SSC19-IX-02

Computing Observation Geometry for Small Satellites

Charles H. Acton
California Institute of Technology/Jet Propulsion Laboratory
4800 Oak Grove Dr., Pasadena CA 91109; (818) 354-3869
charles.h.acton@jpl.nasa.gov

ABSTRACT

Most solar system science missions need a variety of observation geometry – quantities such as position and velocity; range and altitude; viewing latitude and longitude; and lighting angles – to support mission engineering, science planning, and science data analysis activities. NASA's "SPICE" system offers one popular, multi-mission means for doing just that. SPICE comprises both data files, called kernels, and a SPICE software. A mission operations center produces the SPICE kernel files. Scientists and engineers write their own applications programs to address some need, and they include a few SPICE subroutines within that code to do the needed geometry computations. The SPICE system has been in use throughout NASA's planetary science mission domain since 1991, and it has slowly spread to most major space agencies around the globe since then. The SPICE software is available in most popular languages, and for most popular platforms. The code is thoroughly tested before being released, and new versions of the Toolkit are always backwards compatible. The SPICE components are freely offered to everyone, and have no export, licensing or similar restrictions. Maybe using SPICE would work for your CubeSat or SmallSat mission?

INTRODUCTION

For well over 25 years NASA's robotic planetary missions have relied on an ancillary information system named SPICE[1] for computing high-precision observation geometry data, parameters such as spacecraft and target body position and velocity, spacecraft and target body orientation, location of an instrument field-of-view projection onto a target body (latitude/longitude), lighting angles at an instrument's view point, time system conversions, etc.

SPICE comprises both ancillary data files and a software toolkit used to read those data and then compute the derived observation geometry quantities of interest. Scientists use SPICE to help plan scientific observations and to analyze the data returned from those observations. Engineers use SPICE to help design a mission and to carry out a variety of mission operations functions during flight. Even NASA's Deep Space Network uses SPICE to generate parameters used in pointing its antennas and tuning its transmitters and receivers.

GEOMETRY COMPUTATION CHALLENGES

In the case of deep space science missions nearly everything is moving or rotating, and estimates of target body size, shape, and orientation are often uncertain and subject to change. Missions generally use a variety of reference frames and coordinate systems. Returned telemetry can be spotty, and measures of time that glue the geometry calculations together are provided in several systems. Those not familiar with this mélange of ancillary information, as well as the means to process it to confidently obtain needed observation geometry parameters could find themselves struggling to make timely and accurate calculations. The challenge might be exacerbated in the small satellite environment where funding may be sparse, where time for development may be limited, and where an ever-changing student population can limit continuity. The challenge grows larger when the satellite is operating in "deep space," out of the range of Global Positioning System facilities.

HERITAGE

SPICE has been used by "large" science missions since the time of the Magellan mission to Venus. Originally embraced just by NASA's planetary missions, over the years it has been adopted by most major space agencies– ROSCOSMOS, ESA, JAXA, ISRO, and soon KARI. It has also been used in support of some Heliophysics and Earth Science missions, and in a number of technology

[1] **S**pacecraft, **P**lanet, **I**nstrument, **C**amera-matrix, **E**vents

demonstration flights. While developed as a professional tool, SPICE has nevertheless been used by some planetariums and other education and outreach activities.

OVERVIEW OF HOW SPICE WORKS

SPICE operations are initiated by collecting an assortment of so-called ancillary data from the data producers. Spacecraft trajectory, spacecraft orientation, instrument mounting alignment and field-of-view specifications, and target body size, shape and orientation are prime examples of ancillary data. Such data could arrive in a variety of forms. No matter the form, these are processed into SPICE data files, called "kernels." In some cases, SPICE software is used to do this processing, and in some cases a simple text editor is used. The kernel producers are responsible for validating the production, and for adding important metadata that describes the intended purpose of the kernels. The SPICE kernels are placed in a project's file system where all flight team personnel can access them.

Now it's the kernel consumers turn. A scientist or engineer that needs to make geometry computations as part of a science or mission engineering task writes, or obtains from a colleague, an application designed to do the job at hand. This application will include a handful of subroutines (APIs), taken from the SPICE Toolkit, that will do the needed geometry computations.

SOME DETAILS

The SPICE Toolkit source code is available for most popular computation platforms and it is available in many popular programming languages: Fortran 77, C, IDL, MATLAB, and Java Native Interface. Additionally, third parties have written "wrappers" around the C APIs in the Python, Ruby, Swift and Julia languages. All of the NAIF-supplied code is thoroughly tested and highly documented before being released. (Yes, some SPICE users complain there is too much documentation!) New versions of the SPICE Toolkit suite are released whenever enough new capability has been developed to warrant the effort. When a new version is released it is always 100 percent backwards compatible by design.

All Toolkit computations are double precision. The software contains an exception handling mechanism that is able to trap most user-initiated errors and provide an indicator of what the problem is. All the code contains extensive descriptions of inputs and outputs, and much of the code contains working examples of how it may be used.

TWO KINDS OF COMPUTATIONS

Early SPICE Toolkits contained just one type of computation: x = f(t), Fig. 1. As example, compute the spacecraft altitude at time T, or at a series of times between T_1 and T_2 with step size ΔT.

Current Toolkits also offer what is called the "geometry finder subsystem," Fig. 2. With this code one can search for times, within a user-specified containment window, when a binary condition such as occultation is true or false, or when a particular geometric parameter is within a given range or has reached a maximum or minimum. Different kinds of geometry-finder computations can be chained together to carry out rather complex searches.

DO IT YOURSELF

Most SPICE end users count on mission operations teams to provide them with the best possible spacecraft trajectory, spacecraft orientation, planetary ephemerides, and similar ancillary data. But many scientists wish to try out different target body size, shape and orientation parameters, or to compute the spacecraft location in a variety of different reference frames. The SPICE system makes these kinds of custom tunings quite easy to do.

LEARNING SPICE – IS IT WORTH IT?

In addition to extensive code documentation and subsystem technical reference manuals, the NAIF Team provides a series of SPICE Tutorials and several open-book programming lessons to help new users come up the learning curve. The team also offers the occasional three-day, free, SPICE training class. (Some foreign space agencies also offer SPICE training classes.) Nevertheless, it takes non-trivial time and effort to become proficient in using the SPICE system; is it worth this effort? There's not an easy answer. Perhaps the best we can say is that hundreds of persons just like you have successfully made this effort. Once one has become familiar with at least the most important pieces of SPICE, one should be able to re-use this knowledge.

WHAT DOES SPICE COST?

All of the items discussed so far are totally free to everyone worldwide. Also freely available are an assortment of "generic" SPICE data such as planet and satellite ephemerides, and more; these are data (kernels) not tied to a specific mission. Similarly, all of the SPICE data from past and on-going missions that have been archived at the NAIF Node of the Planetary Data System are free. No fees, no licenses, no registrations, no export restrictions, no restrictions as to commercial uses.

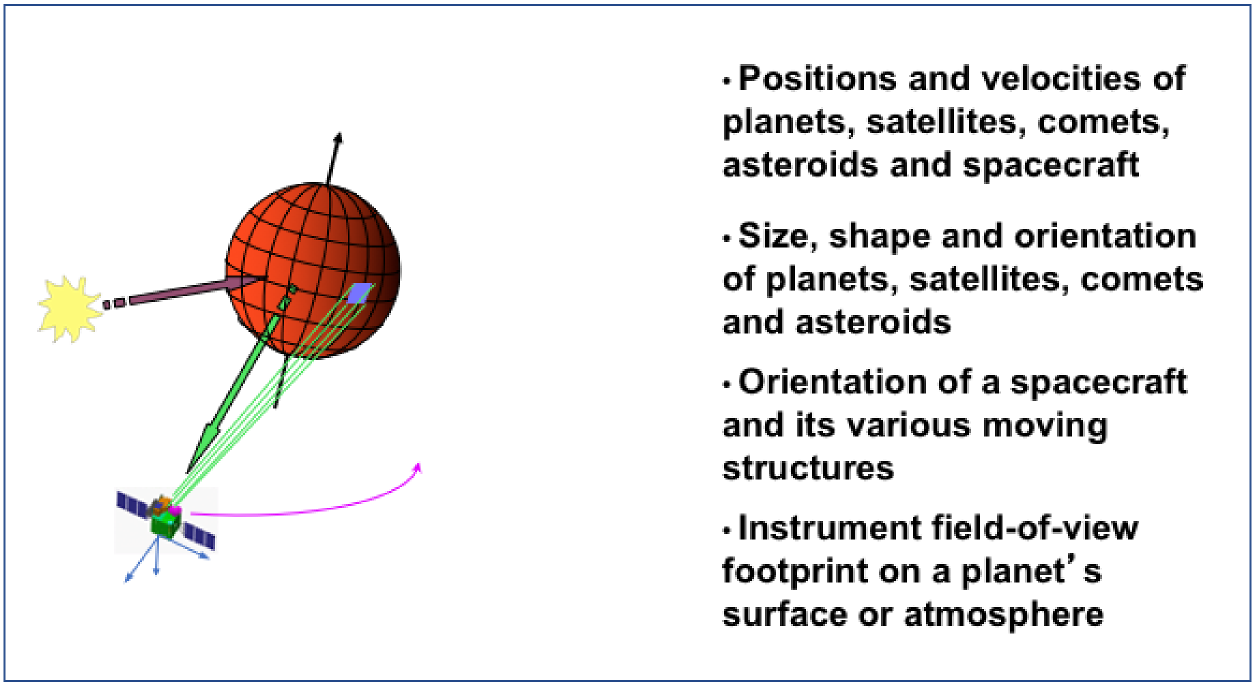

Figure 1: Original Kinds of SPICE Computations

Geometric parameter as a function of time

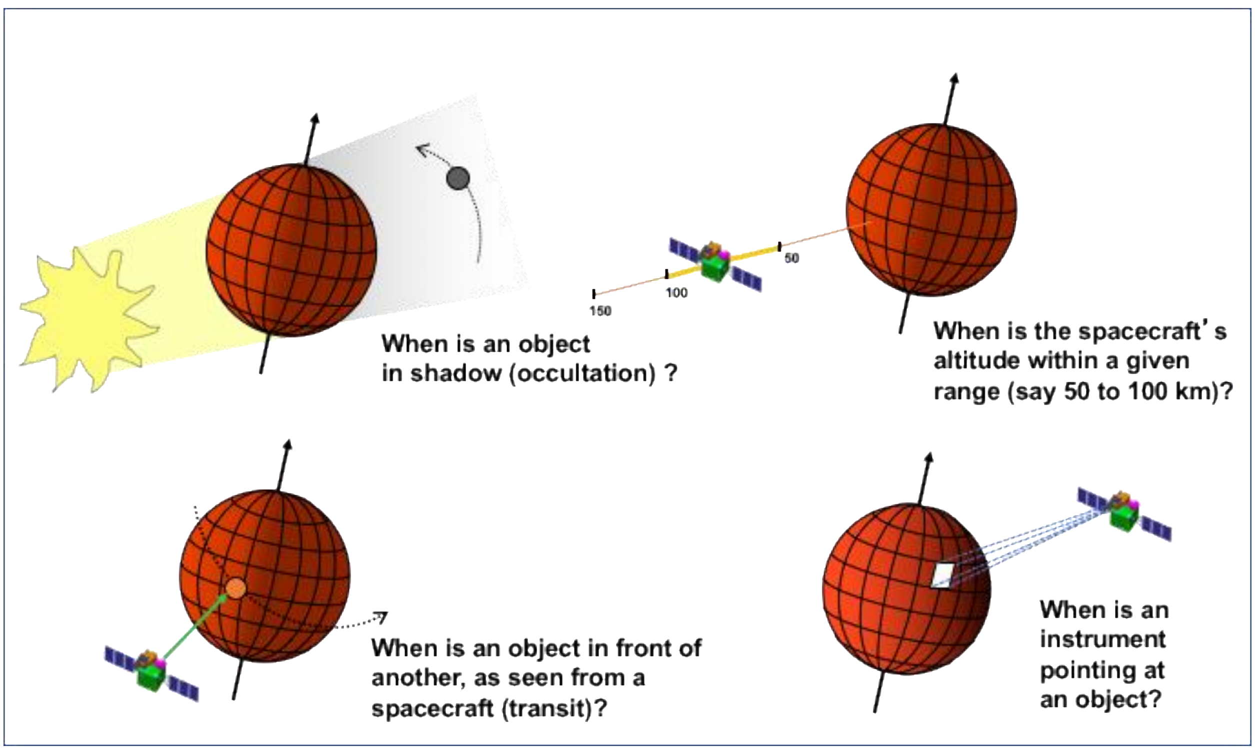

Figure 2: Geometry-finder SPICE Computations

Find times when a geometric condition exists or is within a specified range

Any project is free to use the SPICE system on their own. Such teams should take into account that learning to use SPICE requires non-trivial effort on the part of the SPICE kernel producers and on the part of the SPICE users. The free SPICE training classes can be of some help in this regard, but are likely not sufficient. Reading SPICE documentation, examining existing SPICE archives, and making trial runs using available SPICE software should be worked into one's schedule.

The NAIF Team is unable to provide user support to flight projects or other endeavors except where specifically funded to do so. The cost for NAIF's help varies depending on what is required, but projects have so-far found these costs to be very reasonable. Some details about the kinds of support available may be found at this link: https://naif.jpl.nasa.gov/naif/support.html

SUMMARY

Projects having substantial observation geometry requirements are encouraged to select the approach to be used well before launch so that everyone involved can plan accordingly. Time for testing of the selected system should be included in the project's development schedule.

Perhaps NASA's SPICE observation geometry system would be a workable solution for some SmallSat and CubeSat missions, especially those having space science objectives. Time spent learning SPICE constitutes a reusable investment. More information about SPICE and NAIF may be found on the NAIF website: https://naif.jpl.nasa.gov/naif/.

Acknowledgments

The research described in this publication was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

1. Acton, C.H.; "Ancillary Data Services of NASA's Navigation and Ancillary Information Facility;" Planetary and Space Science, Vol. 44, No. 1, pp. 65-70, 1996.
2. Charles Acton, Nathaniel Bachman, Boris Semenov, Edward Wright; A look toward the future in the handling of space science mission geometry; Planetary and Space Science (2017); DOI 10.1016/j.pss.2017.02.013 https://doi.org/10.1016/j.pss.2017.02.013