Autonomous Vehicle Simulation (AVS) Laboratory, University of Colorado

Basilisk Technical Memorandum

Document ID: Basilisk-RadiationPressure

RADIATION PRESSURE MODEL

Prepared by	S. Carnahan

Status: Initial document draft
Scope/Contents
The radiation pressure module is responsible for calculating the dyamic effects of radiation pressure on a spacecraft. Effects can be calculated either by use of a simplified "cannonball" method or table look-up. A unit test has been written which checks both calculation methods against expected output values.

Rev:	**Change Description**	**By**
Draft	Initial document creation	S. Carnahan

Contents

---

1 Introduction

The Basilisk radiation pressure module (radiation_pressure.cpp) is responsible for calculating the effects of radiation pressure on a spacecraft. The mathematical models used are described below. The code unit tests are then presented and discussed.

2 Mathematical Model

Maths are modeled.

2.1 Radiation Model

Radiation is modeled by...

2.2 Radiation Pressure Model

Radiation Pressure is modeled by...

2.3 Recursion Formulas

The above formulas for the radiation pressure are broken down into iterative steps in order to be modeled in the simulation. The recursive formulas follow.

2.4 Variable Definitions

The variables in Table 2 are available for user input. Variables used by the module but not available to the user are not mentioned here. Variables with default settings do not necessarily need to be changed by the user, but may be.

Table 2: Definition and Explanation of Variables Used.

Variable	LaTeX Equivalent	Variable Type	Notes
stateInMsgName	N/A	string	Default setting: "inertial_state_output". This is the message from which the radiation pressure module receives spacecraft inertial data.
sunEphmInMsgName	N/A	string	Default setting: "sun_planet_data". This is the message through which radiation pressure gets information about the sun and planets.
area	N/A yet	double	[m2] Default setting: 0.0f. Required input for cannonball method to get any real output. This is the area to use when approximating the surface area of the spacecraft.
coefficientReflection	N/A yet	double	Default setting: 1.2. This is a factor applied to the radiation pressure based on spacecraft surface properties.
useCannonballModel	N/A	bool	Default setting: True. To switch cannonball mode off, use the call setUseCannonballModel("False")

3 Library

This section means nothing to me.

4 Unit Test

This test is located at `SimCode/dynamics/RadiationPressure/_UnitTest/test_radiationPressure.py` In order to get good coverage of all the aspects of the module, the test is broken up into two sub-tests. In each sub-test, a spacecraft is placed in the solar system and acted upon by the Sun.

4.1 "Cannonball" Method

This test utilizes the "cannonball" method to calculate the effects of radiation pressure on spacecraft dynamics. The cannonball method approximates the spacecraft as a sphere. External forces in the inertial and body frame, as well as external torques in the body frame, are checked against known values.

4.2 Table Look-up Method

This test uses a stored table of known effects of radiation pressure. It looks up values and compares them to the expected result to validate radiation pressure table look-up capabilities.

4.3 Test Parameters

This section summarizes the test input/output for each of the checks.

- Error Tolerance

  There are specific error tolerances for each test. Error tolerances are determined based on whether the test results comparison should be exact or approximate due to integration or other reasons. Error tolerances for each test are summarized in table 3.

Table 3: Error tolerance for each test.

Test	Tolerated Error
"Cannonball"	1.0e-12
Look-up	1.0e-12

4.4 Test Results

All checks within test_radiationPressure.py passed as expected. Table 4 shows the test results.

Table 4: Test results.

Test	Pass/Fail	Notes
"Cannonball"	PASSED	
Look-up	PASSED	

4.5 Test Coverage

The test covers 2000% of the code.

5 Conclusion

What a great piece of code.