Dungeon Session Application:

SAND2016-1041PE

1. Target problem and sample input decks

There are two categories of tests of interest for the Intel dungeon session scheduled for February 16-18, 2016. The first one deals with sparse linear solvers (ic-test executable) and the second one with the BDDC domain decomposition preconditioner/solver (ic-test-bddc executable). Instructions on how to run the tests and example input decks are described in the README_TESTS.txt file (see tar file provided).

2. Create A Profile Of Your Application for the Sample Input Deck

We intend to gather profiling information during the dungeon session. A copy of a recent presentation showing the performance of various linear solvers on different platforms is provided.

3. Can kernels be created out of the hotspots in your code?

Kernels are self-contained code snippets intended to represent a single nodes work for a bigger problem run at scale.

The most important kernel for domain decomposition solvers/preconditioners is the sparse linear solver. We hope to use the ic-test executable to gain insights into improving the performance of our own linear solvers as well as how to best use MKL/Pardiso. The ic-test-bddc executable itself exercises three basic kernels for domain decomposition solvers. That is, sparse matrix vectors products, sparse linear solvers, and dense linear algebra (plan is to depend on MKL/BLAS for dense linear algebra).

4. Provide MPI vs MPI + OpenMP Scaling Curve for your sample input deck or hotspot kernels

5. How much memory does your sample input run consume? How will this change between your sample input and 2017 target problem?

6. Is your application sensitive to changes in memory bandwidth?

7. Is your application sensitive to changes in clock speed?

8. What is your vector utilization?

We expect to achieve significant vector utilization by using MKL/BLAS in at least two of the basic kernels involving dense linear algebra operations.

9. IO + Communication