A Study on Optimal Ethernet-based Industrial Networks Construction Using Asynchronous Traffic Shaping in IEEE 802.1TSN and Downhill Simplex Method

Akari Yoshimura and Yoshihiro Ito
Graduate School of Engineering Nagoya Institute of Technology
Nagoya, 466-8555, Japan
Email: {yosimura@en., yoshi@}nitech.ac.jp

***Abstract*—This paper studies using Asynchronous Traffic Shaping (ATS), which is defined by IEEE 802.1TSN (Time-Sensitive Networking), to transmit time-sensitive traffic over Ethernet-based industrial networks. In addition, the authors use the Downhill simplex method, a numerical optimization method, to calculate optimal ATS parameter values to meet industrial network requirements. The paper applies the method to an actual industrial network by experiment and confirms its effectiveness from the experimental results.**

I. Introduction

As we move toward Industry 5.0, more diverse and large amounts of data will be required on factory networks to realize collaboration between humans and robots. Therefore, the adoption of high-speed Ethernet-based networks is underway to transmit the above data. Moreover, since the above human-robot collaboration requires low latency and highly reliable data transmission, Quality of Service (QoS) control over Ethernet is inevitable.

To realize such QoS control, different organizations define various Ethernet-based industrial networks. This makes the development of interconnections among factory networks difficult. Consequently, it is currently considering the adoption of IEEE 802.1TSN (Time-Sensitive Networking) standards as a common QoS control for industrial networks. A project, IEC/IEEE P60802[1], is especially standardizing a profile of IEEE 802.1TSN standards for industrial networks. However, even if the industrial profile has been standardized, the profile alone cannot provide a network that meets the requirements of individual networks. That is, it is necessary to configure the network and set appropriate parameters for the IEEE 802.1TSN standard. However, this can be challenging in industrial networks with various network devices and their requirements.

To solve the problem mentioned above, [2] targets Time-Aware Shaper (TAS) as the QoS control in IEEE 802.1TSN and proposes a method to find appropriate parameters that satisfy the requirements of an industrial network by numerical optimization method; it confirms the effectiveness by experiment. However, since TAS requires time synchronization and complicated implementation, adopting and stabilizing it in large-scale industrial networks can be costly. Therefore, it is necessary to consider QoS control other than TAS.

Thus, this study will target Asynchronous Traffic Shaping (ATS) in IEEE 802.1TSN instead of TAS. Although ATS cannot provide as precise control as TAS, it only requires time synchronization between devices and complex implementation. On the other hand, Credit-Based Shaper (CBS) is defined in IEEE 802.1TSN as a technology similar to ATS. However, CBS has only one control parameter and cannot be precisely controlled, while ATS allows for more accurate control than CBS.

In this way, ATS can be expected to be effective in industrial networks, but it has never been clear whether ATS can meet the requirements of industrial networks. Thus, this paper tackles adopting ATS to an industrial network and finds optimal parameters using the numerical optimization method similar to [2]; it evaluates the effectiveness of applying ATS to industrial networks and also clarifies whether an optimal parameter can be obtained by experiment.

The rest of this paper is organized as follows: Sections II and III introduce an overview of TSN and a numerical optimization method, respectively. The experiments and their results are presented in Sects. IV and V, respectively. Finally, Sect. VI concludes this paper.

II. IEEE 802.1TSN

A. Outline

IEEE 802.1TSN is a set of standards developed to address real-time communication requirements over IEEE 802 networks and standardized by the 802.1 Working Group of the IEEE 802 Committee. IEEE 802.1TSN standards provide various technologies for transferring data over IEEE 802 networks with low latency and high reliability. Moreover, the IEC/IEEE P60802 project mentioned above, the IEEE P802.1DP[3] project, which defines a profile for in-aircraft networks, and the IEEE P802.1DG[4] project, which develops a profile for in-vehicle networks, and other IEEE 802.1TSN standard some projects are considering profiles.

Among the IEEE 802.1TSN standards, IEEE 802.1Q[5] specifies the definition and implementation of Virtual LANs (VLANs); it also defines time synchronization, network management, QoS control, etc. ATS is defined as one of the QoS controls of IEEE 802.1Q. The following is a description of the ATS discussed in this paper.

B. Asynchronous Traffic Shaping

ATS aims to efficiently schedule the transmission of data packets in a network and use the bandwidth effectively while minimizing delay. Figure 1 displays an outline of ATS. In Fig. 1, each switch has an independent clock and transmits frames based on a shaper implementing the ATS algorithm attached to the queue. The shaper equalizes the traffic by delaying excess traffic and holding frames in the queue when traffic exceeds a specified amount.

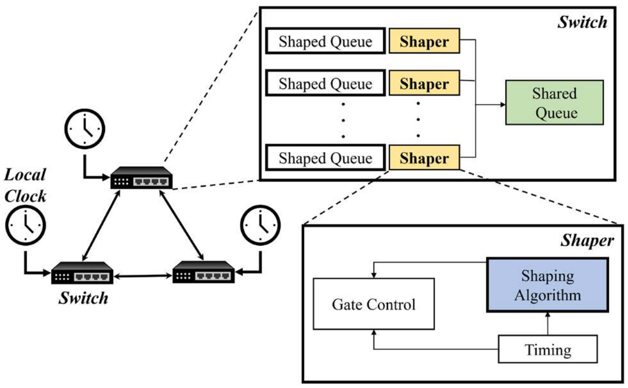

Fig. 1: Asynchronous Traffic Shaping

The ATS's scheduler uses the Token Bucket Emulation (TBE) method. Figure 2 indicates the behavior of TBE. As shown in Fig. 2, transmission rights called tokens are continuously emitted at a specific rate into a buffer called a token bucket. The capacity of the token bucket is limited. When a frame is transmitted, several tokens equal to the frame size are removed from the token bucket. When frames come in at a rate exceeding the average rate of tokens, transmission is either held back until more tokens accumulate, discarded, or sent with a lower priority.

ATS has three parameters: Max Residence Time (MRT), Committed Information Rate (CIR), and Committed Burst Size (CBS). MRT is the maximum time a switch can stay on. CIR and CBS are the token input rate and the number of tokens that can be deleted at one time, respectively. Consequently, frame delay can be limited in ATS without time synchronization between devices. As mentioned above, since ATS does not rely on time synchronization, it is expected to reduce implementation costs.

III. Numerical Optimization Methods

A. Outline

Numerical optimization methods are mathematical techniques for solving problems numerically to find the optimal solution. An optimization problem is formulated as the problem of finding the maximum or minimum value of a function under given conditions. Numerical optimization is widely used to find approximate solutions using algorithms, especially in complex problems for which no analytical solution is available. There are various numerical optimization methods, and the method to be employed should be selected according to the characteristics of the problem to be optimized. This study adopts the Downhill simplex method since it is easy to implement and has the potential to converge with a small number of function evaluations for nonlinear problems in high-dimensional spaces. The following subsection introduces an overview of the Downhill simplex method.

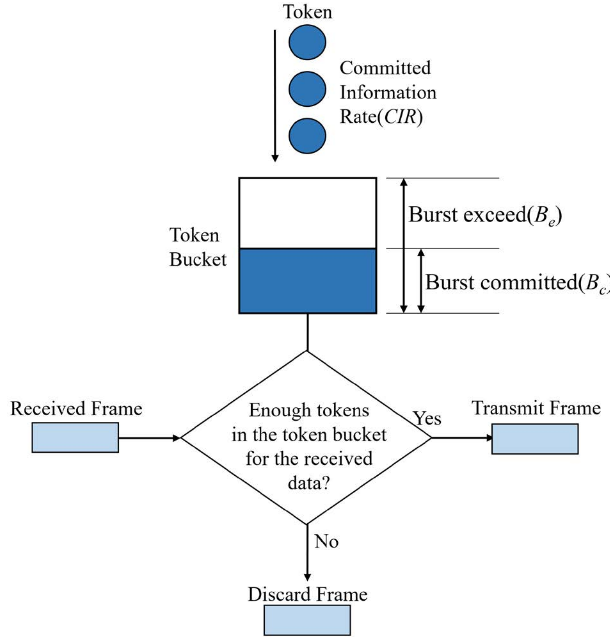

Fig. 2: Token Bucket Emulation

B. Downhill Simplex Method

The Downhill simplex method [6] can search for optimal solutions without using derivatives like the gradient method. It is, therefore, suitable for problems where the function could be more complex or smoother in shape. Its search is performed with a polyhedron (simplex), such as a triangle when the variable to be optimized is 2-dimensional and a tetrahedron when it is 3-dimensional. This simplex process is based on evaluating the objective function, gradually changing its shape as it approaches the optimal solution. Simplex deformation uses reflection, expansion, and contraction operations on the worst points to obtain new search points and gradually approach the optimal solution. Its derivative-free feature makes it a valuable algorithm for many real-world problems. However, since it tends to fall into local solutions and its efficiency decreases for high-dimensional problems, it is important to set appropriate initial conditions and, in some cases, to use it in combination with other methods.

IV. Experiments

A. Outline

In this experiment, ATS is applied to an Ethernet-based industrial network whose parameters are configured using parameters obtained by numerical optimization. Then, it is evaluated whether the requirements can be met by simulation. This experiment uses the Downhill simplex method as a numerical optimization method to find the optimal parameters of the ATS.

B. Experimental Environment

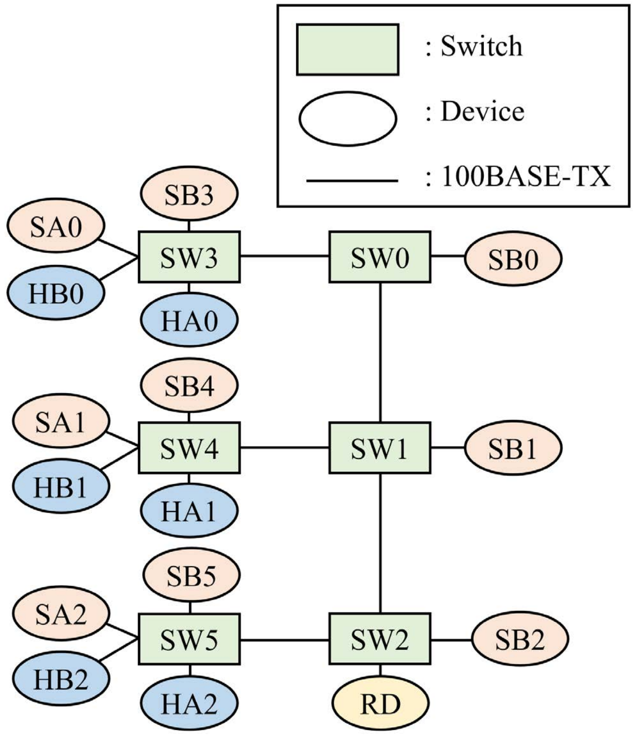

Fig. 3: Experimental Network

Our experimental network is shown in Fig. 3. This experiment uses the same experimental network as in [2] to evaluate the effectiveness of the ATS in the same environment. Assuming a simple actual factory network, the network consists of six devices (HA0, HA1, HA2, HB0, HB1, HB2) that send and receive data, nine devices (SA0, SA1, SA2, SB0 through SB5) that only send data, one device (RD), and six switches (SW0 through SW5) that only receive data. The lines between devices and switches and between switches are 100BASE-TX. Three devices (HA0, HA1, HA2) send three types of traffic received by the corresponding devices (HB0, HB1, HB2). HB0, HB1, and HB2, and SA0, SA1, and SA2 also send three traffic to the corresponding devices (HA0, HA1, HA2). HB0, HB1, and HB2 also send traffic to RD.

ATS is applied to each port of the switch. Table I shows the specifications and requirements for the traffic sent by each device. In Table I, NA means that no requirements are specified. The total number of traffic is 33, of which the number of traffic with requirements is 27. The experiment is conducted in simulation. Each traffic is assigned a traffic number from 1 to 33 for convenience.

In this experiment, the initial parameter values of the Downhill simplex method are selected randomly. However, depending on the initial value, locally appropriate values may be chosen as the optimal ones, so the initial value is changed, and the evaluation is repeated. The simulators used are OMNeT++6.0.1[7] and INET4.5[8].

TABLE I: Specification of Experimental Traffic

Device	Send Interval (μs)	Size (byte)	Priotiry	Requirement(μs)
HA0, HA1, HA2	500	150	7	350
HB0, HB1, HB2	500	150	7	350
SA0, SA1, SA2	1000	750	6	600
SB0, SB1, SB2, SB3, SB4, SB5	280∼520	500	5	NA

C. Criterion

It is necessary to define the evaluation values of the parameters derived from the requirements and results to apply the numerical optimization method to find the optimal ATS parameters. In this paper, a criterion f to be used in the ATS parameter search is defined as follows. First, we define the required delay $D_r(k)$ that the traffic must satisfy and the maximum delay $D_m(k)$ that has been measured. Here, $k(1 \leq k \leq N)$ in these variables represents traffic, N is the total number of traffic, and Nm is the number of traffic that satisfies the requirement. The criterion is obtained from Eq. (1). The criteria has a zero value if the delay requirement is met, and if not, the larger the difference from the delay requirement, the larger the value. This experiment terminates the search for appropriate parameter values at the 100th trial, regardless of whether the requirements are met.

$$f = \sum_{k=1}^{N} sgn(D_r(k) - D_m(k))(D_r(k) - D_m(k)/D_r(k))^2 \quad (1)$$

V. Results

Figure 4 shows the trends of the criterion in all the three experiments. In Fig. 4, the abscissa and ordinate mean the number of trials and the criterion, respectively. From Fig. 4, we see that one of the criterion values converges to 0. However, the other two criterion values converged at 0.3137 and 0.5605, respectively, and did not become smaller than those values.

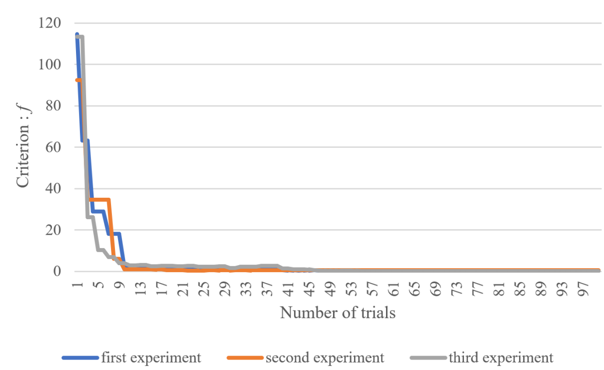

Fig. 4: Criterion vs. Trials

Figures 5 through 7 plot the measured maximum delays for all traffic. In these figures, the abscissa indicates the traffic number, and the ordinate indicates the measured maximum delay value. From Figs. 5 through 7, we see the following. From Figs. 5, we see that one of the traffic with priority 7 did not meet the requirement. Figure 6 indicates that two traffic with priority 6 did not satisfy the requirement. This is because the initial parameters were generated randomly and fell into the locally appropriate value due to the characteristics of the Downhill simplex method. On the other hand, from Fig. 7, all traffic satisfies the requirement. Therefore, when finding the optimal value using this proposal, we found that appropriate initial values for the Downhill simplex method are necessary. Thus, our future work will involve searching for this initial value.

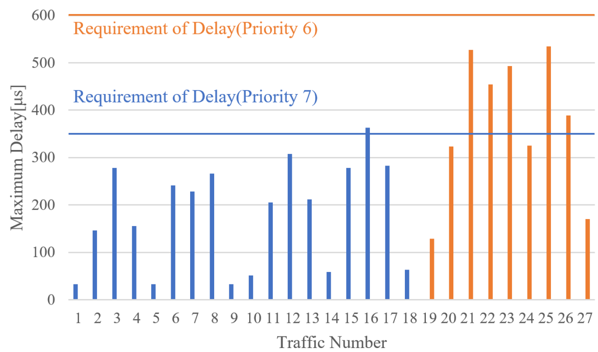

Fig. 5: Maximum delay for the first experiment

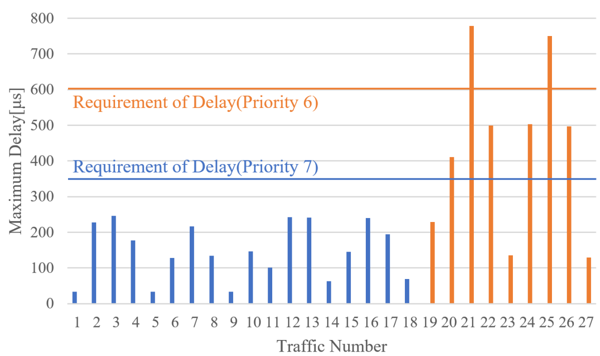

Fig. 6: Maximum delay for the second experiment

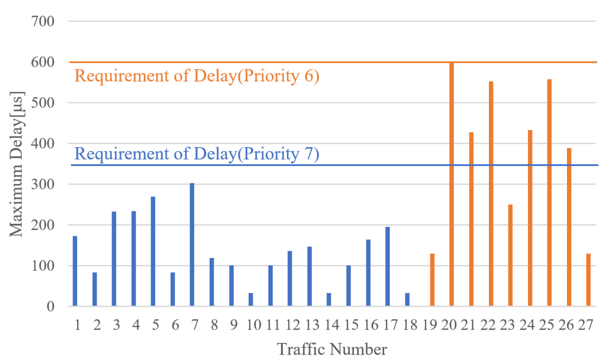

Fig. 7: Maximum delay for the third experiment

VI. Conclusions

We studied the optimal Ehternet-based industrial network configuration with ATS using the Downhill simplex method, a numerical optimization method. We confirmed the effectiveness of the method by experiment. However, we found that depending on the initial values of the simplex method, the optimal solution may not be found, and it is necessary to consider appropriate initial values.

Our future issues are as follows: First, we will consider how to search for the initial value of the Downhill simplex method. Second, we will evaluate the method using other industrial networks.

References

[1] "IEC/IEEE 60802,". https://1.ieee802.org/tsn/iec-ieee-60802/

[2] A. Yoshimura and Y. Ito, "A Study on Determination of an Appropriate GCL of Time-Aware Shaper in Ethernet-Based Industrial Networks," Proc. 2024 Fifteenth International Conference on Ubiquitous and Future Networks (ICUFN), Budapest, Hungary, pp.355-359, 2024.

[3] "IEEE P802.1DP Time-Sensitive Networking for Aerospace Onboard Ethernet Communications," https://standards.ieee.org/ieee/802.1DP/10465/

[4] "IEEE P802.1DG IEEE Draft Standard for Time-Sensitive Networking Profile for Automotive In-Vehicle Ethernet Communications, "https://standards.ieee.org/ieee/802.1DG/7480/

[5] "IEEE 802.1Q IEEE Standard for Local and Metropolitan Area Networks-Bridges and Bridged Networks," https://standards.ieee.org/ieee/802.1Q/6844/

[6] N. A. John and R. Mead, "A simplex method for function minimization," Computer Journal, Vol. 7, pp. 308-313, 1965 doi:10.1093/comjnl/7.4.308.

[7] "OMNeT++ Discrete Event Simulator,"https://omnetpp.org

[8] "INET Framework,"https://inet.omnetpp.org