EXECUTIVE SUMMARY

Continued increases in the performance of large-scale systems will come from greater parallelism at all levels. At the node level, we see both in the increasing number of cores per processor and the use of large numbers of simpler computing elements in general purpose GPUs. The largest systems must network tens of thousands of nodes together to achieve the performance required for the most challenging computations.

Successfully using these systems requires new algorithms. In the last year, we have further improved a new communication model that better fits the performance of multicore nodes to develop new algorithms for sparse matrix-vector products and better understand the behavior of nonblocking algorithms for the Conjugate Gradient method. We also developed a simple implementation of the MPI Cartesian topology routines that significantly outperforms the available implementations and several parallel I/O libraries.

RESEARCH CHALLENGE

This work directly targets current barriers to effective use of extreme-scale systems by applications. For example, Krylov methods such as Conjugate Gradient are used in many applications of extreme-scale systems by applications. For example, Krylov solver performance at scale, to develop more accurate and scalable Krylov solvers for structured grid problems. This includes developing code for measuring and processing parallel runtimes and network performance counters, developing a collection of kernels relevant to structured grid problems, and developing performance models with penalty terms that accurately model parallel performance at scale. Eller has run experiments to determine parameters for the performance models and performed scaling tests for the parallel communication kernels and scalable conjugate gradient solvers. He is currently designing and running experiments with optimizations designed to improve the performance of these kernels on Blue Waters. He has also done some initial work on studies to investigate the impact of network noise on solver performance and to investigate the performance of scalable Krylov solvers within a quantum chromodynamics application. These experiments have helped to better understand Krylov solver performance at scale, to develop more accurate performance models, and to optimize the solvers to obtain better performance.

Ed Karrels, working with William Gropp, has been testing a variety of input/output access patterns and developing tools for improving input/output performance. These tools include:

- **MeshIO**—a tool for reading and writing N-dimensional meshes in parallel. This is being used by the XAPC project, sponsored by the U.S. Department of Energy, to accelerate job startup and shut-down, and is being evaluated by two other science teams;
- **Zchunk**—a tool for reading compressed binary data at arbitrary line offsets efficiently. This is being used by bioinformatics researchers on genomic data;
- **Zchunk and Zlines** are available at https://github.com/oshkosher/meshio.

Finally, William Gropp has developed a new algorithm for implementing process mapping for Cartesian grids, which is needed for many applications that use structured grids. MPI provides a convenient routine for this operation, but few MPI libraries provide a good implementation of this operation. As a result, applications must either forgo the performance or use ad hoc, nonportable techniques to achieve a good mapping. Such tools do exist for Blue Waters, but they do not provide the right solution to the problem. Applications should be able to rely on the features in MPI and not need to use nonstandard, nonportable methods.

In addition, by using insight gained from our new performance model, we developed an alternative implementation of MPI_Cart_create that provides a significant performance benefit, as shown in Fig. 2.

WHY BLUE WATERS

Scalability research relies on the ability to run experiments at large scale and requires tens of thousands of nodes and hundreds of thousands of processes and cores. Blue Waters provides one of the few available environments where such large-scale experiments can be run. In addition, Blue Waters provides a highly capable I/O system, which we used in developing improved approaches to extreme-scale I/O.

PUBLICATIONS & DATA SETS


Zchunk and Zlines are available at https://github.com/oshkosher/meshio.

MeshIO is available at https://github.com/oshkosher/meshio.

The improved implementation of MPI_Cart_create is part of baseenv and is available from wgropp@illinois.edu.