

## **Paper Review: MCRENGINE**

### *Summary*

Similar to the previous paper about FTI, this paper addresses the issue of high overhead caused by large checkpoints being stored in a PFS. Instead of using a multi-level technique like FTI, MCRENGINE aggregates checkpoints from multiple application processes, organizes the data into common groups, and then compresses it. This compression technique requires more computation than normal compression, but the scheme improves compressibility of checkpoints up to 115% and reduces checkpoint overhead by up to 87%.

### *Strengths*

I thought that the authors did a good job of presenting the problem and convincing the reader that it is an important issue in the field of HPC. There are lots of figures and tables that provide excellent visualizations and data that supplements the meat of the paper very well for those who take the time to understand them fully. I like that they used 4 different simulations (ALE3D, Cactus, Enzo Cosmology, Enzo Implosion) to compare their tool with different applications.

### *Shortcomings*

I found that many of the subsections were long and cumbersome to get through (not concise). There seemed to be lots of disparate parts to this paper but in many cases it was difficult to get a feel for how everything worked together in this tool. There were a few *very* minor typos and grammatical errors throughout the paper... this isn't a huge deal but I got confused a few times and had to reread the section to understand what they actually meant.

### *Improvements*

Break up long sections/subsections into more modular pieces. Provide a more cohesive explanation about how all of the pieces work together to make this tool successful. Try to organize figures and tables in a way that is less jumbled and therefore easier on the eyes.

### *Question(s)*

Could you explain the relationship between CNC and ANC more clearly?