

# New Framework of Back Diffusion-based Autonomous Decentralized Control and Its Application to Clustering Scheme

Chisa Takano

Graduate School of Information Sciences,  
Hiroshima City University,  
Hiroshima-shi,  
Hiroshima 731-3194, Japan  
Email: takano@hiroshima-cu.ac.jp

Masaki Aida

Graduate School of System Design,  
Tokyo Metropolitan University,  
Hino-shi, Tokyo 191-0065, Japan  
Email: maida@sd.tmu.ac.jp

Masayuki Murata

and Makoto Imase  
Graduate School of Information Science  
and Technology, Osaka University,  
Suita-shi, Osaka 565-0871, Japan  
Email: {murata, imase}@ist.osaka-u.ac.jp

**Abstract**—We previously proposed a framework for autonomous decentralized control by introducing a partial differential equation based on local-action theory. Although each node in the network acts autonomously on the basis of only the local information directly available to it, our control proposal leads to optimal performance for the whole network. In addition, we have proposed flow control based on a diffusion equation to achieve autonomous load-balancing in the network and have confirmed that realizes congestion avoidance. This paper introduces a new autonomous decentralized control scheme that creates the spatial structures of finite size by using renormalization transformation and back diffusion drift. Moreover, we use the proposed control scheme to realize autonomous decentralized clustering in ad hoc networks.

## I. INTRODUCTION

Mobile ad hoc network (MANET) [1] has attracted a great deal as a likely network configuration technology with which to realize the long-desired ubiquitous networking. MANET has no communication infrastructure and is a self-configuring network of mobile systems connected by wireless links. It can, therefore, offer a flexible communication environment that is robust against disasters and changes in event sites. For ad hoc network control, many techniques have been introduced, especially with regard to route selection. The representative flat-type routing technologies in MANET are classified as either proactive schemes [3], [4] or reactive schemes [5], [6]. Both are effective if the MANET does not have many mobile systems. As the network size grows, however, these protocols have the problem that the size of the routing table stored at each individual node becomes large. As a result, the flooding overhead of route requests increases network load. Against this problem, a large number of studies have looked at clustering [7], [8], [9]. In clustering, mobile nodes are divided into different virtual groups (clusters) and routing is based on intra-cluster and inter-cluster calculations. Cluster-head candidate selection is one of the important problems in clustering schemes, and [10], [11] have proposed algorithms to select cluster-heads appropriately in accordance with network conditions.

The common basic framework demanded in those algorithms is adaptability to network environments, and it is hoped to have the following flexibility: when there is no cluster-head candidate in the neighborhood, the mobile node itself stands as a candidate and becomes a cluster head. On the other hand, when there are two or more cluster-head candidates, we need a highly effective and efficient metric that can realize optimal cluster-head selection. Examples of the metrics possible include node degree, cluster size, mobility speed, and battery energy [12]. In general, the cluster-head must have full information about the state of other nodes or of the overall network to optimize the cluster structure. Unfortunately, it is difficult to collect global information about the network given the network structure and the need to minimize the information exchange frequency over the MANET. Therefore, a MANET needs an autonomous decentralized clustering mechanism in which each node acts autonomously on the basis of just local information they are aware of, while the decisions made by each node optimize the state of the whole network.

In this paper, we introduce an autonomous decentralized clustering method that allows the nodes to act based only on the information they are aware of while realizing environmental adaptability and overall network optimization.

## II. AUTONOMOUS DECENTRALIZED CONTROL MECHANISM BASED ON LOCAL INTERACTION

This section describes the framework of the autonomous decentralized control mechanism based on local interaction [13]. When thinking about the interaction of two objects that are separated in space, there are two theories depending on the action method; “action at a distance (non-local interaction)” and “action through a medium (local interaction)”. “Non-local interaction” is the direct interaction of two objects that are not linked by any field or medium. For example, Newton’s law of universal gravitation is an example of “non-local interaction”; the force of gravity created between object A and object B,  $F$ , is directly proportional to mass  $m_1$  and  $m_2$  of the interacting objects, and inversely proportional to the square of the distance

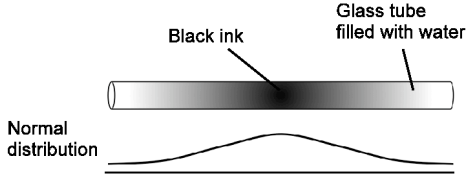


Fig. 1. Example of local interaction.

$|\mathbf{r}_1 - \mathbf{r}_2|$  between the objects as follows:

$$F(\mathbf{r}_1, \mathbf{r}_2) = G \frac{m_1 m_2}{|\mathbf{r}_1 - \mathbf{r}_2|^2},$$

where  $G$  is the gravitational constant. Function  $F(\mathbf{r}_1, \mathbf{r}_2)$  is decided by the distance, that is to say that it is necessary to know the value of the distance, in other word, the positions of objects A and B. From the viewpoint of network control, this is the same as the requirement established by centralized control that all global information about the network state must be collected.

In the case of “local interaction,” on the other hand, any variation in the physical value at a point of space is transmitted to the adjacent point via the medium of the gravitational field at a limited speed. A temporal variation in such a gravitational field can be described by using a partial differential equation. We use the following example to elucidate “local interaction.” : When we let a few drops of black ink fall into a glass tube filled with water, the ink density distribution follows a normal distribution and the ink spreads through the whole tube by diffusion (Fig. 1). In this process, the action within a minute region of water in the glass tube is very simple: the ink diffuses from the higher density side towards the lower density side. The rate of ink diffusion is proportional to the density gradient. Even though each segment acts autonomously and only local information is available, the ink density distribution throughout the glass tube exhibits orderly behavior.

Let the density function (density distribution) of “a certain quantity” at time  $t$  in space be  $p(x, t)$ . In the diffusion phenomenon, flow  $J(x, t)$  of “a certain quantity” in space is expressed as,

$$J(x, t) = -\kappa \frac{\partial p(x, t)}{\partial x}, \quad (1)$$

where  $\kappa$ , which is a positive constant, ( $> 0$ ). This equation means that “a certain quantity” flows at a rate which is proportional to the gradient of the density. When the total amount of “a certain quantity” in the system is unchanged, the continuous equation,

$$\frac{\partial p(x, t)}{\partial t} = -\frac{\partial J(x, t)}{\partial x} \quad (2)$$

holds.

We substitute Eq. (2) for Eq. (1) and get the temporal evolution equation of the density distribution,

$$\frac{\partial p(x, t)}{\partial t} = \kappa \frac{\partial^2 p(x, t)}{\partial x^2}, \quad (3)$$

where this is the ordinary diffusion equation. In this equation, when the density distribution is a delta function as the initial condition, the density becomes uniform over time. Since Eq. (3) is a linear differential, we can obtain general solutions by superposing the normal distribution under even more complex initial conditions. As seen in an example of the diffusion phenomenon, we can consider a system that is based on local interaction as the framework of the autonomous decentralized control from the viewpoint of engineering. In other words, while the sub-systems autonomously run using only local information, the state of the whole system is optimized indirectly through the corresponding temporal evolution equation. Variants of the autonomous decentralized control based on such local interaction can be summarized as follows:

- Think about the macro characteristic that the state of the whole system should have, and find the partial differential equation (corresponding to Eq. (3)) whose solution has such a characteristic.
- Think about local interaction (corresponding to Eq. (1)) that the partial differential equation describes, and design sub-systems so that they act according to the rule that is obtained from the local interaction.
- As a result, while sub-systems act only based on local information autonomously, they push the whole system into the preferable state as a solution of the partial differential equation.

In general, in autonomous decentralized systems whose sub-systems independently perform local interaction, the whole system might fall into system-deadlock if the sub-systems are not designed properly. The framework of autonomous decentralized control based on local interaction avoids this problem and gives us a hint as to how to properly design local actions of sub-systems so that the whole system is pushed into the desired state.

As an example of autonomous decentralized control based on local interaction, we previously proposed a diffusion-type flow control (DFC) mechanism [14], [15] that overcomes the difficulty of controlling high-speed networks. In this control mechanism, the state of the whole network is controlled indirectly through the autonomous action of each node; each node manages its local traffic flow on the basis of only the local information directly available to it, by using predetermined rules. By applying DFC, the distribution of the total number of packets in each node in the network becomes uniform over time, and it exhibits orderly behavior. This property is suitable for fast recovery from congestion. In Fig. 2, we show the temporal evolution of the number of packets stored in each node. The target flow is between node 1 and node 30, while the background traffic flows between node 15 and node 30. The target flow and the background flow start at simulation time  $t = 0$  s and  $t = 0.1$  s, respectively. The horizontal axes denote node ID (1–29) and the vertical axes denote the number of packets stored at the node. In the case with TCP without DFC, after the start of the background traffic flow (0.1 s), all the stored packets were at node 15, resulting in packet

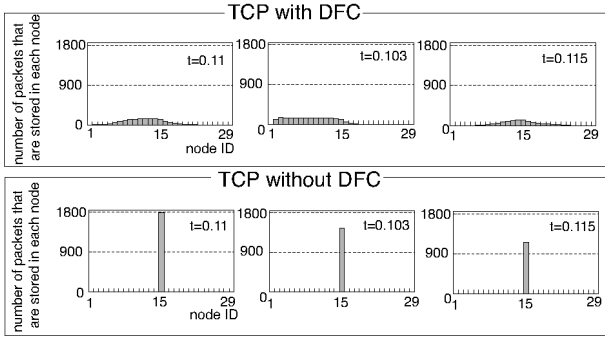


Fig. 2. Diffusion effect of DFC

loss. After that, the TCP window size was reduced and the number of stored packets decreased. In the case of TCP with DFC, the latter prevents the stored packets from building up at a particular node. Since packet loss was avoided, the TCP window size did not reduce and high network efficiency was achieved. Through the introduction of DFC, each node acts cooperatively to avoid packet loss even though the decision-making of each node is based only on local information.

### III. RENORMALIZATION TRANSFORMATION OF THE DIFFUSION PHENOMENON FOR AUTONOMOUS DECENTRALIZED CONTROL

So far, we have referred to the diffusion effect in describing autonomous decentralized control based on local interaction. We turn now to a structure of practical size that offers autonomous decentralized control for patterns other than the diffusion effect.

When we calculate the temporal evolution of density distribution  $p(x, t)$  which is the solution of Eq. (3), while dithering the  $x$  axis simultaneously within the ratio of  $\sqrt{2\kappa t}/\sigma$ , we get density distribution  $q(x, t)$  as follows:

$$q(x, t) := \frac{\sqrt{2\kappa t}}{\sigma} p\left(\frac{\sqrt{2\kappa t}}{\sigma}x, t\right), \quad (4)$$

where  $\sigma$  is a positive constant. Carrying out both the temporal evolution (diffusion) and the scale-variation (dithering) at the same time can be regarded as a kind of renormalization transformation of the density distribution  $p(x, t)$ . The density distribution  $q(x, t)$  for which we execute the renormalization transformation follows

$$\frac{\partial}{\partial t} q(x, t) = \frac{1}{2t} \left( \frac{\partial}{\partial x} x + \sigma^2 \frac{\partial^2}{\partial x^2} \right) q(x, t). \quad (5)$$

This differential equation satisfies the following equation:

$$\lim_{t \rightarrow \infty} q(x, t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}. \quad (6)$$

For the normal diffusion equation (3), the above equation means that the solution approaches a normal distribution asymptotically with time regardless of the initial condition. By use of this mechanism, we can realize a new autonomous decentralized control approach that operates within a limited structure. However, using Eq. (5) instead of expression Eq. (3)

raises two problems with regard to autonomous decentralized control.

- The existence of  $1/(2t)$  on the right hand side of Eq. (5) makes, over time, the change in the distribution small.
- It is necessary to set the coordinate systems in the network because the drift term for Eq. (5) directly depends on  $x$ .

We consider the latter problem in a following section.

Let us then discuss the transformation of the time-scale. The distribution  $q(x, t)$  in Eq. (5) should have a fast temporal evolution so that  $1/(2t)$  on the right hand side of Eq. (5) can be eliminated.

If a constant,  $c > 0$ , is used instead of  $1/(2t)$  in Eq. (5), we get:

$$\frac{\partial}{\partial t} q(x, t) = c \left( \frac{\partial}{\partial x} x + \sigma^2 \frac{\partial^2}{\partial x^2} \right) q(x, t), \quad (7)$$

where we construct the renormalization transformation

$$q(x, t) := \frac{\sqrt{2\kappa e^{2ct}}}{\sigma} p\left(\frac{\sqrt{2\kappa e^{2ct}}}{\sigma}x, e^{2ct}\right). \quad (8)$$

Here, for the reason that time merely passes fast in Eq. (8), the limiting distribution (6) is invariable under the redefinition of renormalization transformation.

It is necessary to specify the flow corresponding to  $J_q(x, t)$  of the continuous equation,

$$\frac{\partial}{\partial t} q(x, t) = -\frac{\partial}{\partial x} J_q(x, t),$$

if we are to associate Eq. (7) with autonomous decentralized control.  $J_q(x, t)$ , which satisfies this equation, is expressed as

$$J_q(x, t) = -cx q(x, t) - c\sigma^2 \frac{\partial}{\partial x} q(x, t), \quad (9)$$

where  $c$  denotes the rate of temporal evolution of the density distribution and  $\sigma^2$  denotes the variance of the normal distribution that is converged on. By the way, the temporal evolution equation Eq.(8) obtained by the renormalization transformation is an example of the Ornstein-Uhlenbeck process [16]. For the reason given above, if we make the local action of each node follow Eq. (9), the limiting distribution is expressed as Eq. (6), regardless of the configuration of the initial distribution.

The next section describes a method for designing autonomous decentralized control schemes that do not depend on the coordinate system.

### IV. LOCAL ACTION THEORY BASED ON BACK DIFFUSION-BASED POTENTIAL

It is necessary to decide the coordinate system in the network, because the drift term  $-cx q(x, t)$  of Eq. (9) depends on  $x$ . Here, we compare the flow of original drift term with the flow for the drift term based on the absolute value of coordinate  $x$ . Figure 3 shows that the direction and the size of the drift are invariable in terms of absolute coordinate values.

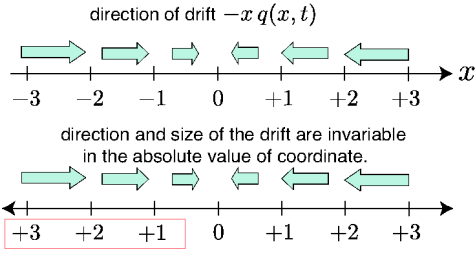


Fig. 3. Coordinate values and the direction of the drift

Therefore, we replace  $x$ , that appears to the drift term for Eq. (9), by some function  $f(x, t)$ , as follows:

$$J_q(x, t) = -c f(x, t) q(x, t) - c \sigma^2 \frac{\partial}{\partial x} q(x, t). \quad (10)$$

The temporal evolution of distribution  $q(x, t)$  that corresponds to this is given by

$$\frac{\partial}{\partial t} q(x, t) = c \left( \frac{\partial}{\partial x} f(x, t) + \sigma^2 \frac{\partial^2}{\partial x^2} \right) q(x, t). \quad (11)$$

The introduction of  $f(x, t)$  eliminates the need to set space coordinates in the network.

As a more intuitive explanation, we consider the following potential function  $\Phi(x, t)$  instead of function  $f(x, t)$ :

$$f(x, t) = -\frac{\partial \Phi(x, t)}{\partial x}. \quad (12)$$

Choosing  $\Phi(x, t)$  appropriately yields autonomous decentralized control that does not depend on the coordinate system.

In general, in order to decide beforehand the potential function  $\Phi(x, 0)$  that satisfies some global condition, we must know the macro-scale information related to the entire system, not the micro-scale information related to system elements. This renders problematic autonomous decentralized systems that operate only with local information. To resolve this conundrum, we consider how to decide the drift term from distribution  $q(x, t)$ . Because the potential function  $\Phi(x, t)$  should have the effect to maintain the distribution within the range of the limited space, in opposition to the effect of diffusion,  $\Phi(x, t)$  is decided as the equation,

$$\Phi(x, t + dt) = - \left( q(x, t) - \kappa' \frac{\partial^2 q(x, t)}{\partial x^2} dt \right), \quad (13)$$

where  $\kappa' > 0$ . This equation can be expressed as follows:

- We let the time progress of the diffusion phenomenon with the diffusion coefficient  $\kappa'$  be reversed (back diffusion).
- Next, we reverse the distribution (up and down), and regard the completed distribution as the potential.

The temporal evolution of the distribution becomes a flat state in the forward direction of time for general diffusion (Fig. 4). For back diffusion, on the other hand, the temporal evolution of the distribution goes against the direction of time and the configuration of the distribution becomes sharp with time. Inherent in back diffusion, the phenomenon of a decrease in entropy does not happen in the natural world, but it can be

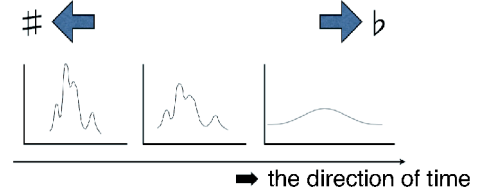


Fig. 4. back diffusion

realized easily in the framework of autonomy decentralized control.

Due to the effect of the drift term, including the potential, the peak of distribution  $q(x, t)$  is emphasized and the distribution shape is sharpened. The effect of the diffusion term, on the other hand, is to flatten the distribution. A structure of finite size can be formed by balancing one effect against the other.

## V. AUTONOMOUS DECENTRALIZED CONTROL CLUSTERING IN AD HOC NETWORKS

This paper applies the above autonomous decentralized control technologies to realize clustering (of finite size configuration) to ad hoc networks. To simplify the discussion, we adopt a one-dimensional network model in which the space coordinate  $x$  corresponds to node ID. Moreover, each node has a numerical value,  $q(x, t)$ , that indicates its fitness to be the cluster head.

We assume that the total number  $\int q(x, t) dx$  for numerical value  $q(x, t)$  of each node  $x$  in the network is constant. Each node interacts with the adjacent nodes and the numerical value is exchanged. As a result, distribution  $q(x, t)$  changes over time. Concrete interaction follows the local action rules (10) described in the previous section, and  $q(x, t)$  follows the temporal evolution Eq. (11). At this time, the peak in the shape of distribution  $q(x, t)$  corresponds to a cluster head, and the bottom corresponds to the boundary of the cluster.

We now explain the concrete process by which  $q(x, t)$  changes and how a cluster structure is generated.

The peak of distribution  $q(x, t)$  is emphasized and the cluster head candidate emerges, for the effect of the potential function  $\Phi(x, t)$  which is obtained by using the back-diffusion of the current distribution  $q(x, t)$  for the drift term in Eq. (11). On the other hand, the effect of the diffusion term of temporal evolution Eq. (11) softens the peak of the distribution  $q(x, t)$ . A structure appears by balancing the effect of the drift term against that of the diffusion term.

We explain the characteristics of the proposed method by Fig. 5. Let us consider the initial condition in which there are two or more cluster-head candidates on the left, and there is one small-scale cluster-head candidate in the right. In areas that are over-endowed with cluster-head candidates, the candidates are integrated autonomously to form one cluster. When there is no candidate in the surrounding oppositely, it becomes a cluster head even if it is not a powerful cluster head candidate. We can invent the structure that adjusts to local circumstances by the autonomous decentralized control.

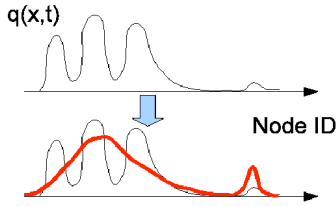


Fig. 5. Integrating and forming clusters according to local circumstances

The control information that each node sends is only  $q(x,t)$ . Each node sends the information to only adjacent nodes, and flooding other than adjacent nodes is unnecessary. Therefore, there are not a lot of amounts of the control information forwarded.

## VI. EVALUATION

### A. Evaluation of clustering method according to local circumstances

First, we evaluate whether to generate a cluster while dynamically changing each node role according to local circumstances. The network model used in the simulation is the one-dimensional model (the number of nodes is 200). The initial state of distribution  $q(x,t)$ , which represents cluster head fitness, is shown in the upper left of Fig. 6. The network model has the feature of torus type and both sides in one-dimensional model are connected (the right side of node 200 is node 0). The initial state exhibits three cluster head candidates on the left side, and one cluster head candidate on the right side. The horizontal axis denotes node ID and the vertical axis denotes the value of distribution  $q(x,t)$ .

If two cluster head candidates close to each other, they should be merged to form one cluster according to local circumstances and parameter  $\sigma^2$ . When there is no cluster-head candidate in a neighborhood, a cluster head candidate should become a cluster head even if it is not necessarily prime candidate. The results of evaluating the cluster head fitness distribution from this viewpoint are shown in the upper right of Fig. 6 and in the lower of this figure. Parameter  $\sigma^2$  is set to values of 0.1, 0.2 or 0.4 in the graphs, and  $c = 1$ ,  $\kappa' = 0.1$ . In this evaluation, the distribution is prone to form smooth clusters by setting the operation interval of the drift term more than the operation interval of the diffusion term. We set the drift interval  $\tau_{\text{drift}} = 10$  while the diffusion interval  $\tau_{\text{diff}}$  is 1. The plots show the results after sufficient time has past ( $t = 100$ ). We can see from the results that when  $\sigma^2$  is small (large), the cluster head candidates in the left side are not (are) integrated. Note that the cluster head on the right side is present in all cases.

### B. Evaluation of clustering method for the state without structure

In Fig. 7, similarly, we show the impact of the cluster structure in the condition where the initial fitness of each cluster head is given at random. Distribution  $q(x,t)$ , which denotes cluster head fitness, is set to a uniform random decimal number between  $[0, 1]$ , the upper left of Fig. 7. The evaluation

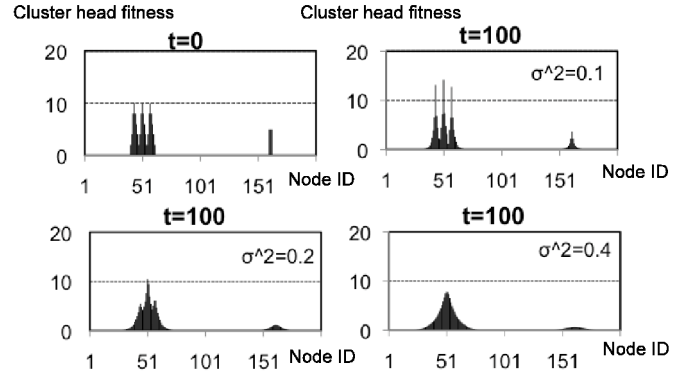


Fig. 6. Evaluation results of clustering method according to local circumstances

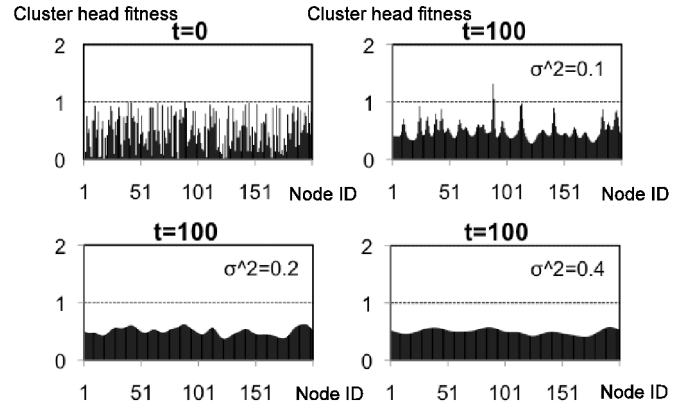


Fig. 7. Evaluation results of clustering method for the state without structure

results of the cluster head fitness distribution are shown in the upper right of and in the lower of Fig. 7. Parameter  $\sigma^2$  is set to values of 0.1, 0.2 or 0.4 in the graphs, and  $c = 1$ ,  $\kappa' = 0.1$ . The drift interval  $\tau_{\text{drift}}$  is 5 and the diffusion interval  $\tau_{\text{diff}}$  is 1. From these results, we can see that if parameter  $\sigma^2$  is small (large), many (few) small (large) clusters are created. This is reasonable because increasing parameter  $\sigma^2$  strengthens the diffusion effect which promotes cluster integration. The proposed method forms clusters autonomously where their number and size are determined by parameter  $\sigma^2$ .

### C. Evaluation of temporal evolution for the number of clusters

Next, we investigate the variation of the number of clusters for parameter  $\sigma^2$  that denotes strength of the diffusion effect. As for the network model, the initial fitness of each cluster is assumed to be at random as well as the previous evaluation. Parameter  $\sigma^2$  is set to values of 0.1, 0.2 or 0.4, and  $c = 1$ ,  $\kappa' = 0.1$  and the drift interval  $\tau = 5$ . For calculating the number of clusters  $N$  we add one to  $N$  when cluster head fitness  $q(x,t)$  of each node is larger than those of adjacent nodes (right and left nodes). We evaluate the number of clusters  $N$  ten times by changing the random seed. In Fig. 8, the horizontal axes denote the simulation time and the vertical axes denote the mean number of clusters  $N_{\text{mean}}$  for ten results. The three lines in this figure show the results for the cases where parameter  $\sigma^2 = 0.1, 0.2, \text{ and } 0.4$ . We can see in Fig. 8

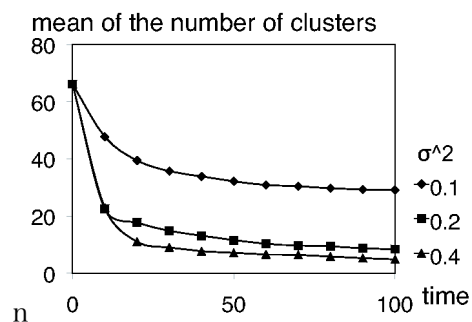


Fig. 8. Temporal variation of the mean number of clusters for  $\sigma^2$

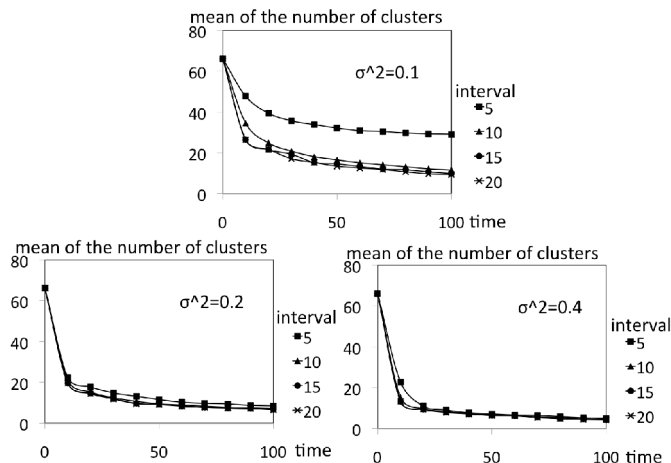


Fig. 9. Temporal variation of the mean number of clusters for  $\tau_{\text{drift}}$

that  $N_{\text{mean}}$  falls sharply at the start (from time  $t = 0$  to  $t = 10$ ) for every  $\sigma^2$ , and the decrease rate of  $N_{\text{mean}}$  becomes small with time. Finally, it settles down in an almost constant value and becomes stable. In the case of  $\sigma^2 = 0.4$ , larger clusters are created and the number of clusters decreases, because the larger  $\sigma^2$  is, the higher the diffusion effect is.

Next, we evaluate the mean number of clusters  $N_{\text{mean}}$  for the drift interval. The small drift interval (the high frequency of the drift operation) enlarges the effect of “the peak emphasis” and “the sharpness of the distribution.” Three graphs in fig.9 show the results when  $\sigma^2 = 0.1, 0.2$ , and  $0.4$ . The lines in this figure show the results for the cases where the drift interval  $\tau_{\text{drift}} = 5, 10, 15$ , and  $20$ .

In the case where  $\sigma^2 = 0.1$ ,  $N_{\text{mean}}$  is large when the drift interval is small ( $\tau_{\text{drift}} = 5$ ). When the drift interval is larger, on the other hand,  $N_{\text{mean}}$  is small and it doesn’t make much difference among three results ( $\tau_{\text{drift}} = 10, 15$ , and  $20$ ) in the lower of this figure. Then, the drift interval  $\tau_{\text{drift}}$  doesn’t influence much the mean number of clusters in the case where  $\sigma^2 = 0.2$  and  $0.4$ , and the values of  $N_{\text{mean}}$  are almost the same regardless of  $\tau_{\text{drift}}$ . To create a lot of clusters with the small cluster size, we have to set the parameter  $\sigma^2$  and  $\tau_{\text{drift}}$  to be small. On the other hand, larger clusters are created by using larger  $\sigma^2$ .

## VII. CONCLUSION

In this paper, we introduce an autonomous decentralized control approach that uses renormalization transformation and

back-diffusion potential; it forms the basis of our new clustering method for ad hoc networks. Though each node acts autonomously based only on local information, the decision-making of each node leads to meaningful cluster formation appropriate for overall network conditions. Evaluations clarified that the proposed method can realize a cluster structure autonomously in response to local circumstances in ad hoc networks. We can see that when there is no cluster-head candidate in the neighborhood, the mobile node itself stands as a candidate and becomes a cluster head, and on the other hand, when there are two or more cluster-head candidates, the node corresponding to metric is selected as cluster-head among them.

## ACKNOWLEDGMENT

This work was partly supported by Grant-in-Aid for Young Scientists (B) (21700086) from the Japan Society for the Promotion of Science, and Early-concept Grants for Exploratory Research on New-generation Network from NICT.

## REFERENCES

- [1] <http://www.ietf.org/html.charters/manet-charter.html>
- [2] E. Royer and C.-K. Toh, “A review of current routing protocols for ad hoc wireless networks,” *IEEE Personal Communications*, pp. 46-55, April 1999.
- [3] R. Ogier, F. Templin and M. Lewis, “Topology dissemination based on reverse-path forwarding (TBRPF),” Request for Comments 3684, February 2004.
- [4] T. Clausen and P. Jacquet, “Optimized link state routing protocol (OLSR),” Request for Comments 3626, October 2003.
- [5] C.E. Perkins, E.M. Royer and S. Das, “Ad hoc on-demand distance vector (AODV) routing,” Request for Comments 3561, July 2003.
- [6] D. Johnson, Y. Hu and D. Maltz, “The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4,” Request for Comments 4728, February 2007.
- [7] S. Basagni, “Distributed clustering for ad hoc networks,” *Proceeding of International Symposium on Parallel Architectures, Algorithms and Networks*, pp. 310-315, 1999.
- [8] A.D. Amis, R. Prakash, T.H.P. Vuong and D. Huynh, “Max-min D-cluster formation in wireless ad hoc networks,” *Proceeding IEEE INFOCOM 2000*.
- [9] T. Ohta, et al., “An adaptive multihop clustering scheme for ad hoc networks with high mobility,” *IEICE Transactions on Fundamentals* vol. E86-A no. 7, pp. 1689-1697, 2003.
- [10] N. Malpani, J.L. Welch and N. Vaidya, “Leader election algorithms for mobile ad hoc networks,” *Proceedings of the 4th international workshop on Discrete algorithms and methods for mobile computing and communications*, pp. 96-103, August 2000.
- [11] C. Johnen and L. Nguyen, “Self-stabilizing weight-based clustering algorithm for ad hoc sensor networks,” *2nd International Workshop on Algorithmic Aspects of Wireless Sensor Network AlgoSensors 2006*, Springer LNCS, vol. 4240, pp. 83-94, 2006.
- [12] 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.
- [13] M. Aida and C. Takano, “Principle of autonomous decentralized flow control and layered structure of network control with respect to time scales,” *Supplement of the ISADS 2003 Conference Fast Abstracts*, pp. 3-4, April 2003.
- [14] C. Takano and M. Aida, “Stability and adaptability of autonomous decentralized flow control in high-speed networks,” *IEICE Transactions on Communications*, vol. E86-B, no. 10, pp. 2882-2890, 2003.
- [15] C. Takano and M. Aida, “Diffusion-type autonomous decentralized flow control for multiple flows,” *IEICE Transactions on Communications*, vol. E90-B, no. 1, pp. 21-30, January 2007.
- [16] G.E. Uhlenbeck and L.S. Ornstein, “On the theory of Brownian Motion,” *Physical Review*, vol. 36, pp. 823-841, September 1930.