Proposal for comparison and measurement of parallel and distributed file systems for training ML models in the healthcare

João V. Vargas[1], Cristiano A. Künas[1], Thiago Araújo [1], Bruno Morales [1], Philippe O. A. Navaux[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{jvvoliveira,cakunas,tsaraujo,bmmorales,navaux}@inf.ufrgs.br

Abstract. *Many scientific fields are increasingly relying on high-performance computing (HPC) to handle and analyze vast amounts of experimental data. At the same time, storage systems in modern HPC environments must adapt to different access patterns. These patterns involve frequent metadata operations, numerous small I/O requests, and randomized file access, whereas traditional parallel file systems have been optimized primarily for sequential and shared access to large files. In this research, we will compare GekkoFS and evaluate its performance against Lustre, a widely used parallel file system that meets the demanding requirements of HPC simulation environments. Our comparison aims to highlight the strengths and limitations of each system for training ML models in healthcare.*

Resumo. *Diversas áreas da ciência passam a depender cada vez mais da computação de alto desempenho (HPC) para processar e analisar grandes volumes de dados experimentais. Ao mesmo tempo, os sistemas de armazenamento em ambientes modernos de HPC devem se adaptar a diferentes padrões de acesso. Esses padrões envolvem operações frequentes de metadados, inúmeras solicitações de E/S pequenas e acesso aleatório a arquivos, enquanto os sistemas tradicionais de arquivos paralelos foram otimizados principalmente para acesso sequencial e compartilhado a arquivos grandes. Nesta pesquisa, compararemos o GekkoFS e avaliaremos seu desempenho em relação ao Lustre, um sistema de arquivos paralelo amplamente utilizado que atende aos exigentes requisitos dos ambientes de simulação em HPC. Nossa comparação tem como objetivo destacar os pontos fortes e as limitações de cada sistema para treinar modelos de machine learning.*

1. Introduction

Advances in HPC have enabled the evolution of various data science applications and Artificial Intelligence (AI) methods for real-world problems, particularly in the healthcare sector. On the other hand, running these models requires substantial computational power due to the complexity of the models and the need to access larger and more diverse datasets [Samsi et al. 2023].

In the context of the research project **Artificial Intelligence Applied to Healthcare** [Dos Reis et al. 2024], various machine learning algorithms are employed. The successful operation of these algorithms depends on having one or more sufficiently large datasets for the training phase, hardware accelerators such as GPUs and TPUs to process the data efficiently, as well as the file system used to train these models.

Most data-driven workloads are based on new algorithms and data structures, which introduce new demands on HPC file systems. These new workloads include data synchronization, small I/O requests, and large numbers of metadata operations [Macedo et al. 2023]; these access patterns show significant differences compared to previous workloads, which mainly executed sequential I/O operations on large files.

In an increasingly diverse AI application landscape, it is crucial to investigate and evaluate different file systems to determine which one is best suited to support learning model training workloads efficiently. Several file systems are currently available, each with its own characteristics and advantages, from distributed file systems to systems optimized for intensive read/write operations [9].

This work proposes a comparison between GekkoFS [Vef et al. 2020], a temporarily deployed, highly scalable distributed file system for HPC applications, and Lustre [Braam 2019], a parallel distributed file system commonly used for large-scale cluster computing. The study aims to evaluate their performance, scalability, and suitability for training ML models. The remainder of this paper is organized as follows. Section 2 presents our proposal, section 3 methodology and section 4 our expected results.

2. Proposal

The goal is to conduct an efficient comparison of the GekkoFS and Lustre file systems. This involves developing an evaluation methodology that enables a comparison of these file systems in terms of performance, efficiency and suitability for training machine learning and deep learning models.

The file systems will be evaluated based on relevant metrics, such as data transfer rate, read/write latency, system resource utilization and scalability. Moreover, this study will investigate the impact of different file systems on training and inference of ML and DP models using datasets and popular benchmarks.

The motivation behind this research lies in the need to optimize storage resources used in ML and DL model training. With the continuous growth of dataset sizes and model complexity, file systems play a crucial role in determining the overall efficiency of the training process. Furthermore, the inappropriate selection of a file system can lead to performance bottleneck, increasing computational costs and extending the time required to train AI models

Additionally, this study aims to provide insights into how different file systems handle the increasing demands of AI-driven workloads, particularly in healthcare applications. As machine learning models become more complex and require larger datasets, the efficiency of file system operations directly impacts model training speed and overall system performance. By analyzing key factors such as parallelism, metadata handling, and fault tolerance, this research will contribute to the understanding of how modern file systems can be optimized for AI applications. The findings will help guide future decisions on file system selection and configuration, ensuring that HPC environments can effectively support machine learning and deep learning workloads with minimal performance bottlenecks.

3. Methodology

To conduct a comprehensive performance evaluation of the GekkoFS and Lustre file systems. We require a high-performance testbed that reflects real-world HPC and AI workloads. The selected testbed consists of Grace1 and Grace2, housed at the high-performance national park at UFRGS.

Grace1 and Grace2 provide an optimal environment for benchmarking distributed file systems due to their high computational power, large memory capacity, and high-speed storage. With 2 x Grace A02, 3.40 GHz, 144 threads, 144 cores, high-performance computing power, along with 480GB LPDDR5 RAM, reduces I/O bottlenecks by enabling effective caching and buffering during file system operations. The combination of 3.5 TB NVMe and 894.3 GB NVMe enables testing of both sequential and random I/O patterns, which is critical for evaluating Lustre and GekkoFS in HPC and AI scenarios.

To ensure a fair and thorough comparison, we will use both synthetic and real-world benchmarks to measure file system performance under different workloads. These benchmarks simulate file system performance using controlled experiments, providing insights into raw I/O efficiency [Gupta et al. 2024]. The IOR (Interleaved Or Random) benchmark measures read and write throughput in sequential and random access patterns. The FIO (Flexible I/O Tester) simulates real-world I/O patterns for different application scenarios. The MLPerf Storage Benchmark evaluates file system performance for machine learning training and inference. The Parallel File System Workload (PFSW) measures performance when multiple nodes access the file system simultaneously.

Failure injection tests simulate node failures to evaluate how the file system recovers from crashes. Data integrity checks verify how reliable the file system is in handling data corruption. Small file benchmarks measure how well the file system handles small file operations, which is crucial for AI and metadata-heavy workloads. The dd command tests provide simple disk read and write latency measurements.By integrating these diverse benchmarking tools and performance tests, we ensure a comprehensive evaluation This rigorous analysis will help identify the strengths and limitations of each file system, guiding researchers and practitioners in selecting the most suitable storage solution for their specific computational needs.

Table 1. Summarizes the main points of the methodology

Category	Details
Testbed	Grace1, Grace2 at UFRGS (2 x Grace A02, 3.40 GHz, 144 cores, 480GB LPDDR5 RAM, 3.5TB + 894.3GB NVMe).
File Systems	GekkoFS; Lustre.
Benchmark Tools	IOR; FIO; MLPerf Storage; PFSW.
Performance Tests	Failure Injection; Data Integrity; Small File Benchmarks; dd Command.
Evaluation Metrics	Read/write throughput; latency; system resource utilization; and scalability.
Objective	Identify strengths and limitations of GekkoFS and Lustre for ML training in HPC and AI workloads.

4. Conclusion

The comparison of GekkoFS and Lustre in the context of HPC and AI workloads is expected to provide valuable insights into their performance, scalability, and suitability for training machine learning and deep learning models. This study will evaluate how each file system handles large-scale data processing, metadata-intensive operations, and parallel I/O workloads, ensuring an in-depth analysis of their impact on training efficiency. The findings from this research aim to help identify the most effective storage solution for optimizing AI-driven research, particularly in healthcare applications.

References

Braam, P. (2019). The lustre storage architecture. *arXiv preprint arXiv:1903.01955*.

Dos Reis, M. A., Künas, C. A., da Silva Araújo, T., Schneiders, J., de Azevedo, P. B., Nakayama, L. F., Rados, D. R., Umpierre, R. N., Berwanger, O., Lavinsky, D., et al. (2024). Advancing healthcare with artificial intelligence: diagnostic accuracy of machine learning algorithm in diagnosis of diabetic retinopathy in the brazilian population. *Diabetology & Metabolic Syndrome*, 16(1):209.

Gupta, A., Dhakshinamoorthy, D., and Paul, A. K. (2024). Studying the effects of asynchronous i/o on hpc i/o patterns. In *2024 IEEE International Conference on Cluster Computing Workshops (CLUSTER Workshops)*, pages 109–112. IEEE.

Macedo, R., Miranda, M., Tanimura, Y., Haga, J., Ruhela, A., Harrell, S. L., Evans, R. T., Pereira, J., and Paulo, J. (2023). Taming metadata-intensive hpc jobs through dynamic, application-agnostic qos control. In *2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, pages 47–61. IEEE.

Samsi, S., Zhao, D., McDonald, J., Li, B., Michaleas, A., Jones, M., Bergeron, W., Kepner, J., Tiwari, D., and Gadepally, V. (2023). From words to watts: Benchmarking the energy costs of large language model inference. In *2023 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–9. IEEE.

Vef, M.-A., Moti, N., Süß, T., Tacke, M., Tocci, T., Nou, R., Miranda, A., Cortes, T., and Brinkmann, A. (2020). Gekkofs—a temporary burst buffer file system for hpc applications. *Journal of Computer Science and Technology*, 35:72–91.