Packet Recovery Method with Redundant Packets for Large Spatio-Temporal Data Retention*

Naoki Tanaka[1], Daiki Nobayashi[1], Kazuya Tsukamoto[1], Takeshi Ikenaga[1]
[1]Kyushu Institute of Technology, Fukuoka, Japan
email: tanaka.naoki845@mail.kyutech.jp, {nova@ecs, tsukamoto@csn, ike@ecs}.kyutech.ac.jp

***Abstract*—Cyber Physical System (CPS) is a system that accumulates information from physical space, analyzes them in cyberspace, and then feeds back the results. We aim to realize a Floating Cyber Physical System (F-CPS), a regionally distributed CPS. We have proposed a large-capacity spatio-temporal data retention system for delivering data and function (application) in the F-CPS. In a previous work, we proposed a data completeness-aware transmission control and evaluated it by simulation. However, considering the movement of obstacles and fluctuation of radio waves in the real environment, packet loss occurs frequently. Thereby, the system may not be able to have complete data and may not function. In this paper, we proposed a method to recover missing data using redundant packets and evaluated its impact on the entire system by testing it with actual equipment.**

***Index Terms*—Large Spatio-Temporal Data，Data Retention，Local Production and Consumption of Data，F-CPS**

I. Introduction

Cyber Physical System (CPS) is a system that accumulates information from physical space, analyzes them in cyberspace, and then feeds back the results. Generally, the CPS has challenges caused by the centralized system, such as limited wireless resources and the cost of maintaining cloud systems. To solve the challenges, we aim to realize a Floating Cyber-Physical System (F-CPS), a regionally distributed CPS. F-CPS focuses on Spatio-Temporal Data (STD), such as traffic and weather information, depends on the location and time of data generation. In F-CPS, edge devices with computing resources near the user, such as smartphones, distribute STD as a regional information hub (InfoHub). Furthermore, in the F-CPS concept, region-specific services also retain as functions (applications) within the area similar to STD. F-CPS can solve the CPS challenges because device edges can build F-CPS.

In privious works [1] [2], we have proposed a large spatio-temporal data retention system (LSTD-RS) for distribute large spatio-temporal data (LSTD), such as images, videos, and functions for realizing F-CPS. In LSTD-RS, edge devices are also regarded as Infohub (hereafter, referred to simply as nodes). Nodes in an area divide the LSTD into multiple packets and broadcast them to retain in the area. Nodes operate according to cycles. Fig. 1 shows LSTD-RS. When an accident occurs in the area, the source nodes divide the LSTD about the accident into multiple packets and broadcast at the first cycle. From the second cycle, nodes broadcast periodicaly to spread and retain in the area. Users in the area can passively receive and utilize the LSTD. In this system, it is necessary to minimize the number of transmissions while delivering LSTD to many more users. In our previous work [2], we proposed a data completeness-aware transmission control for LSTD-RS. The source node of LSTD transmits the LSTD including the sequence number and the total number of packets. The each node that received LSTD judges the LSTD completeness based on the information and transmits the LSTD. This control prevents incomplete data retention and, thus a decline in the wireless environment.

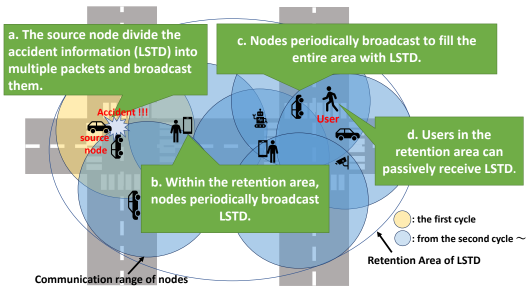

Fig. 1. Large spatio-temporal data retention system.

However, in the real environment, packet loss may occur due to the effects of moving obstacles and fluctuating radio waves, and it may not be possible to provide complete LSTD. In [3] [4], Forward Error Correction (FEC) techniques have been studied to solve packet loss using redundant packets. To mitigate the effects of packet loss, we propose missing data recovering method using redundant packets for LSTD-RS. Additionally, we introduce interleaving into the method to prevent data recovery failures in environments where burst packet loss occurs.

II. Proposed method

In the previous work [2], each node transmitted LSTD only when the complete data was received. Fig. 2 shows each node's action of the proposed method . The source node creates a redundant packet by using the XOR of the divided original data, adds it to the original data, and then transmits them. If nodes receive an incomplete LSTD, the nodes recover the missing data using redundant packets. The nodes that were able to retain the complete LSTD transmit the LSTD.

In addition, where burst packet loss occurs, nodes may be difficult to retain complete data because redundant packets are

*This work was supported in part by the commissioned research JPJ012368C05501 by NICT, JAPAN

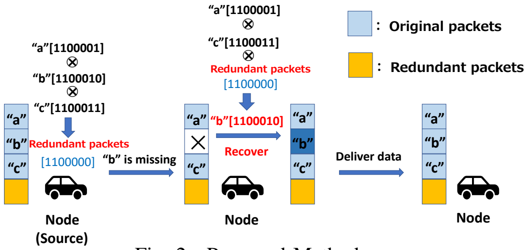

Fig. 2. Proposed Method.

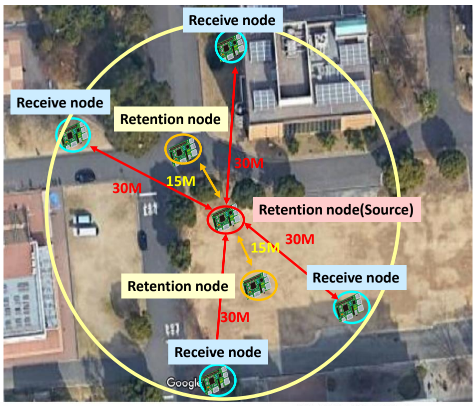

Fig. 3. Experimental Environment.

lost as well. To prevent data recovery failures due to burst packet loss, we introduced interleaving into the missing data recovery method. The source node reorders the original data and redundant packets separately and sends them. The each node restores packets to the original order and recovers.

III. Experimental Evaluation

We experimented in the Tobata campus of Kyushu Institute of Technology. We set up seven fixed nodes using a Raspberry Pi 4 Model B as shown in Fig. 3. The retention node operates data retention, and the receive node only receives to evaluate the retention.

In the experiments, we measured the retention of four cycles. In the first cycle, the source creates redundant packets from the divided original data and broadcasts them to surrounding nodes in the area. The original data is consisted from 24 packets of 1000 bytes each. In the second through fourth cycles, two retention nodes, excluding the source, transmit data by broadcast once in each cycle after the nodes could retain complete data. All nodes receive the data and recover the missing data. If the node cannot retain complete data, it discards data. We defined the ratio of redundant packets to total packets as redundancy, and measured by varying the redundancy from 1/3 to 1/13. The previous method was defined as redundancy 0. We evaluated the effectiveness of the proposed method in the following items. The results are an average of 30 measurements of the four cycles of retention.

We defined the complete data retention rate of how many of the four receiving nodes were able to receive complete data. Fig. 4 shows the rate of the fourth cycle, and Fig. 6 shows the rate of the all cycles how the complete data reception rate is changed with the increase in the redundancy ratio. In Fig.

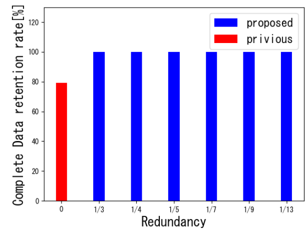

Fig. 4. Complete data retetion rate (The fourth cycle)

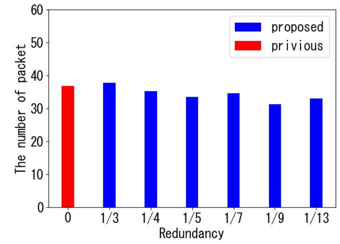

Fig. 5. Average number of required packets to retain the LSTD.

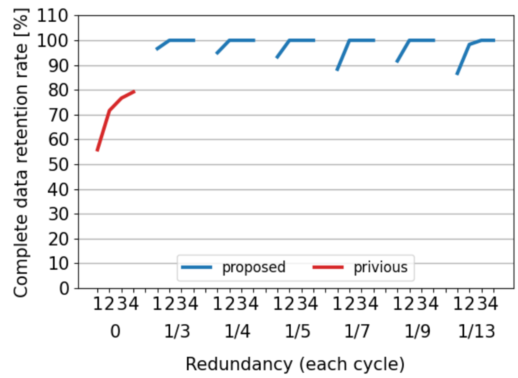

Fig. 6. Complete data retetion rate (The all cycles)

4, the proposed method improves the complete data retention rate from 79.17% to 100%. The results show that many nodes can retain the complete data through all cycles. In Fig. 6, the proposed method improves the complete data reception rate from 55% to around 90% in the first cycle. The results show that the proposed method allows nodes to retain complete data in the earlier cycles.

Fig. 5 shows the number of packets required to retain the complete data for the receive nodes. The number was calculated as the total packets that the retention nodes transmitted until the cycle when the receive node retained the complete data. The results show that the proposed method can reduce the number of packets required to retain complete data by setting the suitable redundancy (1/9) in this environment.

IV. Conclusion

In this paper, we proposed a packet recovery method with redundant packets for LSTD-RS. The proposed method allows many nodes to retain complete data in earlier cycles and to reduce the total number of packets required. In future work, we will study methods that can determine the appropriate redundancy for the environment.

References

[1] D. Nobayashi, et al., "Location-Based Autonomous Transmission Control Method for Spatio-Temporal Data Retention System," In Proc. Of BWCCA 2022, Oct. 2022.

[2] H. Kaneyasu, et al., "Data Completeness-aware Transmission Control for Large Spatio-Temporal Data Retention," ICCE 2022, pp.1-5, Jan 2022.

[3] M.Luby, et al., "Forward error correction (FEC) building block, "IETF RFC3542, Dec.2002.

[4] V.Sharma, et al., "Complementing TCP congestion control with forward error correction," Proc. IFIP Networking2009, pp.378–391, May 2009.