Autonomous Vehicle Simulation (AVS) Laboratory, University of Colorado

Basilisk Technical Memorandum

Document ID: Basilisk-keplerianOrbit

KEPLERIAN ORBIT C++ MODEL

Prepared by	S. Carnahan

Status: Tested
Scope/Contents
The Keplerian orbit class is an object oriented keplerian orbit. Given a planet and orbital elements it can provide a host of outputs including orbital period and position and velocity.

Rev:	**Change Description**	**By**	**Date**
1.0	First version - Mathematical formulation and implementation	G. Chapel	09/07/17

Contents

---

1 Model Description

This model provides a coherent set of data and functionality to represent a Keplerian orbit.

2 Model Functions

This module contains several functionalities described below

- **Interface: Orbit Elements** Classic Orbit elements are provided to define the orbit. Individual elements can be changed.
- **Interface: Outputs** Classic orbital properties like angular momenum and enegy can be access as functions using variable names common in such tests as Vallado. For example, specific orbital angular momentum is access via the member method `h()`.

3 Model Assumptions and Limitations

- **Limitation: Elliptical Orbits** In its current incarnation, this module supports only elliptical orbits. Future work should expand the class to include parabolic, hyperbolic, and circular orbits.

4 Test Description and Success Criteria

The unit test test_unitKeplerianOrbit.py tests the construction and calculations of the class. It specifically checks the mean motion, orbital period, and position and velocity outputs. The test PASSED
.

5 User Guide

This module was meticulously set up to ensure that all variables can only be set or retrieved by setter and getter functions. If the module is instantiated without arguments, the spacecraft is in some arbitrary orbit. Generally, it should be called with the first argument as the desired orbital elements and the second argument as a GravBodyData object.