

Data Synchronization Method in DTN Sensor Network using Autonomous Air Vehicle

Thi Ha Phuong Tran, Hiroshi Yamamoto, Katsuyuki Yamazaki

Nagaoka University of Technology

phuongtran@stn.nagaokaut.ac.jp

Abstract— While sensor network system for monitoring wild animals or observing environment changes becomes more important, collecting those observation data becomes a problem. It is because the observation field is too wide, and is usually in a forest or on a mountain. Therefore, we are studying sensor network system based on Delay Tolerant Network (DTN) using autonomous air vehicle as a data ferry. In this paper, a method of synchronizing data between the multiple sensor nodes and the server using MySQL Replication has been proposed. In the proposed system, the air vehicle does not need to carry MySQL Database, instead should carry only updated information. We have performed experiments to evaluate the practicality of the proposed system. From the evaluation result, it has been clarified that the data synchronization can be completed by our proposed system, but the time for transferring data should be decreased to prevent waste time of flying.

Keywords— Delay Tolerant Network, Autonomous Air Vehicle, AR.Drone, MySQL Replication, Sensor Network

I. INTRODUCTION

Nowadays, researches in environment and ecology have been strongly focused due to the need of observing changes of an eco-system all over the world. In this research field, a sensor network becomes a powerful tool for a long term measurement to observe and collect various sensor data. However, the observation field is too wide and is usually in a forest or on a mountain. Thus, it is difficult to transfer data between sensor nodes and a server managing the sensor data. This is because some special landscapes or planes block a radio wave of the wireless communication (e.g., wireless LAN, ZigBee)[7][8]. Furthermore, scientists of the environment and ecology cannot enter the observation field to collect data without permission of the government or a local nature conservation organization.

On the other hand, Delay Tolerant Network (DTN) has currently been focused as a solution for communication in discontinuous network such as interplanetary internet. One of the popular data transmission methods of DTN is store-carry-forward method, where a mobile node (e.g., satellite, car, truck) is used as a data ferry to collect various data from multiple data sources, store and carry the data back to the main server.

In this paper, a DTN-based sensor network for automatically collecting environment and ecology data using an autonomous air vehicle is proposed. Specifically, unlike a

bike or a car, an air vehicle can fly in the air, so its movement is not affected much by environment landscape. Hence, by using the air vehicle, the proposed system can be deployed in various environments.

We also propose a method for synchronization of data from the sensor nodes to the server through a DTN ferry, using MySQL Database Replication. The benefit of the proposed method is that it can also be executed directly in non-DTN environment, which makes the system more flexible. Furthermore, the data ferry does not need to carry the MySQL Database, hence the load on the air vehicle can be minimized.

II. RELATED STUDIES & OBJECTIVE OF THIS STUDY

A. Sensor Network in DTN Environment

As a solution for discontinuous network, there are many existing researches using DTN sensor network for communication in a disaster area, for gathering agricultural data, or collecting traffic data [9][10][11]. The resemblance of these systems is that it uses store-carry-forward model with a car or a truck carrying DTN device as a mobile node. The mobile node moves on a field to collect data under human control so the work of the data ferry is stable and safety. However, in some observation fields that do not support traditional road transportation such as in a mountainous area or a forest, it is difficult to apply these systems. Furthermore, some observation field even requires the permission of the government or the local nature conservation organization for gathering the data. Therefore, collecting sensor data in such fields becomes a problem.

B. Bundle Protocol for DTN

Bundle Protocol provides fundamental functions for constructing a store-carry-forward overlay network, which is placed on top of the current Internet protocol [3][4][5][6]. A bundle is a protocol data unit of the DTN Bundle Protocol, which contains two or more protocol data block. When connecting to other nodes, the DTN mobile node sends the bundle to them, and these nodes start finding a new node to forward data. In other words, Bundle Protocol supports one-to-one communication, rather than many-to-one communication. In case of environment and ecology sensor network, where there are many sensor nodes placed on a vast field that even communication through a wireless network can

not be established, it is necessary to design a new method that can be used for the many-to-one communication.

C. Objective of This Study

To solve the above mentioned problems, the objective of this study is to develop an automatic system for gathering data from many sensor nodes, with a method for data forwarding and synchronization in a DTN environment. Furthermore, the system must solve the problem of collecting data in some special field that a traditional road-transportation can not move easily.

III. PROPOSED SYSTEM

Fulfilling the above-mentioned purposes, an autonomous air vehicle is proposed to be used as a message ferry for the DTN-based sensor network system. Assume that this system is used for various environmental or ecology observation in a vast field such as a mountainous area, a cliff or a forest. The ferry automatically flies on the observation field to collect the data and then carries them back to the server.

A. Overview of Proposed System

The overview of the proposed system is shown in Figure 1.

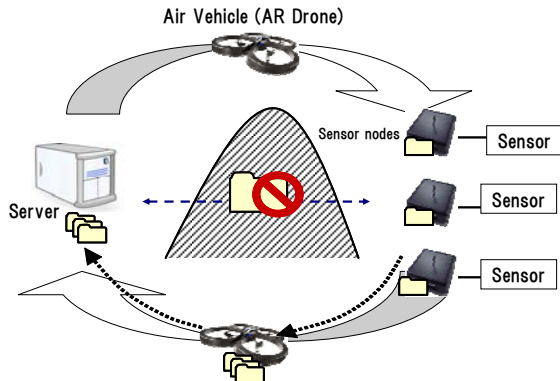


Figure 1. An overview of the proposed system

The position of the sensor nodes and the server are recorded inside the air vehicle beforehand. Using the GPS sensor, the air vehicle obtains its current location, flies to the location of the sensor node, collects data, carries the data back to the server and forwards them to the server. These functions are implemented as an application that works inside the air vehicle. When the application is executed, the air vehicle starts the automatic flight, so that it does not require human control.

The flow of data transmission from the sensor nodes to the server in the proposed system is shown in Figure 2.

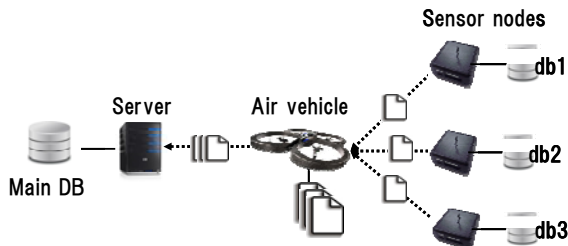


Figure 2. Overview of Data Synchronization Method

Each of the sensor nodes manages its own database. When connecting to the sensor nodes, the air vehicle obtains the necessary data, stores them inside and forwards them to the server when returning. The server uses these new data to update its database.

B. Use of Autonomous Air Vehicle for DTN Ferry

The AR.Drone, shown in Figure 3, a quadricopter developed by Parrot Co., is chosen as a DTN ferry for the proposed system [1]. With a mother board running Linux operating system (modified version of Linux kernel 2.6.27) and many sensors such as an accelerometer, gyroscope sensors, an ultrasonic altimeter, two cameras, it is a powerful and flexible device that can meet the requirements for a data ferry.



Figure 3. AR.Drone

The AR.Drone communicates with another devices by two ways: wifi(b/g), and a universal serial bus. To construct the positioning function of the system, a GPS sensor EM-406A has been attached to the AR.Drone via the serial bus, as shown in Figure 4.

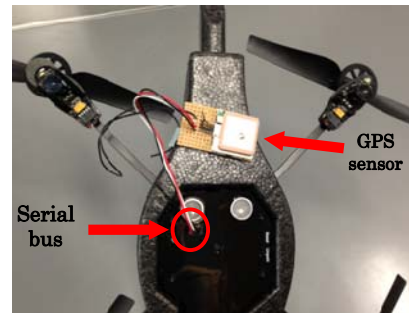


Figure 4. Attach GPS sensor EM-402A to AR.Drone

The EM-406A is a low-cost GPS sensor developed by USGlobalSat with SiRF StarIII chipset [2]. It has an extremely high sensitivity and the weight is only 16 gram, hence it can be attached to the AR.Drone without any impure on the air vehicle's flight.

C. Proposed Autonomous Flight Algorithm

The overview of the proposed autonomous flight algorithm is shown in Figure 5.

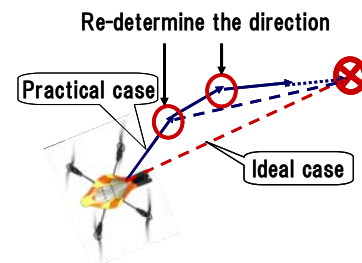


Figure 5. An overview of the proposed autonomous flight algorithm

Theoretically, it is expected that the air vehicle moves directly toward the destination. However, due to many natural factors such as the wind, the practical flight can not be stable. Hence, the air vehicle needs to check if it is going the right way, and correct its moving direction periodically.

To calculate the turning angle, as well as to detect the turning direction, we proposed an algorithm illustrated in Figure 6.

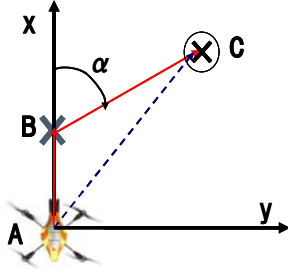


Figure 6. Method of calculating turning angle

Assume that the air vehicle is at point A and the destination is at point C. The position of the destination C is recorded inside the ferry beforehand. The front-camera direction is considered as the front side of the drone. First, the air vehicle obtains its current position information by using the GPS sensor, and then moves forward to get new position information, which means B.

Based on the location information data of three points A, B and C, the turning angle and the turning direction can be detected by Eqs.(1) and (2) as shown below.

$$\alpha = \arccos \left(\frac{x_{AB} \cdot x_{BC} + y_{AB} \cdot y_{BC}}{\sqrt{x_{AB}^2 + y_{AB}^2} \cdot \sqrt{x_{BC}^2 + y_{BC}^2}} \right) \quad (1)$$

$$d = x_{AB} \cdot y_{BC} - y_{AB} \cdot x_{BC} \quad (2)$$

$$d < 0 \Rightarrow \text{left}; d > 0 \Rightarrow \text{right}$$

These steps are repeated until the AR.Drone enters the coverage area of the wireless communication supported by the destination terminal.

D. Proposed Data Synchronization Method

To perform data synchronization from the database of sensor nodes to the database of the server, we propose using MySQL Database. This is because it supports replication method that can be used for synchronizing data [12].

In this paper, we assume to use a light weight but powerful board PC (e.g., BeagleBone Black, Raspberry Pi) as the sensor node. The board PC has a memory that can be extended by using SD-Card, hence it is possible to use MySQL Database for our system.

One of the most benefits of MySQL Database Replication is that it can be executed directly in non-DTN environment, as shown in Figure 7.

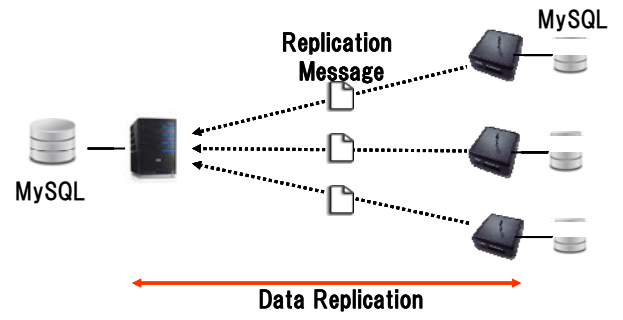


Figure 7. Database synchronization in non-DTN environment

On the other hand, in DTN environment, an air vehicle is used as a ferry that carries Replication Message (RM) files to the server, as shown in Figure 8.

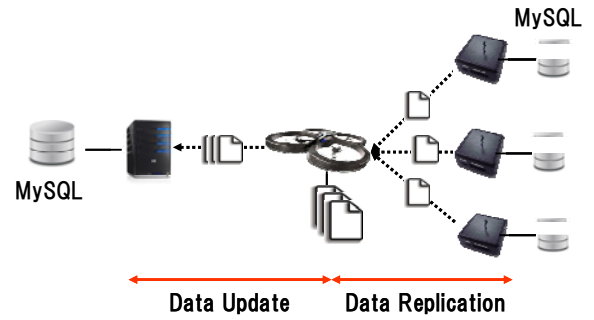


Figure 8. Database synchronization in DTN environment

In both cases, setting of Database Replication for the sensor nodes is the same, therefore, the system becomes more flexible because the sensor node do not need to care about DTN or non-DTN.

The proposed data synchronization is shown in Figure 9.

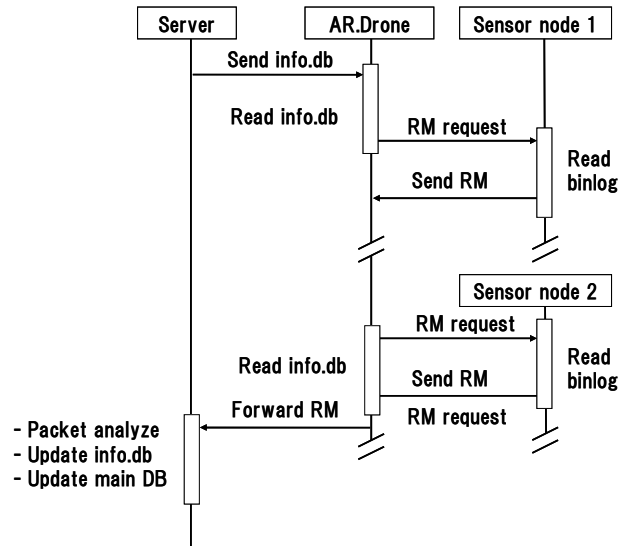


Figure 9. Message sequence of communication between server – air vehicle and air vehicle – sensor node

Before setting up replication between the server and the sensor nodes, an empty database is constructed with the same

name in all machines. Once a MySQL Replication is set up, the master (i.e., the sensor node), starts recording all the changes in its database to a file named binlog file. When a data ferry connects to a sensor node, it sends a Replication Message Request, which contains information of the position of the replication starting point. The sensor node reads its binlog and sends the necessary replication messages back to the ferry. Then, the ferry receives the data and stores them on a text file, which is created for each sensor node. The ferry waits for incoming data in a few seconds before closing the connection. This waiting time is called data receiver time-out. Then, the ferry returns to the server and forwards the obtained data to that. At the server, the replication messages are analyzed to construct MySQL commands for updating the database, and the commands are executed directly to the server database.

The structure of the replication message is shown in Figure 10. There are many types of packet and event in MySQL Replication. However, to get the necessary information, we focus on Rotate Event and Query Event in OK Packet. A Rotate Event is added into a binlog file to tell the client what binlog to request next. And a Query Event is added into a binlog file when a query is executed on the replication database.

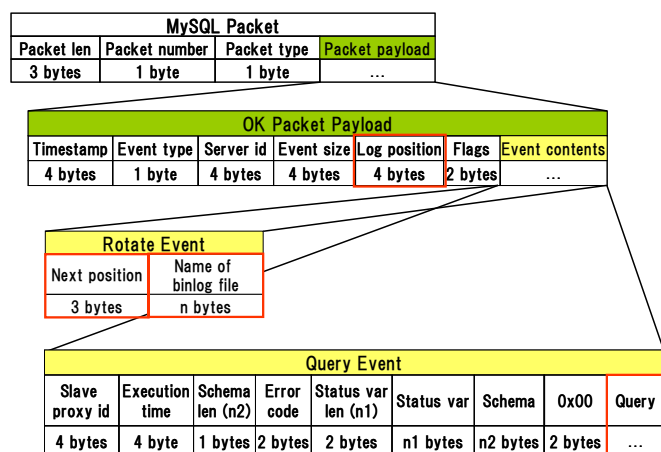


Figure 10. Structure of replication message

Based on Figure 10, a replication message is analysed to get the queries, the binlog position, and the binlog file name. The queries is used to update the main server's database, while the binlog file name and binlog position is saved into a SQLite database file named info.db. It is sent to the air vehicle before performing the autonomous flight, to help the data ferry collect the necessary data from the sensor nodes. The SQLite database is chosen because it is the most powerful embedded database with light-weight and easy to use.

With this method, the DTN ferry does not need to carry a big database like MySQL Database, and hence, reduce the load for the air vehicle.

E. Program Structure of Proposed System

The data synchronization programs that we develop are placed on the server and the air vehicle, as shown in Figure 11. Both the sensor node machine and the server machine work on Ubuntu with the MySQL Database, while the air vehicle (i.e., AR.Drone) runs on Linux. The replication only occurs between the air vehicle and the sensor node. Here, the air vehicle acts like a replication slave, sending a login message and a replication request message to the master database, which means the MySQL Database of the sensor node. Original MySQL packets received from the master database are carried to the server, where they are analyzed and applied directly to the MySQL Database of the server.

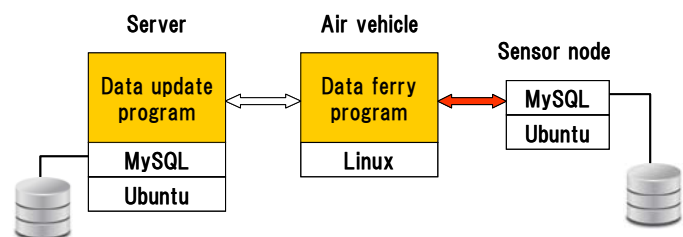


Figure 11. Program Structure of proposed system in DTN environment

In case of non-DTN environment, the synchronization is performed directly and easily in MySQL Database from the sensor node to the server, without creating any additional program, as shown in Figure 12.

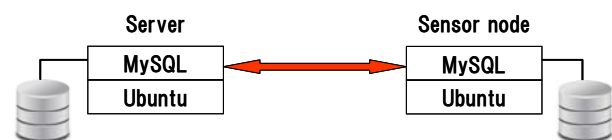


Figure 12. Program Structure in non-DTN environment

In both cases, the database of the sensor node is the same. Only the server database needs to be constructed to match the environment. It is one of the important benefits of this proposed system, because if the condition of the environment changes, we only need to re-construct the program on the server.

IV. EVALUATION OF PROPOSED SYSTEM

A. Accuracy of Autonomous Flight

In order to evaluate the practicality of this system, an experiment of the autonomous flight has been performed. Based on the proposed algorithm mentioned in Section III-C, we have created an autonomous flight application which works on the AR.Drone[13]. With a repeat cycle of 5 second, we expect that the AR.Drone will automatically move to the destination point.

However, as shown in Figure 13, the flight becomes slightly disorder due to the impact of wind, the accuracy of GPS, as well as the effect of angular acceleration. Even though the turning angle and direction has been calculated exactly according to data from Table 1, the air vehicle could

not turn its direction by an angle calculated before. Hence, it is necessary to improve the stability of this flight algorithm.

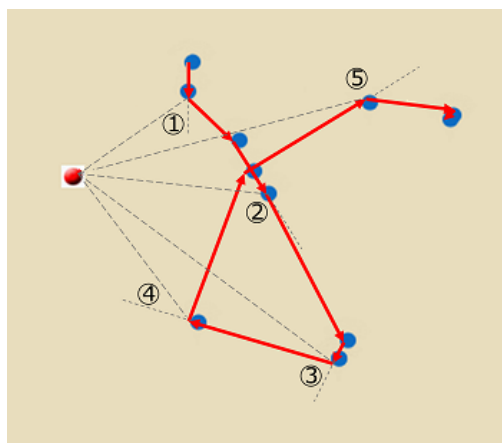


Figure 13. Message sequence of communication between server – ardrone and ardrone – sensor node

TABLE 1. EXPERIMENT RESULT

Position	Direction determination	
	Turning direction	Turning angle
1	Right	50.74022
2	Right	129.2082
3	Right	92.36998
4.	Right	32.03243
5.	Left	164.5351

B. Data Synchronization Time

Experimental evaluations have been performed to observe time taking for transfer data from the sensor nodes to the server database. Here, instead of an AR.Drone, we used a BeagleBone and two PC to set up the experiment environment. The Beagle Bone works as a data ferry, while two PC works as the sensor node and the server. The specification of the experimental devices is shown in Table 2.

TABLE 2. SPECIFICATION OF EXPERIMENT DEVICES

	PC1 (Server)	BeagleBone	PC2 (Sensor node)
CPU	Intel® Celeron® CPU 550 @ 2GHz	TI Sitara AM3359 720MHz ARM Cortex-A8	Intel® Core™ i3 CPU M350@2.27MHz
Hardware Memory	2.0 GiB	256MB	2.9 GiB
OS	Ubuntu 11.10	Ubuntu 12.10	Ubuntu 10.04

Figure 14 shows the processing time of the proposed data synchronization. First, the BeagleBone connects to the sensor

node – PC1, logins and sends replication message request. The sensor node then returns a sequence of MySQL Replication packet. After receiving the replication messages, the connection between the BeagleBone and the sensor node PC is closed, and the connection between the BeagleBone and the server is opened. Next, replication messages are sent to the server PC, to analyze and update the main database.

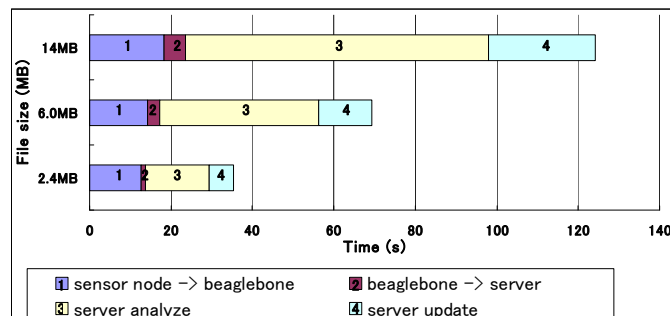


Figure 14. Message sequence of communication between server – ardrone and ardrone – sensor node

According to this result, the time for forwarding data from the data ferry to the server is quite short. However, the time for collecting replication message from the sensor node to the data ferry, and the time for the server to analyze and update data is long. In case of gathering data from the sensor node to the ferry, it takes 5 seconds time-out for waiting for the replication messages. Hence, in each time the data ferry connects to the sensor node, it takes at least 5 seconds. We are working on reducing this time by shortening the time-out time.

The time for server to analyze and update its database is long too. However, this process is performed at the main server, which has no pressure about battery life of a data ferry. Therefore, this result can be accepted.

V. CONCLUSIONS AND FUTURE WORKS

This paper has presented a DTN sensor network system based on store-carry-forward model, using an autonomous air vehicle as a DTN ferry. Specific features of this system are the autonomous flight and the data synchronization from the sensor nodes to the server. With the proposed synchronization method, the data ferry does not need to carry a big database.

Evaluation of the developed system has shown that the time for data synchronization can be accepted, because the longest process is performed at the server, which is not under pressure of battery life. In addition, the algorithm for calculating the turning angle and direction can be ensured. However, it is necessary to control the movement of the air vehicle more exactly to prevent flight disorder.

Currently, we are studying for improving the stability of the autonomous flight as well as completing the entire system.

This study was partly supported by MEXT/JSPS KAKENHI Grant Number 25730055.

REFERENCES

- [1] ARDrone, <http://ardrone.parrot.com>
- [2] GlobalSat, “EM-406A GPS Receiver Engine Board”, http://www.dpcav.com/data_sheets/EM-406A_Product_Guide1_1.pdf

- [3] V.Cerf, "Delay-Tolerant Networking Architectures", IEEE RFC4838, Apr 2007
- [4] Delay Tolerant Networking Group, <http://www.dtnrg.org/>
- [5] Bundle Protocol Specification, <http://tools.ietf.org/html/rfc5050>
- [6] S.Farrell, V.Cahill, D.Geraghty, I.Humphreys, P.McDonald, "When the TCP Breaks", IEEE Internet Computing, vol.10, issue 4, pp 72-78, Sep 2006
- [7] H.Kurazono, S.Uchiyama, M.Yamamoto, K.Nakamura, K.Yamazaki, "Development and Evaluation of Observation Network for Streaked Shearwaters using ZigBee and RFID", IEICE Tech Rep, vol.112, no.430, IA2012-73, pp.13-18, Feb 2013
- [8] H.Yamamoto, S.Uchiyama, M.Yamamoto, K.Nakamura, K.Yamazaki, "Development of Sensor Network for Ecology Observation of Seabirds", IEICE Trans. Inf. & Syst., vol.E95-D, no.2, pp.532-539, Feb 2012
- [9] H.Ochiai, H.Ishizuka, Y.Kawakami, H.Esaki, "A DTN-Based Sensor Data Gathering for Agricultural Applications", IEEE Sensor Journal, vol.11, no.11, pp.2861-2868, Nov 2011
- [10] R.Nakano, H.Yamada, S.Yamamura, A.Nagata, M.Tsuru, Y.Oie, "Field Experiment of Vehicle Information Delivery using Store-Carry-Forward Routing Scheme", IEICE Technical Report, NS2009-253, pp.511-516, Mar 2010
- [11] R.Ayaki, H.Shimada, K.Sato, "A Proposal of Sensor Data Collection System Using Mobile Relay Nodes", Wireless Sensor Network, vol.4, no.1, pp.1-7, Jan 2012
- [12] MySQL Internals Manual, <http://dev.mysql.com/doc/internals/en/>
- [13] P. Tran Thi Ha, et al., "Using Autonomous Air Vehicle in DTN Sensor Network for Environmental Observation", Proc. IEEE COMPSAC 2013, no. DS-2, pp. 447-450, Jul 2013



Thi Ha Phuong Tran received B.E. degree from Nagaoka University of Technology in '12. She is currently a graduate school student in Nagaoka University of Technology. Her research interests include sensor networks and Delay Tolerant Network.



Hiroshi Yamamoto received M.E. and D.E. degrees from Kyushu Institute of Technology, Iizuka, Japan in '03 and '06, respectively. From April '06 to March '10, he worked at FUJITSU LABORATORIES LTD., Kawasaki, Japan. Since April '10, he has been an Assistant Professor in the Department of Electrical Engineering, Nagaoka University of Technology. His research interests include computer networks, distributed applications, and networked services. He is a member of the IEEE.



Katsuyuki Yamazaki received B.E. and D.E degrees from the University of Electro-communications and Kyushu Institute of Technology in '80 and '01, respectively. At KDD Co. Ltd., he had been engaged in R&D and international standardization of ISDN, S.S. No.7, ATM networks, L2 networks, IP networks, mobile and ubiquitous networks, etc., and was responsible for R&D strategy of KDDI R&D Labs. He is currently a Professor of Nagaoka University of Technology.