PDE Preconditioner Resilient to Soft and Hard Fault

F. Rizzi*
Sandia National Laboratories
Livermore, CA, USA
fnrizzi@sandia.gov

K.V. Morris
Sandia National Laboratories
Livermore, CA, USA
knmorri@sandia.gov

K. Dahlgren
Sandia National Laboratories
Livermore, CA, USA
kmdahlg@sandia.gov

K. Sargsyan
Sandia National Laboratories
Livermore, CA, USA
ksargsy@sandia.gov

P. Mycek
Duke University
Durham, NC, USA
paul.mycek@duke.edu

B.J. Debusschere
Sandia National Laboratories
Livermore, CA, USA
bjdebus@sandia.gov

ABSTRACT

We present a resilient domain-decomposition preconditioner for partial differential equations (PDEs). The algorithm reformulates the PDE as a sampling problem, followed by a solution update through data manipulation that is resilient to both soft and hard faults. The algorithm exploits data locality to reduce global communication. We discuss a server-client implementation where all state information is held by the servers, and clients are designed solely as computational units. Focusing on the stages of the algorithm that are most intensive in communication and computation, we explore the scalability of the actual code up to 12k cores, and build an SST/macro skeleton allowing us to extrapolate up to 50k cores. We show the resilience under simulated hard and soft faults for a 2D linear Poisson equation.

1. SUMMARY

As computing platforms evolve towards exascale, several key challenges are arising related to resiliency, power, memory access, concurrency and heterogeneous hardware [4, 1, 5, 2, 3]. There is no consensus or clear idea yet on what a "typical" exascale architecture might look like [1]. One of the main concerns is understanding how the hardware will affect future computing systems in terms of reliability, as well as communication and computational models, and which ones will emerge to become the main reference for exascale.

Exascale simulations are expected to rely on thousands of CPU cores running up to a billion threads [2, 3]. This framework will necessarily lead to systems with a large number of components, with an associated large communication cost for data exchange. This will inevitably lead to simulations that are communication-limited rather than being limited by CPU-time. The presence of many components and the increasing complexity of these systems (e.g. more and smaller transistors, and lower voltages) can become a liability in terms of system faults. Exascale systems are expected to suffer from errors and faults more frequently than the current petascale systems [2, 3]. Current parallel programming models will require a resilient infrastructure to be suitable for fault-free simulations across many cores for reasonable amounts of time.

Resilient EXtreme Scale Scientific Simulation (REXSSS) is our current project focused on developing novel approaches for resilient extreme scale computing. REXSSS is a domain-decomposition preconditioner for the solution of 2D partial differential equations (PDEs) that is resilient to both soft and hard faults. The algorithm consists in recasting the original PDE problem as a sampling problem, followed by a resilient data manipulation to achieve the final solution update. One of the main features of the algorithm is that it does not require detection of all types of faults that can occur, but focus solely on the information that a simulation provides.

The algorithm implementation relies on a server-client (SC) model grouping MPI processes into servers and clients. Currently, the servers are assumed to be safe (or "sandboxed") units holding the data, whereas the clients are designed solely to accept and perform work without any assumption on their reliability. A client is simply defined as a set of MPI processes, which can take up part or all of a computing node. Conceptually, a client does not necessarily need to be limited to reside on a single node, but can spread over multiple nodes. A key advantage of this structure relies in its inherent resiliency to hard faults, provided that the MPI framework is fault-tolerant. Since the actual data is safely held by the servers, the SC is inherently resilient to clients crashing (partial or complete node failures), since this only translates into missing tasks. The asynchronous nature of the SC model is beneficial to reduce the communication wait times, also considering the users have full control on the messages/tasks they can design to limit the communication overhead as much as possible.

The main innovations in this work are: a) the reformulation of the PDE solve so it is reduced to a number of independent tasks to increase concurrency and parallelism, and b) the ability to inherently mitigate the effects of both hard and soft faults that may occur in the execution of those tasks.

*Corresponding Author

The scalability shows excellent results for the major components of the algorithm. The SC approach make it suited for extreme scale applications with many more cores and significant system fault frequencies, which is the target configuration for this resilient PDE solver approach. The real scalability shows efficiency within 90%. This analysis is complemented by building a SST/macro [6] skeleton of our application that allows us to extrapolate up to $50k$ cores. The results from the SST/macro tests show excellent weak and strong scalability.

The resilience of the algorithm is analyzed for two different types of faults, namely hard faults modeling clients crashing, and soft faults occurring during task communication and execution. These faults are modeled using a Poisson process with a failure rate extracted from literature. We demonstrate the resiliency of the approach for a 2D linear elliptic PDE, and explore the effect of the faults. We remark that in all cases, the algorithm always reaches convergence, demonstrating its resiliency. The effect of the faults is to increase the time per iteration and/or the number of iterations that the algorithm needs to run to complete.

2. ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, under Award Numbers 13-016717. 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.

This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

3. ADDITIONAL AUTHORS

Additional authors: C. Safta (Sandia National Laboratories, email: `csafta@sandia.gov` and O. LeMaitre (Duke University, email: `olivier.le.maitre@duke.edu`) and O.M. Knio (Duke University, email: `omar.knio@duke.edu`)

4. REFERENCES

[1] J. A. Ang, R. F. Barrett, R. E. Benner, D. Burke, C. Chan, J. Cook, D. Donofrio, S. D. Hammond, K. S. Hemmert, S. M. Kelly, H. Le, V. J. Leung, D. R. Resnick, A. F. Rodrigues, J. Shalf, D. Stark, D. Unat, and N. J. Wright. Abstract machine models and proxy architectures for exascale computing. In *Proceedings of the 1st International Workshop on Hardware-Software Co-Design for High Performance Computing*, Co-HPC '14, pages 25–32, Piscataway, NJ, USA, 2014. IEEE Press.

[2] F. Cappello, A. Geist, B. Gropp, L. Kale, B. Kramer, and M. Snir. Toward Exascale Resilience. *International Journal of High Performance Computing Applications*, 23(4):374–388, oct 2009.

[3] F. Cappello, A. Geist, W. Gropp, S. Kale, B. Kramer, and M. Snir. Toward exascale resilience: 2014 update. *Supercomputing frontiers and innovations*, 1(1), 2014.

[4] DOE-ASCR. Exascale programming challenges. Technical report, July 2011.

[5] DOE-ASCR. Top ten exascale research challenges. Technical report, Feb. 2014.

[6] J. J. Wilke and J. P. Kenny. Using discrete event simulation for programming model exploration at extreme-scale: Macroscale components for the structural simulation toolkit (sst). Technical report, Sandia Technical Report SAND2015-1027, 2015.