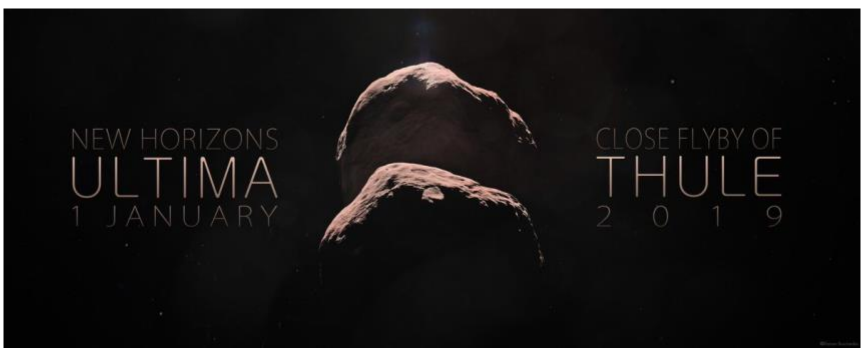
CS486C – Senior Capstone Design in Computer Science

Project Description

Project Title: **Lightcurve modelling of binary systems**	
Sponsor Information:	Will Grundy, Audrey Thirouin Lowell Observatory, Flagstaff thirouin@lowell.edu phone: +1 (928) 233-3233

Introduction:

As interest in travel to small bodies in our solar system, like asteroids, comets, and Kuiper belt objects grows, it is increasingly important to understand the rotational, physical and dynamical properties of these objects. In particular, the Lucy and the New Horizons missions will visit the binary system Patroclus-Menoetius and the probable contact binary 2014 MU_{69} and thus it is vital to understand such systems and, by extension, other binary systems for comparison. Our group at Lowell Observatory is involved in these two missions and we are currently leading an international observational campaign to characterize the Patroclus-Menoetius system, as well as leading a survey to discover Kuiper belt objects similar to 2014 MU_{69}. In order to put in context our observations and extract key information about binary systems regarding their physical, dynamical and rotational properties, accurate modelling is required; providing this modeling is the key focus and contribution of our team at Lowell Observatory.



Of particular interest to us are "binary systems", which are composed of a satellite orbiting a primary (like the Earth-Moon system). Binary systems are found across the Solar System, from the Near-Earth Objects to the Kuiper Belt, and display a large variety of shapes, sizes, orbits, and spin states. From far away, all we can see is the brightness of a binary system, and how it changes over time as objects rotate, but from such observation, it is possible to infer detailed information about the objects and their orbital dynamics. In the study of binary systems, a key method is to carefully observe moment in which the system's two components eclipse and occult one another, known as *mutual events*. By observing such events, we can infer many detailed characteristics of each component and of the system as a whole (sizes, shapes, orbit, spin states).

Problem Overview:

Currently, considerable effort is required to take observations of a binary system, create a model of system behavior, and extract the detailed characteristics of the system because, although the basic algorithms are well-known and some software tools already exist, there is no coherent package available for this purpose. Therefore, the main goal of this project is to create a well-documented package to model a large variety of binary systems with different orbital, physical and rotational properties. Specifically, the aim is to create a proof of concept desktop application running in a Linux environment that focuses on modeling the *lightcurve* of binary systems, which one of the key data extracted from observation of a system. Some key functional features in the software product include:

- A well-designed interface (API) for configuring/controlling the modeling tool, and for inspecting/exporting its output.
- Implementation of 3D shape models of spherical to elongated objects using figures of equilibrium.
- Creation of binary systems with different orbital motions using the Navigation and Ancillary Information Facility (NAIF)/SPICE library (here), and physical and orbital characteristics.
- Will include modeling of light-scattering properties of irregular surfaces using Hapke modelling or the Lommel-Seeliger law.
- Ray-tracing with the software Pov-Ray (here) or similar software.
- The forward model should be callable from a non-linear chi-squared minimization routine to use observations to constrain orbital, scattering, and shape parameters.
- Testing and verification of the product by application to well-known binary systems, with statistical comparison of output to known characteristics of such systems.
- The minimum functionality to satisfy basic needs (minimum viable product) is a working code/package able to simulate binary systems, produce the system's lightcurve, and use this to compute a best set of parameters to fit the observations.
- A more complete product would have additional features to make it truly useful and easy to use. Such features might include:
    - Explore optimization such as more efficient computational techniques and exploring parallel processing.
    - Translation of outputs of the simulation into a set of images representing the rotation of the system which could be used to visually inspect the modeled system behavior.

Knowledge, skills, and expertise required for this project

This project is typical of a software project in scientific computing, where computer scientists work closely with domain experts to create a product that supports the scientific enterprise. Although some terms and computational concepts outlined above may sound intimidating, sponsors will provide technical support for all mathematical or astronomical concepts; the team is there to implement these into software modules. The following are skills the team must have or learn to bring to the project:

- Programming skill in an appropriate programming language for the project (to be determined by team). Some candidate languages include IDL and Python, but others could be considered.
- Familiarity with Linux computing environments.
- Skill in integrating various modeling tools/packages to create complex systems behavior models.
- Some domain knowledge of astronomical observation and associated informatics will need to be learned by the team.

Equipment Requirements

- There is no specific equipment required for this project beyond a standard software development machine.

- The required datasets will be provided for the team by the sponsor, as technical support with respect to doing the modeling and all related tools and technologies.

Deliverables

- The software application as described above, deployed and tested successfully in the Lowell computing environment. Must include a complete and clear User Manual for configuring and operating the software.
- A strong as-built report detailing the design and implementation of the product in a complete, clear and professional manner. This document should provide a strong basis for future development of the product.
- Complete professionally-documented codebase, delivered both as a repository in GitHub, BitBucket, or some other version control repository; and as a physical archive on a USB drive.