Final Project Report
Detailed Form

I. Title of Proposed Project:

Mobile IoT

II. Project Leader:

Full name: Sumei Sun
Institution: Institute for Infocomm Research
Address: 1 Fusionopolis Way #21-01, Connexis South Tower, Singapore 138632
Phone: +65 6408 2529
E-mail: sunsm@i2r.a-star.edu.sg

III. Project Members:

Name	Position/Degree	Department, Institution, Country	Email Address
Hiroshi Emoto	Director General/ PhD	NICT, Japan	jiang@nict.go.jp
Fumihide Kojima	Director/ PhD	NICT, Japan	f-kojima@nict.go.jp
Kentaro Ishizu	Research Manager/ PhD	NICT, Japan	ishidu@nict.go.jp
Nobuyuki Asai	Deputy Director/ PhD	NICT, Thailand	asai@nict.go.jp
Hoang Vinh Dien	Research Scientist/ MSc	NICT, Singapore	hvdien@nict.go.jp
Thu Ngo-Quynh	Associate Professor/ PhD	HUST, Vietnam	thunq@soict.hust.edu.vn
Giang Nguyen-Linh	Associate Professor/ PhD	HUST, Vietnam	giangnl@soict.hust.edu.vn
Binh Huynh-Thanh	Associate Professor/ PhD	HUST, Vietnam	binhht@soict.hust.edu.vn
Nordin Ramli	Researcher/ PhD	MIMOS, Malaysia	nordin.ramli@mimos.my
Ernest Kurniawan	Research Scientist/ PhD	I^{2}R, Singapore	ekurniawan@i2r.a-star.edu.sg

IV. Project Report

i) Introduction

Internet of things (IoT) with billions of devices/sensors makes it possible to build smart cities/smart nations to improve the quality of life for individuals, and brings benefits and opportunities to enterprises. Wireless sensor network (WSN) is a key enabler for IoT, smart city, and smart nation.

Conventional WSN is usually deployed as fixed infrastructure. The IoT sensors are static devices, gathering data from their permanent and fixed locations and sending the data to gateways at fixed locations. While for some use cases, static and fixed deployment of WSN may simplify network management, it requires a large number of sensors and gateways and is subject to limitations such as power supply and aesthetics. For some other use cases, the application environment may experience frequent changes, e.g., container port where containers move in and out constantly. The communication links, therefore, may sometimes get blocked hence degrading the quality of service (QoS).

Mobile IoT, with mobile gateways installed on moving vehicles to help collect measurements from the sensor nodes, is able to alleviate some of the above issues. The benefits associated with Mobile IoT deployment includes:

- Significant reduction in number of sensors and gateways
- Dynamic deployment, better for situation where fixed deployment is costly /impossible and more robust in propagation-hostile environment
- Improved flexibility and scalability
- Easy setup with minimal fixed infrastructure

Despite those advantages, however, there are challenges that need to be addressed for successful deployment of Mobile IoT system. The most prominent challenge is related to connectivity of the different elements involved in the network. One way to solve the connectivity issue is to use a long range communication such as LoRaWAN technology or the LTE cellular network. Particular to LTE network, it is necessary to ensure that a proper prioritization is given to the different group of nodes such that they do not disrupt the other users in the network. In addition, considering the dynamic environment of the sensor network, the level of significance of the different group of sensor nodes may be different at different time period. As such, a mechanism to dynamically allocate different access prioritization to the different group of nodes is needed. In this work, we propose a *"Priority-adaptive hierarchical access control for dense sensor deployment networks"* to allow different access priority to be assigned dynamically to different group of users.

Without long range communications, the information transfer from the sensor devices to the information sink has to go through multiple hops. In such scenario, a strategic placement of relay nodes to ensure connectivity is needed, while making sure that the number of deployed nodes is kept to a minimum. For this purpose, our work on *"Connectivity optimization with minimum number of deployed sensors"* deals with the problem of node deployment to ensure connectivity between the mobile gateway and the information sink using the minimum possible number of nodes. As far as the connectivity from the sensor nodes to the mobile gateway, our work on *"Target Coverage Optimization with Connectivity constraint"* proposed a mechanism for node placement to ensure that every target location is covered by at least one sensor nodes, and at the same time ensures that all deployed sensor nodes is connected through multi-hop communications to the mobile gateway all the time.

Another important challenge associated with Mobile IoT deployment is to reply to the requirement of mobile property of the Mobile Gateway in a Mobile IoT System or to provide low end-to-end delay (between sensors and Mobile Gateway). Our work on *"Large Scale Heterogeneous Mobile IoT Architecture"* addresses this particular issue in the context of large scale deployment consisting of different Wireless Sensors Networks (each is represented by a Relay Node) and Mobile Gateways. The end-to-end delay between sensors and Mobile Gateway is minimized by: i) Proposing path scheduling mechanisms that minimizes delay between Relay Node and Mobile Gateway. ii) Adaptive Scheduling mechanisms within each WSN that provides low delay between sensors and Relay Node adaptively according to traffic patterns of applications. We model our Architecture according to: i) Hardware features of sensors (Zolertia Z1), Relay Node (Raspberry Pi 3) and Mobile Gateway (unlimited buffer and processing capability). ii) Recent protocol stacks of IoT: UDP/RPL/6LoWPAN/802.15.4e TSCH within each WSN and TCP/IPv4/802.11 within Network of Relay Nodes and Mobile Gateways. iii) Intergrated with a street system. iv) Multi-hop communication.

In order to demonstrate our concept of mobile IoT, and to show the different applications where the technology can be used, we develop a *"Simple Mobile IoT System Testbed with Robotic Base and Mobile Gateway"*. In this simple testbed, the sensors obtain temperature and humidity measurements, and upload the data via mobile gateway, which will then be presented via web application server. Our work on *"LORAWAN-based IoT Solution for Environmental Monitoring"* shows our testbed development for environment monitoring system using LoRaWAN long-range communications technology. Our work on *"Mobile Surveillance Technology applications to Mobile-IoT"* and *"Wireless grid technology applications for to Mobile-IoT"* also shows the testbed development for mobile surveillance system and wireless grid systems, along with the techniques addressing some of the technical challenges associate with those applications. Finally, we develop a small testbed that implement the operation of Adaptive Scheduling mechanism based on TSCH 802.15.4e.

ii) Project Activities

(1) Development and Implement

In the following, we describe each individual topic addressing different challenges associated with Mobile IoT system. The first four parts:

- Priority-adaptive hierarchical access control for dense sensor deployment networks
- Connectivity optimization with minimum number of deployed sensors
- Target Coverage Optimization with Connectivity constraint
- Large Scale Heterogeneous Mobile IoT Architecture

deals with the connectivity issues and multi-hop communications protocol design for Mobile IoT systems. Meanwhile, the last four parts:

- Simple Mobile IoT System Testbed with Robotic Base and Mobile Gateway
- LORAWAN-based IoT Solution for Environmental Monitoring
- Mobile Surveillance Technology applications to Mobile-IoT
- Wireless grid technology applications for to Mobile-IoT

deals with the testbed development and the application of the technology on a particular use case.

Priority-adaptive hierarchical access control for dense sensor deployment networks

Machine-type communications (MTC) is a major part of 5G networks. With the number of connected devices in the order of billions, it is crucial to understand the impact they impose to the network and to mitigate them. Overload control is important to deal with radio access networks (RANs) congestion due to concurrent access for random access channel (RACH) resources, and it has been discussed in the Third- Generation Partnership Project (3GPP) Long-Term Evolution (LTE) working item, namely narrow band Internet of Things (NB-IoT) system [1]. The overload control for RACH resource can be broadly categorized into *push-based* method and *pull- based* method. In the *push-based* method such as access class barring [2], dynamic allocation of RACH resources [3], and MTC-specific back-off scheme [4], the policy controls how the MTC devices access the RACH resource. On the other hand, in the *pull-based* method such as paging and group paging [5], the policy controls the evolved NodeB (eNodeB) in deciding when the MTC devices should access the network.

While having the advantage of central control at the eNodeB, the pull-based method lacks the flexibility to introduce access prioritization among MTC devices within the group. To address this, a number of methods have been proposed, such as dynamically allocating the resources at different preamble retransmission attempts [6]-[7], explicitly assigning dedicated resources to the MTC devices within the group [8], introducing probabilistic model when the MTC device can start the random access (RA) contention [9], and extending the contention period [10], etc. However, none of the above schemes considers differentiation among different MTC devices within the group. One exception is the work in [11], where multiple MTC sub-groups are considered, and prioritization to the different sub-groups is proposed by introducing different RA delays. However, the eNodeB is unable to adjust the sub-group priority on the fly across different paging cycles, and all sub-groups are given access to all available RA resources, making the sub prioritization static.
In practice, there are scenarios where different group of MTC devices may have different significance. The priority of the groups, as well as the devices within the group may change, depending on the situation. This work is motivated to address this requirement by proposing a hybrid group paging method, which takes advantage of the benefits from both pull- based and push-based methods. In addition to being centrally controlled, allowing the eNodeB to schedule the contention period of the MTC devices so as to minimize its impact to the human-to-human traffic, the proposed method is capable of dynamically assigning different priority to different sub-group in real time, a feature that will be useful in scenarios where the importance of different MTC sub-groups may vary at different times. In this work, we adopt legacy LTE system for explanation of the proposed hybrid group paging method and performance evaluation. However, we would like to emphasize that the proposed hybrid method can be also applied to NB-IoT system as the operational principle of paging is the same [12] except for the NB-IoT specific operation such as repetition.

System Model

A cellular LTE network is considered, comprising a single eNodeB and multiple MTC devices which are partitioned into F sub-groups that are paged at the same time. The partitioning of MTC devices can be based on the geographical location, sensing capability, or other application-specific requirements. The number of MTC devices in sub-group $f \in [1, F]$ is denoted as N_f, hence there are a total of $\sum_{f=1}^{F} N_f$ devices within the cell. It is also assumed that the number of RA resources (i.e. preamble sequences) available for the MTC devices is R, and they are shared by all the MTC devices within the cell.

Upon receiving a paging message, the MTC devices will initiate an RA procedure in the next available RA slot to establish connection with the eNodeB as per the LTE specification. If unsuccessful, it will back-off and reattempt the RA procedure in the subsequent RA slot immediately after the back-off counter expires, for up to the maximum number of retries N_{PTmax}. In an LTE network, the period of RA slot is determined by a system parameter called $T_{\text{RA-REP}}$, which is specified by the physical RACH (PRACH) configuration index [Table 5.7.1-2][13]. For example, when the PRACH configuration index 6 is used, the RA slot is available for every 5 milliseconds (i.e. $T_{\text{RA-REP}}$ = 5). Within one paging cycle, the total number of RA slots that are dedicated for the group paging is given by the parameter $I_{\max}$, which means that only $I_{\max}$ consecutive RA slots following a paging message are available for the MTC devices to access the network.

The RA procedure is a two-stage process. On the first stage, each device selects randomly a preamble to be transmitted on PRACH at the RA slot. When more than one devices select the same preamble, collision will occur, and it is assumed that the eNodeB will not be able to decode the preamble. Correspondingly, the devices involved in the collision will have to back-off. At the eNodeB, the received signal is correlated with all possible preamble sequences, a process that takes a period of time (T_{RAR}) to complete, and those that are detected successfully will be acknowledged using RA response (RAR) message. In LTE, a power ramping scheme is adopted every time a device re-attempts the RA procedure, hence the device that has failed a number of times will have a higher chance of being detected at the eNodeB. A simplified model to capture this behavior is to set the detection probability at the eNodeB as

$$p_n = 1 - \frac{1}{e^n} \tag{1}$$

where n indicates the number of tries/attempts the current preamble transmission is at. Also, there is a capacity limit on the number of devices that can be granted with a RAR for each RA slot, as determined by the RAR window (W_{RAR}) and the RAR message size (N_{RAR}). Thus, for every RA slot, there are at most $N_{\mathrm{UL}} = W_{\mathrm{RAR}} \times N_{\mathrm{RAR}}$ devices that can successfully complete the first stage of RA procedure, even though there may be more of them which have selected a unique preamble sequence.

The second stage of RA procedure is for signaling message (i.e. Msg 3 and Msg 4) to set up the connection. Non-adaptive hybrid automatic retransmission request (HARQ) is enabled for both signaling messages to protect their fidelity. Once the RAR message is received, the MTC device will send the radio resource control (RRC) connection request message (Msg 3) to the eNodeB, and wait for acknowledgment (ACK). If no ACK is received, the MTC device will resend the Msg 3 after T_{M3} subframes. Once the eNodeB is able to successfully receive the Msg 3, it will reply with HARQ ACK within T_{HARQ} subframes, otherwise it will reply with HARQ NACK, triggering the MTC device to resend the Msg 3. After sending HARQ ACK packet, the eNodeB will send RRC connection setup message (Msg 4). Similarly, the MTC device will wait for T_{HARQ} subframes and respond with HARQ ACK if it has successfully received Msg 4. Otherwise, the eNodeB will retransmit Msg 4 if it does not receive the HARQ ACK within T_{M4} subframes. For both Msg 3 and Msg 4, the maximum number of retransmissions is fixed at N_{HARQ} times.

Hybrid Group Paging

Group paging has been shown to be capable of significantly reducing the overhead, by overcoming the limitation of the paging message with which only up to 16 devices can be paged. Upon receiving the group paging message, all the devices within the group will try to establish connection to the eNodeB simultaneously, therefore congestion will occur especially when the group size is large. The motivation behind the proposed hybrid group paging is to address this issue by introducing prioritization to different sub-groups within the group that is paged. This prioritization is akin to the property of push-based method, whereby separate resources are assigned to different MTC devices, hence we call the proposed scheme a hybrid pull-and-push group paging method.

The underlying idea behind this proposed hybrid paging method is to assign multiple group IDs to each MTC device. Correspondingly, the fraction of RA resources that can be used by an MTC device is determined by how many group IDs that are matched to the list of IDs within the paging message. There are many ways on how the number of matching group IDs can be related to the fraction of RA resources available to use by a particular MTC device. In the following, we describe one such possibility.

Define a fractions-vector $\boldsymbol{\theta}$ of length $2^{(F-1)}$ as follows:

$$\boldsymbol{\theta} = \left[\theta_1, \theta_2, \cdots, \theta_i, \cdots, \theta_{2^{(F-1)}}\right] \tag{2}$$

such that $\theta_i > \theta_j, \forall 0 < i < j$. Also, for convenience, we set $\theta_0 = 0$. It is also natural to set $\theta_1 = 1$ so that all the RA resources can be utilized. The total number of IDs needed in this approach is $2^F - 1$, which are denoted as $\mathrm{ID}_d, \forall d \in [1, 2^F - 1]$. The assignment of these IDs to the *F* sub-groups follows the *F*-bits binary representation of the ID number *d*, and the least significant bit (LSB) indices that are equal to 1. For example, ID_1 is assigned to sub-group 1 (since the binary representation of *1* is *0001*), and ID_3 is assigned to both sub-group *1* and *2* (since the binary representation of *3* is *0011*). Alternatively, we can denote a binary representation of an integer *d* as d_{bin}, and the f^{th} LSB of the binary representation of *d* as $\mathrm{LSB}_f\left(d_{\text{bin}}\right)$. Using this notation, the list of IDs assigned to sub-group $f \in [1, F]$ is given by

$$\left\{\mathrm{ID}_d : \mathrm{LSB}_f\left(d_{\text{bin}}\right) = 1, \forall d \in \left[1, 2^F - 1\right]\right\} \tag{3}$$

It is apparent that using this assignment rule, the maximum number of IDs assigned to the sub-group is $2^{(F-1)}$, hence the length of the fractions-vector $\boldsymbol{\theta}$ is set to $2^{(F-1)}$.

Upon receiving a paging message, the MTC device on each sub-group *f* checks its list of IDs, and calculates the number of matching IDs contained in the paging message. Given that there are *i* matching IDs, the fraction of RA resources available to use is equal to θ_i; hence for this particular example, the devices in sub-group *f* will only have the first $R_f = \theta_i R$ RA resources to use for performing the RA attempt. To better illustrate the concept of using multiple group IDs, consider *F=3* sub-groups with $2^3 - 1 = 7$ IDs, whose assignment is shown in Figure 1. Furthermore, we set $\theta_i = \dfrac{1}{i}$ for simplicity.

	ID_1	ID_2	ID_3	ID_4	ID_5	ID_6	ID_7
Sub Group 1	■		■		■		■
Sub Group 2		■	■			■	■
Sub Group 3				■	■	■	■

Figure 1 Group ID assignment illustration for F = 3 sub-groups.

We describe the following two example scenarios.

Scenario 1: When the paging message contains ID_3 and ID_7, sub-group 1 and 2 will find two matching IDs, therefore they may use only the first $R_1 = R_2 = \theta_2 R = \frac{R}{2}$ RA resources to perform the RA attempt. Meanwhile, sub-group 3 only has 1 matching ID, therefore it may use $R_3 = \theta_1 R = R$ (all of the available) RA resources. As such, this scenario will give higher priority to sub-group 3, while identical lower priority is given to both sub-group 1 and 2.

Scenario 2: Another example is when the paging message contains ID_4, ID_6, and ID_7. In this case, the sub-group 1, 2, and 3 have in total one, two, and three matching IDs, respectively. Correspondingly, sub-group 3 will have the first $R_3 = \theta_3 R = \frac{R}{3}$ RA resources to use, sub-group 2 will have the first $R_2 = \theta_2 R = \frac{R}{2}$ RA resources to use, while the sub-group 1 will have $R_1 = \theta_1 R = R$ (all of the available) RA resources to use when performing the RA attempt. In this scenario, the order of priority is given by sub-group 1 followed by sub-group 2, and finally sub-group 3 which has the least priority.

As we make the convention that $\theta_0 = 0$, it is apparent that the sub-group will remain idle when none of its ID is found inside the paging message, hence it is possible to perform paging to only a subset of the sub-groups. We would also like to emphasize that the limitation of including only 16 IDs within the paging message is limiting the number of sub-prioritization, especially for the case where there are a large number of sub-groups. As such, the number of choices of assigning different priority among different sub-groups may be limited due to the paging message size limit.

Analysis and Discussions

In this section, we analyze the performance of our proposed hybrid pull-and-push paging scheme. We are particularly interested in analyzing the average access delay, the successful access probability, and the preamble collision probability. Using similar approach as in [14], we employ a recursive contending estimation (RCE) to calculate the desired performance metrics.

First we sort the sub-group index based on the available RA resources for each of the sub-groups (which depends on the number of matching IDs in the paging message), and combine the sub-groups having the same total allocated RA resources. For this purpose, we define a set of unique fractions

$$\phi \triangleq \{\theta : \exists R_f = \theta R, \forall f \in [1, F]\} \quad (4)$$

From the set $\phi = \{\phi_g\}_{g=1}^{G}$, it should be clear that $G \leq F$ since there may be more than one sub-groups having the same number of allocated RA resources. Also, we follow a convention that $\phi_i > \phi_j, \forall i < j$; and for convenience, we set $\phi_{G+1} = 0$. The number of MTC devices in sub-group *g* is

$$M_g = \sum_{f:R_f=\phi_g R} N_f \tag{5}$$

From here onwards, any reference to sub-group refers to this newly defined sub-group. Taking as example the two scenarios described earlier, for **Scenario 1** we have two sub-groups with $\phi_1 = 1$, $\phi_2 = 1/2$, $M_1 = N_1$, and $M_2 = N_2 + N_3$. Meanwhile, for **Scenario 2** we have three sub-groups with $\phi_1 = 1$, $\phi_2 = 1/2$, $\phi_3 = 1/3$, and $M_g = N_g, \forall g \in [1,3]$.

We also denote the total number of contending MTC devices at the i^{th} RA slot as $M^{(i)}$ for $i \in [1, I_{\max}]$, and those from sub-group *g* as $M_g^{(i)}$. It is apparent that $M^{(i)} = \sum_{g=1}^{G} M_g^{(i)}$. In order to differentiate between the MTC devices that are performing different RA attempt retries, we denote as $M^{(i)}[n]$ the number of MTC devices contending at the i^{th} RA slot that are performing n^{th} RA attempt for $n \in [1, N_{\mathrm{PTmax}}]$. Out of these $M^{(i)}[n]$ MTC devices, $\mu_g^{(i)}[n]$ of them come from the sub-group *g*. Again, it should be clear that $M^{(i)} = \sum_{n=1}^{N_{\mathrm{PTmax}}} M^{(i)}[n]$, $M^{(i)}[n] = \sum_{g=1}^{G} \mu_g^{(i)}[n]$, and $M_g^{(i)} = \sum_{n=1}^{N_{\mathrm{PTmax}}} \mu_g^{(i)}[n]$.

In terms of the allocation of RA resources to the different sub-groups, we consider again the example described in **Scenario 2** and illustrate how the *R* resources are shared among the different MTC devices from different sub-groups in Figure 2. From the figure, it can be seen that half of the RA resources (which corresponds to $\phi_1 - \phi_2$) are used solely by MTC devices from sub-group 1. Adjacent to that, 1/6 of the RA resources (which corresponds to $\phi_2 - \phi_3$) are used by the MTC devices from both sub-group 1 and sub-group 2. Finally, the remaining 1/3 of the RA resources (which corresponds to $\phi_3 - \phi_4$) are used by all of the sub-groups.

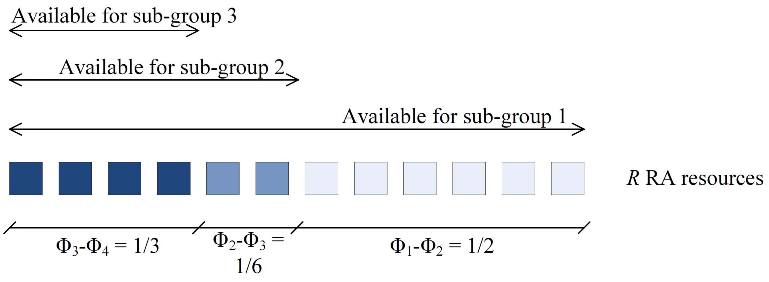

Available for sub-group 3
Available for sub-group 2
Available for sub-group 1
R RA resources
Φ3-Φ4 = 1/3
Φ2-Φ3 = 1/6
Φ1-Φ2 = 1/2

Figure 2 RA resource allocation to different sub-group

The analysis of RA channel makes use of the property of slotted system [15]. Namely, given *N* slots and a total of *M* users trying to access any of the channel with equal probability (occupancy density of *M/N*), the probability that any given slot is occupied by *k* users follows Poisson distribution [16], given by

$$P(k) = \frac{1}{k!}\left(\frac{M}{N}\right)^{k}\exp\left(-\frac{M}{N}\right) \tag{6}$$

Applying the same technique to our hybrid group paging scenario, we can see that during the i^{th} RA slot, $\phi_g - \phi_{g+1}$ fraction of the RA resources has occupancy density of $\sum_{j=1}^{g}\frac{M_j^{(i)}}{\phi_j R}$, hence the probability that any one RA resource in that fraction is selected by *k* MTC devices can be calculated as

$$P_g^{(i)}(k) = \frac{1}{k!}\left(\sum_{j=1}^{g}\frac{M_j^{(i)}}{\phi_j R}\right)^{k}\exp\left(-\sum_{j=1}^{g}\frac{M_j^{(i)}}{\phi_j R}\right) \tag{7}$$

In the beginning when the eNodeB sends a paging message, all the MTC devices with matching group ID will initiate RA attempt in the immediate RA slot. At this first RA slot (*i=1*), all the MTC devices are in the first RA attempt, therefore $M^{(1)}[n] = 0,\ \forall n \neq 1$. Correspondingly, $M^{(1)} = M^{(1)}[1] = \sum_{g=1}^{G}\mu_g^{(1)}[1]$. Out of these $M^{(1)}$ devices, the number of devices that selected unique RA resources can be calculated as $\sum_{g=1}^{G}(\phi_g - \phi_{g+1})\,RP_g^{(1)}(1)$, where $P_g^{(i)}(k)$ is defined in (7). Correspondingly, since all of the contending MTC devices are on their first RA attempt, we can incorporate the power ramping effect defined in (1) to compute the total number of MTC devices that are acknowledged with RAR message as $p_1\sum_{g=1}^{G}(\phi_g - \phi_{g+1})\,RP_g^{(1)}(1)$. However, due to a limitation of RAR grant message, only up to N_{UL} MTC devices can be acknowledged with RAR message.

To perform further analysis, it is necessary to break down the different sub-group among those devices that are acknowledged with RAR message. It can be seen that the MTC devices who selected $(\phi_g - \phi_{g+1})$ fraction of RA resources come from sub-group $j \in [1, g]$, and out of those $p_1(\phi_g - \phi_{g+1})\,RP_g^{(1)}(1)$ MTC devices that are acknowledged with RAR message, the fraction of it that belong to sub-group *j* is $\frac{\mu_j^{(1)}[1]/(\phi_j R)}{\sum_{k=1}^{g}\mu_k^{(1)}[1]/(\phi_k R)}$. Therefore, given the $p_1\sum_{g=1}^{G}(\phi_g - \phi_{g+1})\,RP_g^{(1)}(1)$ non-collided and acknowledged MTC devices, those from sub-group *j* is

$$\sum_{g=j}^{G}(\phi_g - \phi_{g+1})\,RP_g^{(1)}(1)\frac{\mu_j^{(1)}[1]/(\phi_j R)}{\sum_{k=1}^{g}\mu_k^{(1)}[1]/(\phi_k R)}p_1 \tag{8}$$

The above calculation assumes that the MTC devices with unique preamble is less than N_{UL}. For the case where it is larger than N_{UL}, the RAR response message is proportionately distributed among the N_{UL} MTC devices, and the number of acknowledged MTC devices from sub-group *j* is

$$\frac{\sum_{g=j}^{G}\left(\phi_g-\phi_{g+1}\right)RP_g^{(1)}(1)\frac{\mu_j^{(1)}[1]/(\phi_j R)}{\sum_{k=1}^{g}\mu_k^{(1)}[1]/(\phi_k R)}p_1}{\sum_{g=1}^{G}\left(\phi_g-\phi_{g+1}\right)RP_g^{(1)}(1)p_1}N_{\mathrm{UL}} \tag{9}$$

Those devices that did not receive RAR message will initiate a back-off procedure and retry the preamble transmission.

The analysis for the number of contending MTC devices in the subsequent RA slots can be performed in a similar way. Namely, during the i^{th} RA slot, the expected number of MTC devices that select a unique RA resource is $\sum_{g=1}^{G}\left(\phi_g-\phi_{g+1}\right)RP_g^{(i)}(1)$. However, the MTC devices that are contending during this i^{th} RA slot need not necessarily be at the same RA attempt. Out of those MTC devices in the $\left(\phi_g-\phi_{g+1}\right)$ fraction of the RA resources, the portions that is from sub-group *j* that are performing their n^{th} RA attempt is given by $\frac{\mu_j^{(i)}[n]/(\phi_j R)}{\sum_{k=1}^{g}\mu_k^{(i)}/(\phi_k R)}$. Then, the expected number of nodes that are successfully acknowledged with RAR message is

$$\sum_{n=1}^{N_{\mathrm{PTmax}}}\sum_{g=1}^{G}\sum_{j=1}^{g}\left(\phi_g-\phi_{g+1}\right)RP_g^{(i)}(1)\frac{\mu_j^{(i)}[n]/(\phi_j R)}{\sum_{k=1}^{g}\mu_k^{(i)}/(\phi_k R)}p_n \tag{10}$$

Again, there are two possible cases for the above quantity, namely whether or not it is greater than N_{UL}. Using similar approach, we can calculate $\mu_{j,S}^{(i)}[n]$, the number of MTC devices from sub-group $j\in[1,G]$ that are performing their $n\in[1,N_{\mathrm{PTmax}}]$-th RA attempt and successfully acknowledged with RAR message, given by

$$\begin{cases}\sum_{g=j}^{G}\left(\phi_g-\phi_{g+1}\right)RP_g^{(i)}(1)\frac{\mu_j^{(i)}[n]/(\phi_j R)}{\sum_{k=1}^{g}\mu_k^{(i)}/(\phi_k R)}p_n, & \\ \qquad\qquad\qquad\qquad \text{if } (10)\leq N_{\mathrm{UL}} & \\ \frac{\sum_{g=j}^{G}(\phi_g-\phi_{g+1})RP_g^{(i)}(1)\frac{\mu_j^{(i)}[n]/(\phi_j R)}{\sum_{k=1}^{g}\mu_k^{(i)}/(\phi_k R)}p_n}{\sum_{m=1}^{N_{\mathrm{PTmax}}}\sum_{g=1}^{G}\sum_{l=1}^{g}(\phi_g-\phi_{g+1})RP_g^{(i)}(1)\frac{\mu_l^{(i)}[m]/(\phi_l R)}{\sum_{k=1}^{g}\mu_k^{(i)}/(\phi_k R)}p_m}N_{\mathrm{UL}}, & \\ \qquad\qquad\qquad\qquad \text{otherwise.} & \end{cases} \tag{11}$$

It can be checked that substituting *i=1* corresponding to the first RA slot and *n=1* to the above expression, we get back equation (8) and (9) for the two cases, respectively (note that $\mu_k^{(1)}=\mu_k^{(1)}[1]$ as only the first RA attempt is possible in the first RA slot). Also, since during the first RA slot we have $M^{(1)}[n]=0,\ \forall n\neq 1$ is equal to zero for $n\neq 1$.

The number of MTC devices in sub-group *j* that are performing their n^{th} RA attempt but unsuccessful to obtain RAR message can therefore be calculated using

$$\mu_{j,F}^{(i)}[n] = \mu_{j}^{(i)}[n] - \mu_{j,S}^{(i)}[n] \tag{12}$$

Considering that the MTC device cannot switch sub-group when reattempting RA procedure, we can calculate $\mu_g^{(i)}[n]$ from the earlier failed retry $\mu_{g,F}^{(k)}[n-1]$ and those unsuccessful in completing the connection setup message as follows

$$\begin{aligned} \mu_g^{(i)}[n] &= \sum_{k=K_{\min}}^{K_{\max}} \alpha_{k,i}\, \mu_{g,F}^{(k)}[n-1] + \sum_{j=j_{\min}}^{j_{\max}} \beta_{j,i}\, P_{e,\text{msg}}\, \mu_{g,}^{(j)}[n-1] \\ &\approx \sum_{k=K_{\min}}^{K_{\max}} \alpha_{k,i}\, \mu_{g,F}^{(k)}[n-1] \end{aligned} \tag{13}$$

The above calculation uses the same approach as the single group case given in [eqn. (6)][14]. The quantity $P_{e,\text{msg}}$ is the probability that the connection setup (by msg 3 and msg 4) during the second stage is not able to be completed successfully, which is negligible for a typical value of $N_{\text{HARQ}} = 5$ and failed packet transmission probability of *0.1*.

To calculate the $K_{\min}$, $K_{\max}$, and $\alpha_{k,i}$, we use similar analysis provided in [14]. The k^{th} RA slot begins at $t = (k-1)T_{\text{RA-REP}}$. A device that perform RA attempt at the k^{th} RA slot will only start initiating the back-off procedure at $t = (k-1)T_{RA-REP} + T_{\text{RAR}} + W_{\text{RAR}} + 1$. Furthermore, considering the maximum back-off period of W_{BO}, the latest time that the back-off process is completed is $t = (k-1)T_{RA-REP} + T_{\text{RAR}} + W_{\text{RAR}} + W_{\text{BO}}$. In addition, we know that a device will perform the RA attempt on the i^{th} RA slot when its back-off process is complete at any time between $t = (i-2)T_{\text{RA-REP}} + 1$ and $t = (i-1)T_{\text{RA-REP}}$. The value of $K_{\min}$ is obtained when the right boundary of the back-off interval touches the left boundary of the transmission interval of the i^{th} RA slot, resulting in

$$K_{\min} = \left\lceil (i-1) + \frac{1 - (T_{\text{RAR}} + W_{\text{RAR}} + W_{\text{BO}})}{T_{\text{RA-REP}}} \right\rceil \tag{14}$$

Similarly, the value of $K_{\max}$ can be obtained when the left boundary of back-off interval touches the right boundary of the transmission interval of the i^{th} RA slot, resulting in

$$K_{\max} = \left\lfloor i - \frac{T_{\text{RAR}} + W_{\text{RAR}} + 1}{T_{\text{RA-REP}}} \right\rfloor \tag{15}$$

The value of $\alpha_{k,i}$ can be calculated based on the amount of overlap between the back-off window and the transmission interval of the i^{th} RA slot, as follows [eqn. (10)][14]

$$\alpha_{k,i} = \begin{cases} \frac{(k-i+1)T_{\text{RA-REP}} + T_{\text{RAR}} + W_{\text{RAR}} + W_{\text{BO}}}{W_{\text{BO}}} & \text{, if } K_{\min} \le k \le i - \frac{T_{\text{RAR}} + W_{\text{RAR}} + W_{\text{BO}}}{T_{\text{RA-REP}}} \\ \frac{T_{\text{RA-REP}}}{W_{\text{BO}}} & \text{, if } i - \frac{T_{\text{RAR}} + W_{\text{RAR}} + W_{\text{BO}}}{T_{\text{RA-REP}}} < k < (i-1) - \frac{T_{\text{RAR}} + W_{\text{RAR}}}{T_{\text{RA-REP}}} \\ \frac{(i-k)T_{\text{RA-REP}} - (T_{\text{RAR}} + W_{\text{RAR}})}{W_{\text{BO}}} & \text{, if } (i-1) - \frac{T_{\text{RAR}} + W_{\text{RAR}}}{T_{\text{RA-REP}}} \le k \le K_{\max} \\ 0 & \text{, otherwise} \end{cases} \tag{16}$$

Using (14), (15), and (16), $\mu_g^{(i)}[n]$ in (13) can be recursively calculated for any $i \in [1, I_{\max}]$ and $n \in [1, N_{\mathrm{PTmax}}]$.

With the obtained $\mu_g^{(i)}[n]$ for $i \in [1, I_{\max}]$ and $n \in [1, N_{\mathrm{PTmax}}]$, we can calculate the performance metrics of interest. For the preamble collision probability, we use the node-perspective rather than the RA resource-perspective since it does not consider the number of nodes that collide within a particular RA resource [17]. For the MTC devices from sub-group *j*, the collision probability $P_C(j)$ can be calculated as

$$\frac{\sum_{i=1}^{I_{\max}} \left(\mu_j^{(i)} - \sum_{g=j}^{G} \frac{\mu_j^{(i)}/(\phi_j R)}{\sum_{k=1}^{g} \mu_k^{(i)}/(\phi_k R)} (\phi_g - \phi_{g+1}) R P_g^{(i)}(1) \right)}{\sum_{i=1}^{I_{\max}} \mu_j^{(i)}} \tag{17}$$

In the above, the numerator calculates the number of MTC devices that are collided in the i^{th} RA slot, and sum them over all $I_{\max}$ slots. Meanwhile, the denominator gives the total number of contending MTC devices over all $I_{\max}$ slots.

The successful access probability for the MTC devices from sub-group *j* can be calculated by diving those that are successfully acknowledged at any RA slot and any RA attempt, with the initial number of the MTC devices from the same sub-group in the beginning of the paging process as

$$P_S(j) = \frac{1}{\mu_j^{(1)}} \sum_{i=1}^{I_{\max}} \sum_{n=1}^{N_{\mathrm{PTmax}}} \mu_{j,S}^{(i)}[n] \tag{18}$$

In order to calculate the cumulative distribution function (CDF) of the access delay, we divide the total number of MTC devices that are successfully acknowledged by a certain RA slot *d* regardless of the RA attempt, with the total number of MTC devices from the same group that are successfully acknowledged as follows

$$G_j(d) = \frac{\sum_{i=1}^{d} \sum_{n=1}^{N_{\mathrm{PTmax}}} \mu_{j,S}^{(i)}[n]}{\sum_{i=1}^{I_{\max}} \sum_{n=1}^{N_{\mathrm{PTmax}}} \mu_{j,S}^{(i)}[n]} \tag{19}$$

Note that the above CDF for the access delay is calculated in terms of multiples of $T_{\text{RA-REP}}$.

Numerical Results

We will now evaluate the performance of our proposed hybrid group paging and compare it against the conventional group paging. The parameters used throughout the simulation are listed in Table 1. The MTC devices are grouped into three sub-groups (*F=3*), and the number of MTC devices within each sub-group $N_f = N, \forall f$ is varied from 60 to 300 with increment of 30.

Table 1 System Parameters

Parameter	value
PRACH Configuration Index	3
Period of RA slot ($T_{RA\text{-}REP}$)	10 msec
RAR window size (W_{RAR})	5 subframes
Number of RAR in each response message (N_{RAR})	3
Maximum acknowledgement msg (NUL = W_{RAR} x N_{RAR})	15
Back-off indicator (BI)	20 subframes
Back-off window size ($BI + 1$)	21 msec
Preamble sequence size/RA resource (R)	54
Maximum RA retry limit (N_{PTmax})	20
Processing latency at eNB (T_{RAR})	2 msec

The fractions-vector $\boldsymbol{\theta} = [1, 1/2, 1/3, 1/4]$ is used, and the paging message is following the example of **Scenario 2** given earlier, whereby the first, second, and the third group has 1, 2, and 3 matching IDs, respectively. For the conventional group paging, all sub-groups can access all the available RA resources ($\boldsymbol{\theta}^{(\text{conventional})} = [1, 1, 1, 1]$).

Figure 3 shows the collision probability for different sub-group sizes. The solid curves are obtained through Matlab simulation, while the markers represent the theoretical collision probability calculated using (17). It can be seen that the theoretical calculation agrees well with the results obtained via simulation. It is also apparent that the first sub-group achieves the lowest collision probability, while the second sub-group achieves a slightly lower collision probability than the third sub-group. For the conventional group paging case, the achieved collision probability is higher than that from the first sub-group using our proposed hybrid group paging. However, it is better than the second sub-group up to a sub-group size of 210, after which it is comparable to the second sub-group.

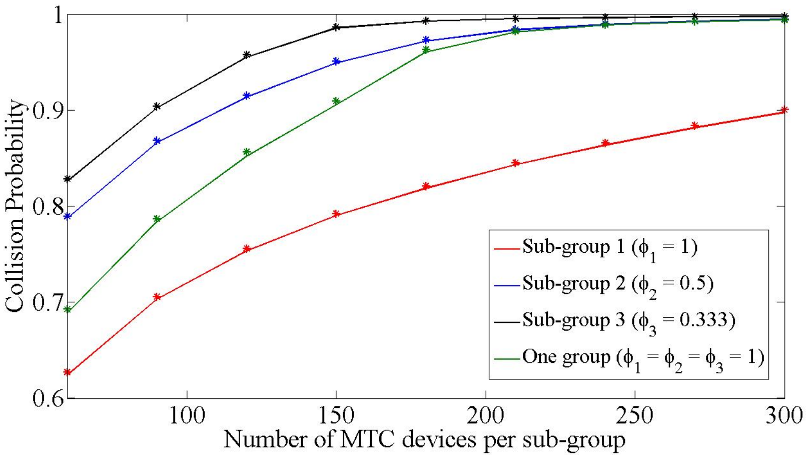


Figure 3 Collision probability for different sub-group sizes

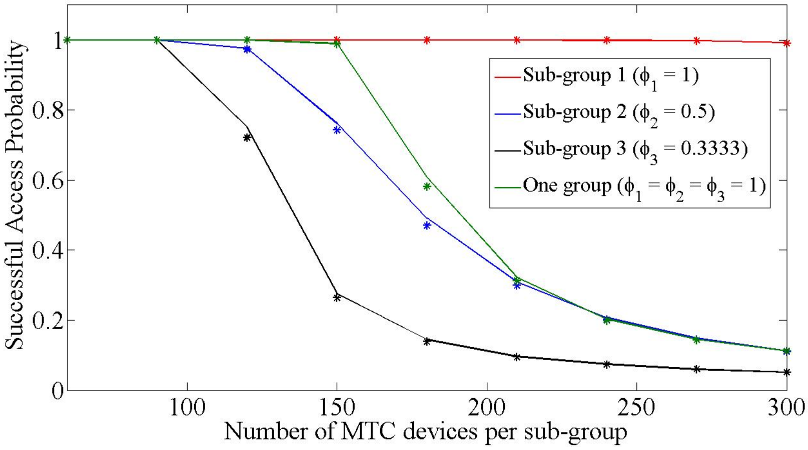


Figure 4 Successful access probability for different sub-group sizes

Similar trend is also observed in the successful access probability, which is shown in Figure 4. Again, the solid line is obtained via Matlab simulation while the markers are obtained using theoretical calculation following equation (18). In this case, the first sub-group is able to achieve very high successful access probability (close to one). Meanwhile, both sub-group 2 and 3 achieve lower successful access probability. As far as the conventional group paging is concerned, the successful access probability is somewhere between the first and the second sub-group (and it is very close to the performance of the second sub-group for sub-group size equal to 210 or larger).

Lastly, the average access delay is shown in Figure 5, where the solid lines are obtained using Matlab simulation and the markers are the theoretical calculation using the formula

$$\overline{G_j} = G_j(1) + \sum_{d=2}^{I_{\max}} d\left(G_j(d) - G_j(d-1)\right) \qquad (20)$$

In the above equation, $G_j(d)$ is the CDF of access delay for the MTC devices in sub-group j given by equation (19). As expected, the first sub-group is performing the best by achieving the smallest average access delay, while the second and the third sub-groups achieve higher access delay. Similarly, the conventional group paging method performs worse than the first sub-group, but it is better than the other sub-groups.

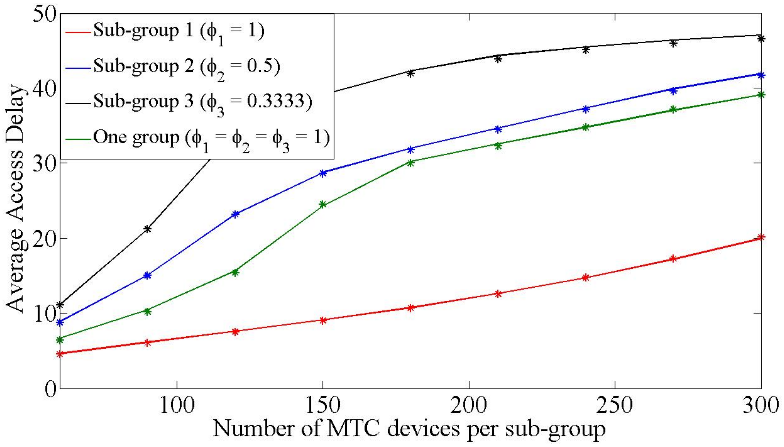


Figure 5 Average access delay for different sub-group sizes

Conclusions

We have proposed a hybrid pull-and-push group paging method for massive machine-type communications. On top of the benefit of central control, the proposed hybrid scheme enables dynamic access prioritization among different sub-groups, which is beneficial for applications where the importance of different MTC devices is varying. The analysis for the collision probability, successful access probability, and the access delay are also presented and shown to agree with the simulation results. Compared to the single group case, the proposed scheme provides better performance to the high priority sub-group at the expense of the low priority sub-group.

Connectivity optimization with minimum number of deployed sensors

Wireless technology in general and wireless sensor networks (WSNs) in particular have been widely applied in a wide range of fields including forest monitoring, disaster forecast, military, internet of things, smart home and healthcare, to name but a few. Additionally, WSNs has enabled a low-cost and low-power deployment. However, regardless of advantages mentioned above, WSN system by itself has severe resource constraints such as energy management, low fault-tolerance, connectivity maintenance, low area coverage and so forth. Therefore, a great number of works have been carried out on various types of WSNs.

Among many kinds of WSNs, mobile WSN (MWSN),which can be referred to as a wireless sensor network (WSN) consisting a base station (sink node) and a set of network nodes distributed in mobile objects such as vehicles and smart phones, appears to be the most propitious. In fact, MWSNs are more versatile than static sensor networks such that they can be deployed in any scenario and cope with rapid topology changes. Moreover, the mobility of systems increases the adaptability of WSNs because sensor nodes are able to move around to collect information more effectively as well as reschedule to achieve different objectives. Despite the great capacity and flexibility coming from the mobility of sensors, MWSNs also contains many challenging problems caused by this property. Not only is the sensors management getting harder but working scheduling is also difficult to get high efficiency. More notably, the connectivity, which acts as one of the most important factors for evaluating the quality of MWSNs is also heavily affected by the continuous change of sensor’s locations. It exerts a great impact on the performance of MWSNs.

From the analysis above, it is clear that studying and proposing effective optimization strategies for solving connectivity problems in MWSNs are strongly demanding. An increasing attention by the research community in the world has been spent on this research topic. In this work, we propose a new model for connectivity optimization problem in vehicular MWSNs.

To be more specific, the new model involves with both static and mobile sensors. While the mobile sensors are attached with vehicles whose routes have been already scheduled, the static ones are used to maintain the connectivity between mobile nodes and the base station. The ultimate mission is to keep the number of static sensors as small as possible while helping the base station gather all information collected from mobile sensors after each period T. To handle this new model, we propose an algorithm which utilizes the K-means clustering algorithm to divide the problem into smaller parts that are solved by a heuristic approach. The remained work is completed by applying Kruskal algorithm, which ensures the connectivity and minimizes the required static sensors.

Our contribution in this topic is as follows:

- Formulating a new model which contains both types of sensors. The combination of mobile and static sensors appears to be fairly desirable since many applications require flexible and proactive systems with simple management tasks. Those properties are easily satisfied by this new model.
- Building the instances for experiment. Our data sets are mainly based on the assumption that mobile sensors are installed on means of transportations. Those vehicle's locations are programmed beforehand after every period of T.
- Proposing an algorithm which introduces a new heuristic method and properly adopts K-means and Kruskal algorithm, to achieve the connectivity of WSNs with a minimum number of static sensors.

Problem formulation

This work investigates the problem formulated as follows: The connectivity optimization problem starts with a two dimensional domain A sized W x H. In that area, there are both a base station and a set of mobile sensors attached on cars whose locations after every fixed period of T are determined. All mobile sensors possess the communication radius Rc .The objective of this problem is to find the minimum number of static sensors which also have the same communication radius with mobile ones to place on such area so that after every period T, the information collected from mobile sensors are able to transmit to base station.

In particular, the problem of connectivity optimization in this model is formulated in details below:

Input:

- *W*, *H*: the width and the length of the 2*D* domain *A* respectively.
- Base station: $s_0(x_0, y_0)$ the 2*D* coordinates of the base station with $(0 \le x_0 \le W, 0 \le y \le H)$
- *M*: the number of mobile sensors on the cars

- *Rc*: the radius communication of both mobile and static sensor.
- *T* is a period of sampling method; *K* is the number of periods
- *CarS* is set up in cars:
  - There is only a sensor in each car.
  - $C_i = \left\{(x_j, y_j) \mid j = \overline{1,K}\right\}$, where (*xj*,*yj*) is the coordinates of car *i*th at *j*th period are the set of the *i*thcar locations.
  - $\mathrm{Car}S = \bigcup_{i=1}^{c} C_i$

Output:

- The coordinates of *n* static sensor nodes.

Constraint:

- Any sensor node $s_i(x_i, y_i)$ must be connected to Base Station $s_0(x_0, y_0)$:

$$\exists M = \left\{s_{i_1}, s_{i_2}, .., s_{i_n}\right\}$$

$$\text{such that} \begin{cases} d(s_i, s_{i_1}) \le R_c \\ d(s_{i_1}, s_{i_2}) \le R_c \\ \ldots \\ d(s_{i_{n-1}}, s_{i_n}) \le R_c \\ d(s_{i_n}, s_0) \le R_c . \end{cases}$$

where: sensor node $s_i(x_i, y_i)$ and sensor node $s_j(x_j, y_j)$,

$$d(s_i, s_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

Objective:

- The number of static sensor nodes $\rightarrow$ *min*

Related works

Recently, MWSNs are getting increasingly popular among researchers of WSNs. With a great flexibility and capacity of reducing the number of necessary sensors thank to their mobility, MWSNs seem to be a promising network structure to deal with acute problems involving with collecting information, tracking objects and monitoring processes. Nevertheless, there are lots of obstacles that hinder the popularity of MWSNs. Because of the mobile nature of sensors, locating and monitoring their tasks are extremely challenging. More importantly, other critical tasks in operating MWSNs such as maintaining the connectivity and planning working schedules turnout to be really complicated compared with the static ones. Such difficulties draw lots of attention of WSN community.

In 2012, an overview about MWSNs was conducted by Javad Rezazadeh et al. [21]. While mentioning some favorable strength of MWSNs such as longer lifetime, channel capacity and data fidelity, those authors also enumerated existing challenges including localization, dynamic network topology, power consumption and mobile sink. In addition, several practical mobility models were also discussed in detail.

In 2007, a survey of clustering algorithms, which are widely applied in WSNs' problems in general and MWSNs' ones in particular was carried out by Ameer Ahmed Abbasi et al. [19]. Depending mainly on the ultimate mission including property, complexity, these algorithms were clearly summarized and categorized into corresponding groups. Other aspects such as the convergence, stability as well as ability of supporting mobility were also carefully considered to improve the precision of the categorization process. There were totally 15 algorithms which have been examined namely LCA, Adaptive clustering, CLUBS, Hierarchical control clustering, RCC, GS3, EEHC, LEACH, FLOC, ACE, HEED, Extend HEED, DWEHC, MOCA and attribute-based clustering. In the conclusion of this survey, the authors also noted that the clustering approach is one of the most popular techniques to build large-scaled systems. Another study involving with MWSNs was introduced in [18]. In this work, the focusing subject was to construct a protocol in order to maintain the physical connection between sensor nodes while providing high quality area coverage. To fulfill these constraints, Spreadable Connected Autonomic Network (SCAN), which not only maintains the network connectivity but also offers low price was proposed. Analysis on the experimental result pointed out that SCAN is bound to be effectively applied on moderately large environment which might have been complicated terrain and keep the systems connected with probability of 99%. Based on co-operation between the information collected from surroundings and the predicting model, SCAN is also capable of handling noise and obstacles that might emerge in reality.

In term of MWSN applications, many researches have been carried out. In [23], Uichin Lee and Mario Gerla conducted a survey which focused on the usage of urban vehicles sensing. More specifically, in their model, sensors are attached to means of transportations to collect and share data among elements of network. This survey thoroughly studied the whole process of information transmission as well as facilities management and concluded that the efficiency of a certain system is dependent to many aspects such as accessing methods, system mobility and sensors deployment. Another application of MWSNs was also carried out by Uichin Lee et al. in [24]. Based on the idea of using the flow of vehicles to diffuse information, the authors proposed MobEyes, which appeared to be an efficient method to support urban supervision. With the great autonomic capacity, MobEyes Diffusion/Harvest Processor (MDPH) is able to expand to thousands of sensor nodes while keeping process of transmitting information smoothly.

In 2013, Mohamed Amine Kaf et al. [22] introduced some applications and constructions of MWSNs for traffic monitoring. So as to deal with the traffic jams which are caused by the increase of the number of means of transportations, the authors compensated current weakness of Intelligent Transport System (ITS) by making use of MWSNs and thus, created a smart traffic network which is easy to expand with low price.

Although the amount of attention to WSNs and MWSNs remarkably increases as shown by many related works, there are still lots of obstacles which have not yet been tackled. Therefore, we attempt to deal with a new model of MWSNs in which sensors are both attached to vehicles and deployed in fixed locations. More precisely, giving a base station and a set of mobile sensors, the main task is to minimize the necessary number of static sensors while still being able to keep the system connected after each period *T*. To fulfill those constraints, our proposed algorithm will make use of K-means clustering and Kruskal algorithms which play important roles in maintaining the connectivity of MWSNs. In addition, we will introduce a heuristic approach to minimize the number of needed sensors. We expect our algorithm will deliver solutions having good quality and stability.

Proposed Algorithm

This work proposes a modified K- means algorithm and Kruskal algorithm (M2K) including three phases to deal with the main problem. In the first phase, K-means algorithm is applied to divide all positions in set *CarS* into clusters with corresponding centroids. After that, our heuristic algorithm named integrated greedy method (IGM) is introduced to ensure the connectivity in each cluster with minimum static sensor nodes. At the last phase, we are going to use Kruskal algorithm to again claim the connectivity in the whole systems at every sampling-time while keeping the number of static nodes low.

Clustering phase: The aim of this phase is to simplify the problem by two processes. In the former one, we treat all the positions in set *CarS* equally regardless of their belonging periods and cars. This action does not violate the constraint of our model since the static sensors deployment will eventually have to create a connection between all locations in *CarS* and base station after all.

In the latter process, M2K adopts K-means cluster algorithm [25], which is a frequent technique to handle large sensors network to divide points in set *CarS* to different group. K-mean is one of the simplest and most effective algorithms that handle the clustering problem [25]. Its main idea is to initialize *k* centroids, each of them belongs to one cluster. Denote $P=\{p_1,p_2,..,p_n\}$ be set of data points and $C=\{c_1,c_2,..,c_k\}$ be the set of centroids, $k \le n$. *C* is randomly created. After calculating the Euclidean distance between each sampling point to all centroids, the algorithm then assigns this point to the cluster containing its nearest centroid. At this point, we need to re-calculate the new cluster center using:

$$c_i = \left(\frac{1}{v_i}\right)\sum_{j=1}^{v_i} p_{ij} \qquad (21)$$

where v_i represents the number of data point in i^{th} cluster and p_{ij} is j^{th} point of i^{th} cluster.

The process of calculating distance between each data point and newly obtained centroids is started over again. If no data point was reassigned, the algorithm will terminate; otherwise, the cluster center recalculation will be continued [25]. The main step of K-means clustering algorithm is illustrated in Figure 6.

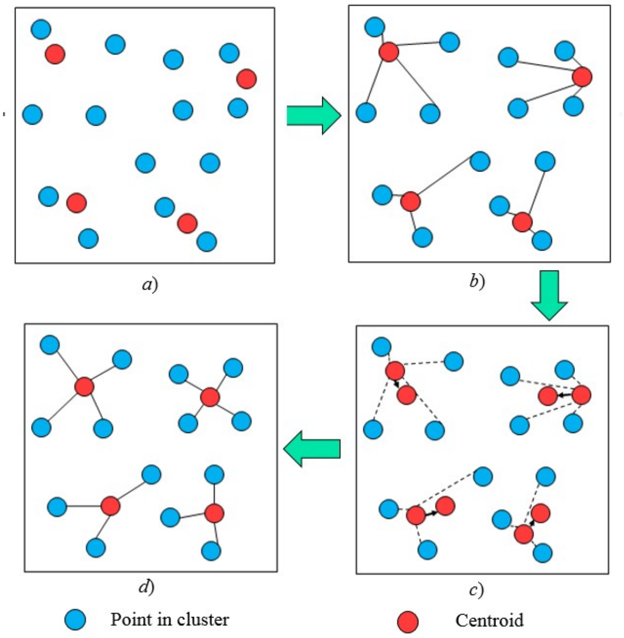


Figure 6 K-means clustering algorithm

After completing the clustering phase, M2K will turn to the next one, which mainly maintains the connectivity in each cluster.

<u>Integrated Greedy Method:</u> In the first phase, all points representing for concerned locations where mobile sensors have to connect to base station at certain time are successfully clustered. In this phase, the main goal is to build the connectivity in clusters before heading to the final phase: making cluster connected to base station.

To fulfill this constraint, we introduce a heuristic method whose name is Integrated Greedy Method (IGM). Let$S_i$be the set of static sensors required to be deployed to maintain the connectivity in i^{th} cluster. Initially, S_i contains only one sensor node that is deployed in the locations of i^{th} centroid. The main idea of this method is to allow the i^{th} cluster's unconnected point U in which possess the smallest distance to a sensor in S_i, namely M, to be bound to transmit information with elements in S_i. This task could be achieved by placing new static sensors to the segment between M and U. The number of needed sensors denoted as w for this process is calculated as follow:

$$\mathrm{w} = \left\lceil \frac{d(M,U)}{R_c} \right\rceil \qquad (22)$$

where *d*(*M*,*U*) is the Euclidean distance between *M* and *U*.

Newly deployed sensors are going to be added to set S_i. The minimum distance between unconnected points and sensor nodes in S_i will be updated. The process will terminate after all points in i^{th} cluster are completely connected to system of sensors in S_i. The pseudo code of this heuristic method could be described as follows.

Algorithm 1: Integrated Greedy Algorithm for k^{th} cluster

```
Input:
        Set $V_k = S_k \cup \{c_k\}$
        The communication radius of sensors: $R_c$.
Output: The deployment of static sensors
$DS = [(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)]$
Begin
    Create an array min containing the minimum distances between points in $V_k$ and a static sensor in DS.
    Create an array used to mark points that have not yet been connected.
    $DS.add(c_k(x_{c_k}, y_{c_k}))$;
    $n = |S_k|$;
    for $i = 1 \to n$
        $min[i] = d(c_k, S_k^i)$;
        $used[i] = 0$;
    endfor
    while(n>0)
        Choose the point p that $used[p] = 0$ and $min[p]$ is minimum among unconnected points.
        Deploy set of static sensors (T) to build connection between p and $S_k$.
        Denote $T = \{t_1(x_{t_1}, y_{t_1}), t_2(x_{t_2}, y_{t_2}), .., t_l(x_{t_l}, y_{t_l})\}$;
        $DS.add(T)$;
        $used[p] = 1$;
        for $i = 1 \to n$
            If( $used[i] == 0$ )
                for $j = 1 \to l$
                    If $(d(t_l, s_k^i) < \min[i]) \min[i] = d(t_l, s_k^i)$;
        n = n-1;
    endwhile.
End.
```

In fact, this method fully makes use of the deployed static sensors. While mobile sensors which appear at points in ith cluster are not permanently present because of their mobility, static sensors appear to be more reliable thanks to their fixed communication area. By repeatedly choosing the shortest distance to place static sensors and using the information of those newly added for further deployment, our algorithm is expected to use a small number of static sensors in this phase.

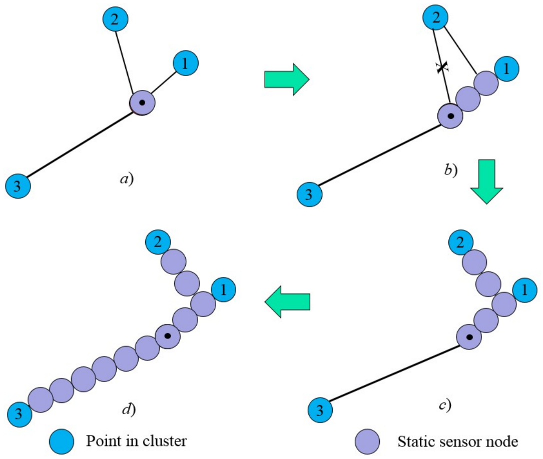


Figure 7 Integrated Greedy Method (IGM) steps.

a) Deploying a static sensor in the positions of cluster center.
b) Choosing the point with the shortest distance centroid and deploying sensors then updating newly shorter distance for unconnected points.
c) Among other unconnected point, pick one that is nearest to a placed static sensor and make those two connected.
d) Repeat this process until the connectivity is achieved.

<u>Minimum Static Sensor Nodes Ensure Connectivity with Base Station:</u> After the job done in the second phase, all points in ith clusters are connected with sensors in S_i. The rest of work to claim the connectivity of the whole system is to build the connection between the base station and centroid of all clusters.

Considering the base station and the set of center clusters as a complete weighted graph G = (V, E), V={s_0} $\cap C$, C is final results of centroids calculated in the clustering phase. Weight of edge e whose vertices are v_1 and v_2, v_1, $v_2 \in V$ is calculated by formula [19]. In order to keep the constraint that maintain connectivity between mobile sensors and the base station while minimizing the number of static ones, we employs the Kruskal algorithm [20], which are capable of satisfying two objectives at the same time. In reality, Kruskal algorithm will construct a minimum spanning tree in order to ensure the connectivity and optimize the total edge weight, which are equal to the necessary sensors.

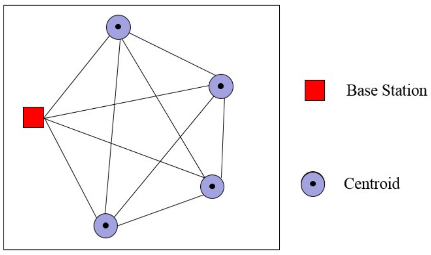


Figure 8 The complete graph whose vertices are centroids and base station.

The final static sensor deployment is the combination of static sensors used in the second and the last phase.

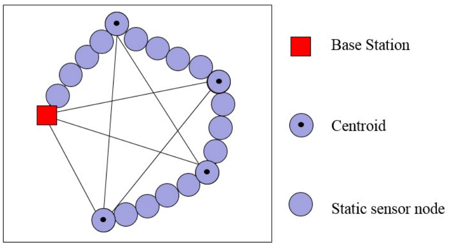


Figure 9 The minimum spanning tree constructed by Kruskal algorithm which ensures the optimization of M2K.

Experimental Results

Problem Instance: This work experiments M2K on 27 instances whose parameters are represented in Table 2.

Table 2 Experimental Instances

(No.*CarS* = number of sampling cars, No.*Ts* = number of periods, R_c = communication radius)

Instance	No. *CarS*	No. *Ts*	R_c
S4_10_12	10	12	4
S6_10_12			6
S8_10_12			8
S4_10_24	10	24	4
S6_10_24			6
S8_10_24			8
S4_10_36	10	36	4
S6_10_36			6
S8_10_36			8
S4_15_12	15	12	4
S6_15_12			6
S8_15_12			8
S4_15_24	15	24	4
S6_15_24			6
S8_15_24			8
S4_15_36	15	36	4
S6_15_36			6
S8_15_36			8
S4_20_12	20	12	4
S6_20_12			6
S8_20_12			8
S4_20_24	20	24	4
S6_20_24			6
S8_20_24			8
S4_20_36	20	36	4
S6_20_36			6
S8_20_36			8

- We assume that the domain *A* which is used for experiment is a square region of 2*D* dimension of the size 100 x 100.
- In the experiment, each instance differentiates by the number of cars, periods and communication radius of static sensors.
- In each instance, the coordinates of base station are provided. The locations of cars after each period are calculated by the information of previous position and the velocity of each car (15 per period).
- In clustering phase, K-means algorithm used parameters calculated in Table 3. This parameter is chosen after many experimenting times and by the observation that the number of static sensors is proportioned to the number of periods and inversely proportioned to the communication radius.

Table 3 Experiment Parameters

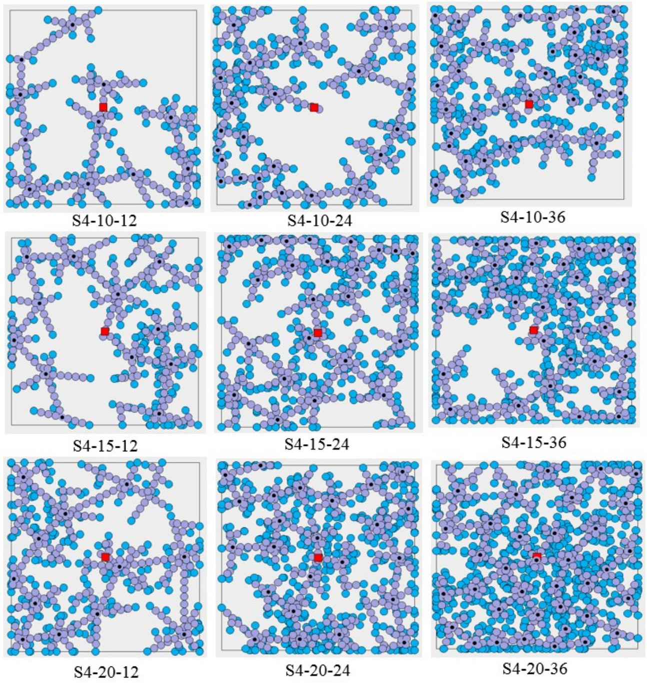
Experimental	Parameter
Domain *A*	100x100
Speed *CarS*	15 per period
Number of Clusters	$No.Ts \times 16 / (c \times R_c \times R_c)$

System Setting: The instances that are created for the experiments on M2K are specified by their parameters.The number of sampling cars are 10, 15 and 20. The number of periods is 12, 24 and 36. The communication radius is 4, 6 and 8 respectively. Each instance is tested over 30 times and the average results are shown in Table 4. The experimental program was implemented in Java language and run on a machine with Intel Core i5 2.5 GHz, RAM 4GB, Ubuntu 64-bit.

Table 4 The average number of static sensor nodes required to maintain connectivity (No. SSN is the average result of static sensor nodes)

Instance	No. SSN	Instance	No. SSN	Instance	No. SSN
S4_10_12	172.47	**S4_15_12**	219.53	**S4_20_12**	254.63
S6_10_12	112.17	**S6_15_12**	139.53	**S6_20_12**	157.33
S8_10_12	74.87	**S8_15_12**	95.57	**S8_20_12**	105.6
S4_10_24	230.37	**S4_15_24**	286.07	**S4_20_24**	324.1
S6_10_24	146.4	**S6_15_24**	179.03	**S6_20_24**	200.27
S8_10_24	103.7	**S8_15_24**	121.8	**S8_20_24**	135.2
S4_10_36	279.43	**S4_15_36**	327.83	**S4_20_36**	369.03
S6_10_36	176.1	**S6_15_36**	203.13	**S6_20_36**	226.67
S8_10_36	124.5	**S8_15_36**	143.5	**S8_20_36**	156.5

Computational Results: The average results from experiments on 27 instances by applying M2K are verified by Table 4. The best solutions derived from each instance is illustrated in Figure 10, Figure 11, and Figure 12, where the red square is the base station, the black dots stand for cluster centers, the blue and purple circles represent the locations of mobile sensors and static ones respectively.

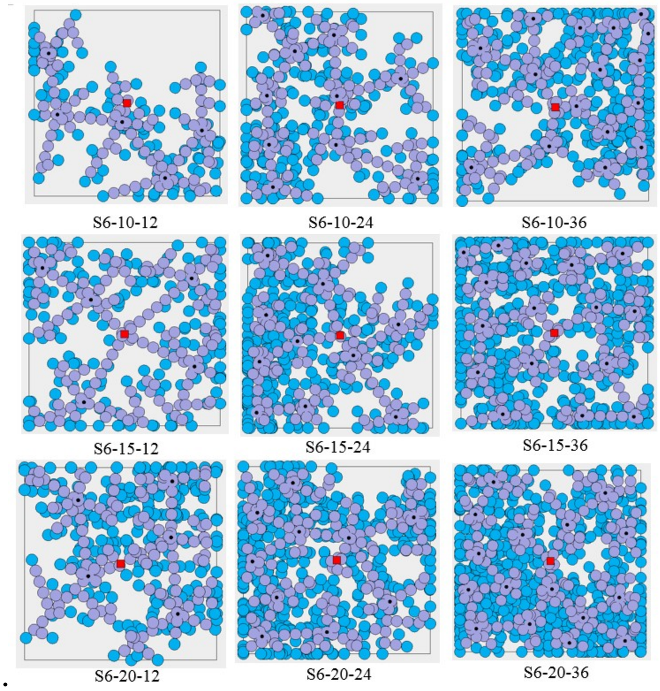


S4-10-12 S4-10-24 S4-10-36

S4-15-12 S4-15-24 S4-15-36

S4-20-12 S4-20-24 S4-20-36

Figure 10 The best solutions given by M2K over 30 running times on the instances with *R_c*=4.



S6-10-12 S6-10-24 S6-10-36

S6-15-12 S6-15-24 S6-15-36

S6-20-12 S6-20-24 S6-20-36

Figure 11 The best solutions given by M2K over 30 running times on the instances with *R_c*=6.

From the above results, we can see that:

- The communication radius has a great impact on the performance of M2K. To be more specific, the number of static sensors required for connectivity decreases significantly when the communication radius grows. In reality, the average number of needed sensors when R_c=8 is about 44,33% that of instances which have R_c=4. It could be explained that the communication radius, which represents for the capacity of transmission are able to make the connectivity more reachable.
- The number of mobile sensor, in the same way, strongly affects the outcome of M2K. When the number of mobile sensors increases, the number of locations needed to be constructed the connection to base station also expands. Therefore, the static sensors required when M = 20 grow up to roundly 140, 12% in compare with those ones in case M = 10.
- With the same effect on M2K, the number of periods, which are able to increase the positions of mobile sensors, causes the great increase in static sensors. This is proven by the fact that the results of M2K in case the number of periods is equal to 12 increase approximately 65,89% comparing to that of periods in the situation when the number of periods is 36.

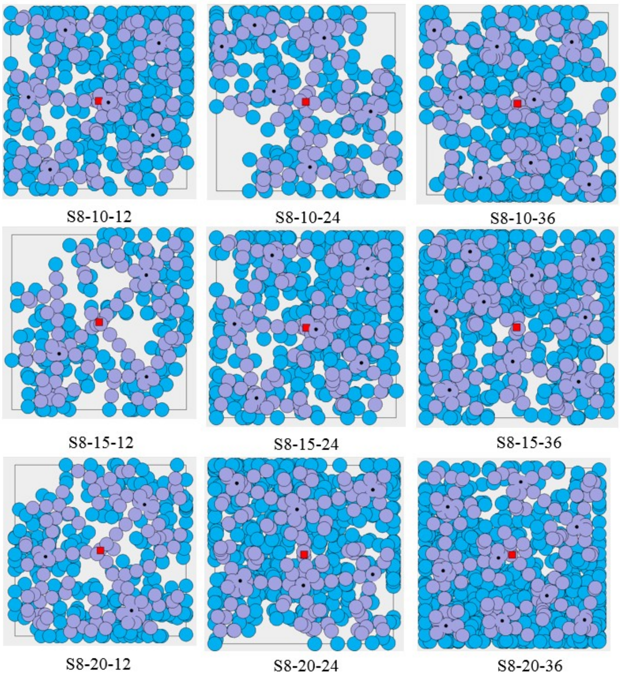


Figure 12 The best solutions given by M2K over 30 running times on the instances with R_c=8.

Conclusion

This work deals with a new model of WSNs which involved the usage of mobile and static sensors. While the call for a more flexible wireless system is increasing, this model seems to be likely to satisfy this demand and used in many applications such as air pollution measurement or traffic jam investigation. In this model, our main objective is to minimize the static sensors needed to secure the connectivity in the whole systems at certain time. By employing K-means clustering and an integrated greedy method (IGM), we are able to succeed maintaining the connectivity in each cluster in the first and second phase of the proposed algorithm. The last phase which aims to build the connection between clusters and base station is done by Kruskal algorithm, creating a minimum spanning tree in the graph forms by centroids and base station. M2K is experimented with 27 instances and the results illustrate how M2K effectively handle the connectivity of the new model.

In future work, we will discuss more detail about other aspects of this model including working schedules and fault-tolerance. Some deeper investigation about practical applications will be also conducted to avoid effects on performance such as lifetime or energy constraints.

Target Coverage Optimization with Connectivity constraint

Wireless sensor networks (WSNs) have been widely used in many applications such as environment monitoring, battle field surveillance, Internet of Things and so on [26], [27]. Typically, WSN consists of a large number of tiny sensors that have the capability to monitor the surrounding, gather information and transfer to sinks. Conventionally, static sinks are exploited to carry out this purpose. However, since all sensors data are forwarded to the sinks, the sensors located in the proximity of the sinks may be imposed a heavier traffic and deplete energy much earlier than the other sensors. Eventually, this will lead to the problem of sink isolation [28]. To tackle this issue, sink mobilization has been introduced [29], in which the mobile sinks can move around to collect information of interest from static sensors.

One of the most challenging issues in WSNs with mobile sinks is to satisfy the following two requirements: (1) all targets are covered by statics sensors (i.e., these sensors are called as target-covering sensors), and (2) all target-covering sensors are connected with at least one mobile sink. We call these two requirements as targets coverage and connectivity, respectively. In the literature, many researches have been devoted to achieve these two requirements. A majority of the existing works have focused on designing mobile sink trajectory in order to guarantee the abovementioned requirements while optimizing some performance factors such as energy consumption, network lifetime, throughput, etc. [30], [31], and [32]. In [30], the author exploited Hilbert Curve Order to design mobile sink trajectory based on network density. The proposed algorithm is proved to achieve full network coverage and better packet delivery ratio. Aiming at reducing latency in data gathering, A. Kinalis et al., proposed a sink mobility scheme with adaptive stop times in [31]. In this proposal, the stop times of mobile sinks are probabilistically adjusted based on some factors such as surrounding node density, remaining energy, and so on. The author in [32] designed an optimal mobile sink trajectory along the perimeter of a hexagon tiling that can improve the network lifetime, reduce the energy consumption while maintaining the sink's connectivity.

In this work, we do not focus on optimizing mobile sink trajectory, but we address the problem of sensor placement with given mobile sink trajectory, instead. We assume that, the mobile sinks move on a random trajectory with a constant speed, and they stop every T period of time for gathering the sensory data from the static sensors. Based on such scenario, we aim at optimizing the number of deployed sensors in order to achieve both the two requirements mentioned above (i.e., targets coverage and connectivity). Firstly, to achieve the full targets coverage with a minimum number of static sensors, our main idea is using K-means clustering algorithm to divide the targets into groups and place static sensors such that one sensor can cover as many targets as possible. Secondly, to connect all static sensors in each cluster, we use Kruskal algorithm to find the minimum spanning tree and place the minimum number of static sensors on the edges found. After that, a greedy algorithm is applied to ensure connectivity between static sensors and mobile sinks at every period of time.

Related Works

Recently, the mobile sink is widely regarded as a mean of relieving traffic burden and enhancing energy efficiency in WSNs. Therefore, many researches have been working on this topic. In 2011, Shuai Gao et al., in [33] proposed a novel data collection scheme called the maximum amount shorter path (MASP) and solved it by making use of a genetic algorithm. Miao Zhao et al., in [34], considered adopting (mobility and space – division multiply access (SDMA)) and (mobile data gathering and SDMA (MDG-SDMA)) to efficient data gathering. Because the MDG-SDMA was NP-hard, the authors proposed an approximate solution by a heuristic algorithm.

Furthermore, studying the effects of different trajectories of the mobile sink and providing important insights for designing mobility scheme in read-world mobile WSNs are promising research topics in the field. Determining a proper placement for sensor nodes is another design strategy to achieve the required performance goals such as network coverage, lifetime, delay and throughput. In 2014, Saim Ghafoor et al., [30] considered an efficient trajectory design for the mobile sink. In this paper, the focusing subject was to take advantage of contact opportunities offered by Hilbert Curve, between the mobile sink, sensor nodes, and the shape of the curve to cover the whole network. In 2016, S. Abirami et al., [35] considered mobile sink based coverage and connectivity to maximize network lifetime of WSNs. In the study, the Meridian Cover Tree (MCT) algorithm with mobile sinks was proposed to achieve coverage as well as connectivity of each sensing node. The target of the MCT algorithm is to sustain full sensing coverage and connectivity of WSNs for a long time.

An extensive series of studies concerning with target coverage and connectivity problems have been proposed [36], [37], and [38]. For example, Zhao et al., [36] developed an approximation algorithm with a distributed version, named Communication Weighted Greedy Cover (CWGC), to maximize the number of cover sets. Hanh et al., [37] solved target coverage with connectivity fault-tolerance problem. One heuristic algorithm was proposed to find a sensors' placement scheme with the largest target coverage and was able to maintain the network connectivity even when one random node defected with the goal using minimize the number of sensors. Besides, Hanh et al., [38] proposed a new model involving connectivity insurance in MWSN. Given a certain number of mobile sensors, a base station *B* and a period *T*, the model asks to find out the minimum number of static sensors required to maintain the connection from the mobile ones to *B* after every time period *T*. To deal with this problem, an algorithm named M2K is proposed.

To wrap things up, there are numerous challenges in WSNs as aforementioned. This work discusses target coverage and connectivity model using mobile sinks in order to overcome mentioned challenges. The main task is to minimize the number of sensors needed to cover all targets and to maintain the connection from each sensor to at least one mobile sink at every period of time *T*. An algorithm that combines K-means clustering and Kruskal algorithm as the cores is proposed to maintain covering and connectivity in WSNs by using mobile sinks. Moreover, we introduce a heuristic method to minimize the number of sensors. We expect our new model and algorithm to achieve good results when testing on practical datasets.

Problem Formulation

In this section, we first give definitions of notations used throughout this work, and then formulate our problem. We consider a network deployed in an interest area with the size of $W \times L$. The network consists of *N* targets and *M* mobile sinks. The locations of the targets are predefined and fixed while the locations of the mobile sinks are changed after every fixed period of time *T*. We assume that after every period, each mobile sink moves a distance of *D* in a random direction. Given the positions of the mobile sinks and the targets, our objective is to deploy sensors such that: (1) each target is covered by at least one sensor, we call a sensor that covers targets *target-covering sensor*; (2) each *target-covering sensor* is connected to at least one mobile sink at every period of time; (3) the total number of deployed sensors in all periods is minimized.

Denote $d(X,Y)$ as the Euclidean distance between two points *X* and *Y*.

Denote the *i*th target as P_i and its coordinates as $\left(P_i(x), P_i(y)\right)$

Denote the *i*th mobile sink as M_i and its coordinates at the *j*th period as $\left(M_i(x,j), M_i(y,j)\right)$.

Denote S^j as the set of static sensors that are newly added at the *j*th period: $S^j = \left\{S_1^j, .., S_{Q_j}^j\right\}$, where S_i^j is the *i*th sensor whose coordinate is $\left(S_i^j(x), S_i^j(y)\right)$.

Denote *K* as the number of the periods.

We assume that all sensors and mobile sinks have the same transmission range, *R_t*, and all sensors have the same sensing range, *R_s*. We adopt the widely-used model (i.e., disk-based) for sensing and transmission, as well as the assumption of $R_t = 2R_s$ [39].

Our problem then can be formulated as an optimization problem as follows:

$$\text{Minimize} \quad \sum_{j=0}^{K} Q_j$$

Subject to

$$\forall k = \overline{1, K}, \forall n = \overline{1, N}$$

$$\exists S_i^j \in \bigcup_{u=1}^{k} S^u, \exists \left\{S_{i_1}^{j_1}, \ldots, S_{i_v}^{j_v}\right\} \subseteq \bigcup_{u=1}^{k} S^u, l \in \{1, ..M\}$$

$$s.t.$$

$$d\left(P_n, S_i^j\right) \le R_s$$

$$d\left(S_i^j, S_{i_1}^{j_1}\right) \le R_t$$

$$d\left(S_{i_h}^{j_h}, S_{i_{h+1}}^{j_{h+1}}\right) \le R_t, \ \forall h = \overline{1, v-1}$$

$$d\left(S_{i_v}^{j_v}, M_l\right) \le R_t$$

Proposed Algorithm

We propose two meta-heuristic algorithms to deal with this model named KHGD and KHKGD. Both KHGD and KHKGD are divided into two phases. In phase I, we cover all targets with a minimum number of static sensors and in phase II, we ensure connectivity between static sensors and mobile sinks at every period of time. Both two algorithms have the same phase I, which uses K-means algorithm and a heuristic method to handle the problem. The difference comes from the phase II. In KHGD phase II only uses a Greedy algorithm to reach the goal that all sensors are connected to mobile sinks in every period of time. Although it gives a good solution, the running time is high. To decrease the running time, we propose KHKGD algorithm to ensure connectivity in each cluster before ensuring connectivity to mobile sink. We will go through the phases of these algorithms.

Phase I (Covering all targets with minimum number of static sensors): The main goal of this phase is to cover all targets with the least static sensor nodes. In order to reduce the complexity, we use divide and conquer strategy. We divide this phase into two stages. Firstly, K-means algorithm is applied to divide all targets into clusters, each of which contains a centroid [19] that are easier to handle. Secondly, to rule each cluster, a proposed heuristic algorithm named Heuristic Integrated Greedy Method (HIGM) is applied to achieve the full coverage in each cluster with minimum sensor nodes.

The K-means algorithm starts with initializing *k* centroids, each of which belongs to one cluster. Denote $C = \{c_1, c_2, \dots, c_k\}$ the set of centroids, $k \le n$. The set *C* is initialized randomly. The next step is to assign each target to the cluster that has the closet centroid, calculated using Euclidean distance matrix. After all targets have been assigned, we recalculate the positions of *k* centroids using the following formula:

$$\begin{cases} c_i(x) = \left(\dfrac{1}{|V_i|}\right)\displaystyle\sum_{j=1}^{v_i} P_{\text{ij}}(x) \\ c_i(y) = \left(\dfrac{1}{|V_i|}\right)\displaystyle\sum_{j=1}^{v_i} P_{\text{ij}}(y) \end{cases} \tag{23}$$

Where V_i represents the set of targets in i^{th} cluster with $|V_i|$ is item of V_i; $(P_{\text{ij}}(x), P_{\text{ij}}(y))$ is the coordinate j^{th} target of i^{th} cluster, and $(c_i(x), c_i(y))$ is the coordinate of *i*th centroid.

The process of assigning the targets to the clusters and re-locating the centroids are repeated until the locations of the centroids become unchanged after re-locating. Figure 13(*a*) illustrates K-means algorithm.

After clustering targets with K-means algorithm, we have a set of clusters $V = \{V_1, V_2, V_3, \dots, V_k\}$. Now we move on to the second stage of phase I, which uses static sensors to fully cover all targets. To this end, we propose HIGM, a greedy algorithm. Denote S_i as the set of static sensors node that needs to cover all targets in the i^{th} cluster. Firstly, we choose a target A located the furthest to the centroid c_i of the i^{th} cluster and put a static sensor on the line connecting A and c_i, which ensures that that static sensor covers the target A and is as near the centroid c_i as possible. The reason we propose this heuristic method is that we expect the other targets would be in the sensor's sensing range with high probability. After placing the static sensor, we remove all targets that reside in the static sensor's sensing range and recalculate the location of the c_i. The above process is repeated until all targets are covered by static sensors. Figure 13(*b*) illustrates our HIGM algorithm.
We keep applying this step to all target clusters until all targets are covered by static sensor nodes. The final result is illustrated in Figure 13(*c*).

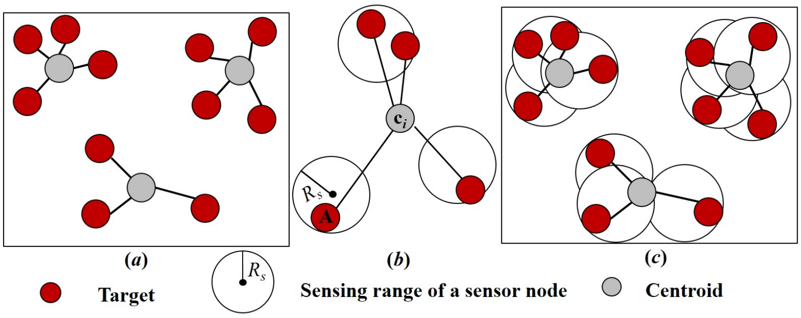


Figure 13 Illustration for covering all targets using K-means algorithm:
***a*): Illustration for clustering all targets. *b*): Illustration for Heuristic Integrated Greedy Method (HIGM) in a cluster. *c*): Illustrated all target covered by static sensor nodes**

Phase II (Ensuring connectivity with minimum number of static sensors): After phase I, all targets have been covered by static sensors. We call a sensor covering a target *target-covering sensor*. In the following, we describe our algorithm to determine static sensors that should be added in each period $T_i\left(i=1,..,K\right)$ in order to ensure the connectivity between target covering sensors and mobile sinks. This is the different point between KHGD and KHKGD, KHGD uses only greedy approach to achieve the goal, while KHKGD utilizes both greedy and Kruskal algorithm.

a) KHGD's phase II

We define a connected component as a set of sensors connected to one or more sink nodes; a static connected component (SCC) is defined as connected component that contains only static sensors; a mobile connected component (MCC) is defined as connected component which contains at least one sink. Our goal is to place additional static sensors such that all SCCs are connected with at least one MCC at any period of time T_i.

The main idea of this method is to find and connect a pair of SCC and MCC until all SCC are connected to at least one MCC. In other words, when a SCC is connected to a MCC, it belongs to the MCC, we keep connecting until no SCC left. By the beginning of this phase, we initiate a set of SCCs from set of static sensors such as with one static sensor, we create a SCC containing only this sensor. At period of time $T_i\left(i=1,..,K\right)$, by the same way, we create set of MCCs from set of sink locations at period T_i. While the set of static connected components is not empty, find the pair of nearest SCC and MCC and connect them. When the set of SCC is empty, we ensure that all static sensors are connected to at least one sink. We move to next period of time T_{i+1} with the input is the set of MCCs where all locations of sinks at period T_i are removed from connected components.

To be more specific, this greedy method is illustrated in the following pseudo-code:

Algorithm: Greedy for Connectivity in WSNs

Input: *numberOfPeriod*: Number of period

sensorsList: list of static sensors after phase I

Output: list of all sensors ensuring every sensor connect to at least one mobile sink at every period of time.

```
SCC_list = $\varnothing$;
for i = 1 to |sensorsList|
    Create a SCC SCC with sensor sensorsList[i];
    SCC_list.add(SCC);
for i = 1 to numberOfPeriod
    MCC_list = $\varnothing$;
    for j = 1 to |M[i]|
        Create MCC MCC with sink M[i][j];
        MCC_list.add(MCC);
    endfor
    while( |SCC_list| > 0)
        Find the pair of nearest SCC and MCC;
        Connect the SCC and the MCC;
        Move all static sensors of SCC and added sensors to the MCC;
        Delete the SCC from SCC_list;
    endwhile
    for m = 0 to |MCC_list|
        if (|MCC_list [m]| <= 1) remove(MCC_list [m]);
        m--;
    endfor
    for m = 0 to |MCC_list|
        Find the mobile sink of MCC_list[m] and remove it;
    endfor
    SCC_list = MCC_list;
endfor
```

b) KHKGD's phase II

Although KHKD can achieve all objectives of this model, KHGD has a weakness that is high running time because it has to handle sensor by sensor. To solve this problem, we propose another algorithm called KHKGD which has the same phase I as KHGD but the second phase of it is divided into two smaller stages to reduce the computation time. To be more specific, in stage 1, we make a pre-processing by using Kruskal to ensure connectivity in each sensor clusters before the move to stage II which applies the same greedy method of KHGD.

In the first stage, we ensure connectivity in each cluster by finding the minimum spanning tree (MST) [20] of a complete undirected weighted graph $G_i(V_i, E_i)$ where V_i is set of positions of the static sensor in cluster *i* and E_i is the set of edges. We propose to use Kruskal algorithm to deal with this stage. Kruskal algorithm is directly based on the generic MST algorithm which builds the MST as a forest. Initially, each vertex is in its own tree in the forest. Then, the algorithm considers each edge in turn, order by increasing weight. If an edge (*u, v*) connects two different trees, (*u, v*) is added to the set of edges of the MST and two trees connected by an edge (*u, v*) are merged into a single tree. Otherwise, if an edge (*u, v*) connects two vertices in the same tree, it will be discarded. After building the MST, we place static sensors to the edges found to ensure every two adjacent vertexes are connected. By the end of the first stage of phase II, all static sensors in each cluster are connected with a minimum number of the static sensor. Each connected cluster we call a static connected component. We move on to the final stage of this phase which ensures connectivity between static sensors and mobile sink at each period of time $T_i\left(i=1,..,K\right)$.

The main target of the second stage is to ensure connectivity between the static sensors and the mobile sink at each period of time $T_i\left(i=1,..,K\right)$. In this stage, the same greedy method with KHGD is used to deal with this problem whose input is a set of static connected components from stage 1. After this stage, all sensors are connected to at least one sink with better computation time than KHGD's. The final static sensor deployment is the combination of static sensors used in the first and the second phase.

Performance Evaluation

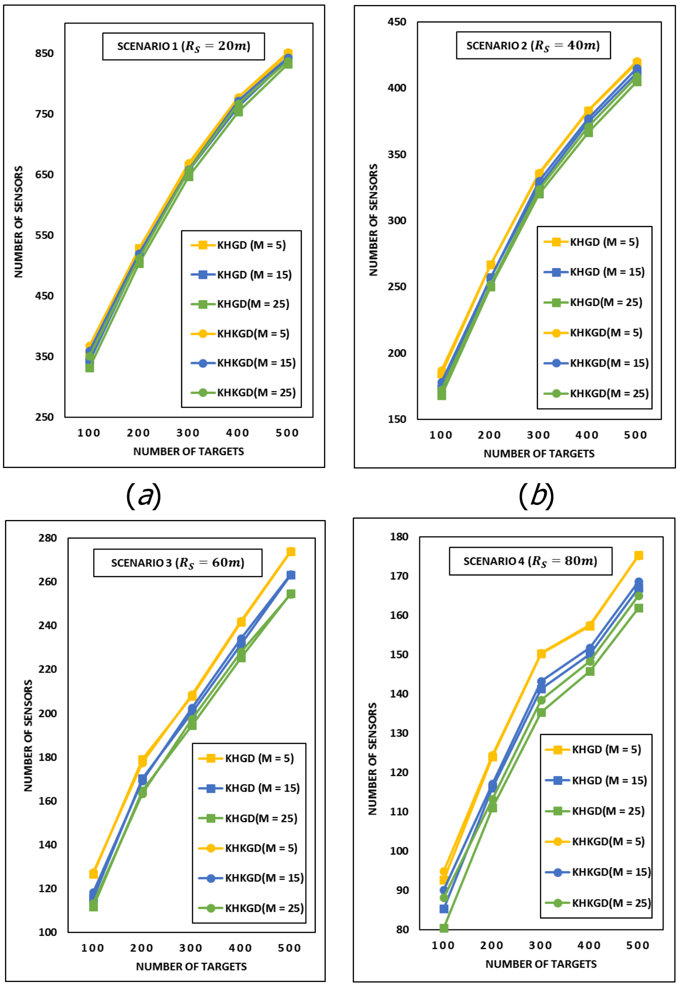
In this section, we will demonstrate how well our proposed algorithms can solve this problem. In the following, we first describe our simulation scenario in simulation setup and then use this scenario to compare performance of proposed algorithms in computational results.

Simulation setup: The simulation was implemented on an Intel(R) Core(TM) i7-5500U (2.4GHz), RAM 8GB computer using Windows 10 Professional 64-bit in Java language. This work experiments the proposed algorithm on 4 scenarios (there are 9 instances in each scenario) whose parameters are shown in Table 5. (R_S as the sensing radius; *A* as the dimension of domain; *K* as the number of the periods; *D* as movement distance of mobile sink after every period; *M* as number of mobile sink, *N* as the number of target). We tested each instance of this model over 30 times and the average results with respect to 4 scenarios are recorded.

Table 5 Scenarios for simulation

Scenario	***R_S***	***A***	***K***	***D***	***M***	***N***
Scenario 1	20 (*m*)	2000 x 2000 (*m^2*)	24 (*periods*)	15 (*m*)	5 15 25 (*pcs*)	100 200 300 400 500 (*pcs*)
Scenario 2	40 (*m*)					
Scenario 3	60 (*m*)					
Scenario 4	80 (*m*)					

Experiment results: We compared KHGD and KHKGD over 4 scenarios with two factors: running time and number of sensors. The experimental data show that KHKGD offered better performance than KHGD did in term of running time with the pre-processing using Kruskal, KHGD gives a better solution in term of the number of required sensors. Specifically, Figure 14 illustrates the performance of two comparing algorithms in term of quality of the solution. Figure 15 illustrates the computational time of two algorithms.



(*a*) (*b*)

(*c*) (*d*)

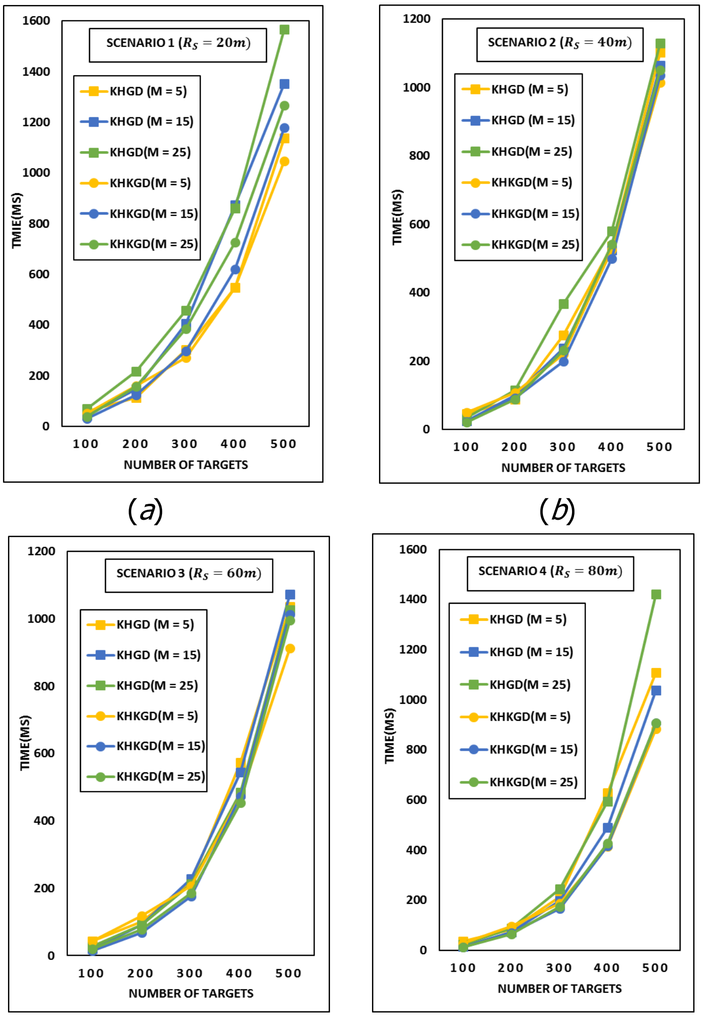
Figure 14 Comparison number static sensor nodes between KHGD and KHKGD algorithm with 4 scenarios.

In terms of number of sensors, Figure 14 shows that the sensing radius of sensors has a great impact on the performance of KHKGD and KHGD algorithm when applying to this model. As we have mentioned before, KHGD has a better performance in term number of required sensors than KHKGD. To be more specific, in most cases, using KHGD decreases the number of sensor by 0.25% - 5.4% compared to KHKGD.

In both case, the sensing radius has a great effect to the outcome. The number of required sensors when R_s = 80 (*m*) in Fig. 2(*d*) is about 19% - 26% that of instances with R_s = 20 (*m*) in Figure 14(*a*). This is understandable because when the sensing radius grows (meaning that communication radius grows in our model), sensors become more reachable, offering better connectivity for the network.

In terms of the number of mobile sinks, in the same way, it also strongly affects the outcome of KHGD and KHKGD. When the number of mobile sinks increase, then sensors required reduces by 3% - 14% on comparison between two scenarios *M* = 25 and *M* = 5 in Figure 14. It can be explained that when the number of mobile sinks increases, the sensors is easier to reach mobile sinks in most of the time, hence the number of sensors decreases as well.

In terms of the number of targets, when the number of targets increases from 100 to 500, the number of sensor increases by 84% - 151% as shown in Figure 14. It is easy to explain that when the number of targets increases, the number of required sensors to cover all targets also increases. This makes the number of required sensors increase.



(*a*) (*b*)

(*c*) (*d*)

Figure 15 Comparison running time between KHGD and KHKGD algorithm with 4 scenarios.

In terms of time complexity, by comparing the running time of KHGD and KHKGD in 4 scenarios as shown in Figure 15, we can see that when the number of targets increases, the running time also increases. In addition, with the same number of targets, the more mobile sensors results in a shorter running time. However, the most important point is that, under the same scenario, KHKGD's runtime is less than KHGD's by about 6% - 50%. This is because KHKGD uses Kruskal's algorithm to ensure connectivity in the cluster as a pre-processing before using greedy to connect sensors and mobile sink. KHKGD has a better running time than KHGD that uses lots of computing resources for greedy method from the beginning of phase II. In reality, KHKGD could be used in scenarios which have a big number of targets and mobile sinks in a large interest area because it is better on computing performance. For other cases, we could use KHGD because it gives better solutions.

Conclusion

In this work, we have proposed two meta-heuristic algorithms to locate the guaranteeing target coverage and connectivity in WSNs with mobile sinks. The main goal is using the minimum number of sensors. To deal with this model we propose two algorithms called KHKGD and KHGD. While the former can achieve the goal with small running time, the latter gives a better solution in the term the number of sensors.

Large Scale Heterogeneous Mobile IoT Architecture

In a large scale heterogeneous sensors networks based on Internet of Things (IoT) with distinct traffic, data collected needs to be transmitted towards the center in a multi-hop manner by implementing different IPv4/IPv6 protocol stacks (CoAP/UDP/RPL/6LoWPAN/ 802.15.4 or MQTT/TCP/IP/802.11). In recent studies, partitioning this large scale heterogeneous sensor networks into different WSNs (each is represented by a Relay Node – RN) and using Mobile Gateways (MGs) as mechanical carriers to collect data from these RNs has been shown to be an effective way (because of significant reduction of numbers sensors, dynamic deployment, etc.) but requires the optimization of the trade-off between system's end-to-end delay, packet loss, energy consumption. Due to the interference and contention over the wireless medium, the limited bandwidth of the radio medium, the limited resources of the battery-operated sensor nodes, this optimization is challenged and requires the appropriate design of data transmission process at each WSN and at network of RNs. In this research, we exploit the multi-channel operation capability of the radio medium and schedule the regular traffic emitted at each WSN by exploiting the past history to find the most appropriate transmission's sequence of packet. In addition, data generating rates at each WSN are not identical, data collecting rates at each RN may vary and some RNs need to be visited more frequently than others in the network of RNs. Thus, we then propose some data collection mechanism that schedules the movements of MGs in a RN's network such that minimizing the data loss due to buffer overflow at RN. We have built this whole architecture on IoT protocol stacks in Contiki (UDP/RPL/6LoWPAN/802.15.4e) and Java/Linux (TCP/IP/802.11) and realize extensive simulations for evaluating the trade-off between end-to-end delay and packet loss of this large scale heterogeneous mobile IoT architecture.

Since the earliest days, Wireless Sensor Networks (WSNs) was envisioned to undertake intensive data collection inside a sensing area. However, the limited energy storage of tiny sensors involved in a WSN largely confines their ability of transmitting the acquired data. As a result, the untimely death of various sensors interfere the connectivity of the network. To address this problem, a novel data gathering technique which combines controlled mobility elements and rendezvous-based data collection [41][42][43] has been introduced. In this approach, sensors first forward their sensing data to several rendezvous points called Relay Nodes (RNs), and then, a Mobile Gateway (MG) is instructed to go to each RN to pick up the cached data for further processing.

In such mobile system, there are several challenges need to be addressed in terms of communication from RN to MG and data transmission between sensors. From the RN-MG communication perspective, there are various types of *real-time* constraints in response to the fast-moving MG must be considered. For example, MG may require RN to process and send data within 1s before moving out of the sensing range. To date, there is a broad range of literatures on improving the utility of this mobile-based strategy. Despite these methods were able to present optimal or near optimal solutions, most of them are based on several impractical assumptions. For example, the link quality is assumed fair enough, data is generated at all sensors at the same rate and it is possible to aggregate data when transmitting. These assumptions are not achievable in practice due to heterogeneous properties, which is the term for interference phenomenon, different data generation rates of different sensors and some infusible data (such as battlefield images) [40].

From the view of data transmission between sensors, achieving a low-latency guarantee for sensor networks is challenging due to a variety of reasons. Although the real-time performance is a key concern, it should be compatible with many other critical issues, such as *energy efficiency* due to the limited power of sensor nodes. For example, keeping sensors always-on only for the purpose of a fast response is inefficient. This naive approach severely reduces system lifetime because turning on the sensors' radio device is actually battery draining. In addition, a large-scale network of unreliable wireless links makes the *packet delivery rate* not quite suitable for low-delay detection. Thus, the two most important objectives of mobile system are *low delay* and *energy efficiency* associated with *packet delivery rate*, which cannot be met concurrently. Therefore, the design of a mobile system requires a trade-off between these considerations.

So far, most of state-of-the-art researches on mobility-based strategy focus only on one aspect of the challenges. In [41], [42], and [43], the authors introduced the mobility strategy for data collection but did not consider the effects of heterogeneous properties as well as data transmission between sensors. Despite attentions to the heterogeneous properties of a mobility system in [40], the author only focused on rendezvous planning in WSNs. Because communication mechanism between sensors is not trivial, it must be taken into account in real-world deployment.

In this work, we propose a combined approach to address these two challenges in a mobile system. The first part of our approach focuses on the connectivity of sensors in each WSN, in which a transmission scheduling algorithm is proposed in combination with Time Slotted Channel Hopping (TSCH) feature of 802.15.4e standard. The second part of our approach addresses the RN-MG communication challenges. We propose mechanisms aimed at finding optimal paths for MGs to visit all RNs in network such that the end-to-end packet delay is minimized. For experimental purpose, we deploy the whole model with Internet of Things (IoT) protocol stack in Contiki and Java programming language, and then run the simulations for evaluation in Linux operating system.

Our contributions can be summarized as follows:

1. A transmission scheduling algorithm based on Reinforcement Learning (RL) technique for WSN's 802.15.4e multi-channel communication protocol. More concretely, we apply an RL-based algorithm to utilize the scheduling part in TSCH feature of 802.15.4e standard.
2. A data collection mechanism that plans the movements of MGs in a RNs network such that minimizing the end-to-end packet delay. More specifically, we propose two algorithms, one is Maximum Weight Sum First (MWSF) in case of one MG and the other is K-Means clustering combined with Simulating Annealing in case of many MGs.
3. We have built this whole model with IoT protocol stacks in Contiki and Java programming language and Linux operating system. We perform extensive simulations for evaluating the trade-off between end-to-end delay and packet-loss of the model.

System Model

Figure 16 gives an intuition on our large scale heterogeneous mobile IoT architecture. A mobility-based approach includes controlled mobility elements and rendezvous-based data collection points. In Figure 16, there is only one mobile element (or mobile gateway - MG) indicated as a green car, whose capability is to go to rendezvous points to collect data. The squares illustrates rendezvous points (called relay nodes – RN), which store data coming from all sensor nodes (in this work, sensors and sensor nodes are used interchangeably) in associated WSN. We address two main challenges of this situation. One challenge is how sensors in a WSN transmit data to the RN, the other is planning the movement of the MGs that minimizes the end-to-end packet delay. These challenges are addressed in the tradeoff between end-to-end packet delay and packet delivery rate.

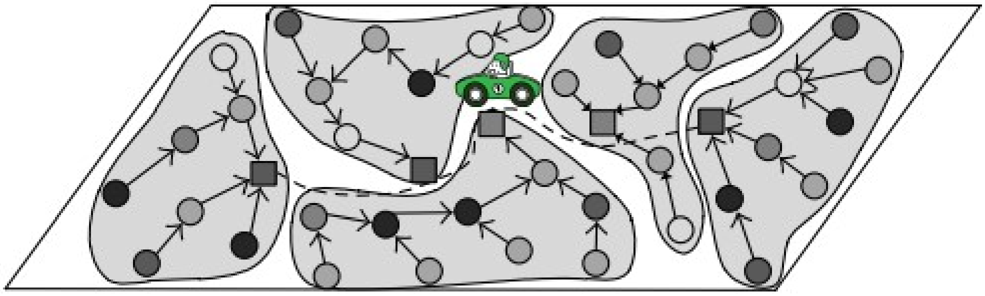


Figure 16 Large Scale Heterogeneous Mobile IoT Architecture

We give some basic assumptions to our model:

1. We assume that wireless communication is the dominating energy-consuming factor. As a result, we omit other energy factors such as sensing and processing.
2. We assume that MGs roams in the network of RNs to collect all sensory data by visiting a series of RNs.
3. We assume that the storage capacity of sensors is large enough to buffer the sensing data during a collection periods (which is the amount of time between two data collection turns).

Some people may argue that the buffer size at each RN should be considered as a metric to address the challenge because if a RN is not visited after a long time, the data would be lost due to buffer overflow phenomenon. However, in practice, this circumstance would be overcome by choosing a device with strong computing ability for RN role. For example, we can choose Raspberry Pi models for RNs (most of them have at least 1 GB of memory and storage could be extended by NAND flash with ultralow power consumption). Increasing data collection frequency is also a practical solution to this problem.

Our model differs from traditional mobility-based models in three points:

1. We mainly focus its application on large scale scenarios, in which a WSN is not interfered by other WSNs. This implies the data collecting range of WSNs are independent, and all data in a WSN is sent to only one RN. In this model, the MGs, whose capabilities are to collect the data, must visit all RNs to get the data in a data collection phase.
2. Our model considers the heterogeneous properties of WSN, which means it takes into account the nature of communication medium, different data generation rate at different

sensors and so on. This property of our model implies its realistic applications.

3. Our model considers the communication between sensors in a WSN and between RNs and MGs based on IoT architecture. More concretely, all sensors in each WSN implements IoT protocol stack up to transport layer, which includes UDP-RPL-6LoWPAN-802.15.4e, and data transmissions between RNs and MGs are upon TCP/IP/802.11. All data communications are achieved over wireless medium.

Related Works

This section presents the works related to the aforementioned challenges in the first section: the connectivity problem of sensors and the RN-MG communication. We present below some relevant literatures in sensor connectivity and dynamic data collection with multiple mobile agents (or mobile gateways).

The field of data transmission in recent years has seen a variety of advances. Many standards have been issued by international bodies to support the development of Wireless Sensor and Actuator Network (WSANs) in different application domains. They include IEEE 802.15.4 [44], ZigBee [45], Bluetooth [46], WirelessHART [47] and ISA-100.11a [48]. In the same period of time, the Internet Engineering Task Force (IETF) has defined a number of protocols to integrate smart objects (i.e., sensor/actuator devices) into the Internet**.** The most important of them are the IPv6 over Low power WPAN (6LoWPAN) [49] adaptation layer protocol, the Routing Protocol for Low power and Lossy networks (RPL) [50], and the Constrained Application Protocol (CoAP) [51] that enables web applications on smart objects.
The IEEE 802.15.4 standard defines the physical and Medium Access Control (MAC) layers of the protocol stack, which is considered the reference standard for commercial WSNs and widely appears in many applications. Many studies have investigated the IEEE 802.15.4 performance in WSANs [52][53]. These works highlighted that IEEE 802.15.4 has a number of limitations (such as low communication reliability and no protection against interferences/fading) that make it unsuitable for applications having stringent requirements in terms of latency, reliability, energy efficiency or operating in harsh environments [54]. In order to overcome such limitations, in 2008 the IEEE set up a Working Group (named 802.15 Task Group 4e) with the aim of enhancing and adding functionality to the 802.15.4 MAC, so as to address the emerging needs of embedded applications, which leads to the release of the 802.15.4e standard in 2012 [54].

The 802.15.4e standard comes with a variety of new features for the aim of adapting to different situations. In order to achieve highly-reliable low-power networking, Time Slotted Channel Hopping (TSCH) [55] is the most notable feature in the 802.15.4e standard due to three reasons: i) Firstly, the feature helps avoid the external interference followed by a rapid increase in number of WSN applications deployed in the same ISM bandwidth. ii) Secondly, it allows multiple simultaneous transmissions on different channels to increase the network packet delivery rate. iii) Thirdly, a carefully-constructed data transmission schedule can help nodes to avoid useless transmission or reception, the main causes of a node’s energy waste. One of the most important parts of TSCH is the scheduling algorithm, which assign links to nodes for communication. Although the TSCH feature provides a default scheduling algorithm called minimal scheduling [55], it is still received strong attention from the research community. Palattella et al. propose a centralized scheduling algorithm, which considers both network topology and traffic conditions of every single node. In [56], Simon Duquennoy et al. proposed an autonomous scheduling method based on setting rules.

The application of Reinforcement Learning (RL) technique in data transmission in WSN has been studied by Kieu-Ha Phung et al. in [57]. In [57], the authors proposed a joint-model of data collection and routing, in which a node chooses a neighbor for data transmission based on feedbacks from previous transmissions. The design of the joint-model was based on time-slotted idea and was experimented successfully in a real test-bed and over the base of 802.15.4 standard. Our transmission scheduling algorithm is developed based on this algorithm, which aims at integrating it to TSCH feature and putting it to the IoT protocol stack.

In recent studies of data collection from sensors, using mobile agents (MAs) as mechanical carriers to relay data has been shown to be an effective way of prolonging sensor network life time and relaying information in partitioned networks. In 2004, Arun A. Somasundara et al. in [58] proposed a mobile agent scheduling for efficient data collection in WSNs with dynamic deadlines. The authors proved this problem of deciding whether a valid schedule exits was NP- Complete and gave an Integer - Linear - Programming (ILP) formulation for it. For that reason, [58] proposed some heuristic algorithms for the kind of dynamic scheduling named Earliest Deadline First (EDF), EDF with k-Look-ahead, Minimum Weighted Sum First (MWSF). In the conclusion of this paper, the authors also noted that introduction a scheduling problem where the mobile agent needs to visit the nodes so that node of their buffers overflow, the usage of controlled mobile agent is a promising approach to collect data from these sensor node and showed that minimum weighted sum first algorithm performs well with computationally inexpensive cost. In this context, the authors expected to find an approximation algorithm for this problem.

Another application of mobile agent scheduling with dynamic deadlines was also carried out by Arun A. Somasundara et al. in [60]. Based on the idea of using mobile agent schedule with dynamic deadline in [58], the authors improved mobile agent schedule with multiple mobiles and also looked at the modification of the vehicle routing problem with time windows for solving this problem. The performances of the heuristics were compared and the experimental results showed that the modified vehicle routing problem with time window gave better results in most of the cases.

The usage of mobile agent to harvest data has recently been considered in the literature. In 2006, Yao Yao Guetal et al. in [59] proposed a Partitioning-Based Scheduling (PBS) algorithm to address this problem. They presented that schedules the movements of MAs in a sensor network such that there is no data loss due to buffer overflow. The authors compared their algorithm with Minimum Weighted Sum First algorithm and showed that their PBS algorithm provided higher performance such that the loss rate and the minimum required speed are reduced while providing high predictability.

In 2011, Ryo Sugihara and Rejesh K.Gupta in [61] investigated the problem of planning the motion of data mules for collecting the data from stationary sensor nodes in WSNs. Using data mules significantly reduces energy consumption at sensor nodes compared to commonly used multi-hop forwarding approaches, but it has a disadvantage such that it increases the latency of data delivery. Therefore, optimizing the motion of data mules, including path and speed, is critical for improving the data delivery and increasing the data mule approach’s performance in practice. This problem is NP hard and the authors approached it by using approximation algorithms for both single-data mule and multiple-data mule cases.

Another study involving with mobile sensor data collection using data mules was introduced in [64]. In this paper, the focusing subject was to construct the mule minimization problem (MMP) with mules being mobile and the objective was to minimize the number of mules required to collect data from all sensors within a pre-specified time. The MMP proved NP complete. The author provided solutions for the MMP by transforming the problem to a generalized version of the minimum flow problem in a network, and then solving it optimally using Integer Linear Programming.

A dynamic model for efficient data collection in WSNs with mobile sink was introduced by Deepak Puthal et al. in [63], which used a mobile sink to collect data. They presented a mobility model for the sink to move in the network and cover the whole network area and proposed Mobile Sink Wireless Sensor Network (MSWSN) model which prolongs the network lifetime. They compared the simulation results with the existing protocol with static sink and found that the proposed model gives the better result in terms of throughput, residual energy and lifetime of the network.

In term of vehicular traffic application, many researches have been carried out. In [62] Jerry John Kponyo et al. conducted a study which focused on novel Distributed Intelligent Traffic System (DITS) and proposed an Ant Colony Optimization (ACO) to solution. The author showed DITS a novel idea which uses ACO to improve vehicular traffic is feasible.

In 2007, Ameer Ahmed Abbasi et al. introduced in [19] overview about clustering algorithms in WSNs. The authors discovered influence of algorithms based on convergence rate, cluster stability, cluster overlapping, location-awareness and support for node mobility.

Despite of the fact that some merits of MWSNs are mentioned in [21] by Javad Rezazadeh et al. such as longer lifetime, channel capacity and data constancy, the authors also listed current issues, for instance localization, network topology, energy cost and mobile sink. Moreover, there were various mobility models being considered in detail.

Regarding urban vehicle sensing in WSNs, Mohamed Amine Kafi et al. in [22] demonstrated methods to create a smart traffic network which have low construction as well as operation. Beside, a survey of urban vehicle sensing platform was conducted by Uichin Lee et al. in [23]. This survey explored recent developments with weakness factors and challenges of vehicle sensor networks.

In spite of the large number of papers involved with WSNs, there are still many obstacles which have not had been tackled yet. Our model focuses on solving two main challenges: data transmission between sensors and data collection using mobile agents. More precisely, with a given set of fixed sensors divided into groups and mobile agents which are categorized into two types: controlled and uncontrolled, the former problem focuses on how data is transmitted in each group of sensors, whereas the latter asks for an agents scheduling that minimize the end-to-end packet delay of all packets over all sensors and the travelling distance of mobile agents. For the first problem, we propose a protocol for data transmission in each WSN. For the second problem, we propose two algorithms. Our first algorithm is a heuristic one which properly utilized K-mean and Minimum Weighted Sum First to construct the schedules of agents. In the second algorithm, we adopt the schedule based on the heuristic algorithm in the initialization phase and apply simulated annealing to search for better results. Our algorithms are expected to deliver good solutions in term of travelling distance and end-to-end packet delay.

Proposed Protocol for Data Transmission in WSN

Here, we propose an algorithm for data transmission in the WSN. The proposed transmission scheduling algorithm is based on Reinforcement Learning (RL) technique and is integrated into the TSCH feature of 802.15.4e standard.

Overview of our solution: In short, we utilize the pre-built architecture of Time Slotted Channel Hopping (TSCH) feature [55] in 802.15.4e standard. For the scheduling part of the feature, we use at the same time the minimal scheduling algorithm [55] and RL-based scheduling algorithm in different states of the network.

Time is divided into timeslots, which are grouped into slotframes. There are two main types of slotframes in our algorithm: unicast slotframes and non-unicast slotframes. Each type of slotframes is capable of dealing with specific traffics in the network. The time synchronization of the network is managed by TSCH in default by adding timing information in all data as well as ACK packets.

An arbitrary node will undergo three stages: bootstrap stage, learning stage and stable stage. In the bootstrap stage, the node will obtain network information necessary for TSCH and routing protocol (RPL). In learning stage, a learning mechanism is executed in each unicast slotframe to obtain a schedule for the node, whereas the minimal scheduling algorithm is performed in all non-unicast slotframes. The stable stage starts right after the learning stage, in which a sensor node has already obtained a stable schedule for unicast slotframes. All nodes will follow their schedules in all unicast slotframes and follow minimal scheduling algorithm in all non-unicast slotframes.

Timeslot management: As mentioned in the previous subsection, there are two types of slotframes in our system: unicast and non-unicast. Unicast slotframes are devoted to traffics transmitted by unicast pattern, i.e. messages sent from one sensor to only one network destination. In contrast, we use non-unicast slotframes to transmit other kinds of traffics, which contain vital data for network maintenance. For example, Enhanced Beacons and DIO, DIS messages of RPL are transferred in these non-unicast slotframes.

We use a simple method to manage these slotframes. Slotframes are grouped into batches such that a batch will contain a fixed number of non-unicast slotframes followed by a fixed number of unicast slotframes. In this configuration, all sensor nodes will switch to a non-unicast slotframes at a defined time, so it is possible for any important data for network maintenance to reach all sensor nodes. In addition, it is easy for network administrator to pre-configure the proportion of unicast slotframes over non-unicast slotframes in a batch to adapt to the requirement of the network traffic. For example, if the application traffic is in demand more than the traffic for network maintenance, the proportion can be set to a large value, such as 3:1 or 5:2.

Schedule in our protocol: A schedule is a combination of actions will be performed in all timeslots in a slotframe. Because of the existence of two types of slotframes in our network, two types of schedules is constructed respectively, which are non-unicast schedule and unicast schedule. While unicast schedule is constructed by our learning mechanism and performed in unicast slotframes, non-unicast schedule are obtained from minimal scheduling algorithm [55] and applied to non-unicast slotframes.

From the perspective of unicast schedule, the action set of a node includes transmit, listen or sleep to a parent or a child node of the given node. Because Routing Protocol for Low-Power and Lossy Networks (RPL) and IP are built on top of protocol, we use the set of parents and children obtained from RPL as the set of destinations for the action of transmitting in our protocol. In the data collection scenario where the traffics are uplink, we often see the action of transmitting data from a node to its parents. In contrast, controlling scenario, in which the traffics are downlink, sees the application of packet transmission from a node to its children. The listening action is dedicated to receiving traffic from transmitter, whereas the action of sleep refers to turning off the radio frequency for power saving.

In detail, assuming we have a node with identifier n. In a timeslot number k, node n may have one of the following actions:

- Transmit data to a node whose identifier is x in channel ch(x, k);
- Listen on its own channel ch(n, k) for incoming data;
- Sleep: turn off its radio for saving power.

The ch(n, k) is the channel hopping mechanism aiming at determining the channel in which the coordination between two nodes may happen. It is possible to use the default channel hopping formula of TSCH, however we use a simpler version of it as (a*n + b*k) % NUM_CHANNELS where a, b are constants and NUM_CHANNELS is the number of channels in use. The parameters a and b are often chosen to co-primes or in other ways such that the distribution of ch(n, k) for arbitrary n and k is uniform, which leads to the utilization of all available channels.

Scheduling algorithm: There are two types of scheduling algorithm in our design, one of them is the minimal scheduling algorithm dedicated to non-unicast slotframes, whereas the other is the scheduling algorithm based on Reinforcement Learning technique devoted to unicast slotframes. There are two main reasons leading to the co-existence of two scheduling algorithms. Firstly, it is because of the nature of the design, in which all nodes behave the same as others in non-unicast slotframes to obtain network maintenance data whereas in unicast slotframes the traffics are point-to-point with specific sender and receiver. Secondly, the co-existence aims at keeping the consistency of timeslot and slotframe structure of each sensor. For example, if we replace non-unicast slotframes by other method for multichannel access control (MAC), it could be difficult for keeping time synchronization correctly, which is an important part of TSCH feature.

The RL-based algorithm presented in this subsection is a modified version of the algorithm proposed in [57]. The algorithm in [57] fits to our design for two main reasons. The first reason is that the algorithm aims at building a communication schedule for each node by considering it as an autonomous agent, acting independently, obtaining feedback from environment and learning from it. It coincides to our target of developing a communication solution for any sensor to transmit and receive data with respect to environment adaption. The second reason is the definition of the action set, which is similar to the action set in our unicast slotframes.

However, these similarities are not enough for the algorithm to fit perfectly well to our model. First of all, in [57], the authors proposed a joint-model of routing and scheduling, in which the action set of a node is associated to its neighbors. In order to build IoT protocols on top of MAC protocol, we define the target of the action set of a node as its parents and children in routing table obtained from RPL protocol. Second of all, we do not use Clear Channel Assessment (CCA) as a check scheme for a node before transmitting. Finally, we define a formula to estimate the successful probability of an action in a timeslot. In case of transmitting data, successful probability is defined as the number of packet transmitted successfully over the number of packets transmitted in a timeslot. In case of listening for incoming data, successful probability is defined as the number of timeslots in which there are any incoming data over the number of timeslots in which a node turns to listening action.

Input, output and parameters		
Input	BOOTSTRAP_END	The end time of bootstrap period, unit of measurement: sec.
	LEARNING_END	The end time of learning period, unit of measurement: frame
	SPF	The number of timeslots per frame
	UC_FRAME	Unicast frame
	MINIMAL	Minimal scheduling algorithm
	LEARNING_ALG	Learning algorithm
	SLEEP	Setting the radio off
	SLEEP_THRES	Threshold to set radio off
	ch(n, k)	Channel hopping formula, determining the channel in which node with identifier n will listen to in timeslot number k
	P_n	Parent addresses of node with identifier n
	A_{nk}	Available actions of node with identifier n in timeslot number k
	$P_{suc}(a)$	Successful probability of action over A_{nk}
	a_0	Action of listening on its channel ch(n, k)
	a_x	Transmit data to node with identifier x on channel ch(x, k)
	current_frame	Current slotframe being examined
	current_uframe	Current unicast slotframe being examined
Output	$S_n = \{s_{nk}\}$ $(0 \leq k \leq SPF)$	Action schedule of node with identifier n in each unicast slotframe

Figure 17 Notations for input, output, and parameters

Scheduling Algorithm

```
BOOTSTRAP STAGE
    while current_time < BOOTSTRAP_END
        do MINIMAL
    endwhile
    if current_frame is UC_FRAME
        $s_{nk}$ <= randomly chosen from $A_{nk}$.
    else
        do MINIMAL
    endif
LEARNING STAGE
```

```
        if current_frame is UC_FRAME
            if current_uframe < LEARNING_END
                for k = 0 to SPF – 1
              do              LEARNING_ALG
              update      packet queue
              update      probability of success $P_{suc}(a)$
                endfor
            endif
            increase current_frame by 1
        else
            do MINIMAL
            increase current_frame by 1
        endif
        /* Selection step */
        for k = 0 to SPF – 1
            select a* with highest probability of success $P_{suc}(a^*)$ in $A_{nk}$
            if $P_{suc}(a^*)$ > SLEEP_THRES then
                $s_{nk}$ <- a*
            else
                $s_{nk}$ <- SLEEP
            endif
        endfor
    STABLE STAGE
        if current_frame is UC_FRAME
            do the action obtained from schedule $S_n$
            increase current_uframe by 1
        else
            do MINIMAL
            increase current_uframe by 1
        endif
```

Figure 18 Scheduling algorithm

The Figure 18 (whose parameters are explained in Figure 17) represents our scheduling algorithm. In the bootstrap phase, in order to create the network topology, each node starts joining the network by applying minimal scheduling algorithm until a determined time. In the first unicast slotframe after the establishment of network topology, each node creates an initial schedule by chance. The learning phase will start subsequently. In this phase, the learning algorithm will be performed in each timeslot associated with a unicast slotframe. Eventually, when learning progress is completed, each node executes a simple selection procedure, which results in a final schedule. This schedule will be followed by each node in all unicast frames afterwards. The stable stage in the figure illustrates the operation of the node in this case. Different nodes have different schedules for unicast slotframes, based on the results of the learning period and the selection step.

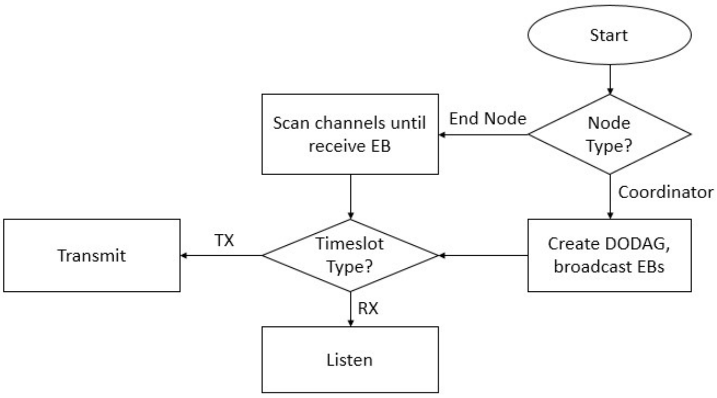


Figure 19 A closer look at the bootstrap stage

A node will check whether it is a coordinator node (root node in RPL). If it is a coordinator node, it will broadcast packets for DODAG and TSCH configuration creation. Otherwise, it is an end node and subsequently scans available channels for DODAG and TSCH configuration packets. After receiving any EB, it has the necessary information to join the network and starts behaving according to minimal algorithm until the end of the bootstrap stage.

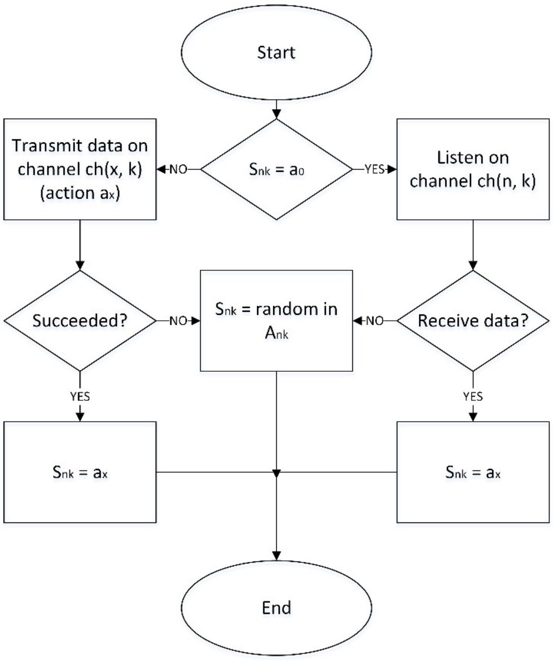


Figure 20 Learning mechanism

In a timeslot, a node (whose identifier is n) will check the action is listening on its channel (ch(n, k)). It will listen on its channel for incoming traffics if the answer is yes and transmit data if the answer is a different action. Base on the feedback of the action, it decides a new action for the next schedule. Note that the action of transmitting is successful if the node receives ACK for its packets.

Proposed Algorithms for Data Collection from Relay Nodes

Here, we formulate the problem of data collection from relay nodes and afterwards, we propose two algorithms to resolve the problem.

Problem formulation: Our data collection model consists of fixed relay nodes in a rectangular field with given width and height. Two types of mobile agents are given in the area, including:

- Controlled Agents (CA): the mobile gateways which can be controlled to serve data collection requests.
- Uncontrolled Agents (UCA): the mobile gateways which have fixed trajectories and move under predicted schedule.

Both of agent types are set up with a gateway to collect data from relay nodes in given rectangular field. When a mobile gateway (MG) visits a relay node, the relay node starts to transmit its data to the MG. The data transmission is done over the TCP/IP protocol, built on top of 802.11 standard. Because the speed at which data transmission speed taken place is usually significant larger than the velocity of the mobile agent, we eliminate the effect of the delay time among transmission time. All relay nodes are assumed to have a communication range, in which they are capable of transmitting the data, of r_c and the mobile agents move with a maximum velocity of v_{max}.

The optimization problem is defined as finding path for each of controlled agents, which at the same time minimizes its moving distance and minimizes the total end-to-end packet delay at each relay node. In practice, the optimization objectives are partly contradicted. For example, when there are many packets waiting for being collected at a relay node far from the current position of a mobile agent while the mobile agent is near another relay node, it must choose whether to collect data at the far relay node or the near one.

We formulate the model as follow:
Input:

- W x L: The rectangular area for data collection with width of W and length of L.
- |UncontrolledAgents| = p
- |ControlledAgents| = q
- The topology of road system

Output:

- Path planning of each MA in the CA set

Objectives:

- Minimize the total end-to-end packet delay at each relay node.
- Minimize the length of the path for each MA.

Proposed algorithm: Here, we propose two algorithms to deal with the scheduling problem. The former is a modified version of MWSF which have been reinforced by K-Means clustering algorithm (K-MWSF). The latter utilizes Simulated Annealing algorithm and K-means (K-SA) to find the optimal solution.

K-MWSF algorithm - In the heuristic algorithm, we modify the minimum weighted sum first (MWSF) at [19] for one mobile agent then we use a combination of K-Means and MWSF to deal with the problem of multiple mobile agents. K-Means is applied to divide all relay nodes into clusters. Each cluster of relay nodes will be visited by one controlled agents according to a modified MWSF algorithm.

Minimum Weighted Sum First (MWSF) is introduced in [58] to optimize a schedule for a mobile agent. The algorithm considers the first relay node to visit based on two components including deadline time and motion cost.

We modify MWSF to adapt to our model. The algorithm aims at helping one controlled agent to plan a schedule to collect data from the sensors. In our algorithm (which is called MMWSF), we are interested in two factors. The first factor is the total delay of all packets in a relay node, which is defined as the different between current time and the time at which the packet is available. The relay node which has the minimal factor should be visit as soon as possible, which is equivalent to the objective of minimizing the total end-to-end delay of the network. The second factor is the length of path from current position of mobile agent to the position in communication range of a relay node which the mobile agent can move in. When the length of a path is small, the car can move quickly.

The algorithm can take into account the connectivity at a position as a third factor. For example, a position has a better connectivity than another position if it is possible to simultaneously collect data from more relay nodes in this position than the other. This optional factor can be useful if the sensors are deployed in a dense environment, in which the communication range of a relay node overlaps on the others'. Because our model mostly fits into a sparse environment, in which the overlapping situations hardly happen, we do not take into account this additional factor. Figure 21 illustrates the algorithm, with the consideration of the additional factor. It is clear that the optimal path is d3 because d3 has a smaller length than d1, d4 and this path can help the car collect data from all two sensors while d2 only collect from one.

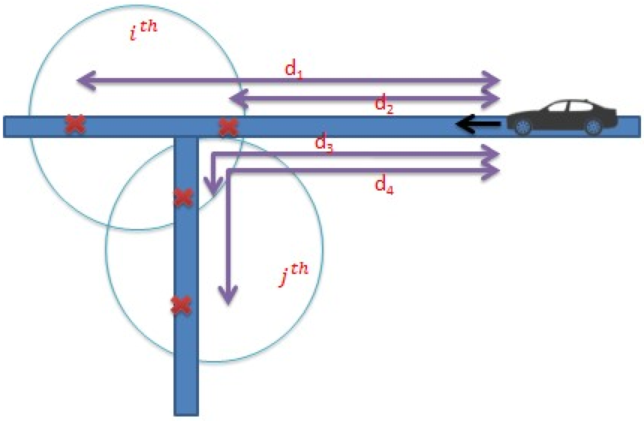


Figure 21 Example for the path planning of the car to collect data from two relay nodes.

Firstly, we define the min_cost function, which aims at finding the most appropriate position in the communication range of a relay node to visit and collect data. The function considers the current position of the controlled agent and the position of relay node it wants to visit to select the best position to visit and collect data. The road system is modeled as an undirected weighted graph and the length of the path is the shortest path obtained from Dijkstra algorithm. Secondly, we also adjust MWSF in each step to solve our model within UCAs. Before considering which node to visit, this algorithm observes whether the node is visited yet. A visited relay node will be skipped until all nodes are visited. After all nodes are visited, the visited states are reset to default to start a new turn of data collection.

```
Function min_cost(ca_pos, i)
Input:
  Position of the controlled agent ca_pos
  Index i of the ith relay node
Output:
  Most appropriate position to visit and collect data.
  The length of the path
Begin
  for position in ith relay's communication range
    if controlled agent can move to position
      w[position] = path_length(ca_pos, position)
    endif
  endfor
  return argmin(w), path_length(ca_pos, position)
End
```

```
Algorithm MMWSF
Input:
  Controlled Agent CA
  Set of Uncontrolled Agents SUCA
  Set of relay node S
  Current time: current_time
  Simulation time: sim_time
Output:
  Path planning for CA
Begin
  while current_time < sim_time
    for node in S
      if visited[node]
        w[node] = INFINITY
      else
        $w[node] = \alpha * sum(current\_time – packet.time\_available) +$
                                  $(1 – \alpha) * min\_cost(ca\_pos, node)$
        /* the sum is taken over all packets which are available in the relay node */
    endfor
  endwhile
  to_be_visited_node = argmin(w)
  update ca_pos
  update visited[node] for node in S
End
```

The alpha parameter in the algorithm illustrates the priority if the mobile agents. If we do not prioritize any of these two factors, alpha is set to a balance value (i.e. 0.4-0.6). If we consider the total packet delay is significantly more important than the distance the mobile agents must move, we set a high value to alpha (i.e. 0.7-0.8). Otherwise, a small value of alpha is preferred (i.e. 0.2-0.3).

In case of multiple controlled mobile agents, we divide the set of relay nodes in to clusters by applying K-Means clustering algorithm. The number of clusters is also the number of controlled agents. Afterwards, we assign each cluster to a controlled agent, so the controlled agent will only collect data in all relays in its cluster. Because we assume relay nodes are deployed in a 2D rectangular field, K-Means algorithm consider Euclidean distance between two relays as the metric to perform clustering. As a result, each cluster contains all relay nodes close to each other, so that the controlled agent does not have to move a long way to collect data.

K-SA algorithm - Simulated Annealing (SA) is local search algorithm which supports to approximate local optimal value from a large search space. Annealing process is simulated accordingly to parameters such as temperature and cooling degree.

In the heuristic algorithm, all clusters represents for concerned locations where mobile agents have to collect data. In many cases, although the locations of relays are close, their total packet delay is so distinct that makes the collecting process face a huge amount of delay. To enhance the grouping of relay nodes, we utilize SA to find local minimal solution from the heuristic solution.

Each group of relays is presented as a tour of the CA whose collect data in this group. A tour includes relay's locations and associated time at which CA visits it. A solution consists of the tours of all CAs.

(s_{i1}, t_{i1})	(s_{i2}, t_{i2})	(s_{i3}, t_{i3})	(s_{i4}, t_{i4})	(s_{i5}, t_{i5})

Figure 22 i^th^ tour of i^th^ CA

From a solution (S_i) we use 1-swap to create a neighbor solution. We select randomly two tours (*i, j*) from all tours of S_i. The chosen tour will exchange two relay nodes, which is also determined randomly, between them. After the exchange, two tours are updated by using MMWSF in previous section.

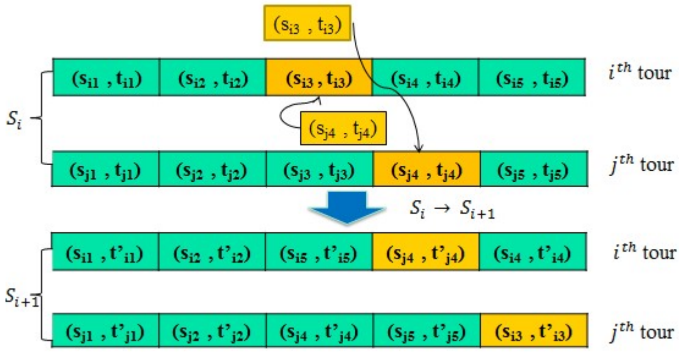

(s_{i3}, t_{i3})
(s_{i1}, t_{i1}) (s_{i2}, t_{i2}) (s_{i3}, t_{i3}) (s_{i4}, t_{i4}) (s_{i5}, t_{i5}) i^{th} tour
S_i
(s_{j4}, t_{j4})
(s_{j1}, t_{j1}) (s_{j2}, t_{j2}) (s_{j3}, t_{j3}) (s_{j4}, t_{j4}) (s_{j5}, t_{j5}) j^{th} tour
$S_i \rightarrow S_{i+1}$
(s_{i1}, t'_{i1}) (s_{i2}, t'_{i2}) (s_{i5}, t'_{i5}) (s_{j4}, t'_{j4}) (s_{i4}, t'_{i4}) i^{th} tour
S_{i+1}
(s_{j1}, t'_{j1}) (s_{j2}, t'_{j2}) (s_{j4}, t'_{j4}) (s_{j5}, t'_{j5}) (s_{i3}, t'_{i3}) j^{th} tour

Figure 23 1-swap for neighbor solution creation

```
Algorithm K-SA
Input:
   Initial clustering result obtained from K-Means $S_0$
   SA's parameters: Temperature T, Cooling parameter $\mu$, Terminated condition £
   Number of iterations: N
Output:
   Optimal clustering result $S_{op}$
Begin
   for t = T to £
      for i = 1 to N
         $S_i' = get\_neighbor(S_i)$
         if fitness($S_i'$) < fitness($S_i$) or uniform(0, 1) < exp(-(fitness($S_i'$) - fitness($S_i$))) / $\sqrt{T}$
            $S_i = S_i'$
            $t = t * \mu$
         endif
      endfor
   endfor
End
```

Figure 24 KSA algorithm

Fitness function is calculated based on the total delay of packets and the total moving distance of all controlled agents weighted by β and (1 – β), respectively. The parameter β illustrates the priority between total delay and total distance to accept a solution.

$$fitness(S'_i) = \beta * total\_delay + (1-\beta) * total\_distance$$

Performance Evaluation

Here, we present the simulation results of the proposed mobile IoT model. We first evaluate the performance of our data transmission within the network of sensors. Second, we show the results the data collection part. In general, we deploy a 1000x1000 m2 simulation field, in which 30 groups of sensors are deployed. A group of sensors consists of from 8 to 10 sensors, whose responsibilities are to collect data in an area. In an anonymous group of sensors, there is a root node called Relay Node to which all collected data is sent to it. The communication range of two arbitrary groups of sensors is mostly not intersected, which is equivalent to a large scale area. As mentioned in the section 1, we aim at solving two problems:

- Data Transmission within a group of sensors.
- Data collection from the network formed by relay nodes.

Our simulation and performance evaluation method is built for the purpose of solving these two problems.

Performance evaluation of data transmission method in a group of sensors: We focus on evaluating the RL-based solution for data transmission. First, we implement the algorithm and integrate it into IoT protocol stack, which includes 802.15.4e in MAC layer, 6LoWPAN in adaption, IPv6 for Internet layer, RPL for routing and UDP for transport layer. The implementation is done in Contiki 3.0, and the code is flashed into Zolertia Z1 device. Second, we deploy a test field with an area of 120 * 110 m2 in which there are up to 10 randomly

distributed sensor nodes. An instance of test field can be seen on Figure 25.
The selection of the test field and the number of sensors are because of 2 main reasons.

- Because the RL-based scheduling algorithm is part of our mobile IoT, the selection of simulation must describe the situation to apply the algorithm. The choosing of test field for a group of sensors must satisfy the large scale condition.
- In this simulation, the hop count in the routing topology formed by RPL is mostly 1, 2 or 3, which illustrate the real-world situation.

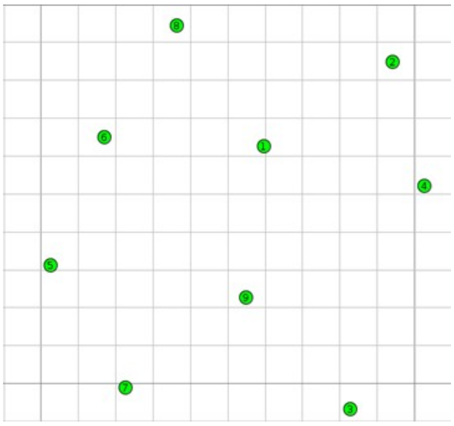


Figure 25 An instance of test field for sensor-group simulation

Input parameters	
BOOTSTRAP_END	4904 (s)
LEARNING_END	150 (frames)
SLEEP_THRES	0.05

Figure 26 Input parameters to RL algorithm

In the last step, we run numerous simulations on provided simulation tool Cooja in Contiki 3.0 and compare the average results obtained to those from minimal scheduling algorithm. The comparison is made between minimal of scheduling algorithm and two versions of our solution, one called RLM 2/1 with 2 unicast slotframes/1 non-unicast slotframe in a batch and the other called RLM 3/1 with 3 unicast slotframes/1 non-unicast slotframe. In this sensor-group simulation, we measure several parameters for performance evaluation as follows:

- The end-to-end packet delivery rate is the ratio between the number of data packets received at the relay node and the number of packets generated at sources nodes. It assesses the capability of the scheduling algorithm to deliver all generated messages to the relay node for further purpose of data collection.
- The average single-hop packet delivery latency is measured at the average ratio of the end-to-end packet delivery delay over the number of intermediate nodes through which a packet travels. The delay is measured at MAC layer.
- The average energy consumption of a node, which is a measurement of the lifetime of a sensor. The larger the energy consumption, the shorter the life of a sensor. We evaluate this parameter indirectly by examine the amount of time that the sensor turned on its radio device for transmission. Because turning on a sensor's radio device is the main source of power consumption, these two measurements are equivalent.

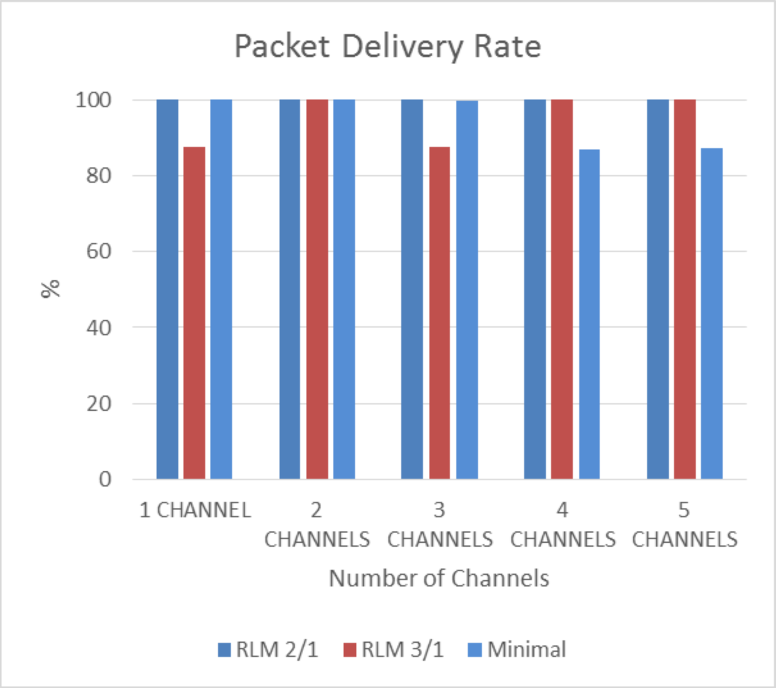


Figure 27 Packet delivery rate

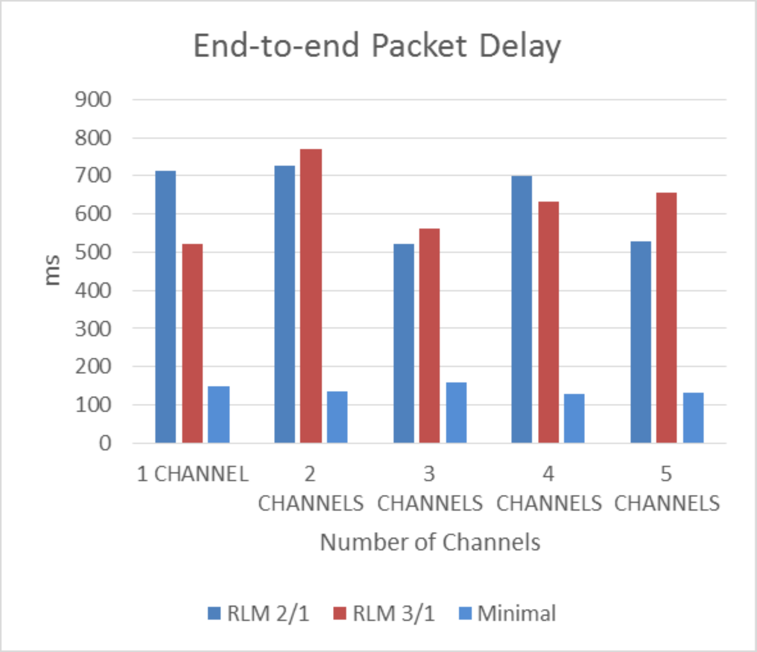


Figure 28 End-to-end packet delay

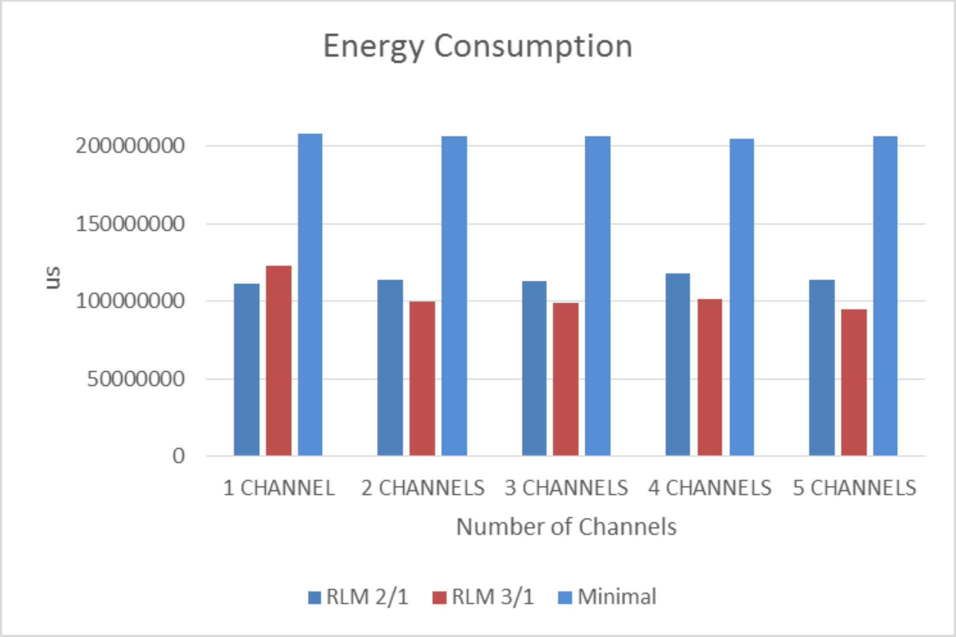


Figure 29 Energy consumption

The results are shown in Figure 27, Figure 28, and Figure 29, in which the x-axis displays the number of channels in use by TSCH. According to the Figure 27, in general, all methods have high packet delivery rate at 100 percent except some cases where a node does not receive any information about the RPL topology. The missing of interference in simulation helps minimal scheduling achieve 100 percentage of packet delivery rate in all cases. However, due to its nature of being prone to interference, the algorithm might not keep the same result if it is deployed in real circumstances. The missing of interference in simulation may be the reason why minimal algorithm maintains a low latency. As can be seen from Figure 28, RLM 2/1 generally achieve a low latency than RLM 3/1 and the latency of the protocol for the cases of 3, 4 and 5 channels are smaller than those of 1 or 2 channels. We may conclude that the more the number of channels are in use, the more the chance to achieve a low latency.

Figure 29 illustrates the superiority of our solution in term of energy consumption. The average energy consumption of the method is half of it in minimal scheduling method, which means that it can prolong the lifetime of a sensor by a factor of two. In addition, RLM 3/1 saves more energy than RLM 2/1 in almost all cases, because there are more unicast slotframes in RLM 3/1 than in RLM 2/1, which is equivalent to a better management of wake-up/sleep actions. This observation indicates the importance of choosing number of unicast slotframes/non-unicast slotframes in a batch.

Performance evaluation of data collection in the network of relay nodes: We evaluate the performance of the data collection algorithm in the network formed by relay nodes. The first step is to get the test dataset. In a 1000x1000 area, we deploy 30 groups of sensors; each of them contains from 8 to 10 sensors and is characterized by a relay node. The relay node has all the data packets collected by the group of sensors within a given simulation time (20 minutes). In order to obtain the information about data packets in a relay node, we run the data transmission model for each group of sensors. In other words, we run 30 different group-sensor simulations associated with 30 different configurations to obtain the information about data packets for 30 relay nodes.

In the second step, we create a simulation program in Python to illustrate the behaviors of algorithms for data collection: MMWSF (MWSF algorithm for one controlled agent), KMWSF (K-Means with MWSF) and K-SA (KMWSF with Simulated Annealing). Given a predefined path system for controlled agents, the positions of all relay nodes are randomly chosen such as (1) communication range of relay nodes mostly do not overlap and (2) it is feasible to collect data from them (which mean there are some intersections between the communication range and the path system). The path system is modeled as a undirected weighted graph where a vertex is a point in the simulation field and the weight of an edge connecting 2 vertices is the Euclidean distance between them. The controlled agents are positioned at random positions in the path system with speed selected randomly from 30 to 40 km/h. The effects of uncontrolled agents is modeled by the action of randomly visit a relay node, which means at a given time, we randomly select some node and collect all data packets have not been collected yet up to that time. To minimize the effects of uncontrolled agents, we rarely perform the above action such that it can be considered as not happened.

In the last step, we examine the performance of 3 algorithms in terms of three factors:

- The average length of the path that each controlled agents has moved. This is one of the optimization factors we considered.
- The average end-to-end packet delay. This is the second optimization factor we considered. This factor estimates the total delay from the time a data packet is generated to the time it is collected. This parameter comprises of three parts: (1) the delay from the time a data packet is generated at a sensor to the time it arrives in relay node, (2) the amount of time it takes a Java program in relay node to parse the content of the UDP datagram and pack the content to a TCP datagram (this delay is because of a relay node received UDP packets from sensors and transmit TCP datagram to a mobile agent, we simulate this behavior through a Java program) and (3) the amount of time it takes a mobile agent to collect data available in the relay node.
- The percentage of data packets collected during the simulation. This parameter is defined as of all packets a relay node received during the simulation, how many of them are collected by mobile agents.

Experimental	Parameter
Domain *A*	1000x1000
Simulation Time	12000 s
Communication Range	4 m
α	0.5
β	0.5
σ	0
T	100
μ	0.9
£	0.5

Figure 30 Experimental parameters

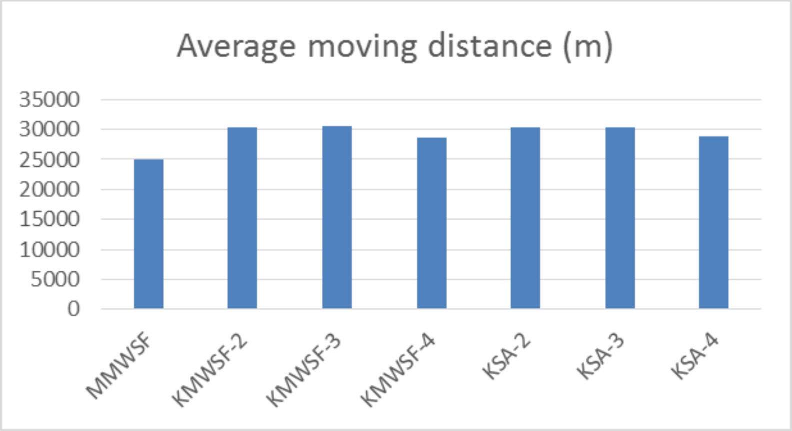


Figure 31 Average moving distance in meters

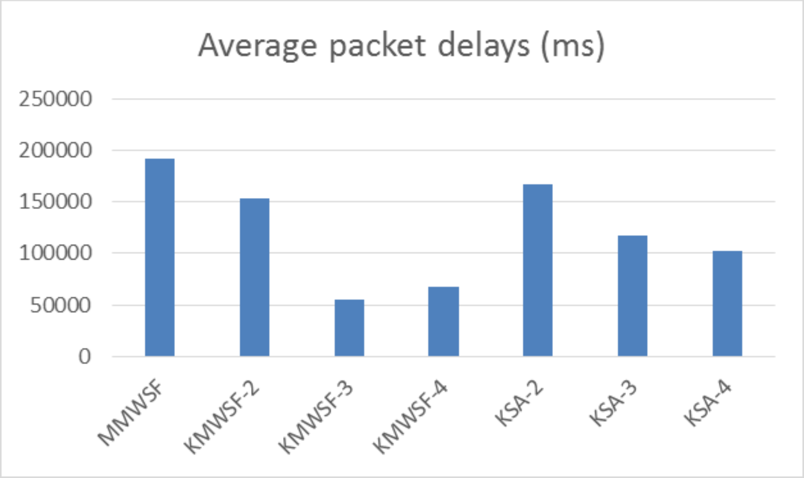


Figure 32 Average packet delay

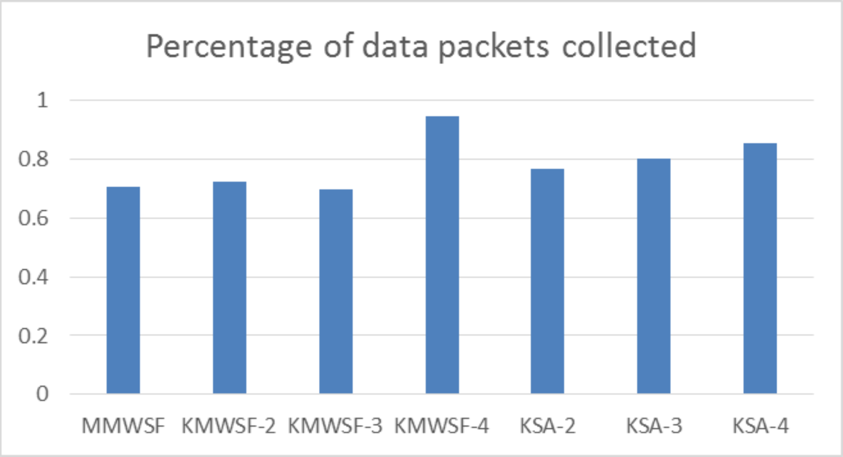


Figure 33 Percentage of data packets collected

MMWSF	KMWSF-2	KMWSF-3	KMWSF-4	KSA-2	KSA-3	KSA-4
2.11	2.16	2.08	2.82	2.29	2.40	2.56

Figure 34 Balance score

The results are shown in Figure 31, Figure 32, and Figure 33. The comparison is made between Minimal (Original) MWSF algorithm (MMWSF), KMWSF algorithm with 2, 3 and 4 controlled agents (KMWSF-2, KMWSF-3 and KMWSF-4) and KSA with 2, 3 and 4 controlled agents (KSA-2, KSA-3 and KSA-3). As can be seen from the Figure 31, the average distance covered by one CA is approximately the same for MMWSF, KMWSF and KSA algorithm. However, the average packet delay of both KMWSF and KSA algorithms are significant better than that of MMWSF. Using more controlled agents in both KMWSF and KSA algorithm leads to lower value of average packet delay. In term of percentage of data packets collected, KSA generally performs better than MMWSF and KMWSF. In addition, a slight increase of percentage of data packets collected can be seen when the number of controlled agents increased from 2 to 4.

In order to estimate which algorithm is better, we define balance score of each algorithm as the harmony average of three factors, including average moving distance, average packet delay and percentage of packets collected. The intuition behind the balance score is that an algorithm with high values in these three factors will achieve a high balance score. According to Figure 34, KMWSF-4 achieves the highest balance score; however the most balance algorithm is KSA because it achieves high scores in all three cases: with 2, 3 or 4 controlled agents.

From above observations, we may conclude that KMWSF and KSA perform better than MMWSF in terms of the percentage of packets collected and average packet delay and KSA achieve a better balance between these two factors.

Conclusions

In this work, we propose a large-scale mobile IoT system which aims at data collection problem. In this model, sensors are deployed into groups, in each group, a sensor with strong processing capability is chosen as relay node. The responsibilities of a relay node are to receive data from other nodes and to transmit data to mobile agents. Mobile agents are robots equipped gateways which are capable of moving inside the data collection area and have to go to relay nodes to get data from them. We proposed solutions for two problems in this large-scale IoT model:

- The first one is data transmission is each group of sensors. We utilize TSCH feature of 802.15.4e standard, build IoT protocols on top of it and modify a RL-based algorithm for transmission scheduling part of the feature.
- The second one is path planing for controlled mobile agents to visit sensor nodes so that minimizes both total moving distance and total end-to-end packet delay. In the large-scale model, the communication ranges of two groups of sensors hardly overlap, and prioritizing moving distance or total end-to-end packet delay is considered as trade-off of the system.

We implemented the large-scale IoT model in simulation environment. We implement the data transmission in each group of sensors in Contiki 3.0 and realize numerous simulations on Cooja to evaluate the results. Our method achieved smaller power consumption than minimal scheduling algorithm and an acceptable latency, high packet delivery rate with consideration of the missing of interference in Cooja simulator. For the relay node-mobile agent part, we simulate the behaviors of the system through a Java program (for parsing UDP datagram and creating TCP connection with the mobile agents) and a Python program (for estimating the performance of MWSF, K-MWSF and K-SA algorithms). The results of K-MWSF and K-SA algorithms are promising compared to MWSF algorithm.

Simple Mobile IoT System Testbed with Robotic Base as Mobile Gateway

A simple testbed of Mobile IoT system has been developed, comprising of a single mobile gateway and two sensor nodes. The testbed comprises of three elements as follows:

- Sensor node
  Raspberry Pi equipped with temperature and humidity sensor and WiFi communications module.
- Mobile gateway
  Iclebo Kobuki robotic base connected to a Linux-based laptop running ROS, with WiFi dongle operating in access point mode via hostapd, and hokuyo short-range lidar.
- Back-end database and Visualization
  PostgreSQL database and Python-based web server interacting with the database to query and generate visualization graph according to the user request.

Hardware Architecture

The sensor node device used in this testbed is shown in Figure 35.

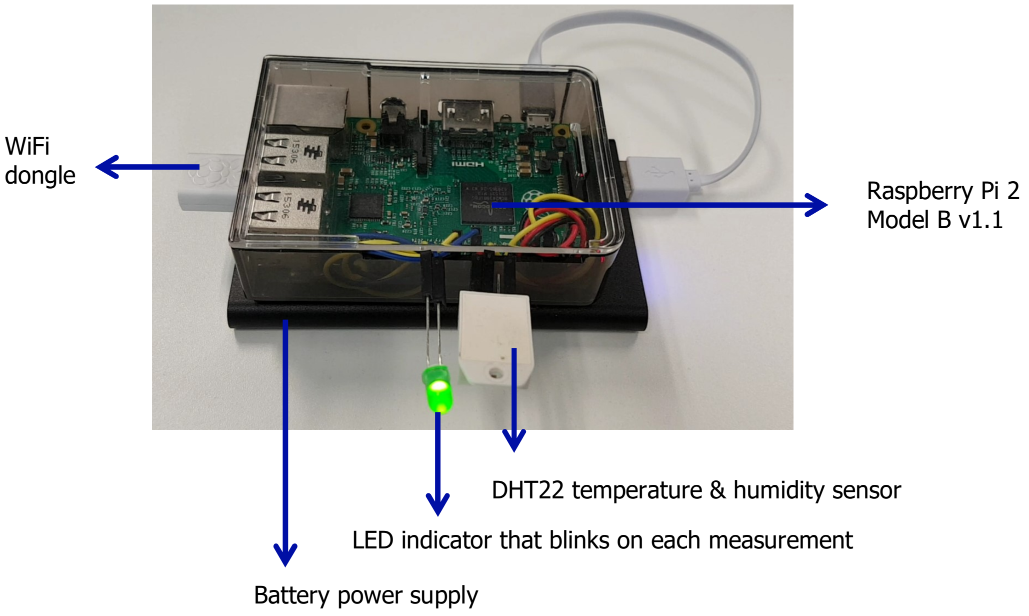

WiFi dongle
Raspberry Pi 2 Model B v1.1
DHT22 temperature & humidity sensor
LED indicator that blinks on each measurement
Battery power supply

Figure 35 Sensor node hardware architecture

The functionality of this sensor node is to periodically obtain the temperature and humidity measurement, and store it into its local buffer. At the same time, the sensor node will monitor if a mobile gateway is within its communication radius. Whenever a connection to a mobile gateway is established, the sensor node will send all the measurement records stored in its buffer to the mobile gateway via socket communications.

Figure 36 shows the mobile gateway used in this testbed.

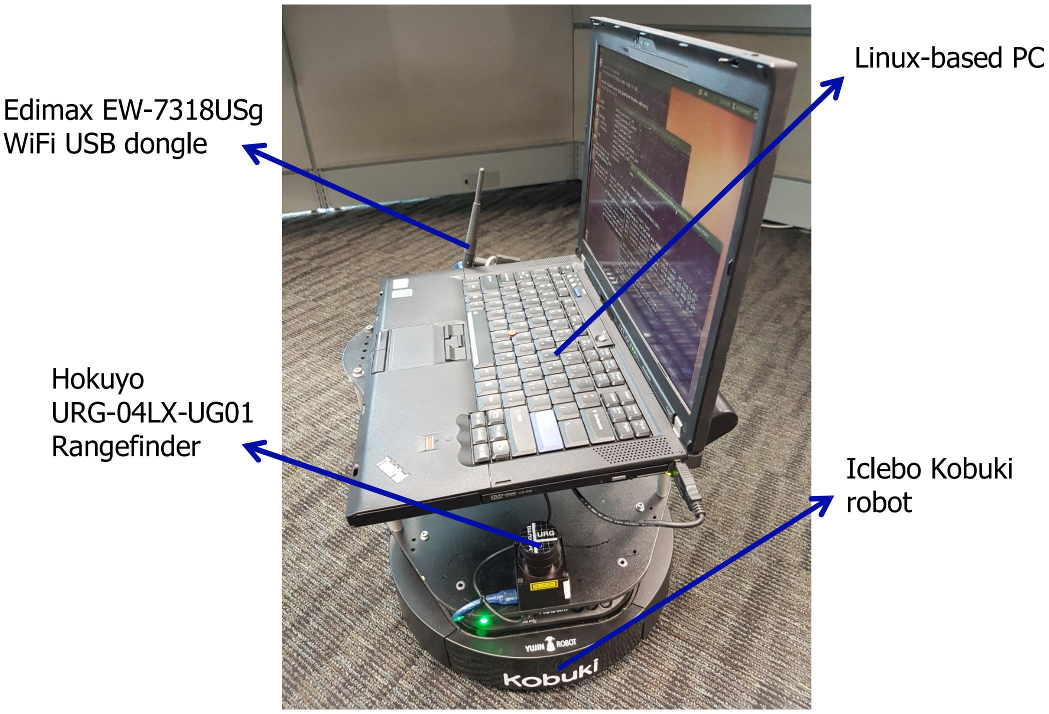


Figure 36 Mobile gateway hardware architecture

The mobile gateway we used in the experiment is in the form of mobile robot, which performs auto navigation implemented in ROS operating system using Hokuyo Rangefinder. In practice, the mobile gateway can also be installed in vehicles such as cars, buses, and trucks. The functionality of this mobile gateway is to roam around a certain area or a pre-determined route, while listening to a possible transmission from sensor nodes. Whenever it enters within a communication radius of sensor node, it receives the measurement data, and stores them into the database. Along with the measurement information, the sensor ID and the time stamp when the measurement is taken are also saved in the database.

Software Architecture

Within the sensor node, there are two Python scripts running, one is responsible to regularly obtain temperature and humidity measurement and store it into a buffer, while another one is responsible to periodically check if there is a mobile gateway detected within its communication radius.

The first script, called “DHT22.py”, interface with the DHT22 temperature and humidity sensor using pigpio to get the measurement readings, and store them into a file buffer. To limit the size of the file buffer, the maximum number of records to be saved within each file buffer is set, and a new file buffer is created to continue saving the measurement records. The files that are sent to the mobile gateway when it is within range are those completed file buffers.

The second script, called “Sender.py”, periodically sense the presence of mobile gateway by establishing socket connection request to it. When the socket connection is successful, the mobile gateway is within range, and the completed buffer files are uploaded to the Linux server that is sitting on the mobile gateway. When the socket connection is unsuccessful, the mobile gateway is outside the communication radius, and the script will wait for 60 second before attempting to establish the socket connection again.

Similarly, on the mobile gateway, there are two Python scripts running. The first one, called “Receiver.py” is used to keep the mobile gateway in a standby mode for socket connection request from the sensor node (particularly from the “Sender.py” script running on the sensor node). Whenever a connection is received and established, the mobile gateway will receive the file buffer from the sensor nodes containing the measurement data, and store it in a specific folder.

The second script, called “ProcessData.py”, is used to process any file buffer received from the sensor nodes, and use the information in the buffer to populate the database. Correspondingly, the script also performs housekeeping to delete all the file buffers that have been processed.

As far as the back-end server is concerned, the database used to store the measurement readings is PostgreSQL, which has one data table for each sensor node, containing the following column information:

- Date/timestamp
- Sensor ID
- Temperature reading
- Humidity reading

To provide visualization to the user, a web application server is created using flask library in Python. Essentially, the web server will interface with the database, and issue SQL queries based on the user request to generate a customized visualization. The result is then presented on the web browser using plotly graph.

Simulation Results

To demonstrate the Mobile IoT prototype, the mobile gateway is setup at Fusionopolis Building, Singapore. The fixed route of the mobile robot is illustrated in Figure 37 (identified with the green arrow). A total of two sensor nodes (indicated by the red dots) are installed and taking the temperature/humidity measurements.

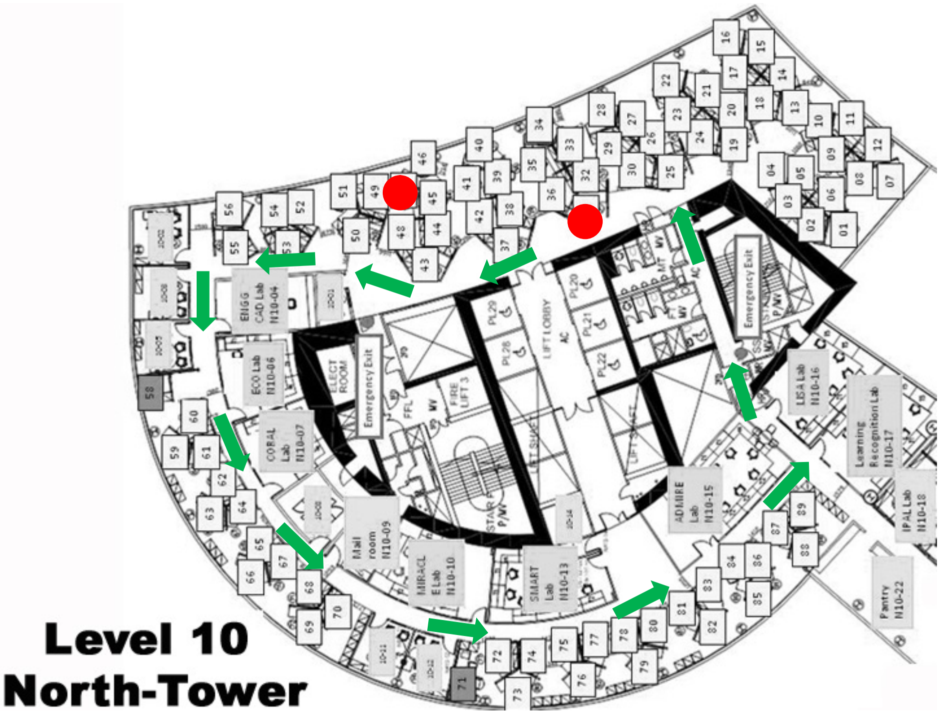


Figure 37 Mobile gateway route

As far as the web application server is concerned, Figure 38 shows an example of the visualization data for one week worth of measurement.

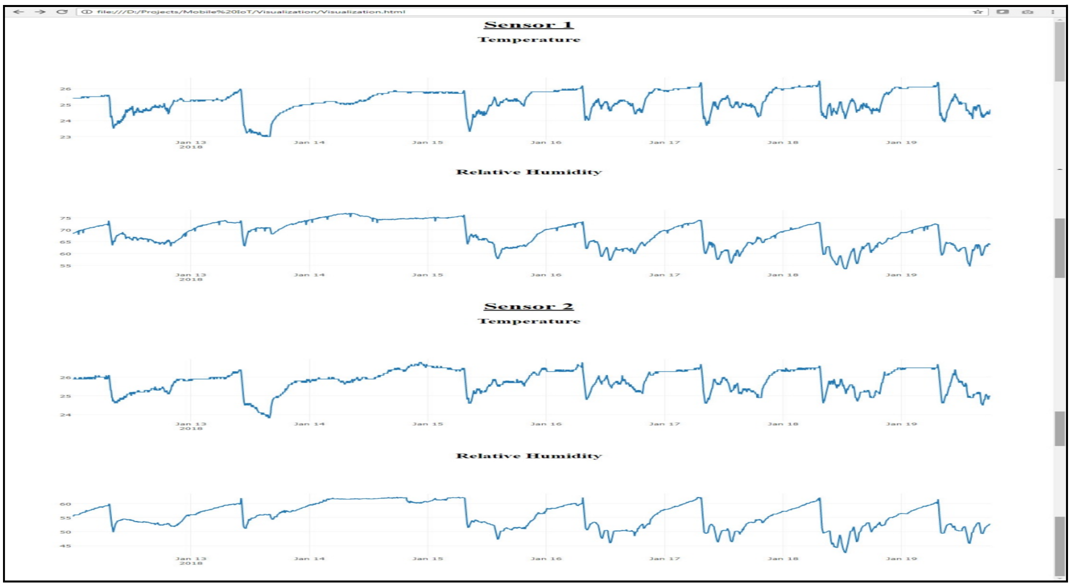


Figure 38 Sample visualization from web application server

Conclusions

We have successfully demonstrated a simple Mobile IoT testbed comprising of mobile gateway and several sensor nodes. The mobile gateway is implemented using a mobile robotic base with auto navigation functionality. Meanwhile, the sensor nodes are implemented using Raspberry Pi device with temperature/humidity sensor device. A demonstration on the Mobile IoT operation is conducted in I^{2}R office building, and the functionality of data collection at the sensor node, data forwarding to the mobile gateway, and the web-based visualization have been successfully shown. This simple testbed can be extended to a larger system with more nodes, and can be used as a starting point to build a full-scale Mobile IoT system.

LORAWAN-based IoT Solution for Environmental Monitoring

Air quality is in the news globally, whether the context is regulatory breaches, poor visibility, traffic congestion or health impacts. Air pollution levels in many cities exceed legal and World Health Organization (WHO) limits for particulate matter and gaseous pollutants which can be found in concentrations that are hazardous to health. Poor air quality is causing a public health problem, since breathing polluted air increases the risk of debilitating and deadly diseases such as lung cancer, stroke, heart disease and chronic bronchitis. Air pollution is now the world's fourth-leading fatal health risk, reported as causing one in ten deaths in 2013.

In particular, transboundary smoke haze from land and forest fires during the traditional dry season from June to October each year has been a perennial problem in the southern ASEAN region in the past decade. These fires are caused mainly by land clearing and 'slash and burn' agricultural practices in Indonesia, particularly Sumatra and Kalimantan. The resultant smoke haze from such activities can be carried over to ASEAN countries and is dependent on factors such as the proximity and extent of the fires, the strength and direction of the prevailing winds, and the incidence and amount of rain.

Transboundary haze pollution affects human health adversely and can also cause significant disruptions to businesses and livelihoods in this region. It is a complex issue, and each country continues to adopt a multi-faceted approach to the problem via regional cooperation and domestic measures.

The Agreement was signed by all 10 ASEAN Member States in June 2002 and came into force in November 2003. The Agreement essentially calls for Parties to undertake, among others:

- Legislative and administrative measures to prevent and control activities related to land and forest fires that may result in transboundary haze pollution; and
- National as well as joint actions to intensify regional and international cooperation to prevent, assess and monitor transboundary haze pollution arising from land and forest fires
- Air pollution & health: key facts
    - Urban air pollution is estimated to cause 1.3 million deaths worldwide per year.
    - Air pollution reduces life expectancy (e.g. 9 months in Europe).
    - Air pollution is a major environmental risk to health.
    - Short and long-term exposure to airborne pollutants enhances mortality and morbidity.

   - There is no evidence of a safe level of exposure or a threshold below which no adverse health effects occur.

Among the pollutants contributing to hazy weather, particulate matter (PM) has attracted much attention. PM can be categorized according to its diameter into PM2.5 micrometer (coarse PM, with a diameter between 2.5 micrometer and 10 micrometer), PM2.5 (fine PM, with a diameter less than 2.5 micrometer), and PM10, the sum of the first two. All of these PM threaten human health, although PM2.5 poses the greatest risk. Particulate matter (PM2.5 and PM10) comprises small, solid particles that often come from traffic and combustion. These particles penetrate airways, lungs and even blood vessels. They are known to be responsible for cardiovascular and respiratory diseases, as well as lung cancers. Nitrogen dioxide (NO2) is a suffocating and irritating gas that comes mainly from combustion. In high concentrations it is known to cause bronchitis, asthma and other respiratory diseases. Ozone (O3) is a gas formed by a chemical reaction between other pollutants. Its concentration is high when there’s a combination of strong sunlight and high concentrations of nitrogen dioxide from combustion. It is known to be responsible for respiratory and heart diseases, asthma and eye irritation.

Air quality is a global challenge for governments, regulators, city administrators and citizens. Many governments are investing multi-billion dollar sums in policies and solutions to improve air quality and they are empowering cities to tackle air pollution locally. In order to implement effective policies and interventions there is an increasing focus on understanding the levels and causes of air pollution. Today, air quality monitoring is performed by large, expensive scientific instruments permanently installed and professionally maintained, at a relatively small number of fixed locations. For example London has around 100 monitoring stations. This makes it difficult for citizens to understand the levels of pollution they experience in their daily lives, as the monitoring data is not available in real time and is very sparse.

Advances in sensors, IoT platforms and mobile communications technologies have led to the emergence of smaller, portable, low cost, mobile-enabled sensors that can measure and report air quality in near real time. “Big Data” capabilities, such as analytics and machine learning, can then be applied to this data and related data sets, such as weather and traffic, to understand the causes and fluctuations in air pollution.

In supporting the above mentioned multipartite agreement, it is a critical to have a tool to monitor the haze at particular location around the region. In this context, the solution will benefit almost the countries within ASEAN.   Thus, it motivates the team project to propose the development of the IoT solution on environmental monitoring with LoRAWAN technology

A complete environmental monitoring with IoT would comprise of the following eco-system elements:

- A cloud/middleware to collect the data and forward to the right channel
- Push notification mechanism that able to response to any changes in the data and notify the end users via different mechanism such as email, short messaging or twitter
- To have platform that able to collect the data from the gateways and further to
- Sensors that able to sense the data real-time with the highest integrity of information (accuracy) to be   forwarded to the cloud server via the gateways
- A analytic mechanism that able to analyze the data and produce a new set of valuable information
- A platform that able to visualize the data in the form of web or mobile application.

With an increase in the number of sensors, there is an increase in spatial density of pollution measurements, which improves both accuracy and reliability of air quality data. With improvement of air pollution data, new applications such as a personal air pollution exposure calculator could be used. Such an application would allow an individual with a global positioning system to track their daily commute and determine an estimation of their air pollution exposure. This application has been developed for Smart Phones which allows a member of the public to take more control over their personal exposures and give them the opportunity to reduce such large exposures. Another application could be real time air pollution monitoring maps, giving a member of the public a quick, easy and informative view of the current air pollution in their environment. Such an application could be accessed over the internet with a simple web browser or dashboard.

In this project, a solution of Low Power Wireless Access (LPWA) with LoRAWAN technology to monitor the PM2.5 and below untul PM1.0 air pollutant index (fine particles that naked to eye) is developed. We have developed the end-to-end solution from the sensor nodes to the user's application. As mentioned earlier, the project chooses the Low Power Wireless Access (LPWA) with LoRaWAN technology at the last mile connectivity to connect with the LoRa sensor nodes.

There are few universal platforms to connect the individual sensor and collect their data to be forwarded to IoT gateway. In this work, we developed the sensor node platform by using the Arduino board. Variety of sensors can be connected to this platform such as haze sensor, temperature (including humidity) and others.

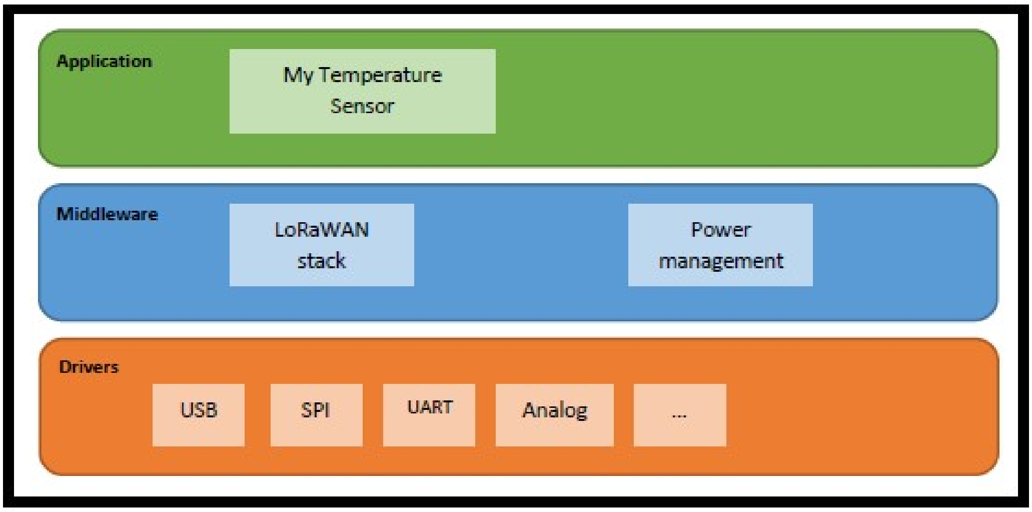


Figure 39 Generic LoRa device software architecture

Schematic Diagram and Data Flow Architecture

Figure 40 and Figure 41 show the schematic diagram and the physical technology components for the mobile IoT air monitoring solution, respectively. In particular in Figure 41, it shows the architecture of the end-to-end technology components for the air monitoring solution. It starts with the data gathering by using the air monitoring sensor (in particular PPD42NJ Particle Sensor). We developed the platform based on Arduino to gather the sensor data and forward it to the LoRa gateway. Here, the LoRa radio is used to communicate with the gateway. Once data received at the gateway, it will forward to the network cloud server, namely as The Things Network (TTN). TTN is one of the community based network cloud based on the LoRa protocol. Meanwhile, it is an option to the user as well to manage the data locally without the requirement to proceed to the network cloud. The data is able to be stored and processed and the local cloud storage. In TTN, the data will be forwarded to the respective application by using the internal MQTT Handler reside in the TTN network cloud. IN this project, we use Node-RED application to process the air data and visualize it by using the dashboard features provided by this application. End user is able to view the real-time time of air quality via this dashboard. Alternatively, data can be forwarded to another third party application to be processed and visualized. In this project, we aim to show the capability of the LoRa technology that enabling the data gathering in the mobile environment. The gateway can be configured and installed in the moving platform (such as truck) to gather the data from the fix sensors that installed at various location.

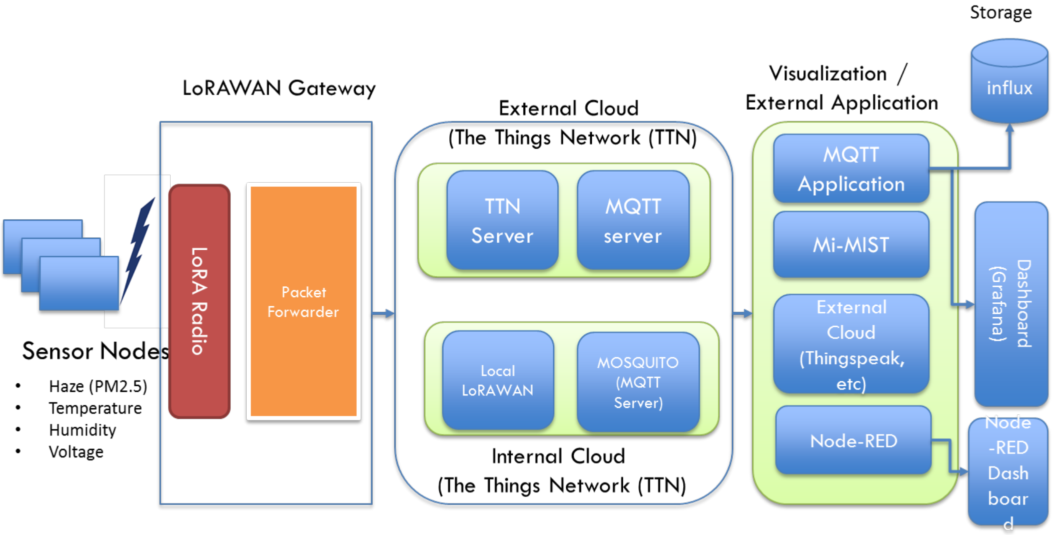


Figure 40 The schematic architecture of the air quality monitoring with LoRaWAN technology

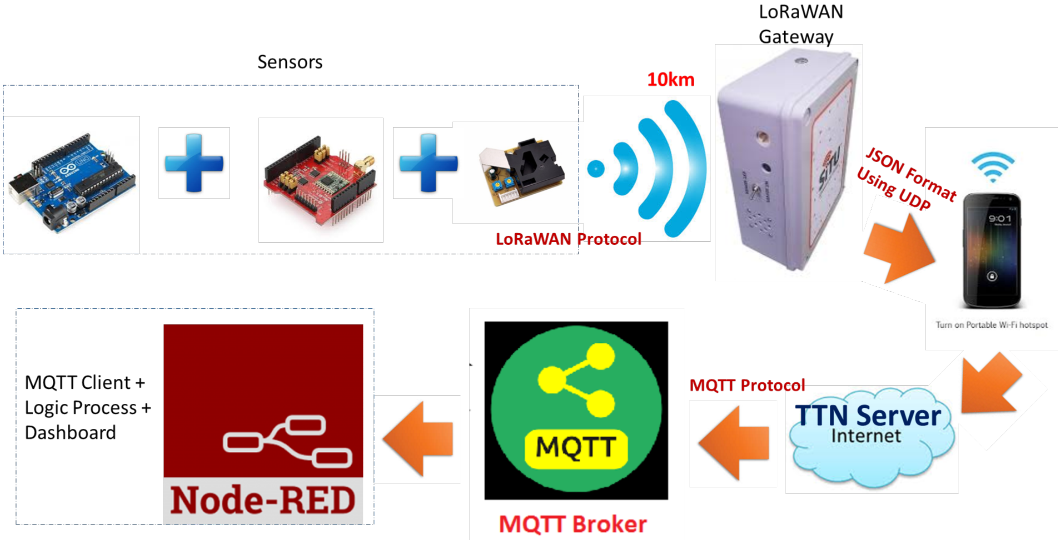


Figure 41 Physical data flow of the environmental monitoring with LoRaWAN technology

Figure 40 shows the schematic diagram. On the other hand, Figure 41 is showing the data flow from the sensors to the gateway, until the visualization at the dashboard. The detail of each technology components will be elaborated in detail in the following section.

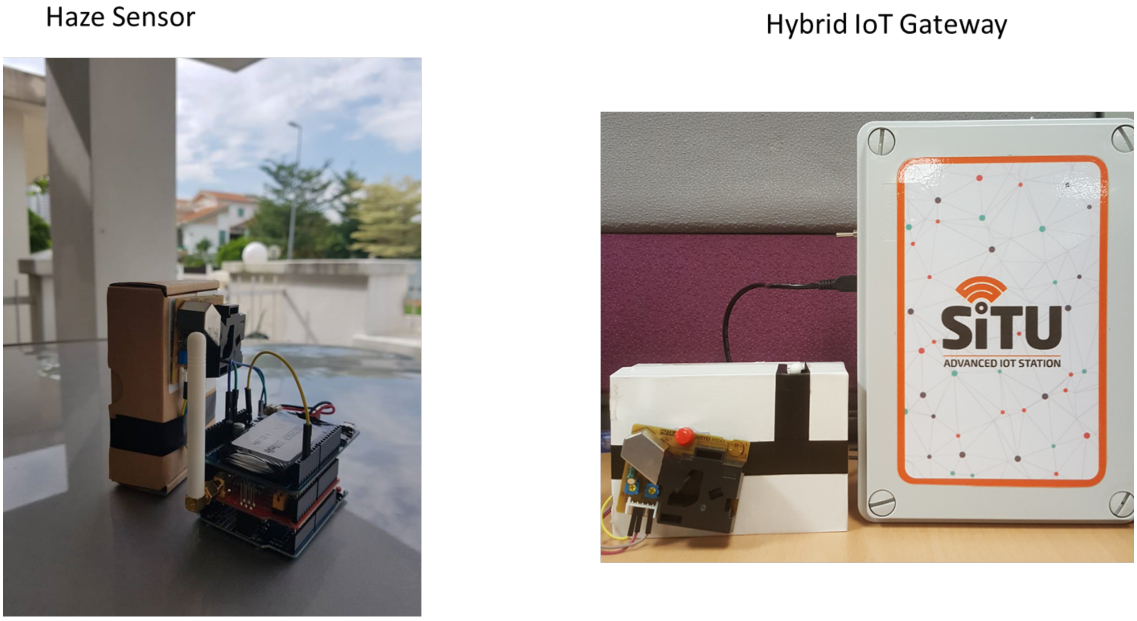


Figure 42 Completed sensor node with the hybrid IoT gateway developed by the project team

Sensor Nodes

In this project, we developed the sensor node by using the Arduino platform of UNO R3 ATmega328P ATMEGA16U2 Board. Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. Here, we attached the LoRa radio module (shield) in order to connect with the LoRaWAN Gateway. This platform is later to be connected with the haze sensor of PPD42NJ Particle Sensor. In addition, we also connect the voltage sensor to the platform. The completed physical view of the sensor node is illustrated in Figure 43 – Figure 44.

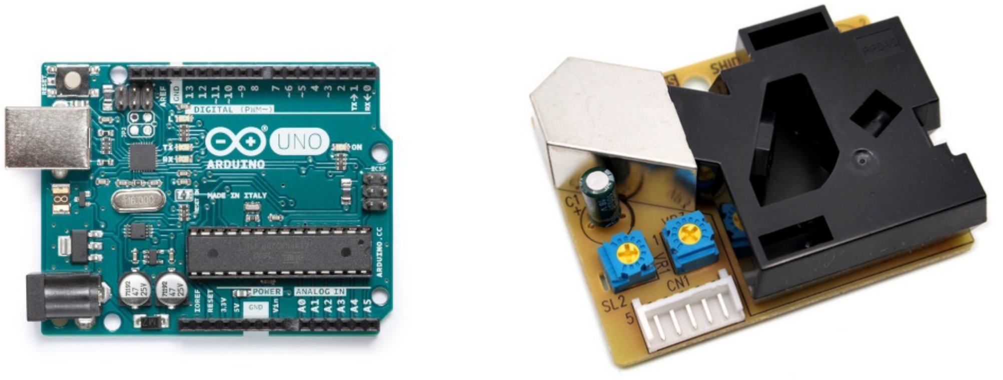


Figure 43 Arduino UNO shield and Dust Haze Sensor (PPD42NJ Particle Sensor)

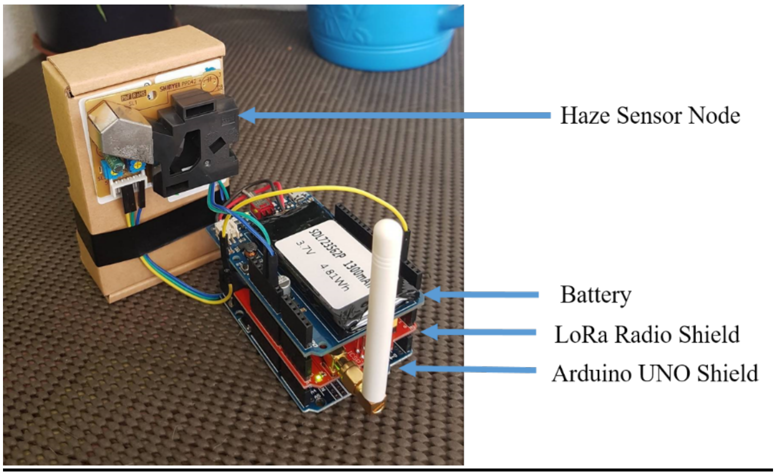


Figure 44 Completed sensor node

Hybrid IoT Gateway

In this project, we have developed the HYBRID LoRaWAN Gateway to receive the LoRa sensor nodes and push the data to the network server (backhaul).   The IoT Gateway based on the LoRaWAN technology is illustrated in Figure 45. This hybrid IoT gateway is able to work in two different modes which are the external LoRAWAN Cloud and internal LoRaWAN cloud (storage). In particular, it is a hybrid mode that allows the end users to extend the data to the external cloud network (via external cloud). On the other hand, the gateway is allowing users to push the data to the local storage (local cloud network).

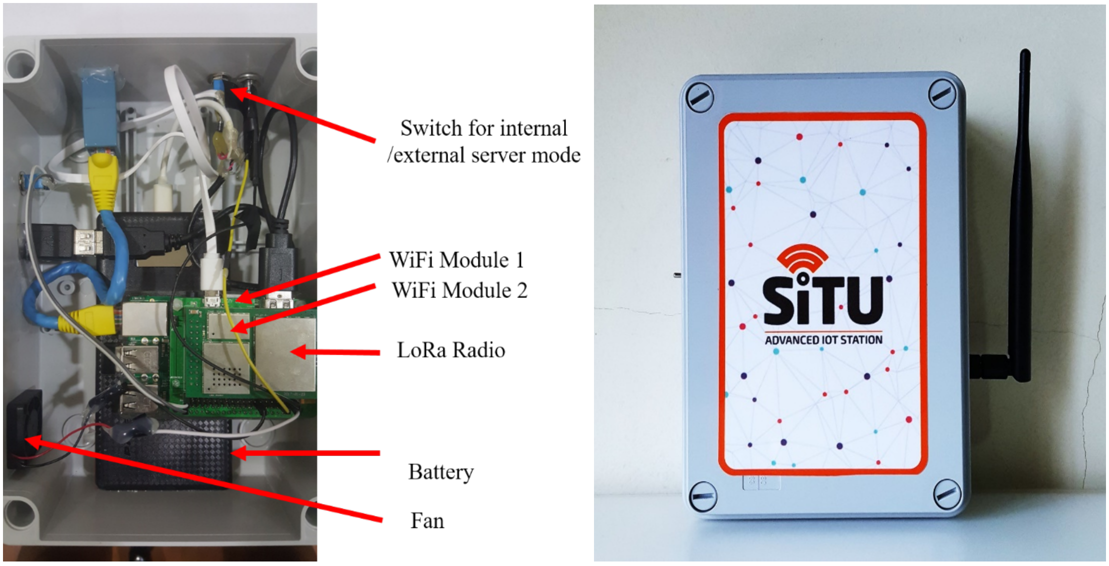
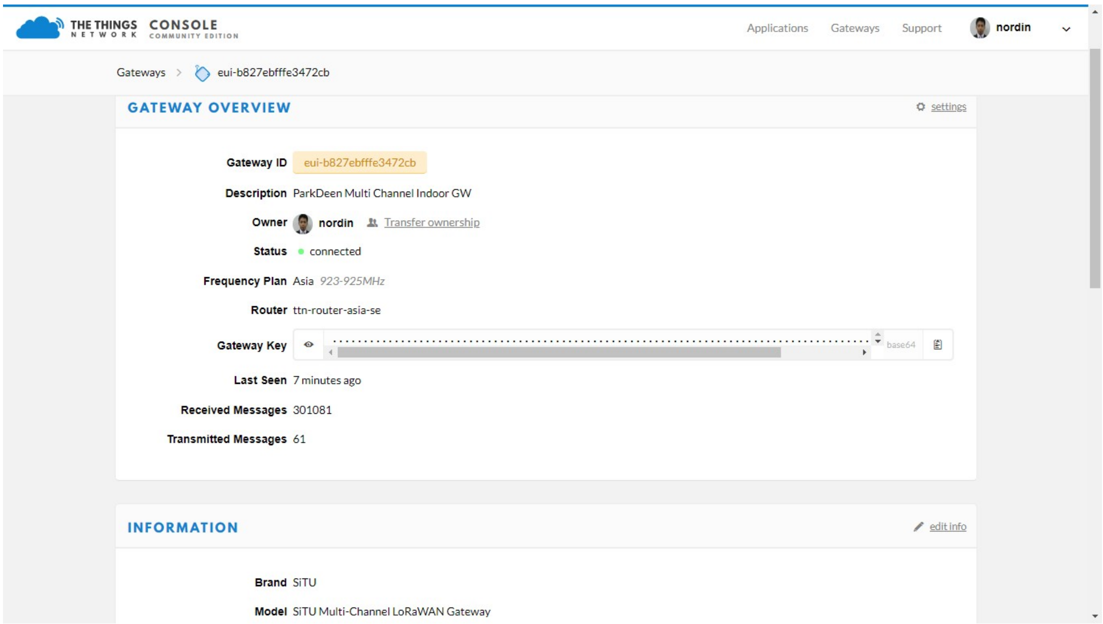


Figure 45 Inside and outside of hybrid IoT gateway based on LoRaWAN technology



Gateways > eui-b827ebfffe3472cb

GATEWAY OVERVIEW settings

Gateway ID eui-b827ebfffe3472cb

Description ParkDeen Multi Channel Indoor GW

Owner nordin Transfer ownership

Status ● connected

Frequency Plan Asia 923-925MHz

Router ttn-router-asia-se

Gateway Key base64

Last Seen 7 minutes ago

Received Messages 301081

Transmitted Messages 61

INFORMATION edit info

Brand SiTU

Model SiTU Multi-Channel LoRaWAN Gateway

Figure 46 Hybrid IoT Gateway status and registered as a public gateway in the TTN network

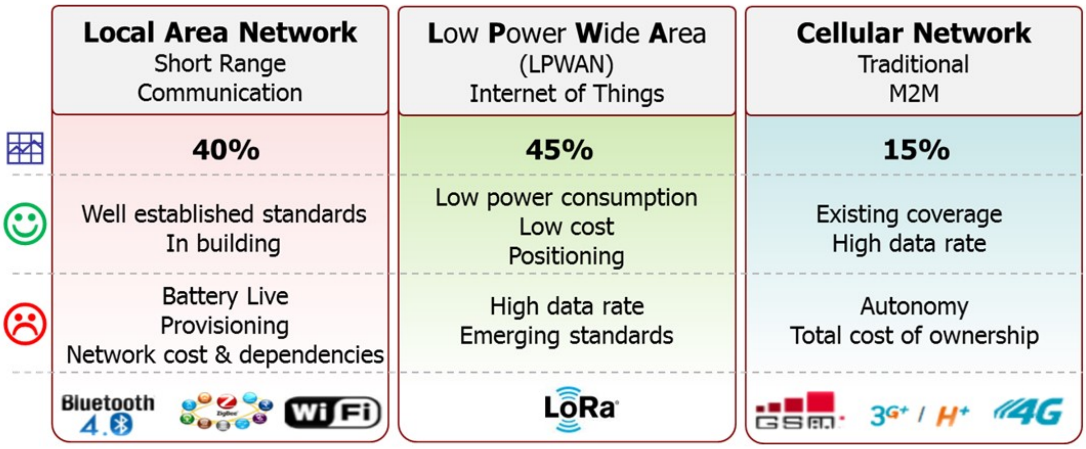
LoRa Connectivity

Here, we have a choice to use different connectivity solution in order to connect to the gateway. It can be either used the cellular (with subscription) or to use WiFi or LPWAN approach which are not required the subscription. In this project, we use the LoRa technology to connect the sensor node to the gateway, due to the long range capability to reach. Here, we explain the information on LoRa technology in brief.

LoRa® is the physical layer or the wireless modulation utilized to create the long range communication link. Many legacy wireless systems use frequency shifting keying (FSK) modulation as the physical layer because it is a very efficient modulation for achieving low power. LoRa® is based on chirp spread spectrum modulation, which maintains the same low power characteristics as FSK modulation but significantly increases the communication range. Chirp spread spectrum has been used in military and space communication for decades due to the long communication distances that can be achieved and robustness to interference, but LoRa® is the first low cost implementation for commercial usage.

The advantage of LoRa® is in the technology's long range capability. A single gateway or base station can cover entire cities or hundreds of square kilometers. Range highly depends on the environment or obstructions in a given location, but LoRa® and LoRaWAN™ have a link budget greater than any other standardized communication technology. The link budget, typically given in decibels (dB), is the primary factor in determining the range in a given environment. Below are the coverage maps from the Proximus network deployed in Belgium. With a minimal amount of infrastructure, entire countries can easily be covered.

In this project, we choose to work on LPWAN. Why LPWAN? One technology cannot serve all of the projected applications and volumes for IoT. WiFi and Bluetooth-Low Energy (BTLE) are widely adopted standards and serve the applications related to communicating personal devices quite well. Cellular technology is a great fit for applications that need high data throughput and have a power source. LPWAN offers multi-year battery lifetime and is designed for sensors and applications that need to send small amounts of data over long distances a few times per hour from varying environments, especially in the case of the moving gateway. Figure 47 shows the LPWAN (in particular, LoRa technology with other technologies). As mentioned in Figure 47, LoRa is suitable for the IoT application's connectivity due to able to reach long distance capability and also to send low date rate transmission.



	Local Area Network Short Range Communication	**Low Power Wide Area** (LPWAN) Internet of Things	**Cellular Network** Traditional M2M
📊	**40%**	**45%**	**15%**
☺	Well established standards In building	Low power consumption Low cost Positioning	Existing coverage High data rate
☹	Battery Live Provisioning Network cost & dependencies	High data rate Emerging standards	Autonomy Total cost of ownership
	Bluetooth 4.0 WiFi	LoRa	GSM 3G+ / H+ 4G

Figure 47 Value proposition of LPWAN (in our case is LoRa)

LoRaWAN™ defines the communication protocol and system architecture for the network while the LoRa® physical layer enables the long-range communication link. The protocol and network architecture have the most influence in determining the battery lifetime of a node, the network capacity, the quality of service, the security, and the variety of applications served by the network.

LoRaWAN™ is a Low Power Wide Area Network (LPWAN) specification intended for wireless battery operated Things (or device) in a regional, national or global network. LoRaWAN targets key requirements of IoT such as secure bi-directional communication, mobility and localization services. The LoRaWAN specification provides seamless interoperability among smart Things without the need of complex local installations and gives back the freedom to the user, developer, businesses enabling the roll out of Internet of Things.

LoRaWAN network architecture is typically laid out in a star-of-stars topology in which **gateways** is a transparent bridge relaying messages between **end-devices** and a central **network server** in the backend. Gateways are connected to the network server via standard IP connections while end-devices use single-hop wireless communication to one or many gateways. All end-point communication is generally bi-directional, but also supports operation such as multicast enabling software upgrade over the air or other mass distribution messages to reduce the on air communication time.

Communication between end-devices and gateways is spread out on different **frequency channels** and **data rates**. The selection of the data rate is a trade-off between communication range and message duration. Due to the spread spectrum technology, communications with different data rates do not interfere with each other and create a set of "virtual" channels increasing the capacity of the gateway. LoRaWAN data rates range from 0.3 kbps to 50 kbps. To maximize both battery life of the end-devices and overall network capacity, the LoRaWAN network server is managing the data rate and RF output for each end-device individually by means of an **adaptive data rate** (ADR) scheme.

In Malaysia, there is no regulation has been imposed by the Malaysia Communication and Multimedia Commission (MCMC) yet. Nevertheless, since the international proposed spectrum for LoRa connectivity is fall within short range device (SRD) band, it is said legally to be used by the public without any license. In particular, in Malaysia is suggested to use channel plan within the range 920-923 MHz: Hong Kong, Japan, Malaysia, Singapore, Thailand, and Vietnam.

National wide networks targeting internet of things such as critical infrastructure, confidential personal data or critical functions for the society has a special need for secure communication. This has been solved by several layer of encryption:

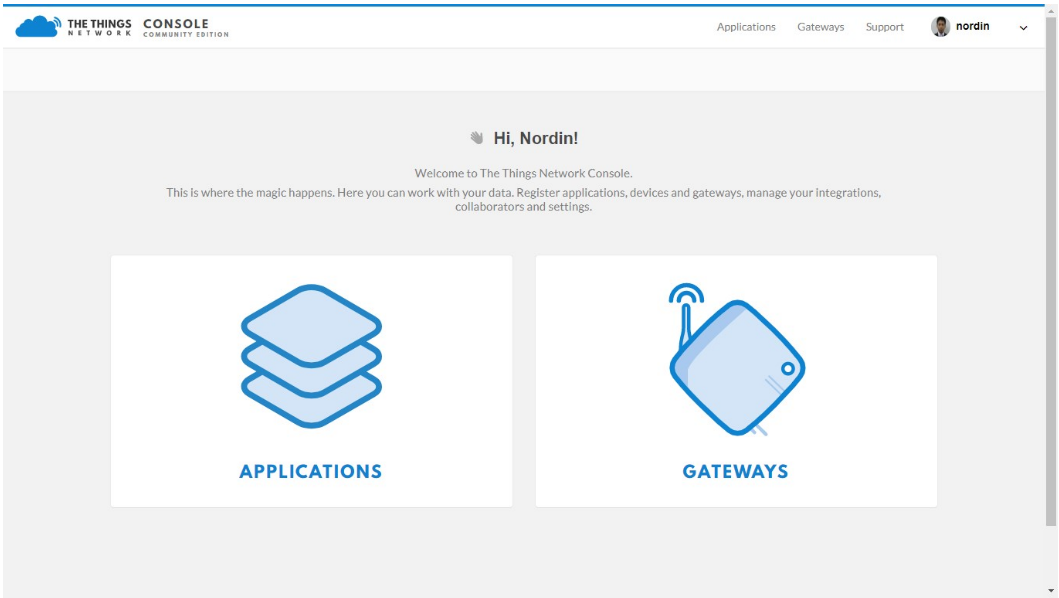
- Unique Network key (EUI64) and ensure security on network level
- Unique Application key (EUI64) ensure end to end security on application level
- Device specific key (EUI128)

The following are some of the brief information of LoRa technology:

- LoRa is pioneered by SEMTECH Corporation
- PHY base technology (Layer 1 (physical))
- Enables long-range communication link
    - Highly dependent on environment
        - Cities / buildings: 2.5km
        - Countryside: up to 15km+
- Data rate 0.3 – 50 kbps
- Frequency band (Malaysia): 919-923 MHz – unlicensed
- Duty cycle : 10%

Network Cloud

Today, LoRaWAN networks are being built by different parties, most of which are commercial. The commercial actors tend to build closed networks with a subscription-based revenue model. In other words, if you want to use their network, you have to pay for it. However, as LoRaWAN is an open standard, it also allows building an open source and free network. This is precisely what the non-profit foundation 'The Things Network' (TTN) aims to do: to create a community LoRaWAN network that that can be used for free by everyone. The software TTN develops is released under an open-source license. The ultimate goal of TTN is to have a set of networks that together cover the whole world (The Things Network, 2016), but they started small. Initially, TTN started with a proof-of-concept network in Amsterdam, The Netherlands, where they built their own software for a backend and crowd-sourced the gateways: local companies and organizations paid for the gateways and put them on their roof. In this project, we are using the network cloud provider that offering the LoRaWAN protocol stacks, offered by The Things Network (TTN). The Things Network is building a network for the Internet of Things by creating abundant data connectivity, so applications and businesses can flourish. The technology we use is called LoRaWAN and it allows for things to talk to the internet without 3G or WiFi. Therefore it requires no WiFi codes and no mobile subscriptions. It features low battery usage, long range and low bandwidth, which is perfect for the internet of things. The front page of the network server of TTN is shown in Figure 48. In particular, the hybrid IoT gateway has been registered and published in the TTN website as one of the public gateway based on the LoRa technology.



Hi, Nordin!

Welcome to The Things Network Console.

This is where the magic happens. Here you can work with your data. Register applications, devices and gateways, manage your integrations, collaborators and settings.

Figure 48 The TTN front page of this project

The backend systems of The Things Network are responsible for routing Internet of Things data between devices and applications. A typical Internet of Things network requires gateways as a bridge between specific radio protocols and the Internet. In cases where the devices themselves support the IP stack, these gateways only have to forward packets to the Internet. Non-IP protocols such as LoRaWAN require some form of routing and processing before messages can be delivered to an application. The Things Network is positioned between the gateways and the applications and takes care of these routing and processing steps.

MQTT Broker and MQTT Client

MQTT is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol. It was designed as an extremely lightweight publish/subscribe messaging transport. The Things Network uses MQTT to publish device activations and messages, but also allows you to publish a message for a specific device in response. In this network server of TTN, it has the built in MQTT broker to forward and publish the message according to the rule.

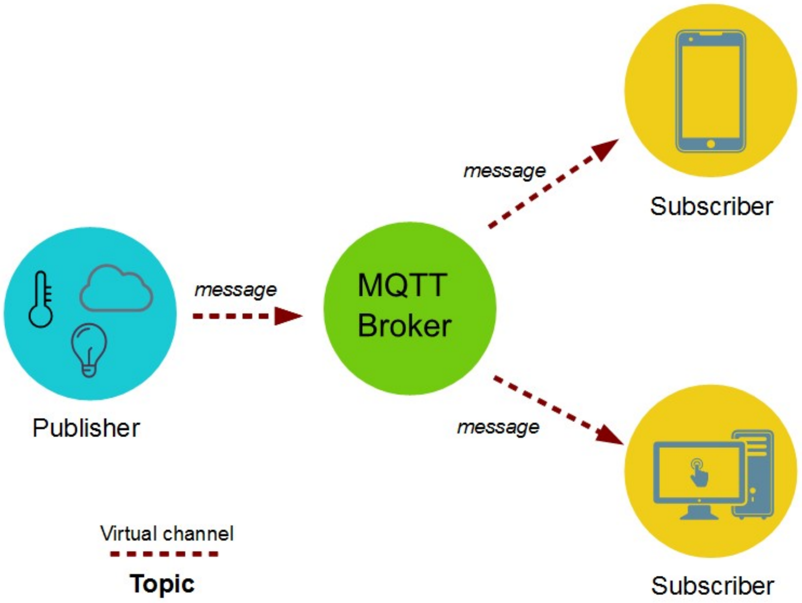


Figure 49 MQTT Broker to push the data to the specific destination

Node Red

Node-RED is a flow-based development tool developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things. Node-RED provides a browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows created in Node-RED are stored using JSON. Since version 0.14 MQTT nodes can make properly configured TLS connections.

In this project, we process the data receive from TTN forwarded by MQTT Broker to be received at Node-RED via its MQTT client based on the subscribed data topics. As shown in Figure 50, the data is received from MQTT Broker in the JSON format to be processed in this Node-RED.

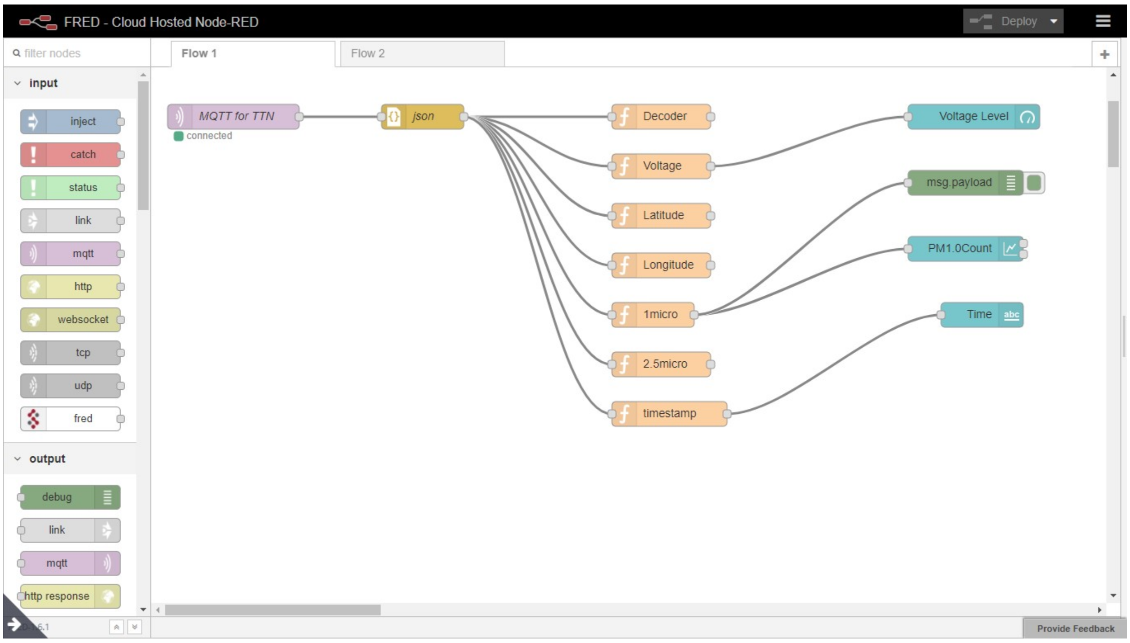
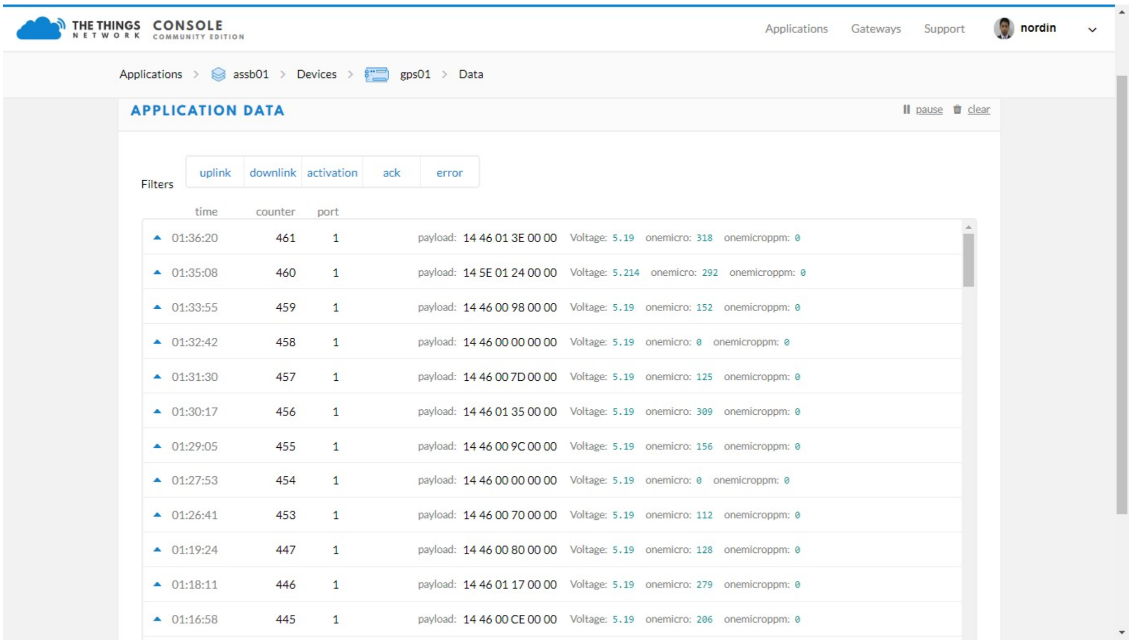


Figure 50 Online Node-RED by FRED to wiring the data subscribed by MQTT Broker of TTN



time	counter	port				
▲ 01:36:20	461	1	payload: 14 46 01 3E 00 00	Voltage: 5.19	onemicro: 318	onemicroppm: 0
▲ 01:35:08	460	1	payload: 14 5E 01 24 00 00	Voltage: 5.214	onemicro: 292	onemicroppm: 0
▲ 01:33:55	459	1	payload: 14 46 00 98 00 00	Voltage: 5.19	onemicro: 152	onemicroppm: 0
▲ 01:32:42	458	1	payload: 14 46 00 00 00 00	Voltage: 5.19	onemicro: 0	onemicroppm: 0
▲ 01:31:30	457	1	payload: 14 46 00 7D 00 00	Voltage: 5.19	onemicro: 125	onemicroppm: 0
▲ 01:30:17	456	1	payload: 14 46 01 35 00 00	Voltage: 5.19	onemicro: 309	onemicroppm: 0
▲ 01:29:05	455	1	payload: 14 46 00 9C 00 00	Voltage: 5.19	onemicro: 156	onemicroppm: 0
▲ 01:27:53	454	1	payload: 14 46 00 00 00 00	Voltage: 5.19	onemicro: 0	onemicroppm: 0
▲ 01:26:41	453	1	payload: 14 46 00 70 00 00	Voltage: 5.19	onemicro: 112	onemicroppm: 0
▲ 01:19:24	447	1	payload: 14 46 00 80 00 00	Voltage: 5.19	onemicro: 128	onemicroppm: 0
▲ 01:18:11	446	1	payload: 14 46 01 17 00 00	Voltage: 5.19	onemicro: 279	onemicroppm: 0
▲ 01:16:58	445	1	payload: 14 46 00 CE 00 00	Voltage: 5.19	onemicro: 206	onemicroppm: 0

Figure 51 Incoming data from sensor node at different interval (from sensor node at Bukit Jalil)

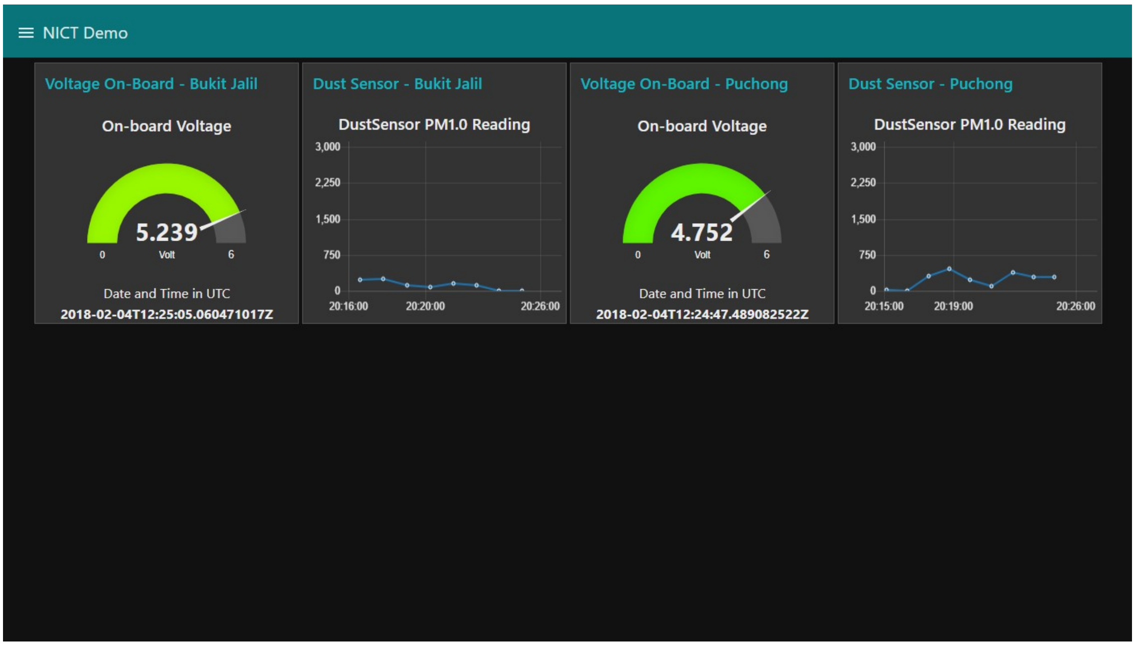
Results

Data traffic: Figure 51 shows the data traffic transferred from sensor node and arrive at the network cloud server via hybrid IoT gateway. It shows the current status of the data of each sensor nodes. In this project, we installed two hybrid IoT gateway at two different location, namely as Puchong and Bukit Jalil, respectively.

Dashboard (Visualization): The data from sensors are able to be visualized at real-time at dashboard of the same online FRED (replication of Node-RED). The dashboard can be viewed lively at the following URL:

https://zaki1575.fred.sensetecnic.com/api/ui/#/6

In the dashboard, we presented the data from two sensor nodes (with the air monitoring sensors). In the data, we published the information on the voltage level of the sensor node, and also the PM1.0 reading. The dashboard snapshots are shown in Figure 52, Figure 53, and Figure 54. Based on the two illustrations of these figures, it is verified that the air quality during the measurement are very good at both respective location.



Date and Time in UTC
2018-02-04T12:25:05.060471017Z

Date and Time in UTC
2018-02-04T12:24:47.489082522Z

Figure 52 The illustration of the air quality monitoring dashboard

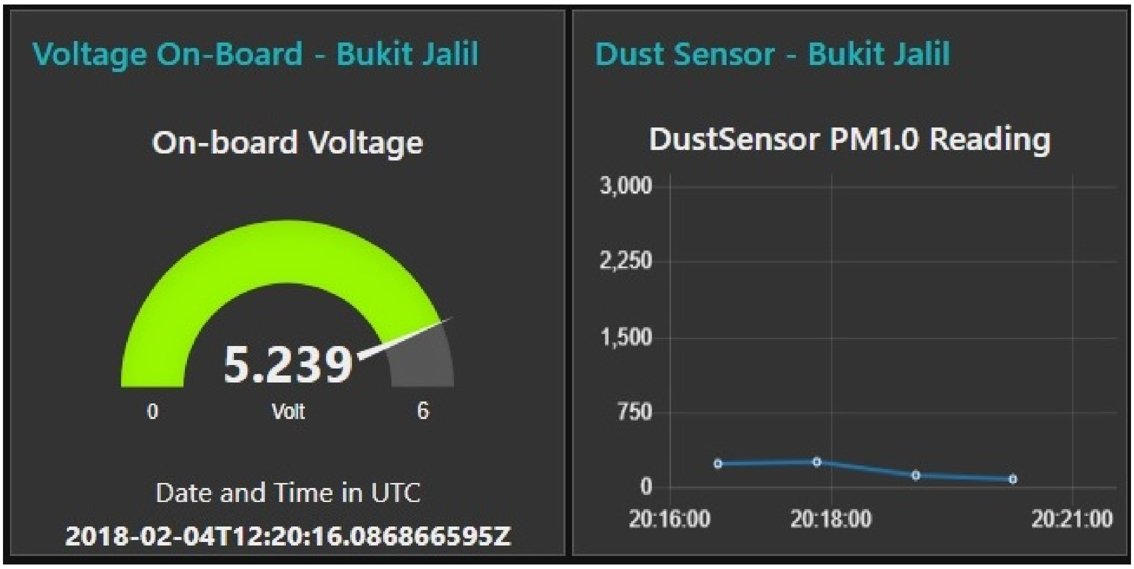


Figure 53 This dashboard shows the illustration of the data collected from the sensor node at Bukit Jalil

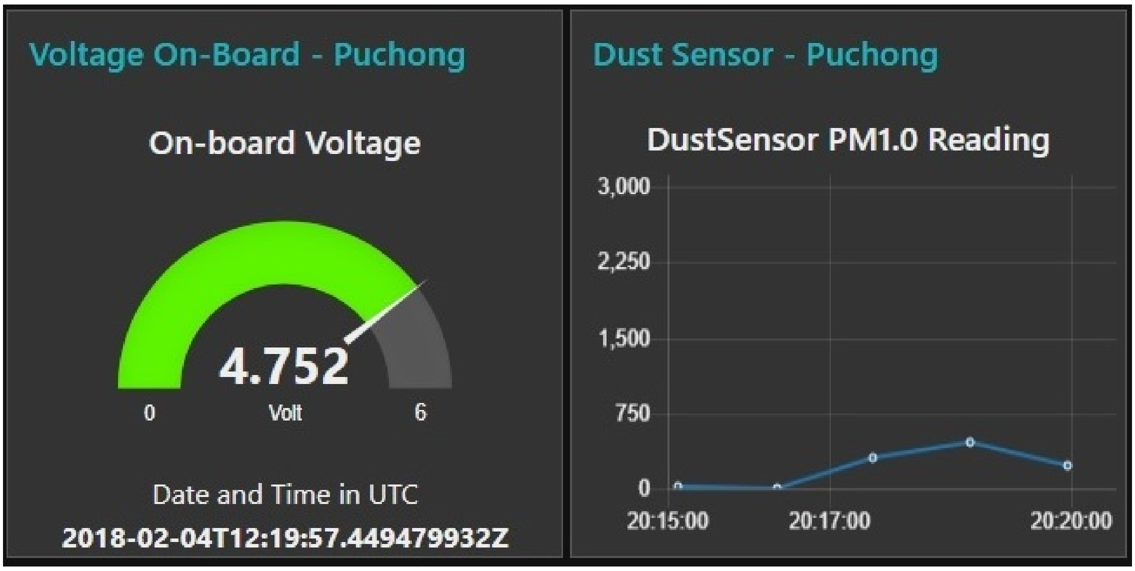


Figure 54 This dashboard shows the illustration of the data collected from the sensor node at Puchong

Conclusions

In this project, we have developed the air monitoring IoT solution with LoRaWAN technology. End-to-end technology components have been developed and tested. In particular, we have developed sensor nodes and also hybrid IoT gateway to support the LoRa connectivity from all the sensor nodes. The system is currently running and working well. It is recommended to expand the installation of the sensor nodes and hybrid IoT gateway nationwide in order to collect the current air quality reading at respective locations.

Mobile surveillance technology applications to Mobile-IoT

Introduction

Terrorist thread in Singapore is growing bigger than ever before. Terrorists consider Singapore as prized target due to:

- Sizable Muslim population
- Safe, secure/stable country
- Links with developed nations: USA, Japan, UK etc.

If terrorists can attack Singapore, it would have significant impact in ASEAN and the world, especially in Indonesia and Malaysia.

Mobile surveillance (Figure 55; fast deployment) using camera/video in anywhere in Singapore using battery powered device/solar panels is hence become essential to combat the threat. Video surveillance will be transmitted back to the control center over the unused VHF/UHF channels. Channel aggregation could be used to increase the system surveillance capacity and reliability.

Figure 55 Mobile video surveillance

Unused VHF/UHF channels in Singapore

With a careful but open approach, Singapore IMDA issued a consultation paper on secondary usage of VHF/UHF to seek comments and feedback from the industry in June 2013 [65]. One year later June 16, 2014, the IMDA published its response to the submissions made during the consultation period on how to regulate secondary usage of VHF/UHF in Singapore. These regulations [66], which came into effect on November 1, 2014, set Singapore alongside the US and the UK as global leaders in the development and usage of VHF/UHF channels on a license-exempt basis.

These regulations allow the secondary usage of the VHF/UHF services with certainty, especially in the amount of spectrum set aside for wireless devices. The regulations also specify the licensing and regulatory requirement, as well as the operating parameters of wireless devices, Geo-location database, etc.

IMDA's regulations specify measures to protect TV broadcasters by limiting the use of adjacent channels and placing limits on the wireless device's power emissions. Existing users of wireless microphones and public mobile services are also protected through a dedicated channel and a registration process for extraordinary events.

From now until 2020, wireless devices can use up to 189 MHz of spectrum when digital TV broadcasting is steadily replaced analogue TV broadcasting. This spectrum consists of 3 VHF channels and 21 UHF channels. After 2020, wireless devices can use 183 MHz when some of the frequencies will be used for other services.

Mobile surveillance can make use of these VHF/UHF channels for fast deployment and cost-effective security monitoring solution.

Mobile surveillance related experiments by NICT

In this section, mobile surveillance related experiment conducted by NICT [67] is introduced. Figure 56 shows an architecture overview of the mobile surveillance system. The mobile surveillance system consists of the mobile component and the back end. The mobile component is typically put on the roadside light poles or inside a mobile device such as car etc. The mobile component consists of a camera, an encoder, a PC. The back-end component consists of the LTE system and storage/monitoring servers. The video captured by the camera is encoded by the encoder and sent back to the monitor server by the PC through the LTE system. The streaming video is then transmitted through the Internet to the storage/monitoring servers in the control room for storage and monitoring purposes.

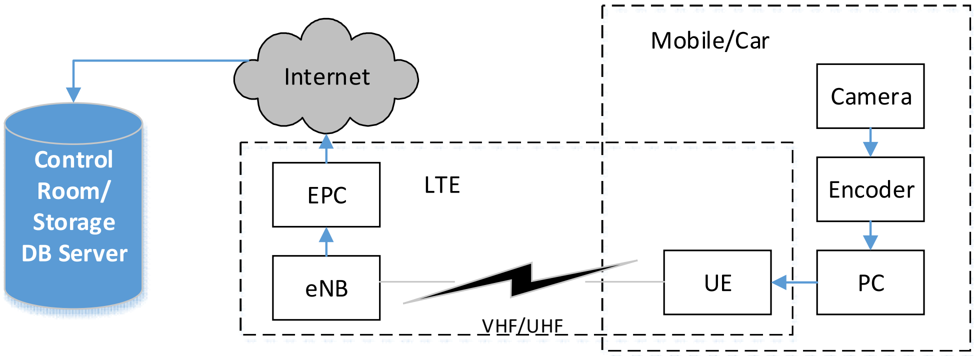


Figure 56 Architecture of mobile surveillance system

The LTE over VHF/UHF consists of the UE in the car, eNB and the EPC in the roadside building as in Figure 57, eNB uses a dipole antenna stationed on the 15m high rooftop. Total antenna height is 17 m. UE uses a whip antenna on car rooftop with total height of 1.5 m. VHF/UHF channels used in the experiment has been approved by Singapore Geo-Location Database.

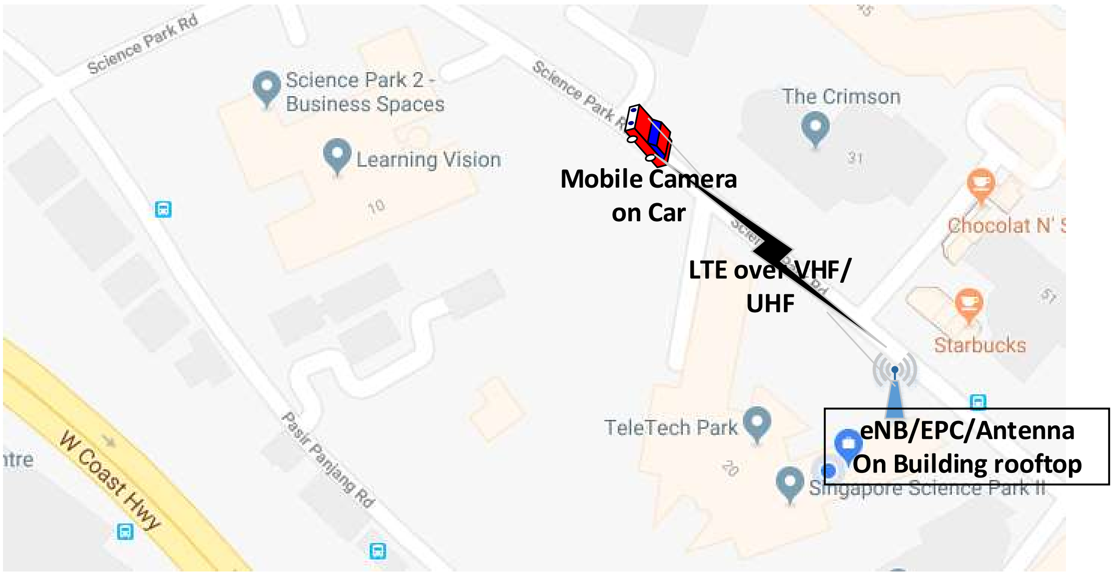
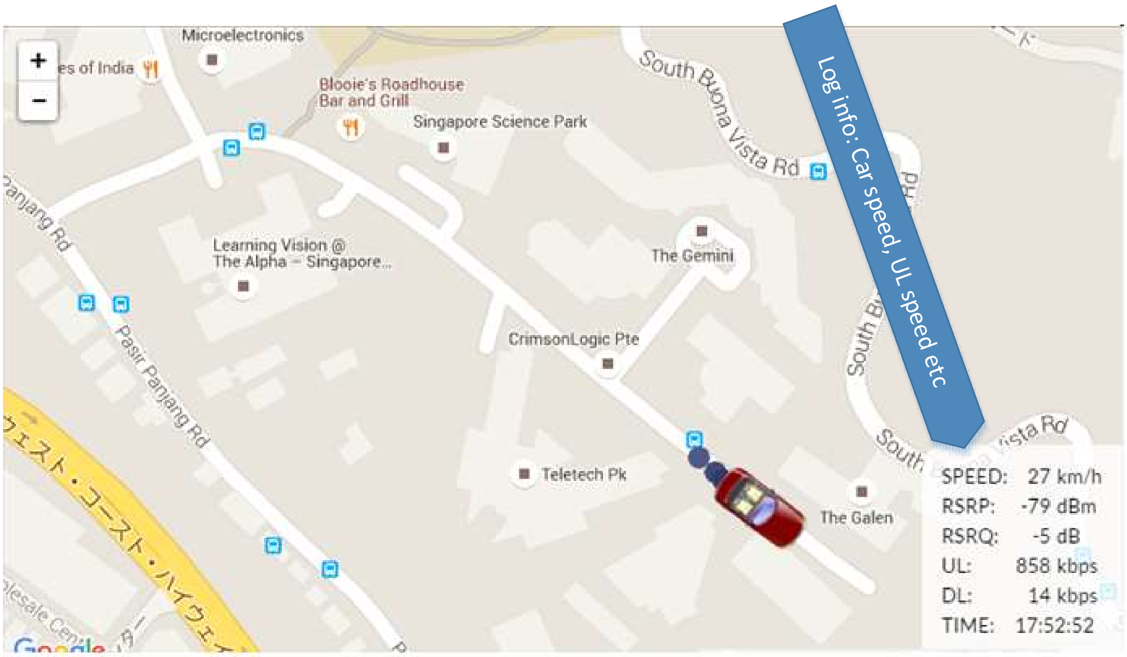


Figure 57 Experimental setup

Figure 58 shows the monitoring screen at the monitoring server for the live video streaming from the car. The monitoring server also has the map consists of live GPS data and other LTE parameters as in Figure 59. The car symbol on the map indicates its current location and direction based on the GPS data sent from the car. The current LTE parameters are shown in the bottom-right corner including the car speed, RF signal strength, uplink/downlink speed etc.

There's delay time of about 2 seconds for the live video due to the encoding time and the transmission delay. The video streaming and GPS data can be stored in the storage database and processed further for monitoring purposes.

Figure 58 Experiment result



Log info: Car speed, UL speed etc

SPEED:	27 km/h
RSRP:	-79 dBm
RSRQ:	-5 dB
UL:	858 kbps
DL:	14 kbps
TIME:	17:52:52

Figure 59 GPS and LTE Logging information

Mobile surveillance using channel aggregation

Further examining the experiment results in the previous section shows that throughput is not stable, especially in the NLOS (Non- Line of Sight) scenarios. Channel bandwidth is also limited in some cases, hindering the surveillance capacity.

Channel aggregation could be used to alleviate these problems and increases the system bandwidth and surveillance ability. Figure 60 shows a typical scenario when channel aggregation could be used. Mobile surveillance stations 1-13 are setup in different spots. Each station could use multiple unused VHF/UHF channels to transfer the surveillance data to the gateway (G0, G1, and G2). The Gateway in turn will forward the data to the control station (CS) through the Internet for further action.

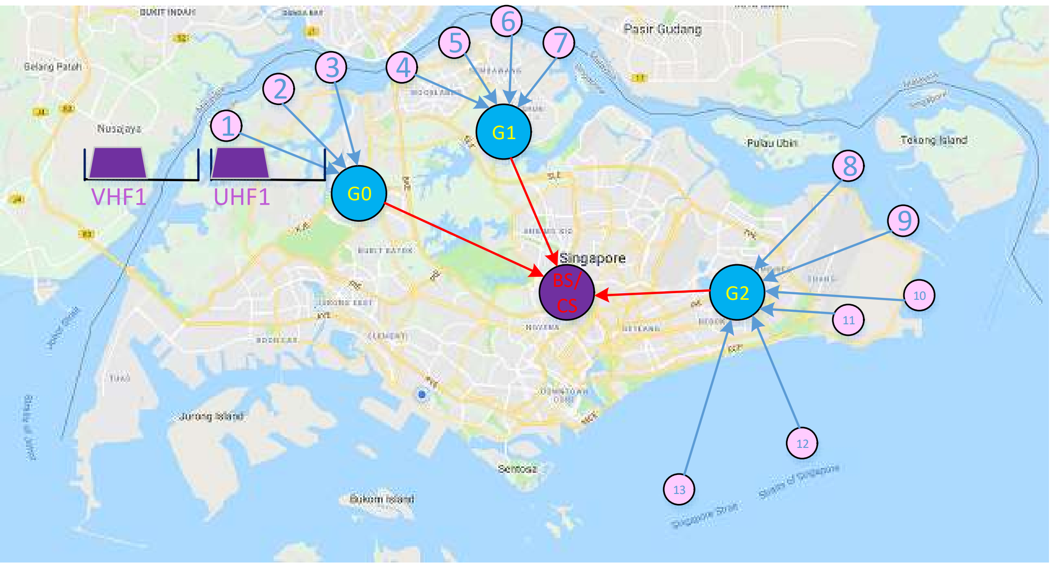


Figure 60 Mobile surveillance using channel aggregation

Evaluation by computer simulation

Simulations were conducted to evaluate the effectiveness of the proposed channel aggregation solution. The simulations are done on the NS2 simulator [68]. Table 6 shows simulation conditions. The proposed solution is evaluated using the IEEE802.11 wireless interface over VHF/UHF channel. Destination Sequence Distance Vector (DSDV [69]) is chosen as the ad hoc routing protocol due to its popularity.

Table 6 Simulation conditions

Simulation parameters	Value
Routing Protocol	DSDV
Simulation time	900s
CBR rate	10Pkt/s, 4 CBR sources
CBR packet	512 Bytes
RTS/CTS/ACK	44/38/38 Bytes
DIFS	50µs
SIFS	20µs
Preamble	144 bits
CBR start at	100s
Mobility	Yes
Mobility space	1000x1000m
Number of mobile nodes	10
Wireless bandwidth	1Mbps

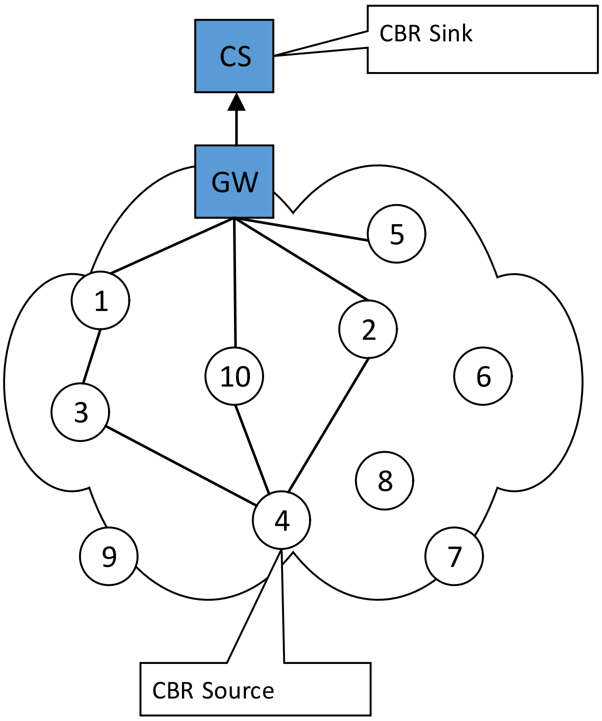


Figure 61 Mobile surveillance using channel aggregation

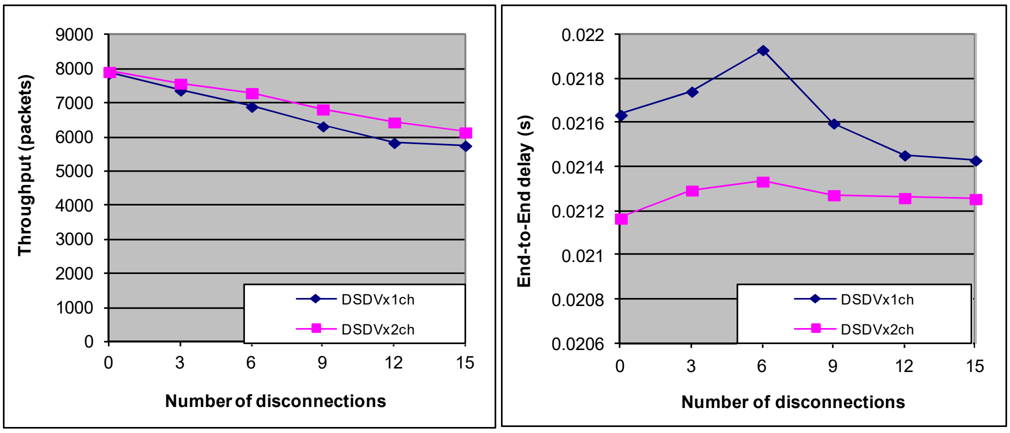
In the simulation, a hybrid network consists of a fixed network and a wireless network as in the Figure 61 will be simulated in various simulation scenarios. The fixed network consists of the Control Station (CS) and the Gateway (GW). The wireless network consists of 10 mobile nodes.

There are 3 CBR traffic sources between mobile nodes, one more CBR source from a mobile surveillance node (node 4) to the GW and in turn to the CS node. So totally there are 4 CBR traffic sources. A CBR source sends 10 packets, 512 bytes each, per second. Each of all mobile nodes uses 1 wireless interface, except node 4 uses 2 wireless interfaces. Mobile nodes move in a 1000m x 1000m area.

The following metrics are used for performance comparisons in the simulation:

- Throughput: Number of packet received at the destination (CS node) over the simulation duration.
- End-to-End Delay: The average time taken from node 4 to CS node over the simulation duration.

All mobile nodes are randomly moving using the random waypoint mobility model [70]. The mobile node's maximum speed is 20m/s with 10s pause time. All the simulations are lasted for 900 seconds. Links between CS and GW are 5Mbps with 2ms delay.



(a) Throughput (b) End-to-end delay

Figure 62 Computer simulation results

The Simulation results using DSDV are shown in Figure 62. The number of disconnection is the number of times node 4 is disconnected from GW due to its movement. It's obvious that as the number of disconnection increases, the throughput decreases because the network needs time to stabilize after each disconnection and packet has to be buffered or may have to take a longer route to the destination.

Figure 62(a) shows the throughput of node 4 using 1 and 2 wireless interfaces respectively (DSDVx1ch and DSDVx2ch). When a connection with current GW is lost, it takes longer time for node with 1 wireless interface to react to the changes. A node with 2 wireless interfaces reacts better when the number of disconnection increases. Figure 62(a) also shows that using 2 wireless interfaces improves the throughput significantly.

Figure 62(b) shows the average end-to-end delay. The end-to-end delay will increase when number of disconnection increase because packets have to buffer at intermediate nodes before the route is re-established or they have to take the longer route. However, a node with 2 wireless interfaces still shows some improvement in end-to-end delay compared with the normal node. This is due to the fact that utilizing 2 wireless interfaces will reduce the disconnection time because multiple interfaces are not disconnected altogether at the same time.

Conclusions

Mobile surveillance anywhere in Singapore using battery powered device/solar panels is essential to combat various security threats for industrial, commercial and military areas. A good mobile surveillance should be rapidly deployed, installed and can be operated easily.

The proposed experimental system provides just that. This system can be developed further to be commercialized and to provide a fast deployed mobile surveillance solution for various purposes from monitoring natural disaster areas to security areas in a cost effective manner.

Further enhancement to the system using channel aggregation is proposed to increase its capacity and reliability. Simulation results show that channel aggregation can improve system throughput and end-to-end delay.

Wireless grid technology applications to Mobile-IoT

Introduction

Although SUN, the most typical embodiment for the wireless grid structure which is one study theme of NICT, was at first based on the application to smart meters, it is easily expected that the application of SUN will not be exclusive to the smart meter system [71]-[73]. In other words, if SUN is developed so as to have such characteristics as long-term action due to low-energy action, etc. and the low-speed and low-capacity communications found in the smart meter use, it is expected to be utilized in infinitely various fields. The IoT (Internet of Things) and M2M (Machine-to-Machine) fields whose significance is of current interest are no exception. The applied areas for the SUN radio are expected to become diversified depending on the diversification of the future radio communication service demand or the diversification of applications.

Figure 63 shows the concept of the extended SUN from such a standpoint. The figure shows three diversification forms in concrete terms. That is:

1. "High capacity data collection network" where the power supply is especially not limited and which is characterized by mesh topology with many radios
2. "Ultra-low-energy operation network" which is characterized by low power consumption assuming the time of battery driving, etc.
3. "Reinforced mesh network" which assumes service development in the environment, such as a devastated district and plants, where the application of radio communication links has not been assumed in the past.

This work especially assumes the effective applications of high capacity data network and ultra-low-energy operation network to the Mobile-IoT systems.

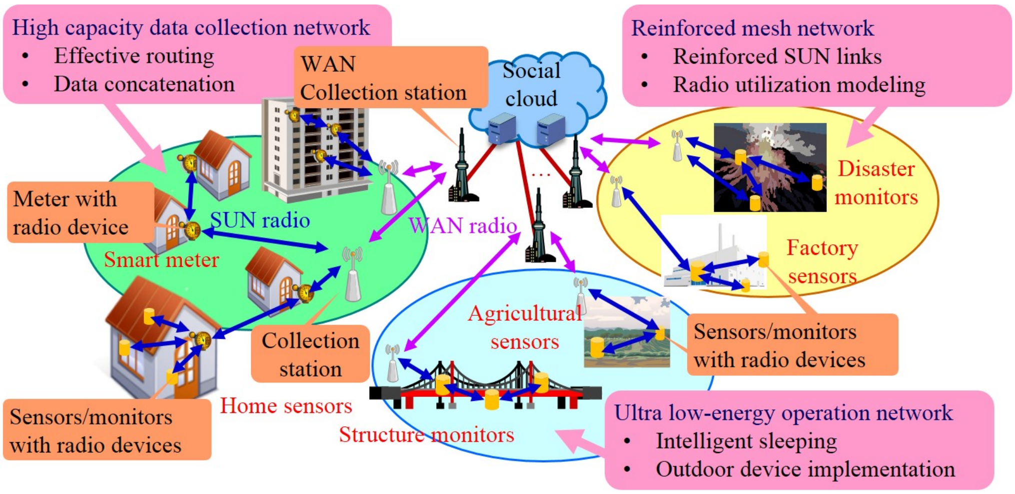


Figure 63 Concept of extended SUN

Layer 2 – Routing control study for the high capacity data collection network

This section reports on NICT's activities related to the Layer2-Routing (L2R) control study for the high capacity data collection network. The greatest characteristic of the routing control technology in this study is to control the action of the information exchange between terminals and so on by using the control by MAC layer known as Layer 2 in the Personal Area Network [75]. Therefore, the information to be handled is described by IE (information element), an information unit handled by the MAC layer, and is exchanged and handled by the function of the MAC layer. It is believed that this reduces the redundancy compared to the conventional routing control handled by IP (internet protocol), as Layer 3, and simple handling becomes possible [76]-[78].

We describe the main technologies of the L2R control for which we implemented research and development. First function is an autonomous mesh construction function. This function allows each radio device, which composes a mesh structure, to sense the whole mesh structure by regularly sending out a mesh topology construction signal including one's own information -- connecting to other radios and so on -- and by receiving the same signal from other radios. The radio which sends out the first mesh topology construction signal is called the mesh route, and signals from other devices are composed on the basis of those from the mesh route.

Figure 64 Autonomous mesh-construction function shows an operating example of the mesh topology construction signals exchanges in the mesh by mesh root R. In Figure 64(i), the mesh root R which attempts to construct a mesh sends out the first mesh-topology construction signals including the information of R, and devices A and B receive the signals. This allows A and B to detect the presence of a mesh and the presence of R adjacent to them at the same time. Furthermore, they can ascertain the communication quality with respect to R based on the received mesh-building signal.

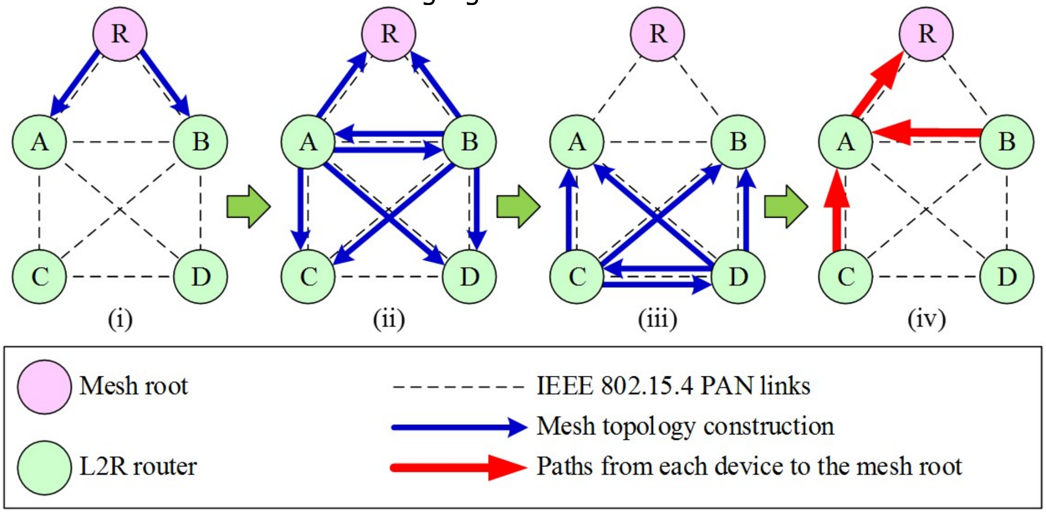

R
A
B
C
D
(i)
(ii)
(iii)
(iv)
Mesh root
L2R router
IEEE 802.15.4 PAN links
Mesh topol ogy construction
Paths from each device to the mesh root

Figure 64 Autonomous mesh-construction function

Next, A and B send out the mesh topology construction signals with an aim to provide the same information to other devices including R (refer to Figure 64(ii)). Devices C and D, which receive the signals, implement the same procedure (refer to Figure 64(iii)). Eventually, five devices, R, A, B, C and D, build a mesh, ascertain adjacent devices, reachable devices and the communication quality between them, and can establish the proper relay path with a special radio as a destination, as shown in Figure 64(iv).

The second function is a data frame concatenation function. This function reduces the number of data frames by combining multiple data frames addressed to the same destination and by relaying them as one data frame to a next device, and prevents performance degradation of data collection by the reduction of bottleneck blockage and collision.

Figure 65 shows an operating example for the data concatenation function. Here, in the same mesh as in Figure 64, the case is indicated that devices A, B, C, D, excluding the mesh root R, send the respective data frames to R and the frames are relayed through the route in Figure 64(iv). In ordinary circumstances, it sometimes occurs that the four data frames from the radios reach R almost at the same time, and collision of frames occurs at that time and the R side cannot successfully receive them. On the other hand, in the data frame coupling function as shown in Figure 65, for this example, one data frame combines the data frame of the same destination, and the one data frame which is relayed while combining the four data frames accordingly reaches R. Therefore, performance degradation due to the collision as described above can be avoided.

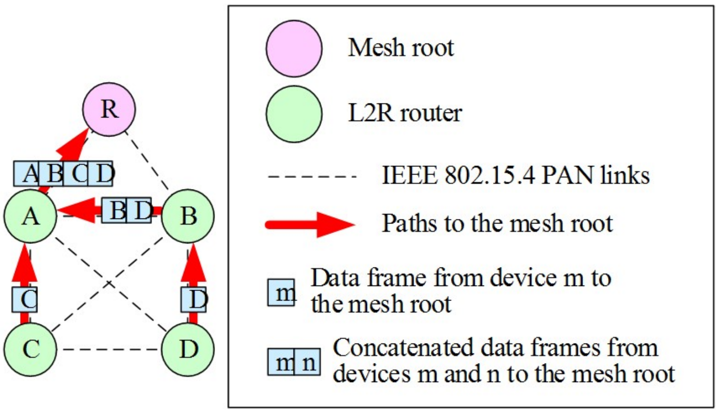

R
A B C D
A
B D
B
C
D
C
D
Mesh root
L2R router
IEEE 802.15.4 PAN links
Paths to the mesh root
m
Data frame from device m to the mesh root
m n
Concatenated data frames from devices m and n to the mesh root

Figure 65 Data frame concatenation function

The third function is a multiple-mesh supporting function. This function differentiates data frames depending on the assumed applications, and the devices properly handle the respective data frames -- selectively construct the routes to the mesh root.

Figure 66 shows an operating example of this function. The case in this function is indicated that two mesh root devices, R1 and R2, compose respective meshes and they provide different services. In this case, the radios, A, B, C and D, can join in any two meshes at the same time, and the data frames related to the respective services are relayed according to the routes on the respective meshes.

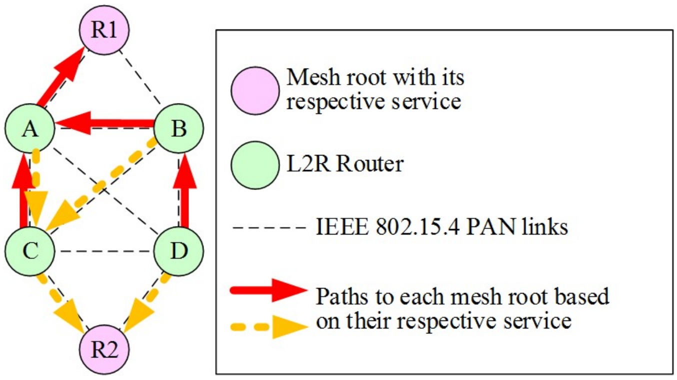

R1
A
B
C
D
R2
Mesh root with its respective service
L2R Router
IEEE 802.15.4 PAN links
Paths to each mesh root based on their respective service

Figure 66 Multiple-mesh supporting function

In order to evaluate the assumed L2R network, NICT has conducted the experiments in Yokosuka, Japan. First, data concatenation function is evaluated in the indoor experiments. Figure 67 shows the employed radio devices to evaluate the L2R mesh and an example of monitored L2R mesh behavior is shown in Figure 68.

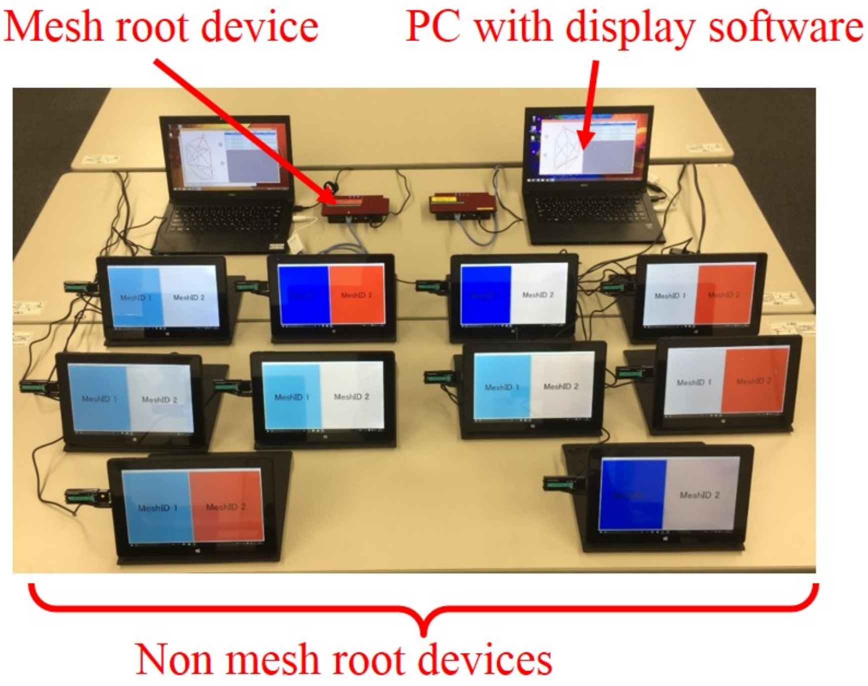
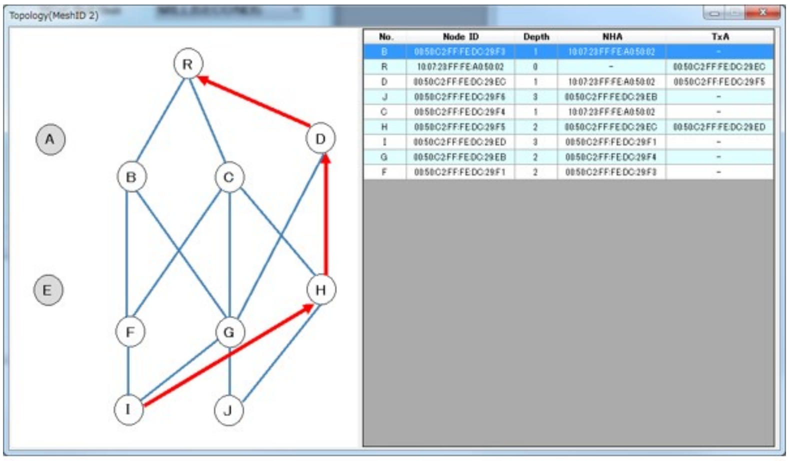


Figure 67 Employed radio devices to evaluate the L2R mesh

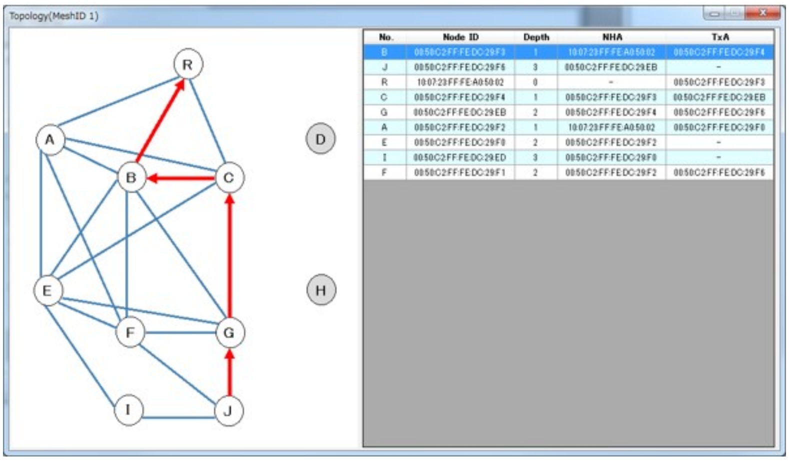
Figure 67 shows the radio devices allocation, where two multiple meshes are constructed by the mesh root R1 and R2 sharing all radio devices, only one of which employs the data concatenation. Experimental results are shown in Figure 68 and Table 7. The results confirm that an employment of data concatenation function effectively works to improve the total sending time and frame success rate performance by decreasing the number of forwarded frames that could results in serious collisions.

Table 7 Experimental results data concatenation evaluation (1)

Exp. ID	Dcat buffering time (s)	Data interval (s)	Ave. Dcat size	Total sending time (ms)		Frame success rate	
				Mesh ID 1	Mesh ID 2	Mesh ID 1	Mesh ID 2
1	5	2	3.46	108643.2	82143.31	93.9%	99.4%
2	10	2	5.77	109056	69089.12	94.4%	98.2%



No.	Node ID	Depth	NHA	TxA
B	00:50:C2:FF:FE:DC:29:F3	1	10:07:23:FF:FE:A0:50:02	-
R	10:07:23:FF:FE:A0:50:02	0	-	00:50:C2:FF:FE:DC:29:EC
D	00:50:C2:FF:FE:DC:29:EC	1	10:07:23:FF:FE:A0:50:02	00:50:C2:FF:FE:DC:29:F5
J	00:50:C2:FF:FE:DC:29:F6	3	00:50:C2:FF:FE:DC:29:EB	-
C	00:50:C2:FF:FE:DC:29:F4	1	10:07:23:FF:FE:A0:50:02	-
H	00:50:C2:FF:FE:DC:29:F5	2	00:50:C2:FF:FE:DC:29:EC	00:50:C2:FF:FE:DC:29:ED
I	00:50:C2:FF:FE:DC:29:ED	3	00:50:C2:FF:FE:DC:29:F1	-
G	00:50:C2:FF:FE:DC:29:EB	2	00:50:C2:FF:FE:DC:29:F4	-
F	00:50:C2:FF:FE:DC:29:F1	2	00:50:C2:FF:FE:DC:29:F3	-



No.	Node ID	Depth	NHA	TxA
B	00:50:C2:FF:FE:DC:29:F3	1	10:07:23:FF:FE:A0:50:02	00:50:C2:FF:FE:DC:29:F4
J	00:50:C2:FF:FE:DC:29:F6	3	00:50:C2:FF:FE:DC:29:EB	-
R	10:07:23:FF:FE:A0:50:02	0	-	00:50:C2:FF:FE:DC:29:F3
C	00:50:C2:FF:FE:DC:29:F4	1	00:50:C2:FF:FE:DC:29:F3	00:50:C2:FF:FE:DC:29:EB
G	00:50:C2:FF:FE:DC:29:EB	2	00:50:C2:FF:FE:DC:29:F4	00:50:C2:FF:FE:DC:29:F6
A	00:50:C2:FF:FE:DC:29:F2	1	10:07:23:FF:FE:A0:50:02	00:50:C2:FF:FE:DC:29:F8
E	00:50:C2:FF:FE:DC:29:F8	2	00:50:C2:FF:FE:DC:29:F2	-
I	00:50:C2:FF:FE:DC:29:ED	3	00:50:C2:FF:FE:DC:29:F8	-
F	00:50:C2:FF:FE:DC:29:F1	2	00:50:C2:FF:FE:DC:29:F2	00:50:C2:FF:FE:DC:29:F6

Figure 68 Monitored L2R mesh behavior

We have further considered on the suitable metric employment in the assumed L2R mesh. We have introduced two different metric employments:

- RSW: to consider only hop number to the root
- RSW+: to consider RSSI for the link as well as the hop number to the root

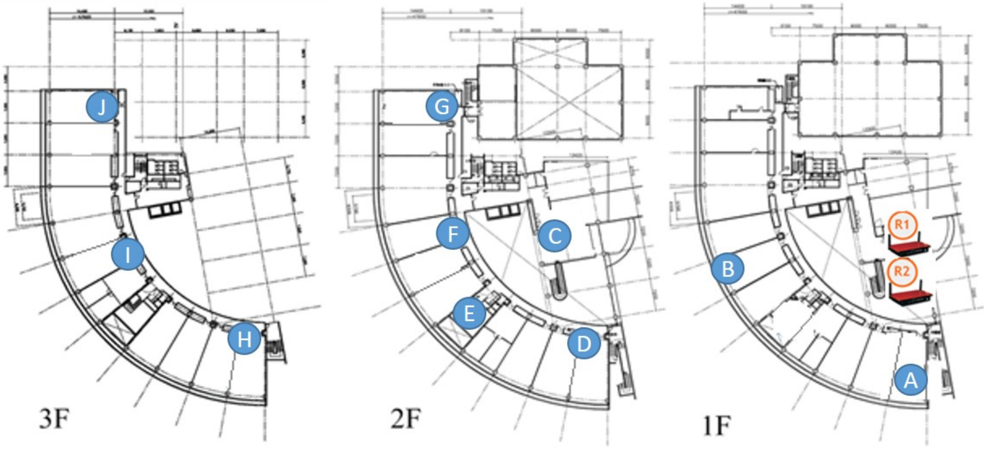


Figure 69 Radio device allocation for data concatenation evaluation

Figure 69 shows the concepts of those metric employments. For the preferable route to the destination R1, a device E simply chooses the direct link though degraded in RSW direction. However, in RSW+ direction the device E can avoid such degraded links by considering RSSI characteristic on each link.

The experimental results are shown in Table 8 and Figure 70. The results show that suitable metric employment could improve the total sending time and frame success rate performance.

Table 8 Experimental result for metric employment (1)

Device	Total sending time (ms)		Frame success rate	
	RSW	RSW+	RSW	RSW+
C	11606.4	19123.2	99.4%	99.6%
E	9043.2	9561.6	94.2%	99.6%

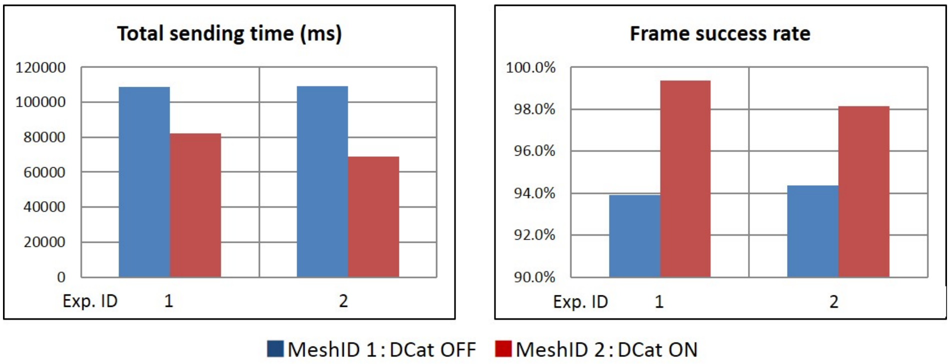


Figure 70 Experimental results for data concatenation evaluation (2)

Low energy access management study for the ultra-low-energy operation network

This section reports on NICT's activities related to the low-energy access management study for the ultra-low-energy operation network. Figure 71 shows an operating example of the low-energy MAC method [79]-[83] applied in this research and development. This method was proposed by NICT, and uses the low-energy superframe structure defined in IEEE 802.15.4e [84]. The superframe is a time period forming the basis of the Time Division Multiple Access (TDMA) control, and is specified by a cyclic beacon signal. The beacon interval (BI), a basic time unit, is divided into the active period (AP) -- sending and receiving, or standby -- and the sleep period (SP) which turns off the power and does not execute the sending and receiving, and the active period is operated as the real communication period. On the other hand, the sleep period allows each device to go into sleep. For the low-energy action, the beacon signal is not sent to every superframe, and essentially, is sent only if temporal synchronization is required. In addition, as shown in the figure, as the active period is sometimes shorter than the data frame, the end of the data frame shall not be limited within the active period and should be before the starting point of the next superframe. As a result, reduction of the active period becomes possible, regardless of the data frame length. In other words, because the reception status for the start of the standby and the sending and receiving of the data frame at each terminal is executed only in the active period and the terminals other than the terminals involved in the sending and receiving of the data frame continuing from the active period can go into sleep during the sleep period, reduction of power consumption can be realized.

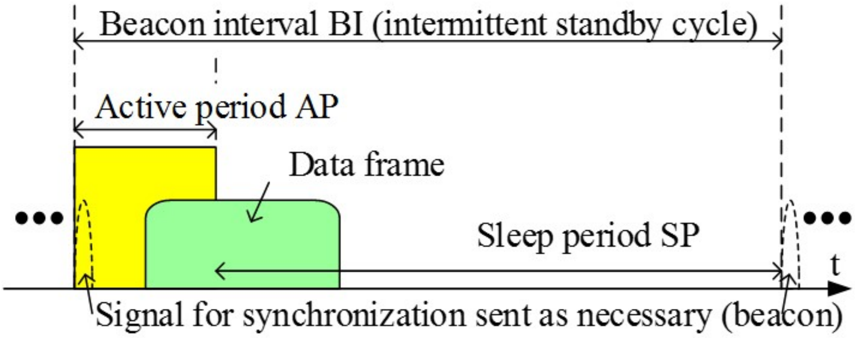

Beacon interval BI (intermittent standby cycle)
Active period AP
Data frame
Sleep period SP
t
Signal for synchronization sent as necessary (beacon)

Figure 71 Low-energy MAC method

Figure 72 shows an operating example which implements the multi-hop communications by using the low-energy MAC method described in the previous section on the topology of the personal area network (PAN) which is the premise for SUN. PAN is the radio device network specified in the IEEE 802.15.4 standard [75], and is composed of a radio device called a PAN coordinator, which is first activated to declare PAN establishment, and a radio device which executes association, a subscribing procedure to PAN. Paying attention to the relationship between the radio device which executes association and the radio device which receives association, we can find the tree-structured PAN topology whose root is the PAN coordinator. We propose a multi-hop communication configuration which collects data from other devices using this tree structure to the PAN coordinator as a collection station (CS). In Figure 72, it can be confirmed that when the D3 or D4 sends the data frames to D1, the sending conforms to the superframe specified in D1, and when D1 sends them to CS, the sending conforms to the superframe specified in CS.

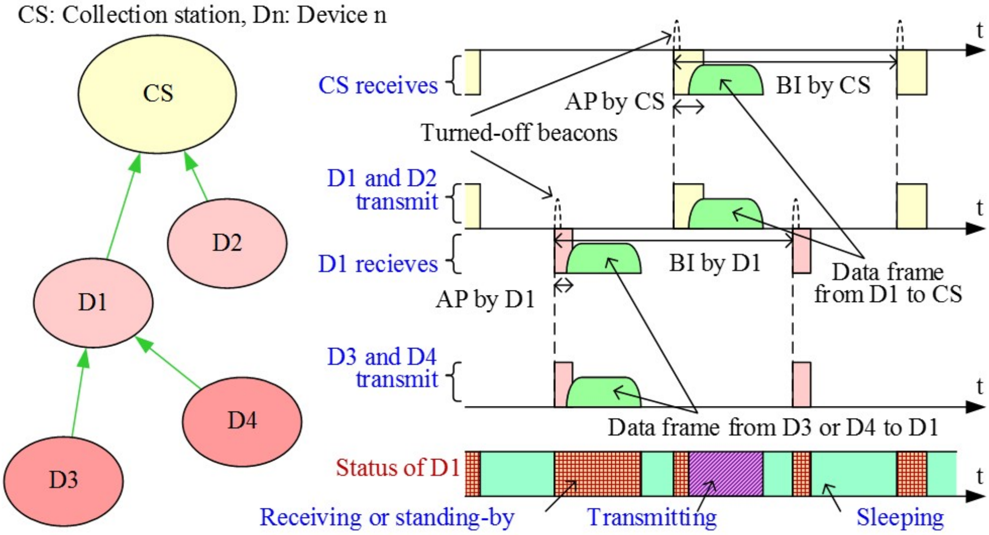


Figure 72 Assumed multi-hop communications

Figure 73 shows the estimated result of battery life when assuming the low-energy multihop communications as referred to above. The assumed battery capacity is three AA batteries. The calculated result shows that if the beacon interval is 80 seconds, data length is 100 octets, and the data generation interval is one minute or so, the battery life can be expected to be 10 years or more.

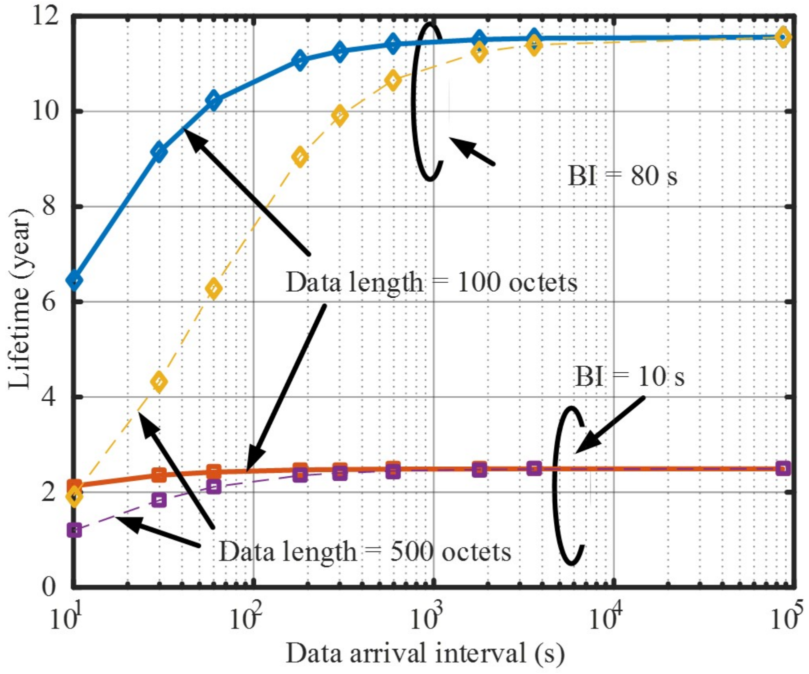


Figure 73 Estimated result of battery life

NICT has conducted the consideration on the mobility supporting capability of such ultra-low-energy operation network. Figure 74 show the concept of radio device mobility in the network. NICT has proposed a mobility supporting capability that assumes a control scheme of dynamic synchronization signal frequency according to the mobility conditions of radio devices in the network.

To evaluate the proposed mobility supporting capability on the ultra-low-energy operation network, NICT has conducted the experimental evaluation. Figure 75 shows the employed low-energy radio module, and specifications of the module are shown in Table 9, which is also referred by the previous estimation as in Figure 73.

Table 9 Specifications of the low-energy radio module

Parameters	Values
Size	20 mm x 40 mm x 3 mm
Weight	4 g
Frequency band	920.6-928.0 MHz
Transmission power	20 mW
Modulation scheme	2 GFSK
Data rate	50, 100, 200 kb/s
Consumption current	Active state: 50 mA Sleep state: 30 μA
I/O interface	Serial

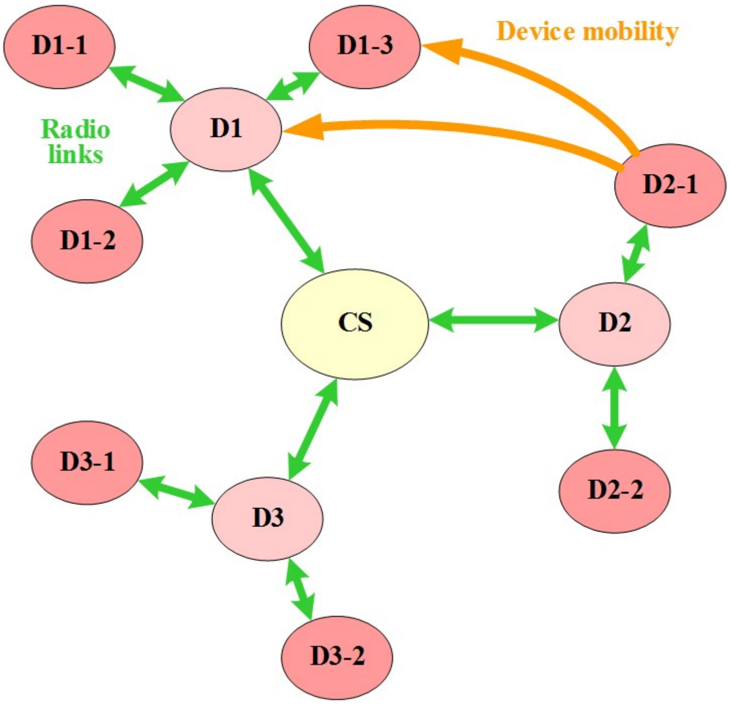


Figure 74 Concept of mobility support in the ultra-low-energy operation network

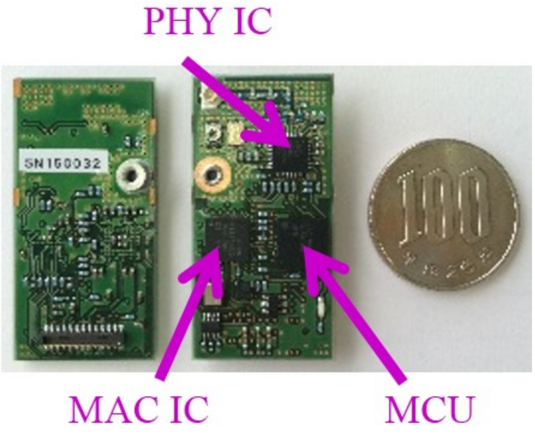


Figure 75 Employed low-energy radio module

In the experiments sensing demonstration by employing the environmental sensor equipping radio device as in Figure 76 is conducted. Specifications of the environmental sensor device are shown in Table 10. Since the device includes the low-energy radio module in Figure 75, all radio specifications are same as those in Table 9. Figure 77 shows an example of measured sensor data.

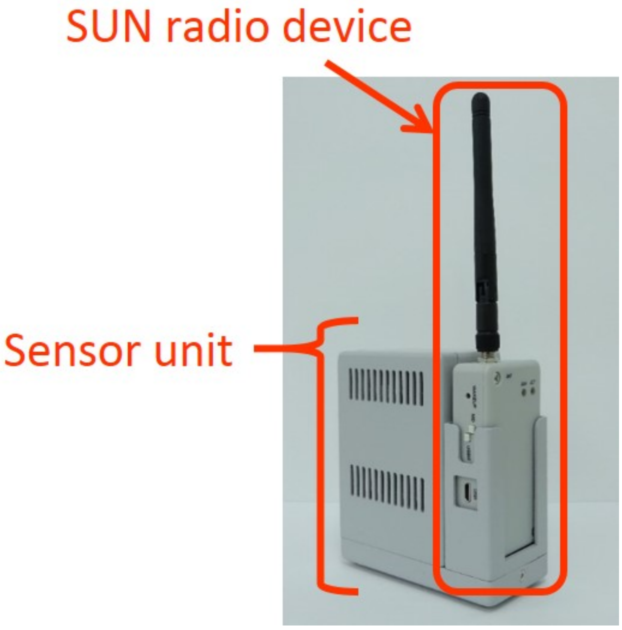


Figure 76 Employed environmental sensor device

Table 10 Specifications of the environmental sensor device

Parameters	Values
Size (w/o antenna)	85mm x 105mm x 40mm
Weight	285g
Sensors	PM2.5, CO2, CO, Air pressure, temperature, humidity

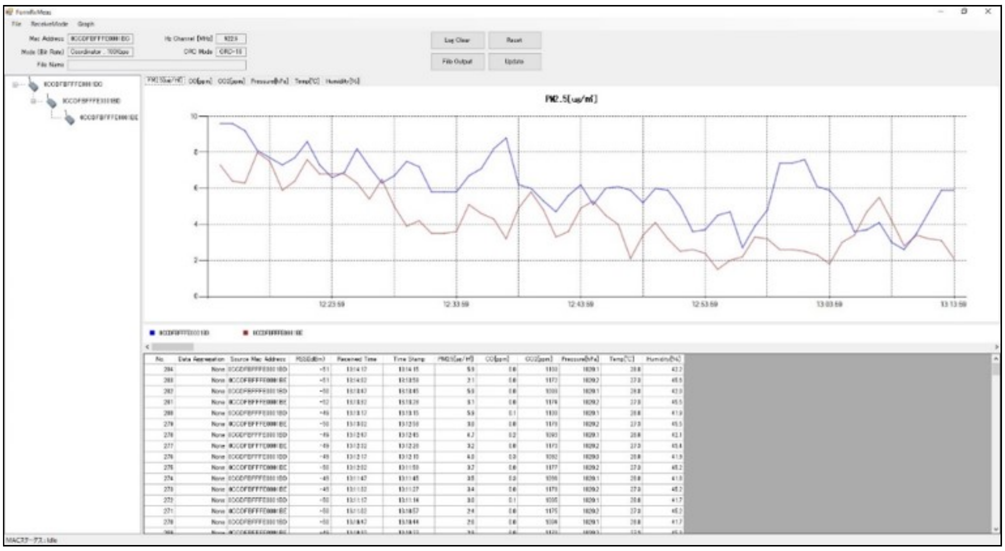
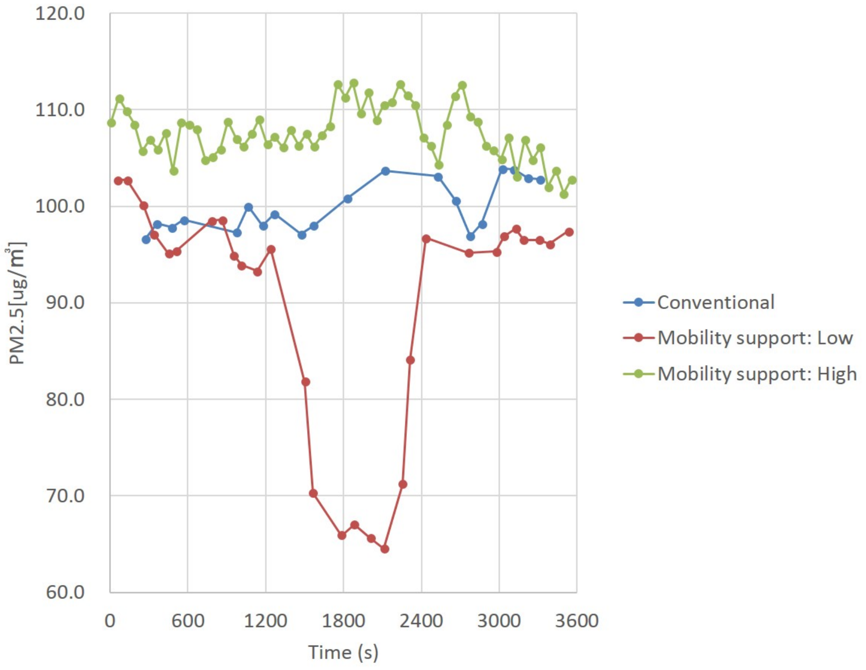
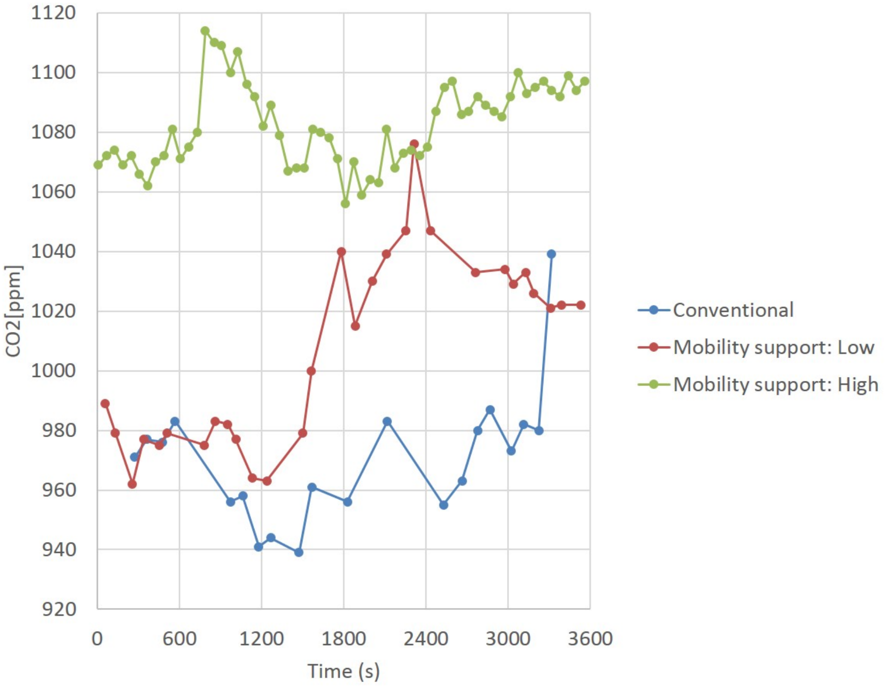


Figure 77 Example of measured sensor data (PM 2.5 case)

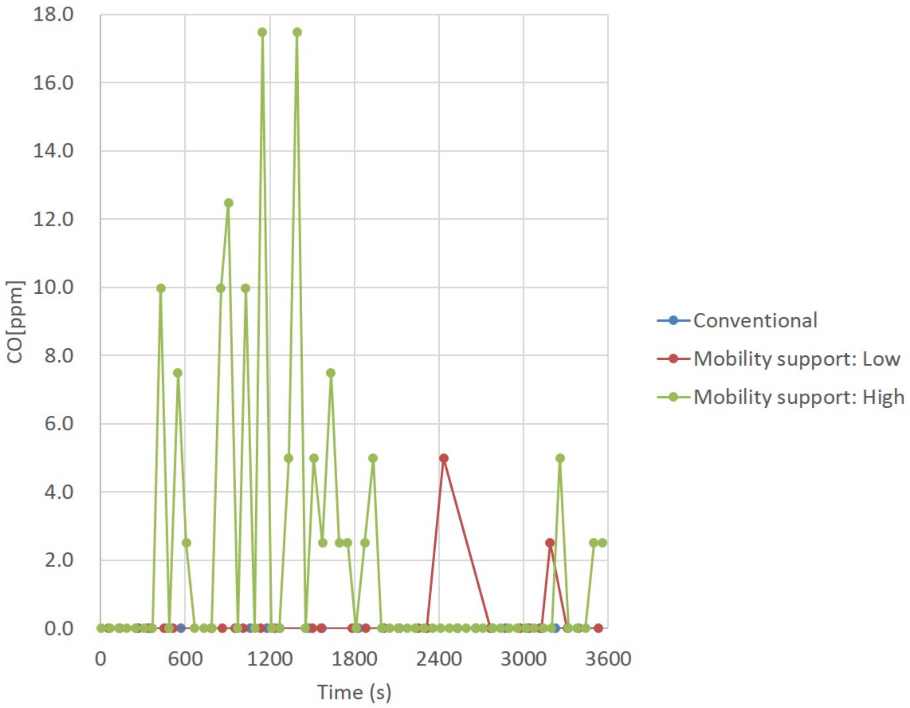
Experimental results are shown in Figure 78. In each graph for the implemented sensors, three conditions of the sensor device that is, conventional, low mobility support and high mobility support are assumed. The Figure 78 confirms that the proposed mobility supporting capability provides more sensor data by realizing early network topology reconstruction according to device movements.



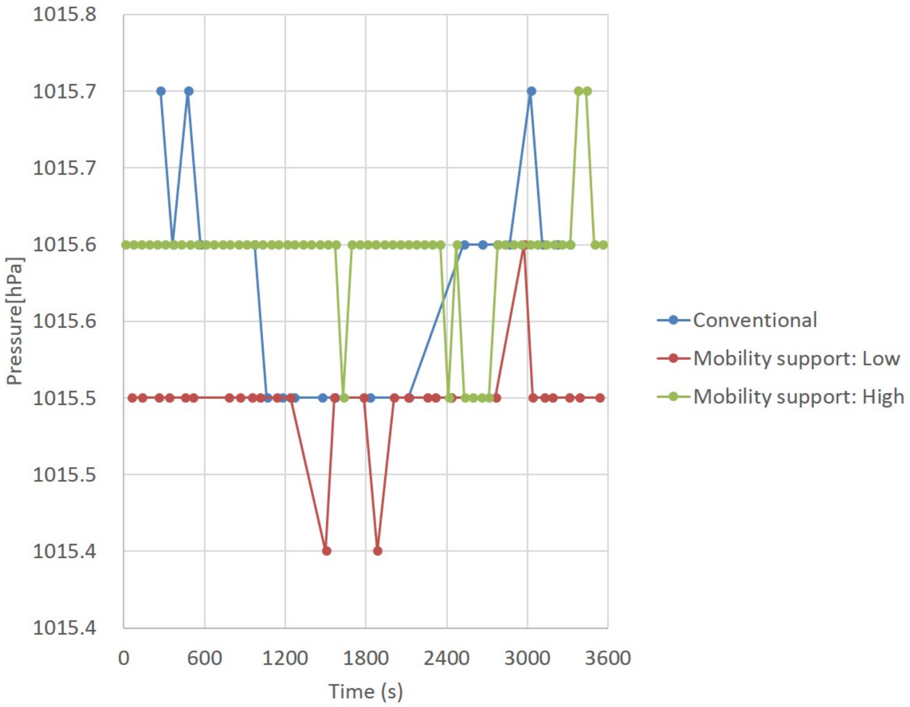
(a) PM 2.5



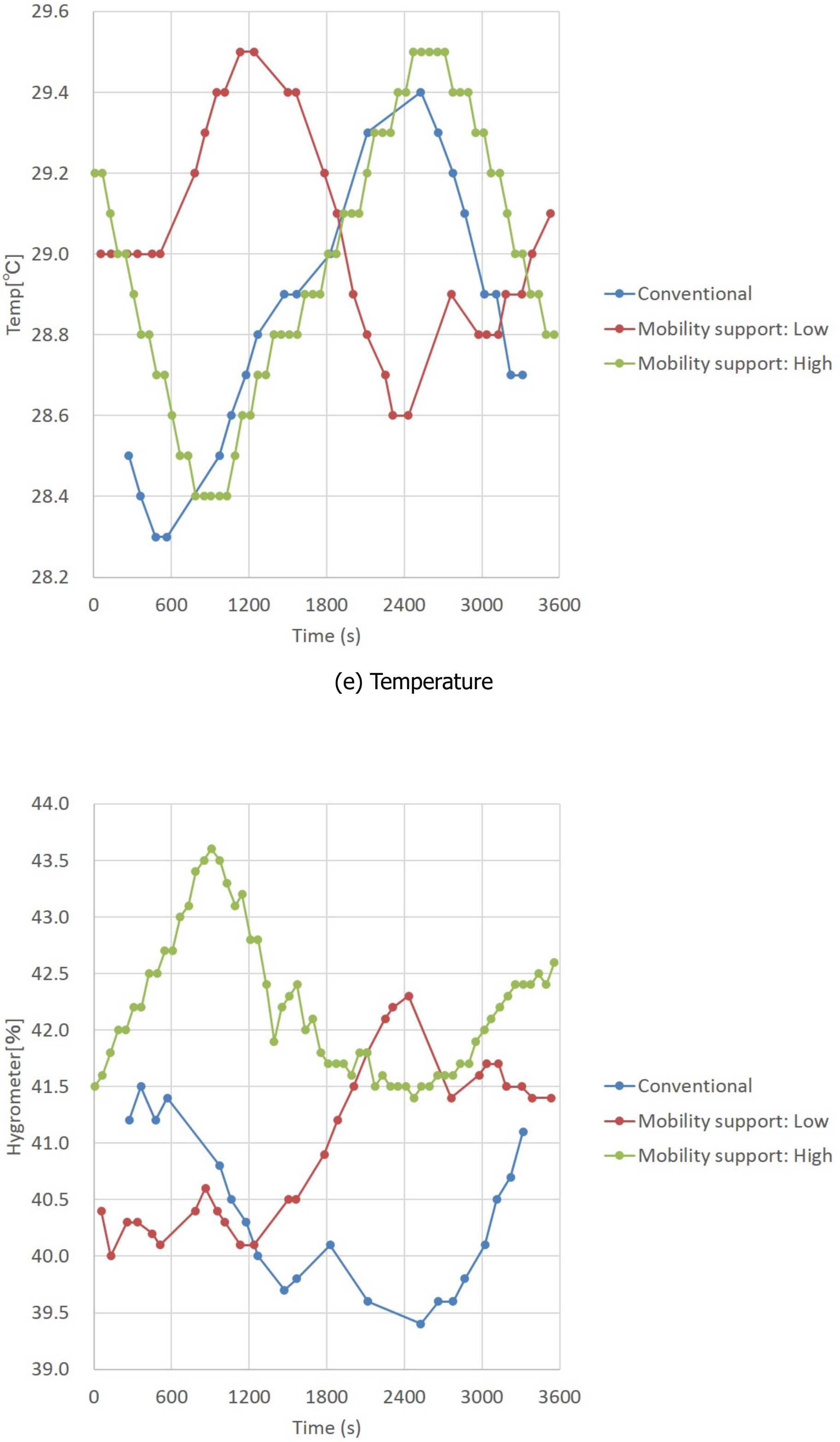
(b) CO2



(c) CO



(d) Air pressure



(e) Temperature

(f) Humidity

Figure 78 Experimental Result

Conclusions

This chapter described the research and development concerning the high-capacity data collection network and ultra-low-energy operation network as operation modes of the wireless grid structures. In the high-capacity data collection network, many terminals relay data in a coordinated manner. This research and development studied the proper building and operation of the radio mesh structure by the application of the L2 R technology, and evaluated the implementation and operation of the autonomous mesh construction function, data concatenation function and multiple mesh supporting functions. On the other hand, the low-energy operation network shows the possible application of the proposed low-energy access management scheme to the lot of networks that also assumes device mobility in the future Mobile-IoT systems.

(2) Leveraged Resources and Participants

Through this project, all project members from NICT, I^{2}R, MIMOS, and HUST have enriched each other through discussions, either from face-to-face meetings or through conference calls. In particular, the following are some of the fruitful collaborations we had throughout the course of the project:

- HUST team has proposed Reinforcement Learning Algorithm and Target Coverage Area. HUST has discussed with I2R, NICT, and MIMOS on Reinforcement Learning Algorithm in WSN 802.15.4 CSMA/CD and on MG's Path Scheduling during kick-off meeting at Singapore January, 2017.
- HUST team and I2R have exchanged papers and ideas through mail and skype sections.
- HUST has proposed a Large Scale Architecture.
- Within the framework of this Mobile IoT Project, the intellectual property of this Large Scale Heterogeneous Mobile IoT Architecture is independently created by HUST team shall be solely reserved by HUST.
- I2R team and HUST team have worked together and exchange ideas on the development of the connectivity optimization and target coverage optimization.
- The proof of concept of the Mobile IoT system testbeds have been developed by the team from MIMOS, NICT, I^{2}R, as well as HUST. Each of these testbeds has its own uniqueness which is catered to a specific scenario or problem being addressed. Through this collaboration, all the project members get to learn from each other on the different techniques used to address certain issues related to Mobile IoT deployment.

Some of the hardware and software platforms that have been used:

- Cooja/Contiki 3.0
- Protocol stack for UDP/RPL/6LoWPAN/TSCH 802.15.4e and TCP/IPv4/802.11
- Z1 Zolertia sensors (Sensor)
- Raspberry Pi3 (Relay Node)
- Iclebo Kobuki
- Java/Linux utilizing TCP/IPv4/802.11.
- Python
- PosgreSQL
- Robot Operating System (ROS)

Broadly, the contributions from each of the member organization can be summarized as follows:

- I^{2}R developed the priority adaptive hierarchical access control for dense sensor deployment in LTE networks.
- I^{2}R has worked together with HUST to propose an algorithm for efficient node deployment to ensure connectivity between the mobile gateway and the information sink.
- I^{2}R has worked together with HUST to propose a node placement algorithm to ensure target coverage while guaranteeing connectivity from the sensor nodes to the mobile gateway.
- HUST has proposed a reinforcement learning algorithm based on TSCH 802.15.4e.
- HUST has proposed this Large Scale Architecture and also modelled the large scale architecture according to 2 IoT protocol stacks (UDP/RPL/IPv6/6LoWPAN/802.15.4e and TCP/IPv4/802.11) and including dynamic features of Mobile IoT system.
- HUST has developed a system testbed for large-scale heterogeneous mobile IoT network, whereby the sensor node is Z1 Zolertia, relay node is Raspberry Pi3, and mobile gateway is a server with unlimited buffer and processing capability.
- I^{2}R has also developed a simple Mobile IoT testbed using mobile robot as the gateway, with several sensor nodes implemented using Raspberry Pi device.
- MIMOS has developed a system testbed IoT solution for environment monitoring using LoRaWAN technology.
- NICT has a role of leader of promotion activities and developed a system testbed of Mobile surveillance technology and Wireless-grid technology.

(3) Findings and Outcomes

Some findings that we obtained from this collaborative work are:

- It is possible to provide dynamic priority assignment for the group paging process in LTE. And this can be achieved using only minimum modification to the standard.
- In ensuring connectivity between sensor nodes to the mobile gateway, and from the mobile gateway to the information sink, a proper placement method of the relay nodes is important to keep the number of deployed sensor minimum. In this work, we have proposed a method to efficiently allocate nodes for this purpose.
- We find that Mobile IoT System that collects data from sensors to Mobile Gateway needs to assure low end-to-end delay in order to reply to mobility requirement of Mobile Gateway.
- A transmission scheduling algorithm based on Reinforcement Learning (RL) technique for WSN's 802.15.4e is useful for multi-channel communication protocol. More concretely, we apply an RL-based algorithm to utilize the scheduling part in TSCH feature of 802.15.4e standard.
- A data collection mechanism that plans the movements of MGs in a RNs network is needed such that the end-to-end packet delay can be minimized. More specifically, we propose two algorithms, one is Maximum Weight Sum First (MWSF) in case of one MG and the other is K-Means clustering combined with Simulating Annealing in case of many MGs.
- We have built this whole model with IoT protocol stacks in Contiki and Java programming language and Linux operating system. We also performed extensive simulations for evaluating the trade-off between end-to-end delay and packet-loss of the model.
- We have developed the end-to-end solution for air monitoring with LoRaWAN technology.

Throughout the project, we have developed the eco-system for LoRaWAN system, which consisted of the hybrid gateway. The hybrid gateway can allow the data to be forwarded to external cloud via internet or to be stored or viewed locally (without the internet connection).

- An application of air monitoring solution to monitor the air quality has been developed, tested and installed to two different locations separated at two different locations in Selangor. The sensor node platform is able to be connected to various types of air quality sensors. In this project, we choose to connect the haze sensor and temperature sensors.
- In the mobile surveillance technology promotion, NICT has shown that the proposed system using battery powered device/solar panels is essential to combat various security threats for industrial, commercial and military areas, and has also shown that a good mobile surveillance should be rapidly deployed, installed and can be operated easily.
- In the wireless grid technologies, the high-capacity data collection network enables data relaying among many terminals in a coordinated manner. This research and development studied the proper building and operation of the radio mesh structure by the application of the L2R technology, and evaluated the implementation and operation of the autonomous mesh construction function, data concatenation function and multiple mesh supporting functions. On the other hand, the low-energy operation network shows the possible application of the proposed low-energy access management scheme to the lot of networks that also assumes device mobility in the future Mobile-IoT systems.

Some outcomes from this collaborative work are:

- We have come up with a novel scheme which can be used in LTE system to support dynamic priority assignment to different group in a group paging process. The achieved performances in terms of successful access probability, average access delay, and collision probability have also been derived and analyzed.
- An algorithm for node placement to ensure target coverage and connectivity (both from the sensor nodes to mobile gateway and from the mobile gateway to the information sink) has been developed. The performance of the algorithm in minimizing the required number of nodes has also been demonstrated via simulation.
- Simulative performance evaluation of RL algorithm has been implemented for TSCH 802.15.4e by utilizing Cooja Contiki 3.0.
- The implementation of this reinforcement learning algorithm has also been demonstrated in a real Testbed.
- We have also evaluated the performance of this Large Scale Architecture based on Mobile Robot Technology via simulation.
- Real implementation of the environment monitoring using LoRaWAN long range transmission technology has been developed, and the web based visualization for the measurement data has also been shown.
- System testbed of mobile surveillance technology and the wireless grid technology has been successfully demonstrated, and the technologies developed along with their advantages have been clearly shown.
- On top of the technical merits and the system testbeds above, we have also published our findings to the international conferences and journals.

As for the potential impact to ICT field or field of application, according to our opinion, this Large Scale Architecture based on Mobile IoT Robot Technology can also be implemented for Hydroponic Greenhouses:

- Where there are different scaffolds. Each scaffold is utilized for planting one type of crop and requires its own way of collecting data. Each scaffold can be represented by a small

WSN where different sensors are used for collecting data and transmitting to a RN. That means different WSNs will generate different traffic patterns. RL algorithm can be implemented within each WSN for providing low delay and reliability.

- A Mobile Robot can be implemented in this Hydroponic Greenhouse for collecting data, capturing images of scaffolds for further data analysis processes. If this mobile robot is implemented for moving on the ground of this Hydroponic Greenhouse, path scheduling algorithms of this Large Scale Architecture can be implemented for this Mobile Robot. However, the implementation of a mobile robot moving around Hydroponic Greenhouse only for collecting data and capturing images is very challenging and expensive.

(4) Broader Impact

Some potential social benefits of the technologies developed include:

- The results of this Large Scale Heterogeneous Mobile IoT Architecture such as traffic-aware communication protocol, optimized path scheduling algorithm for mobile robot, etc. can be used for further research works of institutions in developing countries in Vietnam and Asia: i) A multi-scale adaptive wireless sensor network that adapts to application's traffic and additional traffic. ii) An adaptive large scale mobile IoT system that can control delay levels adaptively based on the speed of Mobile Gateway for achieving the real-time requirement of MGs.
- Other projects within ASEAN-IVO focus on the topics of smart farming, smart aquaculture and smart city can use result of this project, since reliable and real-time communication protocols are the fundamental issues for IoT-based smart system in any field.
- It is also possible to use the proposed framework for generating sustainable practices for the community. Finally, it can enable new collaboration with other partners for relevant activities on IoT-based smart agriculture system.
- The air quality monitoring IoT solution is very useful to understand about the health status of the air. As discussed in the previous section, some of the components in the air are harmful to our health. Air quality monitoring presents an opportunity for operators to create new value propositions, products and services targeted towards a number of different customer segments including the commuters.
- The mobile surveillance technology can be developed further to be commercialized and to provide a fast deployed mobile surveillance solution for various purposes from monitoring natural disaster areas to security areas in a cost effective manner.
- The wireless grid technologies show that current SUN topology can be effectively diversified into at least three directions of high-capacity data collection network, ultra-low-energy operation network and reinforced mesh network. All of them could drastically improve the IoT system performance in the future Mobile IoT systems

(5) Future Developments

The outlook for the future of the technology developed in this project is as follows:

- The developed technology on Mobile IoT can be applied in broad range of applications, such as hydroponic green house as follows:
  - A greenhouse network consists of different scaffolds, each is represented by a WSN can be divided into different WSNs, each is represented by a RN.
  - Each scaffold generates its own traffic pattern based on the type of this scaffold's

crop so at each scaffold's WSN, Reinforcement Learning algorithm can be implemented for adapting to the traffic pattern of this WSN while maintaining low packet delivery rate and delay.

  - A Mobile Robot can be implemented as mechanical gateway for collecting data from different RNs.
  - Path scheduling algorithm of this Large Scale Architecture cannot be implemented for hydroponic greenhouse and aquaculture.
- The system testbeds developed can be scaled up to include more number of sensor nodes and mobile gateway. Further analysis on the data obtained in such a large scale deployment can then be further studied.
- The air monitoring IoT system solution can be deployed in the collaborators countries, namely as the HUST and I2R. The data can be collected and performed the cross analysis to determine the similarities and the anomalies. Through the integrated system, we are able to understand further on the challenges as well as the advantages of this solution.
- Further enhancement to the mobile surveillance technology using channel aggregation is proposed to increase its capacity and reliability. Simulation results show that channel aggregation can improve system throughput and end-to-end delay.
- Moreover, further detailed functions with specifications also from certification and promotion point of view are considered as future enhancements for the wireless grid technologies.

To spread the results from this project to a wider audience, we have submitted the work to the international conferences and journals. Also, based on this Large Scale Heterogeneous Mobile IoT Architecture within the framework of this Mobile IoT Project and on the result of other IVO project (Open Innovation Platform), HUST team has continued to submit a 2018 IVO Project named Scalable Distributed IoT Framework based on Mobile Robot Technology for High Performance Greenhouse Plants. This project has been accepted by Steering Committee during the meeting of March 2018.

iii) Social Contribution

1) Academic papers generated from this project:
   - N. T. Hanh, N. T. Hai, L. Q. Tung, H. T. T. Binh, and E. Kurniawan, "Connectivity Optimization Problem in Vehicular Mobile Wireless Sensor Networks," IEEE 2016 International Conference on Computational Intelligence and Cybernetics (CYBERNETICSCOM 2016), In Proceedings, Aceh, Indonesia, Nov. 2016.
   - E. Kurniawan, T. P. Hui, K. Adachi, and S. Sun, "Hybrid Group Paging for Massive Machine-Type Communications in LTE Networks," 2017 IEEE Global Communications Conference (Globecom 2017), Accepted for publication, Singapore, Dec. 2017.
   - N. T. Hanh, P. L. Nguyen, P. T. Tuyen, H. T. T. Binh, E. Kurniawan, and Y. Ji, "Node Placement for Target Coverage and Network Connectivity in WSNs with Multiple Sinks," 2018 IEEE Consumer Communications and Networking Conference (CCNC 2018), Accepted for publication, Las Vegas, USA, Jan. 2018.
   - F. Kojima, "Wireless Network Customization Technologies in NICT, " NECTEC-NICT Joint Workshop 2017, Sep. 2017
   - F. Kojima, "Wireless Network Customization Technologies in NICT," ASEAN IVO Forum, Nov. 2017

- F. Kojima, "Low-Energy Operation Management Scheme Using Superframe Modification for Wireless Grid Network Structures, " Proc. Of WPMC 2017, Dec. 2017
- F. Kojima, T. Nakaya, A. Namihira and H. Taruya, "Intelligent Water Management System with Wireless-Grid Technology for the ICT Supported Future Agriculture Field," Proc. Of WPMC 2017, Dec. 2017
- Nordin et.al, "Prospects of Education Environment for a Smart Youth City in Malaysia Context", submitted to ACM magazine.
- Hung Nguyen-Duy, Thu Ngo-Quynh, Binh Huynh-Thanh, Giang Nguyen-Linh, E. Kurniawan and other authors, Large Scale Heterogeneous Mobile IoT Architecture, submitted to SCI Journal.

2) Report for international standardization.
   - Wi-SUN global certification standards for "Resource Limited Monitoring and Management (RLMM)"

3) Patent (international or domestic)
   - 3760 20170002009, Method to transmit and receive using HAP with multi-radio and multi-antenna system, 20 Dec 2017.
   - 3766 20180002031, System and method to manage multiple wireless interfaces in moving gateway, 21 Dec 2017.

4) Exhibition of the application or system the project developed
   - "Wireless network customization technologies for IoT society -Large-area/Low-energy capability for the wireless grid according to the applications-," WTP2017, May 2017.
   - F. Kojima, "WSL activities on the system configurations that realize the telestrial wireless network," WTP2017, May 2017.
   - F. Kojima, "NICT's activities on R&D, standardization and promotion of Wi-SUN and Wireless-Grid technologies," APT Training Course, Nov. 2017.
   - "NICT's R&D and Promotion Activities on SUN/Wireless-grid Systems," CSTB Telecom & Media Exhibition and Forum, Jan. 2018.
   - "Wireless Network Technologies in NICT," 5GMF Asia caravan, Feb. 2018.

5) Contributions to NICT's other projects such as:
   - Big Data Integration Research Center (Director General: Koji Zettsu):
     - Real Space Information Analytics Project : Data-thon Event for Environment x Health Smart IoT(http://www2.nict.go.jp/bidal/en/index.html).
     - NEXT Project: Open Innovation Platform for Environmental Smart IoT.

6) Other social contributions:
   - Through the air quality monitoring IoT solution, we can offer the additional information for the public on the air that they breathe. Therefore, the people can do their daily planning and handle their activities in much more healthier way.

Bibliography

[1] "RAN improvement for machine-type communications," Third-Gen. Part. Proj., Sophia-Antipolis, France, 3GPP TR 37.868, Aug. 2011.

[2] S. Y. Lien, T. H. Liau, C. Y. Kao, and K. C. Chen, "Cooperative Access Class Barring for Machine-to-Machine Communications," IEEE Trans. on Wireless Commun., vol. 11, no. 1, pp. 27-32, Jan. 2012.

[3] "MTC Simulation Results with Specific Solutions," Third-Generation Partnership Project, Madrid, Spain, 3GPP TSG R2-104662, Aug. 2010.

[4] W. Yang, A. Fapojuwo, and E. Egbogah, "Performance Analysis and Parameter Optimization of Random Access Backoff Algorithm in LTE," IEEE Veh. Tech. Conf. Fall 2012, pp. 1-5, Quebec, Canada, Sept. 2012.

[5] "Pull Based RAN Overload Control," Third-Generation Partnership Project, Madrid, Spain, 3GPP R2-104870, Aug. 2010.

[6] R. G. Cheng, F. M. Al-Taee, J. Chen, and C. H. Wei, "A Dynamic Resource Allocation Scheme for Group Paging in LTE-Advanced Networks," IEEE Internet of Things Journal, vol. 2, no. 5, pp. 427-434, May 2015.

[7] C. H. Wei, R. G. Cheng, and F. M. Al-Taee, "Dynamic Radio Resource Allocation for Group Paging Supporting Smart Meter Communications," 2012 IEEE Int. Conf. on Smart Grid Commun. (SmartGridComm), pp. 659-663, Tainan, Taiwan, Nov. 2012.

[8] O. Arouk, A. Ksentini, Y. Hadjadj-Aoul, and T. Taleb, "On Improving the Group Paging Method for Machine-Type-Communications," IEEE Int. Conf. on Commun. 2014, pp. 484-489, Sydney, Australia, June 2014.

[9] O. Arouk, A. Ksentini, and T. Taleb, "Group Paging Optimization for Machine-Type-Communications," IEEE Int. Conf. on Commun. 2015, pp. 6500-6505, London, UK, June 2015.

[10] R. Harwahyu, R. G. Cheng, and R. F. Sari, "Consecutive Group Paging for LTE Networks Supporting Machine-type Communications Services," 2013 IEEE 24th Annual Int. Symp. on Personal, Indoor, and Mobile Radio Commun. (PIMRC), pp. 1619-1623, London, UK, Sept. 2013.

[11] C. T. Chang and S. R. Yang, "Modeling LTE group paging mechanism for Machine-Type Communications," 2015 Int. Wireless Commun. and Mobile Comp. Conf., pp. 1295-1300, Dubrovnik, Croatia, Aug. 2015.

[12] "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification (Release 14)," Third-Gen. Part. Proj., Sophia-Antipolis, France, 3GPP TS 36.331, Dec. 2016.

[13] "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation (Release 13)," Third-Generation Partnership Project, Sophia-Antipolis Cedex, France, 3GPP TS 36.211, Dec. 2015.

[14] C. H. Wei, R. G. Cheng, and S. L. Tsao, "Performance Analysis of Group Paging for Machine-Type Communications in LTE Networks," IEEE Trans. on Veh. Tech., vol. 62, no. 7, pp. 3371-3382, Sept. 2013.

[15] C. H. Wei, R. G. Cheng, and S. L. Tsao, "Modeling and Estimation of One-Shot Random Access for Finite-User Multichannel Slotted ALOHA Systems," IEEE Commun. Lett., vol. 16, no. 8, pp. 1196-1199, Aug. 2012.

[16] P. Zhou, H. Hu, H. Wang, and H. H. Chen, "An Efficient Random Access Scheme for OFDMA Systems with Implicit Message Transmission," IEEE Trans. on Wireless Commun., vol. 7, no. 7, pp. 2790-2797, July 2008.

[17] R. G. Cheng, C. H. Wei, S. L. Tsao, and F. C. Ren, "RACH Collision Probability for Machine-Type Communications," IEEE Veh. Tech. Conf. Spring 2012, pp. 1-5, Yokohama,

japan, May 2012.

[18] J. Reich, V. Misra, D. Rubenstein, and G. Zussman, "Connectivity Maintenance in Mobile Wireless Networks via Constrained Mobility," IEEE JSAC Special Issue: Communications Challenges and Dynamics for Unmanned Autonomous Vehicles, vol. 30, no. 5, pp. 935–950, June 2012.

[19] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," Elsevier Journal on Computer Communications, vol. 30, no. 14-15, pp. 2826–2841, Oct. 2007.

[20] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms. 2nd Edition," MIT Press, Cambridge, 2001.

[21] J. Rezazadeh, M. Moradi, and A. S. Ismail, "Mobile Wireless Sensor Networks Overview," International Journal of Computer Communications and Networks (IJCCN), vol. 2, no. 1, pp. 17-22, Feb. 2012.

[22] M. A. Kafi, Y. Challal, D. Djenouri, M. Doudou, A. Bouabdallah, and N. Badache, "A Study of Wireless Sensor Networks for Urban Traffic Monitoring: Applications and Architectures," International Conference on Ambient Systems, Networks and Technologies, pp. 617-626, Canada, June 2013.

[23] U. Lee and M. Gerla, "A survey of urban vehicular sensing platforms," Elsevier Journal on Computer Networks, vol. 54, no. 4, pp. 527–544, March 2010.

[24] U. Lee, E. Magistretti, M. Gerla, P. Bellavista, and A. Corradi, "Dissemination and Harvesting of Urban Data Using Vehicular Sensing Platforms. IEEE Transactions on Vehicular Technology 58(2). (2009), 882 – 901

[25] https://sites.google.com/site/dataclusteringalgorithms/k-means-clustering-algorithm

[26] P. Wang, H. How, X. K. He, C. Wang, T. Xu, and Y. Li, "Survey on Application of Wireless Sensor Network in Smart Gird," in *WTISG*, pp 1212-1217, May 2015.

[27] S. Yinbiao, *Internet of Things: Wireless Sensor Networks*, International Electrotechnical Commission, 2014.

[28] R. Devika, B. Santhi, T. Sivasubramanian, "Increase the Lifetime of WSN by Preventing Sink Isolation using Supercluster Formation," Indian Journal of Science and Technology, Vol 7(S4), pp. 92–98, April 2014.

[29] C. Tunca, S. Isik, M. Y. Donmez, and C. Ersoy, "Distributed Mobile Sink Routing for Wireless Sensor Networks: A Survey," *IEEE Communications Surveys & Tutorials*, Vol. 16, No. 2, pp. 877-897, June 2014.

[30] S. Ghafoor, M. H. Rehmani, S. Cho, and S. H. Park, "An Efficient Trajectory Design for Mobile Sink in a Wireless Sensor Network," Computer and Electrical Engineering, vol. 40, pp. 2089-2100, Oct. 2014.

[31] A. Kinalis, S. Nikoletseas, D. Patroumpa, and J. Rolim, "Biased Sink Mobility with Adaptive Stop Times for Low Latency Data Collection in Sensor Networks," in GLOBECOM - IEEE Global Telecommunications Conference, Dec. 2009.

[32] M. Marta and M. Cardei, "Improved sensor network lifetime with multiple mobile sinks," in Pervasive and Mobile Computing 5(2009), pp. 542-555, Oct. 2009.

[33] S. Gao, H. Zhang, and S. K. Das, "Efficient Data Collection in Wireless Sensor Networks with Path-Constrained Mobile Sinks," *IEEE Transactions on Mobile Computing*, Vol. 10, No. 5, pp. 592-608, April 2011.

[34] M. Zhao, M. Ma, and Y. Yang, "Efficient Data Gathering with Mobile Collectors and Space-Division Multiple Access Technique in Wireless Sensor Networks," *IEEE Transactions on Computers,* Vol. 60, No.3, pp. 400-417, March 2011.

[35] S. Abirami and S. Thilagavathi, "Mobile Sink Based Coverage and Connectivity to Maximize Network Lifetime of Wireless," *International Journal of Research in Science and Technology*, vol. 6, no. 3, pp. 212-223, July 2016.

[36] Q. Zhao and M. Gurusamy, "Lifetime Maximization for Connected Target Coverage in Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol.16, no. 6, pp. 1378–1391, Dec. 2008.

[37] N. T. Hanh, P. T. H. Hanh, H. T. T. Binh, and N. D. Nghia, "Heuristic Algorithm for Target Coverage with Connectivity Fault – tolerance Problem in Wireless Sensor Networks," in *Conference on Technologies and Applications of Artificial Intelligence*, Taiwan, pp. 235-240, Nov. 2016.

[38] N. T. Hanh, N. T. Hai, L. Q. Tung, H. T. T. Binh, and E. Kurniawan, "Connectivity Optimization Problem in Vehicular Mobile Wireless Sensor Networks," in *International Conference on Computational Intelligence and Cybernetics*, Indonesia, pp. 55-61, Nov. 2016.

[39] D. Agrawal and Q.-A. Zeng, *Introduction to wireless and mobile systems*, Cengage Learning, 2011.

[40] J. Du, H. Liu, L. Shangguan, L. Mai, K. Wang, and S. Li, "Rendezvous Data Collection Using a Mobile Element in Heterogeneous Sensor Networks," Hindawi International Journal of Distributed Sensor Networks, vol. 2012, no. 7, pp. 1-10, May 2012.

[41] G. Xing, T. Wang, W. Jia, and M. Li, "Rendezvous design algorithms for wireless sensor networks with a mobile base station," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08)*, pp. 231–239, May 2008.

[42] G. Xing, T. Wang, Z. Xie, and W. Jia, "Rendezvous planning in mobility-assisted wireless sensor networks," in *Proceedings of the 28th IEEE International Real-Time Systems Symposium (RTSS '07)*, pp. 311–320, Dec. 2007.

[43] Y. Tirta, Z. Li, Y. H. Lu, and S. Bagchi, "Efficient collection of sensor data in remote fields using mobile collectors," in *Proceedings of the 13th International Conference on Computer Communications and Networks (ICCCN '04)*, pp. 515–519, Oct. 2004.

[44] IEEE 802.15.4 std., available online at http://ieeexplore.ieee.org/document/7460875/

[45] Zigbee Specification on Zigbee platform, available online at http://www.zigbee.org/wp-content/uploads/2014/11/docs-05-3474-20-0csg-zigbee-specification.pdf

[46] Bluetooth Technology, Core Specification, available online at https://www.bluetooth.com/specifications/bluetooth-core-specification

[47] WirelessHART standard developed by HART Communication Foundation, IEC 62591, IEEE 802.15.4

[48] ISA-100 Wireless Compliance Institute, official website http://isa100wci.org/

[49] IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN), available online at https://tools.ietf.org/html/rfc4919

[50] Routing Protocol for Low Power and Lossy Network (RPL), available online at https://tools.ietf.org/html/rfc6550

[51] The Constrained Application Protocol (CoAP), available online at https://tools.ietf.org/search/rfc7252

[52] William Wallace II, "Limitations of IEEE 802.15.4 based wireless mesh networks for wireless localization" in master thesis at the University of Texas at Arlington.

[53] F. Chen, R. German, and F. Dressler, "Towards IEEE 802.15.4e: A study of performance," *Pervasive Computing and Communications Workshops*, pp. 68-73, Mannheim, Germany, April 2010.

[54] D. De Guglielmo, S. Brienza, and G. Anastasi, "802.15.4e: A Survey," *Elsevier Journal on Computer Communications,* vol. 88, no. 1, pp. 1-24, Aug. 2016.

[55] Using IEEE 802.15.4e Time-Slotted Channel Hopping (TSCH) in the Internet of Things, available online at https://tools.ietf.org/html/rfc7554

[56] S. Duquennoy, B. Al Nahas, O. Landsiedel, and T. Watteyne, “Orchestra: Robust Mesh Networks Through Autonomously Scheduled TSCH” in *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys 2015)*, Seoul, South Korea, Nov. 2015.

[57] K. H. Phung, B. Lemmens, M. Goossens, A. Nowe, L. Tran, and K. Steenhaut, “Schedule-based multi-channel communication in wireless sensor networks: A complete design and performance evaluation” *Elsevier Journal on Ad Hoc Networks*, vol. 26, no. 1, pp. 88-102, March 2015.

[58] R. M. B. S. Arun and A Somasundara, "Mobile Agent Scheduling for Efficient Data Collection in Wireless Sensor Networks with Dynamic Deadlines," in *IEEE International Real-Time Systems Symposium (RTSS)* , Lisbon, Portugal, Dec. 2004.

[59] Y. Gu, D. Bozdag, R. W. Brewer, and E. Ekici, "Data harvesting with mobile elements in wireless sensor networks," *Elsevier Journal on Computer Networks,* vol. 50, no. 17, pp. 3449-3465, Dec. 2006.

[60] A. R. Arun A. Somasundara, "Mobile Element Scheduling with Dynamic Deadlines," *IEEE Transactions on Mobile Computing,* vol. 6, no. 4, pp. 395-410, April 2007.

[61] R. Sugihara and R. K. Gupta, "Path Planning of Data Mules in Sensor Networks," *ACM Transactions on Sensor Networks,* vol. 8, no. 1, pp. 1-27, Aug. 2011.

[62] J. J. Kponyo, Y. Kuang, and Z. Li, "Real Time Status Collection and Dynamic Vehicular Traffic Control Using Ant Colony Optimization," *Intelligent Computer Communications and Processing*, pp. 69-72, Sept. 2012.

[63] D. Puthal, B. Sahoo, and S. Sharma, "Dynamic Model for Efficient Data Collection in Wireless Sensor Networks with Mobile Sink," *International Journal of Computer Science And Technology (IJCST),* vol. 3, no. 1, pp. 623-628, Jan. 2012.

[64] A. Das, A. Mazumder, A. Sen, and N. Mitton, "On Mobile Sensor Data Collection Using Data Mules," *International Workshop on Wireless Sensor, Actuator and Robot Networks - ICNC Workshop*, pp. 1-7, Hawaii, USA, Feb. 2016.

[65] IDA, “Proposed Regulatory Framework for TV white Space Operation in the VHF/UHF Bands,” June 2013.

[66] IDA, “Regulatory Framework for TVWS Operation in the VHF/UHF Band,” June 2014.

[67] T. Matsumura, K. Ibuka, V. D. Hoang, H. Murakami, K. Ishizu, and F. Kojima, “TV White-space Vehicle Communication System in Singapore based on LTE Technology”, IEICE Technical Report, SR2015-56, 2015.

[68] K. Fall and K. Varadhan, Manual. The VINT Project, http://www.isi.edu/nsnam/ns/doc/.

[69] C. Perkins, P. Bhagwat. “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers”, In ACM SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications, pages 234--244, 1994.

[70] G. R. Hiertz, S. Max, E. Wei, L. Berlemann, D. Denteneer, and S. Mangold, “Mesh Technology enabling Ubiquitous Wireless Networks”, WICON’06, The 2nd Annual International Wireless Internet Conference, August 2-5, 2006.

[71] Y. Rachlin, R. Negi, and P. Khosla, "Sensing capacity for discrete sensor network applications," in Conf. Rec. 2005 IEEE Information Processing in Sensor Networks, pp.126–132.

[72] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in Conf. Rec. 2000 Hawaii International Conference on System Sciences, pp.1–10, vol.2.

[73] Q. Li, J. A. Aslam, and D. Rus, "Online power-aware routing in wireless ad-hoc networks," in Conf. Rec. 2001 MOBICOM, pp.97–107.

[74] IEEE802.15.4g, "Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), Amendment 3: Physical Layer (PHY) Specifications for Low-Data-Rate, Wireless, Smart

Metering Utility Networks," 2012.

[75] IEEE802.15.4, "Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPAN)," 2009.

[76] IEEE 802.15.10, "IEEE Recommended Practice for Routing Packets in IEEE 802.15.4 Dynamically Changing Wireless Networks," 2017.

[77] V. Rabarijaona, F. Kojima, H. Harada, and C. Powell, "Enabling Layer 2 Routing in IEEE std 802.15.4 Networks with IEEE Std 802.15.10," IEEE Communications Standards Magazine, vol.1, no.1. pp.44–49, March 2017

[78] V. Rabarijaona, F. Kojima, and H. Harada, "Hierarchical Mesh Routing Implementation for Indoor Data Collection, " IEEE 27th International Symposium on Personal, Indoor and Mobile Radio Communication (PIMRC'16), Valencia, Spain, Sept. 2016

[79] J. L. Hill and D. E. Culler, "Mica: a wireless platform for deeply embedded networks," 2002 IEEE Micro 22, pp. 12-24.

[80] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC protocol for Wireless Sensor Networks," in Conf. Rec. 2002 IEEE Computer and Communications Societies (INFOCOM), pp. 1567-1576.

[81] T. v. Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks," in Conf. Rec. 2003 Embedded Networked Sensor Systems (SenSys), pp. 171-180.

[82] A. El-Hoiydi and J.-D. Decotignie, "WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks," in Conf. Rec. 2004 International Symposium on Computers and Communications, pp. 244-251, vol. 1.

[83] F. Kojima and H. Harada, "Superframe Division Multi-Hop Data Collection withbAggregation on Wi-SUN Profile for ECHONET Lite," Conf. Rec. 2014 IEEE Wireless Communications and Networking Conference Workshops, pp. 116-121

[84] IEEE802.15.4e, "Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), Amendment 1: MAC sublayer," 2012.