# Design of Buffer Management for Industry to Avoid Sensor Data' Conflicts

Dae-ho Won, Jong-wook Hong, Yeon-Mo Yang, and Jinung An

***Abstract*—**To reduce accidents in the industry, WSNs(Wireless Sensor networks)' sensor data is used. WSNs' sensor data has the persistence and continuity. therefore, we design and exploit the buffer management system that has the persistence and continuity to avoid and delivery data conflicts. To develop modules, we use the multi buffers and design the buffer management modules that transfer sensor data through the context-aware methods.

***Keywords*—**safe management system, buffer management, context-aware, input data stream

## I. Introduction

IN Many plants and factories, accidents are steadily increasing. So, these have happened many deaths and injuries. This fact expresses that the attention has been shown to the question of the safety management.

This system, the safe management system, is preventing system that is aware of the context through many wireless sensor nodes' information.

The Wireless sensor nodes' sensor data generally has special features that they acquire and delivery real-time information unlike general data information. Sensor nodes have low cost, low power, persistence and are placed in the plants and factories. The nodes start to transmit sensed data values to neighbor nodes or sink node.

There is a possibility of conflicting each data because much data information in real-time drops in the system. A possibility of conflicting each data increases and the important data information substantially can't access the top monitoring level. Finally, a possibility of accident prevention decreases. Consequently, in this paper, to reduce conflicting between each data, we design buffer management module to manage sensor data and evaluate performance.

The remainder of the paper is organized as follows. In Section 2, we give a short survey of previous works with buffer management system. In Section 3, we describe the general industrial system architecture and proposed system architecture. In Section 4, we evaluate the performance analysis of proposed system's module. Finally, Section 5 outlines the main conclusions.

## II. RELATED WORK

Recently, a number of publications have dealt with methods to avoid conflicting each data and to reconstruct data types.

In Floyd et al.(1997)[1], used buffer's average value in the internet environment and control crowded data. If the Packet's value exceed average value, the system lose buffer. That method common control crowed data in the internet environment.

Dae-ho Won is with Kumoh University, Gumi, Korea (phone : +82-10-4493-0409, e-mail : wdh10828@naver.com)

Jong-wook Hong is with Daegu Gyeongbuk Institute of Science and Techonology (DGIST), Daegu, Korea (e-mail : jwhong@dgist.ac.kr)

Yeon-mo Yang is with Department of Electronic engineering , Kumoh University, Gumi, Korea ( e-mail: yeon-mo yang@vivaldi.kumoh.ac.kr)

Jinung An is with Daegu Gyeongbuk Institute of Science and Technology (DGIST) , Daegu, Korea (Corresponding editor, phone: +82-53-430-8509, fax: +82-53-430-8599, e-mail: robot@dgist.ac.kr)

Shigang et al.(2006)[2], used 1/k-buffer method to avoid congested and conflicting each neighbor node in the wireless network environment. The value of K is a number of using extra buffer and increasing or decreasing according to exceed buffer' value. This method is a level to treat data in the wireless network environment, but this method is unsuitable our proposed system. Our proposed system treats sensor data to avoid conflicting data, is not communicated with each neighbor node.

Soo Han el al.(2008)[3], used multi-buffers to deal with a high speed data of RFID system. This system use increasing or decreasing a number of buffer along the input data velocity. But if the system appears temporarily to increase a number of the buffer, it causes to process system slowly.

## III. System Architecture

### *A. The overall system structure*

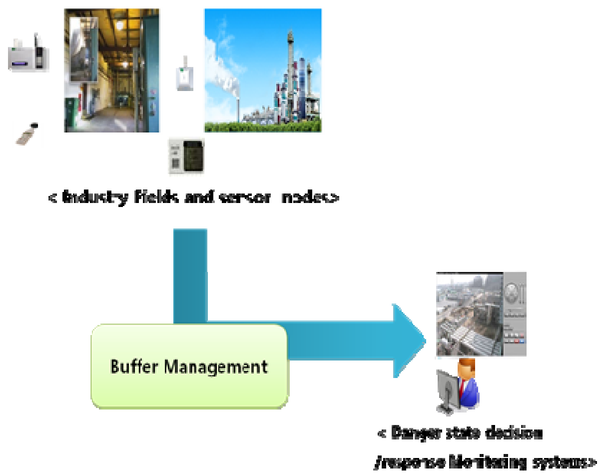


Fig. 1 The Industrial monitoring system architecture

Fig. 1 shows the overall system structure of the danger decision correspondence in the industry. First, there are various sensor nodes to predict danger context in the industry. Values of the sensor data have a proper data structure as numbers of the industry accidents. Second, data move to our proposed buffer management system. They have a priority by the data policies. If they have a correct priority, system has stable. There is not conflicting data. Finally, data move to the danger context decision system and the system gives the alarm to users.[4]

### *B. The buffer management system structure[5]*

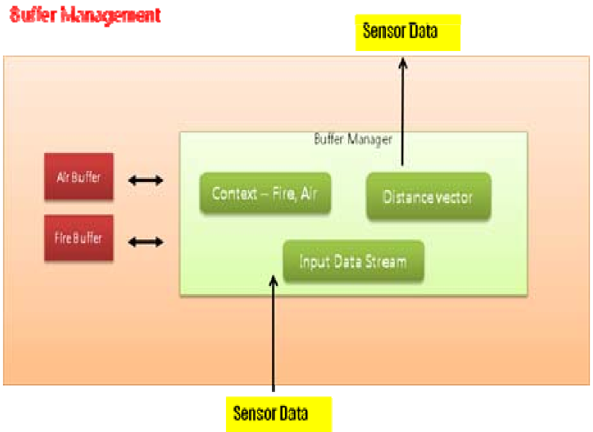


Fig. 2 The Buffer management system architecture

Fig.2 shows that we propose the buffer management system's structure. In this paper, our proposed buffer management system has next following properties and data policies, sets up a priority.

a. The input data stream : The input data stream means that the incoming velocity of the sensor data. The buffer manager checks the incoming velocity of the sensor data, because the incoming velocity is influenced by a number of the sensor data. Eq.(1) shows that the incoming velocity of the sensor data. In Eq.(1), Si means the incoming velocity, S means the Sensor data stream, and Sn means that a number of the sensor data's incoming. In Eq.(1), if a number of data increase, the incoming velocity also increase. Consequently, we control the incoming velocity and reduce a possibility of the confliction.

$$S_i = \frac{\sum_{n=1}^{S_n} S}{t} \qquad (1)$$

b. Context – Fire, Air: In this paper, we propose the second proper that the buffer manager controls data through a context-aware module. The purpose of our designed system processes various accidents such as Fire, Explosion, Air pollution. To handle quickly various contexts, we set up a context-aware module to prepare emergency. Statistically, in this paper, we propose contexts about Fire and Air pollution.

c. Distance vector : Third proper, we use the data structure and set a priority. Table. 1 shows that we propose data structures. One is Fire structure, another is Air pollution data structure.

$$FireCOST(i) = \sqrt{\alpha \times temp_i^2 + \beta \times humid_i^2 + \gamma \times o2_i^2} \qquad (2)$$
$$AirCOST(i) = \sqrt{\alpha \times co2_i^2 + \beta \times no2_i^2 + \gamma \times so2_i^2 + \delta \times nh2_i^2 + \varepsilon \times o2_i^2}$$

In Table. 1 and Eq.2, we use data policies. It defines 'distance vector'. It has a turn as follows. First, we know each data structure' value. Next, we get that each data values square and all of values plus and calculate root-square. This means that we get the average distance which shows a small calculation. A small calculation means that system needs not to operate impractical.

TABLE I
THE FIRE – AIR SENSOR DATA STRUCTURE

```
enum FIRE_V
{
    FIRE_SENSOR_ID = 0,
    FIRE_ARRIVAL_DATE,
    TEMP,
    HUMID,
    FIRE_O2,
    ACT_FIRE,
    FIRE_CO2,
    DEG_FIRE,
    HIGH_TEMP
};
```

```
enum AIR_V {
    AIR_SENSOR_ID = 20,
    AIR_ARRIVAL_DATE,
    CO2,
    No2,
    SO2,
    NH3,
    O2,
    GAS
};
```

*C. The buffer management system's efficient algorithm*

In Table.2, shows a algorithm of the buffer management system. First, we compare the incoming velocity about a number of data (line 1). Next, we select a context-aware value to the incoming velocity.(line 2)

TABLE II
THE BUFFER MANAGEMENT ALGORITHM

| **Algorithm** |
|---|
| **1:** **if** DataStreamV_Weight is high value **then** |
| **2:** Select Context_Values. |
| **3:** **if** data is Context_WarnigValue **then** |
| **4:** maximum priority |
| **5:** **else** |
| **6:** calculate distance vector |
| **7:** **if** distance vector is high **then** |
| **8:** maximum priority |
| **9:** **else** wait in the context_Buffer |
| **10:** **else** select context_WaringValue. |
| **11:** **end if** |

we classify according to value of the context condition. And we give a maximum priority that the data value excceds basic context warning value to data.(line 3~4). Extra data that deal with a priority move to next data policy that is a 'distance vector'. This stage, we calculate and compare data values. The sensor data that get a high value moves to a priority step. The last sensor data wait for buffers, such as Fire buffer or Air buffer.

The values of the context-aware and context-warning are different values according to the system environments.

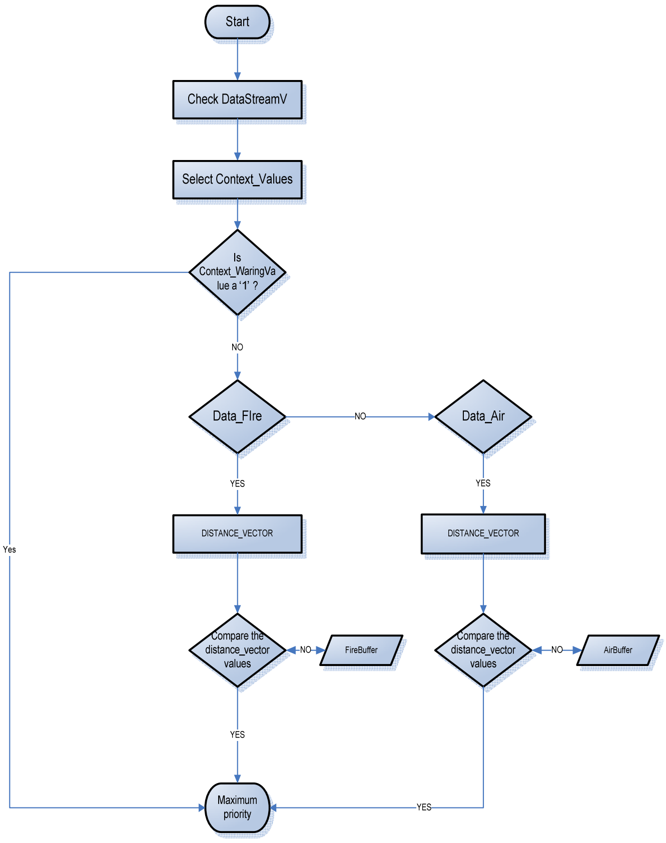


Fig. 3 The Data flow chart

In Fig.3, shows data flows that were embodied in the algorithm of the buffer management system. We can make use of this flow chart for exploiting the main system module. Next section, we evaluate the performance of the proposed module.

## IV. Simulation

In this paper, we evaluate the performance of data processing time according to quality of the module. Basically, we assume, First, the data creation is 2 seconds. Second, we assume that producing data 10, 100, 1000, 10000, 100000, 1000000, for 2 seconds and measure it. Third, values of data are not real values. So we assume values of an arbitrary data.

Fig. 4 and Table. 3 show the simulation result. We know the proposed model and the general model's values too much alike. But, the last value shows the increasing sudden data could not perform data processing. Also, if numbers of data are small, the general model should perform data processing fast. But, finally, our proposed model is a proper system because the industry sensors continue to send millions of data and add another data policy. So, we predict it is a development possibility to improve the system performance. And we design a flexible system because we add a data policy and it handle millions of data.

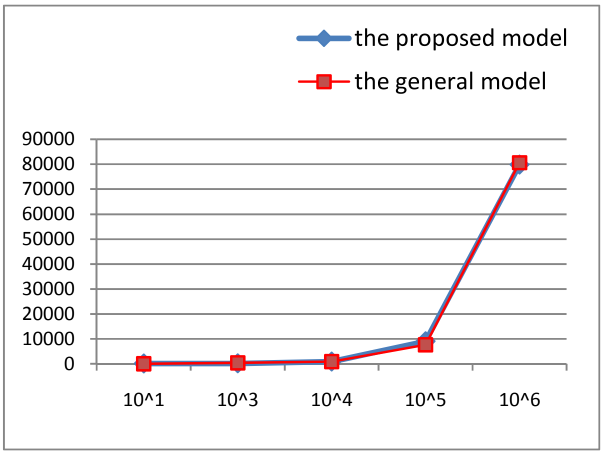


Fig. 4 The simulation result

TABLE III
TIMES AS A NUMBER OF DATA

| Numbers of data | Times(m sec) | |
|---|---|---|
| | The proposed model | The general model |
| 10^1 | 0.19 | 0.05 |
| 10^3 | 0.23 | 0.40 |
| 10^4 | 0.97 | 0.94 |
| 10^5 | 9.08 | 7.69 |
| 10^6 | 79.88 | 80.60 |

## V. Conclusion

In this paper, we have proposed the buffer management system to access the danger decision system and to avoid confliction. We design the buffer management system and evaluate the performance of the module.

Next, we use the designed system model, proposed algorithm, the performance of the main module and exploit the system and reach the perfection. We perform and evaluate our proposed system in the industry. Lastly, we continue to develop the system.

## Acknowledgment

This was supported by the USN R&D Program of MKE/KEIT. (2009-67, 10033643, Development of Core Technology based on USN for Safety Management of Industrial Fields).

## References

[1] S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang, "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing", IEEE/ACM Transactions on Networking, December. 1997
[2] S. Chen, N. Yang, "Congestion Avoidance Based on Lightweight Buffer Management in Sensor Networks", IEEE Transactions on Parallel and Distributed systems", Vol. 17, NO.9, September. 2006
[3] S. Han, S. Park, and S. Shin, " A High-Speed Data Processing Algorithm for RFID Input Data Stream Using Multi-Buffer", KIISE, March.2008

[4] M. Lee, Y. Lee, S. Hyun, I. Ko, and W. Jo, Ubiquitous Smart Space Context Aware Computing Technology, Seoul, CA: Jinhan M&B, 2009, pp.51-70

[5] J. Lee, S. Lim, J. Kim, W. Jo, " Design and Implementation of Monitoring System on Context – aware Middleware In Ubiquitous Environment", KIISE, Jun. 2007.

[6] http://ww.cs.brown.edu/research/aurora

[7] J. Park, H. Jo, " Semantic Load Shedding for Prioritized Continuous Queries over Data Streams", KIISE, April. 2005

[8] E. M. W. Kolb, J. Peck, S. Schoening, and T. Lee, "Reducing Emergency Department Overcrowding Five Patient Buffer Concepts In Comparison",IEEE, Proceedings of the 2008 Winter Simulation Conference, 2008.