PAPER

Evolutionary Game Theoretic Approach to Self-Organized Data Aggregation in Delay Tolerant Networks*

K. Habibul KABIR[†a)], *Nonmember*, **Masahiro SASABE**[†b)], *and* **Tetsuya TAKINE**[†c)], *Members*

SUMMARY Custody transfer in delay tolerant networks (DTNs) provides reliable end-to-end data delivery by delegating the responsibility of data transfer among special nodes (custodians) in a hop-by-hop manner. However, storage congestion occurs when data increases and/or the network is partitioned into multiple sub-networks for a long time. The storage congestion can be alleviated by message ferries which move around the network and proactively collect data from the custodians. In such a scenario, data should be aggregated to some custodians so that message ferries can collect them effectively. In this paper, we propose a scheme to aggregate data into selected custodians, called aggregators, in a fully distributed and autonomous manner with the help of evolutionary game theoretic approach. Through theoretical analysis and several simulation experiments, taking account of the uncooperative behavior of nodes, we show that aggregators can be selected in a self-organized manner and the number of aggregators can be controlled to a desired value.

key words: *delay tolerant networks (DTNs), evolutionary game theory, custody transfer, self-organized, aggregators, message ferry.*

1. Introduction

With the development of networking technologies, many researchers and developers have tried to achieve data communications in challenged networks, called delay tolerant networks (DTNs) [2, 6], e.g., deep space, battle fields, disaster areas, underwater fields, etc. DTNs cause data communications with long delay, asymmetric data rates, and long queueing delay due to lack of continuous end-to-end connectivity. This class of challenged networks may not well match with the current end-to-end TCP/IP model.

In DTNs, a *store-carry-forward* [2] message delivery mechanism is used. A source node combines multiple data into a bundle and transmits it to the destination node in a hop-by-hop manner. However, instantaneous acknowledgment cannot be obtained due to lack of permanent end-to-end connectivity. *Custody transfer* [5] ensures reliable data transfer among nodes in DTNs. It offers that a bundle with custody must be perfectly delivered from a source to the corresponding destination by delegating the responsibility of reliable transfer with the bundle in a hop-by-hop manner. Intermediate nodes keeping bundles with custody are called *custodians*. Note here that to be a custodian, a node must reserve a sufficient amount of storage and energy to receive bundles with custody and hold them until successful delivery or the expiration of the bundle's delivery time. Custodians sometimes face storage congestion when they must refuse to receive a new bundle with custody due to lack of their storages or their sufficient energy to keep awake. An increase of bundles with custody and long-term network partitioning accelerate the storage congestion.

To solve the storage congestion problem, some special mobile nodes, called *message ferries* [29] can be introduced to proactively travel the network and gather bundles from custodians before the congestion occurs. If the network is divided into several isolated networks (clusters), message ferries move around the deployment area and deliver bundles among the clusters. However, if the requests from the storage congested nodes increase, sometimes it is hard for message ferries to visit all of them in a certain period of time.

Any custodian cannot predict how long it should keep bundles with custody. Note that each node in DTNs is basically powered by a battery and it has to be always awake when holding the bundles. Since each custodian also generates its own bundles with custody, it may be selfish and reject requests for custody transfer from other nodes to save its storage as well as its energy. This means that the custody transfer mechanism fails without taking the selfishness of custodians into account.

In summary, we face two challenges: a) It is very difficult for message ferries to communicate all storage-congested nodes in a given period of time and b) nodes are potentially selfish and are not willing to store others' bundles. To tackle these challenges, we propose a system that can a) gather all bundles in a partitioned network to some selected nodes in the network so that message ferries can collect them effectively and b) take the nodes' selfishness into account.

To accomplish such a system, evolutionary game theoretic approach becomes one of the most appropriate mechanisms, which originally explores the dynam-

Manuscript received July 1, 2009.
Manuscript revised November 4, 2009.
[†]The authors are with the Department of Information and Communications Technology, Graduate School of Engineering, Osaka University, Suita, Osaka, 565-0871, Japan.
*This paper is an extended version of the paper presented at IEEE AOC-2009, Greece, 2009.
a) E-mail: habib@post.comm.eng.osaka-u.ac.jp
b) E-mail: sasabe@comm.eng.osaka-u.ac.jp
c) E-mail: takine@comm.eng.osaka-u.ac.jp
DOI: 10.1587/transcom.E93.B.1

ics of a population of players under the influence of natural selection [19, 25]. In evolutionary game theory, we assume that fitness (payoff) of a species is determined by not only its own behavior (strategy), which is programmed by genes, but also the behavior of surrounding individuals: the more the fitness is acquired, the larger the population of the corresponding species is [20]. With the help of this scheme, we can finally select some special custodians referred to as *aggregators*, which are cooperative in nature and willingly hold bundles with custody of other nodes.

The rest of this paper is organized as follows. In Sect. 2 we review the related works. We described our proposed aggregators scheme and the selection procedures of aggregators in Sect. 3. Sections 4 and 5 give the analytical and simulation-based results based on evolutionary game theory, respectively. Finally we conclude in Sect. 6.

2. Related Works

Data aggregation has been studied in wireless sensor networks. In past few years, several researchers proposed LEACH (low energy adaptive clustering hierarchy) [8] and its extended versions [3, 27] for clustering-based data aggregation. In these schemes, sensor nodes play two kinds of roles to achieve data aggregation: Cluster head and regular node. Each node is initially a regular node. Then, it communicates with physically close nodes and elects a cluster head. The cluster head collects data from the regular nodes and forward it to a sink node through multi-hop communication among cluster heads. Here, the cluster head selection follows a stochastic algorithm taking account of nodes' energy consumption.

The aims of these schemes and the proposed scheme are almost the same but LEACH and its extended versions assume that all nodes cooperatively behave each other. On the other hand, in the proposed scheme, aggregator selection totally depends on the nodes' mutual interactions by taking account of selfishness of each node. Thus, the proposed scheme is also applicable to the cluster head selection in a more robust manner.

Inter-cluster communication is also required in DTNs. If the network is partitioned for a long time, the storage congestion frequently occurs in custodians. To alleviate the storage congestion, Zhao et al. proposed message ferry schemes which provide nodes with opportunities of communications among clusters [23, 28, 30]. There are two message ferry schemes [29]: Node-initiated message ferry scheme and ferry-initiated message ferry scheme. In the node-initiated message ferry scheme, nodes know the route of the message ferry in advance and move close to the ferry to transfer bundles on demand. On the other hand, in the ferry-initiated message ferry scheme, the message ferry takes proactive movement to meet the custodians. After receiving the service request from a custodian, the message ferry proceeds to the custodian and collects bundles. It can also supply energy to the custodian if required.

In our research, we use ferry-initiated message ferry scheme to collect bundles proactively from custodians. Sometimes it is difficult for message ferries to visit all custodians because of route limitations and traveling costs. In such a case, aggregating bundles to some selected nodes results in reducing the points where message ferries should visit.

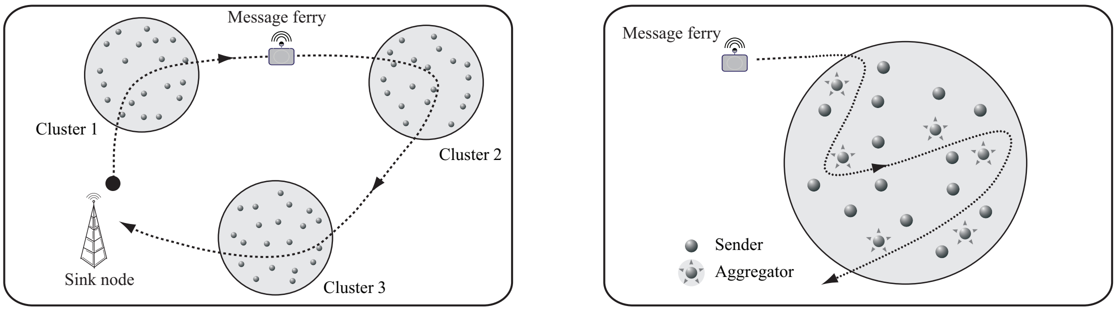
3. Proposed Scheme

3.1 Overview

In this paper, we aim to achieve a system that periodically collects information from multiple isolated networks, e.g., several sensing areas in sensor networks, many evacuation sites in disaster areas, etc. We can model these scenarios as follows. The system consists of one or more sink nodes and lots of clusters. Each node can directly communicate only with other nodes in the transmission range. To collect bundles from the clusters to the sink node, we apply the ferry-initiated message ferry scheme [29], where the message ferry departs from the sink node, visits each cluster to gather bundles, and then brings them back to the sink node as shown in Fig. 1(a). The duration of this cycle should be as short as possible so that the sink node can grasp the current conditions of all the clusters. When there are so many clusters and/or nodes, the duration tends to be longer. In that situation, we may divide clusters into several groups, based on their locations and the expected amount of generated bundles, and assign a single message ferry to each of those groups. Note that the scheme considered in this paper is applicable to such a case because each group of clusters behaves independently.

The duration of the cycle of the message ferry is mainly determined by two factors: The path length of the message ferry and the time for collecting bundles from the clusters and supplying energy to them. In our proposed system, the ferry path is calculated in a hierarchical manner: Inter-cluster path as in Fig. 1(a) and intra-cluster path as in Fig. 1(b). We assume that the length of the intra-cluster path is negligible compared to that of the inter-cluster path because the distance between nodes in an identical cluster is sufficiently shorter than that between clusters. The sink node can calculate the inter-cluster path in advance by obtaining the information on the physical locations of all clusters and solving traveling salesman problem (TSP) [18].

In a cluster, the path length of the message ferry is negligible but the time for collecting bundles from nodes and supplying energy to them linearly increases



(a) Message ferry visits each cluster and delivers collected bundles to the sink node. Sink node calculates an inter-cluster path for the message ferry.

(b) Message ferry visits a limited number of aggregators in a cluster. On arrival at a cluster, it calculates an intra-cluster path in an ad hoc manner.

Fig. 1 Proposed scenario.

with the number of nodes to be visited. To shorten this time, we propose a scheme to aggregate bundles in each cluster to some nodes referred to as *aggregators*. In each cluster, the aggregators are autonomously selected from nodes, called cluster members, by local interactions among them. Each non-aggregator (sender) sends its bundles to the aggregators so that the message ferry requires to visit only the aggregators as illustrated in Fig. 1(b).

In the above scenarios, we assume that each node is equipped with a long range radio and a short range radio. While the message ferry is approaching a cluster, it broadcasts its availability to all members of the cluster. Only aggregators with a specific amount of bundles are allowed to transmit service requests to the message ferry by their long range radio. These service request messages contain the information of each aggregator's location and the amount of bundles it wants to transfer. To guide the message ferry, aggregators occasionally transmit location update messages. On reception of each information, the message ferry calculates the intra-cluster path in an ad hoc manner. When the message ferry and one aggregator are close enough, the aggregator transfers bundles by its short range radio to the message ferry. At the same time, it obtains energy supply from the message ferry. Wireless energy transfer [10] will reduce the overhead and time for energy supply. Note that the range of long range radio transmission of each aggregator may not necessarily cover the whole deployment area due to power constraints. On the other hand, each sender sends its bundles to the aggregators within the transmission range by its short range radio.

At the initial stage, none of cluster members have any bundles, so they act as senders. While some cluster members generate their own initial bundles, they seek for aggregators within the transmission range. If no aggregator is available, the initial bundle's generators become aggregators. Under cluster members' mutual interactions, aggregators in the next round are selected with the help of evolutionary game theory. We describe the selection procedure of a limited number of aggregators in the next sub-section.

We can summarize the above scenario in each cluster as the repetition of the following three phases:

1. *Aggregator selecting phase* - Each node selects to be an aggregator or a sender based on local interactions with the neighboring nodes.
2. *Bundle aggregating phase* - When each sender generates its own bundles, it transmits them to one of the aggregators in the transmission range.
3. *Bundle collecting phase* - Each aggregator transmits its service request to the message ferry and sends all bundles to the ferry. The message ferry supplies energy to aggregators.

We define a *round* as the unit of this repetition as shown in Fig. 2. During each round, each node performs these three phases. We presume that all nodes synchronize each other and know the length of the round. The length of the round is pre-determined by the sink node which can also be updated through the communication between the ferry and nodes if needed.

This scenario not only shortens the duration of the round but also gives all nodes benefits in terms of long battery life. There are two ways to keep their batteries in high levels: 1) Obtaining the battery supply from the message ferry at the phase 3 of the round and 2) reducing the battery consumption by sleeping as long as possible in the round. The former (latter) case can be regarded as being an aggregator (a sender). Aggregators should be awake all the time in the round to receive bundles from senders as shown in Fig. 2. As a result, they consume much energy than senders but can also obtain the battery supply from the message ferry. On the other hand, senders cannot obtain the battery supply but can reduce the battery consumption by waking up only when it needs to generate and transmit its own bundle to the aggregators as shown in Fig. 2. We will give more detailed discussion about the battery life in Sect. 5.3.5.

Taking account of these characteristics, we expect that the system works well under the conditions: 1) There exist a small number of aggregators and many senders, and 2) the role of a node should change per round as shown in Fig. 2. These challenges can be divided into two problems: 1) How to select aggregators

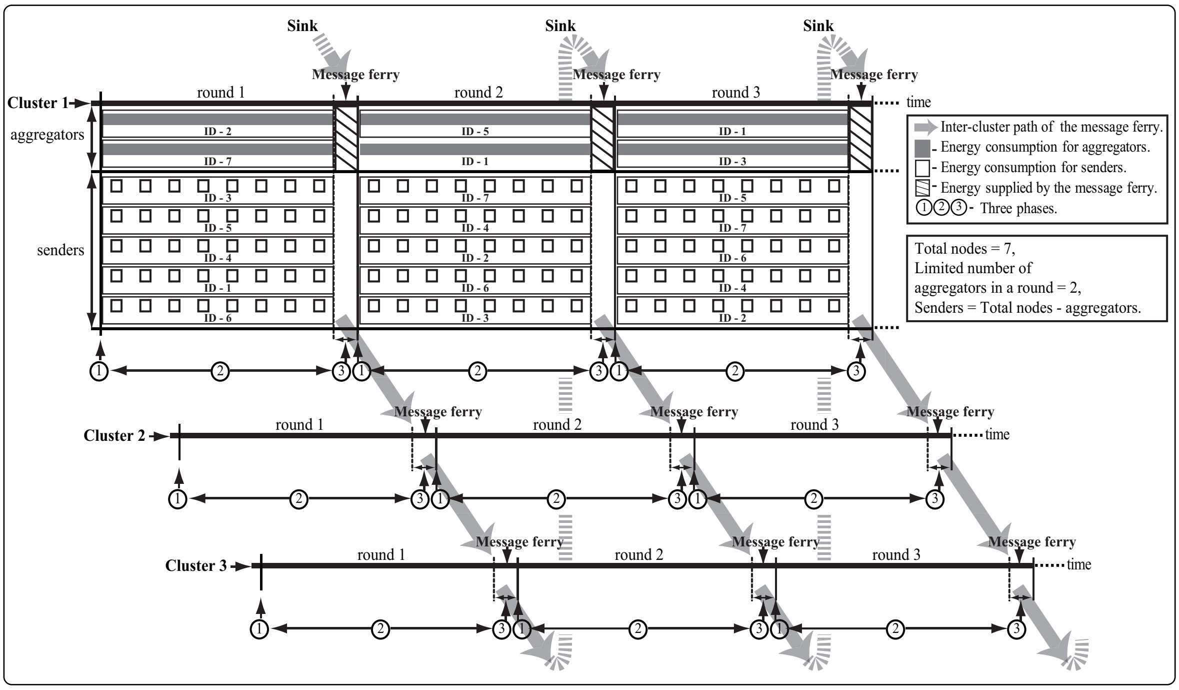


Fig. 2 Intra-cluster timing chart of cluster 1 with awaking period and role of each node, and inter-cluster timing chart for clusters 1, 2, and 3.

autonomously under situations where all nodes are potentially selfish and 2) how to control the number of aggregators. To cope with these problems, we apply evolutionary game theoretic approach.

3.2 Selection of the Aggregators

Since it is difficult to achieve a centralized control in DTNs due to lack of persistent connectivity among arbitrary nodes, the selection of aggregators should be realized in a decentralized way. We assume that each node communicates only with its neighbors and determines to be an aggregator or a sender based on its own benefit depending on the surrounding conditions.

We assume that each node loses energy proportional to the length of time it keeps awake. As mentioned above and illustrated in Fig. 2, aggregators are always awake during a round while senders only wake up when generating and transmitting their bundles. Let c and s denote the amount of energy consumption for aggregators and senders, respectively, per round. s increases with the rate of generating bundles but never exceeds c, i.e., $c > s > 0$. The energy supplied by the message ferry to each aggregator is represented by b. Intuitively, the larger b is, the more the aggregators increase. $b > c$ should also be satisfied to suppress the number of senders.

We first model the bargain among nodes as a game between two neighboring nodes in evolutionary game theory. There are two roles (strategies) for each node: Aggregator (aggregate) and sender (send). There are four possible combinations of the strategies of the two

Table 1 Payoff matrix.

node 1 \ node 2	send	aggregate
send	$-s, -s$	$-s, b-c$
aggregate	$b-c, -s$	$-c, -c$

Table 2 Abstracted payoff matrix.

node 1 \ node 2	send	aggregate
send	R, R	S, T
aggregate	T, S	P, P

nodes as in Table 1. The resulting payoffs for each combination can be modeled by taking the energy supply and energy consumption into account. If both nodes select to be aggregators, they lose the largest energy c without any energy supply from the message ferry, because they are not be able to collect a sufficient number of bundles to request the ferry to visit. An aggregator paired with a sender obtains the largest energy $b - c$; it loses c but obtains b from the message ferry, while the corresponding sender loses the smallest energy s. When both nodes select to be senders, they consume s.

We can abstract Table 1 into Table 2, where $T > S = R > P$. Every node not only has a temptation to be an aggregator ($T > R$) but also a fear to be an aggregator ($S > P$). The larger b is, the more the temptation is. This indicates that the sink node can control the number of aggregators (senders) by changing b. We show the detail in Sect. 4. The condition $T > R$ and $S > P$ also has another significant characteristic; taking a strategy different from the opponent is better than taking the same strategy as the opponent. As a result, both aggregating and sending strategies stably coexist [13]. Thus, with the help of the payoff-matrix and evolutionary game theory, when each node undertakes suitable strategies to optimize its own payoff, then the system converges to a fully stable situation where both senders and aggregators stably coexist.

In the next sections, we clarify the relationship between the parameters in Table 1 and the number of aggregators using evolutionary game theory.

4. Analytical Results

In this section, we discuss the relationship between the ratio of aggregators and the parameters of the payoff matrix through replicator equation on graphs in evolutionary game theory [14, 17]. The basic concept of replicator dynamics is that the growth rate of nodes taking a specific strategy is proportional to the payoff acquired by the strategy. Thus the strategy that yields more payoff than the average payoff of the whole system increases. Replicator dynamics on graphs additionally takes account of the effect of the topological structure of the network which is suitable for our system. We give the details of evolutionary game theory and the replicator equation on graphs in Appendix A.

4.1 Replicator Equation on Graphs

We first consider the replicator equation on graphs [14]. Let x denote the ratio of the number of aggregators to the total number of cluster members. Note that $1-x$ represents the ratio of the number of senders. The expected payoff (fitness) f_1 and f_2 of aggregators and senders are given by

$$f_1 = (1-x)(b-c) - cx, \quad f_2 = -s, \tag{1}$$

respectively.

Let k denote the number of neighbors of each node, called degree [15]. Although we will present the analysis based on the k-regular graphs in [14], the result is also applicable to non-regular graphs, e.g., unit disk graph, random networks, scale free networks, etc [14, 15]. In such a case, k represents the average degree. The modified payoff matrix for evolutionary game theory on graphs is defined as the sum of the original payoff matrix and a modifier matrix [14]. Table 3 shows the modifier matrix, where m describes the local competition between the strategies [14]. The gain of one strategy is the loss of another and local competition between the same strategies results in zero. It follows from Eq. (A·5) that m becomes

$$m = \frac{3b - (k+6)(c-s)}{(k+3)(k-2)}, \quad k > 2. \tag{2}$$

The expected payoff for the local competition g_1 and g_2 of aggregators and senders are obtained to be

$$g_1 = (1-x)m, \qquad g_2 = -xm, \tag{3}$$

Table 3 Modifier matrix.

node 2 / node 1	send	aggregate
send	$0, 0$	$-m, m$
aggregate	$m, -m$	$0, 0$

respectively, where m is given by Eq. (2). The average payoff ϕ of two strategies is then given by

$$\begin{aligned}\phi &= x(f_1+g_1) + (1-x)(f_2+g_2) \\ &= (1-x)(bx-s) - cx.\end{aligned} \tag{4}$$

From Eqs. (1), (3), and (4), we obtain the replicator equation on graphs [14] for $k > 2$ to be

$$\begin{aligned}\dot{x} &= x(f_1+g_1-\phi) \\ &= x(1-x)\left[\frac{b(k^2+k-3)-(c-s)(k^2+2k)}{(k+3)(k-2)} - bx\right].\end{aligned} \tag{5}$$

Substituting $\dot{x} = 0$ yields three equilibria: $x^* = 0, 1$, and

$$x^* = \frac{b(k^2+k-3) - (c-s)(k^2+2k)}{b(k+3)(k-2)}, \quad k > 2. \tag{6}$$

Note that the equilibrium in Eq. (6) is feasible if $0 < x^* < 1$, i.e.,

$$\frac{k^2+2k}{k^2+k-3} < \frac{b}{c-s} < \frac{k^2+2k}{3}, \tag{7}$$

holds. As mentioned above, $c-s > 0$. We also have for all $k > 2$, $0 < (k^2+2k)/(k^2+k-3) < (k^2+2k)/3$. As a result, for any c, s, and k, there exists $b > 0$ which satisfies Eq. (7). Thus the equilibrium in Eq. (6) is controllable. Further, x^* in Eq. (6) is stable because $\dot{x} > 0$ if $0 < x < x^*$, and otherwise, $\dot{x} < 0$. In the next sub-section, we investigate the effects of system parameters on the controllable equilibrium x^*.

4.2 Numerical Results

We have four independent variables, b, c, s and k, which affect x. For simplicity, $c-s$ is assumed to be one. Note that this simplification does not lose generality. Note here that the (average) degree k is a pre-determined parameter representing the density of nodes in the system under consideration. As a result, the ratio of aggregators can be controlled only by b according to Eq. (6). The expected number of aggregators can be obtained by the product of x and the number of cluster members.

Figure 3 illustrates the range of b with the supremum and infimum that satisfy Eq. (7), as a function of k. We observe that the valid range of b widens with k, while the infimum is almost constant. Figure 4 shows the controllable equilibrium x^* as a function of k. As shown in Eq. (6), x^* can take any value between 0 and 1 in both cases, depending on b and k. From those figures, we observe that for each b, x^* does not change when k becomes large, because the modifier m converges to zero with an increase of k, as shown in Fig. 5. We also observe that for a fixed k, the small b leads

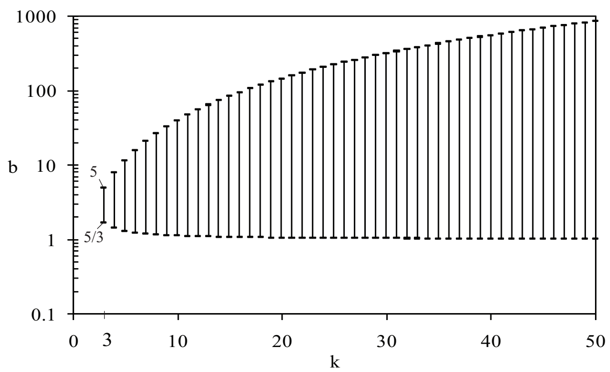


Fig. 3 The supremum and infimum of b ($c - s = 1$).

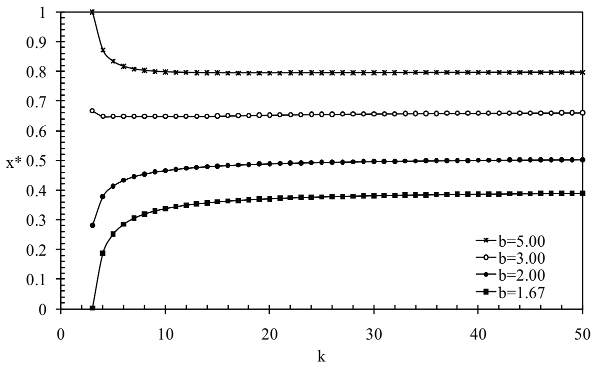


Fig. 4 The controllable equilibrium x^* ($c - s = 1$).

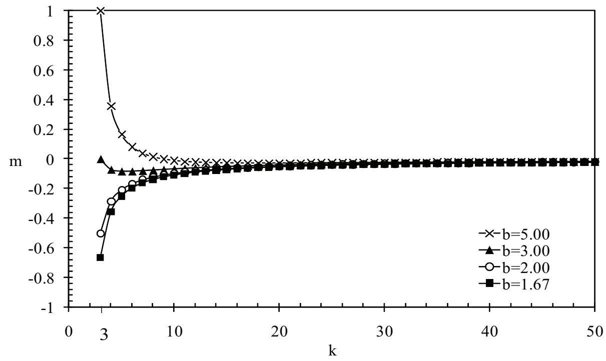


Fig. 5 The modifier m ($c - s = 1$).

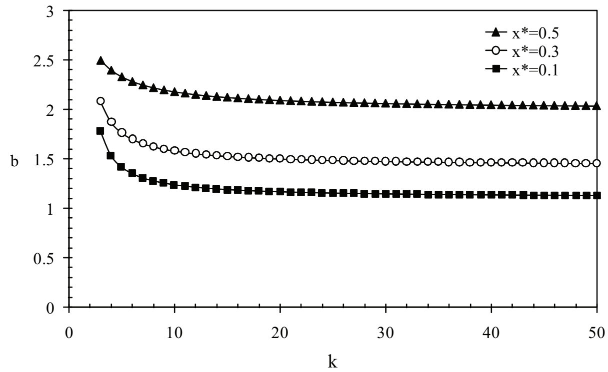


Fig. 6 Appropriate b to achieve x^*.

to the small x^*, which can be shown analytically with Eq. (6). When k is less than 20, the controllable equilibrium x^* shows different characteristics, depending on b. Roughly speaking, if the modifier m is negative (i.e., $b < 3$), x^* is a non-decreasing function of k, and otherwise, x^* is a non-increasing function of k.

Finally, for a given k, Fig. 6 shows appropriate values of b to achieve a specific value of x, where x^* is set to be 0.1, 0.3, and 0.5. We first find that b can be less than 3.00. Note that this value of b is valid under the assumption of $c - s = 1$. We do not need much larger b to achieve our objective that is limiting the ratio of aggregators. Furthermore, if k is larger than 20, b converges to a value, depending on the target level of x^*.

5. Simulation Experiments

Replicator dynamics is a powerful mathematical tool to predict the macro-level system behavior and it clarifies the effect of parameters on it. However, we can gain little insight into the micro-level system behavior such as the influence of irregularity of the topology on the system behavior, the geographical distribution of strategies, transient phenomena (including the convergence time to the equilibrium), and so on. Therefore we conduct simulation experiments based on agent-based dynamics, which is a complementary method to understand the micro-level system behavior in the evolutionary game theory. It models such a phenomenon that a superior strategy spreads over the network in a hop-by-hop manner, where local interactions among neighboring nodes are defined explicitly. In what follows, we first describe agent-based dynamics of our system and then show the results of simulation experiments.

5.1 Agent-based Dynamics

In agent-based dynamics, each agent (i.e., node) interacts only with physically-closed nodes, called neighbors, rather than all other agents in replicator dynamics. In DTNs, nodes within the transmission range of a node can be regarded as neighbors of the node. Each node decides its behavior (a strategy) in the next round based on the information obtained in the preceding round. Agent-based dynamics reveals how the strategies, which are determined from local interactions, affect the performance of the whole system.

In every round, each node determines its strategy by comparing its own payoff with that of a randomly chosen neighboring node at the preceding round. Note that there is no assumption on the initial distribution of strategies. As we will see later, the initial strategy distribution almost does not have any influences on the system performance, except that it slightly affects the convergence time to the expected equilibrium of x discussed in Sect. 5.3. The strategy update of node u is conducted in the follow probabilistic manner, called betters-possess-chance [7, 26]. At the beginning of each round, node u randomly chooses one of neighboring nodes, say, node v. If the average payoff Q_v of node v is greater than the average payoff Q_u of node u, node u then imitates the strategy of node v with probability $H(u, v)$.

$$H(u, v) = \frac{Q_v - Q_u}{T - P}, \tag{8}$$

where, $T - P$ $(= b)$ represents the maximum payoff difference. Otherwise, node u does not change its strategy. Thus, the more a strategy acquires the payoff, the more it spreads over the network through the imitation process in a hop-by-hop manner.

5.2 Simulation Model

Simulation experiments were conducted with NetLogo [12], a multi-agent programmable modeling simulator. Although we assume that the system consists of multiple clusters, we focus on the intra-cluster behavior, and inter-cluster behavior remains as a future work. For simplicity, we assume that the duration of a round is fixed and each node periodically generates a fixed number of bundles per round. Therefore c and s are constant and let $c - s = 1$ as in Sect. 4.2. In the following figures, the average of 100 independent simulation experiments are plotted.

5.3 Simulation Results

We first confirm the range of the number N of cluster members to which the prediction through replicator dynamics is applicable. After that, we discuss system characteristics in detail: The transient behavior, the role transitions of nodes, the effect of topological structures, and the battery life of nodes.

5.3.1 System size valid for replicator dynamics

Figure 7 compares the analytical results of replicator dynamics on graphs with the simulation results of agent-based dynamics, where graphs are regular and b is set to be 1.67. When both the number N of nodes and the degree k are large enough, agent-based dynamics attains the same equilibrium as predicted by replicator dynamics on graphs, because replicator dynamics on graphs assumes that $N = \infty$ and k is sufficiently large. When k is small, however, we observe a slight difference even for a large N. For example, when $N = 100$, the equilibrium in the agent-based dynamics is at most 0.079 greater than that in the replicator dynamics. Contrarily, the results of agent-based dynamics for $N = 10$ totally differ from those of replicator dynamics. Thus the number N of cluster members is essential in applying replicator dynamics to predicting the ratio of aggregators in equilibrium. In what follows, N is set to be 100.

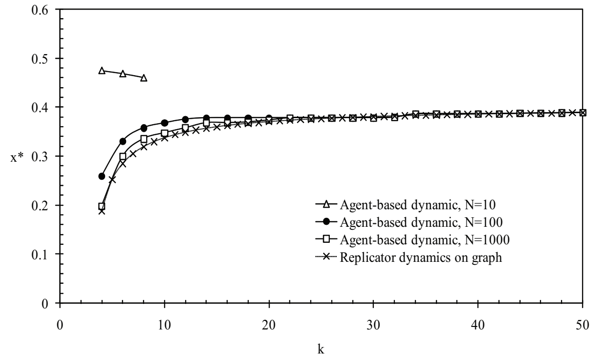


Fig. 7 Equilibrium x^* in k-regular graphs ($b = 1.67$).

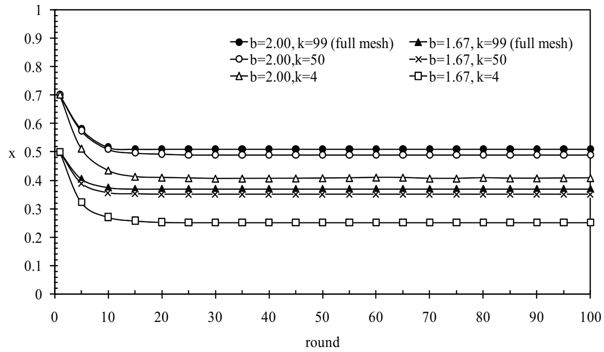


Fig. 8 Transient behavior of ratio x in k-regular graphs ($N = 100$).

5.3.2 Transient behavior

Figure 8 shows how the ratio x of aggregators converges to the equilibrium, where graphs are regular. We observe that x converges after 20 rounds for all cases. This quick convergence property is suitable for achieving a stable system. The resulting equilibrium is not greater than the predicted x^* in general and it coincides with x^* in the full mesh case, as shown in Fig. 7.

Next, we investigate the influence of the initial strategy distribution on the convergence property. Recall that the predicted equilibrium x^* by the replicator dynamics is almost globally stable, i.e., if the initial value of x is in $(0, 1)$, the replicator equation in (5) converges to the equilibrium x^* in Eq. (6). Therefore we expect that the agent-based dynamics inherits the stable convergence property. Although the convergence time depends on the initial value of x, we found that x converges to the same equilibrium at most 20 rounds.

5.3.3 Role transition of nodes

We showed that the ratio of aggregators quickly converges to the equilibrium. The role of each node, however, is not fixed but it alternates dynamically over rounds, because each node selects its own strategy in a probabilistic manner. Figure 9 illustrates the probability of being an aggregator of node i, p_i

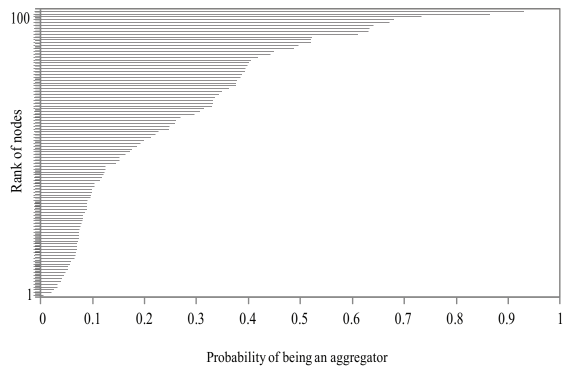


Fig. 9 Probability of each node being an aggregator in a k-regular graph ($k = 4$, $b = 1.67$).

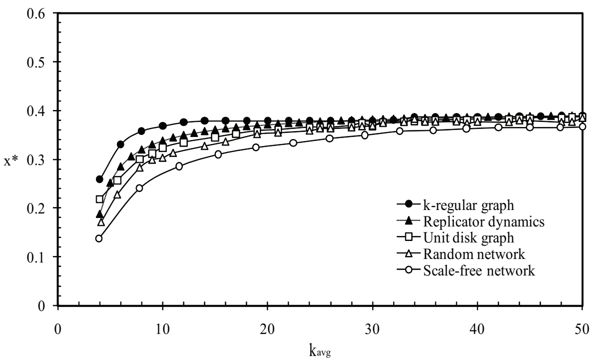


Fig. 10 Influence of network topology on the equilibrium x^* ($b = 1.67$).

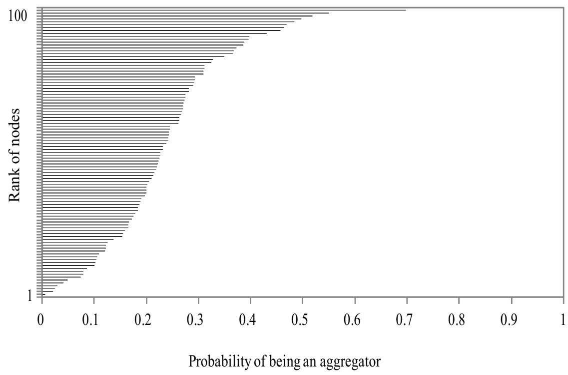


Fig. 11 Probability of each node being an aggregator in a unit disk graph ($k_{\rm avg} = 3.96$, $b = 1.67$).

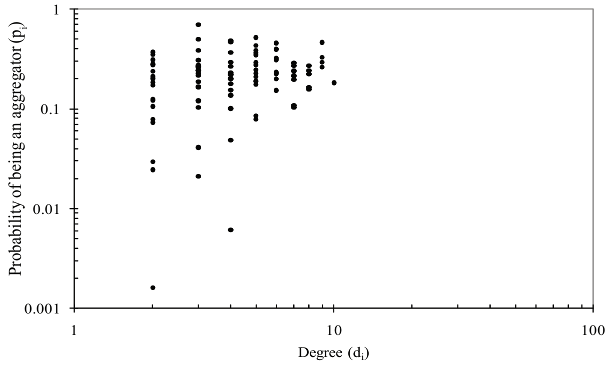


Fig. 12 Degree vs. probability of each node being an aggregator in a unit disk graph ($k_{\rm avg} = 3.96$, $b = 1.67$) (log scale).

($i = 1, 2, \ldots, 100$), in 3,000 rounds, where nodes are sorted in ascending order of p_i. Note that the average $\overline{p}$ of p_i is equal to 0.2508 and the standard deviation of that is equal to 0.2000. The role transition contributes to load balancing and robustness against node failures.

5.3.4 Effect of topological structures

So far we have shown the simulation results with k-regular graphs. We now consider unit disk graphs as more realistic networks. The unit disk graphs are generated by randomly located nodes in 2-dimentional space where two nodes are adjacent if the transmission ranges of the nodes mutually cover each other. We additionally produce two famous network topologies: Scale-free networks and random networks with Barabasi-Albert (BA) model [1] and Erdos-Renyi (ER) model [4], respectively. Note that we can control the average degree $k_{\rm avg}$ by adjusting parameters in those models adequately. With those network models, we discuss the influence of topological structures on the system performance.

Figure 10 shows the equilibrium x^* as a function of average degree $k_{\rm avg}$ in networks with different topological structures, where $N = 100$ and $b = 1.67$. The variance in the degree of nodes for k-regular, unit disk graph, random networks, and scale-free networks are 4.00, 4.73, 5.62 and 9.02, respectively. We observe that the large variance in the degree of nodes leads to the small ratio of aggregators x^* in equilibrium.

To investigate this phenomenon more closely, we observe two figures. Figure 11 shows p_i over 3,000 rounds in a unit disk graph, where nodes are sorted in ascending order of p_i. $\overline{p}$ is equal to 0.2112 and the standard deviation of p_i is equal to 0.1247. Compared with Fig. 9 in a k-regular graph, we observe that $\overline{p}$ becomes small in the unit disk graph. Figure 12 is a scatter graph showing the degree d_i and p_i of node i ($i = 1, 2, \ldots, 100$) in a unit disk graph. We observe that the positive correlation between those two quantities; nodes with high degrees are likely to have large probabilities. In fact, the overall average probability $\overline{p}_W$ of being an aggregator weighted by node degree is equal to 0.2933, where

$$\overline{p}_W = \frac{\sum_{i=1}^{N} d_i p_i}{N k_{\rm avg}},$$

which is greater than the un-weighted average probability $\overline{p}$. Thus we conclude that nodes with large degrees have a stronger impact in playing games than those with small degrees.

Finally, we investigate the speed of the convergence to the equilibrium. Figure 13 shows the transient behavior of the ratio x of aggregators in k-regular graph, unit disk graph, random, and scale-free network with

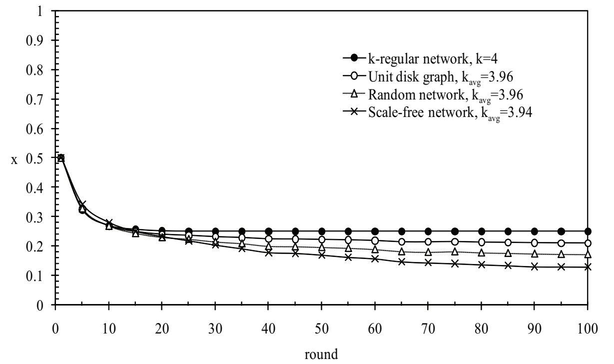


Fig. 13 Transient behavior of ratio x in different topological structures ($b = 1.67$).

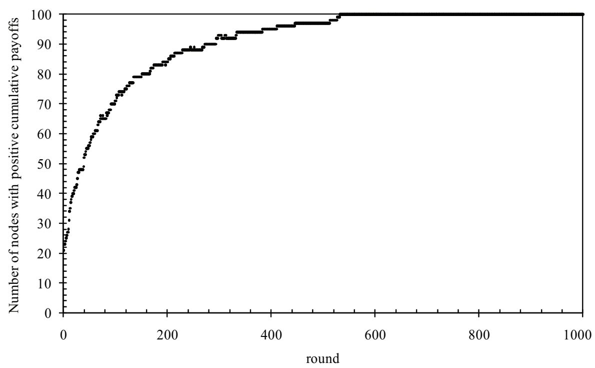


Fig. 14 Transition of the number of nodes with positive cumulative payoffs, (A unit disk graph, area size is 1x1 [km^2], transmission range of each node is 100 [m], $N = 100$, $k_{\text{avg}} = 3.96$, $b = 1.67$, $c = 1.1$, $s = 0.1$, $x = 0.218$).

$N = 100$ and $b = 1.67$, where the average degree k_{avg} is set to be almost the same. We observe that it takes a longer time for networks with high degree variation, compared with networks with low degree variation, yet 100 rounds is enough to converge to the equilibrium in all cases. In summary, the proposed scheme works well in those kinds of non-regular networks.

5.3.5 Battery life

We expect that all nodes can survive forever under appropriate values of the parameters b, c, and s. As mentioned in Sect. 3.2, $b > c > s > 0$ should be satisfied and we suppose that $c - s = 1$ without loss of generality. In steady state, the expectation of payoffs acquired at each node becomes $\mathrm{E}[p] = (b - c)x^* - s(1 - x^*)$. If $\mathrm{E}[p]$ is positive, the system could survive without loss of any node. x^* can be numerically obtained by setting b under $c - s = 1$. After obtaining b and x^*, we can determine the valid combinations of c and s which satisfy $\mathrm{E}[p] > 0$. In actual situations, the desirable c and s can be obtained by controlling the awaking time of aggregators and the generation rate of bundles of senders, respectively.

Figure 14 shows the transition of the number of nodes with positive cumulative payoff. Note that every node initially has no payoff. We observe that every node acquires positive cumulative payoffs after $535th$ round. This indicates that every node can work forever using the proposed scheme if it has a sufficient amount of initial battery which depends on the parameter settings.

6. Conclusion

This paper considered data aggregation for message ferries in delay tolerant networks. Contrary to existing works, we assumed that nodes were selfish and non-cooperative in nature. Applying evolutionary game theory, we proposed the self-organized data aggregation scheme in such an environment. In this scheme, the selection of aggregators is conducted through decentralized processes with the help of strategic decisions of evolutionary game theory. The proposed scheme was evaluated through replicator dynamics and agent-based dynamics, and we showed the excellent performance of the proposed scheme. In particular, we can control the numbers of the aggregators by setting parameters adequately.

Note that the controllable and stable equilibrium of the ratio of aggregators follows from the fact that a strategy different from the opponent yields a larger payoff (i.e., $T > R$ and $S > P$ in Table 2). Therefore our proposed scheme also works well under such a situation that senders need to retransmit bundles when their transmissions fail and therefore they should awake until their successful transmissions. In this case, the payoff matrix should be changed accordingly, yet the overall characteristics of the proposed scheme remain the same.

Acknowledgment

This research was supported in part by "Global COE (Centers of Excellence) Program" of the Ministry of Education, Culture, Sports, Science and Technology, Japan and Grant-in-Aid for Young Scientists (B) 20700061.

References

[1] A. Barabasi and R. Albert, "Emergence of scaling in random networks," Science, vol. 286, no. 5439, pp. 509–512, 1999.
[2] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall and H. Weiss, "Delay tolerant network architecture," work in progress as an IETF RFC 4838 Draft. http://www.ietf.org/rfc/rfc4838.txt
[3] H. Chen and S. Megerian, "Cluster sizing and head selection for efficient data aggregation and routing in sensor networks," Proc. IEEE WCNC, pp. 2318–2323, 2006.
[4] P. Erdos and A. Renyi, "On the evolution of random graphs," Publications of the Mathematical Institute of the Hungarian Academy of Sciences, vol. 5, pp. 17-61, 1960.
[5] K. Fall and W. Hong, "Custody transfer for reliable delivery in delay tolerant networks," Tech. Rep. IRB-TR-03-030,

Intel Research Berkeley, 2003.
[6] K. Fall, "A delay-tolerant network architecture for challenged internets," Proc. ACM SIGCOMM, pp. 27–34, 2003.
[7] C. Hauert and M. Doebeli, "Spatial structure often inhibits the evolution of cooperation in the snowdrift game," Nature, vol. 428, no. 6983, pp.643–646, 2004.
[8] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," Proc. Hawaii International Conference on System Sciences, pp. 3005-3014, 2000.
[9] J. Hofbauer and K. Sigmund, Evolutionary games and population dynamics, Cambridge University Press, 1998.
[10] A. Kurs, A. Karalis, R. Moffatt, J. Joannopoulos, P. Fisher, and M. Soljacic, "Wireless power transfer via strongly coupled magnetic resonances," Science, vol. 317, no. 5834, pp. 83–86, 2007.
[11] E. Lieberman, C. Hauert, and M. Nowak, "Evolutionary dynamics on graphs," Nature, vol. 433, no. 7023, pp. 312–316, 2005.
[12] NetLogo, version 4.0.4. http://ccl.northwestern.edu/netlogo
[13] M. A. Nowak, Evolutionary dynamics: exploring the equations of life, Harvard University Press, 2006.
[14] H. Ohtsuki and M. A. Nowak, "The replicator equation on graphs," Journal of Theoretical Biology, vol. 243, no. 1, pp. 86–97, 2006.
[15] H. Ohtsuki, C. Hauert, E. Lieberman, and M. Nowak, "A simple rule for the evolution of cooperation on graphs," Nature, vol. 441, no. 7092, pp. 502–505, 2006.
[16] H. Ohtsuki, J. Pacheco, and M. A. Nowak, "Evolutionary graph theory: Breaking the symmetry between interaction and replacement," Journal of Theoretical Biology, vol. 246, no. 4, pp. 681–694, 2007.
[17] H. Ohtsuki and M. A. Nowak, "Evolutionary stability on graphs," Journal of Theoretical Biology, vol. 251, no. 4, pp. 698–707, 2008.
[18] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, Numerical recipes 3rd edition: The art of scientific computing, Cambridge University Press, NY, USA, 2007.
[19] L. Samuelson, Evolutionary games and equilibrium selection, MIT Press, 1997.
[20] K. Sigmund and M. A. Nowak, "Evolutionary game theory," Current Biology, vol. 9, no. 14, pp. 503–505, 1999.
[21] J. Smith, Evolution and the theory of games, Cambridge University Press, 1982.
[22] G. Szabó and G. Fáth, "Evolutionary games on graphs," Physics Reports, vol. 446, pp. 97–216, 2007.
[23] M. M. B. Tariq, M. Ammar, and E. Zegura, "Message ferry route design for sparse ad hoc networks with mobile nodes," Proc. ACM MobiHoc, pp. 37–48, 2006.
[24] P. Taylor and L. Jonker, "Evolutionary stable strategies and game dynamics," Mathematical Biosciences, vol. 40, pp. 145–156, 1978.
[25] J. Weibull, Evolutionary game theory, MIT Press, 1995.
[26] Z. Wu, X. Xu, and Y. Wang, "Does the scale-free topology favor the emergence of cooperation?," 2005. http://arxiv.org/abs/physics/0508220
[27] H. Yang and B. Sikdar, "Optimal cluster head selection in the leach architecture," Proc. IEEE International Performance, Computing, and Communications Conference, (IPCCC 2007), pp. 93–100, 2007.
[28] W. Zhao and M. Ammar, "Message ferrying: proactive routing in highly-partitioned wireless ad hoc networks," Proc. IEEE Future Trends of Distributed Computing Systems, IEEE Computer Society, pp. 308-314, 2003.
[29] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," Proc. ACM MobiHoc, pp. 187–198, 2004.
[30] W. Zhao, M. Ammar, and E. Zegura, "Controlling the mobility of multiple data transport ferries in a delay-tolerant network," Proc. IEEE INFOCOM, pp. 1407-1418, 2005.

Appendix A: Replicator Equation in Evolutionary Game Theory

The replicator equation [9, 21, 24, 25] is one of the fundamental equations in evolutionary game theory. Evolutionary game theory assumes that the population of a group (e.g., species) is proportional to the fitness (i.e., payoff) of the strategy that the group selects. Since each group is under mutual dependency with other groups, the superiority of the strategy is determined relatively by the strategy distribution.

We first formalize the general case for two players with n strategies. An $n \times n$ payoff matrix, $A = [a_{ij}]$, represents all possible strategy pairs of the two players. The entries, a_{ij}, $(i, j = 1, 2, \ldots, n)$, denote the payoff of strategy i competing with strategy j.

Let x_i $(i = 1, 2, \ldots, n)$ denote the ratio (relative frequency) of each strategy. All x_i add up to 1, i.e.,

$$\sum_{i=1}^{n} x_i = 1. \tag{A·1}$$

The expected payoff f_i of strategy i is given by

$$f_i = \sum_{j=1}^{n} x_j a_{ij}. \tag{A·2}$$

We can obtain the average payoff of the population to be

$$\phi = \sum_{i=1}^{n} x_i f_i. \tag{A·3}$$

It then follows from Eqs. (A· 2) and (A· 3) that the standard replicator equation is given as

$$\dot{x}_i = x_i(f_i - \phi), \quad i = 1, \ldots, n, \tag{A·4}$$

where a dot represents time derivative. Eq. (A· 4) indicates that the number of players selecting strategy i increases with the relative difference between the expected payoff of strategy i and the average payoff of all strategies. Note that Eq. (A· 4) is applicable only to an infinitely large and well-mixed population where each player can equally play games with all other nodes [24].

Evolutionary game theory on graphs [11, 13, 14, 16, 17, 22] is an extension of the original theory to a finite size population. Members of a population are represented by vertices of a graph and interact with connected individuals. It describes how the expected frequency of each strategy in a game changes over time within the graphs. The pair approximation [15] is applied to regular graphs of degree $k > 2$, i.e., each individual is connected to k other individuals. Each node represents a player with a selected strategy. Each player derives a payoff from interactions with all of its neighbors. Then, it compares the obtained payoff with a randomly chosen neighbor. If it overcomes the opponent, it keeps the current strategy, and otherwise, it imitates the strategy of the opponent. This kind of strategy updating rule is called "imitation updating rule."

By modifying the original payoff matrix A, the evolutionary game dynamics in a well-mixed population can be transformed into that on a k-regular graph. The modified payoff matrix, $A' = [a'_{ij}]$, is defined by the sum of the original $n \times n$ payoff matrix, $A = [a_{ij}]$, and an $n \times n$ modifier matrix, $M = [m_{ij}]$, where, m_{ij} describes the local competition between strategies i and j [14]. The transformed entries a'_{ij} of the modified payoff matrix, A' becomes

$$a'_{ij} = a_{ij} + m_{ij}.$$

In [14], m_{ij} for the imitation updating rule is defined as for $k > 2$,

$$m_{ij} = \frac{(k+3)a_{ii} + 3a_{ij} - 3a_{ji} - (k+3)a_{jj}}{(k+3)(k-2)}. \quad (\text{A}\cdot 5)$$

Note that off-diagonal elements of matrix M is anti symmetric, i.e., $m_{ij} = -m_{ji}$, because the gain of one strategy in local competition is the loss of another. Further, diagonal elements m_{ii} are always zero, suggesting that local competition between the same strategies results in zero. The expected payoff g_i for the local competition of strategy i is defined as

$$g_i = \sum_{j=1}^{n} x_j m_{ij}. \quad (\text{A}\cdot 6)$$

Note that the average payoff of the local competition of strategy i sums to zero, i.e.,

$$\sum_{i=1}^{n} x_i g_i = 0. \quad (\text{A}\cdot 7)$$

We thus obtain the average payoff ϕ of the population on graph to be

$$\phi = \sum_{i=1}^{n} x_i (f_i + g_i) = \sum_{i=1}^{n} x_i f_i, \quad (\text{A}\cdot 8)$$

which is the same as Eq. (A·3).

Let x_i denote the frequency of strategy i on a k-regular graph. Replicator equation on graphs can be obtained as follows [14, 15, 17]:

$$\dot{x}_i = x_i(f_i + g_i - \phi), \quad i = 1, \ldots, n, \quad (\text{A}\cdot 9)$$

where f_i, g_i, and ϕ are given in Eqs. (A·2), (A·6), and (A·8), respectively.

It is interesting to observe that Eq. (A·9) takes the same form as the standard replicator equation in Eq. (A·4), where the payoff matrix $[a_{ij}]$ is replaced by $[a_{ij} + m_{ij}]$. Therefore, many aspects of evolutionary dynamics on graphs can be analyzed by studying a standard replicator equation with the transformed payoff matrix $[a_{ij} + m_{ij}]$. Note that as k increases, the relative contribution of g_i decreases, compared to f_i, and in the limit of $k \to \infty$, Eq. (A·9) is reduced to Eq. (A·4). Therefore the replicator equation on a highly connected graph converges to the standard replicator equation [14].

K. Habibul Kabir was born in Bangladesh. He received B.Eng. in electrical and electronics engineering from Islamic University of Technology (IUT), Bangladesh, in 2002. He received M.Eng. in electrical, electronic and information system engineering from Osaka University, Japan, in 2007. He is currently working toward the Ph.D. degree at Osaka University. He joined as a Lecturer in the Electrical and Electronics Engineering Department of Islamic University of Technology, Bangladesh in 2002, and till date he is on leave of absence from the university. His research interests include design and performance evaluation of delay tolerant networks. He is a student member of IEEE.

Masahiro Sasabe received the M.E. and Ph.D. degrees from Osaka University, Osaka, Japan, in 2003 and 2006, respectively. He is currently an Assistant Professor with the Department of Information and Communication Technology, Osaka University. From 2003 to 2004, he was a Research Fellow with 21COE-JSPS, Japan. From 2004 to 2007, he was an Assistant Professor with the Cybermedia Center, Osaka University. His research interests include QoS architecture for multimedia distribution system, P2P communications, and ubiquitous networking.
Dr. Sasabe is a member of IEEE.

Tetsuya Takine is currently a Professor in the Department of Information and Communications Technology, Graduate School of Engineering, Osaka University, Suita, Japan. He was born in Kyoto, Japan, on November 28, 1961, and received B.Eng., M.Eng., and Dr.Eng. degrees in applied mathematics and physics from Kyoto University, Kyoto, Japan, in 1984, 1986, and 1989, respectively. In April 1989, he joined the Department of Applied Mathematics and Physics, Faculty of Engineering, Kyoto University, as an Assistant Professor. Beginning in November 1991, he spent one year at the Department of Information and Computer Science, University of California, Irvine, on leave of absence from Kyoto University. In April 1994, he joined the Department of Information Systems Engineering, Faculty of Engineering, Osaka University as a Lecturer, and from December 1994 to March 1998, he was an Associate Professor in the same department. From April 1998 to May 2004, he was an Associate Professor in the Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University. His research interests include queueing theory, emphasizing numerical computation, and its application to performance analysis of computer and communication networks. He is now serving as an area editor of Operations Research Letters and an associate editor of Queueing Systems, Stochastic Models, and International Transactions in Operational Research. He received Telecom System Technology Award from TAF in 2003, and Best Paper Awards from ORSJ in 1997, from IEICE in 2004 and 2009, and from ISCIE in 2006. Dr. Takine is a fellow of ORSJ and a member of IEEE, IPSJ, and ISCIE.