Classifiers for the Causes of Data Loss Using Packet-Loss Signatures[*]

Phillip M. Dickens[1, 2] and Jay W. Larson[2]
[1]Department of Computer Science, Illinois Institute of Technology
[2]Mathematics and Computer Science Division, Argonne National Laboratory

ABSTRACT

A necessary step in the development of next-generation congestion control mechanisms is the ability to accurately classify the root cause(s) of observed data loss and to develop responses tailored to the particular cause. Toward this end, we are developing a classification mechanism based on the collection and analysis of what we term packet-loss signatures, which describe the patterns of packet loss in the current transmission window. We are exploring the application of complexity theory to the problem of learning the underlying structure (or lack thereof) of these signatures, and studying the relationship between such underlying structure and the system conditions responsible for its generation. In this paper, we describe the algorithm for determining the complexity of packet-loss signatures, show how complexity measures can be mapped to the underlying causes of packet loss, and provide experimental results demonstrating the effectiveness of our approach.

1 Introduction

Computational Grids create large-scale distributed systems by connecting geographically distributed computational and data-storage resources via high-performance networks. Such systems, which can harness and bring to bear tremendous computational resources on a single large-scale problem, are becoming an increasingly important component of the national computational infrastructure. An important area of research in Grid computing is the development of high-performance communication mechanisms that can take full advantage of the underlying bandwidth when system conditions permit, can back off in response to observed (or predicted) contention within the network, and can accurately distinguish between these two situations.

Our research is addressing the issue of identifying the root cause(s) of data loss as observed by a high-performance data transfer system during the course of its execution. The approach we are pursuing is to analyze what we term *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 our belief (supported by experimental studies) that different classes of error mechanisms have different "signatures." We are exploring the application of complexity theory to the problem of learning the underlying structure (or lack thereof) of these signatures, and studying the relationship between such underlying structure and the system conditions responsible for its generation. Our view, supported by experimental studies provided below, is that complexity measures capture quite well the underlying system dynamics and that understanding such dynamics provides significant insight into the cause(s) of observed data loss. The longer-term goal of this work is to use, in a highly adaptive and efficient data transfer system, information related to the root cause(s) of data loss. However, this paper focuses on techniques to build such classifiers; adaptations based on this knowledge will be the focus of forthcoming papers.

The testbed for this research is FOBS (Fast Object-Based data transfer System), a high-performance data transfer system for computational Grids [9, 10]. FOBS is a UDP-based transfer system that provides reliability through a selective-acknowledgment and retransmission mechanism. As noted above, it is precisely the information contained within the

[*] This work was supported by the Office of Biological and Environmental Research, U.S. Department of Energy, Office of Science, under Contract W-31-109-ENG-38.

selective-acknowledgment packets that is collected and analyzed by our classification mechanism.

Three important factors, whose combination is unique among high-performance data transfer mechanisms for computational Grids, make this research feasible and useful. First, FOBS is an application-level protocol. Thus the congestion control algorithms 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 (but is not currently) 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 [9, 10] for detailed discussions on these issues.

In this paper, we provide substantial experimental data demonstrating the effectiveness of our approach in distinguishing between contention for network resources and contention for CPU resources. This distinction is important for two reasons. First, contention for CPU cycles can be a major contributor to packet loss in UDP-based protocols such as FOBS. This happens, for example, when the receiver's socket-buffer becomes full, additional data bound for the receiver arrives at the host, and the receiver is switched out and thus unavailable to pull such packets off of the network. Second, data loss resulting from CPU contention is completely outside of the network domain and does not represent interference with other network traffic. Thus new and less aggressive responses can be developed to deal with this particular class of losses.

This paper makes two important contributions. First, it presents a simple classification mechanism that is quite powerful in its ability to distinguish between various causes of packet loss. Such distinctions are apparent even at very low loss rates (i.e., around 1%), and the distinction becomes even clearer with increasing loss rate. This represents a major milestone in the development of highly intelligent and adaptive communication mechanisms for Grid computing. Second, the approach outlined here is generally applicable to UDP-based protocols using selective-acknowledgments as part of their reliability mechanism. This work could, in fact, be used to classify and respond to different causes of packet loss by any version of TCP using the selective-acknowledgment mechanism. This paper should be of interest to a large segment of the Grid community given the interest in and importance of exploring new approaches by which data transfers can be made more intelligent and efficient.

The rest of the paper is organized as follows. In Section 2, we discuss related work. In Section 3, we present the complexity analysis used in this paper and show how such techniques can be applied to the packet-loss signatures. In Section 4, we describe our experimental methodology. In Section 5, the results of these experiments are presented. In Section 6, we provide our conclusions and outline future work.

2 Related Work

The issue of distinguishing between categories of losses has received significant attention within the context of TCP for hybrid wired/wireless networks (e.g., [2-4, 6, 14, 20]). The idea is to distinguish between losses caused by network congestion and losses caused by errors in the wireless link, and to trigger TCP's aggressive congestion control mechanisms only in the case of congestion-induced losses. This ability to classify the root cause of data loss, and to respond accordingly, has been shown to improve the performance of TCP in this network environment [2, 14, 19]. These classification schemes are based largely on simple statistics on observed round-trip times, observed throughput, or the inter-arrival time between ACK packets [4, 5, 14]. Debate remains, however as to how well techniques based on such simple statistics can classify loss [14]. Another approach being pursued is the use of Hidden Markov Models where the states are characterized by the mean and standard deviation of the distribution of round-trip times [14]. Hidden Markov Models have also been used to model network channel losses and make inferences about the state of the channel [15].

Our research has similar goals, although we are developing a finer-grained classification system that can distinguish between contention at the NIC, contention in the network, and contention for CPU resources. Also, we believe that complexity measures may prove to be a more robust classifier than (for example) statistics on round-trip times and could be substituted for such statistics within the mathematical frameworks established in these related works. Similar to the projects discussed above, we separate the issue of classification of root cause(s) of data loss from the issue of implementing responses based on such knowledge.

Research into other application-level alternatives to TCP is also related (e.g., [1, 17, 18]). However, none of these approaches attempt to determine the root cause(s) of observed packet loss that is a major focus of our research.

3 Diagnostic Methodology

The packet-loss signatures can be analyzed as time series data with the objective of identifying diagnostics that may be used to characterize causes of packet loss. A desirable attribute of a diagnostic is that it can describe the dynamical structure of the time series. The approach we are taking is the application of *symbolic dynamics* techniques, which have been developed by the nonlinear dynamics community and are highly appropriate for time series of discrete data. We believe this approach to classifying causes of packet loss will work because of the differing timescales over which such losses occur. For example, packet loss due primarily to network-based causes such as router contention or contention at the NIC is likely to show temporal structure over a wide variety of timescales reaching down to the spacing between packets. A platform-based cause such as CPU contention at the host upon which the data receiver is executing will more likely be associated with a narrower range of longer timescales (e.g., the size of the time slice allocated to the receiver in a time-sharing system).

In symbolic dynamics [13], the packet-loss signature is a sequence of symbols drawn from a finite discrete set, which in our case is two symbols: 1 and 0. One diagnostic that quantifies the amount of structure in the sequence is *complexity*. There are numerous ways to quantify complexity. In this discussion, we have chosen the hierarchical approach of d'Alessandro and Politi [8], which has been applied with success to quantify the complexity and predictability of time series of hourly precipitation data [12].

The approach of d'Alessandro and Politi is to view the stream of 1s and 0s as a language and focus on subsequences (or *words*) of length **n** in the limit of increasing values of **n** (i.e., increasing word length). First-order complexity, denoted by $\mathbf{C^1}$, is a measure of the richness of the language's vocabulary and represents the asymptotic growth rate of the number of *admissible words* of fixed length **n** occurring within the string as **n** becomes large. The number of admissible words of length **n**, denoted by **Na(n)**, is simply a count of the number of distinct words of length **n** found in the given sequence. For example, the string **0010100** has **Na(1)** = 2 (0,1), **Na(2)** = 3 (00,01,10), **Na(3)** = 4 (001, 010, 101, 100). The *first-order complexity* (C^1) is defined as

$$C^1 = \lim_{n->\infty} (\log 2 \; Na(n)) / n \, . \qquad (1)$$

The first-order complexity metric characterizes the level of randomness or periodicity in a string of symbols. A string consisting of only one symbol will have one admissible word for each value of **n**, and will thus have a value of C^1=0. A purely random string will, in the limit, have a value of C^1=1. A string that is comprised of a periodic sequence, or one comprising only a few periodic sequences, will tend to have low values of C^1.

As noted, a hierarchy of complexity values is defined in [8]. The next level of the hierarchy is a quantity termed C^2 that captures the fact that random strings are of lower complexity than strings that have rules governing their creation. We do not discuss this quantity here because we have not yet integrated it into our classification mechanism.

Experimental Design

We performed three sets of experiments to evaluate the effectiveness of our approach. The first set compared packet-loss signatures generated when data loss was caused by contention for NIC resources and when the data loss was caused by contention for CPU cycles. The second set of experiments compared the packet-loss signatures generated when data loss was caused by contention at a router and when the data loss was caused by contention for CPU cycles at the host upon the receiver was executing. The third set evaluated the approach using a longer transfer on a shared host during normal business hours.

All data transfers were between hosts at Argonne National Laboratory (ANL) and the

National Center for Supercomputing Applications (NCSA). The host platform at ANL (Chiba City), was a Linux cluster with 256 dual Pentium III 500 MHz processors. The computational platform at NCSA (Titan) was an IA-64 Linux cluster with 128 compute nodes each consisting of dual Intel 800 MHz Itanium 1 processors. The two sites are connected by the Illinois Wired/Wireless Infrastructure for Research and Education (I-WIRE) which operates at 10 Gbps. There was no discernable contention on the I-WIRE at the time these experiments were conducted.

In the first set of experiments the data receiver executed on a dedicated processor within Chiba City, and additional compute-bound processes were spawned on this same processor to create CPU contention. As the number of additional processes increased, the amount of time the data receiver was switched out similarly increased. Since the data receiver was not available to take packets off of the network during the times it was switched-out, there was a direct relationship between CPU load and the resulting packet loss rate. We were interested in analyzing the structure of the bitmaps as a function of both the root cause of data loss (i.e., contention for CPU or NIC resources) and the loss rate. We therefore varied the number of additional processes to obtain loss rates of (approximately) 1%, 5%, 10%, and 15%.

To investigate loss patterns caused by contention for NIC resources, we initiated a second (background) data transfer. The data sender of the background transfer executed on a different node within Chiba City, and the receiver executed on the second processor within the same node as the primary data receiver. Since both processors of a given node share the same NIC, we were able to generate contention at the NIC without causing contention for CPU cycles with the two receivers. Initially, the combined sending rate was set to the maximum speed of the NIC (100 Mbps for the Chiba City compute nodes), and contention for NIC resources was increased by increasing the sending rate of the background transfer. The packet loss experienced by both data transfers was a function of the combined sending rate, and this rate was also set to result in loss rates of (approximately) 1%, 5%, 10%, and 15%.

In the second set of experiments, we used nine parallel UDP data streams (each sending at 100 Mbps) between Titan and Chiba City. We then created a tenth data stream between a HP N4000 node at the Center for Advanced Computational Research (CACR) and a BM IntelliStation Z Pro 6894 workstation within NCSA. The tenth stream shared a router with the 9 parallel UDP streams creating contention for that router's resources. The sending rate of the tenth stream was varied between 50 and 100 Mbps. When data was sent at 100 Mbps packets were dropped at the router. When data was sent at 50 Mbps, the router was able to process all ten streams. The result of interest was a time-series of the packet-loss signatures of a randomly picked stream at the receiving host (NCSA). For comparison, we also conducted experiments resulting in a time-series of packet-loss signatures when data was lost because of CPU load.

We were also interested in validating the classification mechanism under realistic operating conditions. The goal was to look at the range of complexity measures obtained over a (reasonably) long data transfer, and to determine whether those measures appeared to capture the underlying dynamics of the end-to-end system. To examine this issue, we performed a data transfer between ANL and NCSA, where the data receiver was executed on the server for the Titan cluster at NCSA. The experiments were run during normal business hours when there is often significant contention for CPU resources. The data transfer lasted approximately 80 minutes, and the congestion control mechanisms implemented in FOBS were *disabled* during the transfer. The reason for disabling the congestion control mechanisms was to separate out the impact of changes in sending rate as a factor in the complexity measures. The sending rate was set to a constant 100 Mbps, which is well below the capacity of the network (10 Gigabits per second) and the NIC (Gigabit ethernet). We were interested in the first-order complexity measures after each 100 MB chunk of data that was sent. We calculated only the complexity measure for word size n = 15, since this has been sufficient to successfully discriminate between network-based and CPU-based causes of data loss. Similar to the technique of loss pairs [14], we maintained a parallel data transfer (with the same send rate) that traversed the same network path except for the last hop. In this case, one stream branched into the server and the other stream branched into one of the computational nodes within the Titan cluster. This approach was taken to determine the impact of contention within the network path as a cause of data loss. Similarly, we used hardware counters to track the state of the network internal to the server. Finally, we used the available hardware counters to track the percentage of the CPU cycles allocated to the receiver at any given time. This tracking was done to establish whether a strong linear relationship existed between the number of CPU cycles received and the loss rate.

5 Experimental Results

The results of the first set of experiments are shown in Figures 1 and 2. Each figure shows the mean first-order complexity measure (calculated from the five 350,000 bit strings), and the 95% confidence intervals for the mean, for word sizes ***n = 4* to *n = 16.*** These figures compare the complexity measures obtained when data loss was caused by pure CPU contention or pure NIC contention. Each figure shows the complexity measures obtained at each of the four loss rates tested. The complexity of the signatures is clearly distinguishable even at very low loss rates and becomes more pronounced as the loss rate increases. It is interesting to note that the complexity measures associated with CPU contention show little change as the loss rate increases. However, the complexity values associated with NIC contention increase significantly with increasing loss rates. These results are quite encouraging in terms of differentiating between these two classes of error mechanisms using first-order complexity measures.

Figures 3 – 4 show the values of C^1 resulting from contention for router resources with those caused by contention for CPU resources. The disparity in values of C^1 depicted in Figure 3 is quite striking. The complexity measures associated with contention for resources at the router are cyclic and track quite well the gradual filling and draining of the router buffers. In the case of the router, the loss rate fluctuated between 0% and 2.5%. The loss rates associated with CPU contention fluctuated between approximately 1% and 3%, with little fluctuation in the values of C^1. This is another clear demonstration of how different causes of packet loss can produce significant differences in the underlying structure of the packet-loss signatures.

Figure 4 depicts the relationship between the loss rate and the corresponding values of C^1 as a function of the circumstances under which the data was lost. As can be seen, the values of C^1 are significantly higher (across all loss rates) when the data was lost because of contention in the network as opposed to contention at the receiving host. This result again demonstrates that the structure of packet-loss signatures is quite sensitive to the root cause of the data loss even when the loss rate is quite low.

The C^1 values obtained under real operating conditions is shown in Figure 5. The C^1 values obtained during this experiment are quite similar to the results obtained with a dedicated host in one important way: After an initial spike (at very low loss rates) the complexity measures were largely unchanged with increasing loss rates. In fact, the largest C^1 value obtained over the entire run was 0.47 at a loss rate of 22%. This result is particularly important given that the loss rate got as high as 80%! For comparison, the complexity measure associated with NIC contention was 0.69 when the loss rate was set at 15% (Figure 2). This result is consistent with other experiments we have conducted when data loss was due to contention for router resources (not presented in this paper because of space constraints). Based on these results, the classification mechanism would have attributed all loss to contention for CPU resources. This of course brings up the issue of how to validate this classification.

As noted in Section 4, we used several mechanisms to try to rule in or out the various factors that could have contributed to data loss. The parallel data stream experienced virtually no data loss during this experiment making it highly unlikely that contention within the network was a cause of packet loss. This is not surprising given that the experiments were conducted over very high-speed networks that are in general lightly loaded. Also, the network delay between ANL and NCSA is very small (on the order of 3 milliseconds based on results from the **traceroute** function). Similarly, the statistics collected from the **/proc** pseudo-file system showed relatively little network traffic interior to the host and registered no data loss due to errors within the system.

Finally, we wanted to determine whether there existed a linear relationship between the percentage of CPU cycles allocated to the data receiver and the observed loss rate. To this end, we calculated a simple least-squares regression line of CPU utilization on loss rate (shown in Figure 6). Visually, there appears to be a strong linear relationship between the CPU utilization and loss rate. This relationship is formalized by the value of the *sample coefficient of determination* (R^2), which measures the proportion of variability due to regression [11]. This value was 0.98, indicating that 98% of the variability in loss rate was attributable to the relationship between these two variables. The combination of information from the parallel data transfer, the **/proc** pseudo-file system, and the regression analysis strongly suggests that contention for CPU cycles was responsible for almost all of the observed loss and that the C^1 values accurately captured the cause of data loss in this experiment.

6 Conclusions and Future Research

In this paper, we have described a strategy for analyzing packet-loss signatures from a high-speed data transfer mechanism and we have showed how this strategy enables classification of the dominant cause of packet loss in theThe current transmission window. These techniques are based on first-order complexity measures (introduced for the first time in this paper) of packet-loss signatures to determine the root cause of packet loss. We outlined a series of simple experiments to test the efficacy of this technique, demonstrating it is easily capable of distinguishing packet loss caused purely by CPU contention or NIC contention. We have also been successful in detecting network contention using first-order complexity analysis. In actual Grid settings packet loss will likely be caused by a combination of factors, and the resulting signals from the complexity measures will be harder to discern. However, the results presented here provide strong evidence that contention for network resources at the communication endpoints can be detected even when such contention is quite low. This information can be used by the control mechanism to identify conditions under which it may be very damaging to increase the sending rate. What is not clear, and is the focus of current research, is whether there can be significant contention within the wide area network(s) connecting the communication endpoints that can (or likely to) produce complexity measures that appear to represent periodic behavior. This issue is strongly related to the queuing discipline used by the routers in the end-to-end path, and is being investigated through experimental and simulation studies.

The ability to classify the temporal dynamics of packet loss behavior (as expressed by the packet-loss signatures) offers two significant advantages. First, such classification allows the control mechanisms to apply corrective actions based on the particular cause of packet loss. For example, the control mechanisms may be able to migrate the data receiver, rather than drastically reducing the sending rate, when the root cause of packet loss is determined to be contention for CPU (rather than network) resources. Second, if the underlying dynamics has structure, it may be possible to construct simple predictors that allow the data transmitter to shape its behavior in such a way as to increase the probability that a sent packet is received successfully. These are enticing possibilities, and the exploration, evaluation, and integration of these techniques to the problem of large-scale data transfers represents a focus of current research activities.

Acknowledgments

We thank Dr. William D. Gropp (Argonne National Laboratory) for his kind guidance and support throughout this project.

References:

[1] Allcock, W., Bester, J., Breshahan, J., Chervenak, A., Foster, I., Kesselman, C., Meder, S., Nefedova, V., Quesnel, D., and Tuecke, S. Secure, Efficient Data Transport and Replica Management for High-Performance Data_Intensive Computing. In *Proceedings of IEEE Mass Storage Conference*, 2001.

[2] Balakrishnan, S., Padmanabhan, V., Seshan, S., and Katz, R. A Comparison of Mechanisms for Improving TCP Performance Over Wireless Links. *IEEE/ACM Transactions of Networking*, *5(6)*. 756-769. 1997

[3] Balakrishnan, S., Seshan, S., Amir, E., and Katz, R. Improving TCP/IP performance over wireless networks. In *Proceedings of ACM MOBICON*, November 1995.

[4] Barman, D., and Matta, I. Effectiveness of Loss Labeling in Improving TCP Performance in Wired/Wireless Networks. In *Proceedings of ICNP'2002: The 10th IEEE International Conference on Network Protocols*, Paris, France, November 2002.

[5] Biaz, S., and Vaidya, N. Performance of TCP Congestion Predictors as Loss Predictors, Texas A&M University, Department of Computer Science Technical Report 98-007, College Station, Texas.

[6] Biaz, S., and Vaidya, N. Discriminating Congestion Losses from Wireless Losses using Inter-Arrival Times at the Receiver. In *Proceedings of IEEE Symposium ASSET'99*, Richardson, TX, March, 1999.

[7] Crutchfield, J., and Feldman, D. Regularities Unseen, Randomness Observed: Levels of Entropy Convergence, Santa Fe INstitute Working Paper 01-02-012, 2001.

[8] D'Alessandro, G., and Politi, A. Hierarchical Approach to Complexity with Applications to Dynamical Systems. *Physical Review Letters*, *64* (14). 1609-1612. April, 1990

[9] Dickens, P. FOBS: A Lightweight Communication Protocol for Grid Computing. In *Proceedings of Europar 2003*, 2003.

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

[11] Dougherty, E. *Probability and Statistics for the Engineering, Computing, and Physical Sciences*. Prentice Hall, 1990.

[12] Elsner, J., and Tsonis, A. Complexity and Predictability of Hourly Precipitation. *Journal of the Atmospheric Sciences*, *50* ((3)). 400-405. 1993

[13] Hao, B.-l. *Elermentary Symbolic Dynamics and Chaos in Dissipative Systems*. World Scientific, 1989.

[14] Liu, J., Matta, I., and Crovella, M. End-to-End Inference of Loss Nature in a Hybrid Wired/Wireless Environment. In *Proceedings of Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'03)*, Sophia-Antipolis, France, 2003.

[15] Salamatian, K., and Vaton, S. Hidden Markov Modeling for Network Communication Channels. In *Proceedings of ACM SIGMETRICS 2001 / Performance 2001*, Cambridge, Ma, June 2001.

[16] Shannon, C. A mathematical theory of communication. *Bell System Technical Journal*, *27*. 379-423.

[17] Sivakumar, H., Bailey, S., and Grossman, R. PSockets: The Case for Application-level Network Striping for Data Intensive Applications using High Speed Wide Area Networks. In *Proceedings of Super Computing 2000 (SC2000)*.

[18] Sivakumar, H., Mazzucco, M., Zhang, Q., and Grossman, R. Simple Available Bandwidth Utilization Library for High Speed Wide Area Networks. *Submitted to Journal of SuperComputing*.

[19] Tsaoussids, V., and Matta, I. Open Issues on TCP for Mobile Computing. *Journal of Wirelesss Communications and Mobile Computing- Special Issue on Reliable Transport Protocols for Mobile Computing*, *2(1)*. February, 2002

[20] Vaidya, N., and Biaz, S. Discriminating Congestion Losses from Wireless Losses Using Inter-Arrival Times at the Receiver. In *Proceedings of IEEE Symposium ASSET'99*, March, 1999.

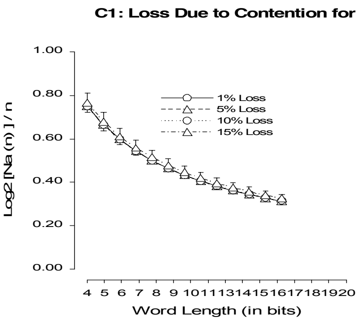

Figure 1. This figure shows the values of C^1 when the root cause of data loss was contention for CPU resources. The loss rate varied between 1% and 15%, and the complexity measures for word sizes n=4 to n=16 are shown.

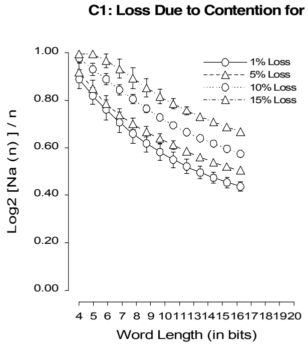

Figure 2. This figure shows the values of C^1 when the root cause of data loss was contention for NIC resources. The loss rate varied between 1% and 15%, and the complexity measures for word sizes n=4 to nn=16 are shown.

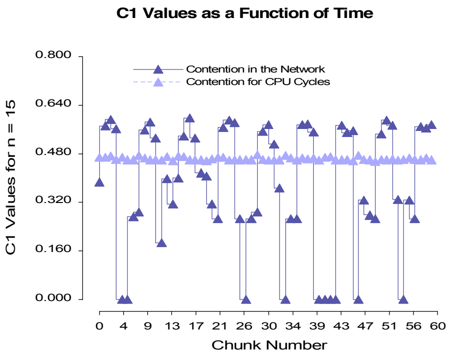

Figure 3. This figure tracks a time series of C^1 values when data loss was caused by contention for router resources versus contention for CPU resources.

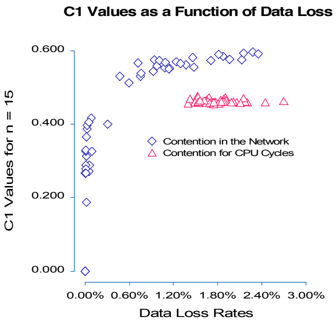

Figure 4. This figure depicts the C^1 values from Figure 3 as a function of the loss rate.

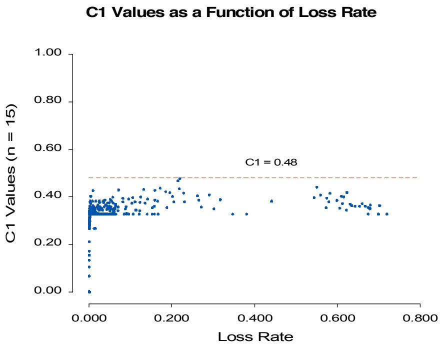

Figure 5. This figure shows the value of C^1 as a function of loss rate in a real-world setting. The red line indicates the largest value of C^1 obtained over the transfer (0.47).

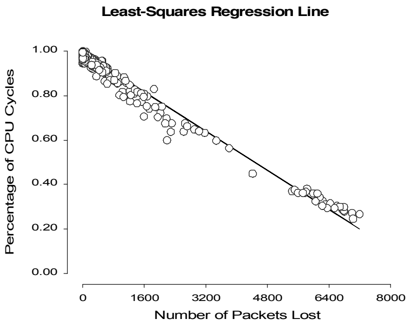

Figure 6. This figure shows the linear relationship between the percentage of CPU cycles allocated to the receiver and the loss rate.