

**ASTEROID SHAPE MODELING WITH CUDA.** S. P. Levensgood<sup>1</sup> and M. K. Shepard<sup>1</sup>, C. Magri<sup>2</sup>, M. Nolan<sup>3</sup>, Department of Environmental, Geographical, and Geological Sciences, Bloomsburg University, Bloomsburg, PA 17815, [sp106880@huskies.bloomu.edu](mailto:sp106880@huskies.bloomu.edu), [mshepard@bloomu.edu](mailto:mshepard@bloomu.edu), <sup>2</sup>University of Maine at Farmington, Farmington, ME 04938, <sup>3</sup>Arecibo Observatory, NAIC, Arecibo, PR 00612.

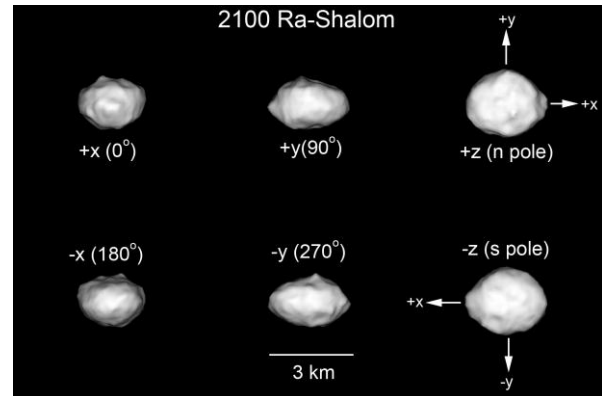
**Introduction:** We modified the existing asteroid SHAPE software package [1,2] to take advantage of modern graphics processing units (GPUs) on Compute Unified Device Architecture (CUDA) platforms, allowing it to generate simulated radar images faster. Data gathered from observations of 2100 Ra-Shalom will be used to test the implementation as well as provide an improved shape model.

**Discussion:** The SHAPE software [1,2] uses a constrained-least-squares model to attempt to find a “best-fit” 3D shape model for a given set of asteroid radar and lightcurve data. In the modeling process, an asteroid is represented by three successively more complex models. An ellipsoid is the simplest shape with the fewest fitting parameters and is first fit to the data to constrain the rough size, aspect ratio, and rotation state of the asteroid. This is followed by a low order spherical harmonic model which can incorporate deviations from the ellipsoid in the gross shape; the cost is more parameters to fit and a consequent slowdown in the fitting process. If the data quality allow it, a vertex model is the final stage of fitting and smaller shape anomalies can be accommodated. With high resolution data sets, hundreds or thousands of vertices can be incorporated, but again, at an increasing cost in computing time. At each stage of this process, the software cycles through each parameter that is not held constant and searches for the “best-fit” with regard to the data.

For each radar delay-Doppler image to be modeled, the software creates a simulated optical plane-of-sky image which is used as an intermediate to generate simulated radar data. Each model at this stage is treated as a vertex-polyhedron solid. For each pixel in the simulated image, SHAPE determines which triangular surface facet would be projected at that point and the pixel is assigned a radar scattering value based on the properties of that facet. After the simulated data is determined, it is then compared to the actual data to obtain a chi-square value. The process is repeated for image in the data set and a total chi-square value is computed. The process continues while the total chi-square continues to decrease.

The process that SHAPE uses is computationally intensive, and can take days to run on single CPU machines. But because of the nature of the calculations, the process is readily parallelized. Currently, this is done by distributing the data set across different CPUs,

then adding the chi-square values for each individual sum.



**Fig. 1.** Shape model of (2100) Ra-Shalom based on radar data in 2000 and 2003. After [3].

The most time consuming part of the process is rendering - determining which triangular facet is associated with each pixel. This task is easily parallelizable because each individual calculation is not dependant on the results of another. In other words, determining the triangular surface facet associated with one pixel does not require any knowledge of which triangular surface facet is associated with another pixel. Thus, the more of these calculations that can be done simultaneously, the faster this stage of the process can be. A natural fit for this type of calculation is the modern graphics processing units (GPUs) cards that have been adapted for scientific computing. This is ideal since GPUs were originally designed specifically for determining association between individual pixels and geometric objects.

Because of the inversion is always an underdetermined problem, many shapes can often be found that fit the data equally well. Thus, subjective input by the user is required at multiple stages, slowing the process further. With current computational limitations, users must be careful about initial parameter choices to avoid wasting processing time. Improving the simulation speed will allow users to test a wider range of parameters and, in some instances, allow nearly real-time feedback of those tests.

We have modified the rendering functions in SHAPE and added support for Nvidia CUDA technology, allowing it to take advantage of the processing capabilities of modern GPUs. This should result in simulations that complete much faster. Additionally,

because only the render functions require significant modification, it should still be possible to distribute the data set among multiple computers, allowing the software to benefit from a cluster containing GPUs.

**2100 Ra-Shalom:** In addition to modifying the SHAPE software, the observational data obtained in [3] will be used to generate a new model of 2100 Ra-Shalom that takes into account more recent optical observations, pole solutions, and newly-discovered retrograde rotation [4].

**References:** [1] S. Hudson (1993), *Three-dimensional reconstruction of asteroids from radar observations*. Remote Sensing Rev. 8, 195-203. [2] C. Magri (2007), *Radar observations and a physical model of Asteroid 1580 Betulia*. Icarus 186, 152-177. [3] M. K. Shepard (2008), *Multi-wavelength observations of Asteroid 2100 Ra-Shalom*. Icarus 193, 20-38. [4] Farnocchia et al. (2012) AAS DPS meeting #44, Abs. 305.02.