Measuring and simulating an 802.3 CSMA/CD LAN

TECHNISCHE UNIVERSITEIT EINDHOVEN
Faculteit Wiskunde en Informatica

Memorandum COSOR 91-42

Measuring and simulating an 802.3 CSMA-CD LAN

A. Dekkers
J. van Oorschot

Eindhoven University of Technology
Department of Mathematics and Computing Science
P.O. Box 513
5600 MB Eindhoven
The Netherlands

Eindhoven, December 1991
The Netherlands
ISSN 0926-4493

Measuring and Simulating an 802.3 CSMA/CD LAN

Jan van Oorschot*
Anton Dekkers**

*Delft University of Technology
P.O. Box 5031
2600 GA Delft
The Netherlands
email: J.P.M.vOorschot@et.tudelft.nl

**Eindhoven University of Technology
P.O. Box 513
5600 MB Eindhoven
The Netherlands

Abstract

A single 802.3 Carrier Sense Multiple Access/Collision Detection (CSMA/CD) segment is measured resulting in throughput, response times and workstation parameters for several network nodes. During the measurements, the network carried an artificial workload with the characteristics of a real-life workload. A simulation of the laboratory test is developed using the artificial workload parameters and the 802.3 CSMA/CD standard of the Consultative Committee for International Telephony and Telegraphy (CCITT).

The measurements show that it is possible to determine the workstation parameters with a great accuracy using simple throughput measurements on an otherwise empty network. It is then possible to isolate exact ethernet parameters during throughput measurements on a network with a known workload. The behaviour measured is reproduced in a simple simulation. The results of the simulation conform to the measured values.

Some conclusions are that a 802.3 CSMA/CD segment can be measured and simulated with accurate results. The simulation environment is used to model a real-life ethernet network in circumstances that can not be measured in a real-life situation. Parameters that can be used to fine-tune the simulation are the interframe gap time, and the workstation distance on a network.

Introduction

In the performance analyses of Local Area Networks (LANs), it is necessary to make a simulation model of that LAN. Such a simulation model can be used to determine performance characteristics of the network and to perform tests that can not be performed in the actual network. To build an accurate simulation model of the LAN under investigation the network component parameters, network operational characteristics and network configuration have to be determined. In this research, an ethernet type of LAN is considered, consisting of several network nodes and one ethernet segment. In such a LAN the network configuration and operation are well-known. The network component parameters will have to be measured before an accurate simulation model can be constructed. The most important parameters involved in a simple model of the network nodes are the time needed to copy the frame tot the ethernet card, the time needed by the ethernet card to send the frame and the time the card has to wait before actually sending the frame.
After these parameters are determined, the simulation model can be run with these parameters and a workload description as input. To validate the simulation model, the results of the simulation run should be compared with measurement on the actual network.

The validated simulation model will be a valuable tool in the performance evaluation of ethernet based LANs. The authors have used this model to validate a simple analytic model of ethernet based LANs [13].

In 'The 802.3 CSMA/CD LAN', the main characteristics of the LAN used are discussed. The part 'Measurements' described the determination of the network component parameters. The part 'Simulation' describes the validation of the simulation model of the ethernet LAN used.

The 802.3 CSMA/CD LAN (ethernet)

The ethernet used in the measurements is a local area network conforming to the IEEE 802.3 CSMA/CD standard [1,5]. This type of network is widely used as transport medium for connecting computer equipment.

The computer equipment is connected to the ethernet as depicted in Figure 1.

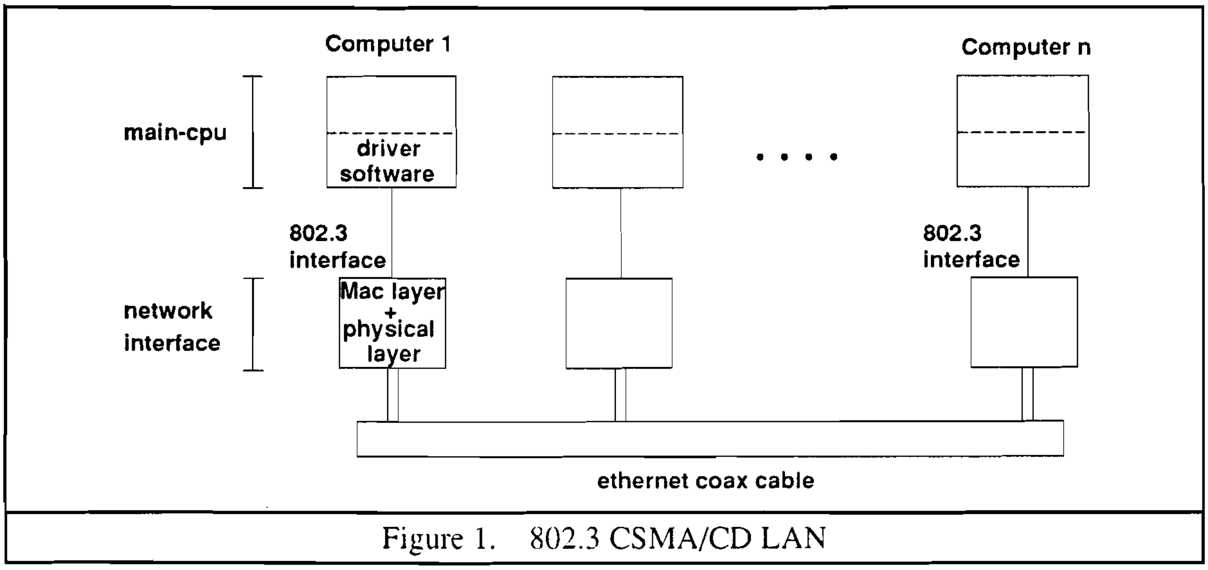

Figure 1. 802.3 CSMA/CD LAN

The exact operation is described in the IEEE standard [1]. In the following paragraphs a quick description of the operation is given, sufficient to explain the principles involved and enables the reader to reproduce the measurements and simulations.

Each computer formats the data to be sent in frames, see Figure 2. This frame is sent to the 802.3 CSMA/CD interface. The interface will wait until the medium is free and then it will try to send the frame over the ethernet broadcast cable. If a collision occurs with another sending workstation, the interfaces of both computers back off for a random time before retrying. After sending a frame, the interface will defer from sending for a fixed period of time to avoid dominating the medium. The exact parameters used for the protocol are given in [1]. Main parameters are the random back-off method, and the minimum interframe gap time. These are well defined for ethernet. The capacity of the ethernet is 10^7 bits per second (10 Mbit/s nominal).

During periods of heavy load on the network, frames can suffer of multiple collisions and long waiting times, and even be lost.
Notice that the total time needed to send a frame is not only dependent on the characteristics of the 802.3 CSMA/CD network, but also on the speed of the computer generating and copying the frame.

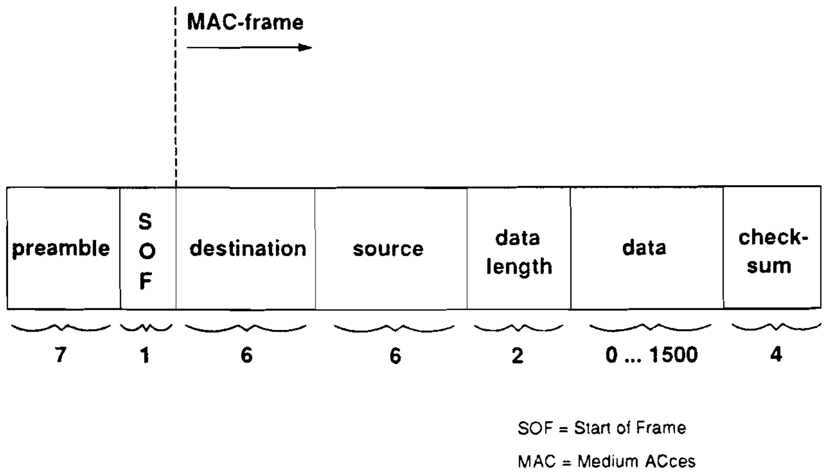

Figure 2. 802.3 CSMA/CD frame layout

Measurements

The goal of the measurements was to determine the behaviour of the ethernet as a function of the network workload.
The behaviour is measured by letting a workstation generate a fixed number of fixed length frames. The total time to send these frames is taken as a performance indicator.
This performance indicator is measured for a range of frame-lengths and different workstations.

In the measuring process there are three time consuming activities which are repeated for each frame sent. Two of these activities, the creation and the copying of the frame, normally take place at the Central Processing Unit (CPU) of the computer involved, the third activity, the sending of the frame, in the network adapter.

The frame is created by the computer CPU. This frame is then copied to the network adapter by the computer CPU as soon as the adapter is available. This will result in some waiting time if the network adapter is occupied.

Once the frame is completely copied to the network adapter, and the network is free, the frame will be sent by the adapter. The adapter then waits for the next frame to be sent.

The computer CPU starts to create the next frame the moment the previous one is copied to the network adapter.

The computer CPU and network adapter can create and send frames independently of the availability of the other, but for the copying of a frame both have to be available. It should also be noted that this model assumes the ethernet card to have just one frame buffer.

This process is depicted in Figure 3.

The main performance parameter in this process is:

T(n) = "*time needed to send one of N frames of length <n> (N = 10.000)*"

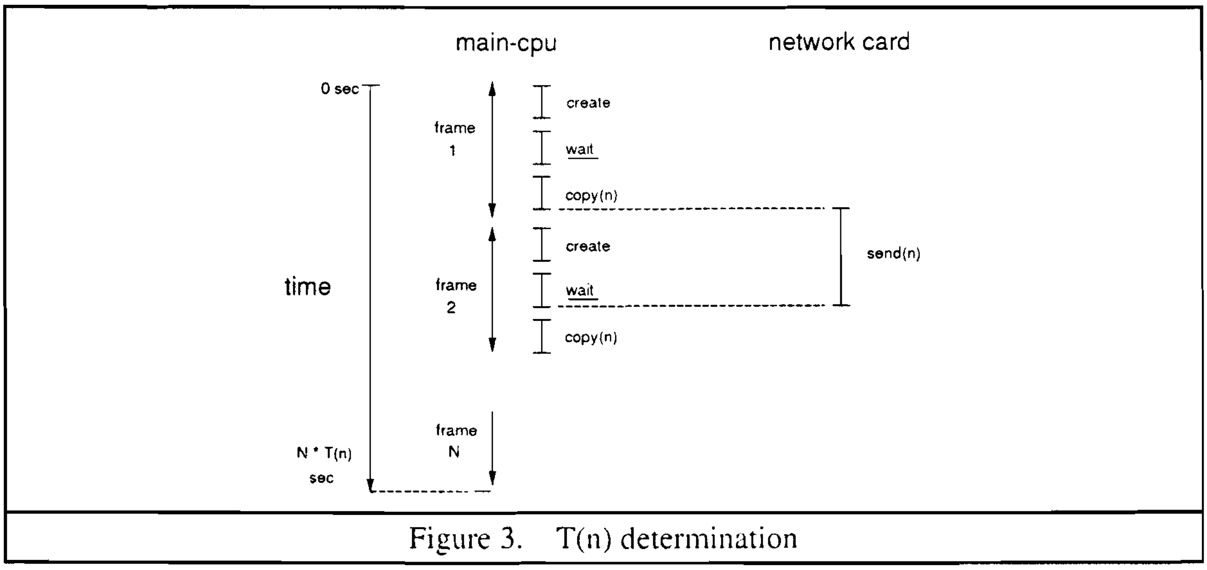

Figure 3. T(n) determination

The following assumptions are made:

(i) send(n) = *time needed by the network card to send an <n> bytes frame*

$$S_a * n + S_b \qquad (1)$$

Time needed to send a packet can be divided in a constant factor needed to do initialisation and other fixed time events. On top of that, time is needed to actually send the bytes ($n * S_a$). Each byte needs a fixed time (S_a sec/byte).

(ii) copy(n) = *time needed by the computer cpu to copy an <n> byte frame to the network card*

$$C_a * n \qquad (2)$$

Time needed to copy a packet to the network card has no fixed overhead, but in lineair/dependent on the number of bytes to be copied.

(iii) wait = *random variable describing the time between the moment a frame is ready to be sent by the network-card and the moment it is actually sent*

(iv) create = *time needed by the measurement program to create a new frame*

On an empty network, T(n) is described by (discarding startup effects):

N = 10.000 is number of packets send
T(n) = N * [copy(n) + create] if send < create
T(n) = N * [copy(n) + send(n)] if create < send

Table 1 and Figure 4 give the results of measuring T(n) for 5 different workstations on an empty ethernet segment. The workstations used are different types of microcomputers. The "m80" is an IBM PS/2 model 80. The "m70-1" and "m70-2" are two slightly different IBM PS/2 model 70. The "m290" is an Olivetti model 90 microcomputer. The "m30" is an IBM PS/2 model 30 microcomputer. All microcomputers were equipped with a Western Digital ethernet adapter.

This adapter has multiple send buffers, but only one is used in this test. Table 2 gives the calculated workstation parameters.
The device driver software for controlling the network adapter consisted of a packet-driver ([6]). All other software was developed in-house.

The results in Figure 4 show the stage where the main-CPU is the bottle-neck (create > send(n)) and where the network interface is the bottle-neck (create < send(n)). Furthermore, the results show that T(n) is linear in n on both areas. The first part of the lines in Figure 4 shows the CPU-bound area. The higher part ($n > n_{eq}$) displays the area where the network card is the bottleneck.

Table 1. T(n) (msecs) for computer parameter determination

n (bytes)	m80	m70-1	m70-2	m290	m30
100	0.33	0.49	0.49	0.89	1.11
200	0.36	0.53	0.53	0.99	1.19
300	0.40	0.57	0.58	1.09	1.27
400	0.48	0.62	0.62	1.19	1.36
500	0.60	0.67	0.67	1.29	1.44
600	0.72	0.76	0.77	1.40	1.53
700	0.84	0.88	0.89	1.50	1.61
800	0.96	1.01	1.02	1.60	1.69
900	1.07	1.13	1.14	1.71	1.78
1000	1.19	1.26	1.27	1.88	1.87
1100	1.31	1.38	1.39	2.06	1.94
1200	1.42	1.50	1.52	2.25	2.05
1300	1.54	1.63	1.64	2.43	2.21
1400	1.66	1.75	1.77	2.62	2.38
1500	1.78	1.88	1.89	2.80	2.55

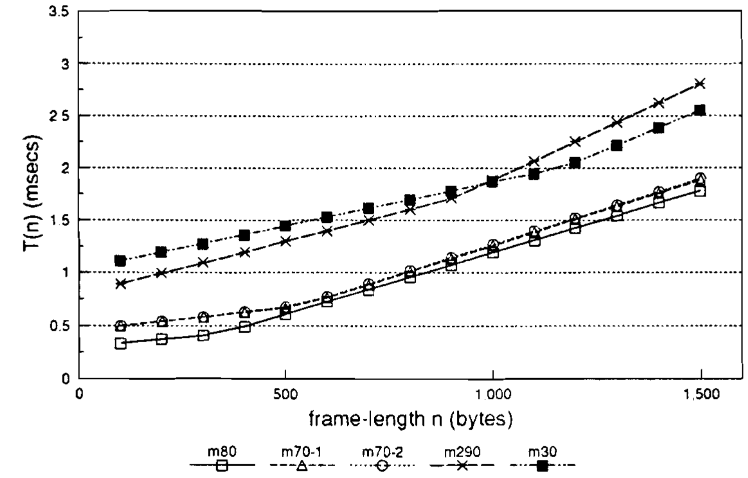

Figure 4. Computer parameter determination

The constants in formulas (1) and (2) are calculated using the measurements results (Table 2). The constants used in the simulation are the workstation characteristics as derived here.

The assumption made about the linearity of the workstation activities (copy (n) and send (n)) are confirmed by the measurements (Table 1 and Figure 4).

Table 2. Computer parameters (msec)

	m80	m70-1	m70-2	m290	m30
create	0 29510	0.45155	0.45065	0 78966	1 02817
Ca	0.00037	0.00042	0.00044	0 00102	0 00084
Sa	0 00080	0.00081	0.00081	0 00083	0 00083
Sb	0.02059	0.02221	0.02216	0.03386	0.05274

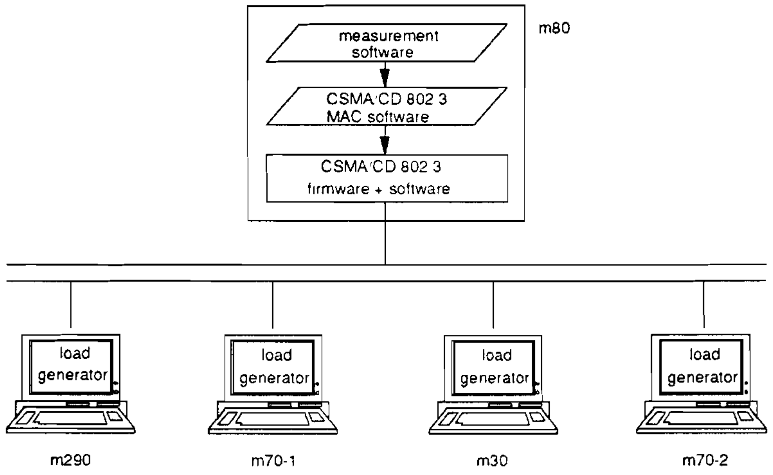

Figure 5. Measurement configuration

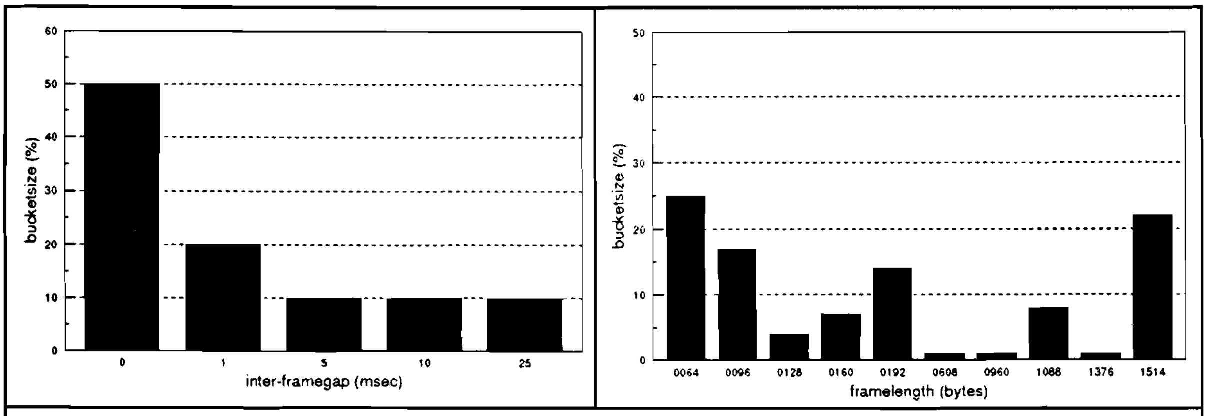

Figure 6. Frame-length and gaptime distributions

The next set of measurements involves the measurement of T(n) on a network with an artificial workload. The workload is generated with one to four load generators (workstations with special software) that generate frames with frame-lengths and interframe-times distributed as measured on the Delft University Campus network (Figure 5). This results in each load generator generating an average of 125 Kbytes/s (average inter-arrival gap 4.2 msec., average length 525 bytes), or a utilisation of 10% of the maximum network capacity. By letting the measurement station generate packets at full capacity, this will result in high utilisation of the ethernet segment.

The results of the measurements are given in Table 3 and Figure 7. It should be noted that the generation of a realistic workload can not be done with just one or two load generators, because of the lack of collisions.

Table 3. T(n) measurements (msecs) on a m-80

n (bytes)	0 gen's	1 gen	2 gen's	3 gen's	4 gen's
100	0.33	0.35	0.37	0.43	0.58
200	0.36	0.39	0.42	0.49	0.65
300	0.40	0.43	0.48	0.57	0.71
400	0.48	0.52	0.60	0.67	0.83
500	0.60	0.65	0.73	0.85	1.09
600	0.72	0.78	0.87	1.01	1.20
700	0.84	0.90	0.99	1.14	1.36
800	0.95	1.03	1.12	1.29	1.54
900	1.07	1.15	1.25	1.42	1.76
1000	1.19	1.26	1.38	1.58	1.90
1100	1.31	1.39	1.51	1.73	2.00
1200	1.42	1.51	1.66	1.88	2.07
1300	1.54	1.65	1.78	1.97	2.24
1400	1.66	1.76	1.92	2.11	2.43
1500	1.77	1.87	2.03	2.21	2.68

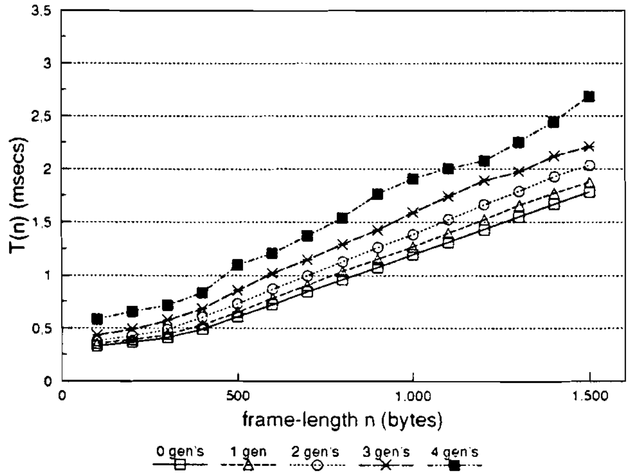

Figure 7. T(n) measurements

Simulation

In this section we will describe the simulation of an ethernet segment with several hosts using this segment as its broadcast medium. Aim of the simulation is to provide a tool in which it is possible to extend the configuration in a lab to a larger scale. The advantage of the simulation tool is that less equipment is involved and that the results are still very accurate. It can also be used to predict the behaviour of an extended configuration.

The simulation program is developed in a UNIX-environment and is written in standard C. It consists of a small set of functions, all especially written for this application. The design of the simulation is event-driven, i.e. after each event the configuration and administration are updated, followed by the determination of the next event that will happen in the system.

In the simulation the reality is imitated as close as possible. The IEEE Standard on CSMA/CD [1] is used as a guide-line. The collision handling is implemented exactly as given in the standard [1], including the interframe gap (9.6 microsec), jam size (32 bits) and slot time (512 bit times). The backoff mechanism in the simulation is the truncated binary exponential backoff algorithm with a backoff limit of 10 and an attempt limit of 16, i.e. the backoff time after a collision is a random integer multiple of the slot time, where the integer is chosen uniformly from the interval $[0, 2^k]$. In this expression k is the minimum of 10 and the number of previous attempts. A station cannot send a second frame until the first one is broadcasted successfully or failed 16 times to gain control over the medium. A frame that failed 16 times is discarded and the protocol layer above will be notified of this failure to broadcast.

The distribution for frame lengths and interarrival times between frames on one machine are defined by the same bucket mechanism as in the measurements.

In our simulation we will consider the same stages of the broadcasting process for a frame as considered in the measurements. Thus our observations start at the moment a frame is presented to the Media Access Control (MAC) layer and end when a frame is broadcasted successfully or discarded.

The workstation parameters used in the simulation are those already presented in Table 2. The functions describing Main-CPU and Network-card are implemented as being independent.

The simulation of a network without other load is a reference point for the parameter setting. Once these parameters are known we can simulate the same configuration as measured. The results can be found in Table 4 and Figure 8.

From Figure 9 one can see that the simulations yield almost the same results as the measurements. The differences can almost totally be explained by the random behaviour of the load generators. We conclude that all relevant aspects of the behaviour of an ethernet segment are included in the simulation model. The simple model of a network node described in 'Measurement' imitates the behaviour of the actual network node correctly.

An important question is the validity of these results for other configurations and distributions with the same overall load on the ethernet segment. Several cases with more machines or different distributions for the frame length and/or for the inter arrival times (e.g. batch arrivals, Poisson gaps) were simulated. In all these cases, the differences with the original configuration with an equal load were at most a few percent. Therefore, we may conclude that the ethernet protocol is rather insensitive for the distribution of the input data given a certain utilization of the net. The response times on an ethernet can be represented as a function of the load. In a forthcoming paper, we will discuss these variations in more detail [13].

Table 4. T(n) simulation (msecs)

n (bytes)	0 gen's	1 gen	2 gen's	3 gen's	4 gen's
100	0.33	0.35	0.38	0.41	0.50
200	0.36	0.39	0.43	0.49	0.56
300	0.40	0.43	0.47	0.54	0.64
400	0.48	0.52	0.58	0.65	0.76
500	0.60	0.65	0.72	0.80	0.96
600	0.72	0.77	0.85	0.96	1.11
700	0.83	0.90	0.98	1.12	1.32
800	0.95	1.02	1.12	1.25	1.48
900	1.07	1.14	1.24	1.39	1.68
1000	1.18	1.26	1.37	1.53	1.84
1100	1.30	1.39	1.50	1.67	1.97
1200	1.42	1.51	1.63	1.83	2.15
1300	1.54	1.62	1.76	1.98	2.28
1400	1.65	1.75	1.88	2.10	2.45
1500	1.77	1.86	2.00	2.23	2.63

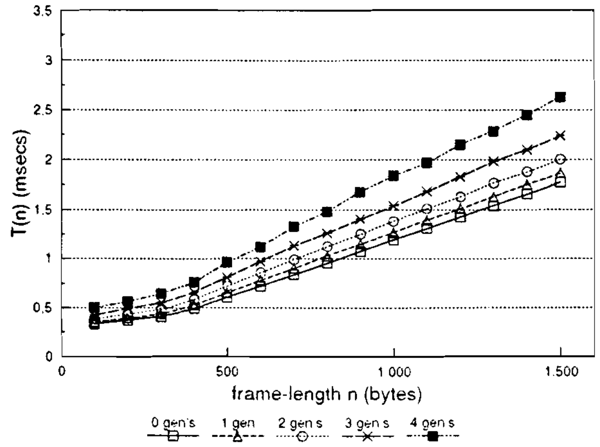

Figure 8. T(n) simulation

The simulation will be very helpful in the modeling and analysis phase. For modeling it is important that assumptions do not violate the reality of an ethernet segment. With a simulation it is easy to check whether an assumption will yield acceptable results or not by implementing this assumption and executing the adapted program.

A further advantage of a simulation is that it is possible to administrate every detail required. In a real system only a restricted small set measurements of a certain type can be done, while in a simulation no (such) restrictions exist.

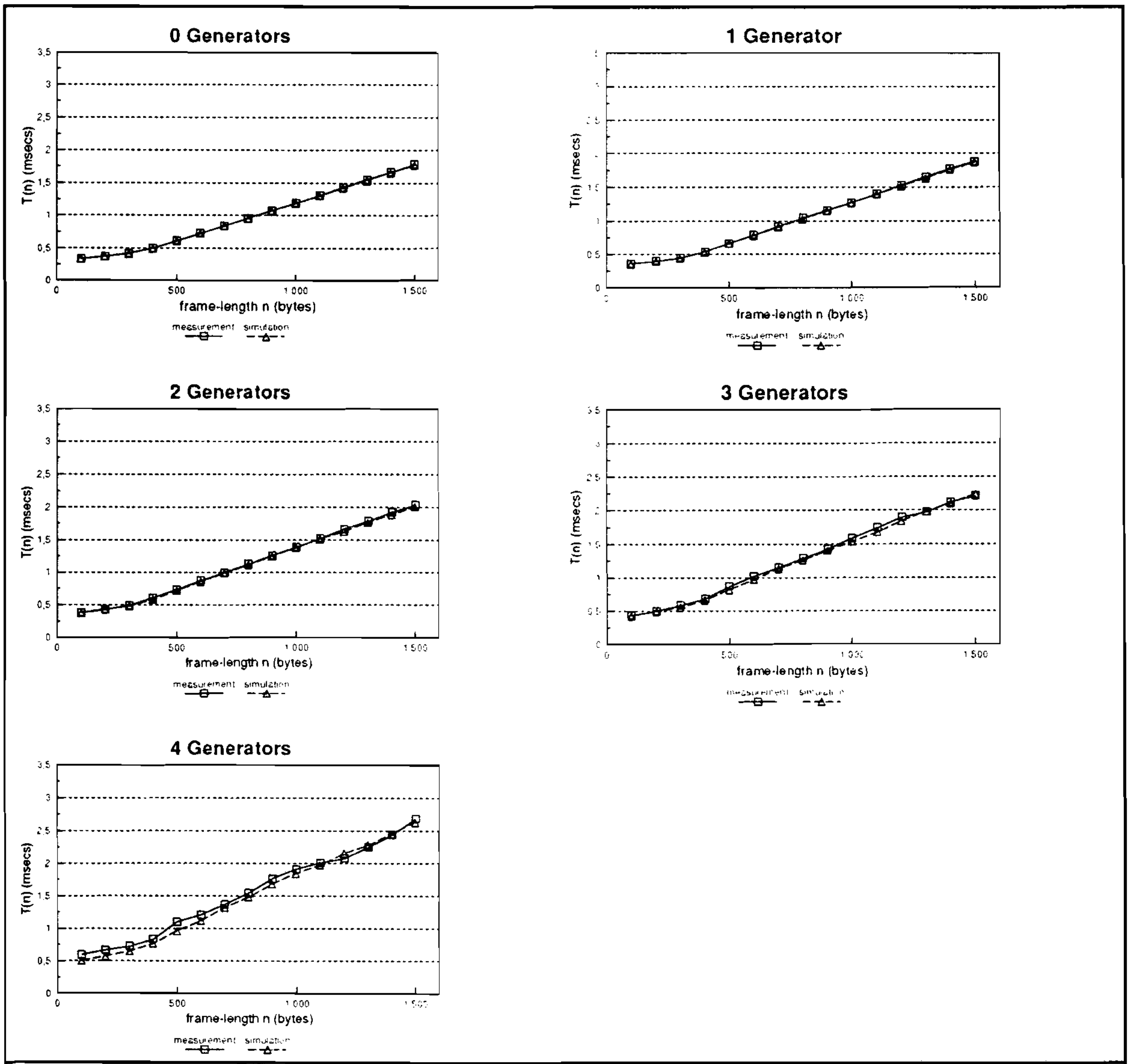

Figure 9. Differences T(n) measured and T(n) simulated

Conclusions

In this paper we have discussed the performance of an ethernet segment with the IEEE 802.3 CSMA/CD protocol. We have used two techniques in our study: measurements on a real ethernet in a laboratory environment and a detailed simulation of the standard protocol for it. We have concentrated on throughput measurements, i.e. we observed a system in which one station tried to transmit a large number of frames as quickly as possible. This was done for various workloads, generated by a different number of machines.

The results for the simulation of the standard protocol are very close to the measurements in a real ethernet. The small differences are due to the randomness of measurements and simulations. A network node can be modelled using a simple linear formula.

A second observation might be that the load in the net is a factor of importance to the response time of a message, for the waiting time is a substantial part of the transmission time. The distribution of inter-arrival time of frames is of little influence on the throughput times.

We assume that a realistic ethernet workload can be generated using four workstations as workload generators, as shown in [13], this assumption is valid.

The research presented here is a basis for new research in two directions. One direction is using these results for a mathematical analysis of the performance model of an ethernet segment. The other direction is to use these results as a start for equivalent research on higher protocol layers and/or an extension from one segment to more than one by using bridges or gateways.

Acknowledgements

We would like to thank prof. G.L. Reijns, G.E. Houtekamer, and J. van der Wal for their valuable suggestions and comments.
We also want to mention Mrs. Gerda Duvé who took care of the illustrations and the layout of this article.

Literature

[1] *IEEE Standards for Local Area Networks: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*, IEEE, New York, 1985.

[2] DDN Protocol Handbook, Volume one, "DOD Militrary Standard Protocols".

[3] *IEEE Standards for Local Area Networks: Logical Link Control*, IEEE, New York, 1984.

[4] Free Software Foundation, Packet driven specification and sources, Free Software Foundation, Cambridge, 1989.

[5] AJMONE MARSAN, M., ROSTAGNO, D. AND TRONCONE, S., "Throughput of CSMA/CD Protocols in LAN with Linear Topologies", pp. 243-262, *in Proceedings of the Third Conference on Data Communication Systems and Their Performance*, ed. L.F.M. de Moreas, E. de Souza e Silva and L.F.G. Soares, North-Holland, Amsterdam, 1988.

[6] "Information processing systems. Open Systems Interconnection - Basic Reference Model", *Draft International Standard ISO/DIS 7498*, International Organization for Standardization, ISO/TC97, April 1982.

[7] BERTSEKAS, D. AND GALLAGER, R., *Data Networks*, Prentice-Hall, Englewood Cliffs, New Jersey, 1987.

[8] COMER, D.E., *Internetworking with TCP/IP, Principles, Protocols, and Architecture*, Prentice-Hall, Englewood Cliffs, New Jersey, 1988.

[9] MURATA, M. AND TAKAGI, H., "Two-Layer Modeling for Local Area Networks", *IEEE Trans. on Comm*, Vol. 36, pp. 1022-1034, 1988.

[10] NEMZOW, M., *Keeping the Link, Ethernet Installation and Management*, McGraw-Hill, New York, 1988.

[11] TAKAGI, H. AND KLEINROCK, L., "Throughput Analysis for Persistent CSMA Systems", *IEEE Trans. on Comm.*, Vol. 33, pp. 627-638, 1985.

[12] TANENBAUM, A.S., *Computer Networks*, Prentice-Hall, Englewood Cliffs, New Yersey, 1988, second edition.

[13] DEKKERS, A. and OORSCHOT, J.P.M. van, *Sensitivity Analysis for a 802.3 Ethernet*, Internal Report TU Delft.

EINDHOVEN UNIVERSITY OF TECHNOLOGY
Department of Mathematics and Computing Science
PROBABILITY THEORY, STATISTICS, OPERATIONS RESEARCH
AND SYSTEMS THEORY
P.O. Box 513
5600 MB Eindhoven, The Netherlands

Secretariate: Dommelbuilding 0.03
Telephone : 040-473130

-List of COSOR-memoranda - 1991

Number	Month	Author	Title
91-01	January	M.W.I. van Kraaij W.Z. Venema J. Wessels	The construction of a strategy for manpower planning problems.
91-02	January	M.W.I. van Kraaij W.Z. Venema J. Wessels	Support for problem formulation and evaluation in manpower planning problems.
91-03	January	M.W.P. Savelsbergh	The vehicle routing problem with time windows: minimizing route duration.
91-04	January	M.W.I. van Kraaij	Some considerations concerning the problem interpreter of the new manpower planning system formasy.
91-05	February	G.L. Nemhauser M.W.P. Savelsbergh	A cutting plane algorithm for the single machine scheduling problem with release times.
91-06	March	R.J.G. Wilms	Properties of Fourier-Stieltjes sequences of distribution with support in [0,1).
91-07	March	F. Coolen R. Dekker A. Smit	Analysis of a two-phase inspection model with competing risks.
91-08	April	P.J. Zwietering E.H.L. Aarts J. Wessels	The Design and Complexity of Exact Multi-Layered Perceptrons.
91-09	May	P.J. Zwietering E.H.L. Aarts J. Wessels	The Classification Capabilities of Exact Two-Layered Peceptrons.
91-10	May	P.J. Zwietering E.H.L. Aarts J. Wessels	Sorting With A Neural Net.
91-11	May	F. Coolen	On some misconceptions about subjective probability and Bayesian inference.

COSOR-MEMORANDA (2)

91-12	May	P. van der Laan	Two-stage selection procedures with attention to screening.
91-13	May	I.J.B.F. Adan G.J. van Houtum J. Wessels W.H.M. Zijm	A compensation procedure for multiprogramming queues.
91-14	June	J. Korst E. Aarts J.K. Lenstra J. Wessels	Periodic assignment and graph colouring.
91-15	July	P.J. Zwietering M.J.A.L. van Kraaij E.H.L. Aarts J. Wessels	Neural Networks and Production Planning.
91-16	July	P. Deheuvels J.H.J. Einmahl	Approximations and Two-Sample Tests Based on P - P and Q - Q Plots of the Kaplan-Meier Estimators of Lifetime Distributions.
91-17	August	M.W.P. Savelsbergh G.C. Sigismondi G.L. Nemhauser	Functional description of MINTO, a Mixed INTeger Optimizer.
91-18	August	M.W.P. Savelsbergh G.C. Sigismondi G.L. Nemhauser	MINTO, a Mixed INTeger Optimizer.
91-19	August	P. van der Laan	The efficiency of subset selection of an almost best treatment.
91-20	September	P. van der Laan	Subset selection for an -best population: efficiency results.
91-21	September	E. Levner A.S. Nemirovsky	A network flow algorithm for just-in-time project scheduling.
91-22	September	R.J.M. Vaessens E.H.L. Aarts J.H. van Lint	Genetic Algorithms in Coding Theory - A Table for $A_3(n,d)$.
91-23	September	P. van der Laan	Distribution theory for selection from logistic populations.

91-24	October	I.J.B.F. Adan J. Wessels W.H.M. Zijm	Matrix-geometric analysis of the shortest queue problem with threshold jockeying.
91-25	October	I.J.B.F. Adan J. Wessels W.H.M. Zijm	Analysing Multiprogramming Queues by Generating Functions.
91-26	October	E.E.M. van Berkum P.M. Upperman	D-optimal designs for an incomplete quadratic model.
91-27	October	R.P. Gilles P.H.M. Ruys S. Jilin	Quasi-Networks in Social Relational Systems.
91-28	October	I.J.B.F. Adan J. Wessels W.H.M. Zijm	A Compensation Approach for Two-dimensional Markov Processes.
91-29	November	J. Wessels	Tools for the Interfacing Between Dynamical Problems and Models withing Decision Support Systems.
91-30	November	G.L. Nemhauser M.W.P. Savelsbergh G.C. Sigismondi	Constraint Classification for Mixed Integer Programming Formulations.
91-31	November	J.Th.M. Wijnen	Taguchi Methods.
91-32	November	F.P.A. Coolen	The Theory of Imprecise Probabilities: Some Results for Distribution Functions, Densities, Hazard Rates and Hazard Functions.
91-33	December	S. van Hoesel A. Wagelmans	On the P-coverage problem on the real line.
91-34	December	S. van Hoesel A. Wagelmans	On setup cost reduction in the economic lot-sizing model without speculative motives.
91-35	December	S. van Hoesel A. Wagelmans	On the complexity of post-optimality analysis of 0/1 programs.
91-36	December	F.P.A. Coolen	Imprecise Conjugate Prior Densities for the One-Parameter Exponential Family of Distributions.

COSOR-MEMORANDA (4)

91-37	December	J. Wessels	Decision systems; the relation between problem specification and mathematical analysis.
91-38	December	J.J.A.M. Brands R.J.G. Wilms	On the asymptotically uniform distribution modulo 1 of extreme order statistics.
91-39	December	C.J. Speck J. van der Wal	The capacitated multi-echelon inventory system with serial structure: 1. The 'push ahead'-effect.
91-40	December	C.J. Speck J. van der Wal	The capacitated multi-echelon inventory system with serial structure: 2. An average cost approximation method.
91-41	December	A. Dekkers J. v. Oorschot	Sensitivity analysis of the load on an ethernet network.
91-42	December	A. Dekkers J. v. Oorschot	Measuring and simulating an 802.3 CSMA-CD LAN.