The Profile-based Data Processing Method Using Wireless Sensor-actuator Networks

TaeWook Heo, Jong-Arm Jun, Nae-Soo Kim
IoT Architecture Research Team
Electronics & Telecommunications Research Institute
Daejeon, South Korea
e-mail: {htw398, jajun, nskim} @etri.re.kr

Hoon Choi
Dept. of Computer Sci. & Eng.
Chungnam National University
Daejeon, South Korea
hc@cnu.ac.kr

***Abstract*—Wireless sensor and actuator networks (WSANs) bring many gains to smart building systems. When a control system is unified by a WSAN, and particularly if the network size is wide, a distributed communication and a control method are necessary. But, multi-hop communication and packet sizes among sensors and actuators cause challenges in making such systems. This paper proposes and evaluates a new profile based data processing scheme for a smart building system with WSANs. Experimental results show that the proposed method effectively achieves the reduction of packet numbers and sizes with self-controlled sensors and actuators. We also discuss how to dynamically extend the service of a WSAN with only profile distributions and updates.**

Keywords-Distributed networks; profile based data processing; profile distribution; wireless sensors-actuator networks; (WSANs).

I. INTRODUCTION

Wireless sensor networks (WSNs) are made by small-sized, low-cost, and wireless communications enabled sensors, which have been installed to form various monitoring systems, e.g. a building environment monitoring system [1].

Wireless Sensor Networks (WSNs) that are used by a centralized way gather the sensing information and perform commands through a central server [2]. Therefore, this method has several disadvantages, e.g., poor response time, heavy network traffic, and centralized bottleneck. However, distributed wireless sensors and actuators close to each other distance and the actuators have a decision making algorithm, these have several strengths, e.g., the data traffic balance and reduction, the battery usage reduction, and the network lifetime increase [3-4].

This paper proposes a new profile based data processing method. There are separated the procedures which have the combinations of events and event processing conditions. And adding an actuator to change the service can be performed dynamically.

For example, when the actuator with ventilation service joins the network, the actuator operation is generated by a combination of events for the ventilation task. Also, the actuator with fire alarm service joins the network, the actuator operation is provided by a union of events for the fire monitoring work.

Now, the server provides events conditions and the combination of events for the frequent changes of the service. Additionally, the response time is fast and it is controlled effectively despite disconnection of the central server.

The problems of the existing systems are the applications-oriented approach. Whenever there are changes in the service they should provide and develop the appropriate applications. These points have been raised as a problem. The development of a service and configuration of it which fit the application are more resources intensive.

Therefore, in this paper we introduce to handle easily commands which execute, change, and update the services. Also the protocol for data processing is introduced. Also, the distributed WSAN are more energy efficient than the centralized WSAN.

Section II discusses the related work. Section III proposes the profile-based data processing scheme, and the Section IV discusses the experimental results. Section V shows the conclusion and future work.

II. RELATED WORK

Smart building systems are used for controlling the environment of smart buildings, such as temperature, humidity, illumination, by means of ventilating, lighting, fire monitoring, and so on [5].

Recently, centralized and distributed methodologies have been studied by various researchers. Among them, a distributed control method is designed and compared with a centralized control method in [6-7].

The traditional methods for modelling and controlling building environment system may become impractical when the control-system loops are closed by the WSANs, in which unreliable and incomplete data and network attributes, such as network traffic, should be paid enough notice [8].

At the sensor nodes in a centralized method, the data are collected from the gateway using WSAN are sent to application servers that are located outside. Therefore, the event collected by the application server is detected and actuators that can handle the event are received to the control command.

In this way, because of all data-intensive to the applications server, there are the prolonged data-path weaknesses.

 ISBN: 978-1-61208-207-3

Also, the shortcomings are that the nodes in the neighbourhood of the application server consume more energy than the other nodes.

In this study, to solve these shortcomings, the self-control method between the sensor nodes are generated through the various sensors and actuators that can handle the data are able to process the events through the autonomous connection.

In WSAN technical adaptation of the various sensing data to be sent to the server, WSAN lifetime (time to provide services) will have the advantage of extending by reducing the power consumption of nodes to deliver a control command from any node in the path.

III. WSAN PROFILE BASED DATA PROCESSING METHOD

A. Centralized WSAN and distributed WSAN

For the centralized WSAN, denote N as the number of nodes, R_b the channel bit rate. Then we define a factor, α_A, taking account of the overhead presented by all protocol stack layers. We consider a WSAN where nodes are requested to send their samples (composed of D bytes each) taken from the monitored space every T_R seconds. Here, we can write:

$$N \leq R_b \alpha_A T_R / (8D) \quad (1)$$

This equation offers an approximate estimation of the number of nodes that can be part of a single-sink single-hop WSAN.

For the distributed WSAN, there are multi-sink multi-hop networks. Let us denote by h_m the average number of hops per data sample taken from the field.

We can assume that each sink (denoting as N_s their overall number in the network) can serve up to N nodes with N limited by expressions. Therefore, we can write:

$$N \leq N_S R_b \alpha_A T_R / (8Dh_m) \quad (2)$$

To give a numerical example, assume R_b=250Kbits/s, TR = 10ms, α_A = 0.1, D = 3; then, if there are N_S = 5 sinks in the network, the maximum number of nodes is approximately 50. But, for the single pan, N cannot exceed 10[9].

The proposed technology using the profile is able to change the service at runtime and to modify the events conditions dynamically. Therefore, there are profile distributions, profile executions, and profile updates.

B. Profile distributions

First, the profile distributions are about how to deliver the boundary event conditions and the combination of sensor events from the server to the node.

In general how to distribute event condition and data for each node is sent to individual nodes.

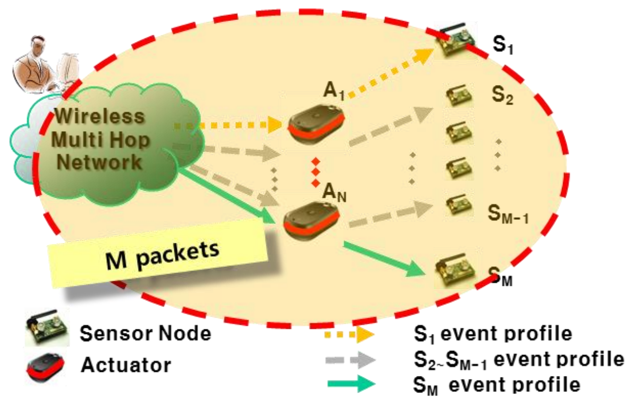
In the process of sending M packets it is about how to send an event to sensor nodes. For example, the condition of sensor node is sent to the sensor node. You can set the event condition that “If the temperature of the sensor reading is greater than 20 degrees, make reports”.

First, Figure 1 (a) shows that each packet sent to the sensor node through the route node M times individually if event profile is sent to the node and the M event profiles through the actuator are sent to the sensor nodes if it sends, or not.

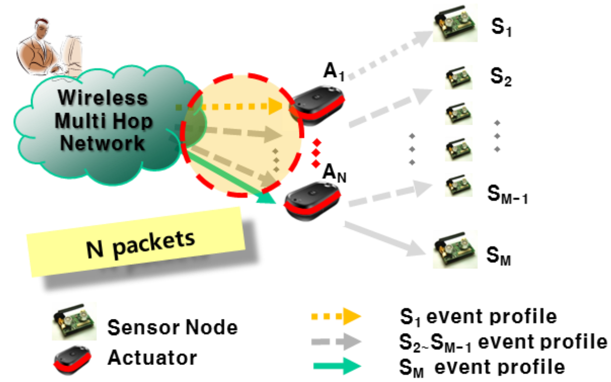
The packet-flow which is sent through the actuator is the most common. In this case, the numbers of the packets sent from network coordinator to the actuator are M.

However, the proposed method sending two messages is separated. First, the step 1 is that the N packets are sent from the network coordinator to the actuator. The step 2 is that the actuator is sending the message to the respective sensor node again. In the two steps, if you place them the same number of packet sent from the actuator to the sensor nodes, the number of packets that are transmitted to the N packets from the M packets.

Typically, the number of sensor node is more than the number of actuator node. Actuator node has the power and performs data processing and routing function.



(a) Generally profile distribution for WSN



(b) The proposed method

Figure 1. Comparison with general WSN method and the proposed method

C. Profile executions

In this paper, we propose that our framework is the logical combination and the separation of the event conditions.

Through the logical combination and union of the event conditions, the specific service can satisfy the various sensor conditions. In this session, we explain the profile based data processing method and the network information for WSAN.

In this paper, the data processing methodology is similar to the node middleware. And our profiles consist of the actuation profile and sensor event profile.

Figure 2 shows the profile based event processing sequence diagram.

The profile based event processing methods are as follows.

1. If the actuator will join the network, it sends the ReportJoin message to the coordinator.
2. The BSI(Building service interface) checks the attribute data of the actuator.
3. The BSI sends to the actuator the actuation event profile packet and the sensor event profile packet.
4. The actuator sends to the sensor nodes in the group the sensor event profile.
5. The sensor node sets the sensor event condition if it receives the profile packet.
6. The sensor node creates the event if the condition of event is met.
7. The sensor node sends the generated event to the actuator.
8. The actuator checks the event generated from the sensor node Actuation and in case the condition of actuator event profile is met, it will perform the actuation control.
9. The actuator sends the actuation event to the server.

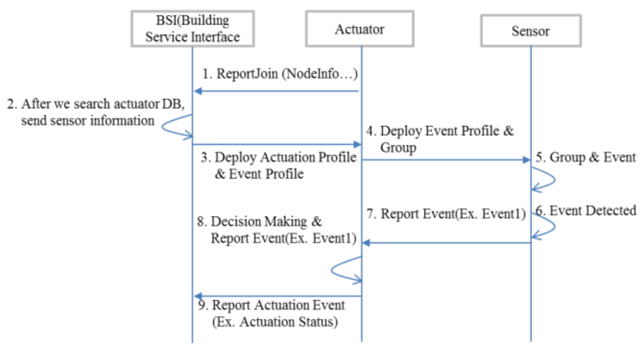


Figure 2. Profile based Event Processing Diagram

1) *Actuator Profile:*

The actuator profile is the combination of the event conditions. For the decision making in the actuator, this profile includes the actuator command that can be performed.

In Table I, there are the actuator profile structures. They are making up a combination of events.

TABLE I. ACTUATION PROFILE FRAME STRUCTURE

Name		Description	Type
Command ID		This field is command identification number (0x02)	U8int
Pay-load	Actuation Profile ID	Actuation profile identification number	U32int
	Group ID	Group identification number	U8int
	Grouping Type	Sets the bitmap to generate a group (bitmap: 0-sensor type, 1-loc, etc.)	U8int
	Sensor-Actuator Type	Sets the detailed information of sensor or actuator	U16int
	Default Control Value	Actuator's default control value (e.g. Outlet: Off(0), On(1), and so on)	U8int
	Actuation Control Value	when the combination of events is true, the actuator do the actuation control (e.g., outlet: Off(0), On(1), and so on)	U8int
	Number of Conditions	The number of event conditions	U8int
	Logical Value	Sets the logical condition value	U8int
	Condition Value	Sets the event identification as the condition value	U32bit
	Event Trigger Rate	When an event connects several nodes, we provide the threshold parameter for each node.	U8int

For example, there are three event conditions. The number of condition is 3, their condition values are {0, E1}, {2, E2}, and {2, E3}.

2) *Sensor Event Profile*

The sensor event profile structure is as follows:

For example, the event structure is A1<Sensor value<A2.

The sensor event profile describes the conditions under which an event occurs. If the condition value of the profile is met the value of the sensor, the event occurs. Table II shows sensor event profile frame structure and Table III shows condition types for the sensor event profile.

TABLE II. SENSOR EVENT PROFILE FRAME STRUCTURE

Name		Description	Type
Command ID		This field is command identification number (0x01)	U8int
Pay-load	Group ID	Group identification number	U8int
	Actuator Address	Actuator address for adapting sensor event profile	U16int
	NumberofEventID	The number of events	U8int
	EventID	Event identification for sensor event profile.	U32int
	EventType	EventType is 0 for sensing EventType is 1 for actuation	U8int
	Sensor-Actuator Type	Detailed identification for sensor and actuator	U16int
	Condition Type	Conditions between events (between, less than, greater than, outer, and so on)	U8int
	Event Condition Value 1	First parameter for condition value 1(A1)	U32int
	Event Condition Value 2	Second parameter for condition value 2(A2)	U32int
	Number of Sensor Node	The number of sensor node belonging to the group	U8int
	SensorNode Address	The address of sensor node (N-number Array)	U16int

 ISBN: 978-1-61208-207-3

TABLE III. CONDITION TYPE

Condition	0x00	A1<x<A2
	0x01	A1<x<=A2
	0x10	A1<=x<A2
	0x11	A1<=x<=A2
	0x0f	A1<x
	0x1f	A1<=x
	0xf0	x<A2
	0xf1	x<=A2
	0x88	x< A1 or x>A2
	0x98	x< =A1 or x>A2
	0x89	x< A1 or x>=A2
	0x99	x< =A1 or x>=A2
	Etc.	Reserved

D. Profile updates

Profile updates as the profile distribution are to update the combination of actuation events and the relation of the sensor event threshold conditions.

In some cases, you may cancel your profile and change it. Therefore, the profile modification with the new conditions has the advantage of dynamically changing the actuator status.

IV. DISCUSSION AND RESULTS

The profile based data processing scheme for verification the technology was applied to the actual five story building. The experimental environments used in the building are as follows:

The number of the actuators is 3, the number of the sensor nodes is 9, and the number of route nodes is 3 [10].

The emulation data is generated for applying this technology. Randomly, it generates for performing actuation.

For the real network, we used 5-story building test-bed and because the situation does not occur, we used the following emulation data.

A. Environment and Emulation data

- Precondition: a sampling interval for 24 hours is 15 minutes.

- Light sensor: The value of luminance set 1 time a day, its range is from 100 lux to 500lux.

- Temperature sensor: The value of temperature set six times a day, its range is from 10 to 55 degrees.

- CO sensor: The value of CO sensor sets to 4 times a day, its range is from 0 to 80.

- CO_2 sensor: The value of CO_2 sensor sets to 4 times a day, its range is from 0 to 2250.

In the N nodes, the packet size is as follows:

Hop counts from the coordinator to each node can be computed in the multi-hop environments. Also, there limited from each node to the actuator in wireless sensor and actuator networks.

The equation (3) represents the relationship between hop counts and the number of nodes. In the N sensor nodes, the total packet sizes from the sensor nodes to the coordinator are the greater than a minimum N and less than a maximum product of hop counts and N.

$$N \le \sum_{i=1}^{N} Hop(i) \le N \times Hop_{\max} \qquad (3)$$

where,

$Hop(i)$: the hop count of the i^{th} node from pan coordinator.

N: total number of nodes.

Hop_{max}: the greater Hop count in the Hop(i).

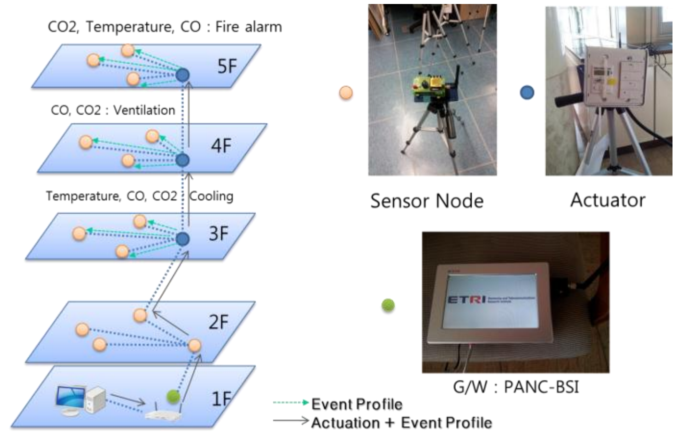


Figure 3. Event condition and test environment

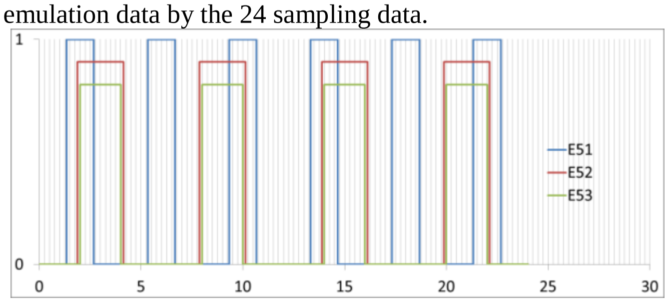
B. Packet numbers and packet sizes

In this experiment, a general centralized way and our distributed way using profile are a comparison of the predicted data.

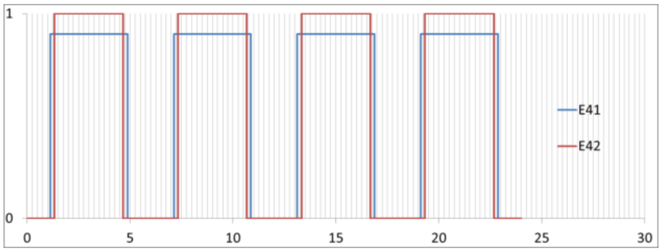
TABLE IV. EVENT PACKET ESTIMATION

Event packet numbers(5F based)		WSN	Proposed method
Sensor event packet numbers (Fire Alarm: Temperature > 40℃ and CO > 50ppm and CO_2 > 1500ppm)	Temperature sensor: 6 Hop	12*6= 72	12
	CO Sensor: 6 Hop	8*6= 48	8
	CO2 Sensor: 6 Hop	8*6= 48	8
Actuation control packet numbers	Actuator: 5Hop	8*5 = 40	8*5 = 40
	Total	208	68

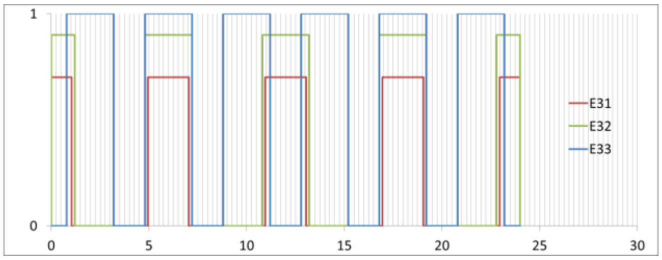
Figure 4 shows the experimental value for the event emulation. The event value of each floor (3F, 4F, and 5F) is emulation data by the 24 sampling data.



(a) Event emulation data for 5F (E51: Temperature, E52: CO, E53:CO2)



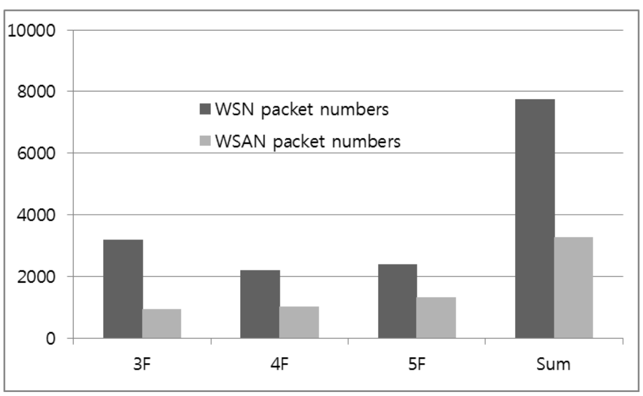
(b) Event emulation data for 4F(E41: CO, E42:CO2)



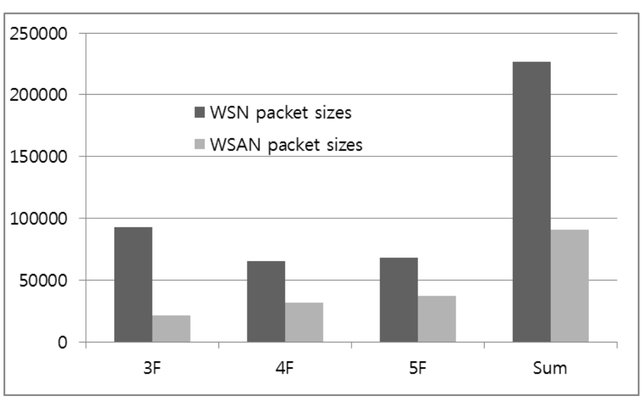
(c) Event emulation data for 3F(E31: Temperature, E32: CO, E33:CO2)

Figure 4. Event emulation data for 3F, 4F and 5F.

Figure 5 shows the experimental result for the event simulation. The packet number and packet sized of the proposed method are reduced about 50% than those of centralized method.



(a) Packet numbers for WSN(centralized) and WSAN(distributed)



(b) Packet sizes for WSN(centralized) and WSAN(distributed)

Figure 5. Test Result (data comparison central WSN and distributed WSAN for 3F, 4F, 5F a)packet numbers, b)packet sizes)

V. CONCLUSION AND FUTURE WORK

In this paper, the packet numbers and packet sizes were compared in the term of the centralized control and the distributed control. And the proposed profile-based distributed data processing technology showed good performance. Therefore we had the benefit of changing the profile easily when the service was run and the profile was distributed. It was good extensibility for the sensor network frameworks.

The profile based data processing techniques to provide a general framework in an effective way. In addition, the events that meet the criteria even more efficient in how data is processed.

The Obstacles in this paper to apply the profile to be limited service on the test adaptation and the actual test-bed is difficult.

Future work includes the accurate sensing data processing scheme such as event handling for the spatial conditions should be considered. And, the voting method is considered how to decide the truth of the event in case same multiple sensor. The way to adapt the aforementioned spatial correlation will increase the accuracy of the data.

ACKNOWLEDGMENT

This work is supported by Development of Self-Powered Smart Sensor Node Platform for Smart and Green Building [10035570] funded by the Ministry of Knowledge and Economy, Republic of Korea.

REFERENCES

[1] X. Cao, J. Chen, Y. Yang, and Y. Sun, “Development of an integrated wireless sensor network micro-environmental monitoring system,” *ISA Trans.*, vol. 47, no 3, pp. 247-255, Jul. 2008.

[2] I. F. Akyildiz and I. H. Kasimoglu, “Wireless sensor and actor networks: Research challenges,” *Ad Hoc Netw*., vol. 2, no.4, pp. 351-367, Oct. 2004.

[3] H. Ramamurthy, B. S. Prabhu, and R. Gadh, “Wireless industrial monitoring and control using a smart sensor platform,” *IEEE Sensors J*., vol. 7, no. 5, pp. 611–618, May 2007.

[4] Y.-J. Wen and A. M. Agogino, “Wireless networked lighting systems for optimizing energy savings and user satisfaction,” *in Proc. IEEE Wireless Hive Netw. Conf*., Aug. 2008, pp. 1–7.

[5] V. C. Gungor and G. P. Hancke, “Industrial wireless sensor networks: Challenges, design principles, and technical approaches,” *IEEE Trans. Ind. Electron*., vol. 56, no. 10, pp. 4258–4265, Oct. 2009.

[6] X. Cao, J. Chen, Y. Xiao, and Y. Sun, “Building Environment Control with Wireless Sensor and Actuator Networks: Centralized Versus Distributed, “ *IEEE Tran. Ind. Electon.*, vol. 57, no. 11, pp. 3596-3605, Nov. 2010.

[7] J. Chen, X. Cao, P. Cheng, Y. Xiao, and Y. Sun, “Distributed Collaborative Control for Industrial Automation With Wireless Sensor and Actuator Networks,” *IEEE Tran. Ind. Electon*., vol. 57, no. 12, pp. 4219-4230, Dec. 2010.

[8] A. I. Dounis and C. Caraiscos, “Advanced control systems engineering for energy and comfort management in a building environment – A review,” *Renew. Sustain. Energy Rev.*, vol. 13, no. 6/7, pp. 1246-1261, Aug/Sep. 2009.

[9] R. Verdone, D. Dardari, D. Mazzini, and A. Conti, Wireless Sensor and Actuator Networks Technologies, Analysis and Design. Academic Press: London, UK, 2008.

[10] IEEE-TG15.4, Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), IEEE standard for Information Technology, 2003.

 ISBN: 978-1-61208-207-3