PERFORMANCE ANALYSIS PROBLEMS IN TSN NETWORKS

Umar Nisar

***Abstract*—Time Sensitive Networking (TSN) is a collection of standards aimed at improving the real time peformance of ethernet interface. The TSN standards, however, lack the formal latency guarantees. For that different models are proposed employing methodologies such as Network Calculus, Compositional Performance Analysis and Machine Learning. In this paper we review different performance analysis techniques at our disposal and draw comparison with respect to their methodology, completeness, accuracy and scalability.**

***Index Terms*—TSN, Performance Analysis, Machine Learning,**

I. Introduction

THe latency requirement for industrial applications are of the order of a few millisecond, traditional networks, however, have latency of tens of millisecond [1]. Time Sensitive Networking (TSN) task group [2] is aimed at standardizing Ultra-Low Latency (ULL) network stack on the standard ethernet interface, in order to incorporate real-time awarness on to an otherwise time agnostic ethernet data link layer protcols. There is a need of comprehensive timing analysis of such protcols to determine schedulability of dynamic time-sensitive traffic in a TSN. There are several approaches that have been explored in order to provide the scheduling entity with the knowledge on the schedulability of a certain traffic. These methods can be broadly categorized in to two categories, mathematical based approach and simulation based approach. Mathematical approaches formulate a time aware mathematical model of a TSN protocol in order to compute latency and other characterstics such as maximum buffer size of scheduled traffic given the network traffic scenario and topology. A simulation model mimics the behaviour of the real system through a set of rules. Simulation based approach has several drawbacks such as it does not usually show the worst-case latencies and design space exploration is cumbersome.

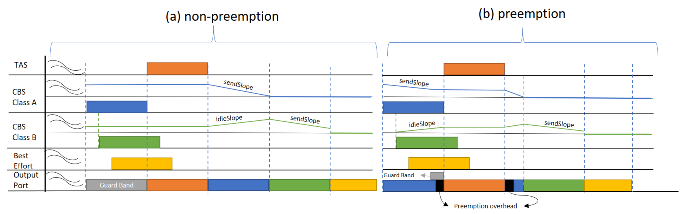

Fig. 1. Flow control at an output port employing TSN CBS and TAS with and without preemption

A unicast or multicast connection from one end station (talker) to another end station (listener) is called as a *flow*, or a *stream*. TSN has defined Time Aware Shaper (TAS) [3] and Credit Based Shaper (CBS) [4] as means to implement QoS strategies such as static priority scheduling and traffic shaping. Fig 1 shows an example of traffic scheduling at an output port. For details, consult [5].

In section II we review two mathematical models primarily for Audio Video Bridging (AVB) traffic in a TSN network. Furthermore, we also review a Machine Learning (ML) based approach to determine schedulability of AVB traffic. In section III we carry out a discussion on the characterstics of the three models in an effort to make a qualitative comparison amongst them.

II. Analysis

As discussed, TSN employs broad range of standards aimed at different use-cases with stringent latency requirements. In the presence of multitude of TSN standards the analysis to determine the schedulability becomes involved. In this section, we review three different techniques aimed at the timing analysis of dynamic traffic of a network.

A. ML Based Analysis

The methodology presented in [6] significantly differs from the other techniques discussed here, it does not explore underlying timing analysis to

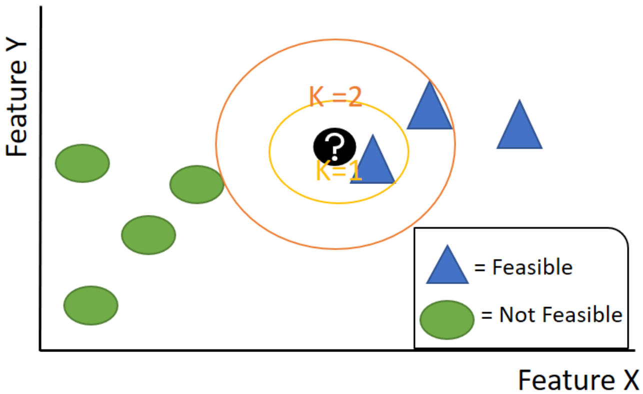

Fig. 2. A toy KNN model for two features, the new configuration will be characterized as 'feasible' for $K < 6$

determine schedulability of a new configuration rather it makes decision based on its similarity with precomputed configurations.

1) Model: In order to evaluate feasibility of a new application, it is evaluated against four possible TSN configurations (or solutions). The Four TSN configurations are FIFO, Manual, Concise Priority (CP8) and Manual with traffic shaping, for details see [6]. The configurations are compatible with the standard static priority scheduling algorithms in order to compare the ML model results with so called "approximate analysis" and "precise analysis" based off [7]–[9].

A generic topology of network is given for which three classes of traffic is scheduled namely Audio, Video and Command and Control (C&C). The ML model that is used to determine feasibility of a configuration is K Nearest Neighbors (KNN). Fig. 2 shows a KNN model for two features X and Y. In order to classify a new configuration (marked as '?') an L2 norm is taken to determine dominant label of nearest neighbors as shown by the circles for the case of K=1 and 2.

In the ML model developed in [6], following five features (D=5) of a configuration are selected based on the empirical results: the number of critical flows, the number of audio flows, the number of video flows, the maximum load of the network and Gini Index. Gini Index [10] is a quantitative characterization of unbalancedness of link loads. The decision sought from the model is binary i-e if a given point in a D dimensional space connotes to either a feasible or an infeasible configuration.

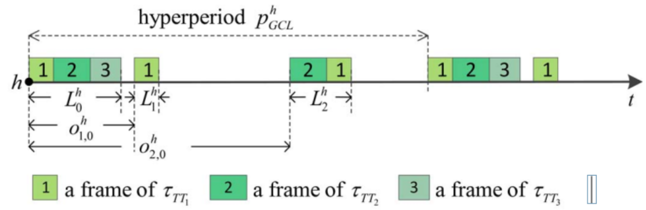

Fig. 3. An example of scheduled TT traffic at an output port h

2) Results: The training data for each solution is generated by labelling randomly generated data of the given topology by the results of 'precise analysis' [7]–[9] which are considered to be 100% accurate. A training data set of N=5200 was deemed optimum. The overall accuracy of the model reduces with increasingly complex solution. For the case of FIFO scheduling the model is a bit biased towards infeasible predictions. The reason behind that is most of the training data set contained not feasible solutions. The accuracy of the model is worst than 'approximate' analysis on average.

B. Network Calculus Based Analysis

Network calculus (NC) is a toolset that is used in deterministic networking to compute bound on queuing, delays, buffers etc. In this paper [11] Network Calculus is used to derive an upper bound on the latency of AVB flow of TSN, which can then be used to ascertain shcedulability of the traffic.

1) Model: The worst case analysis for an AVB traffic flow in presence of Time Triggered (TT) traffic is carried out by calculating the arrival and service curves of the flow. Arrival and service curves are functions that bound arrival and departure process respectively, for details [12]. As TT traffic takes priority over any AVB flow, the paper ingeniously makes use of the arrival curves of TT traffic in order to compute service curve of the AVB flow. This makes sense because when a node is not engaged in TT traffic and credit of the particular flow is positive only then the traffic of that flow is scheduled at an output port. Since TT traffic is scheduled offline assuming in a periodic fashion as show in figure 3, this period is referred to as hyperperiod, the arrival curve for TT traffic of a particular port h would simply be the summation of the bits transferred in each TT traffic window in a hyperperiod.

In the case of absence of IEEE 802.1Qbu, i-e no preemption of the low priority frames, the guard period needs to be catered to in the arrival curve. For the worst case the length of guard band would be either the maximum transmission time of AVB frames or just idle period if that is shorter than the aforementioned time period. Each guard band is appended in front of each TT traffic window. Following the same logic as before, the guards bands can be thought of as an extension of TT windows, since no traffic is scheduled within that time frame.

Now, considering an AVB flow of class M, an interval Δt at an output port h can be broken down as: $\Delta t = \Delta t^{+} + \Delta t^{-} + \Delta t^{0}$ where Δt^{+} refers to the time spent in waiting, Δt^{-} is the time duration of the transmission and Δt^{0} refers to the time duration of TT and GB windows. As service curve is a bound on the departure function, it can be well characterized by bits served in Δt^{-}. This knowledge along with arrival curves of the TT and GB windows help derive the service curve for the AVB flow of class M.

The arrival curve of the AVB flow is just a summation of the bursts arrived at a node at a given point in time and their long-term rate of arrival. The maximum horizontal deviation of arrival and service curve of an output port h draws an upper bound on the latency. The arrival curve of subsequent nodes can be estimated by summing the arrival curve of the current node with upper bound of latency for that node.

2) Results: The results derived for AVB analysis are shown to be scalable in presence of TT flows which are increased from 15 to 100. The analysis works for preemption where the computed Worst Case Delays (WCDs) are slightly better. The analysis also noted that the WCD of AVB flow reduce with an increased value of idle slope, naturally. The analysis also explore the effect of the routing path in a TSN topology and discovers significant impact of routing on the WCD of AVB flows.

C. Compositional Performance Analysis (CPA) of AVB Ethernet Traffic

This technique [13] differs from the NC one primarily because of the mathematical approach towards tackling the timing analysis of the network traffic. This paper makes use of CPA toolset to derive an upper bound on the latency of a traffic stream of class AVB.

1) CPA Background: CPA model is composed of a set of tasks that are then processed by the a set of resources. For example an output port is a resource that executes tranmission of different flows in a network that are reqarded as tasks. Each task, τ_i, is activated by their respective events, q, which are frames of a particular stream that keep the task busy referred to as busy time $B(q)$, part of the busy time is the execution time of the task on the resource associated with the task. Each task has a profile of events referred to as maximum and mimimum arrival curves, these arrival curves provide the information regarding the arrival of the events associated with the task in a given time interval.

2) CPA based Analysis: Busy time activated by q event is given by the equation:

$$B_i^{+}(q, a_i^q) \leq t_{transfer}(q) + I_{LPB} + I_{SPB}(a_i^q) + \\ I_{TSB}(a_i^q) + I_{HPB}(B_i^{+}(q, a_i^q))$$

where: $t_{transfer}(q) = q.C_i^{+}$ is maximum tranfer time for back to back q events and C is the time it takes for a frame to transmit at an output port. $I_{LPB} = \max\limits_{j \in lp(i)} C_j^{+}$ is worst-case time taken by a lower priority task that just started execution before arrival of the q-event. $I_{SPB}(a_i^q) = \sum\limits_{j \in sp(i)} (\eta_j^{+}(a_i^q).C_j^{+})$ is worst-case time taken by same priority events of task τ_j arrived until arrival time of q-event of τ_i given by a_i^q. $I_{HPB}(B_i^{+}(q, a_i^q)) = \sum\limits_{j \in hp(i)} (\eta_j^{+}(B_i^{+}(q, a_i^q) - C_i^{+}).C_j^{+})$ is worst-case time taken by a high priority task, τ_j, following the same logic as same priority task however the subtraction of C_i^{+} is due to the non-premption nature of onging low priority task, I_{TSB} is worst-case time taken by traffic shaper or insufficient credit which is dependent on other terms of the expression as blocking of τ_i due to any reason would result in an increase of the credit. For details consult [13] The upper bound on latency, l_p^{+}, of stream p is given by the summation of the worst-case response time, R_i^{+}, at each node along the path. In order to compute worst-case response time, R_i^{+}, at each node one has to find the worst-case time interval between the foremost q event arrival and the last q event departure of the same busy time characterized by

$R_i^+ = \max_{q \in Q_i} \big\{ \max_{a_i^q \in A_i} \{B_i^+(q, a_i^q) - a_i^q\} \big\}$. So far in the analysis, the effect of traffic shaper has not been incorporated into the calculation of arrival times of the events. For that, an upper bound on the traffic shaped by a shaper on a resource is computed which is then incorporated in to the output event model for each task. For detailed derivation consult [13].

3) Results: A general trend of much higer latencies in the case of AVB are observed which is explainable due to the maximum transmission bound imposed by shaper credit. It was seen that setting the idle slope of AVB traffic 31 times than the required bandwidth achieves the same latency as of strict priority. It was noted that in the case of linear topology the rate of increase of latencies with respect to the number of nodes was much higher than that observed in star topology. This is due to the intermediate shapers at each node that accumulated the effect of shaper at the output node.

III. Discussion and Conclusion

The ML paper [6] was premised around infeasibility of computational time for the case of NC based methods however the choice of ML model could have been improved as improvement in the result is only significant for the configurations $\geq 10^5$. Furthermore, it does not provide a worst-case estimate on the latencies which remains a major draw back even for a relatively accurate model. However, for a large number of configurations it provides better computation time when compared with both the NC based analyses.

The NC based analysis [11] seems to be the most comprehensive analysis amongst the reviewed techniques. The analysis also claim to provide optimistic results when compared with [14], which is definitely a plus for an NC based technique. However, the analysis does not make any reference to the degree of error in the estimate. An estimate on the computational resources and time would also help in determining the applicability of the model.

The CPA analysis [13] seems to be incomplete when TSN configurations with preemption are considered. Simplification in modeling of the shaper in the analysis does not explicitly account for the case of two shapers (such as Class A and B) on the same resource. The effect of TT traffic is not evaluated.

In this paper, we have discussed three different approaches aimed at providing the schedulability analysis of AVB traffic flow in a TSN network. We then discussed different aspects of the model that make them suitable (or not suitable) for the performance analysis of a real-time TSN.

References

[1] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[2] "Time-sensitive networking (tsn) task group." [Online]. Available: https://1.ieee802.org/tsn/

[3] "Ieee standard for local and metropolitan area networks – bridges and bridged networks - amendment 25: Enhancements for scheduled traffic," *IEEE Std 802.1Qbv-2015 (Amendment to IEEE Std 802.1Q-2014 as amended by IEEE Std 802.1Qca-2015, IEEE Std 802.1Qcd-2015, and IEEE Std 802.1Q-2014/Cor 1-2015)*, pp. 1–57, 2016.

[4] "Ieee standard for local and metropolitan area networks - virtual bridged local area networks amendment 12: Forwarding and queuing enhancements for time-sensitive streams," *IEEE Std 802.1Qav-2009 (Amendment to IEEE Std 802.1Q-2005)*, pp. C1–72, 2010.

[5] A. Nasrallah, A. S. Thyagaturu, Z. Alharbi, C. Wang, X. Shao, M. Reisslein, and H. ElBakoury, "Ultra-low latency (ull) networks: The ieee tsn and ietf detnet standards and related 5g ull research," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 88–145, 2019.

[6] T. Mai, N. Navet, and J. Migge, "On the use of supervised machine learning for assessing schedulability: application to ethernet tsn," 11 2019, pp. 143–153.

[7] A. Bouillard and r. Thierry, "An algorithmic toolbox for network calculus," *Discrete Event Dynamic Systems*, vol. 18, no. 1, p. 3–49, 2007.

[8] M. Boyer, J. Migge, and N. Navet, "An efficient and simple class of functions to model arrival curve of packetised flows," 11 2011.

[9] R. Queck, "Analysis of ethernet avb for automotive networks using network calculus," in *2012 IEEE International Conference on Vehicular Electronics and Safety (ICVES 2012)*, 2012, pp. 61–67.

[10] F. Farris, "The gini index and measures of inequality," *American Mathematical Monthly*, vol. 117, pp. 851–864, 12 2010.

[11] L. Zhao, P. Pop, Z. Zheng, and Q. Li, "Timing analysis of avb traffic in tsn networks using network calculus," in *2018 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, 2018, pp. 25–36.

[12] A. V. Bemten and W. Kellerer, "Network Calculus: A Comprehensive Guide," Chair of Communication Networks, Technische Universitat Munchen, Tech. Rep. 201603, October 2016. [Online]. Available: https://mediatum.ub.tum.de/doc/1328613/1328613.pdf

[13] J. Diemer, D. Thiele, and R. Ernst, "Formal worst-case timing analysis of ethernet topologies with strict-priority and avb switching," in *7th IEEE International Symposium on Industrial Embedded Systems (SIES'12)*, 2012, pp. 1–10.

[14] S. M. Laursen, P. Pop, and W. Steiner, "Routing optimization of avb streams in tsn networks," *SIGBED Rev.*, vol. 13, no. 4, p. 43–48, Nov. 2016. [Online]. Available: https://doi.org/10.1145/3015037.3015044