Software Architecture and Use of Satellite Tool Kit's Astrogator Module for Libration Point Orbit Missions

John Carrico, Emmet Fletcher
Analytical Graphics, Inc.

Abstract

The Satellite Tool Kit (STK) Astrogator software module is the third and most recent version of a program originally developed by NASA Goddard Space Flight Center (GSFC). This software lineage - Swingby, Navigator, Astrogator – started in 1989 and has since been used to design and operate many missions, including the non-low Earth orbit missions Clementine, Wind, SOHO, ACE, Lunar Prospector, the AsiaSat 3 rescue, and MAP. This paper describes the history of the software program and reasons behind the numerical methods employed. The authors also discuss the software design methodology and goals that led to this mature software product. Limitations encountered during analysis and operations use are described, as well as subsequent architecture changes made to alleviate them, reduce risk, and support automation.

Introduction

Astrogator is the maneuver planning and trajectory design module of Satellite Tool Kit (STK), a completely commercial off-the-shelf (COTS) software program developed by Analytical Graphics, Incorporated (AGI)[1]. Astrogator is fully integrated within STK and, among other things, can be used to generate the orbit ephemeris and attitude history of spacecraft. These data are then available for subsequent analysis and processing by other modules in STK, such as calculating station acquisitions, lighting times, communication links, coverage effectiveness, and Sensor obscurations.

History and Use

The ISEE-3/ICE[2,3] mission was designed and operated by NASA GSFC in Greenbelt, Maryland, USA. Launched on 12 August 1978, the spacecraft was transferred to the Sun-Earth L1 (interior colinear) libration point, and became the first spacecraft stationed in a libration point orbit. The software used for trajectory design and operations was the Goddard Mission Analysis System, GMAS[4], and an early variation on the General MANeuver program, GMAN[5]. (This special version was known as ICEMan.) In the late 1980's, the NASA GSFC Flight Dynamics Facility (FDF) initiated an effort to get rid of its mainframe computers, and because both these programs ran on the mainframe, alternatives were explored. One initiative, started in 1989, was to create a new trajectory design program based on a personal computer (PC). This task was given to Computer Sciences Corporation (CSC). In addition to meeting the requirements of the previous software, the new software was to incorporate graphics as an aid to trajectory design and analysis. This program was called Swingby[6,7,8], named because the first mission it was designated to support was the double-lunar swingby (DLS) mission Wind[9,10,11]. In 1992, however, the FDF was tasked with supporting trajectory design and maneuver operations for the Deep Space Program Science Experiment, DSPSE, also called Clementine[12,13]. Under the direction of the NASA GSFC FDF, working with the Naval Research Laboratory in Washington D.C., Swingby was enhanced to support Lunar orbit, asteroid rendezvous, and operational maneuver planning tasks. Clementine launched on 25 January, 1994 and performed a successful 2 month lunar orbit. (Unfortunately, afterwards the spacecraft suffered an on-board computer problem and was lost after it left the Moon on the way to the asteroid Geographos.) Subsequent to Clementine, Swingby was used in the NASA GSFC FDF to support Wind[9,10,11]; The Solar and Heliospheric Observatory, SOHO[14,15,16]; the Advanced Composition Explorer, ACE[17]; and Lunar Prospector[18].

NASA GSFC also distributed Swingby to several educational and government organizations within the United States. These included the United States Air Force Academy; University of Colorado, Boulder; University of Texas, Austin; NASA Ames[18], and the Jet Propulsion Laboratory[19].

Swingby earned Computer Science Corporation's Corporate Technical Excellence Award. In 1994, CSC commercialized Swingby and starting selling the software with the name Navigator, in cooperation with AGI.

On 25 December 1997, Hughes Aerospace Corporation was supporting the launch of a geostationary satellite, AsiaSat 3. However, the 4th stage only performed 1 second of an intended 110-second burn, and the spacecraft was stranded in a useless transfer orbit inclined at 51.6 degrees. The spacecraft engineers calculated that they needed 2424 meters/second in order to perform a Hohman transfer to geostationary orbit, but the spacecraft only had 2020 meters/second on board. Using Navigator, however, the engineers designed a trajectory that used Lunar gravity assists. They started their rescue maneuvers on 10 April 1998 and brought the spacecraft to a useful geostationary orbit with extra fuel available for stationkeeping[20]. The engineers used Navigator because their routine maneuver planning software was not designed for such multi-body trajectories.

In 1996 AGI bought Navigator from CSC, and obtained the rights to commercialize Swingby. At the request of the NASA GSFC Flight Dynamics Analysis Branch (FDAB), AGI incorporated Swingby into the STK product line, enhanced the capabilities, and started selling the software under the name STK/Astrogator. The details of the design process follow, but the first major goal was to support the analysis, launch window calculations, transfer trajectory operations, and stationkeeping of the Microwave Anisotropy Probe (MAP). The Astrogator software was ready on time for the pre-launch analysis tasks, such as fuel budget determination, calculation of the launch windows, and contingency planning. STK/Astrogator has successfully supported operations since launch on 30 June 2001[21,22,23,24,25].

STK/Astrogator was also used operationally at the Applied Physics Laboratory in Laurel, Maryland, USA, for the Comet Nucleus Tour mission (CONTOUR). Astrogator was used for planning and analysis of the cislunar phasing loop maneuvers. After the unfortunate loss of contact of the spacecraft after the solid motor firing, STK/Astrogator was used to generate possible burn and no-burn trajectories, two-line element sets, and antenna pointing angles to support the search efforts[26].

A summary of the libration point and deep-space missions that employed STK/Astrogator for operations are shown in Table 1.

Table 1 Operational Lunar, Libration Point, and Multi-Body Missions Supported with Swingby, Navigator, and STK/Astrogator

Mission	Launch	Regime
Clementine	25 January, 1994	Lunar Orbit / Asteroid
Wind	1 November 1994	DLS
SOHO	2 December 1995	Sun-Earth L1
ACE	25 August 1997	Sun-Earth L1
AsiaSat 3 rescue	25 Dec 1997	Lunar Gravity assist
Lunar Prospector	7 January, 1998	Lunar Orbit
MAP	30 June 2001	Sun-Earth L2
CONTOUR	3 July 2002	Cislunar phasing loop/ Comet tour

In addition to being used operationally, Astrogator has been used by many government, educational, and commercial organizations to analyze and design future missions. In particular, Astrogator has been used for the Triana Sun-Earth L1 mission, and the NGST at L2 proposals. It has also been used for some Sun-Mars libration point studies[27,28]. In cislunar space it has been used to analyze some Earth-Moon libration-point missions as well as many lunar orbiting and landing missions, including several missions using the weak stability boundary (WSB) transfer[29].

For these types of multi-body missions, Astrogator has been used for trajectory design, launch-window calculations, fuel estimates, developing station keeping strategies, and evaluating other mission requirements such as shadow analysis and communications link studies.

The majority of Astrogator users have employed it for analysis and operations of many LEO and GEO missions. This, of course, reflects the population of spacecraft missions. This work has included orbit ascent, stationkeeping, ground track control, rendezvous, low-thrust, formation flying, and constellation missions. In addition, Astrogator has supported the analysis of several heliocentric missions, including solar sail missions, and those to the Sun, Mercury, Venus, Mars, Jupiter, Saturn, and Pluto[30].

It should be noted that STK and STK/Astrogator are being used Worldwide, most notably in the USA, but with a growing presence in prominent organisations in Europe and Asia.

Design Methodology and Goals

STK/Astrogator was developed after a requirements definition and design study. This section describes the process that was followed, the resulting requirements, and the specifications and methods used to meet those requirements.

Process

For three months in 1997, AGI technical personnel met several times in small, facilitated design groups with mission analysts from the GSFC FDAB, and from one of its subcontractors, CSC. The design groups listed past mission tasks and future mission requirements. They detailed their current work processes and developed work flow diagrams. These processes were studied, and the team developed suggestions for process improvement. In addition to the processes, the successes and the difficulties of the existing software were discussed. From these meetings requirements were derived.

AGI personnel developed and presented mockup and then prototype graphic user interface (GUI) designs to analysts in the NASA GSFC FDAB. AGI then delivered beta software for analysts to use, and supplied training. After gathering feedback, AGI developed and delivered software to the Goddard FDAB every few months.

During this development phase, the MAP mission required some of the newly implemented features. Consequently, one of the first tasks was to reproduce the existing Swingby setups with Astrogator. After AGI did this, the MAP trajectory design team used Astrogator in parallel with Swingby for a short while. The Goddard FDAB gave a contract to CSC to independently validate and verify Astrogator. When the MAP mission requirements were met, the MAP team switched to using Astrogator exclusively.

Many of the requirements developed in the design groups were standard accuracy and numerical method requirements for a trajectory program. In addition, there were five major areas that the design group developed: analysis through operations support; multiple mission support; seamless operation with STK; automation support; and user extensibility. The reasoning behind these requirements are described in the following sections.

Requirements on Numerical Methods

The design groups developed a set of requirements that would be standard for any operational trajectory software. The requirements can be simplified by stating that all forces must be modeled with sufficient accuracy to match the GSFC FDAB operational orbit determination software, the "Goddard Trajectory Determination System" (GTDS)[31]. These forces include multiple and selectable gravitational bodies, the gravitational effect of a non-spherical central body, atmospheric drag, and solar radiation pressure. Additionally, Astrogator was required to model impulsive and finite maneuvers with accuracy as good as or better than the existing Swingby and the General MANeuver program (GMAN). To meet these requirements, a high-fidelity trajectory propagation system was specified, based on the legacy software.

The details of STK's numerical methods are found in the on-line help system and at AGI's web site, www.stk.com. The algorithms themselves are well known in the industry and the literature: Numerical integration using Cowell's formulation as well as Variation of Parameters for orbit propagation; differential correction and homotopy continuation for targeting; bounded search algorithms for event detection; published atmospheric density models for the Earth and Mars; standardized coordinate and time transformations; planetary ephemerides from the Jet Propulsion Laboratory; et cetera.

The requirements that drove these numerical methods derive from the high-level requirement to support operations. Therefore, as mentioned above, the trajectory propagation was required to match that used for orbit determination. This caused subsequent requirements on the content and accuracy of the force models.

Furthermore, because STK is required to calculate predicted antenna pointing data and timing, requirements on Earth shape models, Earth orientation (rotation, precession, nutation, pole wander), and station mask modeling were imposed. In addition, possible mobile vehicles with satellite tracking or communication equipment imposed an additional requirement on STK that was met with the capability to create station masks from surrounding terrain data.

STK/Astrogator was also required to calculate maneuver plans with thruster timing and attitude data accurate enough for operations. This lead to derived requirements on finite-burn propulsion modeling for pressure regulated, blow-down, and ion-propulsion systems. Furthermore, Astrogator was required to support post-maneuver engine calibration, which led to requirements on the accuracy and consistency of orbit propagation during the maneuver compared to the orbit determination systems.

Analysis Through Operations Support

The design groups desired that the new software, Astrogator, would support a mission from early conceptual phases, through the rigorous pre-launch analysis, and throughout all phases of operations. This created a contradiction: the workflow analysis determined that during the pre-launch and analysis phases of missions, analysts benefit from easy-to-use interactive software, using modern GUI controls, with the ability to quickly set any and all input parameters. Furthermore, sometimes using reduced fidelity (less accurate) force models enable quick studies to be done efficiently. This contrasts greatly with the necessity in operations for high accuracy and strict configuration control of all relevant data and setups.

The design groups noted that when Swingby was first developed, its interactive GUI and graphics streamlined some trajectory design tasks by enabling the analyst to quickly try out many different ideas. The immediate graphical feedback helped analysts develop useful intuitions, especially in the 3- and 4-body dynamics of the missions the GSFC FDF was supporting at that time. However, when the Swingby software was used for simulations and then operations, several difficulties arose. These centered on the fact that to quality-check that the proper data were being used in the system, the analyst must display the user interface panel. When the panel displayed the data, it was possible that the analyst could accidentally adjust and therefore corrupt a numerical value. In fact, even if the analyst did not adjust the data, sometimes the GIU panel would slightly modify a number because the internal binary representation was converted to text for display, and then converted back to binary if the analyst hit the OK button on the panel. This caused a lot of extra work in operations to quality check maneuver plans before delivery.

To get around this problem, STK/Astrogator was architected to store individually configured items in separate XML files. These items represented the objects that make up the numerical simulation. Force model, numerical integrator, and central-body parameters are some examples. During analysis tasks, the analysts can set the parameters in these objects easily through STK's GUI. As launch approaches, and the setups start being used for simulations and finally operations, these individual object files can be flagged as 'read-only' using standard operating system commands. STK/Astrogator honors this setting, and will display the data of these objects in its GUI, but will not allow the analyst to re-set the parameters. Because these files are XML, they can also easily be displayed in human readable form and can be differenced with baseline object files for quality check purposes.

These objects are managed by implementation in C++ using a prototype pattern[32]. This is a software design pattern that allows objects to be instantiated, copied, and otherwise controlled in a convenient way. A prototype of at least one of every object is available for the analyst to use. If a modification is needed, then the analyst copies a prototype, modifies it, and uses the copy instead.

The design groups also noted that during spacecraft contingency and emergency operations, the tasks performed by analysts were similar to pre-launch activities: quick flexibility was needed. Therefore, AGI designed and built a system that allows the analyst to copy any configuration-controlled item to a new object, modify it's parameters, and use the new object instead of the old one. Then, when the contingency is over, the new item can be placed under configuration control by setting its file permissions.

Multiple Mission Support

Because the GSFC FDAB studies and supports such varied topics, Astrogator was required to support missions ranging from low Earth to deep space, libration point, and missions to and around other planets, comets, asteroids, the Moon, and the Sun. In order to meet this requirement, AGI employed a strategy design pattern[32]. This software pattern establishes well-defined interfaces for classes of objects, but allows the implementation to be different according to the need.

For example, all objects of the engine class must calculate thrust and exhaust velocity data. So all engine objects in STK have this interface. However, the polynomial engine model calculates these data in a completely different way than the solar-electric propulsion engine. During numerical integration the force model calls the engine model, selected by the analyst, and it simply asks for certain data. The force model has no knowledge of how the data are calculated. This strategy pattern was employed for many objects, including central bodies, orbital elements, power supplies, vectors and coordinate systems.

The strategy pattern, combined with the prototype pattern mentioned above, allows the analyst to configure the objects comprising the simulation to meet the specific mission needs. This reduces the number of options with which the analyst is presented. For instance, this eliminates the need to 'hard-code' a libration-point coordinate system for every set of appropriate bodies in the solar system, which would be quite a long list. Instead, STK/Astrogator comes installed with just a few configured libration-point coordinate systems. If the analyst wants to study, for example, the Sun-Neptune system, then they duplicate the Sun-Earth system, change the central body, and select which libration point should be the origin. In this way the analyst has easily customized the software to a specific need, without the development team having to guess all possible combinations ahead of time.

Seamless Operation with STK

In the support of various missions, designing the trajectory is only the first step. Previous to Astrogator, after the trajectory was designed, the ephemeris was saved to a file and post-processed with a variety of other software, including STK. Often during trajectory design the acceptance of a trajectory can only be verified by monitoring other parameters, such as ground station communication link coverage or solar lighting geometry. Therefore it was required that the trajectory design system seamlessly interchanges orbit and attitude data with STK. This requirement was met by specifying that Astrogator acted as an ephemeris and attitude source from within STK.

An example of integration with STK that has helped libration-point missions is the ability to calculate shadows from the Earth and the Moon. The MAP mission, for instance, has such strict no-shadow requirements—because of the thermal sensitivities of its scientific payload—that even annular eclipses from the Moon had to be avoided. Because Astrogator is integrated within STK, it became a simple process to run a shadow report after each candidate trajectory was developed. Other examples include analyzing contact times to ground stations; communications link budget and interference analysis; radiation dosage studies, and generic figure-of-merit coverage analysis around the Earth, Mars, or other planets.

The seamless integration with STK also benefits operations tasks by enabling the analyst to create a wide variety of data products for other groups. The data products can be customized and automated using STK's ability to change units and precision; automatically call post-processing scripts for formatting and merging; and by using the STK/Connect module to automate common tasks.

Another benefit of Astrogator's integration with STK is realized with the STK Visualization Option (STK/VO). This is described later in the section "Visualization."

Automation support

The ability to automate tasks such as parametric studies, Monte Carlo analyses, and customized search methods was critical for reducing the workload of the analyst users. In addition, operations personnel had relied on the ability to script routine operations to reduce both the cost of staff as well as reduce the risk of making mistakes. To support this, AGI required Astrogator to allow complete symbolic access (i.e., through a text name) to all input parameters. Symbolic access to each and every variable is useful, of course, only if the analyst can write control logic to change these parameters. AGI developed a custom script language, and this was demonstrated to several users as a prototype. The feedback, however, indicated that the majority of potential users were not willing to commit the time to learning a proprietary language without knowing that it would be successful. As a result, an interface became required that could be run from a variety of standard languages. A generic system was specified to include languages such as ‘C’, C++’, PERL, Python, Java, VBScript, and MATLAB, to name a few. Once these standard languages were supported, the analysts readily started using automation in trajectory design analysis and operations.

One of the more recent uses of the scripting capability has been to take data created with other programs and import them into STK. For example, to support the Triana mission, the company Space Exploration Engineering, Inc. (SEE)[33] was tasked by the GSFC FDAB to write a MATLAB tool to take trajectories designed using the Generator software (developed at Purdue University[34]) and import them into Astrogator as targeting constraints and first guesses at the maneuvers. Several other organizations have written similar scripts that integrate STK within other systems. Some have even gone as far as running STK completely automatically in a “lights out” operations center[35].

Scripting, however, was not sufficient to automate all tasks. In studying past mission analyses, three additional requirements arose. The first can be understood with an example. Take the problem of modeling a trajectory that includes performing a 20 meter/second maneuver at each periapsis until the radius of apogee is halfway to the Moon. This type of problem yielded the requirement that trajectory related actions (such as a maneuver) could be triggered by events (such as periapsis). A generic system was implemented that can trigger when any orbit parameter crossed any user-defined value. In addition, a system was specified that allowed any trajectory sequence of arbitrary complexity, including targeting, to be triggered by any one of these events.

The second additional requirement arose from missions that require more than one maneuver. During analysis and operations, after the initial conditions are changed and a maneuver is modeled, all the subsequent maneuvers must be re-planned. Therefore, the requirement arose that the maneuver targeting capability must allow automatic re-planning, without analyst intervention. This is especially important if the trajectory model is wrapped within the loop of a parametric or Monte Carlo script. This requirement was met by specifying a robust targeting algorithm with specific features to enable convergence even with very non-linear problems. In particular, a differential corrector with normalized parameters, a search step-size control algorithm, and a homotopy continuation method were employed.

The third additional requirement arose after studying how analysts and operators targeted trajectories; the ability to first achieve 'coarse' intermediate goals and then continue with further refinement was required. The analyst must be able to set up one targeting problem, and then, within the same section of the trajectory, target different, refined goals with either the same or, perhaps, different controls constraints. This requirement resulted in the specification of "Targeting Profiles" which enable the analyst to pair up a set of controls with a set of desired constraints, and then name this profile. Furthermore, it was specified that the analyst could create any number of these named profiles, and run them automatically in sequence; as soon as one profile was run successfully, the next in the list must run automatically. This has been quite useful in targeting non-linear problems.

User Extensibility

Before Swingby was built, the Goddard Mission Analysis System (GMAS) was used for trajectory design. GMAS ran on mainframe computers, and was written mostly in FORTRAN. One major feature that was used heavily in the design of complex trajectories was the ability for a programmer/user to create a new FORTRAN subroutine and have it available as a new module within GMAS. The analysts often used this to create new coordinate systems, new parameters for targeting or reporting, and to augment the propagator force models.

The first specification to meet this requirement called for the ability for a programmer/user to write a 'C' function with a specified function signature and compile it as a dynamic link library (DLL) file. Then, by placing the DLL file in the proper system folder, the user could "plug-in" a new function that would be available within Astrogator just as if it was built in from the beginning. This was successfully implemented, and the NASA GSFC FDD built several example plug-in DLL functions specific to deep space missions. However, during testing, it became clear that the typical trajectory analyst was not very proficient in the 'C' language, nor had much desire to learn. After about a year, this DLL feature was no longer used, and was taken out before the commercial release of Astrogator.

To meet the requirement, a new specification was derived with the lessons learned in mind. Specifically, it was desired to allow an analyst/user to augment the functionality of Astrogator without help from a programmer or from technical support. Therefore the new specification called for functions to be in a simple, standard script language. Initially three languages were chosen: MATLAB, PERL, and VBScript. (PERL and VBScript were chosen in particular because they are free to the analyst.) Examples were created with which the analysts can start, as well as documentation of the syntax. The specification further called for a simple text equation capability (not requiring source files) for the most common and simple task of defining a new generic orbit parameter as a function of exiting parameters. These enhancements were made and put into the commercial version of the software in early 2002. Specifically, AGI added plug-in points in Astrogator to allow an analyst to augment it with their own engine models; additional forces for orbit integration; vectors (and therefore axes and coordinate systems); and Astrogator calculation objects (scalar values used for stopping conditions, graphing, reporting, and targeting constraints). Additionally, plug-in points were added to STK for custom access and communication constraints and to model attitude dynamics and control.

Visualization

The stories are told that during the Apollo missions, trajectory analysts strung lights in darkened rooms in order to visualize and understand the 3-body transfer trajectories. Later, the aforementioned GMAS program was able to create simple plots of trajectories using text characters on green & white computer paper. There were a few other attempts within the Goddard organizations at using graphics software, and by the time Swingby was being designed, it became a requirement to display the trajectory graphically, as it was being calculated. Once implemented, this yielded several helpful results. Primarily, when an analyst is trying to develop a trajectory, and attempting to target maneuvers using a shooting method, it is immediately apparent if the trajectories start to diverge. Watching the trajectories target also gives the analyst insight to the cause of divergence, and makes it easier to rectify the problem.

Another benefit of the visualization arises from the fact that there are no easily understood metrics to describe the many trajectories in the 4-Body problem (Sun, Earth, Moon, and spacecraft). Much of the work of a trajectory analyst involves understanding the authority of the control maneuvers to cause a trajectory to meet mission requirements. The deviation and relationship of two trajectories with similar initial conditions in the 4-Body problem can be more easily understood graphically than numerically. Once the analyst gains this understanding, he or she can devise metrics to quantitatively control the trajectory. In addition, the animation of the dynamics helps give insight into trajectory controls that can affect the evolution of a trajectory. These insights help tremendously to determine regimes where linear approximations are appropriate, and where they fail.

With the combination of STK/Astrogator and STK/VO, the analyst can interactively rotate the trajectory with the mouse. This has been quite helpful in understanding the effects of the controls, especially for transfers to the libration points, and even more so for those transfer trajectories involving lunar gravity assists such as MAP. Additionally, the 3-D STK/VO views were helpful in explaining the complex geometries to the reviewers, to the non-trajectory members, and to project management of the MAP team. In particular, the design of MAP's phasing loop trajectories kept the spacecraft out of Earth shadow, but small (annular) Lunar shadows were sometimes problematic. The discussions in design reviews were greatly added by the 3-D images and animations.

The 3-D visualization was also a great help in developing a forward targeting algorithm for Weak Stability Boundary trajectories[29]. While developing a methodology to correct the trajectory due to launch uncertainties and delays, the 3-D views showed clearly where changes behaved linearly, and where they were highly sensitive. This then led to the choice of stopping conditions and targeting constraints to correct the trajectory to meet mission requirements.

Summary

The software linage of Swingby, Navigator, and STK/Astrogator has been used for the analysis and operations of the majority of Libration point trajectories, as well as several Cislunar, Lunar, asteroid, and interplanetary missions, since it's first use for the Clementine (DSPSE) mission. The feedback from use in early missions has caused significant enhancements for the most recent incarnation, STK/Astrogator. The driving requirement to have a single software tool support pre-launch mission analysis, operations, and contingencies has been a major factor in selection of algorithms and methods, and several innovative solutions have been implemented to accomplish this task.

The close working relationship between software industry and government experts has been instrumental in the development of a commercial software product with a mature feature set able to support Libration point and other multi-body missions.

Finally the integration of the trajectory software package Astrogator with the other mission analysis capabilities of STK has proven to be valuable for many analysis and operational tasks: The 'normal' post-trajectory-design numerical tasks have been streamlined and automated, ensuring consistency, reducing risks, and allowing studies that would otherwise be time and cost prohibitive. Additionally, the interactive and advanced computer graphics capabilities have become an invaluable aid to understanding the complex trajectories that exist in the multi-body problem.

References

[1] Analytical Graphics, Inc. web site at www.stk.com

2 R.W. Farquhar, "The Flight of ISEE-3/ICE: Origins, Mission History, and a Legacy," AIAA 98-4464, AIAA/AAS Astrodynamics Specialist Conference, Boston, Massachusetts, USA, August 1998

3 R.W. Farquhar, et al., "Trajectories and Orbital Maneuvers for the First Libration-Point Satellite," J. Guidance and Control, Vol. 3, No. 6, Nov-Dec 1980, pp. 549 – 554

4 "GMAS System Description," from the NASA Goddard Space Flight Center, Flight Dynamics Analysis Branch

[5] "GMAN Mathematical Specification," from the NASA Goddard Space Flight Center, Flight Dynamics Analysis Branch

[6] J.P. Carrico, H.L. Hooper, L Roszman, C. Gramling, "Rapid Design of Gravity Assist Trajectories," Proceedings of the ESA Symposium on Spacecraft Flight Dynamics, Darmstadt, Germany, September/October 1991

[7] J.P. Carrico, C. Schiff, L. Roszman, H.L. Hooper, D. Folta, K.V. Richon, "An Interactive Tool for Design and Support of Lunar, Gravity Assist, and Libration Point Trajectories," AIAA 93-1126, AIAA/AHS/ASEE Aerospace Design Conference, Irvine, California, USA, February 1993

[8]J.P. Carrico, D. Conway, D. Ginn, D. Folta, K.V. Richon, “Operational Use of Swingby-an Interactive Trajectory Design and Maneuver Planning Tool-for Missions to the Moon and Beyond,” AAS 95-323, AAS/AIAA Astrodynamics Specialist Conference, Halifax, Nova Scotia, Canada, August 1995

[9] P. Sharer, H. Franz, D. Folta, “WIND Trajectory Design and Control,” Paper No. MS95/032, CNES International Symposium on Space Dynamics, Toulouse, France, June 1995

[10] H. Franz, “WIND Nominal Mission Performance and Extended Mission Design,” AIAA 98-4467, August 1998

[11] H. Franz, “WIND Lunar Backflip and Distant Prograde Orbit Implementation,” AAS 01-173, AAS/AIAA Spaceflight Mechanics Meeting, Santa Barbara, California, USA, February 2001

[12] J. Carrico et al., “Maneuver Planning and Results for Clementine (the Deep Space Program Science Experiment),” AAS 95-129, AAS/AIAA Spaceflight Mechanics Meeting, Albuquerque, New Mexico, USA, February 1995

[13] D.W. Dunham, R.W. Farquhar, “Trajectory Design for a Lunar Mapping and Near-Earth-Asteroid Flyby Mission,” AAS 93-145, *Spaceflight Mechanics 1993*, Vol. 82, Advances in the Astronautical Sciences, ed. R. Melton, pp. 605-624

[14] D.W. Dunham, et al., “Transfer Trajectory Design for the SOHO Libration-Point Mission,” IAF-92-0066, 43rd Congress of the International Astronautical Federation, Washington, D.C., USA, August/September, 1992

[15] D.W. Dunham, “Handbook on Trajectories, Mission Design, and Operations for the Spacecraft of the Solar-Terrestrial Science Project,” Vol. 3, prepared by the Working Group 3 of the IACG for Space Science, September 1994

[16] D.W. Dunham, C.E. Roberts, “Stationkeeping Techniques for Libration-Point Satellites,” AIAA 98-4466, AIAA/AAS Astrodynamics Specialist Conference, Boston, Massachusetts, USA, August 1998

[17] C.E. Roberts, “Long Duration Lissajous Orbit Control for the ACE Sun-Earth L1 Libration Point Mission,” AAS 01-204, AAS/AIAA Space Flight Mechanics Meeting, Santa Barbara, CA, USA, February, 2001

[18] D. Lozier, K. Galal, D. Folta, M. Beckman, “Lunar Prospector Mission Design and Trajectory Support,” AAS 98-323, pp. 297-311

[19] P.A. Penzo, “A Survey and Recent Developments of Lunar Gravity Assist,” Space Studies Inst., Princeton University, May 1998.

20 J. Salvatore, C. Ocampo, “Mission Design and Orbit Operations for the First Lunar Flyby Rescue Mission,” IAF-99-A.2.01, 50th International Astronautical Congress, Amsterdam, The Netherlands, October 1999

21 K.V. Richon, M.W. Mathews, “An Overview of the Microwave Anisotropy Probe (MAP) Trajectory Design,” Advances in the Astronautical Sciences, Vol. 97, 1998, pp. 1979-1998

22 M. Woodard, O. Cuevas, “MAP Trajectory Design,” Satellite Tool Kit User’s Conference, Georgetown, Washington D.C., USA, June, 2002

23 M. Mesarch “Contingency Planning for the Microwave Anisotropy Probe Mission,” AIAA 2002-4426, AIAA/AAS Astrodynamics Specialist Conference, Monterey, CA, USA

[24] M. Mesarch, S. Andrews "The Maneuver Planning Process for the Microwave Anisotropy Probe (MAP) Mission," AIAA 2002-4427, AIAA/AAS Astrodynamics Specialist Conference, Monterey, CA, USA

25 D. Folta, K.V. Richon, "Libration Orbit Mission Design at L2: a MAP and NGST Perspective," AIAA 98-4469, August 1998

26 Private communication with David Dunham, Dan Muhonen, and Peter Sharer of the Johns Hopkins Applied Physics Laboratory, Laurel, Maryland, July/August, 2002

27 J.D. Strizzi, J. Kutrieb, P. Damphousse, J. Carrico, "Sun-Mars Libration Points and Mars Mission Simulations," AAS 01-159, AAS/AIAA Spaceflight Mechanics Meeting, Santa Barbara, California, USA, February 2001

28 J. Carrico, J. Strizzi, J. Kutrieb, P. Damphousse, "Trajectory Sensitivities for Sun-Mars Libration Point Missions," AAS 01-327, AAS/AIAA Astrodynamics Specialist Conference, Quebec City, Quebec, Canada, July/August 2001

[29] E.A. Belbruno, J.P. Carrico, "Calculation of Weak Stability Boundary Ballistic Lunar Transfer Trajectories," AIAA 2000-4142, AIAA/AAS Astrodynamics Specialist Conference, Denver, Colorado, USA, August 2000

[30] Y. Gao, R. Farquhar "New Horizons Mission for the Pluto-Kuiper Belt Mission," AIAA 2002-4722, AIAA/AAS Astrodynamics Specialist Conference, Monterey, CA, USA, August, 2002

[31] "Goddard Trajectory Determination System (GTDS) Mathematical Reference," NASA Goddard Space Flight Center, Flight Dynamics Division

32 E. Gamma, R. Helm, R. Johnson, J. Vlissides, "Design Patterns" 1995 Addison-Wesley Longman, Inc.

33 Space Exploration Engineering web site at www.see.com

[34] K. C. Howell, "Families of orbits in the vicinity of the collinear libration points," AIAA 98-4465 , AIAA/AAS Astrodynamics Specialist Conference, Boston, MA, USA, August, 1998

35 AGI STK InView article about the PROBA mission, Fall 2002.