# Realization of Mobility-controlled Flying Router in Information-centric Networking

Yuhao Gao, Taku Kitagawa, Suyong Eum, Shingo Ata and Masayuki Murata

*Abstract:* **Constructing a large-scale network in outdoor fields with wired or wireless antennas involves a huge deployment cost. New devices, called movable routers, are able to realize the networks at a low cost. Recently, a large number of internet of things nodes have been deployed into the networks in outdoor fields. In the current communication network, users intend to communicate with devices without knowing where they are, but the current network still focuses on device locations, complicating configuration and implementation of the network. In this study, we design an architecture called router-movable information-centric networking (RMICN) to realize content-based communication instead of host-based communication, for facilitating communications between disjointed networks using movable routers. We incorporate information-centric networking technologies in the design of the system for the naming scheme and the strategy layer, enabling device control over the movable routers, called flying routers. In the simulation results, we show that RMICN can further reduce the response time of content requests compared with delay-tolerant networking. Furthermore, we present a proof-of-concept implementation of the architecture to confirm the feasibility of the design.**

*Index Terms:* **Movable Router, Device Control, Data Delivery, Delay-tolerant Networking, Information-centric Networking**

## I. Introduction

Internet of things (IoT) devices such as sensors and cameras are being used in various fields such as agriculture, disaster management, etc. In such applications, the cost of deployment of wireless base stations and wired networks, which cover huge areas, is extremely high. New movable devices called mobility-controlled routers (MRs) are proposed in this study to resolve this problem. In this paper, an MR is a router that is installed in a moving object such as a car or an unmanned aerial vehicle (UAV) and switches and forwards packets. The router controls the movements of the moving object based on strategies according to a specific application scenario. By using such routers, communication in a large-scale network can be realized at a low cost.

In communications between disjointed networks using mobile routers, there is no connection between different networks. The IP is designed for end-to-end communication where connections are established and hence using it for realizing mobile-based communications is difficult. A new network architecture called information-centric networking (ICN) can greatly deal with the communications seamlessly. Since ICN uses content caching, ICN is able to support communications in disrupted networks. Besides, ICN is considered as a new network paradigm where data exchange is not conducted on the basis of locations (e.g., IP addresses) but on the basis of contents. In ICN, names (structured or unstructured) instead of IP addresses are used for data retrieval. ICN extends a philosophy from location-based content delivery to name-based processing (not only data processing but also device controls) and hence is a flexible and programmable framework with a great potential to realize in-network services. ICN with movable routers not only extends coverage range of communication, but also provides a new communication framework. Here we discuss system design to realize such a communication framework.

In this work, we propose an architecture called router-movable information-centric networking (RMICN) to address the above-mentioned problem. We also propose an implementation of RMICN using one mobility-controlled router commonly called flying router (FR) in a real world. In RMICN, FRs are introduced for delivering messages between disjointed networks where partial networks are not connected with each other. RMICN supports movement control of FRs. RMICN is able to support various application scenarios.

Specifically, we provide an architecture design to realize RMICN, which contains operations for movement control of FRs as well as retrieval and distribution of contents. Our simulation studies show that the response time of content requests in RMICN is shorter than IP-based delay-tolerant networking (DTN) [2].

Since we conceptually propose RMICN, FR design and implementation are still challenging. Especially, strategic path planning may require an explicit device control of FRs. The control sequences should be realized using the principle of ICN. Therefore, we propose an implementation of FR that supports the movement control based on a strategic algorithm within the FR or demand of other nodes. Other nodes can achieve the control by sending packets that include the names of application programming interfaces (APIs) designed in RMICN and invoking the APIs.

We provide a sufficient number of APIs to realize RMICN. As an implementation of ICN, named data networking (NDN) [3] is used that is one of ICN realizations. We presume that this kind of implementation would also be a good practice toward the implementation of ICN in the real world. Through experimental evaluations, we confirm that the operation of RMICN can be

---

Yuhao Gao, Taku Kitagawa, Suyong Eum, and Masayuki Murata are with the Graduate School of Information Science and Technology, Osaka University. Address: 1-5 Yamadaoka, Suita, Osaka, 565-0871, Japan. Email: {u-kou, t-kitagawa, suyong, murata}@ist.osaka-u.ac.jp.

Shingo Ata is with the Graduate School of Engineering, Osaka City University. Address: 3-3-138 Sugimoto, Sumiyoshi-ku, Osaka-shi, Osaka 558-8585, Japan. Email: ata@info.eng.osaka-cu.ac.jp

Yuhao Gao is the corresponding author.

This paper is an extension of proceedings paper [1]. We have added the comparison between the architecture proposed in this paper and delay-tolerant networking and architecture evaluation and discussion from results of evaluation and demonstration to this paper.

realized by using the proposed FRs with APIs.

This paper is organized as follows. In the next section, we introduce related works. We then describe the overview of RMICN in Section III. We propose a design of FR and architecture in Section IV. We present the evaluation results of the architecture in Section V and describe the experimental scenario to verify communication using one FR in Section VI. We present the insights and future works from the results of evaluation and experiment in Section VII. Finally, in Section VIII we summarize this paper.

## II. Related Work

Recent works deploying movable routers have proposed several architectures and protocols. Networks using movable routers, including mobile phone, vehicles, and UAVs, are named as mobile ad hoc networking (MANET), vehicle ad hoc networking (VANET), and flying ad hoc networking (FANET). These networks focus on supporting both characteristics of movable routers (e.g., limited energy [4], mobility model [5]) and routing according to certain application scenarios (e.g., cooperation of movable routers [6], [7], data collection and distribution [8], [9]). Since these networks have similar characteristics, a part of architectures and protocols in MANET, VANET, and FANET are applicable to each other[10]. In this work, we propose an architecture rather than a specific protocol. We presume that this architecture is able to using various protocols of MANET, VANET, and FANET.

For improving the safety of humans and UAVs when UAVs are used in security-sensitive areas, studies on detecting, tracking, jamming, and hunting UAVs have been conducted [11]. For detection, a ground control station (GCS) and other UAVs identify the behaviors of target UAVs based on some signatures of the targets and the background. Recent technologies recognize the signatures based on camera photos. Tracking is performed to calculate the locations of target UAVs. The calculation is mainly based on the global positioning system (GPS) data and channel information. Jamming is performed to disturb the behaviors of evil UAVs. It can be carried out by using excess power and GPS spoofing. Hunting is performed to make target UAVs land safely or leave security-sensitive areas and can be carried out by using nets. In this work, we do not consider the monitoring malfunction of UAVs and evil UAVs. We assume that all UAVs in our proposed architecture behave without malfunction and are used for messaging.

Networks using movable routers are of various types of topologies, which depend on specific application scenarios. Recent works have proposed various architectures for communication using individual or swarmed movable routers [12], [13]. In this work, movable routers in the proposed architecture are individual and are used for communications between disjointed networks.

An existing IP-based architecture similar to our proposal is delay-tolerant networking (DTN) [14]. DTN is considered as an unstable network topology with a long latency, as end-to-end paths may not exist. Delay may be measured in days for some networks.

Communication using mobile routers can also be realized in IP-based DTN architecture, but the DTN architecture is fundamentally different from ICN. DTN supports hop-by-hop communications essential in disjointed networks. In such communication, DTN realizes the store-and-forward technique by using the bundle layer that stores packets when a node is disconnected with the next hop and forwards the packets during connection. However, packets are not copied. When different nodes require the same buffered packets, the requests needs to be send to the endpoint, causing duplicated traffic. ICN can reduce the duplicated traffic by using ICN characteristic of content caching.

In addition, in content retrieval in DTN, a node needs to declare the location of destinations (e.g., IP addresses) and the attributes of contents (e.g., names and types of contents) to identify the destinations and the contents. Actually, if we can realize the name-based routing that identifies a destination with the attributes of contents instead of locations, only the attributes are needed in communications. In ICN, communications are achieved with the technique, allowing developers to develop an application easily.

As for research on mobility support in ICN, an ICN architecture called voice over content-centric networking (VoCCN), supporting consumer mobility, was proposed in [15]. The scenario of VoCCN is that a movable consumer retrieves voice data from the publisher during movement. When a movable consumer is able to connect to more than one network, the architecture chooses the best route to retrieve the data, and measures the response time of requests by using the lifetime of the requests. In this work, we propose an ICN architecture supporting the mobility of relay nodes.

As for research on supporting communications between consumers and IoT devices in ICN, an ICN architecture for content retrieval from IoT devices was proposed in [16]. The architecture includes a hierarchical naming scheme recording content requests and cached content, and is able to aggregate the requests and the data efficiently. In this work, data generated by IoT devices are also retrieved by consumers, but we do not propose a naming scheme for aggregation. We propose a naming scheme for device control of FRs, update of routing tables, and packet transmission with the store-and-forward technique.

## III. Router-movable ICN (RMICN)

### A. Application Scenarios Considered

RMICN is designed for M2M or human-machine communications in outdoor fields where network infrastructures are difficult to be deployed and hence movable routers are needed as relay nodes (e.g., agriculture land, disaster areas). Especially, RMICN is suitable for communication with a large number of IoT devices that retrieve contents within a specific period of time. The distribution of evacuation order in disaster areas and the inspection of fire and smoke in agricultural fields are examples of scenarios in which this type of communication is used. In content retrieval, RMICN only needs the attributes of contents instead of the identifiers of hosts and thus simplifies the communication configuration. In addition, RMICN aggregates requests and contents and reduces traffic, saving the energy of IoT devices. Moreover, RMICN uses content caching in consumers and routers and helps to reduce the response time of requests.

Examples of application of RMICN are the collection and distribution of information in agricultural sensor networks and disaster networks. In the agricultural sensor network, one agriculture field has multiple sensors, a gateway that collects information generated by the sensors, and multiple mobile routers collecting sensor data from gateways. The mobility of mobile routers is controlled for communication enhancement. The battery life of the mobile router is finite, and to save energy, the behavior management of the mobile router is performed by a central network coordinator. As an example of specific communication, an irrigation sprinkler system can be considered. In the system, sensors measure the moisture of the soil and sprinkler distribute water to the field in a controlled way. Sprinkler distribute water based on sensor data only to the part where the moisture is insufficient.

In the disaster area network, instead of a broken network infrastructure, multiple mobile routers provide the instructions from municipalities to residents. At the same time, routers provide municipalities with the safety information of residents and other relevant environmental information. To realize efficient communication, a gateway that can aggregate and cache contents is installed close to the residents. Behavior management of mobile routers is carried out by central network coordinator. As an example of specific communication, risk management immediately after the earthquake can be considered. In that communication, residents provide information about foods, injured people, and the environment. The municipality provides evacuation instructions to residents.

### B. Network Topology and Operation

In RMICN, there are four types of nodes: depot router (DR), FR, gateway (GW), and other RMICN nodes such as sensors and actuators within the communication range of the GW. RMICN nodes within the communication range of the GW construct a network called an intra-region network. Different intra-region networks are not directly interconnected, but FR realizes communications between nodes by delivering messages produced by RMICN nodes. The topology of RMICN is shown in Figure 1.

A DR is a node that centrally manages the movement and the communication of multiple FRs and calculates the moving path of an FR. An FR is a relay node that carries packets between the DR and the GW. The GW collects packets from other nodes within its coverage area and communicates directly with FR. Other nodes within the communication range of the GW do not directly communicate with the FR, but periodically transfer packets to the GW.

All nodes are in the RMICN communication range, and the FR wanders within the RMICN communication range. In the initial state of the system, the DR holds the identifiers of nodes in the RMICN communication area. By visiting each node, the FR exchanges information for updating the routing tables with the node, at the same time, acquires and distributes content requests and the contents.

Path planning of movable routers and routing of packets are based on specific strategies. We assume that FRs do not form a swarm and move individually. RMICN does not standardly provide strategies for forming a swarm and organizing a formation of movable routers. Movement control of FRs can be centralized or decentralized. Reactive or proactive types of routing protocols are applicable to the architecture. In this work, we propose an ICN architecture for movable routers instead of a strategy.

RMICN uses two types of packets: Interest and Data. Interest packets are content requests or instructions for device control or update of routing tables. Data packets are requested contents or responses to instructions.

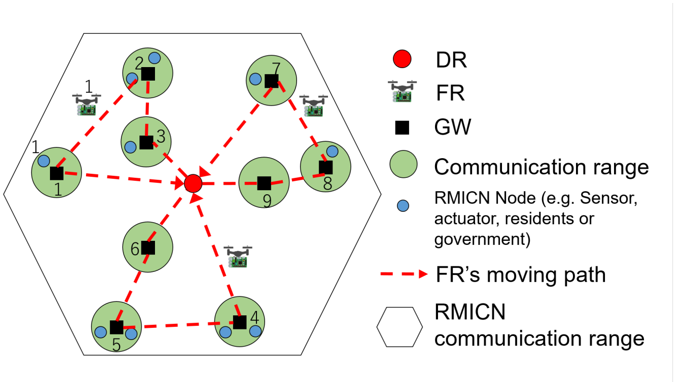


Fig. 1. Overview of FR-based Communication System

As for the main operations of RMICN, we design operations for device control, update of routing tables, and packet transmission with the store-and-forward technique. In communications between different intra-region subnetworks, FRs are supposed to receive and send packets from the GWs of each intra-region subnetwork. The DR and RMICN nodes are able to control the movement of FRs for communication purposes (on-demand device control), or the FRs control themselves based on some information (proactive device control). The DR and RMICN nodes can control the movement of FRs and instruct the FRs to communicate with certain GWs by sending an Interest packet. The name of the Interest packet includes the positions of the GWs. When an FR receives the Interest, its strategy layer invokes APIs of libraries for device control of the UAV and then the FR visits the corresponding GWs.

When the FR arrives at the GW, it exchanges the information with the GR for updating the routing table (e.g. all hosts publishing contents or with cached contents in the whole area) and updates its table. The details of the information and the algorithm used to update the routing table depend on a specific strategy (e.g. a strategy updates the table on the demand of consumers or proactively), and the strategy depends on specific scenarios described in subsection III-A.

The RMICN architecture also enables the store-and-forward technique in which packets are stored in the buffer when two nodes are in disconnection and transmits the packets in the established connection.

### C. RMICN Benefits Compared with DTN

#### C.1 Simplified Configuration in Content-based Communication

DTN identifies a node with an IP address. However, a users' need is to retrieve either content with a name or a type of contents. In this case, DTN needs to map contents to hosts. Actually, an identifier of the content should be converted to an identifier of the host producing or caching the content. The mapping complicates the configuration and the implementation of network services. In RMICN, networks identify a node with the information of contents, and nodes are able to retrieve a content or to control a device by utilizing the information, which simplifies the configuration of communications.

#### C.2 Low Traffic with Packet Aggregation

In DTN, communication focus on where hosts are rather than what packets mean (e.g. name or type of contents within the packets). In RMICN, the information of a packet is opaque to the node receiving the packet. Two packets with the same information are considered to be the duplicated ones (e.g. requests of the same content or the data of the same content) and then one is dropped, reducing traffic and avoids loops of packets.

#### C.3 Early Response with Content Caching

DTN enables the store-and-forward technique by transmitting the buffered packets and deleting the entries from the buffer. For retrieval of the same contents that have been received previously, the contents have to be transmitted from end to end, and this can be enhanced by caching the contents at the end or relay nodes. In RMICN, the end and relay nodes are able to keep copies of contents. By responding to the content request with a copy of the content at the relay nodes, the response time can be shortened.

## IV. Architecture Design

### *A. Component Construction*

The components of each node are shown in Figure 2. Each node has a node program (DR, FR, GW, and RMICN in Figure 2) and the NDN forwarding daemon (NFD)[17], and the FR has a device control program (DCP). As for roles of each component, the node program provides the reference to a table and a buffer. The node program also provides strategy layer processing to achieve operations described in subsection III-B. The NFD transfers packets to the local component or to another node. The DCP performs movement control of the FR node with specific libraries for controlling the moving body. For the modules of the node program, each node program has its own table and strategy, shown in Figure 3.

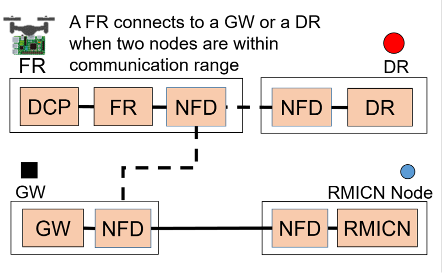


Fig. 2. Component Construction

### *B. Behavior of Individual Modules*

An FRs moving to a host with requested contents, it needs to know the position of the host. The FR deletes the entries of pending Interest table (PIT) from the NFD, which records the upstream Interest and face when the FR is disconnected with the node sending the Interest to the FR. However, the FR needs to return the requested data by referencing the entries of the PIT.

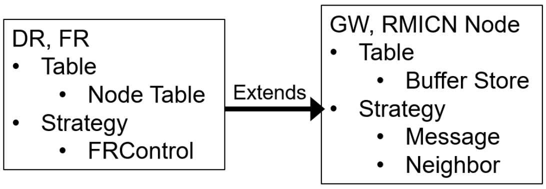


Fig. 3. Modules of Node Program

In addition to the tables of the NFD including the forwarding information base (FIB), the content store (CS) and the PIT, we add the node table and the buffer store in the table. In the node table, the identifier of the destination node, the position of the node, and the face ID are stored as information necessary for communication. The buffer store is a table that stores Interest packets and the face receiving the Interest. An example of the node table is presented in Table 1.

As for the design of strategy, the FRControl strategy is designed for controlling the mobility of an FR based on the name of packets. The neighbor strategy is for exchanging RMICN packets. The message strategy is for buffering and generating RMICN packets. The FRControl strategy stores the traveling path in the name of the Interest packet and transmits the packet, and the FR that receives 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 receives 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.

Table 1. An example of custom data structure: Node Table of DR in Figure 1

| Node Name | Location | Face ID |
|---|---|---|
| FR1 | Not Depot | Face1 ID |
| FR2 | Not Depot | Face2 ID |
| FR3 | Depot | Face3 ID |
| DR | Depot | - |
| GW1 | Loc1 | - |
| GW2 | Loc2 | - |
| ... | ... | ... |
| GW9 | Loc9 | - |

### *C. Packet Structure Design*

In the RMICN architecture, the names of the APIs are appended to the names of Interest packets, and the packets invoke strategy layer processing of a node receiving the packets. The name of the Interest includes the name of the APIs and their parameters. In the ICN communication using mobile routers, to address the problem of the naming scheme, the namespace of the essential APIs and the corresponding parameters have not yet been considered. We design the naming scheme presented in Table 2 for essential processing of contents retrieval, which includes appending names for device control, making routing table, and forwarding packets.

The prefixes of Interest packets are /FRControl, /Neighbor, /RMICN. The nodes receiving the packet call the processing of FRControl strategy, neighbor strategy, and message strategy. We set the lifetime of the Interest packets with prefixes /FRControl and /Neighbor equal to the NFD default lifetime. A node receiving these Interest does not need to return data such as sensor data and connection information, and the node stores the ACK message as the payload of the Data packet and returns it.

### *D. Sequence Design*

#### D.1 Device Control of UAV

When an FR receives Interest packets of which prefix is /crawl, it passes the path recorded in the name of the Interest to the DCP so that the DCP controls the movement of the FR. After that, the FR program periodically acquires the current geographical position of the UAV from the DCP. When the FR arrives at the GW, an update of the forwarding table and the transmission of Interest and Data are performed.

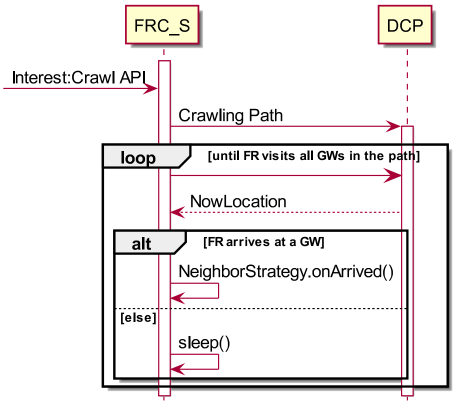


Fig. 4. Sequence in which the FR program controls the movement of UAV

#### D.2 Update of Forwarding Table

When the FR arrives at the GW, the FR transfers the Interest with the name of RetrieveRoute to the GW, and receives a Data packet including the information for updating the FIB of the FR (e.g. the name of the content generated by the GW), and then updates the FIB. Then, the GW also carries out a similar process to update its FIB.

#### D.3 Generation and Transmission of Packet

Content requests are generated after their FIB entries are generated. The requested data packets are generated and are returned to the requester after receiving Interest. Packets are stored in the buffer store or content store, and the packets are transferred when a connection to the destination is established. When the lifetime of the requests and requested data are over, the requests will be deleted from the buffer store, and the data will be deleted from the content store. The lifetime depends on specific strategies.

Table 2. List of Designed APIs

| | |
|---|---|
| API Name | /Neighbor/{NodeName}/RetrieveRoutes |
| Role | Construct routing table in FR and node in connection with FR |
| Input | Name of the connected node |
| Output | Interest and Data packet from the opposite node |
| API Name | /RMICN/{PacketName}/{SystemTime}<br>PacketName={ConsumerName}/{ContentName}*/{PacketIdentifer} |
| Role | Save the packet to the buffer store. |
| Input | Names of consumers, names of contents and the identifer of the packet. |
| Output | - |
| API Name | /FRControl/{FRName}/Crawl/{detourTime}/{Point}*<br>Point = /{North}/{East}/{Altitude} \|{PointName}<br>PointName = {FNName} \|HOME |
| Role | Make the FR create the path and start traveling along the path. |
| Input | The name of the FR, maximum time for traveling around the path, east-west direction distance from the DR to the point, a distance in the north-south direction, height (unit: meters), and the name of the point. |
| Output | - |
| API Name | /FRControl/{FRName}/PathCreate/{PathIdentifer}/{Point}* |
| Role | Generate, save and edit the traveling path |
| Input | The name of the FR, a set of the traveling path's waypoints and the identifer of the travelling path |
| Output | - |
| API Name | /FRControl/{FRName}/PathRun/{PathName} |
| Role | Make the FR start moving along the traveling path. |
| Input | The identifer of the traveling path. |
| Output | - |
| API Name | /FRControl/{FRName}/PathPause |
| Role | Pause and resume FR's movement. |
| Input | The identifer of the FR. |
| Output | - |
| API Name | /FRControl/{FRName}/PathCancel |
| Role | Cancel the movement along the current path and make the FR return to the DR. |
| Input | The identifer of the FR. |
| Output | - |
| API Name | /FRControl/{FRName}/SetVehicleSpeed/{Speed} |
| Role | Set the moving speed of the FR. |
| Input | The moving speed of the FR (unit: m/s). |
| Output | - |

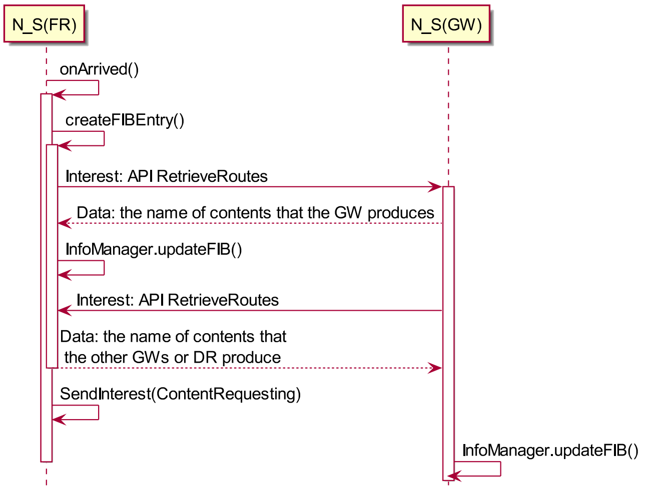
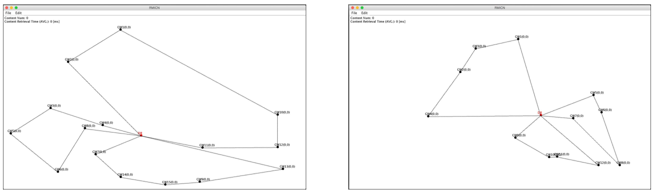
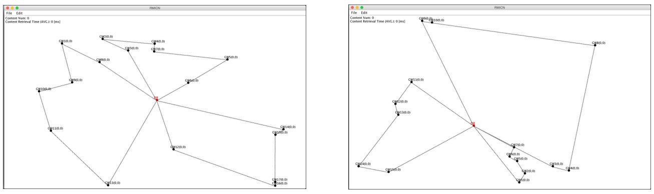


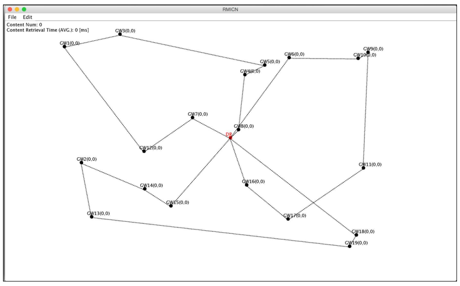
Fig. 5. Sequence for updating the forwarding table



Environment1 Environment2



Environment3 Environment4



Environment5

Fig. 6. Simulation Environment

## V. Evaluation

The section presents the evaluation results of the proposed architecture in terms of content retrieval time. The results are based on simulation as well as the experiment using the implemented RMICN.

### A. Evaluation Methods

In this work, we evaluated the content retrieval time in RMICN in a simulation environment. As a comparison target, we used DTN, which is a method to implement disjoint network communication. As a traffic pattern, a pattern assuming the scenarios of disaster networks was used to evaluate the content cache. In the scenario, the same content (e.g. news from the outside of the disaster area, evacuation orders from local government) was requested multiple times because this happens in disaster situations. The disaster network includes one DR, multiple FRs, and multiple GWs. Multiple GWs are both publishers and consumers of contents. Each publisher generates multiple types of contents and each content is unique. Each consumer randomly chooses one content from all contents and sends a content request to the publisher at a certain time interval. Different consumers request the same content in a certain probability. Each request is for retrieving one of the contents.

The moving paths of FRs are the same in RMICN and DTN. The difference is that when different GWs request the same content, FRs in RMICN do not need to deliver the same request to the publishers and serve the repeated requests by returning the cached content to the GWs. In DTN, the movable relay nodes need to deliver the request and need to retrieve the content from the publishers.

The parameters of the simulation are as follows. The number of requests to contents is 1,000. The probability of requesting the same content is 0.1. The waiting time of an FR in each GW is 1s. The simulation environment is shown in Figure 6.

### B. Evaluation Results

In this section, the results of the simulation described in Section V-A are presented. We use the average content retrieval time, which is a period of time from the generation of an Interest to the arrival of the corresponding Data. The average content retrieval times in the disaster networks are listed in Table 3.

Table 3. The average of content retrieval time in disaster networks

| ENV | ENV1 | ENV2 | ENV3 | ENV4 | ENV5 |
|---|---|---|---|---|---|
| DTN [s] | 184.5 | 119.6 | 165.7 | 190.7 | 161.2 |
| RMICN [s] | 157.3 | 101.6 | 150.1 | 151.2 | 147.9 |
| Reduction Rate [%] | 15.8 | 15.1 | 9.3 | 21.7 | 8.3 |
| AVG. [%] | 13.6 | | | | |

As shown by these results, in RMICN, the content retrieval time was reduced by 20% at the maximum, compared to the DTN case. In the case of RMICN, the retrieval time is smaller due to the content cache in the scenario involving disaster networks. The effect of the content cache reduces content retrieval time by an average of 13.6%. In this simulation, the probability of requesting the same content was set to 0.1. However, in the case where this value becomes large, the content retrieval time can be further shortened.

## VI. Demonstration

In the demonstration, we confirm the communication between the GWs using the APIs of the proposed system. Currently, we use one FR for the verification of the communication between the GWs. We confirm the feasibility of the proposed naming scheme. This section describes the experiment environment, hardware and software, and experiment procedure and result.

### A. *Equipment and Implementation Environment*

Table 4 lists the software, tools, and equipment used for the implementation of the system.

Table 4. Description of tools and equipment

| Software | 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 UAV in the GUI |
| Python | 2.7.0 | Language of device control program |
| c++ | C++ 11 | Language of the ndn-cxx and the NFD |
| java | java 8 | Language of the jndn |
| Ubuntu | 14.04 LTS | User Node's OS |
| Raspbian | Sep 2016 | Raspberry Pi's OS |

| Hardware | Model | Description |
|---|---|---|
| UAV | 3D Robotics Solo | A drone capable of programmable control, WiFi communication, and localization with GPS |
| Computer | Raspberry Pi 3 Model B | With WiFi, receiving power from drone |

Fig. 7. Overall view of the FR

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

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

DR, FR0, GW1, and GW2 are represented as (0, 0), (0, 0), (0, 30), and (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 was executed on the PC, and the programs of GW1 and GW2 were executed on the RPi. GW1 created the content, and GW2 requested the content. The name of the content was /RMICN/GW1Service/<SystemTime>, and the content in the Data packet was null.

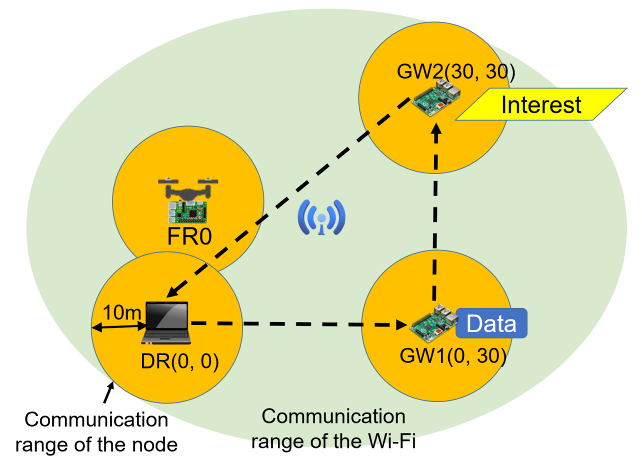


Fig. 8. Experiment Scenario

### B. *Experiment Procedure and Result*

As for the behaviors of each device in the experiment, when we activated the program of each node, the FR took off from the DR and moved to the GWs in the order of shorter distance to the DR. Among those movements, at the first visit to GW1, we set the FR to create an FIB entry in the FR so that Interest requesting content could be transferred from the FR to GW1. At the first visit to GW2, by setting the FIB entry with GW2, Interest requesting content could be transferred from GW2 to the FR. At the same time, GW2 transferred the Interest to the FR. At the second visit to GW1, the FR transferred the interest requesting content to GW1 and at the same time received content from GW1. At the second visit to GW2, GW2 received the content from the FR. As a result, communication between GWs using FR was accomplished.

As an output message to be confirmed, at the first visit to GW1, we have confirmed a message indicating that the FR sends Interest packets of neighbor strategy to GW1 to generate an FIB entry at the FR and the FIB entry was generated. At the first visit to GW2, we confirmed a message indicating that GW2 sent Interest packets of neighbor strategy to the FR to generate an FIB entry in GW2 and the FIB entry was generated. At the same time, we confirmed a message indicating that GW2 forwarded the interest requesting the content. At the second visit to GW1, we confirmed a message indicating that the FR forwarded the request to the GW1 and received the content from the GW1. At the second visit to GW2, we confirmed a message indicating that GW2 received the content from the FR. A picture of the experiment is presented in Figure 9. The video of the experiment has been uploaded at [18].

In the experiment, we have confirmed that the above operation and message are in accordance with the design specification. We showed that ICN real-environment communication using one FR is feasible using the architecture proposed in this paper.

```
Sent            Interest    /Neighbor/GW2/RetrieveRoutes/FR0/1517203443124
Sent            Data        /RMICN/FR0/GW1Service/1513356457794
Sent            Data        /RMICN/FR0/GW1Service/1513356457996
Sent            Data        /RMICN/FR0/GW1Service/1513356458198
Received        Interest    /Neighbor/FR0/RetrieveRoutes/GW2/1513356490668
Sent            Data        /Neighbor/FR0/RetrieveRoutes/GW2/1513356490668
Received        Data        /Neighbor/GW2/RetrieveRoutes/FR0/1517203443124
```

```
Sent            Interest    /RMICN/FR0/GW1Service/1513356458198
Received        Interest    /Neighbor/GW2/RetrieveRoutes/FR0/1517203443124
Sent            Interest    /Neighbor/FR0/RetrieveRoutes/GW2/1513356490668
Sent            Data        /Neighbor/GW2/RetrieveRoutes/FR0/1517203443124
Received        Data        /RMICN/FR0/GW1Service/1513356457794
Received        Data        /RMICN/FR0/GW1Service/1513356457996
Received        Data        /RMICN/FR0/GW1Service/1513356458198
```

Fig. 9. Experimental Verification of Communication Using FR

## VII. Discussion

### A. Evaluation

From the simulation results, we verified that RMICN can shorten the retrieving time of contents further compared to DTN in the disaster network assumed in this research.

In the disaster network scenario, the probability that two GWs request the same content is set to 0.1, but in the case in which this value becomes large, the content retrieving time can be further shortened.

For future works, it is conceivable to further utilize content-based information for the construction method of the point-to-point network and the path planning of the movable router. Specifically, flexible content delivery can be realized by changing the moving path of FRs according to the distribution of content or content requests. In addition, this simulation does not consider the delay and network scale requirements in real-world applications. We leave the proposal for a strategy that can satisfy specific application requirements as a future work.

### B. Demonstration

We verified the following items from the results of the experiment.

- The mobility of the FR can be controlled using ICN naming scheme and the strategy layer.
- The RMICN supported by one movable router can be realized using the components and packets and sequences of the system designed in this research.

In the experiment, all nodes belong to the same network and we spuriously realized communication between disjointed networks. If the distance between the mobile router and other nodes is within a certain value, the movable router can connect to the node. The reason why we realize the communication in a pseudo disjointed network is that we intend to confirm that the architecture is feasible. In the future, the operation of the proposed architecture will be extended to disjointed networks to further support real-world application scenarios.

The scale of the experimental network is a circular region with a radius of 40m, which corresponds to the area of one agriculture filed in Japan (standard size is 20m x 50m) and one community of disaster area. As for the delay, the content retrieval in the demonstration takes totally 60 seconds including 20 seconds for waiting and communication and 40 seconds for moving along the path. The time for waiting and communication in the experiment can be further shortened since the size of contents is small and the time is excessive for the FR exchanging all packets with other nodes. The time depends on the specific application scenarios. The time for moving also can be further shortened if a faster UAV is used in the experiment.

We will devise the requirements on the network scale of a real-world application in future work. We will also realize the operation of multiple FRs simultaneously in the real-world environment of RMICN.

## VIII. Conclusion

In this study, we leveraged naming scheme of ICN and the strategy layer and proposed an architecture for communications among disjointed networks. Simulation results showed that RMICN can retrieve content 13.6% faster than DTN. In the implementation of the proposed RMICN, we introduced the software, hardware, and experimental environment. The experiment scenario and verification procedure of communication using one FR were explained. The experiment showed that the RMICN architecture presented in this paper is feasible, and communication using one movable router can be realized.

In future work, we will discuss the requirements of specific real-world applications and a strategy considering the content characteristics of the applications. In addition, we will implement RMICN communication using multiple movable routers.

### *Acknowledgment*

A part of this work 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] Yuhao Gao and Taku Kitagawa and Suyong Eum and Shingo Ata and Masayuki Murata, "On the use of naming scheme for controlling flying router in information centric networking," in *Proceedings of IEEE International Conference on Communications Information Centric Networking Solution For Real World Applications (ICN-SRA)*, pp. 1–6, May 2018.

[2] M. Tsuru, M. Uchida, T. Takine, A. Nagata, T. Matsuda, H. Miwa, and S. Yamamura, "Delay tolerant networking technology - the latest trends and prospects," *The Institute of Electronics, Information and Communication Engineers*, vol. 2011, no. 16, pp. 57–68, Mar. 2011.

[3] NDN Project, "Named Data Networking (NDN)," http://named-data.net/.

[4] S. Mamechaoui, F. Didi, and G. Pujolle, "A survey on energy efficiency for wireless mesh network," *CoRR*, vol. 3890–3904, Nov. 2013.

[5] M. Ohleger, G. G. Xie, and J. H. Gibson, "Extending uav video dissemination via seamless handover: A proof of concept evaluation of the ieee 802.21 standard," in *Proceedings of System Sciences (HICSS), 2013 46th Hawaii International Conference on*, pp. 5106–5114. IEEE, Mar. 2013.

[6] R. Moazzez-Estanjini, J. Wang, and I. C. Paschalidis, "Scheduling mobile nodes for cooperative data transport in sensor networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 3, pp. 974–989, Jun. 2013.

[7] J. Cruz, G. Chen, D. Li, and X. Wang, "Particle swarm optimization for resource allocation in uav cooperative control," in *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, p. 5250, Aug. 2004.

[8] A. A. Somasundara, A. Ramamoorthy, and M. B. Srivastava, "Mobile element scheduling for efficient data collection in wireless sensor networks with dynamic deadlines," in *Proceedings of Real-Time Systems Symposium, 2004. 25th IEEE International*, pp. 296–305, Dec. 2004.

[9] E. P. De Freitas, T. Heimfarth, I. F. Netto, C. E. Lino, C. E. Pereira, A. M. Ferreira, F. R. Wagner, and T. Larsson, "Uav relay network to support wsn connectivity," in *Proceedings of Ultra Modern Telecommunications and*

*Control Systems and Workshops (ICUMT), 2010 International Congress on*, pp. 309–314. IEEE, 2010.

[10] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in uav communication networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, Nov. 2016.

[11] Z. Kaleem and M. H. Rehmani, "Amateur drone monitoring: State-of-the-art architectures, key enabling technologies, and future research directions," *IEEE Wireless Communications*, vol. 25, no. 2, pp. 150–159, May. 2018.

[12] B. Bellur, M. Lewis, and F. Templin, "An ad-hoc network for teams of autonomous vehicles," in *Proceedings of the First Annual Symposium on Autonomous Intelligence Networks and Systems*, pp. 1–6, May. 2002.

[13] M. Soltan, M. Maleki, and M. Pedram, "Lifetime-aware hierarchical wireless sensor network architecture with mobile overlays," in *Proceedings of Radio and Wireless Symposium, 2007 IEEE*, pp. 325–328. IEEE, Jan. 2007.

[14] University of Missouri-Rolla, "Delay Tolerant Networks," http://web.mst.edu/ mobildat/DTN/index.html, accessed: 2018-02-27.

[15] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, and R. L. Braynard, "VoCCN: Voice-over Content-Centric Networks," in *Proceedings of the 2009 Workshop on Re-architecting the Internet*, ser. ReArch '09, pp. 1–6, Dec. 2009.

[16] E. Baccelli, C. Mehlis, O. Hahm, T. C. Schmidt, and M. Wählisch, "Information centric networking in the IoT: Experiments with NDN in the wild," in *Proceedings of the 1st ACM Conference on Information-Centric Networking*, pp. 77–86, Sep. 2014.

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

[18] Youtube, "Realization of Mobility-controlled Flying Router in Information Centric Networking," https://youtu.be/MEL_YtGaKvY, accessed: 2018-06-20.

**Yuhao Gao** received an B.E. degree from the School of Engineering Science, Osaka University, Japan in 2017. Currently, he is a M.E. candidate. His research interests include the design and implementation of a autonomy movable router using unmanned aerial vehicle for Information Centric Networking.

**Taku Kitagawa** received an M.E. degree from the Graduate School of Information Science and Technology, Osaka University, Japan in 2017. His research interests include the design and implementation of a autonomy movable router using unmanned aerial vehicle for Information Centric Networking.

**Suyong Eum** received B.E. in Electronics Engineering from MyongJi University, Korea and D.E. in Electrical and Computer Engineering Science from RMIT University, Australia in 2007. After his postdoctoral fellowship in Osaka University, Japan, he has been working for the National Institute of Information and Communications Technology (NICT). His current research interests include information centric networking, complex network theory, and network optimization.

**Shingo Ata** received M.E. and D.E. degrees in Informatics and Mathematical Science from Osaka University in 1998 and 2000, respectively. Since 2013, he has been a professor at the Graduate School of Engineering at Osaka City University. His research work includes networking architecture, design of communication protocols, and performance modeling of communication networks. He is a member of IEICE, IEEE, and ACM.

**Masayuki Murata** received his M.E. and D.E. degrees from Osaka University, Japan, in 1984 and 1988, respectively. In 1984, he joined the Tokyo Research Laboratory, IBM Japan, as a researcher. He was an assistant professor from 1987 and an associate professor at Osaka University from 1992 to 1999. Since 1999, he has been a professor at Osaka University, and he is now with the Graduate School of Information Science and Technology, Osaka University. He has more than 500 papers in international and domestic journals and conferences. His research interests include computer communication networks, performance modeling, and evaluation. He is a member of IEICE, IEEE, and ACM.