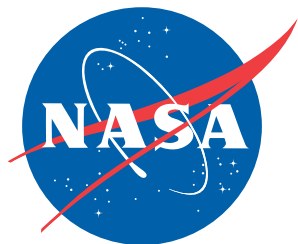


NASA/TM-2012-217328  
NESC-RP-09-00605



# Development of Autonomous Aerobraking (Phase 1)

*Daniel G. Murri/NESC  
Langley Research Center, Hampton, Virginia*

*Richard W. Powell  
Analytical Mechanics Associates, Inc., Hampton, Virginia*

*Jill L. Prince  
Langley Research Center, Hampton, Virginia*

## NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Report Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

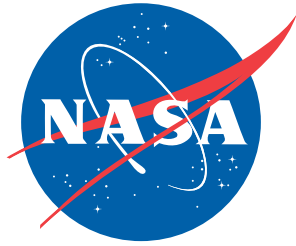
- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.
- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, and organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to [help@sti.nasa.gov](mailto:help@sti.nasa.gov)
- Fax your question to the NASA STI Help Desk at 443-757-5803
- Phone the NASA STI Help Desk at 443-757-5802
- Write to:  
NASA STI Help Desk  
NASA Center for AeroSpace Information  
7115 Standard Drive  
Hanover, MD 21076-1320

NASA/TM-2012-217328  
NESC-RP-09-00605



# Development of Autonomous Aerobraking (Phase 1)

*Daniel G. Murri/NESC  
Langley Research Center, Hampton, Virginia*

*Richard W. Powell  
Analytical Mechanics Associates, Inc., Hampton, Virginia*

*Jill L. Prince  
Langley Research Center, Hampton, Virginia*

National Aeronautics and  
Space Administration


Langley Research Center  
Hampton, Virginia 23681-2199

January 2012

The use of trademarks or names of manufacturers in the report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.


Available from:

NASA Center for AeroSpace Information  
7115 Standard Drive  
Hanover, MD 21076-1320  
443-757-5802

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP- 09-00605</b>	Version: <b>1.0</b>
Title:	<b>Autonomous Aerobraking (Phase 1)</b>		Page #: 1 of 286

## **Development of Autonomous Aerobraking (Phase 1)**

**December 15, 2011**


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 2 of 286	

## Report Approval and Revision History

NOTE: This document was approved at the December 15, 2011, NRB. This document was submitted to the NESC Director on January 10, 2012, for configuration control.

Approved:	<i>Original Signature on File</i> <hr style="width: 80%; margin: 0 auto;"/> NESC Director	1/11/12 <hr style="width: 80%; margin: 0 auto;"/> Date
-----------	--	---


Version	Description of Revision	Office of Primary Responsibility	Effective Date
1.0	Initial Release	Mr. Daniel G. Murri, NASA Technical Fellow for Flight Mechanics, LaRC	12/15/11

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		<b>3 of 286</b>	

## Table of Contents

### Technical Assessment Report (Phase 1)

<b>1.0</b>	<b>Notification and Authorization .....</b>	<b>7</b>
<b>2.0</b>	<b>Signature Page.....</b>	<b>8</b>
<b>3.0</b>	<b>Team List.....</b>	<b>9</b>
<b>4.0</b>	<b>Executive Summary .....</b>	<b>11</b>
<b>5.0</b>	<b>Assessment Plan.....</b>	<b>13</b>
<b>6.0</b>	<b>Problem Description and Proposed Solution.....</b>	<b>14</b>
6.1	Conventional AB .....	14
6.2	Benefits of AA .....	15
6.3	AA Approach.....	15
<b>7.0</b>	<b>Data Analysis.....</b>	<b>17</b>
7.1	Reference and “Truth” Simulations .....	17
	7.1.1 POST2 .....	20
	7.1.2 AAHFS .....	21
7.2	AB Mission Design Methodologies.....	21
7.3	AB Models.....	23
	7.3.1 “Truth” Models.....	23
	7.3.2 Onboard AA Models.....	28
7.4	AADS Model Integration.....	31
7.5	AA Simulation Results .....	32
	7.5.1 AADS Performance: POST2 Analysis .....	32
	7.5.2 AAHFS .....	62
	7.5.3 Future Work.....	88
<b>8.0</b>	<b>Findings, Observations, and NESC Recommendations .....</b>	<b>90</b>
8.1	Findings .....	90
8.2	Observations .....	91
8.3	NESC Recommendations.....	91
<b>9.0</b>	<b>Alternate Viewpoints .....</b>	<b>93</b>
<b>10.0</b>	<b>Other Deliverables .....</b>	<b>93</b>
<b>11.0</b>	<b>Lessons Learned .....</b>	<b>93</b>
<b>12.0</b>	<b>Definition of Terms.....</b>	<b>93</b>
<b>13.0</b>	<b>Acronyms List .....</b>	<b>94</b>
<b>14.0</b>	<b>References .....</b>	<b>96</b>
<b>15.0</b>	<b>Appendices .....</b>	<b>97</b>

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 4 of 286	

### List of Figures

Figure 6.0-1.	A Spacecraft using Apoapsis Maneuvers to Control Periapsis Altitude during AB.....	14
Figure 7.3-1.	MRO Spacecraft used for AA Study Simulation.....	24
Figure 7.3-2.	Mars-GRAM 2010 Density Perturbations.....	26
Figure 7.3-3.	Venus-GRAM Density Perturbations.....	27
Figure 7.3-4.	Titan-GRAM Density Perturbations.....	27
Figure 7.5-1a.	AADS Ephemeris Estimator Performance at Mars with Perturbed Atmosphere (and 7-day Updates): Current Periapsis Time.....	34
Figure 7.5-1b.	AADS Ephemeris Estimator Performance at Mars with Perturbed Atmosphere (and 7-day Updates): Current Periapsis Altitude.....	35
Figure 7.5-2a.	AADS Atmosphere Estimator Performance at Mars (with 7-day Updates): Predicted Periapsis Atmospheric Density.....	36
Figure 7.5-2b.	AADS Atmosphere Estimator Performance at Mars (with 7-day Updates): Predicted Periapsis Atmospheric Scale Height.....	37
Figure 7.5-3a.	AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 3-day Updates.....	38
Figure 7.5-3b.	AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 5-day Updates.....	39
Figure 7.5-3c.	AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 7-day Updates.....	40
Figure 7.5-3d.	AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 14-day Updates.....	41
Figure 7.5-4.	AADS Mission Operations Corridor Performance at Mars with a Nominal Atmosphere (and 7-day Updates).....	43
Figure 7.5-5a.	AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Orbit Period “GlideSlope”.....	44
Figure 7.5-5b.	AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Commanded Maneuver $\Delta v$ .....	45
Figure 7.5-5c.	AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Periapsis Altitude.....	46
Figure 7.5-5d.	AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Periapsis Areocentric Latitude.....	47
Figure 7.5-6a.	AADS Constrained Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere (and 7-day Updates) for a Tight Corridor Centered on the Nominal Corridor Lower Limit.....	49
Figure 7.5-6b.	AADS Constrained Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere (and 7-day Updates) for a Tight Corridor Centered on the Nominal Corridor Upper Limit.....	50
Figure 7.5-7.	AADS Mission Aggressive Operations Corridor Performance at Mars with a Perturbed Atmosphere (and 7-day Updates).....	52
Figure 7.5-8.	AADS Glide Slope Delay and Recovery Scenario at Mars with a Perturbed Atmosphere (and 7-day Updates).....	54
Figure 7.5-9a.	AADS Mission Operations Corridor Performance for the Nominal Corridor.....	55
Figure 7.5-9b.	AADS Mission Operations Corridor Performance for the Glide Slope Delay and Recovery Corridor.....	56




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 5 of 286	

Figure 7.5-10.	AADS Mission Operations Corridor Performance at Venus with a Perturbed Atmosphere (and 7-day Updates) and a Fixed Corridor Target.....	58
Figure 7.5-11.	AADS Mission Operations Corridor Performance at Venus with a Perturbed Atmosphere (and 7-day Updates) and a Varying Corridor Target.....	59
Figure 7.5-12.	AADS Mission Operations Corridor Performance at Titan with a Perturbed Atmosphere (and 7-day Updates) .....	61
Figure 7.5-13.	AAHFS 6-DOF Block Diagram.....	63
Figure 7.5-14.	AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Nominal Atmosphere and 7-day Updates .....	66
Figure 7.5-15.	AAHFS Simulation – AADS Orbital Period Performance at Mars with a Nominal Atmosphere and 7-day Updates .....	67
Figure 7.5-16.	AAHFS Simulation – Glide Slope Performance at Mars with a Nominal Atmosphere and 7-day Updates .....	68
Figure 7.5-17.	AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Nominal Atmosphere and 14-day Updates .....	69
Figure 7.5-18.	AAHFS Simulation – AADS Orbital Period Performance at Mars with a Nominal Atmosphere and 14-day Updates .....	70
Figure 7.5-19.	AAHFS Simulation – Glide Slope Performance at Mars with a Nominal Atmosphere and 14-day Updates .....	71
Figure 7.5-20.	AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere and 7-day Updates .....	72
Figure 7.5-21.	AAHFS Simulation – AADS Orbital Period Performance at Mars with a Perturbed Atmosphere and 7-day Updates .....	73
Figure 7.5-22.	AAHFS Simulation – Glide Slope Performance at Mars with a Perturbed Atmosphere and 7-day Updates .....	74
Figure 7.5-23.	AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere and 14-day Updates .....	75
Figure 7.5-24.	AAHFS Simulation – AADS Orbital Period Performance at Mars with a Perturbed Atmosphere and 14-day Updates .....	76
Figure 7.5-25.	AAHFS Simulation – Glide Slope Performance at Mars with a Perturbed Atmosphere and 14-day Updates .....	77
Figure 7.5-26.	AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere, 7-day Updates, and a Reduced Heat Rate Corridor.....	78
Figure 7.5-27.	AAHFS Simulation – Glide Slope Performance at Mars with a Perturbed Atmosphere, 7-Day Ground Updates, and a Reduced Heat Rate Corridor .....	79
Figure 7.5-28.	AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere, 7-day Updates, and an Elevated Heat Rate Corridor.....	80
Figure 7.5-29.	AAHFS Simulation – Glide Slope Performance at Mars with a Perturbed Atmosphere, 7-Day Ground Updates, and an Elevated Heat Rate Corridor .....	81
Figure 7.5-30.	AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere, Accelerometer Errors, and 7-day Updates.....	82
Figure 7.5-31.	AAHFS Simulation – AADS Orbital Period Performance at Mars with a Perturbed Atmosphere, Accelerometer Errors, and 7-day Updates.....	83
Figure 7.5-32.	AAHFS Simulation – Glide Slope Performance at Mars with a Perturbed Atmosphere, Accelerometer Errors, and 7-day Updates .....	84


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		6 of 286	

Figure 7.5-33. AAHFS Simulation – AADS Heat Rate Performance Histograms for all AAHFS Cases ..... 86

Figure 7.5-34. AAHFS Simulation – Aerodynamic Stability during Drag Passes..... 88

**List of Tables**

Table 7.1-1. Models Contained in “Truth” Simulations and AADS ..... 19

Table 7.3-1. MRO Mass Properties ..... 24

Table 7.5-1. Summary of AADS Heat Rate Estimate Performance Variability for Various Ground Update Frequencies ..... 42

Table 7.5-2. Summary of AADS Performance with a Perturbed Atmosphere (and 7-day Updates) Compared to Reference Simulation at Mars ..... 48

Table 7.5-3. Summary of AADS Performance at Mars with a nominal Atmosphere for Runs using the Nominal Operational Corridor, a corridor Constrained to the Nominal Corridor Upper Limit, and One to the Nominal Corridor Lower Limit..... 50

Table 7.5-4. Comparison of AADS Performance at Mars with a Perturbed Atmosphere (and 7-day Updates) for the Nominal and Aggressive Operational Corridors ..... 53

Table 7.5-5. Summary of AADS Performance with Perturbed Atmosphere (and 7-day Updates) Compared to Reference Simulation at Venus ..... 60

Table 7.5-6. Summary of AADS Performance with Perturbed Atmosphere (and 7-day Updates) Compared to Reference Simulation at Titan..... 61

Table 7.5-7. AAHFS Simulation – Accelerometer Error Sources per Accelerometer ..... 82

Table 7.5-8. AAHFS Simulation – AADS Performance at Mars..... 85

**Appendices (Phase 1)**

Appendix A. Autonomous Aerobraking Planetary Constants and Models Version 0.07 (Supplement to Section 7.2) ..... **98**

Appendix B. Mission Design Appendix (Supplement to Section 7.2) ..... **130**

Appendix C. Aerodynamics and Aerothermodynamics Computational Methods (Supplement to Section 7.3.1.2)..... **140**

Appendix D. Thermal Modeling (Supplement to Section 7.3.2.4) ..... **150**

Appendix E. Mars-GRAM 2010 (Supplement to Section 7.3.1.3)..... **168**


Appendix F. Onboard Atmospheric Modeling and Prediction for AA Missions (Supplement to Section 7.3.2.1)..... **177**

Appendix G. Ephemeris Estimator User’s Guide (Supplement to Section 7.3.2.2)..... **216**

Appendix H. AADS (Supplement to Section 7.5.1)..... **250**

Appendix I. AAHFS (Supplement to Section 7.5.2)..... **255**

Appendix J. AA Interface Control Document ..... **263**

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 7 of 286	


## Technical Assessment Report (Phase 1)

### 1.0 Notification and Authorization

The NASA Engineering and Safety Center (NESC) received a request from Mr. Daniel Murri (NASA Technical Fellow for Flight Mechanics) to develop an autonomous aerobraking (AA) capability. The NESC received this request on December 15, 2009. An initial evaluation for all phases of this assessment was approved to proceed at the February 4, 2010, NESC Review Board. The assessment plan for Phase 1 was approved by the NRB on April 1, 2010.


The stakeholders for this assessment (all phases) are the NESC (including the technical disciplines of Flight Mechanics; Aerosciences; Passive Thermal; Guidance, Navigation, and Control (GN&C); Software; Loads and Dynamics; and Human Factors) and future NASA programs and projects that may utilize aerobraking (AB).




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		<b>Autonomous Aerobraking (Phase 1)</b>	
		Page #: 9 of 286	

### 3.0 Team List

Name	Discipline	Organization
<b>Core Team</b>		
Dan Murri	NESC Lead, NASA Technical Fellow for Flight Mechanics	LaRC
Richard Powell	Assessment Lead	AMA
Jill Prince	Technical Team Lead	LaRC
Hollis Ambrose	APL Simulation	APL
Angela Bowes	LaRC Simulation	LaRC
David Carelli	APL Simulation	APL
Alicia Cianciolo	LaRC Simulation	LaRC
John Dec	Thermal	LaRC
Hilary Justh	Atmospheric Modeler	MSFC
Jere Justus	Atmospheric Modeler, Flight Mechanics TDT	MSFC/MITS
James Kaidy	APL Simulation	APL
Derek Liechty	Aerodynamics/Aerothermal Environments	LaRC
Forrest Lumpkin	Aerodynamics/Aerothermal Environments	JSC
Robert Maddock	LaRC Simulation	LaRC
Dan O'Shaughnessy	APL Simulation Lead	APL
Chris Pastore	Software Support	LaRC/AMA
Holly Ramey	Atmospheric Modeler	Jacobs
David Skinner	Ephemeris Estimator	KinetX
Benedicte Stewart	Aerodynamics/Aerothermal Environments	JSC/Jacobs
Tom Strikwerda	APL Project Manager	APL
Mark Thornblom	Thermal	LaRC
Robert Tolson	Atmospheric Modeler	NCSU/NIA
Bobby Williams	Ephemeris Estimator	KinetX
Laura Leybold	MTSO Program Analyst	LaRC
<b>Consultants</b>		
Neil Dennehy	NASA Technical Fellow for GN&C	GSFC
Starr Ginn	Loads and Dynamics TDT	DFRC
Cynthia Null	NASA Technical Fellow for Human Factors	ARC
Vicki Regenie	Systems Engineering	DFRC

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		10 of 286	

Name	Discipline	Organization
<b>Peer Reviewers</b>		
Mike Aguilar	NASA Technical Fellow for Software	GSFC
Prasun Desai	Flight Mechanics TDT	LaRC
Dan Lyons	JPL Aerobraking SME	JPL
Cynthia Null	NASA Technical Fellow for Human Factors	ARC
Steve Rickman	NASA Technical Fellow for Passive Thermal	JSC
Dave Schuster	NASA Technical Fellow for Aerosciences	LaRC
Scott Striepe	NASA Aerobraking SME	LaRC
Tung-Han You	JPL Aerobraking SME	JPL
<b>Administrative Support</b>		
Terri Derby	Project Coordinator	LaRC/ATK
Donna Gilchrist	Planning and Control Analyst	LaRC/ATK
Carolyn Snare	Technical Writer	LaRC/ATK

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 11 of 286	


## 4.0 Executive Summary

NASA uses aerobraking (AB) to reduce the fuel required to deliver a spacecraft into its desired final orbit around a target planet or moon with a significant atmosphere. Instead of using the propulsion system to decelerate the spacecraft, AB utilizes aerodynamic drag. While flying through the upper atmosphere of the planet or moon multiple times, the spacecraft maintains a periapsis control corridor such that dynamic pressure and thermal loads on the spacecraft remain within designed parameters. AB has been used four times by NASA: once at Venus and three times at Mars.

Although AB reduces the propellant required to reach the final orbit, the reduction comes at the expense of time (typically 3–6 months), continuous Deep Space Network (DSN) coverage, and a large ground staff. The DSN and ground staff are required to design the maneuvers that the spacecraft executes during AB to keep the spacecraft safe and provide the desired final orbit. The combination of duration, staff, and continuous DSN coverage results in a high cost AB operational phase.

As AB has evolved, the operations have matured to the point where many of the operational decisions being made by the ground staff can now be made autonomously onboard the spacecraft. With the development of autonomous aerobraking (AA), much of the daily ground operations could be minimized thereby reducing the AB phase costs. In addition, by relegating the decision making to the spacecraft, which eliminates the dependence on staff work schedules (e.g., spacecraft can only perform maneuvers during prime shift and must minimize maneuvers during weekends and holidays), AA can reduce risk by allowing the maneuver to be conducted at the optimal apoapsis opportunity and executed even if DSN or other currently required ground elements are unavailable. Another advantage of AA is that it could provide the ability to handle multiple AB assets at the same time that would otherwise not be economically feasible using the traditional ground-based operational approach.

Phase 1 of this study investigated the technical capability of transferring the processes of AB maneuver decision making (currently performed on the ground through the DSN and an extensive workforce) to an efficient flight software algorithm onboard the spacecraft. To accomplish this, highly accurate aerodynamic and thermal models for a representative spacecraft were developed for both the onboard algorithm known as Autonomous Aerobraking Development Software (AADS) and a ground-based “truth” simulation developed for testing purposes. Autonomous Ephemeris, Atmosphere, and Maneuver Estimators were developed and incorporated into AADS. Previous AB mission experience indicates that an increase in the error of the predicted time of periapsis passage requires frequent (daily) ephemeris updates from the ground using DSN. One goal of the AA study is to develop an Ephemeris Estimator that can provide state estimates within acceptable tolerances for more than a week before requiring a ground update, thus eliminating the need for continuous DSN coverage.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 12 of 286	


For Phase 1, AADS was tested in simulated AB missions at three destinations, each requiring a unique AB corridor type. At Mars, the corridor was based on aerodynamic heating rate. Due to its proximity to the Sun, the AB mission at Venus utilized a solar-panel temperature corridor. Finally, an altitude corridor was used at Titan. Nominal and off-nominal AB scenarios were simulated and compared with the reference simulation.

Products of Phase 1 included several models and algorithms that were integrated within AADS. An Ephemeris Estimator was developed that used an efficient, easily implemented Runge-Kutta integration scheme, a high order gravity field model, third body gravitational effects, and provided the required accuracy for 7 days. An atmospheric estimator was demonstrated that used traditional atmospheric estimation algorithms that provided periapsis density and density scale height estimates that were adequate for corridor maintenance maneuver calculations. A thermal response algorithm was generated that predicted the maximum temperature of the spacecraft given the periapsis density from the Atmosphere Estimator. The entire system was embedded into two high fidelity simulations that used detailed models of flight subsystems and have vast heritage as simulation tools.

Results of the Phase 1 simulation analysis show that it is feasible for AADS to provide AA control of a spacecraft with ephemeris updates less than once per week at all three sampled destinations. These results have been demonstrated in the presence of atmospheric perturbations and sensor (e.g., inertial measurement unit (IMU)) measurement errors. Longer intervals between ground updates may be possible, as a 14-day ground update interval with atmospheric perturbations at Mars, Venus, and Titan has been demonstrated. This 7-day update cycle meets the goal set for AADS and could significantly reduce the DSN and ground staffing requirements.

Future work has been identified and will include enhancements to AADS, improved capabilities of the simulation tools, and additional stress testing of AADS in both high fidelity simulations.



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 13 of 286	

## 5.0 Assessment Plan

The complete development of the AA capability occurs in four phases. This report summarizes the activities of Phase 1, which included software development sufficient to demonstrate the viability of the proposed AA approach, detailed below.

- Atmospheric, aerodynamic, and thermal models for a representative AA spacecraft, based on Mars Reconnaissance Orbiter (MRO).
- The AADS package, which includes ephemeris and Atmosphere Estimator s, and a corridor control maneuver calculator.
- Two simulation environments for testing the AADS.
  - A 3-degree of freedom (DOF), stand-alone Program to Optimize Simulated Trajectories II (POST2)-based AB simulation used to rapidly develop and mature the AADS software.
  - A 6-DOF, high-fidelity MErcury Surface, Space ENvironment, GEOchemistry and Ranging (MESSENGER)-based simulation that more closely represents the flight operating environment and sensor characteristics (e.g., IMUs and star trackers).


Both simulation environments were used for proof-of-concept testing and preliminary studies of the AADS algorithms and the AA approach in general.

Until this assessment, the method of ephemeris estimation utilized data provided by continuous DSN coverage and ground-based software to provide frequent updates to the stored onboard ephemeris. Phase 1 included simulation testing of AA at Mars, Venus, and Titan.

Phase 2 will explore more sophisticated schemes for the ephemeris and atmosphere estimation as a means of improving AADS robustness and performance. Improved modeling and error checking will be implemented and additional off-nominal stress testing of AADS will be performed.

Phase 3 will incorporate all of the AA software onto a flight-like processor to study the implications of operating the AADS in a real-time environment. This phase will quantify the computational resources required to support AADS in a flight mission.

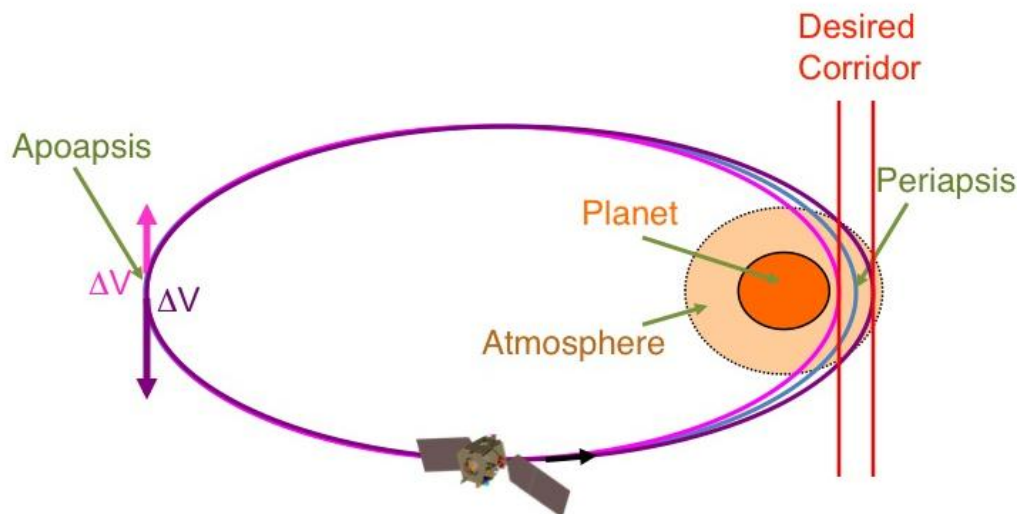
In Phase 4, the AA software will be installed onto a spacecraft that will use AB, and then AA will operate during flight in a shadow-mode, where all the steps for AA are performed but the commands are not executed. The onboard-determined commands will be validated against AB decisions made by the ground staff.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 14 of 286	

## 6.0 Problem Description and Proposed Solution


### 6.1 Conventional AB

NASA uses AB to reduce the fuel required to deliver a spacecraft into its desired final orbit around a target planet or moon with an appreciable atmosphere. Rather than using the propulsion system to decelerate the spacecraft after the initial orbit insertion, AB utilizes aerodynamic drag induced by repeated passes through the atmosphere. Small propulsive maneuvers at apoapsis are used to control the altitude at periapsis to maintain the spacecraft within its designed corridor (see Figure 6.0-1). The periapsis control corridor may be defined by a range of dynamic pressure, a heat rate indicator, atmospheric density, or periapsis altitude, but typically the corridor is constrained by the thermal limit of a spacecraft component. Using a multiple-pass approach enables the spacecraft's loads to remain within designed limits while gradually reducing the apoapsis to achieve the appropriate final orbit.



*Figure 6.0-1. A Spacecraft using Apoapsis Maneuvers to Control Periapsis Altitude during AB*

NASA has used AB 4 times. The first flight demonstration of the technique was the Magellan spacecraft at Venus where AB was completed at the end of the primary mission. AB successfully reduced the spacecraft's orbital period from just over 3 hours to just under 2 hours in 70 days and led to the use of AB as a mission-enabling capability for three Mars spacecraft: Mars Global Surveyor (MGS), Mars Odyssey, and MRO.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 15 of 286	

## 6.2 Benefits of AA

Although AB reduces the propellant required to reach the final orbit and as a result, reduces launch mass, this reduction comes at the expense of time, continuous DSN coverage, and a large ground staff required for AB operations. For example, Mars Odyssey operations required the expertise of trajectory analysts, thermal modelers, aerodynamicists, atmospheric scientists, and systems engineers in multiple locations, to analyze the necessity of an apoapsis maneuver for the following day. This is in addition to the numerous engineers required to build and upload the spacecraft sequence. The process took nearly 12 hours on each day of the 77-day AB phase. Similar analysis was completed every day of the 149-day MRO AB phase, albeit with a reduced staff due to the large thermal margin on the spacecraft. The combined cost of DSN and workforce results in an expensive operational phase.


The development of AA would enable much of the daily ground operations to be moved to the spacecraft, thus reducing the cost of the AB phase by several million dollars [ref. 1] for a single mission.

AA is also an enabling technology for future multiple-vehicle missions. Not only will small communications and science spacecraft benefit from the cost savings of AA, but it should be considered essential to establishing the infrastructure required both prior to and during human missions to Mars. AA enables a cost-effective way to establish sufficient and redundant satellite communication, weather monitoring, and global positioning systems at Mars. AA enables cost-effective means for stereo and three-dimensional (3-D) observations at any body in the solar system with an atmosphere. In addition, AA is capable of handling multiple orbiting and AB assets simultaneously, a task which would not be economically feasible using a traditional ground-based operational approach.

## 6.3 AA Approach

The concept of AA has been studied over the past decade [refs. 2–6]. Steps toward AA were taken when the periapsis timing estimator tested during the Mars Odyssey (2001) AB phase was implemented on MRO (2006). However, further development to include the maneuver execution capability is required to significantly reduce ground operational costs and is the subject of this NESC AA development activity.

To enable a fully AA system, the spacecraft must calculate and predict its own ephemeris. All drag pass activities are referenced to a periapsis time and all periapsis altitude correction burns are performed at apoapsis. As a result, successful AB requires accurate knowledge of these orbital extrema. Using the subsequent orbit's predicted periapsis, the spacecraft estimates the next predicted periapsis density using an onboard atmospheric model. If predicted density causes the corridor control value (e.g., dynamic pressure, heat rate, temperature) to fall outside the specified range, then an onboard algorithm would determine the maneuver direction, magnitude, and epoch that will be required to move the orbital periapsis back into the corridor.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 16 of 286	


The spacecraft then autonomously executes the required maneuvers. Moving the daily maneuver assessment to the spacecraft saves significant cost in staff and DSN usage.

Additionally, because the spacecraft maneuver decision is no longer tied to the work schedule of ground personnel (e.g., previously, maneuvers were ideally performed during staff's prime shift and avoided during weekends and holidays), AA reduces risk, as the maneuver could be conducted at the optimal apoapsis and executed even if DSN or other currently required ground elements were unavailable. Ground-based weekly activities such as updates to the corridor, model parameters, and/or overall mission strategy would continue to be performed from Earth. In addition, some one-way communication via DSN may be expected with the use of AA to monitor the health and safety of the spacecraft. This health monitoring of the spacecraft would be relatively infrequent and does not impede the anticipated cost savings of AA.

The Phase 1 AA study investigated the technical capability of transferring the processes of AB maneuver decision making (currently performed on the ground through the DSN and an extensive workforce) to an efficient flight software algorithm onboard the spacecraft; to do so requires highly accurate models and test simulations.

Therefore, aerodynamic and thermal models of the representative MRO spacecraft were developed; they are described in Sections 7.3.1.2 and 7.3.2.4 respectively (with more detailed descriptions in Appendices C and D). Baseline reference simulations were created at the three destinations: Mars, Venus, and Titan. Reference simulations are those trajectory simulations that are typically performed during the design phase of the mission and serve as the nominal trajectory to which the operational AB trajectory is compared. This reference simulation provides the "glideslope" or metric to which AB status is measured. In designing the AA capability, an additional "truth" simulation was developed. This "truth" simulation is the operational AB trajectory that has full knowledge of the atmosphere, planet, vehicle aerodynamics, etc. As there is little "truth" Mars, Venus, or Titan data available, "truth" in AA is still modeled data (e.g., Mars-Global Reference Atmospheric Model (Mars-GRAM), Venus-GRAM, Titan-GRAM, etc.). Both the reference simulation and the "truth" simulation were developed using two independent tools: POST2 at Langley Research Center (LaRC) and a MESSENGER-based simulation at the Johns Hopkins University (JHU)/Applied Physics Laboratory (APL), the Autonomous Aerobraking High Fidelity Simulation (AAHFS). The POST2 and AAHFS results are described in Section 7.5 with more information in Appendices H and I. Mission design techniques in constructing the reference simulation are described in Section 7.2 and Appendix B.

Finally, the AADS module is a software package onboard the simulated spacecraft with access to the spacecraft IMU data and uplinked parameters but no knowledge of the spacecraft environment. (Phase 2 will address incorporating thermocouple temperature data from the spacecraft to AADS to more accurately predict spacecraft temperature.) The AADS requires accurate onboard models to estimate 1) the atmosphere conditions at periapsis (i.e., maximum

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 17 of 286	

expected density and corresponding atmospheric scale height), 2) the spacecraft ephemeris, and 3) a thermal model estimating the maximum expected temperature of the spacecraft (for use at Venus only). Using these models, AADS determines whether or not a maneuver is required, and if necessary, the magnitude of that maneuver and its location within the orbit, which are sent to the spacecraft. The maneuver logic is an adaptation of the mission design analysis developed at LaRC for the Mars Odyssey AB mission and refined for MRO mission design and operations. The key AADS models, the Atmosphere Estimator and Ephemeris Estimator, are described in Section 7.3.2.1 (with detail in Appendix F) and Section 7.3.2.2 (Ephemeris Estimator User Guide is in Appendix G). AADS system integration is described in Section 7.4.

The overall system goal for the AADS is to demonstrate that the appropriate maneuvers' magnitudes and directions are calculated and execution times are sufficient to maintain a desired flight corridor for up to 1 week compared to the "truth" simulation.

For Phase 1, AADS was tested using simulated AB missions at three destinations, each requiring a unique AB corridor type. At Mars, the corridor was based on heat rate indicator ( $1/2 * \text{atmospheric density} * \text{velocity}^3$ ). Due to the proximity to the Sun, the AB mission at Venus utilized a solar-panel temperature corridor. Finally an altitude corridor was used for AA analysis at Titan. Nominal and off-nominal AB scenarios were simulated and compared with the "truth" simulation. Overall AB mission metrics were compared to the reference simulations.


Follow-on phases are planned that will improve accuracy of models developed in Phase 1; execute rigorous testing of AADS within the mission design simulation used at LaRC, POST2, as well as the APL MESSENGER-based AAHFS; port AADS to a hardware-in-the-loop testbed; and ultimately demonstrate the operability of AA in flight.

## 7.0 Data Analysis

The following sections contain details of the models used and the analysis performed for AA development. Reference simulations were developed that incorporate the "truth" models. The reference simulations provide the overall AB statistics that AADS will attempt to match. Additionally, a "truth" simulation, based on the reference simulation, is used operationally. The AADS data is compared to this "truth" simulation at each orbit periapsis. This comparison provides the confidence in the capability of AA and is provided below.

### 7.1 Reference and "Truth" Simulations


Reference simulations were performed at Mars, Venus, and Titan. These reference simulations are the basis of the "truth" simulation used as a comparison of the accuracy of AADS. Both the reference simulation and the "truth" simulation include full knowledge of the environment, including atmospheric, planetary, and spacecraft properties. The reference simulation utilizes full knowledge of these properties along with a maneuver strategy (to make a maneuver every day or every 3 days, etc.) to obtain the most accurate estimate of realistic AB mission

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 18 of 286	

progression, including glideslope, mission duration, number of maneuvers, amount of fuel required, etc. The “truth” simulation also utilizes full knowledge of the aforementioned properties, but rather than an internal maneuver strategy, it calls the AADS after each drag pass and uses the maneuver provided by AADS at each necessary apoapsis.


Using AADS, maneuver decisions are made in the “truth” simulation based only on knowledge passed to the algorithm by the spacecraft (e.g., parameters uploaded to the spacecraft including initial state and ephemeris and onboard data from the IMU (e.g., acceleration data). The models within AADS predict the atmosphere based on IMU data and make estimates about the subsequent periapsis conditions to estimate whether a maneuver is needed. In the event that one is needed, the magnitude and time at which the maneuver should occur is sent to the spacecraft. The logic within AADS is described in detail in Section 7.4.

”Truth” simulations using both POST2 and AAHFS were developed to analyze AADS performance. The “truth” simulations are described in detail within this document. For reference a table of models contained in both “truth” simulations and in AADS is shown in Table 7.1-1.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		19 of 286	

*Table 7.1-1. Models Contained in "Truth" Simulations and AADS. Green boxes indicate that those models in the same row are identical. A yellow box within the row indicates an identified difference in the model or data used. Red boxes indicate that the model in question is not applicable.*

Model	POST2 "Truth"	AAHFS "Truth"	AADS
Gravity	Mars and Venus gravity fields are truncated to 21x21	Mars and Venus gravity fields are truncated to 21x21 (UT Methodology is used for computing accelerations due to gravity)	Mars and Venus gravity fields are truncated to 21x21
Atmosphere	GRAM models directly integrated with simulation software	GRAM models directly integrated with simulation software	Atmosphere Estimator uses acceleration data to estimate density and scale height based on a simplified (exponential) atmosphere
Planetary Ephemerides	Details in Appendix A	Details in Appendix A	Details in Appendix A
Planetary Shapes, Constants, and Orientations	Details in Appendix A	Details in Appendix A	Details in Appendix A
Solar Radiation Pressure	Based on simplified spacecraft geometry and attitude	High-fidelity model based on spacecraft geometry and true (6DOF) attitude (not used for Phase 1 analyses)	Based on simplified spacecraft geometry and attitude
Aerodynamics	Details in Appendix C	Details in Appendix C	Atmosphere Estimator utilizes same aerodynamic coefficient tables but with modified/simplified equations
Thermal Response Surface	Utilized for monitoring purposes only	To be implemented in Phase 2	Developed for Mars and Venus applications
Spacecraft Shape & Mass Properties	Details in Appendix A	Details in Appendix A	Details in Appendix A

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		20 of 286	

Model	POST2 "Truth"	AAHFS "Truth"	AADS
Sensors	To be implemented in Phase 2	IMU Star Tracker	Not applicable
Actuators	To be implemented in Phase 2	Thrusters Reactions Wheels	Not applicable
Guidance and Control	To be implemented in Phase 2	Attitude /Rate Estimation Wheel Control Thruster Control	Not applicable
Maneuver Execution	Square Pulse Finite Burn	Full thrust profile based on spacecraft thruster model	Not applicable

### 7.1.1 POST2


The POST2 is a generalized point-mass, rigid-body, discrete-parameter targeting and optimization trajectory simulation program based on the POST software initially developed in the 1970s by LaRC in partnership with the Martin Marietta Company to support Space Shuttle development. Throughout the years, POST was continually upgraded and modified by LaRC and Lockheed Martin (LM) to support a large variety of aerospace vehicle development and mission flight operations through trajectory simulation, flight dynamics analyses, vehicle system development and evaluation, and integrated system performance assessments.

POST2 provides state-of-the-art simulation software of endo- and exo-atmospheric flight about a central body to support launch, orbital, and entry vehicle design, development, testing, assessment, and flight operations for either a single vehicle or multiple vehicles working independently or in tandem (e.g., orbital rendezvous or intercept). POST2 has been instrumental in planetary entry, descent, and landing design, as well as development, evaluation, and operations for robotic and human systems. It has been used for the AB mission design and operations for Mars Odyssey and MRO.<sup>1</sup>

Though POST2 can be run in both a 3-DOF and 6-DOF mode, all of the AA Phase 1 work utilized the POST2 3-DOF capability to capitalize on the existing heritage AB trajectory software. In the heritage reference simulation, POST2 was tailored by including the specific planet, gravity, atmosphere, spacecraft, aerodynamic, and third body perturbation models and

<sup>1</sup> Striepe, S.A. *et al.*, "Program To Optimize Simulated Trajectories (POST II): Volume 2, Utilization Manual," Martin Marietta Corporation, 2004 and Brauer, G. L. *et al.*, "Program To Optimize Simulated Trajectories (POST): Volume 1, Formulation Manual," Martin Marietta Corporation, 1990.



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 21 of 286	

incorporated logic required for corridors and maneuver strategies required for AB mission design.

### 7.1.2 AAHFS


The AAHFS is a 6-DOF simulation based on the flight software and truth models previously developed for the JHU/APL-designed MESSENGER spacecraft, currently orbiting Mercury [ref. 7]. Although this is the first time the simulation has been used for AB, the capability demonstration has been developed in the MATLAB/Simulink/Real Time Workshop (RTW) environment. It is comprised of the existing MESSENGER algorithms and software, to which the AB flight system and truth model test bed algorithms have been added. This approach permits rapid study of the AADS and, in the end, generates a high degree of confidence in the technology as a precursor to implementation in a flight program. Another important benefit to developing the software in this way is that the end product is suitable for use in a flight processor, as this study uses the same development environment/process as the MESSENGER flight software.

Like POST2, the AAHFS has included all of the planet and vehicle-specific models used in the POST2 for simulation comparisons.

## 7.2 AB Mission Design Methodologies

As mentioned previously, AB reference simulations utilize full knowledge of environmental properties, along with a flight corridor and maneuver strategy, to obtain an accurate model of realistic AB mission. Both the POST2-based reference and “truth” simulations were used as a basis of comparison and assessment of AA performance using AADS. The reference trajectory was designed to achieve the final desired orbit conditions using AB while maintaining the mission operational constraints and the margin required for the spacecraft design limits [ref. 8]. Desired final orbit conditions can include altitude, inclination, argument of periapsis, and longitude of the ascending node required to attain a specific local mean solar time (LMST) orientation, or combinations of these parameters. However, for this phase of the study all simulations ended on a specified apoapsis altitude.

Mission operational constraints and margin are designed to protect the spacecraft throughout the AB mission and depend on the physical properties of the spacecraft as well as required final orbit conditions. The operational constraints may consist of spacecraft thermal and structural limits (e.g., freestream heating rate, solar array temperature, dynamic pressure, power, attitude, and capability to handle atmospheric density fluctuations) and orbit lifetime requirements, maneuver frequency restrictions, maneuver magnitude limitations, and required propellant remaining post AB to achieve mission objectives. For the study at Mars, the constraints were set similar to those utilized in the MRO AB mission. For the Venus simulation, where temperature of the spacecraft dominates, a temperature range that generated approximately the same AB duration as the Mars mission was selected. It is somewhat arbitrary because it is recognized that the MRO spacecraft is not designed for AB at Venus. Finally, no AB mission has been attempted at Titan, therefore


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 22 of 286	

without specific mission constraints, the decision was made to start AB in an 8-hour orbit period and make the AB duration last at least 2 months utilizing an altitude corridor.

The reference simulations for this study begin after walk-in (a series of orbits intended to incrementally lower the spacecraft periapsis into the sensible atmosphere) and last until the final specified orbit conditions are achieved. A corridor, based on a specific operational constraint as described previously for each destination, is designed to keep the spacecraft within the appropriate design margins. Maneuvers are performed at apoapsis that raise or lower periapsis to maintain the spacecraft within the pre-determined corridor. The upper limit of the corridor is determined by the required operational constraint margin to ensure spacecraft safety, and therefore defines the maximum AB rate (i.e., shortest duration) that can be achieved within that constraint margin. For the corridor constraints selected at Mars and Venus, the lower the spacecraft is in the corridor, the higher altitude the atmospheric passes, the lower the  $\Delta V$  from aerodynamic drag, resulting in overall increased AB mission duration. The lower corridor limit may be set to reduce the frequency of maneuvers required to stay in the corridor and/or to maintain the maneuver magnitudes greater than some minimum threshold. A particular lower corridor limit may also be set to ensure the AB rate is such that the desired final orbit conditions can be reached by a certain time. For instance, in the case where there is a desired final orbit LMST, the initial orbit node must have enough time to precess with respect to the Sun to produce the desired LMST. The amount of time required for the precession varies as a function of the initial orbit conditions, current orbit conditions, central body, gravity, atmospheric environment, and other forces such as third body perturbations and solar radiation pressure. AB either too quickly or too slowly could cause the final desired orbit apoapsis altitude to be reached at a different LMST than required. In the case where a periapsis altitude corridor is used at Titan, the roles of the upper and lower corridor constraints are reversed (i.e., lower corridor limit defines the maximum AB rate rather than the upper corridor limit since the lower the spacecraft is in the corridor means the atmospheric passes are lower and deeper within the atmosphere.)

The corridor limits can change as a function of time since the specific conditions that the spacecraft experiences across the duration of AB are largely a function of orbit geometry. A maneuver target, specified as a percentage of the corridor width, is set and can vary with time or orbit geometry as well. The minimum amount of time allowed between maneuvers is also set. Whether or not a maneuver is performed when it is “allowed” is based on predicting ahead by the minimum time between maneuvers plus one additional day at Mars and Venus, or for Titan, by predicting ahead two orbits. If a corridor violation occurs at any time during the predicted time period, then a maneuver will be performed at the next allowable apoapsis.

Operationally, the reference trajectory is used to establish the spacecraft flight design corridor each week and can be adjusted if necessary during the flight to accommodate observed atmosphere fluctuations. The daily operations are used to determine any required periapsis adjust maneuvers to maintain the spacecraft within the design corridor.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 23 of 286	

Appendix B contains a description of the reference simulation analyses at Mars, Venus, and Titan including all assumptions.


## 7.3 AB Models

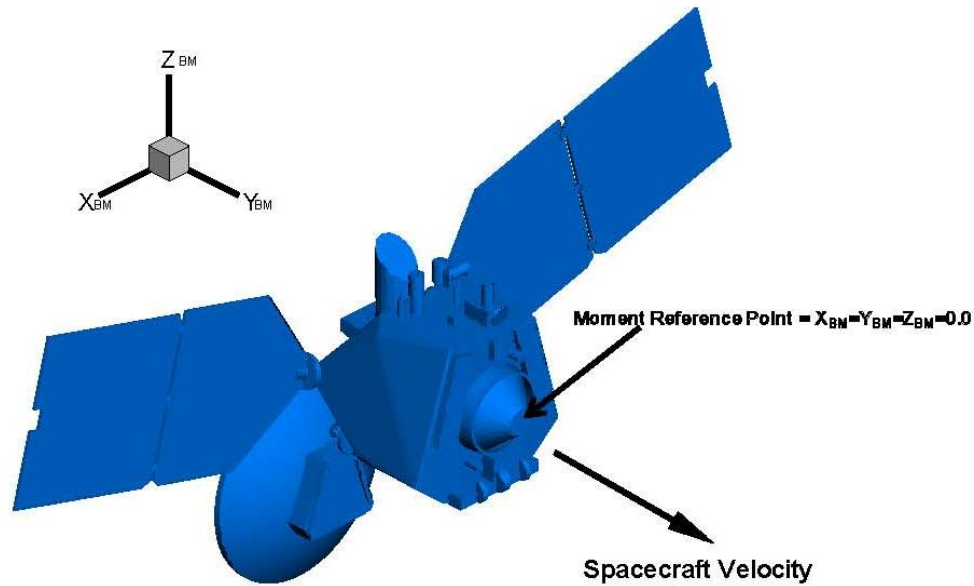
Two types of models were developed in Phase 1 to represent an AB spacecraft in the reference and “truth” simulations (“truth” models) and within AADS (onboard AA models). “Truth” models were incorporated into the reference and “truth” simulations to create a flight environment for comparison. It is understood that they are models and not flight data, but for lack of enough available flight data (specifically for Venus and Titan) and because historically these models have provided a reasonable estimate for flight data, these models are considered truth for AA development purposes and “truth” remains in quotation marks. The “truth” models include a standard planet gravity model, an atmosphere model, and models of the spacecraft and its aerodynamic and aerothermodynamic properties. These are described in Section 7.3.1. In Section 7.3.2, the onboard AA models are discussed: the Ephemeris, Atmosphere, and Maneuver Estimators and thermal model (for Venus only).

### 7.3.1 “Truth” Models

#### 7.3.1.1 Spacecraft Model

The MRO spacecraft shape and ballistic coefficient were chosen for the AA study (see Figure 7.3-1 and Table 7.3-1). Utilizing the MRO spacecraft provided a readily available spacecraft model as well as an aerodatabase and thermal model developed for MRO's use during AB operations. The aerodatabase and thermal model have been flown operationally and have already been correlated with flight data at Mars; thus, they can easily be integrated into the Mars AA technology development and verified. Additionally, these models provide good starting positions for adapting the aerodatabases and thermal models for use at Venus and Titan. However, there is a disadvantage to using the MRO spacecraft shape and ballistic coefficient: because the MRO spacecraft was designed for AB specifically at Mars, it does not handle the corridor conditions at Venus and Titan as well as a spacecraft that is designed exclusively for use at those locations. Details of the spacecraft model can be found in the Autonomous Aerobraking Planetary and Constants Document (AA PCMD) located in Appendix A.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		24 of 286	




*Figure 7.3-1. MRO Spacecraft used for AA Study Simulation*

*Table 7.3-1. MRO Mass Properties*

<b>Mass (kg)</b>		1395
<b>Reference Length (m)</b>	<b>l<sub>ref</sub></b>	13.6
<b>Reference Area (m<sup>2</sup>)</b>	<b>s<sub>ref</sub></b>	37.12
<b>Center of Gravity (m)</b>	<b>X<sub>cg</sub></b>	1.08106
	<b>Y<sub>cg</sub></b>	0.00101
	<b>Z<sub>cg</sub></b>	0.00462
<b>Aerodynamic reference point (m)</b>	<b>X<sub>ref</sub></b>	1.130
	<b>Y<sub>ref</sub></b>	0
	<b>Z<sub>ref</sub></b>	0
<b>Moments of Inertia (kg-m<sup>2</sup>)</b>	<b>I<sub>xx</sub></b>	1800
	<b>I<sub>yy</sub></b>	2400
	<b>I<sub>zz</sub></b>	2600
<b>Products of Inertia (kg-m<sup>2</sup>)</b>	<b>I<sub>xy</sub></b>	0
	<b>I<sub>yz</sub></b>	0
	<b>I<sub>zx</sub></b>	0

### 7.3.1.2 Aerodynamics and Aerothermodynamics

The direct simulation Monte Carlo (DSMC) method of Bird [ref. 9] was used in the computation of the aerodynamic and aeroheating databases for the AA simulation study. This method has

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 25 of 286	


become the standard for simulating low-density gas dynamics. Traditional computational fluid dynamics (CFD) methods are incapable of modeling flows experienced during AB, because assumptions made in developing the differential equations on which they are based break down under rarefied conditions. In contrast to continuum CFD methods, the DSMC method performs a direct physical simulation of the gas at the molecular level. In the DSMC simulation, molecules are tracked in space and time, accounting for both gas-surface interactions and intermolecular collisions. The DSMC Analysis Code (DAC) software [refs. 9-13] represents NASA's state-of-the-art implementation of the DSMC method for analyzing rarefied flows.

The aerodynamic and aeroheating databases are developed by simulating steady-state flows at various altitudes (density ranges) and orientations along the expected trajectories for each atmosphere under consideration. Once completed, the databases are used by the reference simulation and thermal modeling to determine the aerodynamic forces and heating on the vehicle throughout atmospheric flight. In the current study, no new modeling was required to be developed beyond the standard DSMC practices. (Refer to Appendix C for a complete discussion on physical models used, flow conditions, and assumptions made.)

### 7.3.1.3 "Truth" Atmosphere

The Mars-GRAM is an engineering model of the Mars atmosphere used in the AA reference simulations. Mars-GRAM is based on the Ames Research Center (ARC) Global Circulation Model. The model includes spatial (altitude, latitude, and longitude), seasonal, and diurnal variations along with perturbation (tides, gravity waves, etc.) variations. The model also assimilates data from instruments on recent Mars missions and measurements from past Mars AB missions to generate a realistic representation of the atmospheric conditions. For example, the model has extensive heritage with Mars mission design and mission operations and has been used in much of the mission design simulation work associated with Mars. A detailed description of Mars-GRAM 2010 is given in Appendix E.

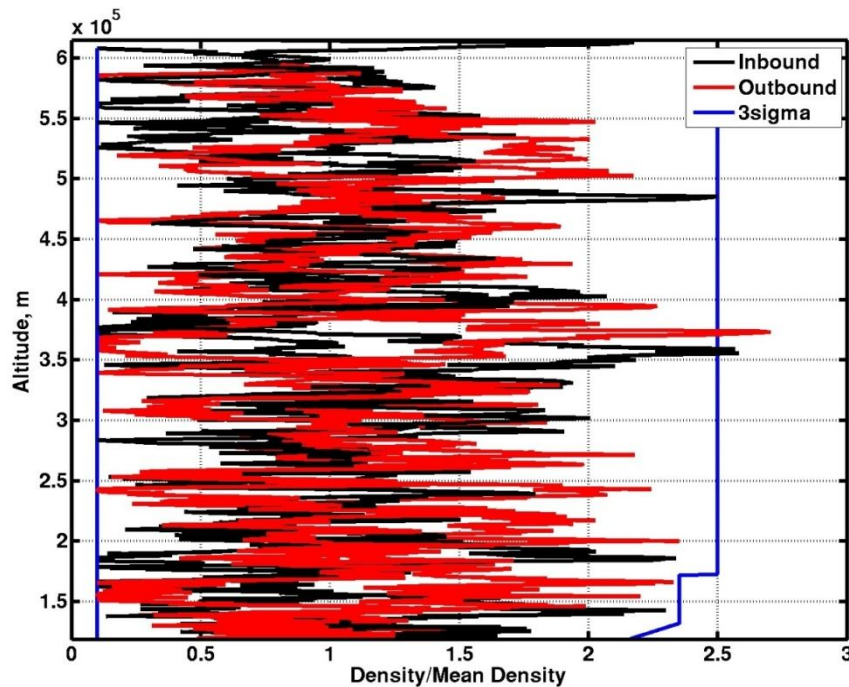
Previous versions of Mars-GRAM were less accurate when used for sensitivity studies for off-nominal conditions. As part of the AA Phase 1 work, analysis was performed to find possible solutions to the differences seen in the sensitivity studies. Mars-GRAM was evaluated at locations and times of Thermal Emission Spectrometer (TES) limb observations, and adjustment factors (ratio of observed TES density to Mars-GRAM density) were determined. Details of the adjustment factor requirements and the development of these factors are given in Appendix E. The addition of the adjustment factors has led to better correspondence to TES Limb data from 0 to 60 km and better agreement with MGS, Mars Odyssey, and MRO data at approximately 90–135 km. Results demonstrating the improvement of Mars-GRAM 2010 results at lower altitudes and AB altitudes are given in Appendix E and generate more realistic atmospheres for the Mars AA "truth" simulation. Improved simulations of the atmospheric density utilizing Mars-GRAM 2010 are vital to developing the onboard atmospheric density estimator for the AA Development Plan.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 26 of 286	

Titan-GRAM and Venus-GRAM were used to generate density data sets for AB design reference simulation trajectories at those destinations. Despite the much smaller data set with which to verify the models, these data sets still provide altitude profiles (both vertical and along a trajectory) GRAM perturbations (tides, gravity waves, etc.) and provided density and scale height values. (See Appendix E for more information.)

All of these atmospheric models have input parameters that allow the user to control the level of density perturbation during an atmospheric pass. The nominal perturbations that would be produced for Mars, Venus, and Titan for two representative orbits are shown in Figures 7.3-2, 7.3-3, and 7.3-4, respectively, as the ratio of the perturbed density to the mean density at that location and time. Each atmospheric pass has an inbound leg (pre-periapsis) and an outbound leg (post-periapsis). As can be seen in the figures below, the density profile sensed on the inbound leg can be significantly different than that of the outbound leg.

The modeled perturbations for Mars are larger than the modeled perturbations for Venus and Titan. Because there have been three AB missions at Mars, there is now sufficient data to develop a Mars atmospheric model based entirely on flight data rather than using Mars-GRAM perturbed atmospheres. This is planned as part of the Phase 2 effort. Because of the limited flight data available for Titan and Venus, Phase 2 will look at increasing the level of perturbations that is provided by the models at Venus and Titan.



*Figure 7.3-2. Mars-GRAM 2010 Density Perturbations*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
27 of 286

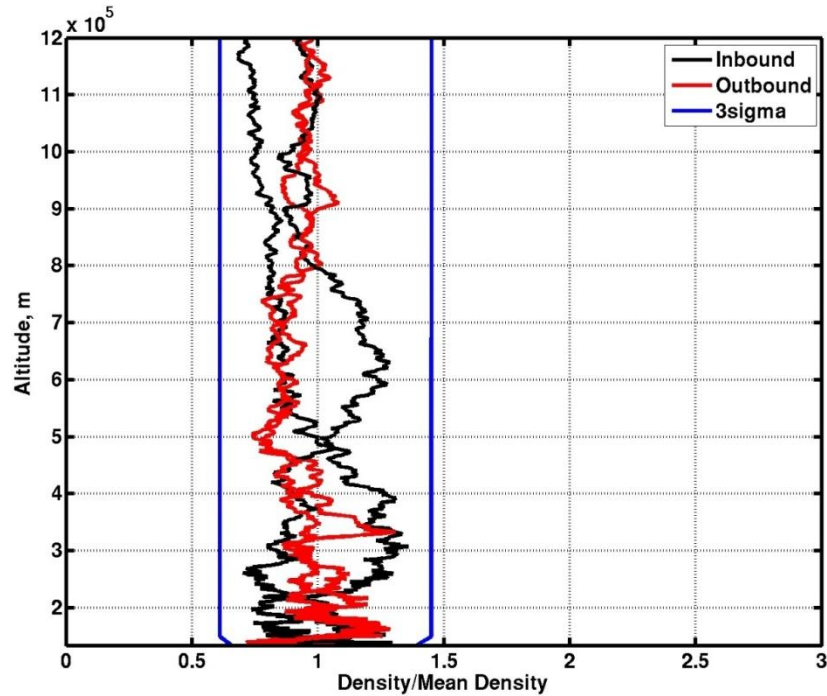


Figure 7.3-3. Venus-GRAM Density Perturbations

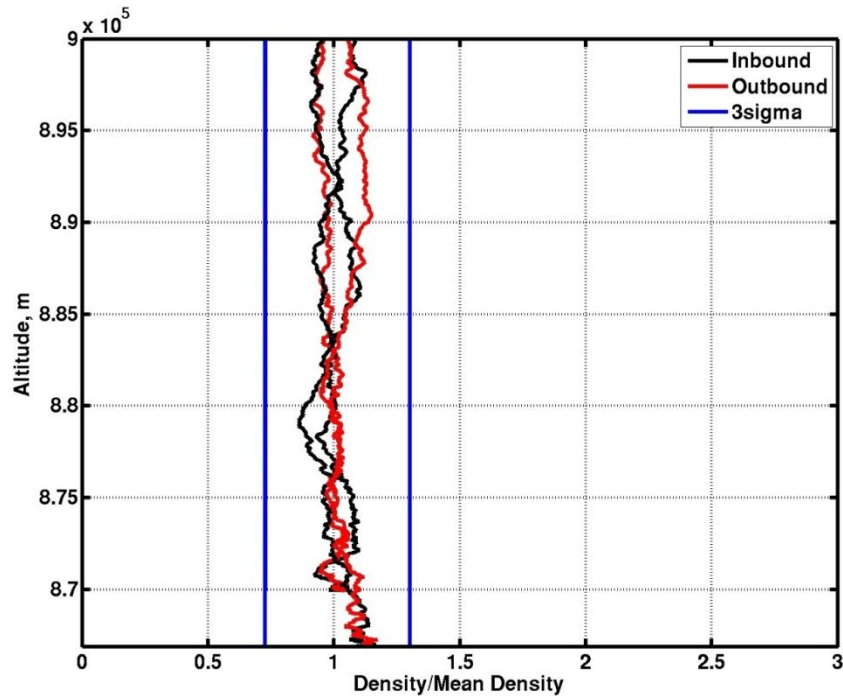



Figure 7.3-4. Titan-GRAM Density Perturbations

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 28 of 286	

### 7.3.2 Onboard AA Models


The following section describes the models that comprise the AADS flight software module. The AADS module has access to only select parameters passed to it by the spacecraft. It uses that information to calculate a maneuver direction, magnitude and epoch of maneuver required to maintain a specified corridor.

#### 7.3.2.1 Autonomous Atmosphere Estimator

The purpose of the onboard Atmosphere Estimator is to determine values of relevant atmospheric parameters for the next AB pass by utilizing IMU accelerometer and gyroscope data from previous AB passes. The parameters are used to predict atmospheric density in the vicinity of the next periapsis, which is subsequently used to determine the maneuver required to raise or lower the periapsis for the next AB pass. Previous AB missions have used a variety of atmospheric parameterizations to analyze IMU data, but in the final analysis they reduce to an estimate of density at periapsis and a density scale height. In addition, radio tracking data (from DSN) have been used to estimate a single atmospheric parameter for each AB pass. After appropriate scaling, these two methods have agreed within 2 percent to 5 percent on the prediction of density at periapsis, well within the uncertainty of each method. Therefore, the methods that use IMU data have been validated and are likely candidates for onboard atmospheric density estimation. However, these methods benefited from having an accurate estimate of the spacecraft's orbit so that the altitude of the vehicle during a pass could be accurately correlated with the measured acceleration due to the atmospheric forces. Onboard atmospheric estimation methods will likely have to be robust enough to account for ephemeris errors and this is probably the issue that will limit the duration of AA before the ephemeris requires updating based on radio-tracking data. Finally, during previous Mars AB operations, several schemes were used to estimate atmospheric parameters from IMU data and human experience was used to select the "best" solution to be used for operations. Thus, onboard estimation methods will make decisions based on knowledge gained from human experience.

During Phase 1, extensive studies were performed using IMU data from the three Mars missions (MGS, Mars Odyssey, and MRO). There are no IMU AB data for Venus (there were no accelerometers onboard Magellan) or Titan (there have been no previous AB missions). Searches were performed to determine the data ranges and atmospheric parameterizations that produced the best estimate of the density at the next periapsis location. Density at the next periapsis was estimated using combinations of different numbers from previous orbits and different filtering methods. In addition, traditional, symmetric parameterizations that depend only on altitude were compared with unsymmetrical parameterizations that depend on time and altitude. To find a possible way of accommodating ephemeris errors, parameterizations that depend only on time were studied in an approximate manner. To evaluate the influence of large ephemeris errors completely would require the complete reanalysis of over 1,500 AB orbits from the three missions, an activity reserved for Phase 2. At the end of all these studies (in Phase 1), a



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 29 of 286	

software system was designed for onboard atmospheric estimation that (1) uses the simplest reliable atmospheric parameterization and (2) is completely parameterized so that all constants which might vary with planet, latitude, local solar time, altitude, or season are constants in only two routines. This provides the ability for ground operations to occasionally update the algorithms to maximize performance during the course of an AB mission. (For more information, see Appendix F.)


### 7.3.2.2 Autonomous Ephemeris Estimator

Because all events that occur in each AB orbit depend on precise knowledge of the location and timing of the spacecraft in orbit, an essential element of the AA software is an algorithm that calculates the spacecraft ephemeris onboard for several days. The Ephemeris Estimator provides an on-going running estimate of the spacecraft orbital state as well as a prediction of the orbital state of each upcoming orbit throughout the AB mission phase. The Ephemeris Estimator is designed to numerically integrate the spacecraft orbital equations of motions about the planet using model parameters and initial conditions that are uplinked from ground-based processing. Once initialized, the Ephemeris Estimator processes onboard accelerometer measurements to account for the accelerations due to atmospheric drag and propulsive maneuvers to produce a trajectory prediction through the next drag pass. The Ephemeris Estimator can be re-initialized by up-linked ephemeris data from DSN, so that a more accurate prediction can be maintained. The longer the Ephemeris Estimator can maintain an accurately predicted orbital state, the longer the autonomous spacecraft operation can be sustained without a revision of the orbital state from the ground using DSN.

The AADS provides the Ephemeris Estimator with high-level interfaces to trajectory-related data stored in the spacecraft memory, to the spacecraft accelerometer measurements, and to spacecraft event flags. The parameters include the initial spacecraft state, dynamic model parameters, and all required natural body ephemerides. The accelerometer data is delivered to the Ephemeris Estimator as time-tagged acceleration data at 10 Hz. The time tags are in ephemeris time in seconds past J2000 format. The accelerations are in units of meters per second squared ( $m/s^2$ ) and are supplied as Cartesian components in Earth mean equator and equinox of J2000 coordinates. This acceleration data includes any orbit trim maneuver that occurred on the current orbit followed by the acceleration data due to drag during the passage through the atmosphere on the current orbit. (For more information, see Appendix G.)

### 7.3.2.3 Autonomous Maneuver Estimator

Among the parameters uploaded to the AADS during a ground update are the limits for the AB design corridor as well as the target within that corridor. The Maneuver Estimator processes data from both the Ephemeris Estimator and Atmospheric Estimator to obtain a prediction of the control parameter at the next periapsis. If the parameter is outside of the designed corridor, then

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 30 of 286	

the Maneuver Estimator logic determines the magnitude, direction, and execution time for the burn and passes that information back to the spacecraft.

For example, at Mars, data from the Ephemeris and Atmosphere Estimators are used to estimate the freestream heat rate at periapsis during the next atmospheric pass. If the estimate is outside of the operational corridor, then a maneuver is calculated such that the predicted heat rate for the next periapsis is at the target location within the corridor. First, a desired change in altitude is calculated using:


$$\Delta h = -H_s \cdot \ln\left(\frac{\rho_{desired}}{\rho_{predicted}}\right) = -H_s \cdot \ln\left(\frac{\dot{q}_{desired}}{\dot{q}_{predicted}}\right) \quad \text{Eq. (7.3-1)}$$

where  $\Delta h$  is the required altitude change and  $H_s$  is the predicted atmospheric scale height. This change in altitude is added to the current estimated periapsis altitude, which is then added to the estimated apoapsis altitude to determine a new orbit semi-major axis, from which a new velocity at apoapsis is determined. The difference between this new apoapsis velocity and the current estimate of the apoapsis velocity is the required maneuver magnitude. This value is positive for a periapsis raise (decrease freestream heating rate) and negative for a periapsis lowering (increase freestream heating rate). The maneuver direction is estimated to be that of the pre-maneuver velocity vector at apoapsis. Since the burn duration of these maneuvers is small compared to the orbital period, this assumption works well, even when considering a finite burn.

#### 7.3.2.4 Thermal Response Surface Algorithms

A high-fidelity thermal model of MRO was developed for use in this AA simulation study. Response surface equations based on 13 environmental, material, and modeling properties were derived from this high-fidelity MRO thermal model and integrated into the AADS. The high-fidelity thermal model was developed using the Thermal Desktop<sup>®</sup> software. The exclusive use of Thermal Desktop<sup>®</sup> represents a simplification of previously developed AB thermal analysis methodologies. Comparisons were made between the Thermal Desktop<sup>®</sup> solutions and those developed for the previous AB thermal analyses performed on the MRO during AB operations. Thermal analysis and response surface equations were developed for AA missions at Mars and Venus. (See Appendix D for detailed information.) A thermal analysis was not constructed for Titan since an altitude corridor was used for design, and thermal constraints are not likely a limiting factor for a Titan mission.

Though the thermal algorithm is used to predict solar array temperature in the AADS, it is also employed in the “truth” simulations as a monitor of the solar array temperature. In Phase 4 of AA development, thermocouple sensors onboard the spacecraft will provide further validation of the models.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 31 of 286	


## 7.4 AADS Model Integration

The AADS is a suite of models and algorithms intended to demonstrate the capability of an AA system. Three separate AADS packages were developed for this study, one each for Mars, Venus, and Titan. The AADS for application at Mars and Titan consists of three distinct modules: (1) the Ephemeris Estimator which processes spacecraft IMU acceleration data to estimate current and future spacecraft states, (2) the Atmosphere Estimator which processes spacecraft acceleration data along with Ephemeris Estimator state data to estimate the atmospheric density and scale height, and (3) the Maneuver Estimator which processes data from both the Ephemeris and Atmosphere Estimators to determine whether a maneuver is required (and the direction, magnitude, and execution time if one is needed) to keep the spacecraft within the desired operational corridor. In addition to these three modules, the AADS for Venus includes a fourth module containing temperature models to predict the maximum temperature the spacecraft will encounter during the next atmospheric pass.

The AADS is designed to output to the spacecraft a maneuver vector and its associated apoapsis time. With this information, the spacecraft can autonomously execute maneuvers at apoapsis to correct its periapsis altitude so that its design parameters are maintained within the specified corridor (e.g., heat rate, temperature, dynamic pressure, or altitude). In addition, the AADS outputs the atmospheric entry/exit state estimates so that the spacecraft can properly slew to AB configuration and begin and end its atmospheric data collection at the appropriate times.

The required AADS input data is passed into the AADS through two data structures; the first data structure includes parameters which are likely to change between AADS calls (e.g., spacecraft acceleration data), and the second data structure contains data not likely to change between AADS calls, but which may be changed and uploaded to the spacecraft during the weekly update cycle (e.g., corridor definition and target). At each AADS call, all calculations are performed and the required data are then passed back to the spacecraft through a separate data structure. At this time, the AADS can be placed in stand-by mode or terminated until the next AADS function call to release spacecraft resources for other activities. Onboard a spacecraft, the AADS will not always be running but instead is called once per orbit, typically after an atmospheric pass ends and prior to the next apoapsis. Some AADS data do need to be preserved between AADS calls (e.g., Ephemeris Estimator current state prediction and Atmosphere Estimator atmosphere archive data), so at least a portion of the spacecraft memory will be allocated and preserved while AADS is not running. Additional details regarding the AADS interfaces, data flow, and operation are included in Appendix H.

It is important to note that if a maneuver is commanded but for some reason is not executed (e.g., due to some issue with the spacecraft or some other operational consideration), this does not affect the AADS in any way. Each call to the AADS is distinct and independent with only limited information being passed from one call to the next. This information does not include any maneuver information. All accelerations (both from maneuvers and drag passes) are passed

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 32 of 286	

through the IMU acceleration data. In this way, any alteration, either deliberate or due to a spacecraft or commanding error or complete elimination of a maneuver, will be represented in the acceleration data it is provided during the next call.

The AADS has been developed for technology capability testing using generalized spacecraft models based on MRO (see Section 7.3.1 for more information). When a flight vehicle is selected for AADS implementation, the simulation aerodynamic and AADS thermal models must be adapted to that specific vehicle. The maneuver calculation, Atmosphere Estimation and Ephemeris Estimation models are not vehicle specific and do not require modification once validated.


## 7.5 AA Simulation Results

The performance of AADS was evaluated using two simulation methods: POST2 and the MESSENGER-based simulation at APL which is referred to as the AAHFS. The POST2 analysis was conducted in 3-DOF using the forces (no moments), which provided a baseline analysis of the key elements of AA. The AAHFS analysis was conducted using a 6-DOF simulation based on the flight software and truth models previously developed for the MESSENGER spacecraft, currently orbiting Mercury. Several advantages of the AAHFS analysis is the inclusion of sensor and actuator models, and the capturing of interactions between rotational and translational dynamics, which provides a higher fidelity testing environment that is closer to flight. Additional discussion of the higher fidelity captured by the AAHFS is provided in Section 7.5.2.

The AADS module is called from both “truth” simulations from POST2 and AAHFS, which provide it with only the required input parameters and IMU data described in the previous sections. Using the autonomous internal models, AADS made a decision to (or not to) perform a maneuver and passes only the maneuver vector and time of apoapsis back to the “truth” simulation. This section demonstrates the performance of AADS via AB mission simulations in both POST2 and AAHFS. Where appropriate, these simulation results have been compared against the reference simulations described in Sections 7.1 and 7.2, and Appendix B.

### 7.5.1 AADS Performance: POST2 Analysis

The key requirement for successful AADS operation is maintaining spacecraft safety throughout the AB mission. To demonstrate that AADS can accomplish this task, several AB simulation analyses were successfully completed for each destination: Mars, Venus, and Titan. For each simulation environment (POST2-based, and AAHFS), AADS predictions are compared with the “truth” simulation. In the AADS POST2 analyses, AADS is used to predict the upcoming atmospheric pass conditions; determine the spacecraft’s likely location with respect to the operational corridor; and if necessary, command the spacecraft to execute a maneuver at the following apoapsis to achieve a specified target within the corridor during the following pass. Unlike the reference simulation, the AADS operation allows these maneuvers to occur on each

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 33 of 286	

and every orbit, if necessary. These operations are all done autonomously within the AADS software, and the maneuver commands are passed back to the “truth” simulation where their execution is simulated, as if AADS were running onboard the spacecraft. Updates to AADS are also simulated. These updates would be used in real mission operations to periodically re-initialize the Ephemeris Estimator with a new truth state, update corridor and corridor target parameters, etc. The goal of these analyses is to show that the AADS system can provide sufficient performance (with margin) without interaction from the ground for at least 1 week.

Because of the high level of flexibility and modularity of POST2 and its extensive heritage and flight validation, it was possible to integrate the AADS code with POST2 in a way that is “flight-like.” The Interface Control Document for AADS can be found in Appendix J. In this simulation environment, POST2 takes on the role of modeling the physical environment as well as standing in as the spacecraft itself, where AADS is then executed through the POST2 flight software interface, in much the same way it would be implemented onboard the spacecraft. The needed interface data structures are created on the spacecraft/POST2 side and passed into the AADS. As would be the case onboard the spacecraft, the AADS code has no other connection to POST2 or the “outside world,” and vice versa, except through this data structure interface. Once integrated, the POST2 and AADS code are compiled into a single executable which is then run using the POST2 user interface.

With the AADS software successfully integrated into the POST2 simulation environment, AADS performance has been assessed for application at Mars, Venus, and Titan. The “truth” simulation (the POST2-based simulation that utilizes AADS maneuver data) utilizes the same planetary, atmosphere, gravity, spacecraft, and aerodynamics models as the reference simulation (see Sections 7.1 and 7.2). However, to initialize the AADS, the state from the seventh orbit is extracted from the reference simulation results and used as the simulation initial state to allow data from the first seven atmospheric passes to be used for building the atmosphere archive needed by the Atmosphere Estimator. (During operations, this archive would likely be constructed during the “walk-in” phase, while there is still ground interaction and prior to initiation of the AADS system.) The same operational corridors as the reference simulation analyses are used for the “truth” simulation and AADS logic; however, because this system is fully autonomous and maneuvers are allowed to occur at any apoapsis, the corridor target selection can differ from that of the reference simulation analyses.

### 7.5.1.1 AADS Module Performance at Mars

#### *Ephemeris Estimator Performance Assessment*

The Ephemeris Estimator performance can be assessed by examining how well the AADS module estimates the current (last occurring) periapsis state as compared to the POST2 “truth” simulation. Figure 7.5-1 illustrates this performance at Mars when assuming a perturbed atmosphere (using the perturbation functionality available within the Mars-GRAM “truth” atmosphere) and weekly ground updates, in terms of the differences in estimated periapsis time



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

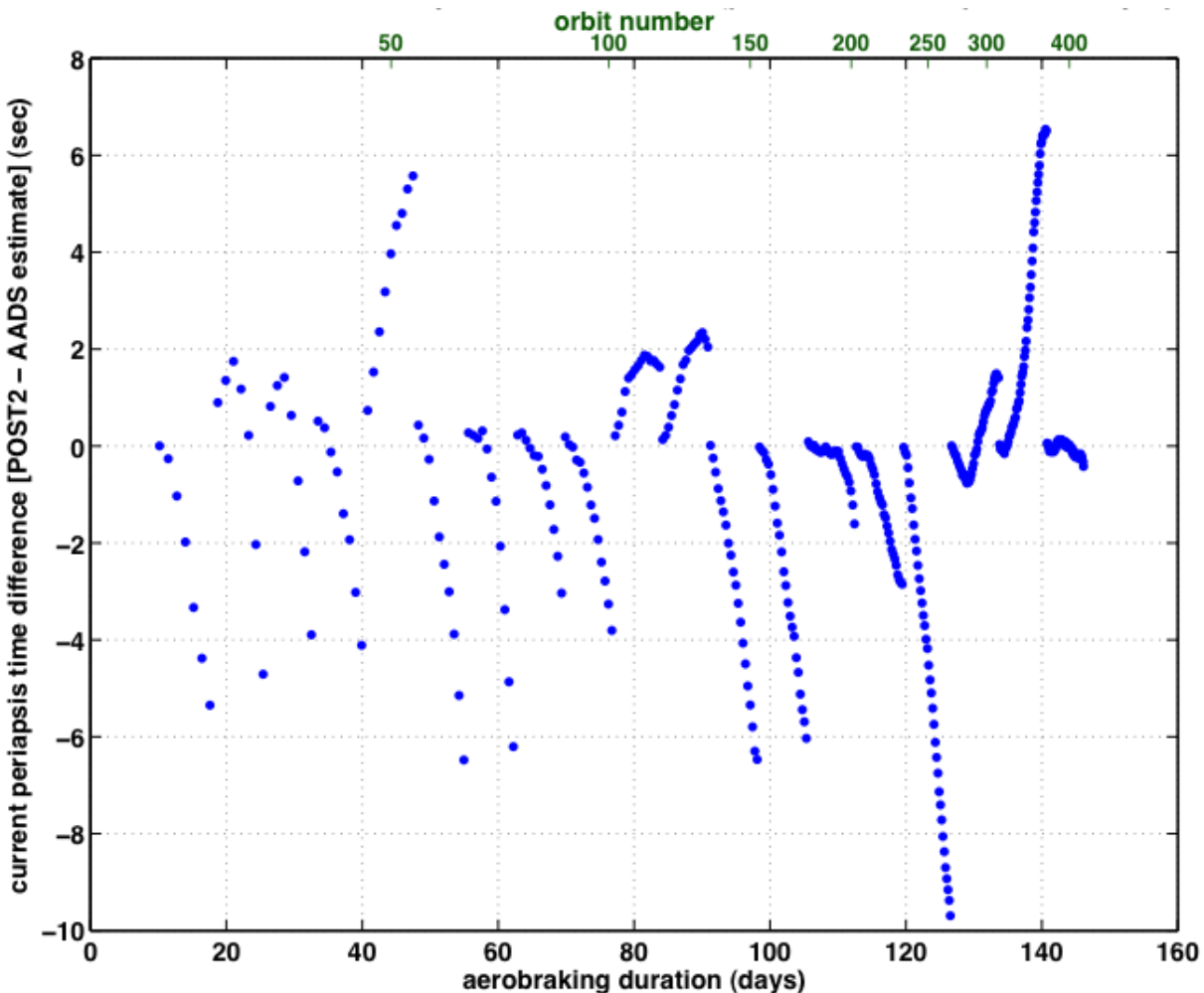
Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

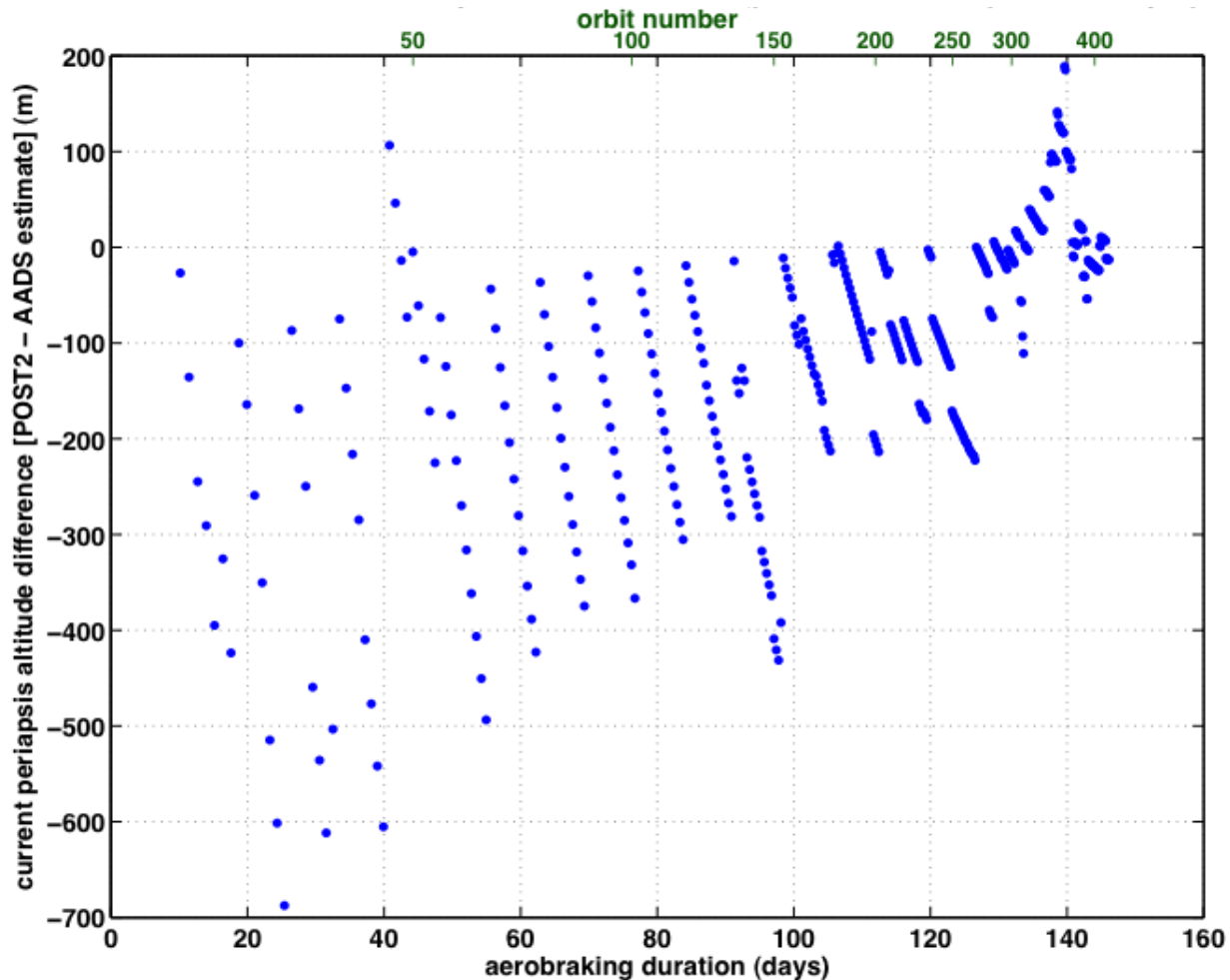
Page #:  
34 of 286

and altitude. At the start of the simulation and immediately following each ground update, when the Ephemeris Estimator is initialized with a POST2 state, the performance is quite good and shows excellent agreement (sub-second and meter level) between the Ephemeris Estimator and POST2 estimates. As time progresses, drifting in the Ephemeris Estimator propagation (due to differences in integrators, etc.) and a growth in error from the lack of precision (i.e., data rate) in the acceleration data provided by the spacecraft (during both the atmospheric pass and maneuver), causes the Ephemeris Estimator estimates to diverge, but never by more than 10 seconds or 700 meters (m).



*Figure 7.5-1a. AADS Ephemeris Estimator Performance at Mars with Perturbed Atmosphere (and 7-day Updates): Current Periapsis Time*


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 35 of 286	



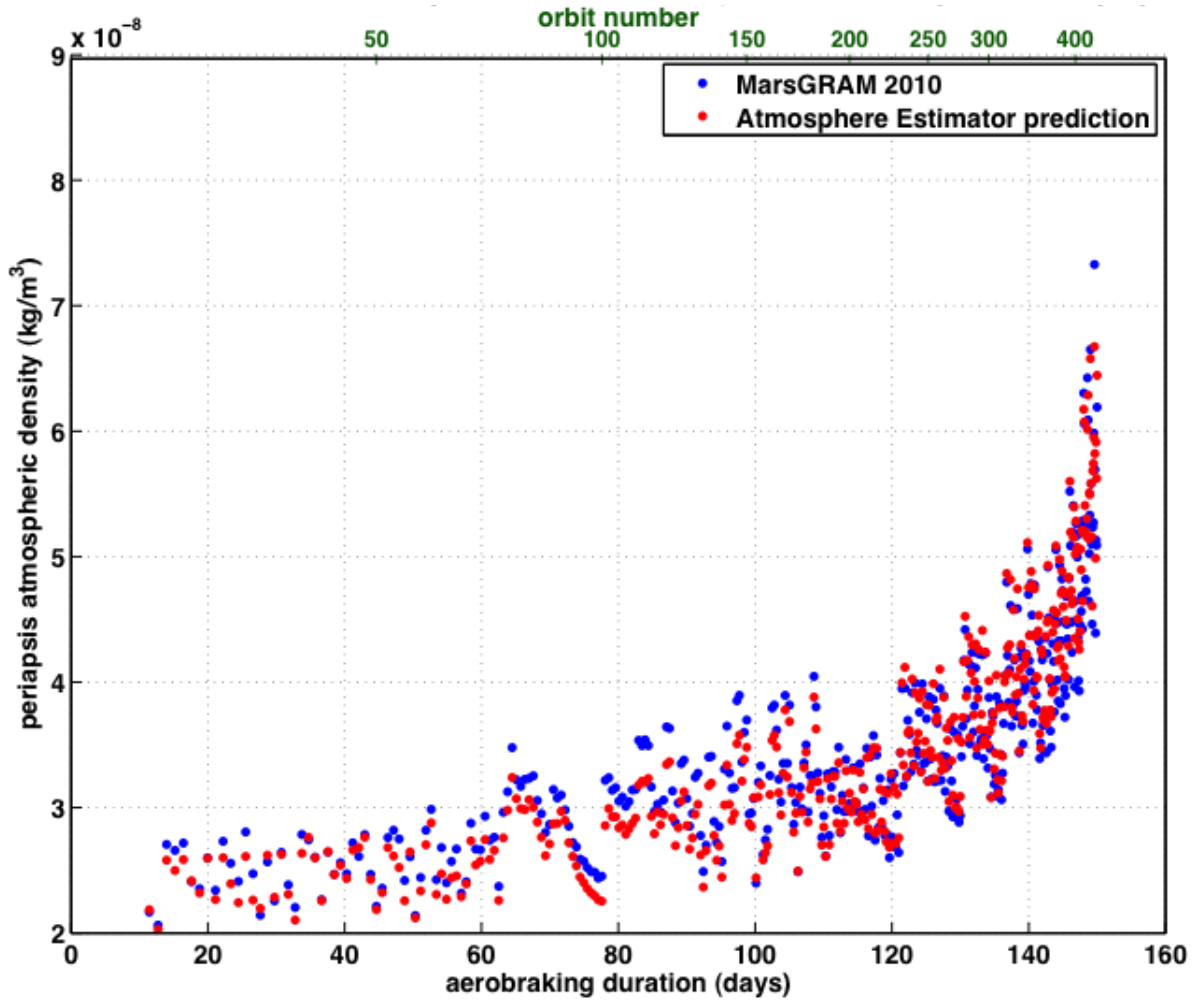
*Figure 7.5-1b. AADS Ephemeris Estimator Performance at Mars with Perturbed Atmosphere (and 7-day Updates): Current Periapsis Altitude*

#### *Atmosphere Estimator Performance Assessment*

The Atmosphere Estimator performance can be assessed by comparing the density and scale height estimates for the next periapsis against the “truth” MarsGRAM-2010 atmosphere model values. Since the Atmosphere Estimator is attempting to fit a mean curve to the acceleration/density prediction, it is appropriate to make this comparison using an AADS run with a nominal atmosphere. As Figure 7.5-2 shows, the Atmosphere Estimator density prediction is generally within 10 percent and the scale height within ~ 1 km (~ 15 percent) of the actual from the Mars-GRAM 2010 model. The difference in the scale height estimates can be tied to the drift in the Ephemeris Estimator periapsis time and altitude (profile) estimates. As previously discussed, improvements to the Atmosphere Estimator to update the algorithm to be independent of altitude


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 36 of 286	

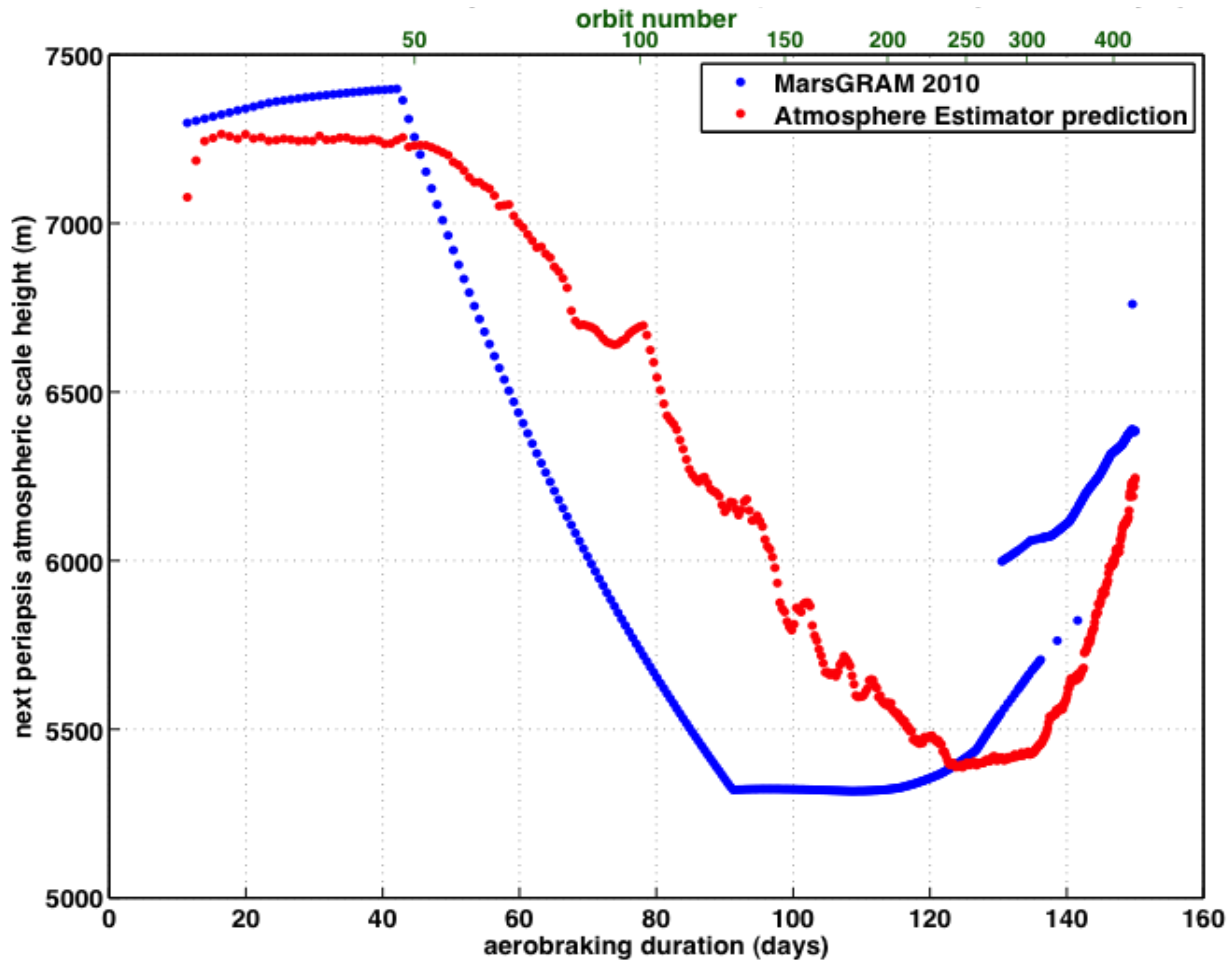
estimates are currently planned for Phase 2 and should provide enhanced Atmosphere Estimator performance.



*Figure 7.5-2a. AADS Atmosphere Estimator Performance at Mars (with 7-day Updates): Predicted Periapsis Atmospheric Density*



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 37 of 286	



*Figure 7.5-2b. AADS Atmosphere Estimator Performance at Mars (with 7-day Updates): Predicted Periaresis Atmospheric Scale Height*

### *AADS Performance Assessment*

To analyze the details of the prediction capability within the designed corridor per orbit, a summary of the AADS performance and margin throughout a Mars AB mission scenario using a perturbed atmosphere is provided in Figure 7.5-3. The capability is measured in terms of how well the spacecraft stays within the mission operations corridor. These plots show that the AADS system successfully keeps the spacecraft within the specified corridor and illustrates the difference between the AADS predicted freestream heat rate (red) and the estimated “truth” heat rate (blue), which is mainly driven by the differences between the Atmosphere Estimator density and scale height estimates from those of the Mars-GRAM 2010 atmosphere model. This is illustrated for ground update frequencies of 3, 5, 7, and 14 days. (See Figure H.2 in Appendix H for a more detailed description of how to interpret the mission corridor plot.)



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

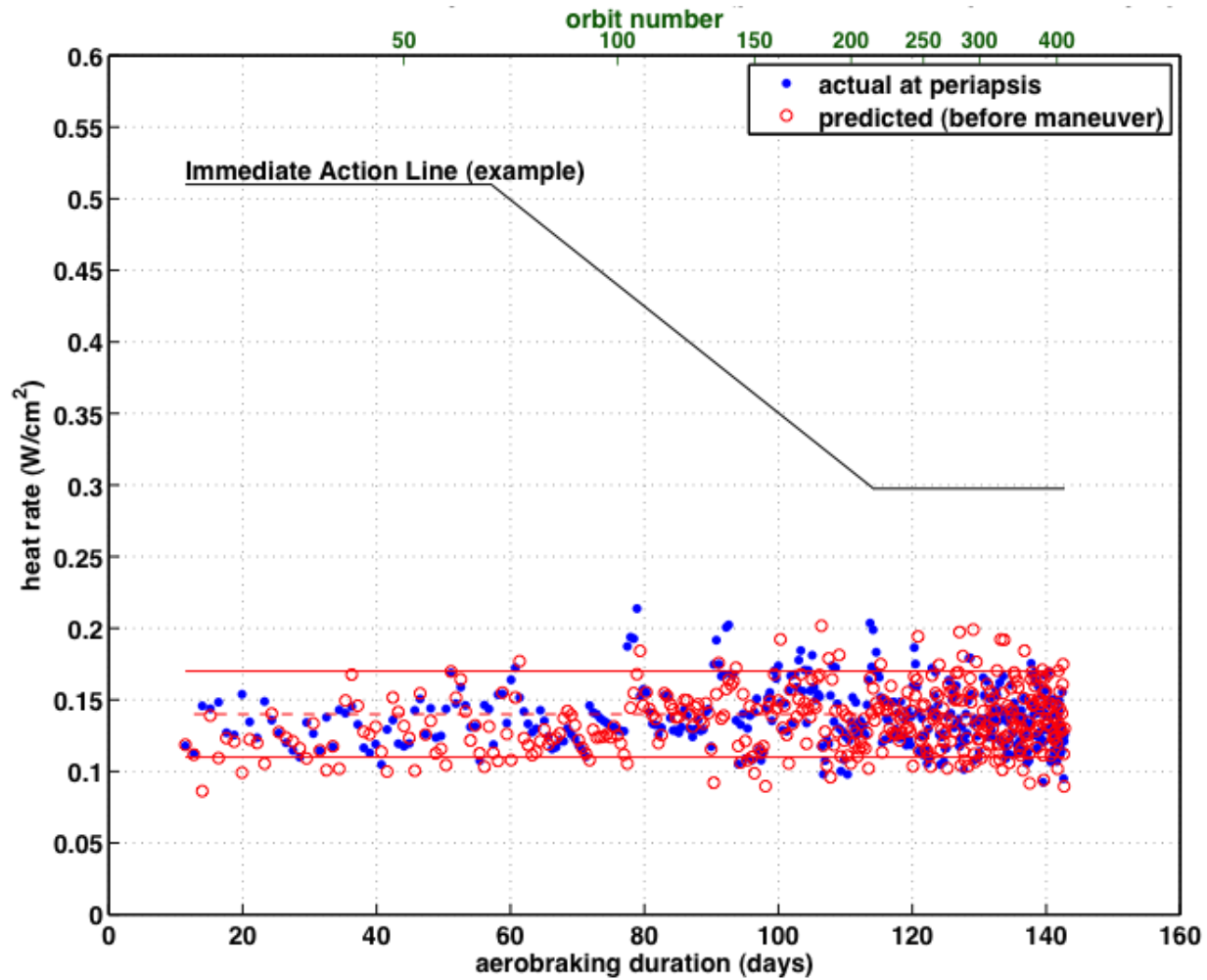
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

38 of 286



*Figure 7.5-3a. AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 3-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

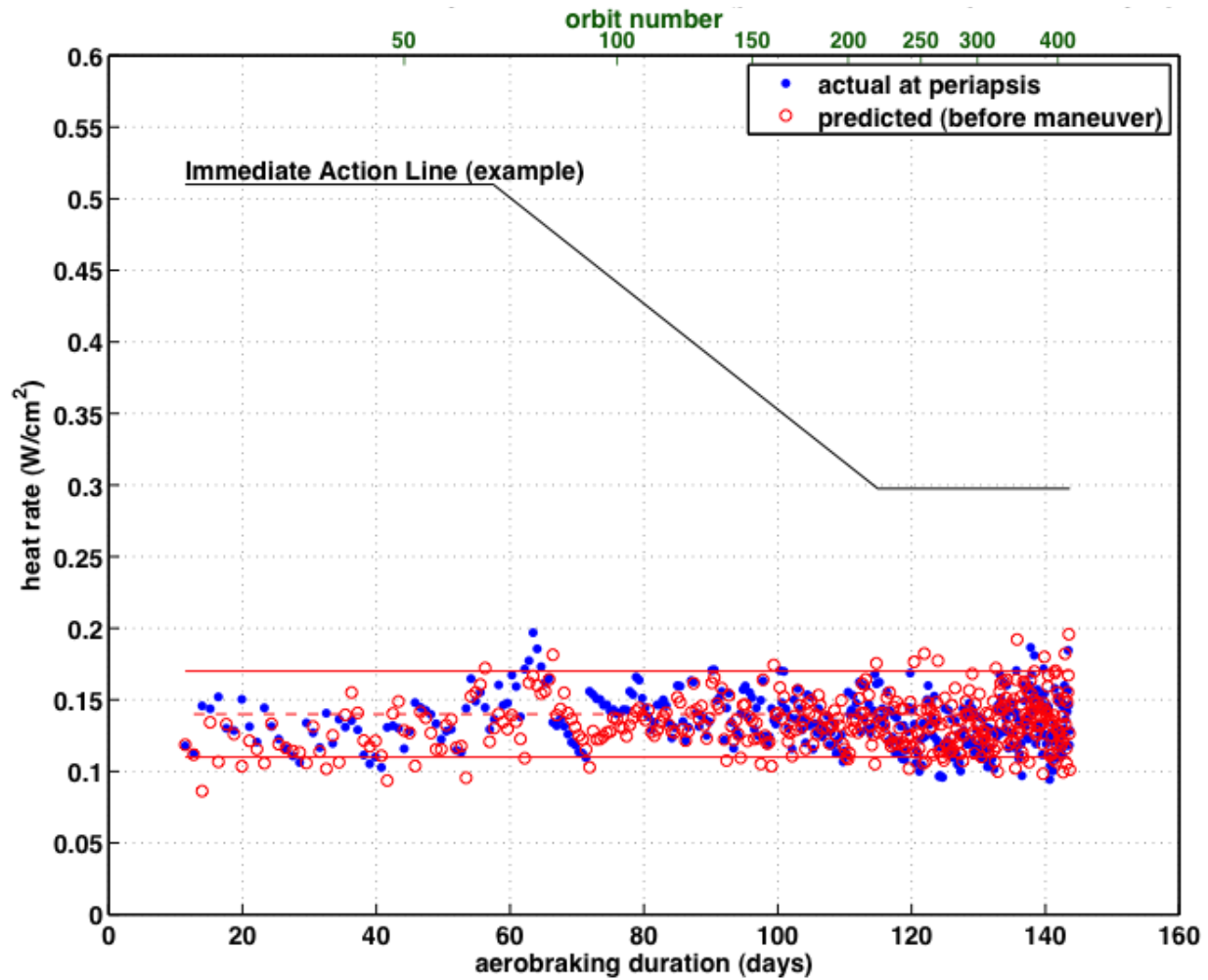
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

39 of 286



*Figure 7.5-3b. AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 5-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

40 of 286

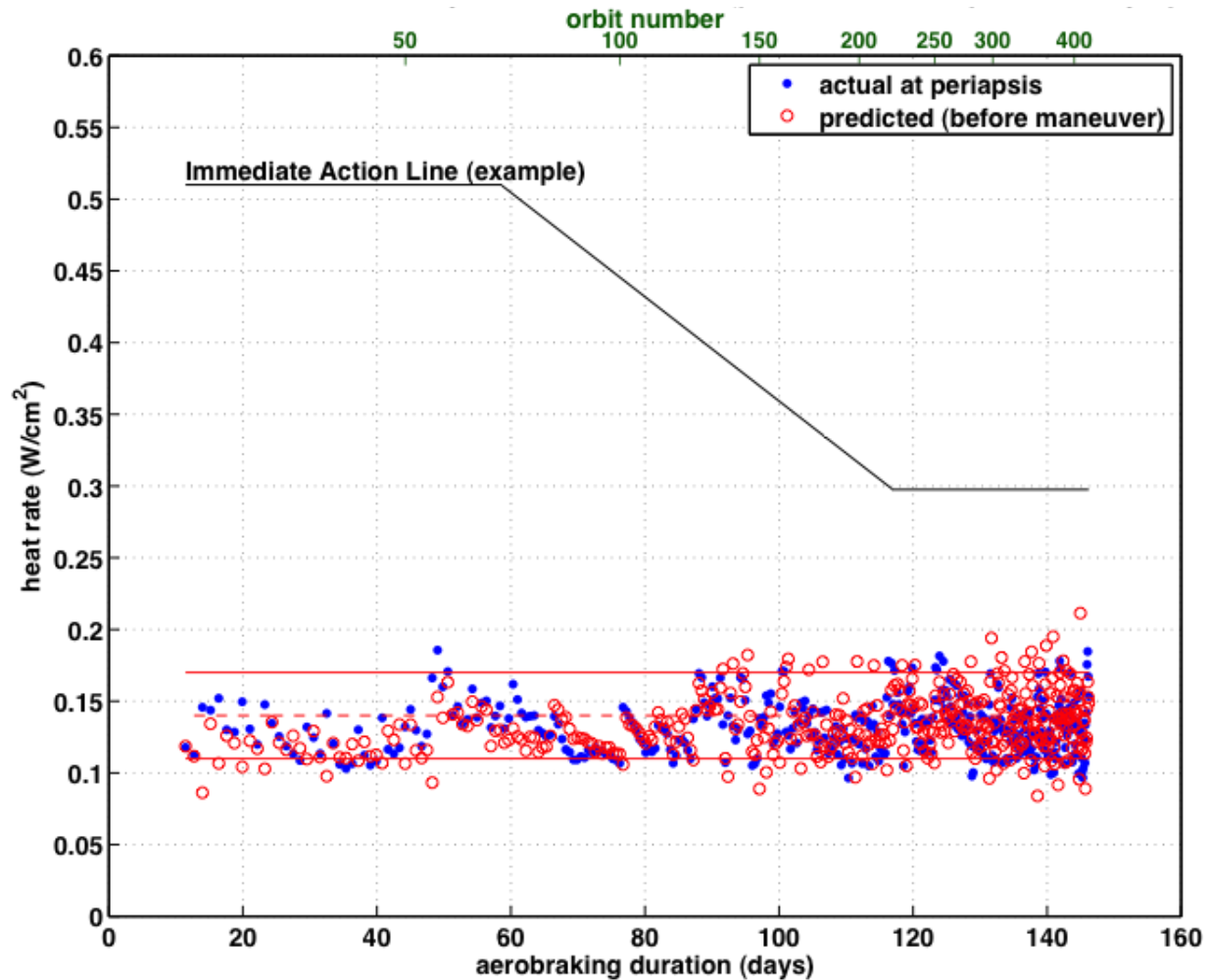


Figure 7.5-3c. AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 7-day Updates



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

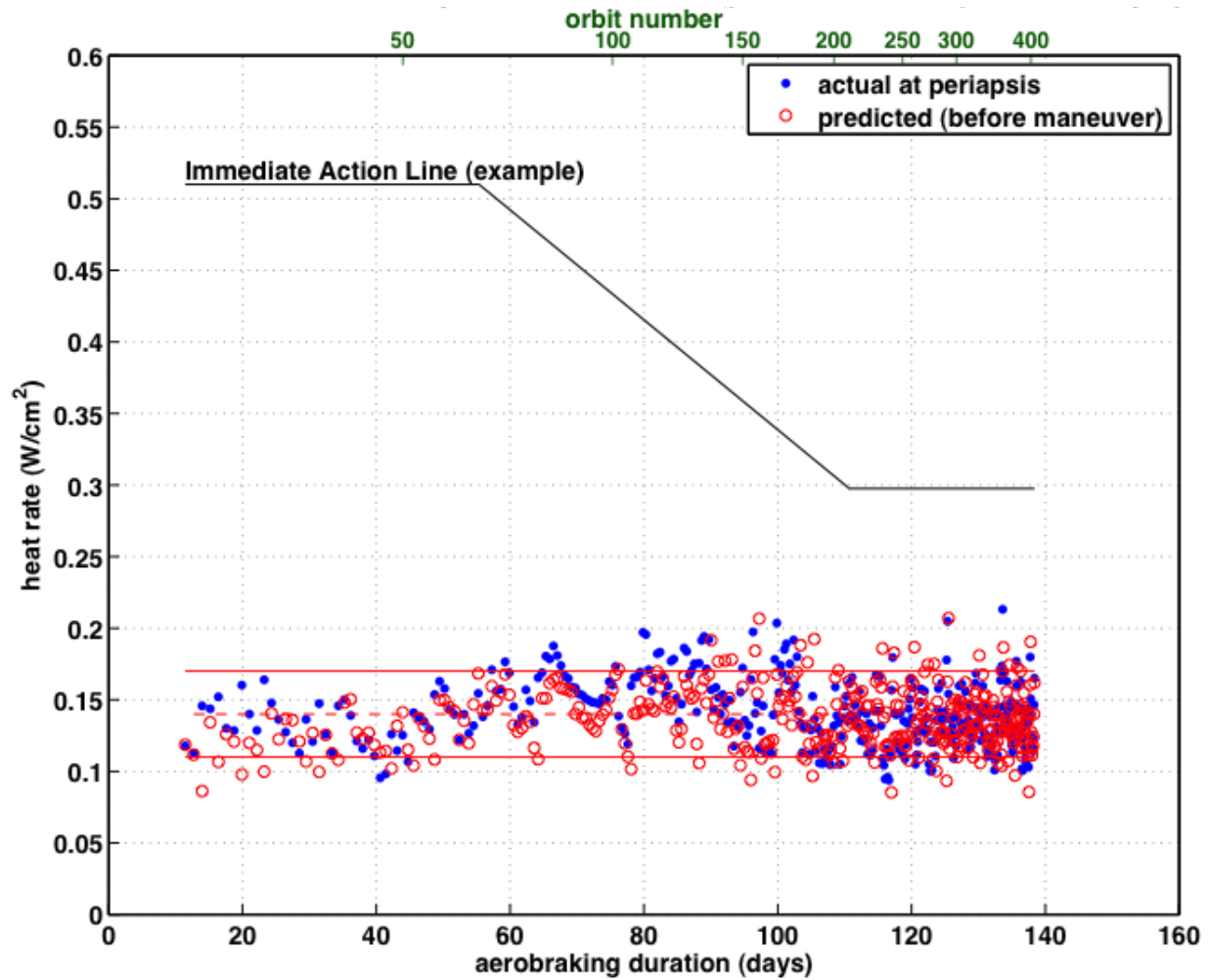
Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

Page #:

41 of 286



*Figure 7.5-3d. AADS Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere and 14-day Updates*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		42 of 286	


As expected, this variation in update frequency has little effect on the AADS performance, as illustrated in Table 7.5-1, which compares the mean of the absolute value percent difference between predicted and “truth” heat rate and  $1\sigma$  variance in those percent difference values. As previously described, the driving constraints on the AADS performance are the drift in the Ephemeris Estimator integration, and to a lesser extent, the Atmosphere Estimator density estimate accuracy.

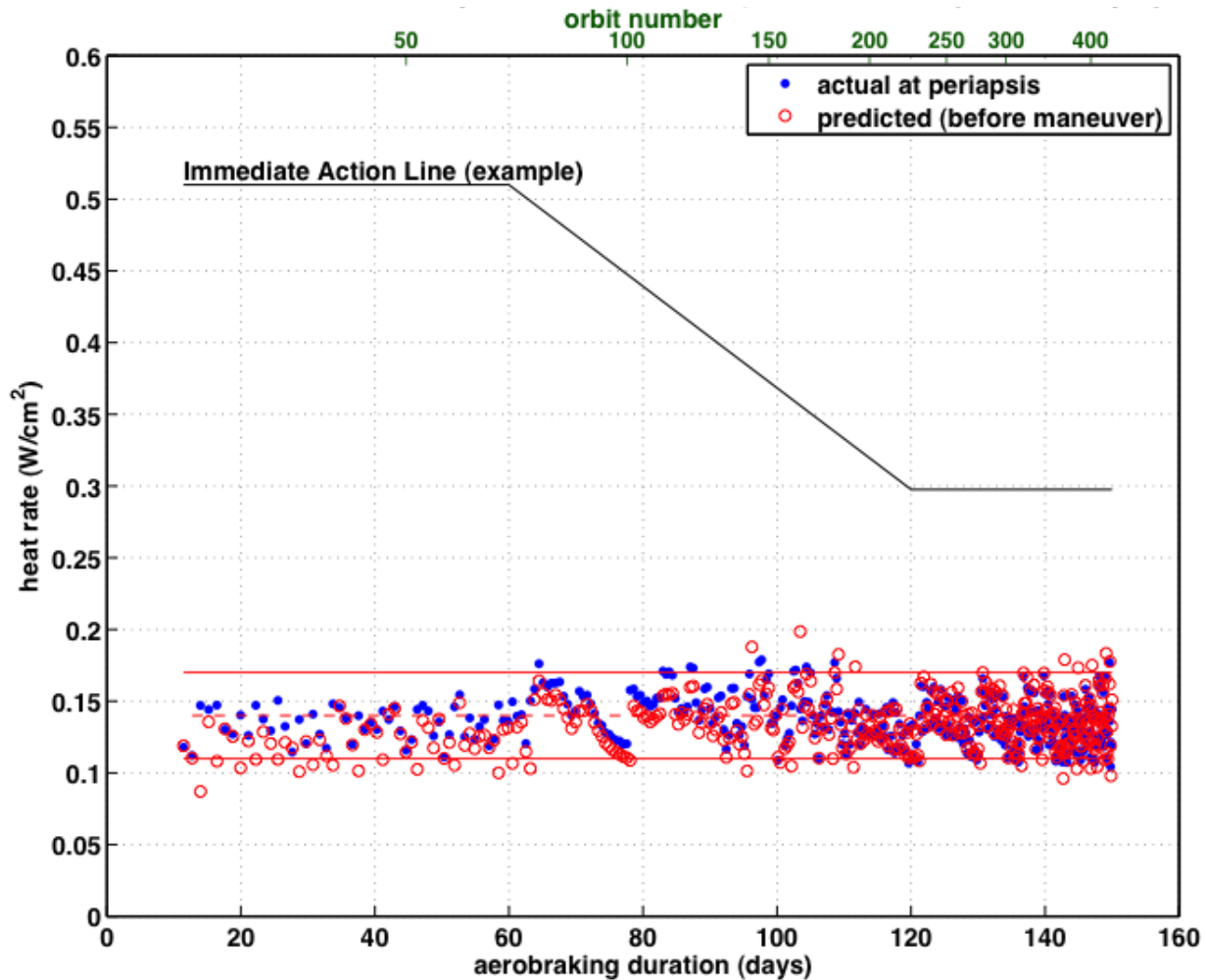
**Table 7.5-1. Summary of AADS Heat Rate Estimate Performance Variability for Various Ground Update Frequencies**

<i>Update Frequency (days)</i>	<i>Mean (percent)</i>	<i>1<math>\sigma</math> (percent)</i>
<b>3</b>	11.11	10.95
<b>5</b>	9.55	8.69
<b>7</b>	11.31	10.65
<b>14</b>	11.34	10.99

This small variability in performance for all update frequencies is another measure of how well the Ephemeris Estimator and Atmosphere Estimator are performing. The perturbations in the atmosphere do introduce additional variability in the heat rate estimate. However, both the Ephemeris and Atmosphere Estimators are now working with acceleration data that reflect these atmospheric variations. This effect can be seen when comparing against a nominal atmosphere, where the corridor performance is much improved (as seen in Figure 7.5-4 and illustrated by the heat rate estimate performance mean being closer to 6.00 percent with a  $1\sigma$  variance of 7.26 percent).

What is most important to note from Figure 7.5-3 and Table 7.5-1, however, is that with a requirement of no less than 7 days between ground updates, the spacecraft could survive with little or no ill effects for up to at least 14 days while maintaining significant operational margin.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 43 of 286	



*Figure 7.5-4. AADS Mission Operations Corridor Performance at Mars with a Nominal Atmosphere (and 7-day Updates)*

### 7.5.1.2 AADS Mars Comparisons to Reference Simulation

With simulations complete for both the Mars AB reference simulation and the “truth” simulation with AADS implementation, it is now possible to compare the system performance between these two analyses. Figure 7.5-5 provides a comparison of the AB mission profile using the AADS (incorporating a perturbed atmosphere and 7-day updates) to the reference simulation, including the difference between the AB “glideslope” (orbit period versus time) and a comparison of the commanded maneuvers, orbit periapsis altitudes, and periapsis locations (areocentric latitude) as a function of time.



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

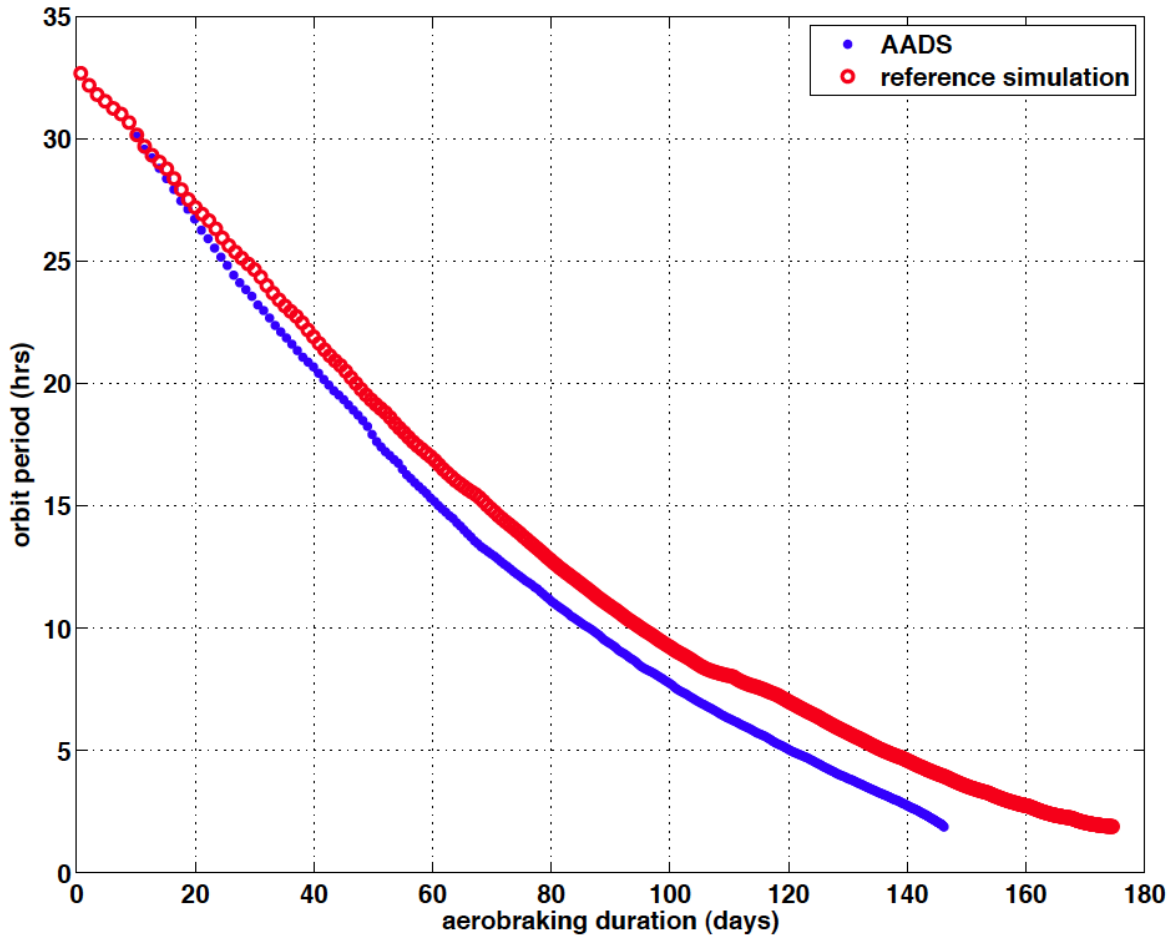
Title:

## Autonomous Aerobraking (Phase 1)

Page #:

44 of 286

### POST2 Simulation: AADS and Reference Simulation Comparison



*Figure 7.5-5a. AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Orbit Period “Glideslope”*





# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

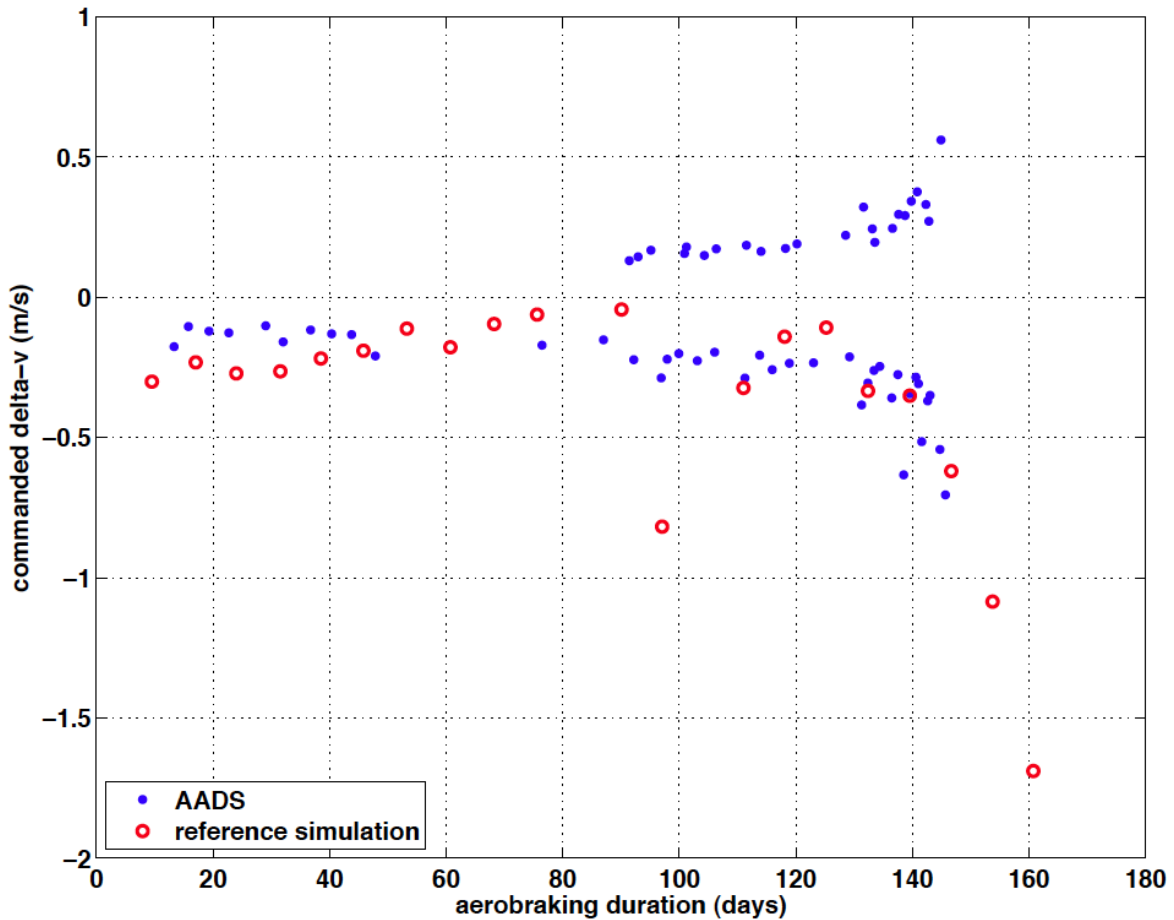
Title:

## Autonomous Aerobraking (Phase 1)

Page #:

45 of 286

### POST2 Simulation: AADS and Reference Simulation Comparison



*Figure 7.5-5b. AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Commanded Maneuver  $\Delta V$*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

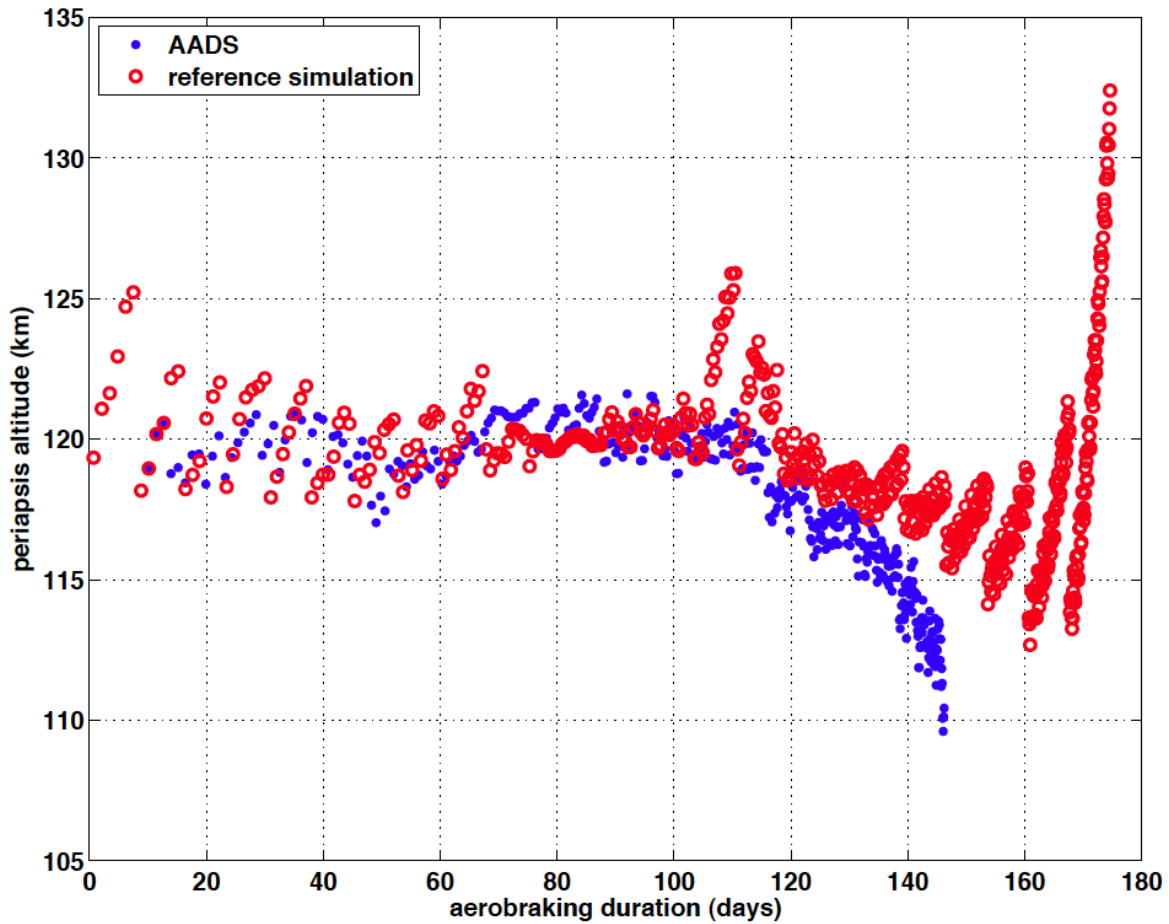
Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

Page #:  
46 of 286

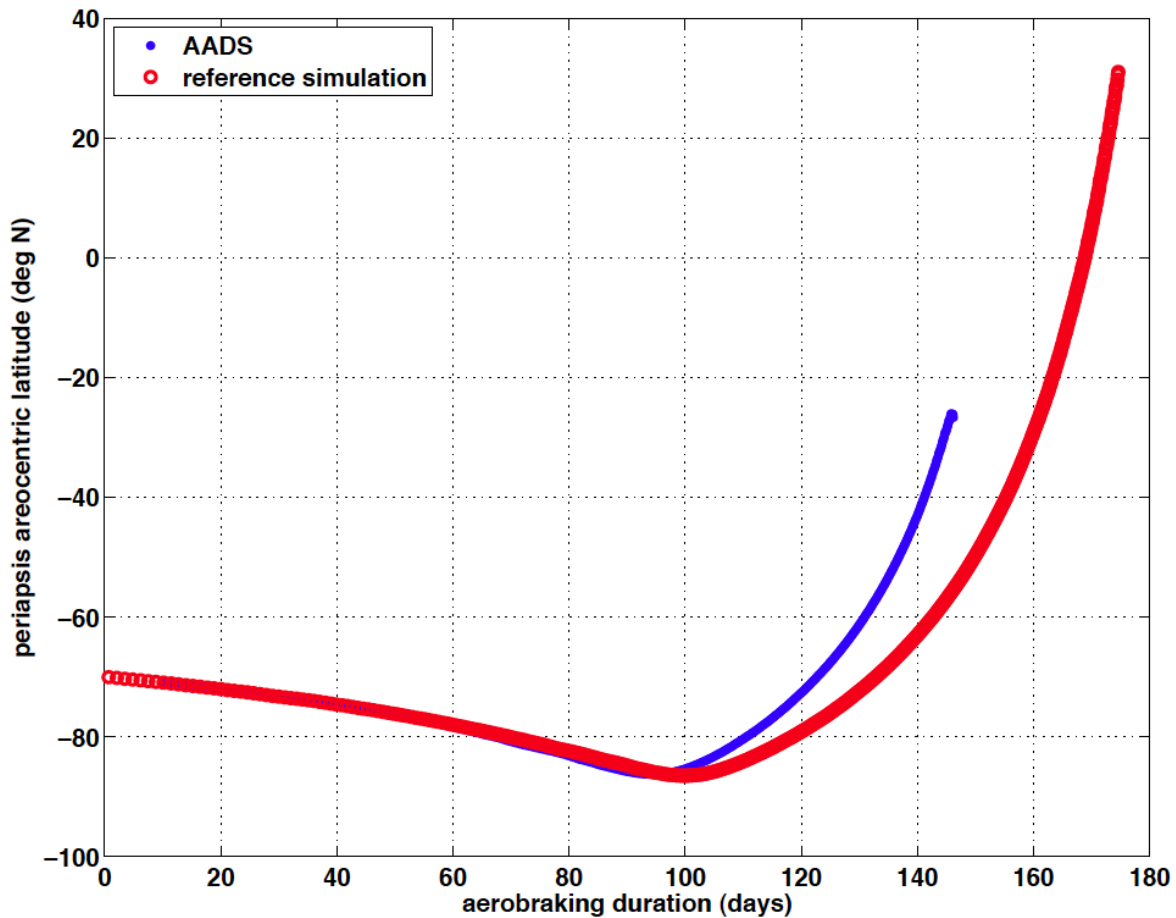
### POST2 Simulation: AADS and Reference Simulation Comparison



*Figure 7.5-5c. AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Periapsis Altitude*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 47 of 286	


**POST2 Simulation: AADS and Reference Simulation Comparison**



*Figure 7.5-5d. AADS with Perturbed Atmosphere (and 7-day Updates) Comparison with Reference Simulation at Mars: Periapsis Areocentric Latitude*

These plots demonstrate that AADS is accurately calculating maneuvers to generate AB missions similar to those simulated using traditional AB design methods (i.e., the reference simulation). The AADS mission is shorter in duration because maneuvers of any size are allowed at apoapsis versus the imposed constraint of one maneuver no more frequently than every 7 days in the reference simulation. In reality there may be some minimum engine thrust that is allowed or collision avoidance criteria that imposes constraints on maneuvers. These considerations will be evaluated in future phases.

The comparison of the simulation using AADS against the reference simulation analysis is summarized in Table 7.5-2 below.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		48 of 286	

**Table 7.5-2. Summary of AADS Performance with a Perturbed Atmosphere (and 7-day Updates) Compared to Reference Simulation at Mars**

	<i>AADS</i>	<i>Reference Simulation</i>
<b>AB Duration (days)</b>	146.2	174.7
<b>Total <math>\Delta V</math> (m/s)</b>	15.9	7.9
<b>Number of Maneuvers</b>	62	21

The reduction in the AB duration in the simulation incorporating AADS as compared to the reference simulation is due to the AADS ability to perform a maneuver on each and every orbit, if necessary, whereas the reference simulation only allows a maneuver once per week (as illustrated in the number of maneuvers). The increase in total  $\Delta V$  is mainly a consequence of comparing the “truth” simulation incorporating AADS, which uses a perturbed atmosphere, against the reference simulation, which uses a nominal atmosphere. These atmospheric perturbations have a strong effect on the AADS estimates, including the required maneuver magnitudes.

### 7.5.1.3 Effects of Changing Corridor Limits

As discussed earlier, when the spacecraft heat rate increases, the orbital periapsis is at lower altitude; likewise, an elevated altitude results in a lower heat rate. It follows then that changing the corridor limits (and/or the target within the corridor) can have a significant effect on the AB mission duration as well as the number of maneuvers performed. The corridor can be modified to accomplish any number of objectives: to ensure proper terminal orbit conditions, to conserve propellant, or to increase or decrease thermal margin. To illustrate the effects of modifying the corridor limits, the AADS AB mission was simulated again using a tight operational corridor (0.02 W/cm<sup>2</sup> in width) centered at the nominal corridor lower limit (0.11 w/cm<sup>2</sup>) and again at the nominal corridor upper limit (0.17 W/cm<sup>2</sup>). Figure 7.5-6 illustrates the corridor performance for these cases; a summary of the mission performance is provided in Table 7.5-3.



# NASA Engineering and Safety Center Technical Assessment Report

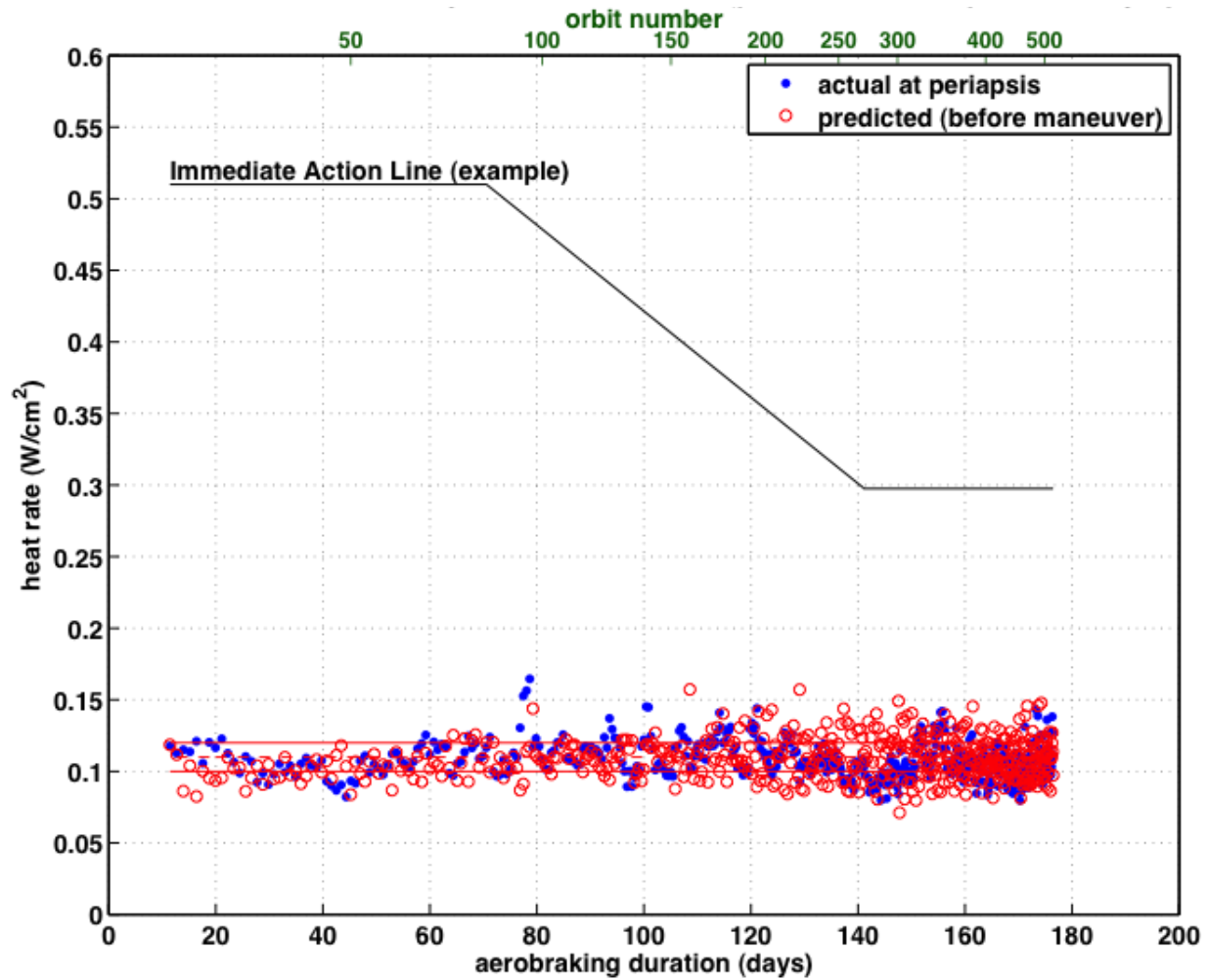
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
49 of 286



*Figure 7.5-6a. AADS Constrained Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere (and 7-day Updates) for a Tight Corridor Centered on the Nominal Corridor Lower Limit*



# NASA Engineering and Safety Center Technical Assessment Report

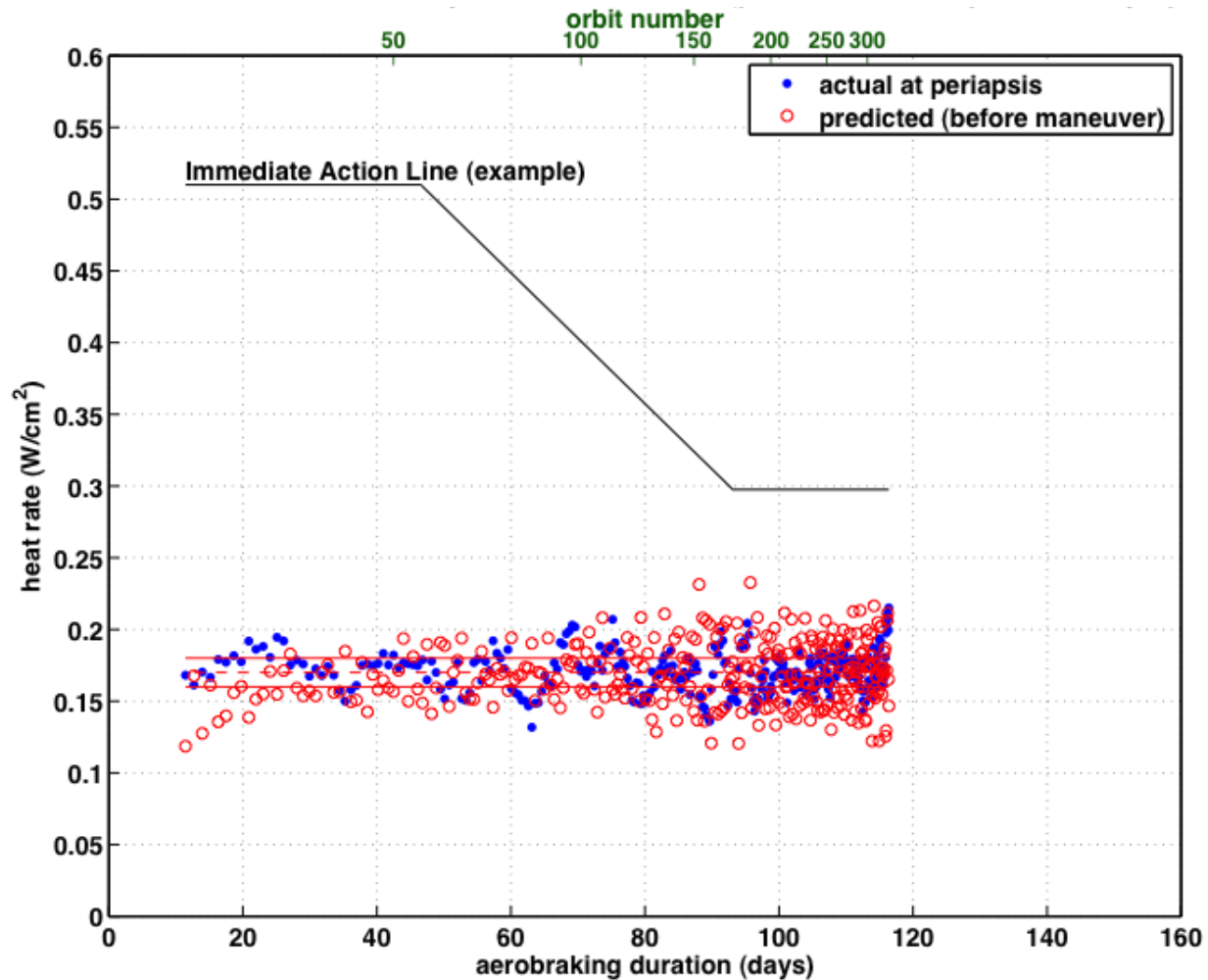
Document #:  
**NESC-RP-09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
50 of 286



*Figure 7.5-6b. AADS Constrained Mission Operations Corridor Performance at Mars with a Perturbed Atmosphere (and 7-day Updates) for a Tight Corridor Centered on the Nominal Corridor Upper Limit*

*Table 7.5-3. Summary of AADS Performance at Mars with a Nominal Atmosphere for Runs using the Nominal Operational Corridor, a Corridor Constrained to the Nominal Corridor Upper Limit, and One to the Nominal Corridor Lower Limit*

	<i>Nominal Corridor</i>	<i>Corridor at Lower Limit</i>	<i>Corridor at Upper Limit</i>
<b>AB Duration (days)</b>	146.2	176.5	116.4
<b>Total ΔV (m/s)</b>	15.9	36.6	28.9
<b>Number of Maneuvers</b>	62	249	227


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 51 of 286	

Figure 7.5-6b does show some increased volatility in the AADS predictions as the AB mission progresses for the more aggressive corridor, as compared to Figure 7.5-6a which uses a more conservative corridor. This volatility is driven by the aggressive nature of corridor (e.g., low altitude = higher density) and its impact on the accuracy of the Ephemeris Estimator's integration of the atmospheric pass acceleration data, and thus the Atmosphere Estimator's density estimates. Since density information is estimated almost directly from acceleration data, and AADS will be more sensitive to these estimates due to the higher density environment, lack of sufficient resolution of the acceleration data will result in earlier and/or more rapid divergence within the AADS. As the mission progresses and the orbit period reduces, the Ephemeris Estimator propagation relies more heavily on acceleration data as the atmospheric passes become longer and more frequent.

The accuracy at which AADS can maintain such a tight operational corridor will also be driven by the accuracy in the predicted freestream heat rate. Any improvements in the Atmosphere Estimator density and scale height estimates will narrow the spread between the predicted and actual, improve the required maneuver estimate to reach the target in the corridor, and thus further improve the overall corridor performance.

This effect of corridor selection on mission duration can be further emphasized by evaluating the mission performance using AADS for a corridor with the same width as the nominal analysis ( $0.06 \text{ W/cm}^2$ ), but shifted up by that same amount ( $0.17\text{--}0.23 \text{ W/cm}^2$ ) to significantly reduce the time required for AB (as shown in Figure 7.5-7) without any impact to the number of maneuvers and/or  $\Delta V$  required (which are driven more by the corridor size and target). Table 7.5-4 provides a comparison between the nominal and this more aggressive AB mission. Figure 7.5-7 shows the same immediate action line limits as for the nominal mission corridor, further illustrating the importance of selecting an operational corridor which ensures sufficient spacecraft safety margin throughout the AB mission.



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

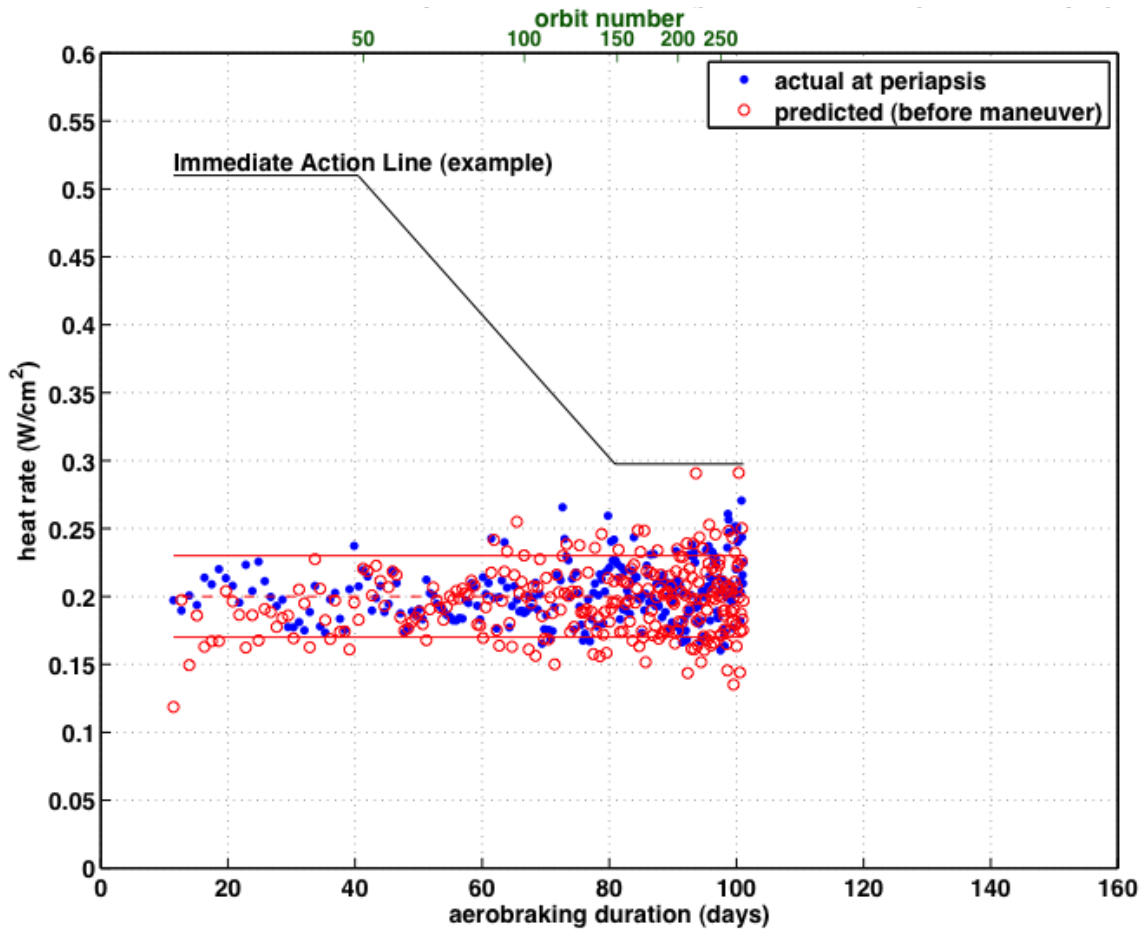
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)


Page #:

52 of 286



*Figure 7.5-7. AADS Mission Aggressive Operations Corridor Performance at Mars with a Perturbed Atmosphere (and 7-day Updates)*



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		53 of 286	

**Table 7.5-4. Comparison of AADS Performance at Mars with a Perturbed Atmosphere (and 7-day Updates) for the Nominal and Aggressive Operational Corridors**

	<i>Nominal Corridor</i>	<i>Aggressive Corridor</i>
<b>AB Duration (days)</b>	146.2	101.1
<b>Total <math>\Delta V</math> (m/s)</b>	15.9	13.9
<b>Number of Maneuvers</b>	62	69

The glideslope, or plot of orbit period or apoapsis altitude reduction over time, provides the metric that determines how quickly an AB mission can be performed. Following the reference simulation glideslope is sufficient to ensure that all desired terminal orbit conditions (e.g., orbital period, periapsis altitude, LMST, etc) are met at the end of the AB phase. Based on the results in the previous section, modifying the corridor can adjust the duration and therefore the current glideslope of an AB mission. In this way, ground operators can easily control the glideslope of an AB mission by simply manipulating the corridor bounds at ground updates. To demonstrate this concept of glideslope control, an additional analysis using AADS was successfully completed using the POST2 simulation at Mars. In this simulation, after some period of time in nominal AB, AADS is instructed (through the spacecraft update capability) to dramatically lower the operational corridor. At Mars, this raises the periapsis altitude higher than it would have been during the nominal mission, thus reducing drag, and results in the spacecraft “falling behind” the nominal glideslope. After approximately 1 week, AADS is commanded to increase the upper corridor limit back to where it was for the nominal mission and to dramatically increase the lower corridor limit. This decreases the periapsis altitude target deeper into the atmosphere, increasing drag, steepening the glideslope, and allowing the spacecraft to eventually recover the nominal glideslope after about 2 weeks. A subtle, but profound implication of this analysis is that it demonstrates how AADS can be used to reduce risk during off-nominal AB operations, as the nominal glideslope was recovered without raising the upper corridor bound (i.e., without reducing thermal margin). Figure 7.5-8 shows the effect of this scenario on the glideslope profile, as compared to the reference simulation, while Figure 7.5-9 shows the effect on the operational corridor and resulting AADS performance.



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

54 of 286

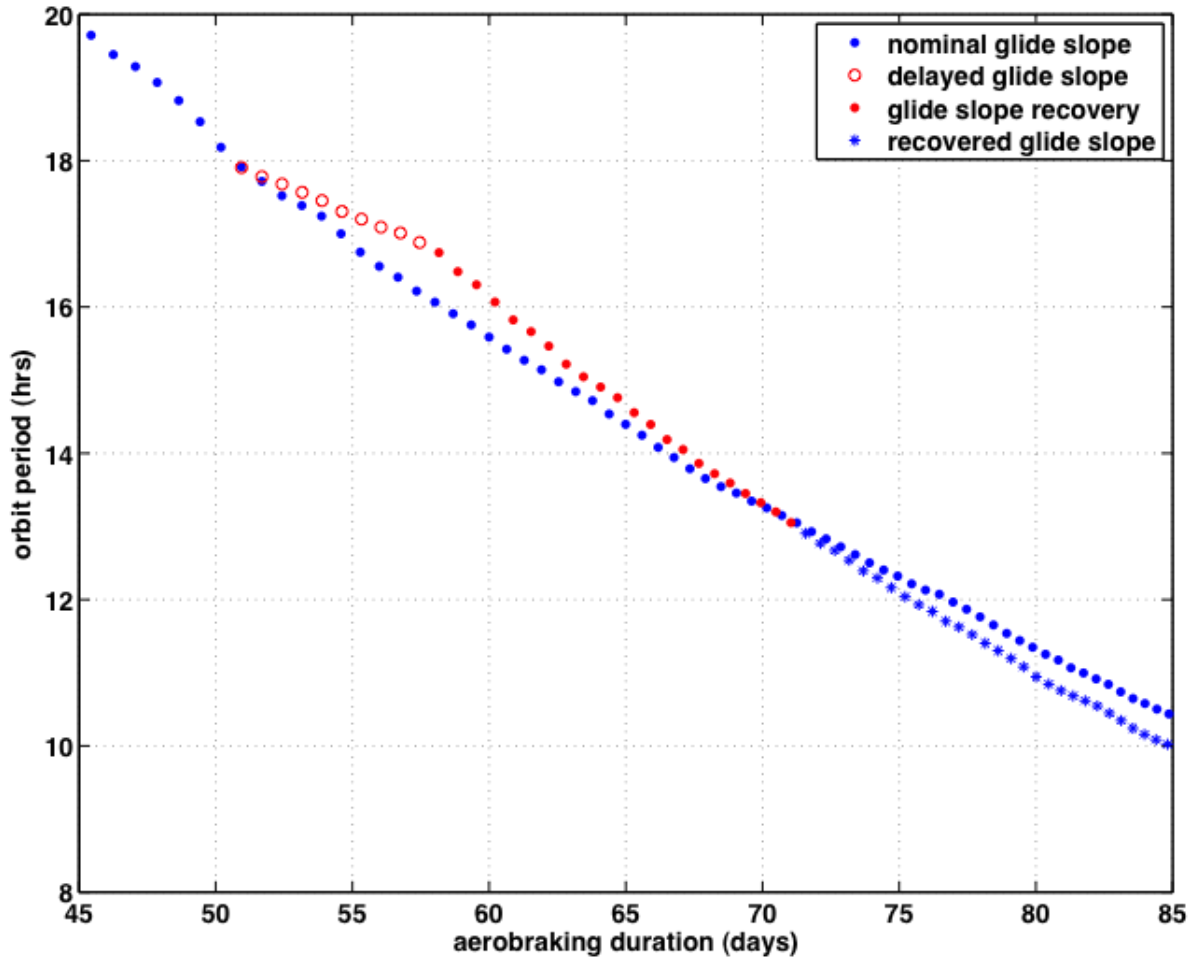


Figure 7.5-8. AADS Glideslope Delay and Recovery Scenario at Mars with a Perturbed Atmosphere (and 7-day Updates)



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

55 of 286

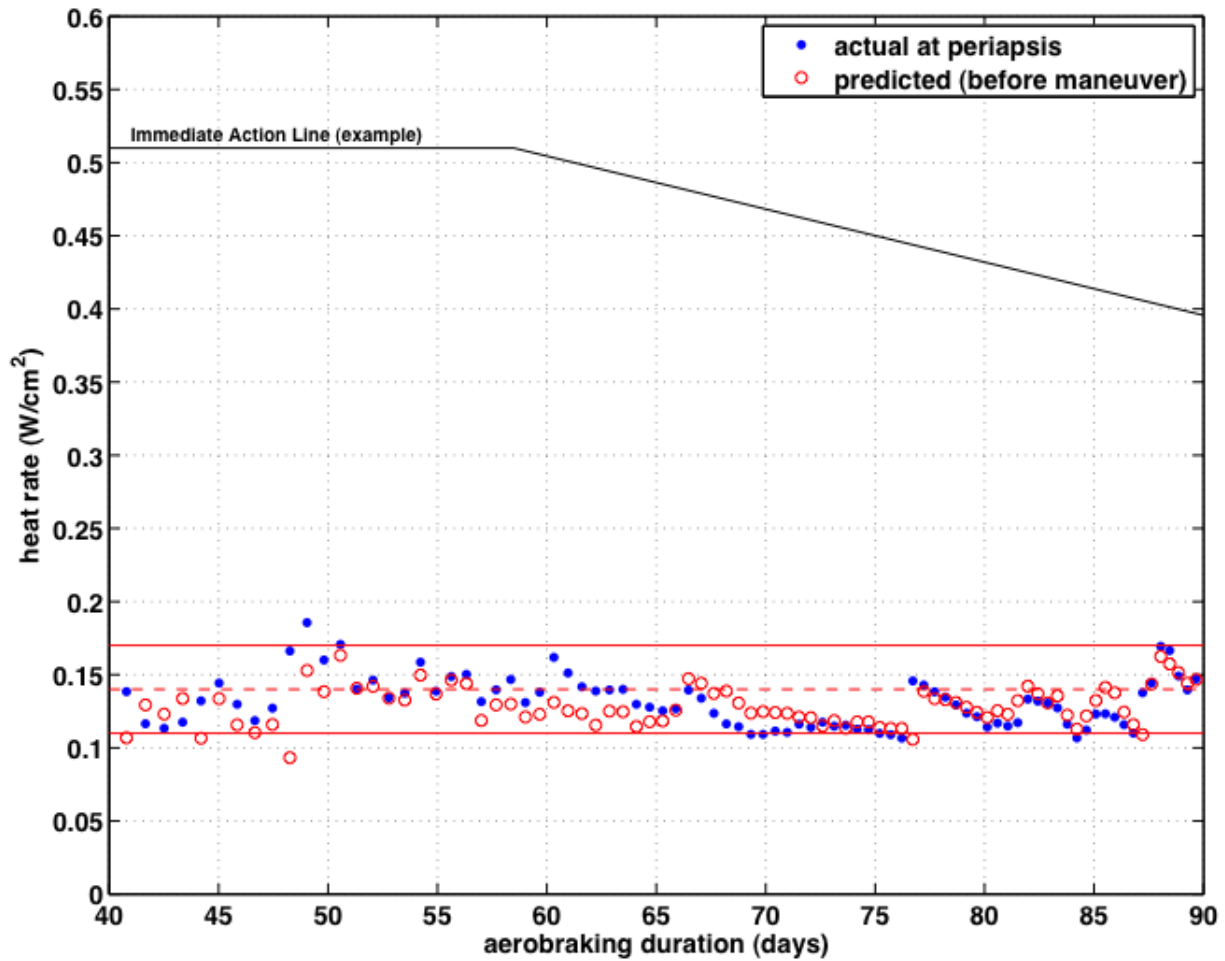


Figure 7.5-9a. AADS Mission Operations Corridor Performance for the Nominal Corridor



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

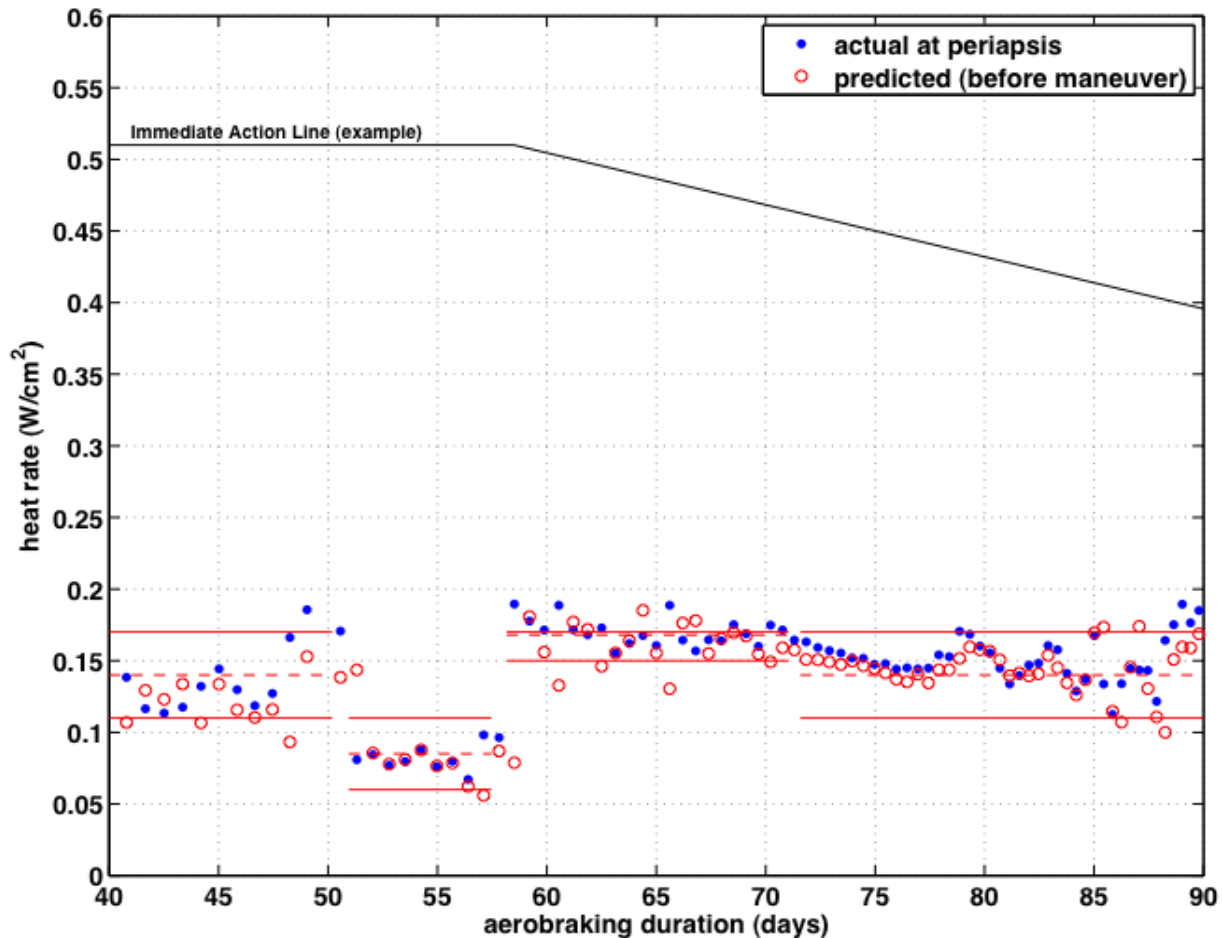
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:


56 of 286



*Figure 7.5-9b. AADS Mission Operations Corridor Performance for the Glideslope Delay and Recovery Corridor*

Note that Figure 7.5-8 shows that not only is the nominal mission glideslope recovered, but the glideslope is steeper. This is a result of the spacecraft being left high in the corridor at the end of the recovery period, much higher than during the same time period in the nominal case (see Figures 7.5-9a and b). Due to the specific orbit geometry, the natural tendency is for the spacecraft to remain near the top of the corridor for over 2 weeks (days 70–85 in Figure 7.5-9b) before the first maneuver is required. If it is deemed necessary to immediately return closer to the nominal glideslope, this could be achieved simply by lowering the top of the corridor (which increases spacecraft safety margin), resulting in AADS commanding a maneuver to push the spacecraft lower into the corridor, until the nominal glideslope is fully recovered. At that time, the corridor upper limit can be returned to its nominal value.

With AADS, this scenario is successfully achieved autonomously, with only three required corridor updates from the spacecraft: (1) delay the glideslope, (2) recover the glideslope, and

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 57 of 286	


(3) return to nominal glideslope. It is accomplished simply by changing the desired corridor and target, thus altering when and how many maneuvers are executed during the affected time period. In addition, it is significant to note that all of this is accomplished without the need to violate the nominal mission upper corridor limit, thus introducing no additional spacecraft risk. With ground-based AB operations, recovery of the nominal glideslope would likely not be possible without increasing the upper corridor limit due to the constraint on maneuver execution frequency.

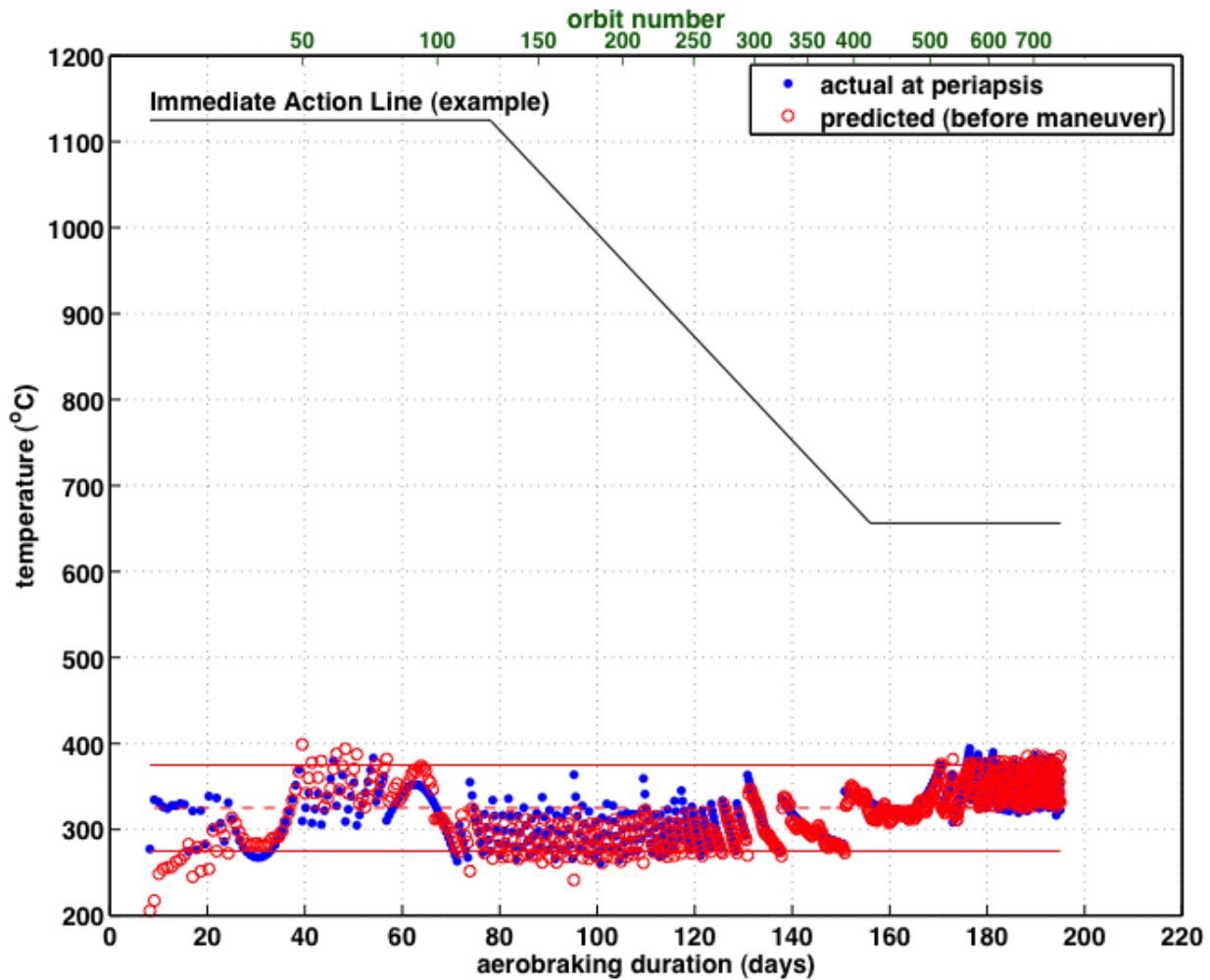
#### **7.5.1.4 AADS Performance at Venus and Comparisons to Reference Simulation**

At Venus, data from the Ephemeris and Atmosphere Estimators are used with a spacecraft thermal model to estimate the (spacecraft solar panel) temperature at periapsis during the next atmospheric pass. If the predicted temperature is outside of the operational corridor, then the same spacecraft thermal model is used to estimate the desired atmospheric density, given the predicted spacecraft orbital state at the next periapsis, and the desired target temperature within the corridor. From this information, a maneuver is calculated similarly to that for Mars (see Section 7.3.2.3).

Many other differences exist between AB missions at Mars and at Venus which affect the performance of the AADS. For example, at Venus, the effects of the Sun, both in terms of third body gravitational effects and solar radiation pressure, have a real influence on the spacecraft orbit. This, coupled with the higher orbital velocities (due to the more massive planet) and the much higher atmospheric density, is what would likely drive a mission operations team to utilize a temperature corridor during AB, as opposed to the heat rate corridor used at Mars. In addition, the atmospheric perturbations at Venus (as provided by Venus-GRAM) are much less than those at Mars. This means that at Venus, as it is at Mars, an update frequency up to 14 days or longer will provide sufficient AADS performance to ensure spacecraft safety throughout the AB mission.


A summary of the AADS performance within the mission operations temperature corridor for the Venus AB “truth” simulation with a perturbed atmosphere (and 7-day updates) is provided in Figure 7.5-10.

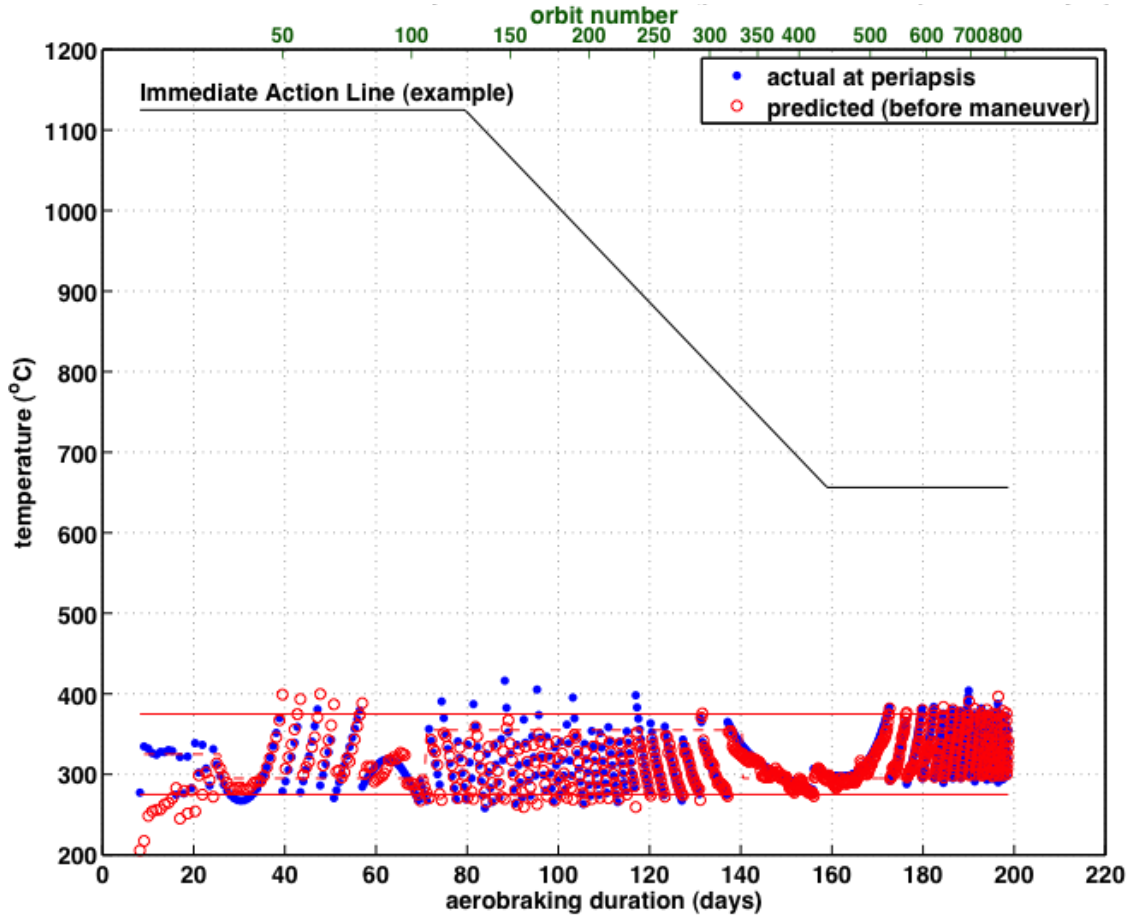
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 58 of 286	



*Figure 7.5-10. AADS Mission Operations Corridor Performance at Venus with a Perturbed Atmosphere (and 7-day Updates) and a Fixed Corridor Target*


The strong solar effects are visible where the spacecraft is pulled to either the top of the corridor or the bottom, depending on where the Sun is located relative to Venus and the spacecraft. To better take advantage of these effects, the target within the corridor can be varied during the mission using the weekly update capability, as illustrated in Figure 7.5-11. This approach can reduce the number of maneuvers required to maintain the operational corridor without the need to change the corridor itself.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 59 of 286	



*Figure 7.5-11. AADS Mission Operations Corridor Performance at Venus with a Perturbed Atmosphere (and 7-day Updates) and a Varying Corridor Target*

Table 7.5-5 provides a summary comparing the Venus AADS baseline performance, using a perturbed atmosphere and 7-day updates, for both the fixed and varying corridor target, with the reference simulation analysis. The difference in the duration between the AADS and reference simulation can be attributed to the differences in the (nominal) corridor target. It is evident that utilizing a variable target enables a reduction in the number of maneuvers.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		60 of 286	

**Table 7.5-5. Summary of AADS Performance with Perturbed Atmosphere (and 7-day Updates) Compared to Reference Simulation at Venus**

	<i>AADS Fixed Target</i>	<i>AADS Variable Target</i>	<i>Reference Simulation</i>
<b>AB Duration (days)</b>	195.0	198.7	174.8
<b>Total <math>\Delta V</math> (m/s)</b>	21.8	22.7	18.0
<b>Number of Maneuvers</b>	86	67	86

### 7.5.1.5 AADS Performance at Titan and Comparisons to Reference Simulation

At Titan, data from the Ephemeris Estimator is used to estimate the spacecraft altitude at periapsis during the next atmospheric pass. If the predicted altitude is outside of the operational corridor, then the difference between the predicted and desired target altitude within the corridor is used to calculate the required correction maneuver. This change in altitude is then added to the current estimated periapsis altitude, which is then added to the estimated apoapsis altitude to determine a new orbit semi-major axis, from which a new velocity at apoapsis is determined. The difference between this new apoapsis velocity and the current estimate of the apoapsis velocity is the required maneuver magnitude. As with Mars and Venus, this value is positive for a periapsis raise (decrease freestream heating rate) and negative for a periapsis lowering (increase freestream heating rate), and the maneuver direction is estimated to be that of the pre-maneuver velocity vector at apoapsis.

More so than for Venus, AB at Titan must account for strong third body effects, but in this case, from Saturn. These gravitational effects can once again be utilized to design the corridor target to minimize the number of maneuvers required to maintain spacecraft safety within the corridor. As was the case with Venus, the atmospheric perturbations provided from Titan-GRAM are small compared to those from Mars-GRAM (and are even smaller than those from Venus-GRAM as well). In addition, since this Titan analysis makes use of an altitude corridor, atmospheric perturbations (in density) (in conjunction with the large atmospheric scale height) have little to no effect on the spacecraft location within the corridor itself.

A summary of the AADS performance for the Titan AB mission with a perturbed atmosphere (and 7-day updates) is provided in Figure 7.5-12. Table 7.5-6 provides a summary comparing the AADS mission with the reference simulation analysis. As is the case with Mars and Venus, an update frequency up to 14 days or longer will provide sufficient AADS performance to ensure spacecraft safety throughout the AB mission.





# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-09-00605**

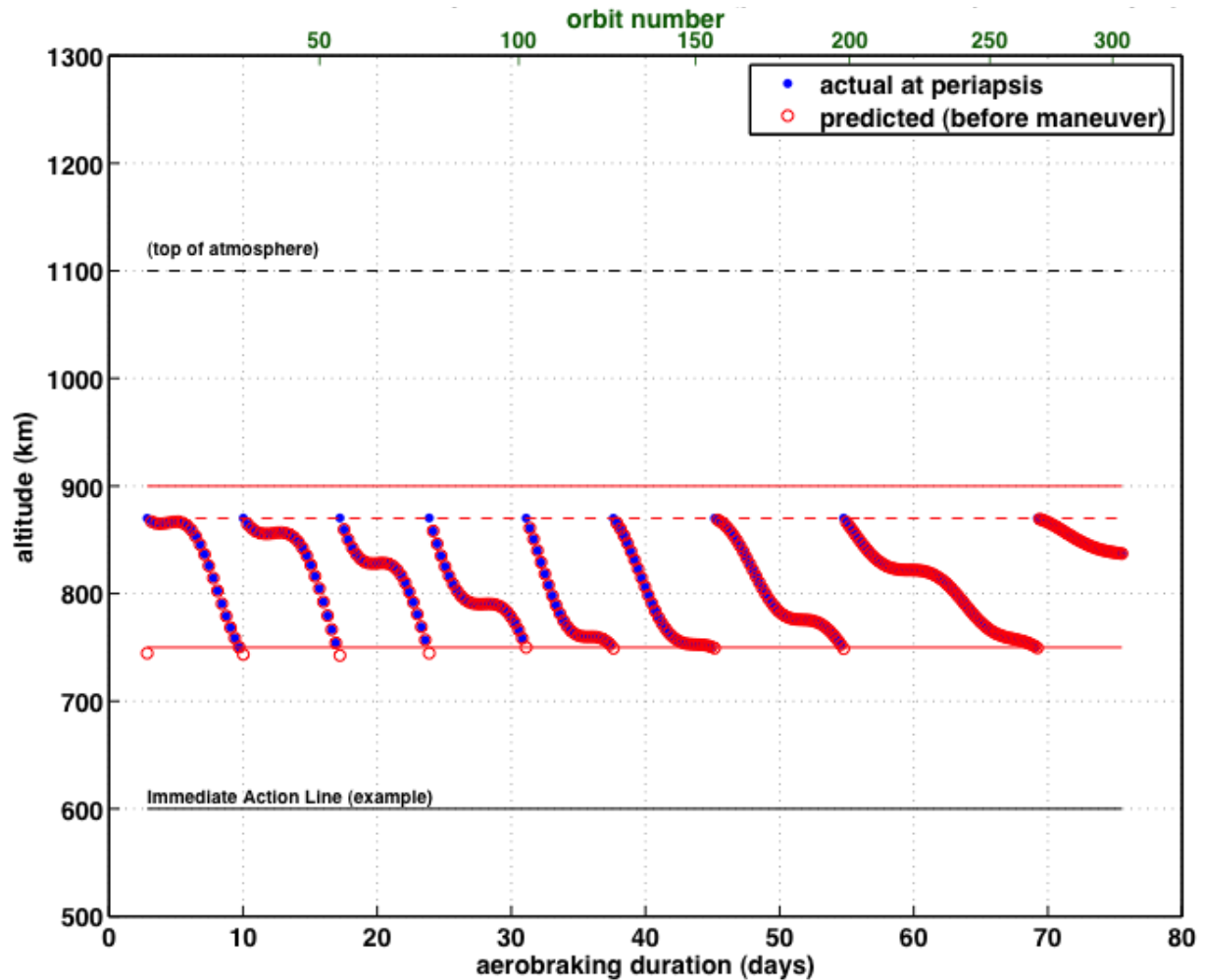
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:


61 of 286



*Figure 7.5-12. AADS Mission Operations Corridor Performance at Titan with a Perturbed Atmosphere (and 7-day Updates)*

*Table 7.5-6. Summary of AADS Performance with Perturbed Atmosphere (and 7-day Updates) Compared to Reference Simulation at Titan*

	AADS	Reference Simulation
AB Duration (days)	75.5	74.1
Total $\Delta V$ (m/s)	105.0	105.4
Number of Maneuvers	9	9


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 62 of 286	

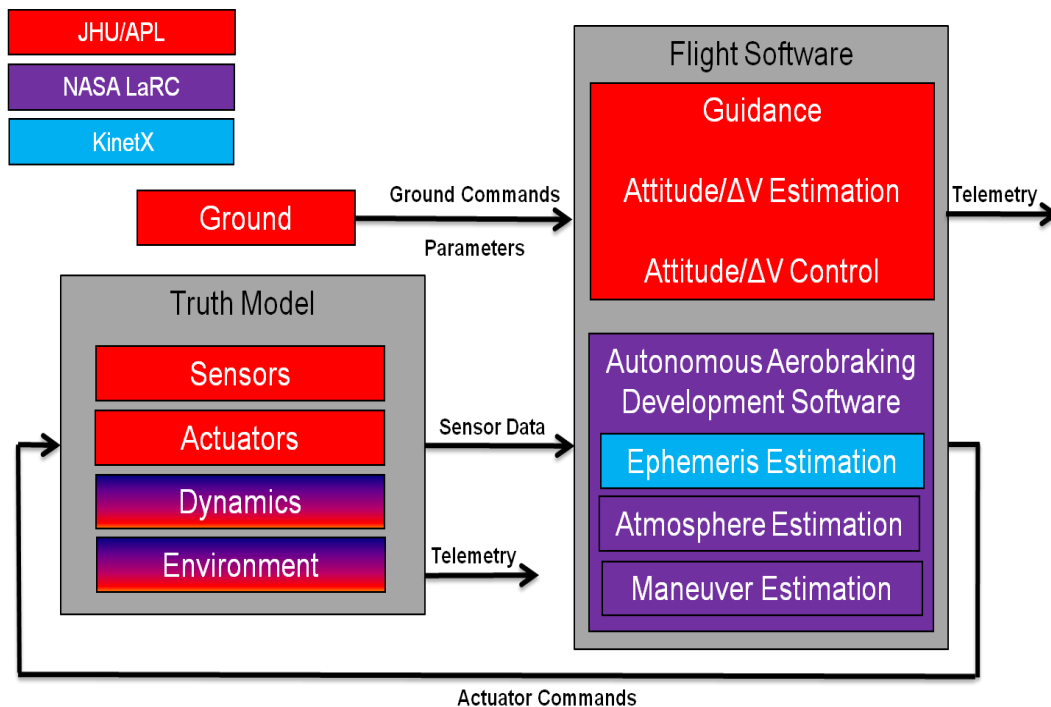
## 7.5.2 AAHFS

### 7.5.2.1 AAHFS Modeling

AAHFS has been developed to assess the performance of the new algorithms required for an AA mission. This 6-DOF simulation is based on the flight software and truth models previously developed for JHU/APL-designed MESSENGER spacecraft, currently orbiting Mercury [ref 14]. This AB demonstration has been developed in the MATLAB/Simulink/RTW environment. It is comprised of the existing MESSENGER algorithms and software, to which the AB flight system and truth model test bed algorithms have been added. This approach permits rapid study of the AADS and, in the end, generates a high degree of confidence in the technology as a precursor to implementation in a flight program. Another important benefit to developing the software in this way is that the end product is suitable for use in a flight processor, as this study uses the same development environment/process as the MESSENGER guidance and control flight software.


Figure 7.5-13 shows the AAHFS software in block diagram form. Detailed block diagrams and descriptions further decomposing the truth model and flight software elements are provided in Appendix I. The color-coding of Figure 7.5-13 indicates the organization responsible for each software element. As previously mentioned, the JHU/APL software is MESSENGER heritage, although some adaptation to the AB application is necessary, particularly in the flight software. The truth model modifications provided by LaRC have a rich AB flight heritage. These elements include an atmosphere model (Mars-GRAM 2010) and an aerodynamics model (based on MRO) and are integrated into the MESSENGER Simulink environment to allow testing of the flight software. The AADS is new algorithm code developed specifically for this project and is described in detail in Sections 7.3 and 7.4 of this report. These algorithms perform the necessary calculations to ensure the spacecraft remains safe and performs AB in the desired manner with limited ground intervention. The AAHFS simulation is developed as a high-fidelity, reliable, validated test environment to demonstrate the performance of the AADS algorithms.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		63 of 286	



**Figure 7.5-13. AAHFS 6-DOF Block Diagram**

AAHFS provides a testing environment with spacecraft dynamic behavior in addition to the 3DOF analysis provided by the POST2 simulations. One key advantage of AAHFS is that the simulation environment has 6-DOF dynamics, enabling study of the interaction between the rotational and translational components. This is of particular importance during AB drag passes, where the vehicle orientation affects the drag accelerations that, in turn, change the evolution of the spacecraft orbit. Another example where AAHFS provides enhanced fidelity is the use of acceleration data in the AADS code. Instead of using the true acceleration in AADS, AAHFS passes this true acceleration through: (1) an IMU model which decimates the true acceleration to 100 Hz, (2) an emulation of spacecraft command and data handling system which collects and buffers these raw IMU measurements, (3) a high-rate (50-Hz) data processing task which takes the high-rate, raw accelerations and averages them down to a more appropriate rate for use in AADS, and (4) a data buffer which collects the acceleration data for use in AADS. Each step in the process can add uncertainty (e.g., noise, bias, scaling/rotational errors, latency, etc.) to these true accelerations, which can impact the fidelity of the trajectory integration in AADS. All of these processes must occur in a flight system and are present in AAHFS. Other examples where AAHFS can add realism to the simulation include the maneuver implementation and execution, use of gyroscope and star tracker data, modeling of attitude dynamics via the real commanding and use of actuator models, system angular momentum modeling, lower-order dynamic effects (propellant slosh, structural dynamics), and additional perturbation models for forces and torques.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 64 of 286	

### 7.5.2.2 AAHFS Testing

The capabilities provided by JHU/APL’s AAHFS allow for thorough testing of the AADS. As with the POST2 simulation results presented in Section 7.5.2.1, the AAHFS test environment allows evaluation of the AADS in a closed-loop simulation. All interactions with AADS in AAHFS are conducted autonomously, mimicking the execution of AADS for a real flight program. To date, the testing with AAHFS has focused on simulations at Mars, with Venus and Titan testing to be done in future phases of this development. The goal of the initial testing done with AAHFS is to demonstrate the technical capability of the AADS algorithms under nominal conditions. More extensive robustness testing of AADS is planned for Phase 2. It is desirable for the AAHFS results to show some level of consistency with the results obtained with POST2. In this way, the AAHFS simulation can be used to check the POST2 simulation results, and vice versa. Although POST2 and AAHFS have a significant number of differences, as highlighted earlier in this section, by configuring some benchmark simulations in a similar way, the results can be shown to be qualitatively similar. It is important to note that the following results are consistent but not identical. Identical results can be obtained in POST2 and AAHFS only when the AADS inputs are precisely controlled. A test where AADS was ported to the Simulink/RTW environment and run as a stand-alone (or unit test) model was conducted. This unit test allowed the inputs to be identical between a POST2 simulation and AAHFS, and under these circumstances, AADS produces identical results and demonstrates that the integration of AADS into AAHFS does not introduce any errors. For the results presented here, all AAHFS simulations use the 6-DOF AAHFS truth/flight software models to generate the input data to AAHFS, and so an identical match to the POST2 results is not feasible.

Several simulations were done to demonstrate the technical capability of AA and to show the consistency of AAHFS with the POST2 simulation results described in Section 7.5.1. As the goal of this effort is to demonstrate the viability of AADS for a week without ground intervention, these initial comparisons are conducted using a ground update interval of 7 days. For these initial comparisons, all disturbance sources to the accelerations have been disabled in AAHFS to mimic the way POST2 was run. This is a recognized shortcoming of the testing to date and will be addressed in Phase 2 of this development effort. Additionally, the only environment models that were enabled for the “truth” model portion of these simulations are those included in the POST2 “truth” simulation. For example, the Mars gravity field uses a harmonic model (of degree and order 21) and the Sun as a third-body perturbation in both of the “truth” models. Adding fidelity to the truth models (e.g., higher order gravity field, or additional orbital perturbations, etc.) to study the robustness of AADS to un-modeled dynamics is left for Phase 2 of this development.

Figure 7.5-14 shows the corridor performance for an AAHFS simulation using a nominal atmosphere at Mars. This figure shows the heat rate corridor performance of AADS versus the true conditions (from the AAHFS truth model), as is plotted using the same scale as the heat rate plots from Section 7.5.1 for easy comparison. This simulation compares favorably with the

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 65 of 286	

results obtained with POST2, shown in Figure 7.5-5. In both the POST2 and AAHFS simulation, the spacecraft remains within the corridor on nearly every orbit. It is important to note that this is despite the AAHFS being a more challenging simulation environment, mostly because the simulation is exercising all 6 DOF. AADS handles the additional complications of imprecision in the corridor control maneuvers and more flight-like accelerometer data collection methods without any notable issues. Figure 7.5-15 demonstrates the accuracy of the AADS in predicting the orbit period between the 7-day ground updates (shown as vertical dashed red lines in the figure). When a ground update occurs, the AADS orbital propagation is reset to the true state, and the error in the periapsis timing estimate is reset. The growth of this timing error in AADS is simply due to imprecision in modeling, capturing, and integrating the forces on the spacecraft, although for the case shown in Figure 7.5-15, the error remains below ~3 seconds. Improving the precision in the integration will be a subject of Phase 2 investigations, as imprecision in predicting periapsis is one of the main factors that would require shortening the ground update interval. Figure 7.5-16 shows the glideslope for this AAHFS simulation versus an identical run with POST2. Despite the differences between the simulations, the glideslopes are consistent between the two runs, as expected. The lower pane of the graphic shows an expanded view of the differences (POST2 period minus AAHFS period), since it is difficult to discern in the upper pane.

An AB phase using AADS was also simulated using a 14-day ground update interval with the nominal Mars atmosphere and these results are presented in Figures 7.3-17–19. While the corridor performance appears similar to the 7-day update case in Figure 7.3-13, the periapsis timing estimates from AADS are drifting farther from the true values, due to the longer time between ground contact. Although the overall performance of this simulation is still good (as indicated by the heat rate corridor), using a 14-day ground update interval is showing signs of stressing the AADS algorithms. Glideslope performance for this AAHFS simulation is compared with the POST2 results in Figure 7.3-19, demonstrating that despite the differences between the simulations, the AA phase is happening at a similar rate.



# NASA Engineering and Safety Center Technical Assessment Report

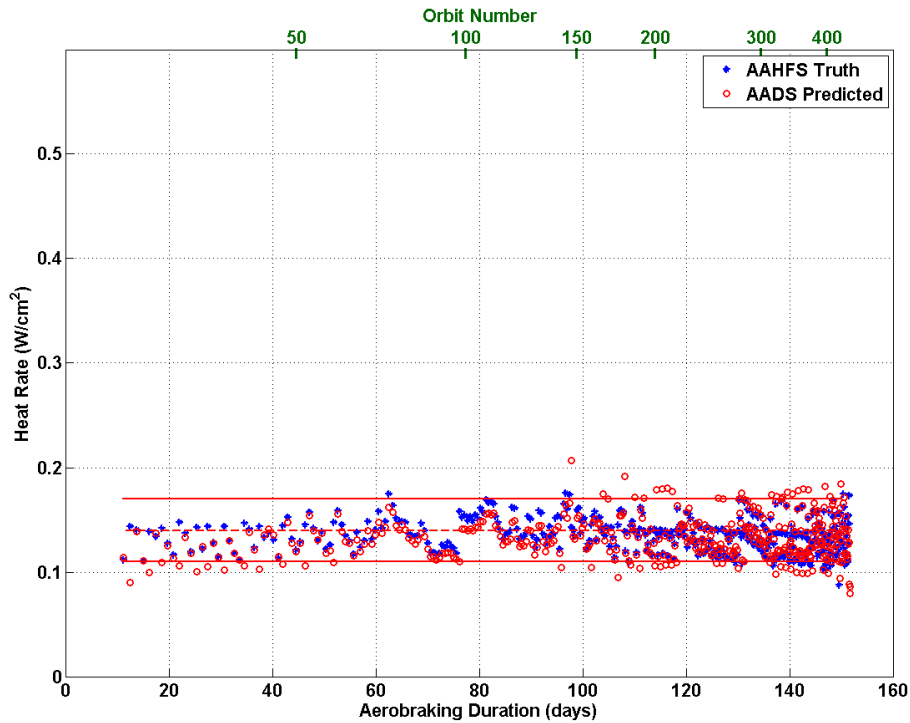
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
66 of 286



*Figure 7.5-14. AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Nominal Atmosphere and 7-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

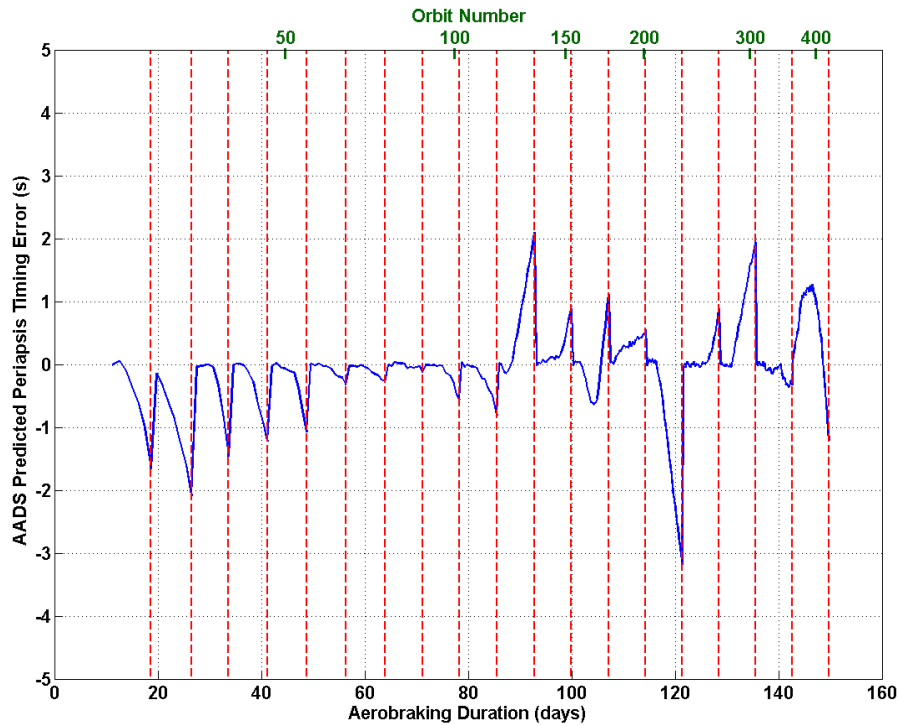
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

67 of 286



*Figure 7.5-15. AAHFS Simulation – AADS Orbital Period Performance at Mars with a Nominal Atmosphere and 7-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

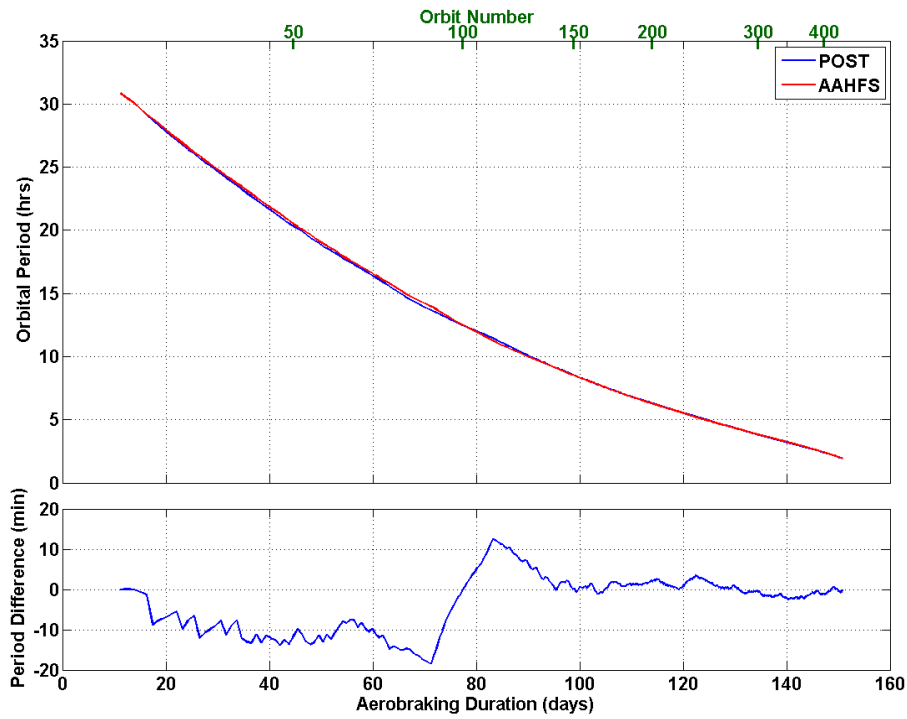
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
68 of 286



*Figure 7.5-16. AAHFS Simulation – Glideslope Performance at Mars with a Nominal Atmosphere and 7-day Updates*





# NASA Engineering and Safety Center Technical Assessment Report

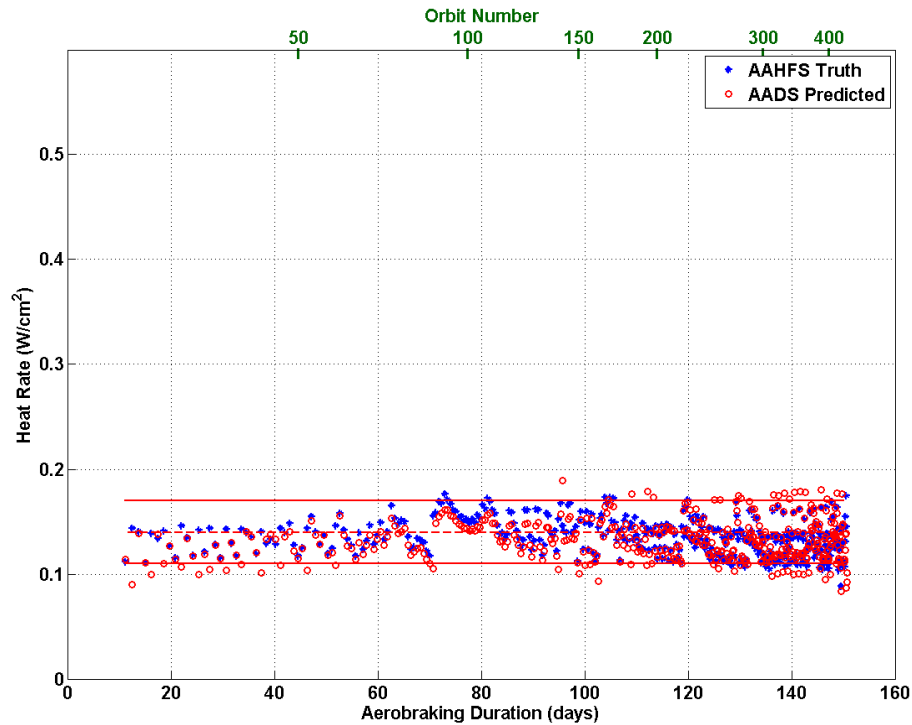
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
69 of 286



*Figure 7.5-17. AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Nominal Atmosphere and 14-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

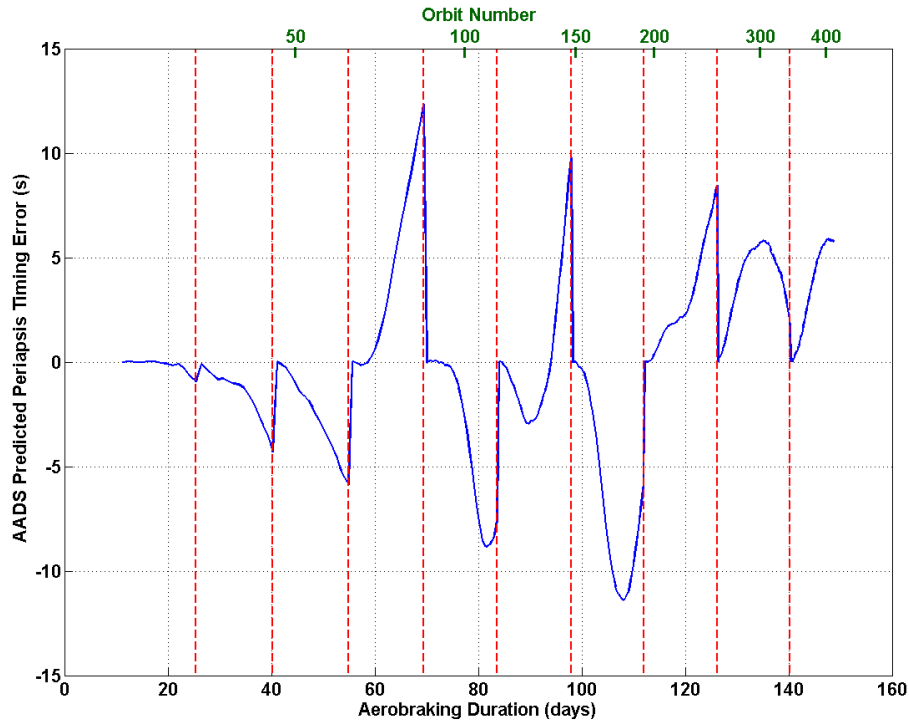
Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

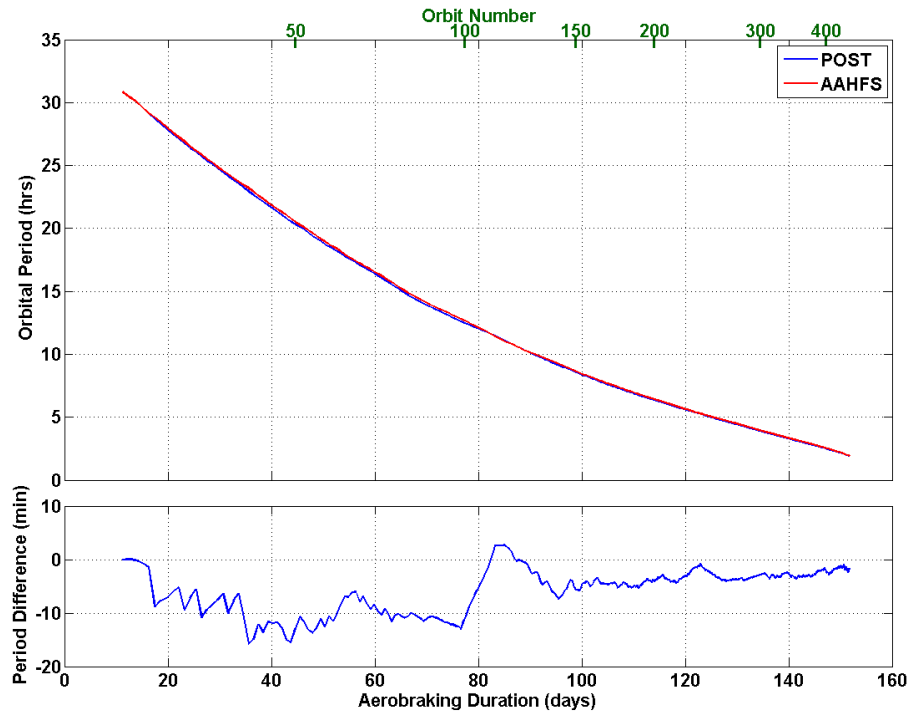
Page #:

70 of 286




*Figure 7.5-18. AAHFS Simulation – AADS Orbital Period Performance at Mars with a Nominal Atmosphere and 14-day Updates*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 71 of 286	

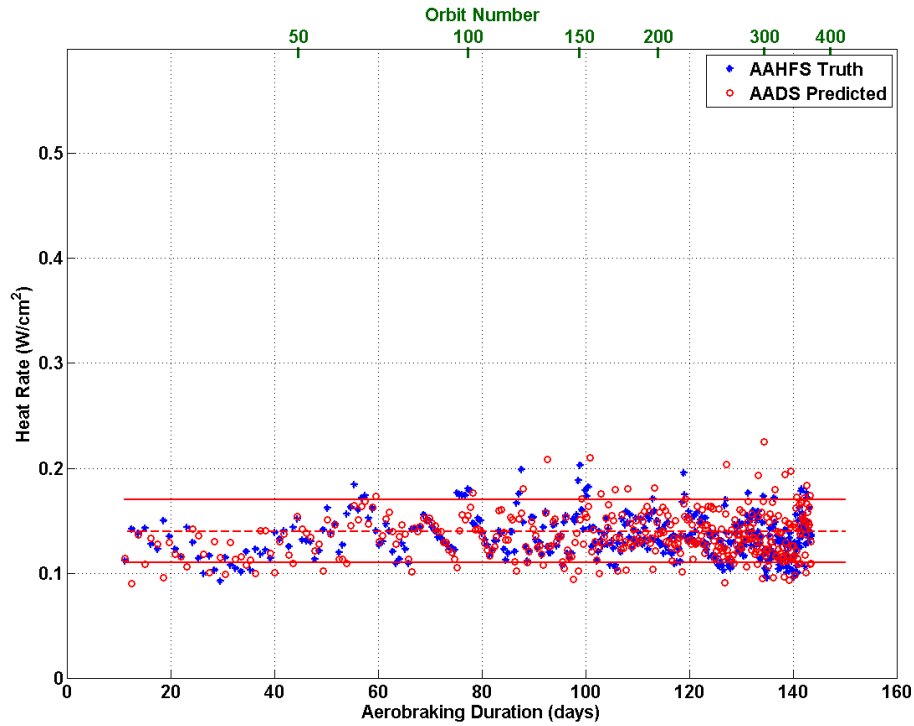


**Figure 7.5-19. AAHFS Simulation – Glideslope Performance at Mars with a Nominal Atmosphere and 14-day Updates**

The results shown in Figures 7.5-14 through 7.5-19 have only used the nominal atmosphere conditions coming from Mars-GRAM 2010. A more useful and challenging study is to use the Mars-GRAM 2010 perturbed densities for the AB simulations. The same two cases (7- and 14-day ground update intervals) were repeated using AAHFS with the density perturbations enabled. Because these density fluctuations are captured by the accelerometer data that is provided to AADS for the ephemeris and atmosphere estimation, these perturbations enhance the realism of the simulations, but they are not expected to significantly impact the AADS performance. Figures 7.5-20 through 7.5-25 demonstrate that as expected, AADS continues to perform well, despite the atmospheric perturbations. The increased variability in the atmosphere does produce a handful of small deviations from the corridor as the Atmosphere Estimator in AADS has more difficulty producing accurate estimates of atmospheric density for the perturbed case. The AADS-computed periapsis timing is consistent with the nominal atmosphere simulations, as expected, and the glideslope remains consistent with the nominal atmosphere cases (and the POST2 nominal atmosphere simulation results). The steady-state deviation in orbital period at the end of the simulation (when comparing AAHFS perturbed atmosphere versus the POST2 nominal atmosphere result), shown in the lower pane of Figures 7.5-22 and

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 72 of 286	

7.5-25, is simply an indication that the perturbed atmosphere case completed the AA phase slightly faster than the nominal atmosphere case.



*Figure 7.5-20. AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere and 7-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

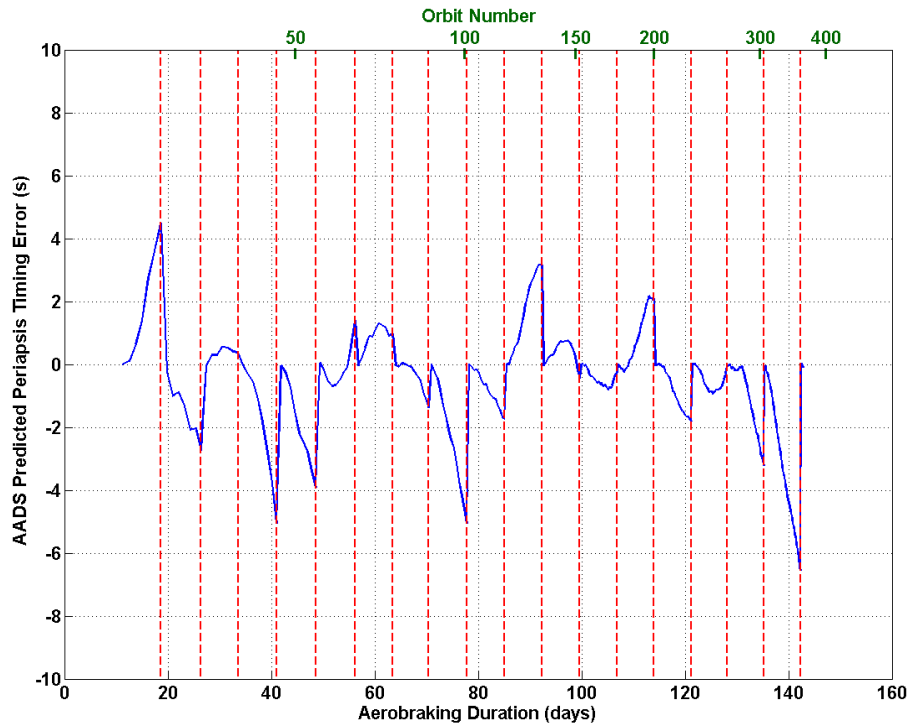
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

73 of 286



*Figure 7.5-21. AAHFS Simulation – AADS Orbital Period Performance at Mars with a Perturbed Atmosphere and 7-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

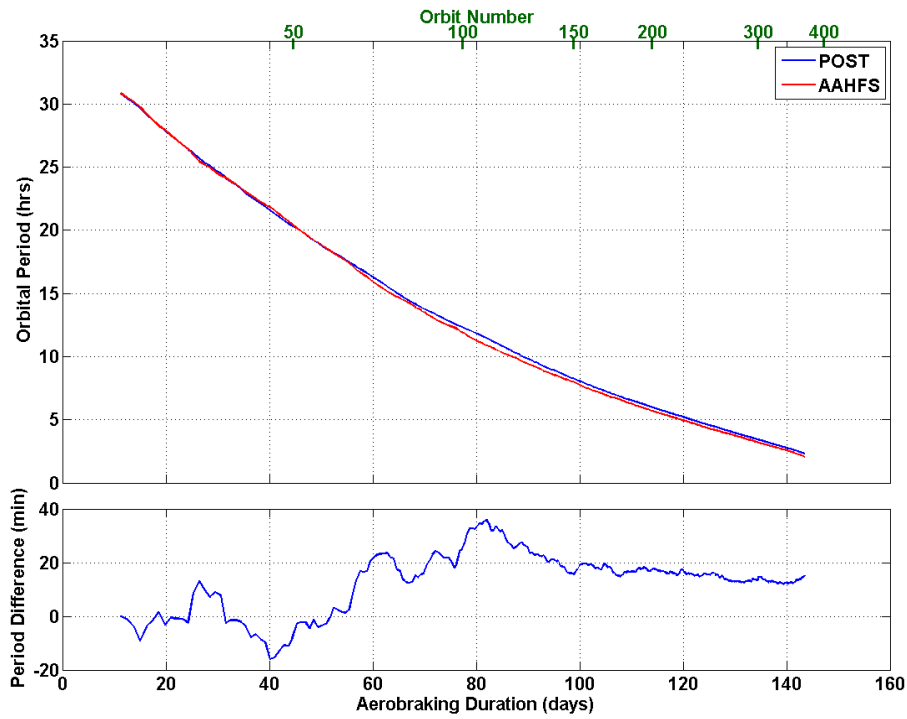
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

74 of 286



*Figure 7.5-22. AAHFS Simulation – Glideslope Performance at Mars with a Perturbed Atmosphere and 7-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

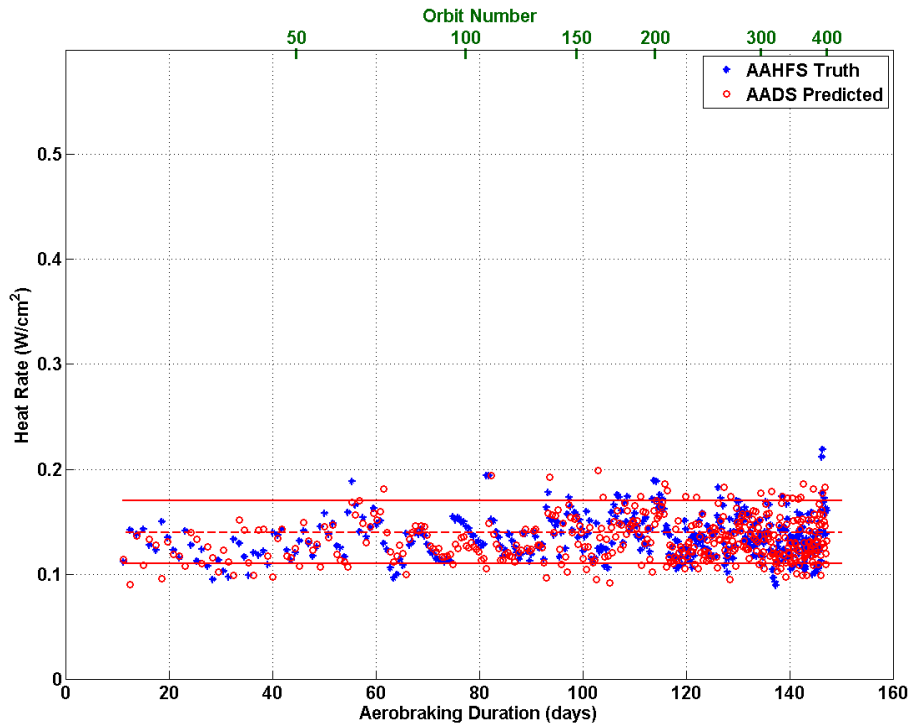
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

75 of 286



*Figure 7.5-23. AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere and 14-day Updates*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

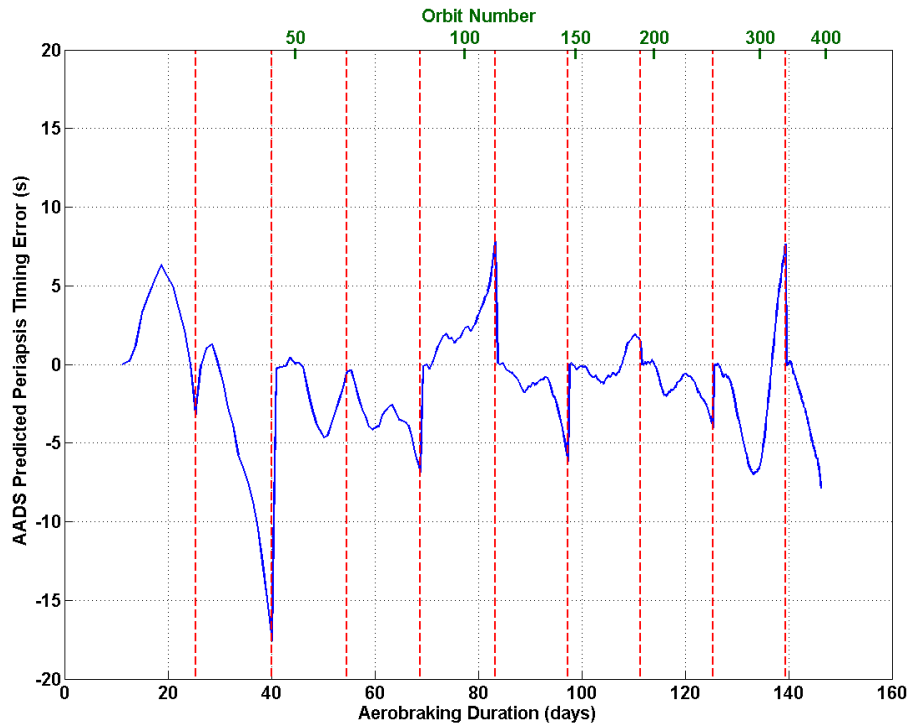
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)


Page #:

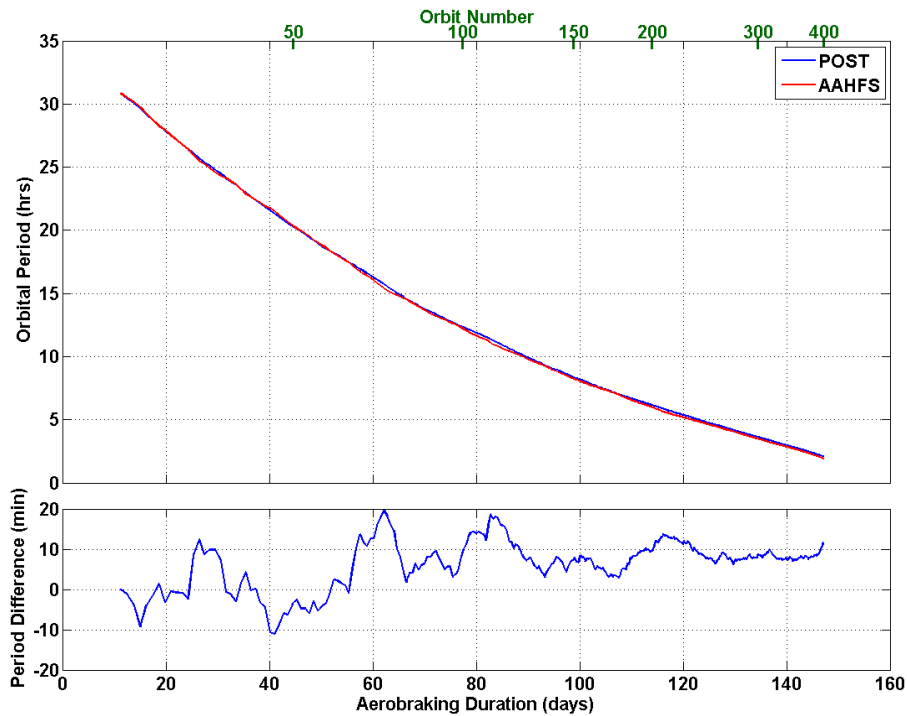
76 of 286



*Figure 7.5-24. AAHFS Simulation – AADS Orbital Period Performance at Mars with a Perturbed Atmosphere and 14-day Updates*



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 77 of 286	



**Figure 7.5-25. AAHFS Simulation – Glideslope Performance at Mars with a Perturbed Atmosphere and 14-day Updates**

Figure 7.5-6 demonstrated that the AB duration seen in the POST2 simulations could be manipulated by adjusting the heat rate corridor. This figure demonstrated that using a narrow corridor results in an increased maneuver frequency and propellant consumption necessary to fly the AB mission. The same conclusions can be demonstrated with AAHFS and are shown in Figures 7.5-26–29. It is more difficult for AADS to remain within the narrow corridor, particularly with the perturbed atmosphere in place, as the atmospheric variability makes accurate atmosphere estimation difficult. Despite this variability, AADS does a good job of keeping the spacecraft near the narrow corridor. This narrow corridor does have a price, for as with the POST2 simulations (Table 7.5-3); it is obvious from the AAHFS results in Figures 7.5-26 and 7.5-28 that there is a marked increase in the maneuver frequency required to maintain this narrow corridor.



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

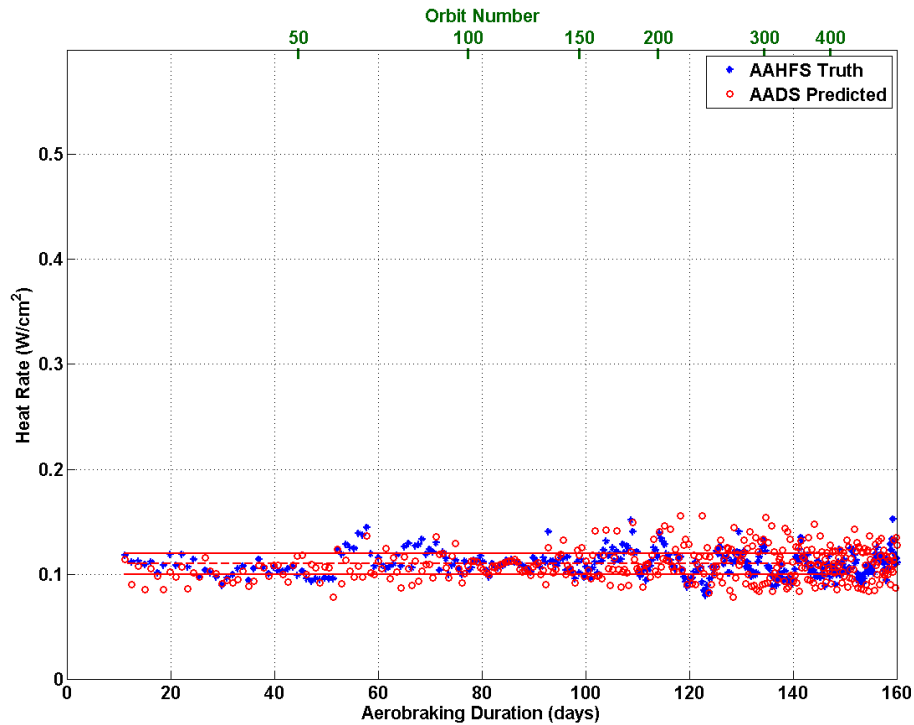
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

78 of 286



*Figure 7.5-26. AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere, 7-day Updates, and a Reduced Heat Rate Corridor*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

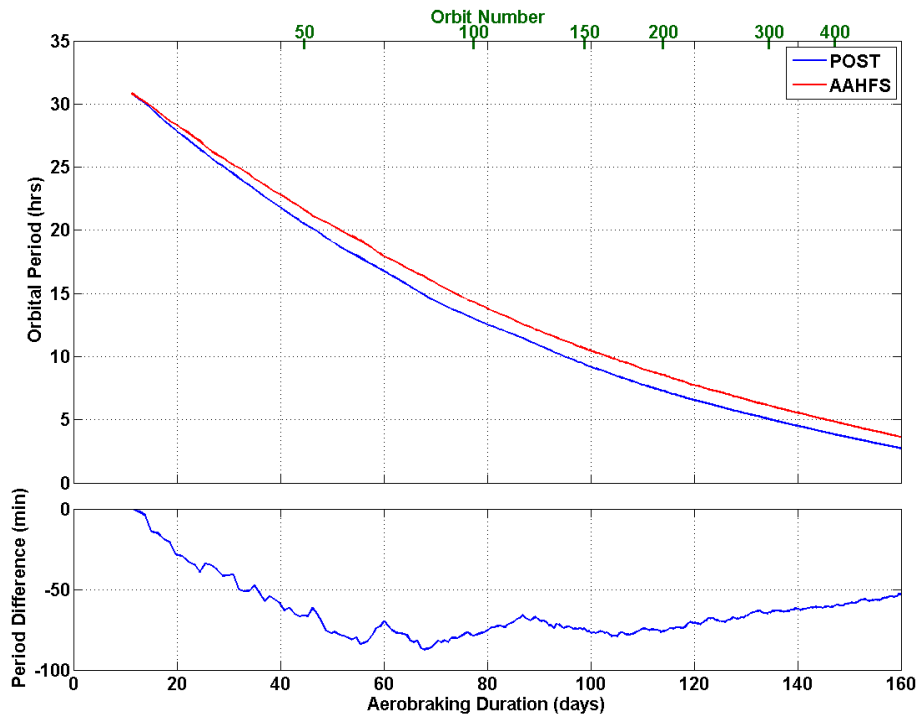
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

79 of 286



*Figure 7.5-27. AAHFS Simulation – Glideslope Performance at Mars with a Perturbed Atmosphere, 7-Day Ground Updates, and a Reduced Heat Rate Corridor*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

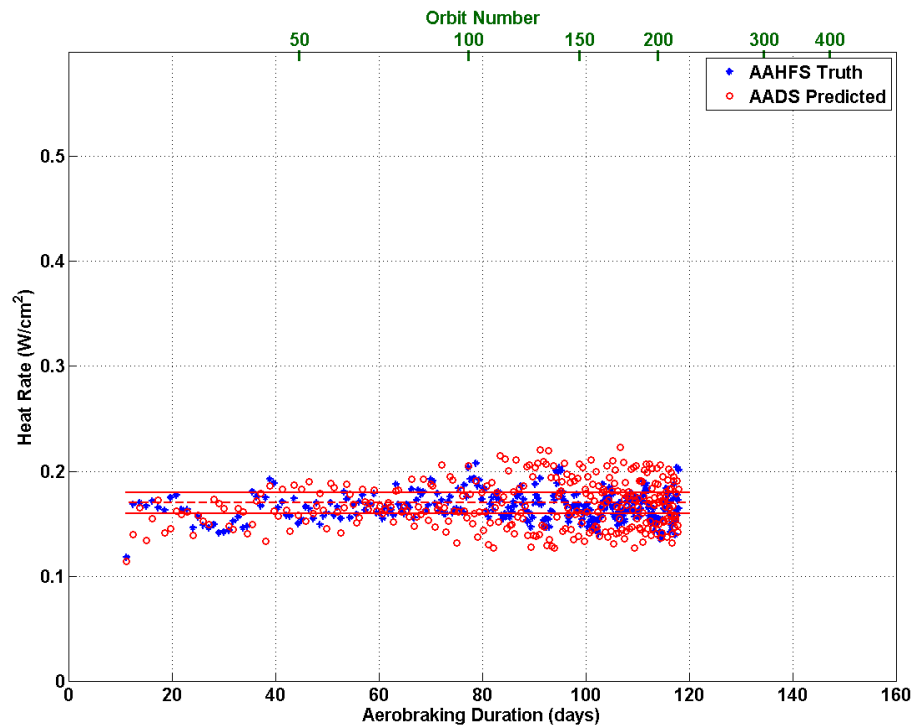
Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

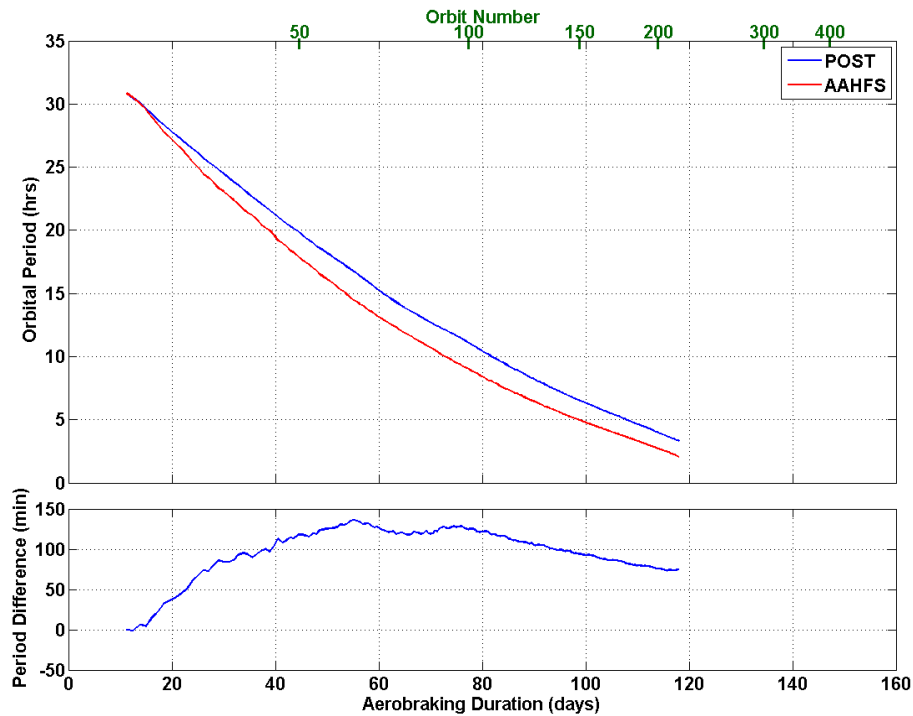
Page #:

80 of 286




*Figure 7.5-28. AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere, 7-day Updates, and an Elevated Heat Rate Corridor*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 81 of 286	



**Figure 7.5-29. AAHFS Simulation – Glideslope Performance at Mars with a Perturbed Atmosphere, 7-Day Ground Updates, and an Elevated Heat Rate Corridor**

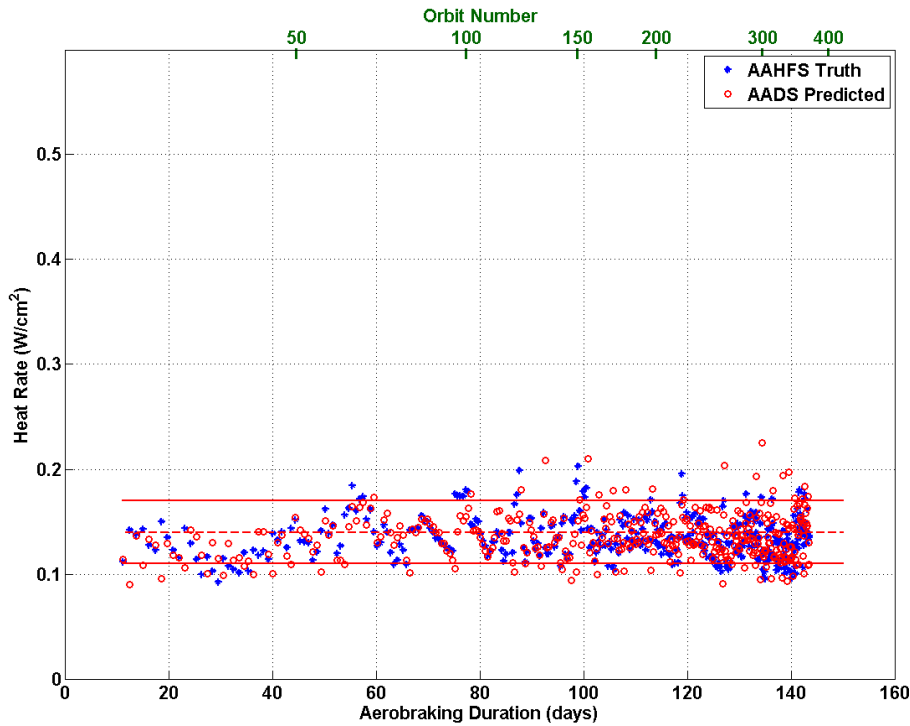
The results of Section 7.5 focused on testing the AADS with near perfect knowledge of the true non-gravitational accelerations. The POST2 results of Section 7.5.1 used decimated truth accelerations directly in AADS. The AAHFS results of Section 7.5.2 used an IMU model, but the error sources typically found in the accelerations had been disabled for the test results presented to this point. Although a full study of AADS robustness to accelerometer errors will be undertaken in Phase 2, it is useful to demonstrate that AADS shows promise even in the presence of typical accelerometer errors. Because AAHFS already has a full IMU model in the software, it is simply a matter of enabling it (with an appropriate parameterization) to study the AADS performance. The results of this simulation are shown in Figures 7.5-30 through 7.5-32, and the accelerometer error sources are tabulated in Table 7.5-7. These error sources were based on the flight performance of the MESSENGER IMU and represent typical performance of an interplanetary spacecraft. This simulation case was simply a repeated run of the case shown in Figures 7.5-20 through 7.5-22 (e.g., 7-day updates, perturbed atmosphere, nominal corridor bounds), and looks similar in terms of the heat rate corridor and glideslope performance. The AADS periapsis timing prediction is drifting more rapidly from the true value, which is not surprising since the orbit knowledge is largely determined by the precision of the accelerations.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		82 of 286	

This increased periapsis timing error does not adversely impact the corridor performance and is an indication of how robust AADS currently is to accelerometer errors. This case is a strong indicator that AADS can be used to conduct AA with a 7-day ground update interval.

**Table 7.5-7. AAHFS Simulation – Accelerometer Error Sources per Accelerometer**

<i>Random Noise</i>	<i>Bias</i>	<i>Readout Noise</i>	<i>Scale Factor</i>	<i>Misalignment</i>
1.5e-4 m/s <sup>2</sup>	7.2e-4 m/s <sup>2</sup>	1 count	240 ppm	150 arcs



**Figure 7.5-30. AAHFS Simulation – AADS Heat Rate Corridor Performance at Mars with a Perturbed Atmosphere, Accelerometer Errors, and 7-day Updates**



# NASA Engineering and Safety Center Technical Assessment Report

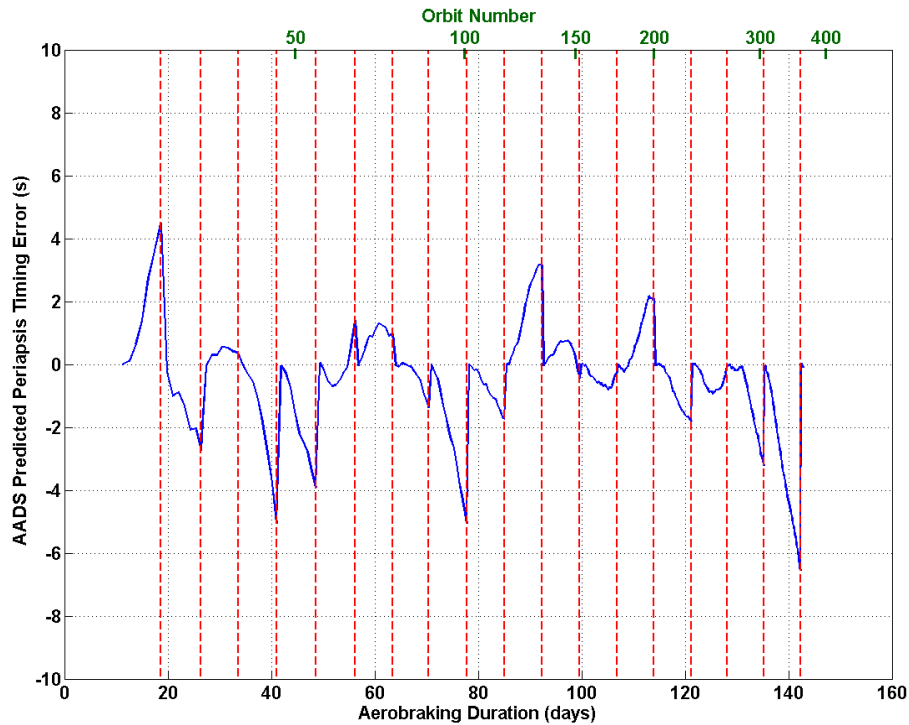
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**


Title:

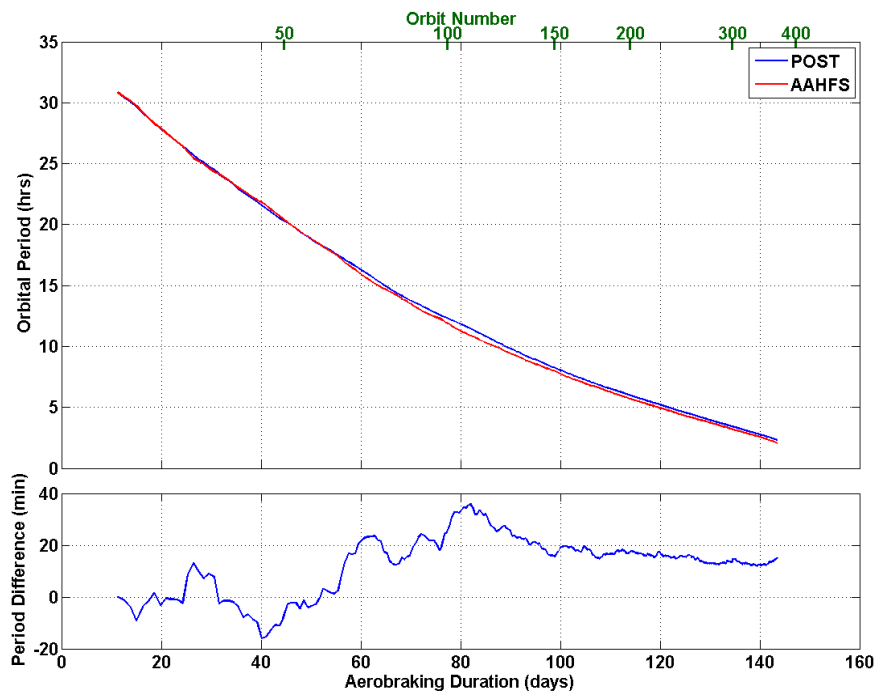
## Autonomous Aerobraking (Phase 1)

Page #:  
83 of 286



*Figure 7.5-31. AAHFS Simulation – AADS Orbital Period Performance at Mars with a Perturbed Atmosphere, Accelerometer Errors, and 7-day Updates*


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 84 of 286	



**Figure 7.5-32. AAHFS Simulation – Glideslope Performance at Mars with a Perturbed Atmosphere, Accelerometer Errors, and 7-day Updates**

Table 7.5-8 summarizes the cases run with AAHFS. This table shows the number of maneuvers and total  $\Delta V$  each simulation required as well as the duration of the AB phase. The relationship between the corridor width and the maneuver frequency (and/or  $\Delta V$  requirements) is a trade study that will be examined in Phase 2, but it is obvious that reducing the width of the corridor can greatly increase the frequency of corridor control maneuvers. Despite the much higher frequency of burns for the narrow corridor cases, it is not expected to pose any problems for the spacecraft propulsion system hardware. It can be seen from Table 7.5-5 that using a perturbed atmosphere and/or changing the ground update interval has little impact on the number of burns or total  $\Delta V$ . The AB duration is slightly shorter for perturbed atmosphere cases versus the nominal atmosphere, but the AB duration is greatly impacted by flying an elevated or reduced corridor, as seen in Cases 5 and 6. (Note that Case 5 only ran the simulation to 160 days, and it is estimated that the AB phase would have required an additional 15–20 days to complete.) For Cases 5 and 6, the corridor changes greatly influence the glideslope behavior, which is readily seen in Figures 7.5-26 and 7.5-28. The glideslope performance of these two cases demonstrates that the duration of AB can be managed by either raising the entire corridor, which may raise mission risk, or by paying a  $\Delta V$  penalty to fly a narrow corridor by leaving the upper heat rate corridor bound alone and raising only the lower corridor bound. Table 7.5-8 demonstrates that when IMU error sources were enabled in Case 7, they did not negatively impact the AB metrics when compared to Case 3.



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		85 of 286	

**Table 7.5-8. AAHFS Simulation – AADS Performance at Mars**

	<i>Atmosphere Model</i>	<i>Ground Update Interval (days)</i>	<i>Heat Rate Corridor Upper Limit (W/m<sup>2</sup>)</i>	<i>Heat Rate Corridor Lower Limit (W/m<sup>2</sup>)</i>	<i>Number of Maneuvers</i>	<i>ΔV (m/s)</i>	<i>AB Duration (days)</i>	<i>Figure Numbers</i>
<b>Case 1</b>	Nominal	7	0.11	0.17	70	18.2	152	7.5-14-16
<b>Case 2</b>	Nominal	14	0.11	0.17	64	16.7	150	7.5-17-19
<b>Case 3</b>	Perturbed	7	0.11	0.17	60	15.5	143	7.5-20-22
<b>Case 4</b>	Perturbed	14	0.11	0.17	68	16.5	147	7.5-23-25
<b>Case 5</b>	Perturbed	7	0.10	0.12	194	24.3	>160	7.5-26 and 7.5-27
<b>Case 6</b>	Perturbed	7	0.16	0.18	222	29.4	118	7.5-28 and 7.5-29
<b>Case 7 (IMU Errors Enabled)</b>	Perturbed	7	0.11	0.17	59	14.5	145	7.5-30-32

Figure 7.5-33 demonstrates the AADS heat rate performance across all AAHFS cases. The left-hand graphics show histograms of the true heat rate from each of the orbits in the various simulations tabulated in Table 7.5-8. The red lines on these histograms represent the corridor bounds used for that case. In this view, every orbit that exceeds these bounds represents one maneuver. With this idea, it can be seen that Cases 5 and 6 with narrow corridor bounds had a great number of cases outside the corridor. Those plots show that by enlarging the corridor by another 0.02 W/cm<sup>2</sup>, it is likely that the number of maneuvers could be greatly reduced. Another observation that is apparent when comparing all the cases in this way is that the corridor performance is not greatly affected by perturbing the atmosphere (Cases 3–4), increasing the ground update interval (Cases 2, 4), or adding accelerometer errors (Case 7). The right-hand graphics show histograms of the heat rate difference, computed as true heat rate minus AADS predicted heat rate. From these histograms, it is clear that perturbing the atmosphere does flatten the histogram versus the unperturbed atmosphere (Cases 3–7 versus Cases 1–2, respectively). This is not a surprising result, as the atmospheric variability will decrease the predictability and therefore the performance of the Atmospheric Estimator. These error statistics could be used to size the width of the corridor, since they provide some information about the reliability of the heat rate predictions for control decisions. More detailed performance estimates and trade studies are planned for Phase 2.



# NASA Engineering and Safety Center Technical Assessment Report

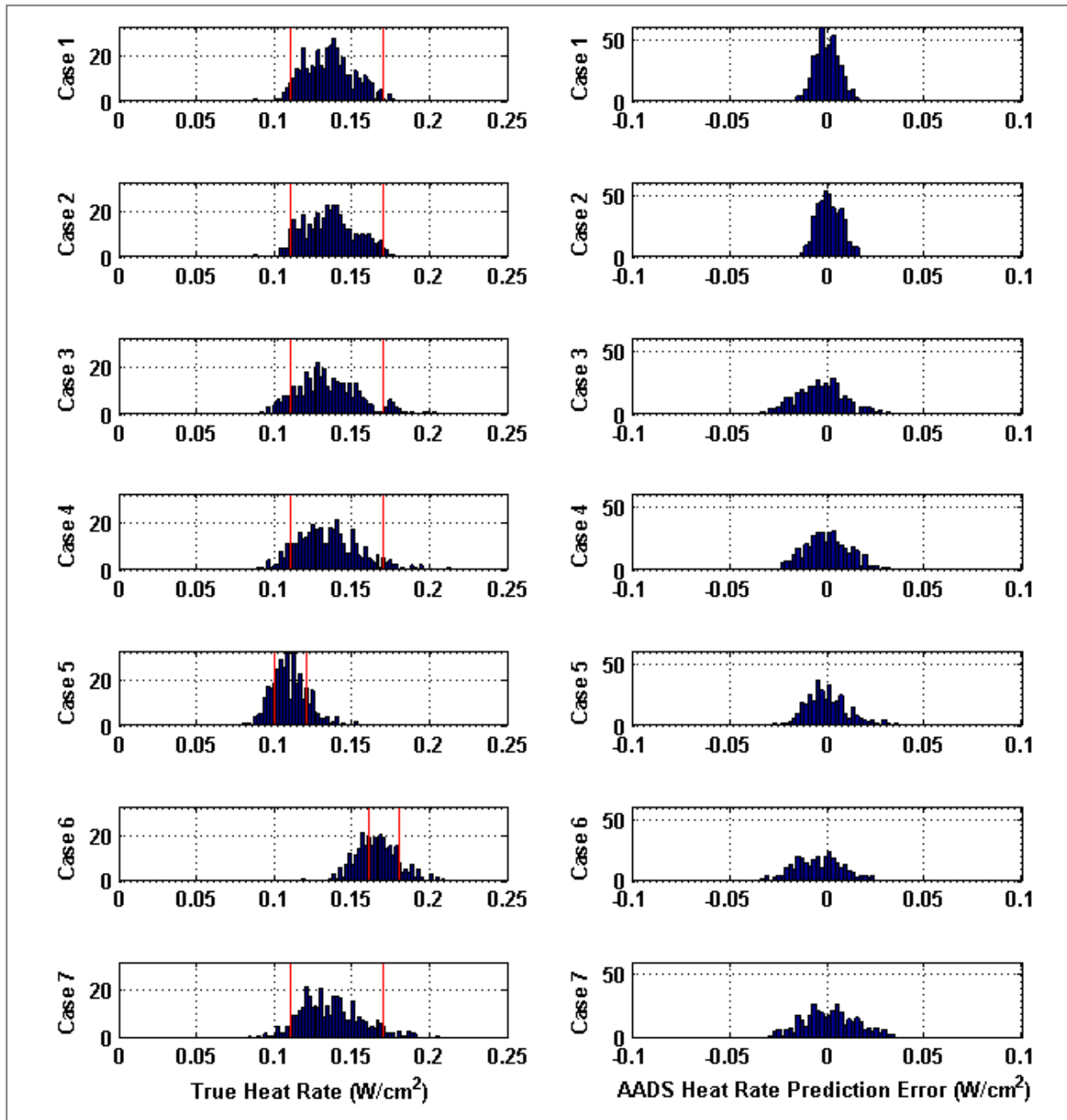
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

Page #:  
86 of 286



*Figure 7.5-33. AAHFS Simulation – AADS Heat Rate Performance Histograms for all AAHFS Cases*


The results shown in Figures 7.5-14–33 demonstrate that under a number of conditions, AADS is able to conduct AB with limited ground interaction. In all the cases presented, the thermal loads on the spacecraft during every drag pass remain well within design limits. While the orbital evolution has been shown to be well managed by AADS, there are attitude dynamics issues to

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 87 of 286	

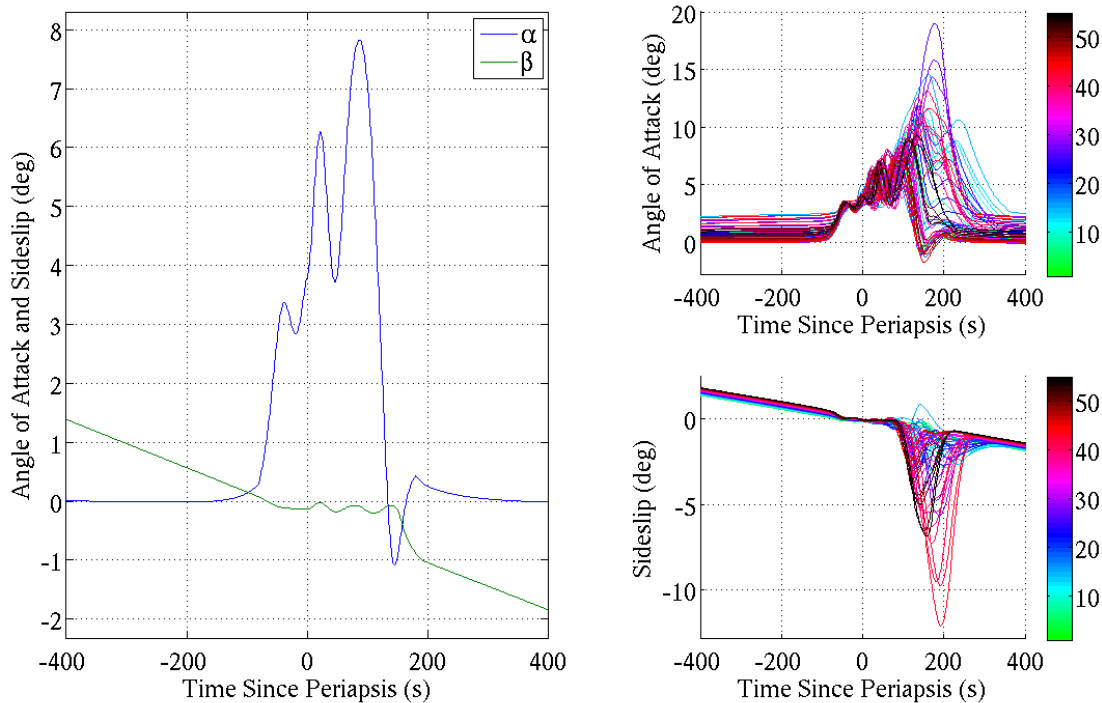
consider as well. While AADS does not directly control the spacecraft attitude, reliable periapsis timing and state information are necessary to ensure aerodynamic stability during the drag passes. For a typical spacecraft, attitude commanding is generally managed with periodic ground involvement, but for an AA spacecraft, the attitude commanding must be autonomous as well. The spacecraft must be capable of taking AADS outputs and generating the necessary commands to configure the spacecraft for the drag passes and any corridor control maneuvers that are necessary. If this process fails, it could lead to an errant (or skipped) corridor control burn or aerodynamic instability during the drag passes. Further, the timing computed by AADS determines the enabling/disabling of the accelerometer data buffers, switching between reaction wheel and thruster control, wheel momentum dump enabling, apoapsis burn ignition timing, and accelerometer bias estimation. Errors in sequencing these commands can rapidly degrade the performance of the AA system as a whole and potentially jeopardize the vehicle.

The aerodynamic stability of the spacecraft during the drag passes is shown in Figure 7.5-34. The left-hand pane of the graphic shows the performance of the aerodynamic angles (angle of attack,  $\alpha$ ; and sideslip,  $\beta$ ) during an initial drag pass. The initial angles entering the drag pass are near enough to the aerodynamic stability point (approximately  $\alpha=6^\circ$ ,  $\beta=0^\circ$ ) that when the aerodynamics overwhelms the reaction wheels as the atmospheric density increases, the vehicle moves to the aerodynamic stability point. After the spacecraft crosses periapsis, the atmospheric density decreases and eventually, the reaction wheels resume control of the vehicle. Throughout the deepest portions of the atmosphere, the control law uses the aerodynamic stability to dump the angular momentum in the wheels. During this time the thrusters are enabled for attitude control with wide deadbands to ensure controllability in the event of unanticipated aerodynamic effects. This is a simple example of how the vehicle may be controlled during a drag pass, although developing a control law for this purpose is beyond the scope of this project. The strategy used for these simulations was intended to be a typical approach. The important factors are that the vehicle enters the atmosphere near the aerodynamic trim point, and that the control law handles the atmospheric entrance and exit without any large ( $>30^\circ$ ) attitude excursions. The right-hand graphics show a number of consecutive (typical) orbits on the same graphic, indicating that as the AB phase unfolds, the attitude dynamics through the drag pass remain stable. In all cases, the attitude at entry (which is governed by the AADS-produced ephemeris information) is close to the aerodynamic trim point, so that the vehicle demonstrates the desired stability. The outbound portion shows some variation in the vehicle behavior, chiefly due to the low torque available to the reaction wheels to reign in the spacecraft body rates as the atmosphere dissipates. In all the simulations presented in this section, the vehicle shows this aerodynamic stability in all drag passes.

The testing presented in Section 7.5 is a precursor to demonstrating the AADS algorithms under more challenging circumstances. This testing has used somewhat idealized circumstances, as the truth model for these simulations matches the AADS configuration (no unmodeled dynamics) and most of the significant perturbations have been disabled. Despite this shortcoming, it is encouraging that AADS appears to be capable of conducting AB in a safe manner without

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 88 of 286	

ground intervention for longer durations (14 days) in the ideal cases. As noise and perturbations are added, it is expected that the ground update interval will be somewhat reduced. Much of this enhanced testing (and additional development) using AAHFS is planned for Phase 2.




*Figure 7.5-34. AAHFS Simulation – Aerodynamic Stability during Drag Passes*

### 7.5.3 Future Work

Results of Phase 1 have shown the functionality of AADS and the feasibility of using AADS to autonomously control the computation of corridor control maneuvers onboard a spacecraft. A number of improvements have been identified to AADS, the simulation tools, and the analysis methods that test AADS, to improve the robustness of the AA system. These improvements include:

AADS:

- Reduce errors in the Ephemeris Estimator by exploring new integration methods, step sizes, and real-time periapsis timing correction.
- Desensitize the Atmosphere Estimator to Ephemeris Estimator timing errors and investigate modeling methods independent of altitude. Incorporate 3-sigma estimates of atmospheric density in Maneuver Estimator logic.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 89 of 286	

- Modify the AADS software so that it is suitable for a flight implementation. This includes streamlining the code execution, adding error handling and fault detection and correction, and accommodating safe-mode events (safety triggers), off-nominal scenarios such as pop-up maneuvers and typical AB spacecraft contingencies.
- Assess the feasibility of incorporating collision avoidance in AADS.
- Add flexibility to the AADS maneuver logic and implement versatile control strategies that can meet project specific requirements. Currently, there are no constraints on the size of executed maneuvers. Operationally, there may be both lower and upper limits to the allowable size of maneuvers, or even a requirement to select from a menu of pre-selected (and pre-tested) maneuvers.

#### POST2:


- Improve the POST2 simulation capability by creating a 6-DOF “truth” simulation version, and by including the effects of IMUs, reaction wheels, thrusters, and other spacecraft models.

#### AAHFS:

- Complete the integration of Venus and Titan environment models into AAHFS and perform testing similar to the POST2 analyses for these bodies. (Support for any central body exists in AAHFS and Venus and Titan atmosphere models are already integrated into AADS, but the aerodynamic database and Venus-specific AADS version require integration and testing.)

#### Analysis Methods:

- Incorporate Mars AB flight-data-derived density profiles into “truth” simulations, identifying the impact of real density profiles with previously simulated “perturbed” atmospheres.
- Develop model uncertainties in AADS “truth” simulations and assess AADS performance against a mission simulation using Monte Carlo methods to identify robustness and further areas of improvement of AADS.
- Stress test AADS using atmospheric random noise and bias, initial ground errors, modeling errors, and fault management logic.
- Perform a sensitivity study on the required fidelity/order of the AADS (Ephemeris Estimator) gravity model to ensure sufficient accuracy while minimizing computational expenditures. This would likely trade performance accuracy against spacecraft resource requirements and availability.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 90 of 286	


- Conduct additional trade studies. Of primary interest is studying the impact of the accelerometer data buffer frequency on the Ephemeris Estimator orbit knowledge. Currently this buffering is done at 10 Hz, but lower data rates reduce the volume of data that must be buffered and therefore reduce the attendant processor memory requirements. Increasing data buffer frequency may improve Ephemeris Estimator performance but memory constraints may require careful architecture to allow real-time implementation in a flight processor. Additional trade studies will include determining accelerometer sensitivities (bias, noise, scale factor, alignment, etc.) and their impact on AADS; analyzing corridor width versus maneuver frequency; evaluating maneuver execution performance sensitivities; and quantifying the impact of solar radiation pressure.
- Benchmark AADS performance on a flight processor. Ensure the proposed software architecture is valid and that the code can run in allotted time (and/or central processing unit cycles).

## 8.0 Findings, Observations, and NESC Recommendations

### 8.1 Findings

The following findings were identified:

- F.1. Results from 3-DOF simulations with nominal and perturbed atmospheres at Mars, Venus, and Titan using POST2 indicate that AADS maintained the simulated spacecraft safely within or near the desired AB corridor, while allowing for 7 days between ground updates to the spacecraft ephemeris.
- F.2. Higher-fidelity 6-DOF results at Mars using AAHFS compared well with the 3-DOF results, where AADS maintained the simulated spacecraft safely within or near the desired AB corridor with 7-day ephemeris updates, and showed adequate performance with 14-day updates.
- F.3. The corridor control parameters used for all analyses were heat rate indicator at Mars, spacecraft temperature at Venus, and periapsis altitude at Titan.
- F.4. Differences between the corridor control parameters predicted by AADS and those calculated by the “truth” simulation were due primarily to errors in the ephemeris and Atmosphere Estimators.
- F.5. AA use may reduce AB risk by:
  - a. Conducting AB maneuvers at the optimal time and executed even if DSN or other required ground elements are unavailable.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 91 of 286	

- b. Narrowing the AB corridor width could provide more corridor margin for a spacecraft without concern for needing ground-based commands for higher frequency and total number of maneuvers.

**F.6. Current AADS limitations include:**

- a. Does not support spacecraft attributes needed for flight implementation, including efficient code execution, error handling, fault detection and correction, safe-mode events (safety triggers), off-nominal scenarios such as pop-up maneuvers and typical AB spacecraft contingencies.
- b. Maneuver logic may trigger excessive and unnecessary maneuver executions, especially in highly variable atmospheric conditions.
- c. Does not support collision avoidance.

**F.7.** Adding atmospheric variability to the simulations did not significantly impact the ability of AADS to ensure reasonable AB corridor performance.

**F.8.** Adding accelerometer errors to the simulation resulted in a more rapid divergence of the onboard ephemeris estimate from the “truth” orbit than atmospheric perturbations.

**F.9.** Ephemeris estimates, provided by AADS at the drag pass entry and exit, are sufficient to control the attitude of the vehicle throughout the drag pass.

**F.10.** Phase 1 analyses did not utilize uncertainty distributions on model parameters, Monte Carlo analyses, or stress testing of potential key error sources (e.g., ground initial conditions (spacecraft state and epoch), or onboard modeling errors).

## **8.2 Observations**


The following observations were identified:

- O.1.** Collision avoidance was a large factor in the MRO AB phase.
- O.2.** Maintaining two independent and synergistic simulations aided the AA effort by allowing a rapid software error diagnosis in simulations and AADS and by independently verifying AB analyses.

## **8.3 NESC Recommendations**

The following NESC recommendations were identified and directed toward the NESC (including the disciplines of Flight Mechanics, Aerosciences, Passive Thermal, GN&C, Software, Loads and Dynamics, and Human Factors) and future NASA programs and projects that may utilize AB:

- R.1.** Further development of AA should address potential improvements to AADS.  
(*F-4, F-6, O-1*)


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 92 of 286	

- a. Reduce errors in the Ephemeris Estimator by exploring new integration methods, step sizes, and real-time periapsis timing correction.
- b. Desensitize the Atmosphere Estimator to Ephemeris Estimator timing errors and investigate modeling methods independent of altitude. Incorporate 3-sigma estimates of atmospheric density in Maneuver Estimator logic.
- c. Modify the AADS software so that it is suitable for a flight implementation. This includes streamlining the code execution, adding error handling and fault detection and correction, and accommodating safe-mode events (safety triggers), off-nominal scenarios such as pop-up maneuvers and typical AB spacecraft contingencies.
- d. Assess the feasibility of incorporating collision avoidance in AADS.
- e. Add flexibility to the AADS maneuver logic and implement versatile control strategies that can meet project specific requirements.

**R.2.** Future AA analysis should include improved analysis methods and additional stress testing and trade studies. (*F-6, F-8, F-10, O-2*)

- a. Improve the POST2 simulation capability by creating a 6-DOF “truth” simulation version, and by including the effects of IMUs, reaction wheels, thrusters, and other spacecraft models.
- b. Complete the integration of Venus and Titan environment models into AAHFS and perform testing similar to the POST2 analyses for these bodies.
- c. Incorporate Mars AB flight-data-derived density profiles into “truth” simulations, identifying the impact of real density profiles with previously simulated “perturbed” atmospheres.
- d. Develop model uncertainties in AADS “truth” simulations and assess AADS performance against a mission simulation using Monte Carlo methods to identify robustness and further areas of improvement of AADS.
- e. Stress test AADS using atmospheric random noise and bias, initial ground errors, modeling errors, and fault management logic.
- f. Perform a sensitivity study on the required fidelity/order of the AADS (Ephemeris Estimator) gravity model to ensure sufficient accuracy while minimizing computational expenditures.
- g. Conduct additional trade studies, including studying the impact of the accelerometer data buffer frequency on the Ephemeris Estimator orbit knowledge; determining accelerometer sensitivities (e.g., bias, noise, scale factor, alignment, etc.) and their impact on AADS; analyzing corridor width versus maneuver



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 93 of 286	

frequency; evaluating maneuver execution performance sensitivities; and quantifying the impact of solar radiation pressure.

- h. Benchmark AADS performance on a flight processor. Ensure the proposed software architecture is valid and that the code can run in allotted time and/or central processing unit cycles.

## 9.0 Alternate Viewpoints

There were no alternate viewpoints identified during the course of this assessment by the NESC team or the NRB quorum.

## 10.0 Other Deliverables


No unique hardware, software, or data packages, outside those contained in this report, were disseminated to other parties outside this assessment.

## 11.0 Lessons Learned

No applicable lessons learned were identified for entry into the NASA Lessons Learned Information System.

## 12.0 Definition of Terms

Corrective Actions	Changes to design processes, work instructions, workmanship practices, training, inspections, tests, procedures, specifications, drawings, tools, equipment, facilities, resources, or material that result in preventing, minimizing, or limiting the potential for recurrence of a problem.
Finding	A conclusion based on facts established by the investigating authority.
Lessons Learned	Knowledge or understanding gained by experience. The experience may be positive, as in a successful test or mission, or negative, as in a mishap or failure. A lesson must be significant in that it has real or assumed impact on operations; valid in that it is factually and technically correct; and applicable in that it identifies a specific design, process, or decision that reduces or limits the potential for failures and mishaps, or reinforces a positive result.
Observation	A factor, event, or circumstance identified during the assessment that did not contribute to the problem, but if left uncorrected has the potential to cause a mishap, injury, or increase the severity should a mishap occur. Alternatively, an observation could be a positive acknowledgement of a


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		<b>Autonomous Aerobraking (Phase 1)</b>	
		Page #: 94 of 286	

Center/Program/Project/Organization’s operational structure, tools, and/or support provided.


Problem	The subject of the independent technical assessment.
Proximate Cause	The event(s) that occurred, including any condition(s) that existed immediately before the undesired outcome, directly resulted in its occurrence and, if eliminated or modified, would have prevented the undesired outcome.
Recommendation	An action identified by the NESC to correct a root cause or deficiency identified during the investigation. The recommendations may be used by the responsible Center/Program/Project/Organization in the preparation of a corrective action plan.
Root Cause	One of multiple factors (events, conditions, or organizational factors) that contributed to or created the proximate cause and subsequent undesired outcome and, if eliminated or modified, would have prevented the undesired outcome. Typically, multiple root causes contribute to an undesired outcome.

### 13.0 Acronyms List

°C	Degrees Celsius
3-D	Three-Dimensional
AA	Autonomous Aerobraking
AADS	Autonomous Aerobraking Development Software
AAG	Atmospheric Advisory Group
AAHFS	Autonomous Aerobraking High Fidelity Simulation
AA PCMD	Autonomous Aerobraking Planetary and Constants Document
AB	Aerobraking
AMA	Analytical Mechanics Associates, Inc.
APL	Applied Physics Laboratory (Johns Hopkins University, Laurel, MD)
ARC	Ames Research Center
ATK	Alliant Techsystems Inc.
CCD	Central Composite Design
CFD	Computational Fluid Dynamics
CSH	Constant Scale Height
CSHIO	Constant Scale Height Inbound and Outbound
CSHT	Constant Scale Height with Time
DAC	DSMC Analysis Code
DFRC	Dryden Flight Research Center
DOE	Design of Experiment
DOF	Degree of Freedom

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 95 of 286	


DSMC	Direct Simulation Monte Carlo
DSN	Deep Space Network
EDL	Entry, Descent, and Landing
GN&C	Guidance, Navigation, and Control
GRAM	Global Reference Atmospheric Model
GRETA	Generic Response-Surface Equation Thermal Analysis
GSFC	Goddard Space Flight Center
IMU	Inertial Measurement Unit
JHU	Johns Hopkins University
JPL	Jet Propulsion Laboratory
JSC	Johnson Space Center
K	Kelvin
kg/km <sup>3</sup>	kilograms per kilometer cubed
KinetX	KinetX, Inc.
km	kilometer
LaRC	Langley Research Center
LM	Lockheed Martin
LMST	Local Mean Solar Time
LST	Local Solar Time
LTST	Local True Solar Time
m	meter
Mars-GRAM	Mars Global Reference Atmospheric Model
MESSENGER	Mercury Surface, Space ENvironment, GEOchemistry and Ranging (Discovery mission)
MGCM	Mars Global Circulation Model
MGS	Mars Global Surveyor
MITS	MSFC Information Technology Services
MOI	Mars Orbit Insertion
MOLA	Mars Orbiting Laser Altimeter
MRO	Mars Reconnaissance Orbiter
m/s	meters/second
MSFC	Marshall Space Flight Center
MTSO	Management and Technical Support Office
MTGCM	Mars Thermospheric Global Circulation Model
NCSU	North Carolina State University
NESC	NASA Engineering and Safety Center
NIA	National Institute of Aerospace
OD	Orbit Determination
PDS	Planetary Data System
POST2	Program To Optimize simulated Trajectories II
PTE	Periapsis Timing Estimator

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		96 of 286	

PVO	Pioneer Venus Orbiter
rms	root mean square
RTW	Real Time Workshop
SA	Solar Array
SME	Subject Matter Expert
SRP	Solar Radiation Pressure
TDT	Technical Discipline Team
TES	Thermal Emission Spectrometer
VIRA	Venus International Reference Atmosphere
w/cm <sup>2</sup>	watts per centimeter squared

## 14.0 References


1. Spencer, D.A., Tolson, R.H., "Aerobraking Cost and Risk," *Journal of Spacecraft and Rockets*. Vol. 44, No. 6, November–December 2007, pp 1285–1293.
2. Hanna, J.L. and Tolson, R.H., "Approaches to Autonomous Aerobraking at Mars," AAS/AIAA, *Astrodynamics Specialist Conference*, Quebec City, Canada, July 30-August 2, 2001. AAS 01-387.
3. Lyons, Daniel T., "Aerobraking Automation Options," AAS-01-385, AAS/AIAA *Astrodynamics Specialist Conference*, Quebec City, California, 2001.
4. Hanna, J.L., Tolson, R.H., Cianciolo, A.M.D., and Dec, J.A., "Autonomous Aerobraking at Mars," presented at *5th International ESA Conference on Guidance Navigation and Control Systems and Actuator and Sensor Product Exhibition*, Frascati, Italy, October 22–25, 2002. (ESA SP-516, February 2003.)
5. Willcockson, W.H. and Johnson, M.A., "Mars Odyssey Aerobraking: The First Step Toward Autonomous Aerobraking Operations," *2003 IEEE Aerospace Conference*, Big Sky, Montana. March 9-14, 2003.
6. Tolson, Robert H. and Prince, Jill L., "Onboard Atmospheric Modeling and Prediction for Autonomous Aerobraking Missions," AAS 11-477, AAS/AIAA *Astrodynamics Specialist Conference*, Girdwood, Alaska, 2011.
7. D.J. O'Shaughnessy *et al.*, "Autonomous Aerobraking Algorithm Testing in a Flight Software Simulation Environment," AAS 11-471, AAS/AIAA *Astrodynamics Specialist Conference*, Girdwood, Alaska, 2011.
8. Prince, J.L. and Striepe, S.A., "NASA Langley Simulation Capabilities for the Mars Reconnaissance Orbiter," AAS/AIAA *Space Flight Mechanics Conference*, Copper Mountain, Colorado, January 23-27, 2005.
9. Bird, G.A., *Molecular Dynamics and the Direct Simulation of Gas Flows*, Oxford University Press, Oxford, United Kingdom, 1994.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 97 of 286	

10. Wilmoth, R.G., LeBeau, G.J., and Carlson, A.B., “DSMC Grid Methodologies for Computing Low-Density, Hypersonic Flows About Reusable Launch Vehicles,” AIAA Paper 1996-1812, 1996.
11. LeBeau, G.J., “A Parallel Implementation of the Direct Simulation Monte Carlo Method,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 174, No. 3–4, 1999, pp. 319–337.
12. LeBeau, G.J., Boyles, K.A., and Lumpkin III, F.E., “Virtual Sub-Cells for the Direct Simulation Monte Carlo Method,” AIAA Paper 2003-1031, 2003.
13. Borgnakke, C. and Larsen, P.S., “Statistical Collision Model for Monte Carlo Simulation of Polyatomic Gas Mixture,” *Journal of Computational Physics*, Vol. 18, No. 4, 1975, pp. 405-420.
14. D.J. Carrelli *et al.*, “Autonomous Aerobraking for Low-Cost Interplanetary Missions,” 9<sup>th</sup> Annual Low-Cost Planetary Missions Conference, Laurel, Maryland, 2011.

## 15.0 Appendices

- Appendix A. Autonomous Aerobraking Planetary Constants and Models Version 0.07 (Supplement to Section 7.2)
- Appendix B. Mission Design Appendix (Supplement to Section 7.2)
- Appendix C. Aerodynamics and Aerothermodynamics Computational Methods (Supplement to Section 7.3.1.2)
- Appendix D. Thermal Modeling (Supplement to Section 7.3.2.4)
- Appendix E. Mars-GRAM 2010 (Supplement to Section 7.3.1.3)
- Appendix F. Onboard Atmospheric Modeling and Prediction for Autonomous Aerobraking Missions (Supplement to Section 7.3.2.1)
- Appendix G. Ephemeris Estimator User’s Guide (Supplement to Section 7.3.2.2)
- Appendix H. AADS (Supplement to Section 7.5.1)
- Appendix I. AAHFS (Supplement to Section 7.5.2)
- Appendix J. AA Interface Control Document

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 98 of 286	


## **Appendix A. Autonomous Aerobraking Planetary Constants and Models Version 0.07 (Supplement to Section 7.2)**

**Autonomous Aerobraking Planetary Constants and Models Version 0.07  
October 11, 2011**

### **Introduction**


The purpose of this document is to provide a common set of planetary constants and models in a single reference for use by the Autonomous Aerobraking Assessment Team. The models and astrophysical quantities necessary to execute the plan and objectives of the Autonomous Aerobraking Assessment Team at Venus, Mars, and Titan are defined and described briefly. The user may consult the provided references for more detail if desired. Current International Astronomical Union (IAU) standards and definitions are adopted wherever possible.

The format of this document is based on several previous Planetary Constants and Models documents: Mars Pathfinder Project [1], Mars Reconnaissance Orbiter [2], Phoenix Project [3], Lunar Constants and Models [4], and Mars Science Laboratory [5]. Portions of this document are taken verbatim from these reference documents and other references listed where applicable; however, there are not any mission-specific data or conventions contained in this document. The data in this document is intended to be applicable to a typical aerobraking mission at Venus, Mars, or Titan. Modifications to the data contained in this document may be necessary for specific missions with unique requirements.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 99 of 286	

## DISTRIBUTION


Mike Aguilar  
Ruth Amundsen  
Jeremy Bauman  
Angela Bowes  
David Carelli  
Alicia Dwyer-Cianciolo  
John Dec  
Neil Dennehy  
Prasun Desai  
Donna Gilchrist  
Starr Ginn  
Sanjay Gowda  
Mark Holdridge  
Hilary Justh  
Jere Justus  
Jim Kaidy  
Gerald Lebeau  
Laura Leybold  
Derek Liechty  
Forrest Lumpkin  
Jeremiah Marichalar  
Melinda Meredith  
Stephen Miller  
Daniel Murri  
Cynthia Null  
Daniel O'Shaughnessy  
Chris Pastore  
Richard Powell  
Jill Prince  
Holly Ramey  
Chris Regan  
Vicki Regenie  
Steve Rickman  
Dave Schuster  
David Skinner  
Carolyn Snare  
Tom Strikwerda  
Mark Thornblom  
Robert Tolson  
Bobby Williams  
Ken Williams  
Peter Wolff

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 100 of 286	


## TABLE OF CONTENTS

<b>Venus Data</b>	<b>5</b>
Venus Gravity Field	
Venus Pole and Prime Meridian	
Venus Shape Parameters	
Venus Atmosphere Model	
Venus Aero Database	
<b>Earth Data</b>	<b>7</b>
Earth Pole and Prime Meridian	
Earth Shape Parameters	
<b>Mars Data</b>	<b>8</b>
Mars Gravity Field	
Mars Pole and Prime Meridian	
Mars Shape Parameters	
Mars Satellites	
Mars Atmosphere Model	
Mars Aero Database	
<b>Saturn Data</b>	<b>14</b>
Saturn Gravity Field	
Saturn Pole and Prime Meridian	
Saturn Shape Parameters	
<b>Titan Data</b>	<b>16</b>
Titan Gravity Field	
Titan Pole and Prime Meridian	
Titan Shape Parameters	
Titan Atmosphere Model	
Titan Aero Database	
<b>Fundamental Constants and Planetary/Satellite Ephemeris Files</b>	<b>19</b>
<b>Coordinate Systems</b>	<b>20</b>
<b>Spacecraft Models</b>	<b>26</b>
<b>Propulsion Models</b>	<b>26</b>
<b>Aerodynamic Force And Moment Calculations</b>	<b>26</b>
<b>Thermal Models</b>	<b>29</b>



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 101 of 286	

<b>Appendix</b>	<b>30</b>
Normalized/Un-normalized Conversions	
<b>References</b>	<b>31</b>

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 102 of 286	

## VENUS DATA

### Venus Gravity Field

The latest Venus gravity field is MGNP180U, which is a 180 degree and order model based on the IAU 1991 Venus pole and prime meridian locations and a reference radius of 6051.0 km. This gravity field was developed using data collected from the Magellan mission. The file can be downloaded from the following site: [http://pds-geosciences.wustl.edu/geo/mgn-v-rss-5-gravity-l2-v1/mg\\_5201/gravity/shgj180u.a01](http://pds-geosciences.wustl.edu/geo/mgn-v-rss-5-gravity-l2-v1/mg_5201/gravity/shgj180u.a01).

$$GM_{\text{Venus}} = 324858.592079 \text{ km}^3/\text{s}^2 \quad (\text{Gravitational Parameter from MGNP180U})$$

$$R_{\text{ref}} = 6051.0 \text{ km} \quad (\text{Reference Radius of MGNP180U})$$

$$J_2_{\text{Venus}} = 1.96972335776 \times 10^{-6} \quad (\text{Venus } J_2 \text{ (Normalized) from MGNP180U})$$

### Venus Pole and Prime Meridian

The Venus pole and prime meridian values listed below are from the most recent (2006) IAU/IAG report [6]. For Venus, these values are the same as those contained in the IAU/IAG/COSPAR 1991 report [7].

$$\alpha_{\text{Venus}} = 272.76 \text{ deg} \quad (\text{Right Ascension of Venus Pole in EME2000})$$

$$\delta_{\text{Venus}} = 67.16 \text{ deg} \quad (\text{Declination of Venus Pole in EME2000})$$

$$J2000 = 2451545.0 \quad (\text{Reference Epoch of J2000 in Julian Days})$$

$$= \text{Jan. 1, 2000 12:00:00 ET} \quad (\text{Calendar date of J2000 Reference Epoch})$$


$$D = (\text{Julian Date} - J2000) \quad (\text{Days Past Epoch of J2000})$$

$$ET = TDT \quad (\text{Ephemeris Time} = \text{Terrestrial Dynamical Time})$$

$$TDT \cong TDB \quad (\text{Terrestrial Dynamical Time is essentially equivalent to Barycentric Dynamic Time. TDB varies from TDT only by periodic variations.})$$

$$W_{\text{Venus}} = 160.20 + \dot{W}_{\text{Venus}} * D \text{ deg} \quad (\text{Prime Meridian with respect to Venus IAU vector at epoch} = D)$$

$$\dot{W}_{\text{Venus}} = -1.4813688 \text{ deg/day} \quad (\text{Rotational rate of Venus})$$

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 103 of 286	

### Venus Shape Parameters

The Venus radii are taken from the IAU/IAG 2006 report.

$R_{\text{Venus-Equator}} = 6051.8 \text{ km}$  (Radius of Venus Equator)

$R_{\text{Venus-Pole}} = 6051.8 \text{ km}$  (Average radius of Venus Poles)

### Venus Atmosphere Model

The Venus atmospheric density model is the Venus Global Reference Atmospheric Model (VenusGRAM) 2005 Version [8]. VenusGRAM is based on data from the Venus International Reference Atmosphere (VIRA). Nominal input parameters should be set as follows:

NVARX = 1 (X-code for plotable output)

NVARY = 0 (Y-code for 2-D plotable output)

profnear = 0 (No auxiliary profile input)

proffar = 0 (Lat-lon radius in degrees beyond which weight for auxiliary profile is 0.0).

rpscale = 1 (Random density perturbation scale factor)

NMONTE = 1 (Number of Monte Carlo runs)

corlmin = 0 (Minimum relative step size for perturbation updates)


Other input parameters for VenusGRAM not listed here are specific to the system executing VenusGRAM and should be set to be consistent with the requirements of each system. Assume no winds.

The reference ellipsoid value utilized in VenusGRAM shall be set in the following manner:

$$R_{\text{ref}} = R_{\text{Venus-Equator}} / \sqrt{1 + [(R_{\text{Venus-Equator}} / R_{\text{Venus-Pole}})^2 - 1] * \sin(\text{lat})^2}$$

### Venus Aero Database Model

The Venus aero database can be downloaded from NSCKN using the following link:  
<https://nsekn.nasa.gov/DMS/ViewDoc.aspx?DocID=208822>

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 104 of 286	

## EARTH DATA

### Earth Pole and Prime Meridian


The Earth pole and prime meridian values are taken from the IAU/IAG 2006 report.

$\alpha_{\text{Earth}}$	= 0.0 deg	(Right Ascension of Earth Pole in EME2000)
$\delta_{\text{Earth}}$	= 90.0 deg	(Declination of Earth Pole in EME2000)
D	= (Julian Date – J2000)	(Days Past Epoch of J2000)
J2000	= 2451545.0	(Reference Epoch of J2000 in Julian Days)
	= Jan. 1, 2000 12:00:00 ET	(Calendar date of J2000 Reference Epoch)
ET	= TDT	(Ephemeris Time = Terrestrial Dynamical Time)
TDT	$\cong$ TDB	(Terrestrial Dynamical Time is essentially equivalent to Barycentric Dynamic Time. TDB varies from TDT only by periodic variations.)
$W_{\text{Earth}}$	= $190.147 + \dot{W} * D$ deg	(Earth Prime Meridian at epoch = D)
$\dot{W}_{\text{Earth}}$	= 360.9856235 deg/day	(Earth rotational rate)

### Earth Shape Parameters

The Earth radii are taken from the IAU/IAG 2006 report:

$R_{\text{Earth-Equator}}$	= 6378.14 km	(Radius of Earth Equator from IAU)
$R_{\text{Earth-Pole}}$	= 6356.75 km	(Radius of Earth Pole from IAU)

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		105 of 286	

## MARS DATA

### Mars Gravity Field

The most recent Mars gravity field model is MRO110B. Because the MRO110B gravity field model requires a new orientation model (which requires extensive software updates), the MGS85F2 gravity field is being used. The MGS85F2 gravity field is an 85 degree and order model based on the IAU 2000 [9] Mars pole and prime meridian locations and a reference radius of 3396.2 km. It is the last Mars gravity field model to use the IAU orientation. This gravity field was developed using data collected from Mariner 9, Viking 1 and 2, and MGS mapping through Nov. 18, 2001. The file can be downloaded from the following site: [http://pds-geosciences.wustl.edu/geo/mgs-m-rss-5-sdp-v1/mors\\_1024/sha/jgm85f02.sha](http://pds-geosciences.wustl.edu/geo/mgs-m-rss-5-sdp-v1/mors_1024/sha/jgm85f02.sha).

$$GM_{\text{Mars}} = 42828.376212 \text{ km}^3/\text{s}^2 \quad (\text{Gravitational Parameter from MGS85F2})$$

$$R_{\text{ref}} = 3396.2 \text{ km} \quad (\text{Reference Radius of MGS85F2})$$

$$J_{2 \text{ Mars}} = 0.874924464436 \times 10^{-3} \quad (\text{Mars } J_2 \text{ (Normalized) from MGS85F2})$$

### Mars Pole and Prime Meridian

The Mars pole and prime meridian values are from the IAU/IAG 2006 report. For Mars and its satellite moons, these values are the same as those contained in the IAU/IAG 2000 report.

$$\alpha_{\text{Mars}} = 317.68143 \text{ deg} \quad (\text{Right Ascension of Mars Pole in EME2000})$$

$$\delta_{\text{Mars}} = 52.8865 \text{ deg} \quad (\text{Declination of Mars Pole in EME2000})$$

$$D = (\text{Julian Date} - J2000) \quad (\text{Days Past Epoch of J2000})$$

$$J2000 = 2451545.0 \quad (\text{Reference Epoch of J2000 in Julian Days})$$


$$= \text{Jan. 1, 2000 12:00:00 ET} \quad (\text{Calendar date of J2000 Reference Epoch})$$

$$ET = TDT \quad (\text{Ephemeris Time} = \text{Terrestrial Dynamical Time})$$

$$TDT \cong TDB \quad (\text{Terrestrial Dynamical Time is essentially equivalent to Barycentric Dynamic Time. TDB varies from TDT only by periodic variations.})$$

$$W_{\text{Mars}} = 176.630 + \dot{W}_{\text{Mars}} * D \text{ deg} \quad (\text{Prime Meridian with respect to Mars IAU vector at epoch} = D)$$

$$\dot{W}_{\text{Mars}} = 350.89198226 \text{ deg/day} \quad (\text{Rotational rate of Mars})$$

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 106 of 286	

### Mars Shape Parameters

The Mars radii are taken from the IAU/IAG 2006 report.

$R_{\text{Mars-Equator}}$	= 3396.19 km	(Radius of Mars Equator)
$R_{\text{Mars-Pole}}$	= 3376.20 km	("Average" radius of Mars Poles for purposes of modeling its shape as an oblate spheroid)
$f$	= 0.00588600756	(Flattening factor)


### Mars Satellites

#### Phobos

$GM_{\text{Phobos}}$	= $7.118 \times 10^{-4} \text{ km}^3/\text{s}^2$	(Gravitational Parameter from MGS85F2)
$R_{\text{Phobos-mean}}$	= 11.1 km	(Mean radius of Phobos from IAU/IAG)
$R_{\text{Phobos-equatx}}$	= 13.4 km	(Subplanet equatorial radius of Phobos (IAU/IAG))
$R_{\text{Phobos-equaty}}$	= 11.2 km	(Along orbit equatorial radius of Phobos (IAU/IAG))
$R_{\text{Phobos-polar}}$	= 9.2 km	(Polar radius of Phobos from IAU/IAG)

#### Deimos

$GM_{\text{Deimos}}$	= $0.786 \times 10^{-4} \text{ km}^3/\text{s}^2$	(Gravitational Parameter from MGS85F2)
$R_{\text{Deimos-mean}}$	= 6.2 km	(Mean radius of Deimos from IAU/IAG)
$R_{\text{Deimos-equatx}}$	= 7.5 km	(Subplanet equatorial radius of Deimos (IAU/IAG))
$R_{\text{Deimos-equaty}}$	= 6.1 km	(Along orbit equatorial radius of Deimos (IAU/IAG))
$R_{\text{Deimos-polar}}$	= 5.2 km	(Polar radius of Deimos from IAU/IAG)


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 107 of 286	

### Mars Atmosphere Model

The Mars atmospheric density model is the Mars Global Reference Atmospheric Model 2010 Beta Version [10]. MarsGRAM outputs high, mean, and low density values at each point. The baseline density is the *mean* value.


Nominal input parameters should be set as follows:

Dusttau	= 0.5	(Atmospheric opacity/dust level)
Dustmin	= 0.3	(Minimum seasonal Dusttau)
Dustmax	= 1	(Maximum seasonal Dusttau)
Dustnu	= 0.003	(Parameter for vertical distribution of dust density)
Dustdiam	= 5	(Dust particle diameter, micrometers)
Dustdens	= 3000	(Dust particle density, kg/m <sup>3</sup> )
ALSO	= 0	(Starting Ls value in degrees for dust storm)
ALSDUR	= 48	(Duration in Ls degrees for dust storm)
INTENS	= 0	(Dust storm intensity)
RADMAX	= 0	(Maximum radius of dust storm, km)
DUSTLAT	= 0	(Latitude for center of dust storm, degrees)
DUSTLON	= 0	(Longitude for center of dust storm, degrees)
MapYear	= 0	(Flag to indicate which GCM input data sets are to be used)
F107	= 160	(10.7 cm solar flux, W/cm <sup>2</sup> )
NVARX	= 1	(X-code for plotable output)
NVARY	= 0	(Y-code for 2-D plotable output)
MOLAhgts	= 0	(Flag to indicate use of MOLA areoid or reference ellipsoid)
hgtasfcm	= 0	(Height above surface, km)
zoffset	= 0	(Constant height offset for MTGCM data, km)
ibougher	= 0	(Bougher height offset term indicating Ls-dependency, km)

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 108 of 286	

deltaTEX	= 0 K	(Exospheric temperature adjustment)
profnear	= 0	(No auxiliary profile input)
proffar	= 0	(Lat-lon radius in degrees beyond which weight for auxiliary profile is 0.0).
rpscale	= 1	(Random density perturbation scale factor)
rwscale	= 1	(Random wind perturbation scale factor)
wlscale	= 1	(Scale factor for perturbation wavelengths)
wmscale	= 1	(Scale factor for mean winds)
blwinfac	= 1	(Scale factor for boundary layer slope winds)
NMONTE	= 1	(Number of Monte Carlo runs)
WaveA0	= 1	(Mean term of longitude-dependent wave multiplier for density)
WaveDate	= 0	(Julian date for primary peak(s) of wave multiplier for density)
WaveA1	= 0	(Amplitude of wave-1 component of longitude-dependent wave multiplier for density)
Wavephi1	= 0	(Phase of wave-1 component of longitude-dependent wave multiplier for density)
phi1dot	= 0	(Rate of longitude movement in degrees per day for wave-1 component)
WaveA2	= 0	(Amplitude of wave-2 component of longitude-dependent wave multiplier for density)
Wavephi2	= 0	(Phase of wave-2 component of longitude-dependent wave multiplier for density)
phi2dot	= 0	(Rate of longitude movement in degrees per day for wave-2 component)
WaveA3	= 0	(Amplitude of wave-3 component of longitude-dependent wave multiplier for density)
Wavephi3	= 0	(Phase of wave-3 component of longitude-dependent wave multiplier for density)
phi3dot	= 0	(Rate of longitude movement in degrees per day for wave-3 component)



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 109 of 286	

iuwave = 0 (No time-dependent wave coefficient data file)  
 Wscale = 20 (Vertical scale of longitude-dependent wave damping, km)  
 corlmin = 0 (Minimum relative step size for perturbation updates)  
 ipelat = 1 (Planetocentric latitude and height input flag)  
 requa = 3396.19 km (Equatorial radius for reference ellipsoid, km)  
 rpole = 3376.2 km (Polar radius for reference ellipsoid, km)  
 idaydata = 0 (No daily max data output)

Other input parameters required by MarsGRAM but not listed here are specific to the system executing MarsGRAM and should be set to be consistent with the requirements of each system. Assume no winds and no dust storms. (The MarsGRAM variables listed above pertaining to dust storms are set to assume no dust storms).

The reference ellipsoid value utilized in MarsGRAM shall be set in the following manner:

$$R_{ref} = R_{Mars-Equator} / \sqrt{1 + [(R_{Mars-Equator} / R_{Mars-Pole})^2 - 1] * \sin(\text{decln})^2}$$

Figure 1 is a graphical depiction of the reference ellipsoid ( $R_{ref}$ ), the decln used in its calculation, and the associated spacecraft altitude (referred to as altito in the figure) that MarsGRAM requires to determine density.

#### **Mars Aero Database Model**

The Mars aero database can be downloaded from NSCKN using the following link:  
<https://nsckn.nasa.gov/DMS/ViewDoc.aspx?DocID=208822>



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

110 of 286

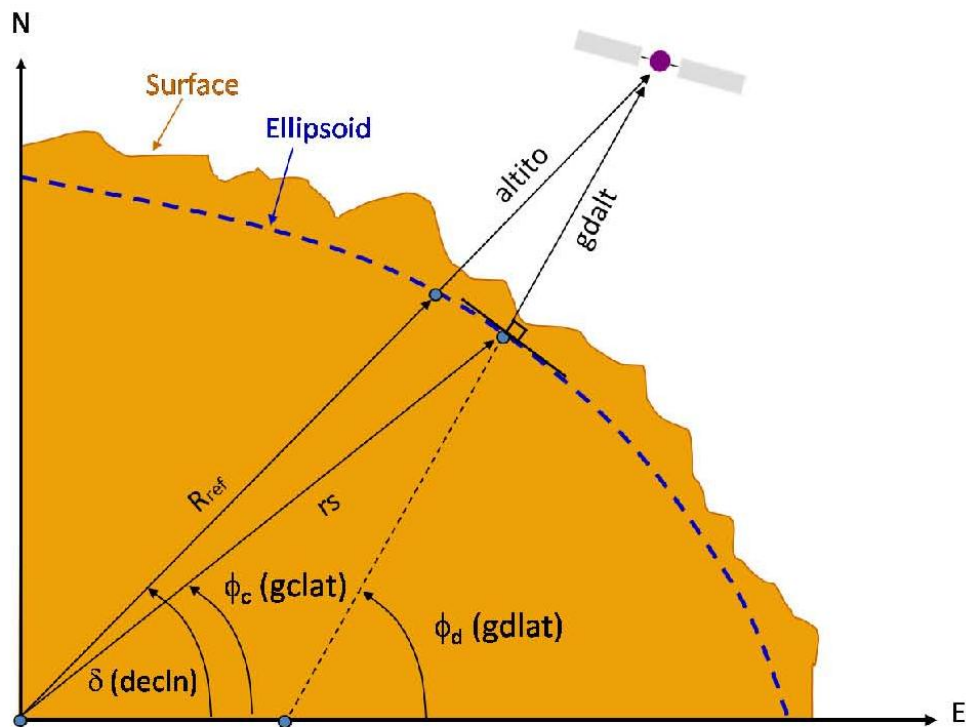



Figure 1: Definitions for Altitude, Latitude, and Declination

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 111 of 286	

## SATURN DATA

### Saturn Gravity Field


The latest  $J_2$ ,  $J_4$ , and  $J_6$  Saturn gravity field coefficients have been determined from data from the Cassini mission as of July 6, 2010. [11]

$GM_{\text{Saturn}}$	= 37931207.6129 km <sup>3</sup> /s <sup>2</sup>	(Gravitational Parameter)
$R_{\text{ref}}$	= 60330.0 km	(Reference Radius)
$J_2_{\text{Saturn}}$	= 0.016290713275	(Saturn $J_2$ , Un-normalized)
$J_4_{\text{Saturn}}$	= -0.000934907426	(Saturn $J_4$ , Un-normalized)
$J_6_{\text{Saturn}}$	= 0.000086813940	(Saturn $J_6$ , Un-normalized)

### Saturn Pole and Prime Meridian

The Saturn pole and prime meridian values listed below are from the IAU/IAG 2006 report.

$\alpha_{\text{Saturn}}$	= 40.589 deg	(Right Ascension of Saturn Pole in EME2000)
$\delta_{\text{Saturn}}$	= 83.537 deg	(Declination of Saturn Pole in EME2000)
D	= (Julian Date – J2000)	(Days Past Epoch of J2000)
J2000	= 2451545.0	(Reference Epoch of J2000 in Julian Days)
	= Jan. 1, 2000 12:00:00 ET	(Calendar date of J2000 Reference Epoch)
ET	= TDT	(Ephemeris Time = Terrestrial Dynamical Time)
TDT	$\cong$ TDB	(Terrestrial Dynamical Time is essentially equivalent to Barycentric Dynamic Time. TDB varies from TDT only by periodic variations.)
$W_{\text{Saturn}}$	= 38.90 + $\dot{W}_{\text{Saturn}}$ *D deg	(Prime Meridian with respect to Saturn IAU vector at epoch = D)
$\dot{W}_{\text{Saturn}}$	= 810.7939024 deg/day	(Rotational rate of Saturn)


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 112 of 286	

**Saturn Shape Parameters**

The Saturn radii are taken from the IAU/IAG 2006 report:

$R_{\text{Saturn-Equator}}$  = 60268.0 km (Radius of Saturn Equator)

$R_{\text{Saturn-Pole}}$  = 54364.0 km (Average radius of Saturn Poles)


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		113 of 286	

## TITAN DATA

### Titan Gravity Field

The latest Titan gravity field is a 3 degree and order model referred to as the SOL2 approach. It is based on radiometric tracking and optical navigation imaging data from the Cassini mission combined with data from the Pioneer and Voyager missions, as well astronomical observations of Saturn and its satellites [12]. The gravity field parameters are un-normalized in Ref. 12 but are shown as both normalized and un-normalized format here.

$GM_{\text{Titan}}$	$= 8978.1394 \text{ km}^3/\text{s}^2$	(Gravitational Parameter)
$R_{\text{ref}}$	$= 2575.0 \text{ km}$	(Reference Radius)
$J_2 \text{ Titan}$	$= 33.462 \times 10^{-6}$ $= 1.496466133262 \times 10^{-5}$	(Titan $J_2$ , Un-normalized) (Titan $J_2$ , Normalized)
$C_{21 \text{ Titan}}$	$= 0.048 \times 10^{-6}$ $= 3.718064012359 \times 10^{-8}$	(Titan $C_{21}$ , Un-normalized) (Titan $C_{21}$ , Normalized)
$S_{21 \text{ Titan}}$	$= -0.620 \times 10^{-6}$ $= -4.802499349297 \times 10^{-7}$	(Titan $S_{21}$ , Un-normalized) (Titan $S_{21}$ , Normalized)
$C_{22 \text{ Titan}}$	$= 10.022 \times 10^{-6}$ $= 1.552601563828 \times 10^{-5}$	(Titan $C_{22}$ , Un-normalized) (Titan $C_{22}$ , Normalized)
$S_{22 \text{ Titan}}$	$= 0.256 \times 10^{-6}$ $= 3.965934946516 \times 10^{-7}$	(Titan $S_{22}$ , Un-normalized) (Titan $S_{22}$ , Normalized)
$J_3 \text{ Titan}$	$= -0.074 \times 10^{-6}$ $= -2.796937100268 \times 10^{-8}$	(Titan $J_3$ , Un-normalized) (Titan $J_3$ , Normalized)
$C_{31 \text{ Titan}}$	$= 1.805 \times 10^{-6}$ $= 1.671105280089 \times 10^{-6}$	(Titan $C_{31}$ , Un-normalized) (Titan $C_{31}$ , Normalized)
$S_{31 \text{ Titan}}$	$= 0.283 \times 10^{-6}$ $= 2.620070882356 \times 10^{-7}$	(Titan $S_{31}$ , Un-normalized) (Titan $S_{31}$ , Normalized)
$C_{32 \text{ Titan}}$	$= 0.136 \times 10^{-6}$ $= 3.981672297630 \times 10^{-7}$	(Titan $C_{32}$ , Un-normalized) (Titan $C_{32}$ , Normalized)
$S_{32 \text{ Titan}}$	$= 0.159 \times 10^{-6}$ $= 4.655043347965 \times 10^{-7}$	(Titan $S_{32}$ , Un-normalized) (Titan $S_{32}$ , Normalized)
$C_{33 \text{ Titan}}$	$= -0.185 \times 10^{-6}$ $= -1.326703756361 \times 10^{-6}$	(Titan $C_{33}$ , Un-normalized) (Titan $C_{33}$ , Normalized)

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 114 of 286	

$$S_{33 \text{ Titan}} = -0.149 \times 10^{-6} \quad (\text{Titan } S_{33}, \text{ Un-normalized})$$

$$= -1.068534376745 \times 10^{-6} \quad (\text{Titan } S_{33}, \text{ Normalized})$$

#### Titan Pole and Prime Meridian

The Titan pole and prime meridian values listed below are from the IAU/IAG 2006 report.

$$\alpha_{\text{Titan}} = 36.41 \text{ deg} \quad (\text{Right Ascension of Titan Pole in EME2000})$$

$$\delta_{\text{Titan}} = 83.94 \text{ deg} \quad (\text{Declination of Titan Pole in EME2000})$$

$$T = \frac{D}{36525} \quad (\text{Centuries Past Epoch of J2000})$$

$$D = (\text{Julian Date} - \text{J2000}) \quad (\text{Days Past Epoch of J2000})$$

$$\text{J2000} = 2451545.0 \quad (\text{Reference Epoch of J2000 in Julian Days})$$

$$= \text{Jan. 1, 2000 12:00:00 ET} \quad (\text{Calendar date of J2000 Reference Epoch})$$

$$\text{ET} = \text{TDT} \quad (\text{Ephemeris Time} = \text{Terrestrial Dynamical Time})$$

$$\text{TDT} \cong \text{TDB} \quad (\text{Terrestrial Dynamical Time is essentially equivalent to Barycentric Dynamic Time. TDB varies from TDT only by periodic variations.})$$

$$W_{\text{Titan}} = 189.64 + \dot{W}_{\text{Titan}} * D - 2.64 * \sin(29.80 - 52.1 * T) \text{ deg}$$

(Prime Meridian with respect to Titan IAU vector at epoch = D)

$$\dot{W}_{\text{Titan}} = 22.5769768 \text{ deg/day} \quad (\text{Rotational rate of Titan})$$

#### Titan Shape Parameters

The Titan radii are taken from the IAU/IAG 2006 report and are consistent with those calculated for the Titan gravity field with the SOL2 approach.


$$R_{\text{Titan-Equator}} = 2575.0 \text{ km} \quad (\text{Radius of Titan Equator})$$

$$R_{\text{Titan-Pole}} = 2575.0 \text{ km} \quad (\text{Average radius of Titan Poles})$$

#### Titan Atmosphere Model

The Titan atmospheric density model is the Titan Global Reference Atmospheric Model (Titan-GRAM) 2004 Version 1.0 [13]. Nominal input parameters should be set as follows:

$$F_{\text{minmax}} = 0 \quad (\text{Factor to vary between min and max allowed mean profiles})$$

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 115 of 286	

IFMM	= 1	(Automatically adjust input Fminmax for seasonal, latitude, and time-of-day effects)
NVARX	= 1	(X-code for plotable output)
NVARY	= 0	(Y-code for 2-D plotable output)
profnear	= 0	(No auxiliary profile input)
proffar	= 0	(Lat-lon radius in degrees beyond which weight for auxiliary profile is 0.0).
rpscale	= 1	(Random density perturbation scale factor)
NMONTE	= 1	(Number of Monte Carlo runs)
corlmin	= 0	(Minimum relative step size for perturbation updates)
famolmeth	= 0	(Use original Yelle methane mole fraction values)


Other input parameters for Titan-GRAM not listed here are specific to the system executing Titan-GRAM and should be set to be consistent with the requirements of each system. Assume no winds.

The reference ellipsoid value utilized in Titan-GRAM shall be set in the following manner:

$$R_{ref} = R_{Titan-Equator} / \sqrt{1 + [(R_{Titan-Equator} / R_{Titan-Pole})^2 - 1] * \sin^2(\text{lat})}$$

#### **Titan Aero Database Model**

The Titan aero database can be downloaded from NSCKN using the following link:  
<https://nsckn.nasa.gov/DMS/ViewDoc.aspx?DocID=208822>

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 116 of 286	

**FUNDAMENTAL CONSTANTS AND PLANETARY/SATELLITE EPHEMERIS**


The planetary ephemeris is DE421 [14]. It is commonly referred to as a SPK file (Spacecraft and Planetary Ephemeris Kernel). The binary version of the SPK file ends with the extension ".bsp" which stands for "binary SPK" [15]. It can be downloaded from the following site: [ftp://naif.jpl.nasa.gov/pub/naif/generic\\_kernels/spk/planets/](ftp://naif.jpl.nasa.gov/pub/naif/generic_kernels/spk/planets/). The gravitational parameters that are used by DE421 are included below [14]; however, these parameters do not necessarily agree with the gravitational parameters listed previously for Venus, Earth, Mars, and Saturn. In cases of inconsistency, the values listed previously in this document supersede values listed here:

Body/System GM (km <sup>3</sup> /s <sup>2</sup> )	GM (km <sup>3</sup> /s <sup>2</sup> )
Mercury	22032.090000
Venus	324858.592000
Earth	398600.436233
Mars	42828.375214
Jupiter	126712764.800000
Saturn	37940585.200000
Uranus	5794548.600000
Neptune	6836535.000000
Pluto	977.000000
Sun	132712440040.944000
Moon	4902.800076

The Martian satellite ephemeris is MAR085 [16]. The SPK file can be downloaded from the following site: [ftp://naif.jpl.nasa.gov/pub/naif/generic\\_kernels/spk/satellites/](ftp://naif.jpl.nasa.gov/pub/naif/generic_kernels/spk/satellites/). In cases of inconsistency between the MAR085 gravitational parameters and the values listed previously in this document, the values listed previously supersede the MAR085 satellite ephemeris.

The Saturn satellite ephemeris, which also includes Titan's GM, is SAT303 [17]. The SPK file can be downloaded from the following site: <ftp://ssd.jpl.nasa.gov/pub/eph/satellites/bsp/>. In cases of inconsistency between the SAT303 gravitational parameters and the values listed previously in this document, the values listed previously supersede the SAT303 satellite ephemeris.



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 117 of 286	

### COORDINATE SYSTEMS

A coordinate system is simply a means of relating points in three dimensional space and their motion as ordered triples (ex. X,Y,Z). The system is fully specified by three fundamental characteristics: frame, center and type. [18]

The following are the preferred coordinate systems.


Coordinate System	Description
EME2000	Body Centered, Earth Mean Equator and Equinox of J2000 (Cartesian) (Fig. 2)
BMEPME	Body Centered, Mean Equator and Prime Meridian of Epoch (Prime Meridian with respect to the Body IAU vector) (Cartesian) (Fig. 3)
BMEPMD	Body Centered, Mean Equator and Prime Meridian of Date (Prime Meridian with respect to the Body IAU vector) (Cartesian) (Fig. 3)
SBBA	Spacecraft Body Centered, Body Axes (Aligned with spacecraft body axes) (Cartesian) (Fig. 4)
SBBR	Spacecraft Body Centered, Body Reference (Aligned with spacecraft body axes) (Cartesian) (Fig. 4)
SBBM	Spacecraft Body Centered, Body Mechanical (Cartesian) (Fig. 5)

The EME2000 coordinate system is an inertial system and can be fixed in any body, although the reference direction is based on the Earth.

To calculate the body rotational rate as a vector in EME2000 coordinates, the body rotational rate as well as the right ascension and declination of the body pole (all listed previously for each body) are utilized in the following equations:

$$\begin{aligned} \text{Body Rotational Rate [0]} &= \dot{W}_{\text{Body}} * \cos(\delta_{\text{BodyPole}}) * \cos(\alpha_{\text{BodyPole}}) \\ \text{Body Rotational Rate [1]} &= \dot{W}_{\text{Body}} * \cos(\delta_{\text{BodyPole}}) * \sin(\alpha_{\text{BodyPole}}) \\ \text{Body Rotational Rate [2]} &= \dot{W}_{\text{Body}} * \sin(\delta_{\text{BodyPole}}) \end{aligned}$$

The BMEPME coordinate system is fixed in a central body located in the solar system and based on the intersection of the central body's prime meridian, as defined by the IAU angle W on the epoch of interest, and the central body's mean equator. This system is an inertial system and does not rotate with the central body's rate of rotation.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 118 of 286	


The BMEPMD coordinate system is fixed in a central body located in the solar system and is initialized based on the intersection of the central body's prime meridian, as defined by the IAU angle  $W$  on the same epoch of interest used for the BMEPME coordinate system, and the central body's mean equator. The difference between the BMEPME and BMEPMD systems is that the central body's prime meridian in the BMEPMD system rotates consistent with the central body's rate of rotation as defined by the IAU angle  $\dot{W}$ . The BMEPMD system is not an inertial system.

The SBBA coordinate system is a right-hand Cartesian system aligned with the axes of the spacecraft body and centered at the spacecraft's center of gravity.  $X_B$  points forward along the longitudinal axis of the spacecraft,  $Y_B$  points out the right side of the spacecraft, and  $Z_B$  completes the right-hand system by pointing downward through the spacecraft.

The SBBR coordinate system is a right-hand Cartesian system aligned with the spacecraft body axes with its origin at an arbitrary point. The  $X_{BR}$  axis is directed along the negative  $X_B$  axis, the  $Y_{BR}$  axis is directed along the positive  $Y_B$  axis, and  $Z_{BR}$  is directed along the negative  $Z_B$  axis. This system is used to locate the vehicle's center of gravity, aerodynamic reference point, and engine gimbals locations for static trim calculations.

The SBBM coordinate system is a right-hand Cartesian system aligned with the axes of the spacecraft body and centered at the spacecraft's 0,0,0 reference point. The  $Y_{BM}$  axis points along the velocity vector, the  $Z_{BM}$  axis points nadir, and the  $X_{BM}$  axis completes the right-hand system. The SBBM system is the coordinate frame used in the aerodatabases. The aerodatabase routines provide the aero moment and force coefficients located at the 0,0,0 reference point which are required for the spacecraft force and moment calculations.

A detailed description of coordinate frames can be found in Ref. 18.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 119 of 286	

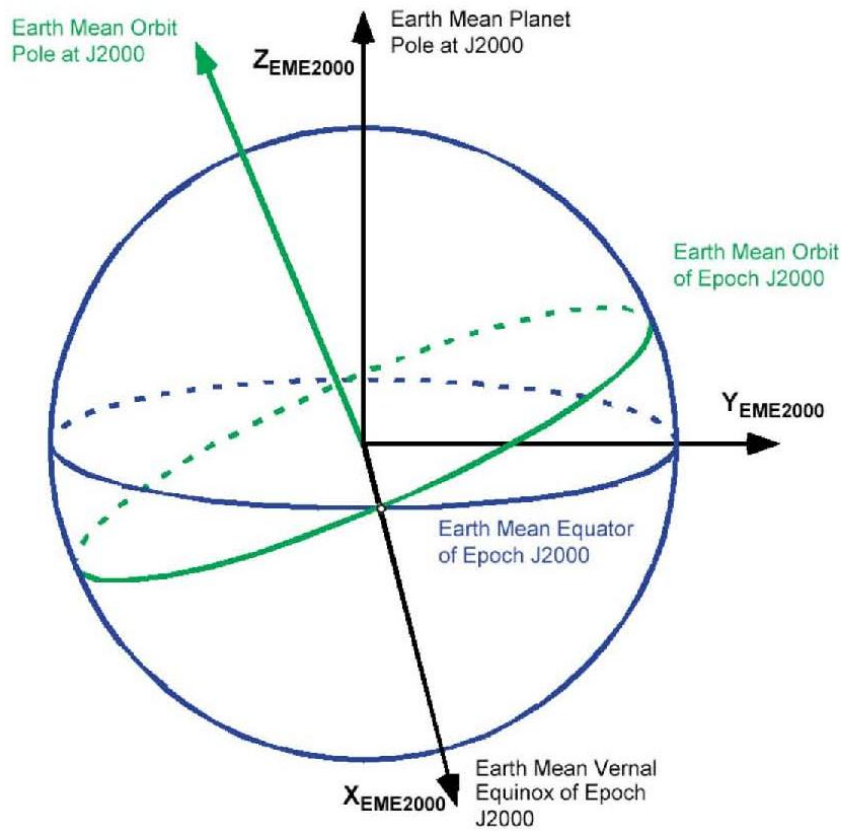

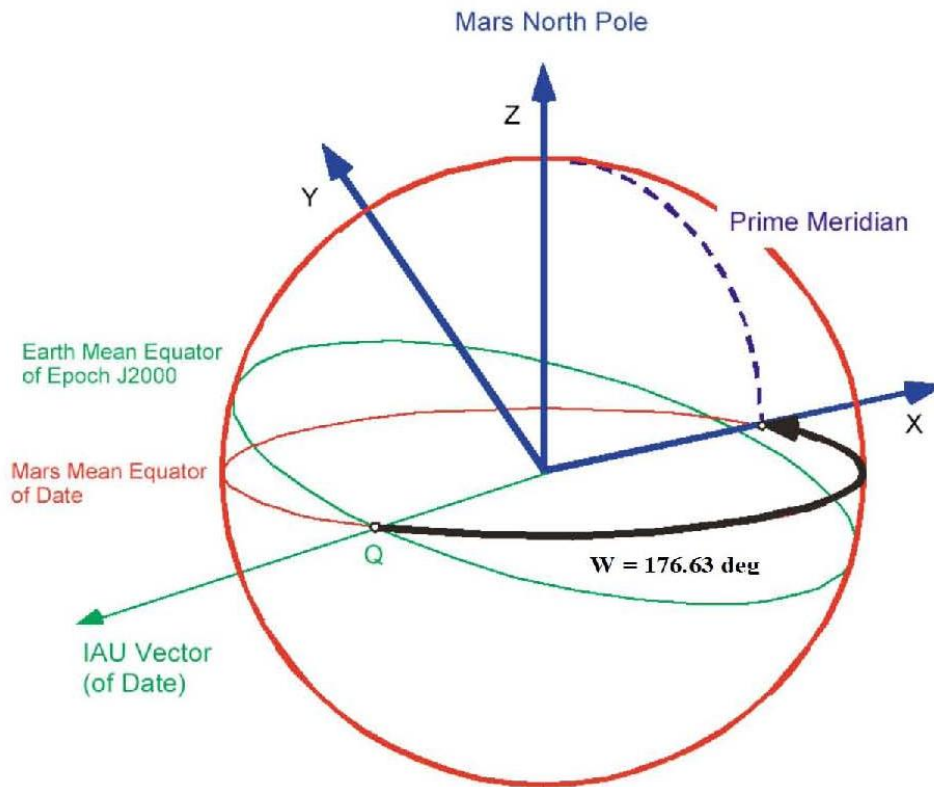



Figure 2: Body Centered Earth Mean Equator and Equinox of J2000 (EME2000) [18]

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 120 of 286	



**Figure 3: Body Centered Body Mean Equator and Prime Meridian of Epoch (BMEPME) shown at J2000 epoch for Mars as Central Body [18]**

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>			Page #: 121 of 286

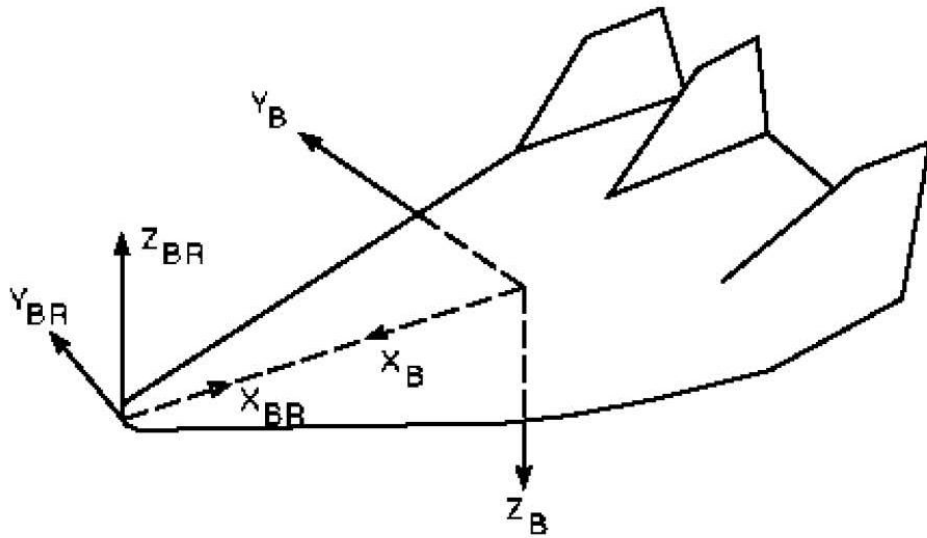



Figure 4: Spacecraft Body Centered, Body Axes (SBA) and Spacecraft Body Centered, Body Reference (SBR) [19]

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 122 of 286	

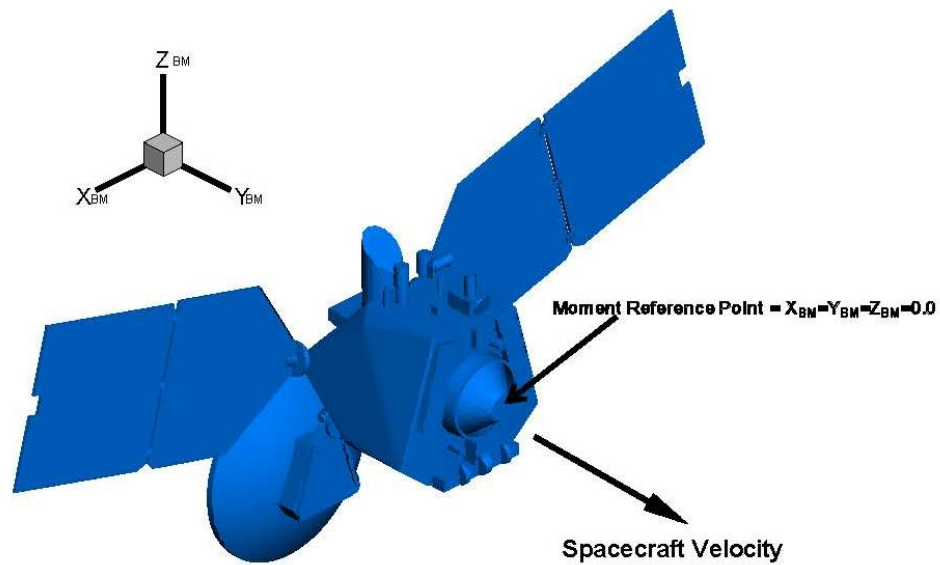



Figure 5: Spacecraft Body Centered, Body Mechanical (SBBM) [20]

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		123 of 286	

### SPACECRAFT MODELS

The assumed spacecraft mass properties are located in Table 1:

<b>Mass (kg)</b>		1395
<b>Reference Length (m)</b>	<b>lref</b>	13.6
<b>Reference Area (m<sup>2</sup>)</b>	<b>sref</b>	37.12
<b>Center of Gravity (m)</b>	<b>Xcg</b>	1.08106
	<b>Ycg</b>	0.00101
	<b>Zcg</b>	0.00462
<b>Aerodynamic reference point (m)</b>	<b>Xref</b>	1.130
	<b>Yref</b>	0
	<b>Zref</b>	0
<b>Moments of Inertia (kg-m<sup>2</sup>)</b>	<b>I<sub>xx</sub></b>	1800
	<b>I<sub>yy</sub></b>	2400
	<b>I<sub>zz</sub></b>	2600
<b>Products of Inertia (kg-m<sup>2</sup>)</b>	<b>I<sub>xy</sub></b>	0
	<b>I<sub>yz</sub></b>	0
	<b>I<sub>zx</sub></b>	0

**Table 1: Spacecraft Mass Properties**

- Note it does not matter where the origin is as long as the aerodynamic reference point and center of gravity are measured in the same frame. The orientation matters, and the required orientation is shown below.

### PROPULSION MODELS


The assumed spacecraft propulsion properties are listed below:

Periapsis Control Thrust = 132 N  
 Periapsis Control Fuel Flow Rate = 0 (Assume no mass is consumed during burns)

### AERODYNAMIC FORCE AND MOMENT CALCULATIONS

The aerodynamic coefficient model requires the following inputs in the order specified:

ichoice = 1 (Integer used to specify linear interpolation for atmosphere relative angle of attack and sideslip angle, and atmospheric density. The aerodynamic coefficients are a function of the above variables).  
  
 rho (Free stream atmospheric density, kg/m<sup>3</sup>)  
  
 alpha (Angle of attack, deg)  
  
 beta (Sideslip angle, deg)

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 124 of 286	

Xcg	(Vector of center of gravity locations in the Spacecraft Body Centered, Body Reference coordinate system)
Xcg(1)	(Distance from the origin of the Spacecraft Body Centered, Body Reference coordinate system to the Xcg, m)
Xcg(2)	(Distance from the origin of the Spacecraft Body Centered, Body Reference coordinate system to the Ycg, m)
Xcg(3)	(Distance from the origin of the Spacecraft Body Centered, Body Reference coordinate system to the Zcg, m)

The aerodynamic coefficient model provides the following aerodynamic coefficients as output:

coef	(Vector of output aerodynamic coefficients)
coef(1)	(Ca= axial force coefficient)
coef(2)	(Cy = side force coefficient)
coef(3)	(Cn = normal force coefficient)
coef(4)	(ClI = roll moment coefficient)
coef(5)	(Cm = pitching moment coefficient)
coef(6)	(Cw = yawing moment coefficient)

The following equations are used to convert coefficients to forces and moments at the center of gravity of the spacecraft in the Spacecraft Body Centered, Body Axes coordinate system. The variable q is the dynamic pressure in Pa units as calculated below:

$$q = 0.5 * (\text{free stream density}) * (\text{atmosphere relative velocity})^2$$


$$XF = -q * sref * Ca \quad (X_B \text{ force})$$

$$YF = q * sref * Cy \quad (Y_B \text{ force})$$

$$ZF = -q * sref * Cn \quad (Z_B \text{ force})$$

$$dx = Xcg - Xref$$



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 125 of 286	

$$\begin{aligned}
 dy &= Y_{cg} - Y_{ref} \\
 dz &= Z_{cg} - Z_{ref} \\
 BRM &= q * s_{ref} * l_{ref} * C_{ll} - (ZF * dy) - (YF * dz) && \text{(Body Rolling Moment)} \\
 BPM &= q * s_{ref} * l_{ref} * C_m + (XF * dz) - (ZF * dx) && \text{(Body Pitching Moment)} \\
 BYM &= q * s_{ref} * l_{ref} * C_w + (YF * dx) + (XF * dy) && \text{(Body Yawing Moment)}
 \end{aligned}$$

The aerodynamic angles used in the aerodynamic database are defined as follows.

- Bank Angle ( $\sigma$ )                      (Positive  $\sigma$  is a positive roll rotation about the atmosphere relative velocity vector.)
- Sideslip angle ( $\beta$ )                      (Positive  $\beta$  is a nose-left (negative yaw) rotation when flying in an upright orientation.)
- Angle of attack ( $\alpha$ )                      Positive  $\alpha$  is a nose-up (positive pitch) rotation for a vehicle flying in an upright orientation.)

An illustration of these angles is shown in Figure 5 relative to the Spacecraft Body Centered Body Axes, where the following definitions apply:

$$\begin{aligned}
 \underline{V}_A & \quad \text{(Velocity vector)} \\
 u_B & \quad \text{(X velocity component)} \\
 v_B & \quad \text{(Y velocity component)} \\
 w_B & \quad \text{(Z velocity component)} \\
 \alpha & = \tan^{-1} \left( \frac{\sin(\alpha)}{\cos(\alpha)} \right) \\
 \beta & = \tan^{-1} \left( \frac{\sin(\beta)}{\cos(\beta)} \right) \\
 \sin \alpha & = \frac{w_B}{\sqrt{u_B^2 + w_B^2}}
 \end{aligned}$$



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
126 of 286

$$\cos \alpha = \frac{u_B}{\sqrt{u_B^2 + w_B^2}}$$

$$\sin \beta = \frac{v_B}{V_A}$$

$$\cos \beta = \frac{\sqrt{u_B^2 + w_B^2}}{V_A}$$

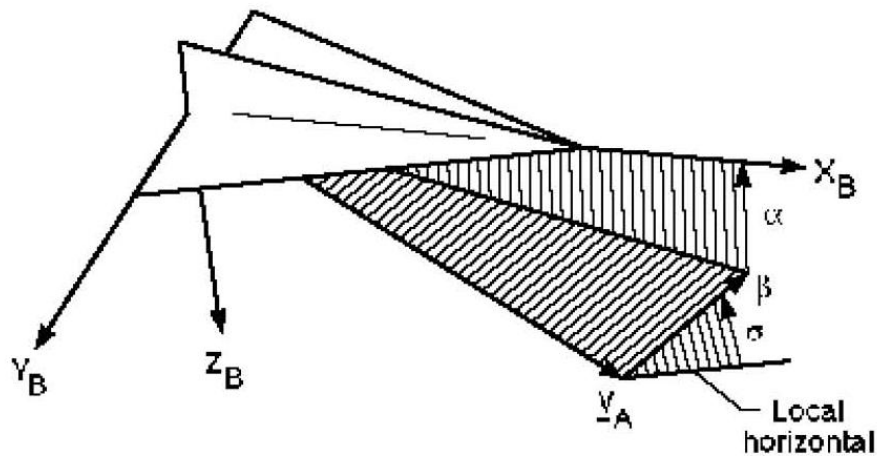



Figure 6: Aerodynamic Angles [19]

### THERMAL MODELS

A high fidelity thermal model of the MRO spacecraft was developed for use at Mars and Venus and can be downloaded from NSCKN using the following link:

<https://nsckn.nasa.gov/DMS/ViewDoc.aspx?DocID=208838>

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 127 of 286	

## APPENDIX

**Converting from Normalized to Un-normalized Gravity Coefficients:**


$$S_{lm} = \frac{\bar{S}_{lm}}{K_{lm}} \quad C_{lm} = \frac{\bar{C}_{lm}}{K_{lm}} \quad J_l = \frac{\bar{J}_l}{K_{lm}}$$

where  $\bar{S}, \bar{C}, \bar{J}$  are the normalized gravity coefficients and  $S, C, J$  are the un-normalized coefficients. The conversion factor  $\Pi$  is given by the formula:

$$K_{lm} = \sqrt{\frac{(l+m)!}{(l-m)!k(2l+1)}}$$


$$k=1 \text{ if } m=0$$

$$k=2 \text{ if } m \neq 0$$


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 128 of 286	

## REFERENCES

1. "Mars Pathfinder Planetary Constants and Models", Robin Vaughan, JPL D-12947, Dec. 1995.
2. "Mars Reconnaissance Orbiter Planetary Constants and Models Document Rev. A", Dolan Highsmith and Bill Strauss, JPL D-22685, May 15, 2003.
3. "Phoenix Project Planetary Constants and Models Document Draft Release", Wyatt Johnson, PHX-377-515, November 22, 2004.
4. "Lunar Constants and Models Document", Ralph Roncoli, JPL D-32296, September 23, 2005.
5. "Mars Science Laboratory Planetary Constants and Models Baseline", Julie Kangas and Angela Bowes, JPL D-27212, May 1, 2006.
6. "Report of the IAU/IAG Working Group on Cartographic Coordinates and Rotational Elements: 2006", P.K. Seidelmann, et al., *Celestial Mechanics and Dynamical Astronomy*, in press, 2007.
7. "Report of the IAU/IAG/COSPAR Working Group on Cartographic Coordinates and Rotational Elements of the Planets and Satellites: 1991", M.E. Davies, et al., *Celestial Mechanics and Dynamical Astronomy*, in press, 1992.
8. Justh, H. L., Justus, C. G., and Keller, V. W., "Global Reference Atmospheric Models, Including Thermospheres, for Mars, Venus and Earth," AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.
9. "Report of the IAU/IAG Working Group on Cartographic Coordinates and Rotational Elements of the Planets and Satellites: 2000", P.K. Seidelmann, et al., *Celestial Mechanics and Dynamical Astronomy*, in press, 2002.
10. Justh, H. L., and Justus, C. G., "Evaluating Mars Science Laboratory Landing Sites with the Mars Global Reference Atmospheric Model (Mars-GRAM 2005)," Third International Workshop on the Mars Atmosphere: Modeling and Observations, 10-13 November 2008, Williamsburg, VA.
11. Email from Fred Pelletier (Cassini OD Team Lead) to A. Bowes, July 29, 2010.
12. "Gravity Field, Shape, and Moment of Inertia of Titan", Luciano Iess, et al., *Science* 327, 1367 (2010), March 12, 2010.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 129 of 286	

13. Justus, C. G., Duvall, A., and Keller, V.W., "Engineering-level model atmospheres for Titan and Mars," Proceedings of the International Planetary Probe Workshop Atmospheric Entry and Descent Trajectory Analysis and Science, 6-9 October 2003, Lisbon, Portugal, ESA Publications Division, ISBN 92-9092-855-7, pp. 311–316, 2004.
14. "The Planetary and Lunar Ephemeris DE 421", W.M. Folkner, J.G. Williams, and D.H. Boggs, JPL IOM 343R-08-003, March 31, 2008.
15. <http://naif.jpl.nasa.gov/naif/tutorials.html>
16. [ftp://naif.jpl.nasa.gov/pub/naif/generic\\_kernels/spk/satellites/mar085.cmt](ftp://naif.jpl.nasa.gov/pub/naif/generic_kernels/spk/satellites/mar085.cmt)
17. "Supporting Online Material for Gravity Field, Shape, and Moment of Inertia of Titan", Luciano Iess, et al., *Science* 327, 1367 (2010), March 12, 2010
18. "MSL Update to Mars Coordinate Frame Definitions", P. Daniel Burkhart, JPL IOM 343B-2006-004, August 15, 2006.
19. "Program To Optimize Simulated Trajectories Formulation Manual", G. L. Brauer, D. E. Cornick, D. W. Olson, F. M. Petersen, and R. Stevenson, September 1990.
20. Chapel, J.D., Schmitz, E., Sidney, W.P., Johnson, M.A., Good, P.G., Wynn, J.A., and Bayer, T., "Attitude Control Performance for MRO Aerobraking and the Initial Science Phase", AAS Guidance and Control Conference, 3-7 February 2007, Breckenridge, CO.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 130 of 286	

## Appendix B. Mission Design Appendix (Supplement to Section 7.2)

Generalized spacecraft geometry and mass properties (as described in Section 7.3.1.1) were used as simulation inputs. Planetary constants and models and the atmosphere GRAM models for each destination were used and are detailed in the AA PCMD (Appendix A) and in Section 7.2. Aerodynamics models developed specifically for use with this spacecraft at each destination were utilized and are described in Section 7.3.1.2. Solar radiation pressure and the Sun as a third-body perturbation were included for each destination. Additionally, Saturn was included as a third-body perturbation at Titan, and a thermal model as described in Section 7.3.2.4 was used in the Venus reference simulation.

For the Mars reference simulation, an initial orbital state was selected from the MRO flight profile after the "walk-in" phase of the AB mission was completed, and an initial epoch near  $L_s=250$  degrees was used to ensure the Martian dust storm season was encompassed during the AB duration. A full AB mission was then simulated until the apoapsis altitude reached 450 km. Maneuvers were constrained to occur no more frequently than once a week. The estimated freestream heat rate at periapsis was used as the corridor control parameter to which the spacecraft must be kept during the main AB phase. This is the same constraint utilized in both the Odyssey and MRO AB missions. For this analysis, the corridor was set to 0.11 watts per centimeter squared ( $w/cm^2$ ) to 0.17  $W/cm^2$ . Since maneuvers were constrained to once a week, it was necessary to bias the target within this corridor as a function of orbit period: 80 percent of the total corridor width for orbit periods greater than 2.5 hours and 30 percent for orbit periods less than 2.5 hours. This bias was necessary to ensure the corridor was maintained (upper limit not exceeded) throughout the AB mission while meeting the maneuver frequency constraint.

Figure B.1 shows the operational corridor performance for the reference simulation at Mars.



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

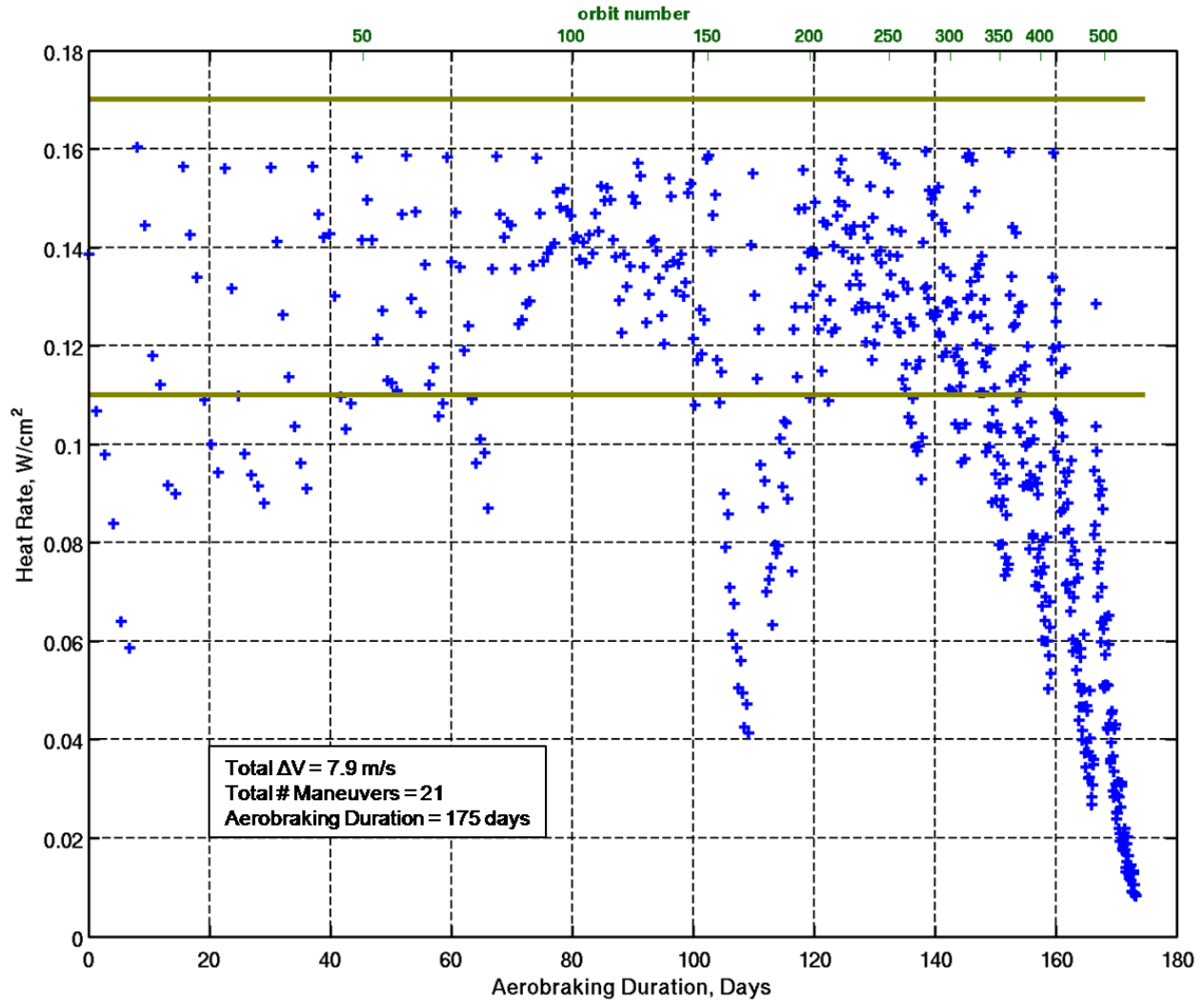
Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

Page #:

131 of 286



*Figure B.1. Mars Reference Simulation Corridor Performance*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 132 of 286	

For the Venus reference simulation, a 24-hour initial orbit period was selected and a full AB mission was simulated until the apoapsis altitude reached 450 km. Due to the third-body effects of the Sun at Venus pulling the spacecraft periapsis altitude higher at some times during the mission and lower at others, maneuvers are required more frequently than at Mars to keep the spacecraft within the operational corridor and thus were allowed to occur once per day. Because of higher solar flux and greater periapsis velocities at Venus, there is increased aerodynamic heating experienced during drag passes than at Mars. Therefore, the temperature of the spacecraft at periapsis was used as the operational corridor to which the spacecraft must be kept since temperature limits are the driving constraint in the structural integrity and functional performance of the spacecraft. For this analysis, the temperature corridor was set from 275 to 375 °C. (The temperatures seen at Venus in this analysis are artificially high because the spacecraft utilized in the analysis was not specifically designed for AB at Venus.) Because the temperatures experienced at periapsis are a direct function of the periapsis velocity and hence orbit period, it was necessary to bias the target within this corridor as a function of orbit period: 80 percent for orbit periods greater than 10 hours, 90 percent for orbit periods less than 10 hours and greater than 2.17 hours, and 5 percent for orbit periods less than 2.17 hours. This bias was necessary to ensure the corridor was maintained (upper limit not exceeded) throughout the AB mission while meeting the maneuver frequency constraint and to maintain the desired AB rate even with the perturbing effects of the Sun. Figure B.2 shows the operational corridor performance for the reference simulation at Venus.





# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

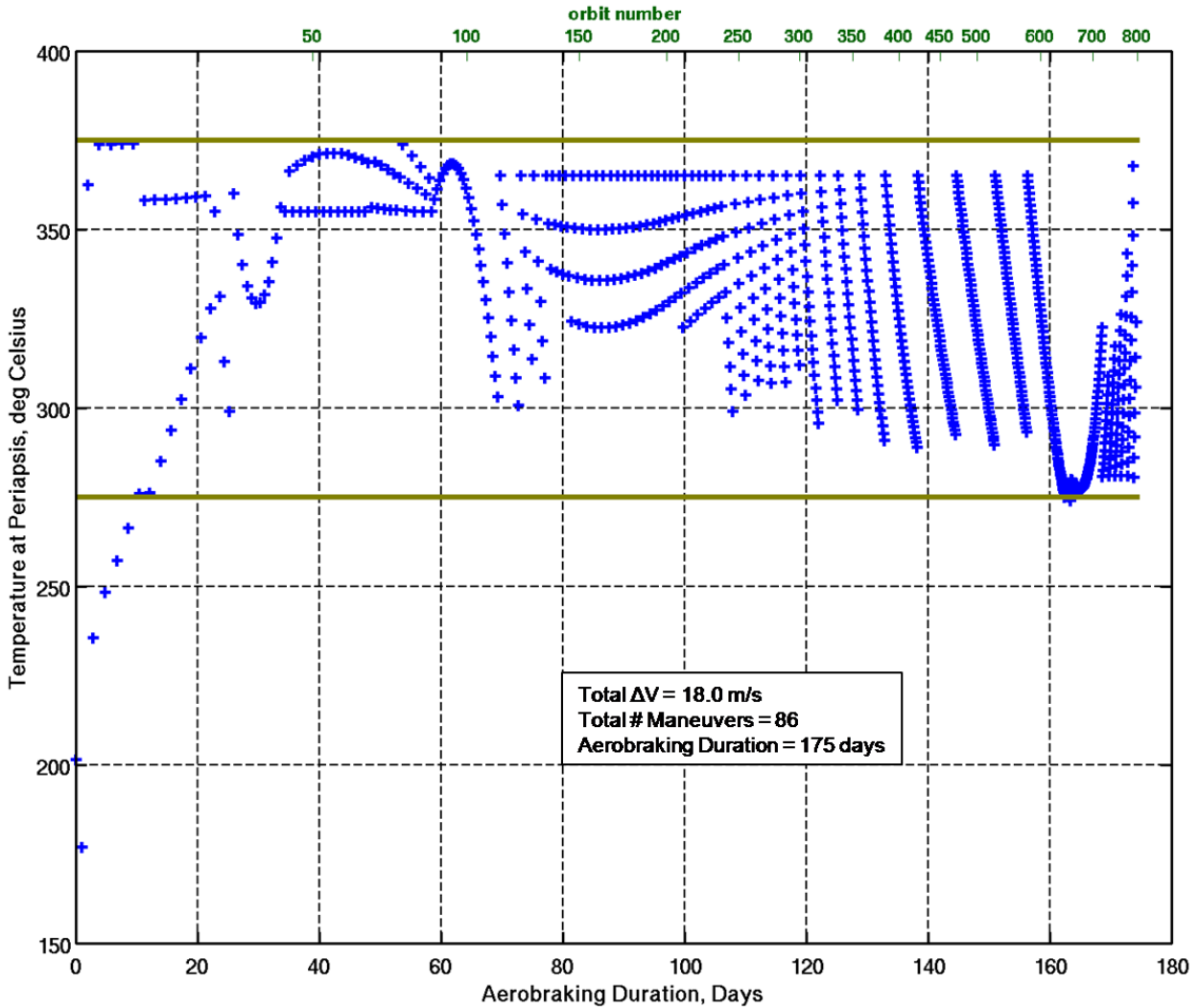
Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

Page #:

133 of 286



*Figure B.2. Venus Reference Simulation Corridor Performance*


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 134 of 286	

AB at Titan is different than at Mars and Venus due to the perturbing effects of Saturn's and Titan's large atmospheric scale height. Saturn can induce large changes in periapsis and apoapsis altitudes and these effects are dependent on the orbit conditions and orientation relative to Saturn. For instance, Saturn can cause shifts in periapsis altitude of over 100 km from one orbit to the next in a 20-hour period orbit studied in this analysis. Whether the altitude swing is upward or downward is dependent on the orbit orientation relative to Saturn. As the orbit period and apoapsis altitudes decrease, the effects of Saturn lessen. In an 8-hour period orbit with the same orbit orientation relative to Saturn as the 20-hour orbit, Saturn causes shifts in periapsis altitude on the order of only 10 km from one orbit to the next. Saturn's pull on the apoapsis and periapsis altitudes is less pronounced when the orbit plane is nearly normal to Saturn.

Due to Saturn's effects, the overall  $\Delta V$  requirement for corridor control at Titan can be large (e.g., up to 4 m/s per day spent in AB (averaged)) when starting in the initial orbit period of 20 hours examined in this analysis. Without Saturn, AB can be performed using the same initial 20-hour orbit period for less than 0.1 m/s  $\Delta V$  per day (averaged) because the large atmospheric scale height at Titan means that there is less orbit-to-orbit variability in density during AB. The large atmospheric scale height has the effect of increasing the size of corridor control maneuvers as compared to the sizes at Mars and Venus.) Thus, Saturn's effect at Titan is to decrease the typical  $\Delta V$  savings achieved by AB in comparison to the savings at Mars and Venus because of the  $\Delta V$  required for corridor control when AB for any significant duration.

One method to minimize the corridor control  $\Delta V$  at Titan is to aerobrake as quickly as possible. Short AB durations (7–20 days when starting in an initial 20-hour period orbit) are feasible at Titan while meeting spacecraft heating constraints because the periapsis velocities at Titan are lower than they are at Mars and Venus, and the spacecraft can therefore experience higher densities for the same heating rate as found at Mars and Venus. However, AB at Titan represents an opportunity for in situ atmospheric sampling which would be of great benefit to science. This science return would be enhanced as the AB duration increases. Hence, a trade space exists when designing a Titan mission where there is some balance between the AB duration and the  $\Delta V$  required for corridor control to still provide reasonable  $\Delta V$  savings to justify AB versus all propulsive maneuvering.


This trade space is heavily dependent on the initial orbit conditions relative to Saturn and has not been extensively examined here. Rather, an initial 8-hour period orbit was selected for the Titan reference simulation since this orbit period seemed to be the threshold where Saturn's effects on the corridor control  $\Delta V$  could be reduced to an average value of  $\sim 1$  m/s per day spent AB. Additionally, even though it is possible to begin AB at Titan from a larger orbit period (such as the 20-hour period orbit mentioned previously), it only takes a few atmospheric passes before the orbit period is reduced to approximately 8 hours. These first few atmospheric passes in the larger orbit period are similar to the walk-in phase of AB at Mars and Venus where the atmosphere is being sampled and therefore would not be utilized with AA.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 135 of 286	

A full AB mission was simulated from the initial 8-hour orbit period with the same epoch as was used in the Mars case until the apoapsis altitude reached 1500 km. The initial orbit was nearly polar and had the periapsis located near the South Pole, similar to a previous Titan AB study [ref. B.1]. An initial periapsis altitude of 810 km was selected to ensure that AB occurred from the beginning of the simulation. A periapsis altitude corridor was chosen for this analysis because it provides a more straightforward control approach in the presence of Saturn than other corridor types when the desire is to stretch the duration of AB. The corridor was set at 750 km to 900 km and an altitude of 870 km was used as the corridor target for the entire AB duration. Both the corridor and target within the corridor were selected with the objective of achieving ~2-month AB duration.

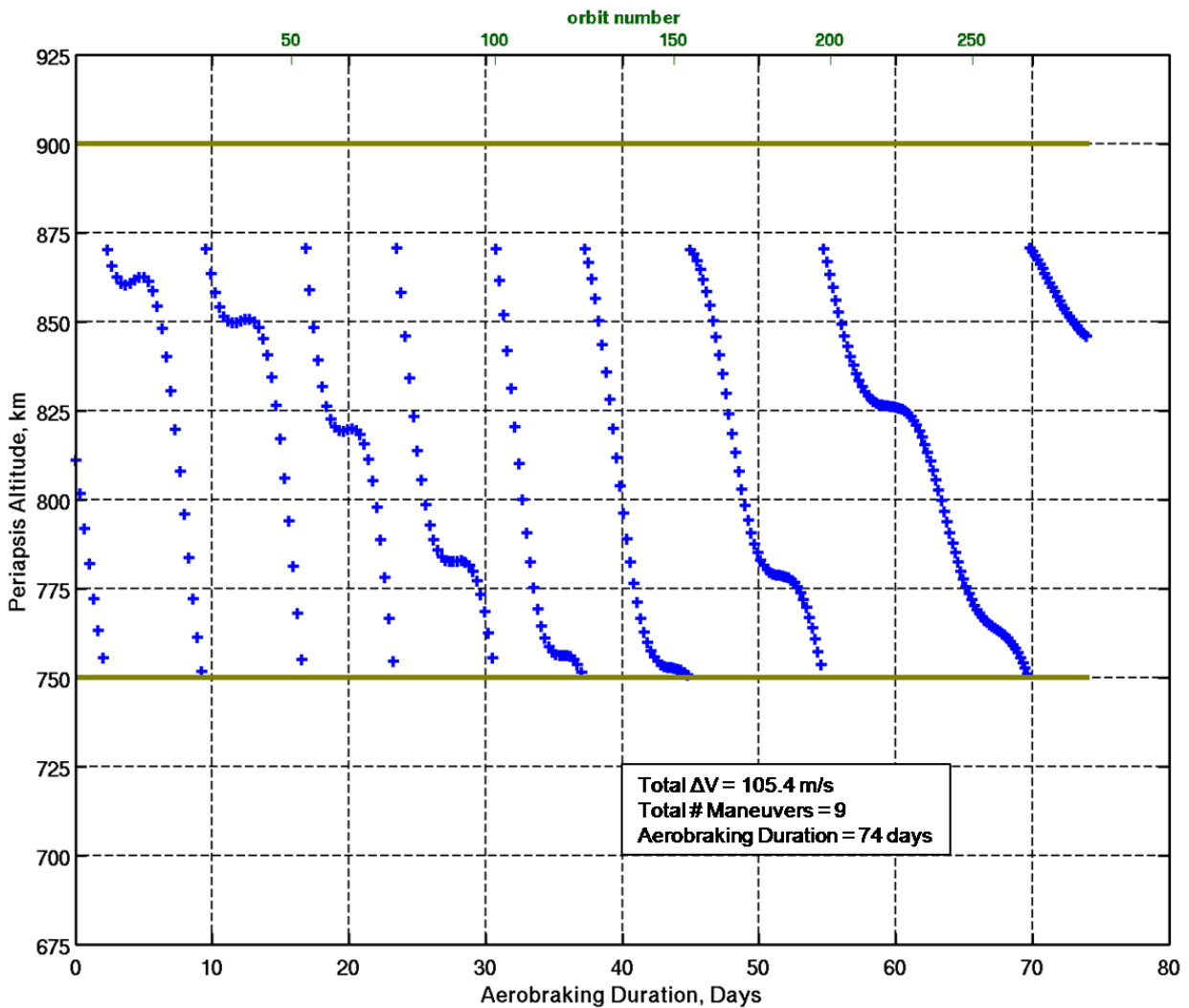
Figure B.3 shows the operational corridor performance for the reference simulation at Titan. Note that there are regularly occurring periods of time spanning ~2 days where the periapsis altitude remains almost constant rather than dropping steadily from one orbit to the next. An examination of the Saturn-Titan-Probe angle as seen in Figure B.4 indicates a relationship exists between the Saturn-Titan-Probe angle and the periapsis behavior. As the Saturn-Titan-Probe angle approaches 90 degrees (i.e., Saturn is nearly normal to the orbit plane examined in this analysis), the periapsis altitude remains almost constant. The occurrences of the Saturn-Titan-Probe angle being near 90 degrees take place at regularly spaced intervals of ~8 days since the orbit period of Titan around Saturn is just under 16 days. Once the spacecraft orbit period shrinks to ~4.5 hours, the effects of Saturn lessen on the periapsis altitude. This can be seen between days 65–70 in Figure B.3 where the periapsis altitude does not remain quite as steady when the Saturn-Titan-Probe angle is near 90 degrees.

By setting a constraint that maneuvers can only take place when the Saturn-Titan-Probe angle is ~90 degrees, and then setting the corridor limits and target within the corridor such that the spacecraft has reached or is below the bottom of the corridor when the Saturn-Titan-Probe angle is ~90 degrees, the amount of required corridor control  $\Delta V$  is reduced. This is illustrated in Figure B.5 where another full AB mission was simulated at Titan using the same initial 8-hour period as presented in Figures B.3 and B.4. The corridor was set at 825 km to 875 km at the onset of AB but was changed to 750 km to 875 km once the orbit period reached just under 4.5 hours. Although the lower corridor limit could have been set to 750 km at the onset of AB, the lower limit would have needed to be shifted upward repeatedly as AB progressed to ensure the spacecraft was in the desired location of the corridor when the Saturn-Titan-Probe angle was ~90 degrees and a periapsis raise maneuver was triggered to prolong AB. Dropping the lower corridor limit to 750 km later in AB was done specifically in this case to avoid performing another periapsis raise maneuver only to reach the desired apoapsis altitude soon after the maneuver. In some mission simulations analyzed, performing another maneuver targeting near the top of the corridor is advantageous to the AB duration rather than unfavorable as was the case with this simulation. The corridor target was an altitude of 860 km for the entire AB duration in this mission.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		136 of 286	

The strategy used here to prolong AB while reducing corridor control  $\Delta V$  can be utilized to increase the AB duration even more than shown for this case if preferred. One such simulation provided 122 days in AB for 120 m/s  $\Delta V$  when beginning from the same initial conditions versus the 69 days and 72 m/s  $\Delta V$  shown here. Further study in the relationship of periapsis/apoapsis altitude behavior to orbit orientation/conditions relative to Saturn will likely provide additional insight into methods that can be employed to optimize an AB mission design at Titan.

Table B.1 summarizes several parameters of interest for the Mars, Venus, and Titan reference simulations.



*Figure B.3. Titan Nominal Reference Simulation Corridor Performance*



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

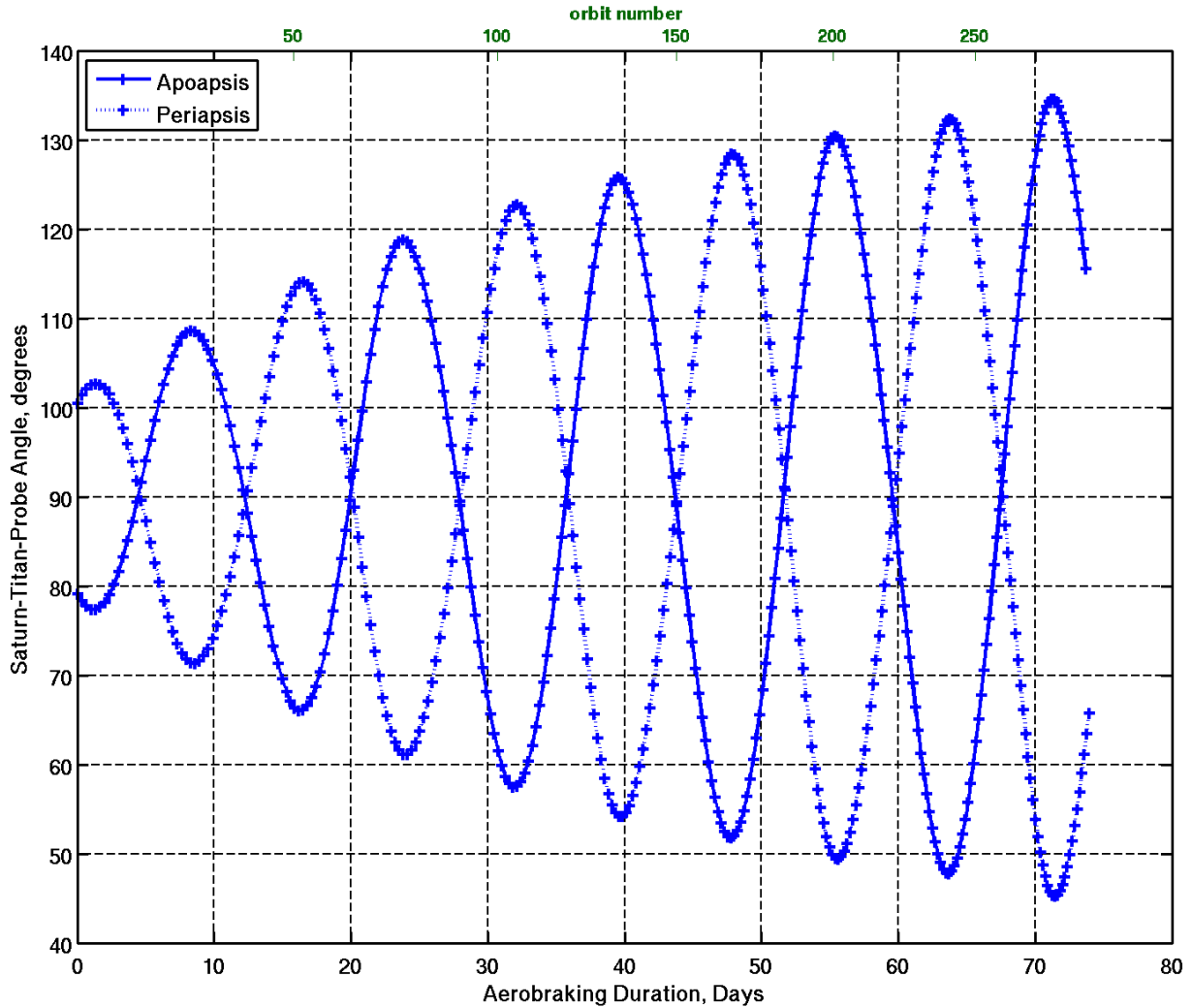
Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

137 of 286



*Figure B.4. Saturn-Titan-Probe Angle during Nominal Reference Simulation*



# NASA Engineering and Safety Center Technical Assessment Report

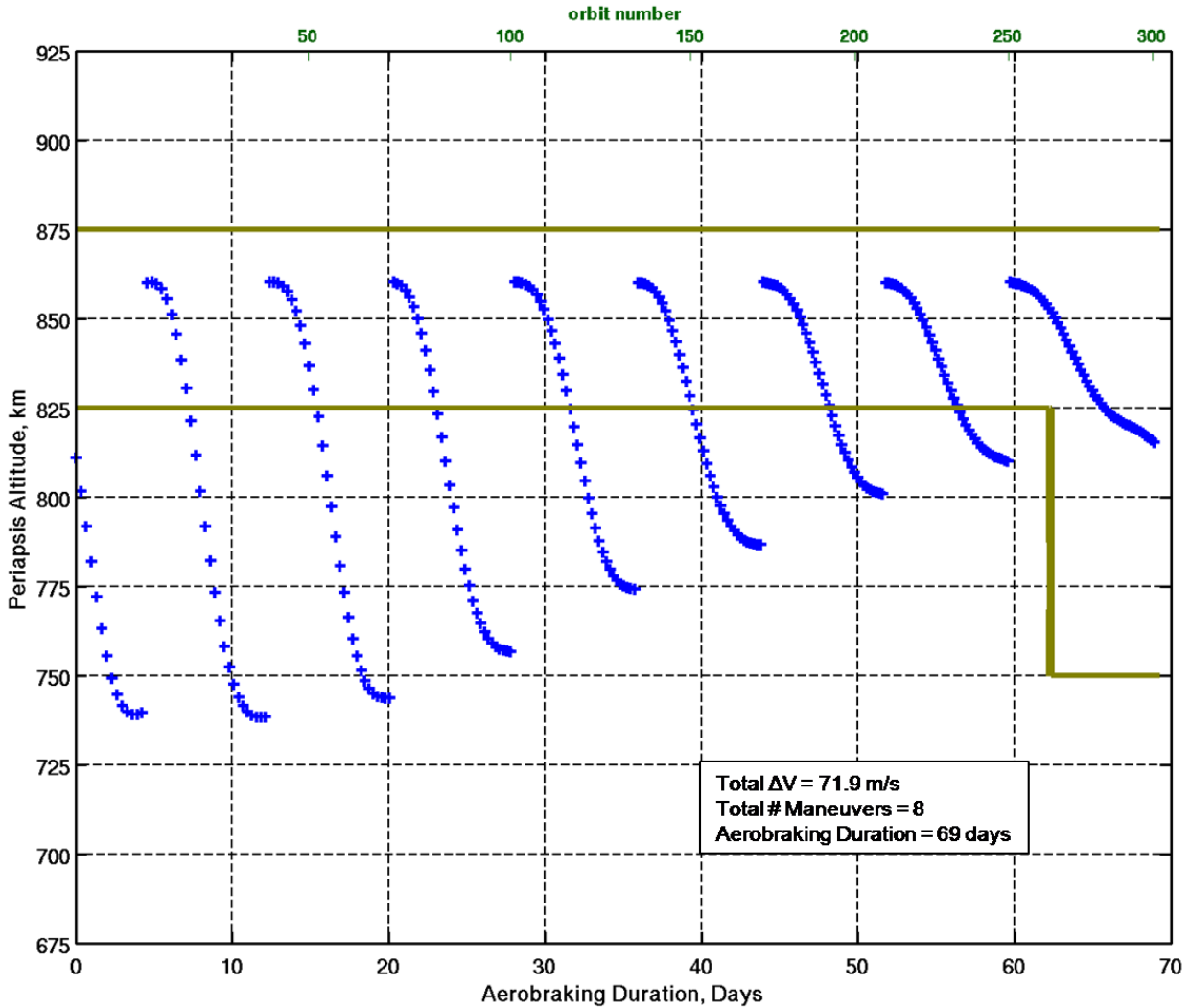
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**


Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
138 of 286



*Figure B.5. Titan Reduced  $\Delta V$  Reference Simulation Corridor Performance*


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:			Page #:
<b>Autonomous Aerobraking (Phase 1)</b>			139 of 286

*Table B.1. AB Summary*

	Mars	Venus	Titan Nominal	Titan Reduced $\Delta V$
Initial Periapsis Altitude (km)	101.65	142.58	810.86	810.86
Initial Orbit Period (hours)	33.43	23.78	8.19	8.19
Range of Periapsis Altitudes (km)	99.06 - 127.03	126.31 - 148.62	750.50 - 870.57	738.34 - 860.32
Maximum Qdot ( $W/cm^2$ )	0.16	1.18	0.007	0.009
Maximum Dynamic Pressure ( $N/m^2$ )	0.41	1.20	0.034	0.05
Maximum Density ( $kg/m^3$ )	5.13e-8	2.50e-8	1.78e-8	2.35e-8

## Reference

Ref. B.1: Lyons, D. and Strange, N., "Aerobraking at Titan," *AAS/AIAA Astrodynamics Specialist Conference*, Pittsburgh, Pennsylvania, August 9–13, 2009.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 140 of 286	


## **Appendix C. Aerodynamics and Aerothermodynamics Computational Methods (Supplement to Section 7.3.1.2)**

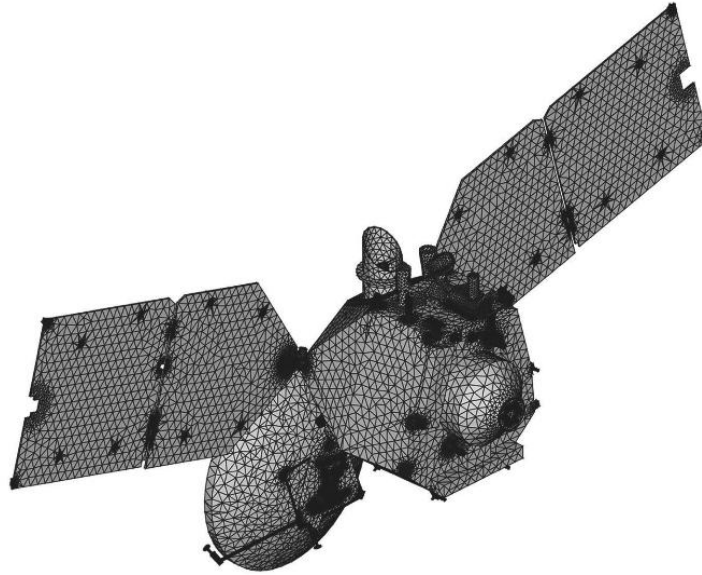
### **C.1 Computational Methods**

The DSMC calculations were performed using Distributed DSMC Analysis Code, the parallel implementation of the program DAC (DSMC Analysis Code) [refs. 2–4]. In DAC, the gas molecular collisions are modeled using the variable hard-sphere model developed by Bird [ref. 1], and the Larsen-Borgnakke model is used for internal energy exchanges [ref. 5]. The surface geometry is represented by unstructured triangular elements that are embedded in a two-level Cartesian grid for the flowfield calculation. The solution from the first level of grid cells, which are uniform in size, is used for grid refinement to create the second-level cells. The grid is refined based on local conditions, thus allowing the program to meet the spatial resolution requirements without excessive global refinement. The grid cells are typically refined such that, on average, the second-level cells have dimensions less than the local mean free path. The local simulation parameters are set such that there are nominally 10 simulated molecules in each cell and the local time step is typically dictated by the local flow time for the problems considered. The simulation is allowed to progress until there is approximately a constant number of particles in the flow domain, then it is run in a steady state mode until a sufficient number of surface collisions are sampled.

For all calculations, the wall collisions were assumed to be fully diffusive; that is, an accommodation coefficient of one was specified, with the spacecraft wall temperature at a constant 300 K. The free stream parameters used for the atmospheres of Mars, Venus, and Titan are presented in Tables C.1-1, C.1-2, and C.1-3, respectively. The baseline vehicle geometry was that of the MRO and can be seen in Figure C.1-1. The surface grid was derived from a computer-aided design file provided by LM Astronautics and represents the best preflight estimate of the nominal AB configuration. The spacecraft itself is about 12 m wide from the outside tip of one solar array to the other. At each trajectory point, angles of attack and side-slip angles of 0 deg,  $\pm 5$  deg, and  $\pm 10$  deg were simulated to span the expected range of orientations expected.




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		141 of 286	



*Figure C.1-1. Space Craft Surface Geometry*

*Table C.1-1. Mars Simulation Parameters*

$n_{\infty}$ (1/m <sup>3</sup> )	$\rho_{\infty}$ (kg/m <sup>3</sup> )	$V_{\infty}$ (m/s)	$T_{\infty}$ (K)	$X_{CO_2}$	$X_{N_2}$	$X_O$	$X_{CO}$	$X_{He}$	$X_N$
7.80E+16	5.60E-09	4811	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000
1.39E+17	1.00E-08	4811	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000
2.48E+17	1.78E-08	4811	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000
4.40E+17	3.16E-08	4811	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000
1.39E+18	1.00E-07	4811	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000
2.09E+18	1.50E-07	4211	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000
3.48E+18	2.50E-07	3911	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000
4.87E+18	3.50E-07	3611	144.77	0.9537	0.0463	0.0000	0.0000	0.0000	0.0000

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		142 of 286	

*Table C.1-2. Venus Simulation Parameters*

$n_{\infty}$ (1/m <sup>3</sup> )	$\rho_{\infty}$ (kg/m <sup>3</sup> )	$V_{\infty}$ (m/s)	$T_{\infty}$ (K)	$X_{CO_2}$	$X_{N_2}$	$X_O$	$X_{CO}$	$X_{He}$	$X_N$
1.08E+15	3.42E-11	9000	127.4	0.0659	0.0548	0.7893	0.0671	0.0175	0.0054
7.10E+15	3.13E-10	9000	127.4	0.3552	0.0842	0.4537	0.1009	0.0033	0.0026
1.09E+17	6.85E-09	9000	127.6	0.7310	0.0676	0.1427	0.0575	0.0003	0.0008
5.53E+17	3.69E-08	9000	128.0	0.8267	0.0539	0.0845	0.0344	0.0001	0.0005
2.92E+18	2.02E-07	9000	129.0	0.8887	0.0461	0.0455	0.0194	0.0001	0.0002

*Table C.1-3. Titan Simulation Parameters*


$n_{\infty}$ (1/m <sup>3</sup> )	$\rho_{\infty}$ (kg/m <sup>3</sup> )	$V_{\infty}$ (m/s)	$T_{\infty}$ (K)	$X_{N_2}$	$X_{CH_4}$	$X_{H_2}$
3.34E+16	1.54E-09	6500	174.63	0.9837	0.013	0.0033
7.01E+16	3.23E-09	6500	171.90	0.9850	0.0124	0.0026
1.55E+17	7.16E-09	6500	166.79	0.9867	0.0117	0.0017
3.67E+17	1.69E-08	6500	159.74	0.9878	0.0111	0.0011
9.37E+17	4.32E-08	6500	151.42	0.9887	0.0106	0.0007

## C.2 Database Uncertainty Analysis for Aeroheating

With any database, some uncertainty must be assigned to the heating levels reported. While some uncertainties are generalized and apply to any data set, there were several that needed to be quantified for the present databases. For the current work, an analysis was performed on the Martian atmosphere and is assumed to be valid for the atmospheres of Venus and Titan as well.

### C.2.1 Effect of Chemical Reactions

For the Martian database, a two-species, non-reacting chemistry model was implemented while a nine-species, reacting chemistry model was implemented for the Venus and Titan databases. To assess the uncertainty associated with the differing chemistry models, comparisons were made at the highest expected density at Mars (32 kg/km<sup>3</sup>). The variation of the non-dimensional incident heating coefficient,  $C_H$ , was compared and the maximum difference was estimated to be less than 5 percent at the edges of the solar panels.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 143 of 286	

### C.2.2 Effect of Surface Grid Resolution

The next uncertainty examined was the effect of changing the surface grid resolution at the highest expected Martian density ( $32 \text{ kg/km}^3$ ). The nominal surface grid was compared with a surface grid for which the size of the surface elements was decreased by approximately one-half. The greatest differences observed were near the corners of the solar panels and were estimated to be less than 1 percent.

### C.2.3 Effect of Surface Temperature


The final simulation parameter to be examined for uncertainty was the variation of the incident heating rate with the wall temperature specified for the spacecraft, once again at the highest expected Martian density ( $32 \text{ kg/km}^3$ ). The nominal wall temperature was chosen to be 300 K. Off-nominal surface temperatures of 150 K and 600 K were chosen for comparison. The value of 600 K was obviously higher than any expected in-flight temperature but is included to get the maximum uncertainty level. As it turns out, the 150 K off-nominal temperature was lower than any of the temperatures observed near the atmospheric entry portion of the MRO AB maneuver, but it provided a reasonable lower bound. The greatest differences observed were approximately 5 percent.

### C.2.4 Summary of Database Uncertainty

A summary of the uncertainties included in the overall estimate of database uncertainty is presented in Table C.2-1. The main sources of uncertainty are computational errors (statistical sampling, gridding errors), physical model errors (gas collision model used, accommodation coefficient used, chemical reactions), boundary conditions (atmospheric temperature, surface temperature), and any errors in the computational geometry model used (whether the multilayer insulation was applied correctly, simplifications to some parts, etc.). While this may not be an all-inclusive list of possible sources of error, the major contributors have been included and examined.

The database uncertainty has been reported with and without the inclusion of the accommodation coefficient uncertainty. The thermal analysis team used the uncertainty with this value because their analysis includes the reflected heating rate in addition to the incident heating rate. The accommodation coefficient affects the incident heating only slightly (by varying the number density near the surface). The total uncertainty was calculated by taking the square-root-of-the-sum of the squares of the contributing uncertainties.

The grid, chemical reaction, and surface temperature have already been addressed here. The statistical sampling error was estimated by approximating the uncertainty as  $1 / \sqrt{N}$ , where  $N$  is the number of surface collisions. Because most of the surface elements accumulated on the order of one million collisions, this uncertainty was estimated to be  $\pm 0.1$  percent. The gas collision model, accommodation coefficient, and atmospheric temperature uncertainties are historic values

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		144 of 286	

that have been used with confidence in previous planetary missions. The geometry error was an uncertainty to which it was difficult to assign a value. This uncertainty is only mentioned and was not assigned a value because a direct comparison between the computational model and the spacecraft in flight cannot be made. The final database uncertainties are therefore assigned values of  $\pm 7.9$  and  $\pm 9.4$  percent, with and without the inclusion of accommodation coefficient uncertainty, respectively.

*Table C.2-1. Summary of Uncertainties*


Source of Uncertainty	Aeroheating (percent)	Aerodynamics (percent)
Computational errors		
Statistical sampling	$\pm 0.1$	$\pm 0.05$
Grid	$\pm 3.0$	$\pm 1.0$
Physical errors		
Gas collision models	$\pm 2.0$	$\pm 1.0$
Accommodation coefficient	$\pm 5.0$	$\pm 2.5$
Chemical reactions	$\pm 5.0$	$\pm 1.0$
Boundary Conditions		
Atmospheric temperature	$\pm 0.2$	$\pm 0.1$
Surface temperature	$\pm 5.0$	$\pm 0.5$
Geometry	Small	Small
RMS uncertainty (excl. acc. coef.)	$\pm 7.9$	
RMS uncertainty (incl. acc. coef.)	$\pm 9.4$	$\pm 3.08$

### C.3 Database Uncertainty Analysis for Aerodynamics

The sources of uncertainty for the aerodynamics databases are the same as for the aeroheating databases, but the sensitivity of the force and moment coefficients is much smaller than for the heating coefficient. The resulting uncertainties for the aerodynamic databases are listed in Table C.2-1.

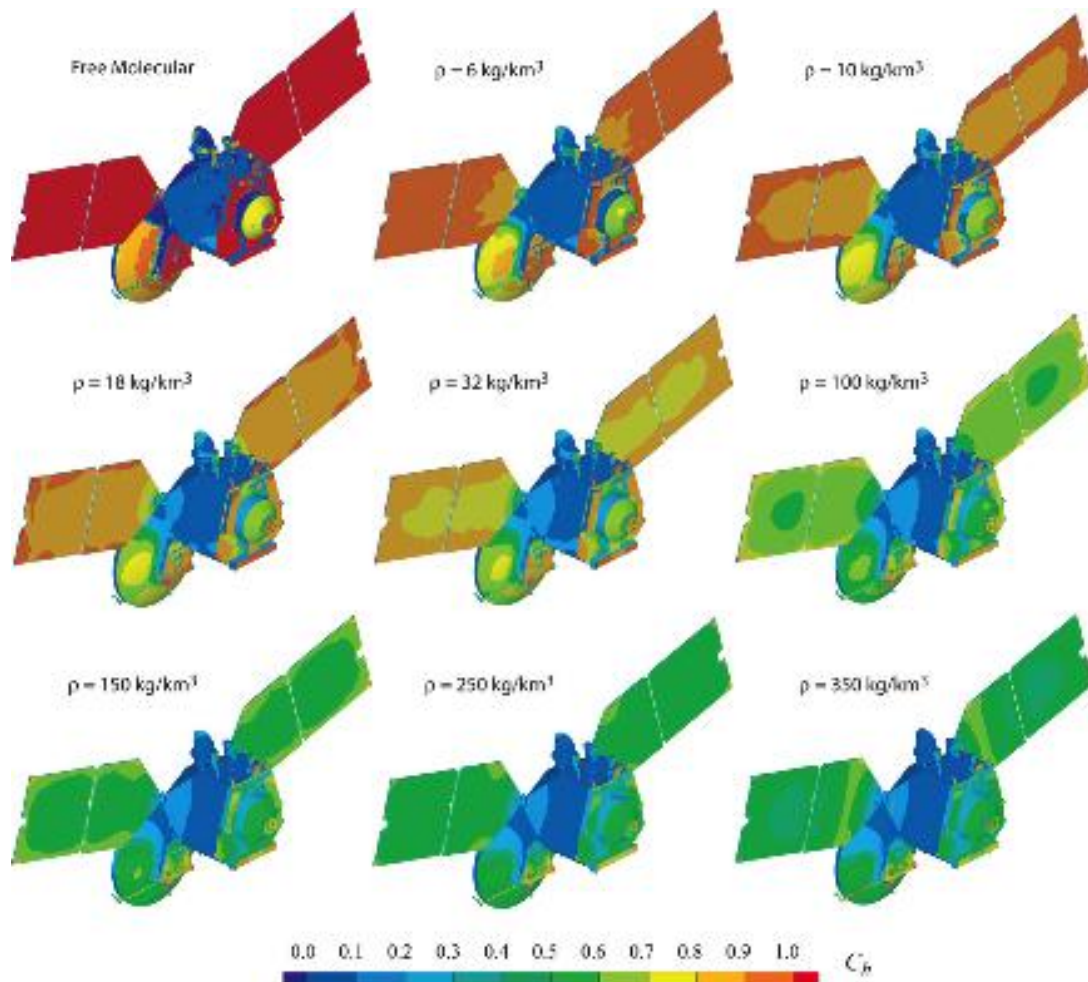
### C.4 Database Trends

The following section gives the reader a general knowledge of how the aerodynamic and aeroheating coefficients vary as the density and orientation vary through an atmospheric pass. No specific comparisons will be given between atmospheres since the free stream conditions vary widely depending on which planet/moon is being considered.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 145 of 286	

### C.4.1 Aeroheating Database

Due to the elliptic nature of AB orbits, a range of densities is generally encountered, from the free-molecular regime down through the maximum density allowed by the thermal limit lines (as determined by the Thermal Team) where the flow may become transitional. Heating results for a range of densities from the Martian atmosphere simulations are presented in Figure C.4-1. Although the non-dimensional heating may be decreasing as the density increases, it should be noted that the heating rates are increasing since the density is increasing. The value of  $C_h$  decreases with density because the kinetic energy of the incident molecules is decreasing due to the increasing number of collisions in the flow.



**Figure C.4-1. Effect of Density on  $C_h$  at  $\alpha = 0\text{-deg}$ ,  $\beta = 0\text{-deg}$**

As discussed, a variety of angles-of-attack and side-slip angles were examined for each trajectory. A sample of how the distribution of  $C_h$  varies with these parameters is presented in


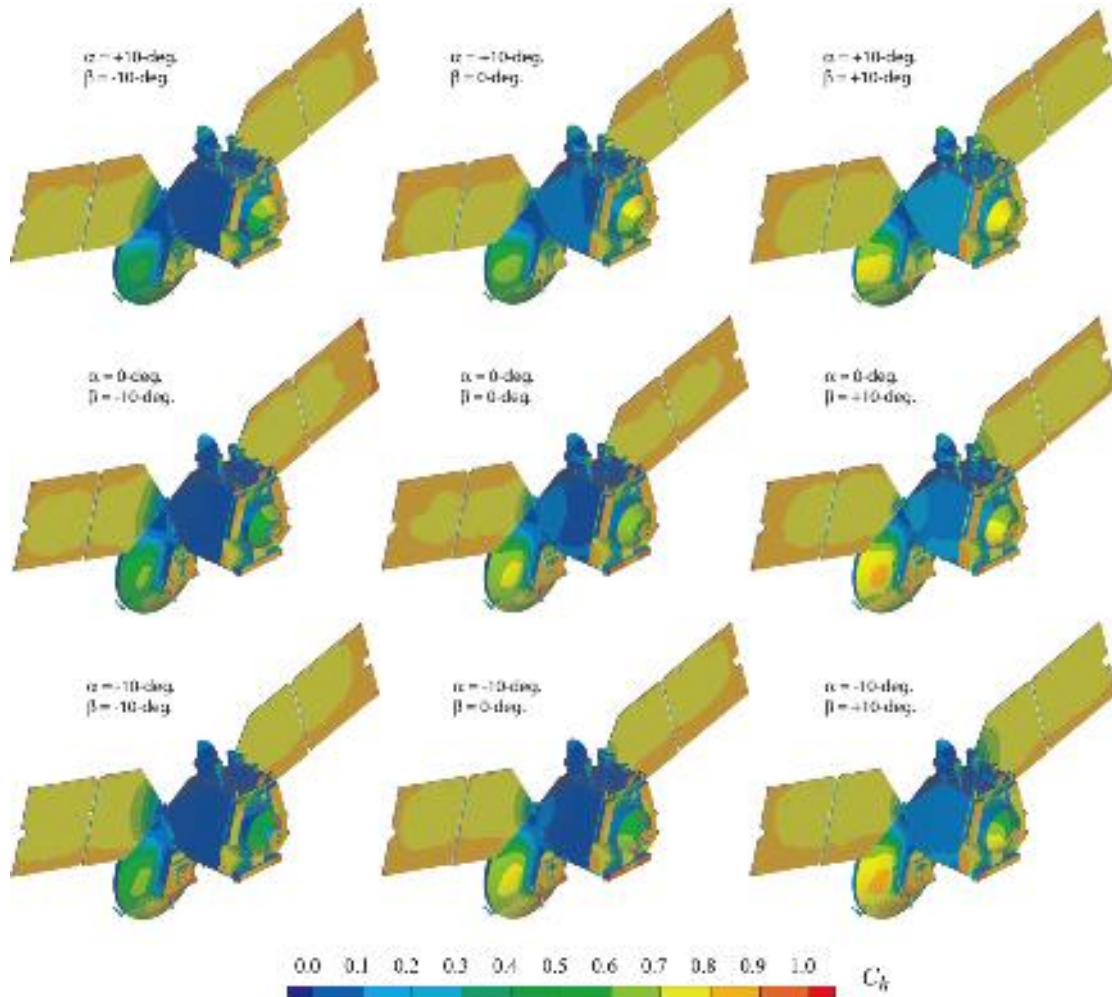
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 146 of 286	

Figure C.4-2. The spacecraft is assumed to pass through the atmosphere with the science instruments pointed downward towards the surface, so in this figure, the ground is “up.”



**Figure C.4-2. Effect of Orientation on  $C_l$**



Title:

Autonomous Aerobraking (Phase 1)

Page #:

147 of 286

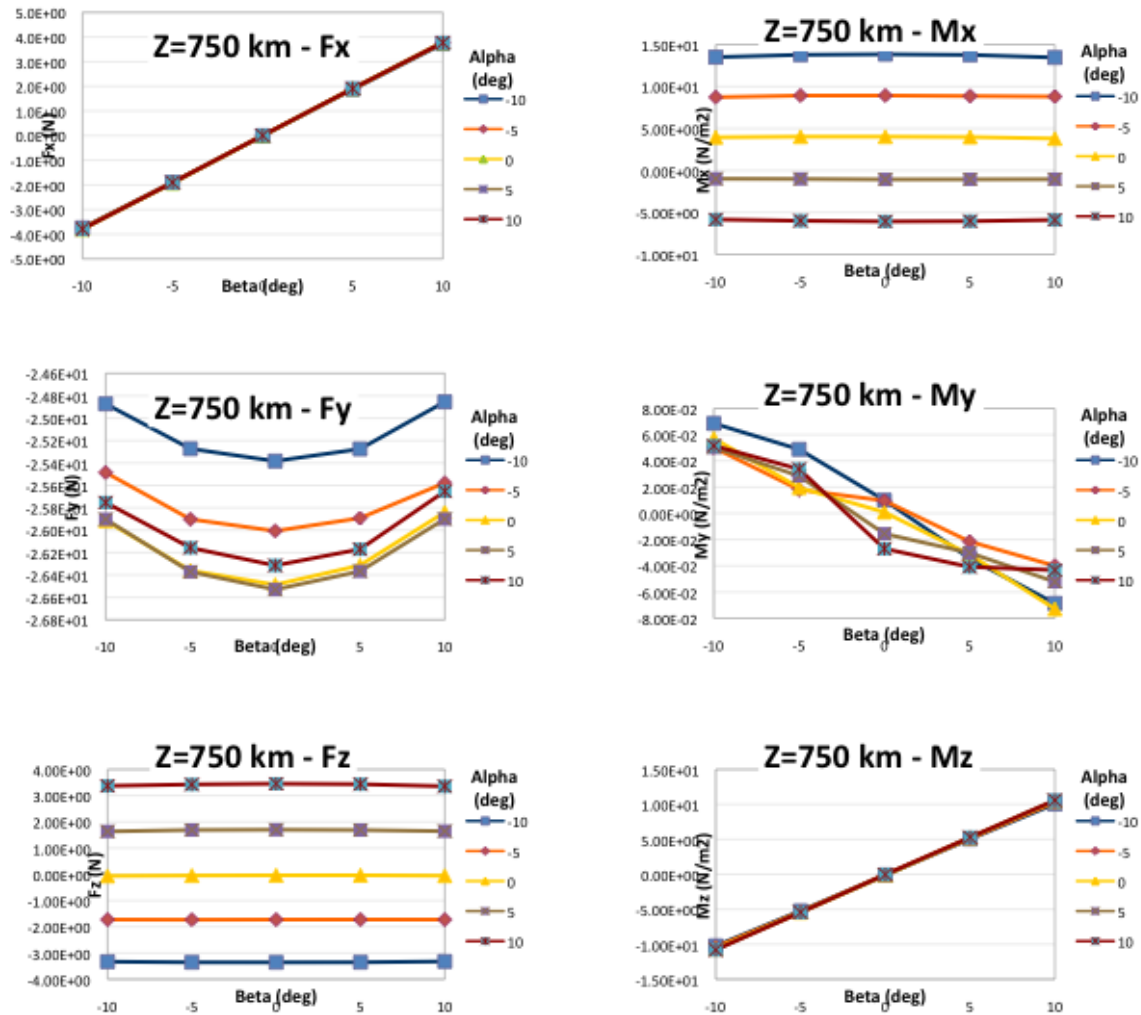


Figure C.4-3. Variation of Forces and Moments with Orientation at 750 km at Titan

### C.4.2 Aerodynamic Database

Samples of the aerodynamics database for Titan are presented in Figures C.4-3 and C.4-4 for altitudes of 750 km and 800 km, respectively, for the six forces and moments (these results are not non-dimensionalized) at various spacecraft orientations. The trends between the two altitudes are generally similar except for the rolling moment ( $M_y$ ) where the slope of the curve changes sign somewhere between 750 km and 800 km.



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
148 of 286

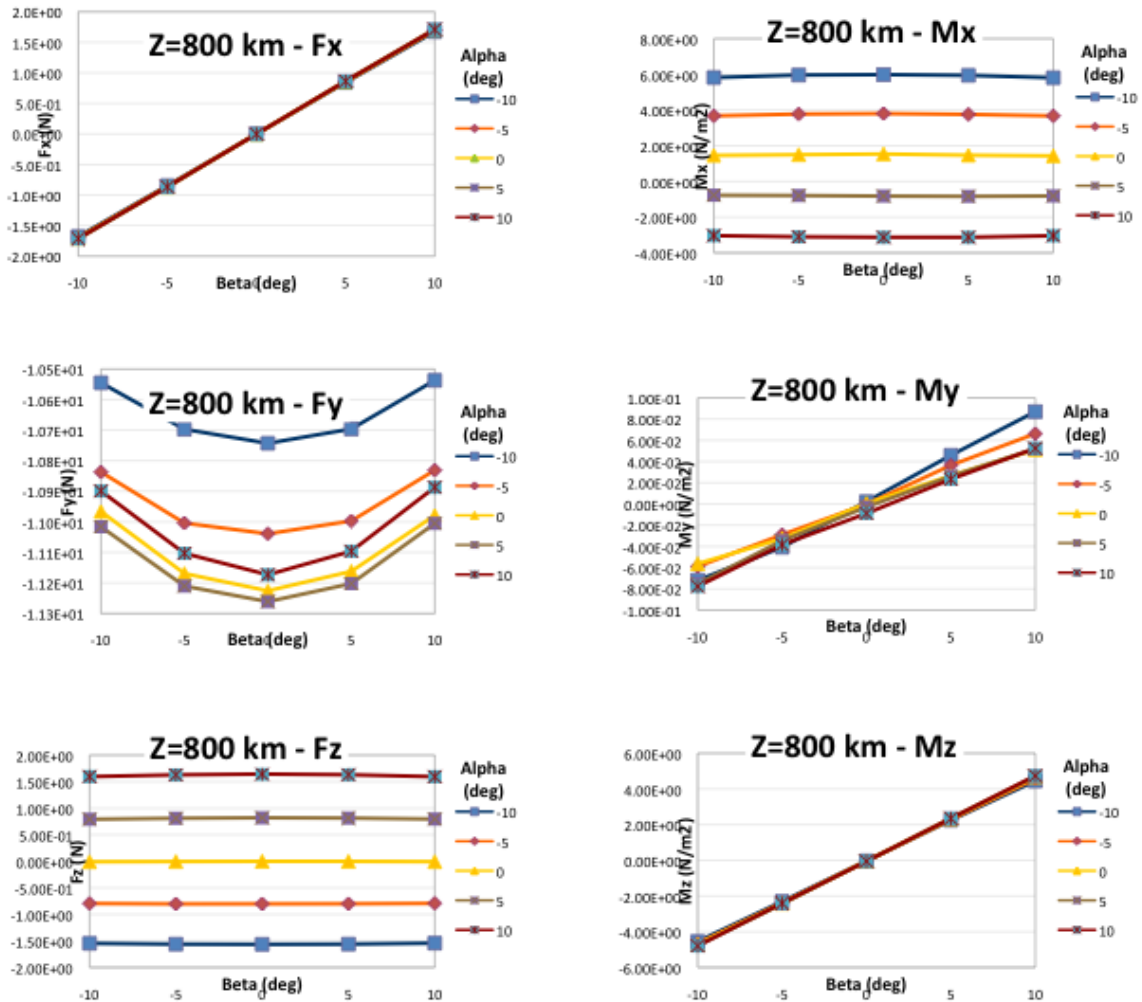



Figure C.4-4. Variation of Forces and Moments with Orientation at 800 km at Titan


## C.5 References

1. Bird, G.A., *Molecular Dynamics and the Direct Simulation of Gas Flows*, Oxford University Press, Oxford, United Kingdom, 1994.
2. Wilmoth, R.G., LeBeau, G.J., and Carlson, A.B., *DSMC Grid Methodologies for Computing Low-Density, Hypersonic Flows About Reusable Launch Vehicles*, AIAA Paper 1996-1812, 1996.
3. LeBeau, G.J., "A Parallel Implementation of the Direct Simulation Monte Carlo Method," *Computer Methods in Applied Mechanics and Engineering*, Vol. 174, No. 3-4, 1999, pp. 319-337.



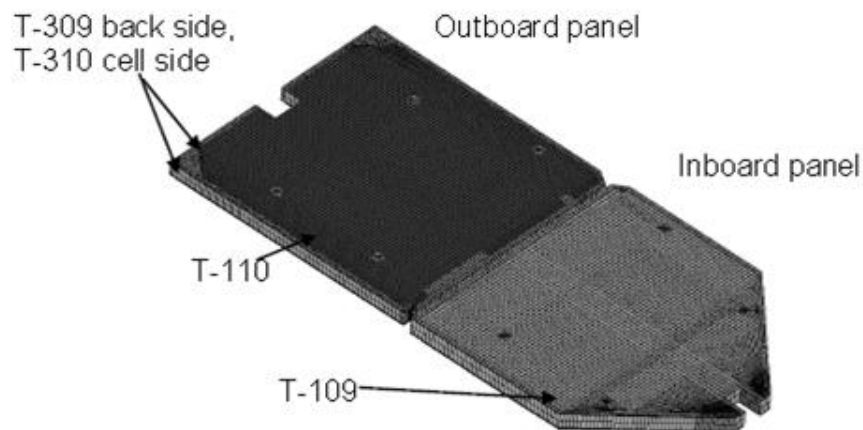
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 149 of 286	

4. LeBeau, G.J., Boyles, K.A., and Lumpkin III, F.E., *Virtual Sub-Cells for the Direct Simulation Monte Carlo Method*, AIAA Paper 2003-1031, 2003.
5. Borgnakke, C. and Larsen, P.S., “Statistical Collision Model for Monte Carlo Simulation of Polyatomic Gas Mixture,” *Journal of Computational Physics*, Vol. 18, No. 4, 1975, pp. 405–420.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 150 of 286	

## Appendix D. Thermal Modeling (Supplement to Section 7.3.2.4)

A high-fidelity thermal model, originally developed in MSC Software Patran™ ref. [i] and Thermal Desktop® ref. [ii] for MRO AB operations ref. [iii], was modified to develop the response surface equations for this AA simulation. Originally, Thermal Desktop® was used to compute the view factors to space and the solar heating. The Patran™ model was used to compute the temperatures during the drag pass, utilizing the view factor and solar heating data from Thermal Desktop®, and the aerodynamic heating from the DSMC code as boundary conditions. The Thermal Team assessed the effect of reflected heating only on the solar panel. They made no direct assessment of the heating reflected from the solar panel to other spacecraft surfaces. In the DSMC calculations, particles reflected from the solar panel surfaces that impinge on other spacecraft surfaces are included in the calculated  $C_h$  value for that surface. Vice versa, particles reflected from other spacecraft surfaces impinging on the solar panel are included in the calculated incident  $C_h$  value. The original high-fidelity Patran™ thermal model was used as a starting point because the model was already correlated to flight data, ref. [iv]. One of the objectives of the AA study was to consolidate the thermal analysis models into one universal model that would compute the view factors, solar heating inputs, and solar array temperatures. To accomplish this objective, the original MRO thermal model (shown in Figure D.0-1) was converted to Thermal Desktop® and correlated to MRO flight data, ref. [v]. The results of the correlation effort compare well to flight data. An example of the correlation results is provided in Figures D.0-2 and D.0-3 for orbit pass 262.



**Figure D.0-1. Original MRO Solar Array Model and Sensor Locations**



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
151 of 286

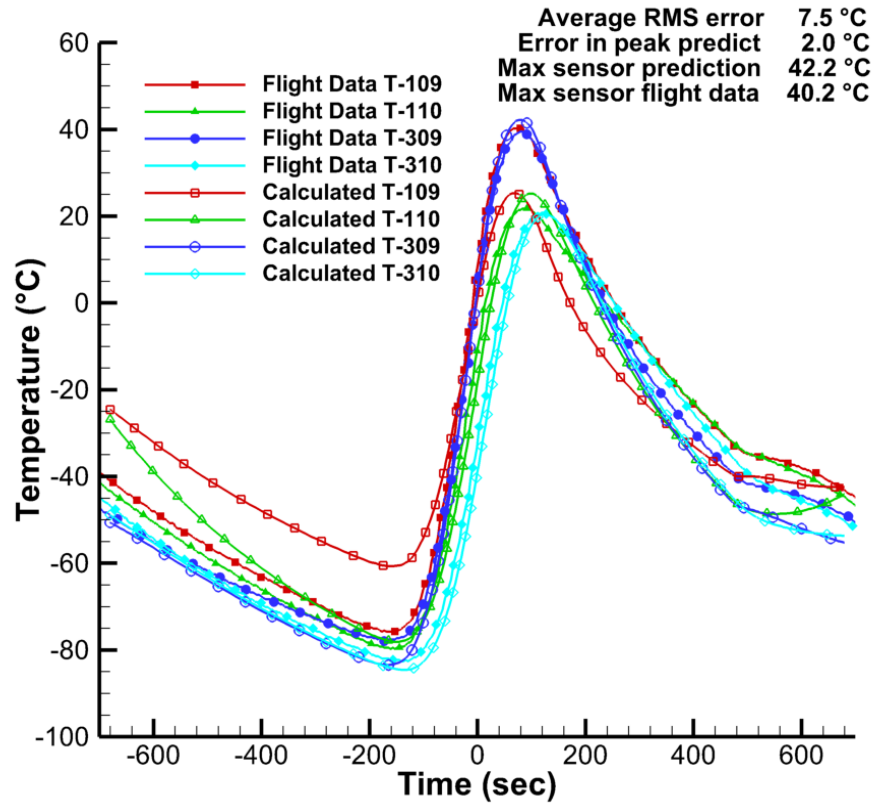

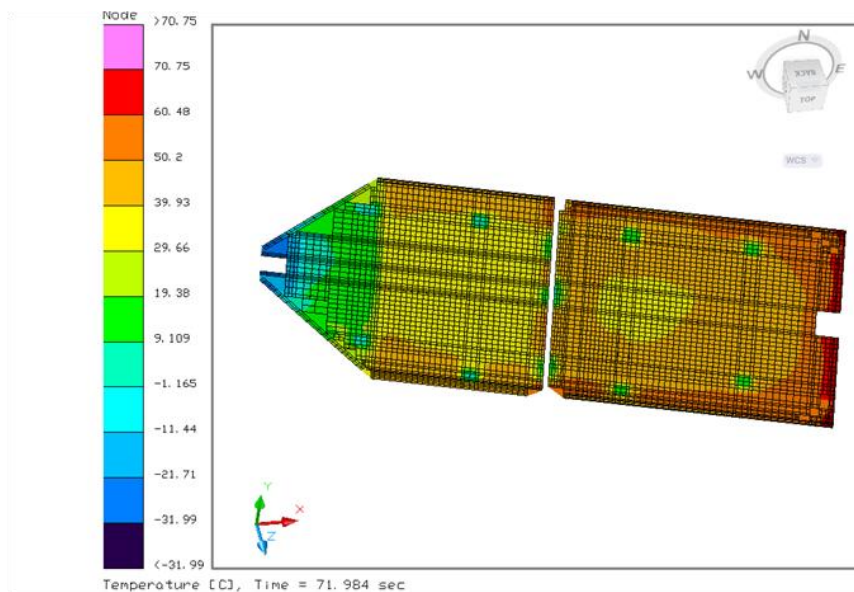


Figure D.0-2. Correlation of the Calculated Temperatures to Flight Data for Drag Pass 262 ref. [v]

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 152 of 286	




**Figure D.0-3. Peak Temperature Distribution for Drag Pass 262 ( $^{\circ}\text{C}$ )**

After the MRO model was converted to Thermal Desktop<sup>®</sup> and correlated to flight data, several modifications were made to utilize the model as a tool for AA and response surface development. First, the model was parameterized to allow variation in the key environmental, material property, and modeling variables needed for response surface development. This parameterization involved creating symbols within the model that either explicitly define the value of specific variables, or, as in most cases, establish a multiplier or bias to known values to represent the defined uncertainty of the variable.

The next modification of the model is made to enable autonomous running of multiple analyses in parametric mode with multiple variables, where the user can select a desired number of variables and change the values between a defined upper and lower limit. Currently, Thermal Desktop<sup>®</sup> has no design of experiment (DOE) capabilities; the code only has the built-in ability to run in parametric mode while varying a single variable. For response surface equation development of the MRO model, it is necessary to vary between 12 and 15 parameters. Therefore, custom logic and operation blocks are added to the Thermal Desktop<sup>®</sup> model that allows multiple cases to be run with variation of a user-defined number of variables. Additionally, these logic blocks allow specification of the total number of cases to run as well as the nominal, the high, and the low values of each variable.

The logic block provides the ability to input a matrix of numbers that define the values of each parameter for each run. For a DOE, this matrix would be N by M elements, where N represents the number of cases in the study and M represents the number of variables being investigated. The values in the matrix consist of either a 0 or  $\pm 1$ . In the case of the MRO model, 0 indicates the nominal value of the variable used in the study and  $\pm 1$  indicates that the  $\pm 3 \sigma$  value is used.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 153 of 286	

The variables are coded to range between -1 and +1 so that they are all on the same scale. This matrix is then input to an array data block, within the Thermal Desktop<sup>®</sup> logic manager. While this approach limits the user to only the nominal, high, and low values, minimal effort would be required to populate this matrix with any values between -1 and +1, based on either a uniform or Gaussian distribution, and the variable set according to the corresponding value, thus allowing the user to run Monte Carlo analyses, but that aspect is beyond the scope of this study.


## D.1 Design of Experiments, Sensitivity Study, and Response Surface Development

For an AA mission, it is impractical (from a time perspective), given current onboard spacecraft computer technology, to run a high-fidelity thermal model onboard the spacecraft. For AA, the spacecraft must be able to compute the temperatures within seconds, or minutes at the most. One solution to satisfy this calculation speed requirement is to develop a response surface model for the temperatures, which is derived from the high-fidelity thermal model. A response surface model is typically a polynomial equation that can be used to determine how a given response is affected by a set of quantitative independent variables or factors over a specified range. In the case of a high-fidelity thermal model the response is the temperature at a discrete point. The general form of the response surface equation representing the thermal response of the spacecraft solar arrays is given in Eq. (D.1-1), ref [vi].

$$T_m = b_0 + \sum_{i=1}^n b_i x_i + \sum_{i=1}^n b_{ii} x_i^2 + \sum_{i=1}^{n-1} \sum_{j=i+1}^n b_{ij} x_i x_j + \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^n b_{ijk} x_i x_j x_k \quad \text{Eq. (D.1-1)}$$

Eq. (D.1-1) captures the main effects (first- and second-order interactions) and non-linearities with the quadratic terms and third-order interaction terms. Main effects are how the response of the system changes as a single factor changes. Interactions occur when the effect of one factor on the response depends on the level of another factor, ref. [vii].


Without a priori knowledge of how the temperatures (calculated via a thermal analysis of a complex system) will respond to variations and uncertainty in the input parameters, analysts are forced to include every variable they can think of in the development of a response surface representation of the thermal analysis. One way to generate the data necessary to create a response surface is to perform a DOE. A DOE is a systematic way of varying the design variables so that the data obtained can be analyzed to yield valid and objective conclusions, ref [vii]. In the case of the thermal analysis for AA, the objective is to create a response surface model of the high-fidelity thermal model. As the number of variables (or factors, as they are called in statistics) increases, the number of runs required for the DOE and thus, required to

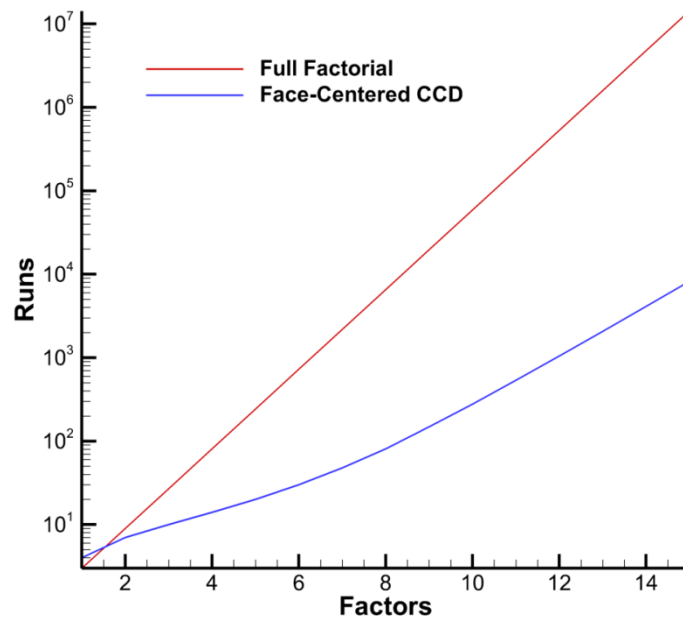
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 154 of 286	

define the response surface increases dramatically. For example, in a full factorial design, which is a DOE that includes all possible combinations of the factors, if there are three levels for each factor and ten factors, then the number of required runs of the thermal analysis model would be 59,049, or  $l^k$ , where  $l$  is the number of levels and  $k$  is the number of factors. A level is defined as a discrete value for a particular factor, hence three levels represents three discrete values for a factor. Typically, when three levels are used the minimum, maximum, and midpoint values are used.

There are other types of DOEs that reduce the number of runs, but the trade off is that not every combination of the factors is represented. A face-centered central composite design (CCD) for example is one type of DOE that reduces the number of runs. A face-centered CCD is made up of three parts: center points, axial points, and fractional factorial points. For the same example of 10 factors at three levels, if a face-centered CCD is chosen with two center points and a  $\frac{1}{4}$  fractional factorial contribution, the number of runs required of the thermal model would be reduced to 278. The variation in the number of required runs as a function of the number of analysis variables for a full factorial design and a face-centered CCD are compared in Figure D.1-1.

The trends in Figure D.1-1 indicate that the number of factors being used to create the response surface should be minimized to reduce the number of required runs of the thermal model. In practical terms, if the thermal model takes 2 hours for one run, the 10-factor face-centered CCD requiring 278 runs would take over 23 days running on a single computer to generate the data required to create the response surface. For AA, updates to the thermal response surface may be required so minimizing the number of required runs, and thus, the time necessary for an update is essential. Additionally, reducing the number of factors reduces the amount of data that needs to be passed back and forth and maintained within the AA simulation software.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 155 of 286	



*Figure D.1-1. Comparison of Required Runs for Different DOEs*

To minimize the number of factors, a sensitivity study can be performed to determine which initially selected factors are significant contributors to the solar array temperature response. Creating a screening DOE is a way to examine which of the factors’ main effects and which interactions are important. A screening DOE is similar to a CCD, except that a screening DOE does not include axial points; may or may not include center points; and the fraction factorial portion is much, much smaller. If a factor is deemed insignificant, it does not mean that the particular factor contributes nothing to the response; it just means that the particular factor’s variation is insignificant.

For this study, the MRO spacecraft is used to simulate AA around both Mars and Venus. The thermal model described in Section 7.1.4.1 is used for both the Mars and Venus mission scenarios. The only differences in the model come from the external heating environments. At Mars, the solar heating input is relatively low and the effect of solar occultation on the initial temperatures is large. The atmospheric density and corresponding aerodynamic heating encountered during the drag pass are relatively low, but due to the low initial temperatures prior to the drag pass, only the aerodynamic heating dominates the thermal response during the drag pass. At Venus, the solar heating inputs are relatively high and the effect of solar occultation in lowering the initial temperatures is lessened. The density and corresponding aerodynamic heating are relatively high and combined with the solar heating; both dominate the thermal response during the drag pass. The differences in the corresponding thermal response for both

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		156 of 286	

mission scenarios necessitate that a screening sensitivity study be performed for each mission scenario.


Starting with the initial list of factors used in the MRO AB thermal response surface analysis, ref. [viii], a screening DOE was generated using the JMP<sup>®</sup> statistical software, ref. [ix]. The factors and their definitions are given in Table D.1-1. The factors can be classified into three general categories: environmental, material property, and modeling. For these 15 factors, the screening DOE only required 129 runs, 128 from the fraction factorial part and 1 center point.

The JMP<sup>®</sup> software performed an analysis of variance on the resulting temperatures calculated for each case in the DOE matrix. The statistical p-value was an indication of whether the variation in the factor contributes significantly to the analysis. P-values less than 0.05 typically indicate a significant contribution. For the Mars AA mission, the main effects for factors that had p-values greater than 0.05 are summarized in Table D.1-2. If the only concern were the main effects, all six of these factors could be eliminated from the subsequent DOE and would not be carried in the response surface equation. However, the interactions between factors must be examined. In the Mars mission scenario, interactions between all but two of the factors had p-values less than 0.05 when interacting with other factors. The only factors that could be dropped were the drag pass duration and the solar cell emissivity; therefore the face-centered CCD DOE for generating the response surface equation for the Mars mission scenario will contain 13 factors.

***Table D.1-1. MRO Thermal Analysis Variables***

Category	Factor	Abbreviation
Environmental	Drag pass duration	DP
	Density	RHO
	Heat transfer coefficient	C <sub>H</sub>
	Periapsis velocity	V
	Initial solar array temperature	IT
	Orbital heat flux	Q <sub>s</sub>
Material Property	M55J graphite emissivity	FSE
	ITJ solar cell emissivity	ITJE
	M55J graphite thermal conductivity	Fsk
	M55J graphite specific heat	FSC <sub>p</sub>
	Aluminum honeycomb core thermal conductivity	ALk
	Aluminum honeycomb core specific heat	ALC <sub>p</sub>
Modeling	Outboard solar panel mass distribution	OFM
	Solar cell layer mass distribution	MD
	Contact resistance	CR



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		157 of 286	

***Table D.1-2. Factor Screening for Mars Mission Scenario***

Factor	Abbreviation	p-value
Drag pass duration	DP	0.8100
Orbital heat flux	$Q_s$	0.5987
ITJ solar cell emissivity	ITJE	0.6443
M55J graphite thermal conductivity	FSk	0.7929
Outboard solar panel mass distribution	OFM	0.4642
Contact resistance	CR	0.7929

Since different environmental conditions are encountered for the Venus mission scenario, the screening sensitivity must be performed again. The drag pass duration was replaced by the orbital period. This new factor was used since it was deemed a better representation of the variation in the orbit geometry, which was the original intent of the drag pass duration factor. Following the same procedure as in the Mars mission scenario, an identical screening DOE was generated and the resulting data analyzed. For the Venus AA mission, the main effects for factors that had p-values greater than 0.05 are summarized in Table D.1-3.


***Table D.1-3. Factor Screening for Venus Mission Scenario***

Factor	Abbreviation	p-value
Orbital period	P	0.1097
Periapsis velocity	V	0.7999
M55J graphite specific heat	FSC <sub>p</sub>	0.5526
M55J graphite thermal conductivity	FSk	0.5232
Aluminum honeycomb core thermal conductivity	ALk	0.9832
Aluminum honeycomb core specific heat	ALC <sub>p</sub>	0.5684
Solar cell layer mass distribution	MD	0.5291
Outboard solar panel mass distribution	OFM	0.5496
Contact resistance	CR	0.5081

For Venus, some of the factors that are found to be insignificant are the same as in the Mars mission scenario; however, there are others that are insignificant for Venus but were significant for Mars, and vice-versa. The difference arises due to how different the missions are in terms of their environment and underscores the need to repeat the screening study for every mission scenario. Both scenarios illustrate the need to examine the interaction between factors. It was found that all but two factors had significant interactions with other factors. For Venus, the periapsis velocity and the contact resistance are dropped; hence the face-centered CCD DOE for generating the response surface equation for the Venus mission scenario will contain 13 factors.

## **D.2 Response Surface Goodness of Fit Determination**

A face-centered CCD with 13 factors was generated using the JMP statistical software. The CCD had 26 axial points, 10 center points, and 128 points from the fractional factorial

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 158 of 286	

contribution. JMP<sup>®</sup> automatically reduces the fraction used to compute the fractional factorial contribution as the number of factors increases; in this case the fraction was 1/64th. The temperatures calculated for each of the 164 total runs for both Mars and Venus were analyzed using JMP<sup>®</sup> where a least squares fit was constructed using the stepwise regression option in JMP<sup>®</sup>. The result of the regression is a quadratic equation, one unique to the Mars mission scenario and one unique to the Venus mission scenario. The coefficient of determination or R<sup>2</sup> adjusted value was measured and used to determine how well the assumed functional form of the response measures the variability of the supplied data. In this case, the R<sup>2</sup> adjusted value measured how well the quadratic response surface represented the variability in the temperatures generated by the DOE cases. In the Mars mission scenario, the resulting response surface equation had an R<sup>2</sup> adjusted value of 0.9948. For the Venus mission scenario, the R<sup>2</sup> adjusted value was 0.9991. An R<sup>2</sup> adjusted value greater than 0.9 was desirable but not sufficient to determine the goodness of fit of the response surface.

To get a clear picture of how well the response surface equation is fitting the response data from the DOE runs, a plot of the actual versus predicted values, a plot of the residual versus predicted values, and the model fit distributions must be examined. The actual versus predicted plot shows the temperatures calculated by the thermal model for the cases described in the DOE plotted against the temperatures calculated by the quadratic response surface equation and is given in Figure D.2-1 for the Mars mission scenario and Figure D.2-2 for the Venus mission scenario. Note that the temperatures for the Venus mission scenario are unrealistically high. The reason the temperatures were unrealistically high is due to the fact that the solar heating was almost 4.5 times higher at Venus as compared to Mars, in addition to a higher aerodynamic heating. Furthermore, the MRO spacecraft was not designed to aerobrake at Venus and hence, the thermal response was not consistent with a spacecraft specifically designed for Venus AB. For the AA simulation at Venus (for demonstration purposes) the maximum temperature obtained from the thermal analysis was scaled to match the maximum temperature calculated for a proposed Venus AB spacecraft; a spacecraft which had a more robust thermal design and had solar panels tailored to minimize the aerodynamic heating.



# NASA Engineering and Safety Center Technical Assessment Report

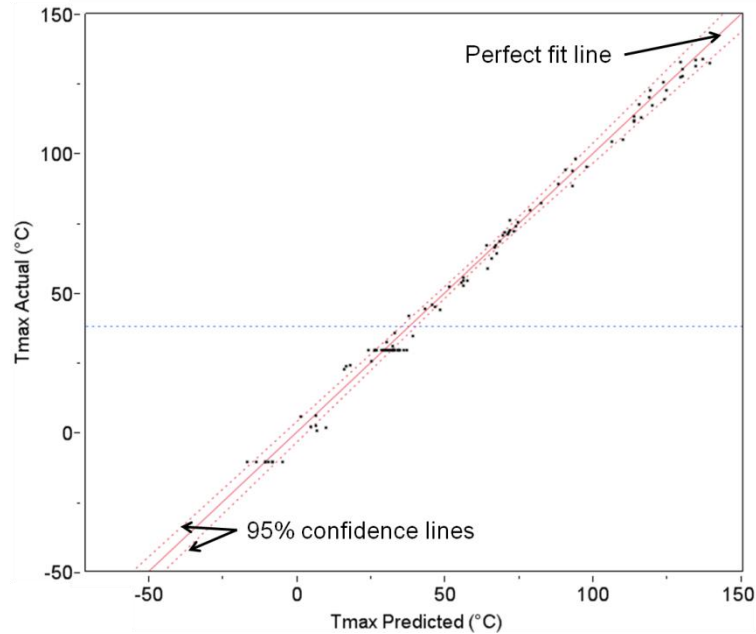
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

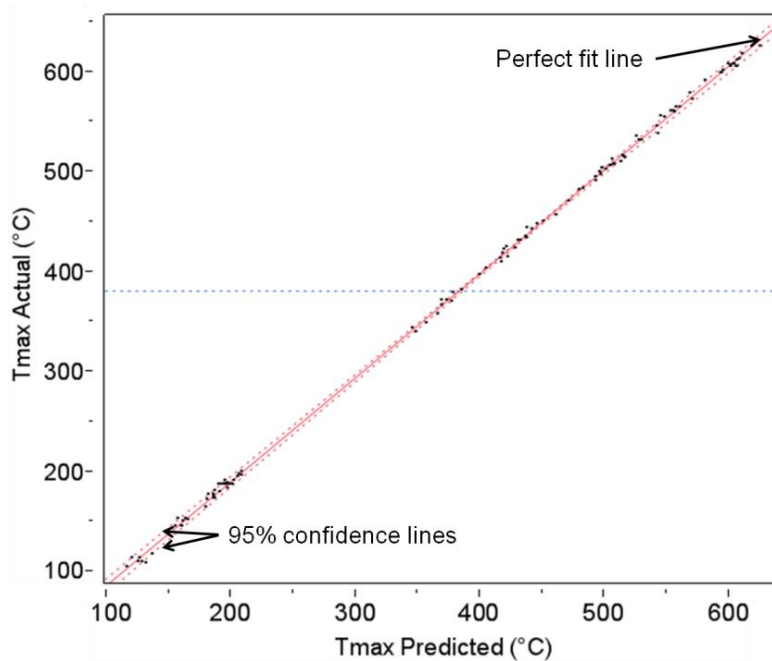
Title:

## Autonomous Aerobraking (Phase 1)


Page #:  
159 of 286



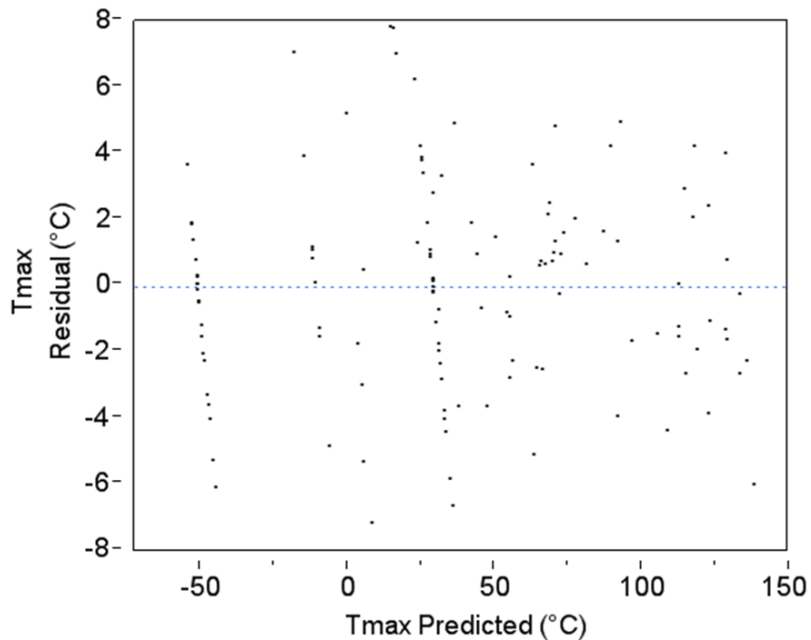
*Figure D.2-1. Mars Mission Scenario Actual versus Predicted Temperatures*




*Figure D.2-2. Venus Mission Scenario Actual versus Predicted Temperatures*

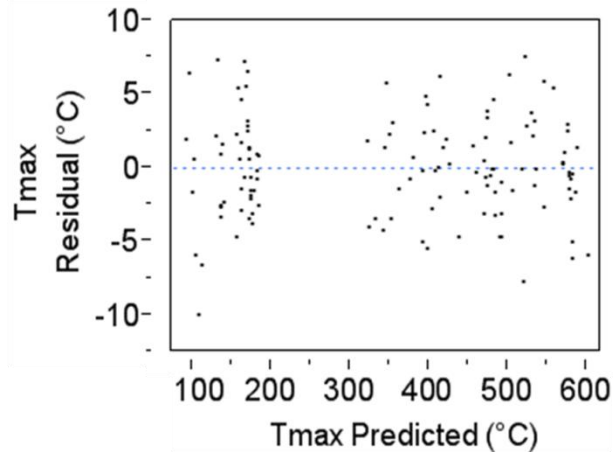
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 160 of 286	

The centerlines of the plots in Figures D.2-1 and D.2-2 represent a perfect fit of the data; the plots show that the data points lie close to the center line, which indicates a good fit. The residual is the error in the fitted model and is the difference between the temperature calculated by the thermal model and the temperature calculated by the response surface equation. The residual for the maximum solar panel temperature versus the predicted maximum temperature is plotted in Figure D.2-3 for the Mars mission scenario and Figure D.2-4 for the Venus mission scenario.



***Figure D.2-3. Mars Mission Scenario Maximum Solar Panel Temperature Residual versus Predicted Maximum Temperature***


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 161 of 286	

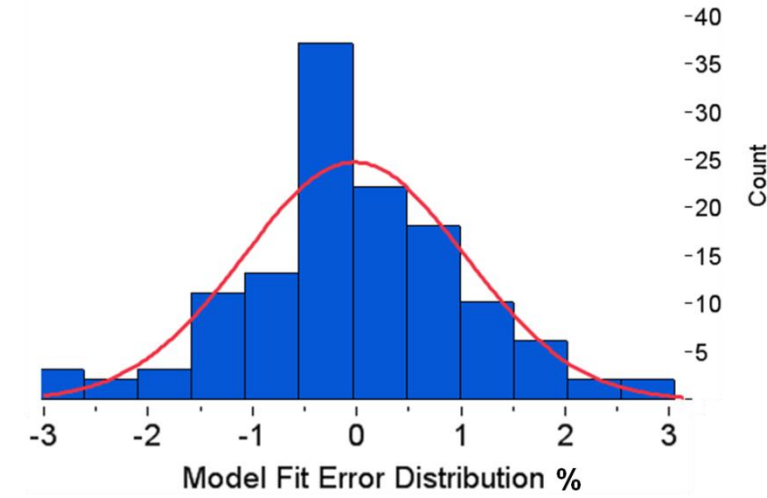


**Figure D.2-4. Venus Mission Scenario Maximum Solar Panel Temperature Residual versus Predicted Maximum Temperature**

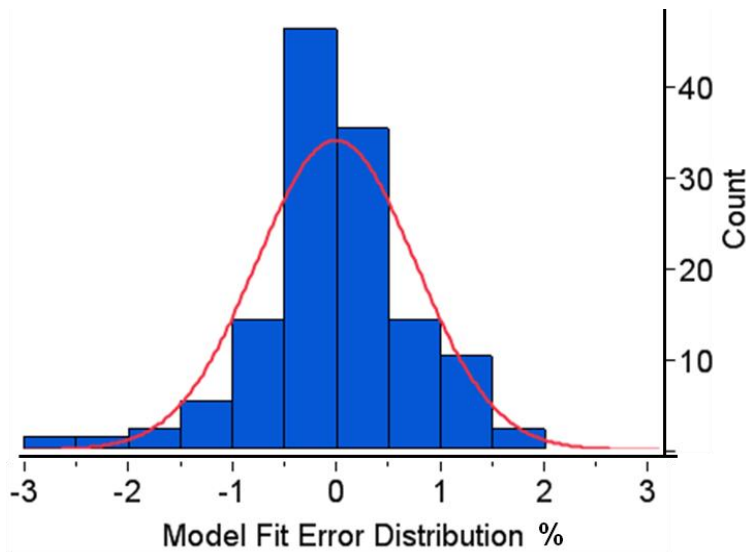
In general, the data points are randomly scattered in Figures D.2-3 and D.2-4, indicating a good fit of the temperature data. However, there are two areas on Figure D.2-3 and one on Figure D.2-4 where the data points are clustered together; this clustering indicates that one of the factors may be dominating the response. For AB, the peak temperatures are highly influenced by the peak density, which is the primary reason for this clustering. One way to alleviate the occurrence of clustering is to break the density into smaller intervals and develop a different response surface equation for each interval as in ref. [viii]. For simplicity in implementing the response surface equations into the AA simulation, a goal is to try to have a single response surface equation. As a result of the goodness of fit analysis, it is recommended that the density be broken into three ranges and three separate response surface equations used.

One final check of the goodness of fit is to examine the model fit and model representation error distributions. Both model error distributions should approximate a normal distribution with a mean around zero and a standard deviation less  $\leq 1.0$ . The model fit error is how well the response surface fits the temperature data in the DOE. The model fit error distribution for the maximum temperature for the Mars mission scenario is plotted in Figure D.2-5 and for the Venus mission scenario is plotted in Figure D.2-6. The model fit error distribution for the Mars response surface equation is approximately normal and has a mean of 0.0158 and a standard deviation of 1.0359. The standard deviation is greater than 1.0, but is sufficiently close to 1.0 to conclude that the model is accurately fitting the DOE temperature data.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 162 of 286	




*Figure D.2-5. Mars Mission Scenario Model Fit Error Distribution*




*Figure D.2-6. Venus Mission Scenario Model Fit Error Distribution*

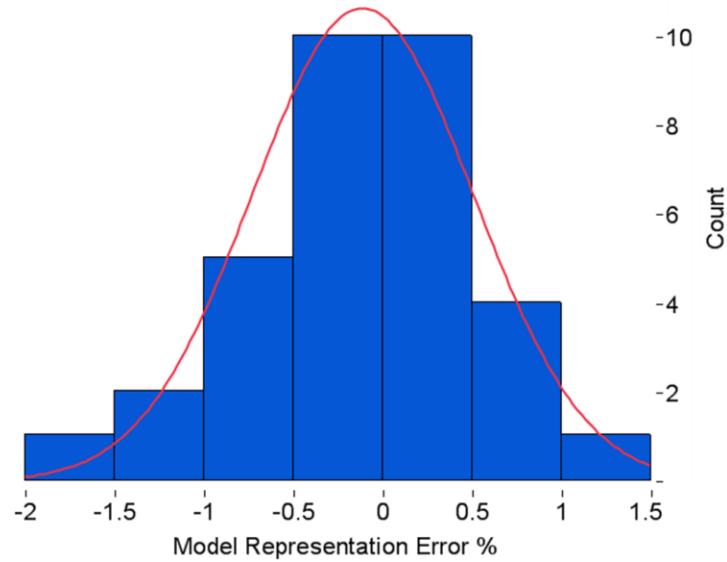
The model fit error distribution for the Venus response surface equation is approximately normal and has a mean of -0.0085 and a standard deviation of 0.7616. Both the mean and standard deviation fall within the desired range, therefore it can be concluded that the model is accurately fitting the DOE temperature data.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 163 of 286	

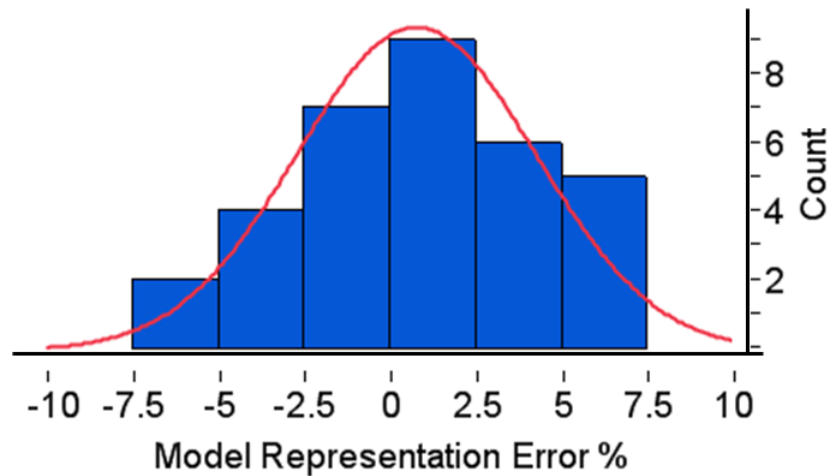
The model representation error is how well the response surface fits temperatures calculated by the thermal model for points other than those on the DOE. For the Mars mission scenario, the model representation error for the maximum temperature is plotted in Figure D.2-7. For the Venus mission scenario, the model fit distribution is plotted in Figure D.2-8. The model representation error distribution for the Mars response surface equation is approximately normal with a mean of -0.1103 and a standard deviation of 0.6177. Hence, it can be concluded that the response surface equation is an accurate representation of the high-fidelity thermal model.

The model representation error distribution for the Venus response surface equation is approximately normal and has a mean of 0.778, but a standard deviation of 3.48. The response surface equation for Venus is modeling the high-fidelity thermal model accurately, but there is a lot of room for improvement. Referring back to Figure D.2-4, there is a region on the plot between 205–355 °C where it appears that there is no temperature response for the runs made from the DOE. The model representation error distribution is found by randomly setting the model factors and calculating the temperatures using both the high-fidelity thermal model; this is due to an error made while constructing the response surface. Unfortunately, the error was not discovered in time to correct for this report. The orbital period was a new variable introduced for the Venus mission scenario and replaced drag pass duration as a way to better track how the variation in orbit geometry affects the thermal response. The error was made by using the early 20-hour orbit period and resulting orbit geometry and only varying this orbit by  $\pm 1.0$  hours. The way it should have been varied was to select the 20-hour orbit as the +1 point in the DOE table, selecting a short period ~2-hour orbit as the -1 point in the DOE table and an orbit somewhere in the middle of the range ~11-hour orbit as the 0 point in the DOE table. After discovering the error, the response surface equation was regenerated eliminating the orbital period from the equation. Some fidelity was lost in eliminating the orbital period variation but the response surface equation is still accurate enough for the purposes of demonstrating the temperature corridor AB strategy as shown by the goodness of fit tests. Correcting this error and breaking the density range into three separate response surface equations will be the first thing accomplished for Phase 2.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 164 of 286	




*Figure D.2-7. Mars Mission Scenario Model Representation Error Distribution*



*Figure D.2-8. Venus Mission Scenario Model Representation Error Distribution*

The model fit and model representation errors are accounted for in the response surface equation when the temperature calculation is made from within the AA simulation to provide a conservative temperature. Another error is added as a bias to the temperature calculated by the response surface. This error is present because the high-fidelity thermal model will typically not be correlated to the AB flight temperature data. This error is typically unknown until the first couple of drag passes are made and the flight temperatures and predicted temperatures compared.



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 165 of 286	

Therefore, a short calibration period is required but this can be accomplished during walk in which makes up the first initial orbits where the spacecraft's periapsis is gradually lowered into the AB altitude corridor. One important aspect of response surface modeling that must be emphasized is that the response surface equations are only valid over the range for which they are defined. It must be stressed that even a small amount of extrapolation in any factor included in the equation can produce invalid results.

### D.3 AA Simulation Software

A generic response-surface equation thermal analysis (GRETA) computer program was written for use in the AADS. There are two versions, one written as a standalone program which includes the ability to run Monte Carlo simulations, the other for use directly with AADS which does not have a Monte Carlo simulation. AADS accesses the GRETA routines via an external function call. This architecture is beneficial in that the response surface equation coefficients or GRETA routines can be updated independently of AADS. The main feature of GRETA is that GRETA will accept any number of variables and hence any number of response surface equation coefficients so long as the program follows the form of Eq. (D.1-1). GRETA will allow the user to modify any set of factors and thus calculate a new response. Additionally, GRETA allows the user to input a value for the response and calculate the value of one specific factor, holding all others constant. For AA, the ability to calculate the value of a factor is crucial. For AA the response is the temperature and the factor which needs to be determined is the atmospheric density. During the AA simulation a temperature within the temperature corridor is sent by AADS to GRETA and the density is calculated. Hence, the temperature can be used to control the spacecraft during AB. Using the temperature represents a major step forward since the temperature is measured directly onboard the spacecraft and can be used to determine what temperature is input to GRETA for the next orbit pass. The temperature and corresponding density for the Mars Mission simulation is shown in Figures D.3-1 and D.3-2.



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

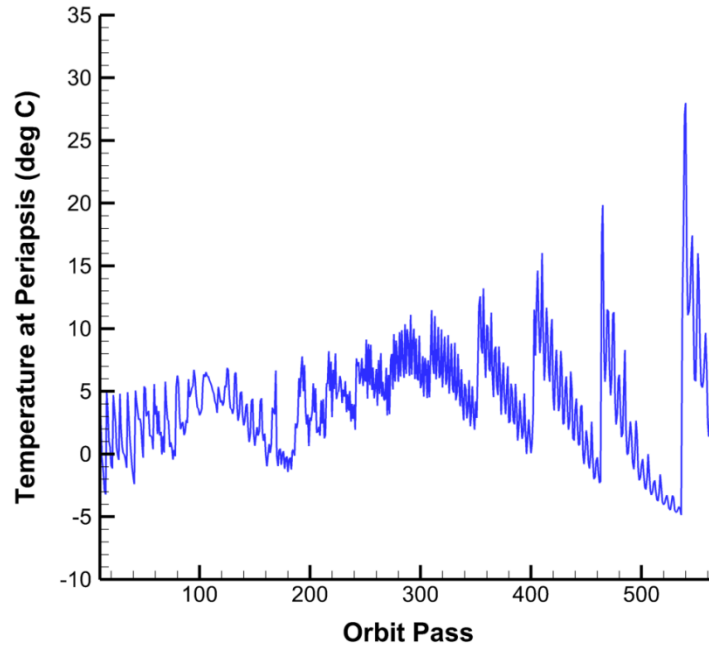
Version:  
**1.0**

Title:

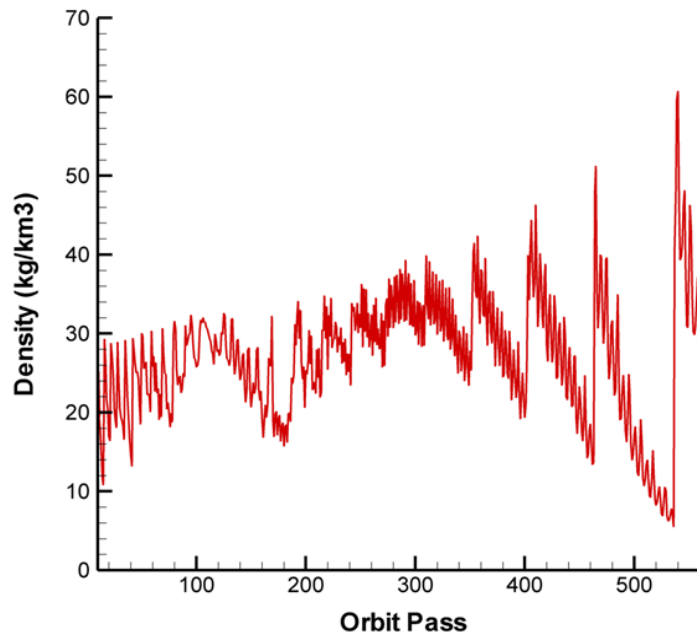
## **Autonomous Aerobraking (Phase 1)**

Page #:


166 of 286



*Figure D.3-1. Periapsis Temperature for a Mars Mission Scenario*



*Figure D.3-2. Periapsis Density for a Mars Mission Scenario*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 167 of 286	

## D.4 References

<sup>i</sup>*MSC/PATRAN User Manual*, MacNeal-Schwendler Corporation, Version 2010, February 2010.

<sup>ii</sup>*Thermal Desktop User Manual*, Cullimore and Ring Technologies, Inc., Version 5.3, January 2010.

<sup>iii</sup>Dec, John A., Gasbarre, Joseph F., and Amundsen, Ruth M., “Thermal Modeling of the Mars Reconnaissance Orbiter’s Solar Panel and Instruments During Aerobraking,” 07ICES-64, *37th International Conference On Environmental Systems*, Chicago, Illinois, July 2007.

<sup>iv</sup>Amundsen, Ruth M., Dec, John A., Gasbarre, Joseph F., “Thermal Model Correlation for Mars Reconnaissance Orbiter,” 07ICES-17, *37th International Conference on Environmental Systems*, Chicago, Illinois, 2007.


<sup>v</sup>Amundsen, Ruth M., “Aeroheating Mapping to Thermal Model for Autonomous Aerobraking Capability,” *22<sup>nd</sup> Annual Thermal Fluids and Analysis Workshop*, Newport News, Virginia, 2011.

<sup>vi</sup>Breyfogle, F. W., *Implementing Six Sigma: Smarter Solutions Using Statistical Methods*, 2<sup>nd</sup> Ed., John Wiley & Sons, Inc., Hoboken, New Jersey, 2003.

<sup>vii</sup>*NIST/SEMATECH e-Handbook of Statistical Methods*, <http://www.itl.nist.gov/div898/handbook/>, 2011.

<sup>viii</sup>Dec, J. A., *Probabilistic Thermal Analysis During Mars Reconnaissance Orbiter Aerobraking*, AIAA Paper 2007-1214, January 2007.

<sup>ix</sup>JMP<sup>®</sup>, Version 8. SAS Institute Inc., Cary, North Carolina, 1989-2011.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 168 of 286	

## Appendix E. Mars-GRAM 2010 (Supplement to Section 7.3.1.3)

### E.1 Mars-GRAM 2010

The Mars-GRAM is an engineering-level atmospheric model widely used for diverse mission applications. Applications include systems design, performance analysis, and operations planning for AB; entry, descent, and landing (EDL); and aerocapture. Mars-GRAM's perturbation modeling capability is commonly used, in a Monte-Carlo mode, to perform high-fidelity engineering end-to-end simulations for EDL<sup>1</sup>. Mars-GRAM 2005 has been validated<sup>2</sup> against Radio Science data and both nadir and limb data from the TES<sup>3</sup>.


There are several traditional Mars-GRAM options for representing the mean atmosphere along entry corridors. The first option is mapping Year 0, with user-controlled dust optical depth and Mars-GRAM data interpolated from ARC's Mars Global Circulation Model (MGCM)<sup>4</sup> results driven by selected values of globally uniform dust optical depth. The second is the auxiliary profile option in which the user can read and use any auxiliary profile of temperature and density versus altitude in the mapping Year 0 option. In exercising the auxiliary profile Mars-GRAM option, the values from the auxiliary profile replace data from the original MGCM databases. Examples of auxiliary profiles include data from TES (nadir or limb) observations or Mars mesoscale model output at a particular location and time. The final option is mapping Years 1 and 2, with Mars-GRAM data coming from MGCM results driven by the observed TES dust optical depth during TES Years 1 and 2. From the surface to 80 km altitude, Mars-GRAM is based on the ARC MGCM. Above 80 km, Mars-GRAM is based on the University of Michigan Mars Thermospheric Global Circulation Model (MTGCM)<sup>5</sup>. Mars-GRAM and MGCM use surface topography from MGS Mars Orbiting Laser Altimeter (MOLA), with altitudes referenced to the MOLA constant potential surface (areoid).

Mars-GRAM standard inputs are geographic position and time. The user can adjust the optical depth of the uniformly mixed background dust level, add a seasonal dust optical depth, set the dust particle diameter and density, and provide the starting Ls, position, duration, intensity, and radius of a dust storm. Mars-GRAM outputs include density, temperature, pressure, winds, and selected atmospheric constituents. Three Mars-GRAM parameters allow standard deviations of Mars-GRAM perturbations to be adjusted: *rpscale* can be used to scale density perturbations up or down, *rwscale* can be used to scale wind perturbations, and *wlscale* can be used to adjust wavelengths (spectral range) of the perturbations.

#### E.1.1 References

<sup>1</sup>Striepe S. A. *et al.*, (2002) *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Abstract # 2002-4412.

<sup>2</sup>Justus C. G. *et al.*, (2005) "Mars Aerocapture and Validation of Mars-GRAM with TES Data," *53rd JANNAF Propulsion Meeting*.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 169 of 286	

<sup>3</sup>Smith M. D. (2004) *Icarus*, 167, 148-165.

<sup>4</sup>Haberle, R. M., Pollack, J. B., Barnes, J. R., *et al.*, (1993) “Mars Atmospheric Dynamics as Simulated by the NASA Ames General Circulation Model 1. The Zonal-Mean Circulation,” *Journal of Geophysical Research*, Vol. 98, No. E2, pp. 3093-3123.

<sup>5</sup>Bougher, S.W., *et al.*, (1990) “The Mars Thermosphere: 2. General Circulation with Coupled Dynamics and Composition,” *Journal of Geophysical Research*, Vol. 95, No. B9, pp. 14,811-14,827.

## E.2 Mars-GRAM 2010 Adjustment Factors

### E.2.1 Adjustment Factor Requirements

The adjustment factors generated by this process had to satisfy the gas law:  $p = \rho RT$  as well as the hydrostatic relation:  $dp/dz = -\rho g$ . If  $T$  is assumed to be unchanged and both  $p$  and  $\rho$  are adjusted by a common factor ( $F$ ), both relations are preserved. The adjustment factors  $[F(z, Lat, L_s)]$  were expressed as a function of height ( $z$ ), latitude ( $Lat$ ), and areocentric solar longitude ( $L_s$ ). This adjustment factor ( $F$ ), is applied to the daily mean ARC MGCM density and pressure (0–80 km) and University of Michigan MTGCM density and pressure (above 80 km). The pressure scale height ( $RT/g$ ) is unchanged by this process. However, since the pressure has been changed by the adjustment factor, the height of the 1.26 nbar pressure level, referred to as  $ZF$  in Mars-GRAM, has been changed.

The daily mean MGCM or MTGCM density,  $DTA0$ , and the daily mean MGCM or MTGCM pressure,  $PTA0$ , depend on height ( $z$ ), latitude ( $Lat$ ), solar longitude ( $L_s$ ), dust amount ( $\tau$ ), and solar activity parameter ( $F10$ ). The adjusted values of  $DTA0'$  and  $PTA0'$  are computed from the adjustment factor ( $F$ ) using the following equations:

$$DTA0' = DTA0 * F(z, Lat, L_s) \quad \text{Eq. (E.2-1)}$$


$$PTA0' = PTA0 * F(z, Lat, L_s) \quad \text{Eq. (E.2-2)}$$

where the adjustment factor ( $F$ ) has been determined.

Adjustment factor ( $F$ ) is used to adjust  $ZF$  by the relation:

$$ZF' = ZF + H \ln(F) \quad \text{Eq. (E.2-3)}$$

where  $H$  is local pressure scale height.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		170 of 286	

## E.2.2 Development of MTGCM Factors

The Mars-GRAM density and pressure need to be consistent at 80 km, where the transition from MGCM to MTGCM data occurs. Thus, the assumption was made that  $F(80, \text{Lat}, L_s)$  for the MTGCM data had to be the same as the adjustment factor at 80 km for the MGCM data. After adjustment factors  $F(80, \text{Lat}, L_s)$  were determined from the MGCM analysis, they were used to determine MTGCM adjustment factors by use of the following equation:

$$F(z, \text{Lat}, L_s) = F(80, \text{Lat}, L_s) * (1 + A\zeta + B\zeta^2) \quad \text{Eq. (E.2-4)}$$

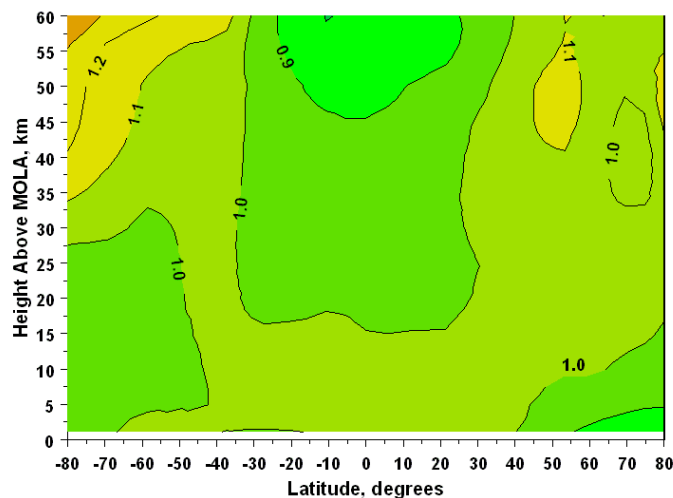
where the height parameter  $\zeta = (z - 80)$  and the coefficients A and B depend on Lat and  $L_s$ .

Final adjustment factors  $F(z, \text{Lat}, L_s)$  for MTGCM data were implemented into Mars-GRAM and a validation run comparing Mars-GRAM 2010 versus MGS, Mars Odyssey, and MRO AB data from the Planetary Data System (PDS) was completed. Any residual variation of AB density about mean values that became apparent during this process was used to update the height dependence of Mars-GRAM perturbation standard deviations.


## E.3 Improvement in Mars-GRAM 2010 Results

### E.3.1 Improvement of Mars-GRAM 2010 at Lower Altitudes

Application of adjustment factors for the ARC MGCM data yields improved comparisons between Mars-GRAM and TES limb data, as shown by density ratios (Mars-GRAM/TES Limb) given in Figure E.3-1. Prior to adjustment these density ratios were as low as 0.65 near 60 km.



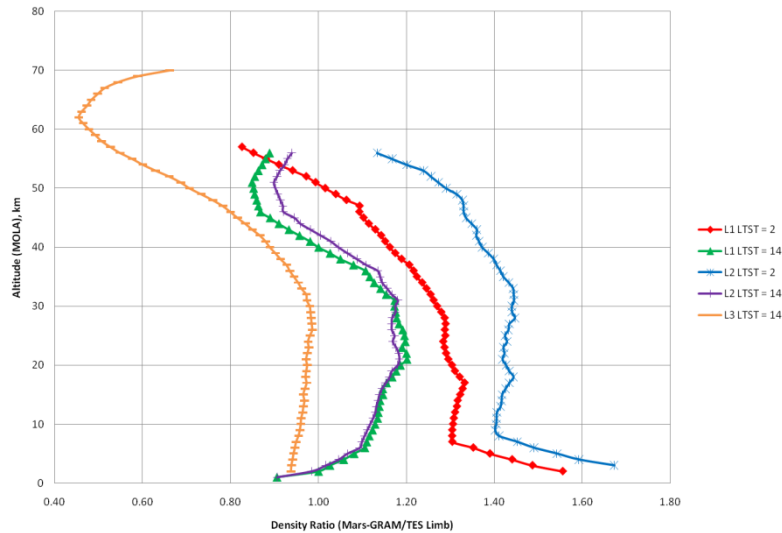
*Figure E.3-1. Latitude-Height Contours of Density Ratio (Mars-GRAM/TES Limb) after Application of MGCM Adjustment Factors*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 171 of 286	


Mars-GRAM 2005 and Mars-GRAM 2010 Map Year = 0 results have been compared for three locations at Local True Solar Time (LTST) 2 and 14.

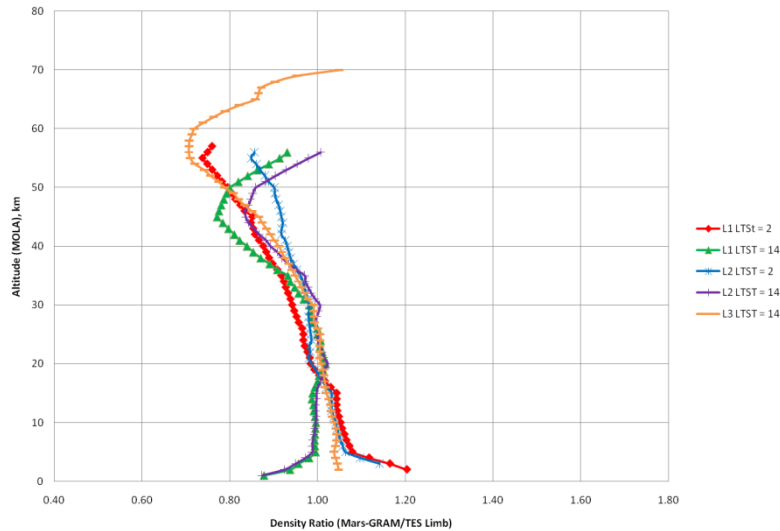
- Location 1 (L1) = 22.5° S, 180° E,  $L_s = 90 \pm 5$ ,  $\tau = .11$
- Location 2 (L2) = 22.5° S, 180° E,  $L_s = 75 \pm 5$ ,  $\tau = .12$
- Location 3 (L3) = 2.5° N, 180° E,  $L_s = 210 \pm 5$ ,  $\tau = 2.65$  \*Dust Storm case\*

Figure E.3-2 provides the density ratios of Mars-GRAM to TES for Mars-GRAM 2005. As Figure E.3-3 shows, the application of the adjustment factor in Mars-GRAM 2010 results in ratios of approximately 1 at lower altitudes.



**Figure E.3-2. Density Ratio (Mars-GRAM/TES) for Mars-GRAM 2005**

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 172 of 286	



**Figure E.3-3. Density Ratio (Mars-GRAM/TES) for Mars-GRAM 2010**


At the higher altitudes, Mars-GRAM 2010 results have corrected the effect of the underestimated dust aloft in the MGCM. At location 3, the Mars-GRAM 2010 density ratio has shifted closer to 1. This demonstrates that the addition of adjustment factors to Mars-GRAM 2010 has improved the results for the Map Year = 0 cases for large tau values.

### **E.3.2 Improvement of Mars-GRAM 2010 at Aerobraking Altitudes**

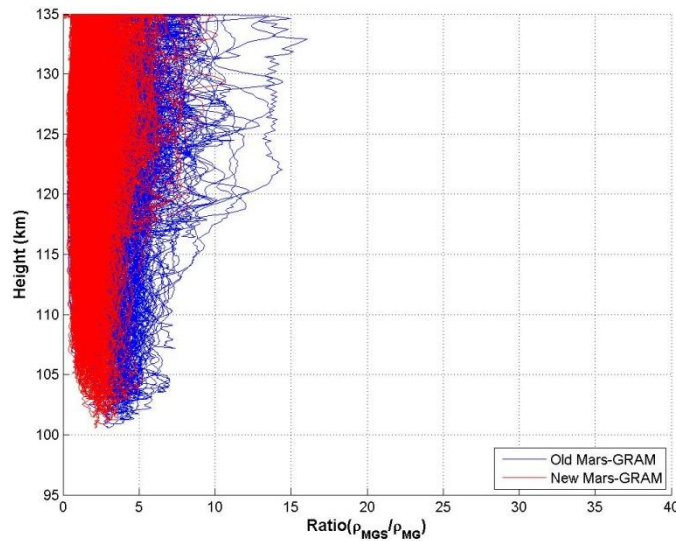
Mars-GRAM modeled data output has improved at AB altitudes by adding University of Michigan MTGCM adjustment factors which included height parameters and thermosphere coefficients. Improvement has been quantified by examining all of the profile data density ratios for each PDS orbiter. The 99<sup>th</sup> percentile profile shows the most extreme cases of ratio values while eliminating outliers that do not contribute to the standard profile. Density ratios for the old and updated Mars-GRAM versions will be shown versus height and latitude globally for Mars; these results will show the variability in certain regions on the red planet. All of these results will show that the updated Mars-GRAM is producing more realistic results, which will assist in future AA procedures.

All of the density ratios from the PDS profile datasets are shown in Figures E.3-4 through E.3-6. Each of these figures shows the density ratio of the PDS density to the Mars-GRAM output density versus height, with the blue lines representing the old Mars-GRAM output and the red lines showing the updated Mars-GRAM 2010 output using the thermosphere coefficients. Each one of the datasets showed an improvement with the ratio values for the latest version of Mars-GRAM. The MGS/Mars-GRAM density ratio originally was an average 2.6 with a maximum value of 16.1, but the updated Mars-GRAM 2010 ratio data averaged 1.8 with a maximum value of 10.7. The initial MRO/Mars-GRAM density ratio reached a maximum of 10.0 and averaged 2.0, whereas the new ratio only reached a maximum of 3.6 and averaged 0.9—close to the

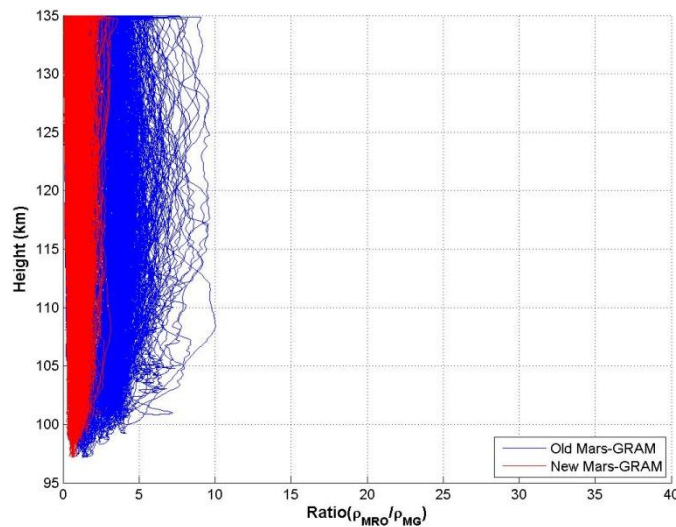


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 173 of 286	


optimal 1.0 ratio. The Mars Odyssey /Mars-GRAM ratio exceeded all of the other ratios with a maximum ratio of 39 but had an average of 3.9, which means that there were several outlying profiles that skewed the average profile. However, the newly modeled Mars Odyssey/Mars-GRAM 2010 ratio only reached a maximum of 8.2 with an average of 0.99. As these results show, the updated Mars-GRAM 2010 with MTGCM adjustment factors including thermosphere coefficients greatly improves the results of the modeled data when compared to observed data.

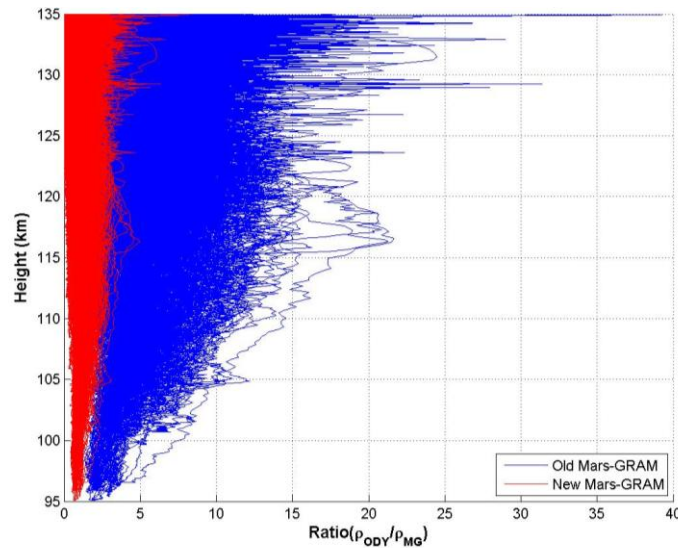


*Figure E.3-4. Density Ratios of MGS Data to the New and Old Mars-GRAM Output Data*




*Figure E.3-5. Density Ratios of MRO Data to the New and Old Mars-GRAM Output Data*

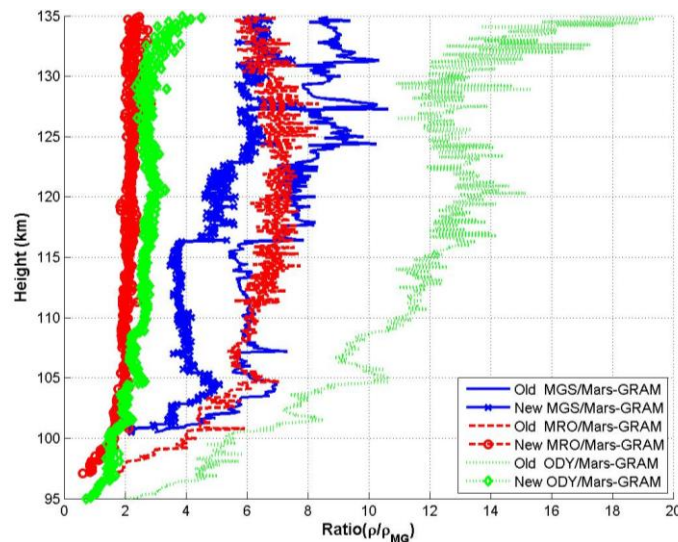
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 174 of 286	



**Figure E.3-6. Density Ratios of Mars Odyssey Data to the New and Old Mars-GRAM Output Data**

Taking the 99<sup>th</sup> percentile of all the density profiles illustrates the significant change the updated Mars-GRAM 2010 has on the profile density ratios. As shown in Figure E.3-7, the least amount of change was observed in the MGS data over the 99<sup>th</sup> percentile profile data, with an overall change of 2.0 units across the altitude range. The MRO data showed a significant improvement from the old version of Mars-GRAM, reducing the higher altitude ratios from 6.0 to close to the optimal value of 1.0 on the updated data. However, the greatest change in ratio values occurred with the Mars Odyssey data where the older data reached values close to 20.0, but the newer data brought the ratios down to a range between less than 1.0 to over 4.0 at the higher altitudes. All of the ratio values of the datasets improved from the old Mars-GRAM data output to the updated Mars-GRAM 2010 version; therefore, the inclusion of MTGCM adjustment factors has shown to be valid in providing more realistic output to be used in future endeavors.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 175 of 286	



**Figure E.3-7. The 99<sup>th</sup> Percentile Density Ratios of the Profile Data from MGS, MRO, and Mars Odyssey to Mars-GRAM 2010 Output Versus Height**

Although AA procedures are sensitive to density values at certain altitude levels, showing the density ratio values according to latitude is beneficial for mission planning operations. Figures E.3-8 and E.3-9 show the ratio of the observed density values to the Mars-GRAM output values for the old version and the updated Mars-GRAM 2010 version versus height and Mars latitude. Before the MTGCM adjustment factors including thermosphere coefficients were added to the Mars-GRAM code (Figure E.3-8), the ratio values were higher than the optimal value of 1.0, especially at locations toward the poles. The contour lines are tight near the poles, meaning lots of variability exists with the comparisons. In the updated plot shown in Figure E.3-9, a large area of the map is covered with the 1.0 ratio value, especially between 30°S and 15°N. Although a large discrepancy of ratio values still exists toward the poles, the variability has decreased with the inclusion of the adjustment factors. Improvement in density ratio values across latitudes can be beneficial for planning AA procedures on Mars.



# NASA Engineering and Safety Center Technical Assessment Report

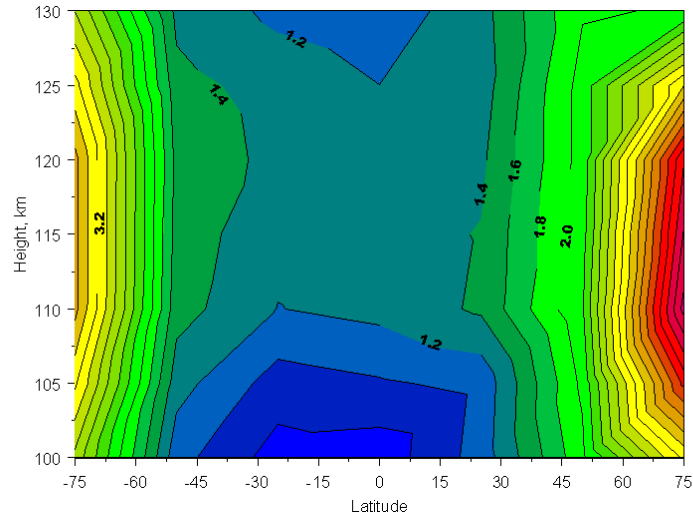
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

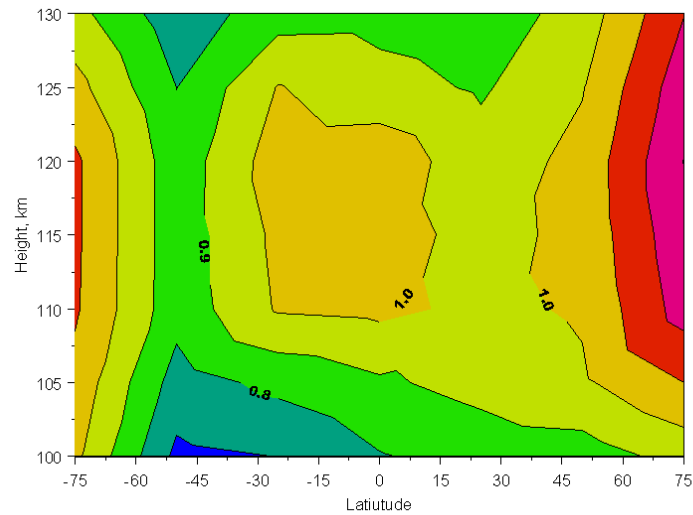
Title:

## Autonomous Aerobraking (Phase 1)


Page #:  
176 of 286



**Figure E.3-8. Contour Plots of the Ratio of Observed PDS Density Values to Mars-GRAM Output Values (before Adjustment) versus Height and Latitude**



**Figure E.3-9. Contour Plots of the Ratio of Observed PDS Density Values to Mars-GRAM 2010 Output Values (after Adjustment) versus Height and Latitude**

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 177 of 286	

## **Appendix F. Onboard Atmospheric Modeling and Prediction for AA Missions (Supplement to Section 7.3.2.1)**


### **F.1 Introduction**

The first planetary AB mission, Venus Magellan, took 70 days and over 700 passes<sup>1</sup> through the atmosphere to enhance the scientific return of the extended mission. AB was performed to reduce the orbital eccentricity after the primary science mission thereby lowering apoapsis altitude and improving the resolution of the gravity mapping. Based on the successful Magellan experience, AB became an enabling technology for recent Mars orbiting missions. These missions had AB operational phases that took about 850 orbits for MGS, 77 days and 325 orbits for Mars Odyssey, and 145 days and 420 orbits for MRO. MGS was anomalistically long due to a broken solar array which constrained the maximum dynamic pressure during an AB pass. These missions used the solar arrays as the primary drag area and consequently, except for MGS, the temperature of the solar arrays was the limiting atmosphere dependent factor in designing the AB corridor<sup>1</sup>, although other subsystems had to be considered.

Although AB has numerous benefits, there are cost and risk. The greatest costs are the large operations team and DSN coverage that have been required to maintain the AB schedule. The greatest risk<sup>2</sup> has been the inability to predict the orbit-to-orbit variability of the Martian atmosphere. One of the functions performed on an orbit-by-orbit basis is an estimation of the atmospheric density profile. These profiles were recovered using telemetric accelerometer and gyroscope data from the IMU. After the AB pass, these data were related to aerodynamic forces and then mapped into atmospheric density at 1-second intervals along the orbit. The recovered density profiles were analyzed to determine atmospheric temperature, gravity wave phenomena, orbit-to-orbit variability, longitude dependent waves, latitudinal gradients, and other information.<sup>3</sup> To predict upcoming atmospheric conditions, this information was evaluated on a day-by-day basis by a team of atmospheric scientists, the Atmospheric Advisory Group (AAG). Implementation of AA will require the development of robust, reliable, and simple methods for the estimation of atmospheric density profiles from the IMU data and the prediction of future atmospheric conditions without the human interpretation provided by the AAG.

Mars, Venus, and Titan are targets for AA missions. It is well known that the Mars atmosphere provides a challenging environment for AB because of the high orbit-to-orbit variability in atmospheric density<sup>3</sup>. An abundance of AB data provides adequate information for testing AA at Mars. High orbit-to-orbit variability has been detected near the terminator and on the night side of Venus<sup>4</sup>. Even though there are no accelerometer data for detection of small-scale variations, Pioneer Venus Orbiter (PVO) mass spectrometer data provide some insight.

Little is known about the variability of the Titan atmosphere on the temporal and spatial scales of interest for AB. However, during the Huygens descent through the atmosphere, significant wave structure was found in the density and temperature profiles in the altitude range of interest<sup>5</sup>, and

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 178 of 286	

Cassini mass spectrometer measurements during Titan flybys in the altitude range from 1,000 km to 1,600 km identified relevant vertical and horizontal wave structure in various constituents and in total density.<sup>6</sup>


The current paper presents various potential methods for representing density profiles derived from IMU data during AA, for recovering profile parameters from IMU data, and for optimal combinations of profiles for prediction. Algorithms are evaluated based on simplicity, robustness, and applicability to onboard limitations. The atmosphere of Mars is the primary focus due to the wealth of data, but Venus and Titan are discussed briefly.

## **F.2 Atmospheric Estimations during Past AB Missions**

Magellan entered orbit in August 1990 with an orbit eccentricity of about 0.4. After the 4th Venusian day, spanning over 7,000 orbits, the AB phase was initiated and reduced the eccentricity to about 0.03 after 70 days and over 700 AB passes. During AB, the active side of the solar array was turned away from the free stream direction to minimize the temperature encountered by the cells, adhesives, and structure. Maximum solar array (SA) temperature was the limiting factor constraining the rate of AB<sup>1</sup>. Pre-AB studies provided a relationship between free stream dynamic pressure and maximum SA temperature, but atmospheric density was required to determine dynamic pressure. The method for determining atmospheric density during each Magellan pass relied on Doppler radio tracking data. Pre-pass and post-pass tracking data were processed in a single orbit determination (OD) that included density at a specified altitude as a solution parameter. This approach provides continuity of the equations of motion across the unobserved AB pass. To provide a unique solution for density, a model for density versus altitude was used. The contemporary Venus International Reference Atmosphere (VIRA) model<sup>7</sup> provided density every 5 km and a constant scale height was used for interpolation. Density at 140-km altitude was the solution parameter in the OD process and the scale heights from the VIRA model were used to map density to other altitudes. For a hydrostatic atmosphere, this is equivalent to assuming that the temperature profile is given and the density profile is defined within a multiplicative factor.

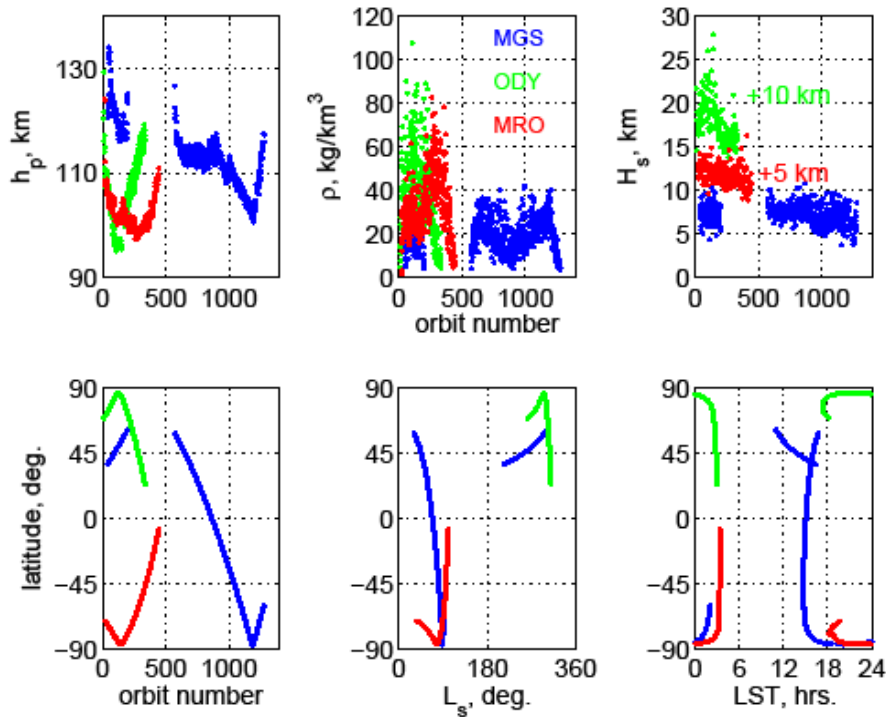
Magellan AB was so successful that AB was considered a validated technology and was enabled for the MGS mission in 1997. The MGS AB corridor was again defined in terms of the surrogate variable, free stream dynamic pressure. However, after the discovery of the broken solar array on orbits 11 through 15, the corridor criteria changed from limiting SA temperature to limiting torque on the broken SA yoke<sup>8</sup> and for the only time, the maximum dynamic pressure became the most relevant control variable.

During MGS operations, density at periapsis was estimated by two different methods. Members of the AAG used the IMU data at a one per second sample rate to model the atmospheric density profile. IMU accelerometer measurements were mapped to the vehicle center of mass using the IMU angular rate data and the resulting center of mass acceleration was converted to


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 179 of 286	

atmospheric density using a database of aerodynamic force coefficients. Density at periapsis and density scale height were extracted using a least squares solution from three data sets that included all data within 1, 1.5, and 2 scale heights of periapsis.<sup>9</sup> The “best” model was selected by visual comparison of the model and the data density profiles. Estimated scale heights were averaged over a few orbits and provided to the Navigation Team to be used for corridor control maneuver calculations and orbit determination. The Navigation Team used this scale height to estimate the density at periapsis using radio tracking data in the same way as was done for Magellan. The need for more autonomy<sup>10</sup> was recognized well before the end of the 15 months required to complete MGS AB. When adjusted for the difference between predicted and observed scale height, the AAG and navigation estimates of periapsis density were within 5 percent,  $1\sigma$ .

The periapsis altitude, latitude, density at periapsis, local solar time (LST), density scale height, and solar longitude as determined during operations, are shown in Figure F.2-1. In an idealized atmosphere, density scale height is proportional to temperature, so this variable can be thought of as the local average atmospheric temperature. The first 202 orbits of MGS were termed “phase 1,” after which there was a 6-month “hiatus” while periapsis regressed over the North Pole at a periapsis altitude near 170 km. AB “phase 2” began on orbit 573 and ended on orbit 1285 about 2 weeks after periapsis regressed over the South Pole during the winter.



**Figure F.2-1. Summary of AB Conditions for MGS, Mars Odyssey, and MRO**

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 180 of 286	

Mars Odyssey, the most aggressive AB mission to date, went to the lowest altitude and experienced the highest densities, unintentionally reaching 107 kg/km<sup>3</sup> on orbit 106. Like MGS and MRO, the science orbit required a particular LST, which meant that the AB phase had to end within a few days of the planned final day. Both Mars Odyssey and MRO used Mars-GRAM<sup>11</sup> to define the density profile and the OD process to determine the density by solving for a multiplier to be applied to the Mars-GRAM density profile. In addition, as the latitude of AB precessed toward the North Pole, it was expected that the thermospheric temperature would decrease. Instead the temperature increased dramatically as indicated by the density scale height in Figure F.2-1. The inferred temperature increase has been interpreted as a polar warming<sup>12</sup> and led to accelerometer derive density scale heights between 7 km and 14 km with an average above 10 km. The nominal atmosphere scale height was expected to be closer to 6 km and did return to that value after the latitude was south of 60°N. The difference in scale height partially led to the large density differences between Mars-GRAM and the IMU-derived densities.

Mars Odyssey tested new techniques. Though Mars Odyssey used maximum dynamic pressure to define the AB corridor, it was the first mission to have a near real time prediction of the solar array temperatures for a comparison with the measured temperatures.<sup>13</sup> Based on this comparison over a number of orbits, the AB safety margin was reduced, permitting Mars Odyssey AB to proceed at a faster rate. During this mission, the first onboard algorithm<sup>14</sup>, called the Periapsis Timing Estimator (PTE) and designed to reduce the work load of the ground flight team, was tested.


MRO had a less risky AB phase than Mars Odyssey because there were 6 months between Mars Orbit Insertion (MOI) and the time when the orbit would have the proper LST. AB was initially performed with nearly a 200 percent safety margin as opposed to the 100 percent margins used for MGS and Mars Odyssey. However, as suggested by the significant increase in density after orbit 200, MRO fell behind the time line during the early conservative approach and AB was more aggressive for the last 200 orbits. PTE was used operationally during this mission with an estimated saving of about \$1M. The operational process for atmospheric estimation was essentially the same as Mars Odyssey.

### F.3 Atmospheric Prediction Performance during Operations

During Mars AB operations, the AAG monitored the characteristics of recent AB passes to anticipate major changes in the atmosphere. The simplest variation used for modeling the density ( $\rho$ ) profile was the exponential or constant scale height (CSH) model

$$\rho(h) = \rho_p \exp\left[\frac{-(h - h_p)}{H_s}\right] \quad \text{Eq. (F.3-1)}$$




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 181 of 286	

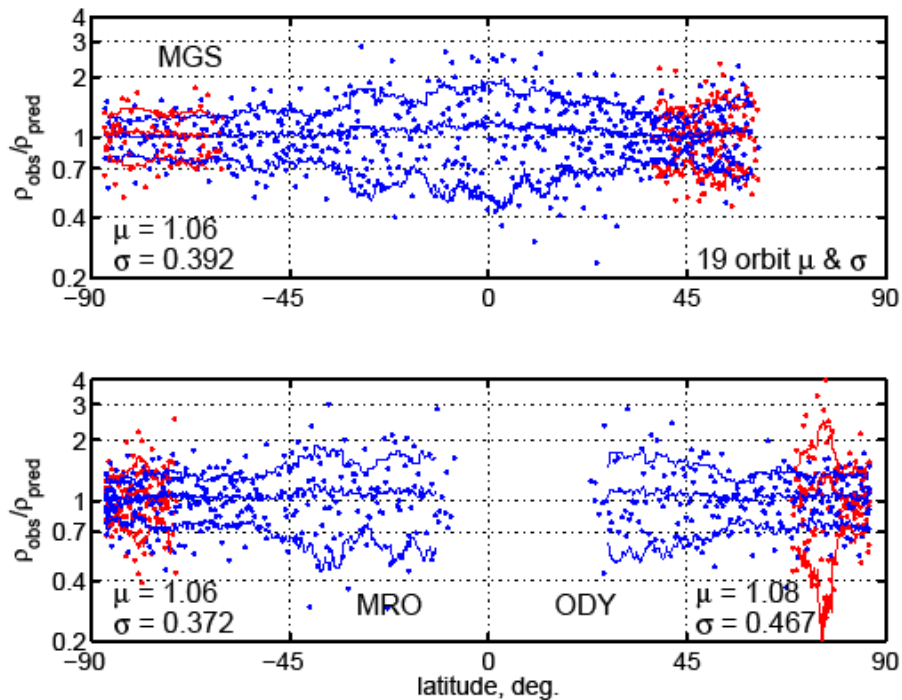
where density, as determined from IMU data<sup>9</sup>, is a function only of the altitude (h) above some reference or base altitude, here taken as periapsis.  $H_s$  is called the density scale height. Such a model results for a homogeneous, isothermal atmosphere in hydrostatic equilibrium, and the density scale height is related to the atmospheric temperature (T), the local gravity acceleration (g), and the mean molecular weight by  $H_s = kT/mg$ , where k is the Boltzmann constant. Using this as the basic model, the AAG studied density and temperature latitudinal gradients, amplitude of gravity waves, and among others, the accuracy of predicting the periapsis density for the upcoming orbit using the density and scale height from the current orbit. This latter metric was called “persistence” and is a measure of the atmospheric variability that the AB system must accommodate. The ratio of observed to predicted periapsis density for orbit n+1 is

$$\frac{\rho_{obs_{n+1}}}{\rho_{pred_{n+1}}} = \frac{\rho_{obs_{n+1}}}{\rho_{obs_n}} \exp\left[\frac{h_{n+1} - h_n}{H_{s_n}}\right] \quad \text{Eq. (F.3-2)}$$

where the altitudes are provided by the OD process and “observed” density and scale height are determined from IMU data. Orbit n is called the “base” orbit and orbit n+1 is the “predict” orbit.

Figure F.3-1 provides the persistence for all three Mars missions. The means over the entire missions are between 1.06 and 1.08, with the deviation from unity mostly being an artifact of averaging a positive ratio. Mars Odyssey has the largest 19 orbiting running mean at 1.38 and a maximum standard deviation of 1.10 (i.e., over a factor of two) variation orbit to orbit. Mission-wide standard deviations range from 37 percent for MRO to 47 percent for Mars Odyssey. The large Mars Odyssey value is perhaps due to the large variations early in the mission between 70° and 80° latitude. Except for Mars Odyssey during this time, the deviations from the means are much smaller at high latitudes than in the mid latitudes and equatorial regions. Poleward of 60° latitude, the 1σ deviations are generally between 20 percent and 30 percent. From a geometric argument, it might be expected that the deviations would become smaller near the pole since great circle distances between successive periapsis locations become shorter. The large Mars Odyssey deviations near the pole are likely due to the polar warming producing strong winds and large, asymmetric temperature variations around the pole.<sup>12, 15</sup> In the tropics, persistence is the largest for all three missions likely due to the global scale tides that appear as stationary waves.<sup>16</sup> These waves were sufficiently persistent and observable during MGS that models were developed during operations to include their influence on predicting subsequent periapsis densities and to plan orbit trim maneuvers.<sup>17</sup> Latitude-dependent empirical models were developed post flight for inclusion of such waves in Monte Carlo simulations of AB missions.<sup>18</sup> These waves appeared for brief periods during Mars Odyssey and MRO, but not with sufficient persistence to be included in operational decisions.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 182 of 286	

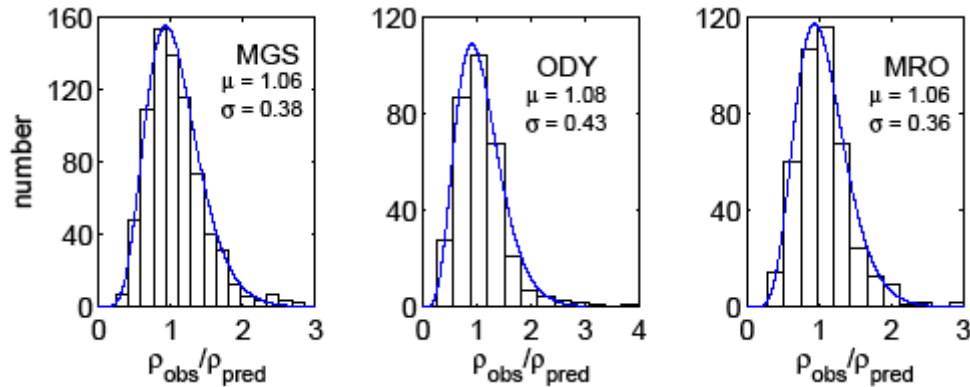


**Figure F.3-1. Persistence for MGS, Mars Odyssey, and MRO Missions. Dots are data and lines are the 19 Orbit Running Mean and Mean  $\pm 1 \sigma$ . Full Mission  $\mu$  and  $\sigma$  are shown. Dots and lines change color as periapsis regresses past the pole.**

Ignoring the latitudinal, seasonal, diurnal, and other dependencies and considering the orbit-to-orbit variability as a random process provide similar results for all three missions. It was found<sup>18</sup> that persistence can be reasonably represented by a gamma probability distribution. Maximum likelihood estimates (95 percent) of the two gamma distribution parameters for each mission result in probability density distributions shown in Figure F.3-2. The histograms are from the same ratios shown in Figure F.3-1. Within the 95 percent confidence interval, the values of  $\sigma$  and  $\mu$  are indistinguishable from each other and compare well with the simple standard deviations in Figure F.3-1.

Since underestimating density usually causes higher mission risk than underestimation, these distributions can be used to approximate the probability associated with any ratio of  $\rho_{\text{obs}}$  to  $\rho_{\text{pred}}$ . For example, for MGS, Mars Odyssey, and MRO, the probabilities that the ratio will be less than 2 are 98.5 percent, 97 percent, and 98.6 percent, respectively. These probabilities are consistent with the AB rule of thumb requiring a design safety factor of 2 for the uncertainty in density. The distributions might be used for Monte Carlo simulations of AB missions.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 183 of 286	



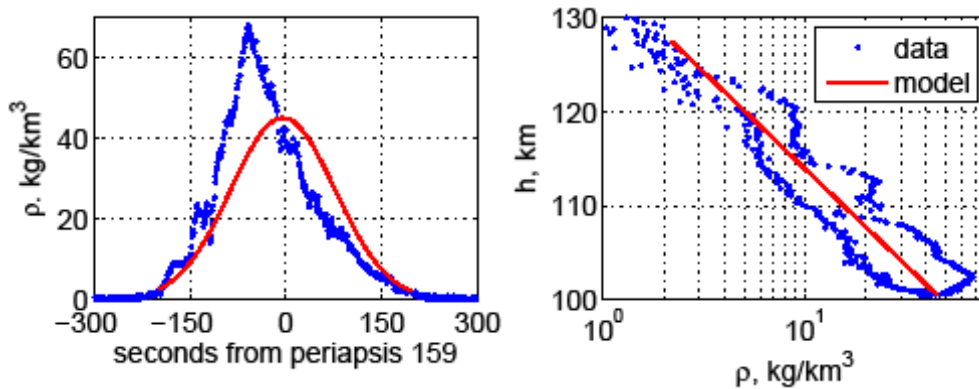
*Figure F.3-2. Gamma Probability Density Distribution Based on Maximum Likelihood Fit to Persistence Data for Three Mars AB Missions*

## F.4 Relevant Atmospheric Parameters for Aerobraking

The relevant atmospheric parameters depend on the criterion selected to define the AB corridor. If the limiting condition is related to maximum aerodynamic force or torque, then maximum dynamic pressure is likely the relevant parameter. If maximum temperature is the limiting factor for a component with rapid thermal response, maximum free stream heat flux might be the relevant parameter. If temperature is the limiting factor for a component with slow thermal response (e.g., high thermal inertia or low radiative cooling), total or integrated heat flux may be most relevant. Here thermal response time is relative to the duration of the AB pass. To calculate any of these parameters requires knowledge of some characteristic of atmospheric density along the trajectory. To predict the variation for subsequent orbits requires an atmospheric model. For this discussion, consider Figure F.4-1, which shows the recovered density versus time and versus areodetic altitude for a typical Mars AB orbit. A least squares fit to data with  $\rho > 2 \text{ kg/km}^3$  using the CSH model produced the “model” results. For this orbit, maximum density occurs 57 seconds before periapsis, a feature not captured by CSH. There is considerable asymmetry in the time profile, with density rising faster than it falls. If maximum dynamic pressure or maximum heat flux are the selected corridor criteria, then recovering the density at periapsis using the data or the model is inadequate. Further, when maximum temperature is the criterion, the shape of the heat flux as well as the total heat flux could become a consideration and only a detailed thermal analysis<sup>19</sup> can address these issues. The CSH scale height of 8.9 km, which might be used to predict density for the next orbit, represents the inbound, outbound, and mean density profiles reasonably well. Maximum density occurs 2 km above periapsis and density varies by nearly a factor of 3 within this altitude range. This gradient is likely due to a strong along track density gradient. Within this altitude range, Mars Odyssey spent about 110 seconds and traveled about 360 km along track. The high-frequency deviation from a “smoothed” density profile is generally attributed to gravity waves<sup>20</sup> and is a common feature at high latitudes. Note that accelerometer noise becomes relevant above

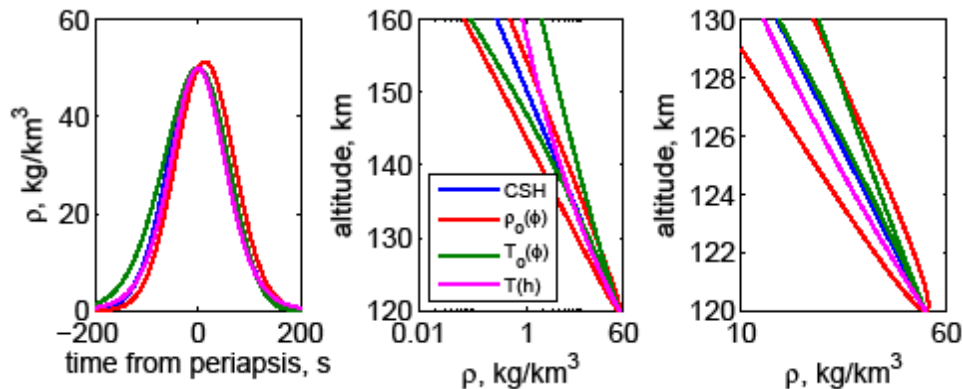
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 184 of 286	

altitudes of 125 km. Early and late in the AB pass, accelerometer data noise dominates the signal and the recovered “density” is often negative. These phases of the pass are used to determine a time linear approximation to the accelerometer bias, which is used to correct the data during the pass.<sup>9</sup>




*Figure F.4-1. Mars Odyssey Orbit 159 Atmosphere Density Inferred from Accelerometer Data*

Although the density variation is usually modeled as a function of only altitude, along track variations may dominate over the altitudinal. Large-scale variations in atmospheric properties, from those assumed for the simple CSH model, might be expected to include an along track variation ( $\phi$ ) in base density and/or base temperature, and an altitudinal variation in temperature. Examples of how such variations affect the density profiles are shown in Figure F.4-2.




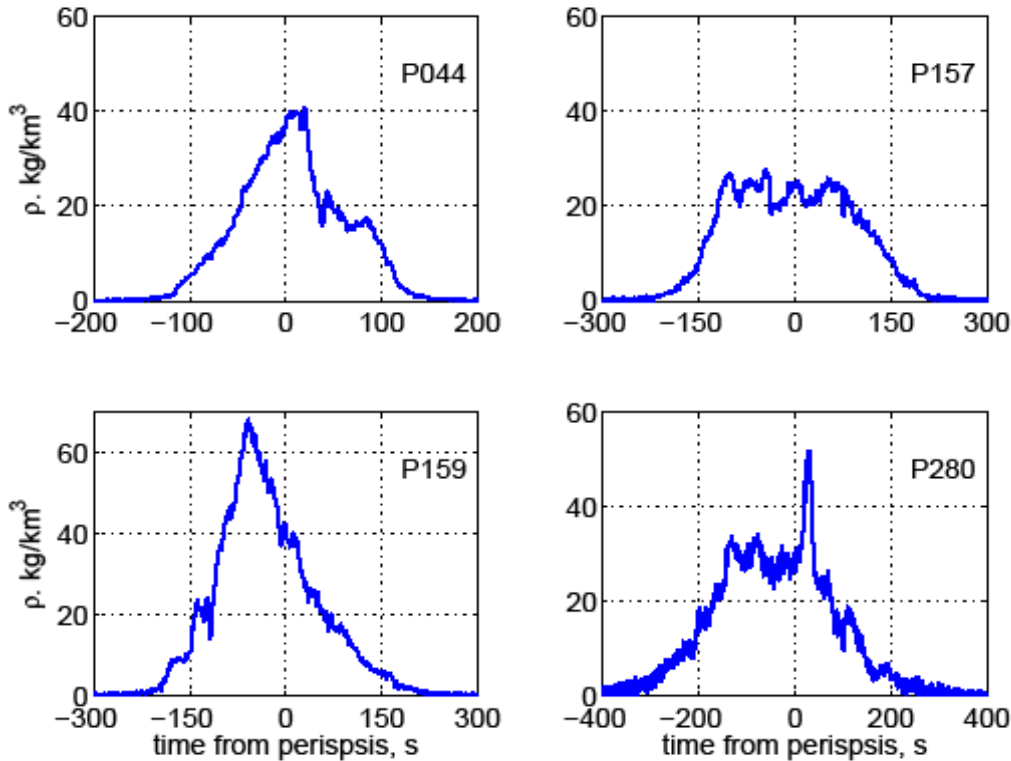
*Figure F.4-2. Effects of Along Track and Altitudinal Density and Temperature Gradients on Density Profiles*

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 185 of 286	

There are obvious deviations in the altitudinal profiles but the differences in the temporal variation are more subtle. The CSH model is of course a straight line in the two right panels and for this case has a scale height of 7 km. Assuming base density varies linearly with along track angle ( $\phi$ ) provides different inbound and outbound profiles and a substantial difference in the two densities within a kilometer or two of periapsis that is comparable to the altitudinal variation in density. An along track linear temperature variation produces a linear variation in scale height resulting again in different inbound and outbound profiles. This variation causes little density differences in the vicinity of periapsis but an increasing difference with altitude. The model with temperature increasing linearly with altitude ( $T(h)$ ) has the same inbound and outbound altitude profiles and no significant deviation in the first 10 km, but deviates significantly 20 km above periapsis and higher. One can assume that some combination of these and other effects influence every AB pass. For AA, the first issue is to quantify such effects and the second issue is to decide whether or not to include them in the atmospheric estimation process.

As examples of some of the multiplicity of effects on real AB passes, consider Figure F.4-3 which provides examples of Mars Odyssey profiles that are representative of the types of phenomena seen during all the Mars AB missions. The noticeable increase in data noise level is due to halving the sample rate on orbit 134 and again on orbit 270. The “bell-shaped” density variation with time, shown for the CSH model in Figure F.4-2, is not representative of any of these orbits. For orbit 44, the factor of two change in density over 10 seconds is not atypical for Mars. The time and altitude of maximum density are meaningless concepts for pass 157. The large asymmetry for orbit 159 results in maximum density occurring a full minute before periapsis. Generally solar power and Earth communications are lost during the AB pass. Thus, there are clear reasons to want to minimize the duration that the vehicle is in the AB orientation. The AB phase is usually designed to be centered on periapsis. With such large asymmetries, extra time may have to be allocated, or if the asymmetries are consistent from orbit-to-orbit, biasing the center of the pass away from periapsis may be desirable. The AB passes for orbits 157 and 159 are 7 hours apart in time, 2 km apart in altitude, and essentially at the same latitude, yet the profiles and the maximum density are dramatically different. These phenomena are the sort of natural orbit-to-orbit variabilities that are difficult to predict and therefore must be included as uncertainties in the design of any AB mission. They require a particularly robust design for an AA mission. It will be seen that there is some orbit-to-orbit persistence in the density, density scale height, and temporal asymmetry.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 186 of 286	



*Figure F.4-3. Four Odyssey Orbits*

Near factor of two density spikes like P280 are uncommon. They would be important if maximum density or heat flux is the consideration and not so important if total heat flux is the consideration. Even in the former case, the characteristic response time of the system will play a role. At Mars, the lack of persistence in the shape and maximum value of the density profile from orbit to orbit and the small-scale deviation are attributed to global scale longitudinal waves and vertically propagating gravity waves. The longitudinal waves during MGS have been modeled<sup>18</sup> and are attributed to non-migrating thermal tides<sup>16</sup> in the lower atmosphere that propagate to the upper atmosphere in the equatorial and mid-latitude regions. On the other hand, the source of the gravity waves is not known, but they are believed to originate in the lower atmosphere, and at high latitudes, and to propagate vertically while increasing in amplitude with subsequent “breaking” in the lower thermosphere.<sup>20</sup> Their latitudinal, seasonal, and diurnal variations of root mean square (rms) amplitude have been partially defined from previous AB data.<sup>3</sup> Whether they are significant for a particular mission depends on the criteria that limit AB. In the modeling approaches that follow, neither of these wave types will be a consideration as they are difficult to model on time scale of interest to AB.

The observed density asymmetries in time could be due to either an along track density gradient at a fixed altitude or to the areodetic altitude gradient at a constant distance from the center of the

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 187 of 286	


planet. First consider a possible density gradient. Mars Odyssey, like the other Mars AB missions, is in a near polar orbit so along track is essentially latitudinal. The polar regions are generally colder than the tropics and consequently the density scale height is smaller in polar regions. This would suggest that, for a fixed altitude, a lower density would be expected near the pole than in the tropics and strong latitudinal density gradients have been seen in all three Mars missions. On the second possible reason for a density asymmetry, periapsis is the point in the orbit that is closest to the center of mass; but, due to planetary flattening, does not usually correspond to the point of lowest areodetic altitude. Planetary flattening is defined by the reference ellipsoid which approximates the equipotential surface at the surface of the planet. The reference ellipsoid is selected to approximate such a surface by defining an equatorial radius ( $a$ ) and a flattening ( $f$ ) that give a polar radius of  $a(1-f)$ . For the Earth, the ellipsoid can be thought of as defining “mean sea level.” On solid, ocean-less planets, the ellipsoid is selected to provide an equatorial radius that approximates the physical mean radius and the flattening is usually selected to represent the equipotential defined by the central gravitational potential,  $J_2$ , and the centrifugal potential due to planetary rotation. In an idealized, isothermal atmosphere, surfaces of constant planetodetic altitude correspond to surfaces of constant pressure and density. Hence, in a real atmosphere, density should be approximately constant on surfaces of constant planetodetic altitude. Many empirical atmospheric models, as will this paper, use this surface as the reference from which altitude is measured.

**Density:** In this paper it is assumed that the AB corridor is defined in terms of variables that require a knowledge of atmosphere density. But readers should remember that density is often a surrogate for some other physical quantity that defines the limits on the execution of AB.

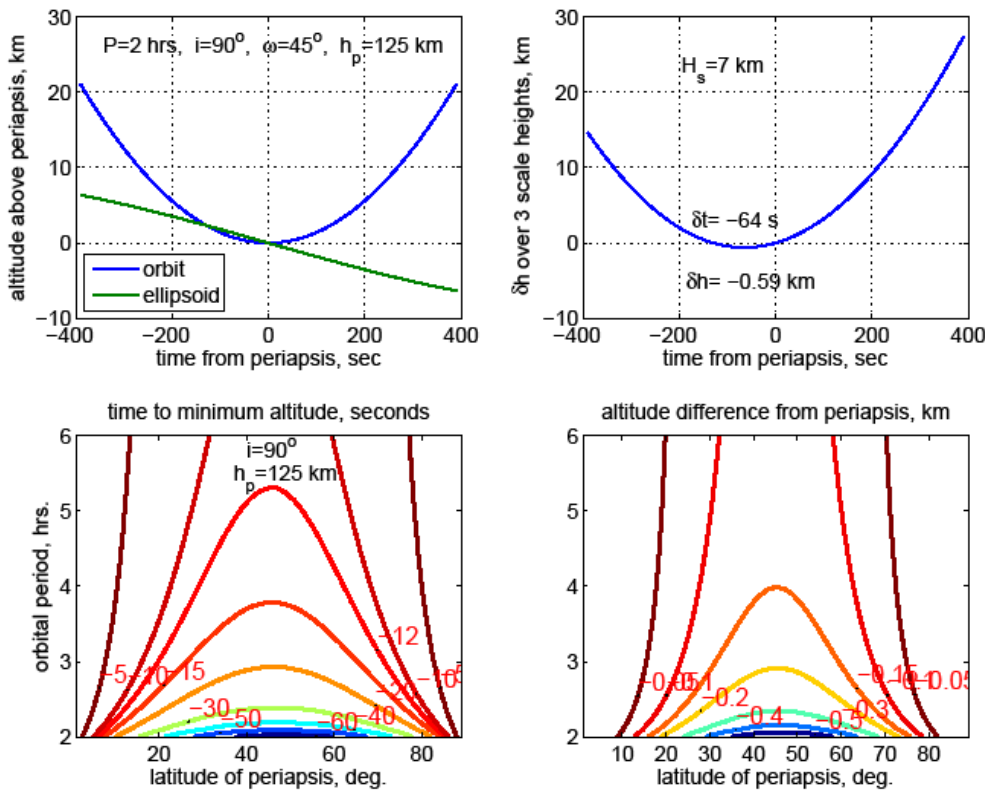
**Density Scale Height:** Density scale height plays two roles in AB. First, for maneuver calculations to stay within a density corridor, the density scale height, or equivalent, must be known to calculate the required  $\delta V$ . Second, if total heat flux or integrated density is important, the integral depends on the reference altitude density and the scale height as discussed in Section F.5.2 below.

**Asymmetry of Density Profile:** As mentioned, during AB at Venus or Mars, the vehicle would be generally turned from sun-point and would be operating on batteries. In this case it may be desirable to minimize the time in the AB orientation. If the density profile is skewed or asymmetric in time, an allowance may be made for the potential skewness. If the skewness is predictable, then it can be included in the design and the AB pass can be accordingly biased in time.

Figure F.4-4 shows the influence of planet flattening on shifting the density profile for a Mars AB mission. The upper left chart provides the variation of altitude above periapsis (blue line) along the orbit for three scale heights (21 km). The orbit parameters are given in the figure. The green line shows altitude of the reference ellipsoid relative to periapsis along the ground track. The upper right panel provides the altitude above the reference ellipsoid. The lowest areodetic altitude and highest density occur 64 seconds before periapsis. The 0.59 km difference in


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 188 of 286	

altitude would cause a 9 percent higher density than the density at periapsis. This shift would cause a least squares density estimation process, centered on periapsis, to overweight the outbound leg of the pass. The time and altitude shifts for other latitudes and orbit periods are shown in the lower two panels. The differences approach zero at the equator and pole, are maximum at mid latitudes, and decrease rapidly with orbit period. This latter effect is due to shortening of the AB pass as eccentricity increases with orbital period. Non-polar orbits will show smaller effects at every latitude. The size of the altitude difference and time shift are increased with planetary flattening, AB pass duration, and angular velocity at periapsis. This phenomenon is not an issue at Venus due to slow rotation and the nearly spherical gravity field. The Titan rotational period is 15.9 days and Saturn produces tidal bulges of less than one kilometer resulting in a flattening that is less than 1/10 of that at Mars, so the effects on Titan AB are likely ignorable.



**Figure F.4-4. Mars Reference Ellipsoid Flattening Effect on Density Profiles**



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 189 of 286	

## F.5 Potential Atmospheric Models

Here it is assumed that no preflight empirical model of the atmosphere exists that is sufficiently accurate to enable AB to be performed without using onboard data to adjust model parameters during the flight. Selection criteria for onboard atmospheric models include (1) capture the relevant characteristics of the atmosphere, (2) be robust against unexpected phenomena, (3) allow for linear estimation of the parameters, (4) permit prediction of atmospheric properties for the next AB pass, and (5) support the calculation of corridor control maneuvers. Several models are discussed and evaluated using Mars AB data. All the models assume that atmospheric density data have been derived from an onboard source (e.g., IMU accelerometer and gyro data).<sup>9</sup>

### F.5.1 Models That Require Altitude Knowledge


Empirical atmospheric models are almost always modeled with altitude as one of the parameter. An onboard Ephemeris Estimator is required to provide altitude versus ephemeris time. On the other hand, onboard density estimation is done using IMU data which has spacecraft time as the independent variable. If these two time references loose synchronization, significant errors, as will be seen in Section F.6 can be produced in the estimate of the variation of density with altitude. Such a loss of synchronization is likely not due to a clock failure, but rather an ephemeris integration error that causes associating an erroneous time with the time of periapsis passage. In anticipation of such issues, the methods studied are divided into those that depend on knowledge of altitude and those that do not.

#### F.5.1.1 Constant Scale Height

The CSH model given Equation (F.3-1) was used successfully as the fundamental model during the MGS mission. Successful use of this model, or any model using altitude, depends on an accurate representation of altitude versus time. This could be an issue for onboard ephemeris integration and is discussed in F.6. There are two disadvantages to using this form for estimation. First, the density is not linear in the estimation parameters  $\rho(h_p)$  and  $H_s$  and second, a simple least squares process will overweight residuals at the lowest altitude and nearly ignore residuals a few scale heights above the reference altitude. One approach is to use  $\log(\rho)$  as the observable and use

$$\log \rho(h) = a + b(h - h_p) \quad \text{Eq. (F.5-1)}$$

where  $a$  and  $b$  are the regression parameters. The equation is linear in  $a$  and  $b$  and the least squares method, within the linear regime, now minimizes the sum of squares of the density difference divided by the density, i.e., the fractional deviation in the density. This approach provides equal weight to high or low density data and is more suitable when scale height is

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		190 of 286	

among the estimated parameters. This model can, to a limited extent, provides asymmetric temporal variation like the left panel in Figure F.4-1, but the inbound and outbound altitude profiles will be identical so that the model is a straight line in the left panel. Equation (F.5-1) is not applicable early and late in the AB pass when accelerometer data noise produces negative density. This model is simple and captures the dominant local variations in density. It permits prediction of the atmospheric density at the next periapsis by assuming that the scale height is the same for the next orbit and that the density at periapsis can be obtained equation (F.3-1) at the next periapsis. Since the AB pass is not vertical, the two parameters in this model absorb an unknown amount of along track variation. The persistence results in Figure F.3-1 show the real world limitations to this approach.

### F.5.1.2 Constant Scale Height with Linear Time

As seen in Figure F.4-3, density profiles need not be symmetric in time from periapsis or in geodetic altitude so models should be considered that might include such asymmetries. One of the simplest models is the hybrid model (constant scale height with time (CSHT)), with constant scale height but different inbound and outbound density profiles. One approach can be obtained by adding a linear time term to get

$$\log \rho(h, t) = a + b(h - h_p) + c(t - t_p) \quad \text{Eq. (F.5-2)}$$


where the reference altitude and time are taken at periapsis. The model permits some variation in local scale height with altitude. It is unlikely that this model should be used to extrapolate beyond the data interval, since, unless  $c=0$ , the predicted density will eventually increase with altitude.

### F.5.1.3 Constant Inbound and Outbound Scale Heights

Another asymmetric three parameter hybrid model (constant scale height inbound and outbound (CSHIO)) permits different inbound and outbound scale heights and one density at periapsis given by

$$\log \rho(h) = a + b(h - h_p) \quad \text{and} \quad \log \rho(h) = a + c(h - h_p) \quad \text{Eq. (F.5-3)}$$

where the first equation is used for the inbound part of the pass and the second model is used for the outbound phase. The parameter  $a = \log(\rho_p)$ ,  $b = -1/H_{\text{sin}}$  inbound and  $c = -1/H_{\text{sout}}$  outbound.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 191 of 286	

## F.5.2 Models that Depend Only on Time

In the definitive reference on drag effects on satellite orbits,<sup>22</sup> King-Hele provides a derivation to show that the CSH altitude model is well represented by the bell shaped Gaussian density function in time. That equation is derived below in a more focused manner and other relations relevant to AB are developed.

### F.5.2.1 Temporal variation of the CSH model for two body motion

Consider an AB pass during for which atmospheric density can be modeled with a constant scale height (equation (F.3-1)), so that the density as a function of altitude above periapsis is

$$\rho(h) = \rho(h_p) \exp\left[\frac{-(h - h_p)}{H_s}\right] \quad \text{Eq. (F.5-4)}$$

where  $h_p$  is the periapsis altitude and  $H_s$  is the density scale height. To produce the familiar “bell” shaped density versus. time profile, assume two-body motion about a spherical planet and expand the altitude in a Taylor series about the time of periapsis to get

$$h(t) = h(t_p) + \frac{1}{2}\ddot{h}(t_p)(t - t_p)^2 + O((t - t_p)^4) \quad \text{Eq. (F.5-5)}$$

where  $\ddot{h}(t_p)$  is the second derivative of altitude with respect to time evaluated at periapsis.


Under the identified assumptions, altitude is symmetric in time so odd derivatives vanish and the truncated terms are of order  $(t - t_p)^4$  and negligible except for very low eccentricity orbits.

Eliminating altitude in equation (F.5-4) in favor of time in equation (F.5-5) yields the “bell” shape variation of density with time.

$$\rho(t) = \rho(t_p) \exp\left[\frac{-\ddot{h}(t_p)(t - t_p)^2}{2H_s}\right] \quad \text{Eq. (F.5-6)}$$

It is of interest to write  $\ddot{h}(t_p)$  in terms of the orbital elements. Starting with the radial acceleration equation for the two-body problem

$$\ddot{r} = r\dot{\theta}^2 - \frac{\mu}{r^2} \quad \text{Eq. (F.5-7)}$$

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 192 of 286	

where  $\mu$  is the gravitational constant for the planet and  $\theta$  is the angular position in orbit. At periapsis  $r=r_p=a(1-e)$  and

$$r^2\dot{\theta}^2 = V_p^2 = \mu(1+e)/r_p \quad \text{Eq. (F.5-8)}$$

where  $V_p$  is the velocity at periapsis and  $e$  is the orbital eccentricity. Since  $\ddot{h}=\ddot{r}$  for a spherical planet, equation (F.5-7) reduces to

$$\ddot{h}(t_p) = \mu e / r_p^2 \quad \text{Eq. (F.5-9)}$$

To relate the orbital and atmospheric parameters, start with the Gaussian density function

$$f(x,\xi,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x-\xi)^2}{2\sigma^2}\right] \quad \text{Eq. (F.5-10)}$$

where  $\xi$  is the mean and  $\sigma$  is the standard deviation and

$$\int_{-\infty}^{\infty} f(x)dx = 1$$


Eq. (F.5-11)

Equation (F.5-10) and equation (F.5-6) suggest the substitution

$$\sigma^2 = H_s/\ddot{h}_p = H_s r_p^2 / \mu e \quad \text{Eq. (F.5-12)}$$

with  $e>0$ , leading to equation (F.5-6) in the desired form

$$\rho(t) = \rho(t_p) \sqrt{\frac{2\pi r_p^2 H_s}{\mu e}} \left[ \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(t-t_p)^2}{2\sigma^2}\right] \right] = \rho_p \exp\left[-\frac{(t-t_p)^2}{2\sigma^2}\right] \quad \text{Eq. (F.5-13)}$$

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 193 of 286	

From equation (F.5-11), the integral over the entire pass is

$$\int_{-\infty}^{\infty} \rho(t) dt = \rho(t_p) \sqrt{\frac{2\pi r_p^2 H_s}{\mu e}} \quad \text{Eq. (F.5-14)}$$

where again  $e > 0$  is assumed. This result shows that the area under the curve is proportional to the density at periapsis times the square root of the scale height, sometime called the “square root of scale height law.” Since the orbital velocity decrease due to drag is approximately proportional to the integral of density, over estimation of the scale height will result in an underestimation of the density as determined by an orbit determination approach that process tracking data before and after the unobserved AB pass.

The only approximation to arrive at equation (F.5-13) is the truncation of the Taylor series in equation (F.5-6) and as long as the scale height is small compared to the periapsis altitude the higher order terms are not significant. For example, for Mars with  $e=0.1$ ,  $H_s=7$  km,  $h_p=125$  km, and  $\rho_p=50$  kg/km<sup>3</sup>, the error in altitude is less than 1 km and the error in density is less than 0.1 kg/km<sup>3</sup> over an altitude range from periapsis to 5 scale heights above periapsis.

The integrals of dynamic pressure ( $0.5\rho V^2$ ) for total drag effect or heat flux ( $0.5\rho V^3$ ) for total heat input may be more important variables than density. Under the same assumptions for which equation (F.5-6) is valid, the velocity variation throughout the AB pass varies by only a few percent from the value at periapsis. So the total heat input during a pass is closely approximated by the value of the heat flux at periapsis times the radical equation (F.5-14).


Finally, the “drag duration” ( $T_d$ ) is often defined as the time from the inbound occurrence of 1 percent of maximum density to the outbound time when the density is 1 percent of maximum density, then the drag duration is twice the time for the spacecraft to increase in altitude above periapsis by  $4.6H_s$ . From equation (F.5-6) and equation (F.5-14)

$$T_d = 2 \sqrt{\frac{9.2 H_s r_p^2}{\mu e}} \quad \text{Eq. (F.5-15)}$$

Equation (F.5-15) has been used to define the AB duration for some AB missions.

### F.5.2.2 Quadratic time

Under the identified assumptions, the CSH model can be represented in time by

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 194 of 286	

$$\rho(t) = \rho_p \exp\left[\frac{-(t-t_p)^2}{2\sigma^2}\right] \quad \text{Eq. (F.5-16)}$$

where

$$\sigma^2 = H_s r_p^2 / \mu e \quad \text{Eq. (F.5-17)}$$

depends on orbit parameters and scale height. Again, to assure linear estimation,  $\log\rho$  is used as the observable and  $a=\log\rho_p$  and  $b=-1/2\sigma^2$  are the parameters. This model has the advantage that a precision trajectory is not required to generate altitude versus time. To predict to other altitudes, the scale height can be approximated from the solution for  $\sigma^2$ . It is seen that a disadvantage is that the model is symmetric in time about the time of periapsis and that maximum density occurs at periapsis, whereas few of the Martian density profiles satisfy either of these conditions and further, if one is only relying on the IMU time tags, the time of periapsis is unknown. A shift in the time of maximum density is easily accomplished by adding a linear term to get

$$\log\rho(t) = a + b(t-t_p) + c(t-t_p)^2 \quad \text{Eq. (F.5-18)}$$


which is still symmetric in time but centered at the model maximum density which occurs at  $t_{\max}=t_p-b/2c$  and has a value of  $\exp(a-b^2/2c)$ . This model (QdT) does however permit different inbound and outbound altitude profiles, but with the same scale height.

### F.5.3 Cubic and Quartic Time

One can introduce both asymmetry and a shift in the time of maximum density by extending the quadratic model to either a cubic (CubT) or a quartic model (QtT) in time, for example:

$$\log\rho(t) = a + b(t-t_p) + c(t-t_p)^2 + d(t-t_p)^3 + e(t-t_p)^4 \quad \text{Eq. (F.5-19)}$$

The quartic term might be included to assure that density decreases with altitude outside the data set or to provide a better estimate of the maximum density during the pass. There are profiles for which the coefficient  $e$  has a value greater than zero, so this model would not be recommended for extrapolation. For both models,  $H_s$  can be extracted from the quadratic coefficient. However, it was found that the  $(t-t_p)^4$  term often absorbed enough of the quadratic dependence that the  $H_s$  estimates were substantially biased. Consequently, no further consideration will be given to the quartic representation.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 195 of 286	

### F.5.4 Periapsis Timing Estimator

Satellite ephemeris propagation errors are usually dominated by along track deviations which for AB are manifested as time of periapsis errors. The PTE<sup>14</sup> was consequently designed to adjust the flight sequence so that it would be centered on the centroid of the density history. Based on the PTE  $\Delta t$  from one orbit, the initiation of the AB sequence for the next orbit is adjusted by  $\Delta t$ . PTE was run in shadow mode and validated during Mars Odyssey and was operational for MRO. Although the details are not exactly known, results of the models will be compared to an implementation based on Reference 14. The implementation is a simple density weighted time from periapsis to provide the location of the density centroid relative to periapsis

$$\Delta t = \frac{\sum t_i \rho_i}{\sum \rho_i} \quad \text{Eq. (F.5-20)}$$

where time is measured from periapsis and the sum is taken over all the density data above a threshold determined by the density noise level.

### F.5.5 Single Orbit Examples

Each of these models was applied to the four Odyssey orbits in Figure F.4-3. Data within 14 km altitude of periapsis are used for the LS solutions. Results are shown in Figure F.5-1 for both density versus time and density-altitude profiles, where the density data are shown as dots. Relevant solution parameters for these orbits are tabulated in Table F.5-1. For the QdT and CubT models, the scale height was calculated using Equation (F.5-17) where the position and velocity at periapsis was used to calculate the eccentricity. For orbits 44, 157 and 280, little difference between the models is seen in the plots. For these orbits, the CSHT and time quadratic (QdT) models are nearly identical and the differences in the parameters in Table F.5-1 are ignorable. Examination of the orbit 44 profile shows that the cubic (CubT) model is beginning to diverge above 110 km and  $\rho=10 \text{ kg/km}^3$  with one branch going to zero density and the other going to an infinite density with further increase in altitude. To fit the “flat” top of orbit 157, all models produced a large scale height near 23 km, whereas the remaining orbits have scale heights of less than 11 km. Similar statements can be made for the time of the maximum density.



# NASA Engineering and Safety Center Technical Assessment Report

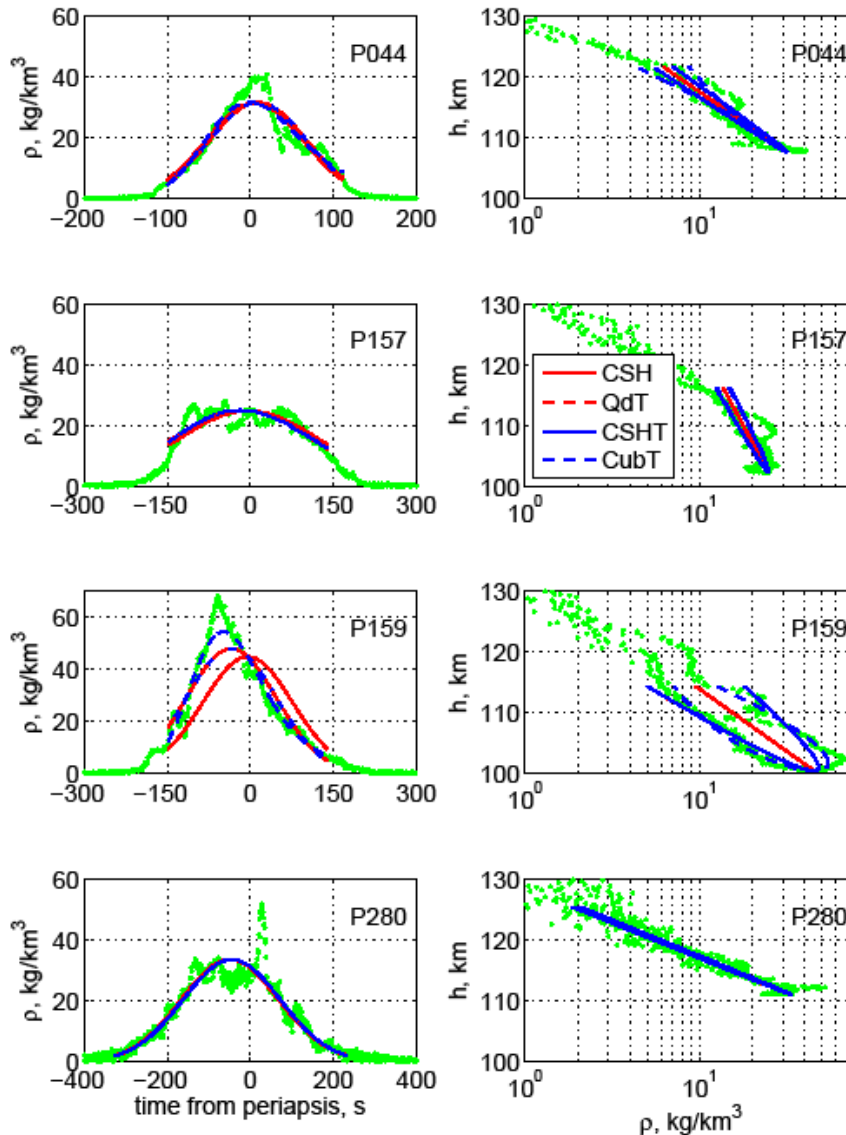
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
196 of 286



*Figure F.5-1. Density Model Least Squares Fit to Four Odyssey Orbits. Data Altitude Range  $\delta h=14$  km*

Estimation of the scale height is consistent among the models, but of course all the orbit 157 estimates are much greater than estimates from nearby orbits and it would likely be unwise to use the large scale height for orbit 157 to predict the density for orbit 158. To do so would give  $21.8 \text{ kg/km}^3$  versus the measured value of  $38.8$ . Predicting orbit 159 from the 157 values gives  $21.9 \text{ kg/km}^3$ . Orbits like this demonstrate the need to combine estimates from a number of orbits and even then, large differences might be expected.




**Table. F.4-1. Model Comparisons for Four Mars Odyssey Orbits**

Model	Odyssey Orbit			
	44	157	159	280
	$\rho_{\max}$ , kg/km <sup>3</sup>			
Data	40.8	27.8	68.1	51.8
CSH	31.5	24.9	45.0	33.6
QdT	31.6	25.0	48.0	33.5
CSHT	31.6	25.0	48.1	33.6
CubT	31.7	25.1	54.5	33.6
	$t_{\max}$ , sec			
Data	20.0	-44.5	-58.5	29.5
CSH	5.0	-3.5	-4.5	-49.4
QdT	9.0	-15.5	-34.5	-45.4
CSHT	9.0	-15.5	-34.5	-46.4
CubT	-1.0	-19.5	-48.6	-44.4
<b>PTE</b>	6.8	-5.4	-28.6	-41.3
	Hs, km			
CSH	8.6	23.1	8.9	5.1
QdT	8.7	23.6	9.2	5.0
CSHT	8.6	23.1	8.9	5.1
CubT	8.3	23.8	9.7	5.0

### F.5.6 Multiple Orbit Comparisons


The four algorithms in Table F.5-1 and the CSHIO algorithm were applied to all three Mars missions. Only data during the “main” AB phase were included. In addition, MGS data for orbits 910 through 980 were excluded because of an onboard computer issue that significantly reduced the quality of the accelerometer data. There is a subtle difference in how the data are selected for the three constant scale height models and the two time polynomial models. For the former, data are selected within a specified altitude range of periapsis, which for all these results is 14 km or about 2 density scale heights. Unless periapsis is at the equator or a pole, planetary flattening results in these data being asymmetric in time. Conversely, for the latter two models, the data are selected symmetric in time around periapsis with a time interval that corresponds to

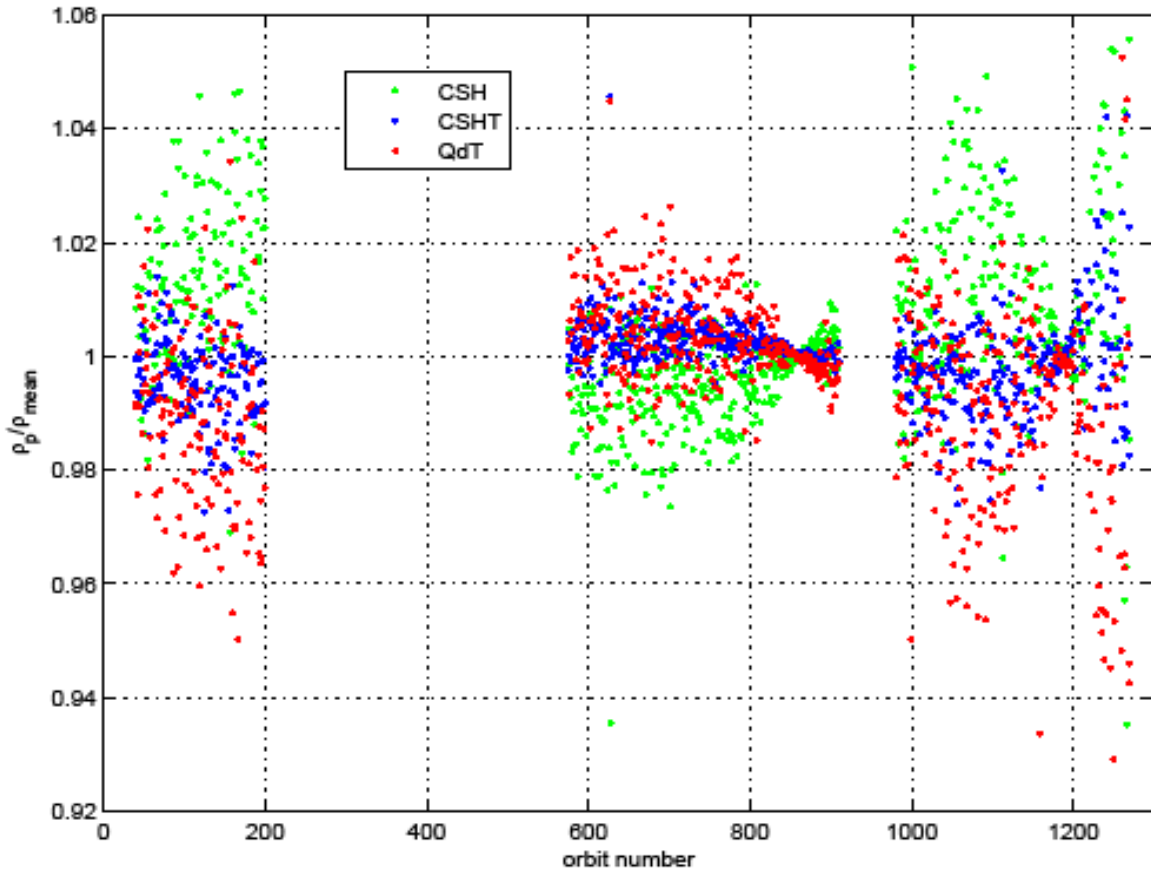
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 198 of 286	

a planetocentric radius change of 14 km. The resulting in planetodetic altitude distribution is generally asymmetry. This small difference has a noticeable effect on the results.

**Density:** As a basis for comparison, the “mean” density for each orbit was calculated by averaging all five solutions for density at periapsis. Results are presented as ratios of recovered density to this mean density. For MGS it was found that this ratio varied from 0.92 to 1.06. The orbit average difference between CSH and CSHIO had a  $\mu=0.0045$  with  $\sigma=0.02$  and between QdT and CubT  $\mu<10^{-5}$  and  $\sigma<10^{-3}$ . Because these pairs of recoveries are so similar, only one of the pair will be shown for some of the results. Figure F.5-2 shows these ratios for the CSH, CSHT and QdT methods. There are a couple of general trends evident. First, when the CSH ratios are generally greater than one, the QdT ratios are generally less than one. Second, the CSHT method provides results closest to unity over the entire mission. Third, there are two places where all three methods give nearly the same density, near orbits 860 and 1190.


It will be seen in the next paragraph that a couple of these trends can be explained in terms of (1) the method of selecting data as discussed and (2) the time between periapsis and the minimum planetodetic altitude. The Mars Odyssey and MRO analyses showed similar trends in ratio ranging from 0.92 to 1.06, model agreement near the pole, and CSH and QdT providing opposite deviations from unity. The CSHT model provides results closer to the mean than the other two models, but with slightly larger deviations than MGS.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		199 of 286	

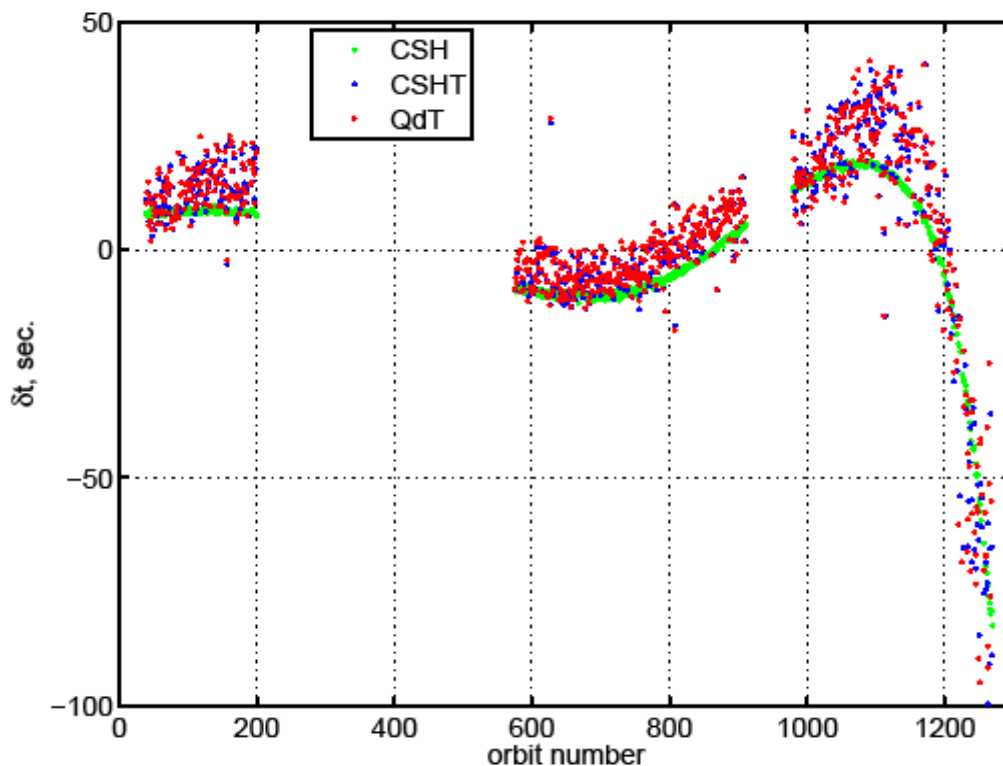


**Figure F.5-2. MGS Estimated Periapsis Density for Three Methods Normalized by Mean Periapsis Density of all Five Methods**

**Time of Maximum Density:** Consider Figure F.5-3 which shows the time from periapsis to the time when the various models predict the maximum density. For the CSH model, this is time from periapsis to minimum planetodetic altitude, i.e., the same “time to minimum altitude,” presented in Figure F.4-4. Time difference changes slowly because latitude and orbit eccentricity are changing slowly until near the final orbits when the orbit is nearly circular. During the first 200 orbits, MGS is passing from north to south as periapsis regresses northward. Consequently, minimum altitude occurs after periapsis. The CSHT and QdT results suggest there is an additional effect that further delays the time. This would be consistent with an equator-ward increase in density, which is the climatological trend. The periapsis regresses past the equator on orbit 860 and here the altitude data distribution will be symmetric in time and conversely. So, the models should predict essentially the same atmospheric parameters. It is seen from Figure F.5-2 that this is true for density, but not so for the  $\delta t$  values. Again there is an along track density gradient that delays the epoch of maximum density. At the risk of over analysis, one possibility is that the maximum density is occurring in the northern hemisphere

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 200 of 286	


since during all of phase 2,  $L_s$  (Figure F.2-1) is between  $0^\circ$  and  $90^\circ$ , so it is hemisphere spring and the Sun is in the northern hemisphere. The balance of the orbits after 860 is readily explained by a minimum density occurring near the pole. Maximum positive  $\delta t$  occurs near orbit 1060 as periapsis regresses past  $45^\circ$  latitude, the region of maximum gradient in the planetocentric altitude. As periapsis passes over the pole, there will be symmetry in the data distributions and the densities are nearly identical. One final note, the difference between the  $\delta t$  for CSHT and QdT has a  $\mu=0.2$  second and  $\sigma=1.9$  second. These two methods are providing excellent agreement in  $\delta t$ .



*Figure F.5-3. MGS Time from Periapsis to Time of Maximum Density as Predicted by Three Models*


Mars Odyssey and MRO analyses provided similar results. MRO had generally smaller deviations from the  $\delta t$  caused by flattening than MGS. Mars Odyssey on the other hand, while periapsis regressed toward the pole, showed up to 40 seconds positive  $\delta t$  deviations which rapidly switched to negative values up to -40 seconds while moving away from the pole. Perhaps these large, rapid variations were due to the polar warming.

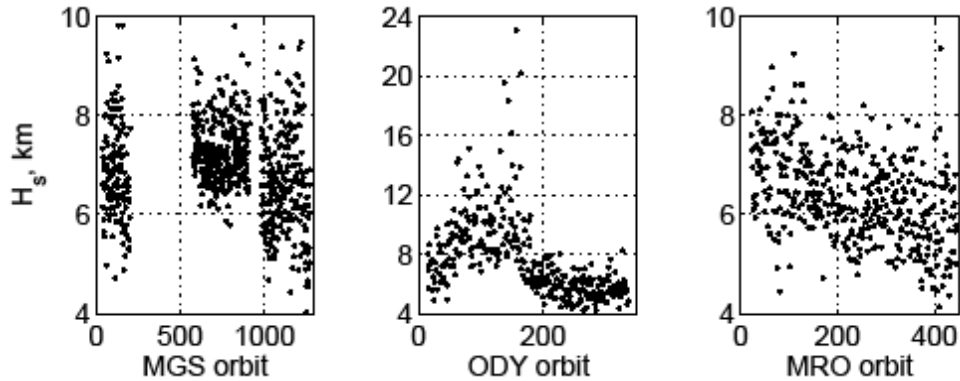
**Density Scale Height:**  $H_s$  is the final variable of interest for predicting the periapsis density at subsequent orbits. As might be expected, the estimation of scale height is more sensitive than density to the altitude span of the data set. Orbit 157 in Figure F.5-1 illustrates the difficulty.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 201 of 286	

The solution used data within  $\delta h=14$  km of periapsis, i.e., about two expected scale heights. The resulting  $H_s=23$  km given Table F.5-1 is not a realistic value to use for predicting density at the next periapsis. From the figure it can be expected that as the altitude range  $\delta h$  is increased, the value of  $H_s$  would decrease perhaps to more realistic values, but the estimated periapsis density will likely increase. The data above 100 km appears to follow a straight line with a scale height of about 7.5 km, but using just these data would yield a periapsis density of about  $100 \text{ kg/km}^3$ . Hence, using a much larger data set would lead to a significantly higher density prediction. Orbit 158 occurred 1 km lower in altitude than 157 and had a density of  $41 \text{ kg/km}^3$ . So the predictions using a 14 km altitude range underestimated the density for 158 and using a very large altitude range would have overestimated the density. Studies for all missions using  $\delta h=7, 11, 14,$  and 21 km altitude ranges showed that for orbits with “bell shaped” density histories, even with time shifts, the estimates of  $H_s$  generally differed by less than 0.5 km between the 11 and 14 km cases and 0.3 km between the 14 and 21 km cases. Differences between the 7 and 14 cases were around 1 km. For orbits that vary significantly for the “bell shape,” the results are mixed. Using an altitude interval of  $\delta h=2H_s$  seems to be a reasonable compromise.

Like the other parameters, the estimation of  $H_s$  within a family of methods, e.g., (CSH, CSHT, CSHIO) or (QdT, CubT) were consistent with standard deviations for the differences of less than  $\sigma=50\text{m}$ . Between the two families,  $\sigma<500\text{m}$ . Consequently, only the CSHT results are shown in Figure F.5-4. The means of the three MGS data segments ( $H_s=6.9, 7.3, 6.6$  km) have trends that are consistent with temperature decreasing toward the poles. The standard deviations are 1.1, 0.7 and 1.1 km, from left to right. Mars Odyssey scale heights (i.e., temperature) were the means of discovering the polar warming but outside the polar region the mean scale height drops to about 5.6 km, well below the expected value. Global circulations model simulations of a polar warming<sup>15</sup> show strong adiabatic heating near the pole due to subsiding flow and an adjacent region of cooling. This cool region may be the reason for these small scale heights. In this region  $\sigma=0.76$  km. MRO shows a decreasing  $H_s$  trend as periapsis regresses toward the pole. If there was a south polar warming, it occurred after periapsis passed the pole. Averaging over 100 orbit blocks,  $H_s$  varies from  $\mu=6.9$  and  $\sigma=0.86$  to  $\mu=5.7$  and  $\sigma=1$  km over the mission.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 202 of 286	

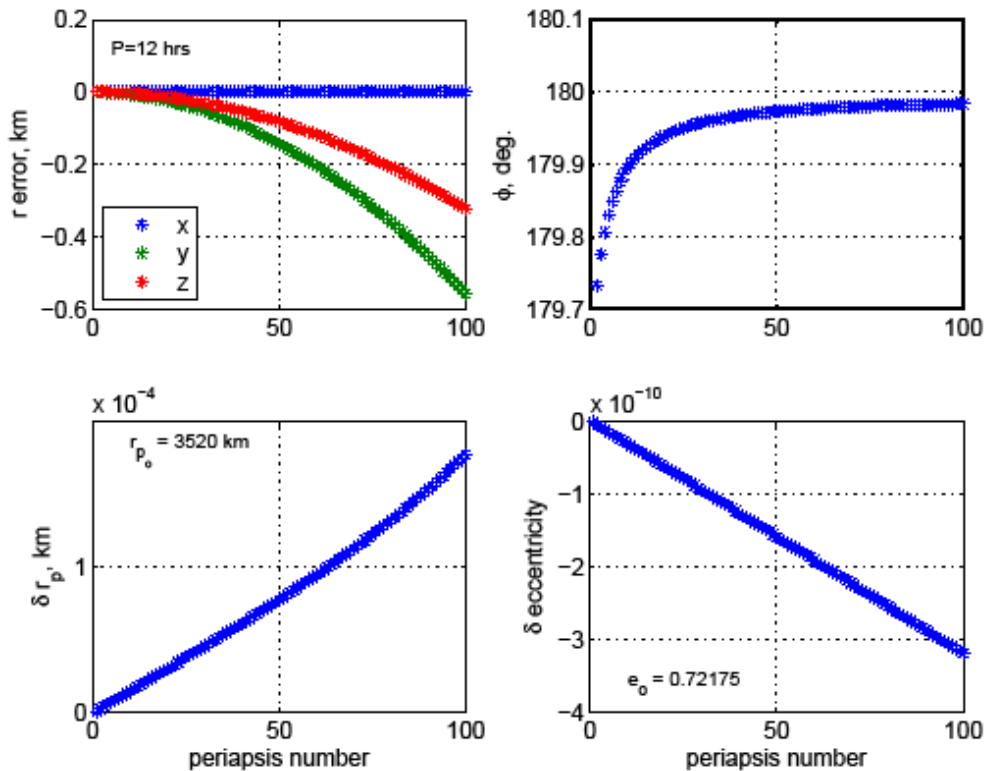


*Figure F.5-4. Density Scale Height Recovered using the CSHT Method and a Data Altitude Range of  $\delta h=14$  km*

## F.6 Influence of Along Track Ephemeris Errors

Studies of numerical solutions of two body satellite orbits usually show that the largest position error is along track<sup>23</sup> and increases with  $t$ ,  $t^{3/2}$  or  $t^2$  depending on the numerical integration scheme, computer specifications and problem formulation. An example is shown in Figure F.6-1 using a simple Runge-Kutta-Fehlberg 4/5 integration method. The nature of the errors is typical, but the deviations will vary in magnitude with integration method particulars. The orbit starts at periapsis at an altitude of 125 km. The initial periapsis location is in the equator on the x-axis and the orbit inclination is  $30^\circ$ .

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		203 of 286	



**Figure F.6-1. Numerical Integration Errors for a RKF 4/5 Method Applied to a Mars Satellite in an AB Orbit with a 12-hour Period**

The upper left panel shows, for 100 orbits, the difference between subsequent periastris position and the initial position. Note the errors are predominately in the y and z directions, i.e., normal to the position vector at periastris. The upper right panel shows the angle ( $\phi$ ) between the position error vector and the periastris velocity vector. After just the first orbit, the predominant position error is essentially along track and becomes even more closely aligned as orbit number increases. From an osculating orbital element view point, the dominate error is in the time of periastris passage. In anticipation of using equation (F.5-17) to obtain an estimate of scale height, note from the lower panels that the errors in  $r_p$  and orbit eccentricity are negligible over 100 orbits. To obtain a first order evaluation of the effects of such errors on the estimation of atmospheric density and scale height, the following study was performed. Assuming a two body orbit and a CSH atmosphere with periastris density of  $1 \text{ kg/km}^3$  and a density scale height of 7 km, density versus time was generated to simulate the onboard recovery of density purely from IMU data. Altitude versus time was generated from the two body equations. Then, the altitude versus time ephemeris data was shifted in time to simulate an ephemeris time of periastris offset.

The CSH, CSHT and QdT methods were then used to recover density at periastris and scale height using the equations (F.5-1), (F.5-2), (F.5-16), and (F.5-17), respectively.



Title:

Autonomous Aerobraking (Phase 1)

Page #:

204 of 286

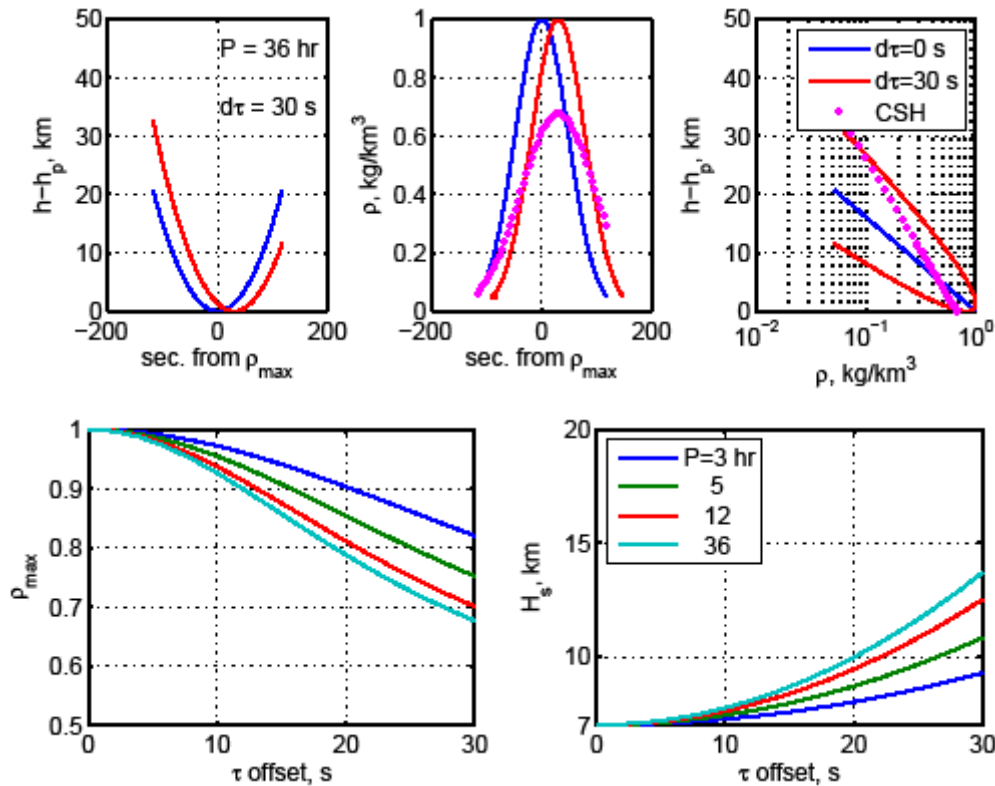


Figure F.6-2. Effect of Ephemeris along Track Offsets on CSH Estimation of Density and Density Scale Height

Figure F.6-2 shows the effect of an ephemeris offsets from 0 to 30 seconds for a range of orbital periods from 3 to 36 hours. The blue line in the upper figures corresponds to IMU derived data except for the left-most figure where the line corresponds to the altitude if there were no ephemeris offset. The red line corresponds to the ephemeris values with a 30-second delay in the ephemeris time. The magenta line corresponds to the CSH model estimation results. The CSH maximum density occurs near the same IMU time as the ephemeris maximum density. As shown in the lower two plots, regardless of the ephemeris offset or the orbital period, the CSH model always underestimates the maximum density and overestimates  $H_s$ . Larger offsets can be tolerated more readily by low eccentricity orbits. For the high eccentricity orbits, a 10-second offset leads to a 10 percent underestimation of maximum density and about a 1 km over estimation of  $H_s$ . For Mars, these deviations are within the  $1\sigma$  natural variability in both density (Figure F.3-1) and  $H_s$  (Figure F.5-4) and may be acceptable. It was evident that 30-second offsets would not be acceptable.





# NASA Engineering and Safety Center Technical Assessment Report

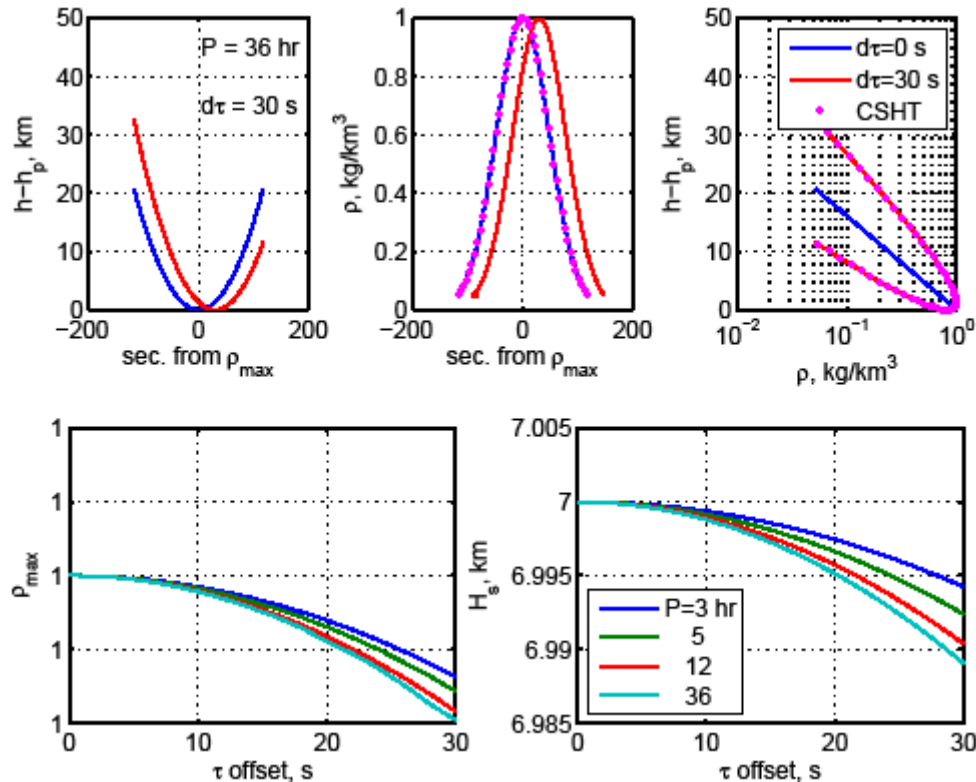
Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:


## Autonomous Aerobraking (Phase 1)

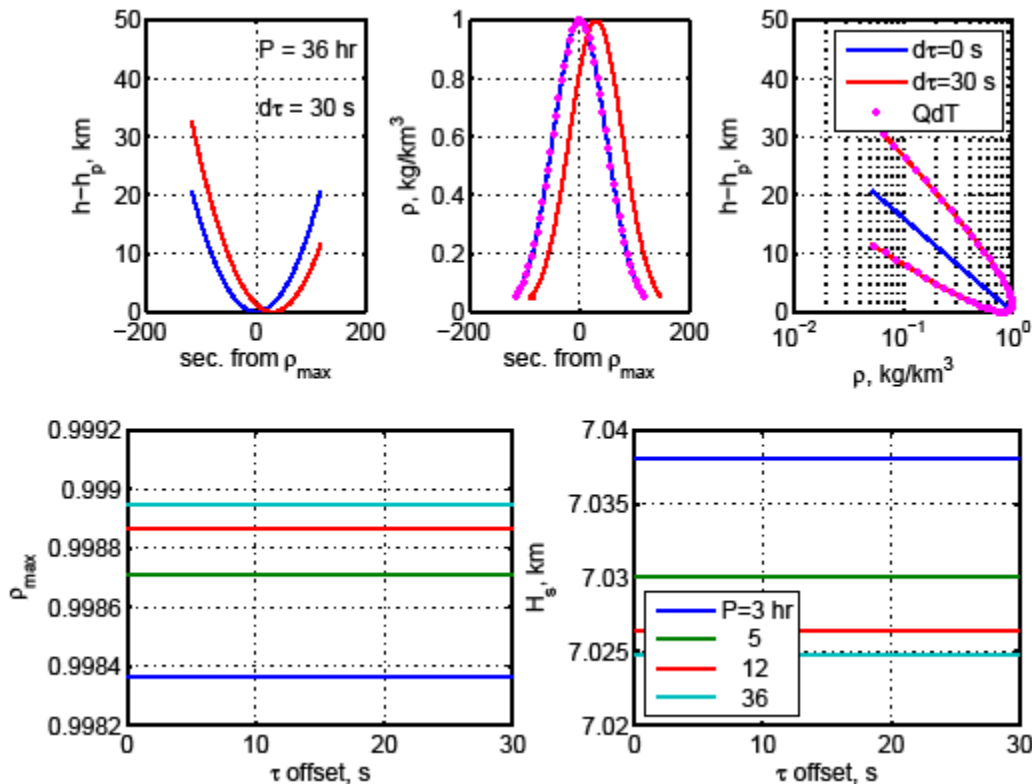
Page #:  
205 of 286



**Figure F.6-3. Effect of Ephemeris along Track Offsets on CSHT Estimation of Density and Density Scale Height**

When the linear time term is added to the CSH model to obtain the CSHT model (equation (F.5-2)) the influence of the ephemeris offset becomes almost nil. The results are shown in Figure F.6-3. Here, only every third magenta point is plotted so the blue and magenta results can be distinguished. In the top center frame, the difference is negligible and the CSHT model recovers the maximum density to 5 decimal places and  $H_s$  to within a few meters over the entire period and time offset sweep. This method still requires a model that produces an altitude profile versus time. The model can either be a precision ephemeris or an approximation based on a set of osculating or mean elements. As a backup to the precision ephemeris, consideration should be given to utilizing other approaches to obtaining altitude vs. time profiles.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		206 of 286	



**Figure F.6-4. Effect of Ephemeris along Track Offsets on OdT Estimation of Density and Density Scale Height**

Finally, the QdT (equation (F.5-18)) method was tested with the results (Figure F.6-4) being similar to those of the CSHT results. The main difference is in the estimates of the max. density and  $H_s$ . Here, an orbital period dependent bias is seen in each parameter and these biases are essentially independent of the time offset. The biases are due to truncating the Taylor series expansion in equation (F.5-9) and are much larger than the errors produced by the ephemeris time offset. The latter are illustrated by the two lower panels in Figure F.6-3. This model has the advantage of only requiring a reasonable estimate of  $r_p$  and orbital eccentricity as seen in equation (F.5-17).

This study was done near the end of the Phase 1 of the project and is by no means complete. The next step would be to apply an ephemeris offset to MGS, Mars Odyssey and MRO density sets to evaluate the effects. This can be done in a relatively short time but is not complete. The complete study would include ephemeris offsets in the density recovery process itself. Though probably a small effect, the ephemeris influences the density recovery in that the position and velocity are necessary to calculate the velocity relative to a rotating atmosphere to determine angles of attack and sideslip for interpolation into the aero data base. The exact influence will vary from orbit to orbit depending on the phasing of the attitude oscillation relative to the time of

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 207 of 286	

periapsis. A second study will be required to determine the accuracy with which periapsis radius and orbit eccentricity are determined by the particular numerical method use for the onboard ephemeris generation. A third study should include determining the precision with which the altitude vs. time profile must be known. The current approach is based on a precision ephemeris, but these results suggest a less precise trajectory may be required. The results of these studies could ameliorate the effects on an ephemeris time offset and substantially extend the duration of AA before human intervention is required.

## F.7 Aerobraking Corridor Maintenance

As mentioned earlier, a model of atmospheric density has a couple of purposes: (1) to quantify characteristics of the atmosphere for the current pass which might be used for heating calculations<sup>19</sup>, (2) to provide a prediction of the characteristics of the next pass, and (3) provide information needed to calculate the orbit trim maneuvers to stay in the AB corridor.

### F.7.1 Corridor Maintenance Maneuvers


To maintain the AB corridor, orbit trim maneuvers are generally performed near apoapsis to adjust the altitude of subsequent periapses and thereby control the atmospheric density.<sup>21</sup> For tangential, impulsive maneuvers, the first order  $\delta V_a$  required at apoapsis to raise periapsis altitude by an amount  $\delta r_p$  is first given for two body motion and secondly is given by relating the altitude change through the CSH model to obtain the desired fractional change in periapsis density  $\delta \rho_p / \rho_p$ .

$$\delta V_a = \frac{n}{4} \sqrt{\frac{r_a}{r_p}} \delta r_p = - \frac{n}{4} \sqrt{\frac{r_a}{r_p}} H_s \left( \frac{\delta \rho_p}{\rho_p} \right) \quad \text{Eq. (F.7-1)}$$

where  $\delta r_p$  has been approximated by  $\delta h_p$ ,  $n$  is the orbital mean motion,  $r_a$  ( $r_p$ ) is the apoapsis (periapsis) radius and  $H_s$  and  $\rho_p$  are the expected density scale height for the next orbit. Even without a precision trajectory, the previous results strongly suggest that the latter two variables are likely to contribute the majority of the uncertainty in calculating the desired  $\delta V_a$ .


### F.7.2 Predicting Atmospheric Parameters for the Next Pass

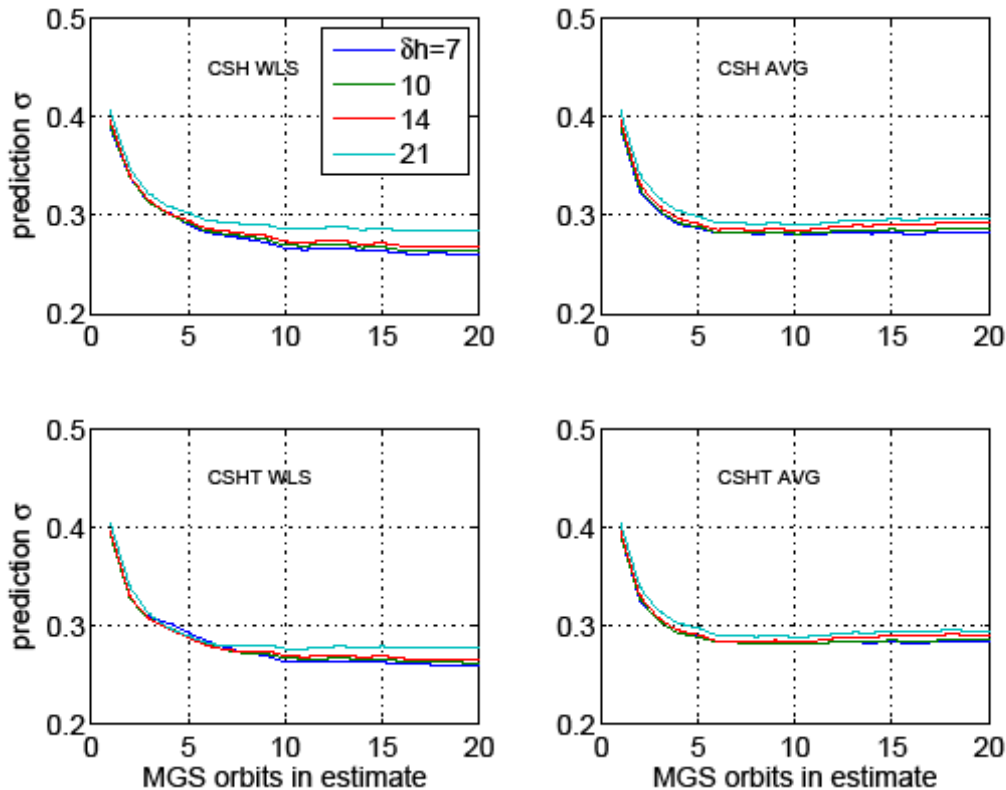
For AA it will be necessary to have a prediction of atmospheric density for the next pass through the atmosphere. From past experience with Mars AB it is clear that there are likely to be large variations between predicted and observed density that are not within the current ability to predict from either empirical or numerical models of the atmosphere. Any of the density models discussed might be used to generate parameters for predicting the conditions at the next periapsis as was demonstrated in the study of “persistence.” All the models have persistence values that

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 208 of 286	

deviate up to a factor of 2 from unity. This naturally raised the question if some averaging of the estimates would produce a better “prediction” ratio. As mentioned, for both Magellan and MGS, some form of atmospheric density was averaged over a number of orbits to be used to predict the density at the next periapsis. Using the extensive set of data from Mars, a study was performed of two averaging methods over a range of altitudes ( $\delta h$ ) used to obtain model coefficients and over the number of orbits used in the averaging process. One would anticipate both of these variables would influence the results. Only the CSH and CSHT models were included as CSH is the simplest model and CSHT provided prediction results that were the closest to the average of all the methods. The data collection altitude ranges studied were  $\delta h=7, 10, 14$  and  $21$  km. This range starts near the mean  $H_s$  averaged all latitudes and times. The values of  $10$  and  $14$  can be thought of as  $1.5$  or  $2$  times the  $H_s=7$  km value or  $1$  and  $1.5$  times the  $H_s\sim 10$  that was seen during the polar warming. The number of orbits for the averaging was varied from  $1$  to  $20$ . It might be anticipated that a low number of averaging orbits will have a large standard deviation for the prediction ratio while estimates using long data arcs may be biased by the time dependent (latitudinal, seasonal, diurnal) variations in atmospheric characteristics.

Figure F.7-1 shows the results for MGS. The prediction  $\sigma$  is the standard deviation over all the values of the ratio of observed to predicted density across the entire mission. Two simple averaging methods were explored for both the CSH and CSHT models. For notational convince, let  $n+1$  be the orbit number at which density is to be estimated using model values from orbits  $n, n-1, \dots, n-k+1$ , where  $k$  is the “number of orbits in the estimate.” When the number of orbits in the estimate  $k=1$ , the prediction value is the standard deviation of all the points in Figure F.5-2 for each model. For the “CSH AVG” results the previous  $k$  values of ‘a’ and ‘b’ in equation (F.5-1) are used to get a value of ‘a’ at the periapsis altitude of orbit  $n+1$ . These  $k$  values are simply averaged to obtain the “estimated” density at orbit  $n+1$ . This value is used in the denominator of the prediction ratio. A similar process is used for the CSHT model where the time dependence in this model is ignored in the prediction. The second averaging method attempted to account for the “accuracy” of the estimated parameters. The LS process was turned into a WLS method by using the reciprocal of the rms deviation between observed and model predicted density to “weight” the data. Orbits with large deviations, like orbit  $159$  (Figure F.5-1), would be weighted lower than orbits which had a smaller rms, like orbit  $280$ . So the second method is the weighted sum of the estimates, much like a minimum variance linear combination of random variables. No probabilistic interpretation is attempted for these results for obvious reasons.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		209 of 286	



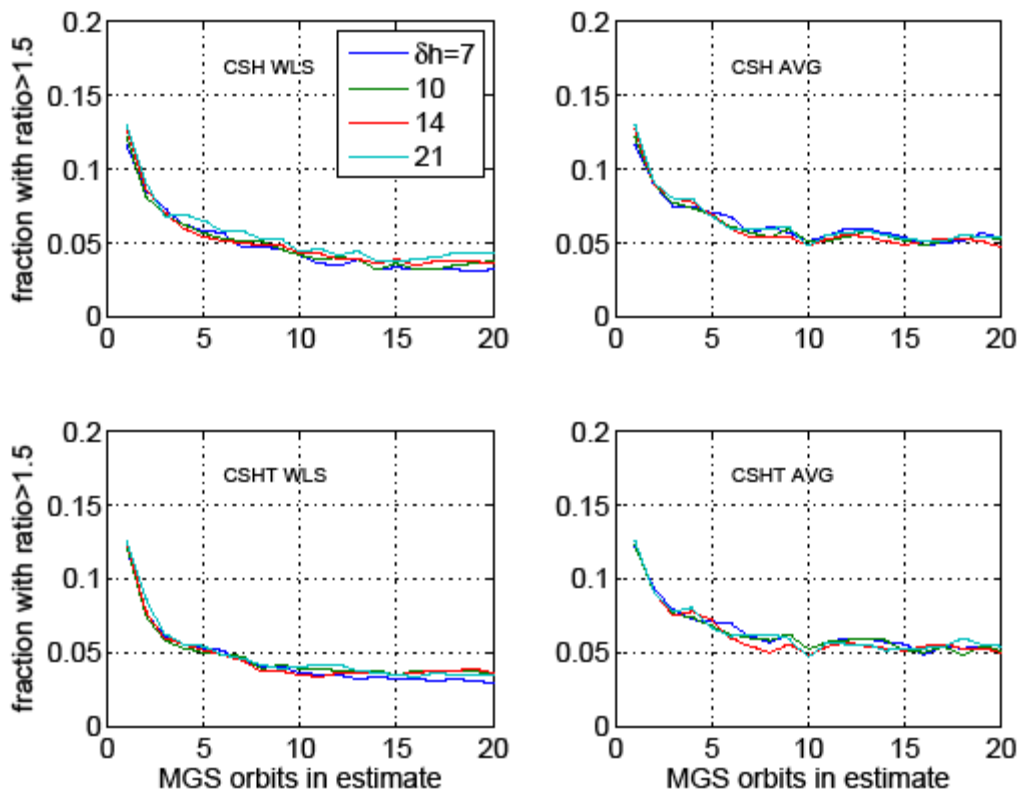
**Figure F.7-1. Density Prediction Capability of Various Methods for the MGS Mission**

Referring to the figure, the standard deviation across all orbits starts near 40 percent  $1\sigma$ . The initial downward trend as the number of orbits increases is to be expected. From a practical standpoint little is to be gained in reducing  $\sigma$  after 10 to 15 orbits and the AVG results start to increase slightly after 10 orbits are averaged. The three lower values of  $\delta h$  provide similar results and noticeable lower than the  $\delta h=21$  case, although this difference is likely not significant from an AA standpoint. This residual deviation of about 28 percent is interpreted as the natural variability from the “mean” atmosphere and cannot be reduced without significantly more knowledge of the atmosphere than is available from onboard measurements alone.

The standard deviation does not tell the whole story on prediction for AA. The probability of the ratio being greater than a specified value may be more relevant. The gamma distributions shown in Figure F.3-2 provide a more rigorous means of making probabilistic statements. Here, a simpler approach is taken by just tracking the fraction of total orbits for which the ratio of observed to predicted density ratio exceeds 1.5. For MGS this result is shown in Figure F.7-2 for the same methods and data as Figure F.7-1. There appears to be little advantage to using more than 10 to 15 orbits for the prediction and WLS provides about 1 to 2 percent improvement over


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		210 of 286	

the AVG approach. Note that WLS is computationally more cumbersome than the AVG approach, which is a consideration for onboard computation. For prediction the shorter data spans,  $\delta h < 10$  provided a small advantage, for Figure F.7-2 the lines cross repeatedly with  $\delta h = 14$  being lower than the others in more cases. Setting  $\delta h$  at about 2 scale heights may be a good rule of thumb and unless time of maximum density is a desirable parameter to estimate, the simple CSH method seems like a good candidate. Selecting between AVG and WLS is less obvious.

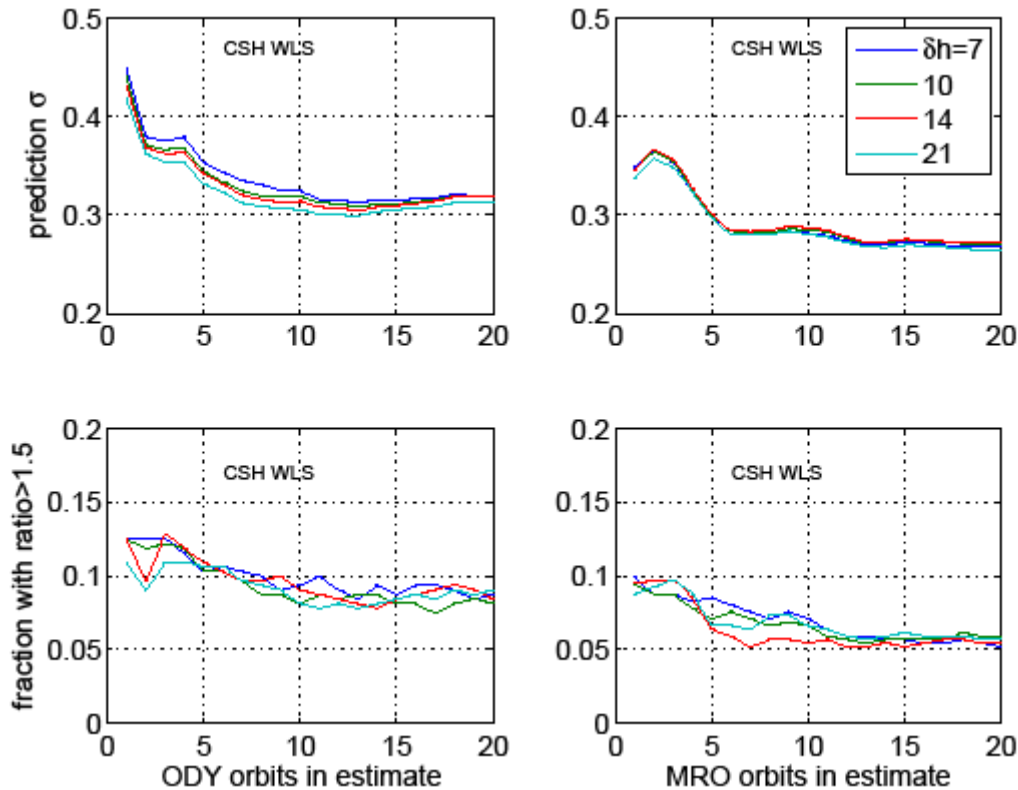


**Figure F.7-2. Fraction of Total MGS Orbits for which the Ratio of Observed to Predicted Density Exceeded 1.5**

Just to complete this story, similar results are shown for Mars Odyssey and MRO in Figure F.7-3 for just the CSH model and the WLS prediction method. The Mars Odyssey prediction has a minimum at 0.3 which is 10 percent higher than either MGS or MRO, but still over a 30 percent reduction below the initial persistence values. This higher variability appears in the prediction greater than 1.5 have nearly 10 percent of the orbits above this limit. The longer the span of orbits used in the prediction, the smaller the  $\sigma$ . The  $\delta h = 21$  appears to be optimal for Mars Odyssey, consistent with Mars Odyssey having  $H_s > 10$  km for a significant fraction of the

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		211 of 286	


mission. Of course, MRO behaves in a different manner from the other two missions with an initial rise in  $\sigma$  followed by a steep fall eventually becoming 0.28 like MGS. The decline in the 1.5 fraction is slower than MGS, but does get near 5 percent eventually.

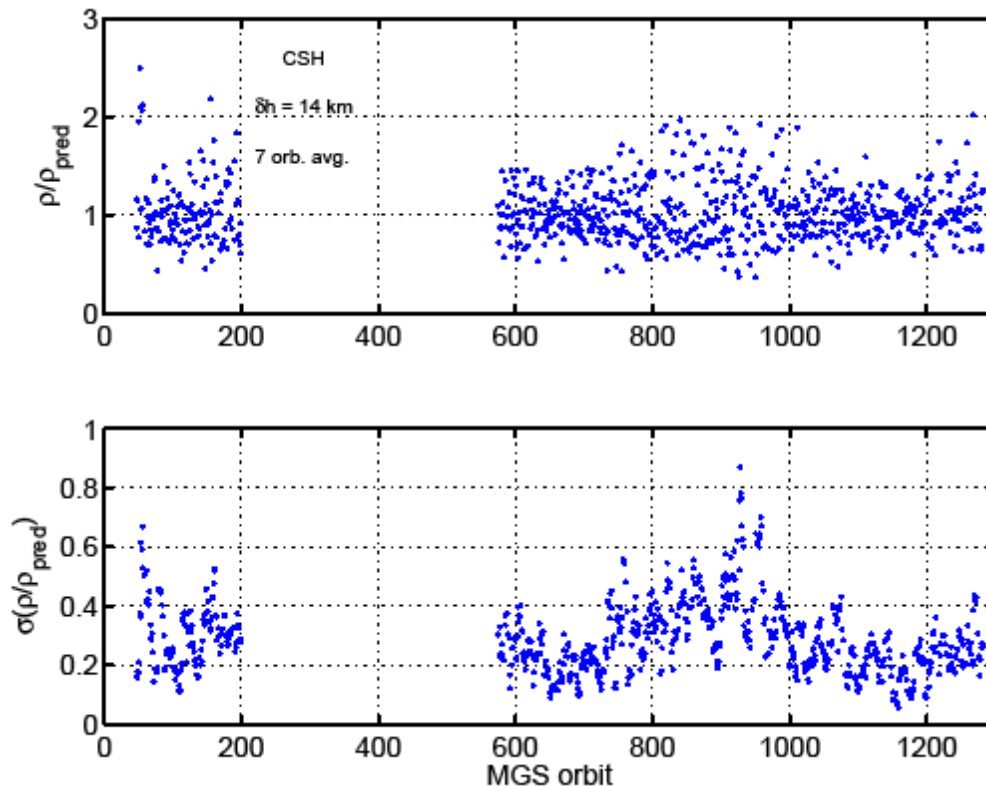


*Figure F.7-3. Prediction and Orbit Fractions > 1.5 for Mars Odyssey and MRO*

### F.7.3 Prediction Errors

Although not an active part of the current AA strategy, it is of interest to have an estimate of the uncertainty of the predicted density at the next periapsis shown in Figure F.7-1, Figure F.7-2, and Figure F.7-3. The only method tested during Phase 1 was to simply calculate the standard deviation over the number of orbits used to obtain the predicted density for MGS for the CSH method. For example, in Figure F.7-1 the upper right panel shows the prediction sample standard deviation averaged over the entire MGS mission for the CSH “average” prediction method. The prediction accuracy varies throughout the mission due to the latitudinal, season and diurnal natural variability of the atmosphere. One method to capture this local variability is to determine the sample standard deviation for each orbit prediction. Figure F.7-4 shows this variation for the MGS case with  $\delta h=14$  km and a 7 orbit averaging, the red line in Figure F.7-1, upper right.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 212 of 286	




*Figure F.7-4. Prediction Standard Deviations for CHS Method Based on Seven MGS Orbit Samples*

The upper panel of Figure F.7-4 shows the ratio of the observed density at the next periapsis location to the predicted density based on the indicated CSH method. The standard deviation over all these orbits would provide one point on the red curve in Figure F.7-1, upper right. The variation here is similar to the persistence shown in Figure F.3-1 for MGS. The local sample standard deviation is shown in the lower panel Figure F.7-4. This sample s can be interpreted as the ability to predict the density at periapsis of the next pass. This approach provides a “trailing” and highly variable indicator of variability. Phase 2 studies are proposed to improve this simple method, to expand the study to other methods (e.g., CSHT, QdT, etc.) and to quantify the accuracy of this and similar approached.

## F.7.4 Summary

1. Mars is a challenging environment for AA due to the large, natural orbit to orbit variability in the density profiles. With the plethora of data from MGS, Mars Odyssey and MRO, some latitudinal, seasonal trends have been identified, but the best that has been done to date is to reduce the variability by about one third by averaging over a number of orbits.




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 213 of 286	


2. Thermospheric waves, due to thermal tides in the lower atmosphere, produce nearly twice the average variability in temperate latitudes than at polar latitudes. AA strategies should be designed to take advantage of this well documented latitudinal variation, recognizing of course that a polar warming can upset the trend.
3. Five different formulations for the density variation with altitude and time were investigated. If maximum density or the time of maximum density is not relevant, then the simple CSH model performs nearly as well as more complicated models. If time is important, the hybrid CSHT is the likely choice.
4. Models that depend on altitude are likely to be sensitive to ephemeris errors and combined studies of atmospheric model parameterization and ephemeris propagation errors should be performed to better quantify this interaction. Consideration should be given to using atmospheric data to adjust the ephemeris time of periapsis passage. If ephemeris errors in altitude versus time become too large, models that depend only on time can be considered without loss of performance. The extent to which such models are applicable should be evaluated. Such models should be considered as the primary onboard algorithm.
5. Two methods were used to combine data from numerous orbits to improve the prediction for subsequent orbits. A weighted least squares method performed a little better than simple averaging, but at the cost of additional software on the vehicle. Both methods reduced the variability by about 30 percent. The remaining  $1\sigma$  deviations of about 30 percent are likely due to natural variability that will have to be included in vehicle design.
6. The AA strategies to date do not rely on statistical considerations to improve performance or to quantify risk. Studies should be performed to determine the advantages and limitations of including a more statistical based approach.
7. Due to the limited data, the extensive analyses reported here cannot be performed for AB at Venus and Titan. However, significant orbit to orbit variation has been noted at Venus and gravity waves have been seen at both bodies. PVO<sup>4</sup> and Magellan orbit determination results show modest orbit to orbit variability on the day side of Venus and large variability on the night side and near the terminator. PVO mass spectrometer data provides some insight into gravity wave structure. Based on these limited data, studies should be performed for Venus AB similar to those performed herein for Mars.

## F.8 References


1. Willcockson, W. H., Magellan Aerobraking Control Corridor: Design and Implementation, *Adv. Astronautical Sciences*, Vol. 87, Part II, 1994, pp. 647-662.
2. Spencer, D.A. & Tolson, R.H., "Aerobraking Cost and Risk," *J. Spacecraft and Rockets*, Vol. 44, No. 6, Nov.-Dec., 2007, pp.1285-1293

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 214 of 286	

3. Tolson, R.H., Keating, G.M., Zurek, R.W., Bougher, S.W., Justus, C.J., Fritts, D.C., "Application of Accelerometer Data to Atmospheric Modeling During Mars Aerobraking Operations." *J. Spacecraft and Rockets*, Vol. 44, No. 6, 2007, pp. 1171-1179.
4. Konopliv, A.S. & Sjogren, W.L., "Venus Gravity Handbook," JPL Pub.96-2, Jan. 1996.
5. Striepe, S.A., Blanchard, R.C., Kirsch, M.F., Fowler, W.T., "Huygens Titan Probe Trajectory Reconstruction Using Traditional Methods and the Program to Optimize Trajectories II," AAS/AIAA Space Flight Mechanics Meeting, Sedona, AZ, 2007, AAS 07-226.
6. Muller-Wodarg, I. C.F., Yelle, R.V., Borggren, N., Waite, J.H., "Waves and Horizontal Structures in Titan's Thermosphere," *J. Geophysical Research*, Vol. 111, A12315, doi:10.1029/2006JA011061, 2006.
7. Keating, G.M., et al., "Models of Venus Neutral Upper Atmosphere: Structure and Composition," *Adv. Space Res.*, Vol. 5, No. 11, 1985, pp. 117-171.
8. Lyons, D.T., Beerer, J.G., Esposito, P., Johnston, M.D., "MGS: Aerobraking Mission Overview," *J. of Spacecraft and Rockets*, Vol. 36, No. 3, 1999, pp. 307-313.
9. Tolson, R.H., Keating, G.M., Cancro, G.J., Parker, J.S., Noll, S.N., and Wilkerson, B.L., "Application of Accelerometer Data to Mars Global Surveyor Aerobraking Operations," *J. Spacecraft and Rockets*, Vol. 36, No. 3, May-June, 1999, pp. 323-329.
10. Hanna, J.L. & Tolson, R.H., "Approaches to Autonomous Aerobraking at Mars," AAS/AIAA Astrodynamics Specialist Conference, Quebec City, Canada, July 30-August 2, 2001. AAS 01-387
11. Justus, C.G., Duvall, A., Keller, V.W., "Atmospheric Models for Aeroentry and Aeroassist," Proceedings of the 2nd International Planetary Probe Workshop, NASA Ames Research Center, Moffett Field, CA, pp 41-48, August 23-27, 2004. Also published in NASA/CP-2004-213456, April 2005.
12. Keating, G.M., et al., "Detection of Winter Polar Warming in Mars Upper Atmosphere," Paper PS1.02-1TH2A-006, EGS XXVII General Assembly, Nice, France, April 2002.
13. Dec, J. A., Gasbarre, J.F., George, B.E., "Thermal Analysis and Correlation of the Mars Odyssey Spacecraft's Solar Array During Aerobraking Operations," AIAA/AAS Astrodynamics Conference, Monterey CA, August 5-8, 2002.
14. Willcockson, W.H. & Johnson, M.A., "Mars Odyssey Aerobraking: The First Step Towards Autonomous Aerobraking Operations" 2003 IEEE Aerospace Conference, Big Sky, MT, March 9-14, 2003.
15. Bougher, S.J., Bell, J.M., Murphy, J.R., Lopez-Valverde, M.A., Withers, P.G., "Polar Warming in the Mars Thermosphere: Seasonal Variations Owing to Change in Insolation and Dust Distributions," *Geophysical Research Letters*, Vol. 33, 2006.
16. Wilson, R.J., "Evidence for Nonmigrating Thermal Tides in the Mars Upper Atmosphere from the MGS Accelerometer Exp.," *Geophys. Res. Lett.*, 27(21), 3563-3566, 2002.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 215 of 286	

17. Tolson, R.H., Keating, G.M., Noll, S.N., Baird, D.T., Shellenberg, T.J., "Utilization of Mars Global Surveyor Accelerometer Data for Atmospheric Modeling," *Astrodynamics 1999, Advances in the Astronautical Sciences*, pp. 1329-1346, 2000.
18. Dwyer, A.M., Tolson, R. H., Munk, M.M., Tartabini, P.V., "Development of a Monte Carlo Mars-GRAM Model for Mars 2001 Aerobraking Simulations," *J. of the Astronautical Sciences*, 2001, Vol. 109, pp 1293-1308.
19. Dec, J.A. & Thornblom, M.N., "Autonomous Aerobraking: Thermal Analysis and Response Surface Development," *AAS/AIAA Astrodynamics Specialist Conference*, July 31-August 3, 2011, Gildwood, Alaska, AAS 11-474.
20. Fritts, D.C., Wang, L., Tolson, R.H., "Mean and Gravity Wave Structured and Variability in the Mars Upper Atmosphere Inferred from Mars Global Surveyor and Mars Odyssey Aerobraking Densities," *J. Geophys. Res.*, Vol. 111, 2006.
21. Maddock, R.W., et al., "Implementation and Simulation Results Using Autonomous Aerobraking Development Software," *AAS/AIAA Astrodynamics Specialist Conference*, July 31-August 3, 2011, Gildwood, Alaska, AAS 11-476.
22. King-Hele, D.G., "Satellite Orbits in an Atmosphere: Theory and Applications," Blackie and Sons, Ltd., Glasgow and London, 1987, ISBN 0-216-92252-6.
23. Arakida, H. & Fukushima, T., "Long-Term Integration Error of K-S Regularized Orbital Motion," *Astronomical Journal*, Vol. 120, pp. 3333-3399, December 2000.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 216 of 286	

## Appendix G. Ephemeris Estimator User's Guide (Supplement to Section 7.3.2.2)

### G.1 Introduction


The Ephemeris Estimator is one of the models in the AADS software package intended for use on-board spacecraft that are placed in orbit around Venus, Mars, or Titan to implement AA. The Ephemeris Estimator provides the AADS a running estimate of the spacecraft orbital state through each atmospheric pass as well as a prediction of both the apoapsis and periapsis of the next orbit to be traversed. With occasional updates of the spacecraft orbital state, this information is provided orbit by orbit throughout the AB mission phase.

Written in C, the Ephemeris Estimator is a package of subroutines that integrates the orbit of a spacecraft in the gravitational field of its central body taking into account the gravitational effects of other relevant bodies including the Sun, the effects of solar radiation pressure, and the effects of orbit trim maneuvers and atmospheric drag as provided by accelerometer data gathered on-board the spacecraft. The integrator itself is an eighth order Runge-Kutta integrator with Fehlberg constants and seventh order automatic step sizing, which can be used for all integrations. Alternatively, a specified fixed step size can be used when integrating either or both of the orbit trim maneuvers and atmospheric drag accelerometer data types.

A list of C extern variables, most of which are associated with the AADS "iload" data, and the arguments of five of the Ephemeris Estimator routines comprise the complete data interface between the Ephemeris Estimator and the AADS. Many of the C extern variables are constants associated with the mission at hand, while the arguments to the five subroutines and the remaining C extern variables are running variables. The arguments can be further categorized as either inputs to the Ephemeris Estimator from the AADS or outputs from the Ephemeris Estimator to the AADS, but never both. There is a programming interface that includes providing the built-in data for the central body  $nxn$  gravitational arrays, i.e., the C and S matrices as well as the J array of zonal coefficients, and the requisite C code modifications that are needed to run the Ephemeris Estimator either as a standalone program or as part of the AADS. Two auxiliary subroutines are provided. One extracts the needed parts of natural body ephemeris files and the other unloads acceleration data from ASCII files generated by the AADS when it is used in a simulation. Both of these are needed as input arrays to the Ephemeris Estimator when used as a standalone program.

### G.2 Extern Variable Interface

Below is a table containing all C extern variables of this interface. At the top of the table are the 19 C extern variables that are associated with the specified AADS "iload" data structure element names, although one of these ("ephinit\_gm\_saturn") is only present when Titan is the central body. To the left of each equal sign ("=") is the C extern variable in the Ephemeris Estimator; to


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 217 of 286	

the right of each equal sign ("=") is the associated AADS "iload" data structure element name. Two of these element names ("ee\_ae\_re" and "ee\_ae\_rp") are divided by a thousand in the table to convert from meters to kilometers. At the bottom of the table are two C extern variables that have no associated AADS "iload" data structure element names. These control error handling and printing.

```

double ephinit_gm_sun          = ee_sun_mu;          /* Sun gravitational
*/                               constant [km^3/s^2]
double ephinit_gm_saturn      = ee_saturn_mu;       /* Saturn gravitational
*/                               constant [km^3/s^2]
Titan                           (only present when
*/                               is the central body)
double ephinit_gm              = cb_mu;              /* central body
constant                         gravitational
*/                               [km^3/s^2] */
double ephinit_alpha          = ee_ae_pole_ra;       /* right ascension of
cental                             body rotational pole
*/                               (EMEJ2000) [deg] */
double ephinit_delta          = ee_ae_pole_dec;      /* declination of central
*/                               body rotational pole
double ephinit_w              = ee_IAU_pm;          /* prime meridian with
*/                               respect to central
double ephinit_wdot           = ee_ae_omega;        /* central body rotation
*/                               rate [deg/day] */
double ephinit_radius          = ee_ae_re/1000.;     /* central body equatorial
*/                               radius [km] */
double ephinit_radius_polar   = ee_ae_rp/1000.;     /* central body polar
radius                             [km] */
double ephinit_radius_oblate  = ee_oblate_radius;   /* central body oblateness
*/                               radius [km] */
double ephinit_deltat_atm     = ee_deltat           /* time offset [s] from
*/                               atmosphere entry and
double ephinit_alt_atm        = ee_alt_atm;         /* bodydetic altitude to
*/                               use as atmospheric
double ephinit_stepsize       = ee_stepsize;        /* integrator step size
*/                               initial guess [s] */
double ephinit_relative_error = ee_relative_error;  /* integrator relative

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 218 of 286	

```

error [] */
double ephinit_absolute_error = ee_absolute_error; /* integrator absolute
error [] */

double ephinit_mnvr_step      = ee_mnvr_step;      /* fixed step size to use
during maneuvers [s]
(0. causes variable
step size) */
double ephinit_atmos_step     = ee_atmos_step;     /* fixed step size to use
during atmospheric
passes [s] (0.
causes
variable step size)
*/
double ephinit_sc_area        = sc_area;           /* spacecraft aerodynamic
reference area [m^2]
*/
double ephinit_sc_mass        = sc_mass;           /* spacecraft mass [kg] */
int ephdriver_error_handling; /* 0 = error off with
appropriate
message
(default)
1 = immediately return
with status code
*/
int ephdriver_print; /* 0 = no printed results
1 = interface argument
printed results
from "ephdriver"
(default)
2 = all printed results
with debugging
except from inside
of integrator
3 = all printed results
with debugging
even from inside
of integrator */


```

What follows are the built-in data for the C extern variables whose values are common to all three central bodies as specified by the "Autonomous Aerobraking Planetary Constants and Models" document [1].

```

ephinit_stepsize      = 60.; /* integrator step size
initial guess [s] */
ephinit_relative_error = 1.e-11; /* integrator relative
error [] */

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		219 of 286	

```

ephinit_absolute_error = 1.e-11; /* integrator absolute
                                  error [] */
ephinit_gm_sun          = 0.13271244004094400e+12; /* Sun gravitational
                                                    constant [km^3/s^2]
*/

```

What follows is the built-in value for the lone C extern variable that is only present when Titan is the central body as specified by the "Autonomous Aerobraking Planetary Constants and Models" document [1].

```

ephinit_gm_saturn      = 37931207.6129; /* Titan */ /* Saturn gravitational
                                                    constant [km^3/s^2]
*/

```

What follows are the built-in data for the C extern variables whose values differ by central body as specified by the "Autonomous Aerobraking Planetary Constants and Models" document [1].

```

ephinit_gm              = 42828.376212; /* Mars */ /* central body
ephinit_gm              = 8978.1394;   /* Titan */ /* gravitational
constant
ephinit_gm              = 324858.592079; /* Venus */ /* [km^3/s^2] */

ephinit_alpha          = 317.68143;    /* Mars */ /* right ascension of
cental
ephinit_alpha          = 36.41;        /* Titan */ /* body rotational pole
ephinit_alpha          = 272.76;      /* Venus */ /* (EMEJ2000) [deg] */


ephinit_delta          = 52.8865;     /* Mars */ /* declination of central
ephinit_delta          = 83.94;       /* Titan */ /* body rotational pole
ephinit_delta          = 67.16;      /* Venus */ /* (EMEJ2000) [deg] */

ephinit_w              = 176.630;     /* Mars */ /* prime meridian with
ephinit_w              = 189.64;     /* Titan */ /* respect to central
ephinit_w              = 160.20;     /* Venus */ /* body IAU vector at
epoch [deg] */

ephinit_wdot           = 350.89198226; /* Mars */ /* central body rotation
ephinit_wdot           = 22.5769768;  /* Titan */ /* rate [deg/day] */
ephinit_wdot           = -1.4813688;  /* Venus */ /*

ephinit_radius         = 3396.19;     /* Mars */ /* central body equatorial
ephinit_radius         = 2575.;       /* Titan */ /* radius [km] */
ephinit_radius         = 6051.8;     /* Venus */ /*

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 220 of 286	

```

ephinit_radius_polar = 3376.20;      /* Mars */ /* central body polar
radius
ephinit_radius_polar = 2575.;      /* Titan */ [km] */
ephinit_radius_polar = 6051.8;     /* Venus */

ephinit_radius_oblate = 3396.2;     /* Mars */ /* central body oblateness
ephinit_radius_oblate = 2575.;     /* Titan */ radius [km] */
ephinit_radius_oblate = 6051.;     /* Venus */

ephinit_alt_atm      = 200.;        /* Mars */ /* bodydetic altitude to
ephinit_alt_atm      = 1000.;      /* Titan */ use as atmosphere
ephinit_alt_atm      = 200.;        /* Venus */ interface [km] */

```

The "ephinit\_radius\_polar" for Mars is the average of two polar radii, namely the one for the Mars North Pole and the one for the Mars South Pole. Also, "ephinit\_gm" and "ephinit\_radius\_oblate" are determined by the choice of gravitational field arrays (C and S matrices as well as the J array of zonals) and are specified along with these array values in routine "oblateness\_perturbation.c" as discussed in the "Programming Interface" section of this document below. Any attempt to override the specified values of "ephinit\_gm" and "ephinit\_radius\_oblate" will result in an error.

The two C extern variables "ephinit\_sc\_area" and "ephinit\_sc\_mass" are spacecraft dependent. Despite this fact, they are currently set by default to 37.12 [m<sup>2</sup>] and 1395 [kg], respectively, in the "ephinit" routine. These are the values associated with the APL Messenger spacecraft that was used in the development of this software.

All other C extern variables associated with "iload" elements default to a value of zero. The two C extern values not associated with "iload" elements default to a value of one as indicated in the table at the top of this section. To override any of the default values of these C extern variables when using the Ephemeris Estimator as a standalone program, see the "Programming Interface" section of this document below.


### G.3 Subroutine Interface

What follows is a description of each of the five routines of the subroutine interface along with paragraph descriptions of how those routines are to be called with respect to one another.

#### (1) Initialize or re-initialize natural body ephemeris data using routine "ephinit"

The first call to this routine must occur before the first call to the "ephupd" routine. Arguments "plan\_eph", "n\_plan\_eph", "sat\_eph", and "n\_sat\_eph" are not present when executing with Mars or Venus as the central body since currently "plan\_eph" is only needed for Saturn when Titan is the central body and "sat\_eph" is only needed for Titan itself. The lack of need for




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 221 of 286	

explicit planetary ephemerides for Venus and Mars is because the barycenter and real center of both Venus and Mars are considered to be co-located. There is a slight offset of the Mars barycenter from its real center due to its two moons Phobos and Deimos, but the offset is small. This means that the associated barycenter data with respect to the Solar System Barycenter is all that is needed for these central bodies.

```

/* cause initialization or re-initialization
of natural body ephemeris data --
output status (0 = outright success,
<0 = outright failure and the particular
negative value indicates where the
failure
occurred, see "Error Codes" section
below) */
int ephinit(sun_eph,n_sun_eph,bary_eph,n_bary_eph,plan_eph,n_plan_eph,
sat_eph,n_sat_eph)
double sun_eph[]; /* input Sun Chebyshevs (see "Format of the
Chebyshev Ephemeris Arrays" section
below) (SPK Type-2 Records) */
int n_sun_eph; /* input number of Developmental Ephemeris
array elements of the Sun */
double bary_eph[]; /* input planetary barycenter Chebyshevs
(see "Format of the Chebyshev Ephemeris
Arrays" section below) (SPK Type-2
Records) */
int n_bary_eph; /* input number of Developmental Ephemeris
array elements of the planetary
barycenter */
double plan_eph[]; /* input planet Chebyshevs (if needed)
(see "Format of the Chebyshev Ephemeris
Arrays" section below) (SPK Type-3
Records) */
int n_plan_eph; /* input number of "satellite ephemeris"
array elements of the associated
planet */
double sat_eph[]; /* input satellite Chebyshevs (if needed)
(see "Format of the Chebyshev Ephemeris
Arrays" section below) (SPK Type-3
Records) */
int n_sat_eph; /* input number of "satellite ephemeris"
array elements of the associated
satellite */

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 222 of 286	


## (2) Initialize or re-initialize spacecraft orbital state and epoch using routine "ephupd0"

The first call to this routine must occur before the first call to the "ephupd" routine. The epoch of the orbital state provided in any call to this routine must precede the time tag associated with the first acceleration datum in the first call to the "ephupd" routine that occurs thereafter, presuming there is any acceleration data provided in that call to "ephupd". The epoch of the orbital state provided in subsequent calls to this routine, where all calls other than the first call to this routine are the subsequent calls, must be greater than or equal to the ephemeris time of the end of the atmospheric pass at the end of the current orbit if there was an atmospheric pass, or greater than or equal to the ephemeris time of the periapsis at the end of the current orbit if there was no atmospheric pass. Not to be confused, this is the situation that exists after the most recent previous call to the "ephupd" routine.

```

/* cause initialization or re-initialization
   of spacecraft orbital state --
   output status (0 = outright success,
   <0 = outright failure and the particular
   negative value indicates where the
   failure occurred, see "Error codes"
   section below) */
int ephupd0(et_entry, reentry, et_peri, rperi, rperir, rperi_dalt, rperi_dlat,
  rperi_lat, et_exit, rexit, et0, x0)
double *et_entry;          /* output ephemeris time associated with next
                           atmospheric entry (seconds past J2000)
*/
double reentry[6];        /* output orbital state vector associated
                           with next atmospheric entry [m, m/s] in
                           Earth mean equator and equinox of J2000
                           coordinates */
double *et_peri;          /* output ephemeris time of next periapsis
                           (seconds past J2000) */
double rperi[6];          /* output orbital state vector of next
                           periapsis [m, m/s] in Earth mean equator
                           and equinox of J2000 coordinates */
double rperir[6];         /* output orbital state vector of next
                           periapsis [m, m/s] in central body
                           equator and prime meridian of date
                           (rotating) coordinates */
double *rperi_dalt;       /* output bodydetic altitude ("gdalt" in
POST)
                           of next periapsis [m] */
double *rperi_dlat;       /* output bodydetic latitude ("gdlat" in
POST)
                           of next periapsis [deg] */
double *rperi_lat;        /* output bodycentric latitude ("decln" in

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 223 of 286	

```

double *et_exit;          POST) of next periapsis [deg] */
                          /* output ephemeris time associated with next
                          atmospheric exit (seconds past J2000)
*/
double rexit[6];         /* output orbital state vector associated
                          with next atmospheric exit [m, m/s] in
                          Earth mean equator and equinox of J2000
                          coordinates */
double et0;              /* input epoch ephemeris time
                          (seconds past J2000) */
double x0[6];            /* input spacecraft orbital state [m, m/s]
                          at epoch in Earth mean equator and
                          equinox of J2000 coordinates */


```

### **(3) Integrate forward one orbit with or without accelerometer data using routine "ephupd"**

Presuming that routine "ephupd" has been previously called, then a call to routine "ephupd" will integrate forward one complete orbit, the so-called "current orbit", which ends at the so-called "current periapsis", or at the point on the orbit just beyond the "current periapsis" where the atmosphere is declared to have been exited. This occurs while possibly processing one or more time-wise contiguous sets of tabular acceleration data. Then routine "ephupd" will integrate beyond the end of the "current orbit" to predict the following apoapsis, the so-called "next apoapsis", followed by the periapsis after that, the so-called "next periapsis". If this is the first call to "ephupd" after a call to routine "ephupd0", then "ephupd" will treat the epoch and orbital state provided by the call to "ephupd0" as defining the "current orbit" no matter where the orbital state is on that orbit. In this situation, "ephupd" will only integrate from that epoch to the end of the "current orbit" before integrating to predict the following apoapsis and subsequent periapsis.

Routine "ephupd" must be called exactly once per orbit and this call should be made as soon as possible after the pass through the atmosphere, with its associated gathering of accelerometer data, has been completed, or if there is no pass through the atmosphere, then as soon as possible after passing through periapsis. In other words, the orbit to be predicted begins at the end of the atmospheric pass at the end of the "current orbit" or at the periapsis at the end of the "current orbit", i.e., the "current periapsis", if there is no atmospheric pass.

All tabular acceleration data provided to this routine must be in chronological order. If there are no maneuvers and no atmospheric pass, i.e., no tabular acceleration data, then simply set integer argument "n\_data" to the literal 0 or a zero-valued integer variable and set double precision arguments "et" and "acc" to the literal 0, or zero-valued double precision variables. If argument "n\_zero" is not set to zero, then the acceleration data can measure the effects for any of the following scenarios: maneuver(s) only, an atmospheric pass only, or maneuver(s) followed by an atmospheric pass. If there are both maneuver(s) and an atmospheric pass in the acceleration data, then the last time-wise contiguous set of tabular acceleration data provided on a given orbit must be the acceleration data associated with the atmospheric pass, i.e., any maneuver(s)

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 224 of 286	

measured in the acceleration data must precede the atmospheric pass. The only acceleration data that can extend in time beyond this "current periapsis" is the acceleration data associated with the atmospheric pass. The acceleration data associated with the atmospheric pass *must* extend beyond the "current periapsis", as this is what identifies that particular set of tabular acceleration data as being an atmospheric pass as opposed to a maneuver. It is possible to contain a maneuver within an atmospheric pass, an atmospheric pass within a maneuver, or one can lead into the other; but a maneuver surrounding "current periapsis" that is not within an atmospheric pass will lead to the creation of data earmarked as being part of an atmospheric pass when it is not, and this will cause bookkeeping problems in other AA routines when the Ephemeris Estimator is used as part of the AADS. Furthermore, no time-wise contiguous set of tabular acceleration data is permitted to start after the "current periapsis", as that would be in the subsequent, i.e., "next" orbit. This routine can only handle one orbit at a time.

```

/* integrate forward one complete orbit
   the so-called "current orbit" which
   ends at the so-called "current
periapsis"

or at the point where the atmosphere
is exited that is associated with the
"current periapsis", while possibly
processing one or more time-wise
contiguous sets of tabular acceleration
data and then integrate beyond that to
predict the following apoapsis, the
so-called "next apoapsis", and

the so-called "next periapsis" --
output status (2 = surface intercept,
1 = altitude intercept, 0 = outright
success, <0 = outright failure to

complete

particular

negative value indicates where in the
source code that the failure occurred,
see "Error Codes" section below) */
int ephupd(et_peri0,rperi0,rperi0_dalt,rperi0_dlat,rperi0_lat,et_apo,rapo,
  et_entry,rentry,et_peri,rperi,rperir,rperi_dalt,rperi_dlat,rperi_lat,
  et_exit,rexit,altmin,n_data,et,acc)
double *et_peri0;          /* output ephemeris time of current periapsis
                           (seconds past J2000) */
double rperi0[6];         /* output orbital state vector of current
                           periapsis [m, m/s] in Earth mean equator
                           and equinox of J2000 coordinates */
double *rperi0_dalt;      /* output bodydetic altitude ("gdalt" in
POST)

```



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**


Title:

## Autonomous Aerobraking (Phase 1)

Page #:

225 of 286

```
double *rperi0_dlat;    /* of current periapsis [m] */
POST)                  /* output bodydetic latitude ("gdlat" in
                        of current periapsis [deg] */
double *rperi0_lat;    /* output bodycentric latitude ("decln" in
                        POST) of current periapsis [deg] */
double *et_apo;        /* output ephemeris time of next apoapsis
                        (seconds past J2000) */
double rapo[6];        /* output orbital state vector of next
                        apoapsis [m, m/s] in Earth mean equator
                        and equinox of J2000 coordinates */
double *et_entry;      /* output ephemeris time associated with next
                        atmospheric entry (seconds past J2000)
*/
double rentry[6];      /* output orbital state vector associated
                        with next atmospheric entry [m, m/s] in
                        Earth mean equator and equinox of J2000
                        coordinates */
double *et_peri;       /* output ephemeris time of next periapsis
                        (seconds past J2000) */
double rperi[6];       /* output orbital state vector of next
                        periapsis [m, m/s] in Earth mean equator
                        and equinox of J2000 coordinates */
double rperir[6];      /* output orbital state vector of next
                        periapsis [m, m/s] in central body
                        equator and prime meridian of date
                        (rotating) coordinates */
double *rperi_dalt;    /* output bodydetic altitude ("gdalt" in
                        POST)
                        of next periapsis [m] */
double *rperi_dlat;    /* output bodydetic latitude ("gdlat" in
                        POST)
                        of next periapsis [deg] */
double *rperi_lat;    /* output bodycentric latitude ("decln" in
                        POST) of next periapsis [deg] */
double *et_exit;       /* output ephemeris time associated with next
                        atmospheric exit (seconds past J2000) */
double rexit[6];       /* output orbital state vector associated
                        with next atmospheric exit [m, m/s] in
                        Earth mean equator and equinox of J2000
                        coordinates */
double altmin;         /* input minimum acceptable bodydetic
                        altitude that will cause the altitude
                        intercept output status of this routine
                        to be set if the periapsis of the "next
                        orbit" dips below this value [m] */
int n_data;            /* input number of ephemeris time tags and
                        acceleration 3-vectors */
double et[];           /* input ephemeris time tag array associated
                        with the tabular acceleration data
```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 226 of 286	

```

double acc[][3];          (seconds past J2000) */
                          /* input tabular acceleration data array
                          [m/s^2] */

```

#### (4) Return atmospheric pass orbital states and bodydetic altitudes using routine “ephاتم”

The first call to this routine must occur after the first call to the "ephupd" routine. The results from this routine are always the results associated with the most recent previous call to “ephupd”.

```

value
                          /* return entry epoch and both the spacecraft
                          orbital states and the associated
                          bodydetic altitudes through the
                          atmospheric pass -- output status
                          (0 = outright success, <0 = outright
                          failure and the particular negative
                          indicates where the failure occurred,
                          see "Error Codes" section below) */

int ephاتم(etاتم0, xاتم, aاتم, n_اتم)
double *etاتم0;          /* output ephemeris time associated with
                          atmospheric entry (seconds past J2000)
*/
double xاتم[][6];        /* output spacecraft orbital states [m, m/s]
                          in Earth mean equator and equinox of
                          J2000 coordinates */
double aاتم[];          /* output spacecraft bodydetic altitudes
                          [m] */
int *n_اتم;             /* output number of spacecraft orbital states
                          and associated bodydetic altitudes [] */

```


#### (5) Return number of orbital states and altitudes available in “ephاتم” using routine “ephاتم0”

```

through
                          /* return number of spacecraft orbital states
                          and associated bodydetic altitudes
                          the atmospheric pass - output status
                          (0 = success and !0 = failure) */

int ephاتم0(n_اتم)
int *n_اتم;             /* output number of spacecraft orbital states
                          and associated bodydetic altitudes [] */

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 227 of 286	

## G.4 Format of the Chebyshev Ephemeris Arrays

The "sun\_eph", "bary\_eph", "plan\_eph", and "sat\_eph" arguments to routine "ephinit" are concatenated one-dimensional arrays with the following format. Yes, the value of array[4], array[4+m+1], ..., array[4+(n-1)\*(m+1)] is the exact same value represented here by m (the number of double precision words per record) for any given body, although the value of m does vary between bodies. The number of contiguous records being stored is represented here by n. This repetitious array loading is done to make each contiguous set of m+1 array elements that begins with one of those duplicate elements be exactly what is expected in the input argument of the appropriate SPK ephemeris record evaluation routine associated with that data. The appropriate SPK ephemeris record evaluation routine for "sun\_eph" and "bary\_eph" data, which always comes from a Development Ephemeris (DE) file, is "spke02" which handles Chebyshev polynomials of position only data, and the appropriate SPK ephemeris record evaluation routine for "plan\_eph" and "sat\_eph" ephemeris data, which always comes from a so-called "satellite ephemeris" file, is "spke03" which handles Chebyshev polynomials of position and velocity data. Both evaluation routines return body orbital state at the specified ephemeris time. In the case of "spke03", the body orbital state is solved for directly from the 6 sets of Chebyshev coefficients that are present. While in the case of "spke02", the body position is solved for directly from the 3 sets of Chebyshev coefficients that are present, and the body velocity is solved for directly as the derivative of those same 3 sets of Chebyshev coefficients. The ability to readily get the derivative of a set of Chebyshev coefficients is one of the fundamental properties of Chebyshev polynomials.


```

array[0]           = beginning ephemeris time (seconds past J2000)
array[1]           = time span of each data record [s]
array[2]           = number of contiguous data records present
array[3]           = (unused)
array[4]           = number of double precision words per record (m)
array[4+1]-array[4+m] = 1st Chebyshev record data words
array[4+m+1]       = number of double precision words per record (m)
array[4+m+1+1]-array[4+2*(m+1)-1]
                  = 2nd Chebyshev record data words
.
.
.
array[4+(n-1)*(m+1)] = number of double precision words per record (m)
array[4+(n-1)*(m+1)+1]-array[4+n*(m+1)-1]
                  = nth Chebyshev record data words

```

## G.5 Error Codes

The following is a list of error messages which are prefixed by the associated error codes followed by the associated routines where the errors will have taken place. The error code will be returned to the highest level calling routine if C extern variable "ephdriver\_error\_handling" is

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>			Page #: 228 of 286


set to 0, i.e., error code -2 will be returned by "ephadm", error codes -3 through -11 will be returned by "ephinit", and all other error codes will be returned by "ephupd" or "ephupd0". Otherwise, the error message will be printed and then "exit(1)" will be executed at the point in the code where the error is detected. The occasional "%d" in these error messages will be replaced with an integer value if and when the error message is printed, and the "\n" in the error code -32 message will cause a new line to be generated at that location.

```

-1: addrot: bad axis = %d input
-2: ephadm: ephupd has not yet been called
-3: ephinit: illegal central body number
-4: ephinit: sun ephemeris array is missing
-5: ephinit: sun ephemeris array is too long
-6: ephinit: barycenter ephemeris array is missing
-7: ephinit: barycenter ephemeris array is too long
-8: ephinit: planet ephemeris array is missing
-9: ephinit: planet ephemeris array is too long
-10: ephinit: satellite ephemeris array is missing
-11: ephinit: satellite ephemeris array is too long
-12: ephupd: ephinit has not yet been called
-13: ephupd: ephupd0 has not yet been called
-14: ephupd: acc and et array arguments are too long to be stored
-15: ephupd: illegal value of specified array sizes
-16: ephupd: integration would start before beginning of ephemeris
-17: ephupd: unsupported central body number %d detected
-18: ephupd: bad time tag data in acceleration data table
-19: ephupd: acceleration data starts before current orbit
-20: ephupd: last acceleration data set is entirely beyond current periapsis
-21: ephupd: last acceleration single datum is beyond current periapsis
-22: ephupd: integration attempting to go off end of ephemeris
-23: ephupd0: ephinit has not yet been called
-24: ephupd0: integration would start before beginning of ephemeris
-25: ephupd0: unsupported central body number %d detected
-26: ephupd0: new epoch is before current epoch in ephupd
-27: ephupd0: integration attempting to go off end of ephemeris
-28: integ: imeth = %d not supported
-29: integ: imeth = %d should never have occurred
-30: integ: illegal imeth = %d
-31: integ: this should not happen (imeth = %d)
-32: integ: integrator choked \n iflag = %d
-33: oblateness_perturbation: could not open VENUSGRV
-34: oblateness_perturbation: Venus MAX_J = %d exceeded
-35: oblateness_perturbation: Venus MAX_CS = %d exceeded
-36: oblateness_perturbation: could not open MARSGRV
-37: oblateness_perturbation: Mars MAX_J = %d exceeded
-38: oblateness_perturbation: Mars MAX_CS = %d exceeded
-39: oblateness_perturbation: could not open TITANGRV
-40: oblateness_perturbation: Titan MAX_J = %d exceeded
-41: oblateness_perturbation: Titan MAX_CS = %d exceeded

```



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 229 of 286	

```

-42: oblateness_perturbation: bad central body number %d
-43: oblateness_perturbation: MAX_J = %d exceeded
-44: oblateness_perturbation: MAX_CS = %d exceeded
-45: oblateness_perturbation: ephinit_gm wrongly altered
-46: oblateness_perturbation: ephinit_radius_oblate wrongly altered
-47: plsang: bad body number
<-47: ephupd: = -int(10.*ephemeris_time_tag) to indicate apparent hole in the
      last acceleration data set

```

## G.6 Programming Interface

### G.6.1 J array, C matrix, and S matrix data

There are three sets of central body gravitational field arrays that are hard coded into file "oblateness\_perturbation.c" according to the central body at hand. These are the normalized C and S matrices as well as the normalized J array of zonals. The arrays associated with this data are loaded in such a way that the degree and order can be readily truncated to any size from 2 to the maximum degree and maximum order available with the unused portions of the arrays being ignored. Each J term is immediately followed by the associated C and S matrix terms as follows.


```

J(n)
C(n,1) S(n,1)
C(n,2) S(n,2)
.
.
.
C(n,n) S(n,n)

```

The exact values of both the central body gravitation C extern variable "ephinit\_gm" and the oblateness radius C extern variable "ephinit\_radius\_oblate" that were identified in the "C Extern Variable Interface" section are mandated by the choice of these gravitational field arrays, i.e., all three arrays, the central body gravitation, and the oblateness radius come as a set. So, neither the central body nor the oblateness radius should be superseded with alternative values, and it is treated as an error if they are.

"The The latest Venus gravity field is MGNP180U, which is a 180 degree and order model based on the IAU 1991 Venus pole and prime meridian locations and a reference radius of 6051.0 km. This gravity field was developed using data collected from the Magellan mission. The file can

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 230 of 286	

be downloaded from the following site: [http://pds-geosciences.wustl.edu/geo/mgn-v-rss-5-gravity-12-v1/mg\\_5201/gravity/shgj180u.a01](http://pds-geosciences.wustl.edu/geo/mgn-v-rss-5-gravity-12-v1/mg_5201/gravity/shgj180u.a01)."

"The most recent Mars gravity field model is MRO110B. Because the MRO110B gravity field model requires a new orientation model (which requires extensive software updates), the MGS85F2 gravity field is being used. The MGS85F2 gravity field is an 85 degree and order model based on the IAU 2000 [2] Mars pole and prime meridian locations and a reference radius of 3396.2 km. It is the last Mars gravity field model to use the IAU orientation. This gravity field was developed using data collected from Mariner 9, Viking 1 and 2, and MGS mapping through Nov. 18, 2001. The file can be downloaded from the following site: [http://pds-geosciences.wustl.edu/geo/mgs-m-rss-5-sdp-v1/mors\\_1024/sha/jgm85f02.sha](http://pds-geosciences.wustl.edu/geo/mgs-m-rss-5-sdp-v1/mors_1024/sha/jgm85f02.sha)."

"The latest Titan gravity field is a 3 degree and order model referred to as the SOL2 approach. It is based on radiometric tracking and optical navigation imaging data from the Cassini mission combined with data from the Pioneer and Voyager missions, as well astronomical observations of Saturn and its satellites [3]."


The three paragraphs of quoted text are from the "Autonomous Aerobraking Planetary Constants and Models" document [1]. The only alterations made thereto are the reference numbers which have been adjusted to be the reference numbers used in this document.

## G.6.2 Inclusion and exclusion of code by choice of central body

The inclusion and exclusion of blocks of code based on the central body at hand is effected with the value assigned to macro CBOD in include file "ephest.h". The relevant values of CBOD are 299 for Venus, 499 for Mars, and 606 for Titan. File "ephest.h" must be edited to assign the correct central body number to CBOD, and then all routines that cause file "ephest.h" to be included must be re-compiled. The largest of these blocks of code is the implementation of the C and S matrices as well as the J array in file "oblateness\_perturbation.c". Lesser examples including the assignment of default values to C extern variables appear in files "aads\_blkdat.h", "eevardefine.c", "ephbuild.c", "ephdriver.c", "ephest.h", "ephload.c", "ephinit.c", "ephupd.c", "ephupd0.c", "fauto.c", and "plsang.c".

## G.6.3 Specifying Whether to Treat code as Standalone or as part of the AADS

The determination of whether to treat the code as a standalone or as a part of the AADS is totally determined by the value assigned to macro STANDALONE in include file "ephest.h". Except for the specification of central body in macro CBOD as described and adjustments to the main program when the code is treated as standalone, *no other modifications to the Ephemeris Estimator source code are needed or should occur*. The relevant values of STANDALONE are 1 for yes and 0 for no. If the code is treated as part of the AADS, then the various C extern variables associated with the "iload" data structure, as specified in the first part of the "C Extern Variable Interface" section of this document, are updated before each orbit is

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 231 of 286	

integrated with the contents of the “iload” data structure. The appropriate “iload” data structure for the central body being Mars or Venus is the first include file below. The appropriate “iload” data structure for the central body being Titan is obviously the second include file below.

aads\_iloards.h  
aads\_iloards\_titan.h

The “iload” updates all take place in routine “ephiload” as called by routine “ephupd0”, or in routine “ephupd” if routine “ephupd0” is not called on a given orbit. If the Ephemeris Estimator code is treated as standalone, then these C extern variables are initially set to their built-in values, as specified in the second, third, and fourth parts of the “C Extern Variable Interface” section of this document, and are never altered after that, except explicitly by the main program. Manipulation of C extern variables by the main program is described next.


#### **G.6.4 Establishing and Possibly Overriding Default Values of C Extern Variables**

To override one or more of the default values described in the “C Extern Variable Interface” section of this document, file “ephest.h” must be included in the main program, and the values in question must be explicitly set before the first call to “ephinit” occurs. After that, these values can continue to be altered as needed.

#### **G.6.5 Generation of Natural Body Ephemeris Arrays using Auxiliary Routine “Ephbuild”**

The generation of natural body ephemeris arrays is effected by calling auxiliary routine “ephbuild”. Arguments “plan”, “n\_plan”, “sat”, and “n\_sat” are not present when executing at Mars and Venus. The setting of macro CBOD to one of the three body numbers mentioned instructs routine “ephbuild” as to which natural body ephemeris arrays to generate.

```
int ephbuild(sun,n_sun,bary,n_bary,plan,n_plan,sat,n_sat,et0,etf)
double sun[];          /* output Sun with respect to Solar System Barycenter
                        ephemeris array */
int *n_sun;           /* output number of double precision words in sun[]
*/
double bary[];        /* output Venus, Mars system barycenter, or Saturn
                        system barycenter with respect to the Solar
                        System Barycenter ephemeris array */
int *n_bary;          /* output number of double precision words in bary[]
*/
double plan[];        /* output Saturn with respect to Saturn system
                        barycenter ephemeris array */
int *n_plan;          /* output number of double precision words in plan[]
*/
```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 232 of 286	

```

double sat[];          /* output Titan with respect to Saturn system
                       barycenter ephemeris array */
int *n_sat;           /* output number of double precision words in sat[]
*/
double et0;           /* input requested ephemeris time start of all
natural
                       body ephemeris arrays (seconds past J2000) */
double etf;           /* input requested ephemeris time end of all natural
                       body ephemeris arrays (seconds past J2000)
                       -- not currently used */

```

### G.6.6 Extraction of Time-tagged Acceleration Data from a File using Auxiliary Routine “Ephunload”

When the Ephemeris Estimator is run in standalone mode, all acceleration data is relayed to the Ephemeris Estimator in ASCII files. Auxiliary routine “ephunload” extracts this data from one of those files that is always named “ee\_accel.dat”, and returns it in three variables: a time-tag array, a Cartesian acceleration array, and a variable that specifies the number of acceleration records that are present. Macro MAX\_ACC, which is used to dimension the arrays, is specified in include file “ephst.h”. These three variables are the form in which acceleration data is to be delivered to routine “ephudp” either directly or via routine “ephdriver”.

```

void ephunload(et, acc, n_data)
double et[MAX_ACC];          /* input ephemeris time tag array associated
                             with the tabular acceleration data
                             (seconds past J2000) */
double acc[MAX_ACC][3];     /* input tabular acceleration data array
                             [m/s^2] */.
int *n_data;                /* input number of ephemeris time tags and
                             acceleration 3-vectors */


```

## G.7 Terminology

The word “bodydetic” is used throughout this document instead of the Earth relative word “geodetic” or the planet relative word “planetodetic”. Given that the Earth is not used as a central body in this document and given that Titan is a natural satellite of a planet but not a planet itself, a more generic word was needed.

## G.8 Future Work

There needs to be additional work on the handling of acceleration tables by the integrator. There is a need for additional gravitating bodies, with Jupiter being included when Mars is the central body being a prime example. This in particular will alter the interfaces to routines “ephbuild” and “ephinit” as well as the code in routine “fauto.” Finally, the solar radiation pressure model needs to be modified to include penumbra effects.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 233 of 286	

### G.8.1 Example 1: Overall Interface Routine “Ephdriver”—a Subroutine Interface Example

This routine was originally developed by the AADS to combine the five Ephemeris Estimator interface routines into one overall interface routine. The main program examples in the below example 2 and example 3 sections both make use of this routine.

The only argument of "ephdriver" below not already described in the interfaces to the five routines in the “Subroutine Interface” section is the first argument "init\_orb". Otherwise, the routine or routines named in the comment that immediately follows each of the other arguments of "ephdriver" below are the associated interface routines where that argument is described.

```

/*
 * Copyright 2011, KinetX, Inc. This software is developed as freeware and
 * may be freely reproduced, distributed, and used by anyone as long as it is
 * not used for profit and as long as any derivative works are also freeware.
 */

/* Written by Robert W. Maddock with occasional modifications by
 * David L. Skinner */

#include <stdio.h>
#include <stdlib.h>

#include "ephest.h"

void ephdriver(

/* input: initialization control flag and state data */
init_orb, tepoch, x,

/* input: natural body ephemeris data */
sun, n_sun, bary, n_bary,
#ifdef CBOD == 606
  plan, n_plan, sat, n_sat,
#endif

/* input: acceleration data */
et, acc, n_data,

/* output: previous "current" periapsis data */
et_peri0, rperi0_dalt,

/* output: predicted "next" apoapsis data */
et_apo, rapo,

/* output: predicted "next" periapsis and atmosphere entry/exit data */
et_entry, reentry, et_peri, rperi, rperir, rperi_dalt, et_exit, rexit,

```



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

**Autonomous Aerobraking (Phase 1)**

Page #:

234 of 286


```
/* output: data for Atmosphere Estimator (AtmosEst) */
n_atm, altitudes_atm, states_atm )

int      *init_orb; /* input control flag where
                    0 do not call routines "ephinit" and "ephupd0"
                    1 call routines "ephinit" and "ephupd0"
                    output is the reset value of the flag which is 0 */

double tepoch;      /* et0 in ephupd0 */
double x[6];        /* x0 in ephupd0 */
double sun[];       /* ephinit */
int      n_sun;     /* ephinit */
double bary[];      /* ephinit */
int      n_bary;    /* ephinit */
#if CBOD == 606
    double plan[];  /* ephinit */
    int      n_plan; /* ephinit */
    double sat[];   /* ephinit */
    int      n_sat;  /* ephinit */
#endif
double et[];        /* ephupd */
double acc[][3];    /* ephupd */
int      n_data;    /* ephupd */
double *et_peri0;   /* ephupd */
double *rperi0_dalt; /* ephupd */
double *et_apo;     /* ephupd */
double rapo[6];     /* ephupd */
double *et_entry;   /* ephupd */
double reentry[6];  /* ephupd0, ephupd */
double *et_peri;    /* ephupd0, ephupd */
double rperi[6];    /* ephupd0, ephupd */
double rperir[6];   /* ephupd0, ephupd */
double *rperi_dalt; /* ephupd0, ephupd */
double *et_exit;    /* ephupd0, ephupd */
double rexit[6];    /* ephupd0, ephupd */
double aatm[];      /* ephatm */
double xatm[][6];   /* ephatm */
int      *n_atm;    /* ephatm0, ephatm */
{
    double altmin=0;
    double etatm0;

    double rperi0[6];
    double rperi0_dlat;
    double rperi0_lat;
    double rperi_dlat;
    double rperi_lat;

    int i;
}
/*
```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 235 of 286	

```

int iflag;
*/
int iistat;
int init;
int istat;

static int first=1;

/***** External references *****/
extern int ephatm0();
extern int ephatm();
extern int ephinit();
extern int ephupd0();
extern int ephupd();

init=(*init_orb);
*init_orb=0;

if (first || (init-10*(init/10))) {
    first=0;
    istat=ephinit(sun,n_sun,bary,n_bary
    #if CBOD == 606
        ,plan,n_plan,sat,n_sat
    #endif
    );
    if (ephdriver_print>=1)
        printf("ephinit=%d\n",istat);
    if (istat!=0)
        exit(1);
}

if (init-10*(init/10)) {
    istat=ephupd0(et_entry,rentry,et_peri,rperi,rperir,rperi_dalt,
    &rperi_dlat,&rperi_lat,et_exit,rexitepoch,x);
    if (ephdriver_print>=1)
        printf("ephupd0=%d\n",istat);
    if (istat!=0)
        exit(1);

    if (ephdriver_print>=1) {
        printf("next atmospheric entry: %f %f %f %f %f %f %f\n",*et_entry,
        rentry[0],rentry[1],rentry[2],rentry[3],rentry[4],rentry[5]);
        printf("next periapsis: %f %f %f %f %f %f %f\n",*et_peri,
        rperi[0],rperi[1],rperi[2],rperi[3],rperi[4],rperi[5]);
        printf("next periapsis rotating: %f %f %f %f %f %f\n",
        rperir[0],rperir[1],rperir[2],rperir[3],rperir[4],rperir[5]);
        printf("next periapsis bodydetic altitude: %f\n",*rperi_dalt);
        printf("next periapsis bodydetic latitude: %f\n",rperi_dlat);
        printf("next periapsis bodycentric latitude: %f\n",rperi_lat);
        printf("next atmospheric exit: %f %f %f %f %f %f %f\n",*et_exit,

```



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

236 of 286

```
        rexit[0], rexit[1], rexit[2], rexit[3], rexit[4], rexit[5]);
    }
}


iistat=ephupd(et_peri0, rperi0, rperi0_dalt, &rperi0_dlat, &rperi0_lat, et_apo,
    rapo, et_entry, reentry, et_peri, rperi, rperir, rperi_dalt, &rperi_dlat,
    &rperi_lat, et_exit, rexit, altmin, n_data, et, acc);
if (ephdriver_print>=1)
    printf("ephupd=%d\n", iistat);
if (iistat!=0)
    exit(1);

if (ephdriver_print>=1) {
    printf("current periapsis: %f %f %f %f %f %f %f\n", *et_peri0,
        rperi0[0], rperi0[1], rperi0[2], rperi0[3], rperi0[4], rperi0[5]);
    printf("current periapsis bodydetic altitude: %f\n", *rperi0_dalt);
    printf("current periapsis bodydetic latitude: %f\n", rperi0_dlat);
    printf("current periapsis bodycentric latitude: %f\n", rperi0_lat);
    printf("next apoapsis: %f %f %f %f %f %f %f\n", *et_apo,
        rapo[0], rapo[1], rapo[2], rapo[3], rapo[4], rapo[5]);
    if (iistat==0) {
        printf("next atmospheric entry: %f %f %f %f %f %f %f\n", *et_entry,
            reentry[0], reentry[1], reentry[2], reentry[3], reentry[4], reentry[5]);
        printf("next periapsis: %f %f %f %f %f %f %f\n", *et_peri,
            rperi[0], rperi[1], rperi[2], rperi[3], rperi[4], rperi[5]);
        printf("next periapsis rotating: %f %f %f %f %f %f\n",
            rperir[0], rperir[1], rperir[2], rperir[3], rperir[4], rperir[5]);
        printf("next periapsis bodydetic altitude: %f\n", *rperi_dalt);
        printf("next periapsis bodydetic latitude: %f\n", rperi_dlat);
        printf("next periapsis bodycentric latitude: %f\n", rperi_lat);
        printf("next atmospheric exit: %f %f %f %f %f %f %f\n", *et_exit,
            rexit[0], rexit[1], rexit[2], rexit[3], rexit[4], rexit[5]);
    }
}

istat=ephatm0(n_atm);
if (ephdriver_print>=1)
    printf("ephatm0=%d n_atm=%d\n", istat, *n_atm);
if (istat!=0)
    exit(1);
if (*n_atm>MAX_STT) {
    printf("ephdriver: MAX_STT = %d exceeded\n", MAX_STT);
    exit(1);
}

istat=ephatm(&etatm0, xatm, aatm, n_atm);
if (ephdriver_print>=1)
    printf("ephatm=%d\n", istat);
if (istat!=0)
    exit(1);
```



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 237 of 286	

```

if (ephdriver_print>=1)
  printf("n_atm=%d etatm0=%f\n",*n_atm,etatm0);

if (ephdriver_print>1) {
  for (i=0; i<*n_atm; i++) {
    printf("atmosphere: %f %e %e %e %e %e\n",etatm0+(double)(i),
      xatm[i][0],xatm[i][1],xatm[i][2],xatm[i][3],xatm[i][4],xatm[i][5]);
    printf("atmosphere bodydetic altitude: %f\n",aatm[i]);
  }
}

/* SAVE DATA FILES FOR TROUBLESHOOTING */
FILE *fp;
fp = fopen("ee_per.csv","a");
fprintf(fp, "%f %f %f %f %f %f %f\n",*et_peri0,
  rperi0[0],rperi0[1],rperi0[2],rperi0[3],rperi0[4],rperi0[5]);
fclose(fp);
fp = fopen("ee_per_a.csv","a");
fprintf(fp, "%f\n",*rperi0_dalt);
fclose(fp);
fp = fopen("ee_apo.csv","a");
fprintf(fp, "%f %f %f %f %f %f %f\n",*et_apo,
  rapo[0],rapo[1],rapo[2],rapo[3],rapo[4],rapo[5]);
fclose(fp);
/*


fp = fopen("alt.csv","a");
for (i=0; i<*n_atm; i++) fprintf(fp, "%.23e %.23e\n",
  etatm0+(double)(i),altitudes_atm[i]);
fclose(fp);
fp = fopen("scpos.csv","a");
for (i=0; i<*n_atm; i++) fprintf(fp, "%.23e %.23e %.23e %.23e\n",
  etatm0+(double)(i),states_atm[i][0],states_atm[i][1],states_atm[i][2]);
fclose(fp);
fp = fopen("scvel.csv","a");
for (i=0; i<*n_atm; i++) fprintf(fp, "%.23e %.23e %.23e %.23e\n",
  etatm0+(double)(i),states_atm[i][3],states_atm[i][4],states_atm[i][5]);
fclose(fp);

*/
}

```

## G.8.2 Example 2: Titan Non-atmospheric Run-out—a Main Program Example

There are nine orbits in this Titan non-atmospheric run-out, which corresponds to the Titan run-out. Only two arguments to "ephdriver" have name changes in this main program, and these are the second argument which is "tepo" instead of "et0" and the third argument which is "x"

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 238 of 286	

instead of "x0". Before the first call to "ephdriver", the epoch in variable "teepoch" and the orbital state in array "x" are initialized. This is indicated to "ephdriver" by setting its first argument "init\_orb" to 1 which indicates to call routines "ephinit" and "ephupd0". After that, the other 8 calls to "ephdriver" are preceded by "init\_orb" being set to 0 which indicates to not call either "ephinit" or "ephupd0". (This is an optional setting given that "ephdriver" always resets "init\_orb" to 0.) Finally, note that run-outs like this have no maneuvers or atmospheric passes and therefore no accelerometer data (n\_data = 0).

```
/*
 * Copyright 2011, KinetX, Inc. This software is developed as freeware and
 * may be freely reproduced, distributed, and used by anyone as long as it is
 * not used for profit and as long as any derivative works are also freeware.
 */
```


```
/* Written by David L. Skinner */
```

```
#include <stdio.h>
#include <stdlib.h>
```

```
#include "ephest.h"
```

```
main()
{
  double aatm[MAX_STT];
  double bary[4+NRECS*(NR+1)];
  double et[MAX_ACC]={0};
  double et_apo;
  double et_entry;
  double et_exit;
  double et_peri;
  double et_peri0;
  double plan[4+NRECS*(NR_699+1)];
  double rapo[6];
  double reentry[6];
  double rexit[6];
  double rperi[6];
  double rperi0[6];
  double rperi0_dalt;
  double rperi_dalt;
  double rperir[6];
  double sat[4+NRECS*(NR_606+1)];
  double sun[4+NRECS*(NR_10+1)];
  double tepoch;
  double x[6];
  double xatm[MAX_STT][6];

  int init_orb;
  int istat;
```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 239 of 286	

```

int n_atm;
int n_bary;
int n_plan;
int n_sat;
int n_sun;

/***** External references *****/
extern int ephbuild();
extern void ephdriver();
extern void ephunload();

ephdriver_error_handling=1;

ephdriver_print=1;

tePOCH = 292507200.0000000;
x[0] = 1.101775903953336e+06;
x[1] = 4.532982772796417e+06;
x[2] = 1.694457972447233e+07;
x[3] = -1.306418564602341e+02;
x[4] = -3.646815435373888e+02;
x[5] = 1.060535718871817e+02;

n_data = 0;


istat=ephbuild(sun,&n_sun,bary,&n_bary,plan,&n_plan,sat,&n_sat,tePOCH,0.);
if (istat!=0) {
    printf("istat = %d after call to ephbuild\n",istat);
    exit(1);
}

init_orb = 1;          /* integrate orbit 1 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
    &et_peri0,&rperi0_dalt,&et_apo,rapo,
    &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
    &et_exit,rexit,&n_atm,aatm,xatm);

init_flag = 0;        /* integrate orbit 2 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
    &et_peri0,&rperi0_dalt,&et_apo,rapo,
    &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
    &et_exit,rexit,&n_atm,aatm,xatm);

init_flag = 0;        /* integrate orbit 3 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
    &et_peri0,&rperi0_dalt,&et_apo,rapo,

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 240 of 286	

```

    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

init_flag = 0;          /* integrate orbit 4 */
ephdriver (&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

init_flag = 0;          /* integrate orbit 5 */
ephdriver (&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

init_flag = 0;          /* integrate orbit 6 */
ephdriver (&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

init_flag = 0;          /* integrate orbit 7 */
ephdriver (&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);


init_flag = 0;          /* integrate orbit 8 */
ephdriver (&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

init_flag = 0;          /* integrate orbit 9 */
ephdriver (&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);
}

```

### G.8.3 Example 3: Mars Aerobraking with Orbital State Updates —a Main Program Example

There are 20 orbits in this Mars example, which is the example that was used to compare the Ephemeris Estimator to DPTRAJ/MIRAGE. Only two arguments to "ephdriver" have name changes in this main program, and these are the second argument which is "tepoch" instead of "et0" and the third argument which is "x" instead of "x0". Before the first call to "ephdriver", the

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 241 of 286	

epoch in variable "tepo" and the orbital state in array "x" are initialized. Before the eighth and fifteenth calls to "ephdriver", the epoch in variable "tepo" and the orbital state in array "x" are updated. In all three cases, this is indicated to "ephdriver" by setting its first argument "init\_orb" to 1 which indicates to call routines "ephinit" and "ephupd0". The rest of the calls to "ephdriver" are preceded by "init\_orb" being set to 0, which indicates to not call either "ephinit" or "ephupd0". (This is an optional setting given that "ephdriver" always resets "init\_orb" to 0.) Note that all 20 calls to "ephdriver" are preceded by eight lines beginning with the line "if (init\_acc) {" . In this example, variable "init\_acc" has previously been set to 1. So, this "if" clause will test true. The lines that will be executed consist of a "system" call, a test to see if the "system" call was successful with error termination if it was not, and a call to "ephunload". This call to "ephunload" reads the file "ee\_accel.dat" created by the "system" call, unloads its time-tagged acceleration data contents into arrays "et" and "acc", and sets variable "n\_data" to the number of acceleration records that are present. Notice how the call to "ephbuild" in this example only has six arguments, i.e., arguments "plan," "n\_plan," "sat," and "n\_sat" are missing. This is because the central body is not Titan.

```

/*
 * Copyright 2011, KinetX, Inc. This software is developed as freeware and
 * may be freely reproduced, distributed, and used by anyone as long as it is
 * not used for profit and as long as any derivative works are also freeware.
 */


/* Written by David L. Skinner */

#include <stdio.h>
#include <stdlib.h>

#include "ephest.h"

main()
{
  double aatm[MAX_STT];
  double acc[MAX_ACC][3]={0};
  double bary[4+NRECS*(NR+1)];
  double et[MAX_ACC]={0};
  double et_apo;
  double et_entry;
  double et_exit;
  double et_peri;
  double et_peri0;
  double rapo[6];
  double reentry[6];
  double rexit[6];
  double rperi[6];
  double rperi0_dalt;
  double rperi_dalt;
  double rperir[6];

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 242 of 286	

```

double sun[4+NRECS*(NR_10+1)];
double tepoch;
double x[6];
double xatm[MAX_STT][6];

int init_acc;
int init_orb;
int istat;
int n_atm;
int n_bary;
int n_sun;

/***** External references *****/
extern int ephbuild();
extern void ephdriver();
extern void ephunload();

ephdriver_error_handling=1;

ephdriver_print=1;

tepoch = 2.93329038208989500999451e+08;
x[0] = 1.72760675314441435039043e+07;
x[1] = -4.52485670035811886191368e+06;
x[2] = 4.07681638074280992150307e+07;
x[3] = -2.70857367232199344186938e+01;
x[4] = 3.69688306831999113910570e+02;
x[5] = 5.25096242285018348638914e+01;


init_acc = 1;

istat=ephbuild(sun,&n_sun,bary,&n_bary,tepoch,0.);
if (istat!=0) {
  printf("istat = %d after call to ephbuild\n",istat);
  exit(1);
}

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit017.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et,acc,&n_data);
}

init_orb = 1; /* integrate orbit 1 */
ephdriver(&init_orb,tepoch,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 243 of 286	

```

&et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
&et_exit, rexit, &n_atm, aatm, xatm);

if (init_acc) {
istat=system("cp mars_roc.ee_accel_orbit018.dat ee_accel.dat");
if (istat!=0) {
printf("istat = %d after 1st call to system\n", istat);
exit(1);
}
ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 2 */
ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
&et_peri0, &rperi0_dalt, &et_apo, rapo,
&et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
&et_exit, rexit, &n_atm, aatm, xatm);

if (init_acc) {
istat=system("cp mars_roc.ee_accel_orbit019.dat ee_accel.dat");


if (istat!=0) {
printf("istat = %d after 1st call to system\n", istat);
exit(1);
}
ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 3 */
ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
&et_peri0, &rperi0_dalt, &et_apo, rapo,
&et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
&et_exit, rexit, &n_atm, aatm, xatm);

if (init_acc) {
istat=system("cp mars_roc.ee_accel_orbit020.dat ee_accel.dat");
if (istat!=0) {
printf("istat = %d after 1st call to system\n", istat);
exit(1);
}
ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 4 */
ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
&et_peri0, &rperi0_dalt, &et_apo, rapo,
&et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
&et_exit, rexit, &n_atm, aatm, xatm);

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 244 of 286	

```

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit021.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 5 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexit,&n_atm,aatm,xatm);

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit022.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et, acc, &n_data);
}


init_orb = 0; /* integrate orbit 6 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexit,&n_atm,aatm,xatm);

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit023.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 7 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexit,&n_atm,aatm,xatm);
tePOCH = 2.940284000000000000000000e+08;
x[0] = -9.31159024916070397011936e+05;
x[1] = -2.42423837009716685861349e+06;
x[2] = -3.07023892755041271448135e+06;
x[3] = 9.67849587909739511815133e+02;
x[4] = -4.17531521449173487781081e+03;

```



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 245 of 286	

```

x[5] = 1.01175500943769566219999e+03;

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit024.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et, acc, &n_data);
}

init_orb = 1; /* integrate orbit 8 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexite,&n_atm,aatm,xatm);

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit025.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et, acc, &n_data);
}


init_orb = 0; /* integrate orbit 9 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexite,&n_atm,aatm,xatm);

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit026.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 10 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexite,&n_atm,aatm,xatm);

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit027.dat ee_accel.dat");
  if (istat!=0) {

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 246 of 286	

```

printf("istat = %d after 1st call to system\n",istat);
exit(1);
}
ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 11 */
ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
&et_peri0, &rperi0_dalt, &et_apo, rapo,
&et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
&et_exit, rexit, &n_atm, aatm, xatm);

if (init_acc) {
istat=system("cp mars_roc.ee_accel_orbit028.dat ee_accel.dat");
if (istat!=0) {
printf("istat = %d after 1st call to system\n",istat);
exit(1);
}
ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 12 */
ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
&et_peri0, &rperi0_dalt, &et_apo, rapo,
&et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
&et_exit, rexit, &n_atm, aatm, xatm);

if (init_acc) {
istat=system("cp mars_roc.ee_accel_orbit029.dat ee_accel.dat");
if (istat!=0) {
printf("istat = %d after 1st call to system\n",istat);
exit(1);
}
ephunload(et, acc, &n_data);
}

init_orb = 0; /* integrate orbit 13 */
ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
&et_peri0, &rperi0_dalt, &et_apo, rapo,
&et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
&et_exit, rexit, &n_atm, aatm, xatm);

if (init_acc) {
istat=system("cp mars_roc.ee_accel_orbit030.dat ee_accel.dat");
if (istat!=0) {
printf("istat = %d after 1st call to system\n",istat);
exit(1);
}
ephunload(et, acc, &n_data);
}

```



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## **Autonomous Aerobraking (Phase 1)**

Page #:

247 of 286

```
init_orb = 0; /* integrate orbit 14 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexit,&n_atm,aatm,xatm);

tePOCH = 2.947097000000000000000000000000e+08;
x[0] = -9.96845590627893456257880e+05;
x[1] = -2.145507169824444468885660e+06;
x[2] = -3.15225289098504697903991e+06;
x[3] = 9.13219270463832799578086e+02;
x[4] = -4.26052627359854159294628e+03;
x[5] = 8.33728815678350883899839e+02;


if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit031.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et,acc,&n_data);
}

init_orb = 1; /* integrate orbit 15 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexit,&n_atm,aatm,xatm);

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit032.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
  ephunload(et,acc,&n_data);
}

init_orb = 0; /* integrate orbit 16 */
ephdriver(&init_orb,tePOCH,x,sun,n_sun,bary,n_bary,et,acc,n_data,
  &et_peri0,&rperi0_dalt,&et_apo,rapo,
  &et_entry,rentry,&et_peri,rperi,rperir,&rperi_dalt,
  &et_exit,rexit,&n_atm,aatm,xatm);

if (init_acc) {
  istat=system("cp mars_roc.ee_accel_orbit033.dat ee_accel.dat");
  if (istat!=0) {
    printf("istat = %d after 1st call to system\n",istat);
    exit(1);
  }
}
```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 248 of 286	

```

    ephunload(et, acc, &n_data);
  }
  init_orb = 0; /* integrate orbit 17 */
  ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

  if (init_acc) {
    istat=system("cp mars_roc.ee_accel_orbit034.dat ee_accel.dat");
    if (istat!=0) {
      printf("istat = %d after 1st call to system\n", istat);
      exit(1);
    }
    ephunload(et, acc, &n_data);
  }


  init_orb = 0; /* integrate orbit 18 */
  ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

  if (init_acc) {
    istat=system("cp mars_roc.ee_accel_orbit035.dat ee_accel.dat");
    if (istat!=0) {
      printf("istat = %d after 1st call to system\n", istat);
      exit(1);
    }
    ephunload(et, acc, &n_data);
  }

  init_orb = 0; /* integrate orbit 19 */
  ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);
  if (init_acc) {
    istat=system("cp mars_roc.ee_accel_orbit036.dat ee_accel.dat");
    if (istat!=0) {
      printf("istat = %d after 1st call to system\n", istat);
      exit(1);
    }
    ephunload(et, acc, &n_data);
  }


  init_orb = 0; /* integrate orbit 20 */
  ephdriver(&init_orb, tepoch, x, sun, n_sun, bary, n_bary, et, acc, n_data,
    &et_peri0, &rperi0_dalt, &et_apo, rapo,
    &et_entry, reentry, &et_peri, rperi, rperir, &rperi_dalt,
    &et_exit, rexit, &n_atm, aatm, xatm);

```

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 249 of 286	

## G.9 References

1. *PRELIMINARY Autonomous Aerobraking Planetary Constants and Models*, Version 0.06, June 9, 2011.
2. Seidelmann, P.K., et al., "Report of the IAU/IAG Working Group on Cartographic Coordinates and Rotational Elements of the Planets and Satellites: 2000," *Celestial Mechanics and Dynamical Astronomy*, in press, 2002.
3. Luciano less, et al., "Gravity Field, Shape, and Moment of Inertia of Titan," *Science* 327, 1367 (2010), March 12, 2010.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 250 of 286	

## Appendix H. AADS (Supplement to Section 7.5.1)

The spacecraft interfaces to the AADS flight software through the use of data structures, as shown in Figure H.1. The required AADS input data is passed into AADS through two data structures. The first input structure includes parameters and/or data which are likely to change between AADS calls, such as the following:


- Ephemeris Estimator initialization: flag which triggers use of provided state and epoch as initial state for integration.
- Ephemeris Estimator ephemerides data: including Sun, and central body barycenter for operation at Mars and Venus; includes Saturn and Titan information for operation at Titan.
- Ephemeris Estimator time-tagged acceleration data: 10 Hz data (provided by the IMU) during both the maneuver (if executed) and the atmospheric pass.
- Atmosphere Estimator acceleration data: 1 Hz data (provided by the IMU) during the atmospheric pass; provided epoch at start of data collection.
- Atmosphere Estimator attitude quaternion data: used to estimate aerodynamics, which, along with acceleration data, can be used to estimate atmospheric density and scale height.

The second input data structure to AADS contains data not likely to change between AADS calls, but which may be changed and uploaded to the spacecraft during the weekly update cycle. This AADS iLOAD structure includes data such as:

- Planetary constants: GM for Sun, central body, as well as Saturn and Titan if applicable, central body radii (equatorial and polar), central body pole definition.
- Ephemeris Estimator setup parameters: integration stepsize initial guess and tolerances, time offset from periapsis to estimate atmospheric entry and exit.
- Spacecraft parameters: mass, aerodynamic reference area, central body to aerodynamic coordinate frame transformation.
- Corridor settings: operational corridor lower and upper limit as well as desired target (as a percentage of the corridor width).


All outputs from AADS to the spacecraft are provided in a single output data structure. This AADS output structure includes data such as:

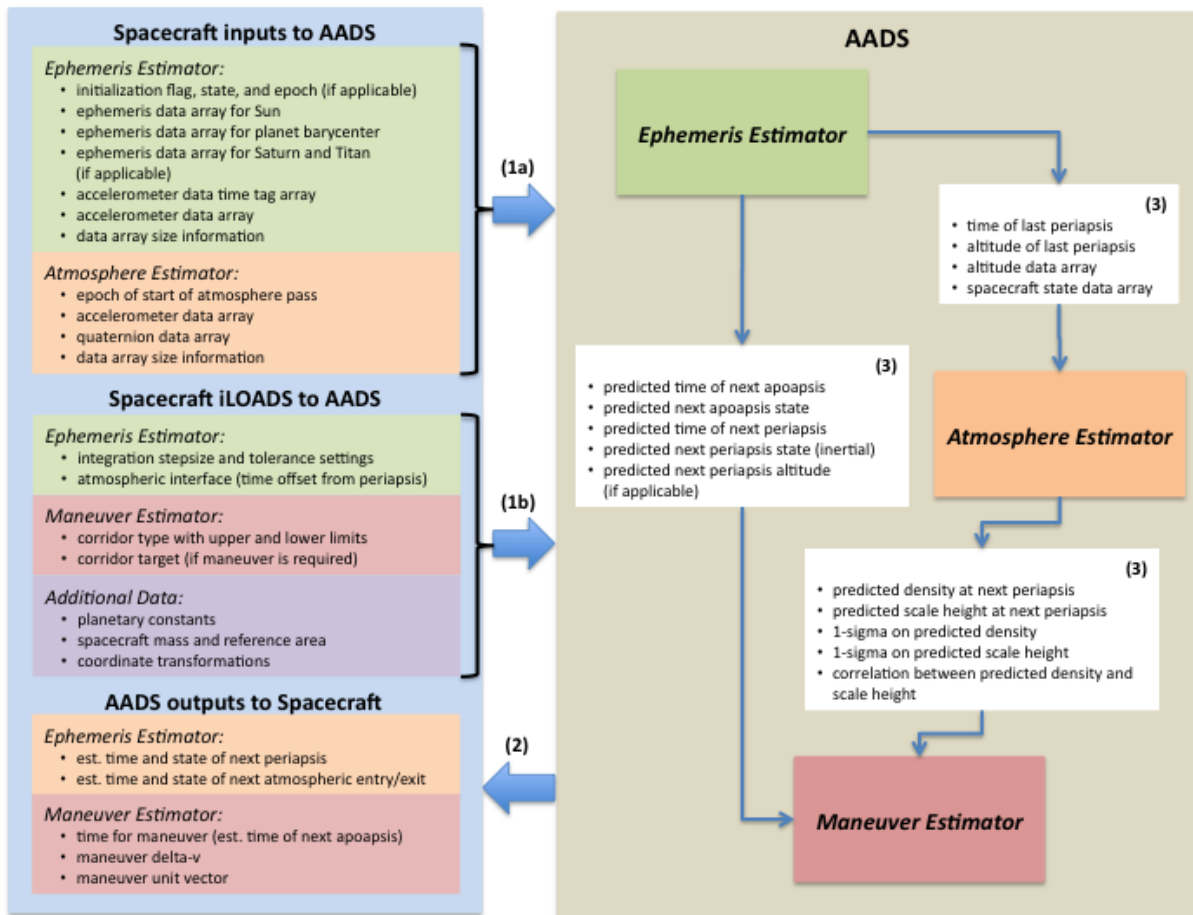
- Ephemeris Estimator atmospheric entry and exit predictions: both epoch and state used by spacecraft to command slew to and from AB configuration.
- Maneuver command: magnitude, direction and epoch; current model places maneuver epoch at the Ephemeris Estimator predicted apoapsis time, with the maneuver assumed to be in the direction of the estimated orbital velocity at apoapsis.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 251 of 286	

In addition to this interface with the spacecraft, an additional data structure is created within AADS to pass information between the Atmosphere Estimator, Ephemeris Estimator, and the Maneuver Estimator. This AADS internal data structure includes:

- Ephemeris Estimator prediction of the next apopsis state.
- Ephemeris Estimator prediction of the next periapsis state (including altitude).
- Ephemeris Estimator estimate of the current (previous) periapsis state (including altitude).
- Current (previous) atmospheric pass information: provided by Ephemeris Estimator to the Atmosphere Estimator and includes both state and altitude estimates.
- Atmosphere Estimator prediction of the next periapsis density and scale height (includes 1-sigma errors as well as correlation between the two predictions).

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		252 of 286	




**Figure H.1. AADS Interfaces with the Spacecraft using Data Structures: (1a) Spacecraft inputs to AADS which will or may change each AADS call, (1b) Spacecraft inputs to AADS which are not likely to change during the AB mission, (2) AADS outputs to the spacecraft, and (3) Intra-AADS.**

## H.1 A Cycle in the Life of AADS

With the interfaces now defined, it is possible to step through the AADS operation, at a high level, to better understand how the various models work together to successfully execute an AA mission while ensuring spacecraft safety. The AADS is called just once during each orbit, at some time after atmospheric exit and prior to the next apoapsis, giving the spacecraft sufficient time to complete any onboard tasks, including execution of the AADS software and preparations for any maneuver commanded by AADS.

The first processes executed during AADS operation involve the Ephemeris Estimator. This module is responsible for integrating the estimated spacecraft state up. This integration begins at either the state provided for initialization, or from the last state propagated to and stored in




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 253 of 286	

memory. Included in this state integration are all applicable models (central body gravity field, 3<sup>rd</sup> body gravitational effects and solar radiation pressure). The Ephemeris Estimator is provided IMU acceleration data during both burns and atmospheric passes. When the Ephemeris Estimator integration reaches the epoch of this data, it processes the data, including it in its integration estimate. So, with this the Ephemeris Estimator can effectively integrate a spacecraft state estimate from the last AADS call, through the provided maneuver and atmosphere acceleration data, up to the current mission time. This is the state the Ephemeris Estimator then stores away to use as its initial state during the next AADS call, if a new initialization state is not provided. However, the Ephemeris Estimator does not stop here. The spacecraft state continues to be integrated through the next apoapsis (assuming no maneuver), up to and through the next atmospheric pass, however the atmosphere itself is not modeled. This is done to first, provide predicted apoapsis conditions for execution of a maneuver, if needed, and second, to provide state and altitude information for the current/previous atmosphere pass to the Atmosphere Estimator.

At this point, the AADS calls the Atmosphere Estimator. The Atmosphere Estimator takes the atmospheric pass acceleration data provided by the spacecraft, along with the atmospheric pass state and altitude information provided by the Ephemeris Estimator, to estimate the atmospheric conditions (density and scale height) during the current/previous pass. This information is then added to an archive where atmospheric condition estimates are stored for each pass. This archive data is then used to predict both the density and scale height of the next periapsis, along with dispersion and correlation predictions.

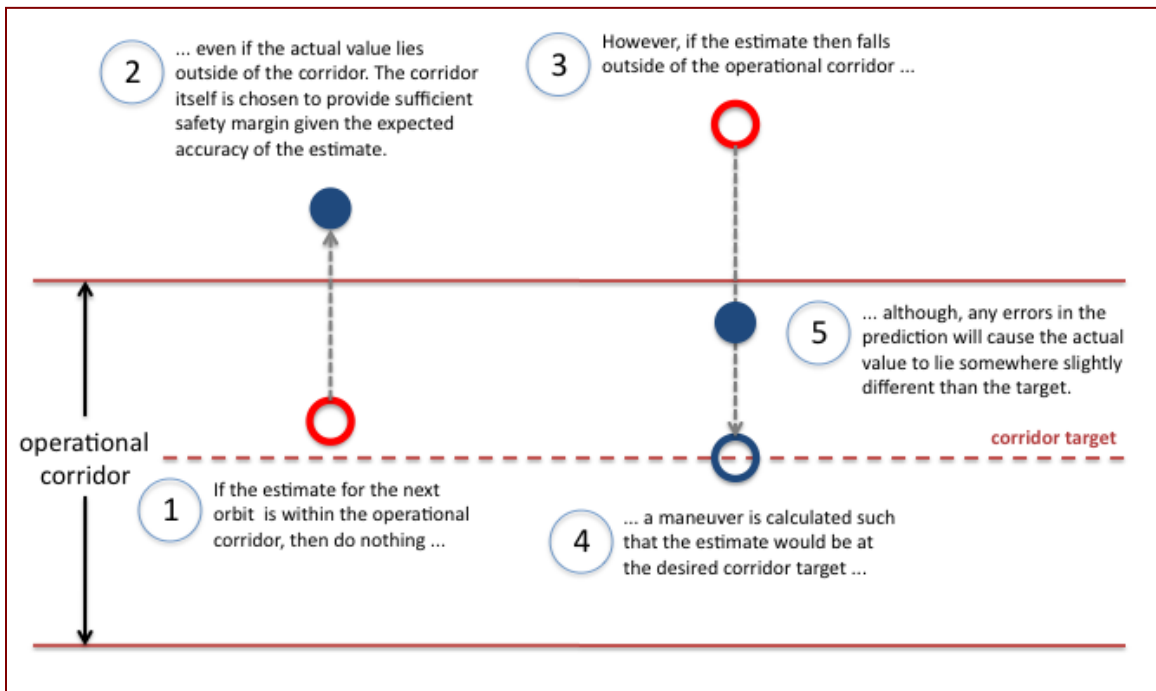
Once the Ephemeris and Atmosphere Estimators have calculated their prediction for the conditions at the spacecraft's next periapsis, the Maneuver Estimator can now determine whether or not a maneuver is required, and if so, what that maneuver should be. Figure H.2 illustrates this process through the following steps:

1. Calculate the predicted operational corridor location (e.g., heat rate, temperature, altitude, etc.) based on the information provided by the Ephemeris and Atmosphere Estimators.
2. If the predicted location is within the corridor, do nothing. Note that this does not guarantee that the true corridor location will be within the corridor. If the AADS has not diverged significantly, then the difference between the predicted and true corridor location should be small, certainly close enough not to trigger any immediate action.
3. If the predicted location is outside of the corridor, a change in altitude is determined to place the spacecraft at the desired corridor target.
4. This change in altitude is added to the current estimated periapsis altitude, which is then added to the estimated apoapsis altitude to determine a new orbit semi-major axis, from which a new velocity at apoapsis is determined. The difference between this new apoapsis velocity and the current estimate of the apoapsis velocity provided by the Ephemeris Estimator is the required maneuver magnitude. This value is


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 254 of 286	

positive for a periapsis raise (decrease freestream heating rate) and negative for a periapsis lowering (increase freestream heating rate). The maneuver direction is estimated to be that of the pre-maneuver velocity vector at apoapsis. Since these maneuvers are typically small ( $< 0.5$  m/s), this assumption works well, even when considering a finite burn.

At this time, the AADS software can be placed in stand-by mode or terminated until the next AADS function call to free spacecraft resources for other activities. Onboard a spacecraft, the AADS software will not always running, but instead is called once per orbit, typically at some time after an atmospheric pass ends and prior to the next apoapsis. Some AADS data does need to be preserved between AADS calls (e.g., Ephemeris Estimator current state prediction and Atmosphere Estimator atmosphere archive data), so at least a portion of the memory will be allocated and preserved while AADS is not running. If a maneuver is executed, the associated acceleration data is collected and stored, along with the subsequent atmospheric pass acceleration data, and provided to the AADS at the next call. This process is then repeated each orbit. Ideally, this would be done completely autonomously, with interactions from the ground no more frequent than once per week. At those times, any changes in the AADS parameters (e.g., corridor and/or target) could be made, along with an Ephemeris Estimator initialization state update.




**Figure H.2. AADS Execution Process with Respect to the Operational Corridor**

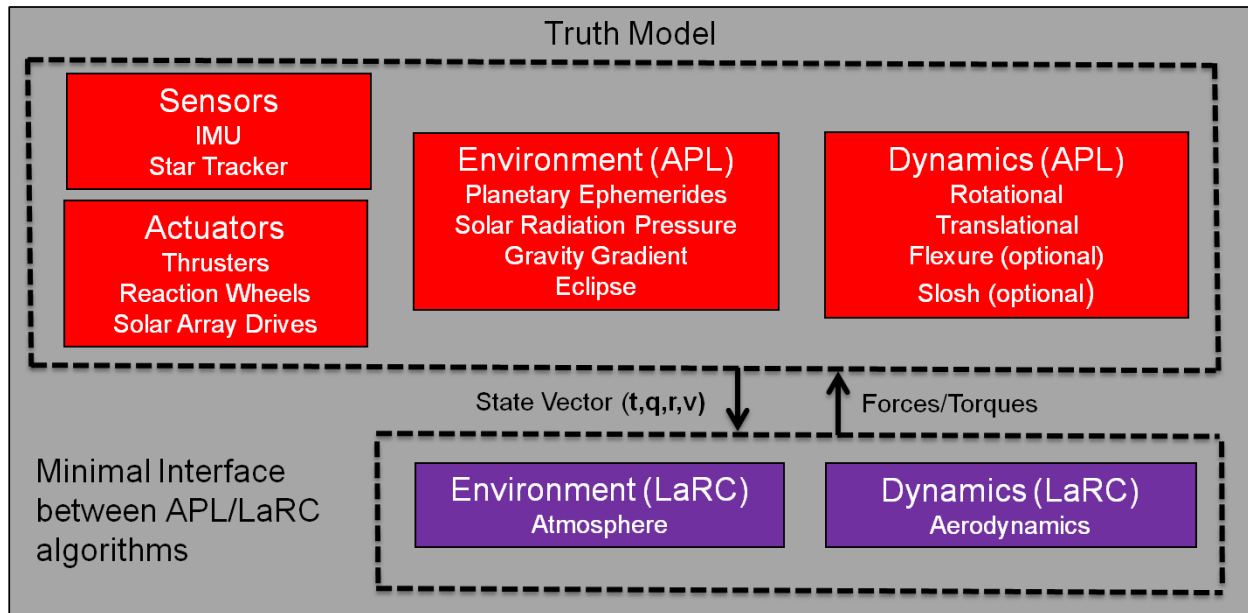
	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 255 of 286	

## Appendix I. AAHFS (Supplement to Section 7.5.2)

### I.1 AAHFS Truth Model

The AAHFS truth model software replicates the true behavior of the spacecraft in its environment. The environment and spacecraft characteristics of a typical AB mission are similar to many other deep-space missions. This makes the MESSENGER truth model a good initial starting point for developing a test environment for AA. The MESSENGER 6-DOF truth model includes the spacecraft dynamics, sensors, actuators and environmental disturbances. Although these actuator models were developed to emulate the characteristics of the MESSENGER flight hardware, they provide a convincing, validated environment for testing the AA approach since these sensors are typical of an AB spacecraft as well. The environment models approximate all known disturbances sources external to the vehicle. These disturbances affect both the trajectory and the attitude dynamics, so it is natural to encapsulate these effects in a full 6-DOF simulation. A key addition is that AA modeling requires detailed atmospheric and aerodynamic models to determine the forces and torques due to atmospheric drag. These models were not a part of the original MESSENGER truth models, but are easily incorporated into the code by adopting the architecture shown in Figure I.1-1. This figure highlights the high-level interface between the MESSENGER heritage software, depicted in red, and the new development to support AA, depicted in purple. Minimizing the interaction between the heritage code and the AB models reduces the complexity of the model and ensures the MESSENGER-based software is unperturbed. The new AB models have their own rich flight heritage as well, as they are based on software used for testing and operations of prior NASA AB missions. The forces and torques that result from these models are integrated by the full equations of motion to predict the vehicle state. Sensor models are used to emulate the data inputs to the GN&C system. These models make use of the MESSENGER flight heritage to include flight performance characteristics, further enhancing the fidelity of the simulation environment.


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		256 of 286	



*Figure I.1-1. AAHFS Truth Model Block Diagram*

### I.1.1 MESSENGER Heritage Truth Models in AAHFS


The MESSENGER heritage software contains complete actuator models that are sufficient for the AAHFS demonstration. These models include both thruster and reaction wheel models, which operate at 200 Hz to ensure that transient characteristics are correctly modeled. Although the MESSENGER reaction wheels are undersized for a typical AB mission (as the AB mission inertias are 3–5 times the MESSENGER values), the model can be scaled appropriately to ensure the torque and momentum characteristics are consistent with the simulated vehicle inertias. There is significant flexibility built into these models, as modifications can easily be made to static and running friction, wheel alignments and inertia variability, allowing for trade study and (if desired) Monte Carlo analysis. The MESSENGER propulsion system is perhaps overly complex for the AA studies, but this model has simplified modes that are consistent with propulsive maneuver operations during prime AB missions. Thruster capabilities allow corridor control maneuvers to be simulated with appropriate fidelity. The propellant supply mechanism can run in a fixed pressure mode or simulate tank blowdown for producing realistic thrust and specific impulse, as well as ensuring mass changes are modeled realistically. Thrust transients due to valve/thruster/thermal effects are modeled to accurately mimic system performance. Additional performance variation is easily accomplished including variable thruster plume impingement model included for Monte Carlo study. These models have undergone an extensive correlation with flight data, ensuring that they mimic the real hardware performance, thereby ensuring a high-fidelity test environment for AAHFS.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 257 of 286	

The MESSENGER heritage software includes high-fidelity sensor models running at rates that are consistent with the device operation. This includes multiple star trackers for attitude determination running at 10 Hz. These star tracker models have easily adjusted alignments and noise characteristics for sensitivity studies. The 100-Hz IMU model is critical for AAHFS simulation and study, since AADS algorithms heavily rely on acceleration data from IMU. A variety of trade studies may be useful involving this model, as it is of interest to determine the accuracy required of the IMU data (in particular, the accelerometer readings through the drag pass) to ensure the proposed AA approach meets performance goals. The IMU model includes a convenient environment for studying AADS sensitivity to accelerometer (and gyro) misalignments, scale factor errors, biases, noise (white and readout), IMU clock walk, data collection frequency and sensor latencies.

The environment models in the AAHFS simulation model all of the salient perturbations that the spacecraft would experience when in orbit about the central body. This simulation can extract ephemeris information for any solar system body of interest from a SPICE SPK file. The bodies of interest are easily changed, allowing the simulation to be quickly adapted to different central bodies of interest (as well as modifying the desired gravitational perturbative bodies). The central body uses an optional harmonic gravity model, which can be of arbitrary degree and order; this provides the highest possible fidelity in integrating the translational equations of motion. Disturbance forces include the effects of solar radiation pressure (SRP) with a full umbra/penumbra eclipse model. This SRP model is specific to the MESSENGER spacecraft, but is scaled to get the effect to be consistent with an AB mission (based on the ratio of the Sun-facing areas between a notional AB mission and MESSENGER). For missions where this force is negligible (at Titan, for instance), this effect can be disabled. Additionally, disturbance torque models include those due to SRP as well as the gravity gradient. In general, most of these environment models are run at 1 Hz as the resulting disturbance forces and torques are slowly varying and don't require higher execution rates. The exception is the ephemeris models, since the planetary states are used to calculate the right-hand side of the translational equations of motion. In this case, the SPICE file extractions are done at 1 Hz, but the planet states are interpolated up to the model integration rate of 200 Hz.

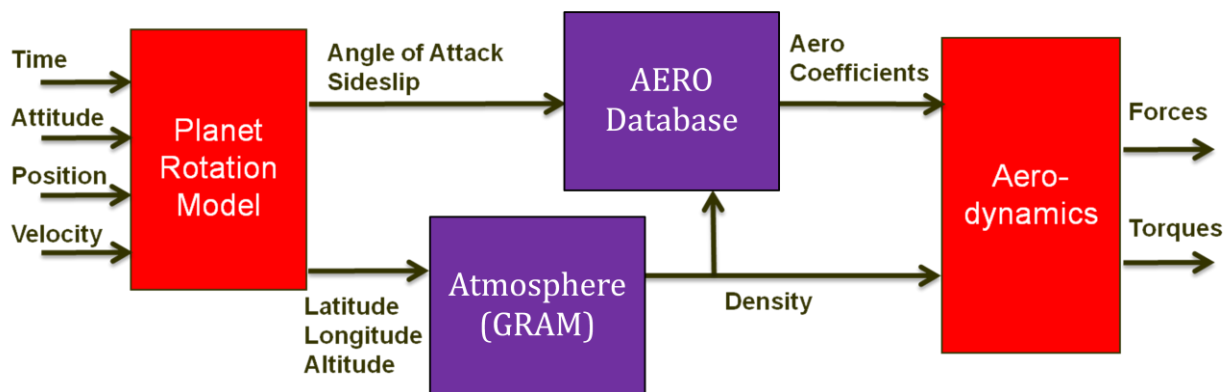
The AAHFS dynamics models integrate the translational and rotational dynamics at 200 Hz. This rate is selected to capture thrust transients for thrusters pulsed at 50 Hz. A smaller step size would be possible; testing reveals that it does not provide additional fidelity but it does reduce the simulation speed. Simulation speed is of critical importance, as the goal of the project is to demonstrate AA is plausible for durations on the order of a week, so the AAHFS code must be capable of running in excess of seven days. Additionally, the AAHFS truth models include propellant slosh dynamics and a structural model, which are optional models for the purposes of this study. While these models do enhance the fidelity of the truth model software, they are spacecraft specific, and would need to be updated to be consistent with the vehicle of interest for a real AB flight program. Further, this model does not respond or get excited by the atmosphere, limiting its utility. It does provide some mechanism to conduct simulations on a non-rigid body,

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		NESC-RP-09-00605	1.0
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		258 of 286	

particularly to model the IMU response to a flexible environment, so it is retained for future study purposes. Both the slosh and the structural model are easily disabled for the initial AAHFS testing.


### I.1.2 AAHFS Truth Model New Development

Figure I.1-2 shows a detailed block diagram for the new AAHFS truth model development. Of particular interest for this discussion are the heritage elements provided by the LaRC team members, the atmosphere model and the aerodynamic models. These new models have been added to support AB specific capabilities these models have been calibrated using AB flight data from prior NASA AB missions. This produces a convincing environment for testing the 6-DOF behavior of an AB spacecraft.



*Figure I.1-2. Block Diagram for New AAHFS Truth Models*

The fidelity of the atmosphere model is of particular importance to the demonstration of an end-to-end AA simulation. Additionally, flexibility of this model is of paramount importance, as this study aims to use atmospheric variability as one means of investigating the robustness of the proposed AADS algorithms. The Mars-GRAM is an engineering-level atmospheric model widely used for prior Mars AB study and mission analyses [refs. 1, 2]. In addition to providing high-fidelity predictions of the mean density, temperature, pressure, and wind components at any planet position and altitude, Mars-GRAM allows for the simulation of perturbed profiles about the mean conditions, thereby offering great flexibility for testing the AA approach. The most recent version of Mars-GRAM (2010) has updates to reconcile the models with the AB data of MRO, Mars Odyssey, and MGS, making it a highly useful model for an AB study. For the two other central bodies in this study a Titan-GRAM model and Venus-GRAM model were utilized. C versions of these GRAM models have been provided, and through the use of the Simulink Legacy Code Tool, these models have been directly integrated into the AAHFS truth model software. It is of minor consequence that this code was originally conceived in FORTRAN, and relies on namelist files for parametric modification. Prior to running the Simulink simulation,


	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 259 of 286	

these namelist files are easily modified by MATLAB scripts. The significant capability that the GRAM code provides does come at the price of execution speed. For this reason, the AAHFS only executes the GRAM model at 1 Hz. The output atmospheric densities are generally quite smooth and slowly varying, so interpolation up to the 200-Hz integration time step is easily accomplished. This allows use of the high-fidelity GRAM models without sacrificing simulation speed. The AAHFS atmosphere modeling strategy allows a wide variety of studies of missions to Mars, Venus, and Titan with a single simulation.

An aerodynamic model has been added to allow conversion of the atmospheric data generated by the GRAM software into the forces and torques operating on the vehicle. This model uses an underlying database of aerodynamic coefficients that are interpolated based on the spacecraft attitude and atmospheric density. This database is spacecraft specific (the current one in use is based on MRO data), and although this step in the process is mission dependent, the data currently in use is consistent with a typical AB spacecraft. If an aerodynamic data set was available for a proposed spacecraft, it is trivial to modify the software to adopt an alternate database. Any future flight program that uses the AA approach would substitute their vehicle model in for the MRO model when that data became available. As such, the use of MRO data for the vehicle aerodynamic properties is notional for this study. However, this database has been validated against an AB flight mission, thereby serving to produce more convincing test results. Once this model produces the necessary aerodynamic coefficients, the forces and torques are produced via standard aerodynamic equations [ref. 3]. It is important that these dynamic quantities are calculated at a high rate, as the attitude can be active during a drag pass. The AAHFS software computes these forces and torques at the integration rate of 200 Hz to ensure faithful modeling of the aerodynamics during a drag pass.

### **I.1.3 AAHFS FLIGHT Software**

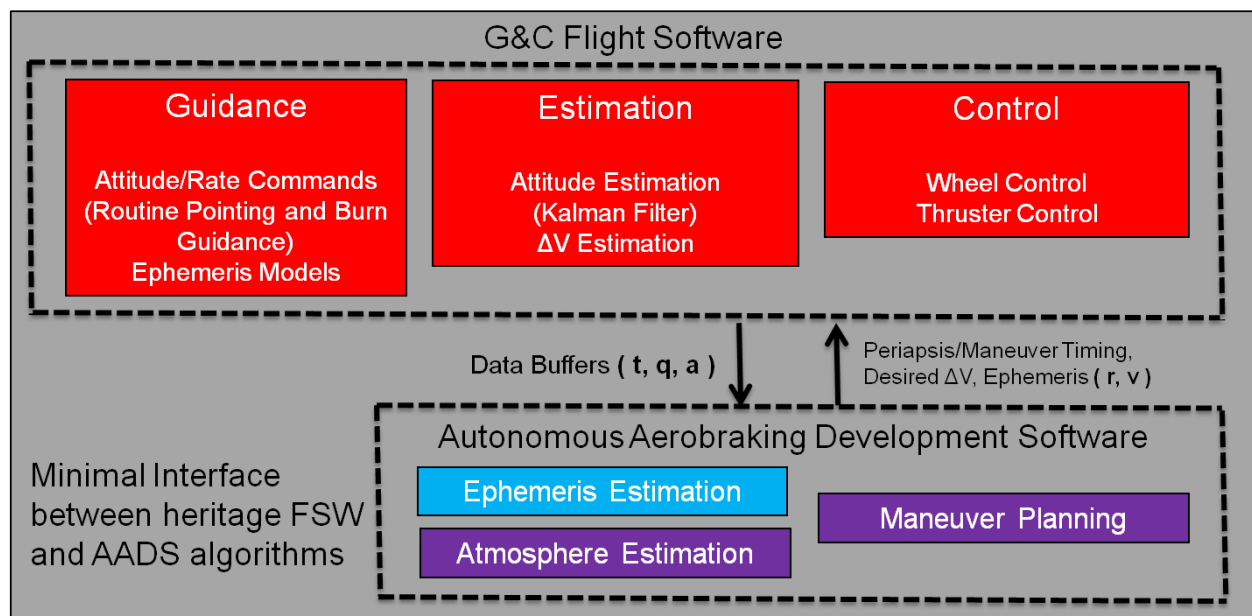
The flight software portion of the AAHFS model represents all of the guidance and control functions and algorithms necessary to ensure control of the spacecraft. As with the AAHFS truth model algorithms, the framework for this software was the MESSENGER guidance and control flight software. Where appropriate, AAHFS uses the same control code for the demonstration of the AA capability. This heritage software includes a typical set of attitude and maneuver guidance, estimation and control algorithms. The primary responsibility of this code is to ensure the attitude follows the desired pointing profiles, that angular momentum remains within desired limits, and that the propulsive maneuvers are executed successfully. One major advantage of this architecture is that the portion of the flight software that is inherited from MESSENGER is the onboard software used for the flight mission. The only significant differences are that the simulation runs in a workstation much faster than real time and responds to simulated environmental data, whereas the onboard guidance and control software runs in real time as an embedded application on the flight processor and is experiencing the true flight environment. The MESSENGER heritage code alone is insufficient to demonstrate AA. The AADS is a new block of software developed to handle the additional functions necessary to implement the AB

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		260 of 286	

corridor control onboard. While this set of algorithms represents a small part of the AAHFS software, the development of these algorithms is the primary focus of the demonstration of the AA capability. The remainder of the AAHFS acts as a test bed for these algorithms.

### I.1.4 Heritage Flight Software Algorithms


Much of the AAHFS flight software block depicted in Figure I.1-3 is heritage code from the MESSENGER mission. A standard set of algorithms is used to ensure the necessary control goals are achieved. The addition of the AADS software as well as the additional drag pass operations of an AB spacecraft levy additional requirements on the standard MESSENGER software. This required modification to all elements of the software functionality to enhance the autonomy of the system and to ensure control is maintained through the drag pass.



**Figure I.1-3. AAHFS Guidance and Control Flight Software Block Diagram**

The MESSENGER heritage software contains a complete set of guidance algorithms. These algorithms maintain knowledge of all relevant solar system bodies to allow construction of a variety of pointing commands [ref. 4], as well as to ensure satisfactory execution of planned  $\Delta V$  maneuvers. Transitions and parameterization of these pointing scenarios are typically handled with ground commands, but in the case of AA, these mode transitions need to happen autonomously. In the case of the pointing scenarios, the spacecraft must determine when and how to configure itself for an AB drag pass as well as handle the appropriate configuration for executing a corridor control maneuver. These events cannot be triggered via ground command,




	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 261 of 286	

as their timing is not known in advance. This forces these events to be orbit-event driven. As an example, the spacecraft guidance system must know the expected time of the ensuing periapsis passage. Based on this periapsis timing, the spacecraft must determine the duration of the AB drag pass, and ensure that the proper reconfiguration is handled to orient a preferred axis into the wind prior to entering the atmosphere to ensure the necessary aerodynamic stability. As the orbit evolves over time due to the varied pass drag environment, the spacecraft must respond accordingly. As a result, the MESSENGER guidance algorithms have been modified to ensure these pointing transitions occur autonomously based on timing information about the orbit. Likewise, to ensure maneuvers are executed correctly, the spacecraft must be able to determine its own attitude command and maneuver timing to ensure the desired corridor control maneuver executes properly. The AADS function computes the desired corridor control  $\Delta V$  and maneuver epoch, and the guidance system must use this information to autonomously implement the maneuver. So while much of the functionality of the onboard guidance system is unchanged, the level of autonomy is increased significantly.

The estimation tasks provided by the MESSENGER flight software are reused directly for AAHFS. This software runs a model replacement Kalman Filter to estimate the spacecraft attitude from the gyro and star tracker data. Although fault scenarios are not a planned part of the AAHFS test program, the attitude estimation is robust to missing or incomplete sensor data, as is typical for flight software. The filter executes at 1 Hz, and attitude estimates are propagated with high-rate gyro data up to the control task rate of 50 Hz. The MESSENGER software supports a high-rate (50 Hz) estimation of accumulated  $\Delta V$  based on the accelerometer data. This includes the onboard estimation of accelerometer biases prior to the maneuver. This process has been modified to execute autonomously, as for the MESSENGER flight program, the bias estimation and maneuver estimation are all accomplished with command sequences carefully planned by ground operators. An identical process to maneuver estimation is proposed for estimating the accelerations (or alternately,  $\Delta V$ ) from the AB drag pass. This estimation must happen autonomously, so the software has been configured to execute the necessary commands autonomously to perform the accelerometer bias estimation as well as to estimate and buffer the accelerations from the drag pass. This buffer of accelerations will be used in the AADS software, discussed in the next subsection.

Much of the control functionality required for an AB mission is a part of the MESSENGER heritage flight software. This software contains algorithms for attitude control on reaction wheels and thrusters. The wheel control law is a hybrid law that performs an eigenaxis slew for large angle maneuvers with a bang-bang control law and is reduced to a PID law when the angle errors are small. The thruster control law uses a phase plane to control each thruster individually, and is typically only employed during maneuvers and momentum dumps. Momentum control is generally accomplished via ground command, but autonomous momentum control is a part of the MESSENGER heritage code, and is used for AAHFS as necessary. The only significant modification to the MESSENGER code to support AAHFS simulations is to develop a control mode for the AB drag pass. Much of the attitude control during AB is

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 262 of 286	

accomplished by the aerodynamic stability of the vehicle. It is assumed that the vehicle is stable in pitch and yaw, so the roll is the only element that requires control, and this torque is assumed to be reasonably small. The control concept adopted for AAHFS simulations is to enter the drag pass on wheel control, ensuring that a pre-defined vehicle axis (the “nose”) is pointed into the wind. Once in the drag pass and the aerodynamics take over, the wheels spin down (off-loading momentum), and pitch and yaw are controlled by the aerodynamic stability. The rolling motion is controlled with thrusters, although few thruster pulses are required, due to the small roll torque and the large thruster deadbands. This strategy maintains vehicle and momentum control with minimal propellant usage. This ensures that the thruster pulses are captured by the buffered acceleration data to allow accurate orbit determination by the AADS software.


The algorithms and data flow used for the AADS software are described in detail in Section 7.2.1 of this report.

Reference 1: C.G. Justus, “A Mars Global Reference Atmospheric Model (MARS-GRAM) for Mission Planning and Analysis.” AIAA-1990-4 28<sup>th</sup> Aerospace Sciences Meeting, Reno, NV. Jan 8-11, 1990.

Reference 2: H.L. Justh *et al.*, “The Next Generation of Mars-GRAM and Its Role in the Autonomous Aerobraking Development Plan.” AAS 11-478, AAS/AIAA Astrodynamics Specialist Conference, Girdwood, AK, 2011.

Reference 3: B.L. Stevens and F.L. Lewis, *Aircraft Control and Simulation*, John Wiley and Sons, Inc., Hoboken, New Jersey, p100-106, 2003.

Reference 4: D.J. O’Shaughnessy and R.M. Vaughan, “MESSENGER Spacecraft Pointing Options,” AAS 03-149, AAS/AIAA Spaceflight Mechanics Meeting, Ponce, Puerto Rico, Feb, 2003.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP- 09-00605</b>	Version: <b>1.0</b>
Title:	<b>Autonomous Aerobraking (Phase 1)</b>		Page #: 263 of 286

## **Appendix J. AA Interface Control Document**

### **Autonomous Aerobraking Development Software**

**DRAFT**


### **Interface Control Document**

Document No.: [AA-001](#)

Document Availability Authorization: [NF 1676L ID 11031](#)

Date of Issue: 20 October 2010

Modified: 17 November 2011

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document # <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 264 of 286	

## Interface Control Document

\_\_\_\_\_

Project Manager

\_\_\_\_\_


Date

\_\_\_\_\_

Program Executive


\_\_\_\_\_

Date

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		265 of 286	

## Table of Contents

<b>1</b>	<b>Introduction.....</b>	<b>266</b>
1.1	Purpose.....	266
1.2	Background.....	266
<b>2</b>	<b>Documents .....</b>	<b>267</b>
2.1	Applicable Documents.....	267
2.2	Reference Documents .....	267
<b>3</b>	<b>Interface Design .....</b>	<b>267</b>
3.1	Interface Identification.....	268
3.2	Interface Diagram .....	268
3.3	Individual Interface Characteristics .....	269
	3.3.1 AAI_01: Spacecraft to Ephemeris Estimator.....	269
	3.3.2 AAI_02: Spacecraft to Atmosphere Estimator .....	270
	3.3.3 AAI_03: Spacecraft to Thermal Model .....	270
	3.3.4 AAI_04: Spacecraft to Maneuver Estimator.....	270
	3.3.5 AAI_05: Ephemeris Estimator to Atmosphere Estimator .....	270
	3.3.6 AAI_06: Ephemeris Estimator to Maneuver Estimator.....	270
	3.3.7 AAI_07: Atmosphere Estimator to Maneuver Estimator .....	271
	3.3.8 AAI_08: Maneuver Estimator and Thermal Model.....	271
	3.3.9 AAI_09: Ephemeris Estimator to Spacecraft.....	271
	3.3.10 AAI_010: Maneuver Estimator to Spacecraft.....	271
3.4	Detailed Interface Data Description.....	271
	3.4.1 Spacecraft iLOADS Interface Data Structure Element Descriptions .....	272
	3.4.2 Spacecraft Input Data Structure Element Descriptions .....	277
	3.4.3 AADS Internal Data Structure Element Descriptions .....	281
	3.4.4 AADS Output Interface Structure Element Descriptions .....	284

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 266 of 286	

## 1.0 Introduction

This document describes the Software Interface assumptions for the AADS. This study is led by the NESC and is tasked with developing and testing the algorithms necessary to safely perform an AB operations mission autonomously on board a spacecraft.

The AADS is a suite of models and algorithms intended to test the feasibility of an AA system. Three separate AADS packages are being developed for this NESC study, one each for Mars, Venus, and Titan. AADS for application at Mars and Titan consists of three distinct modules: (1) the Ephemeris Estimator, developed by KinetX, which processes spacecraft IMU acceleration data to estimate current and future spacecraft states, (2) the Atmosphere Estimator, developed by LaRC in conjunction with the National Institute of Aerospace (NIA), which processes spacecraft acceleration data along with Ephemeris Estimator state data to estimate the atmosphere's density and scale height, and (3) the Maneuver Estimator, developed by LaRC, which processes data from both the Ephemeris and Atmosphere Estimators to determine whether or not a maneuver is required to keep the spacecraft within the desired operational corridor. The AADS for Venus will also include a fourth module containing temperature models, also developed at LaRC, to predict the maximum temperature the spacecraft will encounter during the next atmospheric pass.

The AADS modules are integrated into two separate simulation environments for detailed performance analyses. The first is The Program to Optimize Simulated Trajectories (POST2) at LaRC [refs. 1, 2]. The second is a high-fidelity, software and hardware-in-the-loop simulation (AAHFS) based on the MESSENGER spacecraft testbed, developed at the JHU/APL.


### 1.1 Purpose

The key interfaces between spacecraft (or spacecraft simulator) and the AADS must be defined, understood and accepted for the efficient conduct of the development task and for its results to be credible.

The primary purpose of this document is to describe the agreed upon interface definitions between not only the spacecraft (or spacecraft simulator) and the AADS, but between the various modules internal to AADS.

### 1.2 Background

Several past NASA missions have used the AB technique to reduce the fuel required to deliver a spacecraft into a desired orbit around a target planet or Moon with an appreciable atmosphere. AB was first demonstrated at Venus with Magellan in 1993 and then was used to achieve the science orbit of three Mars spacecraft: MGS in 1997, Mars Odyssey in 2001, and MRO in 2006. Instead of using only the propulsion system to decelerate the spacecraft, AB is used after the initial orbit insertion to further decelerate the spacecraft using aerodynamic drag. The spacecraft traverses the upper atmosphere of the planet or moon multiple times while controlling periapsis

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 267 of 286	

altitude using small propulsive maneuvers at apoapsis to hold the spacecraft within a specified corridor. This corridor is designed to keep the spacecraft safely within required structural and/or thermal design limits until the desired orbit is achieved.

Although AB itself reduces the propellant required to reach the final orbit, this reduction comes at the expense of additional mission time (typically 3–6 months), a large mission operations staff, and significant DSN coverage. This combination of critical resources results in an expensive operational phase of a mission. Aerobraking missions typically require daily monitoring and weekly support to determine the maneuvers required to maintain the spacecraft on the predefined mission glideslope. Significant operational cost could be saved by enabling a spacecraft to calculate the maneuvers required on-board, based on measurement data collected during each atmospheric pass and any executed maneuvers, and execute the same glideslope maintenance maneuvers autonomously. This capability would be enabling for some orbiter missions, where due to the environment and its impact on the required maneuver frequency, or simply due to light-time delay, successful AB mission operations from the ground alone would not be possible.

## 2 Documents


### 2.1 Applicable Documents

### 2.2 Reference Documents

1. Autonomous Aerobraking Planetary Constants and Models Document
2. Detailed Documentation of the Ephemeris Estimator (users guide)
3. Detailed Documentation of the Atmosphere Estimator
4. Detailed Documentation of the Maneuver Estimator
5. Detailed Documentation of the Thermal Response Surface Model
6. Detailed Documentation of the APL High Fidelity Simulation
7. Detailed Documentation of the assumed spacecraft sensor, actuator, etc., models

## 3 Interface Design

Only the software interfaces between the spacecraft (or spacecraft simulator) and the AADS, as well as those between the individual AADS modules, are included within the scope of this document. For a more detailed description of the AADS and its individual modules, as well as other simulation and environment models, see the references provided in Section 2.2.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>			Page #: 268 of 286

### 3.1 Interface Identification

This document outlines the interface between the instruments, models and simulations. The interfaces are provided unique identifiers for use throughout the document (i.e., AAI## - Autonomous Aerobraking Interface ##). The cases are ordered in the general direction of data flow.

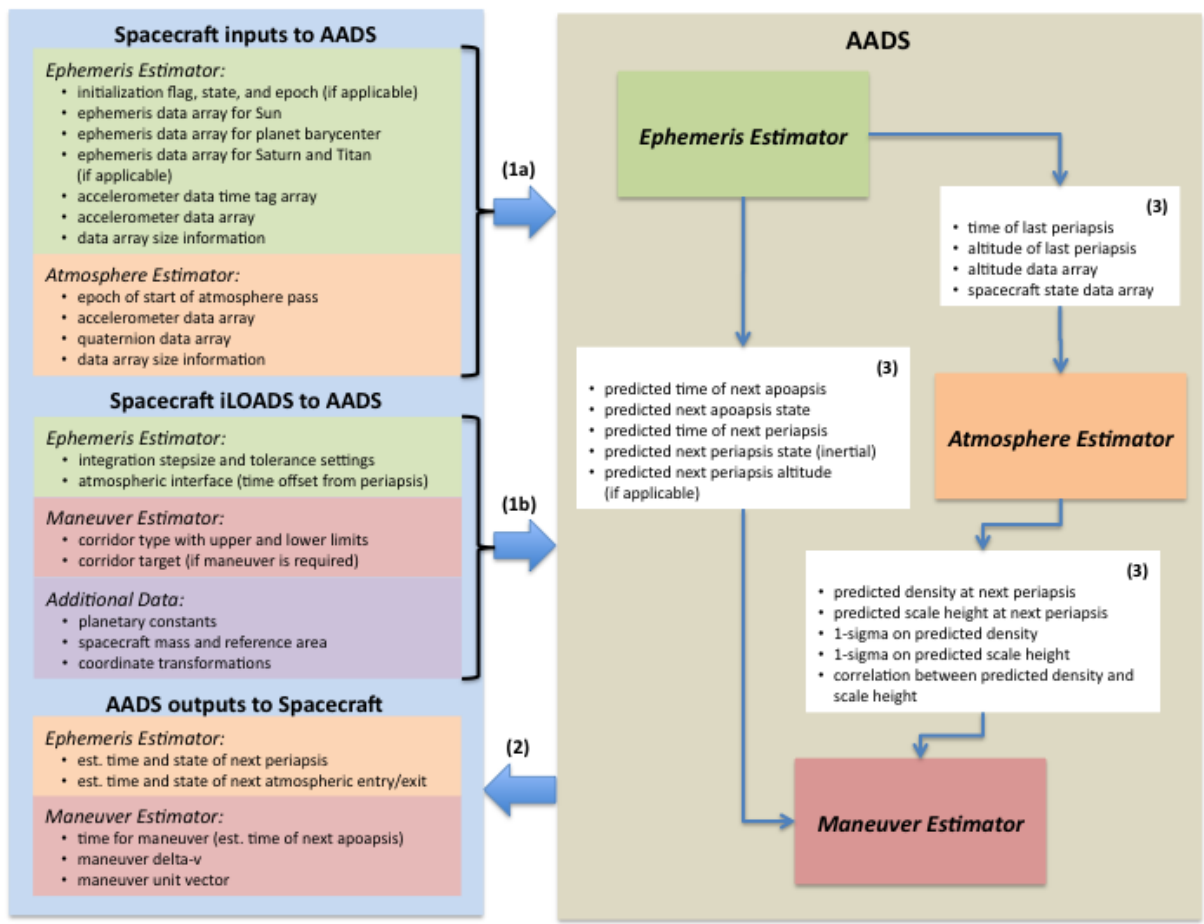
- a. AAI\_01 - Spacecraft to Ephemeris Estimator
- b. AAI\_02 - Spacecraft to Atmosphere Estimator
- c. AAI\_03 - Spacecraft to Thermal Model
- d. AAI\_04 - Spacecraft to Maneuver Estimator
- e. AAI\_05 - Ephemeris Estimator to Atmosphere Estimator
- f. AAI\_06 - Ephemeris Estimator to Maneuver Estimator
- g. AAI\_07 - Atmosphere Estimator to Maneuver Estimator
- h. AAI\_08 - Maneuver Estimator and Thermal Model
- i. AAI\_09 - Ephemeris Estimator to Spacecraft
- j. AAI\_10 - Maneuver Estimator to Spacecraft

Each of these interfaces will be described in detail in Section 3.3. It is important to note that some inputs/parameters may appear in more than one interface.

### 3.2 Interface Diagram

The spacecraft interfaces to the AADS flight software through the use of data structures, as shown in Figure 3.2-1 (in this appendix). The required AADS input data is passed into AADS through two data structures. The first input data structure to AADS contains data not likely to change between AADS calls, but which may be changed and uploaded to the spacecraft during the weekly update cycle (to be referred to as spacecraft iLOADS). The second input structure includes parameters and/or data which are likely to change between AADS calls. All outputs from AADS to the spacecraft are provided in a single output data structure. In addition to this interface with the spacecraft, an additional data structure is created within AADS to pass information between the Atmosphere Estimator, Ephemeris Estimator, and the Maneuver Estimator.





*Figure 3.2-1. Spacecraft and AADS Interface Diagram; (1a) Spacecraft Inputs to AADS which will or may change each AADS Call, (1b) Spacecraft Inputs to AADS which are not likely to Change during the AB Mission, (2) AADS Outputs to the Spacecraft, and (3) Intra-AADS*


### 3.3 Individual Interface Characteristics

This section describes the data passed through the various interfaces defined in this document. More detailed information regarding the individual parameters, data type, size, etc. will be provided in Section 3.4 of this appendix.

#### 3.3.1 AAI\_01: Spacecraft to Ephemeris Estimator

The spacecraft (or spacecraft simulator) provides the Ephemeris Estimator the following data:

1. Planetary constants
2. Spacecraft mass and geometry
3. Integrator settings

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>			Page #: 270 of 286

4. Atmospheric entry/exit specification
5. Spacecraft state initialization
6. Planetary ephemerides
7. Processed spacecraft accelerations

### 3.3.2 AAI\_02: Spacecraft to Atmosphere Estimator

The spacecraft (or spacecraft simulator) provides the Atmosphere Estimator the following data:

1. Planetary constants
2. Spacecraft mass and geometry
3. Atmosphere archive index
4. Coordinate transformation
5. Processed spacecraft accelerations
6. Processed spacecraft quaternions

### 3.3.3 AAI\_03: Spacecraft to Thermal Model

**This interface is not implemented.**

The spacecraft (or spacecraft simulator) provides the Thermal Model the following data:

1. Thermocouple

### 3.3.4 AAI\_04: Spacecraft to Maneuver Estimator

The spacecraft provides the Maneuver Estimator the following:

1. Corridor type
2. Corridor limits
3. Corridor target

### 3.3.5 AAI\_05: Ephemeris Estimator to Atmosphere Estimator


The Ephemeris Estimator provides the Atmosphere Estimator estimated data for the following:

1. Previous periapsis
2. Spacecraft altitudes during previous atmospheric pass
3. Spacecraft state estimates during previous atmospheric pass
4. Next periapsis altitude

### 3.3.6 AAI\_06: Ephemeris Estimator to Maneuver Estimator

The Ephemeris Estimator provides the Maneuver Estimator estimated data for the following:

1. Next apoapsis
2. Next periapsis

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #: <b>NESC-RP-09-00605</b>	Version: <b>1.0</b>
Title: <b>Autonomous Aerobraking (Phase 1)</b>		Page #: 271 of 286	

### 3.3.7 AAI\_07: Atmosphere Estimator to Maneuver Estimator

The Atmosphere Estimator provides the Maneuver Estimator estimated data at the next periapsis for the following:

1. Density
2. Scale height
3. Variance
4. Correlation

### 3.3.8 AAI\_08: Maneuver Estimator and Thermal Model

The Thermal Model is called as a function from the Maneuver Estimator. Because of this, the interface between the two modules is simply through a standard function argument list. Although not included in the data structures previously described, the data exchanged between the Maneuver Estimator and the Thermal model includes the estimated:

1. Density (input)
2. Atmospheric pass duration (input)
3. Periapsis speed (input)
4. Orbit period (input)
5. Max spacecraft (solar panel) temperature (output)

The Thermal Model can be used to output the estimated density for a desired spacecraft temperature. In this mode, all other inputs are unchanged.

### 3.3.9 AAI\_09: Ephemeris Estimator to Spacecraft

The Ephemeris Estimator provides the spacecraft the following estimated data:

1. Atmospheric entry
2. Atmospheric exit


### 3.3.10 AAI\_010: Maneuver Estimator to Spacecraft

The Maneuver Estimator provides the spacecraft the following estimated data:

1. Maneuver (magnitude and direction)
2. Maneuver epoch

## 3.4 Detailed Interface Data Description

This section provides more detailed descriptions of the data parameters which reside within the spacecraft (or spacecraft simulator) to AADS interfaces, as well as intra-AADS interface. The names provided here are the same as used in the software header files to define the interface data structures.

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		272 of 286	

### 3.4.1 Spacecraft iLOADS Interface Data Structure Element Descriptions

<b>ee_sun_mu</b>	gravitational constant for the Sun
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km <sup>3</sup> /s <sup>2</sup>

<b>ee_saturn_mu</b>	gravitational constant for Saturn
<i>Interface Identification</i>	AAI_01 used for application at Titan only
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km <sup>3</sup> /s <sup>2</sup>

<b>cb_mu</b>	gravitational constant for the central body
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km <sup>3</sup> /s <sup>2</sup>

<b>ee_ae_pole_ra</b>	right ascension of central body rotational pole
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	deg



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
273 of 286

<b>ee_ae_pole_dec</b>	declination of central body rotational pole
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	deg

<b>ee_IAU_pm</b>	prime meridian with respect to central body IAU vector at epoch
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	deg

<b>ee_ae_omega</b>	central body rotation rate
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	deg/day

<b>ee_ae_re</b>	central body equatorial radius
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km

<b>ee_ae_rp</b>	central body polar radius
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km

<b>ee_oblate_radius</b>	central body oblateness radius (used in gravity field calculations)
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## **Autonomous Aerobraking (Phase 1)**

Page #:  
274 of 286

<b>ee_alt_atm</b>	altitude at which atmospheric interface is assumed to occur
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km

<b>ee_deltat_atm</b>	time offset from atmospheric interface to specify atmospheric entry and exit
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec

<b>ee_stepsize</b>	variable stepsize integrator stepsize initial guess
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec
<i>Allowable values</i>	unconstrained; ignored if/when using fixed stepsize

<b>ee_relative_error</b>	variable stepsize integrator relative error constraint
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	n/a
<i>Allowable values</i>	unconstrained; ignored if/when using fixed stepsize

<b>ee_absolute_error</b>	variable stepsize integrator absolute error constraint
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	n/a
<i>Allowable values</i>	unconstrained; ignored if/when using fixed stepsize



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:

275 of 286

<b>ee_mnvr_step</b>	fixed stepsize to use while integrating through maneuver acceleration data
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec
<i>Allowable values</i>	unconstrained; variable stepsize used if = 0

<b>ee_atmos_step</b>	fixed stepsize to use while integrating through atmospheric pass acceleration data
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec
<i>Allowable values</i>	unconstrained; variable stepsize used if = 0

<b>ae_start_orbit</b>	orbit number at the start of AADS AB; sets index within the atmosphere archive to begin data retrieval and write output
<i>Interface Identification</i>	AAI_02
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

<b>ae_first_orbit_data</b>	first orbit index within the atmosphere archive which contains data acceptable for use by the Atmosphere Estimator
<i>Interface Identification</i>	AAI_02
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

<b>sc_area</b>	spacecraft aerodynamic / wetted reference area
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	m <sup>2</sup>



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
276 of 286


<b>sc_mass</b>	spacecraft mass
<i>Interface Identification</i>	AAI_01, AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	km

<b>ae_phib2a</b>	spacecraft body to aerodynamic Direction Cosine Matrix
<i>Interface Identification</i>	AAI_02
<i>Type</i>	double
<i>Dimension</i>	[9] vector is created by appending rows from matrix: [(1,1) (1,2) (1,3) (2,1) (2,2) (2,3) (3,1) (3,2) (3,3)]
<i>Units</i>	n/a

<b>me_corr_type</b>	operational corridor type
<i>Interface Identification</i>	AAI_04
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a
<i>Allowable values</i>	1 = temperature 2 = density 3 = heatrate 4 = dynamic pressure 5 = altitude

<b>me_corr_up</b>	operational corridor upper limit
<i>Interface Identification</i>	AAI_04
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	if: me_corr_type = 1, deg C me_corr_type = 2, kg/m <sup>3</sup> me_corr_type = 3, W/cm <sup>2</sup> me_corr_type = 4, Pa me_corr_type = 5, m



	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:			Page #:
<b>Autonomous Aerobraking (Phase 1)</b>			277 of 286


<b>me_corr_low</b>	operational corridor lower limit
<i>Interface Identification</i>	AAI_04
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	if: me_corr_type = 1, deg C me_corr_type = 2, kg/m <sup>3</sup> me_corr_type = 3, W/cm <sup>2</sup> me_corr_type = 4, Pa me_corr_type = 5, m

<b>me_targ_fac</b>	operational corridor target (expressed as a percentage of the total corridor width)
<i>Interface Identification</i>	AAI_04
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	n/a

### 3.4.2 Spacecraft Input Data Structure Element Descriptions

<b>sc2ee_init_flag</b>	flag which indicates whether a new state is available for re-initialization of the AADS integrator
<i>Interface Identification</i>	AAI_01
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	none
<i>Allowable values</i>	0 = no update available !0 = update available

<b>sc2ee_et_init</b>	epoch associated with new AADS integrator initialization state
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)
<i>Allowable values</i>	unconstrained; ignored if sc2ee_init_flag = 0

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:			Page #:
<b>Autonomous Aerobraking (Phase 1)</b>			278 of 286

<b>sc2ee_init_state</b>	AADS integration initialization state
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[6]
<i>Element order</i>	position = [1-3] velocity = [4-6]
<i>Units</i>	position, km velocity, km/s
<i>Coordinate system</i>	central body, EME2000
<i>Allowable values</i>	unconstrained; ignored if sc2ee_init_flag = 0

<b>sc2ee_sun</b>	Sun ephemeris / Chebyshev coefficients
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[4+NRECS*(NR+1)], where NRECS = 20 NR = 35 (sized to accommodate maximum required)
<i>Units</i>	n/a

<b>sc2ee_n_sun</b>	number of (usable) elements in sc2ee_sun
<i>Interface Identification</i>	AAI_01
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

<b>sc2ee_bary</b>	system barycenter ephemeris / Chebyshev coefficients
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[4+NRECS*(NR+1)], where NRECS = 20 NR = 32 for application at Venus NR = 35 for application at Mars NR = 23 for application at Titan (sized to accommodate maximum required)
<i>Units</i>	n/a



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
279 of 286

<b>sc2ee_n_bary</b>	number of (usable) elements in sc2ee_bary
<i>Interface Identification</i>	AAI_01
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

<b>sc2ee_plan</b>	third body planetary ephemeris / Chebyshev coefficients
<i>Interface Identification</i>	AAI_01 - used for application at Titan only
<i>Type</i>	double
<i>Dimension</i>	[4+NRECS*(NR+1)], where NRECS = 20 NR = 68 (sized to accommodate maximum possible)
<i>Units</i>	n/a

<b>sc2ee_n_plan</b>	number of (usable) elements in sc2ee_plan
<i>Interface Identification</i>	AAI_01 - used for application at Titan only
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

<b>sc2ee_sat</b>	satellite ephemeris / Chebyshev coefficients
<i>Interface Identification</i>	AAI_01 - used for application at Titan only
<i>Type</i>	double
<i>Dimension</i>	[4+NRECS*(NR+1)], where NRECS = 20 NR = 80 (sized to accommodate maximum possible)
<i>Units</i>	n/a

<b>sc2ee_n_sat</b>	number of (usable) elements in sc2ee_sat
<i>Interface Identification</i>	AAI_01 - used for application at Titan only
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## **Autonomous Aerobraking (Phase 1)**


Page #:  
280 of 286

<b>sc2ee_accel</b>	spacecraft acceleration vector
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	POST2 = [MAX_ACC][3] HFS = [3*MAX_ACC] where MAX_ACC is sized to accommodate maximum duration drag pass with data provided at 10 Hz
<i>Units</i>	m/s <sup>3</sup>
<i>Coordinate system</i>	central body EME2000

<b>sc2ee_et_accel</b>	spacecraft acceleration vector time tag
<i>Interface Identification</i>	AAI_01
<i>Type</i>	double
<i>Dimension</i>	[MAX_ACC] where MAX_ACC is sized to accommodate maximum duration drag pass with data provided at 10 Hz
<i>Units</i>	sec (past J2000)

<b>sc2ee_n_data</b>	number of (usable) acceleration data elements in sc2ee_et_accel and sc2ee_ee_accel
<i>Interface Identification</i>	AAI_01
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

<b>sc2ae_accel</b>	spacecraft acceleration vector array
<i>Interface Identification</i>	AAI_02
<i>Type</i>	double
<i>Dimension</i>	POST2 = [MAX_STT][3] HFS = [3*MAX_STT] where MAX_STT is sized to accommodate maximum duration drag pass with data provided at 1 Hz
<i>Units</i>	m/s <sup>3</sup>
<i>Coordinate system</i>	central body EME2000

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:			Page #:
<b>Autonomous Aerobraking (Phase 1)</b>			281 of 286

<b>sc2ae_quats</b>	central body EME2000 to spacecraft body frame quaternion
<i>Interface Identification</i>	AAI_02
<i>Type</i>	double
<i>Dimension</i>	POST2 = [MAX_STT][4] HFS = [4*MAX_STT] where MAX_STT is sized to accommodate maximum duration drag pass with data provided at 1 Hz
<i>Units</i>	n/a, scalar last

<b>sc2ae_et_atm_pass</b>	epoch corresponding to start of sc2ae_accel and sc2ae_quats
<i>Interface Identification</i>	AAI_02
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)

<b>sc2ae_num_pts</b>	number of (usable) data elements in sc2ae_accel and sc2ae_quats
<i>Interface Identification</i>	AAI_02
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

### 3.4.3 AADS Internal Data Structure Element Descriptions

<b>ee2me_et_apo</b>	estimated time of next apoapsis
<i>Interface Identification</i>	AAI_06
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)

<b>ee2me_apo_state</b>	estimated next apoapsis state
<i>Interface Identification</i>	AAI_06
<i>Type</i>	double
<i>Dimension</i>	[6]
<i>Element order</i>	position = [1-3] velocity = [4-6]
<i>Units</i>	position, km velocity, km/s
<i>Coordinate system</i>	central body, EME2000



# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## Autonomous Aerobraking (Phase 1)

Page #:  
282 of 286


<b>ee2me_et_peri</b>	estimated time of next periapsis
<i>Interface Identification</i>	AAI_06
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)

<b>ee2me_peri_state</b>	estimated next periapsis state
<i>Interface Identification</i>	AAI_06
<i>Type</i>	double
<i>Dimension</i>	[6]
<i>Element order</i>	position = [1-3] velocity = [4-6]
<i>Units</i>	position, km velocity, km/s
<i>Coordinate system</i>	central body, EME2000

<b>ee2me_peri_rstate</b>	estimated next periapsis state
<i>Interface Identification</i>	AAI_06
<i>Type</i>	double
<i>Dimension</i>	[6]
<i>Element order</i>	position = [1-3] velocity = [4-6]
<i>Units</i>	position, km velocity, km/s
<i>Coordinate system</i>	central body rotating

<b>ee2ae_alt_peri</b>	estimated altitude of next periapsis
<i>Interface Identification</i>	AAI_05
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	m

<b>ee2ae_et_peri_prev</b>	estimated time of previous periapsis
<i>Interface Identification</i>	AAI_05
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:			Page #:
<b>Autonomous Aerobraking (Phase 1)</b>			283 of 286


<b>ee2ae_et_alt_prev</b>	estimated altitude of previous periapsis
<i>Interface Identification</i>	AAI_05
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	m

<b>ee2ae_alt_atm</b>	estimated altitude profile during previous atmospheric pass
<i>Interface Identification</i>	AAI_05
<i>Type</i>	double
<i>Dimension</i>	[MAX_STT], where MAX_STT is sized to accommodate maximum duration drag pass with data provided at 1 Hz
<i>Units</i>	m

<b>ee2ae_state_atm</b>	estimated spacecraft states during previous atmospheric pass
<i>Interface Identification</i>	AAI_05
<i>Type</i>	double
<i>Dimension</i>	[MAX_STT][6], where MAX_STT is sized to accommodate maximum duration drag pass with data provided at 1 Hz
<i>Element order</i>	position = [1-3] velocity = [4-6]
<i>Units</i>	position, km velocity, km/s
<i>Coordinate system</i>	central body, EME2000

<b>ee2ae_num_pts</b>	number of (usable) data elements in ee2ae_alt_atm and ee2ae_state_atm arrays
<i>Interface Identification</i>	AAI_05
<i>Type</i>	integer
<i>Dimension</i>	[1]
<i>Units</i>	n/a

<b>ae2me_rho_plus</b>	estimated atmospheric density at next periapsis
<i>Interface Identification</i>	AAI_07
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	kg/m <sup>3</sup>

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		284 of 286	

<b>ae2me_hs_plus</b>	estimated atmospheric scale height at next periapsis
<i>Interface Identification</i>	AAI_07
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	m

<b>ae2me_sigma_rho_plus</b>	estimated $1\sigma$ variance in ae2me_rho_plus
<i>Interface Identification</i>	AAI_07
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	kg/m <sup>3</sup>

<b>ae2me_sigma_hs_plus</b>	estimated $1\sigma$ variance in ae2me_hs_plus
<i>Interface Identification</i>	AAI_07
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	m

<b>ae2me_corr_rho_hs_plus</b>	correlation between ae2me_rho_plus and ae2me_hs_plus
<i>Interface Identification</i>	AAI_07
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	n/a

### 3.4.4 AADS Output Interface Structure Element Descriptions

<b>ee2sc_et_entry</b>	estimated epoch of next atmospheric entry
<i>Interface Identification</i>	AAI_09
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)





# NASA Engineering and Safety Center Technical Assessment Report

Document #:  
**NESC-RP-  
09-00605**

Version:  
**1.0**

Title:

## **Autonomous Aerobraking (Phase 1)**

Page #:  
285 of 286


<b>ee2sc_entry_state</b>	estimated spacecraft state at next atmospheric entry
<i>Interface Identification</i>	AAI_09
<i>Type</i>	double
<i>Dimension</i>	[6]
<i>Element order</i>	position = [1-3] velocity = [4-6]
<i>Units</i>	position, km velocity, km/s
<i>Coordinate system</i>	central body, EME2000

<b>ee2sc_et_exit</b>	estimated epoch of next atmospheric exit
<i>Interface Identification</i>	AAI_09
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)

<b>ee2sc_exit_state</b>	estimated spacecraft state at next atmospheric exit
<i>Interface Identification</i>	AAI_09
<i>Type</i>	double
<i>Dimension</i>	[6]
<i>Element order</i>	position = [1-3] velocity = [4-6]
<i>Units</i>	position, km velocity, km/s
<i>Coordinate system</i>	central body, EME2000

<b>me2sc_et_burn</b>	epoch of next corridor control maneuver
<i>Interface Identification</i>	AAI_10
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	sec (past J2000)

<b>me2sc_dv_to_burn</b>	delta-v required for next corridor control maneuver
<i>Interface Identification</i>	AAI_10
<i>Type</i>	double
<i>Dimension</i>	[1]
<i>Units</i>	m/s

	<b>NASA Engineering and Safety Center Technical Assessment Report</b>	Document #:	Version:
		<b>NESC-RP-09-00605</b>	<b>1.0</b>
Title:		Page #:	
<b>Autonomous Aerobraking (Phase 1)</b>		286 of 286	

<b>me2sc_dvi_vec</b>	direction / unit vector of next corridor control maneuver
<i>Interface Identification</i>	AAI_10
<i>Type</i>	double
<i>Dimension</i>	[3]
<i>Units</i>	n/a
<i>Coordinate system</i>	central body EME2000

## 4 References

1. S.A. Striepe et al., "Program To Optimize Simulated Trajectories (POST II): Volume 2, Utilization Manual." Martin Marietta Corporation, 2004.
2. G.L. Brauer et al., "Program To Optimize Simulated Trajectories (POST): Volume 1, Formulation Manual." Martin Marietta Corporation, 1990.

**REPORT DOCUMENTATION PAGE**

*Form Approved  
OMB No. 0704-0188*

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.  
**PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

<b>1. REPORT DATE (DD-MM-YYYY)</b> 01-01 - 2012		<b>2. REPORT TYPE</b> Technical Memorandum		<b>3. DATES COVERED (From - To)</b> February 2010 - November 2011	
<b>4. TITLE AND SUBTITLE</b> Development of Autonomous Aerobraking (Phase 1)				<b>5a. CONTRACT NUMBER</b>	
				<b>5b. GRANT NUMBER</b>	
				<b>5c. PROGRAM ELEMENT NUMBER</b>	
<b>6. AUTHOR(S)</b> Murri, Daniel G.; Powell, Richard W.; Prince, Jill L.				<b>5d. PROJECT NUMBER</b>	
				<b>5e. TASK NUMBER</b>	
				<b>5f. WORK UNIT NUMBER</b> 869021.05.07.01.14	
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> NASA Langley Research Center Hampton, VA 23681-2199				<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>  L-20114 NESC-RP-09-00605	
<b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> National Aeronautics and Space Administration Washington, DC 20546-0001				<b>10. SPONSOR/MONITOR'S ACRONYM(S)</b>  NASA	
				<b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b>  NASA/TM-2012-217328	
<b>12. DISTRIBUTION/AVAILABILITY STATEMENT</b> Unclassified - Unlimited Subject Category 20 Spacecraft Propulsion and Power Availability: NASA CASI (443) 757-5802					
<b>13. SUPPLEMENTARY NOTES</b>					
<b>14. ABSTRACT</b> The NASA Engineering and Safety Center received a request from Mr. Daniel Murri (NASA Technical Fellow for Flight Mechanics) to develop an autonomous aerobraking capability. An initial evaluation for all phases of this assessment was approved to proceed at the NESC Review Board meeting. The purpose of phase 1 of this study was to provide an assessment of the feasibility of autonomous aerobraking. During this phase, atmospheric, aerodynamic, and thermal models for a representative spacecraft were developed for both the onboard algorithm known as Autonomous Aerobraking Development Software, and a ground-based "truth" simulation developed for testing purposes. The results of the phase 1 assessment are included in this report.					
<b>15. SUBJECT TERMS</b> autonomous aerobraking; Autonomous Aerobraking Development Software; NASA Engineering and Safety Center; Deep Space Network					
<b>16. SECURITY CLASSIFICATION OF:</b>			<b>17. LIMITATION OF ABSTRACT</b>	<b>18. NUMBER OF PAGES</b>	<b>19a. NAME OF RESPONSIBLE PERSON</b>
<b>a. REPORT</b>	<b>b. ABSTRACT</b>	<b>c. THIS PAGE</b>			STI Help Desk (email: help@sti.nasa.gov)
U	U	U	UU	291	<b>19b. TELEPHONE NUMBER (Include area code)</b> (443) 757-5802