus Dr., Suite 260, Colorado Springs, CO, USA, 80920. Email dvallado@centerforspace.com
[2] Analytical Graphics Inc., 220 Valley Creek Blvd., Exton, PA, 19341. Email rhujsak@agi.com
[3] Analytical Graphics Inc., 220 Valley Creek Blvd., Exton, PA, 19341. Email tj@agi.com
[4] Analytical Graphics Inc., 220 Valley Creek Blvd., Exton, PA, 19341. Email jseago@agi.com
[5] Analytical Graphics Inc., 220 Valley Creek Blvd., Exton, PA, 19341. Email jwoodburn@agi.com

ABSTRACT

The Orbit Determination Tool Kit (ODTK) is a commercial software product from Analytical Graphics, Inc., that performs orbit determination and analysis. Its key features include a tracking-data simulator, an extended (Kalman-like) sequential filter, a fixed-interval smoother and a variable lag smoother. There are a variety of custom tools to aid the researcher and analyst, including estimation of multiple satellite parameters and time-varying measurement biases, autonomous measurement editing, and data reporting and graphing. ODTK Version 6 builds upon previous successful versions and introduces significant new capabilities.

1. INTRODUCTION

Analytical Graphics, Inc., (AGI) has completed a significant upgrade of ODTK, a commercial software product for spacecraft orbit determination and analysis. ODTK is based on a well established algorithm for sequential orbit determination first used by the General Electric Company for satellite operations [1]. This algorithm has been adapted into other software packages by various companies and agencies in support of operational satellite programs [2]. AGI also concluded that this approach would be best for its own orbit-determination software, and ODTK was introduced as the commercial-off-the-shelf product STK/OD in October, 2003. The latest release (ODTK 6) continues to advance the product's capabilities for a world-wide customer base that include aerospace businesses, international governments, and academic institutions.

ODTK includes a simulator, a Kalman-like sequential filter, a fixed-interval smoother and a variable-lag smoother [3]. ODTK also provides Initial Orbit Determination (IOD) capability, and a conventional weighted Least-Squares (LS) estimator. ODTK operation can be automated with scripts and custom "plug-in" models, and reports and graphs can be customized for tailored results. From satellite tracking data, ODTK's filter/smoother can estimate the orbits of multiple satellites simultaneously, as well as their observable satellite parameters such as ballistic coefficients, solar-radiation-pressure coefficients, *etc.* It can also simultaneously solve for biases in ground-based and/or space-based tracking, time-varying transponder biases, and atmospheric-density corrections. A significant feature of the filter is its ability to process through known maneuvers, and to include planned maneuvers (and their uncertainty) in propagation.

ODTK 6 is compatible with the Consultative Committee on Space Data Systems (CCSDS) standards. TDM messages can now be read and OEM ephemerides are output.

2. SYSTEM REQUIREMENTS AND OPERATION

An ODTK 6 installation requires about 250 MB of disk space. Running a typical scenario takes slightly over 110 Mb of memory and about 600 MB disk space for simulation, filter and smoother runs, allowing ODTK 6 to run on smaller, inexpensive platforms. ODTK 6 runs under Windows XP, Vista, and 7 and is about 20% faster than previous versions. It now supports multiple CPU processors or cores and automatically uses additional CPUs if available.

ODTK's graphical user interface (GUI) has an on-screen *object manager* to invoke software options. The user controls object options and saves the settings to a *scenario* file. ODTK also provides a ***message viewer*** window that displays errors, warnings, and informational messages. Software documentation is provided by an on-line context-sensitive help system. Optional ***event-control features*** execute user-defined scripts following filter and/or simulator "events" such as internal errors (*e.g.*, when a satellite has reached re-entry altitude), starting, stopping, resuming, interrupted halting, and ending of measurements. A complete application programming interface (API) is provided.

ODTK also provides a variety of reporting styles with customizable features. The ***Static Product Builder*** is an interface for creating, editing, and displaying reports and graphs, or exporting data and images to other standard software formats. Static graphs are re-scalable and key events such as maneuvers may be displayed directly on the graph. The ***Dynamic Product Selector*** displays reports and graphs that continuously update during the filtering process so that performance can be monitored in real time. A full restart capability is very useful when the analyst seeks to change items part way through a complex simulation or for continuation of operational orbit estimates.

3. TECHNICAL BACKGROUND AND NOTES

A detailed mathematical specification exists with the product [4], and there are also numerous white papers available via the Internet [5].

ODTK models stochastic process noise for state, acceleration, and measurement errors, where process noise is a mathematical model of errors in the system dynamics. ***Dynamic noise compensation*** models unknown acceleration errors by assuming they can be characterized by Langevin's first-order linear stochastic differential equation [6]. The general solution to the Langevin equation is a Gauss-Markov process attributed to Uhlenbeck and Ornstein [7]. The stochastic process-noise model primarily adopted in ODTK is a stationary, two-parameter sequence, shown in scalar form as Eq 1.

$$\begin{aligned} X(t_{k+1}) &= \Phi(t_{k+1},t_k)X(t_k) + \sqrt{1-\Phi^2(t_{k+1},t_k)}\,w(t_{k+1}) \\ \Phi(t_{k+1},t_k) &= e^{\alpha|t_{k+1}-t_k|} \\ w &\sim N(0,\sigma_w^2) \end{aligned} \tag{1}$$

where σ_w is the square-root of the variance of the independent (white) Gaussian noise, and α is a constant < 0 determining the degree of process autocorrelation. This model is also Gauss-Markov, and related to the Ornstein-Uhlenbeck sequence in that both demonstrate an exponentially fading autocorrelation function. In practice, the user defines α through the exponential half-life $\tau_{½} = \ln(½) / \alpha$. The half-life regulates how quickly a bias estimate can change; for a time interval equal to the half-life, a Gauss Markov bias will decay by a factor of two during propagation in the absence of measurements.

A complete stochastic bias representation in ODTK is given by,

$$b = b_C + b_{GM} \ , \tag{2}$$

where b_C is a user-specified static ***bias*** (a constant offset between a predicted value and an observed value) which is not estimated, and where b_{GM} is an estimated time-varying Gauss-Markov offset that behaves according to Eq. (1). Computing the expectation, $E(X(t_{k+1})X(t_k))$, reveals that σ^2_w is the steady state variance of b_{GM} (in the absence of measurements). For b_{GM}, the user must specify the ***bias sigma*** (σ_w) and the so-called ***bias half-life*** ($\tau_{½}$ for b_{GM}). A ***white noise sigma*** is also required to describe the random measurement noise.

4. DATA INPUT

4.1. Facilities and Sensing Trackers

Measurements may be simulated or processed from a sensor. The sensors can be either a ground-based facility or space-based. Additional information is available online in the Help section under "Measurement Types".

Ground-based facility locations are specified in the Earth-fixed frame in a variety of coordinate variables. Precise station-motion models are implemented according to IERS Conventions [8], including plate-tectonic motion, vertical motion due to Earth tides and coastal ocean-tide loading, and rotational deformation (pole tide). Facility location can be estimated by the filter/smoother, and a randomly deviated location can be simulated. Either three components of position or horizontal location can be estimated. GPS receivers, emitters, and/or transponders can be attached to a facility. The ***GPS-receiver*** object allows for simulation and/or processing of GPS measurements. The *emitter* object represents a source beacon from which one can simulate and/or process time difference-of-arrival measurements. The ***transponder*** object allows for simulation and/or processing of 2-way measurements and multi-legged measurements involving relays.

In ODTK, a satellite orbit can be defined from initial conditions with an associated force model, or it can be imported as a tabular ephemeris. The *a priori* orbit uncertainty matrix is input by the user and defines the satellite's initial orbit covariance for the filter and the random distribution from which the simulator will draw to randomly deviate the initial orbit state. The matrix elements represent an error-covariance in radial, in-track, and cross-track, (RIC) position and velocity. The orbit uncertainty matrix can also specify the uncertainty of a satellite orbit defined by an external reference ephemeris; in this case, the orbital uncertainty is assumed constant in the RIC frame. Alternatively, the uncertainty of a satellite orbit defined by an external reference ephemeris can be provided as part of the reference ephemeris.

Satellite attitude may be defined based on a set of geometric profiles or via of tabulated attitude data. The satellite attitude provides the rotation between the inertial frame and the body frame, and is used to compute the inertial location of antenna phase centers which are defined in body coordinates relative to the center of mass of the satellite.

4.2. Measurement Models

ODTK 6 supports numerous measurement types, shown in Table 1 and Table 2. In the interest of brevity the following acronyms are defined: Deep Space Network (DSN), Total Count Phase (TCP), Satellite Laser Ranging (SLR), Time Difference of Arrival (TDOA), Frequency Difference of Arrival (FDOA), time derivative of TDOA (TDOA dot), single differencing (SD), double differencing (DD), Tracking Data and Relay System (TDRS), Bilateral Ranging Transponder System (BRTS), and Dual Frequency (DF).

Table 1. Ground Based Measurements

Traditional	Azimuth / Elevation X / Y Right Ascension / Declination Bistatic range 2-way range Direction cosines
DSN	3-way Doppler 3-way TCP Doppler TCP Sequential range
SLR	Normal point range
Geolocation	TDOA FDOA TDOA dot SD TDOA SD FDOA Ground TDOA Ground FDOA Ground TDOA dot
TDRS	BRTS Range BRTS Doppler
GPS	2-leg CA pseudorange Pseudorange (CA, L1, L2), SD and DD Phase (CA, L1, L2, LA), SD and DD CA and DF navigation solution (X, Y, Z)

Table2. Space Based Measurements

GPS	Pseudorange (CA, L1, L2), SD and DD Phase (CA, L1, L2, LA), SD and DD CA and DF navigation solution (X, Y, Z)
TDRS	4-way range, 5-way doppler 3-way return-link doppler
Space to Space	Range Azimuth / Elevation Right Ascension / Declination
STK Ephemeris	Position (X, Y, Z) Velocity (X dot, Y dot, Z dot)

The user may specify the available measurements, satellites, trackers, and emitters for use in each computational process (simulator, filter, *etc*.). In addition, each satellite may be restricted to a set of allowable measurement types that may be used to track that satellite.

Measurement editing (which rejects suspicious data) and measurement thinning (which decreases the sampling rate of a series of measurements) provides greater flexibility. These settings are controlled on a per-process basis (*e.g*., filter, LS) using the Custom Data Editing attributes. Users can build sets of rules to control which measurements are processed.

4.3. Tracking Data Formats

ODTK natively supports a variety of tracking data formats, as seen in Table 3. Users can create plug-ins to read and write custom data formats.

Table 3. Tracking Data Formats

NASA UTDF and GEOS-C
DSN TRK-2-34
SLR CRD, NPT, LSR
AFSPC archive and transmit B3
CCSDS TDM

4.4. GPS Processing

ODTK supports two primary applications of GPS data processing: orbit estimation for satellites with onboard GPS receivers and estimation of the orbits and clocks for the entire GPS constellation. The L1 and L2 frequencies may be modified to model navigation systems other than GPS.

For user satellite orbit determination, a priori GPS satellite ephemeris and clock information may be provided via SP3c or SP3a formatted files. Observation modeling includes the use of GPS and user satellite specific antenna offsets and attitude profiles. Estimation capabilities include the estimation of user clock offsets and antenna location. The filter can process RINEX formatted pseudo-range and carrier phase observations or navigation solutions. A capability to compute navigation solutions from pseudo-range observations is also provided.

For GPS constellation estimation, all satellite orbits and clocks are estimated in conjunction with clocks on ground based receivers. Clock modeling can be individually tuned for each satellite and receiver.

Raw GPS observations may be processed directly or standard combinations and differences can be constructed to eliminate the effects of the ionosphere or clock errors including dual frequency (DF), single differencing (SD), and double differencing (DD).

DF corrections enable the user to eliminate the ionosphere uncertainty. The process requires two signal measurements from the same satellite, but on different frequencies. SD measurements eliminate the receiver clock error and require observations from two different GPS satellites. The SD approach is especially useful for GPS receivers that have steered clocks rather than a free-running clock. The DD approach takes the SD approach one step further and removes GPS clock errors at the cost of requiring additional GPS receivers.

The preceding approaches can also be combined to simplify the processing. For example, the SD DF approach which removes ionosphere effects and receiver clock errors is particularly useful in user satellite orbit determination.

4.5. Coordinate Frames

ODTK 6 (along with STK 9) fully supports the latest improvements in the realization of the International Celestial Reference Frame (ICRF) [9]. This is the default inertial coordinate frame, replacing the J2000 frame. A backwards compatibility mode is provided which causes all calculations to be performed in the J2000 frame and the ICRF options to be hidden in the GUI. Older scenarios loaded into ODTK 6 have the backwards compatibility mode enabled by default, while new scenarios will use the ICRF frame; the user may change this preference for any scenario.

5. FORCE MODELING

ODTK 6 uses AGI's High Precision Orbital Propagator (HPOP) which provides an extensive selection of equations of motion, numerical integration methods, and force modeling options. This is the same numerical orbit propagation capability that is available in STK with a slightly restricted feature set.

5.1. Gravity

The gravity model for ODTK is provided via text files; the EGM2008 gravity model has been added for use in ODTK 6. The user may specify the maximum degree and order to be used. ODTK's gravity process noise accounts for certain errors associated with integrating through the gravity field. The gravity process noise model allows the user to account for errors of commission (uncertainty in the gravity field coefficients) and/or errors of omission (uncertainty due to truncating the gravity field). Adding gravity process noise increases the orbit covariance. Gravity-field process noise can always be used, always ignored, or dependent upon the satellite's orbit and other pre-defined factors.

Perturbing accelerations due to solid and/or ocean tides follow IERS Conventions [7]. Tide models can also be truncated to speed processing. ODTK's implementation of General Relativity effects (Schwarzschild solution, geodesic and Lense–Thirring precessions) also follows IERS conventions.

Third-body gravitational accelerations due to the Sun, Moon, and/or major planets are available. In ODTK 6, the user may specify the source and value of the gravitational parameter (μ) for each planet.

5.2. Atmospheric Drag

Atmospheric drag can be automatically activated in ODTK when the perigee of the spacecraft's orbit falls below a minimum threshold defined by the user. Available atmospheric models include CIRA 1972, Jacchia-Roberts, MSISE 1990, and NRLMSISE 2000. In ODTK, the satellite ballistic coefficient is defined as $BC = c_D A / m$, where c_D is coefficient of drag, A is frontal cross-sectional area, and m is the satellite mass; the user can work with BC directly, or input c_D, A, and m if they are known.

ODTK can estimate a relative correction to BC while simultaneously estimating a relative correction to the local atmospheric density [10]. The relative corrections to atmospheric density and ballistic coefficient are each modeled as Gauss-Markov processes, wherein the user defines their exponential half-lives. The two states become separable when the half-life of the density correction is significantly different from the half-life of the BC correction and when the atmosphere is active. Except during an initialization period, the root-variance of the density correction is calculated internally from the atmospheric model, while the root-variance of the BC correction is user defined.

5.3. Solar Radiation Pressure

Solar-radiation pressure is modeled using a penumbral cone model for the Earth's shadow and the Moon's shadow. The model can be activated automatically when the spacecraft's orbital apogee crosses a threshold value. A relative correction to the satellite's nominal solar pressure coefficient can be estimated by the filter, modeled as an exponentially decaying Gauss-Markov process, where the analyst can define the exponential half-life of the process. A spherical radiation model is provided and can be configured as either diffusely reflecting or perfectly absorbing. ODTK can account for the change in solar pressure if a satellite is in lunar shadow, a recommended option for high-accuracy satellite programs. White process noise can be further added to the in-ecliptic and out-of-ecliptic directions normal to the Sun-to-satellite vector. This is useful should significant solar pressure accelerations exist in these directions, as solar pressure ordinarily accounts for acceleration along the satellite-to-Sun direction. An Earth albedo model and a thermal radiation pressure model are also available. These computationally intensive models are defined by a user-designated ground-reflection file.

The GPS satellite solar-pressure models are specific to the GPS Block types (II, IIA, IIR, IIRM, etc.). JPL's Block IIA and Block IIR models are recommended for precision work; these are the GSPM.04a models derived from a study of non-eclipsing satellites, and the GSPM.04ae extension for satellites undergoing eclipse. Optional legacy models include the Aerospace T20 (Block IIA) and the Aerospace T30 (Block IIR) models. ODTK's implementations employ two scale parameters: one primarily affects acceleration in the satellite-body-fixed X-Z plane containing the Sun-to-satellite line (K1), and the other primarily affects acceleration along the body-fixed Y axis of the satellite (K2). These scale factors can be estimated and are modeled as exponentially decaying Gauss-Markov processes, where the analyst defines the process variances and exponential half-lives.

Spacecraft-specific solar pressure models can be included through a solar pressure model plug-in point. The attitude of the spacecraft is automatically accounted for as the plug-in interface provides the direction of the Sun in the spacecraft body frame and accepts accelerations computed in the same frame. The solar pressure plug-in can also specify a list of parameters to be exposed through the ODTK user and automation interfaces and supports the estimation of corrections to plug-in model parameters.

5.4. Maneuvers

ODTK 6 models instantaneous (or impulsive) and finite maneuvers. The four types of instant maneuvers are:

- *velocity change* (a physically non-realistic method adequate for short-term thrusting that treats the ΔV as instantaneous),
- *acceleration history* (determined assuming time history of satellite acceleration during the maneuver),
- *constant thrust* (assumes increasing acceleration with decreasing mass per $a(t) = Fconst \,/\, m(t)$), and
- *constant acceleration* (assumes decreasing force with decreasing mass per $F(t) = m(t) \cdot aconst$).

Because an interval of thrusting is approximated as an impulse, any tracking data collected during the thrust interval may be mis-modeled. A data-exclusion interval can be specified to ignore any such tracking data.

Finite (time-interval) maneuvers are also provided for accurate modeling of long-duration thrusting. The three types of finite maneuvers are velocity change, acceleration history, and constant thrust. Multiple impulsive or finite maneuvers can be specified on a single satellite or multiple satellites. When multiple finite maneuvers are specified for a single satellite, the maneuvers can be configured to be independent (i.e., each maneuver is assigned its own estimation states - appropriate if there is a long time between the maneuvers, or, if the maneuvers use different thrusters) or related (multiple maneuvers share common thrusters).

Given tracking data during thrusting, the magnitude and/or direction of a finite maneuver can be estimated by the filter, each component being modeled as an exponentially decaying Gauss-Markov process, where the analyst can define the exponential half-life of the process. Estimates of impulsive maneuvers can be extracted from the output of the smoother [11]. Satellite maneuvers can be defined relative to several coordinate frames in terms of Cartesian components or in terms of direction and magnitude.

The user may further specify ODTK's unmodeled accelerations option to add process noise in the radial, in-track, and cross-track directions, to account for unmodeled phenomena such as out-gassing. The process noise is applied to the filtered covariance at each time update.

5.5. User (plug-in) force models

A single force-model plug-in point is provided for the customization of the accelerations affecting satellite propagation. A typical use might be to implement a specialized non-conservative force model, such as a box-wing model for atmospheric drag. Plug-ins may be written in any language that supports a Microsoft COM interface, such as Perl, VBScript, C++, C#, and Visual Basic .NET. Details regarding the ODTK Plug-in option are found in the ODTK Help.

5.6. Numerical Integration

The following numerical integrators can be set for each satellite: Bulirsch-Stoer integration (based on Richardson extrapolation with automatic step size control), 4th-order Runge-Kutta (no step-size error control), and 7th-order Runge-Kutta-Fehlberg with 8th-order step-size error control.

There are two methods of orbit error transition in ODTK: variational equations, and numerical partial derivatives. The variational equations are the partial derivatives of the accelerations with respect to the satellite position and velocity, and are used to compute the time derivative of the orbit error transition matrix. When using variational equations, the orbit error transition matrix is numerically integrated with the satellite.

When using the numerical partial derivatives method of transition, ODTK propagates six additional trajectories (each one originating from a small deviation having been applied to each component of the position or velocity). Trajectory differences are used to numerically approximate the partial derivatives that contribute to the orbit error transition matrix. This method is comparatively slow, but useful for comparison purposes.

6. ORBIT DETERMINATION TOOLS

ODTK 6 includes several tools that enable the user to perform complete analyses of satellites. These include initial orbit determination (IOD) methods, a least squares (LS) method, filter, fixed interval smoother (FIS), and a variable lag smoother (VLS). With the exception of the IOD processes, all of the orbit determination tools produce output files which can be used in the generation of reports and graphs.

6.1. Initial Orbit Determination

IOD creates position and velocity vectors from satellite tracking data when no prior orbit estimate is available. IOD methods assume two-body dynamics and the results are very sensitive to observation errors. Different IOD methods are available, depending upon the tracking data available.

The ***Herrick-Gibbs*** method requires three sets of range, azimuth, and elevation at different times. The user can also use Satellite Laser Ranging (SLR) Normal Point (NP) Range measurements. ***Gooding's angles-only*** method requires at least 3 sets of angles: azimuth and elevation; right-ascension and declination; or X and Y. The angles can be from ground based or space based tracking systems.

A third IOD option is the GPS navigation solution method. This method uses RINEX-formatted GPS observations as inputs. The method estimates a spacecraft's position and clock-phase using an analytical navigation solution algorithm [12] for several points around the time of interest, then fits of an interpolating polynomial through those positions and uses the derivative to estimate the velocity. Measurements from at least 4 GPS constellation space vehicles are necessary at each time, including the solution epoch.

6.2. Least Squares

ODTK uses a weighted least-squares estimator to refine the orbit derived from the IOD method. The batch-least-squares process is limited in the number of states estimated. These include position, velocity, solar-radiation-pressure correction, ballistic-coefficient correction, and measurement biases. The primary function is to refine the IOD estimate (or other initial estimate) sufficiently using a more complete force model . This refined state is then used to initialize the filter.

6.3. Filter

ODTK employs a current-time sequential filter [13]. The Kalman filter measurement update is used, but the time-update equations are non-Kalman, with process-noise based on the uncertainty in the physical models. This enables the calculation and propagation of a more realistic state error covariance. The filter also estimates time-varying measurement biases as well as time-varying corrections to the force models. The covariance has a variety of uses, including system performance analyses, optimization of tasking, and calculation of the probability of collision. The ODTK filter provides various output controls for data archiving, display, output for the fixed interval smoother, and creation of STK ephemeris files for display and file operations.

The sequential filter is initialized with an initial epoch, state estimate, state-estimate-error covariance matrix, measurement data to update the state estimate, and all support data and model parameters to operate the filter process. ODTK also provides an automatic *filter restart* capability that periodically saves this necessary information to a "restart disk file" at intervals specified by the user. The filter process can then be restarted from any epoch in the restart file as required.

Initialization of the filter from the most recent restart record is the normal processing mode for operational orbit determination. Use of the restart capability also allows the filter to resume operation at an earlier epoch without having to reprocess historical data and without loss of accuracy or information. If the filter diverges during a long computer run during operations (*e.g.*, the user neglected to model an orbit maneuver), use of selectable restart epochs can expedite recovery.

When a state is manually added or changed, the cross-correlations to other states are initialized as zero under the assumption that the new content is independent of existing content. Because flexible-state-space actions change the dimension and content of the filter state, the user may be required to take additional actions to save and use intermediate files, such as restart files, smoother-input files, and output files that drive reports and graphs.

6.4. Simulator

The purpose of the simulator is to create a set of realistic measurements by varying the initial conditions of satellites and measurement biases, and by adding noise to modeled measurements. The user controls where deviations are applied. If no deviations are applied, the resultant simulated measurements should provide nearly zero residuals when processed by the filter. The user can optionally apply a scale factor to the deviations that effectively acts as a sigma multiplier. A parameter is “deviated” when its value is perturbed by a Gaussian pseudorandom number drawn from a N (0, σ) distribution. The deviated result defines a randomly defined “truth” for the simulation. Options include deviating orbits (the initial satellite orbit state is deviated against the full orbit covariance in the RIC frame), atmospheric density, ballistic coefficient, solar-pressure coefficient, transponder delay, measurement biases, troposphere biases, maneuvers, facility locations, and antenna locations. White noise can be added to simulated measurements, and simulated process noise can be further added to density corrections, solar pressure, transponder delays, measurement biases, direction and magnitude of finite maneuvers, and GPS receiver clocks. The sequences for simulated process noise, except for clocks, are described by Equation 1. Details of the clock process noise model are found in [3]. ***Custom tracking intervals*** allow the analyst to define inclusion and exclusion intervals and vary time step by facility and/or satellite.

6.5. Fixed Interval Smoother

ODTK’s fixed-interval smoother combines filtered state and covariance information in reverse chronological order to calculate the best post-fit estimate of the orbit throughout any interval of interest. The smoother produces significantly smaller covariances than the filter, and state estimates that behave smoothly.

ODTK’s fixed-interval sequential smoother is a nonlinear adaptation of Meditch’s linear fixed-interval optimal smoother [14]. The smoother operates using outputs saved from the sequential filter, and the trajectory propagator runs backwards with time to accommodate the smoother’s state-transition function. The smoother runs backwards through time, smoothing ephemeris discontinuities caused by state corrections due to processed measurements. Smoothing operations can be applied over the results of multiple filter runs to provide longer duration smoothed outputs. A mathematical overview of the ODTK smoother is provided by [3], and a detailed description is available via the ODTK Help System.

6.6. Variable Lag Smoother

In addition to the fixed-interval smoother, useful for post-event reconstruction, ODTK 6 includes a forward-running variable lag smoother (VLS). The VLS is a sub-object of the filter and allows the user to smooth the output of the filter as it runs, obtaining a near-real time smoothed solution. The variable lag smoother is constructed as a sequence of fixed epoch smoothers initialized from the filter as it runs. Each fixed epoch smoother solution is updated each time the filter processes a measurement and therefore represents the best possible estimate at each fixed epoch given the current time of the filter. Each fixed epoch smoother is updated independently until one or more of its user defined convergence criteria are satisfied. Combined with the filter restart capability, the variable lag smoother provides a means for continuous generation of a smoothed solution with uniform information content over long durations. The time between fixed epochs is controllable by the user and provides a means of regulating the computation expense associated with the VLS.

The same fixed epoch smoother algorithms used in the VLS are leveraged to allow the estimation of impulsive maneuvers in the forward running filter. In this case, the filter initializes two fixed epoch smoothers at the time of the impulsive maneuver, one which contains the velocity change and one which does not. When both fixed epoch smoothers have converged, the difference between the two smoothed states represents the maneuver estimate.

6.7. State Differencing

ODTK 6 provides the capability to generate differences between state histories produced by its various processes. State differences include all estimated parameters so that difference between transponder bias estimates can be examined as easily as orbit differences. Examples of common state differences include filter minus simulator and filter minus smoother differences. These differences can be used to verify that state estimates are within their formal uncertainty of a simulated truth or to support the computation of filter-smoother consistency test metrics as described below.

7. FILTER-SMOOTHER CONSISTENCY

The *filter-smoother consistency test* is useful for general model validation. It is based on a proof that the difference between the filtered state (X_f) and smoother state (X_s) is normally distributed in n dimensions (assuming n is the size of the state-difference vector), and the variances and correlations of the state-difference vector are equal to the filter covariance subtracted from the smoother covariance [15]. Plotting this difference vector over time creates a population that should be normally distributed.

$$consistency = \frac{X_f - X_s}{\sqrt{\sigma_f^2 - \sigma_s^2}} \tag{3}$$

If the result is abnormal, then this is interpreted as incorrect setting in the filter-smoother model. ODTK provides a graphical method of examining filter-smoother consistency of each state element, instead of a full multivariate test statistic originally proposed by McReynolds [15]. For the single-state case, the filter-smoother state differences are divided by the filter-smoother variance differences. Filter-smoother consistency is generally claimed when this metric stays within

± 3 over the fit interval. This single-element test provides the user with a very powerful diagnostic tool as this test is equally applicable to simulated and real data scenarios. Typical examples of problems are an initial state outside the radius of convergence, incorrect mass/area/other satellite parameter, out-gassing or momentum dumps that are not modeled directly, etc. Sometimes the cause of the mis-modeling is unrelated to a specific force model or parameter, at which time an un-modeled acceleration can be introduced for solution.

8. SCRIPTING

ODTK 6 has an extensive scripting language that lets the user interface with the various technical features, build scenarios, process data, run other executables, etc. There are two primary approaches – scripts to perform some action from within an open scenario, and scripts which actually build scenarios and perform complex analyses. Scripts are commonly written in an html file using VBScript, Perl, Java, etc.

9. CONCLUSIONS

ODTK 6 is a commercial off-the-shelf software product used worldwide by high accuracy users for the analysis, design, development, and operation of various satellite systems. As a 3rd generation tool, it has evolved into a robust tool capable of supporting the accuracy and timeliness requirements of many stringent satellite missions. This was accomplished by conforming to international standards, leveraging proven algorithms and models, continually pursuing state-of-the-art information in open literature, and by continual technical refinement based on analysts' experience and feedback. This paper describes many of the features that have made ODTK functional for the applications it serves. New features will continue to be added based on customer requests.

10. REFERENCES

1. Wright, J.R. (1981). Sequential Orbit Determination with Auto-Correlated Gravity Model Errors, *Journal of Guidance and Control*, Vol. 4, No. 3, May 1981, pp. 304-309.

2. Hujsak, R.S., Woodburn, J.W. & Seago, J.H. (2007). The Orbit Determination Toolkit – Version 5. Paper AAS 07-125, Proceedings of the AAS/AIAA Spaceflight Mechanics Conference, Sedona, AZ.

3. Wright, J.R. & Woodburn, J.W. (2008). Nonlinear Variable Lag Smoother. Paper AAS-08-303, Proceedings of the AAS Landis Markley Astronautics Symposium. Cambridge, MD.

4. Hujsak, R.S., Woodburn, J.W. & Seago, J.H. (2007). The Orbit Determination Toolkit – Version 5. Paper AAS 07-125, Proceedings of the AAS/AIAA Spaceflight Mechanics Conference, Sedona, AZ.

5. http://www.agi.com/resources/help/odtk-help/, http://www.agi.com/whitepapers/

6. Tapley, B.D., Schutz, B.E. & Born, G.H. (2004). *Statistical Orbit Determination*. Elsevier Academic Press, Burlington, MA (ISBN 0126836302), Appendix F.

7. Ulenbeck, G. E. & Ornstein, L.S. (1930). On the Theory of the Brownian Motion. *Physical Review*, Vol. 36, September 1, 1930. pp. 823-41.

8. McCarthy, D,D. & Petit, G. (2003). IERS Technical Note 32. U.S. Naval Observatory.

9. Coppola, V., Seago, J.H. & Vallado, D.A. (2009). The IAU 2000A and IAU 2006 Precession-Nutation Theories and their Implementation. Paper AAS 09-159, Proceedings of the AAS/AIAA Spaceflight Mechanics Conference. Savannah, GA.

10. Wright, J.R. & Woodburn, J.W. (2004). Simultaneous Real-time Estimation of Atmospheric Density and Ballistic Coefficient. Paper AAS-04-175, Proceedings of the AAS/AIAA Spaceflight Mechanics Conference. Maui, HI.

11. Woodburn, J.W., Carrico, J. & Wright, J.R. (2003). Estimation of Instantaneous Maneuvers Using a Fixed Interval Smoother. Paper AAS 03-518, Proceedings of the AAS/AIAA Astrodynamics Specialists Conference, Big Sky, MT.

12. Yang, M. & Chen, K-H. (2001). Performance Assessment of a Noniterative Algorithm for GPS Absolute Positioning. Proceedings of the National Science Council, ROC(A) Vol. 25, No. 2, 2001, pp. 102-

13. Wright, J.R. (2002). Optimal Orbit Determination. Paper AAS 02-192, Proceedings of the AAS/AIAA Space Flight Mechanics Meeting, San Antonio, TX.

14. Meditch, J. S. (1969). *Stochastic Optimal Linear Estimation and Control*. McGraw-Hill, Inc., NY.

15. McReynolds, S.R. (1984). Editing Data Using Sequential Smoothing Techniques for Discrete Systems. Paper AIAA-1984-2053 Proceedings of the AIAA/AAS Astrodynamics Conference, Seattle, WA.