

# Power Consumption Characteristics of Autonomous Decentralized Clustering Based on Local Interaction\*

Ryo HAMAMOTO<sup>†a)</sup>, Student Member, Chisa TAKANO<sup>†b)</sup>, Kenji ISHIDA<sup>†c)</sup>, and Masaki AIDA<sup>††d)</sup>, Members

**SUMMARY** Mobile ad hoc networks (MANETs) consist of mobile terminals that directly connect with one another to communicate without a network infrastructure, such as base stations and/or access points of wireless local area networks (LANs) connected to wired backbone networks. Large-scale disasters such as tsunamis and earthquakes can cause serious damage to life, property as well as any network infrastructure. However, MANETs can function even after severe disasters have destroyed regular network infrastructure. We have proposed an autonomous decentralized structure formation technology based on local interaction, and have applied it to implement autonomous decentralized clustering on MANETs. This method is known to configure clusters that reflect the network condition, such as residual battery power and the degree of each node. However, the effect of clusters that reflect the network condition has not been evaluated. In this study, we configure clusters using our method, the back-diffusion method, and a bio-inspired method, which is a kind of autonomous decentralized clustering that cannot reflect the network condition. We also clarify the importance of clustering that reflects the network condition, with regard to power consumption and data transfer efficiency.

**key words:** *autonomous decentralized control, local-action theory, mobile ad hoc network, clustering*

## 1. Introduction

Large-scale disasters such as tsunamis and earthquakes can seriously damage or destroy network infrastructure. In the aftermath of such disasters, it is crucial to quickly gather information about the disaster and to promptly issue according evacuation orders. In order to realize these goals, failures in network functionality must be corrected as quickly as possible. Because confusion tends to abound immediately after a catastrophe, network protocols designed with the assumption of a normal environment may not satisfy operating requirements, and hence prompt network recovery may not be possible. This problem can be solved by creating an environment in which the remaining devices can operate effectively.

One solution is mobile ad hoc networks (MANETs) [1], where mobile terminals directly connect to one another to

create a communication network without recourse to network infrastructure, such as base stations and/or access points of wireless local area networks (LANs) connected to wired backbone networks. One of the most important issues in MANETs is to reduce the power consumption of the network in order to extend its life span. Research has been conducted on methods to reduce the power consumption of networks [2]–[4]. Clustering mechanisms for MANETs have been proposed for power saving and load balancing [5], [6]. They are important because they help reduce the power consumption of each node and extend the life of the entire network. These mechanisms use metrics such as the battery reserves [7] and the performance (e.g., processing speed, memory, and other parameters) [8] of each node in the network.

Various clustering methods have been studied [9]–[14]. However, all these methods require non-local information. In other words, they are not strictly autonomous decentralized algorithms, and global information regarding the state of the network is needed to obtain a globally optimal solution for the cluster structure. It is practically difficult to acquire global network information because information exchange is structurally limited in MANETs. This emphasizes the importance of autonomous decentralized cluster configuration methods, whereby globally optimum structures can be developed from local information, and can be used to execute traffic control, path control, and network resource management. Methods proposed in [15], [16] are based on local information, and include the well-known bio-inspired method, which uses a Turing pattern of reaction-diffusion equations. However, these clustering methods use seven parameters, and thus parameter setting is difficult for them. Moreover, clusters configured using the bio-inspired method cannot reflect the characteristics of the given network conditions (e.g., the distribution of the residual battery power of terminals, the position of power supplies, or the node degree of mobile terminals).

In the past, we have proposed a framework for a novel autonomous decentralized mechanism based on local interaction [17]. This framework is founded on the interplay between local interaction and the solution provided by a partial differential equation. As a specific example, we proposed the autonomous decentralized formation of structures with finite spatial size and showed the applicability of our method to autonomous clustering in MANETs [18]. Our clustering method, the back-diffusion method, can configure clusters using only local information about neighboring nodes.

Manuscript received January 7, 2014.

Manuscript revised May 19, 2014.

<sup>†</sup>The authors are with the Graduate School of Information Sciences, Hiroshima City University, Hiroshima-shi, 731–3194 Japan.

<sup>††</sup>The author is with the Graduate School of System Design, Tokyo Metropolitan University, Hino-shi, 191–0065 Japan.

\*An earlier version of part of this paper was presented at the Sixth International Workshop on Autonomous Self-Organizing Networks (ASON), 2013 [21].

a) E-mail: ryo@net.info.hiroshima-cu.ac.jp

b) E-mail: takano@hiroshima-cu.ac.jp

c) E-mail: ishida@hiroshima-cu.ac.jp

d) E-mail: aida@sd.tmu.ac.jp

DOI: 10.1587/transinf.2014PAP0001

Moreover, it allows each node to act flexibly based only on the information available to it, i.e., its own situation. Consequently, the back-diffusion method can yield clustering structures that reflect the characteristics of the network condition. [19] shows that the back-diffusion method (see [18]) can configure clusters faster than an existing method [15] by a factor of 10 or more. This means that communication can be recovered more quickly through the back-diffusion method. Furthermore, the clusters yielded by the back-diffusion method have approximately double the lifetime of those generated by the bio-inspired method in the control packet transfer phase [20]. Note that [20] makes no mention of the effect of the routing algorithm on the data packet transfer phase. Moreover, the effect of clusters that reflect the network condition has not been evaluated.

In this paper, we configure clusters using the back-diffusion method [18] and the bio-inspired method [15]. We then evaluate the effect which reflect the network condition for clusters from the point of view of both power consumption and data transfer efficiency [21]. In particular, we show the characteristics of power consumption using the metrics of both the first node die (FND) time [22] (FND time is the time period until the first failure of a node due to battery exhaustion) and the percentage of live nodes. Live nodes are those that have battery power remaining. Moreover, we evaluate the amount of the received packets by the sink node of the network, which is configured by the back-diffusion method and the bio-inspired method, to determine data transfer efficiency. Further, we show the performance characteristics of each method in terms of the number of nodes and node mobility. We use a hierarchical temporally ordered routing algorithm (Hi-TORA) [23] to evaluate the data transfer efficiency of the cluster. Note, however, that our aim here is not to find the most suitable routing algorithm for the back-diffusion method.

This paper consists of the following sections: in Sect. 2, we present the framework of our proposed autonomous decentralized structure formation technology and explain the bio-inspired method, which is an autonomous decentralized structure formation approach that uses Turing patterns. In Sect. 3, we describe Hi-TORA, the hierarchical routing algorithm used in this study. We evaluate the characteristics of the back-diffusion method in Sect. 4, and Sect. 5 provides our concluding remarks.

## 2. Clustering Method Based on Autonomous Decentralized Structure Formation

In this section, we provide an overview of the autonomous decentralized structure formation that uses back-diffusion drift. We also describe the bio-inspired method based on reaction-diffusion equations.

### 2.1 Overview of Back-Diffusion-Based Autonomous Decentralized Structure Formation Technology

We first introduce the autonomous decentralized structure

formation technology (back-diffusion method) for an one-dimensional network model to provide an intuitive understanding of the behavior of our method. Let the density function (density distribution) of a certain *quantity* at time  $t$  and position  $x$  be  $q(x, t)$ . The initial value of  $q(x, 0)$  can be considered as the metric, e.g., the residual battery power of each node in a MANET. Local behavior corresponds to changing the value of  $q(x, t)$  at each point,  $x$ , by controlling the information exchange between adjacent nodes. Note that,  $q(x, t)$  ( $t > 0$ ) is used for cluster configuration and is independent of residual battery power, whereas  $q(x, 0)$  reflects initial battery power. Therefore, changing the value of  $q(x, t)$  at each point does not mean that each node changes its own battery power. In an autonomous decentralized structure formation, flow  $J(x, t)$  (the operation rule that changes the value of  $q(x, t)$ ) is expressed as

$$J(x, t) = -c f(x, t) q(x, t) - c \sigma^2 \frac{\partial}{\partial x} q(x, t), \quad (1)$$

where the first and second terms denote the drift and diffusion terms, respectively. The temporal evolution of distribution  $q(x, t)$  that corresponds to this change is given by

$$\frac{\partial}{\partial t} q(x, t) = c \left( \frac{\partial}{\partial x} f(x, t) + \sigma^2 \frac{\partial^2}{\partial x^2} \right) q(x, t). \quad (2)$$

In the above equation,  $c$  ( $> 0$ ) denotes the rate of temporal evolution of  $q(x, t)$  and  $\sigma^2$  denotes the variance of the normal distribution on which the function converges.  $J(x, t)$  represents the extent of spatial movement of  $q(x, t)$ ; note that the total amount of battery power  $q(x, t)$  does not change over time. Equation (2) is a second-order differential equation. Therefore, this operation rule can be determined through interaction with adjacent nodes.

The introduction of  $f(x, t)$  eliminates the need to set a coordinate system in the network. As a more intuitive explanation, we consider the potential function  $\Phi(x, t)$  instead of  $f(x, t)$ :

$$f(x, t) = -\frac{\partial \Phi(x, t)}{\partial x}. \quad (3)$$

Choosing an appropriate  $\Phi(x, t)$  yields autonomous decentralized control that does not depend on a coordinate system. We now investigate the calculation of the drift term from  $q(x, t)$  at each point  $x$ . Since  $\Phi(x, t)$  should result in the maintenance of the distribution within a finite spatial size, contrary to the effect of diffusion,  $\Phi(x, t)$  is, after discrete time  $\Delta t$ , given by

$$\Phi(x, t + \Delta t) = -\left( q(x, t) - \gamma \frac{\partial^2 q(x, t)}{\partial x^2} \Delta t \right), \quad (4)$$

where  $\gamma > 0$ , and  $\Phi(x, t)$  is periodically renewed at intervals of  $\Delta t$ . The above equation is obtained by the sign inversion of the space derivative term in the diffusion equation. Note that Eq. (4) uses periodic time with interval  $\Delta t$  instead of  $dt$ . This is because the effect of the second term vanishes at the limit where  $dt$  approaches 0 [18]. The method of generating

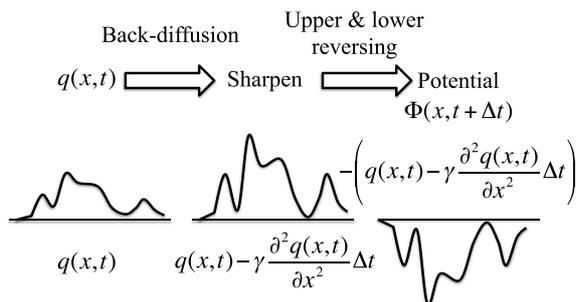


Fig. 1 Determining  $\Phi(x, t)$  according to back-diffusion.

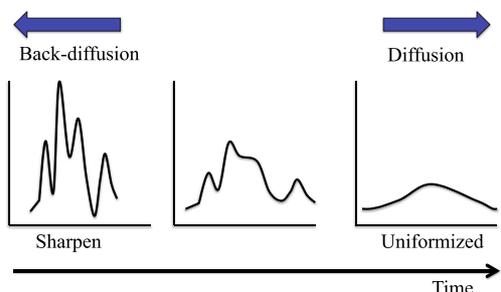


Fig. 2 Diffusion and back-diffusion.

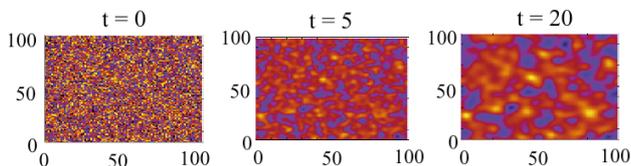


Fig. 3 Cluster structure formation by the back-diffusion method (initial state, after 5, 20).

potential  $\Phi(x, t + \Delta t)$  by using  $q(x, t)$  is shown in Fig. 1. The meaning of this figure is expressed as follows:

- We let the time progression of the diffusion phenomenon with diffusion coefficient  $\gamma$  be reversed (back-diffusion).
- We then reverse the distribution (up and down) and regard the completed distribution as the potential.

Because of the effect of the drift term, including the potential, the peak of  $q(x, t)$  is emphasized and the shape of the distribution is sharpened (Fig. 2). The effect of the diffusion term, on the other hand, is to flatten the distribution. Figure 3 shows an example of the structure with a finite spatial size that can be formed by balancing one effect against the other. In Fig. 3, the peak of  $q(x, t)$  is the representative node of the cluster and the minimum value of the distribution is the boundary of the cluster.

Moreover, [18] has shown that our approach can be applied to an arbitrary network as well as a one-dimensional network because diffusion and back-diffusion can be defined based only on the states of the node and nodes adjacent to it. We now concretely describe the local action rule in the network. First, the set of nodes that are adjacent to node  $i$  (the

set of nodes that are linked to node  $i$ ) is defined as  $N_i$ . We also discretize time and set the time interval of autonomous control as  $\Delta t$ . In the following, we describe the action rule for spatial discretization that corresponds to nodes in the network, and time discretization that corresponds to control timing. The distribution  $q_i(t_k)$  at time  $t_k$  ( $:= k \times \Delta t$ ) at node  $i$  changes (after  $\Delta t$ ) as follows:

$$q_i(t_{k+1}) = q_i(t_k) - \Delta t \sum_{j \in N_i} (J_{i,j}^{\text{drift}}(t_k) + J_{i,j}^{\text{diff}}(t_k)), \quad (5)$$

where  $J_{i,j}^{\text{drift}}(t)$  and  $J_{i,j}^{\text{diff}}(t)$  are variations created by the drift effect and the diffusion effect within each unit time, respectively.  $J_{i,j}^{\text{drift}}(t)$  and  $J_{i,j}^{\text{diff}}(t)$  satisfy the following equations:

$$J_{i,j}^{\text{drift}}(t_k) := \begin{cases} c f_{i,j}(t_k) q_i(t_k), & (f_{i,j}(t_k) > 0), \\ -c f_{j,i}(t_k) q_j(t_k), & (f_{j,i}(t_k) > 0), \end{cases} \quad (6)$$

$$f_{i,j}(t_k) := -(\Phi(j, t_k) - \Phi(i, t_k)), \quad (7)$$

$$J_{i,j}^{\text{diff}}(t_k) := -\sigma^2 (q_j(t_k) - q_i(t_k)). \quad (8)$$

Due to the drift effect, the distribution moves in direction  $i \rightarrow j$  ( $j \rightarrow i$ ) in the case of  $f_{i,j}(t_k) > 0$  ( $f_{i,j}(t_k) < 0$ ). Here, equation  $f_{i,j}(t_k) = -f_{j,i}(t_k)$  holds. The variation is proportional to the product of the velocity of the drift  $f_{i,j}(t_k)$  ( $f_{j,i}(t_k)$ ) and  $q_i(t_k)$  ( $q_j(t_k)$ ) in node  $i$  ( $j$ ). The above description is formalized by Eq. (6). The variation due to the diffusion effect is proportional to the gradient of the distribution in Eq. (8).

We now explain how to determine the potential  $\Phi_i(t_k)$  that is related to drift. The potential value  $\Phi_i(t_{k+1})$  of node  $i$  at time  $t_{k+1}$  is decided by the value of the distribution  $q_i(t_k)$  and the back-diffusion of  $q_i(t_k)$  as follows:

$$\Phi_i(t_{k+1}) = - \left( q_i(t_k) - \gamma \Delta t \sum_{j \in N_i} (J_{i,j}^{\text{back}}(t_k) - J_{j,i}^{\text{back}}(t_k)) \right), \quad (9)$$

where  $J_{i,j}^{\text{back}}(t_k)$  is generated by the back-diffusion of  $q_i(t_k)$ , and is the variation in a unit time period in the direction of node  $i \rightarrow j$ . The variation  $J_{i,j}^{\text{back}}(t_k)$  is given by

$$J_{i,j}^{\text{back}}(t_k) = \begin{cases} q_j(t_k) - q_i(t_k), & (\Delta q_i^{\text{max}}(t_k) \\ & = q_j(t_k) - q_i(t_k)), \\ 0, & (\text{otherwise}). \end{cases} \quad (10)$$

$$\Delta q_i^{\text{max}}(t_k) := \max \left( \max_{j \in N_i} (q_j(t_k) - q_i(t_k)), 0 \right). \quad (11)$$

$\Delta q_i^{\text{max}}(t_k)$  is the difference between the distribution value of node  $i$  and the distribution value of the adjacent node  $j$  in the direction of the steepest ascent from node  $i$ . In the above-mentioned action rule for discretization in the network, local interaction is guaranteed because the summations of nodes  $j \in N_i$  in the above equations involve only the nodes adjacent to node  $i$ .

To achieve the above-mentioned control, it is necessary to exchange information about the values of the distribution

$q_j(t_k)$  for the adjacent nodes at the interval of  $\Delta t$ . The complexity of this information exchange does not depend on network size because the communication range is only one hop. Therefore, it is scalable against the number of nodes.

Our method takes into account only the initial battery capacity of terminals for clustering. A better cluster can certainly be configured if updated information about the network, such as the battery capacity of each terminal, is used for our clustering. We have in the past proposed a vector distribution *quantity* [24], [25] for this purpose. By using this method, it is possible to reflect the network condition (e.g., the residual battery power of each node) for cluster structures at all times. However, our aim here is to evaluate the effect of the clustering which reflects or not the network condition from the point of view of both power consumption and data transfer efficiency. Thus, we do not use the vector process in our evaluation. The vector process is necessary for practical use, but we have positioned this paper as a basic study. In future research, we will evaluate the effect which reflect the network condition for clusters using vector process.

## 2.2 Bio-Inspired Method Based on Reaction-Diffusion Equations

The bio-inspired method [15] is an autonomous decentralized structure formation approach that uses Turing patterns. Invented by Alan Mathison Turing, a Turing pattern is a mathematical method used to describe pattern formation on the bodies of animals. A Turing pattern is formed through reaction-diffusion equations (Eqs. (12) and (13)). Each node in the network contains two factors, activator  $a$  and inhibitor  $h$ , and these values change over time according to the following differential equations:

$$\frac{\partial a}{\partial t} = \frac{ca^2}{h} - \mu a + \rho_0 + D_a \nabla^2 a, \quad (12)$$

$$\frac{\partial h}{\partial t} = ca^2 - \nu h + \rho_1 + D_h \nabla^2 h, \quad (13)$$

where  $c$ ,  $\rho_0$ , and  $\rho_1$  are parameters that enhance the effects of the activator and the inhibitor, and  $\mu$  and  $\nu$  are parameters that reduce the effects of the activator and the inhibitor, respectively. Moreover,  $D_a$  and  $D_h$  are parameters that describe the rate of diffusion of the activator and the inhibitor, respectively. From Eqs. (12) and (13), we see that the spatial pattern appears gradually over time (Fig. 4), and the peak of the created pattern denotes the representative node of the cluster whereas the extent of the pattern denotes the extent of the cluster. [26] presents research on parameter design for systems based on reaction-diffusion equations. It is, in

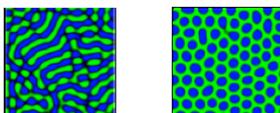


Fig. 4 Examples of Turing pattern.

general, significantly more difficult to design parameters for the bio-inspired method than the back-diffusion method because the former has many more parameters.

## 3. Hierarchical Routing Hi-TORA

In this section, we describe Hi-TORA [23], a hierarchical routing algorithm that we apply to MANETs in this paper.

### 3.1 Overview of Hi-TORA

Hi-TORA is a hierarchical routing (cluster-based routing) scheme used in MANETs. Hi-TORA has two phases according to the domain in which the routing algorithm operates: the intra-cluster (within a cluster) and the inter-cluster (among clusters) phases. The traditional link-state-type routes (shortest path routes) are provided for intra-cluster routing. For inter-cluster routing, on the other hand, Hi-TORA applies a TORA (Temporally Ordered Routing Algorithm) [27]. It regards one cluster as a virtual node in this case. In this way, Hi-TORA calculates the routing path from the source node to the sink node based on the two phases, when the sink node belongs to a different cluster from the source node. We briefly explain the routing algorithm for each phase below.

### 3.2 Routing for the Intra-Cluster Phase

For the intra-cluster phase, Hi-TORA executes the link-state routing algorithm, which finds the shortest path between the source node and the sink node using Dijkstra's algorithm. When the source node and the sink node are in the same cluster, the source node sends the data to the sink node through the shortest path. Otherwise, the source node sends the data to the boundary node adjoining the neighboring cluster on the path to the sink node. If two or more boundary nodes exist, the source node chooses as the boundary node the one that has the a lower numerical node ID.

### 3.3 Routing for the Inter-Cluster Phase

Hi-TORA adopts TORA to execute on-demand path calculations for the routing algorithm among the clusters because TORA is highly adaptable to node mobility. TORA establishes the DAG (directed acyclic graph) in which the sink node is regarded as the root. TORA then determines the logical direction of the links to the sink node by using the DAG, where the calculation of the direction of each link uses a metric called "*height*". TORA controls the entire network to maintain multiple paths between the source and sink nodes. Thus, the overhead of the control packets for TORA inevitably increases with the number of nodes. The overhead can be reduced by assuming that each cluster is a virtual node, so that *height* can be set not to each link but to each cluster.

When a source node wants to communicate with a sink node, the source node sends a request to send (RTS) packet

to the sink node. The *height* is then set for all clusters that receive the packet. The destination cluster's *height* is set to 0. The closer a cluster is to the source node, the larger the value of *height* set for it. Thus, the *height* of the source cluster, to which the source node belongs, is the highest value in the network, and the *height* of the destination cluster, to which the destination node belongs, is the lowest value. The data packets generated by the source node are forwarded through the clusters with lower *height*.

Note that the clusters on which Hi-TORA in [23] operates are configured by the clustering method in [28]. The evaluations of this paper compare the back-diffusion method and the bio-inspired method as clustering methods.

#### 4. Evaluation

In this section, we clarify the effect which reflects the network condition for clusters from the point of view of both power consumption and data transfer efficiency while showing the results of a comparative evaluation of the temporal evolution of the live node percentage, the FND time, and the amount of data received by the sink node.

Our evaluations do not take into account the physical layer and the MAC (media access control) layer. This is because the aim of our study is to evaluate the effect which reflects the network condition for clusters. In other words, we evaluate the effect of the mismatch between the network condition and the configured clusters. Therefore, we focus on the importance of the clustering that reflects network conditions rather than evaluations of accurate power consumption. Power consumption and data transfer efficiency are merely metrics for the evaluation of the effect which reflects the network condition for clusters. In future research, we intend to evaluate the back-diffusion method by considering the effects of real environments, such as protocols and real battery models, using vector process.

##### 4.1 Characteristics of Power Consumption and Data Transfer Efficiency

We examine the power consumption characteristics of data transfer for each method in order to clarify the importance of clustering that reflects network conditions. The network is an UDG (unit disk graph) of 1,000 m × 1,000 m and is constructed by 101 nodes. An UDG is a type of intersection graph based on circles of the same size (Fig. 5). An UDG is suitable as a model of an ad hoc network because it can describe various radio transmission ranges between nodes, but cannot reflect more realistic wireless network characteristics such as packet collisions. We will address this issue in future work. The network model has a torus topology to exclude the influence of the boundary. In other words, each node emerges on the other side when it crosses the boundary. Note that we assume mobile nodes in MANETs. Node movement is set by a random direction model every second, and the average velocity of each node is 1.3 m/s. The random direction model is a kind of famous mobility

model [29]. This model is used for the simulating the movement of users in a mobile wireless networks.

One of the nodes is taken to be a sink node and is set at the center of the network. The position of the sink node is (500 m, 500 m). Note that the sink node is fixed. It assumes that the sink node is a server or a base station. The established servers and base stations can secure battery supply. Therefore, the sink node does not consume its own battery. If the sink node consumes battery, it may become immediately unusable by the mobility pattern of each node. This is because the number of nodes adjacent to the sink node may suddenly increase. As a result, maximum data transfer efficiency cannot be measured, and the analysis of results becomes difficult.

Figure 6 shows an example of the distribution for initial battery power of each node. In Fig. 6, the color represents the battery power of each node. We assume that the initial distribution of each node is identical to the random initial battery power. Other conditions are shown in Table 1. Note that we assume that the user carries his or her terminal, so that node movement does not increase battery consumption, and the battery consumption for each node occurs by packet transfer and the constant processing of each representative node. Therefore, our battery model does not capture its nonlinear characteristics. Furthermore, this study assumes a simplified power consumption model. Specifically, we assume that the same amount of power is consumed for transmitting packets and for receiving them. Naturally, considering a more realistic power consumption model is a very important issue. However, this is a basic study about the effect of the clustering which reflects or not the network condition,

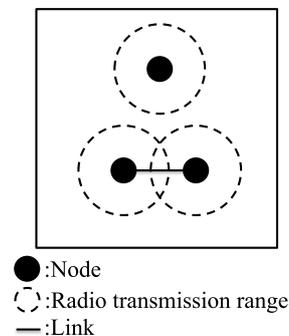


Fig. 5 Unit disk graph network.

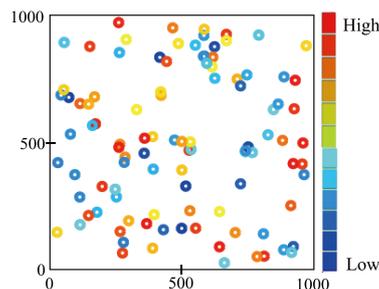


Fig. 6 The distribution of initial battery power.

**Table 1** Experimental environment.

Network	Unit Disk Graph (UDG) of 1,000 m × 1,000 m
Number of nodes	101 (One of which is sink node)
Transmission range of node	250 m
Initial battery power of node	uniform random numbers in the range [5, 15] × 1 J
Battery consumption	1 μJ/bit (transmission) 0.1 μJ/s (processing of representative node)
Simulation time	20,000 s
Number of Simulations	30

**Table 2** Parameters of the back-diffusion method.

$c$	$\sigma^2$	$\gamma$
0.05	0.5	0.15

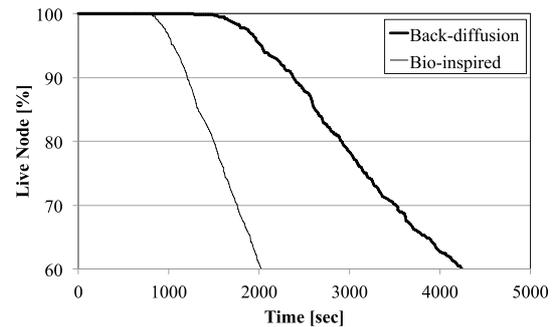
**Table 3** Parameters of the bio-inspired method.

$c$	0.001
$\mu$	0.05
$\nu$	0.1
$\rho_0$	0.04
$\rho_1$	0.02
$D_a$	0.00122273
$D_h$	0.00180619

and our future work will evaluate the characteristics of a network given a realistic power consumption model. Note that clusters are formed from  $t = 0$  s using each method, and the routing, using Hi-TORA, and the transmission of data packets starts at  $t = 1,001$  s. In addition, the results show the average of 30 simulations.

We now describe the routing procedures. First, each node generates data packets (1 pkt = 1.5 kB) with time intervals that obey an exponential distribution with  $\lambda = 0.005$ . The value of  $\lambda$  is set according to [30]. The source node sends data packets to the sink node through the multi-hop path computed by Hi-TORA. Once established, the routing path is maintained until the transmission of packets is completed. For simplicity, we assume that the sink node can receive multiple packets simultaneously; that is, packet collision is not considered. If the path to the sink node cannot be found, the source cancels packet transmission. Only the sink node has a main power supply, and thus its battery power is never exhausted. The parameters of the back-diffusion method and the bio-inspired method are shown in Tables 2 and 3, respectively. The parameters of the back-diffusion method are adjusted so that it yields the same number of clusters as the bio-inspired method at  $t = 1,000$ s. For cluster formation, adjacent nodes exchange one control packet per second (1 pkt = 8 bytes). Routing-control packets (1 pkt = 8 bytes) are used in inter-cluster communication through Hi-TORA to set the *height* of each cluster on the path.

Figure 7 shows the temporal evolution of the percentage of live nodes for both methods. The horizontal axis represents time and the vertical axis is the percentage of live nodes. We can see that the back-diffusion method offers a longer survival time than the bio-inspired method. One reason is the difference in the amount of control information used when configuring the cluster. Another is that the back-



**Fig. 7** Temporal evolution of percentage of live nodes.

**Table 4** FND time.

back-diffusion	bio-inspired
1,565 s	1,031 s

**Table 5** Standard deviation of FND time.

back-diffusion	bio-inspired
159 s	107 s

diffusion method forms clusters according to the distribution of initial battery power. On the other hand, clusters formed by the bio-inspired method are arranged at equal intervals regardless of the distribution of initial battery power. This increases the number of nodes with exhausted batteries and the live node percentage decreases with time.

Tables 4 and 5 show the FND times of each method and the standard deviation of the results respectively. We see from the result that clusters configured by the back-diffusion method have a longer FND time, by 534s, than those configured by the bio-inspired method. However, the standard deviation of the back-diffusion method is greater than that of the bio-inspired method, by 52 s. The difference in the shape of the configured cluster and the number of control packets used influences these results for the same reasons as in the case of live nodes.

Tables 6 and 7 show the amount of data received by the sink node and its standard deviation for the back-diffusion method and the bio-inspired method, respectively. The results show that the sink node can gather more information, by 1,168 packets, in clusters configured using the back-diffusion method than in those configured by the bio-inspired method. However, the standard deviation of data received by the sink node in the cluster configured by the back-diffusion method is greater than that of the bio-

**Table 6** Amount of total received data.

back-diffusion	bio-inspired
1,978 pkt	810 pkt

**Table 7** Standard deviation of total received data by the sink node.

back-diffusion	bio-inspired
99 pkt	84 pkt

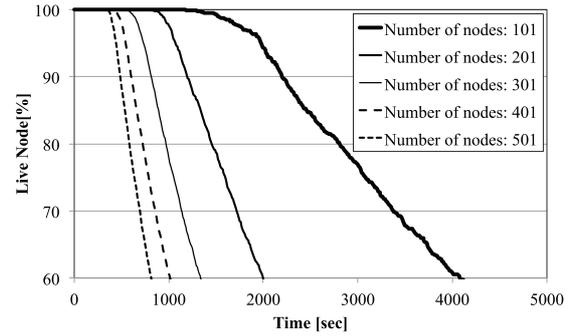
inspired method, as was the case with FND time. We can see from these results that the back-diffusion method can reduce power consumption and permit the transmission of significantly more data than the bio-inspired method. Therefore, reflecting the network condition for clusters is effective from the point of view of both the power consumption and the data transfer efficiency.

4.2 Performance Characteristics with Varying Number of Nodes and Node Mobility

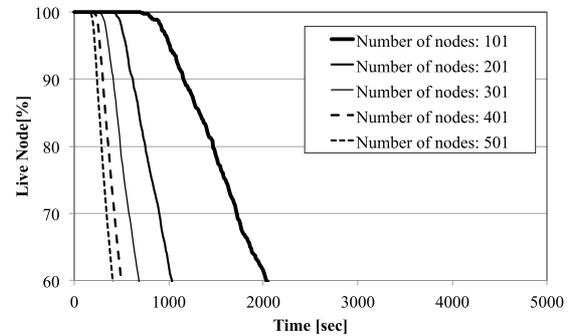
In this section, we evaluate the characteristics of power consumption and data transfer efficiency when we change the number of nodes and their average mobility speed. The network model, the routing procedures and the parameters of clustering are same as those in Sect.4.1. The number of nodes and the mobility of nodes are set to (101, 201, 301, 401, 501) and (1 m/s, 5 m/s, 20 m/s), respectively. The outcome measures are the percentage of live nodes, FND the time, and the number of packets received by the sink node.

First, we describe the behavior of the characteristics if the number of nodes is varied. Figures 8 and 9 show the variation with time in the percentage of live nodes for varying number of nodes using the back-diffusion method and the bio-inspired method, respectively. In each figure, the horizontal axis represents time and the vertical axis represents the live node percentage. These figures show that as the number of nodes increases, the percentage of live nodes decreases more rapidly. This is because each node has more adjacent nodes in this case, thus leading to an increase in the number of control packets sent/received by each node. These results also show that the back-diffusion method reduces the rate at which the percentage of live nodes decreases compared to the bio-inspired method. This is because the back-diffusion method can configure clusters reflecting the network condition.

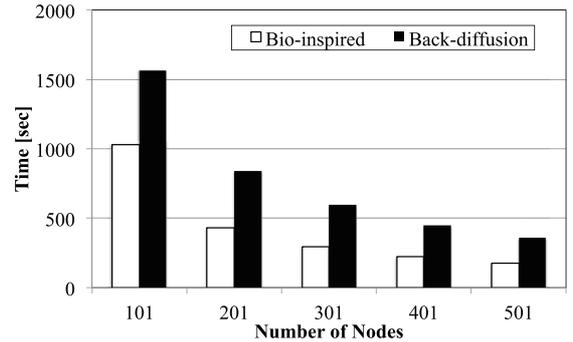
We now discuss the FND time for each method. In Fig. 10, the horizontal axis represents the number of nodes, and the vertical axis is time. This result shows that increase in the number of nodes decreases the FND time. This is for the same reason as the decrease in the percentage of live nodes. The back-diffusion method yields longer FND times than the bio-inspired method regardless of the number of nodes. Moreover, Fig. 11 shows the relationship between the number of nodes and the standard deviation of the FND time. We see that an increase in the number of nodes decreases the standard deviation. This is because a node im-



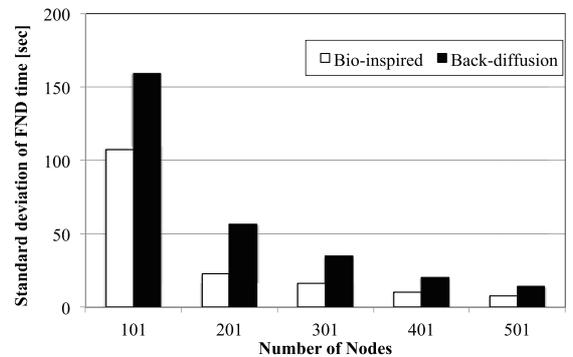
**Fig. 8** Relationship between the percentage of live nodes and the number of nodes (back-diffusion).



**Fig. 9** Relationship between the percentage of live nodes and the number of nodes (bio-inspired).



**Fig. 10** Relationship between the FND time and the number of nodes.



**Fig. 11** Relationship between the standard deviation of FND time and the number of nodes.

mediately runs out of battery regardless of its initial position when the number of nodes increases. From the above results, we see that our method can better reduce power consumption than the bio-inspired method, even if the number of nodes varies.

Figures 12 and 13 show the number of data packets received by the sink node and its standard deviation for varying numbers of nodes, respectively. These results show that increasing the number of nodes decreases the amount of data received by the sink node as well as its standard deviation for both methods. However, the back-diffusion method can transmit more data packets than the bio-inspired method. This is because our clustering method allows nodes to live longer (the percentage of live nodes is high) and the route to the sink node is maintained in the network. From these results, we conclude that taking into account the network condition for clustering is very important from the viewpoint of power consumption and data transfer efficiency, even if the number of nodes increases.

Next, we now present the experimental results of the relation between performance characteristics and node mobility. Figures 14 and 15 show the temporal evolution of the percentage of live nodes for various node mobility values for each method. These results show that the rate at which the percentage of live nodes decreases is independent of average node mobility speed. This is because the number of adjacent nodes for each node changes little in the random di-

rection model, even if the average node velocity increases. As a result, the number of control packets sent/received by each node does not so change. The back-diffusion method again yields a slower fall in the percentage of live nodes than the bio-inspired method.

Figure 16 shows the relationship between the FND time and average node velocity, and Fig. 17 represents the standard deviation of the FND time for different average node velocities. These results show that the FND time and its standard deviation are not so dependent on average node velocity. From these results, we see that the back-diffusion method yields longer FND times than the bio-inspired method, regardless of node velocity.

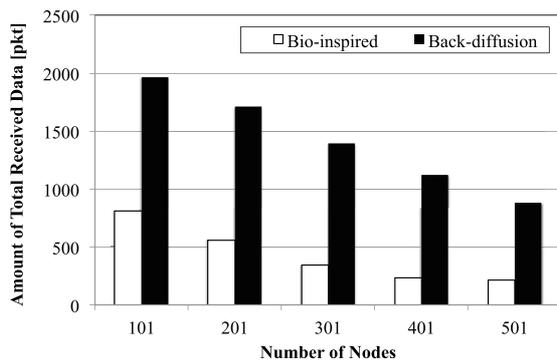


Fig. 12 Relationship between the amount of received data and the number of nodes.

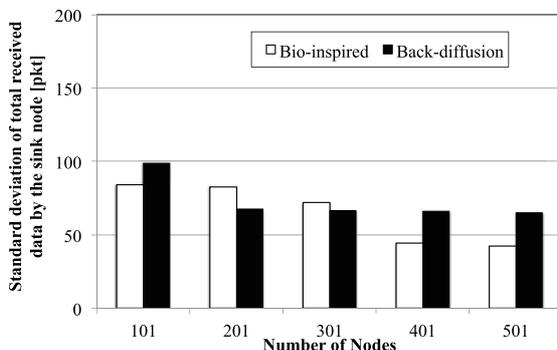


Fig. 13 Relationship between the standard deviation of the amount of received data and the number of nodes.

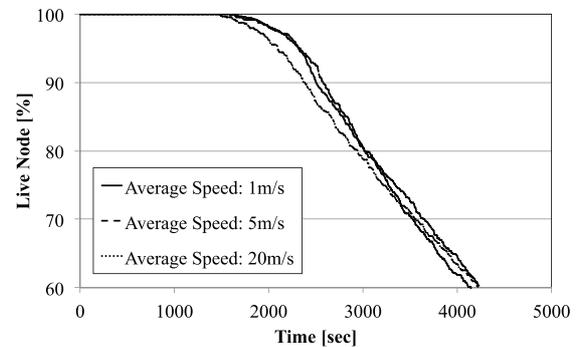


Fig. 14 Relationship between the percentage of live nodes and the average node velocity (back-diffusion).

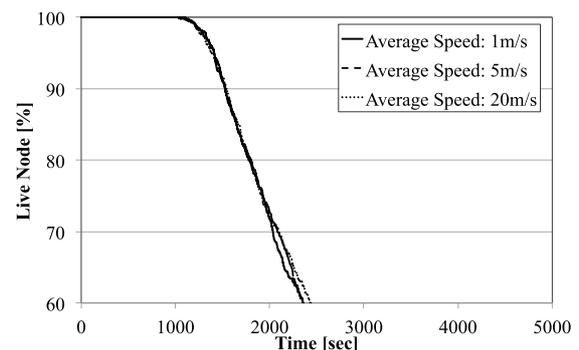


Fig. 15 Relationship between the percentage of live nodes and the average node velocity (bio-inspired).

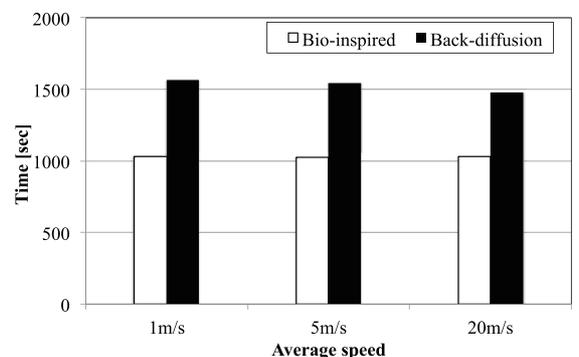


Fig. 16 Relationship between the FND time and the average node velocity.

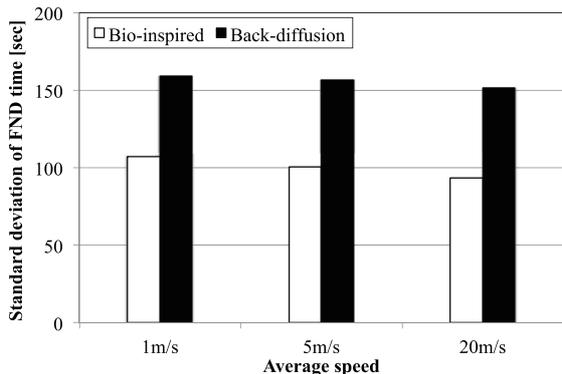


Fig. 17 Relationship between the standard deviation of the FND time and the average node velocity.

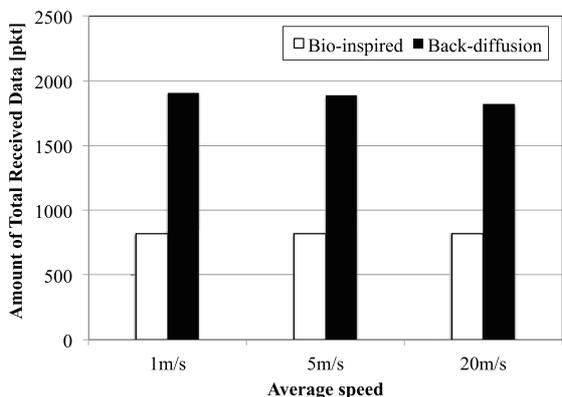


Fig. 18 Relationship between the amount of received data and the average node velocity.

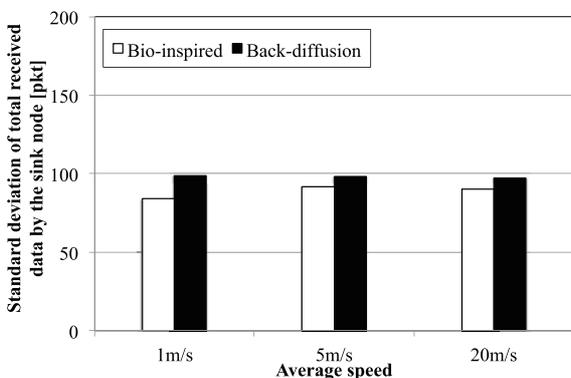


Fig. 19 Relationship between the standard deviation of the amount of received data and the average node velocity.

Figure 18 shows the amount of data received by the sink node at different node velocities. We see from the figure that the number of data packets collected by the sink node does not change, even if average node velocity increases, because the number of live nodes changes only slightly as a consequence. Figure 19 shows the standard deviation of the amount of data received by the sink node. From Fig. 19, we see that the standard deviation is almost independent of average node velocity. Note that our clustering method can

transmit more data regardless of node velocity. These results show that reflecting the network condition for clusters is crucial, even if the node velocity increases.

### 5. Conclusion

In past research, we proposed an autonomous decentralized clustering technology based on local interaction and used it to realize clustering in MANETs. In this study, we compared the back-diffusion method to a bio-inspired method based on the reaction-diffusion equation in order to evaluate the effect of clustering on network condition in terms of power consumption and data transfer efficiency. We used Hi-TORA as our routing algorithm, which offers one kind of cluster-based routing for ad hoc networks. Our evaluation focused on the temporal change in the percentage of live nodes, the FND time, and the amount of the data received by the sink node. We found that the clusters yielded by the back-diffusion method are superior in all respects to those generated by the bio-inspired method. This means that reflecting the network condition for clusters is effective from the point of view of both power consumption and data transfer efficiency. Therefore, the back-diffusion method can configure clusters that can operate for longer periods of time, and thus can help maintain communication after a disaster for longer periods.

Our future research will involve investigating the compatibility of our method with routing algorithms other than Hi-TORA, and enhancing the flexibility of the back-diffusion algorithm. Moreover, we will evaluate the back-diffusion method with vector processes in real environments.

### Acknowledgments

This research was partly supported by Grant-in-Aid for Scientific Research (C) No. 24500091 (2012–2014) and No. 24560471 (2012–2014) from the Japan Society for the Promotion of Science and project research grants from the Graduate School of Information Sciences, Hiroshima City University.

### References

- [1] C.E. Perkins, ed., Ad hoc networking, Addition Wesley, 2000.
- [2] A. Abbas, J. Bahi, and A. Mostefaoui, "Improving wireless ad hoc networks lifetime," Proc. IEEE Int. Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), 2006.
- [3] W. Feng, H. Alshaer, and J.M.H. Elmoghani, "Evaluation of energy consumption and network lifetime in rectangular ad-hoc wireless networks," Proc. 6th Int. Conference on Information Technology: New Generations (ITNG '09), pp.546–551, 2009.
- [4] L. Zhaohua and G. Mingjun, "Survey on network lifetime research for wireless sensor networks," Proc. 2nd IEEE Int. Conference on Broadband Network & Multimedia Technology (IC-BNMT '09), pp.899–902, 2009.
- [5] D. Ganesan, A. Cerpa, W. Ye, Y. Yu, J. Zhao, and D. Estrin, "Networking issues in wireless sensor networks," J. Parallel Distrib. Comput., vol.64, no.7, pp.799–814, 2004.

- [6] J.Y. Yu and P.H.J. Chong, "A survey of clustering schemes for mobile ad hoc networks," *IEEE Communication Survey and Tutorial*, vol.7, no.1, pp.32–48, 2005.
- [7] T. Nagata, H. Oguma, and K. Yamazaki, "A sensor networking middleware for clustering similar things," *Proc. Int. Workshop on Smart Object Systems in Conjunction with International Conference on Ubiquitous Computing*, 2005.
- [8] S. Priyankara, K. Kinoshita, H. Tode, and K. Murakami, "A clustering method for wireless sensor networks with heterogeneous node types," *IEICE Trans. Commun.*, vol.E94-B, no.8, pp.2254–2264, Aug. 2011.
- [9] S. Basagni, "Distributed clustering for ad hoc networks," *Proc. Int. Symposium on Parallel Architectures, Algorithms and Networks*, pp.310–315, 1999.
- [10] A.D. Amis, R. Prakash, T.H.P. Vuong, and D. Huynh, "Max-min D-cluster formation in wireless ad hoc networks," *Proc. IEEE INFOCOM 2000*, vol.1, pp.32–41, 2000.
- [11] T. Ohta, S. Inoue, Y. Kakuda, and K. Ishida, "An adaptive multihop clustering scheme for ad hoc networks with high mobility," *IEICE Trans. Fundamentals*, vol.E86-A, no.7, pp.1689–1697, July 2003.
- [12] P. Sasikumar and S. Khara, "k-means clustering in wireless sensor networks," *Proc. 4th International Conference on CICN 2012*, pp.140–144, 2012.
- [13] S. Thirumurugan and E.G.D.P. Raj, "Ex-PAC: An improved clustering technique for ad hoc network," *Proc. International Conference on RACSS 2012*, pp.196–200, 2012.
- [14] I. Tal and G.-M. Muntean, "User-oriented fuzzy logic-based clustering scheme for vehicular ad-hoc networks," *Proc. IEEE 77th VTC Spring*, pp.1–5, 2013.
- [15] G. Neglia and G. Reina, "Evaluating activator-inhibitor mechanisms for sensors coordination," *Proc. 2nd IEEE/ACM Int. Conference on Bio-Inspired Models of Network, Information and Computing Systems (IEEE/ACM BIONETICS 2007)*, pp.129–133, 2007.
- [16] K. Hyodo, N. Wakamiya, E. Nakaguchi, M. Murata, Y. Kubo, and K. Yanagihara, "Reaction-diffusion based autonomous control of wireless sensor networks," *Int. J. Sensor Networks*, vol.7, no.4, pp.189–198, 2010.
- [17] C. Takano and M. Aida, "Autonomous decentralized flow control mechanism based on diffusion phenomenon as guiding principle: Inspired from local-action theory," *J. IEICE*, vol.91, no.10, pp.875–880, Oct. 2008.
- [18] C. Takano, M. Aida, M. Murata, and M. Imase, "Proposal for autonomous decentralized structure formation based on local interaction and back-diffusion potential," *IEICE Trans. Commun.*, vol.E95-B, no.5, pp.1529–1538, May 2012.
- [19] K. Takagi, Y. Sakumoto, C. Takano, and M. Aida, "On convergence rate of autonomous decentralized structure formation technology for clustering in ad hoc networks," *Proc. IEEE ICDCS 2012 Workshops (ADSN 2012)*, 2012.
- [20] R. Hamamoto, C. Takano, K. Ishida, and M. Aida, "Power consumption characteristics by autonomous decentralized structure formation technology," *Proc. 9th Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT 2012)*, 2012.
- [21] R. Hamamoto, C. Takano, K. Ishida, and M. Aida, "Characteristics of autonomously configured structure formation based on power consumption and data transfer efficiency," *Proc. 6th International Workshop on Autonomous Self-Organizing Networks (ASON 2013)*, 2013.
- [22] S.R. Gandham, M. Dawanda, R. Prakash, and S. Venkatesan, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations," *Proc. IEEE GLOBECOM 2003*, vol.1, pp.377–381, 2003.
- [23] T. Ohta, M. Fujimoto, S. Inoue, and Y. Kakuda, "Hi-TORA: Hierarchical routing protocol in ad hoc networks," *Proc. 7th IEEE Int. Symposium on High Assurance Systems Engineering (HASE2002)*, 2002.
- [24] R. Hamamoto, C. Takano, K. Ishida, and M. Aida, "Guaranteeing asymptotic stability of clustering by autonomous decentralized structure formation," *Proc. 9th IEEE Int. Conf. on Autonomic and Trusted Computing (ATC 2012)*, 2012.
- [25] R. Hamamoto, C. Takano, K. Ishida, and M. Aida, "Guaranteeing asymptotic stability of clustering for MANET by autonomous decentralized structure formation mechanism based on local interaction," *Proc. IEEE ICDCS 2013 Workshops (ADSN 2013)*, 2013.
- [26] M.A. Dewar, V. Kadiramanathan, M. Opper, and G. Sanguinetti, "Parameter estimation and inference for stochastic reaction-diffusion systems: Application to morphogenesis in *D. melanogaster*," *BMC Systems Biology*, vol.4, issue 1, no.21, pp.1–9, 2010.
- [27] V.D. Park and M.S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," *Proc. IEEE INFOCOM 1997*, pp.1405–1413, 1997.
- [28] T. Ohta, K. Ishida, Y. Kakuda, S. Inoue, and K. Maeda, "Maintenance algorithm for hierarchical structure in large ad hoc networks," *Proc. 1st Int. Conference on Fundamentals of Electronics, Communications and Computer Sciences (ICFCS2002)*, 2002.
- [29] P. Nain, D. Towsley, B. Liu, and Z. Liu, "Properties of random direction models," *Proc. IEEE INFOCOM 2005*, pp.1897–1907, 2005.
- [30] D. Kominami, M. Sugano, M. Murata, and T. Hatauchi, "Robust and resilient data collection protocols for multihop wireless sensor networks," *IEICE Trans. Commun.*, vol.E95-B, no.9, pp.2740–2750, Sept. 2012.



**Ryo Hamamoto** received his B.E. and M.E. degrees in Computer Engineering from Hiroshima City University, Japan, in 2011 and 2014, respectively. His research interests are in wireless networks and distributed systems. He received the Information Network Research Award from IEICE in 2012. He is a student member of IEEE (U.S.A.).



**Chisa Takano** received her B.E. in telecommunication engineering from Osaka University, Japan, in 2000, and her Ph.D. in system design engineering from Tokyo Metropolitan University, Japan, in 2008. In 2000, she joined the Traffic Research Center, NTT Advanced Technology Corporation (NTTAT). Since April 2008, she has been an associate professor at the Graduate School of Information Sciences, Hiroshima City University. Her research interests are computer networks and distributed systems. She received the IEICE Young Researchers' Award in 2003, and the Information Network Research Award from the IEICE in 2004 and 2012. She is a member of IEEE (U.S.A.).

received the IEICE Young Researchers' Award in 2003, and the Information Network Research Award from the IEICE in 2004 and 2012. She is a member of IEEE (U.S.A.).



**Kenji Ishida** received his B.E., M.Sc., and Ph.D. from Hiroshima University, Japan, in 1984, 1986, and 1989, respectively. He was at Hiroshima Prefectural University from 1989 to 1997. From 1997 to 2003, he was an associate professor at Hiroshima City University. Since 2003, he has been a professor in the Department of Computer Engineering, Faculty of Information Sciences, Hiroshima City University. His research interests include distributed computing systems and design of control procedures

for computer networks. He is a member of IEEE (U.S.A.), ACM (U.S.A.), and IPSJ (Japan).



**Masaki Aida** received his B.S. and M.S. degrees in theoretical physics from St. Paul's University, Tokyo, Japan, in 1987 and 1989, respectively, and his Ph.D. in telecommunications engineering from the University of Tokyo, Japan, in 1999. In April 1989, he joined NTT Laboratories. From April 2005 to March 2007, he was an associate professor at the Faculty of System Design, Tokyo Metropolitan University. He has been a professor at the Graduate School of System Design, Tokyo Metropolitan University,

since April 2007. His current interests include traffic issues in computer communication networks. He is a member of IEEE and the Operations Research Society of Japan.