SAND2015-1604C

Assembly Algorithms for PDEs with Uncertain Input Data on Emerging Multicore Architectures

Sandia National Laboratories
E.T. Phipps and H.C. Edwards
Sandia National Laboratories, Albuquerque 87185

Problem

Uncertainty quantification (UQ) is an important scientific driver for pushing to the exascale, potentially enabling rigorous and accurate predictive simulation for many problems that are intractable today.

Nearly all UQ approaches repeatedly sample deterministic simulation codes at different realizations of the input data, resulting in performance limited to that of each realization.

Many PDE simulations do not achieve high performance on multicore (CPU/GPU/Accelerator) architectures due to:

- Random, uncoalesced memory accesses
- Inability to exploit consistent vectorization

Approach

In many cases, the code path, processor instructions, and memory access patterns are very similar from realization to realization.

Idea: Propagate a collection of samples (ensemble) together through the forward simulation:

FEM Residual Equations: $f(u,y) = 0 \implies F(U,Y) = 0$

$$U = \sum_{i=1}^{m} u_i \otimes e_i, \quad Y = \sum_{i=1}^{m} y_i \otimes e_i, \quad F = \sum_{i=1}^{m} f(u_i, y_i) \otimes e_i$$

Ensemble PDE Solution | Input Data Ensemble | FEM Residual Ensemble

- Each sample-dependent datum becomes a small array
- Increases fine-grained parallelism: Each sample within ensemble can be assigned to a vector lane/CUDA thread
- Improves memory access patterns: Random memory accesses become block accesses (coalesced/packed)
- Enables sharing of non-sample-dependent data (e.g., mesh) between samples to reduce memory bandwidth
- Amortizes MPI communication latency across ensemble

Apply to C++ PDE codes via template-based generic programming:

- Template assembly on scalar type
- Instantiate template code on ensemble scalar type
- Ensemble scalar type implements all relevant operations using SIMD/SIMT parallelism
- Use *thread team interface* for kernel launch and functor

Approach implemented by **Stokhos** embedded uncertainty quantification library on top of **Kokkos** portable manycore performance library (H.C. Edwards, D. Sunderland, C. Trott).

Results

Idea prototyped within matrix, RHS assembly for 3-D uncertain, nonlinear diffusion equation:

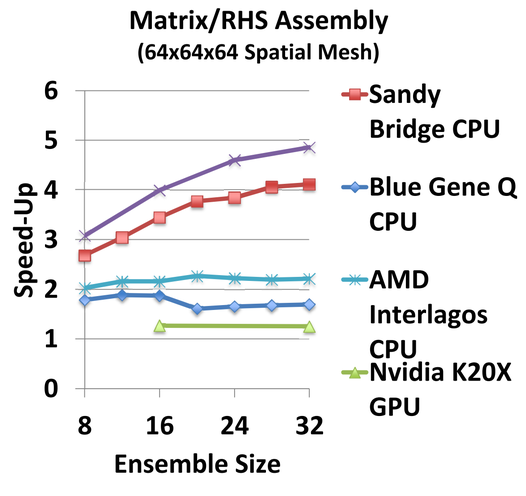
$$-\nabla \cdot (\kappa(x,y)\nabla u) + u^2 = 0, \;\; \kappa(x,y) = \kappa_0 + \sigma \sum_{i=1}^{M} \sqrt{\lambda_i}\kappa_i(x) y_i$$

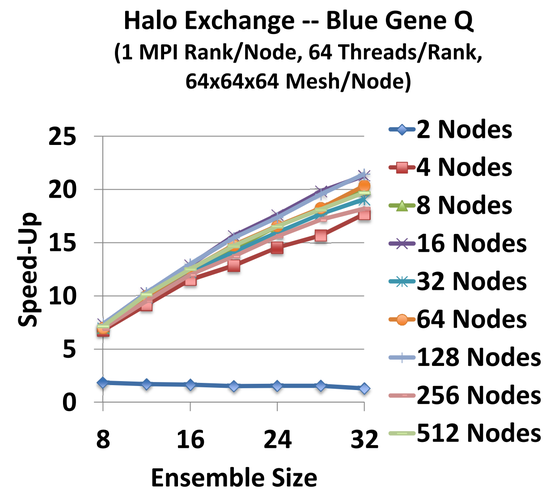
Cost of ensemble assembly compared to propagating a single sample at a time on contemporary multicore architectures:

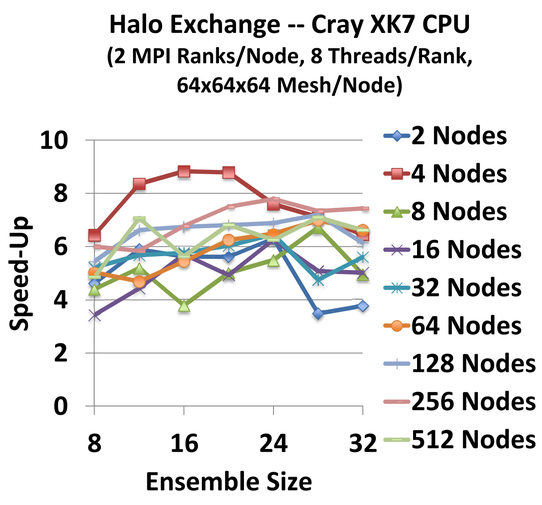
- Intel Sandy Bridge CPU (8 cores, 16 threads)
- IBM Blue Gene Q CPU (16 cores, 64 threads)
- AMD Interlagos CPU (16 cores, 16 threads)
- NVIDIA Kepler K20x GPU
- Intel Xeon Phi 7120p (60 cores, 240 threads)

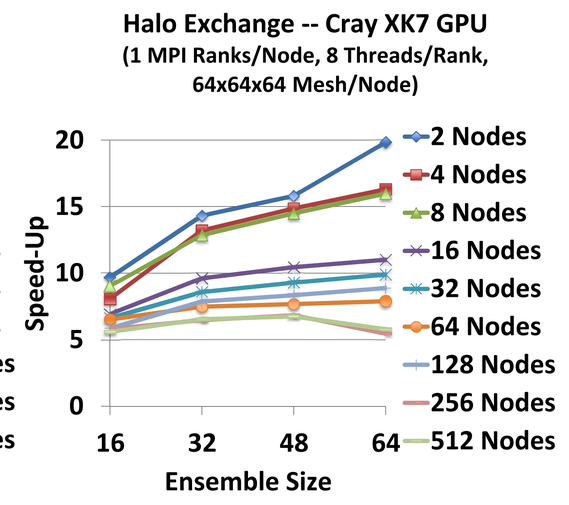
and interconnects:

- IBM Blue Gene Q
- Cray XK7 (CPU-CPU and GPU-GPU)









Significance

- Results demonstrate significant improvements in assembly performance for all multicore architectures and interconnects
- Similar speed-ups achieved for sparse iterative linear system solvers (e.g., CG, GMRES)
- Enables further speed-ups by sharing preconditioners across the ensemble
- A significant challenge is developing algorithmic approach for grouping samples in a real UQ problem to take most advantage of similarity in simulation process through ensemble

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

SAND 2014-xxxxP