Intelligence Balancing for Communication Data Management in Grid Computing

Jong Sik Lee

School of Computer Science and Engineering, Inha University
Incheon 402-751, South Korea
jslee@inha.ac.kr

Abstract. This paper describes a design and development of an intelligence balancing for communication data management in a grid computing. The intelligence balancing allows to execute a complex large-scale grid computing system and share dispersed data assets collaboratively. After a brief review of communication data management, we discuss the design issue of the intelligence balancing. We analyze system performance and scalability, especially with non-intelligence and intelligence-based configuration. The empirical result on the heterogeneous OS distributed system indicates the intelligence balancing is effective and scalable due to the use of communication data reduction.

1 Introduction

A popular trend to execute complex and large-scale distributed systems with reasonable computation and communication resources has been focused on a grid computing. Actually, distributed computing is a subset of grid computing. Grid computing approaches are being applied to a growing variety of systems including process control and manufacturing, military command and control, transportation management, and so on. In order to provide a reliable answer in reasonable time with limited communication and computation resources, a methodology for reducing the interactive data transmission is required in a grid computing environment. In this paper, we propose an intelligence balancing for communication data management to promote the effective reduction of data communication in a grid computing environment.

This paper is organized as follows: Section 2 reviews communication data management [1,2,3]. Section 3 discusses the intelligence balancing for communication data management with an application: satellite cluster management. Section 4 analyzes performance effectiveness and system scalability. Section 5 illustrates a testbed for experiment and evaluates system performance. The conclusion is Section 6.

2 Communication Data Management

A large-scale grid computing system requires an achievement of real-time linkage among multiple systems, and thus has to execute complex large-scale execution and to share dispersed data assets and computing resources collaboratively. The methodology to support the reduction of the interactive messages among grid computing components is called a “communication data management.” It is the goal of a data traffic reduction scheme that a large-scale grid computing is performed with reasonable communication and computation resources. To perform a data traffic reduction scheme reliably, flexibility and efficiency are required. Flexibility does not indicate anything specific to any particular problem domain or technology, but rather indicates being general in nature. Efficiency requires the scaling of grid computing from very small to very large along many dimensions including numbers of the grid component objects, complexity of interactions, fidelity of representations, and computational/network resources.

3 Intelligence Balancing for Communication Data Management

For performance improvement of a complex large-scale grid computing system, we propose an intelligence balancing for communication data management which reduces the required transmission data by providing an intelligence to a transmission-related component. The proposed scheme indicates intelligence distribution to each grid component from a centralized intelligence-related component. The intelligence balancing reduces system execution cost by reducing the communication cost among components with separated small intelligence. The intelligence indicates that each component operates with its own decision beside its basic operations. The scheme increases system modeling flexibility and improves system performance through communication data reduction and computation synchronization and system scalability for a large-scale grid computing system. Note that the component with intelligence needs more computation resource. However, the reduction required transmission data improves a total system performance instead of growth of local computation. In this paper, we apply to the intelligence balancing to a satellite cluster management system [4,5]. We introduce the ground system operation as a case study to discuss non-intelligence and intelligence approaches and evaluate system performance. A ground system commands and controls a cluster of spacecrafts. Basically, a ground system requires operations and manpower to monitor the cluster, makes a decision, and sends the proper command strings. The intelligence approach indicates that it separates ground functions and distributes a set of functions to spacecrafts. Performing a set of functions requires intelligence of spacecraft. Here, we classify the degree of intelligence: low and high. In case of low intelligence, a ground station separates four regions to be observed, makes four different command strings, and sends them to a cluster manager. The cluster manager parses the command strings and forwards them to each proper spacecraft. The parsing and forwarding classifies a lower intelligence of the cluster manager. In case of high intelligence, the ground station does not separate four regions to be observed and sends a total region to the cluster manager. The cluster manager should include the intelligence for division of region to be observed. The division intelligence should understand the technology including region division, image capturing, image visualization, image data transmission, and so on.

4 Performance Analysis

To analyze the system performance of the intelligence balancing, we take the amount of required satellite transmission data which are between ground station and spacecrafts. Notice that transmission data among spacecrafts inside cluster is ignored. As Table 1 shows, the intelligence approach significantly reduces the number of bits passed. Basically, there occurs overhead bits (H) needed for satellite communication when a ground station sends a command. The non-intelligence causes an amount of overhead bits since it makes a ground station to send spacecrafts messages individually. High intelligence significantly reduces the transmission data bits since it transmits the one big region location information irrelevant to the number of spacecrafts (N) in a cluster. The high intelligence still requires the same lower transmission data bits.

Table 1 Analysis of Transmission Data Reduction (Note: N: Number of spacecrafts in a Cluster; M: Number of Clusters; H: Number of overhead bits in satellite communication (160 bits assumed), R: Number of regions at one spacecraft on one transmission (40 assumed))

Approach		Number of bits passed
Non-Intelligence		(H + 4*64) *R*N*M
Intelligence	Low	(H + 4*64*R*N) *M
	High	(H + 4*64) *M

5 Experiment and Result

A cluster of 4 spacecrafts flies on pre-scheduled orbits. One of the spacecrafts acts as the cluster manager that communicates with the ground station. The cluster manager gathers states of each spacecraft and sends telemetry information back to the ground station. At any given time, the ground station can send an observation request to the cluster manager which, in turns, will coordinate with other spacecrafts in the cluster to perform the requested observation in synchronization. The cluster manager then aggregates data collected from the other spacecrafts and send it back to the ground station. In the platform setting, we develop a HLA[6]-compliant heterogeneous distributed system which includes various operating systems including SGI Unix, Linux, Sun Unix, and Windows. The total of five federates are allocated to five machines, respectively, and they are connected via a 10 Base T Ethernet network. As Table 2 shows, the intelligence approach apparently reduces the transmission data bits. Especially, the use of high intelligence approach greatly reduces the transmission data bits. The high intelligence approach is able to allow an execution which requires a small amount of transmission data bits regardless to the number of satellites. The system execution time considers both of communication and computation performance. The non-intelligence approach requires a large amount of communication data, however it does not need the local computation for intelligence. The intelligence approach reduces the amount of communication data and uses operations for intelligence. The system execution time for the intelligence approach is caused by both of data communication time and intelligence operation time. The reduction indicates that the time reduction from transmission data reduction is greater than the time expense from intelligence operation.

Table 2 Transmission data bits and System execution time (ms) in three approaches: Non-intelligence, Low Intelligence, and High Intelligence

#of Satellites	Non-Intelligence		Low Intelligence		High Intelligence	
4	66560	6187.43	41120	5464.75	416	5514.56
6	99840	6287.45	61600	5941.48	416	5911.48
8	133120	6476.34	82080	6273.98	416	6263.38
10	166400	6498.34	102560	6421.44	416	6414.84
12	199680	6507.68	123040	6423.04	416	6410.04

6 Conclusion

This paper presented the design and development of the intelligence balancing scheme for communication data management in a grid computing system. The proposed scheme focuses on the intelligence balancing to each grid component and various degrees of intelligence. Also, the scheme indicates the conversion from a conventional passive component, which processes with given commands, to an intelligent component, which processes with its own decision. These intelligence balancing and intelligent component concepts allow various complex executions for a variety of grid computing systems and improve system performance through data communication reduction and computation load balancing. The empirical results showed favorable reduction of communication data and overall execution time and proved the usefulness of the intelligence balancing for communication data management in a grid computing system. In future work, we will extend the scheme to a real-time grid computing system and execution infrastructure.

References

1. Boukerche and A. Roy, "A Dynamic Grid-Based Multicast Algorithm for Data Distribution Management" IEEE Distributed Simulation and Real Time Application, 2000.
2. Gary Tan et. al. "A Hybrid Approach to Data Distribution Management," 4th IEEE Distributed Simulation and Real Time Application, August 2000.
3. Katherine L. Morse, et. al., “Multicast grouping for dynamic data distribution management,” in Proceeding of SCSC99, 1999.
4. P. Zetocha, “Intelligent Agent Architecture for Onboard Executive Satellite Control, Intelligent Automation and Control,” vol. 9, TSI Press Series on Intelligent Automation and Soft Computing, Albuquerque, N.M., pp. 27–32, 2000.
5. D.M. Surka, M.C. Brito, and C.G. Harvey, “Development of the Real-Time Object-Agent Flight Software Architecture for Distributed Satellite Systems, IEEE Aerospace Conf., IEEE Press, Piscataway, N.J., to be published in 2001.
6. High Level Architecture Run-Time Infrastructure Programmer’s Guide 1.3 Version 3, 1998 DMSO