Vol.2011-MBL-59 No.3
Vol.2011-CDS-2 No.3
2011/9/5

Timely In-Network Data Aggregation in Disaster Areas

Jovilyn Therese B. Fajardo,[†1]
Keiichi Yasumoto,[†1,†3] Naoki Shibata,[†2,†3]
Weihua Sun,[†1,†3] and Minoru Ito[†1]

A common characteristic in disaster areas is the partial or complete failure of the telecommunications infrastructure. This study proposes the use of mobile phones of the people to serve as sensing nodes for collecting disaster-related information in the affected area. Existing delay tolerant network (DTN) technologies allow propagation of disaster-related information created by the people to some extent, but it is difficult to achieve coverage of the area of interest (AoI) by the obtained messages in a short time due to the small data transfer capacity of a DTN. In this paper, a DTN-based data aggregation method is proposed that achieves the AoI coverage in minimal time by merging multiple messages (obtained through DTN) and their affected areas into a new message with the merged area. To keep sufficient details of the aggregated messages, the query sender can specify the maximum area size covered by each message. The evaluation method of the proposed method is also shown.

1. Introduction

According to a report released by the United Nations International Strategy for Disaster Reduction (ISDR), the total number of natural disasters worldwide reached 373 for the year 2010 and the average number from 2000 to 2009 is 387 [1]. Recently, an earthquake of magnitude 9.0 hit the eastern coast of Japan followed by a 15 to 20 m high tsunami that costs more than 10,000 deaths and billions worth of property. Due to severe damages of buildings and infrastructure, major lifelines like electricity, gas, and water were not available [2]. In disaster areas like this, one common characteristic is the partial or total failure of the telecommunications infrastructure [3] so the usual means of communication may not be applicable. However, access to information on the affected area is crucial at this time. Thus, an ad-hoc network may be deployed in an infrastructureless environment like a disaster area.

In an ad-hoc network, there are instances that the nodes are not connected to each other so information cannot be delivered from one node to another affecting network reliability. In networks where an end-to-end routing path between nodes is not guaranteed, a delay tolerant network (DTN) architecture may be utilized [4]. In this type of architecture, information is delivered via a store-carry-and-forward approach that is, information is temporarily stored at intermediate nodes for eventual delivery to the destination node. This approach incurs a certain delay in sending the information as expected but there is still a need to minimize this delay especially in networks that are time-constrained like a disaster area network. Also, minimizing delay indirectly improves the probability of message delivery [5].

In this study, a DTN-based data aggregation method is proposed to achieve timely collection of information from an area of interest (AoI) in a disaster area. Since it is a common scenario in disaster areas to have no communication infrastructure, a DTN-based method is adopted. People with mobile phones within or near the AoI vicinity serve as the nodes for the DTN. They create disaster-related messages at various places in the AoI and collect the messages by exchanging them among the nodes through short-range wireless communication like Bluetooth or WiFi. The proposed method aims to provide timely coverage of the AoI since it is important to quickly provide an accurate and coherent situational overview of the disaster area. However, due to the limited data transfer capacity of a DTN, it may not be possible to quickly collect all the messages created in the AoI with DTN. Thus, data aggregation is applied in order to reduce data size. To minimize the time for information delivery, this study uses the expected time of a node to reach the destination node, in which information is opportunistically forwarded to the node with the lower expected time. Data aggregation is then done depending on the aggregation granularity metric, which ensures the detailedness of the aggregated message by specifying the maximum area covered by one message. The metric and settings for the simulation-based evaluation of the proposed method is also shown.

†1 Graduate School of Information Science, Nara Institute of Science and Technology
†2 Department of Information Processing and Management, Shiga University
†3 Japan Science and Technology Agency, CREST

© 2011 Information Processing Society of Japan

2. Related Work

2.1 Existing Work

In disaster scenarios, it is of utmost importance that there is access to a wide range of information and such information is usually collected by sensors already deployed in the area. However, these predeployed sensors may be damaged in the aftermath of the disaster posing a problem for information collection in the disaster area. In these situations, the mobile phones of the people already present in the affected area may be used to gather information since current mobile phones have a rich set of embedded sensors such as accelerometers, global positioning system (GPS), microphones, cameras, and the like [6]. With the continuous advancement of mobile phones or smartphones in terms of computing and communication power, there is a shift in paradigm towards the use of people-centric sensing or the participatory sensing framework [7]. In this type of framework, humans are the center point of sensing and information collection is aimed towards the benefit of the society [8] [9] [10].

During the 2010 Haiti earthquake, the Twitter platform played an important role in terms of information creation. Users were able to broadcast 140-character messages (tweets) to other groups of users who subscribed to their accounts (followers) using the Tweak-the-Tweet (TtT) syntax intended for use during a disaster by those affected people [11]. This was also true in the recent Tohoku earthquake that happened in Japan, in which Twitter users posted tweets containing information regarding the situation in the affected area. Thus, in this study, the participatory sensing framework is adopted for information collection using the Twitter platform.

Networks in extreme environments, such as environmental habitats, disaster areas, and the like, are usually characterized by delay tolerant networks (DTNs). In [12], a DTN is implemented in order to gather information in extreme environments with no infrastructure. However, the data capacity of a DTN is limited thus, there is a need to reduce the data that is to be transmitted. In addition, in order to minimize energy consumption, only aggregates or statistics (summaries) of the collected information should be sent instead of sending all the information or raw data to another mobile node since mobile nodes have a limited battery life [13] [14]. One way of reducing energy consumption and data size is through in-network data aggregation of nodes. There are a number of studies that use in-network data aggregation to extend the lifetime of nodes and minimize energy consumption [15] [16] [17] but most aggregation schemes in literature only focus on reducing energy consumption. However, in-network data aggregation results in delay thus, some existing literature aims for an effective data aggregation subject to delay constraints [18] [19] [20] [21]. In most literature employing in-network data aggregation, the sensor nodes are static and the aggregation schemes are not applicable in applications using the participatory sensing framework.

2.2 Contribution

This study aims to achieve data aggregation in the AoI of a disaster area that minimizes delay of data delivery. It employs mobile nodes that are part of the participatory sensing framework. Unlike in the previous studies, the proposed method provides the possible minimum time for data collection from the AoI taking into account the aggregation granularity.

3. Data Aggregation Problem for Disaster Areas

3.1 Target Environment

Consider a disaster scenario during the early period of the recovery phase, which is the first 30 days from the onset of disaster recovery. It is vital in this period that information about the affected area is up-to-date but in most cases, the communication infrastructure is destroyed and it is difficult to collect information from the affected areas.

The disaster area is denoted by A_d. As shown in Fig. 1, A_d consists of links (roads) between special spots (e.g. evacuation centers, hospitals, and so on). The set of mobile nodes existing in A_d is denoted by U.

A user (or mobile node)[⋆1] in an evacuation center, a command center, or the like, with a fixed location is able to send a snapshot query on a particular area of interest denoted by $AoI \subseteq A_d$. The user and the corresponding mobile phone, which sent the query, is called the *sink*[⋆2]. Information is collected and aggre-

⋆1 The terms user, mobile node, and node are used interchangeably.
⋆2 The terms query sender or sink refers to the query node and vice-versa.

© 2011 Information Processing Society of Japan

Vol.2011-MBL-59 No.3
Vol.2011-CDS-2 No.3
2011/9/5

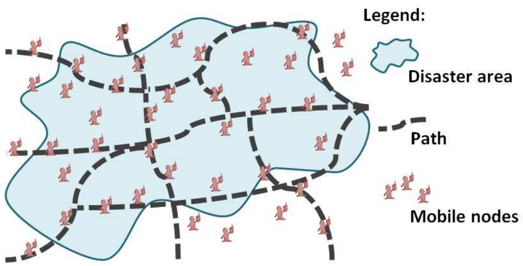
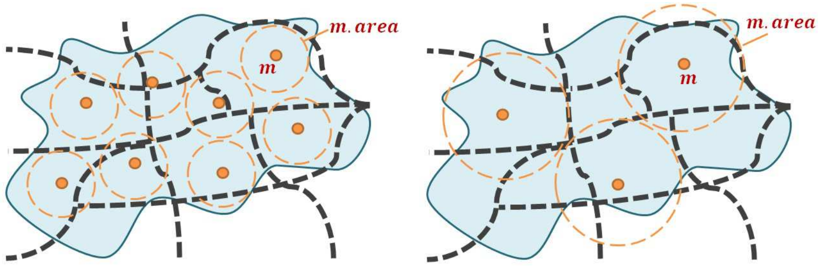
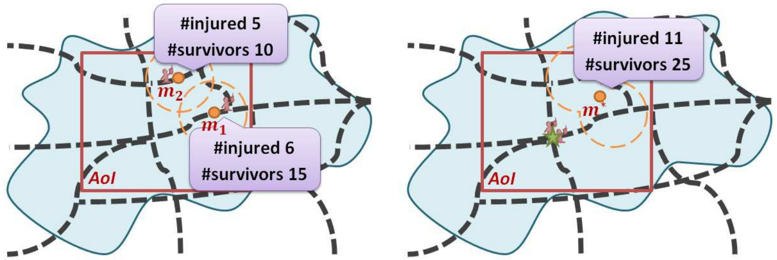


Fig. 1 Example of a Disaster Area



(a) $\phi = 50$ (b) $\phi = 100$

Fig. 2 Aggregation Granularity, ϕ



(a) Atomic Messages (b) Aggregated Message

Fig. 3 Example of Aggregating Messages

gated from different users who pass through the AoI. A query q is denoted by $\langle s_0, AoI, I_t, \phi \rangle$, where s_0 is the query node and AoI is the area of interest from which information type I_t is to be collected with an aggregation granularity ϕ.

Aggregation granularity ϕ is defined as the maximum area that each message is effective. Fig. 2(a) shows an aggregation granularity of 50, which corresponds that each aggregated message has an effective coverage of at most 50 m^2. However, in Fig. 2(b), it shows an aggregation granularity of 100, which corresponds that each aggregated message has an effective coverage of at most 100 m^2. Based on the figures, a lower ϕ value means that the user needs a more detailed data from the area while a higher ϕ value means that the user only wants a summary data from the area.

Each node receiving the query and existing in the AoI creates a message m depending on the specified I_t in the query, which may be statistical information on survivors, shelter capacity, and available resources such as food, first responders, and utilities. m has an effective area denoted by $m.area$. Nodes exchange messages with other nodes upon contact so that the set of collected messages covers the AoI and is delivered to the sink. As an example of the message format, the Tweek-the-Tweet (TtT) format, or commonly known as tweets, is supposed.

Both distributive (max, min, sum, count) and algebraic (average) data aggregation functions f_a are considered and only correlated messages with neighboring areas are combined. Fig. 3 shows an example of aggregating messages. In Fig. 3(a), two users create *atomic messages* m_1 and m_2 with information on the number of injured persons and survivors from different areas. Atomic messages are nonaggregated messages created by the user or the raw message. As shown in Fig. 3(b), when the two nodes come into contact, their messages are aggregated into one message m^* with information on the total number of injured persons and survivors from the two atomic messages. Also, the effective area of the atomic messages are merged and the merged area will then be the effective area of the aggregated message $m^*.area$.

3.2 System Model

For each node $s \in U$, its location at time t is denoted by $s.pos(t)$ determined either through GPS or estimated based on some other means. s is assumed to have a limited storage for the collected information and is capable of short-range wireless communication, Bluetooth or WiFi. A unit disc model is adopted, in

© 2011 Information Processing Society of Japan

which each s has the same communication range r_c and within this distance, s can transmit or receive data from other nodes. The available time while a contact (communication opportunity) occurs between two nodes s and s' is known as *contact duration* and the contact duration between s and s' starting from time t is denoted by $cd(s, s', t)$. The maximum transmission speed of the available short-range wireless communication is denoted by BW and the transferrable data amount of one contact is defined by $cd(s, s', t) \times BW$.

Each s is able to *create*, *send*, *receive*, or *aggregate* messages in different locations. The create action includes sensing information and creating the tweet message m containing the information that covers $m.area$. Each m is location-stamped denoted by $m.pos$ and time-stamped denoted by $m.t$.

The messages are aggregated depending on the aggregation granularity ϕ set by the query node. Let $\phi.a$ and $\phi.r$ denote the maximum area size and radius, respectively, that a message can cover. Also, let m_1 and m_2 denote atomic messages with a coverage area of $m_1.area$ and $m_2.area$, respectively, as shown in Fig. 3(a). m_1 and m_2 will only be aggregated to m^* if areas of the two messages have overlapping areas $m_1.area \cap m_2.area \neq \emptyset$, the merged area is smaller than the maximum area size $|m_1.area \cup m_2.area| \leq \phi.a$, and the farthest distance from the center point of the merged area is not greater than $\phi.r$.

Moreover, it is assumed that a query is issued at time 0 ($t = 0$). As time progresses, the node may move from one location to another or perform a certain action type $A_t = \{create, send, receive, aggregate\}$. Time is divided into time periods $T_0, T_1, \ldots$ with length P. Each period is also divided into two parts: *active interval* and *sleep interval*. In each time period, the first pP portion is assigned as the active interval and the remaining $(1-p)P$ portion as the sleep interval, where p is a system parameter and $0 < p < 1$. Each node is assumed to have an accurate clock wherein it sends a beacon message for finding other nodes only in the active interval. During the sleep intervals, each node turns its wireless communication device to sleep mode if there is no contact with other nodes in order to save energy consumption.

3.3 User Model

Any point in A_d can eventually be covered by some user of U, that is, the point will be visited by some user at some time in the future.

In addition, each user of U will eventually have direct or indirect contact with any other user of U in A_d, where the direct contact represents the situation of two users existing in their common communication range and the indirect contact is defined as the transitive closure of the direct contact.

3.4 Problem Definition

Given a query specifying the AoI and I_t, our problem is to derive the set of actions taken by each node of U that collects the set of messages M covering the AoI in the shortest possible time.

Each node s_i of U has action tuples Act_i, where $a_{i_j} = \langle s_i, A_t, M_i, t\rangle \in Act_i$ refers to the jth action performed by s_i. The tuple $\langle s_i, A_t, M_i, t\rangle$ represents the node s_i performing the action A_t on message set M_i at time t.

For every send action of node s_i, there is a corresponding receive action of node s_j. The send and receive actions must be performed during the contact duration, and the entire message must be transferred within the duration. Thus, the following equation must hold.

$$\forall s_i \in U, \quad \forall \langle s_i, send(s_j), M_i, t\rangle \in Act_i, \quad \exists s_j \in U \exists \langle s_j, receive(s_i), M_i, t'\rangle \\ \wedge \exists cd(s_i, s_j, t'') \\ \text{such that} \quad t'' \leq t \leq t' \wedge t, t' \in cd(s_i, s_j, t'') \wedge |M_i| \leq cd(s_i, s_j, t'') \times BW \quad (1)$$

Each message in M must have been created or aggregated by some nodes according to the following equation:

$$\forall m \in M, \quad IsCreated(m) \vee IsAggregated(m) \quad (2)$$

where

$IsCreated(m) \stackrel{def}{=} \exists s_i \in U \exists \langle s_i, create, \emptyset, t\rangle \in Act_i$ that creates m

$IsAggregated(m) \stackrel{def}{=} \exists s_i \in U \exists \langle s_i, aggregate,_a, t\rangle \in Act_i$ that creates m $\wedge$ $\forall m' \in M_a, \quad IsCreated(m') \vee IsAggregated(m')$.

The set of messages M delivered to s_0 must cover the entire AoI,

$$\bigcup_{m \in M} m.area \supseteq area(AoI), \quad (3)$$

and all messages of M must be received by s_0.

$$\left(\bigcup_{\langle s_0, receive, M_0, t\rangle \in Act_0} M_0\right) \subseteq M \quad (4)$$

Let M_0^* denote the set of messages that s_0 received and satisfy $M_0^* \supseteq M$. Also, let D denote the time when the receive action that completed the set M_0^*

© 2011 Information Processing Society of Japan

occurred.

Thus, given A_d, U, and a query q with s_0, AoI, I_t, and ϕ, the problem is defined as the minimum time data aggregation (MTDA) problem to decide the set of actions Act_i for each node s_i with the objective function defined as:

$$\textbf{minimize } D, \quad \textbf{subject to } (1) - (4)$$

4. Data Aggregation Algorithm

In this section, a greedy algorithm is presented to solve the MTDA problem described in Section 3.4. The NodeAction algorithm shown in Algorithm 1 is our main algorithm. This algorithm is executed at each node $s_i \in U$ independently of the other nodes and determines the action $A_t = \{create, send, receive, aggregate\}$ of s_i on its message set M_i over time. Each node runs the algorithm when it receives the query containing s_0, I_t, AoI, and ϕ. Moreover, each node also has knowledge of A_d. When a query is received, a node sends a beacon message to find its neighbor nodes during its active interval. During its sleep interval, it either creates a message, exchanges messages with its neighbor nodes, aggregates messages, or just sleeps if there are no neighbor nodes or there are no messages that can be aggregated.

As shown in Algorithm 1, it is assumed that the location of s_i is known and the query, which consists of the identity of the sink node s_0 and its position $s_0.pos(0)$, area of interest AoI, information type I_t, and aggregation granularity ϕ, has been received. During this instance, time t is set to 0 (line 1). Variables M_i and M_j are used to represent the message sets retained and received by s_i, respectively. Both message sets are initialized to be empty (line 2). At this point, it is also assumed that s_i is not in contact with any node as represented by a null s_j (line 3). s_i then enters into a loop performing lines 4–26 until reaching the predetermined deadline T, which may be equivalent to the time that the query is not needed anymore preventing s_i from going into an infinite loop.

The following subsections explain in detail the different parts of our main algorithm.

4.1 Neighbor Discovery

During the active interval in each time period, s_i sends a beacon message for neighbor node discovery (line 7). If s_i receives a beacon from node s_j successfully, s_i and s_j are assumed to be in contact (line 8) and s_j is added to the neighbor set N_i (line 9). All of the nodes within r_c of s_i is added to N_i.

Algorithm 1 NodeAction($s_i.pos(t), q$)

Input: Location of mobile node s_i at time t, $s_i.pos(t)$, Query q=$\langle s_0, AoI, I_t, \phi\rangle$
Output: Node action schedule of s_i

1: $t \leftarrow 0$
2: $M_i \leftarrow \emptyset$, $M_j \leftarrow \emptyset$ ▷ M_i is the current message set of s_i, M_j is the message set received by s_i
3: $s_j \leftarrow null$
4: **while** $t \leq T$ **or** M_i covering AoI is received by s_0 **do**
5: $N_i \leftarrow \emptyset$
6: **while** t is in active interval **do**
7: s_i sends a beacon message
8: **if** s_i receives a beacon message from s_j **then**
9: $N_i \leftarrow N_i \cup \{s_j\}$
10: **end if**
11: **end while**
12: **while** t is in sleep interval **and** $N_i \neq \emptyset$ **do**
13: **if** $s_i.pos(t)$ is within AoI **and** outside the covered area of M_i **then**
14: s_i creates m_i
15: $M_i \leftarrow$ Aggregate($M_i, \{m_i\}, \phi$) ▷ Algorithm 3
16: **end if**
17: **while** $N_i \neq \emptyset$ **do**
18: $s_j \leftarrow$ select one node from N_i at random
19: $M_j \leftarrow$ ExchangeMsg($s_i, s_j, s_0, M_i, ert(s_i), ert(s_j), ert(s_k), count$) ▷ Algorithm 2
20: **if** $M_j \neq \emptyset$ **then**
21: $M_i \leftarrow$ Aggregate(M_i, M_j, ϕ) ▷ Algorithm 3
22: **end if**
23: $N_i \leftarrow N_i - \{s_j\}$
24: **end while**
25: **end while**
26: **end while**

4.2 Message Creation

When t is within the sleep interval and s_i discovered neighbor nodes, s_i checks whether its current location is within the AoI and outside the covered area of its

© 2011 Information Processing Society of Japan

M_i (line 13). If it is true, it creates a message m_i (line 14) and m_i is aggregated with the other messages in M_i, which is explained in detail in Section 4.4 (line 15).

4.3 Message Exchange/Duplication

Node s_i then selects a random node s_j from N_i (line 18). Messages between nodes are exchanged (line 19), in which Algorithm 2 shows the ExchangeMsg algorithm. In Algorithm 2, the expected time of each contact node to reach the sink, $ert(s_i)$, is used to determine node action. This can be determined from the node's speed $v(s_i)$, moving direction, and the node's distance from s_0, $d(s_i, s_0)$ that can be computed from the positions of s_i and s_0 as well as the links of A_d. Assume that s_i is travelling at $v(s_i) = 1\ m/s$ to a spot (intersection) 30 m far away from the current location and the spot is 70 m from s_0, $ert(s_i) = \frac{30+70}{1} = 100\ s$. This means that it will take 100 s for s_i to come into the possible earliest contact with s_0.

Three ert values need to be known: $ert(s_i)$, $ert(s_j)$, and $ert(s_k)$. These values correspond to the ert of the current node s_i, the current neighbor node of s_i, and the node with the highest ert among the previous neighbor nodes of s_i to which M_i was copied, respectively. Depending on these values, the messages retained by s_i may be moved or copied and sent to s_j. In order to suppress the number of copied messages, a threshold *count* is introduced.

As an example, the ert may be divided into six range levels: I (< 10 minutes), II (10 minutes – 1 hour), III (1 hour – 5 hours), IV (5 hours – 1 day), V (1 day – 3 days), and VI (> 3 days), in which the ert level of a node s_i is denoted by $ert_{lvl}(s_i)$. When s_i is in contact with s_j, it checks whether the $ert(s_j)$ belongs to the lowest range level of ert (line 3) or the level in which the node is going to be contact with s_0 the soonest. If it is true, s_i moves its M_i to s_j (line 4). If not, s_i compares its $ert(s_i)$ with the $ert(s_j)$ and $ert(s_k)$ (line 5). Moreover, if the $ert(s_j)$ belongs to a range lower than both $ert(s_i)$ and $ert(s_k)$ that is, s_j is going to be in contact with s_0 at an earlier time than both s_i and s_k, s_i copies and sends M_i to s_j (line 6) while decrementing *count* (line 7). If not, s_i receives M_j from s_j (line 9) and the algorithm returns M_j (line 10).

Algorithm 2 ExchangeMsg$(s_i, s_j, s_0, M_i, ert(s_i), ert(s_j), ert(s_k), count)$

Input: Mobile nodes s_i, s_j, s_0, Message set M_i, Expected time to reach sink $ert(s_i), ert(s_j), ert(s_k))$, count $\triangleright s_i$ is the current node, s_j is the current neighbor node of s_i, s_k is the node with the highest ert among the previous neighbor nodes of s_i
Output: New message set M^*

1: Determine the ert_{lvl} of s_i, s_j, s_k
2: **if** $count \neq 0$ **then**
3:    **if** $ert_{lvl}(s_j) = 1$ **and** $ert_{lvl}(s_i) > 1$ **then**
4:       s_i moves (sends and removes) M_i to s_j
5:    **else if** $ert_{lvl}(s_j) < max(ert_{lvl}(s_i), ert_{lvl}(s_k))$ **then**
6:       s_i copies M_i to s_j
7:       $count \leftarrow count - 1$
8:    **else**
9:       s_i receives M_j from s_j
10:       **return** M_j
11:    **end if**
12: **end if**
13: **return** $\emptyset$

4.4 Message Aggregation

When a new message set M_j is received by s_i, it aggregates M_i and M_j depending on ϕ (Algorithm 1, line 21). Algorithm 3 describes the Aggregate algorithm. This algorithm aggregates or concatenates M_j with the local messages M_i depending on ϕ. Let M' contain the aggregated messages, which is set to $\emptyset$ initially (line 1). Aggregation is done until all of the message pairs of M_i and M_j are aggregated (line 2). However, only the message pairs that satisfy the following conditions are aggregated (line 3). First, the message pair m_i and m_j covers neighboring areas, $m_i.area \cap m_j.area \neq \emptyset$. Second, the area covered by m_i and m_j is less than or equal to $\phi.a$. Lastly, the Radius$(m_i.area \cup m_j.area)$, which is the farthest distance from the center point of $m_i.area \cup m_j.area$, is less than or equal to $\phi.r$. If these conditions are met, m_i and m_j are aggregated depending on the aggregation function f_a resulting to the aggregated message m_a (line 4). m_a is then added to the set of aggregated messages M' (line 5) and the algorithm returns M' (line 8).

© 2011 Information Processing Society of Japan

Algorithm 3 AGGREGATE(M_i, M_j, ϕ)

Input: Messages M_i and M_j, Aggregation granularity ϕ
Output: Aggregated message M'

```
M' ← ∅    ▷ M' is the aggregated message set
for each pair (m_i, m_j) ∈ M_i × M_j do
    if m_i.area ∩ m_j.area ≠ ∅ ∧ |m_i.area ∪ m_j.area| ≤ φ.a ∧ RADIUS(m_i.area ∪ m_j.area) ≤ φ.r then
        m_a ← AGGREGATEMSG(m_i, m_j)
        M' ← M' ∪ {m_a}
    end if
end for
return M'
```

Consider node s_1 in the AoI as shown in Fig. 3(a). s_1 receives a query with the following information: identity of s_0 and its position $s_0.pos(0)$, AoI, I_t, and ϕ at $t = 0$. Let I_t be the number of survivors with $\phi = 100$. During the sleep interval of s_1, it creates an atomic message m_1 with $m_1.area = 5\ m^2$. Then, as s_1 becomes active, it sends beacon messages to discover its neighbor nodes. Suppose that s_1 and s_2 comes into contact, s_1 compares its ert with the $ert(s_2)$. Let us suppose that the $ert(s_1)$ is within 10 minutes. Since this ert belongs to the lowest ert level as shown in Section 4.3, m_2 of s_2 is sent to s_1. Aggregation of the messages occurs since the following conditions hold: (1) $m_1.area \cap m_2.area \neq \emptyset$, (2) $|m_1.area \cup m_2.area| \leq \phi.a$, and (3) RADIUS$(m_i.area \cup m_j.area) \leq \phi.r$. The aggregated message m^* will then contain the information on the total number of survivors from the two messages m_1 and m_2.

5. Evaluation Method

5.1 Evaluation Criteria

In this study, the time for the aggregated messages to arrive the sink in response to a query is evaluated. The proposed algorithm, which opportunistically aggregates received messages and considers the estimated time of a node to reach the sink $ert(s_i)$, is compared to other methods. First, the proposed algorithm is compared to the method that does not aggregate messages but considers the ert of the nodes. Second, the proposed algorithm is compared to the method that aggregates the messages but does not consider the ert of the nodes. Lastly, the proposed algorithm is compared to the method that does not aggregate messages and does not consider the ert of the nodes.

5.2 Simulation Configuration

A custom simulator is used since only the contact times of nodes are needed and not its physical and link layer details. However, it is assumed that a node can discover and connect with another node at transmission range instantly. Simulation consists of 500 nodes placed uniformly at random over a two-dimensional plane with a $500\ m \times 500\ m$ area. The area looks like a grid, which represents the road network composed of horizontal and vertical streets. A random waypoint mobility model is adopted in which the nodes may only travel along the grid lines. At an intersection, the node may turn left, right, or straight depending on its destination. Each node moves at a random speed between $1 - 3\ m/s$ and broadcasts beacons every second.

A log-normal distribution is used for the node contacts since the inter-contact times between nodes or the time interval between two contacts of a node is based on a log-normal distribution according to the study by Conan et al. [22]. Their study used three different data sets of contact times between people, specifically students. This is similar to our study, which focuses on the mobility of people. Moreover, a node will only be able to send or receive a message at contact, in which only a certain size of message is exchanged depending on $cd(s, s', t) \times BW$.

The proposed algorithm is then evaluated based on the mentioned assumptions.

6. Conclusion

In this paper, the minimum time data aggregation problem in a disaster area was formulated and a greedy algorithm was proposed to solve the problem. The expected time of each contact node to reach the sink, $ert(s_i)$, was used to determine node action and aggregation was done depending on the aggregation granularity ϕ. Thus, the proposed algorithm ensures the timely delivery of the aggregated messages to the sink with a certain level of detailedness based on ϕ and a maximum coverage of the AoI.

Future work includes a simulation study on the proposed algorithm using the configuration described in Section 5. Moreover, in this study, only a single query is considered to be issued at a time. However, there may be instances that mul-

© 2011 Information Processing Society of Japan

tiple queries are issued by the users thus, future studies will address a multiple query scenario. Then, after a simulation-based evaluation, a more realistic scenario will be used, in which people actually collect information from an area and the proposed algorithm is incorporated. This will prove that the proposed algorithm can really be implemented in a real environment. Finally, improvements on the efficiency of the proposed algorithm will be done and the theoretical bound of its performance will be provided.

References

1) Guha-Sapir, D., Vos, F., Below, R. and Ponserre, S.: Annual Disaster Statistical Review 2010, Technical report, Centre for Research on the Epidemiology of Disasters - CRED (2011).
2) Miyamoto, H. K.: 2011 Tohoku Earthquake and Tsunami, Technical report, Miyamoto Earthquake + Structural Engineers (2011).
3) Townsend, A.M. and Moss, M.L.: Telecommunications Infrastructure in Disasters: Preparing Cities for Crisis Communications, Technical report, The Center for Catastrophe Preparedness and Response New York University (2005).
4) Fall, K.: A delay-tolerant network architecture for challenged internets, *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '03, pp.27–34 (2003).
5) Jain, S., Fall, K. and Patra, R.: Routing in a delay tolerant network, *SIGCOMM Comput. Commun. Rev.*, Vol.34, pp.145–158 (2004).
6) Lane, N., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T. and Campbell, A.: A survey of mobile phone sensing, *IEEE Communications Magazine*, Vol. 48, No. 9, pp.140–150 (2010).
7) Campbell, A.T., Eisenman, S.B., Lane, N.D., Miluzzo, E., Peterson, R.A., Lu, H., Zheng, X., Musolesi, M., Fodor, K. and Ahn, G.-S.: The Rise of People-Centric Sensing, *IEEE Internet Computing*, Vol.12, pp.12–21 (2008).
8) Gaonkar, S., Li, J., Choudhury, R.R., Cox, L. and Schmidt, A.: Micro-Blog: sharing and querying content through mobile phones and social participation, *Proceeding of the 6th international conference on Mobile systems, applications, and services*, MobiSys '08, pp.174–186 (2008).
9) Rana, R.K., Chou, C.T., Kanhere, S.S., Bulusu, N. and Hu, W.: Ear-phone: an end-to-end participatory urban noise mapping system, *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, IPSN '10, pp.105–116 (2010).
10) Ahmed, A., Yasumoto, K., Yamauchi, Y. and Ito, M.: Distance and Time Based Node Selection for Probabilistic Coverage in People-Centric Sensing, *Proceedings of IEEE SECON 2011*, pp.134–142 (2011).
11) Starbird, K. and Palen, L.: "Voluntweeters": self-organizing by digital volunteers in times of crisis, *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11 (2011).
12) Razafindralambo, T., Mitton, N., Viana, A., de Amorim, M. and Obraczka, K.: Adaptive deployment for pervasive data gathering in connectivity-challenged environments, *2010 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp.51–59 (2010).
13) Rasheed, F., Lee, Y.-K. and Lee, S.: Towards Using Data Aggregation Techniques in Ubiquitous Computing Environments, *Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops*, PERCOMW '06, pp.369–372 (2006).
14) Motegi, S., Nishimura, Y. and Yoshihara, K.: Communication Algorithm for Statistic Monitoring in People-Centric Sensing Networks, *International Conference on Networking*, Vol.0, pp.133–138 (2010).
15) Al-Karaki, J.N., Ul-Mustafa, R. and Kamal, A.E.: Data aggregation and routing in Wireless Sensor Networks: Optimal and heuristic algorithms, *Computer Networks*, Vol.53, No.7, pp.945–960 (2009).
16) Rajagopalan, R. and Varshney, P.: Data-aggregation techniques in sensor networks: a survey, *IEEE Communications Surveys Tutorials*, Vol.8, No.4, pp.48–63 (2006).
17) Kalpakis, K., Dasgupta, K. and Namjoshi, P.: Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks, *Computer Networks*, Vol.42, No.6, pp.697–716 (2003).
18) Akkaya, K., Younis, M. and Youssef, M.: Efficient aggregation of delay-constrained data in wireless sensor networks, *ACS/IEEE International Conference on Computer Systems and Applications*, Vol.0, pp.904–909 (2005).
19) Galluccio, L. and Palazzo, S.: End-to-End Delay and Network Lifetime Analysis in a Wireless Sensor Network Performing Data Aggregation, *IEEE Global Telecommunications Conference, 2009. GLOBECOM 2009.*, pp.1–6 (2009).
20) Hu, Y., Yu, N. and Jia, X.: Energy efficient real-time data aggregation in wireless sensor networks, *Proceedings of the 2006 international conference on Wireless communications and mobile computing*, IWCMC '06, pp.803–808 (2006).
21) Du, H., Hu, X. and Jia, X.: Energy efficient routing and scheduling for real-time data aggregation in WSNs, *Computer Communications*, Vol. 29, No. 17, pp. 3527–3535 (2006).
22) Conan, V., Leguay, J. and Friedman, T.: Characterizing pairwise inter-contact patterns in delay tolerant networks, *Proceedings of the 1st international conference on Autonomic computing and communication systems*, Autonomics '07 (2007).

© 2011 Information Processing Society of Japan