

Draft: Work in Progress

Input Range Testing for the General Mission Analysis Tool (GMAT)

DRAFT



Steve Hughes and Edwin Dove
NASA Goddard Space Flight Center
Greenbelt, MD 20771

July 12, 2007

Draft: Work in Progress

2

Draft: Work in Progress

Contents

1	Test Plan Overview	7
1.1	Tests to Verify Disallowed Fields are Rejected	7
1.2	Input Range Tests for Fields that Accept Numeric Values	8
1.3	Input Range Tests for Fields that Accept Strings	8
1.3.1	Tests for Strings that are Restricted to a Predefined List	9
1.3.2	Tests for Strings that are not Restricted to a Predefined List	9
1.4	Behavior When a Disallowed Object Field is Encountered	9
1.4.1	Error Message for a Disallowed Field	9
1.4.2	Error Message for an Invalid Input for an Allowed Field	10
1.5	Notes	10
2	Objects and Resources	11
2.1	Spacecraft and Hardware Fields	11
2.2	Propagator Fields	21
2.3	Maneuvers	25
2.4	Solver Fields	26
2.5	Plots and Reports	28
2.6	Solar System, Celestial Bodies and other Space Points	33
3	Commands and Events	39
3.1	Propagation	39
3.2	Control Flow	41
3.3	Solver-related	45
3.4	Miscellaneous	52

Draft: Work in Progress

CONTENTS

Draft: Work in Progress

List of Tables

1.1	Input Range Tests for Fields that Accept Real Numbers and Integers	8
2.1	Fields Associated with a Spacecraft Orbit State (Orbit Tab)	11
2.2	Fields Associated with Spacecraft Physical Properties (Ballistic/Mass Tab)	15
2.3	Fields Associated with Spacecraft Attitude State (Attitude Tab)	16
2.4	Fields Associated with a Spacecraft Tank (Tanks Tab)	18
2.5	Fields Associated with a Spacecraft Thruster (Actuators Tab)	19
2.6	Fields Associated with a Force Model	21
2.7	Fields Associated with an Integrator	23
2.8	Fields Associated with an Impulsive Burn	25
2.9	Fields Associated with a Finite Burn	26
2.10	Fields Associated with the fmincon Solver	27
2.11	Fields Associated with a Differential Corrector	28
2.12	Fields Associated with OpenGL Plots	28
2.13	Fields Associated with Report Files	32
2.14	Fields Associated with XY-Plots	33
2.15	Fields Associated with the Solar System	34
2.16	Fields Associated with a Libration Point	34
2.17	Fields Associated with a BaryCenter	35
2.18	Fields Associated with Celestial Bodies	35
2.19	Fields Associated with a Coordinate System	36
2.20	Fields Associated with MATLAB Functions	37
3.1	Propagate Command	39
3.2	If Command	41
3.3	While Command	42
3.4	For Command	43

Draft: Work in Progress

3.5	Target Command	45
3.6	Optimize Command	45
3.7	Achieve Command	46
3.8	Vary Command	46
3.9	Minimize Command	48
3.10	NonLinearConstraint Command	51
3.11	Maneuver Command	52
3.12	BeginFiniteBurn Command	52
3.13	EndFiniteBurn Command	52
3.14	CallFunction Command	53
3.15	Toggle Command	54
3.16	Report Command	54
3.17	ScriptEvent Command	56
3.18	Pause Command	56
3.19	Stop Command	56
3.20	Save Command	57

Draft: Work in Progress

Chapter 1

Test Plan Overview

This document contains a test plan for testing input values to the General Mission Analysis Tool (GMAT). The plan includes four primary types of information, which rigorously define all tests that should be performed to validate that GMAT will accept allowable inputs and deny disallowed inputs. The first is a complete list of all allowed object fields in GMAT. The second type of information, is test input to be attempted for each field. The third type of information is allowable input values for all objects fields in GMAT. The final piece of information is how GMAT should respond to both valid and invalid information.

It is VERY important to note that the tests below must be performed for both the Graphical User Interface and the script!! The examples are illustrated using a scripting perspective, because it is simpler to write up. However, the test must be performed for both interfaces to GMAT.

1.1 Tests to Verify Disallowed Fields are Rejected

The tables contained in this document detail all allowed fields that the user should have access to in the GMAT user interfaces including the script and GUI. The tables are organized in terms of objects. Each object has a table that defines the allowable field names. The tables contain a complete definition of what object fields the user should have access to in the user interface. The user should not have access to or be able to set fields that are not contained in the field lists. This includes being able to set the field from the GUI or script, or seeing the field when the "Show Script" button is hit on the object's associated dialogue box or a mission is saved.

These following tests should be performed to ensure that the user does not have access to fields that are read only or some other type internal field.

- Using show script and by saving missions, verify that only fields that are defined in the tables below are user settable and accessible. If a field name does not appear in the tables below, it should not appear in show script, or a saved mission.
- Verify that the setting for fields in a saved mission and show script meet the allowable values. For example, currently if you hit show script on an OpenGL plot, you'll see a line like this

```
OpenGLPlot1.ViewPointVector = Vector;
```

The word "Vector" is not an allowable value for the ViewPointVector field.

- Try GMAT Object.DoesNotExist = 1 for all objects;

1.2 Input Range Tests for Fields that Accept Numeric Values

This section documents tests to be performed to ensure that GMAT accepts allowable values for numeric fields, and denies disallowed values. The tests apply to fields that accept real numbers or integers. The tests ensure that both allowed and disallowed values are attempted in the test process.

The tests are described in Table 1.1. Let's illustrate one of the tests and select the case where the input is a real number such that $Real \geq c$. The ECC field on the Spacecraft object is in this category. According to the Table 1.1, the tests to be performed on the ECC field are to try inputting values of -1, 0, 1, 'DoesNotExist', and '' (Note: according to the notation in the table, $c = 0$ for the ECC field). In the GMAT script syntax, the tests to be performed on the ECC field are:

```
GMAT DefaultSC.ECC = -1.0
GMAT DefaultSC.ECC = 0.0
GMAT DefaultSC.ECC = 1.0
GMAT DefaultSC.ECC = DoesNotExist
GMAT DefaultSC.ECC =
```

We see in Table 2.1 that the allowable options are $ECC \geq 0$, so we know that only the second and third lines above should pass the test. The first, fourth, and fifth lines should throw an error message consistent with the format discussed in Sec. 1.4.

Table 1.1: Input Range Tests for Fields that Accept Real Numbers and Integers

Allowable Input Type	Test Input
Real	-1.01 , 1.01 , 'DoesNotExist' , ''
Real $\geq c$	$c-1$, c , $c+1$, 'DoesNotExist' , ''
Real $\leq c$	
Real $> c$	
Real $< c$	
Real $\neq c$	
$c_1 > Real > c_2$	$c_1 - 1$, c_1 , $(c_1 + c_2)/2$, c_2 , $c_2 + 1$, 'DoesNotExist' , ''
$c_1 \geq Real \geq c_2$	$c_1 - 1$, c_1 , $(c_1 + c_2)/2$, c_2 , $c_2 + 1$, 'DoesNotExist' , ''
Integer	-1 , 1 , 1.01 , 'DoesNotExist' , ''
Integer $\geq c$	$c - 0.5$, $c - 1$, c , $c + 0.5$, $c + 1$, 'DoesNotExist' , ''
Integer $\leq c$	
Integer $> c$	
Integer $< c$	
$c_1 > Integer > c_2$	$c_1 - 0.5$, c_1 , $\text{round}((c_1 + c_2)/2)$, c_2 , $c_2 + 0.5$, 'DoesNotExist' , ''
$c_1 \geq Integer \geq c_2$	$c_1 - 0.5$, c_1 , $\text{round}((c_1 + c_2)/2)$, c_2 , $c_2 + 0.5$, 'DoesNotExist' , ''

1.3 Input Range Tests for Fields that Accept Strings

There are numerous fields in GMAT that accept strings as input values. For some fields, the list of strings that are acceptable is predefined, and the user cannot add to the list of possible values. A good example of this type of field is the StateType field on the Spacecraft object. The allowable inputs for the StateType are restricted to the following: Cartesian, Keplerian, ModifiedKeplerian, SphericalAZFPA, SphericalRADEC, Equinoctial . The user cannot add to this list.

The second type of field that accepts strings as input is best illustrated by the CoordinateSystem field on the Spacecraft object. For the CoordinateSystem field, the user can select from the default list of coordinate systems such as EarthMJ2000Eq, EarthMJ2000Ec, or EarthFixed. However, the user can also select from any user-defined coordinate system. So if the user has defined a coordinate system called EarthSunRotating, then they can choose this as the coordinate system defined by the CoordinateSystem field object on the Spacecraft object.

Draft: Work in Progress

1.4. BEHAVIOR WHEN A DISALLOWED OBJECT FIELD IS ENCOUNTERED

9

For all string fields we should test that numeric values (use 1.0 for value) and the string `DoesNotExist` are not accepted by GMAT. Now let's look at the tests for these two types of fields.

1.3.1 Tests for Strings that are Restricted to a Predefined List

For the first type, where the user can only select from a list of predefined strings, we should attempt two of the strings. The first should be the default value, the second should not be a default value. As an example, for the `StateType` field on the spacecraft object, we should test the following two inputs:

```
DefaultSC.StateType = Cartesian;  
DefaultSC.StateType = Keplerian;
```

We should perform similar tests for all fields of this type.

1.3.2 Tests for Strings that are not Restricted to a Predefined List

For the second type of string input, where the user can select from a predefined list of strings as well as a user defined list, we should test one value from the predefined list. Then we should create an object of the type appropriate to the field, and test that the object name can be used as input to the field. Let's look again at the `CoordinateSystem` field on the spacecraft object as an example. We should perform the following:

```
Create CoordinateSystem MyCoordSys;  
DefaultSC.CoordinateSystem = EarthMJ2000Eq;  
DefaultSC.CoordinateSystem = MyCoordSys;
```

We should perform similar tests for all fields of this type.

1.4 Behavior When a Disallowed Object Field is Encountered

There are two types of error handling described in this section. The first, is when a user tries to set a field that does not exist, or that the user is not allowed access to. The second type, is when the user tries to set an allowed field, but does not use an allowable setting for the field.

1.4.1 Error Message for a Disallowed Field

If a field that is not allowed is encountered by GMAT, GMAT should output the following error message:

The field name " " on object " " is not permitted. As an example, if the user inputs the following,

```
DefaultSC.DoesNotExist = 5
```

then GMAT should respond with the following error message:

The field name "DoesNotExist" on object DefaultSC is not permitted.

Draft: Work in Progress

10

CHAPTER 1. TEST PLAN OVERVIEW

1.4.2 Error Message for an Invalid Input for an Allowed Field

If the user inputs a disallowed value for an allowed field GMAT should respond with the following message

The value of “ ” for field “ ” on object “ ” is not an allowed value. The allowed values are: [insert options here].

For example, if the user input the following line of script

```
DefaultSC.StateType = DoesNotExist
```

then GMAT should respond with the following error message:

The value of DoesNotExist for field StateType on object DefaultSC is not an allowed value. The allowed values are: [Cartesian, Keplerian, ModifiedKeplerian, SphericalAZFPA, SphericalRADEC , Equinoctial].

The allowed values for all other fields are found in the the tables below.

1.5 Notes

ABM integrator settings.

For Differential corrector, UseCentralDifferencing is not implemented in base code.

Normalization of quaternion, are we doing this?

Changed TargetStatus to SolverIterations on XYplots and OpenGL plots, and added the field to Reports.

The following fields have been changed according to this document

SolarSystem.RotationDataSource = DE405 or false IAU2002 (used to be 0 or 1)

Equinoctial elements have changed from the labels on the GUI.

Spacecraft.DateFormat -> Spacecraft.EpochFormat for consistency with the rest of GMAT

ABM integrator settings changed

LowerError -> MinIntegrationError

TargetError -> NomIntegrationError

Chapter 2

Objects and Resources

2.1 Spacecraft and Hardware Fields

Table 2.1: Fields Associated with a Spacecraft Orbit State (Orbit Tab)

Field	Options and Description
StateType	Default: Cartesian. Options: [Cartesian, Keplerian, ModifiedKeplerian, SphericalAZFPA, SphericalRADEC, Equinoctial]. The StateType field allows the user to configure the type of state vector that they wish to use. The Statetype field has a dependency upon the CoordinateSystem field. If the Coordinate System chosen by the user does not have a gravitational body at the origin, then the state types Keplerian, ModifiedKeplerian, and Equinoctial are not permitted. This is because these state types require a μ value. Units: N/A. When the Keplerian or ModifiedKeplerian state types are selected, the Anomaly Type field becomes visible.
Coordinate System	Default: EarthMJ2000Eq. Options: [EarthMJ2000Eq, EarthMJ2000Ec, EarthFixed, or any user defined system]: The Coordinate System field allows the user to choose which coordinate system with which to define the orbit state vector. The CoordinateSystem field has a dependency upon the StateType field. If the Coordinate System chosen by the user does not have a gravitational body at the origin, then the state types Keplerian, ModifiedKeplerian, and Equinoctial are not permitted. This is because these state types require a μ value. Units: N/A.
EpochFormat	Default: TAIModJulian. Options: [A1ModJulian, TAIModJulian, UTCModJulian, TTModJulian, A1Gregorian, TAIGregorian, UTCGregorian, TTGregorian]: The DateFormat field allows the user to specify the format for defining a spacecraft's initial epoch. DateFormat determines both the time system (TAI, TT, etc) and the time format (MJD or Gregorian). Units: N/A.
Epoch	Default: 21545.000000000. Options: [See Comments]: The Epoch field allows the user to specify the initial spacecraft epoch. The format of the epoch must be consistent with the DateFormat field. If DateFormat is of the "MJD" type, then the epoch is in Modified Julian format. If DateFormat is a "Gregorian Type", the format is similar to 01 Jan 2000 12:00:00.000. Units: MJD - days, Gregorian - N/A.

Table 2.1: (Fields Associated with a Spacecraft Orbit State (Orbit Tab). continued)

Field	Options and Description
AnomalyType	Default: TA. Options: [TA, MA, EA, HA]: The Epoch field allows the user to specify the to select the AnomalyType needed for the Keplerian or ModifiedKeplerian spacecraft state. In the scripting environment, AnomalyType is not used. Units: N/A.

Fields associated with Cartesian state.

X	Default: 7100. Options: [Real Number]: X is the x-component of the Spacecraft state in the coordinate system chosen in the Spacecraft CoordinateSystem field. Units: km.
Y	Default: 0. Options: [Real Number]: Y is the y-component of the Spacecraft state in the coordinate system chosen in the Spacecraft CoordinateSystem field. Units: km.
Z	Default: 1300. Options: [Real Number]: Z is the z-component of the Spacecraft state in the coordinate system chosen in the Spacecraft CoordinateSystem field. Units: km.
VX	Default: 0. Options: [Real Number]: VX is the x-component of the Spacecraft velocity in the coordinate system chosen in the Spacecraft CoordinateSystem field. Units: km/sec.
VY	Default: 7.35. Options: [Real Number]: VY is the y-component of the Spacecraft velocity in the coordinate system chosen in the Spacecraft CoordinateSystem field. Units: km/sec.
VZ	Default: 1.0. Options: [Real Number]: VZ is the z-component of the Spacecraft velocity in the coordinate system chosen in the Spacecraft CoordinateSystem field. Units: km/sec.

NOTE: Default values for the remaining state types are obtained through transformations of the default Cartesian spacecraft state values. The Keplerian, ModifiedKeplerian, and Equinoctial are dependant on the origin of the CoordinateSystem, because the state types require a μ value.

Fields associated with Keplerian state.

SMA	Default: 7191.938817629. Options: [Real Number \neq 0]: The SMA field is the spacecraft orbit's osculating Keplerian semimajor axis in coordinate system chosen in the Spacecraft CoordinateSystem field. SMA must be strictly greater than or less than zero. For circular and elliptical ($0 \leq \text{ECC} < 1$) orbits SMA should only be a positive Real Number and for hyperbolic orbits ($\text{ECC} > 1$) SMA should only be a negative Real Number. GMAT does not support the creation of parabolic orbits. Units: km.
-----	---

Draft: Work in Progress

Table 2.1: (Fields Associated with a Spacecraft Orbit State (Orbit Tab). continued)

Field	Options and Description
ECC	Default: 0.024549749. Options: [$0 \leq$ Real Number, $ECC \neq 1$]: The ECC field is the spacecraft orbit's osculating eccentricity. ECC must be greater than or equal to zero but not equal to one (GMAT does not support parabolic orbits). Note: ECC can be greater than one. See the SMA description for additional restrictions to the allowable values of ECC. Units: Dimensionless.
INC	Default: 12.850080057. Options: [Real Number]: The INC field is the spacecraft orbit's osculating inclination, in degrees, w/r/t to the selected coordinate system. Units: degrees.
AOP	Default: 314.190551536. Options: [Real Number]: The AOP field is the spacecraft orbit's osculating argument of periapsis, in degrees, w/r/t to the selected coordinate system. Units: degrees.
RAAN	Default: 306.614802195. Options: [Real Number]: The RAAN field is the spacecraft orbit's osculating right ascension of the ascending node, in degrees, w/r/t to the selected coordinate system. Units: degrees.
TA	Default: 99.887749332. Options: [Real Number]: The TA field is the spacecraft orbit's osculating true anomaly. Units: degrees.
MA	Default: 97.107826639. Options: [Real Number]: The MA field is the spacecraft orbit's osculating mean anomaly. Units: degrees.
EA	Default: 98.498977103. Options: [Real Number]: The EA field is the spacecraft orbit's osculating eccentric anomaly. Units: degrees.
HA	Default: 0.000000000. Options: [Real Number]: The HA field is the spacecraft orbit's osculating hyperbolic anomaly. Units: degrees.
Fields associated with ModifiedKeplerian state.	
RadApo	Default: 7015.378524789. Options: [Real Number $\neq 0$]: The RadApo field is the spacecraft orbit's osculating radius of apoapsis. RadApo must be strictly greater than or less than zero. When RadApo is negative, the orbit is hyperbolic. Units: km.
RadPer	Default: 7368.4991104681 Options: [Real Number > 0]: The RadPer field is the spacecraft orbit's osculating radius of periapsis. RadPer must be greater than zero. Units: km.
INC	See the Keplerian state section for a description on this field.
AOP	See the Keplerian state section for a description on this field.
RAAN	See the Keplerian state section for a description on this field.
TA	See the Keplerian state section for a description on this field.
MA	See the Keplerian state section for a description on this field.

Table 2.1: (Fields Associated with a Spacecraft Orbit State (Orbit Tab). continued)

Field	Options and Description
EA	See the Keplerian state section for a description on this field.
HA	See the Keplerian state section for a description on this field.

Fields associated with SphericalAZFPA state.

RMAG	Default: 7218.03297304. Options: [Real Number > 0]: The RMAG field allows the user to set the magnitude of the spacecraft's position vector. Units: km.
RA	Default: 0. Options: [Real Number]: The RA field allows the user to set the spacecraft's right ascension. Units: degrees.
DEC	Default: 10.3758449200. Options: [Real Number]: The DEC field allows the user to set the spacecraft's declination. Units: degrees.
VMAG	Default: 7.41771528167. Options: [Real Number ≥ 0]: The VMAG field allows the user to set the magnitude of the spacecraft's velocity. Units: km/sec.
AZI	Default: 82.377421681. Options: [Real Number]: The AZI field allows the user to set the spacecraft's azimuth angle. Units: degrees.
FPA	Default: 88.60870365370. Options: [Real Number]: The FPA allows the user to set a spacecraft's flight path angle. Units: degrees.

Fields associated with SphericalRADEC state.

RMAG	See the SphericalAZFPA state section for a description on this field.
RA	See the SphericalAZFPA state section for a description on this field.
DEC	See the SphericalAZFPA state section for a description on this field.
VMAG	See the SphericalAZFPA state section for a description on this field.
RAV	Default: 90. Options: [Real Number]: The RAV field allows the user to set the right ascension of the spacecraft's velocity. Units: degrees.
DECV	Default: 7.7477720361. Options: [Real Number]: The DECV field allows the user to set the declination of the spacecraft's velocity. Units: degrees.

Fields associated with Equinoctial elements.

SMA	See the Keplerian state section for a description on this field.
-----	--

Table 2.1: (Fields Associated with a Spacecraft Orbit State (Orbit Tab). continued)

Field	Options and Description
h	Default: -0.024234314. Options: [Real Number]: The h field is the projection of the eccentricity vector onto the y_{ep} axes. The \mathcal{F}_{ep} system is a system used in calculating the equinoctial elements and is beyond the scope of this discussion. The GMAT Mathematical Specifications document discusses \mathcal{F}_{ep} and the calculation of the equinoctial elements in detail. Units: None.
k	Default: -0.003922779. Options: [Real Number]: The k field is the projection of the eccentricity vector onto the x_{ep} axes. The \mathcal{F}_{ep} system is a system used in calculating the equinoctial elements and is beyond the scope of this discussion. The GMAT Mathematical Specifications document discusses \mathcal{F}_{ep} and the calculation of the equinoctial elements in detail. Units: None.
p	Default: -0.090388347. Options: [Real Number]: The p field is the projection of the N vector onto the y_{ep} axes. The N vector and the \mathcal{F}_{ep} system are used in calculating the equinoctial elements and are beyond the scope of this discussion. The GMAT Mathematical Specifications document discusses N and \mathcal{F}_{ep} and the calculation of the equinoctial elements in detail. Units: None.
q	Default: 0.067164549. Options: [Real Number]: The q field is the projection of the N vector onto the x_{ep} axes. The N vector and the \mathcal{F}_{ep} system are used in calculating the equinoctial elements and are beyond the scope of this discussion. The GMAT Mathematical Specifications document discusses N and \mathcal{F}_{ep} and the calculation of the equinoctial elements in detail. Units: None.
MeanLongitude	Default: 3.16359946. Options: [Real Number]: The MeanLongitude field is the the spacecraft's mean longitude. The GMAT Mathematical Specifications document discusses mean longitude and the calculation of the equinoctial elements in detail. Units: degrees.

Table 2.2: Fields Associated with Spacecraft Physical Properties (Ballistic/Mass Tab)

Field	Options and Description
DryMass	Default: 850. Options: [Real Number ≥ 0]: The DryMass field allows the user to specify the mass of the spacecraft structure, but does not include the mass of tanks, thrusters, or fuel. Units: kg.
Cd	Default: 2.2. Options: [Real Number ≥ 0]: The Cd field allows the user to specify the spacecraft's drag coefficient. Units: None.
Cr	Default: 1.8. Options: [Real Number ≥ 0]: The Cr field allows the user to specify the spacecraft's coefficient of reflectivity. Units: None.
DragArea	Default: 15. Options: [Real Number ≥ 0]: The DragArea is the effective spacecraft area used in calculate the force due to drag. Units: m^2 .

Table 2.2: (Fields Associated with Spacecraft Physical Properties
(Orbit Tab). continued)

Field	Options and Description
SRPArea	Default: 1. Options: [Real Number ≥ 0]: The SRPArea is the effective spacecraft area used in calculate the force due to solar radiation pressures. Units: m^2 .

Table 2.3: Fields Associated with Spacecraft Attitude State
(Attitude Tab)

Field	Options and Description
Attitude Mode	Default: CSFixed. Options:[CSFixed, Spinner]: The AttitudeMode mode field allows the user to specify the attitude dynamics model to be used by GMAT to propagate a spacecraft's attitude. The attitude dynamics model uses the initial attitude state and the algorithm associated with AttitudeMode to advance the attitude state in time. Units: N/A.
Attitude Coordinate System	Default: EarthMJ2000Eq. Options: [EarthMJ2000Eq, EarthMJ2000Ec, EarthMJ2000Eq, or any user defined system]: A spacecraft's initial body axes orientation as defined by the quaternions or some other parameterizations are expressed with respect to the AttitudeCoordinateSystem. Unlike an orbit state, an attitude state is really information that uniquely defines a rotation matrix. A spacecraft's attitude is the orientation of the spacecraft's body-fixed frame with respect to the inertial frame. However, it is often more convenient to define the initial attitude with respect to an intermediate frame than with respect to an inertial frame. The Attitude CoordinateSystem allows the user to define the initial orientation of a spacecraft's body axes, with respect to any frame GMAT knows how to calculate. Units: N/A.
Attitude StateType	Default: EulerAngles. Options:[EulerAngles, Quaternions, DCM]: The AttitudeStateType field allows the user to choose among different attitude parameterizations when defining the attitude initial conditions. Units: N/A.
Attitude Rate StateType	Default: EulerAngleRates. Options:[EulerAngleRates, AngularVelocity]: The AttitudeRateStateType field allows the user to define the attitude parameterization to be used in defining the initial attitude rate. Units: N/A.
Euler Angle Sequence	Default: 312. Options:[123, 132, 121, 131, 213, 231, 212, 232, 312, 321, 313, 323]: The EulerAngleSequence field allows the user to define the Euler sequence used in rotating from the body-fixed to the inertial axes. For example, if EulerAngleSequence is selected as 321, then the first rotation is a 3 rotation through EulerAngle1, the second rotation is a 2 rotation through EulerAngle2, and the third rotation is a 1 rotation through EulerAngle3. Units: N/A.

Fields associated with Spacecraft Attitude State

EulerAngle1 Default: 0. Options:[Real Number]: EulerAngle1 is one of three Euler angles that can be used to define the initial conditions of a spacecraft. EulerAngle1 corresponds to the first rotation performed in the sequence that goes from the spacecraft body frame to the inertial frame. For example, if the EulerAngleSequence field is set to 321, the first rotation from the body to the inertial frame would be a 3-rotation through EulerAngle1. Units: degrees.

Draft: Work in Progress

Table 2.3: (Fields Associated with Spacecraft Attitude State (Attitude Tab)continued)

Field	Options and Description
EulerAngle2	Default: 0. Options:[Real Number]: EulerAngle2 is one of three Euler angles that can be used to define the initial conditions of a spacecraft. EulerAngle2 corresponds to the second rotation performed in the sequence that goes from the spacecraft body frame to the inertial frame. For example, if the EulerAngleSequence field is set to 321, the second rotation from the body to the inertial frame would be a 2-rotation through EulerAngle2. Units: degrees.
EulerAngle3	Default: 0. Options:[Real Number]: EulerAngle3 is one of three Euler angles that can be used to define the initial conditions of a spacecraft. EulerAngle3 corresponds to the third rotation performed in the sequence that goes from the spacecraft body frame to the inertial frame. For example, if the EulerAngleSequence field is set to 321, the third rotation from the body to the inertial frame would be a 1-rotation through EulerAngle3. Units: degrees.
q1	Default: 0. Options:[Real Number]: The q1 parameter is the first element of the quaternion. GMAT normalizes the quaternion to be of length 1. Units: degrees.
q2	Default: 0. Options:[Real Number]: The q2 parameter is the second element of the quaternion. GMAT normalizes the quaternion to be of length 1. Units: degrees.
q3	Default: 0. Options:[Real Number]: The q3 parameter is the third element of the quaternion. GMAT normalizes the quaternion to be of length 1. Units: degrees.
q4	Default: 1. Options:[Real Number]: The q4 parameter is the fourth element of the quaternion. GMAT normalizes the quaternion to be of length 1. Units: degrees.
DCM11	Default: 1. Options:[Real Number]: The DCM11 parameter is the upper left component of the direction cosine matrix that rotates from the spacecraft body frame to the inertial frame. GMAT normalizes the attitude matrix to have a determinant of 1. The default DCM matrix is the identity matrix. Units: None.
DCM12	Default: 0. Options:[Real Number]: The DCM12 parameter is the R_{12} component of the direction cosine matrix that rotates from the spacecraft body frame to the inertial frame. GMAT normalizes the attitude matrix to have a determinant of 1. The default DCM matrix is the identity matrix. Units: None.
DCM33	Default: 1. Options:[Real Number]: The DCM33 parameter is the R_{33} component of the direction cosine matrix that rotates from the spacecraft body frame to the inertial frame. GMAT normalizes the attitude matrix to have a determinant of 1. The default DCM matrix is the identity matrix. Units: None.
EulerAngleRate1	Default: 0. Options:[Real Number]: The EulerAngleRate1 defines the time-rate-of-change of EulerAngle1, expressed in the the system defined by AttitudeCoordinateSystem. Units: deg/sec.
EulerAngleRate2	Default: 0. Options:[Real Number]: The EulerAngleRate2 defines the time-rate-of-change of EulerAngle2, expressed in the the system defined by AttitudeCoordinateSystem. Units: deg/sec.
EulerAngleRate3	Default: 0. Options:[Real Number]: The EulerAngleRate3 defines the time-rate-of-change of EulerAngle3, expressed in the the system defined by AttitudeCoordinateSystem. Units: deg/sec.

Table 2.3: (Fields Associated with Spacecraft Attitude State
(Attitude Tab)continued)

Field	Options and Description
Angular VelocityX	Default: 0. Options:[Real Number]: The AngularVelocityX component is the x-component of the spacecraft's body axes with respect to the system defined by AttitudeCoordinateSystem. Units: deg/sec.
Angular VelocityY	Default: 0. Options:[Real Number]: The AngularVelocityY component is the y-component of the spacecraft's body axes with respect to the system defined by AttitudeCoordinateSystem. Units: deg/sec.
Angular VelocityZ	Default: 0. Options:[Real Number]: The AngularVelocityZ component is the z-component of the spacecraft's body axes with respect to the system defined by AttitudeCoordinateSystem. Units: deg/sec.

Table 2.4: Fields Associated with a Spacecraft Tank(Tanks Tab)

Field	Options and Description
FuelMass	Default: 756. Options: [Real Number ≥ 0]: The FuelMass field is the mass of fuel in the tank. Units: kg.
Pressure	Default: 1500. Options: [Real Number ≥ 0]: The Pressure field is the pressure of the fuel in the tank. Units: kPa.
Temperature	Default: 20. Options: [Real Number]: The Temperature field is the temperature of the fuel in the tank. Units: C.
RefTemperature	Default: 20. Options: [Real Number]: RefTemperature Units: C.
Volume	Default: 0.75. Options: [Real Number ≥ 0]: The Volume field is the volume of the tank. Units: m ³ .
FuelDensity	Default: 1260. Options: [Real Number ≥ 0]: The FuelDensity parameter is the fuel density. Units: kg/m ³
PressureRegulated	Default: true. Options: [true false]: The PressureRegulated flag allows the user to choose between a pressure regulated tank or a blow down tank. If PressureRegulated is true, then the pressure is held constant as fuel mass is depleted. If PressureRegulated is false, then the pressure decreases as fuel is depleted.

Draft: Work in Progress

Table 2.5: Fields Associated with a Spacecraft Thruster
(Actuators Tab)

Field	Options and Description
CoordinateSystem	Default: EarthMJ2000Eq. Options: [EarthMJ2000Eq, EarthMJ2000Ec, EarthMJ2000Eq, or any user defined system]: The CoordinateSystem field for a thruster determines what coordinate system the orientation parameters X_Direction, Y_Direction, and Z_Direction are referenced to. This is a temporary fix in GMAT. Eventually, the user will specify the attitude of a spacecraft, and then X_Direction, Y_Direction, and Z_Direction will be referenced to the spacecraft body frame.
Axis	Default: VNB. Options: [Inertial VNB]: The Axis field allows the user to define a local coordinate system for a thruster. Note that there is a coupling between the Axis parameter and the CoordinateSystem parameter for a thruster. Only one of the two can be specified. Units: N/A.
Origin	Default: Earth. Options: [Sun, Mercury, Venus, Earth, Luna, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto]: The Origin field allows the user to define a local origin for a thruster. Note that there is a coupling between the Origin parameter and the CoordinateSystem parameter for a thruster. Only one of the two can be specified. Units: N/A.
X_Direction	Default: 1. Options: [Real Number]: X_Direction, divided by the RSS of the three direction components, forms the x direction of the spacecraft thrust vector direction.
Y_Direction	Default: 0. Options: [Real Number]: Y_Direction, divided by the RSS of the three direction components, forms the y direction of the spacecraft thrust vector direction.
Z_Direction	Default: 0. Options: [Real Number]: Z_Direction, divided by the RSS of the three direction components, forms the z direction of the spacecraft thrust vector direction.
ThrustScaleFactor	Default: 1. Options: [Real Number > 0]: ThrustScaleFactor is a scale factor that is multiplied by the thrust vector for a given thruster, before the thrust vector is added into the total acceleration. Units: None.
Tank	Default: None. Options: [Tank Name]: The Tank field specifies which tank the thruster draws propellant from.

The constants C_i below are used in the following equation to calculate thrust F_T as a function of pressure P and temperature T

$$F_T(P, T) = \{C_1 + C_2P + C_3P^2 + C_4P^{C_5} + C_6P^{C_7} + C_8P^{C_9} + C_{10}C_{11}^{C_{12}P}\} \left(\frac{T}{T_{ref}}\right)^{1+C_{13}+C_{14}P}$$

C1	Default: 500. Options: [Real Number]: Thrust coefficient. Units: N
C2	Default: 0. Options: [Real Number]: Thrust coefficient. Units: N/kPa.
C3	Default: 0. Options: [Real Number]: Thrust coefficient. Units: N/kPa ²

Table 2.5: Fields Associated with a Spacecraft Thruster
(Actuators Tab) (continued)

Field	Options and Description
C4	Default: 0. Options: [Real Number]: Thrust coefficient. Units: N/kPa ^{C5} .
C5	Default: 0. Options: [Real Number]: Thrust coefficient. Units: None
C6	Default: 0. Options: [Real Number]: Thrust coefficient. Units: N/kPa ^{C7} .
C7	Default: 0. Options: [Real Number]: Thrust coefficient. Units: None
C8	Default: 0. Options: [Real Number]: Thrust coefficient. Units: N/kPa ^{C9} .
C9	Default: 0. Options: [Real Number]: Thrust coefficient. Units: None
C10	Default: 0. Options: [Real Number]: Thrust coefficient. Units: N.
C11	Default: 1. Options: [Real Number]: Thrust coefficient. Units: None
C12	Default: 0. Options: [Real Number]: Thrust coefficient. Units: 1/kPa.
C13	Default: 0. Options: [Real Number]: Thrust coefficient. Units: None.
C14	Default: 0. Options: [Real Number]: Thrust coefficient. Units 1/kPa.
<p>The constants K_i below are used in the following equation to calculate Isp as a function of pressure P and temperature T</p> $I_{sp}(P, T) = \left\{ K_1 + K_2 P + K_3 P^2 + K_4 P^{K_5} + K_6 P^{K_7} + K_8 P^{K_9} + K_{10} K_{11}^{K_{12} P} \right\} \left(\frac{T}{T_{ref}} \right)^{1+K_{13}+K_{14}P}$	
K1	Default: 2150. Options: [Real Number]: Isp coefficient. Units: m/sec
K2	Default: 0. Options: [Real Number]: Isp coefficient. Units: m/(sec·kPa).
K3	Default: 0. Options: [Real Number]: Isp coefficient. Units: m/(sec·kPa ²)
K4	Default: 0. Options: [Real Number]: Isp coefficient. Units: m/(sec·kPa ^{K5}).
K5	Default: 0. Options: [Real Number]: Isp coefficient. Units: None
K6	Default: 0. Options: [Real Number]: Isp coefficient. Units: m/(sec·kPa ^{K7}).
K7	Default: 0. Options: [Real Number]: Isp coefficient. Units: None
K8	Default: 0. Options: [Real Number]: Isp coefficient. Units: m/(sec·kPa ^{K9}).
K9	Default: 0. Options: [Real Number]: Isp coefficient. Units: None

Table 2.5: Fields Associated with a Spacecraft Thruster (Actuators Tab) (continued)

Field	Options and Description
K10	Default: 0. Options: [Real Number]: Isp coefficient. Units: m/sec.
K11	Default: 1. Options: [Real Number]: Isp coefficient. Units: None
K12	Default: 0. Options: [Real Number]: Isp coefficient. Units: 1/kPa.
K13	Default: 0. Options: [Real Number]: Isp coefficient. Units: None.
K14	Default: 0. Options: [Real Number]: Isp coefficient. Units 1/kPa.

2.2 Propagator Fields

Table 2.6: Fields Associated with a Force Model

Field	Options and Description
CentralBody	Default: Earth. Options: [Sun, Mercury, Venus, Earth, Luna, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto]: The CentralBody field allows the user to select the origin for the propagation. All propagation occurs in the FK5 axes system, about the CentralBody chosen by the user. The CentralBody must be a gravitational body and so cannot be a LibrationPoint or other special point. Units: N/A.
PrimaryBodies	Default: {Earth}. Options: [Sun, Mercury, Venus, Earth, Luna, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto]: The PrimaryBodies field is a list of all celestial bodies that are to be modelled with a force model more complex than point mass gravity. Lists are surrounded by curly braces. For each PrimaryBody, the user can choose a drag, magnetic field, and aspherical gravity model. There is a coupling between the PrimaryBodies field and the PointMasses field. A primary body can be any planet or moon not included in the PointMasses field. Units: N/A.

Table 2.6: (Fields Associated with a Force Model...continued)

Field	Options and Description
<code>Gravity.PrimaryBody.PotentialFile</code>	Default: JGM2. Options: [<code>CentralBody</code> -based models, <code>Other</code> . See Comments]. This field allows the user to define the source for the non-spherical gravity coefficients for a primary body. If a gravity file is located in the Primary Body's potential path as defined in the startup file, you only need to specify the model name and not the entire path. For example, if the JGM2 coefficients file is contained in the directory defined in the startup file by the line <code>EARTH.POT.PATH</code> , then you only need to specify the model name JGM2. If the model is not contained in the body's potential path, you must supply the entire path as well as the file name. If GMAT does not successfully find the file requested, it uses the default gravity model as defined in the startup file. From the GUI, only models for Earth appear if Earth is the active primary body. This is to avoid allowing the user to select a lunar potential model for the Earth. If the <code>Other</code> option is selected the user has the ability of selecting a gravity model file on their local computer. Units: None.
<code>Gravity.PrimaryBody.Degree</code>	Default: 4. Options: [Integer ≥ 0 and $<$ the maximum specified by the model, <code>Order</code> \leq <code>Degree</code>]. This field allows the user to select the the degree, or number of zonal terms, in the non-spherical gravity model. Ex. <code>Gravity.Earth.Degree = 2</code> tells GMAT to use only the J2 zonal term for the Earth. The value for <code>Degree</code> must be less than the maximum degree specified by the Model. Units: None.
<code>Gravity.PrimaryBody.Order</code>	Default: 4. Options: [Integer ≥ 0 and $<$ the maximum specified by the model, <code>Order</code> \leq <code>Degree</code>]. This field allows the user to select the the order, or number of tesseral terms, in the non-spherical gravity model. Ex. <code>Gravity.Earth.Order = 2</code> tells GMAT to use 2 tesseral terms. Note: <code>Order</code> must be greater than or equal to <code>Degree</code> . Units: None.
<code>Drag</code>	Default: None. Options: [<code>None</code> , <code>JachhiaRoberts</code> , <code>MSISE90</code> , <code>Exponential</code>]. The <code>Drag</code> field allows a user to specify a drag model. Currently, only one drag model can be chosen for a particular propagator and only Earth models are available. Units: N/A. <i>Note: This field will be deprecated in future versions of GMAT. Currently, the <code>Drag</code> field and the <code>Drag.AtmosphereModel</code> field must be set to the same value.</i>
<code>Drag.AtmosphereModel</code>	Default: None. Options: [<code>JachhiaRoberts</code> , <code>MSISE90</code> , <code>Exponential</code>]. The <code>Drag.AtmosphereModel</code> field allows a user to specify a drag model. Currently, only one drag model can be chosen for a particular propagator and only Earth models are available. Units: N/A.
<code>Drag.F107</code>	Default: 150. Options: [Real Number ≥ 0]. The <code>F107</code> field allows you to set the $F_{10.7}$ solar flux value used in computing atmospheric density. $F_{10.7}$ is the solar radiation at a wavelength of 10.7 cm. Units: $W/m^2/Hz$
<code>Drag.F107A</code>	Default: 150. Options: [Real Number ≥ 0]. The <code>F107A</code> field allows you to set the average $F_{10.7}$ value. $\bar{F}_{10.7}$ is the average of $F_{10.7}$ over one month. Units: $W/m^2/Hz$

Table 2.6: (Fields Associated with a Force Model...continued)

Field	Options and Description
Drag.MagneticIndex	Default:3. Options: [$0 \leq \text{Real Number} \leq 9$]: The MagneticIndex index field allows you to set the k_p value for use in atmospheric density calculations. k_p is a planetary 3-hour-average, geomagnetic index that measures magnetic effects of solar radiation. Units: None.
SRP	Default: Off. Options: [On, Off]. The SRP field allows the user to include the force due to solar radiation pressure in the total sum of forces. Units: N/A.
PointMasses	Default: None. Options [Sun, Mercury, Venus, Earth, Luna, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto]. A PointMass is a planet or moon that is modelled by a point source located at its center of gravity. A PointMass body can be any planet or moon not included in the PrimaryBodies field. Units: N/A.
ErrorControl	Default: RSSStep. Options: [RSSStep, RSSState, LargestState, LargestStep]: The ErrorControl field allows you to choose how a Propagator measures the error in an integration step. The algorithm selected in the ErrorControl field is used to determine the error in the current step, and this error is compared to the value set in the Accuracy field to determine if the step has an acceptable error or needs to be improved. All error measurements are relative error, however, the reference for the relative error changes depending upon the selection of ErrorControl. RSSState is the Root Sum Square (RSS) relative error measured with respect to the current step. RSSState is the (RSS) relative error measured with respect to the current state. LargestStep is the state vector component with the largest relative error measured with respect to the current step. LargestState is the state vector component with the largest relative error measured with respect to the current state. For a more detailed discussion see the GMAT Mathematical Specification. Units: N/A.

Table 2.7: Fields Associated with an Integrator

Field	Options and Description
Fields associated with All Integrators	
Type	Default: RungeKutta89. Options: [RungeKutta89, RungeKutta68, RungeKutta56, PrinceDormand45, PrinceDormand78, BulirschStoer, AdamsBashforthMoulton]: The Type field is used to set the type of numerical integrator. Units: N/A.

Table 2.7: Fields Associated with an Integrator....(continued)

Field	Options and Description
InitialStepSize	Default: 60 (sec). Options: [Real Number]. The InitialStepSize is the size of the first attempted step by the integrator. If the step defined by InitialStepSize does not satisfy Accuracy, the integrator adapts the step according an algorithm defined in the mathematical specifications document to find an acceptable first step that meets the user's requested Accuracy. Units: sec.
Accuracy	Default: 1e-11. Options: [Real Number ≥ 0]. The Accuracy field is used to set the desired accuracy for an integration step. Units: N/A. When you set a value for Accuracy, GMAT uses the method selected in ErrorControl field on the Force Model, to determine a metric of the accuracy. For each step, the integrator ensures that the accuracy, as calculate using the method define by ErrorControl, is less than the limit defined by Accuracy. If an integrator exceeds MaxStepAttempts trying to meet the requested accuracy, and error message is thrown and propagation stops.
MinStep	Default: .001 (sec). Options: [Real Number > 0 , MinStep \leq MaxStep]. The MinStep field is used to set the minimum allowable step size. Units: sec.
MaxStep	Default: 2700.0 (sec.). Options: [Real Number > 0 , MinStep \leq MaxStep]. The MaxStep field is used to set the maximum allowable step size. Units: sec.
MaxStepAttempts	Default: 50. Options: [Integer > 0]. The MaxStepAttempts field allows the user to set the number of attempts the integrator takes to meet the tolerance defined by Accuracy. Units: None.
Fields associated only with Adams-Bashforth-Moulton Integrator	
MinIntegrationError	Default: 1.0e-13. Options: [Real Number > 0 , MinIntegrationError $<$ NomIntegrationError $<$ Accuracy]: The MinIntegrationError field is used by the ABM integrator (and other predictor-corrector integrators when implemented) as the desired integration error to be obtained when the step size is changed. Predictor-Corrector integrators adapt step size when the obtained integration error falls outside of the range of acceptable steps, as determined by the bounds set by the MinIntegrationError and Accuracy fields. The integrator then applies an internal calculation to recompute the step size, attempting to hit the NomIntegrationError, and restarts the integrator. Units: N/A.
NomIntegrationError	Default: 1.0e-11. Options: [Real Number > 0 , MinIntegrationError $<$ NomIntegrationError $<$ Accuracy]: The NomIntegrationError field is used by the ABM integrator (and other predictor-corrector integrators when implemented) as the desired integration error to be obtained when the step size is changed. Predictor-Corrector integrators adapt step size when the obtained integration error falls outside of the range of acceptable steps, as determined by the bounds set by the MinIntegrationError and Accuracy fields. The integrator then applies an internal calculation to recompute the step size, attempting to hit the NomIntegrationError, and restarts the integrator. Units: N/A.

Table 2.7: Fields Associated with an Integrator...(continued)

Field	Options and Description
Script Examples	
	<pre> Create ForceModel MyProp_ForceModel; GMAT MyProp_ForceModel.CentralBody = Earth; GMAT MyProp_ForceModel.PrimaryBodies = {Earth}; GMAT MyProp_ForceModel.PointMasses = {Sun, Luna}; GMAT MyProp_ForceModel.Drag = None; GMAT MyProp_ForceModel.SRP = Off; GMAT MyProp_ForceModel.ErrorControl = RSSStep; GMAT MyProp_ForceModel.Gravity.Earth.Degree = 4; GMAT MyProp_ForceModel.Gravity.Earth.Order = 4; GMAT MyProp_ForceModel.Gravity.Earth.PotentialFile =/JGM2v; Create Propagator MyProp; GMAT MyProp.FM = MyProp_ForceModel; GMAT MyProp.Type = RungeKutta89; GMAT MyProp.InitialStepSize = 60; GMAT MyProp.Accuracy = 9.999999999999999e-012; GMAT MyProp.MinStep = 0.001; GMAT MyProp.MaxStep = 2700; GMAT MyProp.MaxStepAttempts = 50; </pre>

2.3 Maneuvers

Table 2.8: Fields Associated with an Impulsive Burn

Field	Options and Description
Origin	Default: Earth . Options: [Any celestial body]: Together the Origin and Axes fields describe the coordinate system in which a maneuver is applied. The Origin field determines the origin of the maneuver coordinate system. The ability to define the coordinate system locally avoids having to create many coordinate systems, associated with specific spacecraft, in order to perform finite maneuvers for multiple spacecraft. Units: N/A.
Axes	Default: VNB . Options: [VNB,MJ2000Eq]: The Axes field, together with the Origin field, describe the coordinate system in which an impulsive maneuver is applied. If VNB is chosen for Axes, a local coordinate system is created such that the x-axis points in the velocity direction of the spacecraft, with respect to the point defined by Origin, the y-axis points in the normal direction of the spacecraft with respect to Origin, and the z-axis completes the right-handed set. Units: N/A.
VectorFormat	Default: Cartesian . Options: [Cartesian, Spherical]: The VectorFormat field allows the user to define the format of the maneuver vector. Units: N/A.

Table 2.8: (continued)

Field	Options and Description
Element1	Default: 0. Options: [Real Number]: The Element1 field allows the user to define the first element of the impulsive maneuver vector. Element1 is x if VectorFormat is Cartesian. Element1 is the magnitude of the burn if VectorFormat is spherical. Units: km/sec.
Element2	Default: 0. Options: [Real Number]: The Element2 field allows the user to define the second element of the impulsive maneuver vector. Element2 is y if VectorFormat is Cartesian. Units: km/sec.
Element3	Default: 0. Options: [Real Number]: The Element3 field allows the user to define the second element of the impulsive maneuver vector. Element3 is z if VectorFormat is Cartesian. Units: km/sec.

Table 2.9: Fields Associated with a Finite Burn

Field	Options and Description
Origin	Default: Earth . Options: [Any celestial body, libration point, or barycenter]: Together the Origin and Axes fields describe the coordinate system in which a maneuver is applied. The Origin field determines the origin of the maneuver coordinate system. The ability to define the coordinate system locally avoids having to create many coordinate systems, associated with specific spacecraft, in order to perform finite maneuvers for multiple spacecraft. Units: N/A.
Axes	Default: VNB . Options: [VNB, MJ2000Eq]: The Axes field, together with the Origin field, describe the coordinate system in which a finite maneuver is applied. If VNB is chosen for Axes, a local coordinate system is created such that the x-axis points in the velocity direction of the spacecraft, with respect to the point defined by Origin, the y-axis points in the normal direction of the spacecraft with respect to Origin, and the z-axis completes the right-handed set. Units: N/A.
Thrusters	Default: No Default. Options: [Any thruster created by user]: The Thrusters field allows the selection of which thrusters to use when applying a finite maneuver. The user can select more than one thruster, from the list of thrusters previously created, by including all thrusters in curly braces. An example is MyFiniteBurn.Thrusters = {Thruster1,Thruster2,Thruster3}. Units: N/A.
BurnScaleFactor	Default: 1.0 . Options: [Real Number]: The BurnScaleFactor is used to scale the total acceleration before adding the acceleration due to a finite burn into the sum of the accelerations of a spacecraft. The scaling is performed by taking the sum of the accelerations applied by all thrusters specified under the Thrusters field, and multiplying the total thrust by BurnScaleFactor. Units: None.

2.4 Solver Fields

Table 2.10: Fields Associated with the `fmincon` Solver

Field	Options and Description
DiffMax Change	Default: 0.1 . Options: [Real Number > 0]: The <code>DiffMaxChange</code> parameter sets the upper limit on the perturbation used in MATLAB's finite differencing algorithm. For <code>fmincon</code> , you don't specify a single perturbation value, but rather give MATLAB a range, and it uses an adaptive algorithm that attempts to find the optimal perturbation. Units: N/A .
DiffMin Change	Default: 1e-8 . Options: [Real Number > 0]: The <code>DiffMinChange</code> parameter sets the lower limit on the perturbation used in MATLAB's finite differencing algorithm. For <code>fmincon</code> , you don't specify a single perturbation value, but rather give MATLAB a range, and it uses an adaptive algorithm that attempts to find the optimal perturbation. Units: N/A .
MaxFunEvals	Default: 1000. Options: [Integer > 0]: The <code>MaxFunEvals</code> parameter allows the user to set the maximum number of cost function evaluations in an attempt to find an optimal solution. This is equivalent to setting the maximum number of passes through an optimization loop in a GMAT script. If a solution is not found before the maximum function evaluations, <code>fmincon</code> outputs an <code>ExitFlag</code> of zero, and GMAT continues. Units: N/A.
MaxIter	Default: 400. Options: [Integer > 0]: The <code>MaxIter</code> parameter allows the user to set the maximum allowable number of optimizer iterations. Depending upon the nature of the problem, and whether gradients are provided, it may take many function evaluations for each optimizer iteration. The <code>MaxIter</code> parameter allows the user to control the maximum function evaluations, and maximum iterations independently. Units: N/A .
TolX	Default: 1e-4. Options: [Real Number > 0]: The <code>TolX</code> parameter is the termination tolerance on the vector of independent variables, and is used only if the user sets a value. Units: N/A.
TolFun	Default: 1e-4. Options: [Real Number > 0]: The <code>TolFun</code> parameter is the convergence tolerance on the cost function value. Units: N/A .
TolCon	Default: 1e-4 . Options: [Real Number > 0]: The <code>TolCon</code> parameter is the convergence tolerance on the constraint functions. Units: N/A .
Derivative Check	Default: off. Options: [on, off]: If the <code>DerivativeCheck</code> option is set to on, then <code>fmincon</code> will calculate the gradients of the cost and constraints using finite differencing, and compare the values to the values calculated analytically. Units: N/A .
Diagnostics	Default: off. Options: [on, off]: The <code>Diagnostics</code> parameter tells <code>fmincon</code> to output general information on the problem by writing diagnostic information to the MATLAB prompt. The diagnostic information contains the number of independent variables, the number and types of constraints, the sources for derivatives and other information. Units: N/A .
Display	Default: iter. Options: [off, on, iter, notify, final]: The <code>Display</code> parameter allows the user to select between several different options that displays information at the MATLAB prompt that indicates the progress of the optimization process. Units: N/A .
GradObj	Default: off. Options: [on, off]: The <code>GradObj</code> parameter allows the user to tell <code>fmincon</code> to use finite differencing to calculate the cost function derivative, or to use the cost function derivative provided by the user. Units: N/A .

Table 2.10: (Fields Associated with the fmincon Solver....continued)

Field	Options and Description
GradConstr	Default: off. Options: [on, off]: The GradConstr parameter allows the user to tell fmincon to use finite differencing to calculate the constraint function derivatives, or to use the constraint function derivatives provided by the user. Units: N/A.

Table 2.11: Fields Associated with a Differential Corrector

Field	Options and Description
MaximumIterations	Default: 25. Options: [Integer > 0]: The Maximum Iterations field allows the user to define the maximum number of iterations taken in attempt to find a solution. Units: N/A.
ShowProgress	Default: true. Options: [true, false]: When the ShowProgress field is set to true, then data illustrating the progress of the differential correction process are written to the status bar. The status bar is updated with information on perturbation and iteration passes. . Units: N/A.
ReportStyle	Default: Normal . Options: [Normal, Concise, Verbose, Debug]: The ReportStyle field allows the user to control the amount and type of information written to TargeterTextFile. Units: N/A.
TargeterTextFile	Default: DifferentialCorrectorDCName. Options: [Filename consistent with OS]: The TargeterTextFile field allows the user to specify the path and file name for the targeter report. Units: N/A.
UseCentralDifferences	Default: false Options: [true, false]: The UseCentralDifferences field allows the user to choose between one-sided and central differencing for determining the Jacobian matrix. If UseCentralDifferences is set to false, then one-sided differencing is used, if UseCentralDifferences is set to true, then central differencing is used. Units: N/A.

2.5 Plots and Reports

Table 2.12: Fields Associated with OpenGL Plots

Field	Options and Description
Fields associated with Plot Options	
ShowPlot	Default: true. Options: [true, false]: The ShowPlot field allows the user to turn off a plot for a particular run, without deleting the plot object, or removing it from the script. If you select true, then the plot will be shown. If you select false, then the plot will not be shown. Units: N/A.

Table 2.12: (Fields Associated with OpenGL Plots....continued)

Field	Options and Description
DataCollectFrequency	Default: 1. Options: [Integer \geq 1]: The DataCollectFrequency field allows the user to define how data is collected for plotting. It is often inefficient to draw every ephemeris point associated with a trajectory. Often, drawing a smaller subset of the data still results in smooth trajectory plots, while executing more quickly. The DataCollectFrequency is an integer that represents how often to collect data and store for plotting. If DataCollectFrequency is 10, then Data is collected every ten integration steps. Units: Integration Steps
UpdatePlotFrequency	Default: 50. Options: [Integer \geq 1]: The UpdatePlotFrequency field allows the user to specify how often to update an OpenGL plot is updated with new data collected during the process of propagating spacecraft and running a mission. Data is collected for a plot according the value defined by DataCollectFrequency. An OpenGL plot is updated with the new data, according to the value set in UpdatePlotFrequency. If UpdatePlotFrequency is set to 10, then the plot is updated with new data every ten integration steps. Units: Integration Steps.
NumPointsToRedraw	Default: 0. Options: [Integer \geq 0]: When NumPointsToRedraw is set to zero, all ephemeris points are drawn. When NumPointsToRedraw is set to a positive integer, say 10 for example, only the last 10 collected data points are drawn. See DataCollectFrequency for explanation of how data is collected for an OpenGL plot. Units: Integration Steps.
Fields associated with Drawing Options	
WireFrame	Default: Off . Options: [On, Off]: When the WireFrame field is set to On, celestial bodies are drawn using a wireframe model. When the WireFrame field is set to Off, then celestial bodies are drawn using a full map. Units: N/A.
SolverIterations	Default: Off. Options: [On, Off]: The SolverIterations field determines whether or not perturbed trajectories are plotted during a solver (Targeter, Optimize) sequence. When SolverIterations is set to On, solver iterations are shown on the plot. When SolverIterations is Off, the solver iterations are not shown on the plot. Units: N/A.
EclipticPlane	Default: Off. Options: [On,Off, Note: Only allowed for OpenGL plots with Coordinate Systems that use the MJ2000Eq axis system]: The EclipticPlane field allows the user to tell GMAT to draw a grid representing the ecliptic plane in an OpenGL plot. Note, the ecliptic plane can currently only be drawn for plots whose coordinate system uses the MJ2000Eq axis system. Units: N/A .
XYPlane	Default: On. Options: [On,Off]: The XYPlane flag allows the user to tell GMAT to draw a grid representing the XY-plane of the coordinate system selected under the CoordinateSystem field of the OpenGL plot. Units: N/A .
Axes	Default: On. Options: [On,Off]: The Axis flag allows the user to tell GMAT to draw the Cartesian axis system associated with the coordinate system selected under the CoordinateSystem field of an OpenGL plot. Units: N/A .
Grid	Default: On. Options: [On,Off]: The Grid flag allows the user to tell GMAT to draw a grid representing the longitude and latitude lines celestial bodies added to an OpenGL plot. Units: N/A .

Table 2.12: (Fields Associated with OpenGL Plots....continued)

Field	Options and Description
EarthSunLines	Default: On. Options: [On,Off]: The EarthSunLines allows the user to tell GMAT to draw a line that starts at the center of Earth and points towards the Sun. Units: N/A .

Fields Associated with View Definition

CoordinateSystem	Default: EarthMJ2000Eq. Options: [Any default or user defined coordinate system]: The CoordinateSystem field on an OpenGL plot allows the user to select which coordinate system to use to draw the plot data. A coordinate system is defined as an origin and an axis system, and the CoordinateSystem field allows the user to determine the origin and axis system of an OpenGL plot. See the CoordinateSystem object fields for information of defining different types of coordinate systems. Units: N/A.
Add	Default: DefaultSC, Earth. Options: [SpacecraftName CelestialBodyName LibrationPointName BarycenterName]: The Add subfield adds a spacecraft,celestial body, libration point,or barycenter to a plot. When creating a plot the Earth is added as a default body and may be removed by using the Remove command. The user can add a spacecraft, celestial body, libration point, or barycenter to a plot by using the name used to create the object. The GUI's Selected field is the equivalent of the script's Add field. In the event of no Add command or no objects in the Selected field, GMAT should run without the OpenGL plot and a warning message displayed in the message window. The following warning message is sufficient: OpenGL plot will be turned off. No object has been selected for plotting. Units: N/A.
Remove	Default: No Default. Options: [Any object included in the Add list]: The Remove subfield removes a spacecraft,celestial body, libration point, or barycenter from a plot. The user can remove any object that has been added to a plot by using the name used to add the object. Units: N/A.
ViewPointReference	Default: Earth. Options: [SpacecraftName CelestialBodyName LibrationPointName BarycenterName, or a 3-vector of numerical values]: The ViewPointReference field is an optional field that allows the user to change the reference point from which ViewPointVector is measured. ViewPointReference defaults to the origin of the coordinate system for the plot. A ViewPointReference can be any spacecraft, celestial body, libration point, or barycenter. Units: N/A.
ViewPointVector	Default: [0 0 30000]. Options: [SpacecraftName CelestialBodyName LibrationPointName BarycenterName, or a 3-vector of numerical values]: The product of ViewScaleFactor and ViewPointVector field determines the view point location with respect to ViewPointReference. ViewPointVector can be a vector, or any of the following objects: spacecraft,celestial body, libration point,or barycenter. The location of the Viewpoint in three-space is defined as the vector addition of ViewPointReference, and the vector defined by product of ViewScaleFactor and ViewPointVector in the coordinate system chosen by the user. Units: km or N/A.

Table 2.12: (Fields Associated with OpenGL Plots....continued)

Field	Options and Description
ViewDirection	Default: Earth. Options: [SpacecraftName CelestialBodyName LibrationPointName BarycenterName, or a 3-vector of numerical values]: The ViewDirection field allows the user to select the direction of view in an OpenGL plot. The user can specify the view direction by choosing an object to point at such as a spacecraft,celestial body, libration point,or barycenter. Alternatively, the user can specify a vector of the form [x y z]. If the user specification of ViewDirection, ViewPointReference, and ViewPointVector, results in a zero vector, GMAT uses [0 0 10000] for ViewDirection. Units: km or N/A.
ViewScaleFactor	Default: 1. Options [Real Number ≥ 0]: The ViewScaleFactor field scales ViewPointVector before adding it to ViewPointReference. The ViewScaleFactor allows the user to back away from an object to fit in the field of view. Units: None.

Fields Associated with View Up Definition

ViewUpCoordinate System	Default: EarthMJ2000Eq. Options: [Any default or user defined coordinate system]: The ViewUpCoordinateSystem and ViewUpAxis fields are used to determine which direction appears as up in an OpenGL plot and together with the fields associated the the View Direction, uniquely define the view. The fields associated with the View Definition allow the user to define the point of view in 3-space, and the direction of the line of sight. However, this information alone is not enough to uniquely define the view. We also must provide how the view is oriented about the line of sight. This is accomplished by defining what direction should appear as the up direction in the plot and is configured using the ViewUpCoordinateSystem field and the ViewUpAxis field. The ViewUpCoordinateSystem allows the user to select a coordinate system to define the up direction. Most of the time this system will be the same as the coordinate system chosen under the CoordinateSystem field. Units: N/A.
ViewUpAxis	Default: Z. Options: [X, -X, Y, -Y, Z, -Z]: The ViewUpAxis allows the user to define which axis of the ViewUpCoordinateSystem that will appear as the up direction in an OpenGL plot. See the comments under ViewUpCoordinateSystem for more details of fields used to determine the up direction in an OpenGL plot. Units: N/A.

Fields Associated with Field of View

UseInitialView	Default: On. Options: [On, Off]: The UseInitialView field allows the user to control the view of an OpenGL plot between multiple runs of a mission sequence. The first time a specific OpenGL plot is created, GMAT will automatically use the view as defined by the fields associated with View Definition, View Up Direction, and Field of View. However, if the user changes the view using the mouse, GMAT will retain this view upon rerunning the mission if UseInitialView is set to false. If UseInitialView is set to true, the view for an OpenGL plot will be returned to the view defined by the initial settings. Units: N/A.
----------------	---

Table 2.12: (Fields Associated with OpenGL Plots....continued)

Field	Options and Description
PerspectiveMode	Default: Off. Options: [On, Off]: The PerspectiveMode field allows to user to toggle between the Orthogonal or Perspective projections. When PerspectiveMode is set to true, the Perspective projection is used. When PerspectiveMode is set to false, the Orthogonal projection is used. Units: N/A.
UseFixedFov	Default: Off. Options: [On, Off]: Units: N/A.
FixedFovAngle	Default: 45. Options: [Real Number ≥ 1]: Units: Degrees.

Table 2.13: Fields Associated with Report Files

Field	Options and Description
FileName	Default: /RunReports/ReportFile1.txt. Options: [Valid File Path and Name]: The FileName field allows the user to define the file path and file name for a report. Units: None.
Precision	Default: 16. Options: [Integer > 0]: The Precision field allows the user to set the precision of the variable written to a report. Units: Same as variable being reported.
Add	Default: N/A. Options: [Any user-defined parameter. Ex. Variables, Arrays, S/C parameters]: The Add field allows a user to add user-defined variables to a report file. To add multiple user-defined variables, enclose the variables with curly brackets. Ex. MyReportName.Add = {Sat.X, Sat.Y, Var1, Array(1,1)}; The GUI's Selected field is the equivalent of the script's Add field. In the event of no Add command or no objects in the Selected field, GMAT should run without the Report output and a warning message displayed in the message window. The following warning message is sufficient: Report plot will be turned off. No object has been selected for reporting. Units: N/A.
WriteReport	Default: On . Options: [On, Off]: The WriteReport field specifies whether to write data to the report FileName. Units: N/A.
WriteHeaders	Default: On . Options: [On, Off]: The WriteHeaders field specifies whether to include headers that describe the variables in a report. Units: N/A.
LeftJustify	Default: On. Options: [On, Off]: When the LeftJustify field is set to On, then the data is left justified and appears at the left most side of the column. If the LeftJustify field is set to Off, then the data is centered in the column. Units: N/A.
ZeroFill	Default: On. Options: [On, Off]: Units: N/A .
ColumnWidth	Default: 20. Options: [Integer > 0]: The ColumnWidth field is used to define the width of the data columns in a report file. The value for ColumnWidth is applied to all columns of data. For example, if ColumnWidth is set to 20, then each data column will be 20 white-spaces wide. Units: Characters.

Table 2.13: Fields Associated with Report Files....(continued)

Field	Options and Description
SolverIterations	Default: Off. Options: [On, Off]: The SolverIterations field determines whether or not data associated with perturbed trajectories during a solver (Targeter, Optimize) sequence is written to a report file. When SolverIterations is set to On, solver iterations are written to the report file. When SolverIterations is Off, the solver iterations are not written to the report file. Units: N/A.

Table 2.14: Fields Associated with XY-Plots

Field	Options and Description
IndVar	Default: DefaultSC.A1ModJulian. Options: [Any user variable, array element, or spacecraft parameter]: The IndVar field allows the user to define the independent variable for an xy-plot. Only one variable can be defined as an independent variable. For example, the line MyXYPlot.IndVar = DefaultSC.A1ModJulian sets the independent variable to be the epoch of DefaultSC in the A1 time system and modified Julian format. Units: N/A.
Add	Default: DefaultSC.EarthMJ2000Eq.X. Options: [Any user variable, array element, or spacecraft parameter]:: The Add field allows the user to add dependent variables to an xy-plot. All dependent variables are plotted on the y-axis vs the independent variable defined by IndVar. To define multiple dependent variables, they should be included in curly braces. For example, MyXYPlot.Add = {DefaultSC.EarthMJ2000Eq.Y , DefaultSC.EarthMJ2000Eq.Z}; . The GUI's Selected field is the equivalent of the script's Add field. In the event of no Add command or no objects in the Selected field, GMAT should run without the XYPlot and a warning message displayed in the message window. The following warning message is sufficient: XYPlot will be turned off. No object has been selected for plotting. Units: N/A.
Grid	Default: On . Options: [On, Off]: When the Grid field is set to On, then a grid is drawn on an xy-plot. When the Grid field is set to Off, then a grid is not drawn. Units: N/A.
SolverIterations	Default: Off. Options: [On, Off]: The SolverIterations field determines whether or not perturbed trajectories are plotted during a solver (Targeter, Optimize) sequence. When SolverIterations is set to On, solver iterations are shown on the plot. When SolverIterations is set to Off, solver iterations are not shown on the plot. Units: N/A.
ShowPlot	Default: true. Options: [true, false]: The ShowPlot field allows the user to turn off a plot for a particular run, without deleting the plot object, or removing it from the script. If you select true, then the plot will be shown. If you select false, then the plot will not be shown. Units: N/A.

2.6 Solar System, Celestial Bodies and other Space Points

Table 2.15: Fields Associated with the Solar System

Field	Options and Description
EphemerisSource	Default: DE405. Options: [DE405, DE200, SLP, Analytic]: The EphemerisSource field allows the user to select the source used for planetary ephemerides. The source is used globally whenever planetary ephemeris information is required. Units: None.
EphemerisUpdateInterval	Default: 0. Options: [Real Number ≥ 0]. The EphemerisUpdateInterval is used to set how often planetary positions are updated when calculating accelerations during propagation. For low-Earth orbits, EphemerisUpdateInterval can be set to around 60 for faster numerical integration with little effect on the accuracy of the propagation. For deep space propagation, EphemerisUpdateInterval should be set to zero. Units: sec.
UseTTForEphemeris	Default: false. Options: [true, false]: GMAT uses time in the TDB system as the default time system in the JPL ephemeris files. However, often it is possible to use time in the TT time system, without significant difference in propagation accuracy. (TT and TDB are within 1 millisecond of each other). The advantage to using TT is that it avoids the transformation from TT to TDB and therefore orbit propagation will execute faster. The UseTTForEphemeris field allows the user to choose between the default of TDB in the ephemeris files (UseTTForEphemeris=false), or TT in the ephemeris files (UseTTForEphemeris = true). Units: N/A.
EphemerisFile	Default: Same as startup file. Options:[File path and file name consistent with operating system]: The EphemerisFile field allows the user to specify the location and name of the file for each type of ephemeris GMAT supports. For example, if Ephemeris is set to DE405, you can set the path for a DE405 file using SolarSystem.EphemerisFile = c:/MyPath/MyDE405.file. Units: N/A.
AnalyticModel	Default: LowFidelity. Options: [LowFidelity]: Units: N/A.

Table 2.16: Fields Associated with a Libration Point

Field	Options and Description
Primary	Default: Sun. Options: [Sun, Mercury, Venus, Earth, Luna, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto , or any Barycenter. (The Primary and Secondary bodies cannot be the same)]: The Primary field tells GMAT which body to consider the primary body in the calculation of the location of a libration point. Units: N/A.
Secondary	Default: Earth. Options: [Sun, Mercury, Venus, Earth, Luna, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto , or any Barycenter. (The Primary and Secondary bodies cannot be the same)]: The Secondary field tells GMAT which body to consider the secondary body in the calculation of the location of a libration point. Units: N/A.
Point	Default: L1. Options: [L1, L2, L3, L4, L5]: The Point field specifies which libration point the object corresponds to. Units: N/A.

Draft: Work in Progress

Table 2.17: Fields Associated with a BaryCenter

Field	Options and Description
BodyNames	Default: {Earth, Luna}. Options: [Sun, Mercury, Venus, Earth, Luna, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto . (At least one body must be selected!)]: The BodyNames field is list that contains the bodies used to define a barycenter. In a script, the list must be surrounded by curly braces i.e. <code>BaryCenterName.BodyNames = {Earth, Luna} ;</code> Units: N/A.

Table 2.18: Fields Associated with Celestial Bodies

Field	Options and Description
Fields Associated with All Celestial Bodies. (Using Default Values for Earth as an Example)	
Mu	Default: 398600.4414. Options: [Real Number > 0]: The Mu field allows the user to define the gravitational paramter of a celestial body. Units: km ³ /sec ² .
Equatorial Radius	Default: 6378.1363. Options: [Real Number > 0]: The EquatorialRadius field allows the user to define the equatorial radius of a celestial body. Units: km.
Flattening	Default: 0.00335270. Options: [Real Number]: The Flattening field allows the user to define the mass of a celestial body. Units: None.
InitialEpoch	Default: 21544.500371. Options: [Real Number]: The InitialEpoch field allows the user to define the initial epoch, in A1 Modified Julian Date, for a celestial body. The initial epoch is only used when the user selects Analytic for the Ephemeris field on the solar system. In this case, GMAT solves Kepler's problem to determine the position and velocity of a celestial body, using the initial epoch and state information described below. Units: A1ModJulian.
SMA	Default: 149653978.978377. Options: [Real Number ≠ 0]: The SMA field allows the user to define the semimajor axis of a celestial body's orbit about its central body. (Only used when the user selects Analytic for the Ephemeris field on the Solar System.) Units: km.
ECC	Default: 0.017046. Options: [Real Number ≥ 0]: The ECC field allows the user to define the eccentricity of a celestial body's orbit about its central body. (Only used when the user selects Analytic for the Ephemeris field on the Solar System.) Units: None.
INC	Default: 23.439034. Options: [Real Number]: The INC field allows the user to define the inclination of a celestial body's orbit about its central body, in the FK5 coordinate system. (Only used when the user selects Analytic for the Ephemeris field on the Solar System.) Units: deg.
RAAN	Default: 0.000186. Options: [Real Number]: The RAAN field allows the user to define the right ascension of the ascending node of a celestial body's orbit about its central body, in the FK5 coordinate system. (Only used when the user selects Analytic for the Ephemeris field on the Solar System.) Units: deg.

Table 2.18: (Fields Associated with Celestial Bodies...continued)

Field	Options and Description
AOP	Default: 101.741639. Options: [Real Number]: The AOP field allows the user to define the argument of periapsis of a celestial body's orbit about its central body, in the FK5 coordinate system. (Only used when the user selects Analytic for the Ephemeris field on the Solar System.) Units: deg.
TA	Default: 358.127085. Options: [Real Number]: The TA field allows the user to define the true anomaly of a celestial body's orbit about its central body. (Only used when the user selects Analytic for the Ephemeris field on the Solar System.) Units: deg.

Special Fields Associated with Earth

NutationUpdate Interval	Default: 60. Options: [Real Number ≥ 0]: The NutationUpdateInterval field, on the Earth Celestial Body, determines how often GMAT updates the Nutation matrix used in FK5 reduction. If NutationUpdateInterval is set to zero, the Nutation is updated every time a request is made to calculate the orientation of the Earth. If NutationUpdateInterval is set to a real number greater than zero, then GMAT only updates the Nutation matrix if the number of seconds defined by NutationUpdateInterval have elapsed since the last request for the Earth's orientation data. Units: sec.
-------------------------	---

Special Fields Associated with Luna

RotationData Source	Default: DE405. Options: [DE405, IAU2002]: The RotationDataSource, on the Luna Celestial Body, determines what source GMAT uses to obtain data describing the orientation of the moon with respect to the FK5 system. The RotationDataSource field is only used for lunar orientation data when calculating moon-based coordinate systems with the axes types of Fixed and Equator. Units: N/A
---------------------	--

Table 2.19: Fields Associated with a Coordinate System

Field	Options and Description
Origin	Default: Earth. Options: [Any celestial body, barycenter, libration point, or spacecraft]: The Origin field allows the user to select the origin of a coordinate system. Units: N/A .
Axes	Default: MJ2000Eq . Options: [MJ2000Eq, MJ2000Ec, EarthFixed, BodyFixed, TOEEq, TOEEc, MOEEq, MOEEc, TODEq, TODEc, MODEq, MODEc, ObjectReferenced, Equator, BodyFixed, BodyInertial,GSE, GSM]: Units: N/A.
Primary	Default: Earth . Options: [Any celestial body, barycenter, libration point, or spacecraft, except the object chosen as in the Secondary field]: The Primary field is only active when Axes is set to ObjectReferenced. Otherwise, GMAT ignores the Primary field. Units: N/A .

Table 2.19: (Fields Associated with a Coordinate System...continued)

Field	Options and Description
Secondary	Default: Luna . Options: [Any celestial body, barycenter, libration point, or spacecraft, except the object chosen as in the Primary field]: The Secondary field is only active when Axes is set to ObjectReferenced. Otherwise, GMAT ignores the Secondary field. Units: N/A .
Epoch	Default: 21545.0. Options: [Real Number ≥ 0]: The Epoch field is only active if the Axes field is defined by an epoch referenced axis system: MOEEq, MOEEc, TOEEq, TOEEc. Units: Days .
XAxis	Default: R. Options: [R, -R, V, -V, N, -N]: The X field is only active if the Axes field is set to ObjectReferenced. Otherwise, GMAT ignores the X field. Units: N/A.
YAxis	Default: No Default. Options: [R, -R, V, -V, N, -N]: The Y field is only active if the Axes field is set to ObjectReferenced. Otherwise, GMAT ignores the Y field. Units: N/A.
ZAxis	Default: N . Options: [R, -R, V, -V, N, -N]: The Z field is only active if the Axes field is set to ObjectReferenced. Otherwise, GMAT ignores the Z field. Units: N/A.
UpdateInterval	Default: 60. Options: [Real Number ≥ 0]: Units: seconds.

Table 2.20: Fields Associated with MATLAB Functions

Field	Options and Description
FunctionPath	Default: \matlab\work. Options: [Any valid path for Operating System]: Units: N/A.

Draft: Work in Progress

CHAPTER 2. OBJECTS AND RESOURCES

Chapter 3

Commands and Events

3.1 Propagation

Table 3.1: Propagate Command

ScriptSyntax	
Propagate Mode BackProp <i>PropagatorName</i> (SatList1,{StopCondList1}) ... BackProp <i>PropagatorName</i> (SatListN,{StopCondListN})	
Option	Option Description
BackProp	Default: None. Options: [Backwards or None]: The BackProp option allows the user to set the flag to enable or disable backwards propagation for all spacecraft in the the SatListN option. The Backward Propagation GUI check box field stores all the data in BackProp. A check indicates backward propagation is enabled and no check indicates forward propagation. In the script, BackProp can be the word Backwards for backward propagation or blank for forward propagation. Units: N/A.
Mode	Default: None. Options: [Synchronized or None]: The Mode option allows the user to set the propagation mode for the propagator that will affect all of the spacecraft added to the SatListN option. For example, if synchronized is selected, all spacecraft are propagated at the same step size. The Propagate Mode GUI field stores all the data in Mode. In the script, Mode is left blank for the None option and the text of the other options available is used for their respective modes. Units: N/A.
<i>PropagatorName</i>	Default: DefaultProp. Options: [Default propagator or any user-defined propagator]: The <i>PropagatorName</i> option allows the user to select a user defined propagator to use in spacecraft and/or formation propagation. The Propagator GUI field stores all the data in <i>PropagatorName</i> . Units: N/A.
SatListN	Default: DefaultSC. Options: [Any existing spacecraft or formations, not being propagated by another propagator in the same Propagate event. Multiple spacecraft must be expressed in a comma delimited list format.]: The SatListN option allows the user to enter all the satellites and/or formations they want to propagate using the <i>PropagatorName</i> propagator settings. The Spacecraft List GUI field stores all the data in SatListN. Units: N/A.

Table 3.1: Propagate Command ...continued

StopCondListN /Parameter	Default: DefaultSC.ElapsedSecs =. Options: [Any single element user accessible spacecraft parameter followed by an equal sign]. The StopCondListN option allows the user to enter all the parameters used for the propagator stopping condition. See the StopCondListN/Condition Option/Field for additional details to the StopCondListN option. Units: N/A.
StopCondListN /Condition	Default: 8640.0. Options: [Real Number, Array element, Variable, spacecraft parameter, or any user defined parameter]. The StopCondListN option allows the user to enter the propagator stopping condition's value for the StopCondListN Parameter field. Units: Dependant on the condition selected.

Script Examples

```

% Single spacecraft propagation with one stopping condition
% Syntax #1
Propagate DefaultProp(DefaultSC, {DefaultSC.ElapsedSecs = 8640.0});

% Single spacecraft propagation with one stopping condition
% Syntax #2
Propagate DefaultProp(DefaultSC) {DefaultSC.ElapsedSecs = 8640.0};

% Single spacecraft propagation by one integration step
Propagate DefaultProp(DefaultSC);

% Multiple spacecraft propagation by one integration step
Propagate DefaultProp(Sat1, Sat2, Sat3);

% Single formation propagation by one integration step
Propagate DefaultProp(DefaultFormation);

% Single spacecraft backwards propagation by one integration step
Propagate Backwards DefaultProp(DefaultSC);

% Two spacecraft synchronized propagation with one stopping condition
Propagate Synchronized DefaultProp(Sat1, Sat2, {DefaultSC.ElapsedSecs = 8640.0});

% Multiple spacecraft propagation with multiple stopping conditions and propagation settings
% Syntax #1
Propagate Prop1(Sat1,Sat2, {Sat1.ElapsedSecs = 8640.0, Sat2.MA = 90}) ...
Prop2(Sat3, {Sat3.TA = 0.0});

% Multiple spacecraft propagation with multiple stopping conditions and propagation settings
% Syntax #2
Propagate Prop1(Sat1,Sat2) {Sat1.ElapsedSecs = 8640.0, Sat2.MA = 90} ...
Prop2(Sat3) {Sat3.TA = 0.0};

```

3.2 Control Flow

Table 3.2: If Command

Script Syntax	
<i>(Simple If statement)</i>	
<pre>If <logical expression>; <Statements>; EndIf;</pre>	
<i>(Compound If statement)</i>	
<pre>If <logical expression> <logical expression> & <logical expression>; <Statements>; EndIf;</pre>	
<i>(If-Else statement)</i>	
<pre>If <logical expression>; <Statements>; Else; <Statements>; EndIf;</pre>	
Option	Option Description
<logical expression>	Default: DefaultSC.ElapsedDays < 1.0. Options:[Arg1 < Arg2 and < can be >, <, >=, <=, ==, ~=]. Arg1 and Arg2 can be any of the following: Real Number, Array element, Variable, Spacecraft Parameter or any other user defined parameter. Units: N/A.
<Statements>	Default: N/A. Options:[Any script line that can be in the mission sequence]. Units: N/A.
	Default: N/A. Options:[N/A]. The option allows the user to set an OR operator in between <logical expression>s. Units: N/A.
&	Default: N/A. Options:[N/A]. The & option allows the user to set an AND operator in between <logical expression>s. Units: N/A.
Script Examples	
<pre>If DefaultSC.ElapsedDays < 1; Propagate DefaultProp(DefaultSC , { DefaultSC.ElapsedDays = 0.01 }); EndIf;</pre>	
<pre>If MyVariable < MyArray(1,1); MyArray(1,1) = 5; EndIf;</pre>	
<pre>If DefaultSC.Earth.TA < MyArray(1,2); Propagate DefaultProp(DefaultSC); EndIf;</pre>	

Table 3.3: While Command

Script Syntax	
<i>(Simple While Loop)</i>	
<pre>While <logical expression>; <Statements>; EndWhile;</pre>	
<i>(Compound While Loop)</i>	
<pre>While <logical expression> <logical expression> & <logical expression> <Statements> EndWhile</pre>	
Option	Option Description
<logical expression>	Default: DefaultSC.ElapsedDays < 1.0. Options:[Arg1 < Arg2 and < can be >, <, >=, <=, ==, ~=]. Arg1 and Arg2 can be any of the following: Real Number, Array, Variable, Spacecraft Parameter or any other user defined parameter. Units: N/A.
<Statements>	Default: N/A. Options:[Any script line that can be in the mission sequence]. Units: N/A.
	Default: N/A. Options:[N/A]. The option allows the user to set an OR operator in between <logical expression>s. Units: N/A.
&	Default: N/A. Options:[N/A]. The & option allows the user to set an AND operator in between <logical expression>s. Units: N/A.
Script Examples	
<pre>While DefaultSC.ElapsedDays < 1; Propagate DefaultProp(DefaultSC , { DefaultSC.ElapsedDays = 0.01 }); EndWhile;</pre>	
<pre>While MyVariable < MyArray(1,1); MyArray(1,1) = 5; EndWhile;</pre>	

Table 3.4: For Command

Script Syntax	
<i>(Simple For Loop)</i>	
<pre>For Variable = Start:End; <Statements>; EndFor;</pre>	
<i>(Expanded For Loop)</i>	
<pre>For Variable = Start:Increment:End; <Statements>; EndFor;</pre>	
Command Description	
<p>The for loop is a control flow statement that allows portions of code to be executed iteratively using an explicit loop variable (Wikipedia). GMAT for loops are three-expression loops that allow the user to set the initial value of the loop variable, its increment, and the test to exit the loop. A parameter must be defined explicitly using a Create Variable statement or GUI equivalent before it can be used in a for loop command statement. The only parameter type that can be used as a loop variable is the variable type. The parameters used to define Start, Increment, and End can be any of the following GMAT parameters: integer??(real), variable, array element, spacecraft property.</p> <p>GMAT allows the for loop variable to be changed inside the loop by the user, and the resulting behavior of the for loop is equivalent to the behavior defined in ANSI C. If a change is made to the loop variable inside of the loop, if this change causes the exit test to be violated, GMAT will exit the for loop.</p>	
Option	Option Description
<i>Variable</i>	Default: No Default. Options: [Variable]: The <i>Variable</i> option allows the user to define the variable that will store the For Loop numeric range. Units: N/A.
<i>Start</i>	Default: 1. Options: [Real Number, Array element, Variable, or any user defined parameter]. The <i>Start</i> option allows the user to set the starting numeric range value of the For Loop. <i>Start</i> can be equal to <i>End</i> , but the For Loop will not execute. Units: N/A.
<i>Increment</i>	Default: 1. Options: [Real Number, Array element, Variable, or any user defined parameter]. The <i>Increment</i> option allows the user to set the numeric range increment value of the For Loop. When the <i>Increment</i> option is left out of the script syntax the default value is used. If an <i>Increment</i> value of 0 is used, the For Loop should not execute but GMAT should continue to run. If <i>End</i> > <i>Start</i> and <i>Increment</i> < 0, then the For Loop should not execute. If <i>Start</i> > <i>End</i> and <i>Increment</i> > 0, then the For Loop should not execute. Units: N/A.
<i>End</i>	Default: 10. Options: [Real Number, Array, Variable, or any user defined parameter]. The <i>End</i> option allows the user to set the ending numeric range value of the For Loop. <i>End</i> can be equal to <i>Start</i> , but the For Loop will not execute. Units: N/A.
Script Examples	
<pre>% Output the value of the For loop Variable to a file</pre>	

Table 3.4: For Command ...continued

```
For I = 1:1:10;  
    GMAT testVar = I;  
    Report DefaultReportFile I;  
EndFor;
```

3.3 Solver-related

Table 3.5: Target Command

Script Syntax	
<pre>Target <i>SolverName</i> ; <Statements> EndTarget;</pre>	
Option	Option Description
<i>SolverName</i>	Default: DefaultDC. Options:[Any differential corrector existing in the resource tree or created in the script]; The <i>SolverName</i> option allows the user to choose between any previously created differential correctors for use in a targeting sequence. For example, to begin a targeting sequence using DefaultDC, the script is Target DefaultDC. Units: N/A.
<Statements>	Default: None. Options:[Any non-targeter and non-optimizer command lines used in the mission sequence, as well as the targeter dependent command lines Achieve and Vary.]; Units: N/A.
Script Examples	
<pre>% Beginning and ending syntax for the Target command Target DefaultDC; EndTarget;</pre>	

Table 3.6: Optimize Command

Script Syntax	
<pre>Optimize <i>SolverName</i> ; <Statements> EndOptimize;</pre>	
Option	Option Description
<i>SolverName</i>	Default: DefaultSQP. Options:[Any existing optimizer]; The <i>SolverName</i> field allows the user to choose between any previously created optimizer for use in an optimization sequence. For example, to begin an optimization sequence using DefaultSQP, the script is Optimize DefaultSQP. Units: N/A.
<Statements>	Default: None. Options:[Any non-targeter and non-optimizer command lines used in the mission sequence, as well as the optimizer dependent command lines Vary, NonLinearConstraint, and Minimize.]; Units: N/A.
Script Examples	
<pre>% Beginning and ending syntax for the Optimize command Optimize DefaultDC; EndOptimize;</pre>	

Table 3.6: Optimize Command ...continued

Table 3.7: Achieve Command

```
ScriptSyntax: Achieve SolverName (Goal = Arg1, {Tolerance = Arg2});
```

Option	Option Description
<i>Goal</i>	Default: DefaultSC.Earth.RMAG. Options: [Spacecraft parameter, Array element, Variable, or any other single element user defined parameter, excluding numbers]: The <i>Goal</i> option allows the user to select any single element user defined parameter, except a number, to Achieve.
<i>Arg1</i>	Default: 42165. Options: [Real Number, Array element, Variable, or any user defined parameter that obeys the conditions of Chapter 2 for the selected <i>Goal</i>] The <i>Arg1</i> option is the desired value for <i>Goal</i> after the solver has converged. Units: N/A.
<i>Tolerance</i>	Default: 0.1. Options: [Real Number, Array element, Variable, or any user defined parameter > 0]: The <i>Tolerance</i> option sets <i>Arg2</i> . <i>Arg2</i> is the convergence tolerance for <i>Arg1</i> . Units: N/A.
<i>SolverName</i>	Default: DefaultDC. Options: [Any user defined differential corrector]: The <i>SolverName</i> option allows the user to choose which solver to assign to the Achieve command. Units: N/A.

Script Examples

```
Achieve DefaultDC(DefaultSC.Earth.RMAG = 42165.0, {Tolerance = 0.1});
```

Table 3.8: Vary Command

Draft: Work in Progress

ScriptSyntax

Vary *SolverName* (*Variable* = InitialGuess, {Perturbation = Arg1, MaxStep = Arg2, Lower = Arg3, ... Upper = Arg4, AdditiveScaleFactor = Arg5, MultiplicativeScaleFactor = Arg6})

Option	Option Description
--------	--------------------

Parameters Associated with All Solvers.

SolverName Default: DefaultDC. Options: [Any user defined solver]: The *SolverName* option allows the user to choose which solver to assign to the vary command. Units: N/A.

Variable Default: DefaultIB.V. Options: [Spacecraft parameter, Array element, Variable, or any other single element user defined parameter, excluding numbers] The *Variable* option allows the user to select any single element user defined parameter, except a number, to vary. For example, DefaultIB.V, DefaultIB.N, DefaultIB.Element1, DefaultSC.TA, Array(1,1), and Variable are all valid values. The three element burn vector or multidimensional Arrays are not valid values. Units: N/A.

Table 3.8: Vary Command ... continued

InitialGuess	Default: 0.5. Options: [Real Number, Array element, Variable, or any user defined parameter that obeys the conditions of Chapter 2 for the selected <i>Variable</i>]: The InitialGuess option allows the user to set the initial guess for the selected <i>Variable</i> . Units: km/s.
Lower	Default: 0.0. Options: [Real Number, Array element, Variable, or any user defined parameter (Upper > Lower)]: The Lower option allows the user to set Arg3 to the lower bound of the quantity being varied. Units: N/A.
Upper	Default: 3.14159. Options: [Real Number, Array element, Variable, or any user defined parameter (Upper > Lower)]: The Upper option allows the user to set Arg4 to the upper bound of the quantity being varied. Units: N/A.

Parameters Associated with Differential Corrector.

Perturbation	Default: 1e-4. Options: [Real Number, Array element, Variable, or any user defined parameter > 0]: The Perturbation option is set by specifying a value for Arg1. The value of Arg1 is the perturbation size in calculating the finite difference derivative. Units: N/A.
MaxStep	Default: 0.2. Options: [Real Number, Array element, Variable, or any user defined parameter > 0]: The MaxStep option is set by specifying a value for Arg2. The value of Arg2 limits the size of the step taken during an interaction of the differential corrector. Units: N/A.

Parameters Associated with fmincon Optimizer.

Additive Scale Factor	Default: 0. Options: [Real Number, Array element, Variable, or any user defined parameter]: The AdditiveScaleFactor Field is used to nondimensionalize the independent variable. fmincon sees only the nondimensional form of the variable. The nondimensionalization is performed using the following equation: $x_n = (x_d - a)/m$. (x_n is the non-dimensional parameter. x_d is the dimensional parameter. a = additive scale factor. m = multiplicative scale factor.) Units: N/A.
Multiplicative Scale Factor	Default: 1.0. Options: [Real Number, Array element, Variable, or any user defined parameter]: The MultiplicativeScaleFactor Field is used to nondimensionalize the independent variable. fmincon sees only the nondimensional form of the variable. The nondimensionalization is performed using the following equation: $x_n = (x_d - a)/m$. (x_n is the non-dimensional parameter. x_d is the dimensional parameter. a = additive scale factor. m = multiplicative scale factor.) Units: N/A.

Script Examples

```
% Impulsive Burn Vary Command
Vary DefaultDC(DefaultIB.V = 0.5, {Perturbation = 0.0001, MaxStep = 0.2, ...
Lower = 0, Upper = 3.14159});
```

Table 3.9: Minimize Command

Draft: Work in Progress

3.3. SOLVER-RELATED

49

Script Syntax: Minimize *OptimizerName* (*Arg*)

Table 3.9: Minimize Command ... continued

Option	Option Description
<i>OptimizerName</i>	Default: SQP1. Options:[Any existing fmincon solver]: The <i>OptimizerName</i> option allows the user to specify which solver to use to minimize the cost function. Units: N/A.
Arg	Default: DefaultSC.ECC. Options:[Variable, Spacecraft parameter, or Array element]. The Arg field allows the user to specify the function to be minimized upon convergence of the solver given by <i>OptimizerName</i> . Arg can be any of the following: Variable, Array element, or Spacecraft Parameter or any other 1x1 numeric user defined parameter. Units: N/A.
Script Examples	
<pre>% Minimize the eccentricity of Sat, using fminconSQP Minimize fminconSQP(Sat.ECC);</pre>	
<pre>% Minimize the Variable DeltaV, using fminconSQP Minimize fminconSQP(DeltaV);</pre>	
<pre>% Minimize the first component of MyArray, using fminconSQP Minimize fminconSQP(MyArray(1,1));</pre>	

Table 3.10: NonLinearConstraint Command

Script Syntax: `NonLinearConstraint OptimizerName(<logical expression>)`

Option	Option Description
<code>OptimizerName</code>	Default: SQP1. Options:[Any existing fmincon solver]: The <code>OptimizerName</code> option allows the user to specify which solver to use in satisfying nonlinear constraints. Units: N/A.
<code><logical expression></code>	Default: <code>DefaultSC.SMA = 7000</code> . Options:[<code>Arg1 ≤ Arg2</code> where <code>≤</code> can be <code>>=</code> , <code><=</code> , <code>=</code>]. The logical expression field allows the user to specify the constraint to be satisfied upon convergence of the solver given by <code>OptimizerName</code> . <code>Arg1</code> and <code>Arg2</code> can be any of the following: Real Number, a 1-D Array (column vector), Array element, Variable, Spacecraft Parameter or any other numeric user defined parameter. If <code>Arg1</code> is a 1-D Array, then <code>Arg2</code> must be a 1-D Array with the same dimensions and vice-versa. Units: N/A.

Script Examples

```
% Constrain the SMA of Sat to be 7000 km, using fminconSQP
NonLinearConstraint fminconSQP( Sat.SMA = 7000 );

% Constrain the SMA of Sat to be less than or equal to 7000 km, using fminconSQP
NonLinearConstraint fminconSQP( Sat.SMA <= 7000 );

% Constrain the SMA of Sat to be greater than or equal to 7000 km, using fminconSQP
NonLinearConstraint fminconSQP( Sat.SMA >= 7000a );
```

3.4 Miscellaneous

Table 3.11: Maneuver Command

Script Syntax: Maneuver *BurnName* (*SpacecraftName*);

Option	Option Description
<i>BurnName</i>	Default: DefaultIB. Options:[Any impulsive burn existing in the resource tree or created in the script]: The <i>BurnName</i> field allows the user to choose between any previously created impulsive burn. As an example, to maneuver DefaultSC using DefaultIB, the script line would appear as Maneuver DefaultIB(DefaultSC). Units: N/A.
<i>SpacecraftName</i>	Default: DefaultSC. Options:[Any spacecraft existing in the resource tree or created in the script]: The <i>SpacecraftName</i> field allows the user to select which spacecraft to maneuver using the maneuver selected in the <i>BurnName</i> field. Units: N/A.

Script Examples

```
% Impulsive Burn
Maneuver DefaultIB(DefaultSC);
```

Table 3.12: BeginFiniteBurn Command

Script Syntax: BeginFiniteBurn *ManeuverName* (*SpacecraftName*);

Option	Option Description
<i>ManeuverName</i>	Default: DefaultFB. Options:[Any finite burn existing in the resource tree or created in the script]: The <i>ManeuverName</i> option allows the user to choose between any previously created finite burn. As an example, to maneuver DefaultSC using DefaultFB, the script line would appear as Maneuver DefaultFB(DefaultSC). Units: N/A.
<i>SpacecraftName</i>	Default: DefaultSC. Options:[Any spacecraft existing in the resource tree or created in the script]: The <i>SpacecraftName</i> option allows the user to select which spacecraft to maneuver using the maneuver selected in the <i>ManeuverName</i> option. Units: N/A.

Script Examples

```
% Default BeginFiniteBurn syntax
BeginFiniteBurn DefaultFB(DefaultSC);
```

Table 3.13: EndFiniteBurn Command

Script Syntax: `EndFiniteBurn ManeuverName (SpacecraftName);`

Option	Option Description
<i>ManeuverName</i>	Default: DefaultFB. Options:[Any finite burn existing in the resource tree or created in the script]: The <i>ManeuverName</i> option allows the user to choose between any previously created finite burn. As an example, to maneuver DefaultSC using DefaultFB, the script line would appear as <code>Maneuver DefaultFB(DefaultSC)</code> . Units: N/A.
<i>SpacecraftName</i>	Default: DefaultSC. Options:[Any spacecraft existing in the resource tree or created in the script]: The <i>SpacecraftName</i> option allows the user to select which spacecraft to maneuver using the maneuver selected in the <i>ManeuverName</i> option. Units: N/A.

Script Examples

`% Default EndFiniteBurn syntax
EndFiniteBurn DefaultFB(DefaultSC);`

Table 3.14: CallFunction Command

Script Syntax

Function call with Inputs and Outputs

`GMAT [OutputList] = Function (InputList)`

Function call with Outputs only

`GMAT [OutputList] = Function`

Function call with Inputs only

`GMAT Function (InputList)`

Function call with no Inputs or Outputs

`GMAT Function`

Option	Option Description
OutputList	Default: None. Options:[Variables, Arrays, S/C Parameters, any other user-defined parameters, or blank. Multiple outputs must be expressed in a comma delimited list format]: The OutputList option allows the user to set the output of <i>Function</i> to a user defined parameter. Units: N/A.
InputList	Default: None. Options:[Variables, Arrays, S/C Parameters, any other user-defined parameters, or blank. Multiple inputs must be expressed in a comma delimited list format.]: The InputList option allows the user to set the input of <i>Function</i> to a user defined parameter. Units: N/A.
<i>Function</i>	Default: None. Options:[GMAT or Matlab Function]: The <i>Function</i> option allows the user to set the function that will be called in a specific location of the mission sequence. The function has to be defined before it can be used in the CallFunction Command. Units: N/A.

Script Examples

Table 3.14: CallFunction Command ... continued

```

% Matlab function call without inputs or outputs
% Syntax 1
GMAT clearAll;

% Matlab function call without inputs or outputs
% Syntax 2
GMAT [] = clearAll( );
    
```

Table 3.15: Toggle Command

Script Syntax: Toggle *OutputNames* Arg

Option	Option Description
<i>OutputNames</i>	Default: DefaultOpenGL . Options:[Any OpenGL, Report, XYplot, or any other Plot/Report type]: The Toggle option allows the user to assign the Plot/Report(s) to be toggled. When more than one Plot/Report is being toggled they need to be separated by a space. Units: N/A.
Arg	Default: On. Options:[On or Off]: The Arg option allows the user to turn off or on the data output to a Plot/Report. Units: N/A.

Script Examples

```

% Turn off Report file for the first day of propagation
Toggle ReportFile1 Off
Propagate DefaultProp(DefaultSC, DefaultSC.ElapsedDays = 1);
Toggle ReportFile1 On
Propagate DefaultProp(DefaultSC, DefaultSC.ElapsedDays = 1);

% Turn off XYPlot and Report file for the first day of propagation
Toggle XYPlot1 ReportFile1 Off
Propagate DefaultProp(DefaultSC, DefaultSC.ElapsedDays = 1);
Toggle XYPlot1 ReportFile1 On
Propagate DefaultProp(DefaultSC, DefaultSC.ElapsedDays = 1);
    
```

Table 3.16: Report Command

Draft: Work in Progress

Script Syntax: Report *ReportName* *DataList*

Option	Option Description
<i>ReportName</i>	Default: N/A. Options:[Any ReportFile created]: The <i>ReportName</i> option allows the user to specify the ReportFile for data output. Units: N/A.
<i>DataList</i>	Default: N/A. Options:[Spacecraft parameter, Array, Variable, String, or any other single user defined parameter]: The <i>DataList</i> option allows the user to output data to the Filename specified by the <i>ReportName</i> . Multiple objects can be in the <i>DataList</i> when they are separated by spaces. Units: N/A.

Table 3.16: Report Command ... continued

Script Examples
<pre>% Report the time and position of DefaultSC Report DefaultReport DefaultSC.A1ModJulian DefaultSC.X DefaultSC.Y DefaultSC.Z;</pre>

Table 3.17: ScriptEvent Command

Script Syntax				
<pre>BeginScript; <Statements>; EndScript;</pre>				
<table border="1"> <thead> <tr> <th>Option</th> <th>Option Description</th> </tr> </thead> <tbody> <tr> <td><Statements></td> <td>Default: N/A. Options:[Any valid line of GMAT script]. Units: N/A.</td> </tr> </tbody> </table>	Option	Option Description	<Statements>	Default: N/A. Options:[Any valid line of GMAT script]. Units: N/A.
Option	Option Description			
<Statements>	Default: N/A. Options:[Any valid line of GMAT script]. Units: N/A.			

Script Examples
<pre>% Assignment command inside Script Event BeginScript; GMAT testVar = 24; EndScript;</pre>

Table 3.18: Pause Command

Script Syntax: Pause		
<table border="1"> <thead> <tr> <th>Command Description</th> </tr> </thead> <tbody> <tr> <td>The Pause command allows the user to pause a running GMAT script.</td> </tr> </tbody> </table>	Command Description	The Pause command allows the user to pause a running GMAT script.
Command Description		
The Pause command allows the user to pause a running GMAT script.		
<table border="1"> <thead> <tr> <th>Script Examples</th> </tr> </thead> <tbody> <tr> <td> <pre>% Pause between propagation sequences Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedSecs = 8640.0; Pause; Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedDays = 10.0;</pre> </td> </tr> </tbody> </table>	Script Examples	<pre>% Pause between propagation sequences Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedSecs = 8640.0; Pause; Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedDays = 10.0;</pre>
Script Examples		
<pre>% Pause between propagation sequences Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedSecs = 8640.0; Pause; Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedDays = 10.0;</pre>		

Table 3.19: Stop Command

Script Syntax: Stop		
<table border="1"> <thead> <tr> <th>Command Description</th> </tr> </thead> <tbody> <tr> <td>The Stop command allows the user to stop a running GMAT script.</td> </tr> </tbody> </table>	Command Description	The Stop command allows the user to stop a running GMAT script.
Command Description		
The Stop command allows the user to stop a running GMAT script.		
<table border="1"> <thead> <tr> <th>Script Examples</th> </tr> </thead> <tbody> <tr> <td> <pre>% Stop between propagation sequences Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedSecs = 8640.0; Stop;</pre> </td> </tr> </tbody> </table>	Script Examples	<pre>% Stop between propagation sequences Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedSecs = 8640.0; Stop;</pre>
Script Examples		
<pre>% Stop between propagation sequences Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedSecs = 8640.0; Stop;</pre>		

Table 3.19: Stop Command ...continued

```
Propagate DefaultProp(DefaultSC) DefaultSC.ElapsedDays = 10.0;
```

Table 3.20: Save Command

Script Syntax: Save *ObjectList*

Option	Option Description
<i>ObjectList</i>	Default: DefaultSC. Options:[Any user-defined objects, excluding variables and arrays]: The <i>ObjectList</i> option allows the user to save the properties of the objects selected to the output path. Multiple objects can be in the <i>ObjectList</i> when they are separated by spaces. Units: N/A.

Script Examples

```
% Save DefaultSC data after a 1 day propagation
Propagate DefaultProp(DefaultSC, DefaultSC.ElapsedDays = 1);
Save DefaultSC;
% Save Impulsive Burn and DefaultSC data after a Targeter sequence
EndTarget;
Save DefaultIB DefaultSC;
```

Draft: Work in Progress

Index

AAttitudeRateStateType, 16
AOP, 13, 36
AZI, 14
Accuracy, 24
Add, 30, 32, 33
AnalyticModel, 34
AngularVelocityX, 18
AngularVelocityY, 18
AngularVelocityZ, 18
AnomalyType, 12
AttitudeCoordinateSystem, 16
AttitudeMode, 16
AttitudeStateType, 16
Axes, 25, 26, 29, 36
Axis, 19
Backwards Propagation, 39
BeginFiniteBurn, 52
BodyNames, 35
BurnScaleFactor, 26
CallFunction, 53
Cd, 15
CentralBody, 21
ColumnWidth, 32
CoordinateSystem, 11, 30
Cr, 15
DCM11, 17
DCM12, 17
DCM33, 17
DECV, 14
DEC, 14
DataCollectFrequency, 29
DateFormat, 11
Degree, 22
DerivativeCheck, 27
Diagnostics, 27
DiffMaxChange, 27
DiffMinChange, 27
Display, 27
Drag.AtmosphereModel, 22
Drag.F107A, 22
Drag.F107, 22
Drag.MagneticIndex, 23
DragArea, 15
Drag, 22
DrawWireFrame, 29
DryMass, 15
EA, 13
ECC, 13, 35
EarthSunLines, 30
EclipticPlane, 29
EndFiniteBurn, 53
EphemerisFile, 34
EphemerisUpdateInterval, 34
Ephemeris, 34
Epoch, 11, 37
EquatorialRadius, 35
ErrorControl, 23
EulerAngle1, 16
EulerAngle2, 17
EulerAngle3, 17
EulerAngleRate1, 17
EulerAngleSequence, 16
FPA, 14
FileName, 32
FixedFovAngle, 32
Flattening, 35
For, 43
FuelDensity, 18
FuelMass, 18
FunctionPath, 37
GradConstr, 27, 28
Gravity.PrimaryBody.Degree, 22
Gravity.PrimaryBody.Model, 22
Gravity.PrimaryBody.Order, 22
Grid, 29, 33
HA, 13
INC, 13, 35
If, 41
IndVar, 33
Initial Value, 48
InitialEpoch, 35
InitialStepSize, 24
Integrators
 Script Fields, 23
LeftJustify, 32
Libration Point, 34
Lower, 48
MA, 13
Maneuver, 52
MaxFunEvals, 27
MaxIter, 27
MaxStepAttempts, 24
MaxStep, 24, 48
Maximum Iterations, 28
MeanLongitude, 15
MinIntegrationError, 24

Draft: Work in Progress

59

INDEX

MinStep, 24
Minimize, 48
MultiplicativeScaleFactor, 48
Mu, 35
NomIntegrationError, 24
NonLinearConstraint, 51
NumPointsToRedraw, 29
NutationUpdateInterval, 36
Optimize, 45
Order, 22
Origin, 19, 25, 26, 36
Pause, 56
PerspectiveMode, 32
Perturbation, 48
PointMasses, 23
Point, 34
Precision, 32
PressureRegulated, 18
Pressure, 18
PrimaryBodies, 21
Primary, 34, 36
Propagation Mode, 39
RAAN, 13, 35
RAV, 14
RA, 14
RMAG, 14
RadApo, 13
RadPer, 13
RefTemperature, 18
Remove, 30
ReportStyle, 28
Report, 54
RotationDataSource, 36
SMA, 12, 35
SRPArea, 16
SRP, 23
Save, 57
ScriptEvent, 56
Secondary, 34, 37
ShowPlot, 28, 33
ShowProgress, 28
Solar System
 Script Fields, 34
SolverIterations, 29, 33
StateType, 11
Stop, 56
TA, 13, 36
Tank, 19
TargeterTextFile, 28
Target, 45
Temperature, 18
ThrustScaleFactor, 19
Thrusters, 26
Toggle, 54
TolCon, 27
TolFun, 27
TolX, 27
Tolerance, 46
Type, 23
UpdateInterval, 37
UpdatePlotFrequency, 29
Upper, 48
UseCentralDifferences, 28
UseFixedFov, 32
UseInitialView, 31
UseTTForEphemeris, 34
VMAG, 14
VX, 12
VY, 12
VZ, 12
VectorFormat, 25
ViewDirection, 31
ViewPointReference, 30
ViewPointVector, 30
ViewScaleFactor, 31
ViewUpAxis, 31
ViewUpCoordinateSystem, 31
Volume, 18
While, 42
WriteHeaders, 32
XYPlane, 29
X.Direction, 19
X, 12, 37
Y.Direction, 19
Y, 12, 37
Z.Direction, 19
ZeroFill, 32
Z, 12, 37
h, 15
k, 15
p, 15
q1, 17
q2, 17
q3, 17
q4, 17
q, 15