Data Compression and Transformation with Wavelets

As datasets become increasingly large, there is a need to store the data in a compressed form. In some instances, data can be transformed in a lossy manner, yet still retain most of the original characteristics. After reconstituting the data, further processing would yield results similar enough to the original that they remain useful. Wavelets are one possible method that can be used to perform such a transformation. Introduced relatively recently, wavelets are able to capture frequency and time data, providing for a new kind of signal processing. Wavelets have found many uses, one of them famously in image compression with the JPEG format.

The goal of the project is to use wavelets to compress 2-D and 3-D datasets, namely data generated from the Navier-Stokes equations, in order to determine whether the compressed and reconstituted data can still be reliably processed without a significant loss of important information. There are two dependent types of data in question, velocity and vorticity, which can be derived from each other. Compressing the data and then comparing the error to the originals will give a sense of how useful wavelets can be for this data set. Additionally, it may be possible to perform calculations directly on the compressed data, an option that will be explored. Wavelets are comprised of some operations that can be run in parallel. Ideally, the compression of the data will be able to be done in parallel. One of the goals of the project will be to use a threaded architecture at first that will lend itself to be transcribed into a multi-node setup that can run quickly.

Milestones:
1. May 1st – Have explored operations using Matlab and determined error margins when working with compressed data.
2. May 15th - Wavelet compressor is coded in C/C++ that can read in the Navier-Stokes data and output a smaller file. Ensure that the data is valid and similar results can be obtained from the data by checking error margins.
4. June 2nd - Extend to multiple computers through a simple XML based message passing system that will process individual segments and return to a coordinator node.