On the efficiency of sporadic servers on Ethernet with FTT-SE

Zahid Iqbal, Luis Almeida
Instituto de Telecomunicações
Fac. Eng., University of Porto
Porto, Portugal
dee11021@fe.up.pt
lda@fe.up.pt

Mohammad Ashjaei, Moris Behnam
MRTC
Mälardalen University
Västerås, Sweden
mohammad.ashjaei@mdh.se
moris.behnam@mdh.se

ABSTRACT

Ethernet is generating a growing interest as a network for real-time embedded systems such as airplanes and automobiles. In this realm, network reservations appear as important design elements that favor composability in the time domain. One Ethernet protocol that provides such reservations is FTT-SE. In this work-in-progress we make initial steps to assess the efficiency of one particular worst-case network delay analysis for sporadic reservations associated to asynchronous messages using extensive simulation. With over 1000 message sets we found that analytic values match the observations, on average, in 31% of the messages across all generated sets whereas for 85% of the cases, the analysis upper bounds are within 2.85 times the observed values.

1. INTRODUCTION

The increasing complexity of current distributed embedded systems has risen the amount and heterogeneity of information communicated across nodes. To handle such complexity, the reservation-based design paradigm is a suitable approach. Generally, reservations must provide *temporal isolation* across different applications. This can be achieved with reservation-based scheduling, e.g., guaranteeing Q time units of network usage, at most every P units of time. Such abstraction known as *server*, or even *shaper*, enforces a limit on the impact that applications can have on each other upon integration in the network.

A networking technology that has been gaining momentum in complex distributed embedded systems is switched Ethernet, particularly provided with real-time extensions. In this work, we use Flexible Time-Triggered Switched Ethernet (FTT-SE) [3] that enforces reservations and allows any desired traffic scheduling policy. Among other features, this protocol allows reserving separate bandwidth for synchronous and asynchronous communications and, within each of these classes, it creates individual message channels with reserved bandwidth. In particular, each asynchronous channel uses a special sporadic server that fully depletes its capacity every time it is invoked and enforces a minimum message inter-transmission time.

Previously, we provided an analytic model for the worst-case response time of asynchronous messages within FTT-SE [2]. In this work we aim at assessing the efficiency of this analysis, which impacts on the network bandwidth efficiency, and how it varies with properties of the message set such as span of message periods and transmission times or understanding how pessimism spreads across messages in a given set. This knowledge can help system designers better tuning their designs to achieve high network efficiency under strict timing guarantees. We took inspiration from a similar effort developed in the past for processor scheduling, namely the Hartstone benchmark [5].

We are currently assessing such efficiency with extensive simulations using the FTT-SE master scheduler and comparing the analytic delay upper bounds with the observed maximum message response times. We focus on asynchronous messages and a single switch, which is a common configuration in many distributed embedded systems such as small vehicles. Here we extend our presentation at CRTS 2016.

2. SYSTEM MODEL AND ANALYSIS

The system consists of a single full-duplex switch in a micro-segmented architecture with N nodes. These send data to the switch via *uplinks* and receive it from the switch via *downlinks*. The protocol operates in Elementary Cycles (EC) of fixed duration (LEC) that set the timing resolution for counting intervals. The details of the FTT-SE schedule construction for different traffic classes can be found in [4]. In this work, we consider asynchronous streams, only, subject to a global asynchronous reservation of LAW every LEC. An asynchronous message stream (AS_i) is modeled using the sporadic model (Eq. 1), where C_i is its transmission time, T_i the minimum inter-arrival time and D_i the deadline. We assume that a message may include several packets, with the maximum size of $Mmax_i$. Src_i is the source node producing the stream, whereas $Dest_i$ is the destination node for the stream. The scheduling of these streams is managed by special sporadic servers that have full capacity depletion upon each invocation and enforce a minimum message inter-transmission time T_i.

$$AS_i = (C_i, T_i, D_i, Mmax_i, Src_i, Dest_i) \tag{1}$$

FTT-SE uses online scheduling ensuring that all scheduled traffic for each EC (the EC-schedule) can be delivered to the destination in that EC, while respecting the associated reservations. This prevents backlogs in the switch queues from one EC to the next.

We determine message response time (RT_i) upper bounds with the analysis presented in [2], where $RT_i = \min(x^l)$: $x^l = x^{l-1}$, $x^l \leq D_i$ and x^l is computed as follows, with initial value $x^0 = C_i'$:

$$x^l = C_i' + \left(\frac{SFD(m_i) + SLD}{\alpha}\right) + \sum_{\forall m_j, Src_j = Src_i} \left\lceil \frac{x^{l-1}}{T_j} \right\rceil C_j' + \sum_{\forall m_k, Dest_k = Dest_i} \left\lceil \frac{x^{l-1}}{T_k} \right\rceil (C_k' + \left(\frac{SFD(m_k) + SLD}{\alpha}\right))$$

where $SFD(m_i)$ is the switch store and forward delay equals to the maximum packet among the packets composing m_i, i.e., $SFD(m_i) = Mmax_i$. The message transmission times are adapted to $C_i' = C_i/\alpha$ to account for *inserted idle time* (upper bound by iit), with $\alpha = (LAW - iit)/LEC$. Note that iit is calculated as the maximum packet size among all messages.

3. EXPERIMENTS

In the experiments presented in this section we target at comparing the analytic upper bounds (RT) with the observed maximum response times (RT_o) with random message sets. A preliminary comparison was already shown in [2]. Here we confirm those results with larger sets that lead to smoother distributions as seen below.

3.1 Analysis results vs. observation

We consider a system with 10 slave nodes that send a maximum of 5 messages each, $LEC = 2000\mu s$, $LAW = 1049\mu s$ and an MTU of $128\mu s$. The inter-arrival periods are considered constant for maximal load and chosen randomly from the set $\{5, 7, 13, 19, 37\}$ and each set is simulated for 3000 ECs. The message set is generated considering each link scheduled independently and using Rate-Monotonic (RM) scheduling. The target utilization U per link is computed using RM's least utilization upper bound reduced by a factor accounting for the asynchronous reservation and inserted idle time in each EC computed as $U^{RM} \times (LAW - iit)/LEC$. To account for the variability in message arrivals, downlink utilization is further reduced by 70%, thus avoiding conditions near the schedulability threshold. Then, we use the UUniFast algorithm [1] to distribute the target bandwidth among the messages in each link. With these constraints, we generated 1000 random message sets that exhibited an average of 33 messages per data set.

Figure 1 shows the percentage of cases in which the analysis results match the observations, i.e., $RT = RT_o$. We can see that such match occurs, on average, in 31% of the messages of the generated sets. The mode for this histogram is given with 224 message sets reporting matches in $30\% - 36\%$ messages of their respective sets. The median is at $25\% - 30\%$ approximating the midpoint of the histogram i.e., 488 message sets exhibiting distribution between $3\% - 30\%$ and approximately a similar number of sets with distribution varying between $30\% - 58\%$.

We also quantify how much the analytic response time upper bounds exceed the values observed in the scheduler execution. For each message set, we note that the case in which the analytic bound (RT) differs the most from the observed values (RT_o). The results are plotted in Figure 2. We can see that for 85% of the cases, the analysis reports upper bounds that are up to 2.85 times (185%) greater than the observed values.

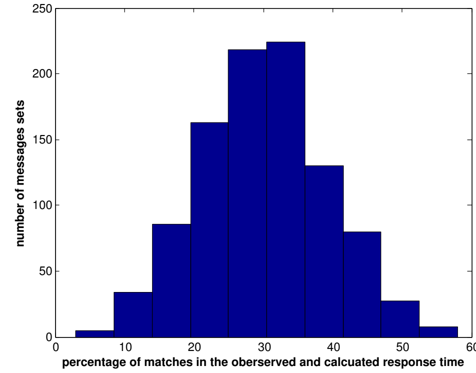

Figure 1: Percentage of matches between calculated results and observed values

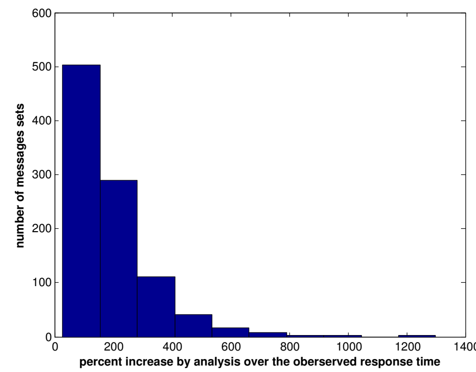

Figure 2: Percentage increase in RT with analysis method over the observed values

3.2 Impact of activation pattern

We also study the impact of message activation pattern on schedulability. For this case, we generate 1200 message sets with their minimum inter-arrival periods taken randomly from the set $\{4, 8, 16, 32, 64, 128\}$ and we trigger each message set periodically (with no offsets) as well as sporadically. With sporadic activation, m_i's next activation is triggered at an EC randomly chosen in the interval $(T_i, 2T_i - 1)$ from its previous activation. In each case, FTT-SE master scheduler runs 20000 elementary cycles. We measure message maximum response time (RT_o) from this simulation. Then, for each message, we take the difference between RT_o values reported with its periodic and sporadic activation. For the message set, we count the number of messages that reported larger RT_o with each activation pattern, and the number of messages that experience the same RT_o. Our observations show that none activation pattern favors the other significantly. Results indicate that, on average, the three cases are evenly distributed each being 33% (Figure 3).

Interestingly, we would expect RT_o for periodic activations to be larger given their higher load. Note that the average inter-arrival periods of the sporadic case are $1.5 * T_i$. Moreover, releasing the periodic messages simultaneously at $EC = 0$ leads to a well defined worst-case busy interval easy to capture in the observations.

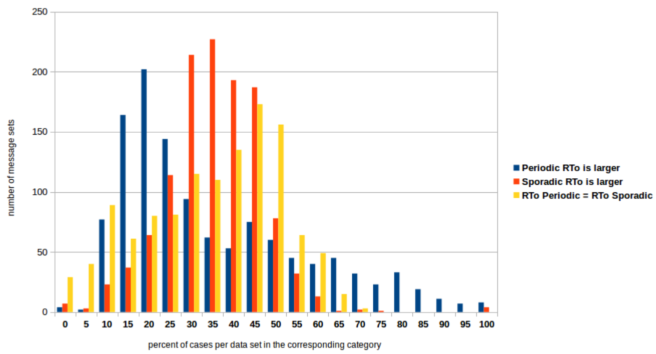

Figure 3: Percentage of cases per data set for each category

Yet, a significant number of sets with sporadic activations present RT_o larger than or equal to that of the periodic case. In our system, however, with multiple links, simultaneous message transmissions do not interfere if they do not share links. This effect softens the difference between both cases.

To illustrate one case in which a given message triggered sporadically exhibits a larger RT_o than when triggered periodically, we provide the following example. We choose message m_{43} from one data set that reports $RT_o = 14$ with periodic activation and $RT_o = 31$ when messages in the set are activated sporadically. Table 3.2 lists the messages that share links with m_{43}.

Description	Message(s)	T_i	Src_i	$Dest_i$
Message under study	m_{43}	128	9	2
Uplink interference	m_{41}	128	9	6
	m_{42}	32	9	8
	m_{45}	16	9	8
Downlink interference	m_{25}	8	5	2
	m_{19}	128	4	2
	m_{28}	4	6	2
	m_{35}	64	7	2
	m_{37}	16	8	2
	m_{48}	128	10	2
	m_{50}	8	10	2

Table 1: Interference set of message m_{43}

We analyzed the FTT-SE scheduler simulation traces to observe the ready queue state during the time between activation and dispatch of m_{43} in each case. Figure 4 shows these traces displaying the messages in Table 3.2, only, for the sake of clarity, and highlighting new messages activations. We observe m_{43} worst case response time that occurred after activation in EC_{4107} and dispatch in EC_{4138}. For the periodic case, RT_o occurs after activation in EC_0 and dispatch in EC_{14}. This is when all messages are released simultaneously in EC_0. However, the worst-case pattern is a complex combination of interferences in the uplinks and downlinks. In this case, the high initial backlog triggered by the periodic release generates the maximum interference in the uplinks but not in the downlinks. Conversely, sporadic releases, given their random nature, tend to be more efficient in finding pernicious interference patterns in the downlinks.

EC_{4107}: $m_{25} - m_{42} - m_{41} - m_{43}$
EC_{4108}: $m_{42} - m_{41} - m_{43}$
EC_{4109}: $m_{50} - m_{42} - m_{35} - m_{41} - m_{43}$
EC_{4110}: $m_{35} - m_{41} - m_{43}$
EC_{4111}: $m_{28} - m_{35} - m_{41} - m_{43}$
EC_{4112}: $m_{35} - m_{41} - m_{43}$
EC_{4113}: $m_{35} - m_{41} - m_{43}$
EC_{4114}: $m_{35} - m_{41} - m_{43}$
EC_{4115}: $m_{45} - m_{41} - m_{43}$
EC_{4116}: $m_{25} - m_{45} - m_{41} - m_{43}$
EC_{4117}: $m_{28} - m_{45} - m_{41} - m_{43}$
EC_{4118}: $m_{45} - m_{41} - m_{43}$
EC_{4119}: $m_{45} - m_{41} - m_{43}$
EC_{4120}: $m_{41} - m_{43}$
EC_{4121}: $m_{41} - m_{43}$
EC_{4122}: $m_{41} - m_{43}$
EC_{4123}: $m_{50} - m_{41} - m_{43}$
EC_{4124}: $m_{28} - m_{41} - m_{43}$
EC_{4125}: $m_{37} - m_{41} - m_{43}$
EC_{4126}: $m_{37} - m_{41} - m_{43}$
EC_{4127}: $m_{25} - m_{41} - m_{43}$
EC_{4128}: $m_{28} - m_{41} - m_{43}$
EC_{4129}: $m_{41} - m_{43}$
EC_{4130}: $m_{41} - m_{43}$
EC_{4131}: $m_{41} - m_{43}$
EC_{4132}: $m_{45} - m_{43}$
EC_{4133}: $m_{45} - m_{43}$
EC_{4134}: $m_{28} - m_{50} - m_{45} - m_{43}$
EC_{4135}: $m_{45} - m_{43}$
EC_{4136}: $m_{25} - m_{45} - m_{43}$
EC_{4137}: m_{43}

EC_0: $m_{28} - m_{50} - m_{25} - m_{45} - m_{37} - m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_1: $m_{25} - m_{45} - m_{37} - m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_2: $m_{45} - m_{37} - m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_3: $m_{45} - m_{37} - m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_4: $m_{28} - m_{45} - m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_5: $m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_6: $m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_7: $m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_8: $m_{28} - m_{50} - m_{25} - m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_9: $m_{25} - m_{42} - m_{35} - m_{43} - m_{41} - m_{19}$
EC_{10}: $m_{43} - m_{41} - m_{19}$
EC_{11}: $m_{43} - m_{41} - m_{19}$
EC_{12}: $m_{28} - m_{43} - m_{41} - m_{19}$
EC_{13}: $m_{43} - m_{41} - m_{19}$

Figure 4: Reduced simulation trace for message m_{43}: sporadic (left) versus periodic release (right)

4. ONGOING WORK

This work fits in the context of reservation-based design of complex embedded systems using Ethernet. We addressed the network reservations and gave a first step towards assessing the efficiency of a previously proposed response-time analysis. With random message sets this analysis may produce singular upper bounds that are several times the maximum observed values, though 85% are below a factor of 2.85. On the other hand, the upper bounds matched the observed values in 31% the messages across all sets. We aim at studying how these results vary with high level properties of the message sets to provide guidelines for efficient designs.

5. REFERENCES

[1] E. Bini and G. C. Buttazzo. Measuring the performance of schedulability tests. *Real-Time Syst.*, 30(1-2):129–154, May 2005.

[2] Z. Iqbal and L. Almeida. Towards an analysis for hierarchies of sporadic servers on ethernet. In *2016 11th IEEE Symposium on Industrial Embedded Systems (SIES)*, pages 1–6. IEEE, 2016.

[3] R. Marau, L. Almeida, and P. Pedreiras. Enhancing real-time communication over COTS Ethernet switches. In *6th IEEE International Workshop on Factory Communication Systems*, pages 295–302, June 2006.

[4] R. Marau, P. Pedreiras, and L. Almeida. Signaling asynchronous traffic over a Master-Slave Switched Ethernet protocol. In *Proc. on the 6th Int. Workshop on Real Time Networks (RTN'07)*, Pisa, Italy, 2 July 2007.

[5] N. H. Weiderman and N. I. Kamenoff. Hartstone uniprocessor benchmark: Definitions and experiments for real-time systems. *Real-Time Systems*, 4(4):353–382, 1992.