GPU-accelerated Optimization of CubeSat Constellation Design Considering Cloud Cover Uncertainty

1. Background

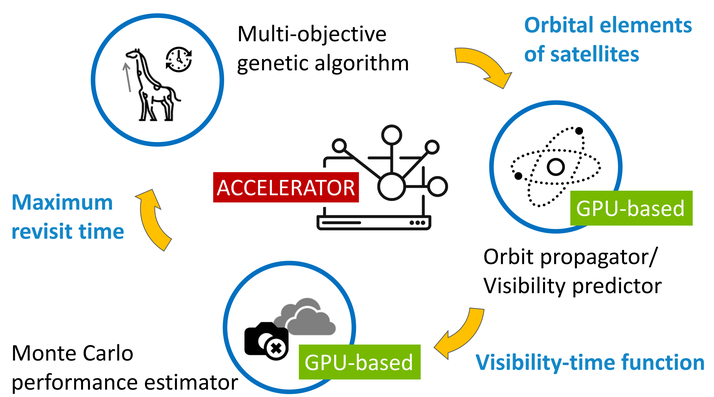
- Constellation of CubeSats has potential to provide comparable or even greater capabilities than a single large satellite but with lower cost and higher robustness.
- To design such a CubeSat constellation entails solving large-scale multi-objective optimization problems, but the computation time is not reasonable with existing tools. One promising way to address this issue is probably the GPU computing.
- Cloud cover will dramatically limit the number of images that can be taken and therefore can seriously affect scientific or commercial returns of satellite constellations. However, this uncertainty has seldom been considered during the design phase.

2. Problem Statement

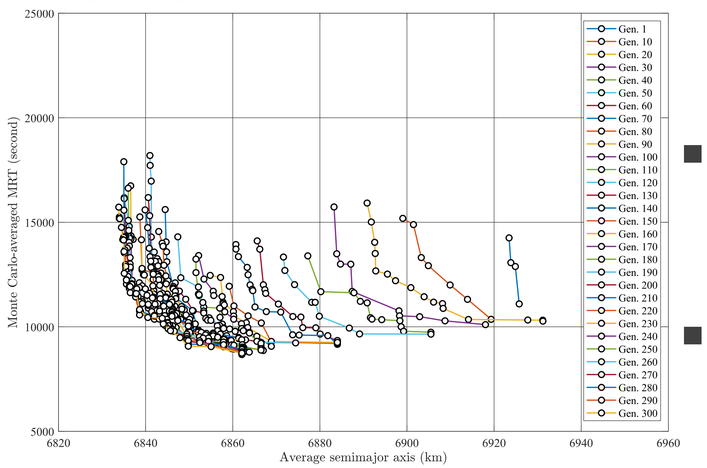
The aim of this study was to develop a design tool, called ACCELERATOR, which can be employed to characterize the performance of constellations (may contain hundreds or thousands of satellites) and to help reduce the burden on design engineers in the mission analysis and design phase of future constellation missions.

Advantages of the tool are threefold:

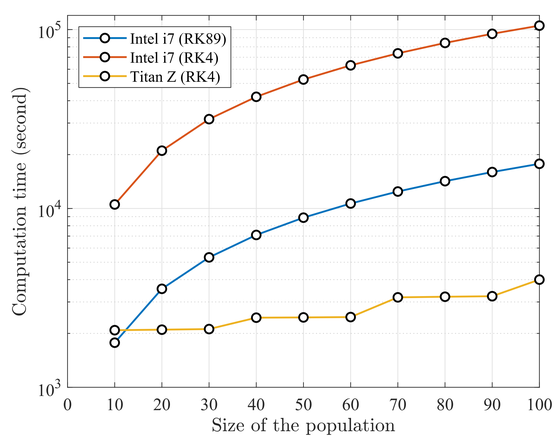
- It incorporates a sufficiently high fidelity orbit propagator which takes into account necessary orbital perturbations and therefore can give fairly accurate state information of satellites.
- It adopts parallel computing techniques to speed up the computation process and make the optimization of large-scale constellations more tractable.
- It produces robust constellation design by dealing with the limiting factor (cloud cover uncertainty) of satellite operations while in the design phase.



3. Results



- Bi-objective optimization: spatial & temporal resolution
- 100 constellations in one generation; each constellation has 150 CubeSats
- Optimized over 300 generations; figure shows their Pareto fronts



- Computation time for different hardware and integrator
- Titan Z can ultimately achieve 7.58 times faster then Intel i7, if size is set to 600

4. Conclusion

- ACCELERATOR helps the designers evaluate performance of constellations and gives Pareto-optimal frontiers.
- The self-developed, GPU-based orbit propagator indeed mitigates the bottleneck, which is the long computation time required for doing orbit propagation and determining the fitness values, in the optimization of satellite constellations.
- Robustness of resulting optimal constellations against cloud cover is improved since the uncertainty has been introduced during the optimization process.

Hsiu-Hsuan Huang, Hyochoong Bang
Korea Advanced Institute of Science and Technology (KAIST)
ernest.huang@kaist.ac.kr, hcbang@kaist.ac.kr

SSC20-WP1-33