TRAFFIC MODELING IN THE ATM-TN TELESIM PROJECT: DESIGN, IMPLEMENTATION, AND PERFORMANCE EVALUATION

Martin F. Arlitt   Ying Chen
Remi J. Gurski   Carey L. Williamson
Department of Computer Science
University of Saskatchewan
Saskatoon, SK, Canada S7N 5A9
e-mail: carey@cs.usask.ca

Keywords: Network, Simulators, Model Design, Telecommunications, Performance Analysis

ABSTRACT

The purpose of this paper is to provide an overview of the traffic models that have been designed and built for the ATM-TN TeleSim Project. A brief description of each of the three traffic models (an MPEG compressed video model, an aggregate Ethernet LAN traffic model, and an Internet mosaic session model) is presented, including the purpose, the design methodology, and the parameters for each model.

All three models have been implemented in SimKit, a C++-based simulation interface built on top of Shared Memory TimeWarp (SMTW), which provides an environment for efficient parallel simulation. Preliminary performance results for the sequential implementations of the three traffic models are now available, and are reported in this paper.

1 INTRODUCTION

Workload modeling is a vital component of any simulation study (Jain 1991). Outputs from a simulation model are highly dependent on the inputs provided to the model, and without realistic input workload models, the simulation results are of little value.

The goal in the traffic modeling effort for the TeleSim project is to design a small set of accurate and representative workload models to use as inputs for the overall ATM-TN simulation engine. *Synthetic* workload models have been chosen to provide concise mathematical representations of the workload characteristics of existing network applications, eliminating the need for lengthy packet-level traces as input to the simulation (which are undesirable for reasons of flexibility, storage, and simulation execution time).

Three specific input traffic types have been chosen as representative examples of applications on high speed networks: MPEG (Motion Pictures Experts Group) compressed video, aggregate Ethernet local area network (LAN) traffic, and Internet mosaic traffic. Video traffic is one of the most bandwidth-intensive applications on high speed networks, and variable-bit-rate video (i.e., compressed video) is one of the most important classes of video to consider in the context of ATM (Asynchronous Transfer Mode) networks. The MPEG video model is designed to model this compressed video traffic. The Ethernet LAN traffic model is designed to model the aggregate data packet traffic generated across an ATM backbone network connecting campus-area LANs. The Internet mosaic model is designed to capture the workload characteristics of `mosaic` (Andreessen 1993), an application that is the fastest-growing consumer of bandwidth on the NSFNET backbone (NSFNET 1995).

Each of the three workload models has its own specific workload characteristics and parameters. To the extent possible, the construction and parameterization of the workload models has been based on empirical network traffic measurement and workload characterization in our own local area network environment. The models also draw upon measurement and modeling results in the published literature (Frost and Melamed 1994; Garrett and Willinger 1994; Leland *et al.* 1994; Paxson 1994; Paxson and Floyd 1994).

The remainder of this paper is organized as follows. Section 2 provides some background on the ATM-TN TeleSim project, setting the context for our simulation work. Section 3 describes the design and implementation of our traffic models. Section 4 presents preliminary performance results for the sequential implementations of the three traffic models. Finally, Section 5 concludes the paper.

2 THE TELESIM PROJECT

TeleSim is a collaborative research project involving researchers from four universities: the University of Alberta, the University of Calgary, the University of Saskatchewan, and Waikato University in New Zealand. The project, which began in Q1 1994, is headquartered at the University of Calgary, where it is led by Brian Unger in the Department of Computer Science.

The goal of the TeleSim project is to develop high performance parallel simulation tools for the design and analysis of telecommunications networks. Funded by the Canadian federal government's CANARIE program, and several industry sponsors, TeleSim is one of six projects under the umbrella of WURCNET (Western University Research Consortium on High Performance Computing and Networking).

Of particular interest to the TeleSim project (and its sponsors) are high speed ATM networks. The TeleSim project brings together researchers from the parallel simulation research community (Calgary, Waikato) and researchers from the computer networking research community (Alberta, Saskatchewan). Both groups share a common interest in building efficient, accurate, high performance simulation tools for ATM networks.

The overall network simulator being constructed by the TeleSim project is called ATM-TN, which stands for Asynchronous Transfer Mode Traffic and Network model. The first release of the ATM-TN software was made available (internally) to the TeleSim project participants in January 1995. This release supports only sequential execution. Performance experiments and simulation experience using this release will be used to guide the development of a parallel version of ATM-TN, expected by summer 1995. Commercial development and release of ATM-TN may follow within the next two years.

Responsibilities in the one-year TeleSim project have been divided amongst the project participants according to the expertise available at each site. The University of Saskatchewan has constructed traffic models for the TeleSim project. The University of Alberta has constructed models of switches, networks, and signalling protocols. The University of Calgary and Waikato University have defined and implemented the simulation interface for ATM-TN, as well as a simulation executive to support efficient parallel simulation.

The simulation interface for ATM-TN is called SimKit. SimKit is an object-oriented C^{++}-based simulation language to support discrete-event simulation. SimKit defines a class (`sk_lp`) to represent logical processes (LP's), a class (`sk_event`) to represent the messages exchanged between LP's, and a framework to transparently support both sequential and parallel execution. Parallel execution is supported by a lower layer called WarpKit, which interfaces directly to Shared Memory TimeWarp (SMTW) (Fujimoto 1989). These lower layers deal with the assignment of logical processes to processors, event scheduling, state saving, and other technical issues related to parallel execution. The target hardware platform for ATM-TN is a 14-processor SGI Power Challenger at the University of Calgary. However, sequential execution is supported on standard UNIX-based workstations, such as Sun SPARCstations and SGI workstations.

This paper describes only the traffic modeling work that has been done by the University of Saskatchewan as part of the TeleSim project. These efforts all take place at or above the SimKit layer in ATM-TN.

3 TRAFFIC MODELS: DESIGN AND IMPLEMENTATION

Three synthetic workload models have been constructed for use within ATM-TN as representative examples of applications on high speed networks: MPEG compressed video, aggregate Ethernet LAN traffic, and Internet mosaic traffic. This section overviews the design and implementation of each of these three models.

3.1 MPEG Video Traffic Model

The MPEG video traffic model in ATM-TN is designed to model the unidirectional transmission of a single variable bit rate compressed video stream in an ATM network, where the video source is compressed according to the MPEG standard (LeGall 1991). MPEG defines a standardized method for compressing full-motion video for storage as digital data. For viewing, this data is transmitted over a network as an MPEG stream and decoded at its destination.

The MPEG standard is intended primarily for use with stored video applications (e.g., entertainment video), but it can also support some limited real-time (i.e., "live") video applications, such as desktop video-conferencing, depending on the compression hardware available. Because the encoding of an MPEG video sequence usually takes much longer than decoding, a video sequence is usually encoded only once and stored in compressed form. Playback can then be done as many times as desired. The decoding of MPEG video sequences can be done in real time.

The MPEG video stream consists of a sequence of images called *frames* that are displayed one after the other, at short periodic intervals (e.g., 30 frames per second). The MPEG standard defines three types of compressed frames:

I I (Intraframe) frames are encoded using only the image data available in the current frame. Since I frames represent a complete image, they provide an absolute reference point for the other two image types in the MPEG sequence, which are encoded using interframe coding (i.e., they express the relative differences from one frame to the next).

I frames are compressed using Discrete Cosine Transform (DCT), similar to the JPEG (Joint Photographic Experts Group) standard for single image compression. I frames take the least time to compress, but have the least compression.

P In addition to DCT compressed data, P (Predictive) frames contain motion-compensated data predicted from the preceding (I or P) frame. P frames take longer to encode than I frames, are faster to decode than I frames, and achieve higher compression than I frames.

B B (Bidirectional interpolative) frames are similar to P frames, but they contain motion-compensated data from both the previous and the next frame (I or P). B frames take the longest time to encode, but offer the highest levels of compression.

An MPEG sequence then consists of a pattern of I, P, and B frames, called a Group Of Pictures (GOP). The GOP pattern must be specified at the time of encoding, and must start with an I frame. The same encoding pattern is then used repeatedly for the duration of the video sequence, with each new I frame starting a new GOP. An example of a typical MPEG sequence is IBBPBBPBBIBBPBBPBBI…, where the GOP pattern is IBBPBBPBB. Significant correlations (i.e., autocorrelation and cross-correlation) have been observed in the sizes of the MPEG frames in a video sequence (Pancha and El Zarki 1994).

In the TeleSim project, the MPEG video sources are modeled using the Transform-Expand-Sample (TES) modeling methodology established by Melamed *et al.* (Melamed 1992; Geist and Melamed 1992). TES provides a fast method to generate autocorrelated time series. In this particular case, TES is used to generate autocorrelated time series data that closely resembles the output of an MPEG stream.

Three separate TES processes are used — one for each type of MPEG frame. The marginal distribution (i.e., frequency histogram) and autocorrelation function of frame sizes (done separately for I, P, and B frames) are the key inputs to the TES modeling process. These inputs have been used to derive a separate TES model for each frame type. Each of these models is saved in a TES model specification file that is used by the MPEG source model. The three frame types are then interleaved as determined by the GOP pattern. Dependence between frame types (as has been empirically observed in MPEG sequences) is created by duplicating the TES background variate of the I frame at the beginning of each GOP. The resulting cross-correlation between frame sizes increases the accuracy of the model (Melamed 1992; Garrett and Willinger 1994). Thus the final MPEG model is a composite TES model. The model has been parameterized based on empirical studies of MPEG sequences (i.e., using MPEG data provided with `testool` (Geist and Melamed 1992)).

The final MPEG model has four parameters: (1) the GOP structure of the MPEG stream; (2) a scaling factor for the average bit rate; (3) the frame rate for the displayed video; and (4) a smoothing parameter for shaping the video source.

The MPEG model has been implemented in SimKit (1474 lines) using two logical processes (LPs): one for the source, one for the sink. A TES model is used to generate the frame sizes within the MPEG model. An entire GOP is generated at a time, and then buffered for simulated transmission according to the other traffic parameters.

3.2 Ethernet LAN Traffic Model

The Ethernet LAN traffic model is designed to represent the self-similar workload characteristics (Leland *et al.* 1994) of aggregate data packet traffic on existing local area networks, such as campus LANs. Self-similarity has been observed in video traffic (Garrett and Willinger 1994) and wide area Internet traffic (Paxson and Floyd 1994) as well.

Self-similar Ethernet traffic has been modeled using the TES (Melamed 1992; Geist and Melamed 1992) modeling methodology. The resulting model has two parameters: (1) the Hurst parameter $0.5 \leq H < 1$ expresses the degree of self-similarity in the traffic; and (2) the network utilization $0 \leq U \leq 1$ determines the average load level offered by the Ethernet LAN, as a fraction of its total 10 Mbps bandwidth. The Ethernet model has been implemented in SimKit (1905 lines) using two LPs: one for the source, one for the sink.

A more complete description of the Ethernet traffic model is contained in a separate paper elsewhere in this conference proceedings (Chen *et al.* 1995).

3.3 Internet Mosaic Traffic Model

The Internet mosaic traffic model is designed to represent the wide area network traffic generated by users of mosaic (Andreessen 1993).

Table 1: Performance Results for Standalone Sequential Implementations

Traffic Model	Events Generated	ATM Cells Generated	Elapsed Seconds	Events per Sec	μsec per Event	Events per Cell	μsec per Cell
MPEG	3703	2775113	0.080	46082	21.70	0.001	0.03
Ethernet	2413616	2313635	51.892	46512	21.50	1.043	22.43
Mosaic	53949	23169	0.874	61740	16.20	2.328	37.71

Table 2: Performance Results for ATM-TN Sequential Implementations

Traffic Model	Events Generated	ATM Cells Generated	Elapsed Seconds	Events per Sec	μsec per Event	Events per Cell	μsec per Cell
MPEG	184551	184300	5.651	32657	30.62	1.001	30.66
Ethernet	484392	474389	21.781	22239	44.97	1.021	45.91
Mosaic	154532	16998	41.514	3722	268.64	9.091	2442

The mosaic session model has six parameters: (1) number of conversations per session; (2) conversation arrival distribution; (3) conversation destination; (4) number of connections per conversation; (5) connection arrival distribution; and (6) bytes exchanged per connection. The mosaic model has been implemented in SimKit (1678 lines) using 107 LPs: 1 for the source, 100 for destinations, and 6 to model the segmentation and reassembly between TCP/IP packets and ATM cells.

A more complete description of the mosaic traffic model is contained in a separate paper elsewhere in this conference proceedings (Arlitt and Williamson 1995).

4 PERFORMANCE RESULTS

Tables 1 and 2 summarize the preliminary performance results for the (unoptimized) sequential implementations of the three traffic models. Two different implementations exist for each traffic model: one that runs in standalone mode outside of ATM-TN, and one that runs within the entire ATM-TN framework. Table 1 shows the performance results for the standalone implementation. Table 2 shows the performance results for the ATM-TN implementation. The two implementations differ in the number of events and the amount of overhead per ATM cell, since the standalone implementations use only traffic sources and sinks, while the ATM-TN implementations include switches, ports, links, end nodes, and signalling messages. Both implementations are written in SimKit. All timings were taken on a Sun SPARC-10.

Table 1 shows that the per event and the per cell overhead is fairly low for each of these models in the standalone implementation. All three models provide on the order of 50,000 events per second, where an event is either a frame arrival (for the MPEG model) or an ATM cell arrival (for the Ethernet and mosaic models). The per cell overhead is greatest for the mosaic model because of the detailed modeling of the ATM Adaptation Layer (AAL), which does segmentation and reassembly of ATM cells using AAL-5. The per cell overhead is lowest for the MPEG model because the standalone implementation generates events at the frame level, and does not do detailed modeling of AAL segmentation and reassembly. A simple calculation for the Ethernet traffic model shows that sequential simulation run times on the order of 20 seconds will be needed to simulate 10^6 ATM cells, for the standalone implementation.

Table 2 shows that the per cell overhead increases in the ATM-TN implementation, since a single cell transmission by the traffic source generates events at all switch, port, and link components on the path to the traffic sink. These performance measurements were taken using a single traffic source in loopback mode through a 4-port shared memory switch model. For this network configuration, Table 2 shows that the Ethernet and MPEG models provide on the order of 20,000 to 30,000 events per second, respectively, while the mosaic model provides about 3,700 events per second. Again, the overhead is greatest for the mosaic model because of the detailed modeling of the segmentation and reassembly of ATM cells using AAL-5. A simple calculation for the Ethernet model shows that simulation run times on the order of 45 seconds will be needed to simulate 10^6 ATM cells on a very simple network topology.

The performance results reported above are all for sequential implementations of the traffic models in SimKit. Parallel implementations of the ATM-TN models are now underway. Performance results for these implementations should be available by mid-to-late summer 1995.

5 CONCLUSIONS

This document has outlined the design of the network traffic workload models for ATM-TN. Three specific traffic models have been chosen for implementation: an MPEG compressed video traffic model, an Ethernet LAN traffic model, and an Internet mosaic traffic model.

Each model is a non-trivial synthetic workload model that is intended to accurately capture the salient characteristics of network traffic workloads. Each model has its own parameters and application-specific workload characteristics. Each model has been designed and implemented using the latest methods and results in traffic modeling and workload characterization. Each model has been parameterized based on empirical measurement data, either from our own network measurements, or from the published literature. Models have been translated into SimKit, validated, and incorporated into the overall ATM-TN simulation model.

Preliminary performance results for the sequential implementations of the three traffic models are now available. Performance results for the parallel implementations should be available by late summer 1995.

REFERENCES

Andreessen, M. 1993. "NCSA Mosaic Technical Summary". National Center for Supercomputing Applications.

Arlitt, M. and C. Williamson. 1995. "A Synthetic Workload Model for Internet Mosaic Traffic". In *Proceedings of the 1995 Summer Computer Simulation Conference* (Ottawa, Canada, July 24-26).

Chen, Y.; Z. Deng; and C. Williamson. 1995. "A Model for Self-Similar Ethernet LAN Traffic: Design, Implementation, and Performance Implications". In *Proceedings of the 1995 Summer Computer Simulation Conference* (Ottawa, Canada, July 24-26).

Frost, V. and B. Melamed. 1994. "Traffic Modeling for Telecommunications Networks". *IEEE Communications*, Vol. 32, No. 3 (Mar.), pp. 70-81.

Fujimoto, R. 1989. "Time Warp on a Shared Memory Multiprocessor". *Transactions of the Society for Computer Simulation*, Vol. 6, No. 3 (July), pp. 211-239.

Garrett, M. and W. Willinger. 1994. "Analysis, Modeling and Generation of Self-Similar VBR Video Traffic". In *Proceedings of the 1994 ACM SIGCOMM Conference* (London, UK, Aug. 31-Sept. 2), pp. 269-280.

Geist, D. and B. Melamed. 1992. "TEStool: An Environment for Visual Interactive Modeling of Autocorrelated Traffic". In *Proceedings of the 1992 International Conference on Communications* (Chicago, Illinois, June), pp. 1285-1289.

Jain, R. 1991. *The Art of Computer Systems Performance Analysis.* Wiley & Sons.

LeGall, D. 1991. "MPEG: A Video Compression Standard for Multimedia Applications". *Communications of the ACM*, Vol. 34, No. 4 (Apr.), pp. 46-58.

Leland, W.; M. Taqqu; W. Willinger; and D. Wilson. 1994. "On the Self-Similar Nature of Ethernet Traffic (Extended Version)". *IEEE/ACM Transactions on Networking*, Vol. 2, No. 1 (Feb.), pp. 1-15.

Melamed, B. 1992. "TES Modeling of Video Traffic". *IEICE Transactions on Communications*, Vol. E75-B (June), pp. 1292-1300.

NSFNET. 1995. Data available by anonymous ftp from nis.nsf.net:/nsfnet/statistics.

Pancha, P. and M. El Zarki. 1994. "MPEG Coding for Variable Bit Rate Video Transmission". *IEEE Communications Magazine*, Vol. 32, No. 5 (May), pp. 54-66.

V. Paxson. 1994. "Empirically Derived Analytic Models of Wide-Area TCP Connections". *IEEE/ACM Transactions on Networking*, Vol. 2, No. 4 (Aug.), pp. 316-336.

Paxson, V. and S. Floyd. 1994. "Wide-Area Traffic: The Failure of Poisson Modeling". In *Proceedings of the 1994 ACM SIGCOMM Conference* (London, UK, Aug. 31-Sept. 2), pp. 257-268.

Martin Arlitt is an M.Sc. student in the Department of Computer Science at the University of Saskatchewan. His interests include Internet traffic characterization, and file caching for Internet Web servers.

Ying Chen is an M.Sc. student in the Department of Computer Science at the University of Saskatchewan. Her interests include workload modeling, self-similarity, and mathematical modeling of traffic flows in high speed networks.

Remi Gurski is an undergraduate student at the University of Saskatchewan, pursuing a combined degree in Electrical Engineering and Computer Science. His interests include video compression, video traffic modeling, network simulation, and large software systems.

Carey Williamson is an Associate Professor in the Department of Computer Science at the University of Saskatchewan. His research focuses on performance issues in high speed ATM networks: workload modeling, admission control, congestion control, and quality of service guarantees. He is currently supervising six graduate students working in these areas.