CPU-GPU Parallel Implementation of a Time Domain Volume Integral Equation Solver for Characterizing Large Scale Electromagnetic Wave Interactions

Project description:

A distributed memory scheme and an unstructured mesh partitioning technique has recently been developed for the efficient parallelization of a time domain volume integral equation (TDVIE) solver [1-5]. It has been shown that the parallelized solver exhibits almost ideal scalability, when executed on Shaheen - an IBM BG/P supercomputer [2-5].

The parallelization scheme can employ a standalone MPI or a hybrid MPI/OMP strategy that rely on the so called rotating-tiles paradigm [4, 5]. The major attribute of this parallelization scheme is that all data exchange is executed using local-based inter-process communications [4, 5].

In the development of the parallelization strategy of the TDVIE solver, the GPU implementation is to be exploited next. To date, a proof of concept utilizing one GPU is readily in place: producing speed-ups in the order of 30-50 folds when compared to a single core CPU. The measured speedup depends on the GPU hardware and the number of degrees of freedom being used. For the further exploitation of the GPU technology for the TDVIE solver, we propose the following.

In the first phase:

a) The implementation of a shared-memory parallelization scheme for the full computation of the solver, which makes use of the existing x86 resources along with multi-GPU boards.

b) The integration of a dynamic runtime scheduler (StarPU, OmpSS) in order to manage the workload distribution and host-devices data movement.

In this phase, the elimination of any load imbalance that may appear as a result of the hybrid CPU-GPU parallelization is addressed. In addition, this overall task-based parallelism will provide the suitable strategy required to solving electrically large problems. In particular, in the case where the memory required for the complete solution of a given problem is greater than the available resource of the combined memory of all GPUs residing in the system (out-of-core).

In the second phase: the extension of the parallelized solver to accommodate for the use of distributed memory systems with multi-GPUs is to be implemented. In this scenario, a combined MPI - at the higher level - and the task-based programming model from phase I, within each node, is to be applied.

Keywords: MPI, CUDA, OpenACC, Dynamic runtime systems

References:

1. A. Al-Jarro, M. A. Salem, H. Bağcı, T. M. Benson, P. Sewell and A. Vukovic, "Explicit solution of the time domain volume integral equation using a predictor-corrector scheme", IEEE Trans. Antennas and Propagat., vol. 60, no. 11, 2012.
2. A. Al-Jarro, and H. Bağcı "A hybrid MPI/OpenMP parallelization scheme for explicit solution of the Volterra integral equation" International Workshop on Computational Electromagnetics (CEM'11), Izmir, Turkey, Aug. 10-13, 2011.
3. A. Al-Jarro, M. Cheeseman, and H. Bağcı, "Distributed-memory parallelization of an explicit time-domain volume integral equation solver on Blue Gene/P," Applied Computational Electromagnetics Society (ACES) Journal, vol. 27, no. 2, pp. 132-144, 2012.
4. A. Al-Jarro, M. Cheeseman, and H. Bağcı, "A distributed-memory parallelization of the explicit time-domain volume integral equation solver using a rotating tiles paradigm," ACES, Ohio, USA, April, 2012.
5. A. Al-Jarro, M. Cheeseman, and H. Bağcı, "An unstructured mesh partitioning and a memory efficient hybrid MPI/OpenMP parallelization scheme for the explicit marching-on-in-time solution of the time domain volume integral equation," Journal of Computational Physics, *to be submitted,* 2012.