Queueing Model for Performance Analysis of a Network Interface Card

Naveen Cherukuri
82 Devonshire Street, #V7B
Boston, MA 02109

Natarajan Gautam
Marcus Department of Industrial and Manufacturing Engineering
Penn State University
University Park, PA 16802

Gokul Kandiraju and Anand Sivasubramaniam
Department of Computer Science and Engineering
Penn State University
University Park, PA 16802

Abstract

We consider a Network Interface Card (NIC) and develop an analytical model to predict its performance. The complexity of the problem is due to the fact that it is a combination of (a) multi-class queueing network with class switching, (b) polling system with limited service discipline, and (c) finite-capacity queues with blocking. However by identifying the bottleneck node and modeling it accurately, and modeling the rest of the nodes using approximations, we are able to analyze the system performance. We test our analytical results using simulations.

Keywords

Network Interface Cards, Performance Analysis, Queueing Model, Simulation

1. Introduction

A network interface card (NIC) is a computer circuit board or card that is installed in a computer so that the computer can be connected to a network. Personal computers and workstations on a local area network (LAN) typically contain a network interface card specifically designed for the LAN transmission technology, such as Ethernet or token ring. NICs are also used to interconnect clusters of computers or workstations such that the cluster can be used for high performance or massively parallel computations. Although clusters are slowly replacing large supercomputers due to their low cost, one of the biggest stumble blocks for clusters to reach the performance of supercomputers is that their NICs are inefficient. Research is under way to improve the efficiency of the NICs, however there are no analytical models for NICs that the researchers could use to quickly test design alternatives. In fact currently only simulations are used by these researchers and each design alternative could take several minutes to hours to evaluate. In this paper we develop one of the first and accurate analytical model for a NIC known as the Virtual Interface Architecture (VIA) NIC. VIA [1] is an effective architecture for the interface between high performance network hardware and computer systems. The VIA approach is an important step in achieving efficient and affordable performance improvements in microprocessor-based server platforms.

Our analytical model is based on a network of queues. In particular, a multi-station and multi-class open queueing network model is proposed to capture the multitude of operations and queues in the NIC. This approach has not been cited or used in the existing literature on NICs, and the present work can act as a precursor for future improvements on the proposed model. The complexity of the problem is due to the fact that it is a combination of (a) multi-class queueing network with class switching, (b) polling system with limited service discipline, and (c) finite-capacity queues with blocking. However by identifying the bottleneck node and modeling it accurately, and modeling the rest of the nodes using approximations, we are able to analyze the system performance. We develop simulations based on ARENA [2] for benchmarking and the results will be compared with the performance measures obtained in the analytical model.

The rest of this paper is structured as follows. Section 2 deals with some preliminaries including a description of the VIA NICs with some details on the information flow in the VIA NICs. Section 3 provides a detailed explanation of the various aspects in the queueing network model and the simplifications used in obtaining the analytical model. In Section 4, performance of the analytical model is compared against simulation studies. This paper concludes with highlights of the summary of the study, contributions from this research, and recommendations for future work in Section 5.

2. Preliminaries

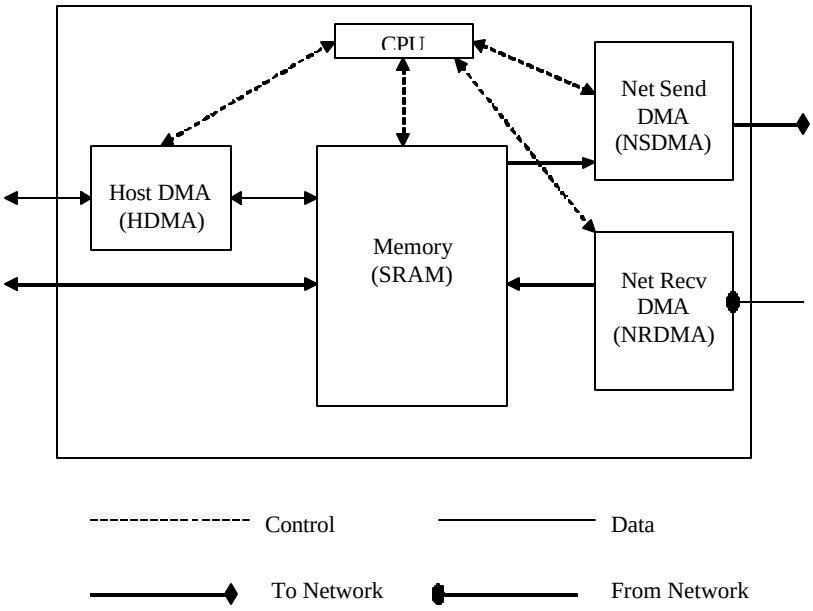

Figure 1: Network Interface Card

Figure 1 [3] shows a Myrinet (brand name for a commercial VIA NIC) NIC. Myrinet [4] is popular for deploying clusters because it provides high hardware transmission rates. The several hardware features that it provides make VIA implementations more efficient. It contains a CPU called LANai, a Direct Memory Access (DMA) engine (represented as HDMA) which is used to transfer the data between the host memory and card buffer (SRAM), a DMA engine (represented as NSDMA) to transfer the data from SRAM onto the network, and a DMA engine (represented as RSDMA) to transfer data on to the SRAM from the network. From a modeling point of view, sending can be translated to appending a message to a queue in the card buffer and receiving can be translated to removing a message from the card buffer. Under program control, a NIC copies data from memory to the network medium, transmission, and from the medium to memory, reception, and implements a unique destination for messages traversing the network.

The LANai goes through the following operations cyclically: polling the doorbell queue, polling the descriptor queue on SRAM and polling the data queue. In addition, it programs NSDMA and NRDMA to send and receive the data to and from the network respectively. LANai polls the doorbell queue and makes them available for HDMA to obtain the corresponding descriptors. Polled doorbells wait in a queue at HDMA to get serviced on a FCFS basis. They are processed by HDMA and the corresponding descriptors are stored in the descriptor queue on SRAM. The descriptors in this queue are polled by LANai and it makes them available for HDMA to obtain the corresponding data. In the case of a send descriptor, LANai initiates the transfer of data from the host memory on to the data queue on SRAM using HDMA. In the case of a receive descriptor, LANai initiates the transfer of data (if any) from the network queue at NRDMA to the data queue on SRAM using NRDMA. LANai polls the data queue and if the polled data is of type “send”, it checks whether NSDMA is busy. If not, it initiates the transfer of send data from

SRAM data queue to NSDMA. If the polled data is of type "receive", it initiates the transfer of data from SRAM data queue to host memory using HDMA.

Notice that the system is a multi-server queueing network. Since there are multiple commodities in the network we need to use a multi-class queueing network model with class switching. Also the LANai can be thought of as a polling system with limited service discipline. Finally the NSDMA queue is a finite-capacity queue with blocking. The system is analyzed by actually obtaining inter-arrival time data (of doorbells) and service time data from a Myrinet NIC used in the Computer Science and Engineering department. We observe that the bottleneck is the HDMA queue.

3. Analytical Model

We model the performance of the NIC of a send station, where the data flow is from the station to the network. In essence, the hardware features of NIC consisting of only NSDMA are modeled in an open queueing network. This simplified version of NIC consists of send doorbell, send descriptor and send data as three classes of traffic in the network. Before describing the model we first explain some simplifications made. The following simplifications are made in order to model the system as a multi-class network of queues by taking advantage of the fact that the bottleneck is at the HDMA queue, however the modeling complications are at the LANai and NSDMA queues:

(1) Mathematical approximation for polling by LANai: In a NIC, the LANai polls three queues, namely, doorbells, descriptors, and data. In the analytical model, we approximate LANai to be a node with a single queue with multiple class traffic corresponding to doorbells, descriptors, and data. This approximation converts the polling system into a single multi-class queue with a single server.

(2) Mathematical approximation for programming of NSDMA by LANai: In a NIC, the NSDMA is a node with zero waiting space and the service of data traffic at NSDMA depends on four scenarios outlined in Table 1 below. Analytically, NSDMA is modeled as a node with a single queue and infinite waiting space. This approximation converts the NSDMA queue which is a finite-capacity queue with blocking into a single queue of infinite capacity.

Scenario	Availability of Data	Status of NSDMA	Programming of NSDMA by LANai
1	Available	Busy	No
2	Available	Idle	Yes
3	Not Available	Busy	No
4	Not Available	Idle	No

Table 1 Four different scenarios for programming of NSDMA by LANai

(3) Estimated probability for Scenario 2 in Table 1: An approximation is used in estimating the probability of Scenario 2, i.e., that there is data available in SRAM when NSDMA is idle. This approximation is needed to obtain an estimated service time for data traffic at node LANai and model NSDMA station as having a single queue with infinite waiting space. Analytical results will be compared with the simulation results, where in the analytical results we use $p = 0.5$, which is not done in the simulations.

(4) Class switching: There is a class switching involved in the network. After getting served at node HDMA, doorbells are converted to descriptors. Similarly, after getting served at node HDMA, descriptors are converted to data. Standard decomposition algorithms in the queueing network literature can be applied only when class switching is not present in the queueing network. However, as an approximation, we will ignore this and use the existing algorithms.

Using the above approximations, we now build an analytical model of the system. Figure 2 shows an approximate model of the three-station and three-class open queueing network proposed for the performance analysis of the VIA NIC. The servers in the network correspond to LANai, HDMA and NSDMA and henceforth will be referred to as nodes. We use the QNA approach by Whitt [5,6] for our analysis of the multi-station and multi-class open queueing network.

For the analysis, we will use Poisson arrivals of doorbells and constant service times at the various nodes. This is based on the measurements made in a VIA NIC at a cluster in the Computer Science and Engineering department in this university. Table 2 describes the various service times used (note that p is the probability there is data available in the SRAM when NSDMA is idle – see (3) above).

Class	Node 1 (LANai)	Node 2 (HDMA)	Node 3 (NSDMA)
1 (doorbells)	22	21	N/A
2 (descriptors)	0.12	68.3154	N/A
3 (data)	10*p	N/A	52.6887

Table 2 Service times in microseconds

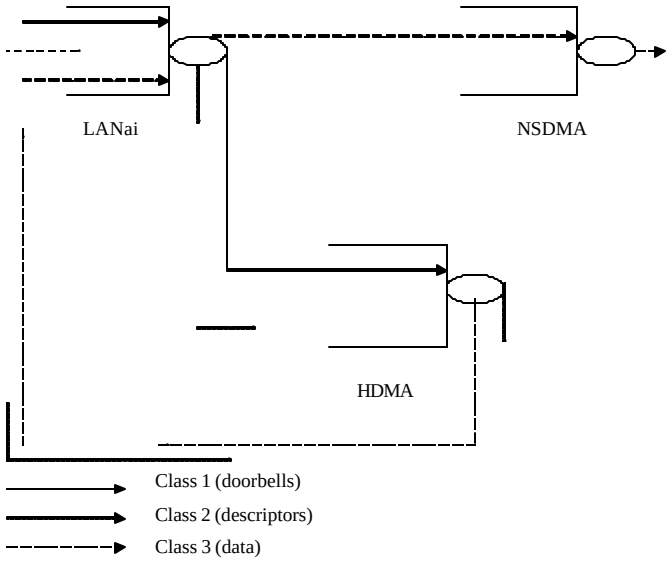

Figure 2: Three-station and three-class open queueing network approximation for NIC

4. Numerical Results

In this section we present a comparison between the analytical results derived in Section 3 against simulations performed on ARENA. None of the approximations made in the analytical model are made in the simulation. Thereby the analytical approximations will be valid if the results are close to the simulations. Table 3 shows the comparison in terms of the utilization of the different nodes for various arrival rates. The utilization results match very closely for HDMA and NSDMA nodes, with the maximum error percentage being 0.35. The deviation in the utilization values for the node LANai is due to the approximation involved in the estimated probability of 0.5 that is chosen for the probability of messages being present in the data queue when NSDMA is idle.

Arrival Rate λ	Analytical Utilization of LANai	Simulated Utilization of LANai	Analytical Utilization of HDMA	Simulated Utilization of HDMA	Analytical Utilization of NSDMA	Simulated Utilization of NSDMA
0.00273	0.0721	0.0875	0.2438	0.2430	0.1438	0.1433
0.00493	0.1273	0.1586	0.4403	0.4393	0.2597	0.2591
0.00786	0.1969	0.2551	0.7020	0.7015	0.4141	0.4138
0.00900	0.2227	0.2935	0.8039	0.8032	0.4742	0.4738
0.01079	0.2620	0.3564	0.9637	0.9637	0.5685	0.5683
0.01100	0.2664	0.3640	0.9825	0.9822	0.5796	0.5794

Table 3 Comparison of utilizations of nodes in analytical and simulation models

In terms of performance measures, the analytical model is compared against the simulation by studying the mean queue lengths (Tables 4 and 5). Notice that the HDMA queue is the bottleneck and is therefore studied in detail in Table 4 as well as Figure 3. The HDMA queue length that is predicted using the analytical model is less than the actual queue length, which is obtained using the simulation model. Since the analytical utilization of LANai is lower than the simulation utilization, a better estimation for the probability of Scenario (2) explained in Section 3 may be needed which could bring the analytical mean queue length of HDMA close to the simulated mean queue length of HDMA. Also, the last two rows in Table 4 correspond to utilizations of HDMA that are greater than 0.95,

and the analytical queue length that is obtained is within 23% of the actual value. The queue length of node LANai can be compared by summing up the lengths of the queues that are polled by LANai in the simulations (Table 15).

Arrival Rate λ	Analytical mean queue length at node HDMA	Simulated mean queue length of HDMA Queue	% Error in the analytical value
0.00273	0.0480	0.0465	3.22
0.00493	0.1922	0.2002	-4.00
0.00786	0.8007	0.9438	-15.16
0.00900	1.5285	1.8653	-18.05
0.01079	11.2929	14.576	-22.52
0.01100	24.1981	30.499	-20.66

Table 4 Mean queue length at HDMA in analytical and simulation models

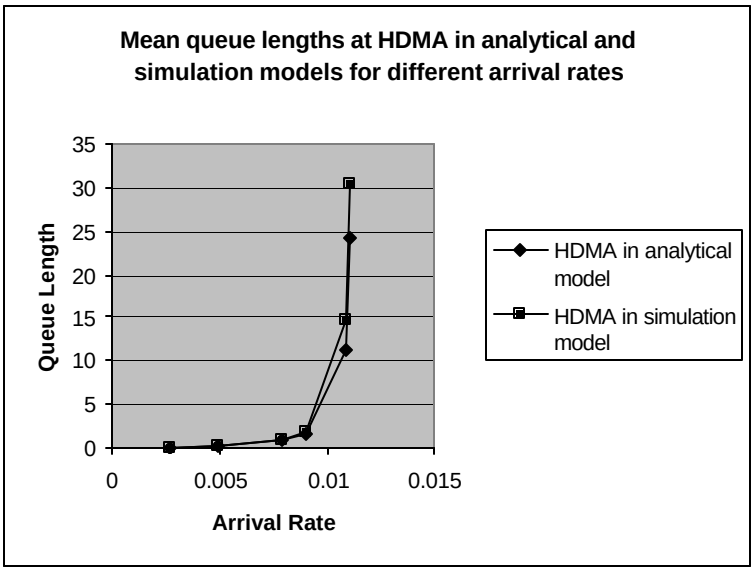

Figure 3: Mean queue lengths at HDMA in analytical and simulation models

Arrival Rate λ	Analytical queue length of node LANai	Sum of the lengths of three individual queues which LANai polls in simulation model
0.00273	0.0059	0.0064
0.00493	0.0191	0.0222
0.00786	0.0486	0.0626
0.00900	0.0642	0.0854
0.01079	0.0940	0.1317
0.01100	0.0980	0.1378

Table 5 Mean queue lengths at LANai in analytical and simulation models

6. Concluding Remarks and Future Work

This section summarizes this research work and provides pointers to future developments possible in this area. The first section provides a short synopsis of modeling, simulation and the comparison between them to check the validity. The second section highlights the research accomplishments, and the third section presents possible enhancements that can be made in the future.

6.1 Summary of the Research Work

A multi-station and multi-class queueing network model is used to study the performance of Myrinet VIA NIC. The stations correspond to LANai, HDMA and NSDMA. Different messages in the system, which are doorbells, descriptors and data, are modeled as different classes of traffic in the queueing network. Various simplifying

assumptions are made which are essential for applying the proposed analytical model. From the point of view of design, two mathematical abstractions (modeling LANai as a node where as in reality it is a processor which visits certain number of queues in a predetermined order and modeling the programming of NSDMA by LANai to pick the available data messages on SRAM when it is idle) are needed to develop a mathematically tractable model. Important performance measures for the design of NIC are obtained analytically. The bottleneck node of the system is the HDMA queue. Summarizing the comparison between the analytical and simulation performance measures: in most cases, analytical performance measures are less than simulated performance measures; utilization values of HDMA and NSDMA match in both models; utilization values of LANai in analytical model are lower than the values in simulation model; the analytical model predicts the mean queue length of the bottleneck node in the network (HDMA) within an average error of 14% and a peak-utilization error of 20-25%, which are fairly good estimates. At lower utilizations, the model predicts the mean queue length of HDMA with higher accuracy.

6.2 Contributions from this Research Work

The contributions fromthis research work are as follows.

(1) Applying the existing queueing modeling principles to obtain the performance measures of a NIC. It provides an alternative to obtaining performance measures by testing or by simulations, which may be expensive and computationally time intensive.

(2) Identifying the bottleneck in the system to be HDMA, which is the main direct memory access DMA engine for processing doorbells and descriptors.

(3) Ability to study various design alternatives for the components of NIC quickly. For example, changing the order of service for LANai is not going to affect the performance.

(4) Reducing the effort in computation to obtain the performance measures approximately in a much easier and faster way. Once the required parameter values are put in the algorithm for a NIC, the analytical results are obtained in less than a second. Table 6 provides the time taken for the simulation.

Arrival Rate λ	Time taken in minutes for the simulation model
0.00273	3.55
0.00493	7.55
0.00786	16.12
0.00900	20.93
0.01079	35.07
0.01100	37.58

Table 6 Time taken in minutes for the second simulation model to run

6.3 Recommendations for the Future Work

Interesting opportunities are possible for enhancing the present research work.

(1) Analytical design of a send-and-receive NIC. Additional approximations need to be made for modeling the behavior of NRDMA that increases the complexity.

(2) Proposing a model for developing the estimated probability in an analytical manner. Present research work proposed a rough and arbitrary value.

(3) Different policies for the operation of LANai. Changing the order of operations is not going to affect the performance measures. Other changes in operations for LANai might be of interest for the performance improvements. An example would be to think of operations where LANai, at a time, polls more than one message at specified queues.

References

[1] Compaq Corp., Intel Corp., and Microsoft Corp. Virtual Interface Architecture Specification, Version 1.0. Available at http://www.viarch.org.

[2] Rockwell Software, Sewickley, PA, USA. http://www.arenasimulation.com.

[3] Nagar, S., Liu, C., Kandiraju, G., Sivasubramaniam, A., and Gautam, N., "Incorporating Quality-of-Service in the Virtual Interface Architecture", Proceedings of IPDPS, April 2002.

[4] Myricom, Inc., Arcadia, CA, USA. http://www.myri.com.

[5] W. Whitt, “The Queueing Network Analyzer”, The Bell System Technical Journal, Vol. 62, No. 9, November 1983.

[6] W. Whitt, “Performance of the Queueing Network Analyzer”, The Bell System Technical Journal, Vol. 62, No. 9, November 1983.