Geomancy: Automated Performance Enhancement through Data Layout Optimization

Oceane Bel*, Kenneth Chang*, Nathan R. Tallent‡, Dirk Duellmann§,
Ethan L. Miller*†, Faisal Nawab* and Darrell D. E. Long*
Emails: obel@ucsc.edu, kchang44@ucsc.edu, tallent@pnnl.gov, Dirk.Duellmann@cern.ch,
elm@ucsc.edu, fnawab@ucsc.edu and darrell@ucsc.edu
*University of California, Santa Cruz, †Pure storage, ‡Pacific Northwest National Labs, §CERN

***Abstract*—The size and complexity of large storage systems, such as high-performance computing (HPC) systems, inhibit rapid effective restructuring of data layouts to maintain performance as workloads shift. To address this issue, we have developed Geomancy, a tool that models the placement of data within a distributed storage system and reacts to drops in performance. Our approach to optimizing throughput offers benefits for storage systems such as avoiding potential bottlenecks and increasing overall I/O throughput from 11% to 30%.**

I. Introduction

High-Performance Computing (HPC) and High Throughput Computing (HTC) systems deliver ever-increasing levels of computing power and storage capacity; however, the full potential of these systems is limited by the inflexibility of data layouts to rapidly changing demands. A shift in demand can cause a system's throughput and latency to suffer, as workloads access data from contended regions of the system. In a shared environment, computers may encounter unforeseen changes in performance. Network contention, faulty hardware, or shifting workloads can reduce performance and, if not diagnosed and resolved rapidly, can create slowdowns around the system.

To mitigate contention, system designers implement static or dynamic algorithms that place data based on how recently the files have been used similar to the caching algorithm Least Recently Used. However, existing strategies require manual experimentation to compare various configurations of data which is expensive or in some cases infeasible. These algorithms are not sufficient for all workloads because they do not adapt as workloads change, and they may not be optimal for all workloads.

To address this issue, we have developed Geomancy, a tool that improves system performance by finding efficient data layouts using reinforcement learning in real-time. Geomancy targets systems that serve and process petabytes of data, such as particle collision analysis [1]. Workloads on these systems are commonly spread across multiple storage devices which can lead to storage devices becoming contended over time. If a heavily used storage device becomes contended, the delay can be felt across the system. Many of these systems collect workload metrics describing the operation of such systems, and we can leverage these metrics as performance data to train a predictive model.

Performance data includes parameters such as average access latencies, remaining storage space, number of previous reads and writes, restrictions on reads or writes, file types that are read or written, and number of bytes accessed. Using this data, we build a predictive model using artificial neural networks that relates system time, data location, and performance. Geomancy's neural network uses this model to forecast when and where a bottleneck can happen due to changing workloads. Additionally, it preempts future drops in performance by moving data between storage devices. If the model predicts an improved location for a piece of data, Geomancy sends the new location ID to the target system, which moves the data to the new location. These models are built from the analysis of two workloads provided by our collaborators. We experimentally test our method in a small scale system against algorithmic modeling, and observe an 11% to 30% performance gain in our experiments including moving overhead compared to policies that place data dynamically or statically according to how frequently or recently the data has been used.

II. Approach

Geomancy was motivated by workload analysis of two workloads: traces generated from a Monte Carlo physics workload provided by Pacific Northwest National Laboratories (PNNL) and workload traces from CERN. The exact methods to generate the traces do not matter for the purposes of our experiments, however each trace follows a similar setup. The traces used all have features that describe the I/O throughput of the system one wishes to optimize. For example, the CERN EOS trace contains information about when a file was opened, closed and where the action took place. We care about when the file was opened since if a file is opened at a time when the storage device is contended it will affect the access latency. We also care about where since some storage devices are more contended then others.

We approach the file layout problem as an unsupervised deep reinforcement learning problem where the throughput of the system is the reward. Our neural network predicts the throughput of accessing a piece of data at every potential location it can exist. To calculate the future throughput of an access at a certain location, we model how each input feature (file location, file size, or any feature describing the action executed on the file such number of bytes read or written)

DOI 10.1109/ISPASS48437.2020.00025

interacts with other input features. Additionally, to avoid future bottlenecks, Geomancy needs to know when to change the data layout to preempt potential accesses that could cause a bottleneck. Given a large trace of throughput measurements, file locations, and transfer overheads, we use these features from the traces to train a neural network.

To determine a useful model, we compared 23 neural network architectures. We used a number of features that we selected from the PNNL server, equalling six. Each layer in the neural networks is represented using the following format: number of neurons (type of layer) selected activation function. Architectures in bold are the networks that performed the best out of the 23 networks in terms of accuracy when predicting future throughput of each storage point on the PNNL system. A through model search can reveal other architectures with better accuracy, however for the scope of the paper we limit our search to these 23 architectures. This gives us a wide range of networks to experiment with from fully dense networks to common recurrent networks. From this experimentation, we determined that a fully connected dense network of four layers provided the highest accuracy and most reliable predictions given our dataset.

Once the neural network has predicted the future performance value of each available locations, the action checker checks for the permissions and availability of the location with the highest performance. If the location is value the file used for prediction is moved to that location. To tackle larger storage systems of millions of files and dozens of mount points, we will need a data movement scheduler (implemented either as a second neural network or algorithm) that determines a cooldown between file movement. We have left this as future work, and we intend on implementing this as further development in improving larger and multi-user workloads.

III. Result

Experimentally, Geomancy outperforms both static and dynamic data placement algorithms by at least 11%, as shown in Figures 1 and 2. In all figures, a vertical gray line represents when Geomancy decides to move data. The blue lines below each graph corresponds to how many files are moved by Geomancy at that access number. We can see that most of the time Geomancy only moves a small subsets of the file to other nodes. Most static placement methods have lower performance, and only a few randomly chosen data placements challenges the performance gain made by Geomancy.

Geomancy accurately captures changes in performance as the workloads runs on the target system. Those predictions are then able to be used to change the data layout of the system. Experimentally, we demonstrate inter-workload congestion reduction and increases in overall throughput from 11% to 30%. Compared to algorithmic or manual data placement, Geomancy is superior in that it reduces bottlenecks and can anticipate when performance fluctuations may happen. By predicting when and where performance may drop, moving data before the slowdowns occurs stabilizes performance and prevents fluctuations in throughput.

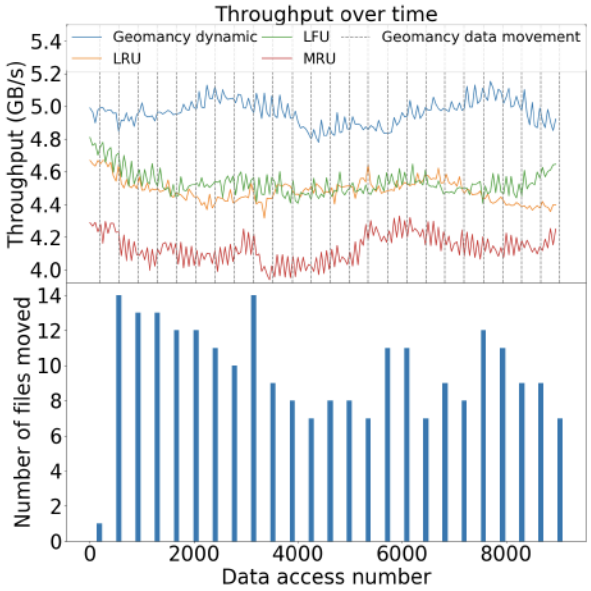

Fig. 1: Geomancy's performance compared to dynamic solutions on the live system. Size of the data that Geomancy moves ranges from 583 KB to 1.1 GB.

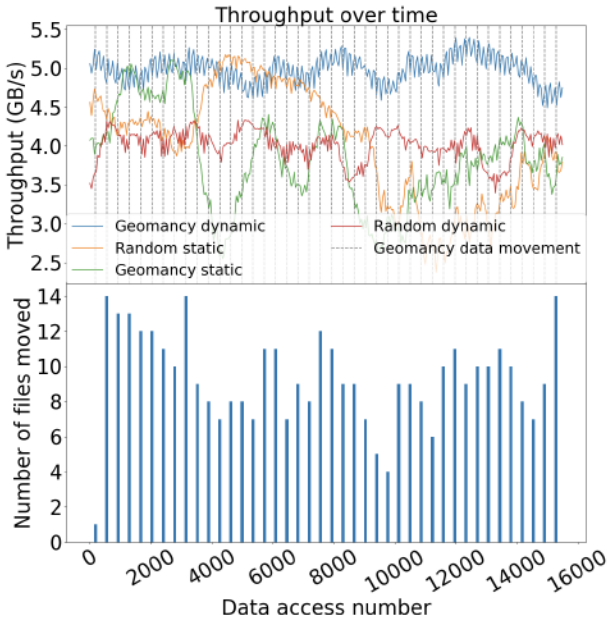

Fig. 2: Geomancy's performance compared to static solutions on the live system. Size of the data that Geomancy moves ranges from 583 KB to 1.1 GB.

References

[1] A. Peters, E. Sindrilaru, and G. Adde, "EOS as the present and future solution for data storage at CERN," *Journal of Physics: Conference Series*, vol. 664, no. 4, p. 042042, 2015.