ID: 1.6.1b

Title: Resiliency in massively parallel quantum simulations

Name: Schulthess, Thomas

Affiliation: ETH Zurich

In recent years, a number of quantum-based simulation codes were developed achieving sustained petaflops performance on supercomputing systems [1]. Production runs with 200+ thousand cores for job lengths of 12 to 24 hours are common. However, much longer jobs that run for days, or on even larger computer systems, with ten or hundred times the concurrency of today?s peta-scale systems, will likely fail because of individual node failures. Today, this situation is dealt with using checkpoint-restart methods, where the state of the simulation is regularly written to disk (checkpoint), and in case of a failure only needs to be restarted from the last completed checkpoint. A more appropriate method is to build fault tolerance directly into algorithms and have simulations continue to run when nodes fail. We show how the Monte Carlo algorithms used in DCA++, a quantum Monte Carlo code, have to be modified to allow for a fault tolerant implementation, and we discuss the implementation of the production code in terms FT-MPI [2] as well as a preliminary implementation of the run-though stabilization proposal to the MPI standard [3]. We will also discuss modifications to the distributed Bethe-Salpeter equation solver used in TD-DFT and self-consistent GW calculations. In this particular case, redundant storage and recovery techniques had to be applied. [1] See for example the Gordon Bell Prize winners and finalist of the ACM/IEEE SC08, SC09, and SC10 conferences. [2] G. E. Fagg et al., Extending the MPI Specifications for Process Fault Tolerance on HPC systems, Proc. Of the ISC2004 conference, Heidelberg 2004. [3] J. Hursey et al., Run-though Stabilization Interfaces and Semantics (https://svn.mpi-forum.org/trac/mpi-forum-web/wiki/ft/run_through_stabilization, 2011.