

EXOSIMS Separate Mass Decrement Enhancement MEng Project

Jayson Figueroa

December 20, 2021

Abstract

An enhancement for the software package Exoplanet Open-Sourced Imaging Mission Simulator is developed. The changes involve simulations of a space observatory that uses a separate occulting spacecraft, also known as a starshade, to suppress incoming starlight to image exoplanets. The development process emphasized planning and documentation prior to implementation. Changes needed to add new functionality without causing older input scripts to encounter new errors. Once the code was developed, a cycle of reviewing proposed changes and requested fixes occurred until the final code was deemed complete. Once pulled into the main repository, individual function testing and end to end testing is performed for validation. The proposed enhancement development was successful, and now enables further investigations regarding propulsion requirements for future planning of starshade missions.

Contents

1	Introduction	3
2	Background and Literature Review	3
3	Methodology	5
4	Results And Discussions	6
5	Conclusion	6

List Of Figures

- Figure 1: Starshade formation.

List of Acronyms

- CR3BP: Circular Restricted Three Body Problem
- EXOSIMS: Exoplanet Open-Sourced Imaging Mission Simulator
- DRM: Design Reference Mission
- ICD: Interface Control Document

List Of Variables

- DRM: (dict) Design Reference Mission, stores observation data and spacecraft status for each scheduled observation
- scMass: (float) Occulting spacecraft wetmass in kg
- skMass: (float) Occulting spacecraft fuel mass for station-keeping maneuvers in kg
- slewMass: (float) Occulting spacecraft fuel mass for slewing maneuvers in kg
- twotanks: (boolean) Boolean signifying if the occulting spacecraft has two separate tanks for maneuvers

1 Introduction

Exoplanet Open-Source Imaging Mission Simulator (EXOSIMS) is a software framework created by Professor Dmitry Savransky to generate end-to-end simulations of exoplanet imaging missions. The difficulty of planning exoplanet imaging missions is uncertainty in what results the mission will provide; to be certain if an exoplanet is able to be imaged its location is needed, however with exoplanets information is typically limited to only its existence. Instead, EXOSIMS provides a means to virtually test how a mission would perform in a randomly generated universe, so given a planet population model one can see how the mission would perform not in our exact universe, but many different simulated universes. Simulation results can be used as figures of merit during a mission’s early life-cycle and formulation stages, or as a means to understand parameter sensitivities and effects through Monte-Carlo analysis [1]. My project was to extend the available input arguments for a particular mission architecture so that a wider set of potential spacecraft designs can be simulated.

2 Background and Literature Review

In direct imaging, the primary obstacle for taking images of exoplanets is the light from the stars they orbit. Stars emit light many orders of magnitude higher than planets, and so to only collect light from the target planet for an image, the light from these stars must be suppressed in the space telescope’s view. This can be done using a physical light shield between the star and the telescope.

Within EXOSIMS, two high-level spacecraft architectures can be used to suppress starlight: light shielding internal to the observational spacecraft using a coronagraph [1], or light shielding external to the observation spacecraft using a separate spacecraft along the telescope’s line-of-sight. The focus of this project is on the occulting spacecraft of the latter mission architecture, called a starshade spacecraft.

The formation and modeling of the starshade is described in Soto et al. [2]. The starshade has two primary maneuvers during the mission: slewing and station-keeping. Slewing is performed when the observatory spacecraft changes its line-of-sight from one target star to another through a large-angle attitude change, and the starshade must maneuver to stay within the changing line-of-sight. A pointing change for the observatory requires a position change for the starshade. Station-keeping is performed to maintain the starshade’s orbit during the mission, as disturbance forces if left unaccounted for will perturb the starshade’s position away from the observatory’s line-of-sight. The formation between the starshade and the observatory spacecraft is shown in Figure 1.

The starshade must use a propulsion system with enough thrust for each maneuver and enough propellant for the life of the mission. The maneuvers are inherently different, as slewing occurs at the command of the observatory and an increased duration of the maneuver decreases observation time, while station-keeping occurs continuously and happens frequently enough that the

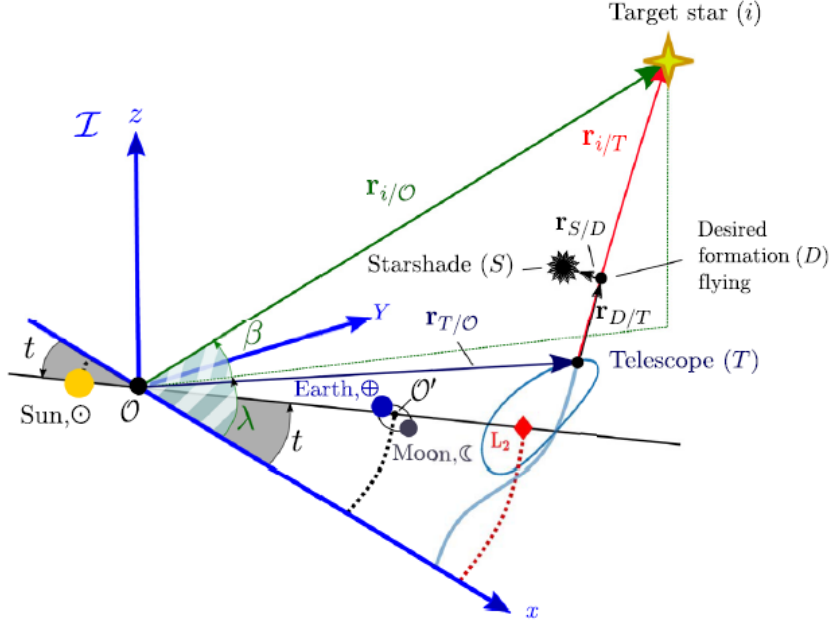


Figure 1: Starshade formation. The coordinates are based on the circular restricted three body problem (CR3BP) and has its origin between the Sun and Earth-Moon-Barycenter. The observatory spacecraft is orbiting about the second Lagrange point pointed at a target star. The starshade is between the observatory and the star to reduce starlight entering the observatory's view. Figure taken from Soto et al. [2].

perturbations do not remove the starshade from the observatory's line-of-sight within some margin.

Prior to this project, EXOSIMS used a propulsion system architecture where both maneuvers are done using the same type of fuel, while the true design space could include different propulsion systems with different fuel types. My project was to allow for separate fuel tanks, one for slewing and one for station-keeping. This extends the design space that EXOSIMS can simulate, which can now include a combination of chemical monopropellant and bipropellant thrusters or even a chemical propulsion system for one maneuver and an electric propulsion for the other. Expanding on the range of designs EXOSIMS can simulate is valuable to future uses of the code, especially considering the emphasis on direct-imaging exoplanet missions from the recent NASA Decadal survey [3]. This contribution allows for further investigation of the propulsion requirements of a starshade spacecraft and how it affects the overall mission.

3 Methodology

The development schedule given by Professor Savransky was structured to emphasize documentation before implementation. My first task was to understand the requested changes and how the proposed enhancement should behave. This step included the literature review and reading through the interface control documentation (ICD), as well as meetings with Professor Savransky to establish the desired inputs and outputs [4]. For this project, the desired functionality was to split up the fuel tanks for starshade maneuvers, and so it was decided that the new optional inputs included a boolean value *twotanks* to turn on the tank separation mode, and two float values *slewMass* and *skMass* which represented the available fuel masses for each maneuver. The desired output was to expand the Design Reference Mission (DRM), a dictionary variable storing simulation output, to include tracked values of *slewMass* and *skMass*. After this functionality was decided, I updated the ICD to include the new inputs and outputs with descriptions of their behavior.

Once Professor Savransky approved of the updates to the documentation, I began the implementation. To start, I needed to know where mass decrements computations were performed. Since prior to this project all mass decrements acted on the total spacecraft wetmass variable *scMass*, I needed to find each place in the software this variable was referenced. This was done using a program called grepWin, a free Windows software tool that can search for phrases in individual files. This is essentially the Windows version of the Linux command line utility grep. Using grepWin, I found exactly where in the software the fuel mass was currently being changed, which was done in the Observatory and Survey Simulation prototype modules.

The mass decrement functions prior to any changes already calculated and distinguished slewing and station keeping maneuver fuel costs, and so the needed changes were to include a conditional if statement, which executes if the input *twotanks* was true. Inside the if statement, the inputs *slewMass* or *skMass*, depending on the maneuver, was decremented by the same value computed outside of the if statement. The if statement ended with *slewMass* or *skMass* being updated in the DRM. Default values for *twotanks*, *slewMass*, and *skMass* were such that older input scripts would have the same outputs as before the new functionality was added.

Other changes included updating any modules beyond the prototype modules to work with the new optional inputs and outputs. The Time Keeping prototype module needed to be updated, specifically the `mission_is_over()` function. This function checks if there is any fuel remaining on the spacecraft, and if there is not the simulation declares the mission is over and stops running. The changes made here were to end the mission if *slewMass* or *skMass* is 0, assuming *twotanks* is true. Finally, EXOSIMS includes unit testing of its functionality, and so the new inputs, outputs, and Time Keeping changes were added to the testing scripts to assure that any future changes work with the fuel mass separation enhancement.

Once the coding was completed on my personal repository, the next step

was to send the changes to the main EXOSIMS repository, which is hosted on Professor Savransky's GitHub page. To do this, first I forked the main repository to create my own repository for changes, which was done at the start of the project. After the changes described above were coded on my local machine, I updated my repository hosted on GitHub, then submitted a pull request to the main EXOSIMS repository. This started an iterative review and change request process between Professor Savransky and I, where he would review the proposed changes line by line and request alterations based on how I coded the functionality. Change requests included removing redundancy found in some of the Survey Simulation modules and adding an assertion when the tank separation is activated to assure legal inputs. Once all the requested changes were made, my changes in the code were pushed to the main EXOSIMS repository, making the enhancement available to anyone using the software. The final line by line alterations in code are included in appendix [PLEASE CITE].

4 Results And Discussions

Once added to the main repository, the GitHub performs all unit tests and several end to end tests to validate any changes made, which takes about half an hour to complete. Initially, my changes did not pass all the unit tests due to an unseen bug in the unit test changes, nor was the functionality properly tested by the end to end tests. Once the unit tests were fixed and the tank separation was included in the end to end tests, the changes passed all tests to validate the updated code.

Further work is planned as of writing this report. Currently, I have a pull request submitted adding optional inputs to adjust maneuver efficiencies, essentially simulating the fuel costs for each maneuver if there are losses present. These losses are common since in practice spacecraft maneuvers always have non-idealities, for example if thrusters are slightly misaligned the maneuver is more costly. The added efficiency factors can be used to simply capture these non-idealities in the simulation. Beyond this, as the project is open-source I can continue to add requested changes and bug-fixes to the software through GitHub using the same process performed in this project.

5 Conclusion

In conclusion, I successfully developed and merged code into EXOSIMS which adds the ability to separate fuel mass used on starshade spacecraft into two tanks, one dedicated to slewing maneuvers and one dedicated to station-keeping maneuvers. This process included a literature review of the starshade mission architecture and the EXOSIMS documentation, determining how the new code would behave, adding proposed changes to the documentation with explanations of functionality, implementing these changes in the code, a review and change request cycle of the submitted pull request on GitHub, and finally a merge

of the enhancement following a successful passing of all unit tests and end to end tests. This new functionality enables a wider design space that can be investigated using EXOSIMS by allowing the starshade spacecraft to have a propulsion system with two separate fuel tanks, such as a chemical propulsion system for slewing and an electric propulsion system for station keeping. The need for these changes is clear given the recommendation for an exoplanet direct imaging space observatory mission in the recent NASA Decadal Survey.

References

- [1] Savransky, Dmitry, and Daniel Garrett. “WFIRST-AFTA Coronagraph Science Yield Modeling with EXOSIMS.” *Journal of Astronomical Telescopes, Instruments, and Systems*, vol. 2, no. 1, Dec. 2015, p. 011006. www.spiedigitallibrary.org, <https://doi.org/10.1117/1.JATIS.2.1.011006>.
- [2] Soto, Gabriel J., Dmitry Savransky, and Rhonda Morgan. “Analytical Model for Starshade Formation Flying with Applications to Exoplanet Direct Imaging Observation Scheduling.” *Journal of Astronomical Telescopes, Instruments, and Systems*, vol. 7, no. 02, Jan. 2021. DOI.org (Crossref), <https://doi.org/10.1117/1.JATIS.7.2.021209>.
- [3] National Academies of Sciences, Engineering, and Medicine 2021. *Pathways to Discovery in Astronomy and Astrophysics for the 2020s*. Washington, DC: The National Academies Press. <https://doi.org/10.17226/26141>.
- [4] Keithly, Dean, et al. *Exoplanet Open-Source Imaging Mission Simulator (EXOSIMS) Interface Control Document*. 1 Dec. 2021, https://exosims.readthedocs.io/en/latest/_downloads/da748bd624ec388fa7f26dbed4f1d1f6/icd.pdf.
- [5] Soto, Gabriel J., Dmitry Savransky, Daniel Garrett, et al. “Parameterizing the Search Space of Starshade Fuel Costs for Optimal Observation Schedules.” *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 12, Dec. 2019, pp. 2671–76. DOI.org (Crossref), <https://doi.org/10.2514/1.G003747>.