ComEX

LETTER

Proposal of TAS hypercycle expansion suppression method by transmission cycle period conversion

Yuhei Kawakami[1, a)], **Hideo Kawata**[1], **Akifumi Tanase**[1], **Hironao Abe**[1], **Shinichi Yoshihara**[1], **and Tomoaki Yoshida**[1]

Abstract IEEE802.1Qbv (Time-aware Shaper: TAS) has recently been applied within user networks (NWs) for realizing factory automation and remote control. As mass customization matures, TAS is expected to extend its application coverage to not only user NWs but also wide-area NWs to enable real-time communication for remote control and cloud-based application use. However, designing a TAS schedule is an NP-hard problem, which becomes more difficult as the gate control list (GCL) hypercycle increases. In this paper, we propose a method to convert the transmission cycle of Talker to a different cycle time for TAS scheduling in a carrier NW. This results in a smaller GCL hypercycle in the carrier NW. Our evaluation experiments show that the GCL computation time of the proposed method is shorter. This paper contributes to the realization of TAS in the wide-area NW, which has not been discussed so far.

Keywords: time-aware shaper (TAS), time-sensitive networking (TSN), ethernet, carrier network, delay, jitter
Classification: Network system

1. Introduction

In fields such as industrial networks (NWs) and automotive NWs, there is a need to control end-to-end delay and jitter with high precision and to establish ultra-low latency (ULL) communication. In particular, IEEE802.1Qbv (Time-aware Shaper: TAS) [1] standard, a component of IEEE802.1TSN (Time-Sensitive Networking), enables delay control on the order of µs. TAS was originally intended to be used in closed NWs, such as within a factory. However, ULL communication is expected to be realized in wide-area NWs. TAS controls transfers in accordance with a transmission schedule called gate control list (GCL). Scheduling GCL is known to be an NP-hard problem. Related works [2, 3] have shown that the computational difficulty of GCL depends on variations in the cycle period of the flows and the NW topology. Time-sensitive (TS) flows in a closed NW are due to the applications used by a single user, and the variations of transmission intervals and paths are assumed to be relatively small. However, when TAS is realized in a carrier wide-area NW, the characteristics of user applications are diversified and scheduling becomes more difficult than in a closed NW.

In this paper, we propose a scheduling method for multiplexing flows with various transmission cycles, assuming that TAS is applied to a shared large-scale NW.

[1] NTT Access Network Service Systems Laboratories, NTT Corporation, Midori-cho, Musashino-shi, Tokyo 180–8585, Japan
[a)] yuuhei.kawakami@ntt.com

DOI: 10.23919/comex.2024XBL0065
Received March 26, 2024
Accepted April 16, 2024
Publicized May 9, 2024
Copyedited July 1, 2024

2. Related work

In recent years, much work has been done to support the ULL communications required in TS applications such as industrial NWs. Nasrallah et al. [4] surveyed and summarized the latest developments in TSN and Deterministic Networking (DetNet) standardization from IT and OT perspectives. In a survey paper reviewing more than 170 academic papers on TSN, Seol et al. [5] elucidated the interoperability and economics of Ethernet-based ULL and low-jitter technologies.

Yan et al. [2] and Ansah et al. [6] discussed the computational feasibility of TAS scheduling, and Hellmanns et al. [3] proposed a convergent TAS scheduling model, stating that applying TAS to a large NW is an NP-hard problem because it requires computing the schedule of the entire NW. We think that their approach to easing the computational difficulty of designing TAS is also important when applying TAS to wide-area NWs.

Using Satisfiability Modulo Theories (SMT) for solving deterministic network scheduling problems was first proposed by Steiner [7]. Craciunas et al. [8] presented an example of using a constraint-satisfaction SMT solution for NP-hard scheduling problems. We assume and discuss a GCL that follows the constraints of their study.

Nasrallah et al. [9] evaluated IEEE 802.1Qcr (Asynchronous Traffic Shaper: ATS) developed to provide low-latency NW services without network-wide time synchronization. Joung et al. [10] also aimed to minimize jitter without time synchronization. Many research papers have been published on TSN that do not require time synchronization as described above. Delay control without time synchronization eliminates scheduling problems like TAS, but it does not provide as accurate delay control or low-latency transmission as TAS. This paper takes as its subject ULL communication across a high-precision time-synchronized NW.

3. Research challenge

In the computation of TAS scheduling, the SMT problem is commonly solved. In our previous work [11], we classified the patterns of unschedulability in TAS scheduling into two patterns. In the first pattern, no solution satisfies the SMT problem (unsat). This is the case where there is no solution for timeslot placement when performing no-wait scheduling. In the second pattern, the computation of the SMT problem

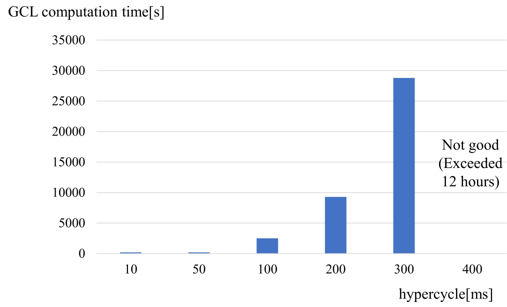

Fig. 1 GCL computation time per hypercycle.

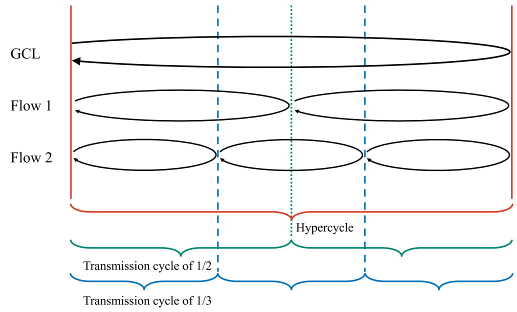

Fig. 2 Image of relationship between hypercycle and transmission cycle.

does not finish within a finite time (unknown).

In our previous work [11], we proposed a method to partition the NW into some domains to solve the first problem (unsat). It reduced the rate of unschedulability due to timeslot collisions and improved the capacity ratio. On the other hand, the second problem, computational complexity, is also important in the SMT problem. Jin et al. [12] stated that even the scheduling computation of several hundred flows could not be completed within two days. Our previous works have not evaluated computational complexity.

In this study, we addressed the length of the GCL hypercycle, which is a major cause of the second problem. Conventionally, the value of the hypercycle is the least common multiple of each transmission cycle, and the periodicity of each flow is expressed in terms of GCL. Carrier NWs multiplex scheduled traffic (ST) with various transmission cycle periods to accommodate various user applications. Therefore, the combination of ST cycles causes the hypercycle to grow. Figure 1 shows a graph evaluating the GCL computation time for a single switch (SW) when the hypercycle is varied in the case where 20% ST is accommodated relative to the physical bandwidth. As the hypercycle lengthens, the GCL computation space increases and the computational complexity grows exponentially. GCL computation is an NP-hard problem, and there are possible cases where the solution does not converge within a finite time. Therefore, in this paper, GCL calculations that are not completed within 12 hours are treated as GCL calculation infeasible.

4. Proposed method

4.1 Transmission cycle period conversion

We focused on the fact that the growth of the hypercycle can be suppressed by accommodating only those transmission cycle periods that are a divisor of the hypercycle. As shown in Fig. 2, the hypercycle of the GCL is set as the least common multiple of the transmission cycle variations of all flows. In other words, if only flows with transmission cycles that are a divisor of the hypercycle are allowed, the hypercycle will not increase.

As shown in Fig. 3, the Talker~Listener continues fixed-cycle communication with an arbitrary transmission cycle period, while the carrier NW performs TAS-SW scheduling limited to only those transmission cycle periods that reduce the hypercycle. The users do not need to be aware that the cycle period is converted at the carrier NW and can use the TAS-SW as a fixed-cycle communication.

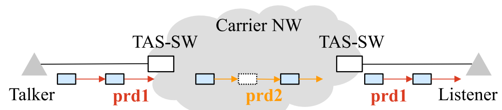

Fig. 3 Image of periodic transformation by the proposed method.

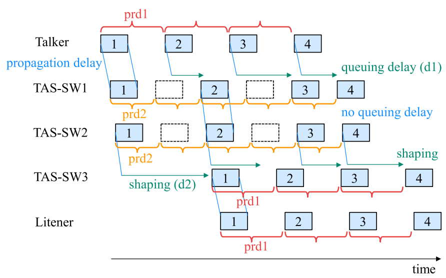

Fig. 4 Image of the delay caused by the periodic transformation.

When the transmission cycle period of prd1 of the added Talker is not a divisor of the hypercycle of the carrier NW, the TAS schedule is performed at the carrier NW by replacing prd1 with prd2, which has a shorter cycle than prd1 and is a divisor of the hypercycle. At this time, as shown in Fig. 4, at the first TAS-SW of the carrier NW closest to the Talker, the timing when the gate is opened does not synchronize with the timing of frame reception due to the different cycle, resulting in queuing for a time slot of prd2 at the maximum. This queuing delay is d1, and the maximum value of d1_max is as follows.

$$d1_{max} = prd2 \tag{1}$$

The TAS-SWs in the second and later stages of the carrier NWs transmit STs with minimal propagation delay only, without queuing, to perform normal TAS forwarding in prd2. In Fig. 4, only TAS-SW1~2 perform normal non-queuing TAS transmission, but all TAS-SWs perform non-queuing transmission, no matter how many stages they are configured and transmitted by prd2.

The TAS-SW at the last stage of the carrier NW near the Listener transmits STs with shaping by adding a delay to restore the period of prd1. As shown in Fig. 4, the last-stage TAS-SW3 defers transmission of the first packet until the second packet arrives. For example, if the last-stage TAS-SW grants only prd1 delay, TAS-SW1 and TAS-SW2 will have empty timeslots, and TAS-SW3 may have timeslots where frames do not arrive. For this reason, the prd1 interval cannot be restored only by giving a prd1 delay. The sum of the above added delay and the queuing delay for waiting

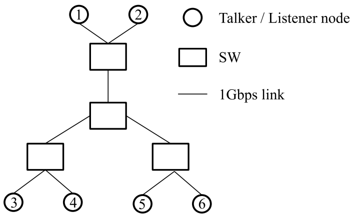

Fig. 5 Simulation NW configuration.

for the prd1 time slot is d2, and the maximum value of d2_max is as follows. d2_max represents the maximum frame interval including empty timeslots in TAS-SW2, and TAS-SW3 assigns this delay d2_max to the first frame. The second and later frames are not given above delay, but are queued until the time slot in which the gate is open in TAS-SW3 and are transmitted as soon as the gate is open.

$$d2\_max = ROUNDUP(prd1/prd2) * prd2 \quad (2)$$

The implementation described above enables flows to be added for Talker/Listener applications with arbitrary transmission cycle periods without bloating the TAS-SW hypercycle in the carrier NW.

4.2 Reuse of empty bandwidth generated by proposed method

Since prd2 is a shorter period than prd1, the carrier NW needs to design the GCL at a rate higher than the actual effective bandwidth of the ST. Normally, ST time slots do not allow transmissions other than scheduled STs, so this empty bandwidth becomes an invalid bandwidth, which creates a problem. Even if best-effort (BE) communications are queued, they are blocked to guarantee ST transmission.

We considered that invalid bandwidth could be prevented by transmitting BE in the time slots that will be invalid. In this paper, as a countermeasure for the invalid bandwidth caused by the proposed method, the periodic discrepancy caused by the combination of prd1 and prd2 is calculated in advance. By opening unused time slots for BE transmission, invalid bandwidth is not generated. The number of the transmitted STs is counted, and the BE gate opening timing is controlled by referring to the counter, thereby enabling countermeasures against periodic misalignment. Note that the time slots to which these measures against invalid bandwidth are applied cannot be reused as time slots that allow STs to be scheduled though it is possible to send BE communications.

5. Evaluation

5.1 Evaluation of proposed method

For an evaluation system, we set up a NW with traffic confluence as shown in Fig. 5. Communication between users must be handled exclusively, and this issue has been discussed in our previous papers [13]. The assumption is that the TAS-SW has enough queues for the number of flows. Traffic conditions are shown in Table I. Each Talker/Listener sends STs to arbitrary destinations. The bandwidth of all STs was 10 Mbps, and the transmission cycle period was randomly selected among 100, 200, . . . , and 1000 µs (in 100-µs increments). Five flow set patterns were prepared for evaluation. We evaluated the maximum ST capacity and GCL computation time under the condition of accommodating five flow sets using the existing and proposed methods, respectively.

Table I Simulation traffic configuration.

Talker node [node No.]	1, 2, 3, 4, 5, 6
Listener node [node No.]	1, 2, 3, 4, 5, 6
Bandwidth [Mbps]	10
Transmission cycle [µs]	100, 200, 300, 400, 500, 600, 700, 800, 900, 1000

Table II Example of periodic transformation.

prd1 [µs] (Talker transmission cycle)	prd2 [µs] (Carrier transmission cycle)
100	100
200	200
300	200
400	400
500	400
600	400
700	400
800	800
900	800
1000	800

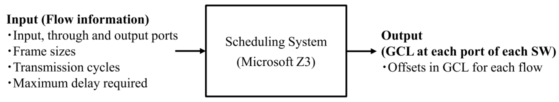

Fig. 6 Inputs and outputs of the scheduling system.

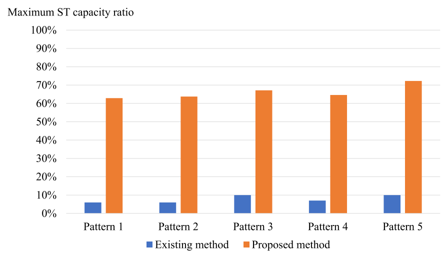

Fig. 7 Evaluation results (maximum ST capacity ratio).

The existing method has a hypercycle of 252,000 µs. In the proposed method, TAS-SW converts the ST transmission period to 100, 200, 400, and 800 µs as shown in Table II. For example, if Talker transmits STs with a 500-µs period, the carrier NW schedules STs with a 400-µs period. In this case, the TAS-SW hypercycle of the proposed method is 800 µs.

In the design of the GCL, SMT solver Z3 was used on the basis of the constraint-satisfaction problem proposed in Craciunas et al. [8]. As shown in Fig. 6, the flow information to be accommodated in the NW is entered into the scheduling tool to calculate the GCL. The SMT problem was solved by the Microsoft solver Z3 [14] and run on a Linux machine with a 2.4 GHz CPU and 376.6 GB of memory.

The evaluation results are shown in Fig. 7, Table III. The ST capacity ratio is used as the evaluation metric. The maximum ST capacity ratio is the total bandwidth used by the STs that can be scheduled. Because ST is a fixed-cycle communication that does not tolerate timeslot collisions, ST may not be possible to schedule at a ratio of 100% of the physical bandwidth. Increasing the maximum ST capacity ratio is equivalent to increasing the multiplicity of guaranteed-

Table III Evaluation results (execution time).

Flow sets	Existing method		Proposed method	
	Time	Failure Factors	Time	Failure Factors
Pattern 1	8h 47m	unknown	2m	unsat
Pattern 2	8h 6m	unknown	2m	unsat
Pattern 3	6h 27m	unknown	2m	unsat
Pattern 4	5h 40m	unknown	2m	unsat
Pattern 5	1h 55m	unknown	2m	unsat

delay communications and can produce cost benefits for guaranteed-delay services. In addition, since the extra bandwidth can be used to transmit BE communications, almost 100% of the physical bandwidth can be used for BE communications and STs combined.

With the conventional method, the maximum ST capacity ratio ranged from 6% to 10%. This is due to the fact that the computation time of the conventional method becomes very long (about 10 hours), even though the ST accommodation rate is as low as 10%, and the computation does not converge. In the conventional method, the computational complexity of checking whether the constraint satisfaction is satisfied is very large due to the large hypercycle, which is the reason for the low ST capacity ratio.

The maximum ST capacity ratio for the proposed method is 63%~72%. This is due to it having a significantly shorter computation time than the conventional method, and the calculation can be completed within one minute for 10% accommodation. The main reason the proposed method cannot accommodate STs is not that the computation time exceeds 12 hours as in the conventional method, but that the time slots of each flow collide and the constraint condition cannot be satisfied. Note that the maximum ST capacity ratio in the proposed method excludes the invalid bandwidth allocated for periodic conversion of the proposed method. These evaluations show that the proposed method not only suppresses TAS-SW hypercycle bloat but also accommodates about 6 to 11 times more flows than the conventional method due to efficient timeslot allocation, even when considering the invalid bandwidth.

5.2 Evaluation of latency

In the evaluation system described in Section 4.1, the sum of the delays shown in Eqs. (1) and (2) is the maximum additional delay added to ST by the proposed method. In this setup, a maximum delay of 2.4 ms is added compared to a normal TAS. The delay added by the proposed method increases as the transmission cycle period of the ST increases. In this evaluation of the proposed method, we converted the transmission cycle to prd2, which is smaller than prd1 and closest to prd1. On the other hand, the delay given by the proposed method can be reduced by setting prd2 to a smaller value, for example 200 µs when prd1 is 1000 µs. These are trade-offs, since a larger difference between prd1 and prd2 also results in a larger invalid bandwidth.

Referring to Nasrallah et al. [4], applications such as telemedicine, haptic feedback have much stricter requirements for jitter than for delay. The proposed method, which can control jitter to a few µs even when a fixed delay of a few ms is given, proves to be effective for use cases that require low jitter. When applying the proposed technique, the prd2 parameters need to be tuned to meet the delay and jitter requirements of the use case.

6. Conclusion

We proposed a method to reduce the gate control list (GCL) computation time by converting scheduled traffic (ST) of an arbitrary cycle period into a limited period that does not bloat the hypercycle in the time-aware shaper switch when accommodating them in the carrier network (NW). We showed that the proposed method reduces the GCL computation time and improves the ST accommodation ratio by a factor of 6 to 11. The delay added by the proposed method was evaluated in a numerical simulation, but delay has not been evaluated by using a real verification system. In the future, we plan to apply the scheduling based on the proposed method to a real large-scale NW and evaluate the performance of time-sensitive networking forwarding, including time synchronization performance.

References

[1] "IEEE 802.1: 802.1Qbv — Enhancements for scheduled traffic," https://www.ieee802.org/1/pages/802.1bv.html.

[2] J. Yan, W. Quan, X. Jiang, and Z. Sun, "Injection time planning: Making CQF practical in time-sensitive networking," IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, pp. 616–625, 2020. DOI: 10.1109/infocom41043.2020.9155434

[3] D. Hellmanns, A. Glavackij, J. Falk, R. Hummen, S. Kehrer, and F. Dürr, "Scaling TSN scheduling for factory automation networks," 2020 16th IEEE International Conference on Factory Communication Systems (WFCS), pp. 1–8, 2020. DOI: 10.1109/wfcs47810.2020.9114415

[4] 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 Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 88–145, 2019. DOI: 10.1109/COMST.2018.2869350

[5] Y. Seol, D. Hyeon, J. Min, M. Kim, and J. Paek, "Timely survey of time-sensitive networking: Past and future directions," *IEEE Access*, vol. 9, pp. 142506–142527, 2021. DOI: 10.1109/access.2021.3120769

[6] F. Ansah, M.A. Abid, and H. de Meer, "Schedulability analysis and GCL computation for time-sensitive networks," 2019 IEEE 17th International Conference on Industrial Informatics (INDIN), pp. 926–932, 2019. DOI: 10.1109/indin41052.2019.8971965

[7] W. Steiner, "An evaluation of SMT-based schedule synthesis for time-triggered multi-hop networks," Proc. 31st IEEE Real-Time Syst. Symp., pp. 375–384, Nov. 2010. DOI: 10.1109/RTSS.2010.25

[8] S.S. Craciunas, R.S. Oliver, M. Chmelík, and W. Steiner, "Scheduling real-time communication in IEEE 802.1Qbv time sensitive networks," Proc. 24th International Conference on Real-Time Networks and Systems (RTNS '16), pp. 183–192, 2016. DOI: 10.1145/2997465.2997470

[9] A. Nasrallah, A.S. Thyagaturu, Z. Alharbi, C. Wang, X. Shao, M. Reisslein, and H. Elbakoury, "Performance comparison of IEEE 802.1 TSN time aware shaper (TAS) and asynchronous traffic shaper (ATS)," *IEEE Access*, vol. 7, pp. 44165–44181, 2019. DOI: 10.1109/access.2019.2908613

[10] J. Joung and J. Kwon, "Zero jitter for deterministic networks without time-synchronization," *IEEE Access*, vol. 9, pp. 49398–49414, 2021. DOI: 10.1109/access.2021.3068515

[11] Y. Kawakami, H. Kawata, N. Yasuhara, H. Abe, S. Yoshihara, and T. Yoshida, "TAS scheduling with network domain division," ICC 2023 - IEEE International Conference on Communications, Rome, Italy, pp. 6066–6071, 2023. DOI: 10.1109/icc45041.2023.10278585

[12] X. Jin, C. Xia, N. Guan, C. Xu, D. Li, Y. Yin, and P. Zeng, “Real-time scheduling of massive data in time sensitive networks with a limited number of schedule entries,” *IEEE Access*, vol. 8, pp. 6751–6767, 2020. DOI: 10.1109/access.2020.2964690

[13] Y. Kawakami, H. Kawata, T. Kubo, N. Yasuhara, S. Yoshihara, and T. Yoshida, “Applying time-aware shaper considering user identifier to service provider network,” 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC), pp. 505–506, 2022. DOI: 10.1109/CCNC49033.2022.9700587

[14] L. de Moura and N. Bjørner, “Z3: An efficient SMT solver,” Proc. 14th Int. Conf. Tools Algorithms for Construct. Anal. Syst., pp. 337–340, 2008. DOI: 10.1007/978-3-540-78800-3_24