

---

Eric Phipps  
**Preliminary Investigations of Multi-Core Accelerated  
Embedded Uncertainty Quantification**

Sandia National Laboratories  
PO Box 5800 MS-1318  
Albuquerque NM 87185  
`etphipp@sandia.gov`  
H. Carter Edwards  
Jonathan Hu

Quantifying uncertainties in computational simulations is a key challenge for predictive computational science. An important task within uncertainty quantification is the propagation of input data uncertainty to the corresponding simulation output quantities of interest. While many uncertainty propagation approaches have been investigated throughout the literature, one approach of interest is the embedded stochastic Galerkin method. Unlike sampling-based methods that only require evaluating the forward simulation at an independent set of realizations of the uncertain input data, and thus decouple the corresponding deterministic and stochastic discretizations, stochastic Galerkin methods couple these degrees-of-freedom together. For a given level of accuracy, they trade fewer stochastic degrees-of-freedom for the additional burden of formulating and solving the resulting coupled (and much larger) system. When applied to partial differential equations (PDEs) with random input data, they have generally been found to be efficient for certain types of linear PDEs, but inefficient for PDEs with nonlinear dependence on either the random variables or the unknown solution. This difference is attributed to the often dramatically greater number of non-zeros in the resulting stochastic Galerkin operator for the nonlinear case.

The stochastic Galerkin operator has a Kronecker product structure arising from the stochastic and deterministic discretizations, and thus it is typically organized hierarchically as a block-operator where the sparsity between the blocks is dictated by the stochastic discretization and each block has the structure of the deterministic discretization. This allows for greater code reuse when modifying a code to implement the stochastic Galerkin method, as well as reuse of solvers and preconditioners designed for the deterministic problem. However, as described above, this block operator is much denser in the nonlinear case, and can even be fully block-dense. Therefore in this work, we investigate commuting this layout to arrive at a sparse block operator arising from the deterministic discretization where each (possibly dense) block has the structure of the stochastic discretization. We investigate the amenability of this layout to emerging multi-core architectures containing a high-degree of on-node shared-memory parallelism. We develop several approaches for implementing

the stochastic Galerkin matrix-vector product using threading for the stochastic blocks, and study their performance on contemporary multi-core CPUs and many-core GPUs.