Maintaining High Transmitted Image Quality in Congested ATM Networks [*]

Chun Wong and Albert M. K. Cheng

Real-Time Systems Laboratory
Department of Computer Science
University of Houston, University Park
Houston, TX 77204-3475
E-mail: comphys, cheng@cs.uh.edu
Phone: (713) 743-3353
Fax: (713) 743-3335

Abstract

This paper presents an approach to pre-process/post-process TIFF image files in real-time applications for maintaining an acceptable image quality when these files are transmitted in a congested ATM network. Compared to existing techniques, our approach introduces a very low transmission and computational overhead, allows a higher cell-loss rate, and produces less image degradation. In this approach, the pre-processor rearranges the content of image files, and then packs these fixed image files into ATM cells for transmission. The post-processor unpacks the cells, restores the original content orders, and if part of the data has been discarded, performs data recovery to reconstruct imprecise image files as close to their original versions as possible. By marking an appropriate number of cells, a user can specify an acceptable lower bound on the quality of the image that is delivered at the destination. This makes it possible to control and know which part of the image data can be discarded. In case congestion occurs at any intermediate node along the preselected route, cells with a marked cell-loss priority bit can be dropped by the congestion control scheme to ease the congestion. This guarantees the timely delivery of the image files while satisfying the image-quality lower bound at the destination. Our experimental results show that by using the proposed approach, the quality of the received/recovered images decreases more slowly than the proportional increase in network congestion.

---

[*] This material is based upon work supported in part by the National Science Foundation under Award Nos. CCR-9111563 and IRI-9526004, by the Texas Advanced Research Program under Grant No. 3652270, and by a grant from the University of Houston Institute of Space Systems Operations.

Motivation

Internet traffic is increasing at an extremely fast rate as the population using this information superhighway grows exponentially. Extensive research has focused on real-time networks and protocols which can support the transfer of multimedia data, such as a combination of voice, image, and video, in order to satisfy the application requirements. In image/video applications, unpleasant viewing results such as movie screen freeze, unsynchronized speech and vocal motions, or incomplete image display may be caused by data that have missed their transmission deadlines (timing faults). As a result, it is important for real-time multimedia messages to be transmitted from the source node to the destination node within its deadline, as specified by the user or application. The high transmission speed of *Asynchronous Transfer Mode* (ATM) networks makes them the best future *broadband integrated service digital networks* (BISDNs) which integrate a variety of different services, including multimedia data transmission. However, the network bandwidth remains to be the potential bottleneck while communicating workstations are continually upgraded with lower-cost and higher-speed processors. The focus of this paper is on optimizing the transmission of image files in real-time ATM network channels. The goal of our research is to develop an image file transmission approach which, when compared to existing techniques, (1) introduces much lower transmission and computational overhead, (2) allows a higher cell-loss rate, and (3) produces less image degradation. Note that the cells that miss their deadlines are considered lost.

Current applications that involve image transfer use a variety of image processing and transmission techniques to reduce the amount of image data that have to be sent. Their assumption is based on the fact that there is little or no data loss. Here, we present a technique to ease network congestion by discarding a portion of the image data, thus degrading the quality of the image while meeting its delivery time constraints. Simply compressing an image to the smallest possible size would not guarantee that the image will be transmitted in a congestion-free network. Also, when the network is congested, it is very risky to allow the network congestion controller to decide which cells to drop if all of them have equal priority, since it has no idea of how vital is the information in each cell. So, it is the responsibility of the higher-level application to know and help the congestion controller distinguish the important data from the less significant data that can be discarded.

The concept of an ATM real-time channel [5] has been developed to guarantee the timely delivery of real-time messages by partitioning and scheduling network resources. A real-time channel is defined as a three-tuple (T, M, D), where T represents the minimum message inter-generation time, M represents the maximum message size, and D represents the maximum message end-to-end delay. In other words, D is also the worst-case deadline for the transmission of the message. A real-time channel is thus a unidirectional connection between a source and a destination which ensures that every message transmission guaranteed at the source will be sequentially delivered to the destination and in a time period no greater than D if the following two conditions are satisfied:

1. The message inter-generation time is at least T.
2. The message size does not exceed M.

However, these assumptions do not always hold in practice for two reasons [5]:

1. A source may generate messages faster and larger than the initially specified values, T and M, due to the use of estimated (instead of actual) channel parameters. Since no bandwidth-enforcement mechanisms are used at the source node, all of these messages are sent over the channel. This would affect the delay bounds guaranteed to all other channels over the link. For example, the size of movie data may increase sharply over a relatively short period of time by the occurrence of fast-moving motion which cannot be foreseen.
2. Even if a source node does not violate its traffic specification, the message inter-arrival time at an intermediate link can still be smaller than T (clumping effect) due to uneven queueing delays at upstream links. For instance, limited buffer space may not be adequate to accommodate early arriving cells.

Therefore, network congestion is unavoidable and can directly affect the timeliness of a real-time image transmission. Congestion control is thus essential for real-time message transmission. In this paper, we adopt the idea of priority control by *selective cell discarding* (SCD) [1, 4] as the basic congestion control scheme, with the concept of imprecise computations as our model [2].

Imprecise computation techniques provide a solution to real-time applications by trading off the result quality with time to meet computation deadlines. The basic strategy to minimize the negative effects of timing faults is to leave the less important tasks unfinished if necessary. This technique divides a task into mandatory and optional subtasks. The mandatory subtask is required for an acceptable result and must be executed completely before the task deadline. The optional subtask refines the result. It can be left unfinished and terminated at its deadline. If the intermediate results produced by a task are recorded at appropriate instances during its execution, the user can examine the error bound and decide whether an imprecise result is acceptable. This method for returning imprecise results is called the *milestone* method. Otherwise, if the deadline is approaching and an imprecise result is produced as soon as possible by skipping all the remaining computation steps, the method is called the *sieve* approach.

Since real-time image applications can distinguish different kinds of image file formats, an image file can be arranged into mandatory and optional parts. All cells containing exclusively data belonging to the optional part of the image file can be marked discardable. In case congestion occurs at an intermediate node, discardable cells can be discarded from the input links when the buffer is full. Also, discardable cells can be removed from the highly occupied buffer in order to make space for the cells containing data in the mandatory part of the successor tasks. This kind of congestion control scheme has been proved effective [3]. Here, we introduce a method for marking cells.

References

[1] Mazlan Abbas and Zainal Abidin Ahmad. “Performance Evaluation of Selective Cell

Discarding Control in ATM Networks," *Communications on the Move. ICCS/ISITA '92*, Singapore, pp. 142-146, vol. 1, November 1992.

[2] Jane W. S. Liu, Kwei-Jay Lin, Wei-Kuan Shih, Albert Chuang-shi Yu, Jen-Yao Chung, and Wei Zhao. "Algorithms for Scheduling Imprecise Computations," *Computer*, pp. 58-68, vol. 24, iss. 5, May 1991.

[3] V. Millan-Lopez, W. Feng and J. W.-S. Liu. "Using the Imprecise-Computation Technique for Congestion Control on a Real-Time Traffic Switching Element," *Proceedings of 1994 International Conference on Parallel and Distributed Systems*, pp. 202-208, December 1994.

[4] David W. Petr and James S. Wineinger. "End-to-End Priority Cell Discarding Analysis for ATM Networks," *IEEE INFOCOM '93. The Conference on Computer Communications Proceedings*, San Francisco, pp. 1178-1185, vol. 3, April 1993.

[5] Qin Zheng, Kang G. Shin, and Chia Shen. "Real-Time Communication in ATM Networks," *Proceedings of 19th Conference on Local Computer Networks*, pp. 156-164, October 1994.