Erkin Verbeek
18.335 - Spring 2024

# Final Project Proposal

I propose a project that covers the Fast Multipole Method (FMM) for the n-body problem, with a special focus on the High-Performance Computing aspects of the current solution space. Solutions to the n-body problem are relevant to various fields, but are particularly relevant to my current master's research focus of using HPC principles to improve cosmological simulations. My current work revolves around using GPUs to improve the performance and scalability of large-scale simulations.

From a simplified perspective, solutions to the n-body problem are calculation schemes that estimate the inter-body force values exerted between n bodies in 3D space. To directly calculate all the forces on a single body, one would need to consider the given body's interaction with all n-1 other bodies. To get the forces over all bodies, this calculation would need to be repeated once for each body, resulting in $O(n^2)$ scaling. The Fast Multipole Method improves on this scaling significantly. FMM achieves a general scaling of O(n) by, in essence, treating closely clustered objects as one single object. In my project I will understand and explain where this scaling improvement comes from much more deeply.

There have been some interesting recent developments in the n-body solution space related to leveraging GPUs to further reduce the computation times. As someone that works with GPUs to improve large-scale simulations, I am very interested in learning about how current algorithms utilize GPUs to further improve simulation performance.

I plan to implement the direct algorithm and FMM in C++. If I am successful early enough then I plan to implement GPU improvements with CUDA & C++. I am very familiar with both of these languages.

I plan to compare FMM scaling to the direct $O(n^2)$ algorithm through plotting computing times over various N. I also intend to perform strong and weak scaling tests. I plan to analyze FMMs behavior under ideal and non-ideal scenarios. Lastly, I would like to see how a GPU version of FMM affects any of these results.

Following are some initial references that I have gathered and plan to reference:
n-body problem / Barnes-Hut:

- http://portillo.ca/nbody/

FMM:

- https://aip.scitation.org/doi/pdf/10.1063/1.4773727
- https://comp-astrophys-cosmol.springeropen.com/articles/10.1186/s40668-014-0001-7

GPU-parallelizing:

- PhotoNs-GPU: A GPU accelerated cosmological simulation code: https://iopscience.iop.org/article/10.1088/1674-4527/21/11/281/meta
- Fast multipole methods on graphics processors: http://users.umiacs.umd.edu/~ramani/pubs/Gumerov_Duraiswami_GPUFMM_JCP_2008.pdf
- Scalable Fast Multipole Methods on Distributed Heterogeneous Architectures: http://users.umiacs.umd.edu/~ramani/pubs/Qi_Gumerov_Duraiswami_SC11_FMM_Heterogeneous.pdf