JSC-61777-models/interactions/gravity_torque

JSC Engineering Orbital Dynamics Gravity Torque Model

Simulation and Graphics Branch (ER7)
Software, Robotics, and Simulation Division
Engineering Directorate

Package Release JEOD v5.2

Document Revision 1.1
September 2024

National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
Houston, Texas

JSC Engineering Orbital Dynamics
Gravity Torque Model

Document Revision 1.1
September 2024

Blair Thompson

Simulation and Graphics Branch (ER7)
Software, Robotics, and Simulation Division
Engineering Directorate

National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
Houston, Texas

Abstract

The JEOD Gravity Torque Model computes the gravitational torque acting on a spacecraft due to any number of gravitational bodies, both spherical and non-spherical. The torque is computed from the gravity gradient, which is computed by the JEOD Gravity Model. The spacecraft mass properties, in the form of an inertia tensor (matrix), are assumed to be known. The total gravitational torque acting on the spacecraft is computed in the spacecraft body-fixed frame and then transformed into the body structural frame for output.

This document describes the implementation of the Gravity Torque Model. It is part of a series of inter-related documents that describe the model requirements, specifications, mathematical theory, and architecture of the model. A user guide is also provided to assist with implementing the model in Trick simulations.

Contents

Chapter 1

Introduction

1.1 Model Description

This documentation describes the design and testing of the gravity torque routines in the JSC Engineering Orbital Dynamics (JEOD) Gravity Torque Model. These routines are derived from published models that are commonly accepted in the aerospace industry.

Included in this documentation are verification and validation cases that describe tests done on the algorithms to verify that they are working correctly and computing the correct values for given input data. There is also a User Guide which describes how to incorporate the above mentioned routines as part of a Trick simulation.

The parent document to this model document is the JEOD Top Level Document [3].

1.2 Document History

Author	Date	Revision	Description
Blair Thompson	Nov 2009	1.0	Initial document
Blair Thompson	Mar 2010	1.1	Separate gradient degree/order controls

1.3 Document Organization

This document is formatted in accordance with the NASA Software Engineering Requirements Standard [6] and is organized into the following chapters:

Chapter 1: Introduction - This introduction contains three sections: description of model, document history, and organization. The first section provides the introduction to the Gravity Torque Model and its reason for existence. It contains a brief description of the interconnections with other models, and references to any supporting documents. It also lists the document that is parent to this one. The second section displays the history of this document which includes author, date, and reason for each revision. The final section contains a description of the how the document is organized.

Chapter 2: Product Requirements - Describes requirements for the Gravity Torque Model.

Chapter 3: Product Specification - Describes the underlying theory, architecture, and design of the Gravity Torque Model in detail. It is organized in three sections: Conceptual Design, Mathematical Formulations, and Detailed Design.

Chapter 4: User Guide - Describes how to use the Gravity Torque Model in a Trick simulation. It is broken into three sections to represent the JEOD defined user types: Analysts or users of simulations (Analysis), Integrators or developers of simulations (Integration), and Model Extenders (Extension).

Chapter 5: Verification and Validation - Contains Gravity Torque Model verification and validation procedures and results.

Chapter 2

Product Requirements

The Gravity Torque Model shall meet the JEOD project requirements specified in the JEOD top-level document.

Requirement gravitytorque_1: Gravity Gradient Matrix

Requirement:

The gravity gradient shall be modeled as a 3x3 Jacobian (matrix) that is the partial derivative of the gravity acceleration vector with respect to position in rectangular inertial coordinates.

Rationale:

The JEOD Gravity Model computes the gravity gradient for all gravitational bodies as a 3x3 Jacobian (matrix).

Verification:

Inspection

Requirement gravitytorque_2: Spacecraft Mass Properties

Requirement:

The mass properties of any spacecraft shall be modeled as a 3x3 inertia tensor (matrix) expressed in the spacecraft body-fixed coordinate system.

Rationale:

A 3x3 inertia tensor is the widely accepted industry standard for representing the mass properties of any rigid body, and JEOD uses this representation.

Verification:

Inspection

Requirement gravitytorque_3: Spacecraft Attitude

Requirement:

The attitude representation of any spacecraft shall represent a transformation from inertial coordinates to spacecraft body-fixed coordinates.

Rationale:

The mass properties of the spacecraft are expressed in spacecraft body-fixed coordinates. The gravity gradient matrix is expressed in inertial coordinates. In order to compute the gravity torque on the spacecraft in the body-fixed system, the attitude of the spacecraft must be known (inertial to body-fixed).

Verification:

Inspection

Requirement gravitytorque_4: Gravity Torque

Requirement:

The model shall compute the inertial torque vector acting on a spacecraft from a gravity gradient Jacobian matrix, an inertia tensor (matrix), and the spacecraft attitude.

4.1 The torque vector shall be expressed in the spacecraft body-fixed coordinate system. The torque vector is output in the structural reference frame.

Rationale:

The gravity torque acting on a spacecraft affects the attitude of the spacecraft and must be modeled for accurate simulations.

Verification:

Test

Chapter 3

Product Specification

3.1 Conceptual Design

The Gravity Torque Model is a function that computes torque due to gravity gradient acting on a spacecraft. The model is not a stand-alone function. It requires the spacecraft mass properties (inertia tensor) and the gravity gradient (at the spacecraft's current location) to be computed before calling the Gravity Torque Model. The gravity gradient is computed by the JEOD Gravity Model and is used to compute the force of gravity acting on each infinitesimal mass of the spacecraft body, as defined by the spacecraft inertia tensor. The forces, in combination with the mass distribution, are combined to form the total gravity torque acting on the spacecraft.

The Gravity Torque Model is based on the derivation and Ada code of Gottlieb[2]. Sign changes have been made to the inertia tensor in order to be compatible with other JEOD models and to be consistent with aerospace industry standards.[1]

3.2 Mathematical Formulations

3.2.1 Gravity Gradient

The gravity gradient is a 3x3 Jacobian (matrix) that is the partial derivative of the inertial gravitational acceleration vector with respect to the inertial position vector (rectangular coordinates).

$$\nabla \bar{a} = \frac{\partial \bar{a}}{\partial \bar{r}} = \begin{bmatrix} \partial a_x/\partial x & \partial a_x/\partial y & \partial a_x/\partial z \\ \partial a_y/\partial x & \partial a_y/\partial y & \partial a_y/\partial z \\ \partial a_z/\partial x & \partial a_z/\partial y & \partial a_z/\partial z \end{bmatrix} \tag{3.1}$$

The trace of the gravity gradient (sum of the diagonal terms) is the Laplacian of the gravitational potential and will be equal to zero for any gravity field.

$$\nabla^2 V = \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2} = \frac{\partial a_x}{\partial x} + \frac{\partial a_y}{\partial y} + \frac{\partial a_z}{\partial z} = 0 \tag{3.2}$$

[1] Gottlieb's Ada code ([2] p.49) contains sign changes compared to his equations on p.21-23. The signs in the Gravity Torque Model formulation match Gottlieb's Ada code.

where V is the gravitational potential[5].

3.2.2 Spacecraft Mass Properties - Inertia Tensor

The mass properties (i.e., mass distribution) of a rigid body can be expressed as a tensor (matrix) in the spacecraft body-fixed coordinate system as

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \tag{3.3}$$

where

$$\begin{aligned} I_{xx} &= \int_B (\rho_y^2 + \rho_z^2) dm \\ I_{yy} &= \int_B (\rho_x^2 + \rho_z^2) dm \\ I_{zz} &= \int_B (\rho_x^2 + \rho_y^2) dm \end{aligned} \tag{3.4}$$

are the moments of intertia, and

$$\begin{aligned} I_{xy} &= \int_B (-\rho_x \rho_y) dm \\ I_{xz} &= \int_B (-\rho_x \rho_z) dm \\ I_{yz} &= \int_B (-\rho_y \rho_z) dm \end{aligned} \tag{3.5}$$

are the products of inertia[4]. The vector $\bar{\rho}$, whose components are ρ_x, ρ_y, ρ_z, locates each infinitesimal mass of the spacecraft in the body-fixed system. The integrals are computed over the entire spacecraft body in the spacecraft body-fixed frame. Therefore, the inertia tensor is also expressed in the body-fixed frame.

3.2.3 Gravity Torque

The gravity gradient matrix can be transformed from inertial coordinates to spacecraft body-fixed coordinates using a similarity transformation[7].

$$G = B \frac{\partial \bar{a}}{\partial \bar{r}} B^T \tag{3.6}$$

where B is the inertial to body-fixed attitde matrix.

The gravitational torque acting on the spacecraft can then be expressed as[2]

$$\bar{\tau} = \int \bar{\rho} \times G \bar{\rho} dm \tag{3.7}$$

The vector $\bar{\rho}$ is defined in the previous section. The torque can be expressed in terms of I and G as

$$\bar{\tau} = \begin{bmatrix} G_{2,3}(I_{zz} - I_{yy}) - G_{1,3}I_{xy} + G_{1,2}I_{xz} - I_{yz}(G_{3,3} - G_{2,2}) \\ G_{1,3}(I_{xx} - I_{zz}) + G_{2,3}I_{xy} - G_{1,2}I_{yz} - I_{xz}(G_{1,1} - G_{3,3}) \\ G_{1,2}(I_{yy} - I_{xx}) - G_{2,3}I_{xz} + G_{1,3}I_{yz} - I_{xy}(G_{2,2} - G_{1,1}) \end{bmatrix} \tag{3.8}$$

See Gottlieb[2] (p.21-23) for details. The final gravity gradient torques are output in the structural reference frame.

3.3 Detailed Design

The Gravity Torque Model was adopted from Gottlieb's Ada code (see Gottlieb[2] p.49). The inertia tensor I and the gravity gradient G are computed in JEOD by the Dynamics Manager and the Gravity Model. Gottlieb's derivation and Ada code assume the attitude matrix B transforms from body-fixed to earth-fixed coordinates. In JEOD, the B matrix transforms from inertial to body-fixed coordinates, so a change from Gottlieb's derivation in equation 3.6 was needed. Details of the derivation of the model are shown in [2] and are not repeated here. Details of the design in the Gravity Torque Model can be found in JEOD Gravity Torque Model Reference Manual [1].

Chapter 4

User Guide

The Analysis section of the user guide is intended primarily for users of pre-existing simulations. It contains:

- A description of how to modify Gravity Torque Model variables after the simulation has compiled, including an in-depth discussion of the input file,
- An overview of how to interpret (but not edit) the S_define file,
- A sample of some of the typical variables that may be logged.

The Integration section of the user guide is intended for simulation developers. It describes the necessary configuration of the Gravity Torque Model within an S_define file, and the creation of standard run directories. The latter component assumes a thorough understanding of the preceding Analysis section of the user guide. Where applicable, the user may be directed to selected portions of Product Specification (Chapter 3).

The Extension section of the user guide is intended primarily for developers needing to extend the capability of the Gravity Torque Model. Such users should have a thorough understanding of how the model is used in the preceding Integration section, and of the model specification (described in Chapter 3).

4.1 Analysis

4.1.1 Using the Gravity Torque Model

The mass properties of a spacecraft must first be defined in the form of an inertia tensor (matrix). An example of configuring the inertia tensor of a spacecraft (sim object) named VEH_OBJ is:

```
VEH_OBJ.mass_init.properties.mass              {kg} =  400000.0;
VEH_OBJ.mass_init.properties.position[0]        {M} = -3.0, -1.5, 4.0;
VEH_OBJ.mass_init.properties.inertia[0][0] {kg*M2} =  1.02e+8,-6.96e+6,-5.48e+6;
VEH_OBJ.mass_init.properties.inertia[1][0] {kg*M2} = -6.96e+6, 0.91e+8, 5.90e+5;
VEH_OBJ.mass_init.properties.inertia[2][0] {kg*M2} = -5.48e+6, 5.90e+5, 1.64e+8;
```

In order to compute gravity torque, the *gradient* flag must be set in the gravity controls for each spacecraft and each gravitational body (i.e., planet) desired. Setting the gradient flag to "false" for a particular planet will result in the null matrix, thus removing that planet's contribution to the overall gravitational torque. An example of setting this flag to true for a spacecraft called VEH_OBJ for the planet Earth is:

```
VEH_OBJ.grav_controls[GC_EARTH].gradient        = true;
VEH_OBJ.grav_controls[GC_EARTH].gradient_degree = 5;
VEH_OBJ.grav_controls[GC_EARTH].gradient_order  = 5;
```

The degree and order of the gravity gradient torque can be set independently of the acceleration degree and order for each gravitational body for each spacecraft. The gradient degree and order must be less than or equal to the acceleration degree and order. In the example above, the gradient degree and order are both 5. If the gradient degree and order are not specified, JEOD will compute the spherical body gradient as a default. Spherical body gradient can also be invoked by setting the gradient degree and order both to 0.

An example of setting the Earth gravity accleration degree to 70 and the order to 70 for a single spacecraft is:

```
VEH_OBJ.grav_controls[GC_EARTH].degree    = 70;
VEH_OBJ.grav_controls[GC_EARTH].order     = 70;
```

Any gravitational body can be modeled as a perfect sphere to simplify and speed up gravity computations. Setting the gravity field to *spherical* will also cause the gravity gradient to be based on the gravity of a spherical body. An example of setting a gravitational body to spherical is:

```
VEH_OBJ.grav_controls[GC_EARTH].spherical = true;
```

If a body is set to spherical, the gravity degree and order are forced to be zero (by definition). As previously mentioned, spherical body gradient can also be invoked by setting the gradient degree and order both to 0.

4.1.2 How to Interpret the S_define File

Each spacecraft (sim object) requires its own structure for computing gravity torque. This structure will be of type "GravityTorque" and declared in the S_define file. The gravity torque function is initialized with a call to the function grad_torque.initialize. At the desired job cycle rate, the gravity torque is computed (updated) by a call to the function grad_torque.update. The resulting torque is accumulated by a vcollect statement in the S_define file.

4.1.3 Data Logging

For logging the gravity torque acting on any spacecraft, include a line similar to this in the log setup file that is used to log forces and torques:

```
"#(VEH_OBJ).grav_torque.torque[0-2]",
```

The exact name of the variable to be logged will depend on the structure declaration for each spacecraft. Details can be found in the next section.

4.2 Integration

In the S_define file, for each vehicle object, include this structure:

```
interactions/gravity_torque: GravityTorque grav_torque;
```

Include this initialization job (sv_dyn is the vehicle sim object in this example):

```
(initialization) interactions/gravity_torque:
   sv_dyn.grad_torque.initialize( In DynBody & subject = sv_dyn.body );
```

Include this derivative class job:

```
P_BODY Idynamics (derivative) interactions/gravity_torque:
   sv_dyn.grav_torque.update();
```

Include grad_torque in the non-transmitted torque collection for each vehicle. An example that also includes aerodynamic torque is:

```
vcollect sv_dyn.body.collect.collect_no_xmit_torq CollectTorque::create {
   sv_dyn.aero_drag.aero_torque,
   sv_dyn.grav_torque.torque
};
```

Gravity gradient must be made active in the gravity controls for each vehicle (and each gravitational body) in the simulation input file or in a modified data file. The gradient degree and order can be specified separately from the acceleration degree and order. An example is shown below. If not specified, the default gravity gradient will be that due to a spherical body.

```
/* Configure gravity gradient torque. */
VEH_OBJ.grav_torque.active = true;

/* Set up the gravity controls for the Earth. */
VEH_OBJ.grav_controls[0].source_name     = "Earth";
VEH_OBJ.grav_controls[0].active          = true;
VEH_OBJ.grav_controls[0].spherical       = false;
VEH_OBJ.grav_controls[0].degree          = 36;
VEH_OBJ.grav_controls[0].order           = 36;
VEH_OBJ.grav_controls[0].gradient        = true;
VEH_OBJ.grav_controls[0].gradient_degree = 5;
VEH_OBJ.grav_controls[0].gradient_order  = 5;
```

The last three lines in this set activate the gravity gradient and set the degree and order both to 5. Setting the gradient flag to “false” will result in the null matrix being computed for the gravity gradient.

4.3 Extension

The Gravity Torque Model is a function based directly on first principles of physics/dynamics. Therefore, the model is not extensible.

Chapter 5

Verification and Validation

5.1 Verification

The Gravity Torque Model was verified for the spherical body case by comparing the model results with the analytical results for spherical body gravity gradient torque. The analytical expression for spherical body torque expressed in the spacecraft body frame is[7]

$$\bar{\tau} = \frac{3\mu}{r^5} \left(\bar{r} \times I\bar{r} \right) \tag{5.1}$$

where I is the inertia matrix in the spacecraft body frame, and the vector $\bar{r}$ has been transformed into the spacecraft body frame.

Six separate runs of the SIM_torque_compare_simple simulation of a low Earth orbiting satellite were executed to verify the functionality of the Gravity Torque Model. In the first run (RUN_01), the Earth was specified to be a spherical body and the gravity gradient was *not* activated using these lines in the input file:

```
/* Set up the gravity controls for the Earth. */
VEH_OBJ.grav_controls[0].source_name     = "Earth";
VEH_OBJ.grav_controls[0].active          = true;
VEH_OBJ.grav_controls[0].spherical       = true;
VEH_OBJ.grav_controls[0].degree          = 20;
VEH_OBJ.grav_controls[0].order           = 20;
VEH_OBJ.grav_controls[0].gradient        = false;
VEH_OBJ.grav_controls[0].gradient_degree = 0;
VEH_OBJ.grav_controls[0].gradient_order  = 0;
```

With the gradient set to false, the Gravity Torque Model computed zero torque. The results are shown in the figure 5.1, where the "benchmark" curve is a plot of the analytical results of equation 5.1. As expected, the computed torque was zero in all three body frame directions.

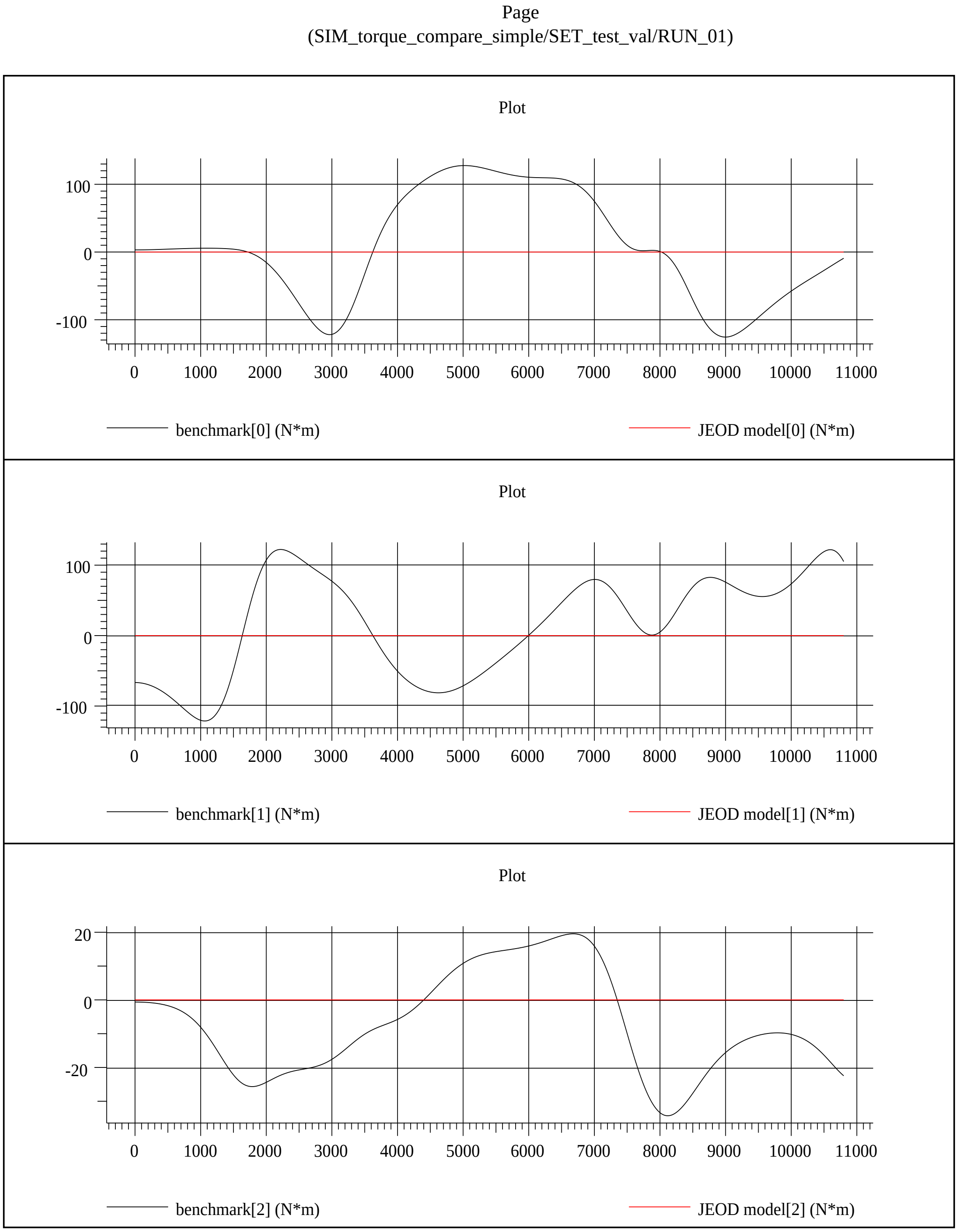

Figure 5.1: RUN_01 results.

In the second sim run (RUN_02), the Earth was specified to be a spherical body and the gravity gradient was activated using these lines in the input file:

```
/* Set up the gravity controls for the Earth. */
VEH_OBJ.grav_controls[0].source_name     = "Earth";
VEH_OBJ.grav_controls[0].active          = true;
VEH_OBJ.grav_controls[0].spherical       = true;
VEH_OBJ.grav_controls[0].degree          = 20;
VEH_OBJ.grav_controls[0].order           = 20;
VEH_OBJ.grav_controls[0].gradient        = true;
VEH_OBJ.grav_controls[0].gradient_degree = 0;
VEH_OBJ.grav_controls[0].gradient_order  = 0;
```

The results are shown in the figure 5.2. Again, the "benchmark" curve is a plot of the analytical results of equation 5.1. As expected, the computed torque was equal to the analytical spherical body torque in all three body frame directions.

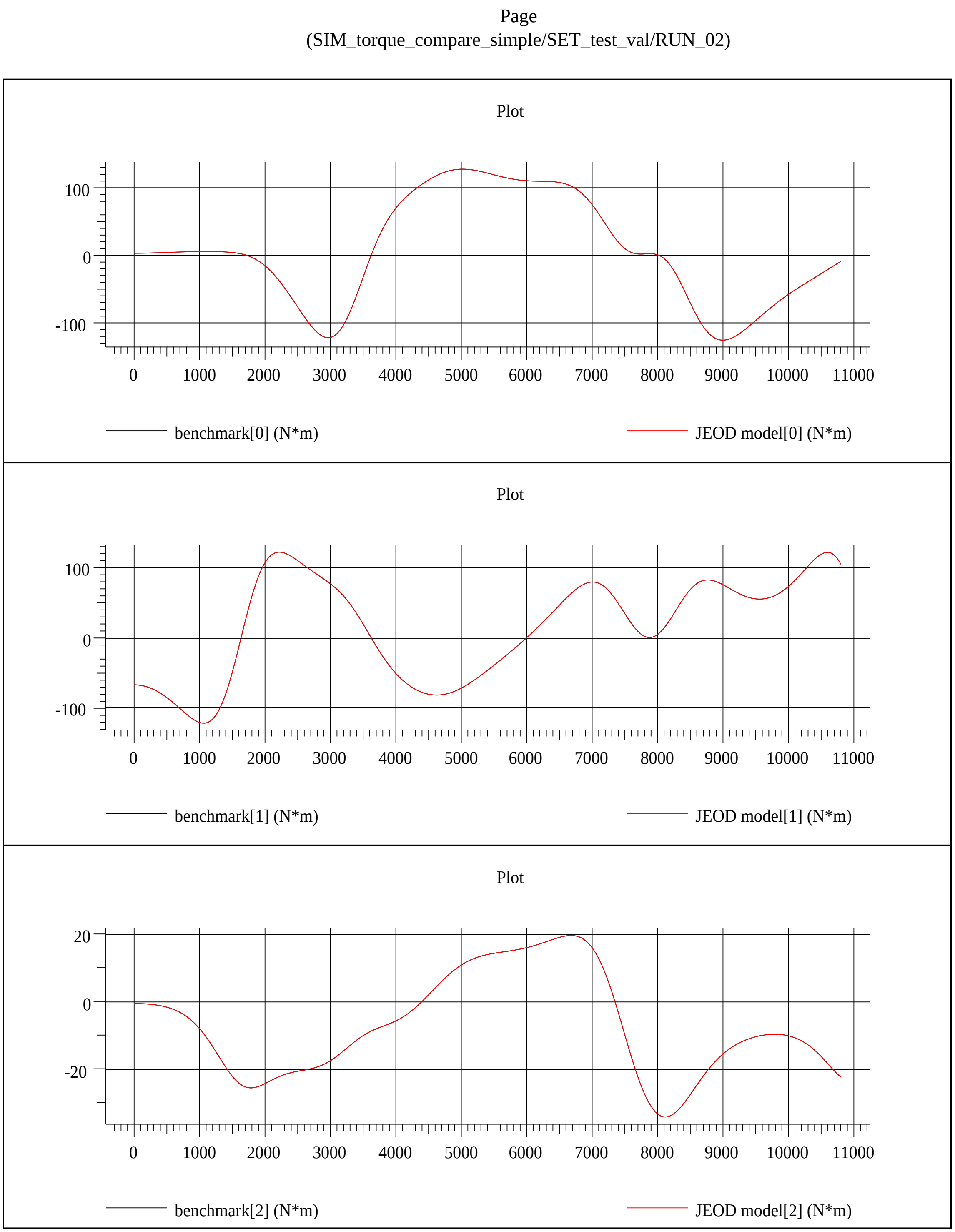

Figure 5.2: RUN_02 results.

In the third sim run (RUN_03), the Earth was was specified to be a spherical body and the gravity gradient was activated. The gravity gradient degree and order were both specified to be 4 using these lines in the input file:

```
/* Set up the gravity controls for the Earth. */
VEH_OBJ.grav_controls[0].source_name     = "Earth";
VEH_OBJ.grav_controls[0].active          = true;
VEH_OBJ.grav_controls[0].spherical       = true;
VEH_OBJ.grav_controls[0].degree          = 20;
VEH_OBJ.grav_controls[0].order           = 20;
VEH_OBJ.grav_controls[0].gradient        = true;
VEH_OBJ.grav_controls[0].gradient_degree = 4;
VEH_OBJ.grav_controls[0].gradient_order  = 4;
```

The results are shown in the figure 5.3. Even though the gradient degree and order were specified for a non-spherical body, the spherical gradient was computed because the Earth was specified to be a spherical body. The resulting torque was equal to the analytical spherical body torque in all three body frame directions.

Page
(SIM_torque_compare_simple/SET_test_val/RUN_03)

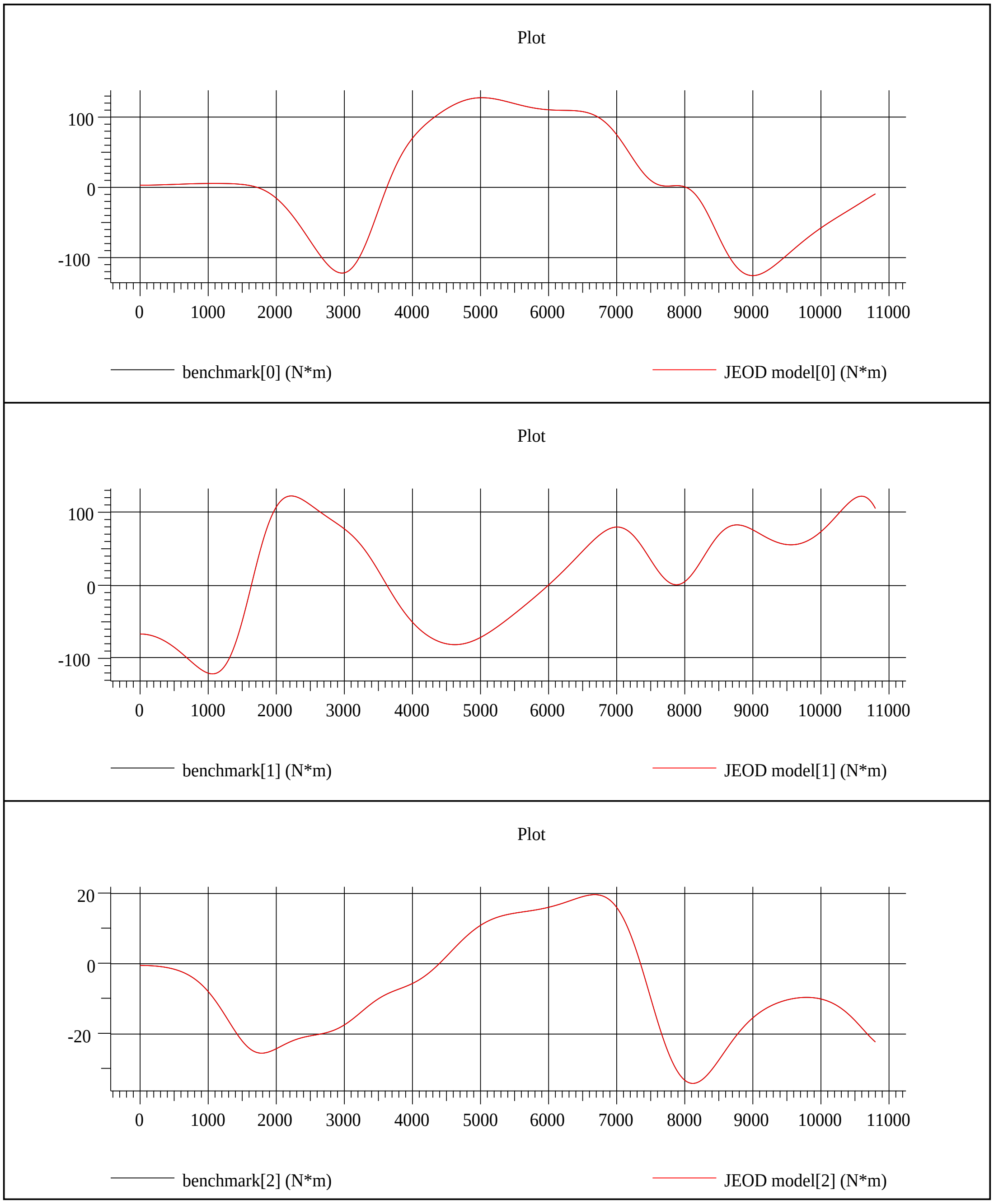

Figure 5.3: RUN_03 results.

In the fourth sim run (RUN_04), the Earth was was specified to be a non-spherical body with gravity acceleration degree and order 20. The gravity gradient was *not* activated. The input file lines are:

```
/* Set up the gravity controls for the Earth. */
VEH_OBJ.grav_controls[0].source_name     = "Earth";
VEH_OBJ.grav_controls[0].active          = true;
VEH_OBJ.grav_controls[0].spherical       = false;
VEH_OBJ.grav_controls[0].degree          = 20;
VEH_OBJ.grav_controls[0].order           = 20;
VEH_OBJ.grav_controls[0].gradient        = false;
VEH_OBJ.grav_controls[0].gradient_degree = 0;
VEH_OBJ.grav_controls[0].gradient_order  = 0;
```

The results are shown in the figure 5.4. Because the gradient was not activated, the resulting torque was computed to be zero in all three body frame directions regardless of the fact that the Earth was specified to be non-spherical.

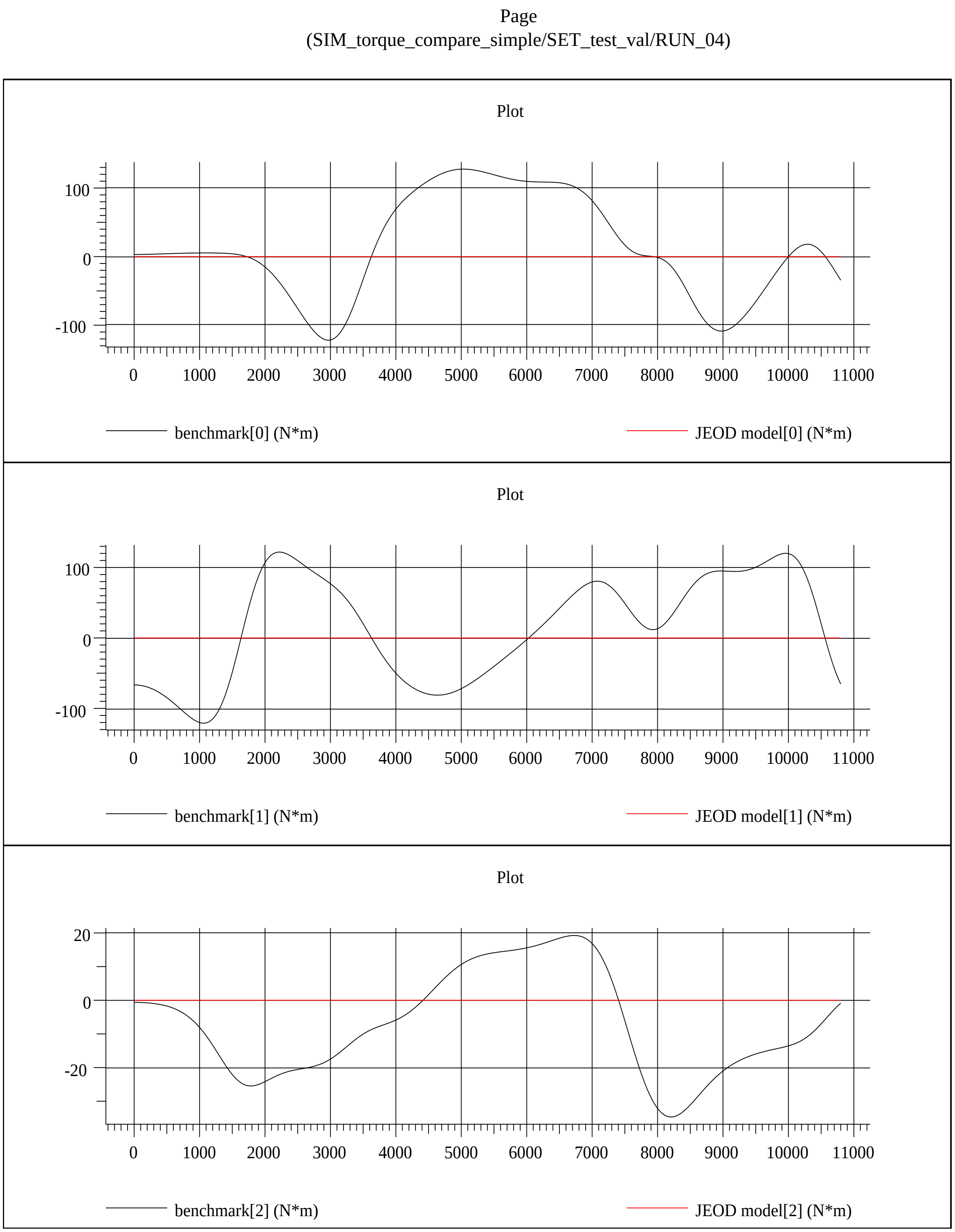

Figure 5.4: RUN_04 results.

In the fifth sim run (RUN_05), the Earth was specified to be a non-spherical body with gravity acceleration degree and order 20. The gravity gradient was activated, and the degree and order were both specified to be 0, invoking spherical gravity gradient. The input file lines are:

```
/* Set up the gravity controls for the Earth. */
VEH_OBJ.grav_controls[0].source_name     = "Earth";
VEH_OBJ.grav_controls[0].active          = true;
VEH_OBJ.grav_controls[0].spherical       = false;
VEH_OBJ.grav_controls[0].degree          = 20;
VEH_OBJ.grav_controls[0].order           = 20;
VEH_OBJ.grav_controls[0].gradient        = true;
VEH_OBJ.grav_controls[0].gradient_degree = 0;
VEH_OBJ.grav_controls[0].gradient_order  = 0;
```

The results are shown in the figure 5.5. With the gradient activated and set for spherical gradient, the resulting torque was identical to the analytical solution in all three body frame directions.

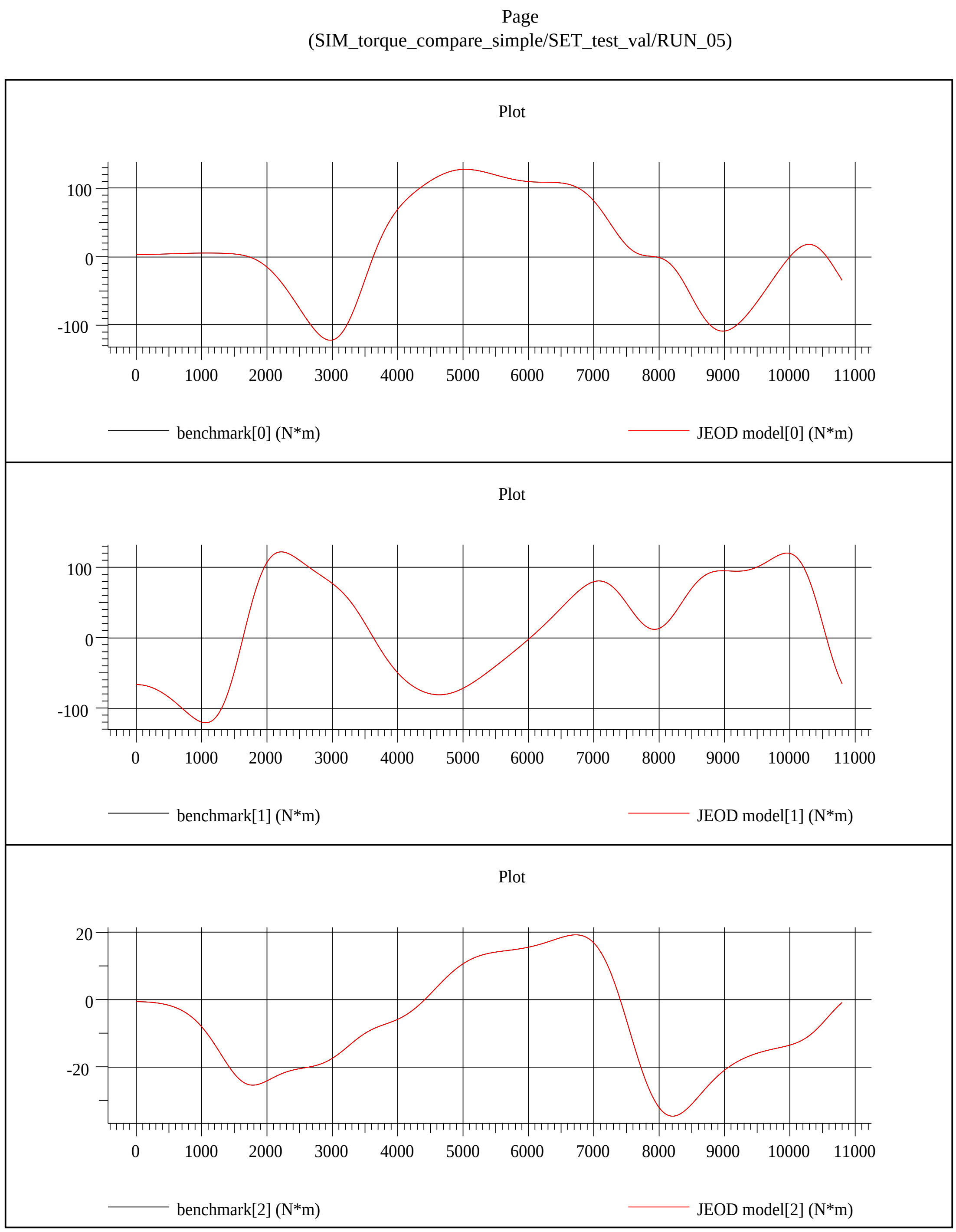

Figure 5.5: RUN_05 results.

Finally, in the sixth sim run (RUN_06), the Earth was specified to be a non-spherical body with gravity acceleration degree and order 20. The gravity gradient was activated, and the degree and order were both specified to be 4, invoking non-spherical gravity gradient. The input file lines are:

```
/* Set up the gravity controls for the Earth. */
VEH_OBJ.grav_controls[0].source_name     = "Earth";
VEH_OBJ.grav_controls[0].active          = true;
VEH_OBJ.grav_controls[0].spherical       = false;
VEH_OBJ.grav_controls[0].degree          = 20;
VEH_OBJ.grav_controls[0].order           = 20;
VEH_OBJ.grav_controls[0].gradient        = true;
VEH_OBJ.grav_controls[0].gradient_degree = 4;
VEH_OBJ.grav_controls[0].gradient_order  = 4;
```

A detailed sample of the results are shown in the figure 5.6. With the gradient activated and set for non-spherical gradient, the resulting torque was slightly different from the analytical spherical body torque in all three body frame directions. Whether or not the computed torque was the correct value was not addressed by this test. The numerical correctness of the computations is the subject of the next section, Validation.

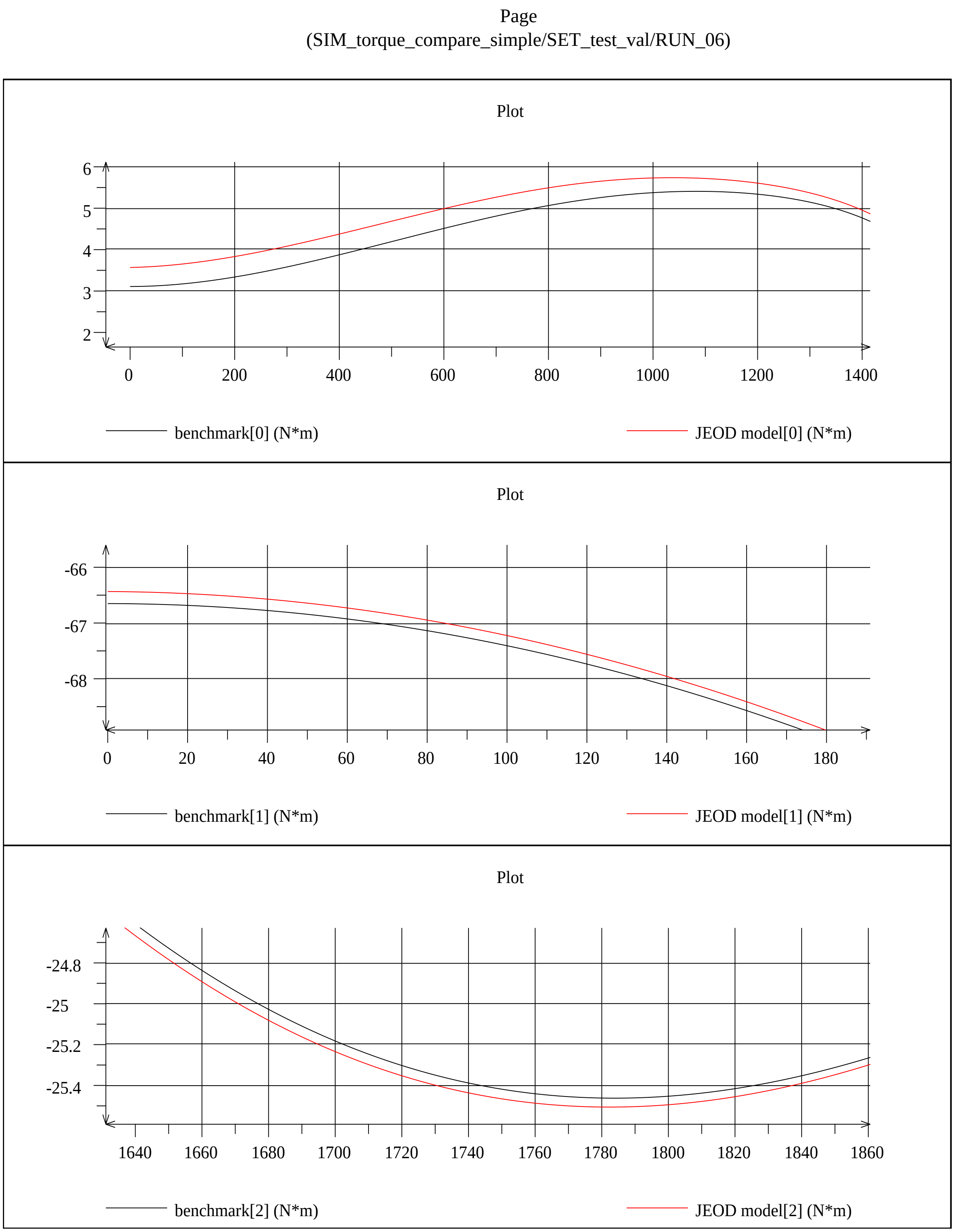

Figure 5.6: RUN_06 results (zoomed in for detail).

5.2 Validation

To validate the Gravity Torque Model, a fictitious "planet" was constructed from a set of rigid point masses at known locations. The spherical harmonic gravity coefficients representing the point mass system were computed and used to compute the gravity gradient by the JEOD Gravity Model. The gravity gradient was used as input to the Gravity Torque Model. Because the mass and location of each point of the point mass system was known, the torque acting on the spacecraft due to the gravity of the point masses was directly computed from first principles of physics. The results of this direct computation were compared to the results of the Gravity Torque Model for validation of the model. See Thompson *et al.* for details[8].

Point Mass System

A system of 12 point masses was configured by experimentation. The total mass of the system was approximately equal to the mass of the Earth roughly divided among all of the points (see table 5.1). The first six point masses were chosen to result in a center of mass at the origin and equal mass in the northern and southern hemispheres. The north/south symmetry of mass caused the odd zonal ($m = 0$) coefficients to be zero. This was unsuitable for evaluation of the spherical harmonic algorithms, so the final six point masses were included such that more mass was positioned in the southern hemisphere (similar to Earth). These points were added to the system in pairs such that the center of mass remained at the origin of the coordinate system in spite of the unequal north/south mass distribution. In terms of radial distance from the origin, the center of mass of two points is

$$c.o.m = \frac{r_1 m_1 + r_2 m_2}{m_1 + m_2} \tag{5.2}$$

This assumes the masses are situated in diametrically opposing locations with respect to the origin. Setting equation 5.2 to zero and solving for r_2 yields

$$r_2 = \frac{m_1}{m_2} r_1 \tag{5.3}$$

This is the radial position of the second point mass in each pair that kept the center of mass at the origin.

Table 5.1 shows the 12 point masses that were selected. This system resulted in degree-one gravity coefficients on the order of $\sim 10^{-20}$, indicating that the center of mass of the system was very nearly located at the origin. Therefore, the degree-one terms were considered negligible and forced to be exactly zero for all computations. This was necessary because many spherical harmonic algorithms make this assumption and begin computing with the degree-two terms.

Test gravitytorque_1: Gravity Torque Test

To validate the Gravity Torque Model a fictitious spacecraft model consisting of three point masses was developed. The point masses and locations in the body fixed frame are shown in Table 5.2. The equivalent inertia tensor was

$$I = \begin{bmatrix} 630.00 & -277.50 & 372.50 \\ -277.50 & 773.75 & 285.00 \\ 372.50 & 285.00 & 573.75 \end{bmatrix} \tag{5.4}$$

Table 5.1: Point mass model of the gravitational body.

Point	Mass(kg)	Lat. (deg)	Lon. (deg)	Radius (m)
1	$M_\oplus/12$	45.0	0.0	4000.0
2	$M_\oplus/12$	45.0	120.0	4000.0
3	$M_\oplus/12$	45.0	240.0	4000.0
4	$M_\oplus/12$	-45.0	180.0	4000.0
5	$M_\oplus/12$	-45.0	300.0	4000.0
6	$M_\oplus/12$	-45.0	60.0	4000.0
7	$0.8M_\oplus/12$	23.0	73.0	4000.0
8	$1.2M_\oplus/12$	-23.0	253.0	(0.8/1.2)4000.0
9	$0.6M_\oplus/12$	77.0	303.0	4000.0
10	$1.4M_\oplus/12$	-77.0	123.0	(0.6/1.4)4000.0
11	$0.6M_\oplus/12$	51.0	12.0	4000.0
12	$1.4M_\oplus/12$	-51.0	192.0	(0.6/1.4)4000.0

Table 5.2: Point masses and locations of the spacecraft model.

Point	Mass(kg)	x (m)	y (m)	x (m)
1	50.0	1.625	1.250	-1.750
2	20.0	-2.375	-1.750	3.250
3	10.0	-3.375	-2.750	2.250

Table 5.3: Test point locations.

Test Point	Lat. (deg)	Lon. (deg)	Radius (m)
1	90.00	120.00	6800000.0
2	89.99	353.00	6800000.0
3	73.00	9.00	6800000.0
4	13.00	100.00	6800000.0
5	0.00	0.00	6800000.0
6	0.00	180.00	6800000.0
7	−26.00	210.00	6800000.0
8	−52.00	310.00	6800000.0
9	−89.99	10.00	6800000.0
10	−90.00	160.00	6800000.0

For validation purposes, a pitch-yaw-roll (2-3-1) sequence was assumed. Attitude angles of pitch = 20 deg, yaw = 30 deg, roll = 40 deg were used for all test cases shown in table 5.3.

Results

A set of test points was selected somewhat randomly to compare the acceleration computed by the spherical harmonic algorithms to the point mass acceleration. All test points were selected to have a radial distance of 6800000.0 meters, approximating the altitude of a typical low-earth orbiting satellite.

The results in table 5.4 show by-component differences typically on the order of six or more orders of magnitude smaller than the torque magnitude. This was considered acceptable in light of the linearity approximations used in deriving the torque model (see Gottlieb [2] p.19).

Table 5.4: Results comparing torque vectors computed by the spherical harmonic algorithm vs. torque vectors computed directly from the point mass planet.

Test Point	Sph. Harm. Torque	Pt. Mass Torque	Difference
1	-6.576995906413200E-04	-6.576998403033940E-04	2.5E-10
	1.269164209592045E-04	1.269165028503494E-04	8.2E-11
	-4.723482089916854E-04	-4.723484289002045E-04	2.2E-10
2	-6.577475179730652E-04	-6.577477676614763E-04	2.5E-10
	1.270487433086179E-04	1.270488252202995E-04	8.2E-11
	-4.722975646974573E-04	-4.722977844267007E-04	2.2E-10
3	-4.667054619988188E-04	-4.667057505685079E-04	2.9E-10
	2.230918182780193E-04	2.230919879480098E-04	1.7E-10
	-2.308991703765578E-04	-2.308993437338813E-04	1.7E-10
4	9.643230821702463E-04	9.643234462686223E-04	3.6E-10
	3.894826293977236E-04	3.894828155921459E-04	1.9E-10
	1.192559213912787E-03	1.192559729815912E-03	5.2E-10
5	5.947635780445906E-05	5.947628430647001E-05	7.3E-11
	-1.824208387207120E-04	-1.824204566105436E-04	3.8E-10
	-1.311220703920095E-04	-1.311218693729188E-04	2.0E-10
6	5.947635779717441E-05	5.947643117565349E-05	7.3E-11
	-1.824208387253428E-04	-1.824212203018760E-04	3.8E-10
	-1.311220704109403E-04	-1.311222710853599E-04	2.0E-10
7	7.585422706336786E-04	7.585422953582111E-04	2.5E-11
	-1.035277845536193E-03	-1.035277890196085E-03	4.5E-11
	-5.060880799457042E-05	-5.060879752250003E-05	1.0E-11
8	5.494210701133441E-04	5.494212856262948E-04	2.2E-10
	1.890337404179465E-04	1.890338145713599E-04	7.4E-11
	6.827228269632132E-04	6.827230673707163E-04	2.4E-10
9	-6.577823003575999E-04	-6.577820505526688E-04	2.5E-10
	1.268171069274990E-04	1.268170250483536E-04	8.2E-11
	-4.724987427239774E-04	-4.724985225834644E-04	2.2E-10
10	-6.576995906610781E-04	-6.576993405928988E-04	2.5E-10
	1.269164210444634E-04	1.269163389849837E-04	8.2E-11
	-4.723482089531574E-04	-4.723479888752991E-04	2.2E-10

5.3 Requirements Traceability

The table below cross-references each requirement of the Gravity Torque Model to a corresponding verification and/or validation test described in sections 5.1 and 5.2 of this document.

Table 5.5: Requirements Traceability

Requirement	**Inspection or Test**
gravitytorque_1 - Gravity Gradient Matrix	Test gravitytorque_1 - Gravity Torque Test
gravitytorque_2 - Spacecraft Mass Properties	Test gravitytorque_1 - Gravity Torque Test
gravitytorque_3 - Spacecraft Attitude	Test gravitytorque_1 - Gravity Torque Test
gravitytorque_4 - Gravity Torque	Test gravitytorque_1 - Gravity Torque Test

Bibliography

[1] generated by doxygen. *JEOD Gravity Torque Model Reference Manual*. NASA, Johnson Space Center, Software, Robotics & Simulation Division, Simulation and Graphics Branch, 2101 NASA Parkway, Houston, Texas, 77058, September 2024.

[2] Gottlieb, R.G. Fast Gravity, Gravity Partials, Normalized Gravity, Gravity Gradient Torque and Magnetic Field: Derivation, Code and Data. Technical Report NASA Contractor Report 188243 (JSC 23762), National Aeronautics and Space Administration, Johnson Space Center, Houston, Texas, February 1993.

[3] Jackson, A., Thebeau, C. JSC Engineering Orbital Dynamics. Technical Report JSC-61777-docs, NASA, Johnson Space Center, Houston, Texas, September 2024.

[4] M.H. Kaplan. *Modern Spacecraft Dynamics and Control*. John Wiley and Sons, New York, 1976.

[5] W.M. Kaula. *Theory of Satellite Geodesy: Applications of Satellites to Geodesy*. Blaisdell Publishing Company, Waltham, Massachusetts, 1966.

[6] NASA. NASA Software Engineering Requirements. Technical Report NPR-7150.2, NASA, NASA Headquarters, Washington, D.C., September 2004.

[7] Schaub, H. and Junkins, J. *Analytical Mechanics of Space Systems*. American Institute of Aeronautics and Astronautics, Inc., Reston, Virginia, 2003.

[8] Thompson, B.F., Hammen D.G., Jackson, A.A., Crues, E.Z. Validation of Gravity Acceleration and Torque Algorithms for Astrodynamics. *Proceedings of the 18th AAS/AIAA Space Flight Mechanics Meeting*, Galveston, Texas, Jan. 2008, pp.1593-1604.