Fan, X. and M. Jonsson, "Guaranteed real-time services in switched Ethernet networks with deadline scheduling in the end nodes," Proc. Swedish National Computer Networking Workshop (SNCNW'04), Karlstad, Sweden, Nov. 23-24, 2004.

Guaranteed Real-Time Services in Switched Ethernet Networks with Deadline Scheduling in the End Nodes

Xing Fan and Magnus Jonsson

Centre for Research on Embedded System, Halmstad University, Halmstad, Sweden,
Box 823, S-301 18, Sweden. {Xing.Fan, Magnus.Jonsson}@ide.hh.se, http://www.hh.se/ide

Abstract

This paper proposes a switched Ethernet network that supports real-time communication with guaranteed bit rate and end-to-end delay bound. In our proposal, the source nodes use Earliest Deadline First (EDF) scheduling, while the switch uses First Come First Served (FCFS) to control periodic real-time traffic belonging to logical real-time connections. The schedulability condition is drawn and the end-to-end worst-case delay bound is derived for guaranteed real-time traffic, which also gives, as a sub-result, the needed buffer space in the switch. Moreover, different traffic classes are prioritized and put into different output queues in the end nodes and the switch, which minimizes the interference with other traffic when transmitting periodic time-critical messages. The solution requires no additional hardware or software modifications of the switch and the underlying standard. It is based purely on software implementation in the end nodes. Furthermore, the EDF scheduling strategy used in the source nodes allows good resource utilization, while the FCFS scheduling used in the switch limits the switch processing overhead. The paper presents simulation analysis for Fast Ethernet, which verifies our approach of guaranteeing real-time transmission at the same time as achieving high utilization.

1. Introduction

Real-time applications, for example, industry process control, radar signal processing systems, automation control and other industry applications, require strict time-deterministic transmissions. In such kind of systems, the frames containing real-time information, such as control commands and alarm signals, have to be delivered within a certain time limit. In other words, the correct performance is specified as packet delivery within a certain delay bound.

Switched Ethernet is prevalent used now and will probably take over much of the industrial bus and network markets in the future [1], because it is cost-effective and enables some key benefits over traditional Ethernet, such as full duplex and flow control. A number of protocols and schemes have been proposed to improve the real-time characteristics of switched Ethernet [2-10]. However, these results are either based on some assumptions about traffic arrival which are not exactly representative to the messages sent by the applications [2-5], or do not introduce the Quality of Service (QoS) mechanisms in their models [6].

The goal of the EtheReal project [7] [8] was to build a scaleable real-time Ethernet switch, which supports bandwidth reservation and guarantee without any hardware or operating system modification. However, EtheReal was only throughput-oriented, i.e., no explicit treatment of hard real-time communication.

Results on establishing real-time channels using deadline sorting in the switch to gain real-time support are found, in switched Ethernet networks [9] [10] and in packet-switched networks [11] [12]. However, the deadline-based sorting in the switch brings high processing overhead, which motivates us to investigate a new strategy to achieve the same approach.

Similar method [9-12], of using logical real-time connections abstraction for packet scheduling and supporting real-time capability is proposed in this paper. Nevertheless, there are several important differences. First, the EDF algorithm is used in the source nodes while the FCFS policy is used in the switch for real-time traffic scheduling. Second, we obtain the schedulability condition and the needed buffer size for hard real-time traffic under our scheduling policy, and prove that we can guarantee

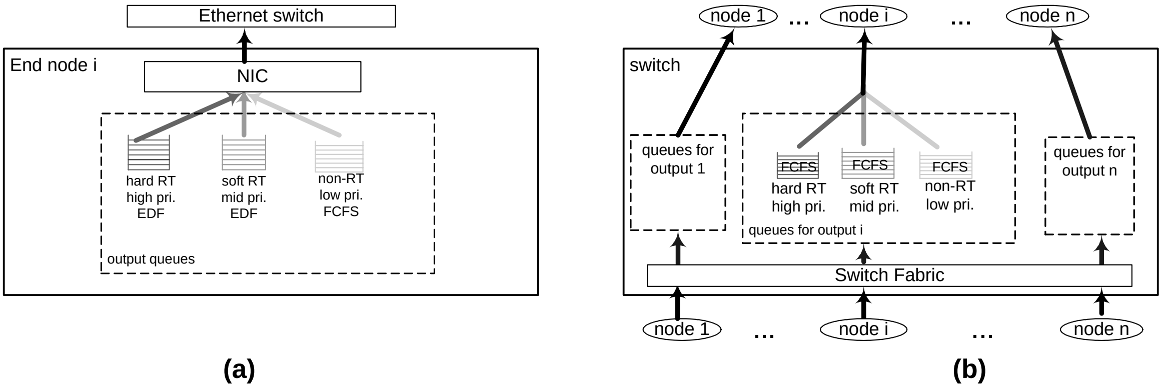

Figure 1. Output Queues and traffic handling. (a) in an end node, (b) in the Ethernet switch.

the end-to-end delay bound. Third, real-time services are provided in different QoS levels, for example, hard real-time, soft real-time and non-real-time traffic [13], in order to ensure that time-critical packets are not hindered by a stream of non-critical traffic.

The rest of the paper is organized as follows. In Section 2, we present the network architecture and traffic handling. The real-time analysis is described in Section 3. In Section 4, the simulation analysis is reported. Finally, Section 5 concludes the paper.

2. Network architecture and traffic handling

We consider a network with the topology of an Ethernet switch and a number of end nodes (a single switch is assumed in this paper). Every node is connected to other nodes via the switch. The switch establishes a direct line of communication between two points for each frame, and maintains multiple simultaneous links between various ports.

The IEEE 802.1p [14] queuing feature, which enables Layer 2 switches to prioritize traffic and perform dynamic multicast filtering, is used to classify different traffic classes in our system.

Figure 1a and 1b show the output queues and traffic handling policy in the end node and in the switch, respectively. In each end node, before entering the Network Interface Card (NIC), the hard real-time frames are put into the hard real-time traffic queue, which has the highest priority among all traffic classes, while soft real-time frames are put into the soft real-time queue having lower priority than the hard real-time queue (Figure 1a). Both the hard real-time queue and the soft real-time queue are sorted according to EDF [15]. Outgoing non-real-time traffic from the end-node is treated as lowest priority and put into a FCFS sorted non-real-time queue.

In the same way, there are three different output queues for each port in the switch too (Figure 1b). In our configuration, only the hard real-time queues at the end nodes use the EDF algorithm, while all queues in the switch follow FCFS scheduling. Our network supports dynamic addition of real-time channels, each being a virtual unidirectional connection between two nodes in the system with guaranteed bit rate and bounded delay.

3. Real-time analysis

The real-time guarantee for the hard real-time traffic is uphold by a real-time channel with index i, which is characterized by:

$$\{ S_i, D_i, T_{period,i}, C_i, T_{deadline,i} \}, \quad (1)$$

where S_i is the index number of the source node, D_i is the index number of the destination node, $T_{period,i}$ is the period of data generation, C_i is the amount of data per period, and $T_{deadline,i}$ is the relative deadline used for the end-to-end scheduling. $T_{period,i}$, C_i, and $T_{deadline,i}$ are expressed as the number of maximum-sized Ethernet frames.

Before we describe the feasibility test, we make the following definitions, by looking upon the real-time channel as a periodic task.

HyperPeriod: The *HyperPeriod* for a set of periodic tasks is defined as the length of time from when all tasks' periods start at the same time, until they start at the same time again.

BusyPeriod: A *BusyPeriod* is any interval of time in which a link is not idle.

Workload function: The *workload function* *h(t,s,d)* is defined as the sum of all the capacities of the tasks with deadline less than or equal to *t*, running on the physical link from node *s* to node *d* (the switch is always one of the nodes in our case), where *t* is the time duration elapsed from the start of the *hyperPeriod*.

Feasible link: A *feasible link* is a link with a set of channels traversing it that can be feasibly scheduled using our scheduling policy.

***Feasible system state**: A feasible system state* is a system state with every link in the system being feasible.

Following the above discussion, the problem for the admission controller to test if the channel can be added or not is to test if the new state is still feasible, given that the new channel has been added. The feasibility test is done in two steps, the utilization constraint and the workload constraint, each step being a test of its own.

The utilization constraint is checked first. According to basic EDF theory, the utilization for a physical link, *U*, should be less than or equal to the maximum value, 100%:

$$U = \sum_i \frac{C_i}{T_{Period,i}} \le 100\% \,. \qquad (2)$$

The second constraint (workload constraint) is that for all values of *t*, the workload function *h(t,s,d)* has to be less than or equal to *t.*

$$h(t,s,d) \le t \,. \qquad (3)$$

Since EDF is not used in the switch for scheduling in our scheme, the workload function The workload function *h(t,s,d)* here is not defined as the sum of all the capacities of the tasks with *absolute deadline*, $T_{deadline,i}$, less than or equal to *t*, but as the sum of all the capabilities of the tasks with the *first hop deadline*, $T_{d1,i}$, less than or equal to *t*.

Several real-time channels might exist in the network at the same time. The workload function *h(t,s,d)* are calculated for each physical link from the source node *s* to the destination node *d*, as follows:

$$h(t,s,d) = \sum_{(i:S_i=s)} \left(1 + \left\lfloor \frac{t - T_{d1,i}}{T_{period,i}} \right\rfloor \right) C_i \,, \qquad (4)$$

where the *first hop deadline*, $T_{d1,i}$, is calculated as:

$$T_{d1,i} = T_{deadline,i} - T_{d2,i} \,, \qquad (5)$$

where $T_{d2,i}$ is the worst case delay in the switch. The instance for a packet to arrive at the switch will never be worse than that when all packets for all the guaranteed logical real-time channels, with the same destination, arrive just before. Hence $T_{d2,i}$ is calculated as:

$$T_{d2,i} = \max\left(\bigcup_t \left\{ \sum_{(j:D_j=D_i)} \left(\left\lceil \frac{t}{T_{period,j}} \right\rceil \right) C_j \right\} \right) . \qquad (6)$$

Therefore, the buffer size for the hard real-time traffic in the output port *i* of the switch, B_i, expressed by the number of maximum-sized frames, is derived as:

$$B_i = \max\left(\bigcup_t \left\{ \sum_{(j:D_j=i)} \left(\left\lceil \frac{t}{T_{period,j}} \right\rceil \right) C_j \right\} \right) . \qquad (7)$$

The workload constraint does not lead itself out particularly well to computation, since the condition should hold for all values of *t*. It is shown in [16] how to reduce the time and memory complexity of the test. The following upper bound would therefore be an improvement of our feasibility algorithm:

$$1 \le t \le BusyPeriod(s,d), \qquad (8)$$

where the *BusyPeriod* is the first *busyPeriod* in the schedule at the start of the *hyperPeriod*.

Furthermore, all values of *t* are not required to be checked, but only the integers *t*:

$$t \in \bigcup_{i=1}^{K} \{ mT_{period,i} + T_{d1,i} : m = 0,1,... \} \,, \qquad (9)$$

assuming that *K* denotes the number of real-time channels traversing the considered physical link.

If the above utilization constraint and workload constraint are met, the new logical real-time channel can be accepted. When a logical real-time channel *i* has been established, the network guarantees to deliver each real-time frame with a bounded delay:

$$T_{db,i} = T_{deadline,i} + T_{node} + T_{switch} + T_{trans} + 2T_{pro} \qquad (10)$$

where T_{pro} is the maximum propagation delay over a link between an end-node and the switch, T_{trans} is the transmission time for the considered frame. T_{switch}

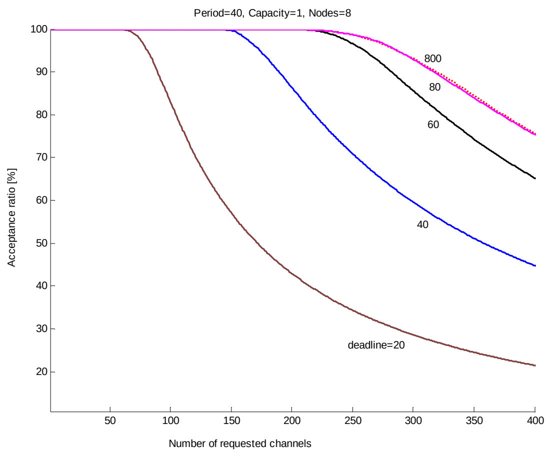

Figure 2: Acceptance ratio of real-time channels vs total number of requested real-time channels.

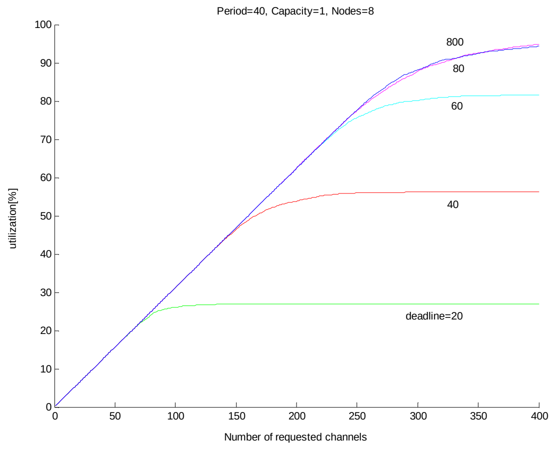

Figure 3: Utilization of real-time channels vs total number of requested real-time channels

and T_{node} is the worst case process latency for an Ethernet frame on the top of the hard real-time queue to leave the source node and to leave the switch, respectively.

4. Simulation analysis

Simulation experiments have been conducted to evaluate our method presented in this paper. In the simulation, each real-time channel is randomly generated with uniformly distributed source and destination nodes. We have simulated a network with a single 100 Mbit/s full-duplex Ethernet switch. Each real time channel is characterized by three parameters: the period, the capacity and the deadline, expressed in the form {period, capacity, deadline}.

For performance metric, we have used the real-time channel acceptance ratio and the utilization. In each simulation, logical real-time channels are added one by one and checked whether accepted or not. A number of such simulations are run to get the average utilization and the average acceptance ratio at different traffic loads, i.e., the total number of requested logical real-time channels. We have compared these performance metrics for different traffic and network characteristics.

Figure 2 shows the relation between the average acceptance ratio of real-time channels and the total number of requested real-time channels. In each simulation, the same parameters are used for all the real-time channels in the system, where the period, deadline and capacity are expressed as the number of maximum-sized Ethernet frames. The number of nodes is set to 8, the period is set to 40 and the capacity is set to 1. We observe different results when the deadline is varied between 20 and 800. When the period equals to the deadline, the acceptance ratio is 100%, until the number of requested real-time channels has passed 200. Then it is decreased slowly, because the traffic intensity is increased. Note that the acceptance ratio is lower if the deadline is shorter, because shorter deadline makes it harder to meet the workload constraint for the link.

If we replace the acceptance ratio by the average utilization of the physical channels in terms of guaranteed logical real-time channels, we get the result shown in Figure 3 (average values after 100 simulations). The utilization is increased fast when the traffic load is increased, then it becomes stable when coming close to the theoretical limits, because it is hard to accept a new logical real-time channel under the high workload in the system. The figure reveals that our system can reach a rather high utilization, for example, up to the theoretical limit of 100 % when the deadline is twice the period. Although a longer deadline achieves higher utilization, as illustrated in Figure 3, we find that there is no big utilization difference for deadlines longer than twice the period.

5. Conclusions

In this paper, we have investigated a switched Ethernet based network concept with guaranteed bit rate and delay bound for periodic real-time streams. The study covers prioritizing and the use of different scheduling policies for different traffic classes, and the feasibility analysis for hard real time traffic under the scheduling policies defined in this paper. The simulation analysis has shown the proposed scheme performs well.

References

[1] G. Kaplan, "Ethernet's winning ways," *IEEE Spectrum*, vol. 38, Jan. 2001.

[2] B.Y. Choi, S. Song, N. Birch, and J. Huang, "Probabilistic approach to switched Ethernet for real-time control applications", *Proc. of 7th International Conference on Real-time Computing Systems and Applications*, pp. 384-388, Dec. 12-14, 2000.

[3] Y. Song, "Time constrained communication over switched Ethernet", *Proc. of International Conference on Fieldbus Systems and Their Applications(IFAC'2001)*, pp. 152-169, Nov. 2001.

[4] Y. Song, A. Koubaa, and F. Simonot, "Switched Ethernet for real-time industrial communication modelling and message buffering delay evaluation", *Proc. of 4th International Workshop on Factory Communication Systems*, pp. 27-35, Aug. 2002.

[5] J. Jasperneite, P. Neumann, M. Theis, and K. Watson, "Deterministic real-time communication with switched Ethernet", *Proc. of 4th International Workshop on Factory Communication Systems*, pp. 11-18, Västerås, Sweden, Aug. 28-30, 2002.

[6] J.P. Georges, E. Rondeau, and T. Divoux, "How to be sure that switched Ethernet networks satisfy the real-time requirements of an industrial application?", *Proc. of 2002 IEEE International Symposium on Industrial Electronics*, pp. 158-163, July 8-11, 2002.

[7] S. Varadarajan and T. Chiueh, "EtheReal: a host-transparent real-time Fast Ethernet switch," *Proc. of 6th IEEE Conference on Network, Protocols*, pp. 12-21, Oct. 1998.

[8] S. Varadarajan, "Experiences with EtheReal: a fault-tolerant real-time Ethernet switch," *Proc. of 8th IEEE International Conference on Emerging technologies and Factory Automation*, pp. 183-194, vol. 1, 2001.

[9] H. Hoang, M. Jonsson, A. Larsson, R. Olsson, and C. Bergenhem, "Deadline partitioning issues and software implementation experiments," *Proc. of the 2002 International workshop on Real-time LANs in the Internet Age*, pp. 69-71, Vienna, Austria, June 2002.

[10] H. Hoang, and M. Jonsson, "Switched real-time Ethernet in industry applications," *Proc. of the 9th Asian Pacific Conference on Communication*, Penang, Malaysia, Sept. 2003.

[11] Q. Zhang, and K. G. Shin, "On the ability of establishing real-time channels in point-to-point packet-switched networks", *IEEE Transactions on Communications*, vol. 42, No. 2/3/4, pp. 1096-1105, Feb./Mar./Apr. 1994.

[12] J. Rexford, J. Hall, and K. Shin, "A router architecture for real-time communication in multicomputer networks", *IEEE Transactions on Computers*, vol. 47, no. 10, Oct. 1998.

[13] H. Zhang, "Service disciplines for guaranteed performance service in packet switching network", *Proceedings of the IEEE*, vol. 83, no. 10, Oct. 1995.

[14] IEEE Std 802.1D, *Information Technology – Telecommunications and Information Exchange between Systems-Local and Metropolitan Area Networks – Common Specification – Media Access Control (MAC) Bridges*, 1998 Edition.

[15] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in hard real-time traffic environment", *Journal of the Association for Computing Machinery*, vol. 20, no. 1, Jan. 1973.

[16] J. A. Stankovic, M. Spuri, K. Ramamritham, and G. C. Buttazzo, *Deadline Scheduling for Real-Time Systems - EDF and Related Algorithms.* Kluwer Academic Publishers, 1998.