Computers and Electrical Engineering 57 (2017) 281–293

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

# A prediction-based dynamic replication strategy for data-intensive applications☆

Vijaya Nagarajan*, Mulk Abdul Maluk Mohamed

*Software System Group, Department of Computer Science and Engineering, M. A. M. College of Engineering and Technology, Tiruchirappalli, Tamil Nadu, India*

ARTICLE INFO

*Article history:*
Received 31 December 2015
Revised 28 November 2016
Accepted 28 November 2016
Available online 7 December 2016

*Keywords:*
Replication
Intelligent Replica Manager
Association rules
Modified apriori algorithm
Prediction

ABSTRACT

Data-intensive applications produce huge amount of data sets which need to be analyzed among geographically distributed nodes in grid computing environment. Data replication is essential in this environment to reduce the data access latency and to improve the data availability across several grid sites. In this work, an Intelligent Replica Manager (IRM) is designed and incorporated in the middleware of the grid for scheduling data-intensive applications. IRM uses a Multi-criteria based replication algorithm which considers multiple parameters like storage capacity, bandwidth and communication cost of the neighboring sites before taking decisions for the selection and placement of replica. Additionally, future needs of the grid site are predicted in advance using modified apriori algorithm, which is an association rule based mining technique. This IRM based strategy reduces the data availability time, data access time and make span. The simulation results prove that the proposed strategy outperforms the existing strategies.



## 1. Introduction

With the growing technological advancements in the scientific field, the modern instruments and simulation tools used in e-science applications produce huge amount of data sets. These data sets need to be analyzed and distributed to the researchers located across diverse geographical regions. Here Grid serves as a promising infrastructure by integrating globally distributed heterogeneous resources across different administrative domains [1]. Eventually, grid can be classified into two major categories such as computational grids and data grids. The objective of computation grids is to split the computation into several parts and execute them across the different resources in the grid. The objective of the data grids is to handle huge amount of data sets and to distribute them among several grid resources. Scientific applications can be categorized as computation-intensive and data-intensive. As computation intensive applications demands more CPU usage, the data-intensive applications process data ranging from tera bytes to peta bytes. Applications such as global weather prediction, Digital sky project, Brain imaging analysis, mammographic analysis and high energy physics produce huge data sets which need to be transferred, processed and analyzed across distributed data repositories [2]. This work concentrates on scheduling of data-intensive applications in the grid environment. Each grid site consists of data hosts and computation hosts. Any e-science application can be considered as set of independent jobs assigned to various grid sites and each job require large amount of data sets stored in various data hosts. In a data grid environment, scheduling data-intensive application is a

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. M. S. Kumar.

* Corresponding author.
*E-mail addresses:* ssg_vijaya@mamce.org (V. Nagarajan), ssg_malukmd@mamce.org (M.A. Maluk Mohamed).

http://dx.doi.org/10.1016/j.compeleceng.2016.11.036

challenging task. Since these applications have large-scale runs and also thousands and thousands of tasks where each task in turn processes hundreds and hundreds of input files where the size of each input data sets is huge in terms of petabytes. Hence, the following issues like Heterogeneity, Granularity, Replication, Storage, Security, Fault tolerance, and Locality are to be taken care of during the planning, execution and storage phases [2–8].

Replication is one of the primary issue which have a major impact in the make span. In order to reduce the access latency, bandwidth and storage server load in the internet, most frequently accessed data sets may be replicated across different sites. Imagine datasets are stored in site 1 and these data sets are needed for the execution of a task at site 2. Now the data sets at site 1 is replicated and the replica is sent to site 2 to improve the performance of job executed at site 2. In future if the same data sets are needed for the tasks executed at site 3 which is nearer to site 2 then datasets will not be transferred from site 1(the permanent storage of data sets) but the replicas available at site 2 may be used. Based on the locality of data sets, temporal or spatial data locality, the replication strategy adopted may be changed.

Data replication not only optimizes the data access cost but also the following [9,10]:

- *Availability:* When a job failure or resource failure happens at a particular site then the system can restore the replicated data from the other site. This will enhance the availability of the data.
- *Reliability:* When replica is available at all sites, then the probability of servicing the user request will be high. Hence the system is more reliable.
- *Performance:* The data access delay and make span is improved, when the replica is available nearer to the execution site.

There are several challenges associated with the dynamic replication [11]. In the grid environment, replication is a serious issue because grid is dynamic in nature. Users may join and leave the virtual organization of the grid at any time. Hence the replication strategy should adapt to the changing nature of grid in order to provide better performance. The replication strategy must be designed according to the topology of the grid. The data grid may have different architectures: Multi-tier architecture, Hierarchical architecture, Graph based architectures, peer to peer architecture and hybrid architecture. Data replication strategy depends upon dynamic decision making which involve when to replicate the data, where to replicate the data and which data has to get replicated. Even if a strategy is adopted, that should ensure the benefit of the replication should always be higher than the cost of the replication. Applying optimization techniques to data replication results in faster data access, increased data availability and decreased make span.

The main contributions of this paper are:

(1) A unique model is proposed where Intelligent Replica Manager (IRM) is designed for scheduling data-intensive applications in grid by considering multiple parameters for replica selection and placement.
(2) A novel Multi-criteria based replication algorithm is proposed and deployed in IRM. The algorithm considers multiple parameters like storage capacity, bandwidth and communication cost of the neighboring sites before taking decisions for placement of replica.
(3) IRM uses association rule based mining technique that predicts the datasets needed for a site accurately by finding the frequent data sets before replication. This IRM based strategy reduces the data availability time, data access time and job execution time.

This paper is structured as follows: Section 2 presents the problem formulation. Section 3 describes the related works. Section 4 proposes IRM. Section 5 describes the simulation results. Finally, In Section 6, we conclude our discussion with future plans of our work.

## 2. Problem formulation

Even though many issues are identified as challenging in the grid scheduling process, data replication plays a prominent role as long as data-intensive applications are considered. During scheduling transferring huge data sets from one site to another site requires more network bandwidth. Also, the delay incurred during the data transfer and the data availability will result in degradation of performance and affect the make span. Optimization of make span is the ultimate aim of the grid scheduling process. In this work, when a task executing at a site requires data sets for further processing, then it places the request to the IRM. The IRM will store the request in the Knowledge Base (KB) and the replica selector present in the IRM will search the data catalogue for finding the location of the required data sets. If the data sets are available in more than one location, then the least cost table is searched to find the location with minimum cost. The Least cost table is constructed based on the bandwidth of the sites. If the bandwidth of a site is high then the cost assigned is low. So the site with the least cost is selected for forwarding the request. At the remote IRM, when a new request arrives, it is stored in the KB. The correlation analyzer will find the frequent data sets using the modified apriori algorithm. Hence, the frequently used data sets and currently requested data sets are sent to the requesting site. This pre-fetching of datasets will reduce the overhead of requesting site to place the request again. This in turn reduces the queuing delay, data access cost and data availability time. If the space is not available for storing the predicted data sets in the requesting site, then the deletor present in the IRM will delete the least frequently used replica and the storage space is allocated to the predicted data sets.

Let us assume an application W consists of n number of tasks $W = \{t_1, t_2, t_3 \ldots\ldots t_n\}$ and these tasks are data-intensive in nature where huge amount of data sets $d_1$, $d_2$, $d_3 \ldots\ldots d_m$ are to be processed. The performance of the application processing is characterized by the following parameters [12].