Bringing Task and Data Parallelism to Analysis of Climate Model Output

Robert L. Jacob*, Jayesh Krishna*, Xiabing Xu*, Sheri Mickelson*, Tim Tautges*, Mike Wilde*,
Robert Latham*, Ian Foster*, Rob Ross*, Mark Hereld*, Jay Larson*,
Pavel Bochev†, Kara Peterson†, Mark Taylor†, Karen Schuchardt‡, Jian Yin‡,
Don Middleton§, Mary Haley§, David Brown§, Richard Brownrigg§, Wei Huang§, Dennis Shea§,
Mariana Vertenstein§, Kwan-Liu Ma¶, Jinrong Xie¶
*Argonne National Laboratory
†Sandia National Laboratory
‡Pacific Northwest National Laboratory
§National Center for Atmospheric Research
¶University of California, Davis

***Abstract*— Climate models are both outputting larger and larger amounts of data and are doing it on more sophisticated numerical grids. The tools climate scientists have used to analyze climate output, an essential component of climate modeling, are single threaded and assume rectangular structured grids in their analysis algorithms. We are bringing both task- and data-parallelism to the analysis of climate model output. We have created a new data-parallel library, the Parallel Gridded Analysis Library (ParGAL) which can read in data using parallel I/O, store the data on a compete representation of the structured or unstructured mesh and perform sophisticated analysis on the data in parallel. ParGAL has been used to create a parallel version of a script-based analysis and visualization package. Finally, we have also taken current workflows and employed task-based parallelism to decrease the total execution time.**

I. Introduction

As climate model's reach the petascale, they are outputting more and more data on structured and unstructured grids [1]. Analysis of this output is essential to determining what the model is saying about the climate system. The programs currently used to perform this analysis are often not nearly as flexible or high-performing as the primary applications. They either break or require workarounds for the ultra-large unstructured-grid data that is becoming the norm in climate science [2]. Ultra-large data sets also present a memory challenge because often these programs cannot read all the data in to memory.

Programs such as Parallel-R [3] provide data-parallel versions of some of its statistical analysis functions. However it does not support operations on a grid. Tools such as GLEAN [4] or DIY [5] provide facilities for data staging and movement in an HPC environment but not the discretization-aware data model we need. In this poster, we describe how to speed up climate data post processing by using a new parallel gridded analysis library and a parallel scripting language.

II. Data Parallelism: ParGAL

ParGAL (Parallel Gridded Analysis Library) leverages the capabilities of the Mesh Oriented datABase (MOAB) [6],

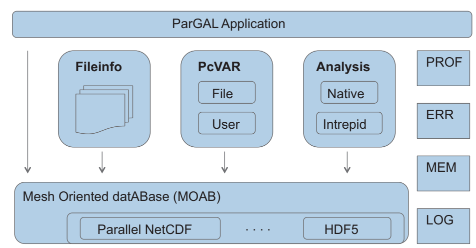

Fig. 1. ParGAL Architecture.

Parallel netCDF (PnetCDF) [7] and Intrepid [8] libraries, to accomplish efficient, parallel, discretization-accurate data analysis. MOAB is a library for representing mesh data that supports structured, unstructured finite element and polyhedral grids and also supports parallel operations on those grids including loading to and from disk using parallel I/O. The PnetCDF library allows reading NetCDF [9] data in parallel using MPI-IO [10] and is used by MOAB to read climate data. Intrepid is an extensible library for computing operators (such as gradient, curl, divergence, etc.) acting on discretized fields.

ParGAL is highly scalable and is designed to simplify implementation and evaluation of a wide variety of discretization-specific algorithms on large scale structured and unstructured grid data sets. ParGAL provides simple interfaces for performing data analysis and encapsulates details on parallel partitioning, mesh-based communication etc.

There are four main components in ParGAL: Fileinfo, PcVAR, Analysis and Support. The Analysis module contains the data analysis algorithms implemented in ParGAL. We have implemented algorithms to calculate maximum, minimum, median, average, vorticity and divergence on climate datasets. PcVAR provides an abstraction of the climate data loaded into MOAB and is used to perform data analysis on the data via ParGAL algorithms. Fileinfo provides the meta data information pertaining to NetCDF files loaded by the application. The

Support module provides miscellaneous support features like exception handling.

III. Task Parallelism

Swift is a parallel scripting system for Grids and clusters. Swift uses a simple, high-level C-like functional language with which users can describe their workflow in terms of data inputs, functions that operate on the input, and resulting outputs. The Swift engine then automatically parallelizes all tasks that have no data dependencies and schedules work automatically on resources described in a configuration file. Swift requires only a java environment to execute locally and can transparently ship tasks and data to remote resources.

IV. Application to Climate

A. ParNCL

To demonstrate the ability of ParGAL to encapsulate parallel analysis at a high level, we are using ParGAL to create a data-parallel version of the NCAR Command Language (NCL). NCL [11] is a free interpreted language that is widely used for data analysis and visualization especially in the climate community. NCL offers a wide array of data analysis operations ranging from simple math operations like finding the minimum element in an array to sophisticated domain-specific operations. We have developed a parallel version of the NCL interpreter, ParNCL, that performs data analysis in parallel using ParGAL and MOAB. ParNCL reads the climate data from NetCDF files using MOAB and performs data analysis using the ParGAL library. Once the data analysis is complete the single threaded visualization algorithms are used to plot the results.

B. AMWG Diagnostics package

The Atmospheric Model Working Group (AMWG) for the DOE/NSF Community earth system model has built a diagnostics package [12] that creates several hundred 2D plots and tables from the output of the atmospheric component of a climate model. The package consists of unix shell scripts that use NCO [13] to perform data reduction and NCL to construct plots. We parallelized the workflow by modifying the diagnostic package to use Swift. The task-parallel versions are 3 to 4 times faster then the original serial code.

V. Conclusion

Post-processing analysis of petascale model output is a crucial component of the scientific process in computational science. We have leveraged several well-engineered software libraries to build a parallel data analysis library. Our early results comparing performance to a well established visualization and analysis package are encouraging. We also showed how to significantly speed up diagnostic workflows using a parallel scripting language. Future work involves implementing more data analysis operations in ParGAL.

Acknowledgment

This work was supported in part by the Office of Biological and Environmental Research, Office of Science, U.S. Department of Energy, under contract DE-ACO2-O6CH11357. The project is co-sponsored by the U.S. National Science Foundation via contributions from the National Center for Atmospheric Research, Boulder, CO. We gratefully acknowledge the computing resources provided on "Fusion," a 320-node computing cluster operated by the Laboratory Computing Resource Center at Argonne National Laboratory. 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.

References

[1] J. M. Dennis, M. Vertenstein, P. H. Worley, A. A. Mirin, A. P. Craig, R. Jacob, and S. Mickelson, "Computational performance of the ultra-high resolution capability in the community earth system model," *Int. J. High Perf. Comp. Appl.*, vol. 26, no. 5, pp. 5–16, 2012.

[2] W. Washington, "Scientific grand challenges: Challenges in climate change science and the role of computing and the extreme scale," http://science.energy.gov/~/media/ber/pdf/Climate_report.pdf, 2008.

[3] N. Samatova, M. Branstetter, A. Ganguly, R. Hettich, S. Khan, G. Kora, J. Li, X. Ma, C. Pan, A. Shoshani, and S. Yoginath, "High performance statistical computing with Parallel R: Applications to biology and climate modeling," in *Journal of Physics: Conference Series SciDAC 2006*, vol. 46, 2006, pp. 505–509.

[4] V. Vishwanath, M. Hereld, and M. E. Papka, "Simulation-time data analysis and I/O acceleration on leadership-class systems using GLEAN," in *IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, Oct. 2011.

[5] T. Peterka, R. Ross, W. Kendall, A. Gyulassy, V. Pascucci, H.-W. Shen, T.-Y. Lee, and A. Chaudhuri, "Scalable parallel building blocks for custom data analysis," in *Proceedings of Large Data Analysis and Visualization Symposium LDAV'11*, Providence, RI, 2011.

[6] T. J. Tautges, R. Meyers, K. Merkley, C. Stimpson, and C. Ernst, "MOAB: a Mesh-Oriented database," Sandia National Laboratories, SAND2004-1592, Apr. 2004, report.

[7] J. Li, W. keng Liao, A. Choudhary, R. Ross, R. Thakur, W. Gropp, R. Latham, A. Siegel, B. Gallagher, and M. Zingale, "Parallel netCDF: A high-performance scientific I/O interface," in *Proceedings of SC2003: High Performance Networking and Computing*, ser. SC '03. Phoenix, AZ: IEEE Computer Society Press, November 2003, pp. 39–. [Online]. Available: http://www.sc-conference.org/sc2003/paperpdfs/pap258.pdf

[8] M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, and K. S. Stanley, "An overview of the trilinos project," *ACM Trans. Math. Softw.*, vol. 31, no. 3, pp. 397–423, 2005.

[9] R. Rew and G. Davis, "NetCDF: an interface for scientific data access," *Computer Graphics and Applications, IEEE*, vol. 10, no. 4, pp. 76 –82, july 1990.

[10] "MPI-2: Extensions to the message-passing interface," The MPI Forum, July 1997. [Online]. Available: http://www.mpi-forum.org/docs/docs.html

[11] "NCL – NCAR Command Language (version 6.0.0) [software]. Boulder, Colorado: UCAR/NCAR/CISL/VETS," http://dx.doi.org/10.5065/D6WD3XH5, 2012.

[12] N. Community Earth System Model, "The AMWG diagnostics package," http://www.cgd.ucar.edu/amp/amwg/diagnostics/. [Online]. Available: http://www.cgd.ucar.edu/amp/amwg/diagnostics/

[13] C. Zender, "NCO – NetCDF operator," http://nco.sourceforge.net/. [Online]. Available: http://nco.sourceforge.net/