On the Use of Naming Scheme for Controlling Flying Router in Information Centric Networking

Yuhao Gao*, Taku Kitagawa*, Shingo Ata†, Suyong Eum* and Masayuki Murata*
* Graduate School of Information Science and Technology, Osaka University, Japan
1–5 Yamadaoka, Suita, Osaka, 565–0871
Email: {u-kou, t-kitagawa, suyong, murata}@ist.osaka-u.ac.jp
† Graduate School of Engineering, Osaka City University, Japan
3–3–138 Sugimoto, Sumiyoshi-ku, Osaka-shi, Osaka 558-8585
Email: ata@info.eng.osaka-cu.ac.jp

***Abstract*—Recently, Information-Centric Networking (ICN), is attracting attention as a network capable of flexible communication especially in IoT mobility environments. As a research on flexible controls in ICN, we have been involved in a movable router using Unmanned Aerial Vehicles (UAVs) or drones (called Flying Routers (FRs)) to realize communication in disjoint networks. In previous studies, control of Flying Routers was targeted reactively where FRs move according to the events on receiving Interest or Data. However, in order to realize a collaborative movement strategy such as content retrieval in consideration of cooperation by multiple FRs, more strategic ways which directly control FRs according to decision of path optimization. For this purpose, in this paper, we design a communication control system which strategically controls FRs by sending explicit Interests. We incorporates ICN technologies in the design of the system: naming scheme and strategy layer, which enable flexible control over FRs. Furthermore, we present a Proof-of-Concept (PoC) implementation of the system to confirm the feasibility of the design.**

I. Introduction

Information Centric Networking (ICN) is being considered as a new network paradigm where data exchanges are not conducted location basis (e.g., IP addresses) but content basis. In ICN, a name (structured or unstructured) is used for data retrieval. By extending a philosophy from location-based content delivery to name-based processing (not only data processing but also device controls), ICN has a great potential to realize a flexible and programmable framework to construct an in-network service. Combination of name-based process discovery and in-network processing provides a strong capability and flexibility especially in IoT (Internet of Things) and/or M2M (Machine-to-Machine) environments, where small devices are collaboratively working.

In recent years, IoT devices such as sensors and cameras have come to be used in various fields such as agriculture, environmental measurement and so on. In such field, the cost of deployment of wireless networks which covers huge area relays the communication with IoT equipments is extremely high. Movable devices such as message ferry in DTN (Delay Tolerant Network) are considered to resolve such a problem. An autonomous movable router is a router that is installed in a moving object such as a car or a ship for collecting and delivering packets. The router controls actual movements of the body by itself based on the information such as the position of the node and the movement strategy.

ICN can greatly deal with communications seamlessly achieved by physical movement of routers, because the control of movement can be considered as a kind of in-network processing in ICN. ICN with moving routers extends not only the coverage area of communication, but also provides a new communication framework like data sharing in disaster area. Previously we propose an architecture called RMICN (Router Movable Information Centric Network) to address above-mentioned background [1]. In RMICN, Flying Routers (FRs) are introduced which supports physical movement of ICN router embedded on UAV (Unmanned Aerial Vehicle) for delivering messages between disjoint networks where partial networks are placed disjointly (not connected directly each other). RMICN is considered to support collaborative movement of multiple Flying Routers and provide a path planning method based on the demand of Interest/Data exchanges.

However, since RMICN is conceptually proposed, design and implementation of Flying Routers are still challenging. Especially, collaborative movement of multiple Flying Routers according to strategic path planning may require an explicit control of Flying Routers based on a movement plan. For unified communication in both control and data processing, these control sequences should be realized in the principle of ICN. We therefore propose an architectural design of Flying Routers which supports physical movements by invoking APIs (Application Programming Interfaces), which are directly associated with named functions in ICN (i.e., controlling FRs by sending Interests).

Namely the contribution of this paper is following,

- We design a set of APIs (in other words naming scheme in ICN) to support direct control of Flying Routers which are necessary to realize RMICN.
- We design a system architecture of Flying Router to handle above APIs as a kind of in-network processing on ICN router.

Specifically, we provide APIs sufficient to realize RMICN, i.e., path planning strategies are achieved by calling APIs we implemented. As an implementation of ICN, it is necessary to implement NDN (Named Data Networking), extending NFD

(NDN Forwarding Daemon) which is packet forwarder of NDN (NDN Forwarding Daemon). We consider such kind of implementation would also a good practice toward ICN in real-world. Through experimental evaluations we confirm the operation of RMICN can be realized by proposed Flying Routers with APIs, and discuss lessons learned.

This paper is organized as follows, in the next section we introduce related works. We then describe the proposed API set to support RMICN, and a design of Flying Router in real environment in Section III. Next, we describe the experimental scenario to verify the system in Section IV. Finally in Section V we summarize this paper.

II. Related Work

In this research, in order to show the implementation for communication using autonomous movable routers in ICN, we show the implementation for communication using RMICN. As an autonomous mobile router, we use the FR proposed in RMICN.

FR is a router able to move by loading the router to an UAV, then controlling the movement of UAV with a program in the router [1]. The role of FR is to relay communication between the other nodes by doing a round trip between those nodes. In particular, FR is useful in realizing communication between the disjointed networks. Examples of applications include disaster area networks and agricultural sensor networks. Examples of movable routers similar to FRs include base station vehicles and message ferries. Compared to those mobile routers, the advantage of FR is that it can be used in various places without being affected by geographical features.

RMICN is an ICN-based communication protocol that realizes communication between disjointed networks by carrying packets generated from nodes in FRs [1]. RMICN can realize high-speed and flexible communication by cooperation of multiple FRs. As another protocol realizing communication between disjointed networks using a movable router similar to RMICN, a message ferry method of DTN can be cited as an example. In such method, the mobile router moves along the same route without changing the route [2], whereas in the RMICN, the movable router itself changes the route to satisfy the requests of contents early.

The reason why ICN is used for communication using mobile routers is that ICN can realize high speed communication. Specifically, in a mobility environment where the IP address frequently changes, ICN can transfer packets with the name of the content instead of the IP address, so that overhead of processing can be reduced compared to TCP/IP. Furthermore, in ICN, by using the strategy layer and the packet naming schema, which is a standard equipment mechanism that realizes additional processing after receiving packets, processing on packets can be realized with less overhead than TCP/IP [3].

Implementation problems in real environment are design modification on interfaces for communicating with real environment nodes and communication mechanisms between RMICN systems and UAV. Specifically, regarding the problem of the design modification of the interface, in the previous work, the communication used the interface of the OS in the local environment [1], but in real environment, the communication needs to use ICN's interface. Regarding the problem of the design of the communication mechanism between the RMICN system and the UAV, in the previous work, the program performed the movement control of the FR with the simulator of the UAV [1], but in real environment the program communicates with the real machine of the UAV. Design of communication mechanism between RMICN system and UAV is necessary.

The reason for implementing RMICN in real environment is to show the method of realizing communication using autonomous mobile router in ICN by implementing it in the real environment. As the merits of showing its realization method, it becomes easier to measure the speed of communication using autonomy movable router in ICN, or to develop an application using a mobile router in ICN.

III. System Design

This section explains the overview of communication control system in Flying Router, system components, packet structure, and sequence specifications. In the FR communication control system, a part of managers, tables, strategies, and sequences are used among the components conceptually proposed in [4], and we design a set of APIs of the NDN used for packet communication and the physical control of Flying Routers.

A. Overview

In this subsection, we outline the system by explaining the network topology of the FR communication control system, and the system structure by explaining the system components and communication flow.

We show the overview of FR-based communication system (RMICN) in Figure 1. There are four types of nodes: Depot Router (DR), FR, GateWay (GW) and other nodes such as sensors, actuators, or other types of devices within the communicable range of GW. DR and GW are not directly interconnected, but FR realizes communication between nodes by physically delivering messages within its storage.

DR is a node that centrally manages the movement and communication of multiple FRs and calculates the traveling path of FR. FR is also a relay node that carries packets between DR and GW. GW collects packets from other nodes within its coverage area and communicates directly with FR. Other nodes within the communicable range of the GW does not directly communicate with FR, but it periodically transfers NDN packets to GW.

All nodes are in the RMICN communication range, which is defined in prior, and FR crawls within the RMICN communication range. In the initial state of the system, DR holds the coordinates and node names of each node in the RMICN communication area, and does not hold the name of the content generated by each node. By visiting each node, FR announces the name of the content generated by each GW to all the nodes in the network, at the same time, acquires and distributes the content request and the content. All moving paths are initiated at DR so that FR installs a new path when FR comes back to DR after traveling.

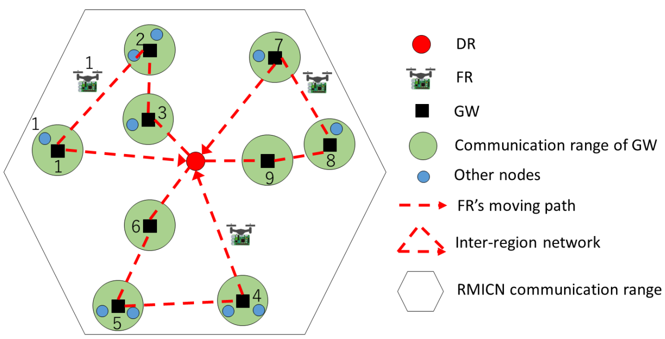

Fig. 1. Overview of FR-based Communication System

Regarding the communication control of FR provided by the system, Table I summarizes the name, operation outline, and input/output of APIs. In this paper, APIs are designed and invoked by Interest packets of ICN, i.e., name of APIs follow the naming scheme of ICN. To invoke an API, the name of API is specified in the Interest packet, and node that receives the packet calls the function of the strategy to execute the operation of the API.

B. Component Construction

As for the components of the program of each node, each node has a program (named node program in this paper) named DR, FR0, GW1, GW2 shown in figure 2 and NDN Forwarding Deamon (NFD) [5], and FR has another Drone Control Program (DCP). As a role of each component, Node program performs table buffer reference and strategy processing, NFD transfers the packet to the local component or to another node, DCP performs drone movement control.

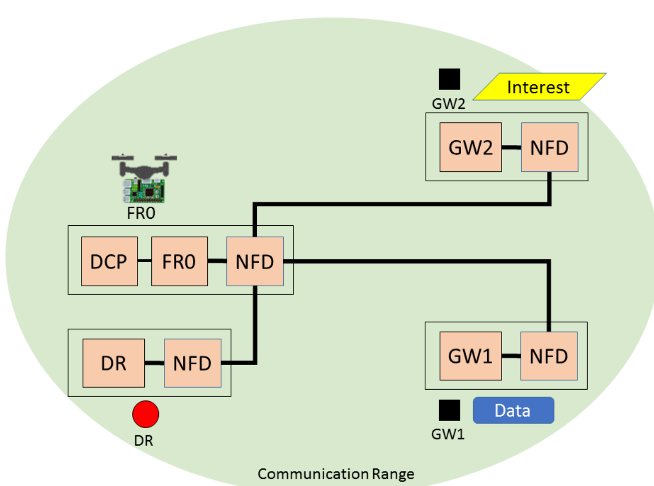

Fig. 2. Component Construction

TABLE I
List of Designed APIs

API Name	/Neighbor/{NodeName}/onFaceConnected
Role	Exchange packets of sensor data when FR connected to DR or GW
Input	Name of the connected node
Output	Interest and Data packet from the opposite node
API Name	/RMICN/{PacketName} PacketName={ConsumerName}/ {ContentName}*/{SystemTime}
Role	API Name is the name of packet with sensor data or packet requesting sensor data. Save the packet to buffer store.
Input	Name of consumer, name of contents and time when packet was generated
Output	-
API Name	/FRControl/{FRName}/Crawl/ {detourTime}/{Point}* Point = /{North}/{East}/{Altitude}{PointName} PointName = {FNName} HOME
Role	Transfer traveling path and limit time to FR
Input	Name of the FR, a limit time for running the traveling path, an east-west direction distance from DR to the point, a distance in the north-south direction and a height (unit: meters), or name of the point
Output	-
API Name	/FRControl/{FRName}/PathCreate/ {PathName}/{Point}*
Role	Generate, save, edit traveling path
Input	Name of the FR, a set of traveling path's waypoints and the name of the travelling path ,
Output	-
API Name	/FRControl/{FRName}/PathRun/ {PathName}
Role	Start moving along the traveling path
Input	A character string of traveling path
Output	-
API Name	/FRControl/{FRName}/PathPause
Role	Pause and resume FR's movement of currently executing path
Input	Name of the FR
Output	-
API Name	/FRControl/{FRName}/PathCancel
Role	Cancel the movement along the currently executing path and stop the movement of FR
Input	Name of the FR
Output	-
API Name	/FRControl/{FRName}/SetVehicleSpeed/ {Speed}
Role	Set FR moving speed
Input	FR moving speed (unit: m/s)
Output	-

For the modules of each node program, each node program has its own table, manager and strategy, as shown in Fig.3.

C. Behavior of Individual Components

NFD sends and receives NDN packets based on ICN forwarding pipeline and data structure. In this system, the packet transmission/reception function is extended to the table, manager and strategy for RMICN communication.

In addition to NFD's FIB (Forward Information Base), CS (Content Store), PIT (Pending Interest Table), Node Table, Buffer Store and Path Table are added in the table. In the

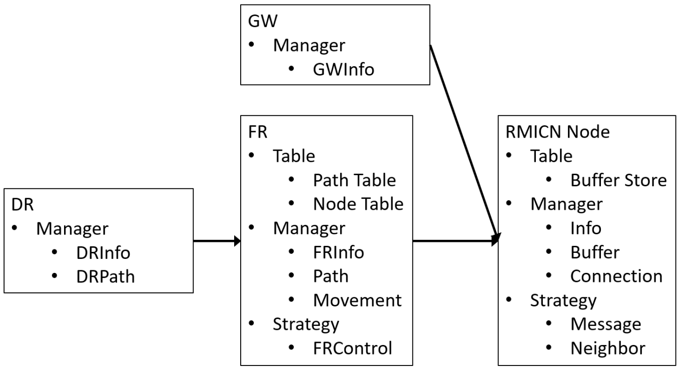

Fig. 3. Modules of Each Node Program

TABLE II
AN EXAMPLE OF CUSTOM DATA STRUCTURE: PATH TABLE AND NODE TABLE OF DR IN FIGURE 1

Node Name	Path
FR0	GW1, GW2, GW3
FR1	GW4, GW5, GW6
FR2	GW7, GW8, GW9

Node Name	Node Type	Location	Face ID	Connected
FR1	FR	Not Depot	Face1 ID	F
FR2	FR	Not Depot	Face2 ID	F
FR3	FR	Depot	Face3 ID	F
DR	DR	Depot	-	-
GW1	GW	Loc1	-	-
GW2	GW	Loc2	-	-
...	...	...	...	...
GW9	GW	Loc9	-	-

Node Table, the name of the connection destination node, the type and position of the node, the face ID, and the connection state are stored as information necessary for communication. Regarding the judgment of the connection state, if the distance between the nodes is within a constant meter, it is judged that the connection has been made, a new face is created and communication is started, otherwise the face is deleted. Buffer Store is a table for storing Interest packets requesting sensor data and Data packets storing sensor data. Path Table is a table that stores a pair of a traveling path and a name of a FR moving along the path.

A manager performs connection setting and communication by controlling the operation of tables and strategies. There are four types of managers: Connection Manager, Information Manager, Buffer Manager, Path Manager, and Movement Manager. The Connection Manager creates faces and sets listeners when nodes are connected. Information Manager manages transmission and reception of tables and packets. Buffer Manager saves and deletes the RMICN packet stored in the node. The Path Manager calculates and saves the traveling path. Movement Manager controls drone movement by calling an external program that controls the drone.

In the strategy, FRControl Strategy is designed for controlling the mobility of FR based on the name of the packet, Neighbor Strategy for exchanging RMICN packets, and Message Strategy for buffering and generating RMICN packets were added. FRControl Strategy stores the traveling path in the name of the Interest packet and transmits the packet, and the FR that received the packet carries out movement control by calling the drone control program based on the name of the packet. The Neighbor Strategy stores the name of the node performing packet switching in the name of the Interest packet and transmits the packet, and the node that received the packet exchanges the packet of the Buffer Store by the requested amount to the node that sent the packet. Message Strategy performs Interest packet buffering and Data packet generation.

D. Packet Structure Design

In the FR communication control system, the Interest packet and the Data packet of the NDN packet are used. The prefix of Interest packet is /FRControl, /Neighbor, /RMICN, and the node which received each packet calls the processing of FRControl Strategy, Neighbor Strategy, Message Strategy. The name space of Interest packet is shown in Figure 4. We set the lifetime of Interest packets with prefixes /FRControl and /Neighbor equal to NFD default lifetime. In the Interest packet that does not need to return data such as sensor data and connection information, it stores the message successfully received the packet in the Data packet and returns it. On the other hand, regarding the lifetime of the Interest packet having the prefix of /Message, all the FRs are fed back to the DR and synchronized in order to efficiently relay the packet in consideration of the time required for FR movement and communication. Among all the FRs, let T_r be the longest time to complete the traveling path and $4T_r$ be the lifetime of the Interest packet. Because delivery of Interest packets of sensor data can be divided into three stages of Interest retrieval, Data retrieval, Data delivery in order of time. Therefore, the Interest packet with the longest time to be satisfied is an Interest packet generated by the GW immediately after the FR visits the GW. We set the lifetime to $4T_r$ to satisfy such a packet within its lifetime. In the namespace of the Data packet, make the same name as the Interest packet so as to satisfy the corresponding Interest packet of the Data packet.

E. Sequence Design

Overall Sequence: Immediately after the FR communication control system is started, DR, FR, GW updates information such as node name and position in each Node Table. At this time, FR is near DR, and mutually connects and exchanges Interest packet requesting sensor data and Data packet storing sensor data. Then, DR calculates the traveling path based on the position information of the GW using VRP and IACO [6] of the path planning method and stores it in its own Path Table and at the same time stores the path in the

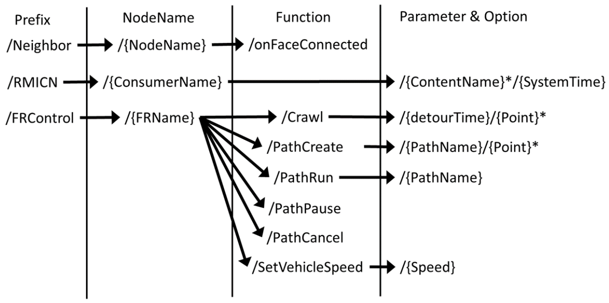

Fig. 4. Name space of Interest packet

name of Interest packet and sends the Interest packet to the FR. When the FR receives the Interest packet, it stores the reception success message in the Data packet, returns it to the DR, and then starts moving according to the traveling path. When arriving near the GW in the traveling path, FR stops moving and connects with GW and exchanges packets. After packet exchange is over, the FR calculates a new traveling path based on its own packet and strategy and moves to the next GW of the path. After visiting all the GWs on the traveling path, the FR will return to the DR and if it arrives near the DR it will exchange the packet. Then, by repeating the computation of the traveling path and the movement control as described above, communication between the nodes in the RMICN communication area is realized.

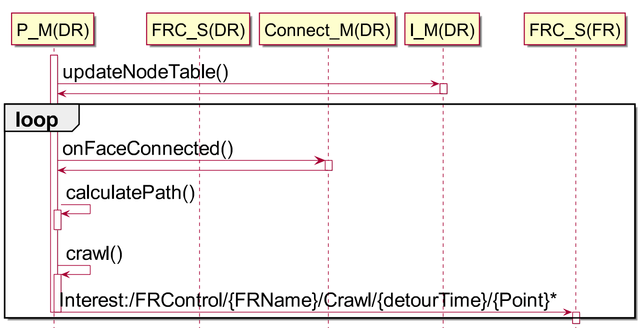

Fig. 5. Sequence when FR is close to DR

Movement Control of UAV: When FR receives Interest of Crawl, it passes Crawling Path in the name of its Interest to DCP and starts DCP so that FR controls movement. After that, the FR program periodically acquires the current coordinates of UAV from DCP. When the FR arrives at the GW, registration of the FIB entry and transfer of the content-requesting interest and the content data are performed as a process upon arrival.

Registration of FIB Entry: When FR arrives around the GW, it transfers the Interest of Retrieve Route to GW, receives the name of the content generated by that GW from GW, and registers it in its own FIB of that name. Similarly, the GW receives the content name generated by another GW from the FR and the name of the content owned by the DR, and registers the name in its own FIB.

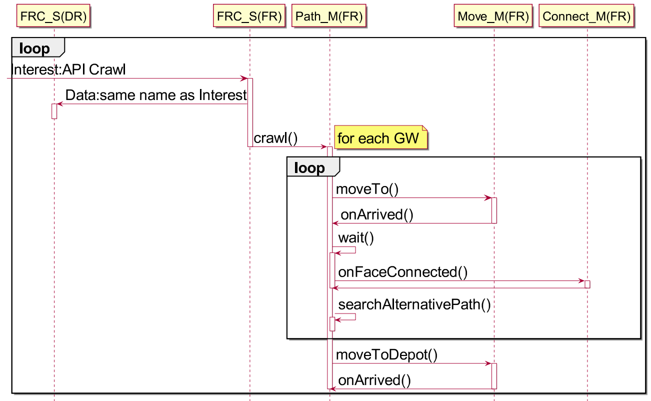

Fig. 6. Sequence when FR departs from DR and visits each GW

Generation and Transmission of Packet: Interest is generated after FIB entry is generated, and Data is generated after receiving Interest. In addition, packets are transferred only when connected to the transfer destination, and otherwise stored in the Buffer Store until connection with the transfer destination. However, Interest will be deleted from the Buffer Store when the Life Time is over. Data does not take Life Time into account in this study and does not delete from Data Buffer Store.

IV. Experiments

In the experiment, we use the FR communication control system incorporating the movement control mechanism and confirm the operation of the system's API. The movement control mechanism consists of a program called User Interface (UI) which transfers Interest packets storing the name of the API, FRControl Strategy and a drone control program. With this mechanism, movement control of FR can be performed by sending an Interest packet. Currently, implementation of communication mechanism is in progress for sensor network communication using FR communication control system. We also confirm that the ICN naming scheme can be used as API by confirming the operation of this experiment and that strategic mobility control of FR is possible by using this system. First, this section describe the mounting environment, equipment and experimental environment. Next, the operation and procedure of experiments are described as a system verification method.

A. Equipment and Implementation Environment

Table III lists the software, tools, and equipment used for the implementation of FR communication control system.

Drone and Raspberry Pi (RPi) communicate via the Wi-Fi provided from the controller of the drone, and power is supplied from the drone via the serial port.

Location of the real machine experiment is the place of the circular area with a radius of 30m centered on the controller of the drone. The radius of Wi-Fi emitted from the controller is 30m, and the pseudo communication range of each node is 10m.

TABLE III
DESCRIPTION OF TOOLS AND EQUIPMENT

Tool	Version	Description
ndn-cxx	0.5.0	C++ NDN library
jndn	0.15	Java NDN library
NFD	0.5.0	NDN packet forwarder
DroneKit-Python	2.0	Control tool for drone
Mission Planner	1.3.44	A tool to display the state of the drone in the GUI
Python	2.7.0	language of Drones control tool
c++	C++ 11	language of ndn-cxx and NFD
java	java 8	language of jndn
Ubuntu	14.04 LTS	User Nodes OS
Raspbian	Sep 2016	Raspberry Pi's OS
Equipment	**Model**	**Description**
Drone	3D Robotics Solo	Programmable control, WiFi communication possible, GPS loaded
Computer	Raspberry Pi 3 Model B	With WiFi, receiving power from drone

Fig. 7. Overall view of FR

DR, FR0, GW1, GW2 are represented as (0, 0), (0, 0), (0, 30), (30, 30) when the position of the node is expressed by a pair of numbers (distance in the north direction to DR, distance in the east direction to DR). The DR program is executed on the PC, and the programs of GW1 and GW2 are executed on the RPi. GW1 creates the content, and GW2 requests the content. The name of the content is /RMICN/GW1Service/<SystemTime>, and the content in the Data packet is null.

B. Experiment Procedure and Result

As for the behaviors of each device in the experiment, when we activate the program of each node, FR takes off from the vicinity of DR and it repeats movement in the order of DR, GW1, GW2, DR. Among those movements, in the first visit to GW1, we set FR to create an FIB entry in FR so that Interest requesting content is transferred from FR to GW1. In the first visit to GW2, by setting the FIB entry with GW2, it is set to transfer Interest requesting content from GW2 to FR. At the same time, GW2 transfers Interest requesting content to FR. At the second visit to GW 1, FR transfers the interest requesting content to GW1 and at the same time receives content from GW 1. In the second visit to GW2, GW2 receives the content from the FR. As a result, communication between GWs using FR is completed.

As an output message to be confirmed, at the first visit to GW1, a message indicating that FR sent Interest of Neighbor Strategy to GW1 to generate FIB entry, and that the FIB entry generated at FR in the first visit to GW1 Check the message indicating. In the first visit to GW2, in order to generate an FIB entry, a message indicating that the Interest of the Neighbor Strategy has been transferred to the FR and a message indicating that the FIB entry has been generated by the GW2 are confirmed. At the same time, it confirms a message indicating that the GW2 has forwarded the interest for requesting the contents to the FR. In the second visit to GW1, a message indicating that the FR has forwarded the content requesting content to the GW1 and a message indicating that the FR has received the content from the GW1 are confirmed. At the second visit to GW2, we check the message that GW2 received contents from FR.

In the experiment, we confirmed the above operation, showed the operation and message according to the design specification, and showed that ICN real environment communication using one FR is possible by using the system designed in this paper.

Fig. 8. Experimental Verification of Communication Using FR

V. CONCLUSION

In this paper, we have focused on the naming scheme of ICN and the flexibility of the strategy layer, and have proposed FR communication control system that can actively control mobile routers. Firstly, we have explained the topology, use case, component configuration and flow of communication of FR communication control system. Next, for the implementation of the system, we have described the packet specifications, component specifications and sequences of the communication control system. In the real machine experiment, we have introduced the installation environment of the tool to be used, the OS, the equipment used, and the experimental environment. Next, we have explain the experiment contents and verification procedure of the main API, and have confirmed the operation as specified by each API as the result of the experiment. It is shown that the naming scheme of ICN can be used as API by real machine experiment and that strategic mobility control to FR can be performed by using this system.

Acknowledgment

A part of works in this paper is supported by the EU-JAPAN initiative by the EC Horizon 2020 Framework Programme Grant Agreement No.723014 and NICT under contract No. 184.

REFERENCES

[1] T. Kitagawa, S. Ata, and M. Murata, "Mobility-controlled flying routers for information-centric networking," in *Proceedings of 2018 IEEE Consumer Communications & Networking Conference (CCNC)*, Jan 2018, pp. 1–2.

[2] W. Zhao and M. H. Ammar, "Message Ferrying: Proactive routing in highly-partitioned wireless ad hoc networks," in *Proceedings of Distributed Computing Systems, 2003. FTDCS 2003. Proceedings. The Ninth IEEE Workshop on Future Trends of*, Mar. 2003, pp. 308–314.

[3] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, Dec. 2009, pp. 1–12.

[4] T. Kitagawa, "Mobility-controlled flying routers for information centric networking," Master's thesis, Osaka University, Feb. 2017.

[5] A. Afanasyev, J. Shi, and B. Zhang, "NFD developers guide," *Dept. Comput. Sci., Univ. California, Los Angeles, Los Angeles, CA, USA, Tech. Rep. NDN-0021*, Oct. 2016.

[6] B. Yu, Z.-Z. Yang, and B. Yao, "An improved ant colony optimization for vehicle routing problem," *European Journal of Operational Research*, vol. 196, no. 1, pp. 171 – 176, Jul. 2009.

[7] R. Sugihara and R. K. Gupta, "Path planning of data mules in sensor networks," *ACM Trans. Sen. Netw.*, vol. 8, no. 1, pp. 1:1–1:27, Aug. 2011.

[8] G. Tyson, N. Sastry, I. Rimac, R. Cuevas, and A. Mauthe, "A survey of mobility in information-centric networks: Challenges and research directions," in *Proceedings of the 1st ACM Workshop on Emerging Name-Oriented Mobile Networking Design - Architecture, Algorithms, and Applications*, Jun. 2012, pp. 1–6.