James Webb Space Telescope Initial Mid-Course Correction Monte Carlo Implementation using Task Parallelism

Jeremy Petersen[1], **Jason Tichy**[2], **Geoffrey Wawrzyniak**[3] **and Karen Richon**[4]
[1, 2, 3] *a.i. solutions, Inc., 10001 Derekwood Lane, Lanham, MD 20706, 301-306-1756, first.last@ai-solutions.com*
[4] *Code 595, NASA/Goddard Space Flight Center, Greenbelt, MD 20771, 301-286-5478, karen.v.richon@nasa.gov*

Keywords: *James Webb Space Telescope, Task Parallelism, Monte Carlo, Mid-Course Correction, Libration Point Orbit*

Scientific objectives for the James Webb Space Telescope (JWST) include identifying the first luminous sources to form and determining the ionization history of the early universe, investigating the assembly of galaxies and the birth of stars and proto-planetary systems, and determining the physical and chemical properties of planetary systems. Because of the sensitivity involved with observations of these types, JWST will be placed in orbit about the Sun-Earth L2 libration point (1.5e6 km from Earth), and its optical telescope element (OTE) will be pointed away from the Earth and the Sun. A large sun shield on the back of the observatory will block stray light from terrestrial and solar radio and light sources. The science phase of the mission is expected to last 10.5 years.

The observatory will be launched in late 2018 into a highly elliptical orbit that will not possess enough energy to achieve an orbit about the Sun-Earth L2 libration point. To make up that energy, three mid-course correction (MCC) maneuvers will be performed. The first maneuver, MCC-1a, is designed to be performed 12 hours after launch, depending on the launch-vehicle injection errors. The plan for MCC-1a is to execute 95% of the maneuver that nominally transfers the observatory into a Lagrange Point Orbit (LPO). This 5% knock-down accounts for unmodeled errors in the design and execution of the maneuver. Attitude requirements to prevent stray light from corrupting the OTE, as well as thruster placement on the spacecraft, effectively limit the propulsion system from removing energy from the trajectory. Because of the dynamics of the multi-body regime, an excess of energy would send the observatory into an unrecoverable state beyond the LPO. The second maneuver, MCC-1b, is a short burn two days after MCC-1a that notionally makes up the 5% difference. The final maneuver, MCC-2, occurs 30 days after injection and is necessary to correct for any errors remaining prior to entry into the LPO.

Propellant is necessary for stationkeeping and momentum-unload maneuvers during the science phase of the mission, as well as for the mid-course corrections necessary to deliver the observatory to the LPO from nominal and off-nominal injections over a variety of launch dates. At present, the propellant-budget analysis is separated into the transfer and the science phases of the mission. The transfer phase is the focus of this study. A Monte Carlo analysis that incorporates MCC maneuver modeling and execution is employed to validate the propellant budget. These types of analyses are often computationally expensive, but modern computing and parallel processing offer a solution to the computational-cost problem.

Modern computing employs two main types of parallelism. The most common type is data parallelism, where the same calculation is performed on the same types of data at the same time. The other type of parallelism is known as task parallelism, where each thread performs a completely different task on the same or different sets of data. Rarely employed for Monte Carlo studies in space-flight mechanics, task parallelism is exploited in this study of the JWST MCC propellant budget. First, each portion of the mission is represented with modular sets of code that can be changed for individual studies without compromising the overall integrity of the code. Second, task parallelism provides the ability to view the results as they are produced, one-by-one, rather than waiting for large sets of data to be produced. Finally, task parallelism allows for tasks to be broken down into small pieces, allowing nodes to opt in and out of the processes without losing large amounts of the computed data.

This paper is organized in four sections. The first is a summary of the JWST mission and trajectory design with an emphasis on the concept of operations for the MCC maneuvers. The second section focuses on subsystem-level trades used to determine the scope of the MCC design. Task parallelism as applied to the JWST MCC Monte Carlo analysis is presented in the third section. The final section highlights initial results from the MCC Monte Carlo analysis.