IEEE Aerospace Abstract Proposal (https://aeroconf.org/admin/manage/my-submissions)

Title (100 characters or less)

Mission Planning Simulation and Design Software Scaling for Shared and Distributed Memory Computing

Abstract Authors – Sam Siewert, Jacob Collins, Chirag Dhamange

Potential Additional Paper Co-authors: Stevie Littleton, Eric Galvan, Angel Martinez

Abstract (500 words or less)

Mission planning and simulation software architecture should ideally provide an open, scalable solution for a wide range of aerospace systems and allow for analysis and design optimization. Existing open-source mission planning and simulation tools such as GMAT (General Mission Analysis Tool), Nyx-space, Godot, ODtbx, and NOS3 provide propagators for space systems celestial mechanics and trajectory determination along with analysis and design tools. However, they all could be improved with parallel scaling software design patterns for propagation and optimization features. Specifically, use of the most current object oriented, and multi-paradigm programming language shared and distributed memory parallel programming methods used in the simulations as well as for optimization. This paper presents a software architectural pattern for use by the core simulation and optimization features of mission planning by providing a simple stand-alone example along with work to create a new prototype extension to existing open source.

Many mission planning, simulation and optimization objectives benefit from open-source parallel scaling, especially with Monte Carlo optimization, and problem scaling to simulate many celestial bodies and space vehicles, over long durations. Present tools either lack parallel scaling or use proprietary methods, often limited to one type of parallelism (e.g., shared memory). The goal of this investigation is to re-engineer the core propagation and Montel Carlo features for speed-up and scaling with portable parallel software engineering design patterns that integrate both shared memory and distributed memory parallel methods. While co-processing can also be used in addition, based upon lack of open solutions for co-processors such as GP-GPU (General Purpose – Graphic Processing Units), proprietary methods such as “CUDA” (Compute Unified Device Architecture), Advance Micro Devices “ROCm”, and Apple “Metal” are avoided in this study to avoid incorporating proprietary parallel software. Longer term, open co-processing methods such as OpenCL can be investigated, but here, we have found that a focus on open shared and distributed memory scaling provides significant advancement.

Our intent is to evaluate the relative merits of combining OpenMP (shared memory) and MPI (distributed memory) to extend existing mission planning and simulation tools. Specifically, the tools benefit from scaling for Monte Carlo analysis and multi-body simulation supported by current supercomputing machines which have millions of distributed memory cores. Not only has the trend for mission design and analysis tools been incorporation of proprietary methods such as Intel TBB (Thread Building Blocks) in extensions to GMAT, but also full replacement of open-source tools like GMAT with proprietary tools like ANSYS System Toolkit for mission design. Preserving open-source mission design and analysis tools that are competitive in terms of performance, scaling, and features is beneficial to the broader space science community by providing full source options that are portable and extensible. The open-source design pattern will be presented by example in a small-scale simulation along with

experience using this pattern in existing larger open-source mission design and analysis tools. Results showing comparison of runtimes for sequential compared to parallel as well as existing proprietary threaded runs are presented and analyzed in this paper.

Full References

1. Conway, Darrel J., and John McGreevy. "Using the GMAT Application Programmer's Interface." *Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference, AAS Paper*. 2020.
2. Restrepo, Carolina, and John Hurtado. "Pattern recognition for a flight dynamics Monte Carlo simulation." *AIAA Guidance, Navigation, and Control Conference*. 2011.
3. Pirayeshshirazinezhad, Reza, et al. "Designing Monte Carlo simulation and an optimal machine learning to optimize and model space missions." *IEEE Access* 10 (2022): 45643-45662.
4. Harris, Gage W., Ping He, and O. O. Abdelkhalike. "A coupled spacecraft system and trajectory optimization framework using GMAT and OpenMDAO." *AAS/AIAA astrodynamics specialist conference*. 2022.
5. Conway, Darrel J. "Paramat: Parallel Processing with the General Mission Analysis Tool." *AIAA/AAS Astrodynamics Specialist Conference. AIAA/AAS*. 2015.
6. https://github.com/nyx-space/nyx
7. https://odtbx.sourceforge.io/
8. https://github.com/nasa/nos3
9. https://godot.io.esa.int/docs/1.4.0/
10. Branch, Mission Design. "ODTBX: Orbit Determination Toolbox." *Astrophysics Source Code Library* (2018): ascl-1810.
11. Grubb, Matthew, et al. "Nasa operational simulator for small satellites (nos3): Tools for software-based validation and verification of small satellites." (2016).
12. https://ntrs.nasa.gov/citations/19760017203

Consider submission for session: 10.03

Track 10 Software and Computing

Kristin Wortman
kristin.wortman@jhuapl.edu

Principal professional staff, Space Exploration Sector's Space Mission Assurance group at Johns Hopkins University Applied Physics Laboratory located in Laurel, MD. B.S., Computer and Information Science; M.S., Software Engineering, University of Maryland University College.

Virgil Adumitroaie
virgila@jpl.nasa.gov

Data Scientist, JPL. Working on planetary atmospheric and magnetospheric modeling. Past research in high-speed turbulent combustion modeling, data dimensionality reduction, neural networks, decision support, climate data assimilation, and scientific software development. Ph.D., ME, University at Buffalo.

Session 03 Software Architecture and Design

Appropriate software architecture is critical to the design, development and evolution of all software systems, and its role in the engineering of software-intensive applications in the aerospace domain has become increasingly important. This session solicits novel ideas on the foundations, languages, models, techniques, tools, and applications of software architecture technology. Topics include software architecture for space mission systems; architecture across software, system and enterprise boundaries; architectural patterns, styles and viewpoints; architecture frameworks; design reasoning, capturing and sharing design decisions; and open architectures, product-line architectures, and systems of systems software architects' roles and responsibilities.

Martin Stelzer **martin.stelzer@dlr.de**
Research Associate, German Aerospace Center (DLR)
Peter Lehner **peter.lehner@dlr.de**
Team Lead Mobile Manipulators, German Aerospace Center (DLR)