

# Using Mobile Elements as Dynamic Bridges in Sparse Wireless Sensor Networks

Dang Hai Hoang<sup>1)</sup>, Thorsten Strufe<sup>2)</sup>, Thieu Nga Pham<sup>3)</sup>,  
Hong Ngoc Hoang<sup>2)</sup>, Anh Son Nguyen<sup>3)</sup>, Immanuel Schweizer<sup>2)</sup>

<sup>1)</sup>Ministry of Information and Communications, <sup>2)</sup>Technical University of Darmstadt (Germany),

<sup>3)</sup>Hanoi University of Civil Engineering.

Email: hdhai@vncert.vn

**Abstract:** Sparse Wireless Sensor Networks using several mobile nodes and a small number of static sensor nodes have been widely used for many applications, especially for traffic-generated pollution monitoring. This paper proposes a method for data collection and forwarding using Mobile Elements (*MEs*), which are moving on predefined trajectories in contrast to previous works that use a mixture of *MEs* and static nodes. In our method, *MEs* can be used as data collector as well as dynamic bridges for data transfer. We design the trajectories in such a way, that they completely cover the deployed area and data will be gradually forwarded from outermost trajectories to the center whenever a pair of *MEs* contacts each other on an overlapping road distance of respective trajectories. The method is based on direction-oriented level and weight assignment. We analyze the contact opportunity for data exchange while *MEs* move. The method has been successfully tested for traffic pollution monitoring in an urban area.

**Keywords –** Sparse WSNs, mobile elements, mobile data collection, opportunistic networks, forwarding protocol.

## I. INTRODUCTION

In recent years, Wireless Sensor networks (*WSNs*) have been widely used for many environmental applications, especially for traffic-generated air pollution monitoring [1,2,3,4,5]. Typical examples of WSN applications for pollution monitoring are DaSense [2,6], Rescatame [3], PermaSense [4], OpenSense [5]. In such applications, a *WSN* is composed of a large number of battery-powered tiny devices (i.e. nodes) equipped with onboard sensors to collect environment data and transmit them to the monitoring center. Nodes can communicate with each other through their low-rate (10-100 kbps) and short-range (usually less than 100m) wireless interfaces. Typical WSN nodes use IEEE 802.15.4 standard

(ZigBee) [7]. The transmission can be either one-hop from a *WSN* node to a sink node, i.e. the base station, or multi-hop using several *WSN* nodes [8].

Most of the traditional *WSNs* consist of static nodes which are densely deployed over a sensing area and mobility is not considered as an option [9]. Such "dense" *WSNs* are not effective to cover a wide environment area. Although we use a large number of sensors, coverage holes (uncovered zones) remain present. This problem may overcome by increasing the number of static nodes (i.e. make a dense *WSN*) or increasing the sensitivity of the sensors (i.e. increasing the sensing range). Both solutions are not suitable due to a higher cost and excessive radio interference [10]. A "sparse" *WSN* would have fewer nodes in the field of interest. Dense *WSNs* are used in situations, where it is very important to monitor every event using multiple sensors. Sparse *WSNs* may be used with regard to the low implementation cost and when we want to cover a large area with a small number of sensors. In sparse *WSNs*, nodes may be out of radio range of others. This kind of partitioned sensor networks is called "disconnected" networks, which are very common in practice, e.g. in traffic-generated pollution and wide area environment monitoring. Many applications require spreading the monitoring network over a relative large area. It is usually not possible to use a large number of nodes to cover the area and the radio range of each node is not sufficient to assure a fully and permanently connected network. Actually, coverage is one of the most attractive research problems in such *WSNs* [10]. On the other hand, intermittent connectivity is a challenging issue for data collection in disconnected *WSNs* [11,12].

Recently, several *WSN* architectures based on

mobility have been proposed in order to improve performance of traditional *WSNs* [9,10,13,14,15,16,17,18]. A number of mobile sensor nodes are usually used in such sparse *WSNs* together with a limited number of static sensor nodes. Mobility is useful for sparse *WSNs*, in which special Mobile Elements (*MEs*) are moving in the area to collect data from fixed sensor nodes. Fixed (static) nodes build up disconnected *WSNs* using multiple isolated groups separately and *MEs* are used as a compliment for fixed *WSN* nodes. *MEs* visit static nodes from time to time in order to collect data from them. *MEs* may be Mobile Data Collector (MDCs) [19], or Data MULEs [20], or Mobile Relays (MRs) [21]. Mobility of *MEs* can be either controlled or uncontrolled, and *MEs* may have deterministic or completely random mobility patterns. When mobility is uncontrolled, sensor nodes can only conform to the way the *MEs* move throughout the network. Otherwise, when mobility is controlled, the *ME* movements can be designed so as to achieve specific goals (see survey in [9,10,14,17]).

However, mobility in *WSNs* brings significant challenges, which do not arise in static *WSNs*. The network becomes highly dynamic, and the topology is thus more unstable, even unpredictable. In such *WSNs*, the presence of a "continuous" path between nodes is not expected. Communication is only possible, when the nodes are in the transmission range of each other. This is what typically is called as "opportunistic" communications [22,23,24], which means the mobile nodes only have *opportunistic contact* during their movement. Such contact may not be frequent. Thus, data collected from isolated nodes should be temporarily stored in the local memory of sensors, and then forwarded to the center relying on opportunistic contact between the nodes, which is occasional. The lack of continuous connectivity gives motivation for *Delay Tolerant Networks (DTN)*, which uses *store and forward* approach.

Using mobility, several critical issues need to be addressed in *WSNs* including: coverage, connectivity, motion control (mobility control), contact detection, routing, reliable data transfer, data collection, data transfer (forwarding). These issues have been extensively studied in recent works (see survey in [9,10,16,17,25,26,27,28]).

This paper presents our solution for data collection and forwarding in sparse *WSNs* using *MEs* as dynamic bridges. Contrary to previous works, we focus the coverage on traffic-generated pollution areas, i.e. only on the main streets of a city similar to the model in our previous work [29]. The coverage is interpreted as how well a *WSN* could monitor a field of interest. Similar to the approach in [28], we consider a transport network using a small number of mobile sensor nodes. The design of node movement strategy for *MEs* needs to address the issue: where to move and how to efficient move *MEs* so that the desired monitoring coverage can be achieved. Typical solutions proposed installing mobile nodes on top of public transport vehicles (e.g. trains, trams, buses) that regularly traverse the city and release collected data whenever they meet the base station (i.e. the monitoring center) [2,5,6,28].

In this paper, we propose a more general approach using predefined trajectories, which have overlapping road distances. Our *MEs* will move on these trajectories and act as sensor nodes, as data collectors for other *MEs*, as well as dynamic bridges for intermediate data transfer. The data forwarding is based on a flexible level assignment to *MEs* and is following the principle: forwarding from *MEs* with higher levels on outermost trajectories to *MEs* with lower levels on inner trajectories. The level of each *ME* is frequently adapted. This method allows a controllable hop-by-hop routing and forwarding in contrast to pre-defined routing and forwarding in previous works. Moreover, we address the issue of intermittent connectivity using an opportunistic contact analysis. To the best of our knowledge, this is the first work addressing the contact opportunity of *MEs* for mobile *WSNs*. Using the proposed method, we developed a protocol for collecting and forwarding data in a mobile *WSN* for traffic pollution monitoring. That are the contributions of this paper.

The rest of this paper is organized as follows. Section II presents related works. Section III gives an overview of our method. Section IV presents an opportunistic contact analysis. Section V describes our proposed protocol. Section VI presents a case study deployment. Finally, Section VII concludes the paper.

## II. RELATED WORKS

In this section, we present related works regarding coverage, connectivity, motion control or mobility control, contact detection, routing, data collection, and data forwarding.

The problem of area coverage with sensors is often considered in past works (see survey in [25]). Most of traditional *WSNs* use deterministic placement of static sensor nodes, which are densely deployed over the sensing area [9]. However, this approach still has problem of coverage holes (uncovered zones) due to the occurrence of node failures. As mobility has been introduced to *WSNs*, a dense *WSN* architecture may not be a requirement [9]. Sparse *WSNs* using mobile nodes and a small number of static nodes become a feasible option, which makes the coverage problem more complex. *MEs* move throughout the *WSN* to collect data coming from all static nodes. Since *MEs* can visit different regions in the networks, the coverage may be improved. Motion planning for *MEs* may help to achieve better area coverage either using random or fixed routes [10]. Different node movement strategies may apply in order to maximize the coverage. Mobile nodes may adjust their positions after initial deployment in order to reduce the coverage holes [30]. Other works proposed to identify coverage holes and to compute the desired positions where sensors should move so as to improve the coverage [10,25,27,31]. Recently, a path planning algorithm has been proposed that allows mobile nodes to autonomously navigate through the field for improving the area coverage [27]. As alternative, our method uses pre-defined trajectories for moving *MEs* in order to cover the desired area [29,32]. From the viewpoint of traffic-generated pollution, only main streets are of interest for the coverage. A similar approach can be found in [28].

Coverage and connectivity are closely related issues. Connectivity implies that the communication range of the sensors is at least twice that of the sensing range [10]. Several protocols attempt to combine coverage and connectivity as described in [10,16,17,25,26]. These protocols are very similar in their nature to address the same problem. Our work is motivated by the concept proposed in [2,29].

Mobility may be either controlled or uncontrolled. Mobility can be characterized by means of trajectory, i.e. the path, on which a *ME* would move. Trajectory control (or motion control) may be static or dynamic. Static trajectory control refers to a path which does not change over time. On the other hand, dynamic trajectory control refers to change of trajectory of the *ME* on-the-fly. For total accessible sensor fields, many works focused on the development of moving trajectories of *MEs* [9,14,16,17]. However, sensor fields are only partly accessible and reachable trajectories are limited in most scenarios. Thus, the motion of *MEs* needs to be controlled with respect to the path and sojourn time of *MEs* in order to improve the performance. In order to dispatch *MEs*, special routes for *MEs* around the deployed area are usually pre-defined.

In opportunistic communication, the connectivity is only intermittent when nodes are in range. Timely *ME* discovery and contact detection is critical in order to effectively exploit the short contact time for data exchange and forwarding. Several works have investigated the issue of opportunistic contact between *MEs* [8,9,12,23,24]. Such contact may not be frequent and contact time is unpredictable in mobile *WSNs* [12]. It is necessary to detect the presence of neighbor nodes correctly and efficiently. This is true when the duration of contact is short [9]. Therefore, we propose in this paper predefined trajectories, which have overlapping road distances in order to increase the contact opportunity. By this design, we can derive a opportunistic contact analysis as described later in Section V. To our best knowledge, this paper is the first work addressing the contact opportunity.

Routing is only possible when a *ME* is in contact with another. In fact, routing in *WSNs* can be considered as the process of data forwarding toward another node, i.e. the selection of the path to the next destination [9]. This process consists of three phases: discovery, routing and data forwarding. Due to the intermittent connectivity, it is not always possible to determine the end-to-end route at the time of data delivery. Routing and forwarding are not possible in the classical way. In contrast, routing and forwarding will be better performed "on-the-fly", while messages are being forwarded. Hop-by-hop routing and

forwarding are desirable. Until now, a number of research works addressed the issue of routing in *WSNs* [9,12,16,17,18,28,33]. Three main categories of routing/forwarding are described in [16] including: data centric, hierarchical and position-based schemes. New schemes has been proposed for disconnected *WSNs* [33] including hybrid protocols and cooperative routing protocols. Several author proposed to combine opportunistic routing and data gathering in one scheme [9]. Other emergent protocols are route planning method [28], Data MULE routing scheme [33], SNIP [23]. In general, there are two main concepts of routing/forwarding protocols for *WSNs*. Firstly since topological information is unreliable, routing should exploit local information to build routes. Secondly, any communication opportunity should be exploited for carrying messages closer to the eventual destination. Suitable control strategy for forwarding is seeking to exploit any communication opportunity in order to carry measured data to the next destination closer to the monitoring center. In this paper, we consider routing as hop-by-hop forwarding similar to previous works. However, contrary to other works, forwarding direction from one *ME* to the other one is depending on level assignment, as we describe in the next section.

Data collection and data forwarding are essential functions of *WSNs*. In a sparse *WSN*, special mobile data collectors (MDCs) are used to gather data from ordinary sensor nodes [9,34,35]. MDCs can be either mobile sinks or mobile relays. Another types of *MEs* are Mobile Relays (MRs) [9,34,35], which are support nodes to gather data from sensor nodes, store them and forward them to sinks or base stations. MR-based data collection in *WSNs* has been proposed in the data-MULE system [20]. Data-MULE consists of a three-tier architecture, where the middle tier is represented by relays, called Mobile Ubiquitous LAN Extensions (MULEs) [9]. The MULE collects data from other nodes and moves to a different location, eventually the base station. One of the most well-known approaches is given by the message ferrying scheme (see [9] for reference). Similar approaches have also been used in the context of opportunistic networks (see [9,11,12,15,16,17,22,34,35]). Fixed nodes build up disconnected *WSNs* using multiple

isolated groups separately. *MEs* visit them from time to time in order to collect data.

Recently, some approaches rely on the concept of mobile peers [9,17,36]. Mobile peers have been used in ZebraNet for wildlife tracking [36], where *MEs* move on random routes. Mobile peers can be both originator and relays of messages in the network, thus, they can transfer their own data as well as those gathered from other peers while moving in the sensing area [9]. The network with mobile peers is homogeneous. These approaches still received little attention in the literature and is the focus of our paper. In another aspect, approaches using *MEs* introduce an increased latency for data transfer. Nevertheless, many applications including traffic pollution monitoring do not require a huge data volume on a day, thus, the latency is acceptable in such delay-tolerant scenarios.

### III. METHOD OVERVIEW

In this section, we present our new method called MBALA (**M**obile **B**ridge using **A**ddaptive **L**evel **A**ssignment) for data forwarding using Mobile Elements (*MEs*) as dynamic bridge in sparse *WSNs*. Our *MEs* act as mobile peers as described in the previous section. However, contrary to previous works, our method uses pre-defined trajectories for moving *MEs* on the main streets in order to collect traffic-generated pollution data. This approach simplifies the issue of coverage and motion control. Our *MEs* will move on pre-defined trajectories and act as data collectors as well as dynamic bridges for intermediate data forwarding to other *MEs*. For data forwarding, we propose a flexible mechanism using level assignment in contrast to fixed forwarding direction in previous works. The level of each *ME* is frequently adapted, in order to determine the forwarding direction. Data forwarding is center-oriented, i.e. forwarding from *MEs* with higher levels on outermost trajectories to *MEs* with lower levels on inner trajectories until the data reaches the center. Using this method, we could provide a certain level on contact opportunity for *MEs* as indicated in the next section. In the next paragraph, we describe the proposed method in details.

We consider a sparse WSN consisting of only Mobile Elements (*MEs*). The deployed area is divided by  $N$  trajectories for moving *MEs*. We design the trajectories to completely cover the deployed area and there is at least one trajectory, where *MEs* can see the base-station. In addition, any two trajectories should have at least one overlapping road distance in order to effectively exploit the contact time for possible intermittent data exchange. While *MEs* are moving on these predefined trajectories, they will gather data as a sensor node, collect measured data from other *MEs* they meet, and act as dynamic bridges for intermediate data transfer in the direction of the base-station. Each *ME* will be assigned a S-Level (Status Level) and a S-Weight, which are frequently updated whenever it meets another *ME* in its coverage. The S-Level is used to orient the hop-by-hop transfer of collected data to the other *ME* (or the base-station). In this sense, *MEs* will have dynamic S-levels, i.e. different S-levels while they move on their trajectories. The data forwarding is base-station oriented based on the S-levels of *MEs*. Collected data will be gradually forwarded from *MEs* with higher S-levels on outermost trajectories to *MEs* with lower S-levels on inner trajectories, until they reach the center. If a node sees many nodes having the same S-level, it then sends data to the node with higher S-Weight.

The S-Level of each *ME* is determined as follows. If a *ME* moves on a trajectory in the base-station's coverage, its S-Level is equal 1. If a *ME* is in the coverage of another, its S-Level is equal to the S-Level of the *ME* it meets plus 1. If a *ME* meets many *MEs*, its S-Level is equal the smallest S-Level of the *MEs* it meets plus 1. *MEs* will frequently update their S-Levels trending towards level improvement (lower S-Level, level decreases as the *ME* closes to the center, S-Level=1 is the closest to the center).

The S-Weight of each *ME* is determined as follows. Initial value of S-Weight is  $K$ , where  $K$  is a sufficiently large number. If a *ME* successfully sent data to another *ME* or the base-station, its S-Weight increases by  $K$ . If a sensor successfully receives data from another sensor, its S-Weight decreases by 1. Since the S-Weight of each *ME* will be decreased by one (1) after each time it received data, if a *ME* can only receive (not send), the S-Weight will be negative

at some time. If the S-Weight becomes negative ( $<0$ ), the S-Level of this *ME* needs to be reset to the S-Level from beginning. This is the flexibility of the proposed method, helps to change the S-Level of a *ME*, when it does not perform his duty as a bridge for receiving and forwarding data, since negative S-Weight means that this *ME* was not able to forward data, but only received data after a long time (depending on  $K$ ). This level change can lead to find out a new path towards the center, if the old path is no longer available due to breakdown (e.g. a *ME* on the path is faulty).

Figure 1 shows an example of the presented method. Assume that there are five trajectories  $T_1...T_5$ , and five *MEs*, namely  $ME_1...ME_5$  moving on them, respectively. The base-station (*BS*) is in the coverage of  $T_1$  and  $T_2$ .

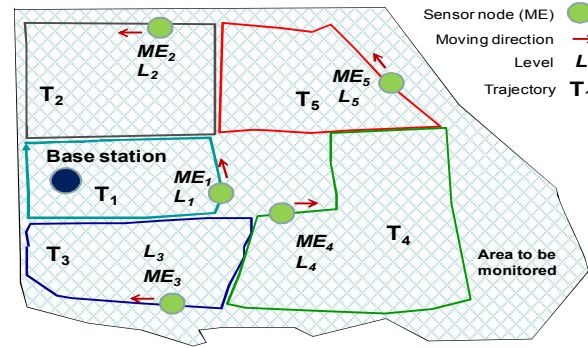


Figure 1. Example for adaptive S-Level changing

$ME_1$  and  $ME_2$  have the opportunity to see the *BS* at some time, so that their S-Levels are updated by  $L_1 = L_2 = 1$ . By moving around their trajectories, the S-Levels of corresponding *MEs* are updated as follows.

- If  $ME_3$  sees  $ME_1 \Rightarrow L_3 = L_1 + 1 = 2$ .
- If  $ME_5$  sees  $ME_2 \Rightarrow L_5 = L_2 + 1 = 2$ .
- If  $ME_4$  sees  $ME_3$  or  $ME_5 \Rightarrow L_4 = L_3 + 1 = 3$  or  $L_4 = L_5 + 1 = 3$ , respectively.

Based on this S-Level assignment, the data forwarding from  $ME_4$  to the *BS* can be in one of the following paths:  $ME_4 \rightarrow ME_3 \rightarrow ME_1 \rightarrow BS$  or  $ME_4 \rightarrow ME_5 \rightarrow ME_2 \rightarrow BS$ . If  $ME_1$  is faulty,  $ME_3$  could not forward data to  $ME_1$ , thus it could only receive data from  $ME_4$ . In this case, its S-Weight is decreased by one ( $W_3 = W_3 - 1$ ) whenever it receives a packet

successfully. Its S-Weight is continuously decreased after each successful receiving until the S-Weight reaches negative. If  $W_3 < 0$ ,  $ME_3$  is reset and its S-Level is re-initialized by  $-1$ . Its S-Level requires to be updated. When  $ME_3$  sees  $ME_4$ , its S-Level is updated based on the S-Level of  $ME_4$ :  $L_3 = L_4 + 1 = 4$ . From now,  $ME_3$  does not receive data from  $ME_4$ , but it will forward data to  $ME_4$  in order to seek other way to the BS. The new forwarding path is:  $ME_3 \rightarrow ME_4 \rightarrow ME_5 \rightarrow ME_2 \rightarrow BS$ .

#### IV. CONTACT OPPORTUNITY ANALYSIS

Obviously, the opportunistic communication between any two  $MEs$  is only possible when the  $MEs$  are in the transmission range of each other. This opportunity depends mainly on the design of trajectories, the mobility control of  $MEs$  and the forwarding mechanism. The trajectories should be designed in such a way, that  $MEs$  can get typical values from the area, the contact possibility is as high as possible, and the contact time should be as much as possible. For mobility control,  $MEs$  should consider the path and speed (or sojourn time) in order to improve the reliable data transfer. A suitable forwarding mechanism needs to be investigated in order to transmit measured data gradually towards the base-station.

Table 1. Notations

$AB$	The overlapping distance of two trajectories with $d$ (meter)
$R$	The coverage diameter of each $ME$ in meter
$a$	The length of the smaller trajectory 1
$ka + b$	The length of the larger trajectory 2
$v$	The average velocity of $ME_1$ and $ME_2$
$[x]$	The integer part of $x$

As mentioned, mobility introduces additional challenges for data collection and transfer. The data exchange can only happen, if any two  $MEs$  can meet each other and have enough contact time for data transfer. Contact opportunity of  $MEs$  is one of the essential issues we have to solve in our method. This opportunity is depending on the design of trajectories, the overlapping distance, the velocity, the radio coverage and the moving direction of  $MEs$ . In fact, the

connectivity may be only intermittent, when  $MEs$  in the radio coverage of each other. In this Section, we investigate the issue of trajectory design and analyze the contact opportunity of  $MEs$ . This analysis will prove the feasibility of our method.

Without loss of generality, we investigate the case with two any adjacent trajectories using two scenarios as follows:

- i) Two  $MEs$  are moving in the same direction;
- ii) Two  $MEs$  are moving in the reverse direction.

Intuitively, we can deduce that  $k=1$  and  $b < a$ , i.e.  $b \in [0, a]$ . We consider the following major problems: the necessary condition for contact of any two  $MEs$ , the contact probability and the forwarding probability within a given time, contact time of meetings. Here, we consider the general situation, where each of  $MEs$  does not have any information on the position of other one.

##### A. Determination of contact opportunity

###### 1) Case 1: $MEs$ are moving in the same direction on $AB$

Without loss of generality, we assume that  $ME_1$  is at the position  $A$  and  $ME_2$  is at the position  $M$  at time  $t_0$ . The distance between  $M$  and  $A$  is  $l$  in the motion direction of  $ME_2$  (Fig. 2), where  $l \in [0, a]$ . If  $l > a$ , we can proceed  $ME_1$  in one round more to get  $l \leq a$ .

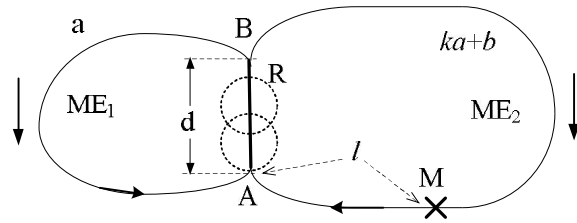


Figure 2. Two  $MEs$  are in the same direction on  $AB$

Therefore,  $ME_1$  will be at the position  $A$  at each time  $t_{01}(n)$ , as follows:

$$t_{01}(n) = t_0 + n a / v$$

Where  $n$  is the number of rounds  $ME_1$  should move on  $a$ , in order to contact  $ME_2$ . Accordingly,  $ME_2$  will be at position  $A$  at every time  $t_{02}(m)$ :

$$t_{02}(m) = t_0 + l / v + m (ka + b) / v$$

**Remark:** For each real number  $l \in [0, a]$ , the

necessary condition that two  $MEs$  can contact with each other and transfer data from one to the other is as follows:

$$\exists m, n: |t_{01}(n) - t_{02}(m)| \leq R/v$$

$$\Leftrightarrow na - m(ka + b) - R \leq l \leq na - m(ka + b) + R \quad (1)$$

$$\Leftrightarrow n - m(k + b/a) - R/a \leq l/a \leq n - m(k + b/a) + R/a \quad (2)$$

**Remark:** If two  $MEs$  meet each other with same speed, the constant contact time of them is:

$$t_{contact} = (d + 2R) / v \quad (3)$$

**Remark:** From Table 2, if  $l \in [4714, 4854] \cup [5351, a]$ , then two  $MEs$  can contact to each other.

Table 2. Test case for  $a=5421m$ ,  $ka+b=6058m$  for  $MEs$  in the same direction on  $AB$

Test case: $a=5421m$ , $ka+b=6058m$ , $k=1$ , $b=637m$ , $d=886m$ , $R=70m$			
$n$	$m$	$na - m(ka + b) - R$	$na - m(ka + b) + R$
2	1	5351	5491
3	2	4714	4854

## 2) Case 2: $MEs$ are moving in the reverse direction on $AB$

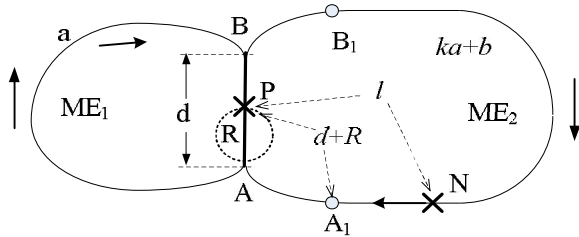


Figure 3. Two  $MEs$  are in the reverse direction on  $AB$

Without loss of generality we assume that  $P$  is the position of  $ME_1$ , and  $N$  is the position of  $ME_2$  at time  $t_0$ , where  $P$  is the middle point of distance  $d$ .  $ME_1$  will be at position  $P$  at each time  $t_{03}(n)$  as follows:

$$t_{03}(n) = t_0 + n a / v$$

The distance between  $N$  and  $P$  is  $l$  in the motion direction of  $ME_2$  (Fig.3), where  $l \in [0, a]$ . If  $l > a$ , we can proceed  $ME_1$  another round to get  $l \leq a$ .

Consider the position  $A_1$  of  $ME_2$ , where  $A_1 \in (A, N)$  and  $A_1$  has a distance  $(d/2 + R)$  to  $A$ . Accordingly, consider the position  $B_1$  of  $ME_2$ , where  $B_1$  of has a distance  $(d/2 + R)$  to  $B$ .  $A_1$  and  $B_1$  are the positions,

where  $ME_2$  can meet  $ME_1$ .  $ME_2$  moves from  $N$  to  $A_1$  during the time.

$$(l - (d + R)) / v$$

$ME_2$  will be at position  $A_1$  at each time  $t_{04}(m)$ :

$$t_{04}(m) = t_0 + (l - (d + R)) / v + m(ka + b) / v$$

**Remark:** For each real number  $l \in [0, a]$ , the necessary condition that two  $MEs$  can contact with each other and transfer data from one to the other is as follows:

$$\exists m, n: |t_{04}(m) - t_{03}(n)| \leq 2(d + R) / v$$

$$\Leftrightarrow na - m(ka + b) \leq l \leq na - m(ka + b) + 2(d + R) \quad (4)$$

$$\Leftrightarrow n - m(k + b/a) \leq l/a \leq n - m(k + b/a) + 2(d + R)/a \quad (5)$$

Let we denote  $X$  and  $Y$  as follows:

$$X = n - m(k + b/a)$$

$$Y = n - m(k + b/a) + 2(d + R) / a$$

we have:  $Y - X = 2(d + R) / a$

**Remark:** If  $ME_1$  and  $ME_2$  move in reserve direction, the necessary condition for that they meets each other is:  $l \in [0, d + R]$  and  $l/a \in [0, 1]$ . The contact time of two  $MEs$  will be:

$$R / v \leq t_{contact} \leq 2R / v \quad (6)$$

Table 3. Test case for  $a=5421m$ ,  $k*a+b=6058m$  for  $MEs$  in reverse direction on  $AB$

Test case: $a=5421m$ , $ka+b=6058m$ , $k=1$ , $b=637m$ , $d=886m$ , $R=70m$			
$n$	$m$	$na - m(ka + b)$	$na - m(ka + b) + 2(d + R)$
2	1	4784	6696

**Remark:** if  $l \in [4784, a]$  then two  $MEs$  can contact to each other.

## B. Contact probability

**Remark:** If we randomly choose a real number  $l \in [0, a]$ , the probability that  $l \in [c1, c2] \subset [0, a]$  is  $(c2 - c1) / a$ .

### 1) Case 1: $MEs$ are moving in the same direction on $AB$

For each  $l \in [0, a]$ , two  $MEs$  will meet each other only if there exist two positive integers  $m$  and  $n$  that satisfy the inequality (2), i.e.:

$$n - m(k + b/a) - R/a \leq l/a \leq n - m(k + b/a) + R/a$$

Since  $l/a \in [0, 1]$ , we can have:

$$n - m(k + b/a) - R/a \leq l/a \leq 1$$

$$\Rightarrow n \leq m(k + b/a) + 1 + R/a$$

$$\Leftrightarrow n < m(k + 1) + 1 + R/a$$

Because  $R/a < 1$  (in reality  $R < a$ ), we have:

$$n \leq m(k + 1) + 1$$

According to the assumption in 1.1 (Fig.2), we consider two sub-cases as follows.

**Sub-case 1:** For  $l \in [0, R]$ , the distance between two *MEs* is less than or equal to  $R$ . Thus, two *MEs* can always contact to each other whenever the minimum contact time is  $R/v$ . No condition is needed on  $m$  or  $n$  for the contact probability of two *MEs* in this case. The probability for contact of two *MEs* can be estimated by the following:

$$P(\text{contact}) \geq R/a$$

**Sub-case 2:** For  $l \in (R, a)$ , two *MEs* will meet each other only if there exist two positive integer  $m$  and  $n$  that satisfy the inequality (1), i.e.:

$$na - m(ka + b) - R \leq l \leq na - m(ka + b) + R$$

We denote  $C_{mn}$  the random event of randomly taking a real number  $l$  that satisfies:

$$l \in \alpha_{mn} := [na - m(ka + b) - R, na - m(ka + b) + R] \cap [0, a]$$

Let us denote  $N_1$  and  $N_2$  as follows:

$$N_1 = na - m(ka + b) - R$$

$$N_2 = na - m(ka + b) + R$$

We can get the following relation:

$$P(C_{mn}) = \begin{cases} \frac{R}{a} + \frac{a - N_1}{a}, & \text{if } R \leq N_1 < a \leq N_2 \\ \frac{R}{a} + \frac{(N_2 - N_1)}{a} = \frac{3R}{a}, & \text{if } R \leq N_1 < N_2 < a \\ \frac{R}{a} + \frac{N_2 - R}{a}, & \text{if } N_1 \leq R \leq N_2 < a \\ \frac{R}{a}, & \text{others} \end{cases}$$

The contact probability of any two *MEs* moving in the same direction is:

$$P(\text{contact}) = P(\bigcup_{m,n} C_{mn})$$

where  $\bigcup C_{mn}$  is an event randomly taking a real number:  $l \in \bigcup_{m,n} \alpha_{mn}$

## 2) Case 2: *MEs* are moving in the reverse direction on AB

For each  $l \in [0, a]$ , two *MEs* will meet each other only if there exist two positive integers  $m$  and  $n$  that satisfy the inequality (5), i.e.:

$$n - m(k + b/a) \leq l/a \leq n - m(k + b/a) + 2(d + R)/a$$

Since  $l/a \in [0, 1]$ , we have:

$$n - m(k + b/a) \leq l/a \leq 1$$

$$\Rightarrow n \leq m(k + b/a) + 1 < m(k + 1) + 1$$

Therefore, we can conclude that:

$$n < m(k + 1) + 1$$

For the case two *MEs* are moving in the reverse direction, we investigate two sub-cases.

**Sub-case 1:** If  $l \in [0, d + R]$ , then  $ME_1$  and  $ME_2$  can obviously always in contact with each other. Thus, we do not need to put attention on the condition for  $m$  and  $n$ . The contact probability for two *MEs* can be estimated by the following item:

$$P(\text{contact}) \geq (d + R)/a$$

That is, the more the value of  $R$  is, the more is the contact probability of *MEs*.

**Sub-case 2:** If  $l \in (d + R, a]$ , then  $ME_1$  and  $ME_2$  can meet each other, only if there exist two positive integers  $m$  and  $n$  satisfying (4), i.e.:

$$na - m(ka + b) \leq l \leq na - m(ka + b) + 2(d + R)$$

We denote  $B_{mn}$  the random event of randomly taking a real number  $l$  that satisfies:

$$l \in \beta_{mn} := [na - m(ka + b), na - m(ka + b) + 2(d + R)] \cap [0, a]$$

Let us denote  $M_1$  and  $M_2$  as follows:

$$M_1 = na - m(ka + b)$$

$$M_2 = na - m(ka + b) + 2(d + R)$$

we get the following relation:

$$P(B_{mn}) = \begin{cases} \frac{(d+R)}{a} + \frac{a-M_1}{a}, & \text{if } d+R \leq M_1 \leq a \leq M_2 \\ \frac{(d+R)}{a} + \frac{(M_2-M_1)}{a}, & \text{if } d+R \leq M_1 < M_2 < a \\ \frac{M_2}{a}, & \text{if } M_1 \leq d+R \leq M_2 < a \\ \frac{(d+R)}{a}, & \text{if } M_2 \leq d+R \\ 1, & \text{if } M_1 \leq 0 < a \leq M_2 \end{cases}$$

We can conclude the probability for contact of any two *MEs* moving in the reserve direction:

$$P(\text{contact}) = P\left(\bigcup_{m,n} B_{mn}\right)$$

Where  $\bigcup B_{mn}$  is an event randomly taking a real value:  $l \in \bigcup_{m,n} \beta_{mn}$

### C. Forwarding probability within a given time

#### 1) Case 1: *MEs* are moving in the same direction on AB

With a given time  $T$ , we can assume that the positive integer parameters  $m$  and  $n$  in symbol  $C_{mn}$  should satisfy the following conditions:

$$\text{Where: } n_1 = \left\lceil \frac{T}{a/v} \right\rceil \text{ and } m_1 = \left\lceil \frac{T-l/v}{(ka+b)/v} \right\rceil$$

The probability that two *MEs* can forward data to each other within a given time  $T$  is follows:

$$P_{\text{samedirection}} = P\left(\bigcup_{\substack{m,n \\ n \leq n_1, m \leq m_1}} C_{mn}\right)$$

**Example 1:** Suppose  $T=8$  hours, the velocity of two *MEs* is 10 km/h. We can deduce that each *ME* will move about 16 rounds ( $n, m \leq 16$ ). We only have intervals  $\alpha_{mn}$  listed belows that intersect with the interval  $[0, a]$  is non-empty (Table 4).

Table 4. Test case for forwarding within a given time for 2 *MEs* in the same direction on AB

Test case: $a=5421m$ , $ka+b=6058m$ , $k=1$ , $b=637m$ , $d=886m$ , $R=70m$					
No	n	m	$na-m(ka+b)$	$na-m(ka+b)+2(d+R)$	$\alpha_{mn}$
1	2	1	4714	4784	[4714,4784]
2	3	2	4077	4147	[4077,4147]

3	4	3	3440	3510	[3440,3510]
4	5	4	2803	2873	[2803,2873]
5	6	5	2166	2236	[2166,2236]
6	7	6	1529	1599	[1529,1599]
7	8	7	892	962	[892,962]
8	9	8	255	325	[255,325]
9	11	9	5039	5109	[5039,5109]
10	12	10	4402	4472	[4402,4472]
11	13	11	3765	3835	[3765,3835]
12	14	12	3128	3198	[3128,3198]
13	15	13	2491	2561	[2491,2561]
14	16	14	1854	1924	[1854,1924]

**Remark:** Since the intervals  $\alpha_{mn}$  presented above is non-intersect paired with each other, we conclude:

$$P\left(\bigcup_{\substack{n \leq 16 \\ m \leq 16}} C_{mn}\right) = \bigcup_{m \leq 16} P(C_{mn}) = 14 * 70 / a \approx 0.1808$$

**Meaning of the result:** The notation  $P(A/B)$  is the probability of an event A when that event B has occurred. According to table 4, the above probability is small ( $< 20\%$ ). However, this calculation is based on the general basis (i.e., two *MEs* do not know the location of each other at each time). Furthermore, one advantage is that the contact time (the communication time) of two *MEs* is large enough in comparison to the demand (i.e.  $(2R+d)/v$ ). If we adjust the starting time of two *MEs* depending on each other based on the previous calculation, the contact probability can be further improved (may reach 100%). The expense for improving the communication time is reasonable small because we only need to adjust the starting time of each *ME* (no need to increase the transmission power). Intuitively, we can have:

$$P(\text{contact} / C_{12}) = P(\text{contact} / C_{23}) = \dots = P(\text{contact} / C_{89}) = 1$$

$$P(\text{contact} / C_{9,11}) = P(\text{contact} / C_{10,12}) = \dots = P(\text{contact} / C_{14,16}) = 1$$

#### 2) Case 2: *MEs* are moving in the reverse direction on AB

With a given time  $T$ , we can assume that the positive integer parameters  $m$  and  $n$  in symbol  $B_{mn}$

should satisfy the following conditions:

$$n \leq n_2, m \leq m_2$$

$$\text{Where: } n_2 = \left\lceil \frac{T}{a/v} \right\rceil \text{ and } m_2 = \left\lceil \frac{T - (l - (d+R))/v}{(ka+b)/v} \right\rceil$$

The probability that two *MEs* can forward data to each other within a given time  $T$  is follows:

$$P_{\text{reverse\_direction}} = P\left(\bigcup_{n \leq n_2, m \leq m_2} B_{mn}\right)$$

**Example 2:** Suppose  $T=8$  hours, the velocity of two *MEs* is 10 km/h. We can deduce that each *ME* will move about 16 rounds ( $n, m \leq 16$ ). However, we can see that,  $n$  and  $m$  are only needed to be less than 9 in order to have the contact probability of 1 (Table 5).

Table 5. Test case for forwarding within a given time for 2 *MEs* in the reverse direction on AB

Test case: $a=5421m, ka+b=6058m, k=1, b=637m, d=886m, R=70m$					
No	$n$	$m$	$na - m(ka+b)$	$na - m(ka+b) + 2(d+R)$	$\beta_{mn}$
1	2	1	4784	6696	[4784, 5421]
2	5	4	2873	4785	[2873, 4785]
3	8	7	962	2874	[962, 2874]
4	1	1	-637	1275	[0, 1275]

**Remark:** From the Table 5 we conclude:

$$\bigcup_{\substack{n \leq 9 \\ m \leq 9}} \beta_{mn} = [0, a]$$

$$P(\text{contact}) = P\left(\bigcup_{\substack{n \leq 16 \\ m \leq 16}} B_{mn}\right) = 1$$

**Meaning of the results:** The result of this calculation gives a contact probability of 100%. based on the fact that two *MEs* can start at any time (i.e. two *MEs* do not know any information on the location of each other at each time). If we adjust the starting time, two *MEs* could definitely meet each other in a possible smallest time ( $n=1, m=1$ ). According to table 5, we have:

$$P(\text{contact} / B_{12}) = P(\text{contact} / B_{45}) = \dots = P(\text{contact} / B_{78}) = P(\text{contact} / B_{11}) = 1$$

However, the communication time of two *MEs* is very small (the value is from  $R/v$  to  $2R/v$ ), which is

completely depending on the radius  $R$  of the radio coverage. There is a large expense in order to increase the communication time due to the need to increase the transmission power of the radio equipment. Thus, if we adjust the starting time of each *ME* to have longer contact time, the approach using same moving direction for two *MEs* will be less expensive.

#### D. Contact time of each meeting

##### 1) Case 1: *MEs* are moving in the same direction on AB

Let us denote with  $t_{con,S}$  the contact time for each time two *MEs* can meet each other. In case two *MEs* are moving in the same direction, we have the following equation according to (3):

$$t_{con,S} = (2R+d) / v$$

##### 2) Case 2: *MEs* are moving in the reverse direction on AB

Let us denote with  $t_{con,R}$  the contact time for each time two *MEs* can meet each other. In case two *MEs* are moving in the reserve direction, from (6) we have the following inequality:

$$t_{reverse}^{\min} \leq t_{con,R} \leq t_{reserve}^{\max}$$

$$\text{Where: } t_{reserve}^{\min} = R/v, t_{reserve}^{\max} = 2R/v$$

#### E. Discussion on higher contact probability

The contact probability analysis did not need position information of any *ME* at any time. Remarks of Table 2 and 3 can conclude that, if two *MEs* move into suitable positions as indicated by some way, they can surely meet each other. As long as  $ME_2$  is in the intervals indicated in Table 2 and 3, it will wait until  $ME_1$  achieves the corresponding position  $A$  or  $P$  before continuing to move, then the contact probability of two *MEs* is certainly equal to one. By comparing two moving directions, we can recognize the following advantages and disadvantages. If two *MEs* are moving in the reserve direction, the contact probability is higher, but the contact time is shorter. Contrary, if two *MEs* moving in the same direction, the contact probability is lower. However, if two *MEs* meet each other, the contact time is much larger, even the whole distance  $d$ .

## V. PROTOCOL DEVELOPMENT

### A. Protocol design

The overall data collection and forwarding process can be split into four states describing the behavior of a *ME* (Fig. 4). During initiation, a *ME* will set its S-Level to -1 and its S-Weight to  $K$ .

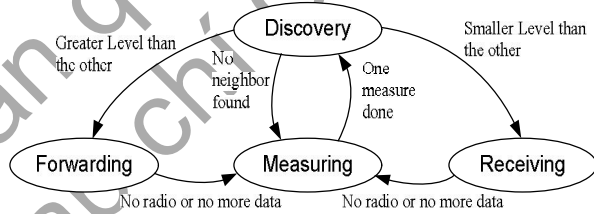


Figure 4. Protocol state diagram

**Discovery State:** It is the basic state, to which a *ME* will return after each measurement. In this state, *MEs* scan the network to find possible neighbors. The scan time will last for few seconds. *ME* gets the information from other neighbors including node identifier, S-Level and S-Weight. The S-Level of the current node is then updated as follows.

$$L_i = \begin{cases} 1 & \text{if } i \text{ sees the base station} \\ L_{\min} + 1 & \text{if } L_i = -1 \text{ \& } L_{\min} \geq 0 \text{ or } L_i \geq L_{\min} \\ -1 & \text{if } W_i < 0 \end{cases}$$

Where  $i$  is the ID of the node  $i$ ,  $L_i$  is the current S-Level of *ME*  $i$ ,  $W_i$  is the S-Weight of node  $i$ ,  $L_{\min}$  is the minimum S-Level of  $n$  neighbor nodes, that is:

$$L_{\min} = \min_{j \in n} \{L_j\}$$

The S-Weight of current node will be calculated:

$$W_i = \begin{cases} W_i + K & \text{if } m > 0 \\ W_i - 1 & \text{if } n > 0 \end{cases}$$

Where  $m$  is the number of successful transmitted packets in the last sending phase,  $n$  is the number of successful received packets.

**Forwarding State:** A *ME* will change to this state from the discovery state, if its S-Level is greater than the S-Level of its neighbor node. Data forwarding is a center-oriented process, i.e. a node tries to send to the inner trajectory. The current node will read a block of locally stored data and send them to the next node,

either the base station or the next intermediate node determined in the discovery state. The block transfer will stop, when the radio channel becomes unavailable. After successful transfer of a block, the S-Weight of corresponding node will be updated as described above. Then, the current *ME* changes to the measuring state.

**Receiving State:** A *ME* will change to this state from the discovery state, if its S-Level is smaller than the S-Level of its neighbor node. In this state, *ME* checks the radio channel for possible incoming packets from other nodes. If data are available, *ME* tries to get data and stores into its local memory. This state finishes after a block of incoming data has been received or after timeout. The next state is measuring state, if *ME* detects no radio signal or no more data from the other *MEs*.

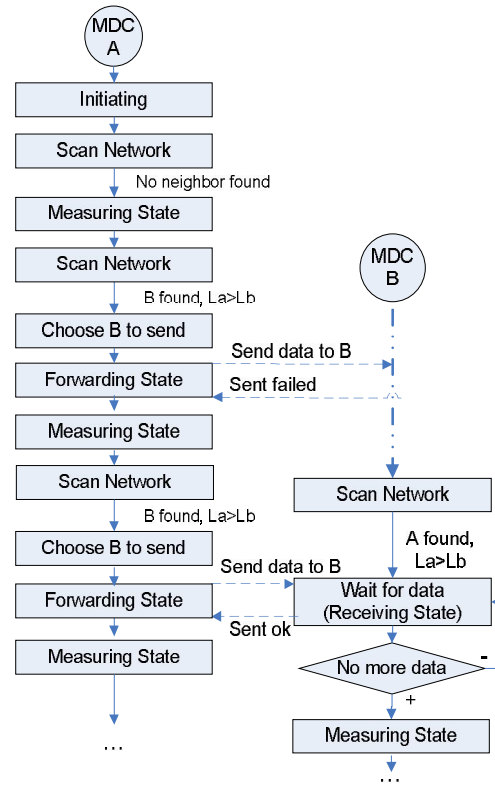


Figure 5. Synchronization process

**Measuring State:** A *ME* will change to this state from the discovery state, whenever it detects no neighbor nodes in its coverage, or *ME* detects no radio signal or no more data from the other *MEs*. In this state, *ME* will sample the pollution data and stores in

the local memory of the *ME* for sending to the other intermediate nodes in the next sending state. After getting a block of measured data, the *ME* will return to the discovery state.

### B. Synchronization issues

Figure 5 presents the synchronization between four states of two *MEs* during forwarding and receiving states. The measuring state lasts only few seconds. The node spends the most time for waiting received data from other nodes in order to gap the mismatch between different states of multiple nodes.

### C. Protocol algorithm

The mnemonic codes of the protocol is as follows.

```

Initiating() {
    Identify_Current Node ID();
    Prepare_Signaling_Information( NodeID, Li, Wi );
    state = Discovery; }

Discovery() {
    scanNetwork(NodeID, Li, Wi);
    while ( noNeighbor > 1 ) {
        getNeighbor(IDj, Lj, Wj);
        if (IDj == BS_ID) Li = 1;
        nextDestination = Base_Station;
        return( state = Forwarding);
    }
    noNeighbor = noNeighbor - 1; }
    while ( noNeighbor > 1 )
        { Lmin = get_min_S-Level(IDj, Lj, Wj); }
    if ( ( (Li == -1) && (Lmin ≥ 0) ) || (Lmin ≥ Li) )
        Li = Lmin + 1;
    nextDestination = WSN_Nodei;
    return( state = Forwarding);
    endif
    if (Wi ≤ 0) { Li = -1; return(state = Measuring); }
    else return (state = Receiving);
    endif
    endif }

Forwarding() {
    while ( Sending_TX_OK ) {
        if (getLastMemRecord( current_Rec ))
            return( state = Measuring );

```

```

    else
        send( nextDestination, getLastMemRecord);
        updatedLastMemRecord(current_Rec);
    endif }
    if ( mPacket_sent_Ok ) { Wi = Wi + K; }
    endif
    return( state = Measuring ); }

```

```

Receiving() {
    while ( (notTimeout) && (RF_data_Available)) {
        while ( incoming_Packet ) {
            store_packet_in_Mem(); } }
        if ( nPacket_received_Ok ) { Wi = Wi - 1; }
        return( state = Measuring ); }

```

```

Measuring() {
    get_environment_Data();
    store_Data_in_Mem();
    return( state = Discovery ); }

```

### D. Performance parameters

We use four parameters for evaluating the protocol performance, namely number of packet delivered or received, packet loss rate, throughput and packet delay. Denote  $N_{loss}$  the number of lost packets,  $N_{received}$  the number of received packets at destination,  $N_{sent}$  the number of packets sent from a source, the packet loss rate is defined as follows.

$$P_{loss} = N_{loss} / (N_{loss} + N_{received}) = N_{loss} / N_{sent}$$

Packet delay is defined as the time different of the receiving time of a packet and the sending time of the corresponding packet.

$$Delay = t_{rec} - t_{sent}$$

Throughput  $\theta_{i-j}$  of packet transfer from node  $ID_i$  to node  $ID_j$  in a time interval  $(t_2 - t_1)$  is:

$$\theta_{i-j} = (N_2 - N_1) / (t_2 - t_1)$$

Where  $N_2$  and  $N_1$  is the number of packets sent at time  $t_2$  and  $t_1$ , respectively. In fact, throughput is the number of sent packets per second.

## VI. CASE STUDY

In this section, we describe the deployment of our proposed protocol for collecting pollution data from an urban district of Hanoi City. We have implemented

a number of *MEs* for testing the proposed protocol using a ZigBee mesh network. Our network is based on a Meshlium base station and eight Wasmotes [32]. The sensor nodes move on eight trajectories based on the roads with most traffic pollution in order to cover an urban district of Hanoi City with an area of about  $14.6 \text{ km}^2$  as shown in the Fig. 6.

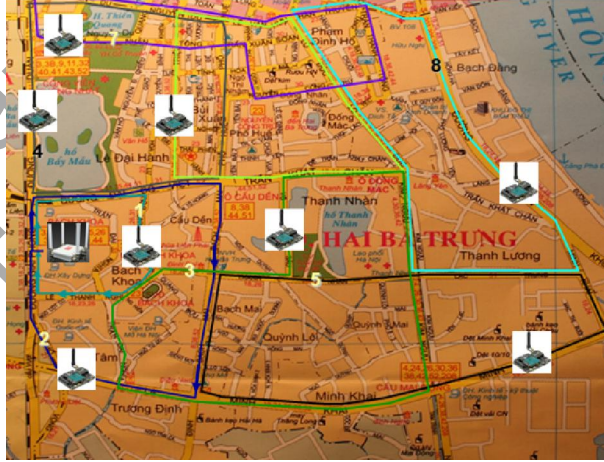


Figure 6. Designed trajectories in the deployment area

The trajectories have been designed so that each mote can meet at least one other mote with enough time to be able to send the data. Additionally two of the trajectories have some parts, on which Wasmotes can meet the Meshlium. Each mote runs on a trajectory and is numbered according to the trajectory number, i.e. from 1 to 8. The average moving speed of each *ME* is  $10 \text{ km/h}$  ( $\sim 2.8 \text{ m/s}$ ).

We built two scenarios to test the protocol operation in various circumstances as follows.

**Scenario 1:** Test the forwarding mechanism. In this scenario, Wasmotes move on 8 trajectories (Fig. 6). For convenience, we denote a node with *ID* (e.g.  $ID_8$  for the Wasmote  $S_8$ ), the S-Level of a node with *L* (e.g.  $L_3$  for the S-Level 3). We investigated the forwarding route  $ID_8(L_3) \rightarrow ID_3(L_2) \rightarrow ID_1(L_1) \rightarrow \text{Meshlium}$ . After a time of measuring data,  $ID_8$  saw  $ID_3$  on an overlapping road distance of  $665\text{m}$  (contact time is about  $238 \text{ seconds}$ ). After that,  $ID_3$  moved alone on his trajectory that has a distance of  $3066\text{m}$ . Then,  $ID_3$  saw  $ID_1$  on an overlapping road distance of  $365\text{m}$  (contact time is about  $130 \text{ seconds}$ ).

The average forwarding speed of each *ME* is  $100 \text{ kbps}$ . Using Libelium Wasmotes [37], one protocol cycle lasts for about  $40 \text{ seconds}$ , i.e. at least one data packet would be sampled within this cycle and stored in the local memory of the Wasmote. Therefore, the *maximum packet delay* will be  $40 \text{ seconds}$  in order to avoid buffer overflow in Wasmotes. Accordingly, the *minimum throughput* will be  $1/40 = 0.025 \text{ packet/s}$ .

Figures 7 and 8 present the detailed results of our experiments according to the scenario 1. Figure 7 shows the packet delay by forwarding packets from  $ID_8$  to  $ID_3$  (solid line) and from  $ID_3$  to  $ID_1$  (dotted line). As shown in the figure, the average delay varies from  $1.0\text{s}$  to  $2.0\text{s}$  for both routes, less lower than the allowed maximum packet delay.

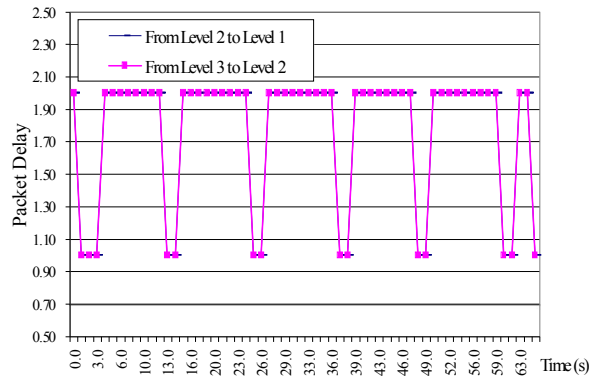


Figure 7. The packet delay by forwarding packets

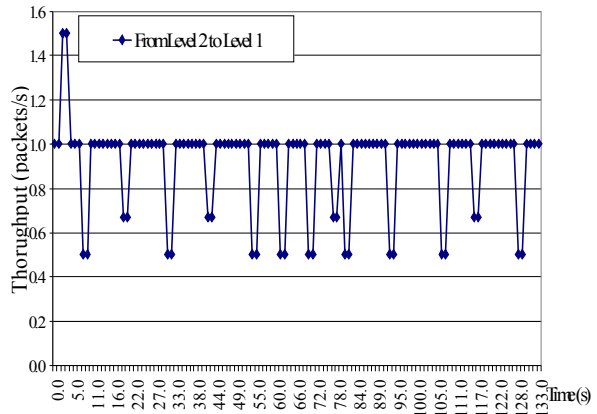


Figure 8. The throughput of data transfer

Figure 8 presents the throughput (packets/s) of data transfer from  $ID_3$  to  $ID_1$ . We calculate the throughput for each interval of one second. As shown, the throughput of data transfer from  $ID_3$  to  $ID_1$  is from  $0.5$

to 1.5 packets/s, which is higher than the required *minimum throughput*.

In scenario 1, we have sent 121 packets from  $ID_3$  to  $ID_1$ . Among them, we counted that 10 packets are lost. Thus, the corresponding packet loss rate is about 10%. In our experiments, the loss rate for data transfer from  $ID_8$  to  $ID_1$ , from  $ID_8$  to  $ID_3$  is 18% and 20%, respectively. This loss rate is reasonable, since the Wasmotes can have a lot of measurements at a location, thus the loss of some packets might not affect the data processing at the center.

**Scenario 2:** Test the possibility to find out a new path, when the old path is faulty. To test the adaptive S-Level change, we have stopped the operation of Wasmote 2 and 3 in scenario 1 in order to show the adaptive change of the forwarding path in scenario 2. The results are given in Table 6 as follows.

Table 6. Adaptive S-Level changing in scenario 2 in comparison to previous S-Level in scenario 1

Wasp -mote ID	S-Level		Wasp- mote ID	S-Level	
	Scena- rio 1	Scena- rio 2		Scena- rio 1	Scena- rio 2
1	1	1	5	2	5
2	1	Stop	6	2	3
3	2	Stop	7	3	3
4	2	2	8	3	4

Explanation:

- Since  $ID_6$  can only meet  $ID_4$  (S-Level 2), its S-Level has changed to 3 instead of 2.

- Since  $ID_8$  can meet  $ID_6$  and  $ID_7$  (S-Level 3), its S-Level has changed to 4 instead of 3 as before.

- Since  $ID_5$  can meet  $ID_8$  (S-Level 4), its S-Level has changed to 5 instead of 2 as before. That is,  $ID_5$  was not an intermediate node for  $ID_8$  anymore. Instead, it used  $ID_8$  as an intermediate node for sending its data to the Meshlimum via indirect routes (5-8-6-4-1 or 5-8-7-4-1).

**Comparison with conventional solutions:** Our case study was carried out on 23 main streets with the total length of more than 25 km. The radio coverage for each sensor node is 70 m. If we use stationary sensors as in conventional solutions, 180 sensor nodes are necessary. By our method, we only need 8 sensor nodes in order to be able to collect data on the whole

23 main streets. This is the benefit of our method in comparison with other conventional solutions.

## VII. CONCLUSION

In this paper, we presented a method for data forwarding using *MEs* as dynamic bridges in sparse *WSNs*. Using this method, we can collect measured data in a wide area with a limited number of mobile nodes. The way to design trajectories aiming to cover a deployed area is suitable for traffic-generated pollution monitoring, because it can rely on the main traffic roads causing significant pollution. In addition, the paper analyzed the opportunistic contact of *MEs* while moving on overlapping road distances of respective trajectories and proved the feasibility of this method. Our proposed method has been used for developing a protocol, and has been successfully tested using a ZigBee *WSN* for traffic-generated pollution monitoring in an urban area of Hanoi city.

Further works will be in deeper analysis of contact opportunity of *MEs* in order to provide useful recommendations for designing trajectories as well as further testing of the method with the trajectories that are the public bus lines for traffic air pollution monitoring.

## VIII. ACKNOWLEDGEMENTS

This research has been supported by the Alexander von Humboldt Foundation (Germany). This work has also been co-funded by the DFG as part of the CRC 1053 MAKI. We would like to thank the sponsors for offering the opportunity, by which German and Vietnamese researchers can cooperate in this joint research work.

## REFERENCES

- [1] Khedol, K. K., Perseedoss, R., Mungur, A., 2010, Wireless sensor network air pollution monitoring system, International journal of wireless and mobile networks, May 2010, Vol.2, No.2, pp.31-45.
- [2] TU Darmstadt, Germany, "Da-Sense: Monitoring system for Smart cities", <http://www.da-sense.de>.
- [3] Gómez, M., 2012, RESCATAME Project Pervasive Air-Quality Sensors Network for an Environmental Sustainable Urban Traffic Management, Intergroup meeting: AIR POLICY, Project report 26/09/2012.
- [4] Beutel, J., et al., 2009, PermaDAQ: A scientific instrument for precision sensing and data recovery in

- environmental extremes, In *Processing of the International Conference on Sensor Networks (IPSN)*, IEEE Computer Society, April 2009, pp.265-276.
- [5] ETH Zurich and EPF Lausanne, OpenSense: Open sensor networks for air quality monitoring, <http://data.opensense.ethz.ch>.
- [6] Schweizer, I., Bärthel, R., Schulz, A., Probst, F., Mühlhäuser, M., 2011, NoiseMap: Real-time participatory noise maps, In *2nd International Workshop on Sensing Applications on Mobile Phones*, 2011.
- [7] Suciu, D., 2010, A study of RF link and coverage in ZigBee, *Scientific Bulletin of the "Petru Major" University of Targu Mureq*, Vol. 7 (XXIV), No. 1, 2010, p.5-10.
- [8] Akyildiz, I.F., Su, W., Sankarasubra-Maniam, Y., Cayirci, E., 2002, *Wireless Sensor Networks: A Survey*, *Computer Networks*, vol. 38, issue 4, 2002, p.393-422.
- [9] Francesco, M., Das, S., 2011, *Data Collection in Wireless Sensor Networks with Mobile Elements: A Survey*, *Journal ACM Trans. on Sensor Networks*, Vol.8, Issue 1, Aug. 2011, NY, USA.
- [10] Gaojun, F., Shiyao, J., 2010, Coverage Problem in Wireless Sensor Network: A Survey, *Journal of Networks*, Vol 5, No 9 (2010), Sep. 2010, 1033-1040.
- [11] Luo, L., Huang, C., Abdelzaher, T., Stankovic, J., 2007, Envirostore: A cooperative storage system for disconnected operation in sensor networks, *Proc. of IEEE INFOCOM (2007)*, p.1802-1810.
- [12] Yu-Chee, T., Fang-Jing, W., Wan-Ting, L., 2013, Opportunistic data collection for disconnected wireless sensor networks by mobile mules, *Journal Ad Hoc Networks*, Vol.11, Issue 3, May 2013, p.1150-1164.
- [13] Diggavi, S., Grossglauser, M., Tse, D., 2002, Even one-dimensional mobility increases ad hoc wireless capacity. In *Proc. of IEEE International Symposium on Information Theory*, 2002, p.352.
- [14] Ekici, E., Gu, Y., Bozdog, D., 2006, Mobility-based communication in wireless sensor networks. *IEEE Communications Magazine*, 44(7), p.56.
- [15] Anastasi, G., Conti, M., and Di Francesco, M., 2009. Reliable and energy-efficient data collection in sparse sensor networks with mobile elements. *Performance Evaluation* 66, 12 (December), p791-810.
- [16] Chia-Ho, O., 2011, Data delivery with mobility control in sparse wireless sensor networks, *Intl. Journal of Innovative Computing, Information and Control*, Vol.7, No.4, Apr.2011, p.1621-1638.
- [17] Rawat, P., Singh, K.D., Chaouchi, H., Bonnin, J.M., 2014, Wireless sensor networks: a survey on recent developments and potential synergies", *Journal of Supercomputing*, Vol.68, Issue 1, Apr.2014, p.1-48.
- [18] Saukh, O., Hasenfratz, D., Walser, C., Thiele, L., 2013, On rendezvous in mobile sensing networks, *Real-World Wireless Sensor Networks*, *Lecture Notes in Electrical Engineering*, Vol.281, 2014, p.29-42.
- [19] Wang, G., Cao, G., La Porta, T., Zhang, W., 2005. Sensor relocation in mobile sensor networks. In *Proc. of the 24th IEEE Conference on Computer Communications (INFOCOM 2005)*, Vol. 4., p2302-2312.
- [20] Shah R. C., Roy S., Jain S. & Brunette W., 2003, Data MULEs: Modelling a Three-tier Architecture for Sparse Sensor Networks, *Proc. IEEE Int'l Workshop on Sensor Network Protocols and Applications (SNPA 2003)*, p.30- 41.
- [21] Jain, S., Shah, R., Brunette, W., Borriello, G., and Roy, S., 2006, Exploiting mobility for energy efficient data collection in wireless sensor networks. *ACM/Springer Mobile Networks and Applications* 11, 3 (June), p.327-339.
- [22] Conti, M., Pelusi, L., Passarella, A., and Anastasi, G., 2008, *Adaptation and Cross Layer Design in Wireless Networks*. CRC Press, Chapter Mobile-relay Forwarding in Opportunistic Networks, p.389-418.
- [23] Wu, X., Brown, K. N., Sreenan, C. J., 2011, SNIP: A sensor node-initiated probing mechanism for opportunistic data collection in sparse wireless sensor networks. In *IEEE INFOCOM 2011*.
- [24] Lee, C.H., Kwak, J., Eun, D.Y., 2013, Characterizing link connectivity for opportunistic mobile networking: Does mobility suffice? In: *INFOCOM 2013*, p.2124-2132.
- [25] Bisht, M., Chhetri, S., 2013, A survey on the coverage of WSNs, *Int'l Journal of Advanced Research in Computer Science and Software Engineering*, Vol.3, Issue 3, Mar.2013, p.295-300.
- [26] KalaiMagal, R., Revathy, M., 2014, A survey on wireless sensor network protocols, *Int'l Journal of Innovative Science, Engineering & Technology (IJSET)*, Vol.1, Issue 6, p.548-562.
- [27] Lambrou, T.P., Panayiotou, C.G., Felici, S., Beferull, B., 2010, Exploiting mobility for efficient coverage in sparse wireless sensor networks, *Wireless Personal Communications*, Vol.54, Issue 1, July 2010, p.187-201.
- [28] Saukh, O., Hasenfratz, D., Thiele, L., 2014, Route selection for mobile sensor nodes on public transport networks, *Journal Ambient Intell Human Computing* (2014) 5, p.307-321.
- [29] Hoang, D.H., Strufe, T., Pham, T.N., Hoang, H.N., Nguyen, C.T., Tran, V.T., Schweizer, I., 2013, A smart data forwarding method based on adaptive levels in wireless sensor networks, *SenseApp, IEEE LCN 2013*, Sydney, Australia, Oct. 2013, p.100-103.
- [30] Kansal, A., Kaiser, W., Pottie, G., Srivastava, M., Sukhatme, G., *Reconfiguration Methods for Mobile Sensor Networks*, *ACM Trans. on Sensor Networks*, 2007, vol. 3, no. 4.
- [31] Wang, B., Beng, L.M., Ma, D., A survey of movement strategies for improving network coverage in wireless

- sensor networks, Computer Communications, August, 2009, Vol. 32, No. 13-14, p.1427-1436.
- [32] Hoang,D.H., Strufe,T., Le, D.Q., Bui, P.T., Pham,T.N., Thai,N.T., Le,T.D., Schweizer,I., Processing and Visualizing Traffic Pollution Data in Hanoi City From a Wireless Sensor Network, SenseApp, IEEE LCN 2013, Sydney, Australia, Oct. 2013, p.48-55.
- [33] Datta,N.N., Gopinath,K., 2006, A survey of routing algorithms for wireless sensor networks, Journal Indian Institute of Science, November-December 2006, 86, p.569-598.
- [34] Wu, X.,Brown,K.N., Sreenan,C.J, 2012, Data pre-forwarding for opportunistic data collection in wireless sensor networks. Proc. of 9th Int'l Conference on Networked Sensing Systems (INSS 2012), p.1-8.
- [35] Shah, K., Francesco, M., Anastasi, G., Kumar, M., 2011, A Framework for Resource-Aware Data Accumulation in Sparse WSNs, Computer Communications, Vol.34, Issue 17, p.2094–2103.
- [36] Hardisty, A., Goossens, B., Ambu, L.N., Gwilliam, C., Preece, A., 2012, K-Has: An Architecture for Using Local and Global Knowledge in Wireless Sensor Networks, The International Conference on Informatics and Applications (ICIA2012), p.331-341.
- [37] <http://www.libelium.com/development/waspmote/>
- [38] Ronald E.W., Raymond, H.M., Sharon, L. M., Keying, Y., 2011, Probability and Statistics for Engineers and Scientists (9th Ed.), Pearson Prentice Hall, Jan., 2011.

#### AUTHORS' BIOGRAPHIES



**Dang Hai Hoang**, Dr.-Ing. (1999), Dr.-Ing.habil. (2003) at the TU Ilmenau (Germany). Ministry of Information and Communications of Vietnam. A/Professor (2009) and invited lecturer at the Post and Telecommunications Institute of Technology. Research interests: telematics, wireless networks, communication networks,

protocol development and network security.



**Thorsten Strufe**, Dr.-Ing. (2007) at the TU Ilmenau (Germany). Professor for peer - to - peer networks at TU Darmstadt (Germany) from 2007-2014. visiting professor at University of Mannheim, Germany during 2011. Professor for Data Protection and Privacy at TU Dresden (Germany) since 2014. Research interests: peer-to-peer networks, large scale distributed systems and social networking services, privacy and resilience, data protection.



**Thieu Nga Pham**, Dr.-Ing. (2000) at the TU Ilmenau (Germany). Senior Lecturer at the Faculty of Information Technology, University of Civil Engineering, Vietnam. Research interests: optimization, control systems, fuzzy control, modeling and simulation, decision support systems, fuzzy decision making.



**Anh Son Nguyen**, MSc. at Viet Nam Mathematics Institute (2003), Lecturer at the Faculty of Information Technology, University of Civil Engineering, Vietnam (from 2001 to present). Research interests: Probability mathematic, numerical methods, optimization mathematics, fluid dynamics.



**Hong Ngoc Hoang**, BSc. at the Hanoi University of Science and Technology, Vietnam (2012). Currently, she is Master candidate at the TU Darmstadt (Germany). Research interests: wireless sensor networks, protocol algorithms, Internet and web-based services and applications.



**Immanuel Schweizer**, MSc. (2009), Dr.-Ing. (2012) at the TU Darmstadt (Germany). Head of Smart Urban Networks, FB 20 FG Telekooperation, TU Darmstadt (Germany). Research interests: energy-efficient wireless sensor networks, urban applications and virtual sensors, resilient data and service networks, mobile and complex networks.