Supplementary Material for FooDog

This supplementary material covers commercial Time-Sensitive Networking (TSN) switches that support Per-Stream Filtering and Policing (PSFP). It also includes the complete constraints of Comp algorithm to allow reviewers to check the solution's validity. It also describes how we calculate the pre-planned expected arrival time windows using the traffic planning algorithm.

I. Commercial TSN switches with PSFP

Industry practices have now integrated the PSFP feature into various switches, as shown in Table. I. Notably, two switches provide detailed information about gate and GCL parameters: Broadcom's BCM56070 and Marvell's 88Q5152.

The BCM56070 switch from Broadcom supports 128 gates, with each gate capable of holding up to 16 entries in its GCL [1]. Marvell's 88Q5152 switch supports 8 gates, with each gate also able to accommodate a maximum of 16 entries in the GCL [2]. Specific details about the 88Q5152's parameters can be found in the confidential register specifications.

NXP's SJA1105TEL switch supports time-based policing, although it does not disclose specific gate parameters [3].

Other switches, such as Broadcom's BCM53570, NXP's SJA1110, and NXP's LS1028A, also support PSFP, according to their product briefs, but detailed specifications are confidential and not available to the public.

Additionally, two other switches claim to support PSFP, but they only offer filtering and metering functions, without supporting time-based policing. These are the Broadcom BCM53112 and BCM53162.

To our knowledge, we argue two main points: first, the current scalability of PSFP implementation in TSN switches may be limited. Second, there is a lack of publicly available information about the gate mechanism of PSFP, indicating a gap in the literature.

II. Constraints for Comp algorithm

The algorithm that calculates the transmission time of each stream instance uses a system model that differs from the algorithm for FooDog. Therefore, the system model and the constraints used by the comparison algorithm is described here.

System model. The network is represented using $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is a set of vertices including switches and end systems, and $\mathcal{E}$ is a set of directed edges. (a,b) denotes a network link from vertex a to b, $(a,b) \in \mathcal{E}$. (a,b) consists of four-tuples, $< BD^{(a,b)}, Q^{(a,b)}, MaxDly^{(a,b)}, MinDly^{(a,b)}, t^{(a,b)} >$, which includes the bandwidth, available transmission queues for TS streams, maximum link delay, and planning granularity respectively. The planning granularity refers to the minimum size of the time unit used for scheduling. The synchronization precision is denoted as δ. $\mathcal{F}$ is a set of TS streams. f_i is the i-th TS stream, $f_i \in \mathcal{F}, i \in [0, N-1]$, where N is the total number of TS streams. f_i consists of five-tuples $< \mathcal{R}_i, C_i, T_i, L_i, J_i >$, which includes route path, frame size in bytes, period, maximum allowed end-to-end delay, and maximum allowed end-to-end jitter, respectively. Note T_i is the interval between two consecutive frames sent by the source device and is a constant value for a TS stream. T_i, L_i and J_i are in the unit of the planning granularity. The $f_{i,j}^{(a,b)}$ is the j-th frame of f_i at link (a,b) in a network cycle period. The network cycle period, denoted by T, is the LCM of periods of all TS streams. $\mathcal{F}^{(a,b)}$ is the set of the frames that are routed through link (a,b). Each frame $f_{i,j}^{(a,b)}$ is attached with two unknown variables that need to be solved, which are $\omega_i^{(a,b)}$ and $\rho_i^{(a,b)}$. $\omega_{i,j}^{(a,b)}$ is the transmission time that frame $f_{i,j}^{(a,b)}$ is scheduled at link (a,b), which is aligned with the planning granularity. $\rho_{i,j}^{(a,b)}$ is the transmission queue to buffer frame $f_{i,j}^{(a,b)}$ at link (a,b).

Core Constraints. The problem of traffic planning in TSN is a constraint-solving problem. The constraints of the comparison algorithm are described below.

Period constraint requires that the TS frame of a period must be scheduled during the current period, as shown in Equation (1). There are two reasons. First, this constraint can reduce the buffered frames in the switch. If the $k+1$-th frame has arrived while the k-th frame has not been scheduled, then the switch needs to buffer two frames. The on-switch memory is limited, however. Second, the constraint can reduce the search space to a reasonable range.

$$\begin{gathered}\forall f_i \in \mathcal{F}, \forall (a,b) \in \mathcal{R}_i, \forall j \in [0, \frac{T}{T_i} - 1] : \\ \left(\omega_{i,j}^{(a,b)} \geq j \times T_i\right) \wedge \left(\omega_i^{(a,b)} < (j+1) \times T_i\right)\end{gathered} \tag{1}$$

Contention-free constraint ensures that frames that are transmitted through the same physical link should not overlap in the time domain, which is the basic principal of transmission in time-triggered network including TSN. The constraint is shown in Equation (2).

$$\begin{gathered}\forall (a,b) \in \mathcal{E}, \forall f_i^{(a,b)}, f_j^{(a,b)} \in \mathcal{F}^{(a,b)}, i \neq j, \\ \forall k \in [0, \frac{LCM(T_i, T_j)}{T_i} - 1], \forall l \in [0, \frac{LCM(T_i, T_j)}{T_j} - 1] : \\ \left(\omega_{i,k}^{(a,b)} \geq \omega_{j,l}^{(a,b)} + \frac{C_j}{BD^{(a,b)}}\right) \vee \left(\omega_{j,l}^{(a,b)} \geq \omega_{i,k}^{(a,b)} + \frac{C_i}{BD^{(a,b)}}\right)\end{gathered} \tag{2}$$

TABLE I: Commercial TSN switches integrating with PSFP

Vendor	Product	Time-based policing	Supported gate	Supported entries	Application
Broadcom	BCM53112	✗	✗	✗	High-end Fast-Ethernet applications
	BCM53162	✗	✗	✗	Industrial, and service provider markets
	BCM53570	?	?	?	Industrial Ethernet, 5G, and transport networks
	BCM56070	✔	128	16	Enterprise, Service Provider
NXP	SJA1110	?	?	?	automotive applications
	SJA1105TEL	✔	?	?	automotive applications
	LS1028A	?	?	?	Industrial and automotive applications
Marvel	88Q5152	✔	8	16	Automotive applications

Sequence constraint. The routing of a TS frame is sequential. The scheduling time on the upstream device must be earlier than the scheduling time on the downstream device. Besides, the scheduling time between neighbor links (a,x) and (x,b) must be larger than $MaxDly^{(a,x)}$ (Equation (3)). Otherwise, the frame has not arrived at the downstream device, but the device has already scheduled the frame.

$$\begin{aligned} &\forall f_i \in \mathcal{F}, \forall (a,x),(x,b) \in \mathcal{R}_i, \forall j \in [0, \frac{T}{T_i}-1]: \\ &\omega_{i,j}^{(x,b)} \times t^{(x,b)} \geq \\ &\omega_i^{(a,x)} \times t^{(a,x)} + \frac{C_i}{BD^{(a,x)}} + MaxDly^{(a,x)} + \delta \end{aligned} \tag{3}$$

End-to-end constraint. TS streams require stringent determinism. The end-to-end delay must be less than the allowed maximum end-to-end delay, as shown in Equation (4).

$$\begin{aligned} &\forall f_i \in \mathcal{F}, \exists src(\mathcal{R}_i), dst(\mathcal{R}_i), \forall j \in [0, \frac{T}{T_i}-1]: \\ &\omega_{i,j}^{src(\mathcal{R}_i)} \times t^{src(\mathcal{R}_i)} + L_i \geq \\ &\omega_{i,j}^{dst(\mathcal{R}_i)} \times t^{dst(\mathcal{R}_i)} + \frac{C_i}{BD^{dst(\mathcal{R}_i)}} + \delta \end{aligned} \tag{4}$$

Frame isolation constraint avoids frame interleaving. The transmission time for any two successive TS frames from upstream links can only be in either case: (1) After the former frame has been scheduled to the downstream link, the latter frame is scheduled at the upstream link; (2) The two frames use different queues. The formulation of the first case is as Equation (5).

$$\begin{aligned} &\forall (a,b) \in \mathcal{E}, \forall f_i^{(a,b)}, f_j^{(a,b)} \in \mathcal{F}^{(a,b)}, i \neq j, \\ &\forall k \in [0, \frac{LCM(T_i,T_j)}{T_i}-1], \forall l \in [0, \frac{LCM(T_i,T_j)}{T_j}-1]: \\ &\left(\omega_{j,l}^{(a,b)} \times t^{(a,b)} + \delta \leq \omega_{i,k}^{(x,a)} \times t^{(x,a)} + MaxDly^{(x,a)}\right) \\ &\vee \left(\omega_{i,k}^{(a,b)} \times t^{(a,b)} \geq \omega_{j,l}^{(y,a)} \times t^{(y,a)} + MaxDly^{(y,a)}\right) \\ &\vee \left(\rho_{i,k}^{(a,b)} \neq \rho_{j,l}^{(a,b)}\right) \end{aligned} \tag{5}$$

The jitter constraint is not presented here. Since there is only a single frame in the queue, the transmission of the frame is deterministic. Ideally, the end-to-end jitter depends on the synchronization precision δ.

Queue constraint. The number of queues in the switch is typically less than 8. The queues that can be used by TS streams are limited (Equation (6)).

$$\begin{aligned} &\forall \mathcal{F}_i \in \mathcal{F}, \forall (a,b) \in \mathcal{R}_i, \forall j \in [0, \frac{T}{T_i}-1]: \\ &(\rho_{i,j}^{(a,b)} \geq 0) \wedge (\rho_{i,j}^{(a,b)} < Q^{(a,b)}) \end{aligned} \tag{6}$$

III. Generation of Period-wise GCL

The planning algorithm calculates the transmission timing of frames at each egress port and determines which transmission queue to buffer the frame. Since the expected arrival time window of a frame at downstream PSFP gates is determined by the time when the frame is scheduled by the upstream device, so the configuration of the Period-wise GCL is described by Equation (7).

The upper and lower bounds of the expected arrival time window at the downstream switch meet Equation (7).

$$\begin{aligned} inf|\kappa_i^{(a,b)}| &= \omega_i^{(a,b)} + MinDly^{(a,b)} - \delta, \\ sup|\kappa_i^{(a,b)}| &= \omega_i^{(a,b)} + MaxDly^{(a,b)} + \delta \end{aligned} \tag{7}$$

It describes the upper and lower bounds of the expected arrival time at the downstream switch. (a,b) is a network link from device a to device b and $f_i^{(a,b)}$ is a frame that is routed through (a,b). $\omega_i^{(a,b)}$ is the time that the first bit of a frame f_i is scheduled by the upstream device a, and f_i is the first frame of the TS stream $\mathcal{F}_i$ within the network cycle period. $MinDly^{(a,b)}$ and $MaxDly^{(a,b)}$ are the minimum and the maximum link delay of (a,b), respectively. δ is the maximum deviation of any two devices' synchronized clock, i.e., synchronization precision. $\kappa_i^{(a,b)}$ is the expected arrival time of the first bit of f_i at device b. Thus $inf|\kappa_i^{(a,b)}|$ is the earliest expected arrival time and $sup|\kappa_i^{(a,b)}|$ is the latest expected arrival time. The earliest expected arrival time and the latest expected arrival time match the gate's opening and closing times.

References

[1] BROADCOM, "Bcm56070 switch programming guide." https://docs.broadcom.com/doc/56070-PG2-PUB. [Accessed 13-05-2024].

[2] Marvell, "Marvell® Brightlane™ 88Q5152 Secure Managed Switch." https://www.marvell.com/content/dam/marvell/en/public-collateral/automotive-solutions/marvell-brightlane-88q5152-secure-managed-switch-product-brief.pdf. [Accessed 13-05-2024].

[3] NXP, “Sja1105: 5-port automotive ethernet switch.” https://www.nxp.com.cn/docs/en/data-sheet/SJA1105.pdf. [Accessed 13-05-2024].