AMGmk

Summary Version

1.0

Purpose of Benchmark

This microkernel contains three compute-intensive sections of the larger AMG benchmark. Optimizing performance for these three sections will improve the figure of merit of AMG.

Characteristics of Benchmark

AMGmk, like the full AMG benchmark, is written in C. The three sections chosen to create this benchmark perform compressed sparse row (CSR) matrix vector multiply, algebraic multigrid (AMG) mesh relaxation, and a simple a * X + Y vector operation. OpenMP directives allow additional increases in performance.

Limitations of Benchmark

AMGmk uses no MPI parallelism and is meant to be studied as a single-CPU benchmark or OpenMP benchmark only. The run time of this benchmark is not linearly related to the figure of merit of the larger AMG benchmark because the exact proportion of time spent performing these three operations varies depending on the size of the problem and the specific linear system being solved.

Mechanics of Building Benchmark

One Makefile is used to build the code. It will require manual modifications (such as compiler, compiler flag, etc.) prior to attempting to build the code. The executable created is named AMGMk.

Mechanics of Running Benchmark

No input decks or command-line arguments are required. Simply execute AMGmk

```
./AMGMk > myoutput
```

where `myoutput` is an arbitrarily named output file. The variable named “testIter” in main.c is normally set to 500, but it can be adjusted to any value desired.

Verification of Results

An error message is printed if the answer is not correct. A sample output file “sample_output” is included in the tar file. The timings in this file are from running on one of LLNL’s POWER5 systems.

(UCRL-CODE-222953)