Accelerating TauDEM as a Scalable Hydrological Terrain Analysis Service on XSEDE

Ye Fan, Yan Liu, Shaowen Wang
University of Illinois at Urbana-Champaign
Urbana, IL, USA
{fanye, yanliu, shaowen}@illinois.edu

David Tarboton, Ahmet Yildirim
Utah State University
Logan, UT, USA
david.tarboton@usu.edu,
ahmetartu@aggiemail.usu.edu

Nancy Wilkins-Diehr
San Diego Supercomputer Center
La Jolla, CA, USA
wilkinsn@sdsc.edu

ABSTRACT

In this paper, we present the experience of scaling a parallel hydrological analysis software - TauDEM - to thousands of processors and large elevation datasets through XSEDE ECSS effort and multi-institutional collaboration.

Categories and Subject Descriptors

C.2.4 [**Distributed Systems**]: Distributed Applications

General Terms

Algorithms, Measurement, Performance, Experimentation.

Keywords

Parallel computing, hydrological analysis, scalability

1. INTRODUCTION

Terrain analysis using Digital Elevation Models (DEM) has been widely used in hydrology to derive topographic information for hydrologic analysis and modeling. Finer resolutions on DEMs have been shown to have significant impact on hydrologically important variables and improve the accuracy and reliability of results [1]. However, as high-resolution DEMs become increasingly available, e.g., LiDAR-based DEMs (http://opentopography.org) and the high-resolution DEMs in the National Elevation Dataset produced by the U.S. Geological Survey (http://ned.usgs.gov), the computation cost of DEM-based hydrological modeling significantly increases [1] and, thus, makes desktop computer-based analyses extremely difficult.

TauDEM is a parallel computing solution to tackle this computational challenge. A set of parallel hydrological terrain processing algorithms was developed in TauDEM to leverage the Message Passing Interface (MPI, http://mpi-forum.org) and MPI IO for efficient handling of the processing and input/output (I/O) of DEM data and TauDEM results on multiple processors. A multi-institutional effort has been ongoing to leverage expertise in multiple disciplines (i.e., hydrology, computational science, geographic information science, and geography) to scale TauDEM for analyzing large DEMs on national cyberinfrastructure such as XSEDE (http://xsede.org) and publish TauDEM as a widely accessible cyberinfrastructure-empowered science gateway application service. This is a collaboration among the TauDEM team at Utah State University (http://hydrology.usu.edu/taudem), XSEDE Extended Collaborative Support Services (ECSS [2]), the National Science Foundation (NSF)-funded CyberGIS project (http://cybergis.org), and the NSF OpenTopography data facility (http://opentopography.org). This paper presents the findings and experience in enhancing TauDEM for large-scale terrain analysis on massive computing resources provided on XSEDE through rigorous computational performance profiling and analysis. As a result, computational bottlenecks that were not observed on local clusters were identified and successfully eliminated. The improved TauDEM software scales to thousands of processors and is capable of handling 36GB DEM data, which was not possible on small-scale clusters. The software is now deployed on XSEDE and paves the way for the development of TauDEM science gateway services in order to provide our communities with TauDEM capabilities on high-resolution DEMs (Figure 1).

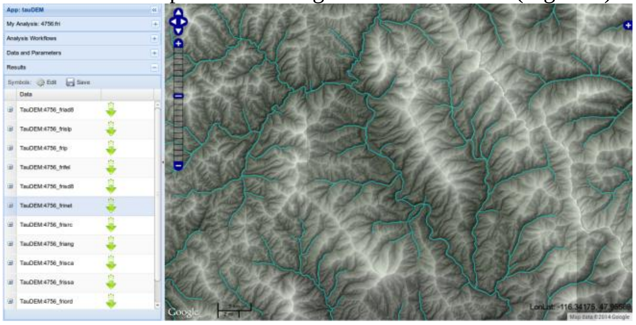

Figure 1. TauDEM in CyberGIS Gateway. Left panel: TauDEM products. Right panel: visualization of two products: hydrologically correct DEM and stream network.

2. COMPUTATIONAL PERFORMANCE

TauDEM is open source software written in C++. It includes a suite of functions for deriving hydrological terrain analysis products. Dependencies among these functions and their input/output make a typical TauDEM analysis a computational workflow. Each TauDEM function uses MPI to harness multiple processors for parallel processing and produces one or more raster (i.e., rectangular grids representing the study area) and/or vector (e.g., points, lines, polygons) products. The spatial domain of the study area is partitioned into sub-regions and distributed to multiple MPI processes. Since the derivation of hydrological information such as flow direction, contributing area, stream network require global knowledge, intensive communication is involved among MPI processes. Output raster information distributed on each MPI process is written into one or multiple GeoTIFF files using MPI IO. Experiments on local cluster based on Network File System (NFS) showed limited scalability to input DEM size (<=2GB) and the number of processors (<=64 cores).

XSEDE '14, Jul 13-18 2014, Atlanta, GA, USA
ACM 978-1-4503-2893-7/14/07.
http://dx.doi.org/10.1145/2616498.2616510

The Trestles cluster at San Diego Supercomputer Center (SDSC) was used for comprehensive performance profiling in order to identify potential computing, data, memory, or network bottlenecks. Initial experiments applied general profiling techniques and quickly identified four of the most time consuming functions of TauDEM (i.e., PitRemove, D8FlowDir, DInfFlowDir, and StreamNet). An I/O bottleneck was identified in the TiffIO library used by all TauDEM functions when used for large datasets spanning multiple files distributed across multiple processes. The handling of multiple files was restructured, eliminating this I/O bottleneck resulting in marginal I/O cost on all of the TauDEM functions. Scalability test on the improved TauDEM code using large DEM data showed consistent speedup.

2.1 I/O Bottleneck

TauDEM uses MPI IO to read and write GeoTIFF raster files and exhibited apparent I/O slowdown on NFS (e.g., PitRemove on a 2GB DEM took 681 seconds on an 8-core PC *vs.* 3,759 seconds on a 64-core cluster). Further profiling on the Lustre file system on Trestles showed that the slowdown got worse as more processor cores were used. Darshan – a profiling tool (http://mcs.anl.gov/research/projects/darshan/) for I/O was employed to investigate I/O patterns of the PitRemove function, the first function in a typical TauDEM workflow. We found that a large number of file read operations attributed to the majority of I/O time and then located GeoTIFF file header reading code as the bottleneck. In original coding, if n TauDEM processes read m DEM files, totally $n*m$ header reading operations were performed to retrieve spatial context information. As n becomes large, the total number of I/O operations significantly slows down the header reading. Such behavior was not obvious when n is small, which is true on local cluster. Accordingly, code was changed to have one process read all file headers and propagate header information to all other processes using MPI broadcast. We also identified a deadlock issue in the reading of a large number of files due to a limitation on the number of files that can be open simultaneously. This problem was resolved by closing files immediately following the reading of data from them. TauDEM uses spatial domain partition as its parallelization strategy. On output we found slowdown where the domain covered by an output file spanned the domain partition used by a single process, requiring multiple processes to write to the same file. This write bottleneck was resolved by restructuring the arrangement of output files so that separate output files are written from each domain partition process. Table 1 shows the comparison of execution time of the PitRemove function on a 2GB DEM. With above changes, I/O is no longer a bottleneck of scalability.

Table 1. I/O Time Comparison (before / after; in seconds)

#cores	Compute	Header Read	Data Read	Data Write
32	42.7 / 42.8	193.5 / 3.8	0.4 / 0.4	153.5 / 3.5
64	35.3 / 34.8	605.5 / 3.9	1.5 / 1.1	160.2 / 2.3
128	33.7 / 33.0	615.2 / 2.6	0.9 / 1.0	173.2 / 2.3
256	37.5 / 38.0	831.7 / 2.3	0.5 / 0.9	391.3 / 1.6

2.2 Scalability on Large DEM Data

Before this ECSS project, the maximum DEM size which TauDEM could process was 6GB using a limited number of processors. . Figure 2 shows the scalability results on a 36GB DEM after the aforementioned I/O bottlenecks were resolved. Further scaling effort targets 180GB DEM data on 4,096 cores, with the ultimate goal of processing the entire 10-meter NED dataset (0.5TB) on XSEDE. A set of performance optimization techniques are also being developed to maximize the resource utilization given different computational demands of each TauDEM function. Methodological exploration on how to accelerate the three most costly functions is the next step to further address the computing challenges in TauDEM.

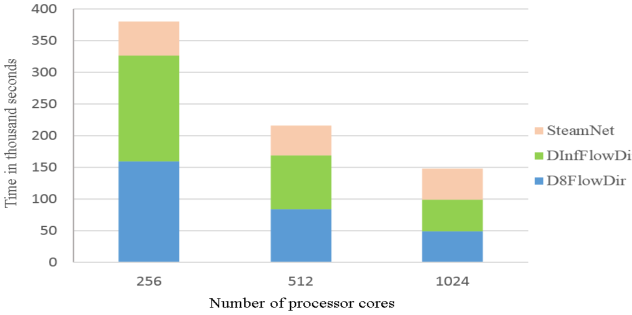

Figure 2. Execution time of the three most costly TauDEM functions on a 36GB DEM dataset.

3. CONCLUDING DISCUSSION

The aforementioned ECSS project for scalable hydrological terrain analysis was successfully executed to identify and resolve major computational bottlenecks in the TauDEM software code and opened the door to further performance enhancements and the establishment of TauDEM as a science gateway service.

Close and effective collaboration coordinated through XSEDE ECSS was crucial to achieve the success of resolving computational bottlenecks of TauDEM. Four teams coordinated closely to streamline the process of profiling, debugging, and integrating solutions into newer versions of TauDEM disseminated in GitHub (https://github.com/dtarb/taudem). The success in enhancing TauDEM capability has motivated extending TauDEM services in two science gateways: CyberGIS and OpenTopography. The collaboration has also generated research ideas for enhancing existing hydrological terrain analysis methodologies, developing more efficient ways to exploit parallel computing resources in TauDEM workflow execution, and resolving integration challenges in data, analysis, and visualization for establishing highly usable TauDEM services.

4. ACKNOWLEDGMENTS

This work used XSEDE (NSF Grant No. 1053575). Tarboton and Yildirim's TauDEM work is supported by the US Army Research and Development Center contract No. W912HZ-11-P-0338 and the NSF Grant No. 1135482. This material is based in part upon work supported by NSF under Grant No. 0846655 and 1047916.

5. REFERENCES

[1] T. Tesfa, D. Tarboton, D. Watson, K. Schreuders, M. Baker and R. Wallace, "Extraction of hydrological proximity measures from DEMs using parallel processing," *Environmental Modelling & Software*, vol. 26, no. 12, pp. 1696-1709, 2011.

[2] M. Romanus, P. Mantha, M. McKenzie, T. Bishop, E. Gallichio, A. Merzky, Y. Khamra and S. Jha, "The anatomy of successful ECSS projects: lessons of supporting high-throughput high-performance ensembles on XSEDE," in *Proceedings of XSEDE'12 Conference*, Chicago, IL, 2012.