Design of An Improved Ethernet AVB Model for Real-time Communication in In -Vehicle Network

Xiaoyu Liu[1,2,3] , Zhenbang Nie[1,2,3] ,Dong Li[1,2] ,Haibin Yu[1,2]
1. Key Laboratory of Networked Control System of CAS, Shenyang Institute of Automation of Chinese Academy of Sciences, Shenyang, China
2. Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang, China
3. University of Chinese Academy of Sciences, Beijing, China
liuxiaoyu1@sia.cn, niezhenbang@sia.cn, lidong@sia.cn, yhb@sia.cn

***Abstract*—The amount of traffics is increasing rapidly due to the increase of the number of electronic components and vehicle applications in vehicles. They require more network bandwidth and more stringent end-to-end latency and jitter constraints. In this situation, IEEE Audio/Video Bridging (AVB) is expected to be a promising technology in In-Vehicle Network since it guarantees high level of Quality of Service (QoS) for real-time communication. AVB defined tow SR-Classes for time sensitive traffics. However, as a result of the complexity of stream reservation domain construction, in-vehicle networks tend to use single traffic classes. In addition, the AVB jitter constraint is too weak. In this paper, an improved Ethernet AVB model with single SR-Class and time-triggered scheduling is proposed. In this model, time-sensitive traffics with high jitter requirements are forwarded through time-triggered scheduling, and ordinary time-sensitive flows are forwarded using SR-Class. Through simulations we confirm that proposed model not only satisfies the Qos requirements, but also has some extra advantages.**

Keywords—IEEE 802.1 Audio/Video Bridging; time-triggered scheduling; in-vehicle network

I. INTRODUCTION

Recently years, with the development of pilotless and intelligent transportation technology, Advanced Driver Assistance Systems (ADAS) with a large number of HD cameras, various sensors and infotainment equipments are used extensively, and diverse and complicated applications have been developed to enhance safety of vehicles and to meet requirements of infotainment systems, obviously increasing the number of traffics in-vehicle network. More stringent requirements were raised on bandwidth and end-to-end delay by these new changes.

However, hundreds of electronic control units （ECUs） are interconnected over heterogeneous and proprietary communication technologies in vehicles, including CAN, FlexRay, LIN, MOST and so on. But, they all cannot provide enough bandwidth, and the special mechanisms between different technologies have led to the difficulty of interoperability between them. In a word, traditional in-vehicle networks result in the difficulty of design, development and promotion of network architecture and application.

Ethernet already has proven to be a flexible, highly scalable and open standard network technology. Mature Ethernet application development experience and using the two-wire unshielding twisted pair cables of 100 Mbps, make the software and hardware development cost of vehicular network reduce significantly. Nevertheless, CSMA/CD and no difference of queuing forwarding between various traffic in switches could cause Ethernet frames be delayed or even discarded in certain network load situations. In order to handle this problem, IEEE 802.1 Audio / Video Bridging (AVB)[1]becomes one of the solutions with the QoS mechanisms. AVB enables time-synchronized low latency streaming services in standard 802 networks. It specified tow SR-Classes(SR-A and SR-B) for satisfying the QoS of low latency. The priority of SR-A is higher than that of SR-B. For SR-A, a maximum latency of 2ms over 7 hops is guaranteed, and for SR-B, maximum latency over 7 hops is 50ms.

But, standard AVB network can not fully match the characteristics of in-vehicular network. In-vehicle network is static and small-flow, applications usually do not generate traffics exceeding 100Mbps. Besides, the creation of the stream reservation domains of SR-A and SR-B takes twice time, increasing the complexity of the procedure of processing the data in queue. Therefore, we consider only using single SR-Class to complete data transmission under the QoS. AVB is an essentially event-triggered network, which only can guarantee the end-to-end delay. However, some functions of in-vehicle network(e.g. for autonomous driving)not only have rigid real-time requirements, but also have rigid requirements of jitter. So adding synchronous time-triggered scheduling to AVB is a simple and effective way. Time-triggered scheduling mechanism uses a time-division multiple access(TDMA)multiplexing strategy where real-time traffic only could be sent in corresponding time windows without any interference to meet the latency required with predictable minimal jitter.

In this work, AVB with single SR-Class and time-triggered scheduling are combined to allocate bandwidth for different traffics according to the different QoS of them. An improved Ethernet AVB model is built by OMNeT++[2], which is a switched network over 7 hops. Under the premise of meeting the QoS, the end-to-end delay and jitter of the network are compared in the cases of multiple SR-Classes, single SR-Class, and single SR-Class with time-triggered scheduling. Through the results of simulation, we verify that smaller end-to-end delay and lower jitter could be guaranteed

This work is supported by the National Science Foundation of China(Grant No.61533015).

in the case of single SR-Class with time-triggered scheduling than standard AVB.

This paper is organized as follows: In Section II , we introduce the IEEE 802.1 AVB standards with main mechanisms and related work. In Section III, we present our improved model and Section IV analyzes the simulation results in various scenarios. Finally, Section V concludes this work and suggests our future work.

II. BACKGROUND & RELATED WORK

This section introduces the IEEE 802.1 Audio/Video Bridging (AVB) standards sets and presents related work.

A. IEEE 802.1 Audio/Video Bridging (AVB)

AVB standard sets defined by IEEE Audio Video Bridging (AVB) Task Group (now renamed Time Sensitive Networking Task Group) consist of IEEE 802.1 AS, Qat, Qav, BA and IEEE 1722, 1733, 1722. The proposed protocols are based on Layer-2(MAC Layer) to support the QoS in switched Ethernet.

IEEE 802.1 AS[3]is time synchronization protocol which enables a MAC layer synchronization between distributed network nodes in switched Ethernet. The synchronization process is executed in two steps which includes the selection of a grandmaster (GM)clock node by the best master clock algorithm (BMCA) and broadcasting the synchronization information between distributed network nodes.

IEEE 802.1 Qat[4]is stream reservation protocol(SRP) which enables reserving bandwidth along the path between talker and listener. It provides a three step process to allocate or release bandwidth in switched Ethernet for AVB stream, including stream advertisement, registration and un-registration. SRP can reserve up to 75% of the total network bandwidth for SR-Classes. In addition, interval time of SR-A, B was defined 125us, 250us for each and the worst-case latency over 7 hops of SR-A, B was defined 2ms and 50ms.

IEEE 802.1 Qav[5]specifies queuing and forwarding rules. A credit based shaping(CBS) algorithm is proposed in 802.1 Qav. When the credit value is greater or equal 0, the transmission of a frame belonging to SR-Classes is allowed. An upper and lower bound of the credit limits the end-to-end delay, as shown in Figure 1. Streams of nodes that are unaware of the 802.1 Qav are mapped to IEEE 802.1Q, and their priorities are lower than SR-Classes.

B. Related Work

Various works have been dedicated to Ethernet-based communication in vehicles. Shane Tuohy et al.[6]present a overview of current research on advanced in-vehicle networks which comprehensively describe the most recent developments in the field of in-vehicle networking. They compare CAN, LIN, MOST, LVDS, FlexRay, AVB and TTEthernet in various types of traffic scenarios. At the beginning, IEEE 802.1Q is used to improve the performance of in-vehicle network. The papers[7][8]make a performance comparison of IEEE 802.1Q and AVB and come to conclusions that AVB is more suitable for in-vehicle network. Li Seul Kim et al.[9]design a single stream reservation class in AVB Networks. However, it only compares the maximum end-to-end delay in small-flow environment and lacks formal analysis and the QoS of jitter. Then, performance assessments of TTEthernet and AVB reveals strengths and weaknesses in[10][11], which aim to assess the behavior of AVB and TTEthernet when supporting ADAS and multimedia traffic under varying payload. As a result, both protocols all fit the QoS of ADAS and in-Car infotainment, but TTEthernet offers lower jitter that is suitable for the cross-domain communication. IEEE 802.1 Qbv[12]which is a enhancements for scheduled traffic of time-triggered traffic are rising. 802.1 Qbv combines stream reservation, time-triggered scheduling, frame preemption reasonably, and it uses priority values encoded into the VLAN tag to determine between scheduled and credit-based traffic.

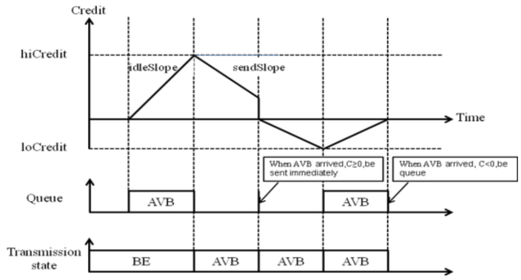

Fig. 1. Credit Based Shaping(CBS) Algorithm

III. IMPROVED AVB MODEL

This section presents the improved model from three parts of network topology, parameters configuration and experiment scenarios.

A. Network Topology

OMNeT++ with the CoRE4INET-framework[13]is used for establishing a switched in-vehicle network over 7 hops. Fig 2 shows the simulated network topology. The simulated in-vehicle network consists of 6 switches and 6 hosts in a tree topology. Tree topology offers a good trade-off between performance and the costs of installation and maintenance. Node 1,2,3,4,5 represent the talkers which generate different types of traffic depending on the simulation scenarios. The types of traffic that generated by talkers are driver assistance camera traffic, control information, audio traffic, video traffic and best effort traffic. Some of them are time-sensitive traffics, and some are ordinary Ethernet traffics. Node 6 represents the listener which receives the traffics from the talkers. From talker to listener, the maximum number of hops is 7 hops, and the minimum number of hops is 4.

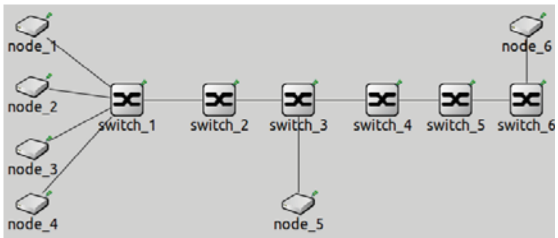

Fig. 2. Simulated Network Topology

B. Parameters configuration

Nodes and switches are connected through 100Mbps links as it allows the utilization of a physical layer and cables optimized for in-vehicle network.

As defined in IEEE 802.1 Qat, SR-Classes can occupy up to 75% of the total network bandwidth. Because we employ 100Mbps Ethernet in simulation model, so the bandwidth occupied by SR-Classes is 75Mbps. Here, we define the maximum transmission unit(MTU) for SR-Classes , which means the maximum size of the Ethernet frame that can be transmitted within the bandwidth allowed range[4].

$$MTU = \left(BW \times IT \right) \div MIF \tag{1}$$

Among them, BW expresses network bandwidth, IT indicates interval time, and MIF represents the maximum interval of frame which is set to 1in all scenarios.

In scenario 1,the MTU is calculated as follow:

$$\frac{2*MTU}{125us} + \frac{2*MTU}{250us} = 75Mbps \tag{2}$$

the MTU in the scenario 1 is 390.625 bytes.

The MTU in scenario 2 is calculated as follows:

$$\frac{4*MTU}{125us} = 75Mbps \tag{3}$$

the value of the MTU is 292.96875 bytes.

In order to obtain the end-to-end delay, T_{hop} , which represents the delay of each hop, is calculated, including forwarding delay, propagation delay and sending interval. We assume that the frame size is M bytes, and the forwarding delay of the switch is defined as 8us. As mentioned in Section II-A, the send interval time of SR- Classes is 125us or 250us,which is expressed to be ST . Then,

$$T_{hop} = \frac{M}{100Mbps} + 8us + ST \tag{4}$$

To guarantee a maximum latency of 2ms over 7 hops, 7 times of T_{hop} must be less than 2ms.

Moreover, the whole network is globally synchronous, global clock cycle is 12.5MHz, and cycle period is 4ms.

C. Scenarios

Three experiment scenarios are set up to simulate traffic conditions of different traffics in in-vehicle network.

- Scenario 1: Because the driver assistance camera traffics and control information are more time-sensitive, node 1 and node 2 use SR-A to transfer. Meanwhile, node 3 and 4 transfer audio traffics and video traffics by SR-B, because they are relatively less time-sensitive. Node 5 generates best effort traffics which has the lowest priority.
- Scenario 2: Except node 5 generates ordinary best effort traffics, node 1,2,3,4 use SR-A to transfer traffics.
- Scenario 3: Time-triggered scheduling is used by node 1 and node 2 to transfer the driver assistance camera traffics and control information. Because they are more sensitive to the determinacy of transmission time. Meanwhile, node 3 and 4 transfer audio traffics and video traffics by SR-A, and node 5 generates best effort traffics.

IV. Results Analysis

Based on the previous model, experiments have been done with various MTU . End-to-end delay and jitter are measured to see whether the QoS is met.

First of all, experiments with the payload(i.e. MTU) setting separately to100 bytes and 250 bytes were all done to verify the QoS in scenario 1and 2. In both cases, node 5 generates best-effort traffic frames of 1500 bytes at sending interval time of uniform distribution (2ms, 3ms), and broadcasts them to the rest of the nodes. Node 6 is a listener, receiving all the traffics from other nodes. The difference lies in that different SR-Classes are used by node 1,2,3 and 4. We count the end-to-end delay of all types of traffic in table I and II.

TABLE I. End-to-end Delay of Various Traffics in Scenario 1

Traffic	Size(Byte)	Max[us]	Mean[us]	Min[us]
Driver Assistance Camera	100	978.22	761.08	457.8
	250	1056.92	851.15	521.4
Control Information	100	980.58	765.71	519.8
	250	1054.45	849.42	583.72
Audio	100	1117.47	806.64	491.08
	250	1495.46	1132.39	607.56
Video	100	1104.46	823.38	617.16
	250	1457.51	1134.91	732.12
Best Effort	1500	706.27	655.99	643.4

TABLE II. End-to-end Delay of Various Traffics in Scenario 2

Traffic	Size[Byte]	Max[us]	Mean[us]	Min[us]
Driver Assistance Camera	100	978.03	783.42	457.88
	250	1027.42	852.09	521.48
Control Information	100	985.58	782.05	488.92
	250	1037.07	853.04	563
Audio	100	982.24	783.84	519.96
	250	1025.47	853.59	604.52
Video	100	983.55	789.56	551
	250	NaN	NaN	NaN
Best Effort	1500	680.68	649.74	643.4

It is worth noting that the maximum end-to-end delay over 7 hops is far less than 2 ms, which meet the QoS of the end-to-end delay. And the end-to-end delay of nodes 1 and 2 is almost unchanged, regardless of the case of multi SR-Classes

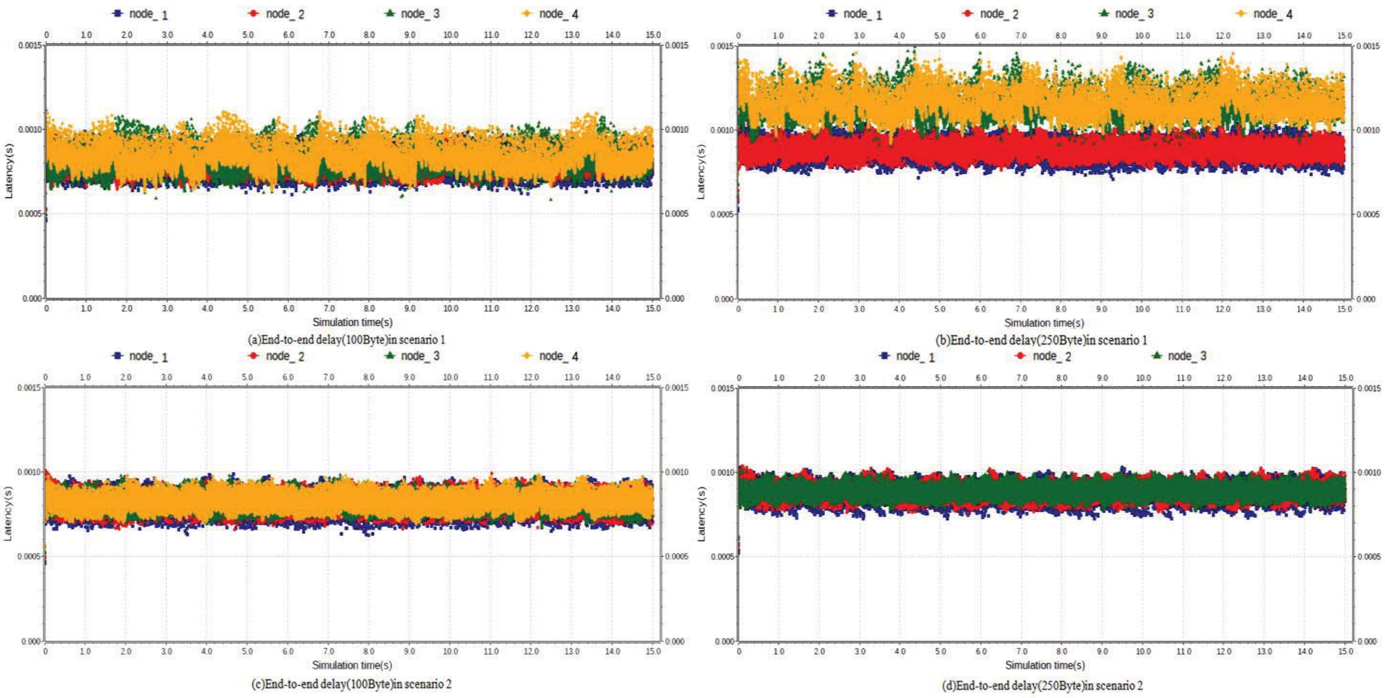

Fig. 3. End-to-end latency and jitter in scenario 1,2

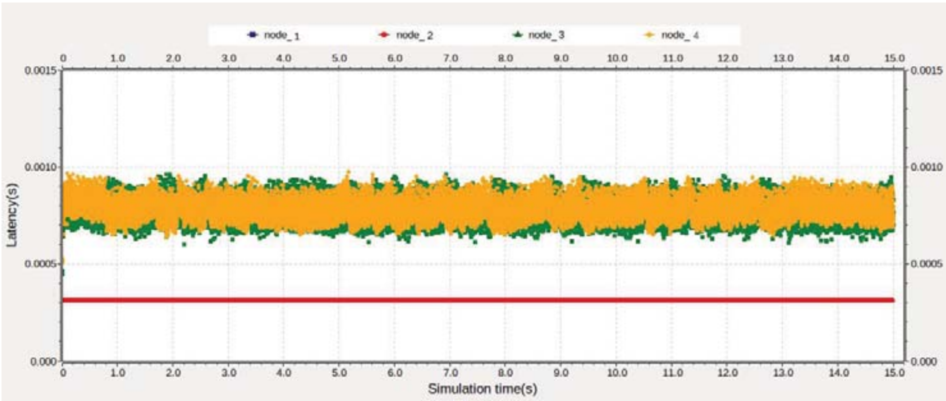

Fig. 4. End-to-end latency and jitter in scenario 3

or single SR-Class. However, the situation is quite different for nodes 3 and 4. The maximum end-to-end delay for them is reduced by almost 125us. The total end-to-end delay of scenario 1 and 2 is calculated as follows,

$$T1 = 978.22 + 980.58 + 1117.47 + 1104.46 = 4180.73us \quad (5)$$

$$T2 = 978.03 + 985.58 + 982.24 + 983.55 = 3929.4us \quad (6)$$

$T1$ represents the total end-to-end delay under scenario 1 and $T2$ represents it under scenario 2. We can come to conclusion that using only SR-A to transfer the time-sensitive traffics reduce the delay greatly and enhance the QoS of the end-to-end delay in-vehicles network.

However, node 4 can not transmit frame when payload is greater than 250 bytes under the case of only SR-A. It means that the reserved link bandwidth has all been occupied. As mentioned in Section II-A, the send interval time of SR-A is 125us, which is half of SR-B. So traffics of SR-A will occupy two times the bandwidth of SR-B when it is transmitted on the link. The reserved bandwidth will soon be depleted when all time-sensitive traffics are SR-A. In addition, the jitter is not optimistic that the difference between the maximum and the minimum of the end-to-end delay is hundreds of microseconds. Jitter brings randomness to the arrival of traffics which will have serious adverse effects on some traffic, such as engine control.

The end-to-end delay and jitter of various types of traffics are more intuitively in Fig 3. From Fig 3-(a)(b), we can see that the larger the payload, the shorter the transfer latency needed by traffics of SR-A. From Fig 3-(a)(c), we can conclude that the cases of single SR-Class is more time-saving than multiple SR-Classes. In Fig 3-(d), node 4 has no data, indicating that the reserved bandwidth has been exhausted. All the data in Fig 3 fluctuates considerably, indicating that Ethernet AVB does not have rigid jitter constraints. It is difficult to meet the QoS of jitter.

In order to reduce jitter of time-sensitive traffics, we introduced time-triggered scheduling in scenario 3. For time-triggered communication, we will configure the scheduling time window offline. These preconfigured schedules assign dedicated transmission slots to each participant, including talkers, listeners and switches. Time-triggered mechanism is a time division multiple access (TDMA) multiplexing strategy which allows for deterministic transmission with predictable delays. It prevents congestion on egress port and enables isochronous communication with low latency and jitter.Node1 and 2 use time-triggered traffic instead of SR-A. Other network devices configurations are the same as scenario 2.The results of the experiment are shown in Table III. Under the same payload, the end-to-end delay of time-triggered traffic is much less than SR-A, and almost zero jitter is achieved.

Fig 4 shows the results of the experiment in scenario 3. The blue and red data points in the diagram represent time-triggered traffics, which is almost a straight line, meaning almost no jitter. The yellow and green data points represent the traffics of SR-A, which fluctuate greatly, meaning that the jitter is high. In addition, from the Y- axis, yellow and green data points are far above blue and red data points, indicating that time-triggered scheduling can significantly reduce the end-to-end delay. Network bandwidth has also been released to a certain extent due to the introduction of TDMA, and SR-class can transmit larger payload.

TABLE III. END-TO-END DELAY OF VARIOUS TRAFFICS IN SCENARIO 3

Traffic	Size[Byte]	Max[us]	Mean[us]	Min[us]
Driver Assistance Camera	100	315.64	315.04	314.44
	250	455.65	455.03	454.4
	1500	1075.78	1075.1	1074.37
Control Information	100	315.65	315.06	314.39
	250	455.72	455.03	454.38
	1500	1105.8	1105.12	1104.36
Audio	100	963.48	747.96	448.04
	250	1086.42	846.49	521.48
	1500	1169.01	854.94	690.94
Video	100	975.07	752.09	510.12
	250	1065.12	846.26	584.2
	1500	1162.86	859.9	728.72
Best Effort	1500	1049.55	676.49	643.4

V. CONCLUSION & OUTLOOK

AVB is one of the promising technologies for a real-time Ethernet based in- vehicle networking infrastructure. Dynamic stream reservation and credit based shaping algorithm guarantee end-to-end latency of 2ms over 7 hops. But for the rigid requirements of jitter, it is insufficient. As the performance of time-triggered traffic in Ethernet is already well known, we improve the AVB model by using only SR-A and time-triggered scheduling and analyze the effects of concurrent synchronous and asynchronous traffics. Our results show that the new model can achieve low jitter under the QoS of end-to-end latency.

Although the present model have used time-triggered scheduling, the problem of too large frames invading the time windows of adjacent traffics still exists, which is a serious harm to time-sensitive traffic. Some measures need to be taken to prevent this from happening. In future work, we plan to extend the model with time aware shaper and frame preemption, as proposed in IEEE 802.1Qbv. 802.1Qbv is one of the important features of Ethernet AVB II(i.e. Time-Sensitive Network, TSN). TSN is considered to be the unified Ethernet backbone in the vehicular network industry.

ACKNOWLEDGMENT

This work is supported by the National Science Foundation of China(Grant No.61533015).

REFERENCES

[1] IEEE 802.1 AVB Task Group, “IEEE 802.1 Audio Video Bridging.”[OL].Available:http://www.ieee802.org/1/pages/avbridges.html.

[2] OMNeT++ Community, “OMNeT++ 5.0.” [OL]. Available: https://www.omnetpp.org/.

[3] IEEE 802.1 AVB Task Group, “IEEE 802.1AS-Timing and Synchronization for Time-Sensitive Applications.” [OL]. Available: http://www.ieee802.org/1/pages/802.1asbt.html.

[4] IEEE 802.1 AVB Task Group, “IEEE 802.1Qat- Stream Reservation Protocol.” [OL]. Available: http://www.ieee802.org/1/pages/802.1at.html.

[5] IEEE 802.1 AVB Task Group, “IEEE 802.1Qav- Forwarding and Queuing Enhancements for Time-Sensitive Streams.” [OL]. Available: http://www.ieee802.org/1/pages/802.1av.html.

[6] Tuohy S., Glavin M. and Hughes C. Intra-Vehicle Networks: A Review[J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16(2):534-545.

[7] Lim H T., Herrscher D. and Chaari F. Performance comparison of IEEE 802.1Q and IEEE 802.1 AVB in an Ethernet-based in-vehicle network[C]// International Conference on Computing Technology and Information Management. IEEE, 2012:1-6.

[8] Moon J C., Bo M C. and Park J D. Comparison of IEEE802.1Q and IEEE802.1AVB in in-vehicle network implemented in embedded system[C]// IEEE International Conference on Information and Automation. IEEE, 2015:1491-1496.

[9] Li S K., Park C. and Park S K. Design of single stream reservation class for efficient data transmission in IEEE 802.1 AVB networks[C]// IEEE, Malaysia International Conference on Communications. IEEE, 2016:135-138.

[10] Steinbach T., Lim H T. and Korf F. Tomorrow's In-Car Interconnect? A Competitive Evaluation of IEEE 802.1 AVB and Time-Triggered Ethernet (AS6802)[C]// Vehicular Technology Conference. IEEE, 2012:1-5.

[11] Alderisi G., Caltabiano A. and Vasta G. Simulative assessments of IEEE 802.1 Ethernet AVB and Time-Triggered Ethernet for Advanced Driver Assistance Systems and in-car infotainment[C]// Vehicular NETWORKING Conference. IEEE, 2012:187-194.

[12] IEEE 802.1 TSN Task Group, “IEEE 802.1Qbv-Enhancements for Scheduled Traffic.” [OL]. Available: http://www.ieee802.org/1/pages/802.1bv.html.

[13] Communication over Real-time Ethernet Group, “CoRE4INET Framework.” [OL]. Available: http://core.informatik.haw-hamburg.de/.