RESEARCHERID
THOMSON REUTERS

[Sam *et al.*, 12(4): April, 2025] ISSN 2349-0292
Impact Factor 3.802

GLOBAL JOURNAL OF ADVANCED ENGINEERING TECHNOLOGIES AND SCIENCES

PERFORMANCE-GUARANTEED DATA REPLICATION FOR DATA-INTENSIVE SCIENTIFIC APPLICATIONS

Anirban Sam *[1], Arpit Solanki*[2]

*[1]Student, Dr. A.P.J. Abdul Kalam University, Indore, M.P., India
*[2]Assistant Professor, Dr. A.P.J. Abdul Kalam University, Indore, M.P., India

ABSTRACT

A Data Grid is composed of multiple interconnected sites organized in a hierarchical structure with one top-level site and several institutional sites. The top-level site functions as the central management unit and is responsible for maintaining the Replica Catalogue (RC), which stores metadata about data files and their replicas across different sites. Each site possesses both computational and storage capabilities, enabling job execution and data storage within the Grid environment. Communication within a site is assumed to have negligible delay due to high internal bandwidth.

In the data replication framework, multiple data files are generated by designated source sites, and a single site may act as the source for more than one file. Since each Grid node has limited storage capacity, it can replicate and store only a limited number of data files. Efficient replication strategies are therefore essential to optimize storage utilization, data availability, and overall system performance. This study focuses on addressing the data file replication problem in such a distributed Grid environment while considering storage constraints and system architecture.

KEYWORDS: Data Grid, Data Replication, Scheduling

INTRODUCTION

Data-intensive scientific applications generate massive volumes of distributed data that must be processed across geographically dispersed computing resources. Data Grids provide an effective infrastructure for integrating storage, computation, and networking resources to support such large-scale collaborative environments. In these systems, Grid sites execute jobs that require access to multiple data files, which may be stored at remote locations. Retrieving remote data introduces significant communication delay and bandwidth consumption, thereby increasing overall job execution time.

Since each Grid site has limited storage capacity, it is not feasible to replicate all data files everywhere. Consequently, efficient data replication strategies are required to determine where replicas should be placed in order to minimize total access cost while satisfying storage constraints. The data replication problem in Data Grids is computationally challenging and has been shown to be NP-hard, making optimal solutions impractical for large systems. Existing approaches either rely on heuristic methods without formal guarantees or focus on theoretical models with limited applicability in distributed environments. In this paper, we present a performance-aware data replication framework that balances theoretical guarantees with practical implementation. The replication problem is modeled using a graph-theoretic approach, and a centralized greedy algorithm with provable performance bounds is proposed. To improve scalability and adaptability, the centralized solution is extended into a distributed replication algorithm in which each site makes localized caching decisions based on observed access traffic. The framework incorporates a Centralized Replica Catalogue (CRC) and a Nearest Replica Catalogue (NRC) to efficiently locate replicas and manage updates.Simulation results demonstrate that the proposed approach significantly reduces total access cost and adapts effectively to dynamic access patterns compared to traditional replication strategies. The proposed framework offers an efficient and scalable solution for data replication in large-scale Data Grid environments.

RELATED STUDY

This section outlines, some noteworthy contributions has been reported :

In a closely related study, Cibej et al. [1] treat data replication in Data Grids as a static optimisation problem. According to their research, the replica placement problem is both NP-hard and non-approximable, meaning that there isn't a polynomial-time approximation method unless P = NP. They offer integer programming solutions as well as more straightforward approaches. However, their method only takes into account static replication, which is unable to adapt to user access patterns that are always changing. Moreover, centralised integer programming techniques are difficult to apply in distributed Data Grid environments. A.Aazami, et al. [2] proposes a near-optimal strategy for replicating audio and video clips in wireless home-to-home (H2O) ad-hoc networks to

maximize the number of simultaneous streaming users. The authors prove that the number of replicas for each clip should be proportional to the square root of the product of its bandwidth requirement and frequency of access ($\sqrt{(\beta_i \times f_i)}$), which minimizes total bandwidth consumption and reduces the average distance between users and media copies. Through analytical modeling and simulations on string and grid topologies, they show that this square-root-based replication strategy significantly outperforms alternatives based on size, bandwidth alone, popularity alone, or random replication, making it highly effective for efficient storage and bandwidth utilization in continuous media streaming systems. I. Foster et al. [15] Large datasets are processed and generated by several loosely connected activities in domains like bioinformatics and high energy physics, necessitating the deployment of geographically dispersed systems known as data grids. The complexity of scheduling in these kinds of settings stems from the need to balance a number of variables, including response time, resource utilisation, allocation policies, and coordination among several compute, storage, and network resources. According to data access patterns, the framework that is being presented offers flexible scheduling in which data transfer (replication) can be managed independently through an asynchronous process or closely connected with job scheduling. According to simulation studies, data migration and work scheduling may not necessarily need to be closely related, enabling them to be controlled separately and streamlining the system design overall, even when replication has an impact on scheduling performance. A framework for managing massive data in smart grids by A. Zainab et al. [20] They clarify that traditional systems are unable to effectively manage the vast amounts of historical and real-time data generated by smart grids. They propose a three-phase design to address this, which includes data gathering, Hadoop (HDFS)-based distributed storage, and Apache Spark-based machine learning analytics. The necessity of scalable and distributed technology is emphasised in the study along with issues like data size, security, and real-time processing.

PROBLEM STATEMENT

A Data Grid consists of multiple interconnected sites arranged hierarchically, with a top-level site managing a centralized Replica Catalogue (RC) that maintains information about data files and their replica locations across institutional sites. Each site has both computational and storage capabilities, and users submit jobs that require access to distributed data files. Since each data file originates from a specific source site and every Grid node has limited storage capacity, it is not feasible to store all replicas at every site. Therefore, the core problem is to determine an efficient data replication strategy that decides which files should be replicated at which sites so that data access is optimized while satisfying storage constraints.

PROPOSED SYSTEM ARCHITECTURE

The proposed system architecture begins with a Centralized Server that executes the Nominal Distribution Algorithm to determine optimal replica placement. Once the replication decision is made, data files are replicated, registered, and transferred to the designated sites. During the transfer process, the system continuously monitors for faults. If no fault occurs, the data is distributed to the Distributed Replica Servers (S1, S2, …, Sn), where CRC/NRC updates are performed, followed by redistribution to the nearest replicas and implementation of localized data caching. In case of a fault, the process is redirected to the Optimal Makespan module, where failed jobs are placed into a Job Pool and rescheduled using an efficient scheduling mechanism. After successful execution, the system updates the replica catalogues and resumes normal operation. This integrated framework ensures efficient replica distribution, adaptive caching, fault tolerance, and optimized job scheduling within the Data Grid environment.

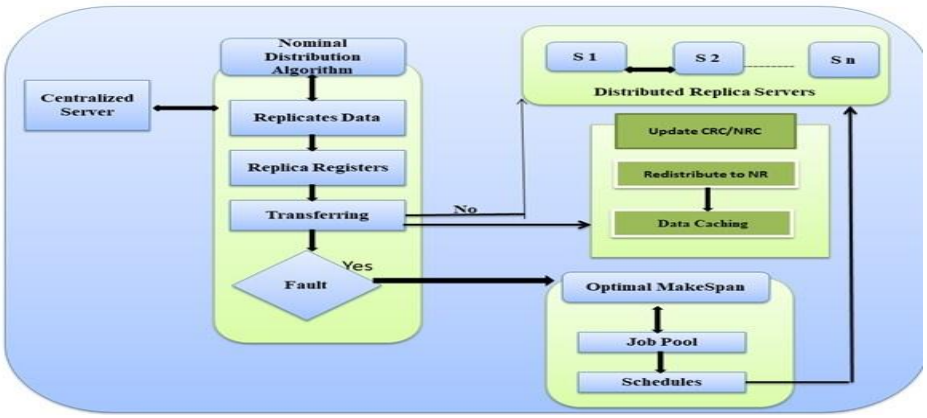

Fig : 1 Proposed System Architecture

ISSN 2349-0292
Impact Factor 3.802

RESULT ANALYSIS

The performance of the proposed replication strategy was evaluated through extensive simulations using a Java-based simulator. The proposed Optimal Algorithm was compared with existing approaches including Greedy, Local Greedy, and Random algorithms by varying parameters such as the number of files, storage capacity, and number of Grid sites .The results demonstrate that the Optimal Algorithm consistently achieves lower Total Access Cost compared to the Greedy, Local Greedy, and Random strategies The Greedy and Local Greedy approaches replicate files immediately upon generation, while the Random strategy makes arbitrary replication decisions, leading to higher access costs. In contrast, the proposed algorithm strategically places replicas at appropriate locations, ensuring better utilization of limited storage capacity It was also observed that the Total Access Cost remains stable even when the number of Grid sites increases, highlighting the scalability of the approach. Additionally, the integration of the Makespan algorithm enhances system reliability by handling faulty sites and rescheduling jobs efficiently. Overall, the experimental evaluation confirms that the proposed Optimal replication strategy significantly reduces data access cost and improves system performance compared to traditional replication methods.

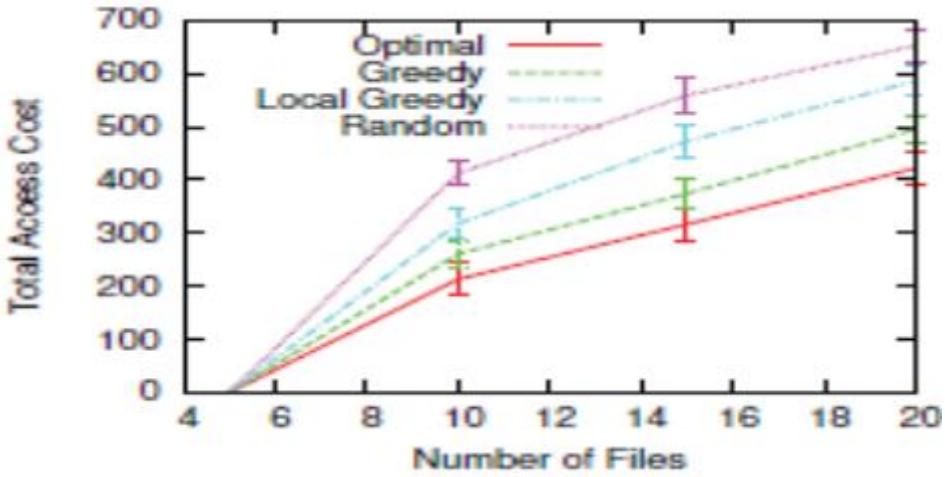

Fig: 2 Varying number of files

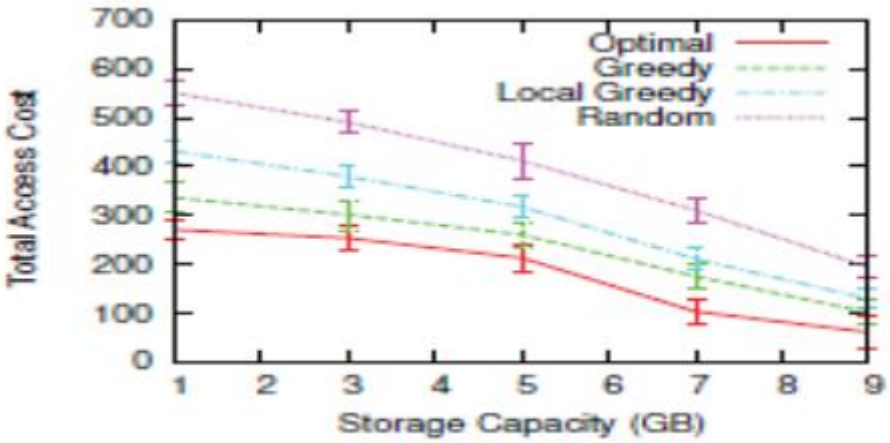

Fig: 3 Varying storage capacity

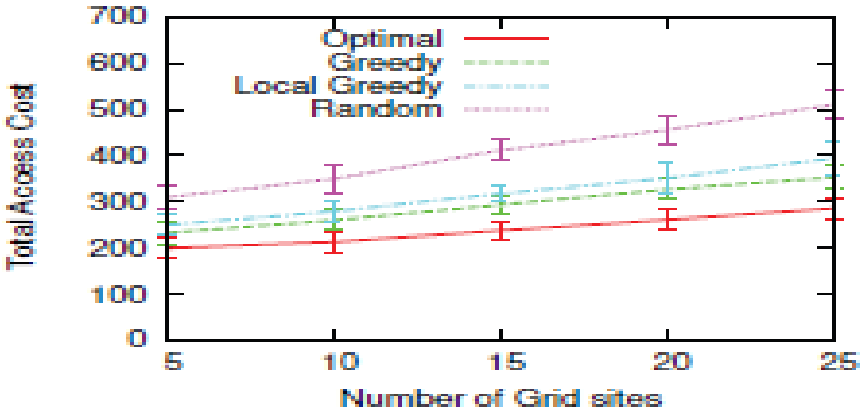

Fig: 4 Varying Number of Grid Sites

CONCLUSION AND FUTURE SCOPE

In data-intensive scientific applications, Grid sites execute multiple jobs that require distributed input data files. When files are not locally available, they must be transferred from remote sites, increasing execution time. This work addressed the data replication problem by modeling it as a graph-theoretical optimization problem with storage constraints. The proposed replication strategy aims to minimize overall job execution time by intelligently placing replicas across Grid sites. Experimental results demonstrate that the proposed optimal algorithm outperforms Greedy, Local Greedy, and Random approaches in terms of total access cost. Furthermore, the integration of the Makespan scheduling mechanism improves fault handling, while the use of Centralized Replica Catalogue (CRC) and Nearest Replica Catalogue (NRC) ensures efficient identification of nearby replicas. Overall, the proposed approach provides an effective and scalable solution for geographically distributed Data Grid environments.

Future work can extend this replication framework to large-scale cloud computing environments, where higher storage capacity can be leveraged to further optimize replica placement. By incorporating data derivation history and intelligent decision-making mechanisms, replication strategies can be made more adaptive and cost-efficient. Additionally, a more dynamic replication model can be developed in which each site automatically adjusts its data observation window based on real-time traffic patterns, leading to improved adaptability and performance in highly dynamic distributed systems.

REFERENCES

[1] Cibej, B. Slivnik, and B. Robi , (“The Complexity of Static Data Replication in Data Grids,” Parallel Computing, vol. 31, nos. 8/9, pp. 900-912, 2005.

[2] A. Aazami, S., Ghandeharizadeh, and T. Helmi. Near optimal number of replicas for continuous media in ad-hoc networks of wireless devices. (In Proceedings of International Workshop on Multimedia Information Systems, 2004.)

[3] B. Alco, J. Bester, J. Brenham, A.L. Chervenak, C. Kesselman, S. Meder, V. Nefedova, D. Quesnel, S. Tuecke, and I. Foster. Secure, efficient data transport and replica management for high-performance data-intensive computing.(In Proceedings of IEEE Symposium on Mass Storage Systems and Technologies, 2001.)

[4] W. H. Bell, D. G. Cameron, R. Cavajal-Schiaffino, A. P. Millar, K. Stockinger, and F. Zini. Evaluation of an economy-based file replication strategy for a data grid. (In Proceedings of International Workshop on Agent Based Cluster and Grid Computing at CCGrid 2003.)

[5] D. G. Cameron, A. P. Miller, C. Nicholson, R. Carvajal-Schiaffino, K. Stockinger, and F. Zini. Analysis of scheduling and replica optimization strategies for data grids using optorsim.(Journal of Grid Computing, Volume 2,No:1,pp:57-69, 2004.)

[6] M. Carman, F. Zini, L. Serafini, and K. Stockinger. Towards an economy-based optimization of file access and replication on a data grid. (In Proceedings of International Workshop on Agent Based Cluster and Grid Computing at CCGrid 2002.

[7] A. Chakrabarti and S. Sengupta. Scalable and distributed mechanisms for integrated scheduling and replication in data grids. (In 10th International Conference on Distributed Computing and Networking (ICDCN 2008)).

[8] R.-S. Chang and H.-P. Chang. A dynamic data replication strategy using access-weight in data grids.(Journal of Supercomputing,Volume 45,pp:277-295, 2008).
[9] R.-S. Chang, J.-S. Chang, and S.-Y. Lin. Job scheduling and data replication on data grids. (Future Generation Computer Systems,Volume 23,No:-(7),pp:846-860).
[10] A. Chervenak, E. Deelman, M. Livny, M.-H. Su, R. Schuler, S. Bharathi, G. Mehta, and K. Vahi. Data placement for scientific applications in distributed environments. (In Proceedings of Grid Conference 2007, Austin, Texas, September 2007).
[11] A. Chervenak, R. Schuler, C. Kesselman, S. Koranda, and B. Moe. Wide area data replication for scientific collaboration. (In Proceedings of The 6th IEEE/ACM International Workshop on Grid Computing, 2005).
[12] N. N. Dang and S. B. Lim. Combination of replication and scheduling in data grids.(International Journal of Computer Science and Network Security, Volume 7,No:-(3),pp:304-308, March 2007).
[13] J. Rehn et. al. Phedex: high-throughput data transfer management system.(In Proceedings of Computing in High Energy and Nuclear Physics (CHEP 2006)).
[14] I. Foster. The grid: A new infrastructure for 21st century science. Physics Today, 55:42-47,2002.
[15] I. Foster and K. Ranganathan. Decoupling computation and data scheduling in distributed data-intensive applications. (In Proceedings of the 11th IEEE International Symposium on High Performance Distributed Computing, HPDC-11, pages 352-358, 2002).
[16] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. (In Proceedings of ACM International Conference on Mobile Computing and Networking (MOBICOM), 2000).
[17] J. C. Jacob, D.S. Katz, T. Prince, G.B. Berriman, J.C. Good, A.C. Laity, E. Deelman, G.Singh, and M.-H Su. The montage architecture for grid-enabled science processing of large, distributed datasets.(In Proceedings of the Earth Science Technology Conference, 2004).
[18] S. Jin and L. Wang. Content and service replication strategies in multi-hop wireless mesh networks. (In Proceedings of ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), 2005).
[19] H. Lamehamedi, B. K. Szymanski, and B. Conte. Distributed data management services for dynamic data grids.
[20] A.Zainab, A. Ghrayeb D. Syed, H. Abu-Rub, S. S. Refaat, And O.Bouhali2"Big Data Management in Smart Grids:Technologies and Challenges."