AN INTEGRATED APPROACH TOWARDS THE SIMULATION ANALYSIS OF NETWORK PROTOCOLS*

Alessandro Merlo

Dipartimento di Informatica e Sistemistica

Università di Pavia

Via Abbiategrasso, 209 – 27100 PAVIA (Italy)

E–mail: merlo@gilda.unipv.it

ABSTRACT

Computer system performance evaluation can be approached from two different viewpoints. Measurements provide direct results about the actual behavior of a system. Modelling techniques provide predictions of the performance indices and are employed for system tuning and reconfiguration studies where performance and workload are forecasted.
In general, measurements and modelling are regarded as "alternative" techniques [Heid84]. The purpose of this paper is to combine these two approaches into an integrated one. A practical application of such an approach aimed towards the evaluation of the CSMA/CD protocol for an Ethernet local area network is also presented.

1 INTRODUCTION

Performance evaluation represents an important aspect at every stage in the life cycle of computer systems. Design, reconfiguration, and tuning activities are strongly affected by the results of performance evaluation studies, which require a deep understanding of both the hardware and the software components of the system, and the knowledge of the processing mechanisms of the system itself.
The architectures of computer systems, and the applications they support, are so different that computer performance evaluation has become a complex process. The first step of every performance study is to select the level of abstraction at which the system is to be evaluated, and to choose the most significant measures of performance.

Measurements are the only way to derive direct results about the actual behavior of a system, although, usually, neither the hardware nor the software of a system have been designed to be measured. Hence, the type and the amount of information that can be obtained are often limited. Measurements provide the values of performance indices for a system while it is processing a given workload. If we are interested in system design, tuning or reconfiguration, direct measurements are not feasible and modelling techniques must be adopted [Ferr83]. However, since performance cannot be expressed by quantities that are independent of the load, models cannot be adequately analyzed if the workload to be processed by the system is not specified. As a consequence, in order to drive system models it is necessary to develop first a workload model. This process, known as *workload characterization* [Ferr72] [Agra76], consists in describing from a quantitative and qualitative viewpoint the typical components of the load submitted to the system. This description is usually performed in terms of parameters selected according to the objectives of the study. After this phase, the parameters and their representation in terms of single numerical values, distribution descriptors, distributions, and so on, are available for the system.

Workload characterization represents the bridge between the measurement and the modelling activities, in that it yelds to an integrated approach towards system performance evaluation [Calz90].

The paper is organized as follows. The overall structure of the integrated approach proposed here is briefly outlined in Section 2. Section 3 presents an experimental application for the evaluation of the

CSMA/CD protocol of an Ethernet local area network. Finally, a few conclusions are drawn in Section 4.

2 AN INTEGRATED APPROACH

A correct performance evaluation activity can be completely realized by means of an integrated approach, in which different methodologies in the field of computer system performance evaluation are combined together as shown in Fig. 1.
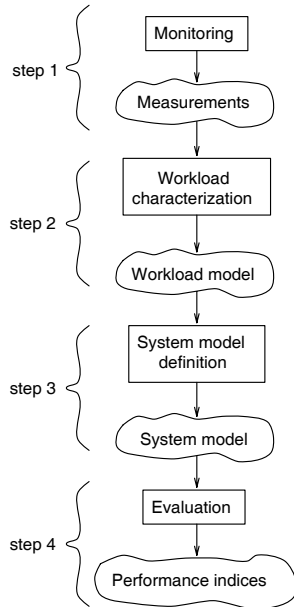

Figure 1: An integrated approach towards performance evaluation.

Starting from a real system, the first step of our integrated approach is represented by the measurements activity (step 1 in Fig. 1).

Measurability of a computer system can be defined in terms of the information accessible through the use of performance monitors [Svob76]. The amount of data to be collected in this phase varies considerably as a function of the measurement technique adopted (e.g., event detection or sampling) and of the objective of the analysis to be performed. Indeed, event detection records all the events of a given type (monitoring the changes in the system), while sampling is a statistical techniques dealing with a reduced set of collected measurements. Furthermore, a study of a local area network protocol only requires data about channel activities, whereas the evaluation of the network itself as a whole involves data collected from all the system components.

The data set collected in step 1 represents the input of step 2 (see Fig. 1) which deals with the workload characterization.

With respect to the objectives of the analysis a workload component can be an entire application, a user session, or simply a packet transmitted over a network. In any case, since the amount of data collected is often considerable, a manual analysis is impossible. Furthermore, the construction of accurate workload models can require multiple and combined applications of various types of techniques (e.g., scaling, fitting, cluster analysis) interacting together in different ways. Most of the tools for workload characterization implement only some of these techniques. As a result, from a user viewpoint, the modelling process is, in general, quite difficult. More than one tool is to be used, and, usually, there is no integration between one tool and the others.

The last two steps (step 3 and step 4 in Fig. 1) deal with the construction of a system model and with the evaluation of its performance indices, respectively. A model is an abstraction of the system under test, where the most significant components are represented. Note that before using a model for performance prediction it is necessary to test it against the real system.

The behavior of a computer system can be completely specified by the evolution in time of appropriate state variables. As a consequence, a model can be viewed as a representation of the system able to reproduce its state transitions [Lave89]. Input variables to step 3 (i.e., parameters) represent both the description of the characteristics of the workload submitted to the model, and the hardware and software specifications of the system under test. Output variables obtained from step 4 (i.e., the values obtained by solving the model) are the performance indices.

As can be seen in Fig. 1, the intermediate results produced by every step of this integrated approach can be used as input to the following phases.

It is important to notice that the integrated approach shown in Fig. 1 is not a pure sequential process. Indeed, according to the intermediate results obtained at each step, some feedback may be requested in order to obtain the best insight into the system behavior with respect to the objective of the analysis performed.

3 EXPERIMENTAL APPLICATION

As stated in Section 2, the application of our integrated approach towards the evaluation of the CSMA/CD protocol is such that the results obtained in each step can be viewed as input to the following phase. In this section, we will outline how this flow of information, from the measurements up to the final evaluation of the performance indices of the model, may be used on a real environment.

3.1 Monitoring

The system under test was a large DEC site, with several Ethernet local area networks connected by bridges. During the measurement interval, a number of transmitting stations varying from 372 up to 426 has been observed.
A hybrid monitor, namely, the SpiderAnalyser [Spid89], has been used to collect the packets flowing on the networks. Such a monitor consists of a data-gathering component, which is a hardware interface that senses the signals transmitted over the network channel, and of a software data reduction module.
The tool provides the values of performance indices, such as the percentage of collisions, and the utilization of the network bandwidth with respect to the total number of transmitting stations. The characteristics of each packet flowing on the channel has also been stored for further analyses.

The information concerning the packets transmitted over the communication channel are as follows:

```
Packet 1 (60 bytes) at T + 0.0010 secs
Delta t 0.0010 secs
Source = 08002B0A225D
Destination = AA0004007FB8
Type = 6004h
```

Every packet is characterized by:

- a unique identifier (e.g., `1`);

- its size expressed in bytes (e.g., `60 bytes`);

- the elapsed time between the instant (`T`) Spider-Analyser started monitoring the channel and the instant the current packet is detected (`0.0010 secs`);

- the interarrival time between the current packet and the previous one (`Delta`);

- the source and the destination addresses (e.g., `Source = 08002B0A225D` and `Destination = AA0004007FB8`);

- the packet type (`Type = 6004h`).
  Eight measurement sessions were collected with a global number of 5,336 packets which have been analyzed in the workload characterization phase.

3.2 Workload Characterization

The tool we used to characterize the Ethernet workload is the MEasuremets Description Evaluation and Analysis tool (MEDEA), developed at the University of Pavia [Merl93]. As already pointed out in Section 2, most of the tool for workload characterization implement only some specific functionalities. The main objective of MEDEA is to define an integrated and friendly environment, which allows the user to perform systematic workload modelling activity.
In order to make easier the application of standard statistical techniques for workload characterization, the tool provides all the facilities for selecting and reducing the measurements obtained by the monitoring tools. The integration of numerical methods, such as, scaling algorithms, sampling, functional characterization, fitting techniques, and cluster analysis, provides the user with a complete environment to perform workload modelling studies.

As far as our experimental application is concerned, the packet length (expressed in bytes) and the interarrival time (expressed in seconds) have been selected as representative parameters of the load of the Ethernet network. In the sequel, we will refer to these parameters as "packet_length" and "interarrival_time", respectively. In Figs. 2 and 3, the distributions of these two parameters obtained by means of the MEDEA tool are shown.

After this preliminary analysis, the clustering algorithm is applied. The statistical properties of the measured data set are examined through multidimensional data analysis techniques with the objective of identifying groups of items having homogeneous characteristics. The classification process, which represents the core for the construction of workload models, is based on the nonhierarchical *k−means* clustering algorithm [Hart75]. This algorithm groups together the components that are "similar" according to the minimization of some criteria.
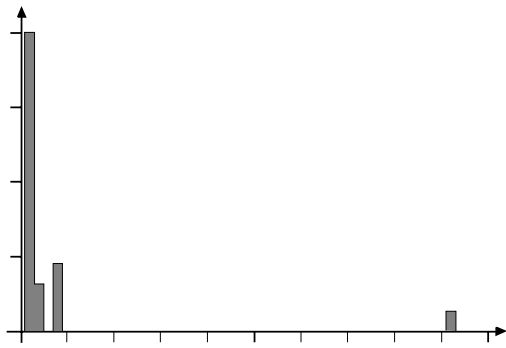
Figure 2: Distribution of parameter packet_length.
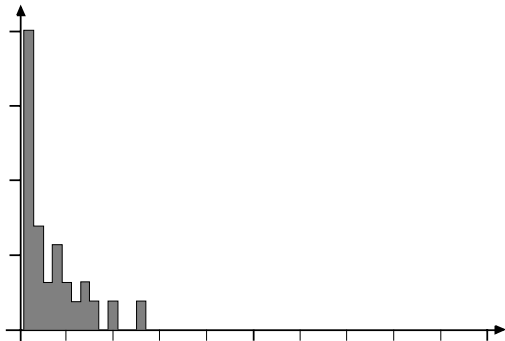

Figure 3: Distribution of parameter interarrival_time.

We have obtained three clusters, that correspond to the optimal number of classes with respect to the overall mean square ratio (see [Hart75]). Table 1 shows the values of centroids and standard deviations for the parameters selected.

	packet_length [bytes]		interarrival_time [seconds]	
	μ	σ	μ	σ
class 1	105.406	103.751	0.001	0.001
class 2	98.725	97.075	0.008	0.003
class 3	1322.447	92.931	0.004	0.003

Table 1: Centroids and standard deviations for the two parameters.

3.3 System Model Definition

Following the communication scheme provided by the 1–persistent CSMA/CD protocol for an Ethernet local area network [Lam80], a queueing network model of the protocol has been developed. The package we used for the system modelling activity is QNAP2, a software tool for the description and evaluation of queueing network models [Vera84]. Fig. 4 shows a graphical representation of the main components of the CSMA/CD model as obtained by means of QNAP2.
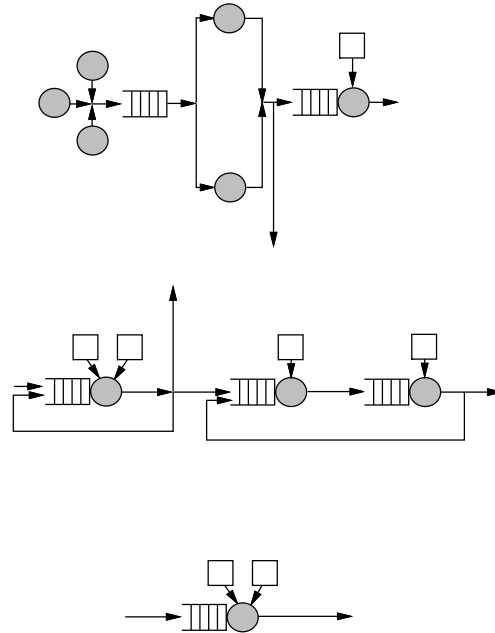

Figure 4: Overall structure of the CSMA/CD queueing network model.

The three queues labelled *source i* ($i = 1, 2, 3$) are responsible for the generation of all the packets that the stations connected to the Ethernet transmit over the communication channel. We have assumed as input parameters the statistical distribution of the identified classes of workload components. As far as system modelling is concerned, this probabilistic behavior is reproduced using exponentially distributed packet lengths and interarrival times with means μ and standard deviations σ derived from Table 1.

The queue *buffer_dispatch* is used to select an appropriate buffer where a packet can be stored before a station undertakes its transmission. This queue is also responsible for emptying a buffer once the packet it contains is successful transmitted.

The queue *bus_request_grant* is a local semaphore that allows the accesses to the bus for a specific station.

Once a station is ready for a transmission, the control is routed to the queue *listen_channel* which represents the listening phase of the CSMA/CD protocol. If the bus is sensed idle, the transmission starts and the queue *emit_try* is used to simulate the propagation delay over the communication channel. During this interval, if a collision is detected, the queue *call_detect* is responsible for the emission of the *jam* signal. In this case, the control is returned to the queue *emit_try*, otherwise the transmission is completed by the queue *complete_emit*.

As explained in Section 2, the input parameters for the model of the system includes also the hardware and/or software specifications of the system itself. The values assumed for the parameters related to the network configuration are summarized in Table 2.

Bus rate	10 Mbps
Intra–frame time	9.6 μs
Buffer length	1486 bytes
Jam length	32 bytes
Number of buffers for station	2

Table 2: Input parameters for the queueing network model of the CSMA/CD protocol.

3.4 Evaluation

The main objective of the evaluation phase is to derive estimations from the model by means of simulation in order to obtain the utilization of the communication channel and the percentage of packet collisions as the number of transmitting station increases.

The solution of the queueing network model of the CSMA/CD protocol is based on event–driven simulation. To choose a suitable simulation interval, we have made an analysis on the accuracy reached on the confidence intervals for the performance indices of some queue, and, finally, we have chosen a period of 3,250 time units. To enforce the generation of performance indices corresponding to a steady–state condition of the workload, we have adopted the *initial data deletion* technique (see e.g., [Jain91]). We have collected the data produced by the simulation starting after 250 units of time, up to 3,500 units.

The model has been solved for various configurations, i.e., for a number of workstations connected to the Ethernet local area network, ranging from 1 up to 15.

In Fig. 5 the percentage of utilization of the bus as a function of the number of workstations is shown.

This diagram was obtained with a total number of 1,375 packets/second generated by the three sources, in order to test the model of the protocol under heavy load conditions.
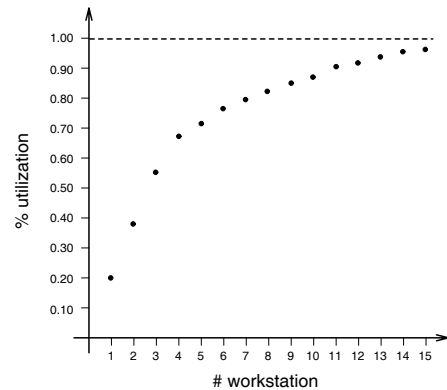

Figure 5: Channel utilization for the CSMA/CD model as a function of the number of workstations.

As far as the collision rate is concerned, the simulation results show the the maximum percentage of collisions is equal to 0.06%. This reflects a "good" behavior of the CSMA/CD protocol even when the communication channel is heavily utilized.

4 CONCLUSIONS

In this paper, we have presented an integrated approach towards the simulation analysis of network protocols. The main steps of this integrated methodology combine together both measurements and system modelling techniques in order to obtain significant performance indices. The bridge between these techniques is represented by the workload characterization process, as explained in Section 2. An application to the evaluation of the CSMA/CD protocol for an Ethernet local area network has also been discussed in Section 3. Future works will deal with the utilization of the integrated approach proposed in this paper towards performance evaluation studies of parallel and distributed environments.

References

[Agra76] Agrawala, A. K. R. M. Bryant, and J. M. Mohr. 1976. "An approach to the workload characterization problem." *IEEE Computer*, no. 9: 18–32.

[Calz90] Calzarossa, M. and L. Massari. 1990. "IMSE: An Integrated Modelling Support Environment". In *Atti 11° giornata A.I.C.A. "Valutazione delle Prestazioni dei Sistemi Informatici: Modelli e Sistemi Paralleli.*

[Ferr72] Ferrari, D. . 1972. "Workload characterization and selection in computer performance measurement." *IEEE Computer*, no. 4 (Apr.): 18–24.

[Ferr83] Ferrari, D.; G. Serazzi; and A. Zeigner. 1983. *Measurement and Tuning of Computer Systems.* Prentice–Hall, Engleewood Cliffs, N.J.

[Hart75] Hartigan, J. A. . 1975. *Clustering Algorithms.* John Wiley & Sons, Inc., New York.

[Heid84] Heidelberg, P. and S. S. Lavenberg. 1984. "Computer Performance Evaluation Methodology." *IEEE Transactions on Computers*, no. 12 (Dec.).

[Jain91] Jain R. 1981. *The Art of Computer System Performance Analysis.* John Wiley & Sons, Inc., New York.

[Lam80] Lam, S. S. .1980. "A Carrier Sense Multiple Access Protocol for Local Area Network". *Computer Networks* 4: 254–259.

[Lave89] Lavenberg, S. S. . 1989. "A Perspective on Queueing Models of Computer Performance." *Performance Evaluation* 10: 57–76.

[Merl93] Merlo, A. .1993. "MEDEA: MEasurements Description Evaluation and Analysis tool." Technical Report. Progetto Finalizzato C.N.R. "Sistemi Informatici e Calcolo Parallelo" (Apr.)

[Spid89] Spider Systems Limited. 1989. *SpiderAnalyser: User and Programming Guide.* Spider System Limited, Edinburgh.

[Svob76] Svobodova, L. . 1974. *Computer Performance Measurement and Evaluation Methods: Analysis and Application.* Elsevier, New York.

[Vera84] Veran M. and Potier, D. . 1984. "QNAP2: A Portable Environment for Queueing System Modelling." In *Proc. Int. Conf. on Modelling Techniques and Tools for Performance Evaluation.* Paris.