A Longest-Path Problem for Evaluating the Worst-Case Packet Delay of Switched Ethernet

Klaus Schmidt
Department of Electronic and Communication Engineering
Cankaya University, Ankara, Turkey
Email: schmidt@cankaya.edu.tr

Ece G. Schmidt
Department of Electrical and Electronics Engineering
Middle East Technical University, Ankara, Turkey
Email: eguran@metu.edu.tr

***Abstract*—In the recent years, the use of real-time *Ethernet* protocols becomes more and more relevant for time-critical networked industrial applications. In this context, this paper presents a method to compute the worst-case packet delays on *switched Ethernet*. Based on an evaluation of the packet delays at each switch port and the network topology, we construct a weighted directed graph that allows to find the worst-case end-to-end packet delay by solving a conventional longest-path problem.**

I. Introduction

In the recent years, there is an increased effort to employ the low-cost and high-speed Ethernet technology as the communication network in industrial automation [1], [2], [3] and various *real-time Ethernet* (RTE) protocols were standardized. In this paper, we study the transmission of time-critical *data packets* that are transmitted on full-duplex switched RTE solutions such as *Ethernet/IP* (EIP) [4] that employ customary switches with packet prioritization (e.g., [5]). With EIP, real-time (RT) packets obtain the highest priority and are hence not queued behind lower-priority packets. However, since there is no further RT support, protocols such as EIP require a detailed analysis of the packet delays in order to ensure RT behavior.

In this paper, we develop a novel method for the computation of worst-case packet delays on switched Ethernet networks such as EIP. We first propose a traffic characterization that is suitable for industrial automation systems: we assume that a certain maximum number of packets can be present in the network for each controller node at any time. Then, we recursively evaluate the maximum sizes of the switch packet queues to compute the maximum packet delay at each switch port. Using these packet delays and the network topology, we construct a weighted graph whose longest path corresponds to the worst-case packet delay in the overall network.

Related work employs the *network calculus* (NC) [6], [7] or chooses an explicit characterization of the transmitted packets [8], [9]. In the first case, the use of approximate models of the packet traffic flow leads to conservative results as shown in [8]. In the latter case, periodicity of the packets is required in [8], while [9] only allows one packet per controller node. An overview of various analysis techniques is provided in [10].

The organization of the paper is as follows. In Section II, we introduce the formal description of the network components. Our packet delay computation method is developed in Section III, and conclusions are given in Section IV.

II. Switched Ethernet Architecture

We study networks that consist of multiple switches in a tree topology analogous to [9]. Fig. 1 shows an example topology. The switches are interconnected with full-duplex connections and the queues at the switch output ports are served in a first-in first-out (FIFO) manner.

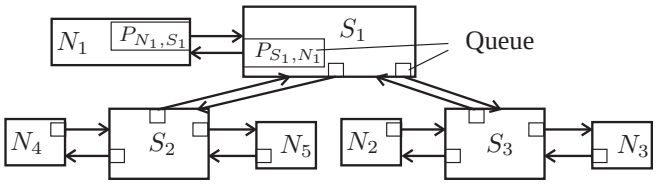

Fig. 1. Example topology.

The following notation is introduced to allow a concise characterization of the network. It consists of a *set of switches* $\mathcal{S}$ and a *set of nodes* $\mathcal{N}$, where both switches and nodes are referred to as network *units*. The switch topology is captured by a tree representation $T_{\mathcal{S}} = (\mathcal{S}, S_{\mathrm{R}}, c_{\mathcal{S}}, p_{\mathcal{S}})$: each switch $S \in \mathcal{S}$ constitutes a *vertex* of the tree; S_{R} is the *root vertex*; $c_{\mathcal{S}} : \mathcal{S} \to 2^{\mathcal{S}}$ is the *children map*; $p_{\mathcal{S}} : \mathcal{S} \to \mathcal{S}$ is the *parent map*. Here, $2^{\mathcal{S}}$ denotes the power set of $\mathcal{S}$, and we will write $|\mathcal{S}|$ for the number of elements of $\mathcal{S}$. Hence, S_{R} is the highest-level switch in a hierarchy with multiple levels, and for a switch $S \in \mathcal{S}$, $c_{\mathcal{S}}(S)$ represents the set of children, and $p_{\mathcal{S}}(S)$ represents the parent of S in that hierarchy. A switch that does not have any children is called a *leaf*. In the example in Fig. 1, we have $\mathcal{N} = \{N_1, N_2, N_3, N_4, N_5\}$, and $\mathcal{S} = \{S_1, S_2, S_3\}$. The root of the tree representation $T_{\mathcal{S}}$ is S_1. In addition, it holds for instance that $c_{\mathcal{S}}(S_1) = \{S_2, S_3\}$ and $p_{\mathcal{S}}(S_3) = S_1$.

Furthermore, the map $n_{\mathcal{S}} : \mathcal{S} \to 2^{\mathcal{N}}$ characterizes the set of nodes $n_{\mathcal{S}}(S)$ that are connected to $S \in \mathcal{S}$. For example, $n_{\mathcal{S}}(S_3) = \{N_2, N_3\}$ in Fig. 1. Finally, we introduce the map $u_{\mathcal{S}} : \mathcal{S} \to 2^{\mathcal{S} \cup \mathcal{N}}$. It represents the set of units $u_{\mathcal{S}}(S) := n_{\mathcal{S}}(S) \cup c_{\mathcal{S}}(S) \cup \{p_{\mathcal{S}}(S)\}$ connected to each switch $S \in \mathcal{S}$.

Regarding the packet transmission, an *output port* $P_{X,Y}$ is introduced for each existing connection between network units, where X denotes the sending unit and Y represents the receiving unit as shown for N_1 and S_1 in Fig. 1. We denote the *set of output ports* of each unit $U \in \mathcal{N} \cup \mathcal{S}$ as $\mathcal{P}_U$, and the overall set of output ports as $\mathcal{P} := \bigcup_{U \in \mathcal{N} \cup \mathcal{S}} \mathcal{P}_U$.

In this paper, we employ switches that support packet prioritization as for example defined by the IEEE 802.3p standard [11]. Such switches are standard in networking applications and are also common in industrial communication [5]. Hence, each switch port comprises FIFO queues in order to store and forward the incoming packets for each priority level. In this work, we restrict our attention to packets of RT applications that are transmitted with the highest priority. Accordingly, we only explicitly describe the queue for the highest-priority packets at each port $P_{X,Y} \in \mathcal{P}$ by its *maximum queue length* $Q_{X,Y}$.[1] Furthermore, we introduce the fixed delay D_{LP} that characterizes the maximum delay introduced by a lower priority packet that is currently transmitted and cannot be preempted by an incoming higher-priority packet. Similarly, we denote the maximum number of packets queued at the output port $P_{N,S}$ of a node $N \in \mathcal{N}$ as $Q_{N,S}$ assuming the same packet prioritization as in the switches.

III. Worst Case Delay Computation

A. Assumptions

Our worst-case end-to-end delay computation is performed for RT traffic. Following the observations in [2], [12], RT traffic involves both periodically and sporadically generated packets with usually small packet sizes. Consequently, we assume that the high-priority packets under investigation are transmitted in minimum size Ethernet frames with 576 bits.[2]

Moreover, we propose a suitable traffic characterization for industrial communications. We capture the packet generation properties by requiring that a certain maximum number of generated packets per node can be present in the network at any time. Precisely, this means that, after an initial transmission phase, each node can only transmit a new packet if a previously transmitted packet leaves the network. In particular, this traffic model covers the sporadic and periodic packets that are relevant for industrial automation systems.

Sporadic RT packets commonly allow the information exchange among different network nodes about event occurrences. They normally trigger a response of the destination node such as an acknowledgment or information about new event occurrences. Thus, new packets are only sent after the response packet has been processed. Each periodic RT packet is transmitted by some node $N \in \mathcal{N}$ with a given period p_N. If p_N is larger than the maximum packet delay D^* that is computed by our approach, then the proposed traffic model is appropriate. More importantly, if our approach yields that $p_N < D^*$, then our computation can be reevaluated assuming that at least $\lceil D^*/p_N \rceil$ packets instead of one packet are transmitted by the node N ($\lceil \bullet \rceil$ denotes the ceiling operation).

Hence, it is ensured in both cases that the inter-transmission time of packets that are considered for the maximum delay computation is larger than the maximum packet delay.

As the maximum number of packets per node N_i in the network at any time also determines the maximum number of packets from N_i that can be present at any output port P_{N_i,S_k} at any time, the notational convention C_{N_i,S_k} for this *maximum packet count* is chosen. Finally, in this work-in-progress, we assume that no information about each packet destination is available, i.e., any message is broadcasted to all destination nodes.

Remark 1: Note that the packet generation assumed in [9] conforms with the restrictive case of $C_{N_i,S_k} = 1$ for each $N_i \in \mathcal{N}$ with the connected switch S_k in our framework.

B. Computation of Maximum Packet Counts

Based on the given maximum packet counts for all network nodes, the corresponding maximum packet counts $C_{S_k,U}$ at each switch port $P_{S_k,U}$ can be evaluated ($S_k \in \mathcal{S}$, $U \in u_{\mathcal{S}}(S)$). Here $C_{S_k,U}$ describes the maximum number of packets that can be present in the network (queued or in transit) that go through $P_{S_k,U}$. The values of $C_{S_k,U}$ are necessary to perform the delay computation as elaborated in Section III-C and III-D.

At each port $P_{S_k,U}$, the value of $C_{S_k,U}$ is the sum of the maximum packet counts that can be switched to that port at the same time.

$$C_{S_k,U} = \sum\nolimits_{Y \in u_{\mathcal{S}}(S_k)-\{U\}} C_{Y,S_k}. \tag{1}$$

An iterative evaluation of (1) starting from leaf switches in $T_{\mathcal{S}}$, where all addends in the right hand sum of (1) are known allows to compute all maximum packet counts.

For the example in Fig. 1, and according to the assumptions in Section III-A, the values of $C_{N_1,S_1} = 6$, $C_{N_2,S_3} = 5$, $C_{N_3,S_3} = 3$, $C_{N_4,S_2} = 4$, and $C_{N_5,S_2} = 2$ have been chosen for the 5 network nodes. The resulting values of the remaining maximum message counts are listed in Table I.

TABLE I
Maximum Packet Counts for the Network Switches

C_{S_2,S_1}	C_{S_1,S_3}	C_{S_3,N_2}	C_{S_3,N_3}	C_{S_3,S_1}	C_{S_1,N_1}	C_{S_1,S_2}	C_{S_2,N_4}	C_{S_2,N_5}
6	12	15	17	8	14	14	16	18

C. Computation of Maximum Queue Lengths

After establishing the maximum packet counts in the previous section, it is now possible to compute the *maximum queue size* $Q_{U,V}$ for each output port $P_{U,V}$, i.e., the maximum number of packets queued at that port, where $U, V \in \mathcal{N} \cup \mathcal{S}$.

First, the output ports of the network nodes are considered. Let $N_i \in \mathcal{N}$ with the output port P_{N_i,S_k}, $N_i \in n_{\mathcal{S}}(S_k)$. Then, $Q_{N_i,S_k} = C_{N_i,S_k}$ as stated in Section III-A. For the example configuration in Fig. 1, this means that $Q_{N_1,S_1} = C_{N_1,S_1} = 6$, $Q_{N_2,S_3} = 5$, $Q_{N_3,S_3} = 3$, $Q_{N_4,S_2} = 4$, and $Q_{N_5,S_2} = 2$.

Next, we compute the queue sizes of each switch output port $P_{S_k,U}$, $S_k \in \mathcal{S}$, $U \in u_{\mathcal{S}}(S_k)$. The packets directed to $P_{S_k,U}$ originate from the remaining connected ports P_{Y,S_k} with $Y \in u_{\mathcal{S}}(S_k) - \{U\}$. Denoting the maximum packet count of nodes connected to S_k as $C^{\max}_{S_k,U} = \max_{Y \in u(S_k)-\{U\}} C_{Y,S_k}$, the maximum queue size $Q_{S_k,U}$ evaluates to

$$Q_{S_k,U} := C_{S_k,U} - C^{\max}_{S_k,U} + 1. \tag{2}$$

[1] The evaluation of $Q_{X,Y}$ is the subject of Section III-C.

[2] Note that this assumption complies with the study in [9].

Equation (2) is described in the timing diagram in Fig. 2, which refers to the nodes N_4 and N_5 and the switch S_2 in the example topology in Fig. 1. In order to study the worst case, we assume that, initially, the maximum number of 4 (A1,...,A4) and 2 (B1,B2) packets are present at N_4 and N_5, respectively. After a *processing delay* D_{N}, both nodes can send their first packets A1 and B1 to the switch S_2, where they are queued in the output port P_{S_2,S_1} after experiencing the *framing delay* D_{F} and the *propagation delay* D_{P}. After that, the packets stored at the switch port are forwarded one after another, while new packets are coming in from the nodes N_4 and N_5. Note that after the transmission of a packet, each unit has to wait for the *interframe delay* D_{I} until a new packet can be sent. The last packet that leaves P_{S_2,S_1}, i.e., the packet with the maximum delay, is A4 from N_4. In compliance with (2), it holds that although 3 packets are queued in front of A4 at N_4, the maximum number of packets queued in front of A4 at P_{S_2,S_1} is $Q_{S_2,S_1} - 1 = C_{S_2,S_1} - C^{\max}_{S_2,S_1} = 2$. This is due to the fact that the C_{S_2,S_1} packets that are switched to the output port P_{S_2,S_1} cannot be present at the same time: while the C_{S_2,S_1} packets arrive piece by piece from the connected nodes, at least $C^{\max}_{S_2,S_1}$ already queued packets leave P_{S_2,S_1}.

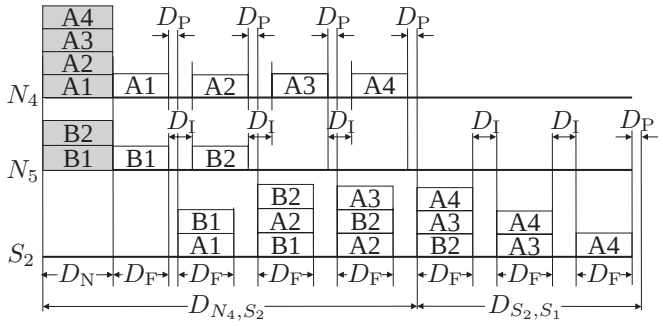

Fig. 2. Timing diagram for packet queueing.

The maximum queue sizes of all switch ports are determined by an iterative evaluation of (2) starting from the leaf switches. Table II summarizes the results for our example topology.

TABLE II
MAXIMUM QUEUE SIZES FOR THE SWITCH PORTS

Q_{S_2,S_1}	Q_{S_1,S_3}	Q_{S_3,N_2}	Q_{S_3,N_3}	Q_{S_3,S_1}	Q_{S_1,N_1}	Q_{S_1,S_2}	Q_{S_2,N_4}	Q_{S_2,N_5}
3	7	4	6	4	7	7	3	5

D. Components of the End-to-End Delay

Having determined the maximum queue sizes of all output ports in the network, this information can now be used to derive the resulting communication delays. In this respect, three different situations can be distinguished: 1) *node-to-switch* connections, 2) *switch-to-node* connections and 3) *switch-to-switch* connections.

1) Node-to-Switch Connection: For each node N_i with its connected switch S_l, the value of Q_{N_i,S_l} is known, i.e., in the worst case, a packet has to wait for $Q_{N_i,S_l} - 1$ packet transmissions until it can leave the node. Let D_{N_i,S_l} denote the maximum time from the packet generation at N_i to its arrival at the tail of its output queue to S_l. Then, D_{N_i,S_l} comprises the *processing delay* D_{N} at the node, the *frame transmission delay* D_{F}, the *propagation delay* D_{P} and the delay that is potentially caused by the longest lower-priority packet D_{LP}. Furthermore, the *queueing delay* $(Q_{N_i,S_l} - 1) \cdot (D_{\mathrm{F}} + D_{\mathrm{I}})$ for $Q_{N_i,S_l} - 1$ packet transmissions including the *interframe delay* D_{I} between consecutive packets contributes to the packet delay. The maximum time from the packet generation at N_i to its arrival at the tail of P_{S_k,S_l} of a switch S_k is

$$D_{N_i,S_l} = D_{\mathrm{N}} + (Q_{N_i,S_l} - 1)(D_{\mathrm{F}} + D_{\mathrm{I}}) + D_{\mathrm{F}} + D_{\mathrm{P}} + D_{\mathrm{LP}}. \quad (3)$$

For example, it holds that $D_{N_4,S_2} = D_{\mathrm{N}} + 3(D_{\mathrm{F}} + D_{\mathrm{I}}) + D_{\mathrm{P}}$ as can be seen in Fig. 2. Table III shows these values together with the delay parameters for node-to-switch connections, where realistic values for D_{F}, D_{P}, D_{I} and D_{N} are taken from [9] for the use of 10-BASE-T Ethernet.

2) Switch-to-Node Connection: The case, where a switch $S_k \in \mathcal{S}$ transmits to a node $N_i \in \mathcal{N}$ is considered. Here, the same components as in (3) are relevant.

3) Switch-to-Switch Connection: If a switch S_k transmits to a switch S_l, then the same components as in Section III-D1, except for the processing delay at the sender contribute to the packet delay. Thus, the maximum delay between the output port P_{S_k,S_l} and the switch S_l is

$$D_{S_k,S_l} := (Q_{S_k,S_l} - 1) \cdot (D_{\mathrm{F}} + D_{\mathrm{I}}) + D_{\mathrm{F}} + D_{\mathrm{P}} + D_{\mathrm{LP}}. \quad (4)$$

Fig. 2 illustrates the computation of $D_{S_2,S_1} = 2(D_{\mathrm{F}} + D_{\mathrm{I}}) + D_{\mathrm{F}} + D_{\mathrm{P}}$. Table III depicts the delay parameters for all switch-to-switch connections of the example topology in Fig. 1.

TABLE III
MAXIMUM PACKET DELAYS

D_{F}	D_{P}	D_{I}	D_{N}	D_{N_1,S_1}	D_{N_2,S_3}
57.6μs	0.1μs	9.6μs	42.3μs	436μs	368.8μs
D_{N_3,S_3}	D_{N_4,S_2}	D_{N_5,S_2}	D_{S_1,N_1}	D_{S_2,N_4}	D_{S_2,N_5}
234.4μs	301.6μs	167.2μs	503.2μs	234.4μs	368.8μs
D_{S_3,N_2}	D_{S_3,N_3}	D_{S_1,S_2}	D_{S_1,S_3}	D_{S_2,S_1}	D_{S_3,S_1}
301.6μs	436μs	460.9μs	460.9μs	192.1μs	259.3μs

Based on the above results, the maximum delay of a packet on an end-to-end path from any sender node to any destination node can be determined by adding up the delays for the respective connections. For example, the path from N_2 to N_5 in Fig. 1 can be divided into the node-to-switch connection N_2 to S_3, the switch-to-switch connections from S_3 to S_1 and S_1 to S_2, and the switch-to-node connection from S_2 to N_5. Then, the maximum delay for a packet from N_2 to N_5 is $D_{N_2,S_3} + D_{S_3,S_1} + D_{S_1,S_2} + D_{S_2,N_5} = 368.8\,\mu\mathrm{s} + 259.3\,\mu\mathrm{s} + 460.9\,\mu\mathrm{s} + 368.8\,\mu\mathrm{s} = 1.4578\,\mathrm{ms}$.

If this computation is carried out for all possible paths in the network, then the maximum packet delay in the network can be computed. However, observing that all packets that share a sub-path also share the corresponding part of the potential maximum delay, we propose a more efficient solution to the maximization problem in the next section.

E. Longest Path Problem Formulation

We model the delay computation by a *weighted directed graph* $G = (\mathcal{V}, \mathcal{E}, w)$. Here, the *set of vertices* $\mathcal{V}$ contains all nodes in $\mathcal{N}$ and all ports $P_{X,Y} \in \mathcal{P}$, i.e., $\mathcal{V} = \mathcal{N} \cup \mathcal{P}$. Semantically, each vertex $P_{X,Y} \in \mathcal{V}$ is associated with a packet at the tail of the respective queue, and each vertex $N_i \in \mathcal{V}$ corresponds to the arrival of a packet at that destination node. The *set of edges* $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ characterizes the connections among the units with the following conditions.

$$P_{X,N_i} \in \mathcal{P} \text{ and } N_i \in \mathcal{N} \Leftrightarrow (P_{X,N_i}, N_i) \in \mathcal{E},$$
$$P_{Y,S_k}, P_{S_k,S_l} \in \mathcal{P} \text{ and } k \neq l \Leftrightarrow (P_{Y,S_k}, P_{S_k,S_l}) \in \mathcal{E},$$

That is, there is a directed edge for each port that is connected to a destination node and there is a directed edge from each port that transmits packets to the switch S_k to each outgoing port of S_k (except for the port returning to the sender unit Y).

Finally, each edge $e = (P_{X,Y}, V) \in \mathcal{E}$ with $X, Y \in \mathcal{N} \cup \mathcal{S}$ and $V \in \mathcal{V}$ gets the *weight* $w(e) := D_{X,Y} \in \mathbb{R}^+$, that is computed according to (3) - (4). That is, $w(e)$ represents the maximum packet delay from arriving at $P_{X,Y}$ to being sent to the destination unit Y. Note that $w(e)$ is always positive and all edges $(P_{X,Y}, V)$ have the same weight $D_{X,Y}$.

The weighted directed graph for the example topology in Fig. 1 is depicted in Fig. 3. Circles represent vertices and arrows indicate the direction of the edges that are labeled by their delay values. For instance, the edge from P_{S_2,S_1} to P_{S_1,S_3} has the weight $w((P_{S_2,S_1}, P_{S_1,S_3})) = D_{S_2,S_1} = 192.1\mu$s.

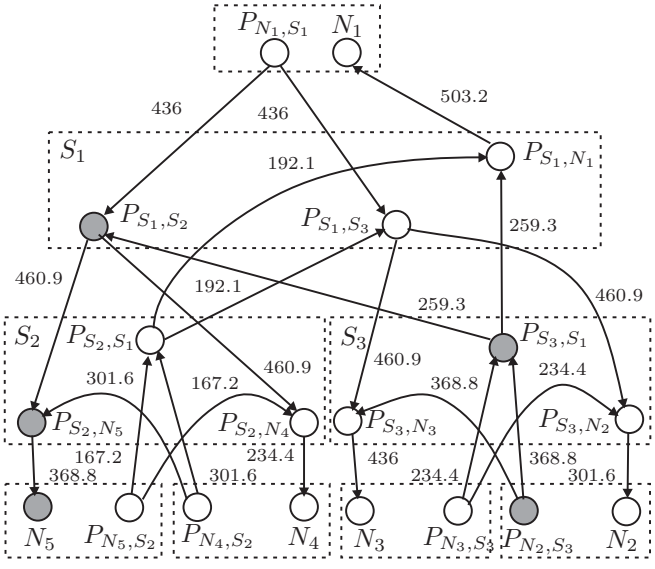

Fig. 3. Example Graph

It holds that the end-to-end delays are experienced on paths from sender ports P_{N_i,S_k}, $N_i \in \mathcal{N}$, $S_k \in \mathcal{S}$ to destination nodes $N_j \in \mathcal{N}$, $i \neq j$. Let $\mathcal{V}_\mathrm{S} := \{P_{X,Y} \in \mathcal{P} | X \in \mathcal{N}\} \subseteq \mathcal{V}$ be the set of *sender vertices* and $\mathcal{V}_\mathrm{D} := \mathcal{N} \subseteq \mathcal{V}$ be the set of *destination vertices* in G. We define the set of all possible *end-to-end paths* in the network as

$$\Pi := \{p | p = v_1 v_2 \cdots v_a \text{ is a path in } G, a \in \mathbb{N} \text{ and } v_1 \in \mathcal{V}_\mathrm{S}, v_a \in \mathcal{V}_\mathrm{D}\}.$$

Then, the maximum packet delay in the network D^* is equal to the length of the longest path in G.

$$D^* = \max_{p \in \Pi} \sum_{i=1}^{a-1} w((v_i, v_{i+1})). \quad (5)$$

Since (5) constitutes a standard *longest path problem* it can be efficiently solved by *dynamic programming* [13].

For the example topology in Fig. 1 with the packet generation assumption in Section III-A, the largest packet delay of $D^* = 1.4578\,\mathrm{ms}$ occurs on the path from N_2 to N_5 (the associated vertices in Fig. 3 are shaded in gray).

IV. Conclusion

In this paper, we propose a method for the worst-case end-to-end packet delay computation for real-time control packets on switched industrial Ethernet networks. Our approach is based on an explicit characterization of the packets transmitted on the network in the sense that the exact numbers of packets that can be generated by each node at a time is known. Based on this assumption, we develop a recursive procedure in order to evaluate the maximum lengths of the packet queues in all switches. Using these queue lengths, we construct a weighted directed graph that captures the packet delays and allows to find the worst-case packet delay as the solution of a longest-path problem. As an extension of this work-in-progress study, we are currently generalizing the framework to the practical case of multicast switching and validating the maximum delay bounds by means of a manufacturing system example.

References

[1] M. Felser, "Real-time Ethernet - industry prospective," *Proceedings of the IEEE*, vol. 93, pp. 1118– 1129, 2005.

[2] P. Neumann, "Communication in industrial automation - what is going on?" *Control Engineering Practice*, vol. 15, pp. 1332–1347, 2007.

[3] J.-D. Decotignie, "The many faces of industrial Ethernet," *IEEE Industrial Electronics Magazine*, vol. 3, no. 1, pp. 8–19, 2009.

[4] "Real-time Ethernet: Ethernet/IP with time synchronization: Proposal for a publicly available specification for real-time Ethernet," Doc. IEC 65C/361/NP, 2004.

[5] (2008) Scalance x-400 product line: The modular high-performance switches. [Online]. Available: http://www.automation.siemens.com/net/html_76/produkte/scalance/040_scalance_x_400.htm

[6] J. Jasperneite, P. Neumann, M. Theis, and K. Watson, "Deterministic real-time communication with switched ethernet," in *Factory Communication Systems, IEEE International Workshop on*, 2002, pp. 11–18.

[7] J.-P. Georges, T. Divoux, and E. Rondeau, "Confronting the performances of a switched Ethernet network with industrial constraints by using the network calculus," *Int. J. Commun. Syst.*, vol. 18, no. 9, pp. 877–903, 2005.

[8] X. Fan, M. Jonsson, and J. Jonsson, "Guaranteed real-time communication in packet-switched networks with FCFS queuing," *Comput. Netw.*, vol. 53, no. 3, pp. 400–417, 2009.

[9] K. C. Lee, S. L. Lee, and M. H. Lee, "Worst case communication delay of real-time industrial switched Ethernet with multiple levels," *IEEE Trans. on Industrial Electronics*, vol. 53, no. 5, pp. 1669–1676, 2006.

[10] S. Perathoner, E. Wandeler, L. Thiele, A. Hamann, S. Schliecker, R. Henia, R. Racu, R. Ernst, and M. G. Harbour, "Influence of different system abstractions on the performance analysis of distributed real-time systems," in *Proc. ACM Conference on Embedded Software*, 2007.

[11] "IEEE 802.3p-1993 supplement to 802.3 - layer management for 10 mb/s baseband medium attachment units (maus) (section 20)," Published with 802.3q-1993. Institute of Electrical and Electronics Engineers, 1993.

[12] J. Moyne and D. Tilbury, "The emergence of industrial control networks for manufacturing control, diagnostics, and safety data," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 29–47, Jan. 2007.

[13] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. 1*. Athena Scientific, 3rd Edition, 2005.