Autonomous Vehicle Simulation (AVS) Laboratory, University of Colorado

Basilisk Technical Memorandum

Document ID: Basilisk-MagneticField

MAGNETIC FIELD ENVIRONMENT

Prepared by	H. Schaub

Status: Released

Scope/Contents

The `MagneticFieldCenteredDipole` class is used to calculate the magnetic field vector above a body using the centered dipole model. This class is used to hold relevant planetary magnetic field properties to compute answers for a given set of spacecraft locations relative to a specified planet. Planetary parameters, including position and input message, are settable by the user. In a given simulation, each planet of interest should have only one magnetic field model associated with it linked to the spacecraft in orbit about that body.

Rev	**Change Description**	**By**	**Date**
1.0	Initial release	H. Schaub	03-12-2019
1.1	updated with new magnetic field base class info	H. Schaub	03-21-2019

Contents

1 Model Description

1.1 General Module Behavior

The purpose of this module is to implement a magnetic field model that rotates with a planet fixed frame $\mathcal{P}: \{\hat{\boldsymbol{p}}_1, \hat{\boldsymbol{p}}_2, \hat{\boldsymbol{p}}_3\}$. Here $\hat{\boldsymbol{p}}_3$ is the typical positive rotation axis and $\hat{\boldsymbol{p}}_1$ and $\hat{\boldsymbol{p}}_2$ span the planet's equatorial plane.

`MagneticFieldCenteredDipole` is a child of the `MagneticFieldBase` base class and provides a simple centered dipole magnetic field model. By invoking the magnetic field module, the default values are set such that the dipole parameters are zeroed and the magnetic field output is a zero vector. The reach of the model controlled by setting the variables `envMinReach` and `envMaxReach` to positive values. These values are the radial distance from the planet center. The default values are -1 which turns off this checking where the atmosphere model as unbounded reach.

There are a multitude of magnetic field models.* The goal with Basilisk is to provide a simple and consistent interface to a range of models. The list of models is expected to grow over time.

1.2 Planet Centric Spacecraft Position Vector

For the following developments, the spacecraft location relative to the planet frame is required. Let $\boldsymbol{r}_{B/P}$ be the spacecraft position vector relative to the planet center. In the simulation the spacecraft location is given relative to an inertial frame origin O. The planet centric position vector is computed using

$$\boldsymbol{r}_{B/P} = \boldsymbol{r}_{B/O} - \boldsymbol{r}_{P/O} \tag{1}$$

---

* https://geomag.colorado.edu/geomagnetic-and-electric-field-models.html

If no planet ephemeris message is specified, then the planet position vector $\boldsymbol{r}_{P/O}$ is set to zero.

Let $[PN]$ be the direction cosine matrix[1] that relates the rotating planet-fixed frame relative to an inertial frame $\mathcal{N} : \{\hat{\boldsymbol{n}}_1, \hat{\boldsymbol{n}}_2, \hat{\boldsymbol{n}}_3\}$. The simulation provides the spacecraft position vector in inertial frame components. The planet centric position vector is then written in Earth-fixed frame components using

$$ {}^{\mathcal{P}}\boldsymbol{r}_{B/P} = [PN]\, {}^{\mathcal{N}}\boldsymbol{r}_{B/P} \tag{2}$$

1.3 Centered Dipole Magnetic Field Model

The centered dipole model is a first order result of the more complex spherical harmonic modeling of the planet's magnetic field.[2] There are several solutions that provide an answer in the local North-Earth-Down or NED frame,[3] or in the local spherical coordinates. Let $\boldsymbol{m}$ be the magnetic dipole vector which is then defined as[2]

$$ V(\boldsymbol{r}_{B/P}) = \frac{\boldsymbol{m}\cdot\boldsymbol{r}_{B/P}}{|\boldsymbol{r}_{B/P}|^3} \tag{3}$$

with the dipole vector being defined in Earth fixed frame $\mathcal{E}$ coordinates as

$$ \boldsymbol{m} = {}^{\mathcal{E}}\begin{bmatrix} g_1^1 \\ h_1^1 \\ g_1^0 \end{bmatrix} \tag{4}$$

The magnetic field vector $\boldsymbol{B}$ is expressed at the spacecraft location as

$$ \boldsymbol{B}(\boldsymbol{r}_{B/P}) = -\nabla V(\boldsymbol{r}_{B/P}) = \frac{3(\boldsymbol{m}\cdot\boldsymbol{r}_{B/P})\boldsymbol{r}_{B/P} - \boldsymbol{r}_{B/P}\cdot\boldsymbol{r}_{B/P}\boldsymbol{m}}{|\boldsymbol{r}_{B/P}|^5} = \frac{1}{|\boldsymbol{r}_{B/P}|^3}\left(3(\boldsymbol{m}\cdot\hat{\boldsymbol{r}})\hat{\boldsymbol{r}} - \boldsymbol{m}\right) \tag{5}$$

where

$$ \hat{\boldsymbol{r}} = \frac{\boldsymbol{r}_{B/P}}{|\boldsymbol{r}_{B/P}|} \tag{6}$$

The above vector equation is evaluated in $\mathcal{E}$-frame components, while the output is mapped into $\mathcal{N}$-frame components by returning

$$ {}^{\mathcal{N}}\boldsymbol{B} = [PN]^T\, {}^{\mathcal{P}}\boldsymbol{B} \tag{7}$$

2 Module Functions

This module will:

- **Compute magnetic field vector**: Each of the provided models is fundamentally intended to compute the planetary magnetic vector for a spacecraft.
- **Subscribe to model-relevant information:** Each provided magnetic field model requires different input information to operate, such as spacecraft positions or time. This module automatically attempts to subscribe to the relevant messages for a specified model.
- **Support for multiple spacecraft and model types** Only one magnetic field module is required for each planet, and can support an arbitrary number of spacecraft. Output messages for individual spacecraft are automatically named based on the environment type.

3 Module Assumptions and Limitations

Individual magnetic field models are complex and have their own assumptions. The reader is referred to the cited literature to learn more about the model limitations and challenges.

4 Test Description and Success Criteria

This section describes the specific unit tests conducted on this module. This unit test only runs the magnetic field Basilisk module with two fixed spacecraft state input messages. The simulation option `useDefault` checks if the module default settings are used that lead to a zero magnetic field vector, or if the centered dipole parameters are setup manually. The option `useMinReach` dictates if the minimum orbit radius check is performed, while the option `useMaxReach` checks if the maximum reach check is performed. The option `usePlanetEphemeris` checks if a planet state input message should be created. All permutations are checked.

5 Test Parameters

The simulation tolerances are shown in Table 2. In each simulation the neutral density output message is checked relative to python computed true values.

Table 2: Error tolerance for each test.

Output Value Tested	**Tolerated Error**
`magneticField vector`	1e-05 (relative)

6 Test Results

The following two tables show the test results. All tests are expected to pass.

Table 3: Test result for `test_unitTestMagneticField.py`

`useDefault`	`useMinReach`	`useMaxReach`	`usePlanetEphemeris`	**Pass/Fail**
False	False	False	False	PASSED
False	False	False	True	PASSED
False	False	True	False	PASSED
False	False	True	True	PASSED
False	True	False	False	PASSED
False	True	False	True	PASSED
False	True	True	False	PASSED
False	True	True	True	PASSED
True	False	False	False	PASSED
True	False	False	True	PASSED
True	False	True	False	PASSED
True	False	True	True	PASSED
True	True	False	False	PASSED
True	True	False	True	PASSED
True	True	True	False	PASSED
True	True	True	True	PASSED

7 User Guide

7.1 General Module Setup

This section outlines the steps needed to add a `MagneticField` module to a sim. First, the planet magnetic field model must be imported and initialized:

```
from Basilisk.simulation import magneticFieldCenteredDipole
magModule = magneticFieldCenteredDipole.MagneticFieldCenteredDipole()
magModule = "CenteredDipole"
```

By default the model the dipole parameters are zeroed, resulting in a zeroed magnetic field message.

The model can be added to a task like other simModels.

```
unitTestSim.AddModelToTask(unitTaskName, testModule)
```

Each `MagneticField` module calculates the magnetic field based on the output state messages for a set of spacecraft. To add spacecraft to the model the spacecraft state output message name is sent to the `addScToModel` method:

```
scObject = spacecraft.Spacecraft()
scObject.ModelTag = "spacecraftBody"
magModule.addSpacecraftToModel(scObject.scStateOutMsg)
```

7.2 Planet Ephemeris Information

The optional planet state message name can be set by directly adjusting that attribute of the class:

```
magModule.planetPosInMsg.subscribeTo(planetMsg)
```

If SPICE is not being used, the planet is assumed to reside at the origin and $\boldsymbol{r}_{P/O} = \mathbf{0}$.

7.3 Setting the Model Reach

By default the model doesn't perform any checks on the altitude to see if the specified magnetic field model should be used. This is set through the parameters `envMinReach` and `envMaxReach`. Their default values are -1. If these are set to positive values, then if the spacecraft orbit radius is smaller than `envMinReach` or larger than `envMaxReach`, the magnetic field vector is set to zero.

7.4 Centered Dipole Magnetic Parameters

The parameters of the dipole model are set by calling

```
    magModule.g10 = g10Value
    magModule.g11 = g11Value
    magModule.h11 = h11Value
    magModule.planetRadius = planetRadius # in meters
```

where g_1^0, g_1^1 and h_1^1 are the first three expansion terms of the IGRF spherical harmonics model.[*]

The python support file `simSetPlanetEnvironment.py` contains a helper function called

```
centeredDipoleMagField()
```

which helps setup common NASA centered dipole models for a range of planets that contain global magnetic fields. This possible planet names includes mercury, earth, jupiter, saturn, uranus and neptune. The function is then called with

```
simSetPlanetEnvironment.centeredDipoleMagField(testModule, "jupiter")
```

to setup the named planets dipole model.

REFERENCES

[1] Hanspeter Schaub and John L. Junkins. *Analytical Mechanics of Space Systems*. AIAA Education Series, Reston, VA, 4th edition, 2018.

[2] F. Landis Markley and John L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Springer, New York, 2014.

[3] Michael D. Grifin and James R. French. *Space Vehicle Design*. AIAA Education Series, Reston, VA, 2005.

---

[*] `https://www.ngdc.noaa.gov/IAGA/vmod/igrf.html`