Classifiers for the Causes of Data Loss: An Important Step Towards Intelligent, Adaptive, and Efficient Communication Services

Phillip M. Dickens
Department of Computer Science
University of Maine
E-Mail: dickens@umcs.maine.edu

Abstract

The goal of this research is to develop a set of intelligent, adaptive, and efficient communication services for Grid computing. An important milestone on the path to such next-generation communication systems is the development of a classification mechanism that can distinguish between the various causes of data loss in cluster/Grid environments. In this paper, we discuss our approach to developing such a system.

Key Words: Communication protocols, Adaptive computing, Grid computing.

Introduction

Computational Grids create large-scale distributed systems by connecting geographically distributed computational and data-storage facilities via high-performance networks. A critical component of such systems is the high-performance communication infrastructure that allows the geographically distributed computational elements to function as a single (and tightly-coupled) computational platform. Given the importance of such systems to the high-performance computing community, it is vital to create communication services that are intelligent, adaptive, and efficient. An important milestone on the path to such next-generation communication systems is the development of a classification mechanism that can distinguish between the various causes of data loss in cluster/Grid environments. The idea is to use the classification mechanism to respond to data loss in a way that is appropriate for the particular set of system dynamics responsible for creating such loss. That is, when provided with such knowledge, the communication system should be able to either adapt its behavior to match current system conditions or change its execution environment.

The approach that we are pursuing is to analyze what may be termed *packet-loss signatures,* which show the distribution (or pattern) of those packets that successfully traversed the end-to-end transmission path and those that did not. These signatures are collected by the receiver and delivered to the sender upon request. Thus the packet-loss signatures are essentially large selective-acknowledgment packets, and are so named based on a growing set of experimental results [5, 6] showing that different classes of error mechanisms have different "signatures". We are applying complexity theory to the problem of learning the underlying structure (or lack thereof) of these signatures, and mapping the relationship between such underlying structure and the system conditions responsible for its generation. Our research has shown that complexity measures capture quite well the underlying system dynamics, and that understanding such dynamics provides significant insight into the root cause(s) of observed data loss.

The test-bed for this research is FOBS[1]: a high-performance data transfer system for computational Grids. FOBS [2, 4] is a UDP-based data transfer system that provides reliability through a selective-acknowledgment and retransmission mechanism. As noted above, it is precisely the information contained within the selective-acknowledgment packets that is collected and analyzed by our classification mechanism. Three important factors, whose combination is unique among high-performance data transfer

[1] Fast Object-Based data transfer System

mechanisms for computational Grids, make FOBS an excellent test-bed for this research. First, FOBS is an application-level protocol. Thus the congestion-control mechanism can collect, synthesize, and leverage information from a higher-level view than is possible when operating at the kernel level. Second, the complexity measures can be obtained as a function of a *constant* sending rate. Thus the values of the variables collected are (largely) unaffected by the behavior of the algorithm itself. Third, FOBS is structured as a feedback control system. Thus the external data (e.g., the complexity measures) can be analyzed at each control point, and this data can be used to determine the duration of the next control interval and the rate at which data will be placed onto the network during this interval. We do not discuss further the design, implementation, or performance of FOBS here. The interested reader is directed to [2, 3] for detailed discussions on these issues.

In our talk, we will show the relationship between packet-loss signatures and the system conditions responsible for their creation. Also, we will outline the mathematics behind the computation of the complexity measures of these signatures. Finally, we will discuss a new set of responses to data loss that are made possible by having knowledge of the underlying causes of such loss.

Classification of the Causes of Packet Loss

As discussed above, preliminary research has shown that different error mechanisms have different signatures. The question then is whether the statistical properties of such signatures are *different enough* to allow the construction of rigorous hypothesis tests for the causes of packet loss. To investigate this issue, we performed approximately 10,000 data transfers each of which consisted of 100 MB. In 5000 of the transfers we injected competing network traffic. In the other 5,000 trials, we created contention for CPU resources by spawning additional processes on the CPU upon which the data receiver was executing. We controlled the amount of competing network traffic, and the number of competing processes, to create loss rates in the range of 0% to 10%. We collected the complexity measures after each transfer. All experiments were conducted on the TeraGrid [1]: a high-performance computational Grid that connects various supercomputing facilities via networks operating at speeds of up to 40 gigabits per second. The two facilities used in these experiments were the Center for Advanced Computing Research (CACR, located at the California Institute of Technology), and the National Center for Supercomputing Applications (NCSA, located at the University of Illinois, Urbana). The host platforms at both facilities were IA-64 Linux clusters where each compute node consisted of dual Intel Itanium2 processors. The compute nodes at CACR were 1.3 GHz and those at NCSA were 1.5 GHz. The operating system at CACR was Linux 2.4.19-SMP, while the operating system at NCSA was Linux 2.4.21-SMP. Each compute node had a gigabit Ethernet connection to the TeraGrid network. The sending rate was a constant Gigabit per second, and the experiments were run late at night where there was little, if any, competing network traffic.

The results are shown in Figure 1. This figure shows the mean complexity value with 95% confidence intervals of the mean, for two causes of data loss, contention at the NIC[2] and contention at the CPU, for loss rates from 0% - 10%. As can be seen, the statistical properties are strikingly different, and we conclude that the development of sophisticated hypothesis tests is indeed possible.

In our talk, we will discuss our progress in constructing a Bayesian statistical model for hypothesis testing.

[2] Note that we are using contention at the NIC as a proxy for contention within the network for two reasons: First, it is very difficult to create contention in a 40 gigabit per second network. Second, in experiments performed on less powerful networks we have observed that the statistical properties of these two causes of packet loss are quite similar. This is not to say that this is always the case, and the investigation of this issue is an area of current research.

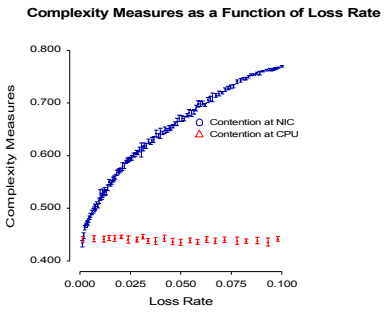

Figure 1. This figure shows the mean complexity measure and 95% confidence intervals around the mean as a function of the cause of data loss and the loss rate.

The Next Step: Adaptations Based on Classification

We define a truly adaptive communication service as one that can *adapt its behavior* in response to changes in its execution environment or to *change its execution environment*. While our research in this important area is still in its infancy, the following scenario serves as an example of each type of response.

Assume a second process is initiated on the CPU upon which a data receiver is executing, and that the subsequent contention for CPU resources causes data loss[3]. Further, assume that the classification mechanism has correctly classified the cause of data loss as being outside of the network domain. Then one option would be for the sender to experiment with different sending rates until an acceptable loss rate is obtained (the definition of such a loss rate is as-of-yet undefined, but arguably it can be greater than 0%). The second option would be to migrate the data receiver to a dedicated node. A very important point to be made is that ***neither*** option makes sense if the true cause of data loss is not known to be CPU- rather than network-related.

References:

[1] The TeraGrid Homepage.
http://www.teragrid.org

[2] Dickens, P., FOBS: A Lightweight Communication Protocol for Grid Computing.
in the Proceedings of *Europar 2003*, (2003).

[3] Dickens, P., A High Performance File Transfer Mechanism for Grid Computing.
in the Proceedings of *The 2002 Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*. (Las Vegas, Nevada, 2002).

[4] Dickens, P. and Gropp, B., An Evaluation of Object-Based Data Transfers Across High Performance High Delay Networks.
in the Proceedings of *the 11th Conference on High Performance Distributed Computing*, (Edinburgh, Scotland, 2002).

[5] Dickens, P. and Larson, J., Classifiers for Causes of Data Loss Using Packet-Loss Signatures.
in the Proceedings of *IEEE Symposium on Cluster Computing and the Grid(ccGrid04)*, (2004).

[6] Dickens, P., Larson, J. and Nicol, D., Diagnostics for Causes of Packet Loss in a High Performance Data Transfer System.
in the Proceedings of *Proceedings of 2004 IPDPS Conference: the 18th International Parallel and Distributed Processing Symposium*, (Santa Fe, New Mexico, 2004).

[3] Recall that we are discussing an application-level UDP-based system for which contention for CPU resources can be a significant cause of data loss.