

Parallelized Model of Low-Thrust Cargo Spacecraft Trajectories and Payload Capabilities to Mars

Wesley Yu

Department of Aerospace Engineering and
Engineering Mechanics

The University of Texas at Austin

Austin, TX 78712

+1(915)309-7972

yuwesley@utexas.edu

Hans Mark

Department of Aerospace Engineering and
Engineering Mechanics

The University of Texas at Austin

Austin, TX 78712

+1(512)471-5077

hmark@mail.utexas.edu

ABSTRACT

As a Blue Waters Student Internship Program project, we have developed a model of interplanetary low-thrust trajectories from Earth to Mars for spacecrafts supplying necessary cargo for future human-crewed missions. Since these cargo missions use ionic propulsion that causes a gradual change in the spacecraft's velocity, the modeling is more computationally expensive than conventional trajectories assuming instantaneous spacecraft velocity changes. This model calculates the spacecraft's time of flight and swept angle at different payload masses with other parameters kept constant and correlates them with known locations of the planets. With parallelization using OpenMP on Blue Waters, its runtime has decreased from 10.55 to 1.53 hours. The program takes a user-selected Mars arrival date and outputs a given range of dates with maximum payload capabilities. This parallelized model will greatly reduce the time required for future mission design projects when other factors like spacecraft solar panel power output may vary with new mission specifications. The internship experience has enhanced the intern's ability to manage a project and will impact positively on his future graduate studies or research career.

CCS Concepts

•Applied computing → Physical sciences and engineering
→Aerospace

Keywords

Computational Modeling; Orbital Mechanics; Low-Thrust Trajectories

1. INTRODUCTION

Sending humans to Mars has presented difficult problems to solve in the past few decades since the Moon landings. With the advent of new technology and launch methods, such as NASA's Space Launch System (SLS) and SpaceX's reusable first stage, the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Copyright ©JOCSE, a supported publication of the Shodor Education Foundation Inc.
DOI: <https://doi.org/10.22369/issn.2153-4136/8/2/7>

possibility of reaching Mars within the next few decades is becoming a reality.

There are still many aspects of a crewed mission that need work, like the effect of galactic cosmic radiation on the crew's health, and the ability to get all necessary equipment and backup supplies to the Martian surface to keep the crew safe there. The latter problem leads into the focus of this study.

Ionic engines can be the safe and efficient source of propulsion [1,2,3] for cargo missions to supply the Martian surface with food, water, habitats, scientific instruments, and transportation equipment. With the low-thrust characteristics of ionic engines, we cannot model their trajectories with instantaneous velocity changes [2]. Instead, the calculations must take into account the continuous velocity change along the entire trajectory at infinitesimally small time steps. This study explores the use of parallel computing to speed up the calculation of these low-thrust cargo transfers.

2. BACKGROUND

The possibility of travelling to Mars hinges largely on our ability to support astronauts with necessary sustenance and equipment for the length of their stay on the surface. This requires consideration of backup safety factors that call for multiple cargo launches to ship the necessary supplies to the Martian surface [3]. The eventual goal would be continual shipments from Earth to Mars to setup surface stations and provide a growing community of shelter and supplies for the astronauts to utilize. These cargo missions would be unmanned and much more frequent than the crewed missions.

Each individual cargo mission itself does not need to reach Mars quickly because of the continual nature of the shipment process. Therefore, we do not require current high-thrust chemical propulsion to complete Earth to Mars transfer in the usual eight months [2]. In order to increase safety and conserve fuel, we can utilize ion propulsion for this operation. This would lengthen the trip to a year and a half, but each shipment would be arriving within a few months of each other, giving the astronauts a steady supply, much like the current supply system for the International Space Station.

The main difference between ionic and chemical propulsion is the method in which the thrust is applied. In chemical engines, cold gas is ignited to cause an expulsion of propellant at a very high mass flow rate. The thrust, in turn, is very high but fuel runs out within a few minutes [4]. Thus, chemical engines are fired only

during a very short period of time, and can be modeled assuming an instantaneous change in velocity, ΔV [5]. Using solar panels to provide electric power, ionic engines operate by using energetic electrons to bombard and ionize noble gases, such as argon and xenon [2,3,6]. A potential difference then develops, accelerates the ions, and emits a positively charged beam as the exhaust. Unlike conventional chemical engines, the mass flow rate is very low [7,8], so the thrust must be applied throughout the entire transfer. These two different propulsion systems using ionic and chemical engines would produce two contrasting types of orbital transfers, known as the Ward spiral and Hohmann transfer respectively (see Figure 1). Modeling the Ward spiral trajectory requires a calculation of velocity change at each point along the spacecraft's path. Optimizing these low-thrust trajectories for a proper launch date becomes computationally expensive, and would benefit from a parallelized computational model.

There are two main questions we are trying to answer in this study. The first is the relationship between the time of flight and payload mass of the spacecraft. The second is the effect of launching at non-ideal times on the payload mass capabilities of a certain spacecraft. Having a model to calculate this will help determine safety factors and contingencies in a mission design project.

3. METHODS

The main model in this study centers on a low-thrust Ward spiral, with the inputs being the thrust and mass of the spacecraft. After accounting for the inclination change between Earth's and Mars' orbits, we correlate the positions of the two planets in the coming decades, called ephemerides, to the possible trajectories. These ephemerides tables are obtained from the NASA Jet Propulsion Laboratory's (JPL) HORIZONS online tool [9] by inputting a start and end time, then specifying the time step size. Since we are

testing all the launch dates over a five-year period as well as altering the payload mass levels, parallelizing the code would speed up the process significantly.

3.1 The Ward Spiral

Using a method developed for satellite orbit lowering maneuvers that employ low thrust ion engines [10,11], we reversed the direction of travel to an orbit raising trajectory [10]. Since Earth and Mars are on slightly different orbital planes, the change in inclination also adds additional ΔV to the trajectory, so we had to consider this in our calculations as well. This allowed us to model a spiral of gradually increasing orbital radius that reached from Earth to Mars. The Ward spiral starts with a circular orbit where the orbital velocity, v , is calculated by the equation:

$$v = \sqrt{\frac{\mu}{r}}$$

where μ is the gravitational parameter of the Sun and r is the spacecraft's distance from the Sun [4]. From the inputted thrust, F , of the spacecraft we can calculate the power, P , of the spacecraft:

$$P = Fv = F\sqrt{\frac{\mu}{r}}$$

and correlate it to the vis-viva energy [4], defined as the total energy of the spacecraft in a circular orbit, E :

$$E = -\frac{\mu m}{2r}$$

where m is the inputted spacecraft mass. Since the time derivative of energy is power, we can equate \dot{E} to P and therefore:

$$F\sqrt{\frac{\mu}{r}} = \frac{\mu m}{2r^2}\dot{r}$$

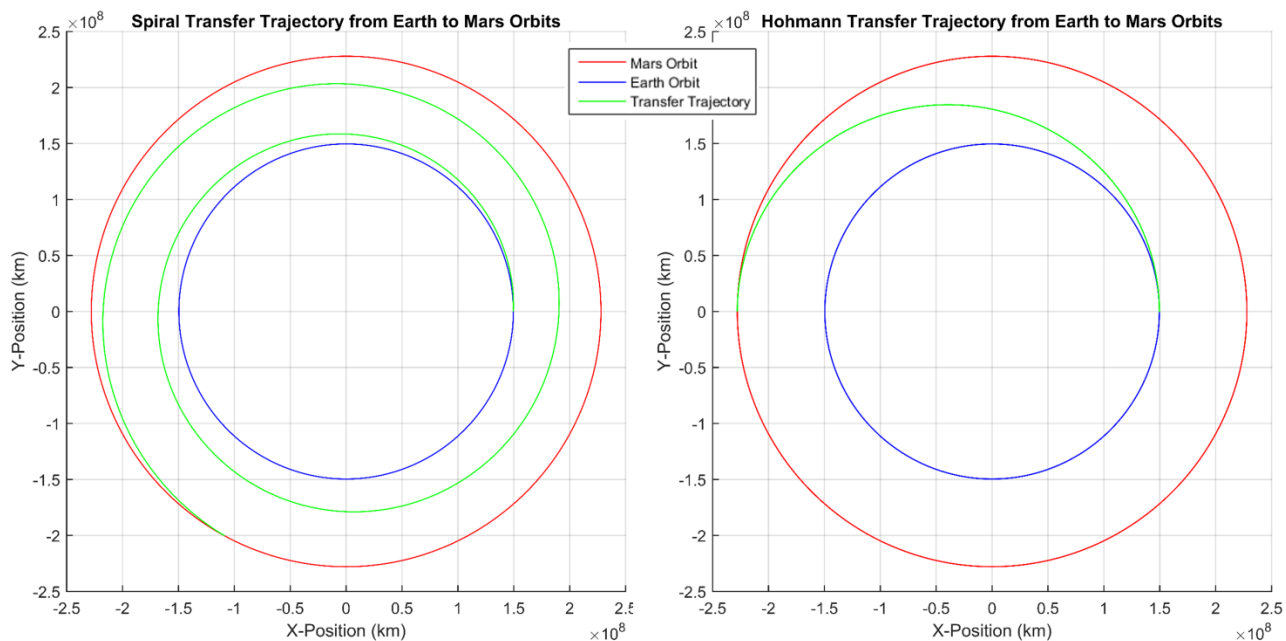


Figure 1. Comparison between a low-thrust Ward spiral and conventional Hohmann transfer from Earth to Mars. The blue circle represents Earth's orbit around the Sun, red is Mars' orbit, and green is the transfer trajectory.

We now have a first-order differential equation that directly relates the time derivative of the orbital radius and the thrust of the spacecraft's ion engine, with initial conditions of $r = r_0$ (Earth's orbital radius around the Sun), at $t = 0$.

$$\frac{dr}{dt} = \frac{2F}{\sqrt{\mu m}} r^{3/2}$$

Solving the above differential equation gives:

$$r = \frac{r_0}{\left(1 - \frac{F}{m} \sqrt{\frac{r_0}{\mu}} t\right)^2}$$

This calculates the orbital radius at any given time, t . We can now use the initial and final orbital radii, r_0 and r_1 , to find the time of flight, t_f , and integrate the stored orbital radius and velocity values to find the total swept angle, θ , of the transfer orbit:

$$t_f = \frac{m}{F} \sqrt{\frac{\mu}{r_0}} \left(1 - \sqrt{\frac{r_0}{r_1}}\right)$$

$$\theta = \int_0^{t_f} \frac{v(t)}{r(t)} dt$$

With these relations it is straightforward to code a model for a single trajectory, with an array of time and an inputted thrust and mass. The next part of the model calculates the inclination change between Earth's and Mars' orbits. Just as energy is required to change the orbital radius, the change in orbital inclination requires an amount of ΔV given by:

$$\Delta V = 2v \sin\left(\frac{\Delta i}{2}\right)$$

where Δi is the change in inclination. Since the orbital velocity v is constantly decreasing as the orbital radius increases, the total ΔV is a summation of all the inclination changes along the trajectory [4].

3.2 Trajectory Optimization

3.2.1 Time of Flight

For conventional propulsion, the optimization is conducted with ΔV as the main objective. Minimizing the ΔV in a maneuver will minimize the fuel needed for a certain transfer, thus lowering the overall cost of the mission. [5,6]. However, with a low-thrust trajectory, the ΔV is determined by the difference in orbital velocities of Earth and Mars around the Sun, and is therefore a constant. So regardless of how long it takes to reach Mars on a certain trajectory, the spacecraft needs the same amount of ΔV . Therefore the main optimizing objective is the time of flight of the transfer, which is mostly dependent on the locations of the planets at a given time [12]. Although there will be a limit to how fast the spacecraft can reach Mars due to the extremely low thrust of the ion engines, we still would like a reasonably quick transfer. This would minimize exposure of food/water and even some scientific equipment to the harmful effects of galactic cosmic radiation.

3.2.2 Computational Approach

With the two inputs of thrust and payload mass, we chose to keep thrust constant for each run, as an assumption that we are using one engine with readily available electrical power. The thrust of 0.023 N is used for our computations based on the NASA-released information on the NEXT ion thruster [3]. We then

varied the payload mass from 1 to 20 metric tons (t) with increments of 0.1 kg, and calculated the time of flight and swept trajectory angle for each mass value. This range was chosen based on payload masses in NASA heritage missions as a lower bound, up to anticipated future necessary mass levels as an upper bound [3]. With these values stored in a text file, we correlated the data with ephemerides to find potential launch windows for missions with different payload masses.

As we make the payload mass increments smaller to increase the accuracy of time of flight calculations, the number of data points also increases, making the text files larger in size. To minimize the inconvenience of loading large text files we separated the outputs into smaller organized files labeled with the first mass value in the file. The main use for the files is to create a table that relates mass and time of flight so that when needed, one can extract the values on a desktop computer, without having to access Blue Waters. The most important application would be a mission plan to arrive on Mars by a certain date, given the maximum payload mass and necessary thrust. This would require many calculations of the time of flight versus mass, and with the tabulated values for a given thrust, it will be much easier to read through them instead of calculating the time of flight each time.

All the code that was run on Blue Waters to calculate time of flight was written in C, and compiled using cc in the default craype/2.5.0 programming environment with Cray Linux Environment (CLE 5.2). The programs for further payload capability analysis were also written in C and compiled with gcc/4.9.2 in a Cygwin 6.3 terminal installed on a Windows 10 desktop computer. We used gnuplot 5.0 to generate payload capability figures and MATLAB 2015b to generate the three-dimensional Earth-Mars trajectory plots.

3.2.3 Maximum Payload

Since the tabulated values are separated by thrust level, one can also start with a time of flight and arrival date, and use the tables to find the maximum payload mass capability for a given thrust. The model takes the required time of flight and associates it to a payload mass, and if the ephemerides suggest that the launch cannot be completed in time, the model will go to the next thrust level.

This application takes advantage of the tabulated values extracted from the parallelized code run on Blue Waters. The user can input the desired arrival date and a maximum time of flight, and the program can find a range of launch dates with their corresponding payload mass and present them in a plot. For mission design projects this would be a very useful tool to visualize the relationship between an actual launch date and the possible payload mass.

Another application is to lock the launch date at the ideal case, and then match it with the user's needed arrival date. Then we can vary the arrival date within a range around the user-selected date to see its effect on the maximum payload mass. This allows us to test the effect of launching at non-ideal launch windows on the maximum payload capability (including the mass spacecraft mass) of a mission. Using this model can help test contingency and develop safety margins for the mission design.

3.3 Launch Date Calculation

In order to calculate the proper launch dates we require the ephemerides of Earth and Mars for the coming decades. This allows the determination of the initial and final positions of the

planets during the transfer orbit, which can be correlated with the outputted times of flight to determine the necessary launch date of the cargo shipment. We generated the ephemerides from JPL's HORIZONS software [9] and loaded them into text files. Using a 50-year range from January 1, 2018 to December 31, 2067, we have the positions of both planets for the next five decades.

To avoid overly long or unrealistically short times of flight, we set an upper and lower bound on the times of flight calculated from the varying payload mass. Then the times were matched with their corresponding arrival dates. Using the known ephemerides we could work backwards to find the appropriate launch dates. This allowed us to identify "windows of opportunity" and optimal launch dates.

3.4 Parallelization

3.4.1 Parallelization Method

Finding the times of flight from varying payload masses represented the bulk of the computations needed in this project. The time required for extraction and analyses of the resulting data was negligible. For simplicity, we separated the parallelized time of flight calculations and tabulations using Blue Waters from the subsequent payload and launch dates analyses that can be run by a user on a typical desktop computer.

To parallelize the time of flight computations, we used OpenMP in the default Cray compiler on Blue Waters. The programming environment was the default craype/2.5.0, and we coded in C on OpenMP v3.0 to parallelize the for loops calculating the vector of different transfer orbits. The code was then compiled on Blue Waters using gcc/4.9.2, running on XE compute nodes.

We assigned one potential transfer orbit to each thread, with eight cores total for the parallel runs. With the thrust kept constant, each orbit had a different payload mass assigned to it and produced a time of flight. Then the trajectory's time of flight was correlated to the different launch dates with the same window for travel. The program would print out the time of flight, swept angle, and corresponding payload mass so that extracting the data later

would be easier. Both the serial and parallel versions of the code are available upon request.

3.4.2 Measuring Speed Increase

The most convenient way to track the run times of all our tests is to use the 'time' function for all tasks we run on Blue Waters. We ran several small data sets with around 1000 payload mass inputs for testing, where hardly any speedup was observed. We also ran 100 large datasets with millions of payload mass inputs to increase the accuracy in time of flight calculations, where more significant speed increase is expected due to the smaller percentage of overhead.

4. RESULTS

The products of the model we created are three-dimensional plots of the trajectories and possible launch dates for the mission. These are helpful in visualizing and planning missions, where the plots can show us the general shape and characteristic of the trajectory, the possible launch dates give us concrete numbers on the frequency and viability of the missions.

The more interactive product is a program that can determine the payload capability of the spacecraft based on given launch or arrival dates. The user can input a necessary Mars arrival date and time of flight, and use the model to find the maximum payload mass for a given thrust. If the mission does not meet the requirements for the given thrust, the model will find the next level suitable thrust level by adding additional engines to increase the thrust.

4.1 Plots of Trajectories

The plots take into account the inclination change of the spacecraft, and show the spacecraft's path from Earth's orbit around the Sun to Mars'. The inclination change is exaggerated in the plots for visual effect, but the actual change in inclination between Earth and Mars orbit is 1.85° . The x-y plane in this reference frame is the orbital plane of Earth around the Sun. Examples of trajectory plots are shown in Figures 2 and 3.

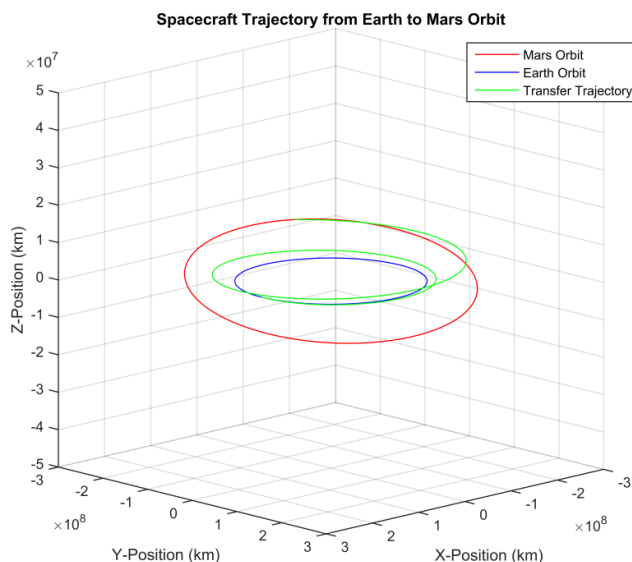


Figure 2. A typical Ward spiral from Earth to Mars, a trajectory that would take two years to complete for a 15t payload with a single NEXT engine.

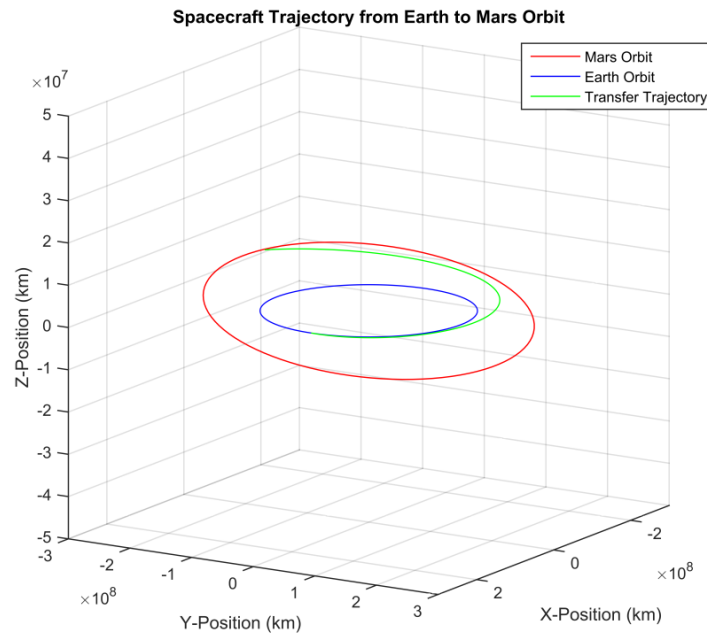


Figure 3. A much faster trajectory (10 months transit time) than in Figure 2, representative of a small probe's trajectory, where the payload would be relatively light.

4.2 Effective Launch Dates

As mentioned earlier, the model outputs possible launch dates given a Mars arrival date, which is very helpful in mission planning. The model can predict the date of launch, or even the hour of the launch if the user chooses. We represent the date as a Julian Date, which is the number of days since noon Universal Time on January 1, 4713 BCE, to facilitate ease of calculation. Julian dates can be converted back to the common Gregorian calendar dates easily using the U.S. Naval Observatory Julian Date Converter [13]. Table 1 below shows two examples of the outputs that the program will give with a 14.9t payload and 0.023 N thrust, with two different user inputted arrival dates:

Table 1. Ideal and Latest Possible Launch Dates

	Mission 1	Mission 2
Inputted Mars Arrival Gregorian Date	5/2/2023	6/17/2045
Corresponding Julian Date	2460066.5	2468148.5
Ideal Launch Julian Date Based on Time of Flight	2459406.5	2467384.5
Ideal Launch Gregorian Date	7/11/2021	5/15/2043
Latest Launch Julian Date Adjusted for Ephemerides	2459415.5	2467416.5
Latest Launch Gregorian Date	7/20/2021	6/16/2043

Launching either earlier or later than the ideal date would require more fuel. However, an earlier launch can give the mission greater scheduling flexibility. Launching later would lower the safety margin, limiting the flexibility of the mission. Beyond the latest launch date, it will not be possible to reach Mars by the desired arrival date with the given payload mass and thrust level.

4.3 Maximum Payload

With the program that outputs payload capability from a given time of flight, we were able to plot the maximum payload mass against a range of launch dates. With the added constraint of the arrival date and maximum time of flight, we could use this tool to narrow down the search of ideal launch windows.

The outputs from this program are once again tabulated in a text file as maximum payload mass and launch date. Such outputs for Mission 1 in Table 1, with required arrival date of May 2, 2023 are plotted in Figure 4. At the ideal launch date of July 11, 2021, the maximum payload mass of 14.9t is attained, but the payload capability decreases parabolically as the launch date deviates from ideal.

When locking the launch date to the ideal case in the aforementioned model and using the user-inputted arrival dates as a reference, we plotted the maximum payload mass while varying the arrival date. Once again we can see that the maximum payload mass remains at the ideal arrival date, but the loss of payload capability is less than that generated by deviations from the ideal launch date.

This computational tool is useful for trade studies when determining the cost of a mission and its resulting trade-off with payload mass. We can now see the effects of delaying or moving up the timetable of a mission, and how it will limit certain equipment or resources to be transported on the mission. With this information we can effectively plan shipments and their fail-safe launch dates.

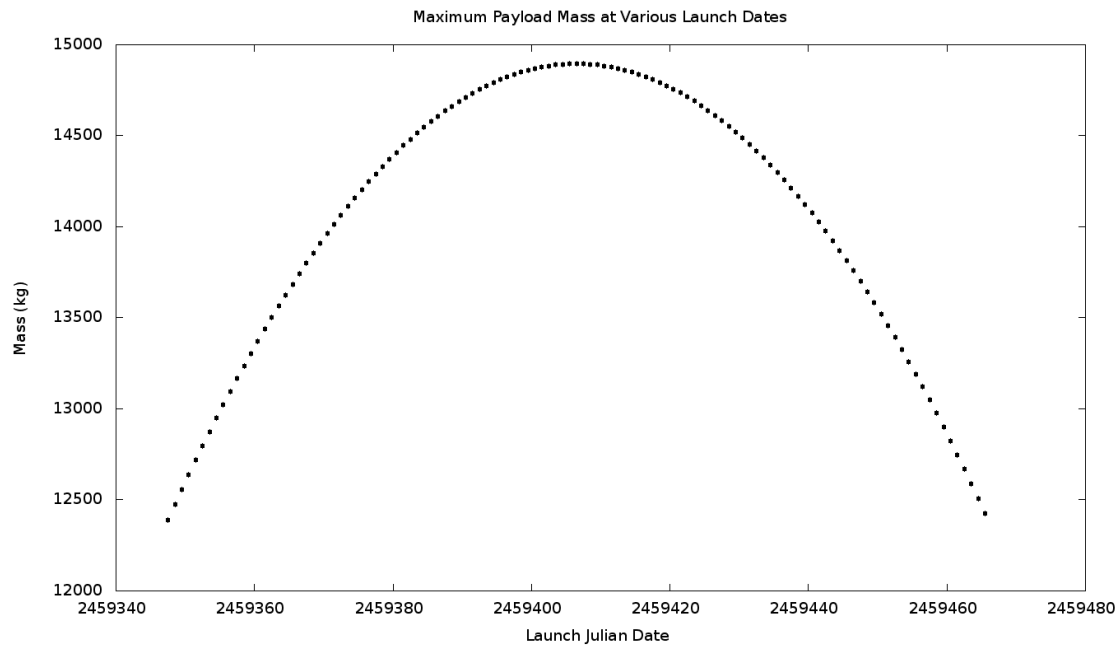


Figure 4. Maximum payload mass the spacecraft can carry at different launch dates with a constant time of flight. The suggested ideal date of July 11, 2021 (Julian 2459406.5, see Table 1) is on the apex of the graph.

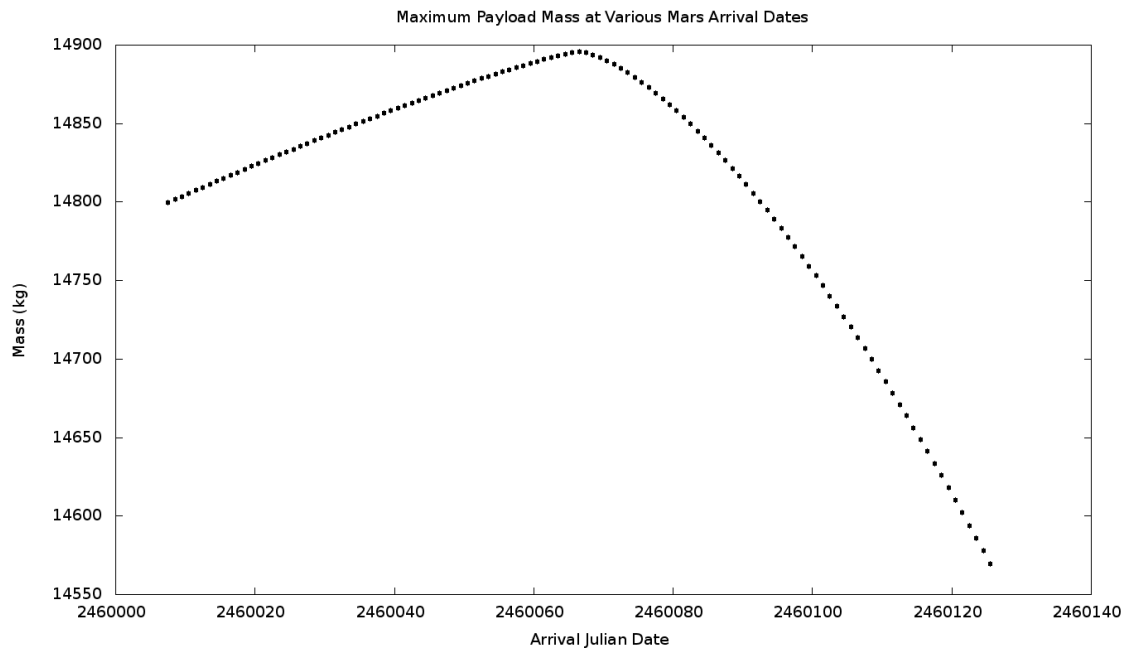


Figure 5. Maximum payload mass the spacecraft can carry at different arrival dates with the launch date kept at the original ideal from Table 1. Note the asymmetry of the decrease in payload capability on either side of the ideal arrival date. Due to the change in positions of Earth and Mars, arriving earlier versus later yields different payload capabilities.

4.4 Speedup After Parallelization

We ran both the serial and parallelized versions of the program on Blue Waters' Cray XE nodes, at a peak performance of 313.6 GFLOPS per node. As expected, when running small data sets of 1000 payload masses for initial testing, there was no noticeable speedup since the effects from overhead canceled out the parallelization. When running the parallelized version to calculate

10 million payload mass data points we used eight cores and the runtime decreased from 10.55 hours to 1.53 hours. This reduced runtime facilitated our development of the Ward spiral code without the need for overnight testing and outputting. The increased speed was utilized primarily to speed up the intensive calculations to prepare the tabulated values, allowing us to quickly start analyzing the resulting data for mission planning.

5. DISCUSSION

5.1 Usefulness of Parallelization

While using Blue Waters was extremely helpful in generating the time of flight tables, a user with the intent to input one mass and thrust value will see no difference in a single calculation of the trajectory versus a program that reads tabulated values. So the parallelization in this case did not affect the end user. However, with the case where the user is looking for maximum payload mass given a time of flight and arrival date, it would be very inconvenient to calculate the values serially on a laptop. Having the tabulated values to scroll through in this case would greatly reduce the computing time.

5.2 Viability of Low-Thrust Missions to Mars

From the trajectory plots, time of flights, and payload capabilities obtained from this study, we can see that the use of ionic propulsion is a viable method to provide support for future missions to Mars. The mission parameters that we have calculated are well within the reasonable range of required cargo shipment set by NASA standards. This will be key in future development of routine cargo missions when we continue our mission designing process.

5.3 Educational Impact of the Project

The project is a one-year Blue Waters Student Internship with an initial training workshop for C programming on the supercomputer, followed by required monthly reports submitted to the internship program. Throughout the course of the program, the intern has learned to structure a research project with a well-defined timeline to meet clear and realistic goals and steps. The experience has enhanced the intern's ability to plan and manage a research project and helped find solutions in a timely fashion for challenges such as parallelization I/O problems and memory-related segmentation errors encountered in this project. This will have a positive impact on the intern's future graduate studies or research career.

The main educational goal of this project is to expose the intern to methods of parallel computing. During the short session in Summer 2015, the intern learned the basics of various parallel programming languages like OpenMP and MPI, and also strengthened his skills in using C. Afterwards during the year, he was able to use this knowledge and apply it to trajectory modeling problems. While the integration of OpenMP into the trajectory model was essentially used to run more cases in the same time, it was a good introduction to parallel computing, and we hope to continue utilizing this tool for future studies of this nature.

Through this process the intern also greatly strengthened his ability to comfortably use Linux command terminals to communicate remotely with a supercomputer, a skill that will be extremely useful in upcoming research. Additionally, the intern has gained insight into trajectory modeling and general knowledge in spacecraft mission design and planning. The program generated for this project will be valuable in the intern's continuing research project to design a cost effective, human-crewed mission to Mars. The code (Version 1.0) was fully developed by the intern, and is available upon request.

6. CONCLUSION

Throughout the study we evaluated the use of parallel computing to assist in trajectory modeling. The most effective course of

action was to increase the volume of trajectories we could model, and generate tables to use offline. This method is like a higher mission level tool similar to how ephemerides would be used to plan a trajectory. Parallelization greatly increased the efficiency of the developmental stage of the model, as it saved time by speeding up testing of the Ward spiral code and results were obtained quickly for payload and launch date analyses. The ability to handle large amounts of data and organize it on Blue Waters saved a lot of disk space and computing time on the desktop computer.

From the study it is now possible to use the model to plan cargo missions to Mars in the concept study phase. The model can be used on any machine as long as the data and ephemerides are present in the package. With the input of thrust and mass in text file form, additional parameters that alter those two main variables can be added on without disrupting the original code. We can build on this template for future, higher-fidelity models.

7. LIMITATIONS AND FUTURE PROJECTS

The main limitation in this study was the assumption of full available power to the spacecraft at all times. The model accounted for an ideal transfer with the maximum thrust of the NEXT engine present throughout the trajectory. However, in an actual mission, the orientation of the solar panels and other factors like system reliability can come into play and alter the available power levels, thus changing the thrust of the engine unexpectedly. In addition, minute gravitational perturbations are not accounted for, and with Jupiter's large gravitational influence, it can alter the trajectory enough to require thrust changes for course correction. These extra factors are subjects we can explore in future studies that add to this model. We can still use the thrust and mass as inputs, but append extra calculations that will alter the thrust based on power levels and course corrections.

8. ACKNOWLEDGEMENTS

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications. We thank Therese Larson for her assistance in communication with Shodor and NCSA and Natalie Wolfenbarger for her help on generating ephemerides.

9. REFERENCES

- [1] Brophy, J. R., Noca, M. 1998. Electric Propulsion for Solar System Exploration. *Journal of Propulsion and Power* 14, 5, 700-707.
- [2] Curtis, H. 2013. *Orbital Mechanics for Engineering Students* 3rd edn. Butterworth-Heinemann, Oxford, United Kingdom.
- [3] Wertz, J. R., Everett, D. F., Puschell, J. J. 2011. *Space Mission Engineering: The New SMAD*. Microcosm Press, Hawthorne.
- [4] Bate, R. R., Mueller, D. D., White, J. E. 1971. *Fundamentals of Astrodynamics*. Dover Publications, Inc, New York, United States.
- [5] Hughes, K. M., Edelman, P. J., Saikia, S. J., Longuski, J. M.,

- Loucks, M. E., Carrico Jr., J. P., Tito, D. 2015. Fast Free Returns to Mars and Venus with Applications to Inspiration Mars. *Journal of Spacecraft and Rockets* 52, 6, 1712-1735.
- [6] Colasurdo, G., Casalino, L. 2003. Characteristics of Electric Propulsion Systems for Optimal Interplanetary Trajectories. In : *54th Annual Astronautical Congress of the International Astronautical Federation, the International Academy of Astronautics, and the International Institute of Space Law*, Bremen, Germany, 1-7.
- [7] Purvis, J. W. 1992. Closed Form Low-Thrust Trajectories for Mars Missions. In : *AIAA/SAE/ASME/ASEE 28th Joint Propulsion Conference and Exhibit*, Nashville, Tennessee, 1-4.
- [8] Rayman, M. D., Fraschetti, T. C., Raymond, C. A., Russell, C. T. 2005. *Preparing for the Dawn Mission to Vesta and Ceres.*, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California.
- [9] NASA Jet Propulsion Laboratory HORIZONS Web-Interface. In: *Jet Propulsion Laboratory Solar System Dynamics*. Available at: <http://ssd.jpl.nasa.gov/horizons.cgi>
- [10] Rimrott, F. P.J., Cleghorn, W. L. 2002. Orbit Transfer by Means of a Ward Spiral. *Technische Mechanik* 22, 4, 283-290.
- [11] Rimrott, F. P.J., Salustri, F. A. 2001. Open Orbits in Satellite Dynamics. *Technische Mechanik* 21, 3, 207-214.
- [12] Nah, R. S., Vadali, S. R., Braden, E. 2001. Fuel-Optimal, Low-Thrust, Three-Dimensional Earth-Mars Trajectories. *Journal of Guidance, Control, and Dynamics* 24, 6, 1100-1107.
- [13] U.S. Naval Observatory Julian Date Converter. In: *Naval Oceanography Portal*. Available at: <http://aa.usno.navy.mil/data/docs/JulianDate.php>