A VO-based Two-Stage Replica Replacement Algorithm

Tian Tian, Junzhou Luo

School of Computer Science and Engineering, Southeast University,
210096 Nanjing, P.R. China
{tian_tian,jluo}@seu.edu.cn

Abstract. Due to high latency of the Internet, it becomes a challenge to access large and widely distributed data quickly and efficiently on data grids. Replication is a process to address this issue, by storing data in different locations to reduce access latency and improve data locality. However, because of the limited storage capacity, a good replica replacement algorithm is needed to improve the efficiency of the access to the replicas. In this paper, a VO-based two-stage replica replacement algorithm is proposed, which provides a good solution to deal with the relations between value and cost. In a VO, replica value is determined according to popularity to make sure which replica will be replaced, and the bandwidth is predicted to make replacement cost as low as possible. Experiments show that compared with traditional replacement algorithms our new replica replacement algorithm shows better performance and efficiency of the data access on Data Grids.

Keywords: Data Grid, Replica Replacement

1 Introduction

Data Grid [1] is a new kind of data management infrastructure to manage large amounts of data produced by some large-scale data-intensive science research. It enables geographically distributed data to be accessed and analyzed by different communities of researchers which are always geographically distributed. The main challenge Data Grid faced with is how to well support efficient data access and sharing, which is hindered mostly by the high latency of Wide Area Networks. Therefore, large amounts of data need to be replicated at several distributed sites, to make data access as quick as possible.

As the study on Data Grid becoming more popular, techniques of replica management become increasingly well into research [2, 3]. Replica management service discovers the available replicas and selects the best replica that based on some selection criteria. However, because of the limited storage capacity, an efficient replica replacement algorithm is needed to replace low value replicas with high value replicas.

The more popular replica replacement algorithm now is based on an economic model. In the algorithm, a grid is recognized as a market, and the eventual purpose of the algorithm is to maximize the profits gained by the storage sites and to minimize the cost paid by the computing sites. However, this kind of model just considers the value factor well during the replacement. In real circumstances, cost factors such as network latency and bandwidth consumption should be taken into account.

Such kind of research work has also been done in Southeast University. A prediction-based and cost-based replica replacement algorithm was proposed in [7], which combined prediction factor and cost factor together. The product of the prediction factor and the cost factor is considered to be a new replacement factor. The main disadvantage of this algorithm is that a replica with high predicted value may be replaced because of its too much cost. Therefore, how to find a point at the best combination of value and cost is the key in replica replacement algorithms.

Moreover, all prediction methods for replica value are based on some particular file access patterns [4] and the content similarity between files. The access pattern itself is a variation. A method adaptive to one access pattern may not perform well if access pattern changes. In addition, the content similarity between files is still an open problem, which has great difficulties in implementation. Therefore, the prediction methods do not have good performance in real grid systems even if they perform well in simulation.

A VO-based two-stage replica replacement algorithm is proposed in this paper. In the first stage, replica value is calculated according to popularity, and higher value replicas are confirmed to replace low value replicas. In the second stage, replacement cost can be made as low as possible through the prediction of network bandwidth periodically.

The rest of the paper is structured as follows. In section 2, related work is discussed briefly in the research field of replica replacement. In section 3, a detailed description is made about our new replica replacement algorithm. Section 4 is the experiments and results of the algorithm. Section 5 gives a conclusion and discusses shortly about future work.

2 Related Work

A definition of replica replacement algorithm was made at the first time in [5].

In [6], a marketplace interaction model was used to optimize access and replication of data on a Data Grid. Optimization is obtained via interaction of the actors in the model, whose goals are maximizing the profits and minimizing the costs of data resource management. The model consists of four kinds of actors: Computing Element, Access Mediator, Storage Broker and Storage Element.

A cost-driven replication algorithm was proposed in [8], which can dynamically replicate data on the Grid. Replica decisions were driven by the estimation of the data access gains and the replica's creation and maintenance costs that based on some factors such as replica size and bandwidth. However, its bandwidth factor doesn't change dynamically.

In order to make the replica replacement cost as low as possible, a proper choice of prediction is crucial. Currently, a number of prediction approaches have been used in many fields. In [9], the authors constructed two component models, network and disk access, and then combine these performance models for a prediction of I/O performance. However, this approach may not be practical, because in a shared environment the performance may be dynamic.

Another approach for predicting is to use observations on the whole system. The Network Weather Service [10] provides the TCP/IP throughput forecasts based on observations from past end-to-end network performance. This approach of forecasting end-to-end behavior from historic performance of the entire system has been applied to predict file transfer time in some applications.

There are also some Grid simulators for study dynamic data replication algorithms. OptorSim [4] is such a simulator, which uses a prediction function to make decisions of replica replacement. The disadvantage is that in OptorSim bandwidth and replica size are fixed. As a result the cost factor is not taken into account carefully.

3 The VO-based Two-Stage Replica Replacement Algorithm

3.1 Scenarios

A local storage site may receive several file requests from computing sites. A request may request only one file or more. All the files requested form a requested file list ($F_1,\ldots,F_n$). Excluding those files which have already been stored in the storage site, the rest should be transferred from other storage sites to local storage site in the form of replicas, and these files also form a file list ($f_1,\ldots,f_m$). Replica replacement will be triggered if there is not enough space left for all the new files. As a result, a selection should start in the file list ($f_1 ,\ldots, f_m$).

3.2 The First Stage: Value Determinations

First, we need to give a definition of replica value in this algorithm. As we know, request patterns for the files can exhibit various locality properties [5], including:

Temporal Locality: Recently accessed files are likely to be accessed again.

Geographical locality (Client locality): Files recently accessed by a client are likely to be accessed by nearby clients.

Spatial Locality (File Locality): Files near a recently accessed file are likely to be accessed.

The definition of Spatial Locality was used in [11] to put forward a prediction method of replica value. In that paper, replica value is defined as the number of times that a replica will be requested in a fixed future time window. The method has two assumptions, one is sequential correlation between file requests, which means that, if a file f0 was accessed last time, files have bigger content similarity to f0 (what “near”

means exactly in the definition of Spatial Locality) will be accessed more likely next time. The content similarity is reflected by the difference of file identifiers. Assuming that there are two files f_1 and f_2 with identifiers ID_1 and ID_2 respectively, the smaller the difference $|ID_1-ID_2|$, the bigger the content similarity between file f_1 and f_2 is.

However, in the real implementation of a Data Grid, the file identifier is no more than a sign that makes a file be unique in the grid. The identifiers can not reflect the content similarity well, because they are generated randomly. For example, in European DataGrid, the UUID is a long string composed of numbers and letters, which can be obtained by some particular algorithm [15].

As mentioned above in section 1, prediction methods for replica value are based on some particular file access pattern, which itself is a variation. A method adaptive to one access pattern may not perform well if access pattern changes. And it is very hard to confirm access patterns before the replacement starts. Therefore, the prediction method of replica value based on content similarity does not have good performance in real circumstances.

In this paper, replica value is defined as something like popularity in different virtual organizations (VO). A virtual organization is a set of individuals and/or institutions defined by some sharing rules [16]. It could be, for example, consultants engaged by a car manufacturer to perform scenario evaluation during planning for a new factory, or members of an industrial consortium bidding on a new aircraft and members of a large and international high-energy physics collaboration.

From the definition we can know that members in a VO are more likely to be interested in the same content or similar content. As a result, the jobs they submit may request files that have similar content more probably. Then a definition of replica popularity is given. Replica popularity reflects how popular a file is in a VO. In a grid system, it can be expressed by the total times of requests in a fixed time period. The initial value of popularity of a file and all its replicas is zero. And the popularity will increase by one unit every time a file or its replica is requested. Meanwhile, the popularity of all files is reduced periodically, in order to prevent some disturbance from replicas that is just very popular in the past. In addition, replicas can't be replaced if their ages are smaller than a pre-configured value of age, since some relative new replicas which can show high value in future may be replaced because of their current low value. The age of a replica is the time a replica has been stored in a storage site.

Replica value in this paper can be measured by replica popularity. The bigger the replica popularity, the higher the replica value is. In a VO where members have similar interests, files with bigger replica popularity may also be requested more times in the future. So files of bigger popularity should be kept, and files of smaller popularity should be replaced when the storage capacity is out of limit.

In fact, the formation of VOs reflects Geographical locality (Client locality) well. Files recently accessed by a client in a VO are likely to be accessed by another client in the same VO. Although clients may be physically distributed, they logically form a VO with similar interests. In addition, the popularity also reflects the Temporal Locality. Recently accessed files are likely to be accessed again in the future.

According to the results in our former research, a rule was made by us that the value factor is more important than the cost factor. Storing a replica with high value will be helpful to improve the speed of data access. However, a replica with low value

does no good to the rise of data access speed, even if it has a low replacement cost. That's because it won't be requested much times in the future.

3.3 The Second Stage: Prediction of Bandwidth

At the beginning, we also give a definition of replacement cost factor, which is the quotient of replica size and bandwidth. Cost factor is an important factor during the whole process of replica replacement. Replicas with a variety of sizes have different replacement cost, even replicas of the same size may have difference in replacement cost due to different bandwidths. Our purpose is to decrease the cost factor, so the jobs will spend shorter time in waiting for replicas they need. Consequently, the grid computing will have a better performance, especially for some real-time applications. In a word, the lower the cost factor is, the better is the performance of replacement algorithm.

In the first stage, replicas which will go to the second stage have been conformed, so the size factor is fixed. Therefore, the bandwidth should be considered carefully. Algorithms employed previously do not pay much attention to dynamic bandwidth changes. Instead, bandwidths between sites are all configured as constants, which do not conform to real circumstances. There are also some algorithms predicting the bandwidth. For example, mean-based techniques, the general formula of which is the sum of previous values over the number of measurements. Another method is based on evaluating the median of a set of values. These two prediction methods are all based on historical data. However, they do not have good accuracy.

In order to improve accuracy, a new prediction method is proposed here, which is also based on historical data. By prediction, a network link with the biggest predicted bandwidth will be chosen to transfer the replica. Of course, the source site should have the replica requested.

As we know, the bandwidth of every network link changes dynamically in a range, which has a max value and a min value. As a result, the bandwidth of a link could be recognized as a random variable.

For every network link, the total traffic every time interval D is measured. Then the bandwidth of the link during the interval D can be figured out. After measuring for N times, we have a time series of bandwidth f(1),…,f(N). To improve the prediction precision, we calculate the average bandwidth $f_m(1),\ldots f_m(n)$ by

$$f_m(i) = \sum_{(i-1)m}^{im} f(j) \Big/ m \tag{1}$$

Then we predict the future network bandwidth based on this averaged time series of bandwidth. The method applied in the prediction is the minimum mean square error (MMSE) method. Assuming that the real bandwidth in the future time interval mD is $F_m(n+1)$ and the predicted bandwidth is $f_m(n+1)$, so $f_m(n+1)$ could be expressed as

$$f_m(n+1)=a_1f_m(1)+a_2f_m(2)+\ldots+a_nf_m(n)+b \quad (2)$$

By the way, $a_1,\ldots a_n$ and b are the coefficients to minimize the mean square error of the prediction, which could be expressed as $E[\{F_m(n+1)-f_m(n+1)\}^2]$. Detailed solutions to this function could be referenced by [12]. Thus, we get the predicted bandwidth of every network link in a future time window mD.

In our former research, we just chose the best one——the one with the largest bandwidth to transfer the needed replica at the beginning of the transfer. However, some problems happened in real circumstances. The network link we chose could become slower than others, which would cause the whole replacement to be delayed greatly. To the contrary, some network link that is slow previously may become faster than others. In order to address these problems, a little change was made to our algorithm. We divided the whole replica into several equal parts. At the beginning, a best network link is chosen to transfer one part of the replica. When the transfer of one part is finished, the algorithm chooses a new best link to transfer another part also based on the prediction results. The algorithm does not go to an end until all the parts of all needed replicas have been transferred to local storage site.

4 Evaluation

4.1 Experimental Setup

SEU-SOC [13] is a grid system built for the AMS (Alpha Magnetic Spectrometer) project. The AMS project [14] is an experiment with the goal of searching for antimatter and dark matter in space, which will be carried out in the year of 2008. Currently, large amounts of simulation data have been produced for storage and analysis. The computing part of the SEU-SOC consists of three virtual organizations. One is for Monte Carlo simulation, one is for data processing and the left one is for data analysis.

Our evaluation testbed is deployed on the VO for data analysis. Some machines have been chosen to do this evaluation and the topology of these machines is show in Figure 1. The jobs for data analysis are submitted to CEs (computing element), and a CE can get all files needed from the SEs (storage element). In order to make the evaluation easier, we make a rule that every CE has the same priority and processing ability as well as a storage space of 1GB. Every SE has a storage space of 2GB. A job may need only one file or more and the file size ranges from 1M to 1000M. Moreover, all machines have been installed Globus toolkit 4.0 including GridFTP server configured for data transfer.

Table 1. Some parameters in the experiment

CE numbers	3
SE numbers	8
Max Bandwidth	100Mb/sec
Workloads	10 jobs、50 jobs、100 jobs

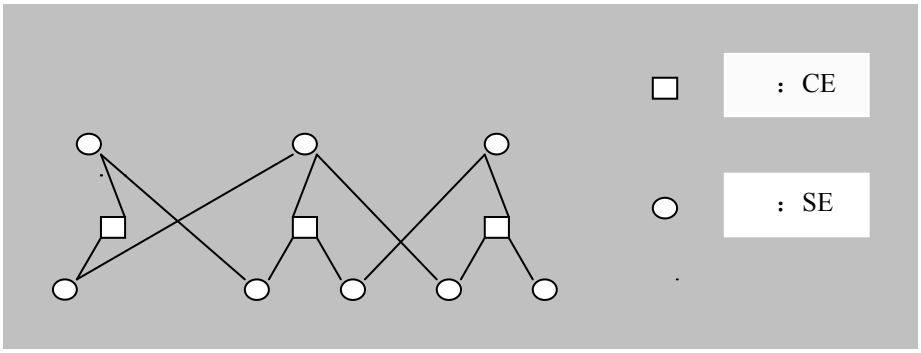

Fig. 1. Topology of the experiment

4.2 Results and Discussion

Some traditional replica replacement algorithms are introduced first for a comparison. For example, least frequently used algorithm (LFU), least recently used algorithm (LRU) and random replacement algorithm.

The purpose of the research of replica replacement algorithm is to improve the grid performance, which can be best evaluated by the parameter of mean job time. As Table 1 shows, we submitted 10 jobs, 50 jobs, 100 jobs separately in order to have a look at the performance of our new replica replacement algorithm (RPA1) under different levels of workloads. We also made a comparison of mean job time among RPA1, LFU and LRU. The results depicted by Figure 2 indicate that RPA1 algorithm has a smaller mean job time and shows better performance as workload increases. It is because that keeping high value replicas reduces the times of requesting replicas from distant sites. Additionally, since prediction is based on the historical data, which will show more accuracy as time increases, a better prediction of bandwidth would happen more likely.

Next, we make a little modification of RPA1. We don't predict the bandwidth and choose the best one. Instead, the algorithm which we called RPA2 picks a network link randomly to transfer replicas. There is another algorithm which has been introduced in section 3.3. Instead of dividing a replica into several equal parts and predicting the bandwidth periodically, it makes a prediction of bandwidth just at the beginning and chooses the best one to transfer the whole replica. We call this algorithm RPA3 as short. Figure 3 shows the mean job time of RPA1, RPA2 and RPA3 for 10 jobs, 50 jobs and 100 jobs. As we can see, RPA1 and RPA3 have shorter mean job time than RPA2, which proves that the prediction of bandwidth has a good effect on grid performance. Moreover, RPA1 has a smaller mean job time than RPA3. The

reason is that the bandwidth of a link changes dynamically. Once the chosen best link begins to slow down, the transfer might cost more time. Therefore, predicting the bandwidth periodically and transferring one part of a replica one time could reduce this kind of risk.

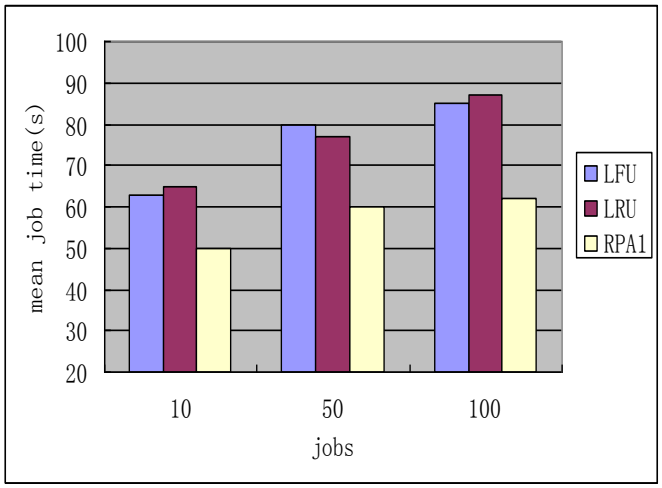

Fig. 2. Mean job time of RPA1, LRU and LFU for 10 jobs, 50 jobs and 100 jobs

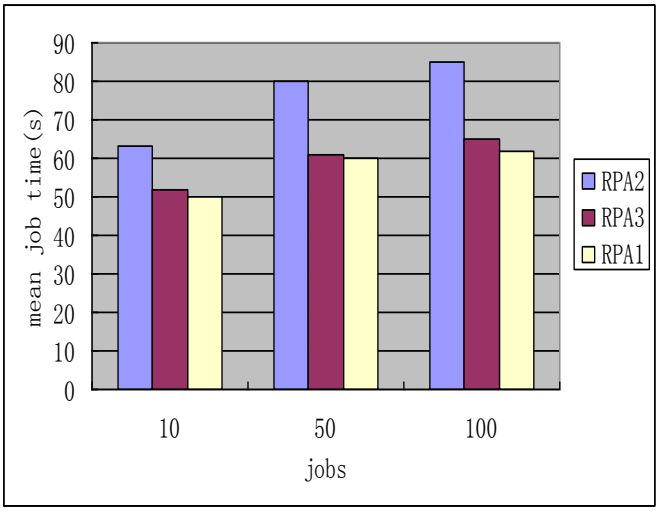

Fig. 3. Mean job time of RPA1, RPA2 and RPA3 for 10 jobs, 50 jobs and 100 jobs

The results shown in Figure 4 indicate that RPA1 decreases the bandwidth consumption. It is mainly because the increase in replica value reduces the times of replica replacement. Of course, the prediction of bandwidth also contributes to it.

In all, our promising results demonstrate that our new algorithm improves the grid performance.

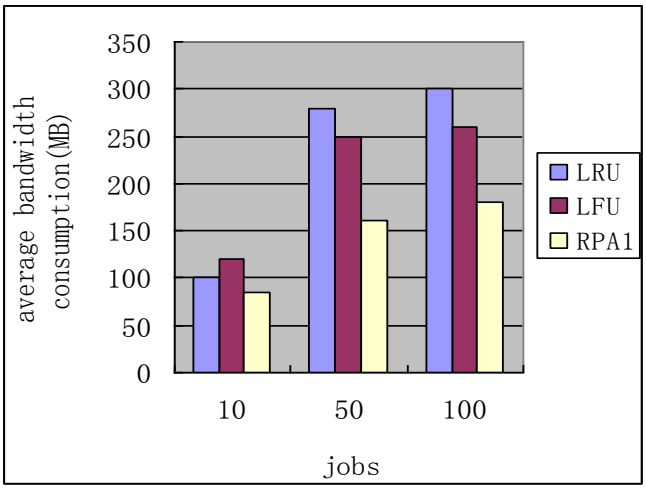

Fig. 4. Average bandwidth consumption for 10 jobs, 50 jobs and 100 jobs

5 Conclusion

A VO-based two-stage replica replacement algorithm is proposed in this paper. We find it is very hard to make a prediction of replica value in real grid systems, for the file access pattern can't be easily determined. Therefore, replica value is calculated according to popularity in the first stage, and high value replicas are fixed to replace low value replicas. In the second stage, replacement cost can be made as low as possible through the prediction of network bandwidth periodically. The experiment results demonstrate that our new algorithm contributes to better grid performance.

In future work, we plan to use multiple replicas to improve the speed of data transfer during the replacement process, and design a new mechanism to meet QoS requirements in parallel transfer if needed.

Acknowledgement

This work is supported by National Natural Science Foundation of China under Grants No. 90412014 and 90604004 and Jiangsu Provincial Key Laboratory of Network and Information Security under Grants No. BM2003201.

References

1. A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, S. Tuecke, "The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets", Journal of Network and Computer Applications, 23:187-200, 2001
2. L. Guy, P. Kunszt, E. Laure, H. Stockinger K. Stockinger, "Replica Management in Data Grids", Technical report, GGF5 Working Draft, July 1, 2002
3. D. Cameron, J. Casey, L. Guy, P. Kunszt, S. Lemaitre, G. McCance, H. Stockinger, K. Stockinger. "Replica Management in the European DataGrid Project", Journal of Grid Computing (2004) 2: 341–351, Springer 2005
4. William H. Bell, David G. Cameron, Luigi Capozza, A. Paul Millar, Kurt Stockinger, Floriano Zini, "OptorSim - A Grid Simulator for Studying Dynamic Data Replication Strategies". International Journal of High Performance Computing Applications, Vol. 17, No. 4, 403-416 (2003)
5. K. Ranganathan, I.Foster, Identifying Dynamic Replication Strategies for a High Performance Data Grid , 2nd International Workshop on Grid Computing (Grid2001), Denver,Colorado,USA.
6. M. Carman, F. Zini, L. Serafini, K. Stockinger, "Towards an Economy-Based Optimization of File Access and Replication on a Data Grid". Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID.02).May 2002
7. M Teng, L Junzhou, "A Prediction-based and Cost-based Replica Replacement Algorithm Research and Simulation". Proceedings of the 19th International Conference on Advanced Information Networking and Applications, Volume 1, 935 – 940,2005
8. Lamehamedi, H. Zujun Shentu Szymanski, B. Deelman, E. "Simulation of Dynamic Data Replication Strategies in Data Grids", Parallel and Distributed Processing Symposium, 2003. Proceedings. International, Page: 100.2, 2003
9. Shen,X. and A. Choudhary,A Multi-Storage Resource Architecture and I/O, Performance Prediction for Scientific Computing. Proceedings of 9th IEEE Symposium on HPDC (2000)
10. Rich Wolski, Neil T. Spring, Jim Hayes, The Network Weather Service: A Distributed Resource Performance Forecasting Service for Metacomputing, The Journal of Future Generaton Computer Systems Page 757-768(1999)
11. Luigi Capozza, Kurt Stockinger, Floriano Zini, "preliminary evaluation of revenue prediction functions for economically effective file replication", DataGrid-02-TED-020724, July 24, 2002
12. Lonnie C. Ludeman, "Random Processes: Filtering, Estimation, and Detection". John Wiley&Sons, New Jersey, 2003, 608 pages, ISBN 0-471-25975-6
13. Luo Junzhou, Song Aibo, Zhu Ye, Wang Xiaopeng, Ma Teng, Wu Zhiang, Xu Yaobin, Ge Liang. Grid Supporting Platform for AMS Data Processing. In: Guihai Chen, Yi Pan, Minyi Guo eds. Parallel and Distributed Processing and Applications – ISPA 2005 Workshops. Nanjing, China: Springer, Nov. 2005.276-285.
14. P.Fisher, A.Klimentov, A.Mujunen, J.Ritakari. AMS Ground Support Computers for ISS mission. AMS Note 2002- 03-01, March 12, 2002
15. Heinz Stockinger, Andrew Hanushevsky. Http Redirection for Replica Catalogue Lookups in Data Grids. Proceedings of the 2002 ACM symposium on Applied computing, Pages: 882 – 889, 2002
16. Ian Foster, Carl Kesselman, Steven Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations International Journal of High Performance Computing Applications, Vol. 15, No. 3, 200-222 (2001)