Proposal and Evaluation of Ant-based Routing with Autonomous Zoning for Convergence Improvement

Naomi Kuze, Naoki Wakamiya, and Masayuki Murata
Graduation School of Information Science and Technology
Osaka University
1-5 Yamadaoka, Suita-shi, Osaka 565-0871, Japan
{*n-kuze, wakamiya, murata*}*@ist.osaka-u.ac.jp*

***Abstract*—To tackle problems emerging with rapid growth of information networks in scale and complexity, bio-inspired self-organization is considered one of promising design principles of a new generation network which is scalable, robust, adaptive, and sustainable. However, self-organizing systems would fall into a local optimum or never converge under some environmental conditions. Controlled or guided self-organization is a novel concept attracting many researchers in these years, where loose and moderate control is imposed on a self-organizing system to push it toward a desired state. In this paper, we take AntNet, an ant-based routing protocol, as an example and consider a mechanism to accelerate convergence by limiting the search space. The proposed mechanism is compared with AntNet and HOPNET from viewpoints of the convergence time, path length, and control overhead. Simulation results show that our proposal can accelerate convergence of ant-based routing to a shorter path than AntNet and with lower control overhead than HOPNET.**

***Keywords*-self-organization, Ant Colony Optimization (ACO), zone-base routing, convergence**

I. INTRODUCTION

Due to rapid growth of information networks in scale and complexity, conventional information network systems and technologies, which are based on central control or distributed control with global information, are to face limitations. An information network system adopting conventional control technologies suffers from the considerable overhead in managing up-to-date information to grasp dynamically changing conditions as the scale and mobility increase. Considering the problems that would emerge in the future networking, there have been research activities such as FIND [1], GENI [2] in the USA, Euro-NGI/FGI [3] and FIRE [4] in Europe, and the AKARI Project [5] in Japan to establish a novel network architecture and relevant technologies. Taking into account the requirements for new generation networks, i.e. scalability, adaptability, robustness, and sustainability higher than ever before, the paradigm shift is needed to organize and control the whole network system in a fully distributed and self-organizing manner.

Self-organization is a natural phenomenon of distributed systems, where components behave individually and autonomously. In a self-organizing system, components behave in accordance with simple rules and information locally available to a component. Though direct or indirect interaction among components, a global behavior or pattern emerges on a macroscopic level without central control. In a self-organizing system, system-level information management can be considerably reduced since none needs up-to-date information of the entire system or many other components. Moreover, local failures and small environmental changes are handled locally and immediately by neighbor components without involving the entire system.

However, it is pointed out that self-organizing control has some disadvantages [6]. First, in a large-scale system, it may take long time for a global pattern to emerge because it appears as a consequence of interaction between autonomous components. Second, self-organization, which uses only local information, would fall into a local optimum while a conventional system using global information can reach an optimal solution in most cases. Furthermore, a self-organizing system is not controllable in general, whereas unnecessity of control is one of the significant aspects of self-organization. These disadvantages and complaints about them from engineers brought an idea of controlled or guided self-organization where a self-organizing system is moderately controlled through a feedback mechanism or adaptation of control parameters [7], [8], [9], [10].

Biological systems are inherently self-organizing and the biology is one of mines of self-organization models that can be applied to information networking [11], [12]. For example, foraging behavior of ants is well-known biological self-organization [13]. An ant coming out from a nest randomly wanders around looking for food. When it finds food, it returns to the nest while leaving chemical substances called *pheromone* on the ground. Pheromones laid on the ground attract other ants and guide them to the food. Since pheromone stimulation is not deterministic, there are ants that reach the food by taking other routes. They make other pheromone trails and they also attract ants. However, since pheromone evaporates over time, a shorter trail has more pheromone than longer trails. As such, more ants traverse a shorter trail and those ants further add pheromone on the trail. This feedforward-based reinforcement mechanism makes ants concentrated on the shortest path.

Because of the similarity, the foraging behavior of ants or its mathematical model called ACO (Ant Colony Optimization) has been adopted as a routing mechanism by many researchers [14], [15], [16]. Previous research shows that AntNet is superior to conventional mechanisms in robustness against failure, control overhead, and communication performance [17]. However, the time required for path establishment to converge depends on the length of the path, i.e. the distance between a source node and a destination node [18]. Therefore, it is not scalable to the size of a network. Moreover, a considerable amount of control messages generated in path establishment deplete network bandwidth and hinder data message transmission.

In this paper, we take AntNet as an example of self-organization based control and propose a mechanism of controlled self-organization to accelerate path convergence. The poor convergence performance of AntNet comes from the fact that ants randomly explore the whole network for a destination node and once found preferential pheromone reinforcement is performed on paths established across the whole. Then we consider to reduce convergence time by limiting the area of exploration of ants. The whole network is autonomously divided into subareas by ants and path finding and establishment are performed within a subarea. The path from a source node to a destination node can be formed by concatenating sub-paths. Though simulation experiments, we compare the proposal with the original AntNet and HOPNET from viewpoints of the convergence time, path length, and control overhead. HOPNET is a hybrid mechanism adopting ant-based routing in proactive and reactive manners [19]. In HOPNET, zones are constructed per node in accordance with the proximity of nodes. The size of zone is determined by a parameter $radius$ and a zone of node k consists of a set of nodes which can be reached in $radius$ hops from node k. Each node proactively maintains paths for nodes within its zone and reactively finds a path for a destination node if a destination node is outside its zone.

In this paper, we first describe abstract of AntNet in section II and detail of our proposal in section III. Then, we perform comparative analysis with AntNet and HOPNET in section IV. Finally in section V we conclude the paper and show future directions.

II. AntNet

In AntNet, a next hop node is selected in accordance with the amount of pheromones, which are laid by control messages, called ant. Source node s establishes and maintains a path to destination node d by sending *forward ants* at regular intervals. A forward ant stochastically selects a next hop node to visit. The probability p_{nd} that neighbor node $n \in N_k$, where N_k is a set of neighbor nodes of node k, is selected as a next hop node of node k for destination node d is given as follows. If there is no pheromone information for destination node d at node k, a next hop node is randomly chosen.

$$p_{nd} = \begin{cases} 1 & (|N_k| = 1) \\ \frac{1}{|N_k|-1} & (|N_k| > 1, n \neq v_{i-1}) \\ 0 & (otherwise) \end{cases} \tag{1}$$

Otherwise, selection is performed based on the pheromone value τ_{nd}.

$$p_{nd} = \begin{cases} 1 & (|N_k| = 1) \\ \frac{1}{|N_k|-1} & (|N_k| > 1, \forall n \in V_{s\to k}, n \neq v_{i-1}) \\ \frac{\tau^k_{nd}+\alpha l_n}{1+\alpha(|N_k|-1)} & (|N_k| > 1, n \notin V_{s\to k}) \\ 0 & (otherwise) \end{cases} \tag{2}$$

$V_{s\to k} = \{s, v_1, v_2, \cdots, v_{i-1}\}$ is a list of nodes that the forward ant has visited before arriving at node k at the i-th step and v_{i-1} is an identifier of the $(i-1)$-th node on the path. l_n is a variable indicating the degree of congestion for neighbor node n at node k, which is given by $1 - \frac{q_n}{\sum_{j\in N_k} q_j}$ and q_n is the number of messages waiting in a sending buffer for neighbor node n. $\alpha \in [0, 1]$ is a coefficient ranging from 0.2 to 0.5 [15]. A larger α allows forward ants to select a next hop node in accordance with local traffic condition. As a consequence, path convergence becomes hard to accomplish. On the contrary, with α close to 0, a path traversing congested links would be established. A forward ant whose travelled hop count reaches the predetermined TTL is discarded at a node.

A forward ant changes to a *backward ant* when it reaches the destination node d and returns to the source node s following the path that the forward ant traversed while updating pheromone values at visited nodes. The pheromone value τ^k_{nd} for neighbor node $n \in N_k$ at node k is updated by Eq. (3).

$$\tau^k_{nd} \leftarrow \begin{cases} \tau^k_{nd} + r(1-\tau^k_{nd}) & (n = f) \\ \tau^k_{nd} - r\tau^k_{nd} & (n \in N_k - \{f\}) \end{cases} \tag{3}$$

f corresponds to the previous node that the backward ant visited just before arriving at node k, i.e. the first node of the path from the node to the destination node. r reflects the goodness of the path, which is calculated on the transmission delay from node k to the destination node d. The shorter the path is, the larger r is. Consequently, the shortest path among paths that forward ants found has the largest amount of pheromones and attracts most of forward ants.

A data message also selects a next hop node based on pheromone values, where the selection probability R^k_{nd} that neighbor node n is chosen as a next hop node for destination node d is given as $\frac{(\tau^k_{nd})^\epsilon}{\sum_{j\in N_k}(\tau^k_{jd})^\epsilon}$ $(\epsilon \geq 0)$. Therefore, data messages follow the shortest path established by forward and backward ants. For further details of AntNet refer to [17].

III. AntNet Routing Mechanism with Autonomous Zoning

In this section, we propose an ant-based routing mechanism with autonomous zoning to limit the area of exploration of forward ants and accelerate path convergence. First we give an overview of our proposal and then the following sections describe details.

A. Overview

In our proposal, there are two types of forward ants, called *exploration ants* and *maintenance ants* as illustrated in Fig. 1. To establish a path to a destination node, a source node sends exploration ants at regular intervals. An exploration ant wanders looking for a destination node like a forward ant of AntNet, but it has a new role to determine a border of zones. When the TTL expires, an exploration ant sets the halfway node a *border node*. We call an area surrounded by border nodes *zone*.

An exploration ant setting a border node changes to a backward ant and returns to the source node while leaving pheromones on visited nodes in similar to a backward ant of AntNet. Then, on receiving the backward ant the source node begins sending maintenance ants in addition to exploration ants to maintain and improve the path to the border node. Each maintenance ant goes to a border node by selecting next hop nodes in a stochastic manner like a forward ant and returns to the source node as a backward ant to update pheromone values. Simultaneously an established border node begins sending exploration ants to look for a destination node. They also set border nodes on expiration of TTL and go back to the border node. On their return, the border node sends maintenance ants to maintain paths to new border nodes. By repeating the exploration, border nodes are scattered over a network among which the shortest path is established by maintenance ants.

When any exploration ant finds a destination node, a backward ant returns to its originating border node. It raises the ceiling of the amount of pheromone at nodes on the path so that those nodes can deposit more pheromones than nodes on other paths. The backward ant eventually returns a border node from which the corresponding forward ant departed. Then backward ants which depart from the border node also begin raising the limit at nodes on the path to the previous border node. Consequently, a chain of paths from the source node to the destination node have stronger pheromones than other paths leading to nodes other than the destination node. Finally, the source node starts emission of data messages.

Each node has a pheromone table T^k as routing information. $T^k = \{T^k_d\}$ where T^k_d is a list of pheromone values τ^k_{nd} for all neighbor node $n \in N_k$ for destination node d, i.e. $T^k_d = \{\tau^k_{nd}\}$. N_k is a set of neighbor nodes of node k. T^k_d is made when node k has or receives a message to send to destination node d and discarded when it is not used for

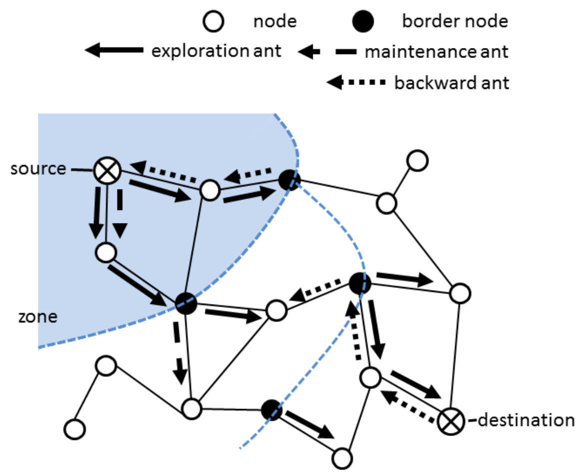

Figure 1. Snapshot of our proposal

a fixed period of time, i.e. 100 times as long as an interval of ant emission. At the beginning, τ^k_{nd} is initialized to $\frac{1}{|N_k|}$. The pheromone value is updated by backward ants, but we set limits on the amount of pheromones that a node can deposit. The pheromone value used for next-hop selection by ants and data messages.

B. Path Exploration and Autonomous Zoning by Exploration Ant

When a source node intends to start a new session to a destination node for which it does not have routing information, it first generates $|N_k|$ exploration ants and sends one exploration ant for each neighbor node at the same time. Then, it keeps sending an exploration ant per interval of Δt. An exploration ant looks for a destination until the number of travelled hops reaches the given TTL of 2ρ hops. If there is no pheromone for destination d on the arrived node, an exploration ant chooses a next hop node at random by Eq. (1). Otherwise, it chooses a next hop node based on the probability p_{nd} given not by Eq. (2) but by Eq. (5), which will be explained later. In the course of the search, an exploration ant records all visited nodes in a visiting order.

If an exploration ant cannot find a destination node or does not arrive at a border node within 2ρ hops, it returns to the node visited at ρ-th hop from the source node and appoints the node as a border node. A border node remembers ρ as its distance from the source node. To avoid placing border nodes next to each other, an exploration ant does not set a border node if there exists any of source, border, or destination node in the immediate vicinity. In this case, an exploration ant immediately dies. Otherwise, an exploration ant which set or arrived at a border node changes to a backward ant and returns to the source node. Each node that receives a backward ant from neighbor node f updates pheromone values for destination node d by

$$\tau_{nd}^k \leftarrow \begin{cases} \min(\tau_{nd}^k + r(1-\tau_{nd}^k), \theta) & (n = f) \\ \max(\tau_{nd}^k - r\tau_{nd}^k, & \\ \quad \tau_{nd}^k - r\tau_{nd}^k + \frac{\tau_{nd}^k + r(1-\tau_{nd}^k)-\theta}{|N_k|-1}) & (n \in N_k - \{f\}) \end{cases} \tag{4}$$

where θ is an upper limit of the amount of pheromone and it is set as $\theta = \theta_l$ $(0 < \theta_l < 1)$. An exploration ant reaching a destination node also becomes a backward ant, but its behavior is different from a backward ant of AntNet and will be described in section III-D.

Similarly to a source node, a new border node generates $|N_k|$ exploration ants and sends one exploratory ant for each neighbor node at the same time. Then, it keeps sending one exploration ant per interval of Δt. An exploration ant sent from a border node looks for a destination node, sets a new border node, and changes to a backward ant as an exploration ant from a source node does. In this way, border nodes are distributed in a network. An exploration ant remembers the distance of the originating border node from the source node. A new border node created by the exploration ant considers its distance as the sum of the distance of the previous border node that the exploration ant remembers and the number of hops that the exploration ant made from the previous border node. Therefore, the distance that a border node remembers is always a multiple of ρ. The distance information is used to avoid making a path going back to the source node. When an exploration ant arrives at a border node which is not the originating border node, it first compares its distance from the source node with that of its originating node. If the arrived border node is closer to the source node, it immediately dies. Otherwise it returns to the originating border node as a backward ant.

In order to scatter border nodes, i.e. bases of exploration, over a network an exploration ant selects a next hop node by avoiding pheromones that show a path to other border node. For this purpose, we use the following equation to give the probability p_{nd} that an exploration ant whose previous node is f chooses neighbor node $n \in N_k - \{f\}$ as a next hop node for destination node d at node k [20].

$$p_{nd} = \frac{\frac{1}{\tau_{nd}^k}}{\sum_{j \in N_k} \frac{1}{\tau_{jd}^k} - \frac{1}{\tau_{fd}^k}} \tag{5}$$

If node f is the only neighbor node of node k, an exploration ant moves to node f.

C. Path Maintenance by Maintenance Ant

On reception of a backward ant, a source node and border nodes start sending maintenance ants to all neighbor nodes whose pheromone value τ_{nd}^k is larger than the initial pheromone value, i.e. $\frac{1}{|N_k|}$, at regular intervals Δt. A maintenance ant chooses a next hop in accordance with the amount of pheromones to maintain an existing path to a border node. A maintenance ant whose travelled hop count reaches the given TTL, i.e. 2ρ hops, is discarded at the node.

The probability p_{nd} that a maintenance ant which arrives at node k from node f chooses neighbor node $n \in N_k - \{f\}$ as a next hop for destination node d is given by Eq. (6). τ_{nd}^k shows the pheromone value for neighbor node $n \in N_k$ at node k for destination node d.

$$p_{nd} = \frac{\tau_{nd}^k}{\sum_{j \in N_k} \tau_{jd}^k - \tau_{fd}^k} \tag{6}$$

If node f is the only neighbor node of node k, a maintenance ant moves to node f.

Similarly to an exploration ant, a maintenance ant reaching a destination node becomes a backward ant to raise the ceiling of pheromone values and leave pheromones as will be described in section III-D. A maintenance ant becomes a backward ant to reinforce its path when it arrives at a border node except for the case that the arrived border node is closer to the source node than the originating border node.

D. Construction of Paths between Source and Destination

In our proposal, paths are constructed and maintained on a per zone basis. A path from a source node to a destination node is established as a chain of those sub-paths of zones which lie between them. So that maintenance ants and data messages are concentrated on the path leading to the destination node, we need to differentiate it from other paths on each node. For this purpose, we put another upper limit θ_h $(0 < \theta_l < \theta_h = 1)$ on the amount of pheromone deposited at a node.

An upper limit of θ_l is applied to all τ_{nd}^k when node k makes entry T_d^k for a new session to destination node d. Eventually node k receives a backward ant departing from a destination node or a next border node to a destination node if its zone is located between the source node and the destination node. Then, it raises the ceiling to θ_h and stops sending exploration ants. At the same time, the border node sets a timer at ten times as long as Δt, an interval of ant emission. The timer is restarted when it receives either of an exploration ant, a maintenance ant, or a backward ant with a role of threshold raising. When the timer expires, the threshold returns to θ_l.

In addition to the favoring mechanism, we have a pruning mechanism to reduce redundant border nodes after a path from a source node to a destination node is found. A border node using θ_l starts a pruning timer at 100 times as long as Δt when it receives an exploration ant or a maintenance ant departing from a border node using θ_h. It implies that the border node is located at the border of zone having a path to the destination node but not on the path itself. The pruning timer is cancelled and discarded when the border node can raise the threshold to θ_h. When the pruning timer expires on the other hand, it stops generating both of exploration ants and maintenance ants. Then, it moves to a standby state. In the standby state, as explained in section III-A, a node set another timer, waits for its expiration, and discards the pheromone list T_d^k on timer expiration. In addition a border node which receives a backward ant departing from a border node in the standby state also starts a pruning timer. It eventually moves to the standby mode unless it raises the threshold until timer expiration and finally discards T_d^k. In this way, zones next to those zones constituting a path to the destination node disappear first and then pruning proceeds to the edge of a network.

IV. Simulation Results

We evaluate and compare our proposal with AntNet and HOPNET from viewpoint of the convergence time, path length, and control overhead.

A. Simulation Setting

We change the size of network while keeping the node density. We experimentally distribute 150 nodes at random locations in the area of 200 m×200 m. We call this setting $scale = 2$. Then, we consider different sizes from $scale = 1$ (100 m×100 m) to 10 (1,000 m×1,000 m). Independently of the size of network we set the communication range at 30 m and the one-hop transmission delay at 2 msec. We appoint a node at the top-left corner as a source node and one at the bottom-right corner as a destination node. The interval Δt of control message emissions is set at 10 msec in AntNet and our proposal. The TTL parameter ρ is set at 3 in our proposal. Other parameters of AntNet and HOPNET are set in accordance with their default settings [15], [19]. Data messages are not generated in simulation experiments.

Regarding performance measures, the convergence time is defined as the time from the beginning of a simulation run where no routing information exists in a network till when the same path is selected for 10 consecutive times or a cyclic selection of the fixed set of paths in the same order, e.g. path A, path B, path C, path A, path B, and path C, for 100 consecutive times. Convergence check is done everytime a backward ant reaches a source node. In the case of HOPNET, reactive path establishment is performed for a destination node outsize a zone. Therefore in our simulation we first allow nodes to proactively establish paths within zones for 300 msec. Then we start construction of a path from the source node and the destination node. Taking into account this the convergence time of HOPNET is defined as the time from 300 msec to convergence. The path length is defined as the number of hops of a created path. The control overhead is defined as the total number of travelled hops of control messages until convergence.

Figure 2 shows an example of paths created by AntNet and our proposal in networks of $scale = 3$. Each small dot corresponds to a node and double circles are border nodes. A filled circle at a top-left corner is a source node and one at a bottom-right corner is a destination node. Each thin line means that nodes of its ends can communicate with each other, that is, they are neighbors. Thick lines show a path established by ants. In the figure, the path length of AntNet is 49 hops and that of our proposal is 25 hops. In the following figures, we show results averaged over 100 simulation runs for each $scale$ except for cases that convergence cannot be achieved by the end of a simulation run.

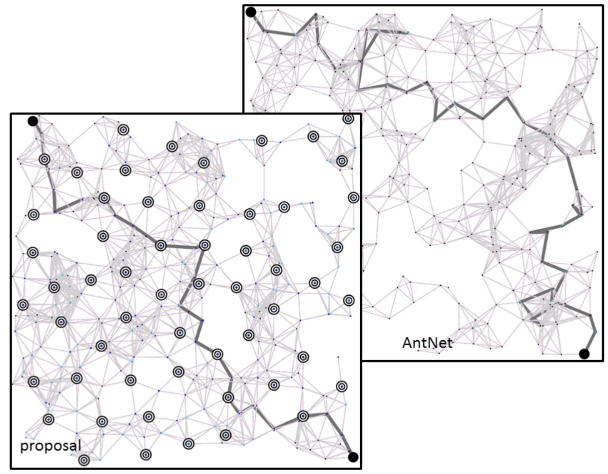

Figure 2. Examples of established path

B. Comparison with AntNet and HOPNET

Figure 3 shows the average convergence time against different $scale$ setting. As shown in the figure, the convergence time of our proposal is much smaller than that of AntNet because path convergence is accelerated by zoning. In ant-based routing, a path has to accumulate pheromones as fast as possible and distinguish an appropriate neighbor node from other neighbors in next-hop selection for faster convergence. However, in AntNet, nodes on a path have to wait long for a backward ant to come from a distant destination node in order to increase the amount of pheromone. While it is waiting, forward ants visiting the node randomly choose a next hop node. Then, there appear multiple paths from a source node to a destination node, which disturbs fast convergence. In addition, a forward ant explores the whole area with AntNet. It delays the discovery of a path. On the contrary, because of zoning, exchanges of forward and backward ants are limited within a zone with our proposal. The length of path connecting a pair of a source node and a border node, two border nodes, or a border node and a destination node, is considerably shorter than a path between a source node and a destination node. As a result, the speed of reinforcement of a path is much faster with our proposal than AntNet. In conclusion our proposal is more scalable than AntNet.

HOPNET converges the fastest among the three mechanisms. The first reason is that paths within a zone have

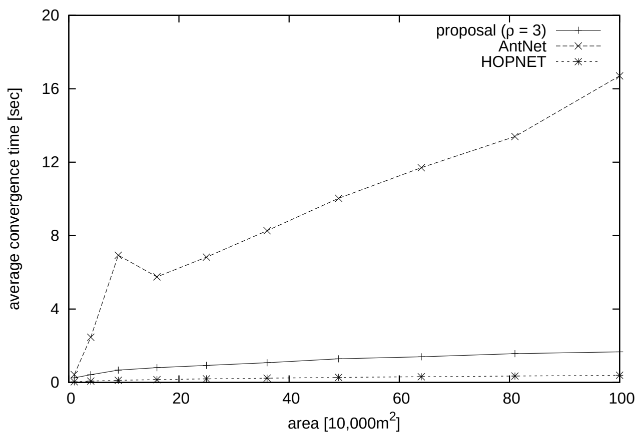

Figure 3. Average convergence time

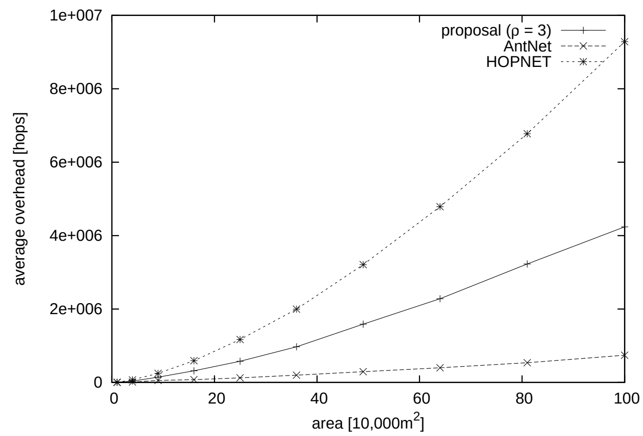

Figure 5. Average overhead

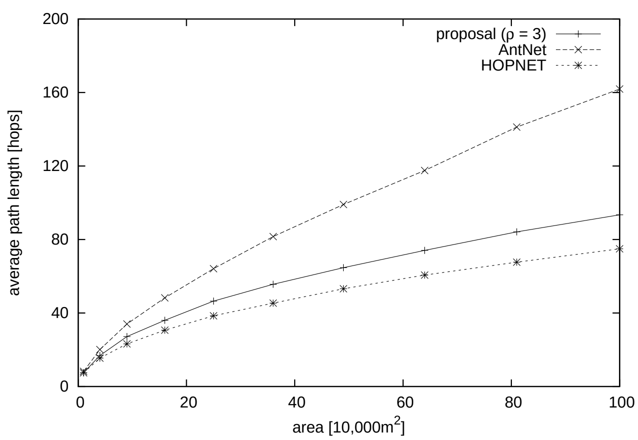

Figure 4. Average path length

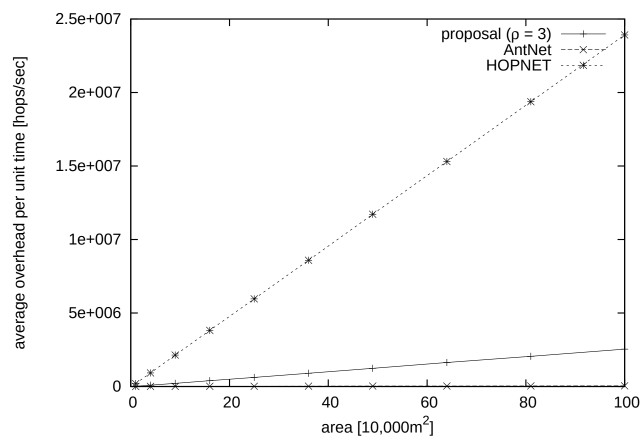

Figure 6. Average of overhead per unit time

already been established before a session starts in our simulation. In addition, with HOPNET a source node sends forward ants to all nodes at a border of its zone if it does not have routing information for a destination node. The greedy exploration is repeated by border nodes and as a result the fast path discovery is accomplished. Finally, HOPNET uses a path which is found first and does not search for a better path. Therefore, when a path is found, HOPNET converges.

We also evaluate the path length constructed by each mechanism in Fig. 4. Since HOPNET conducts greedy and exhaustive search, the length of a path becomes close to the optimal shortest path. Compared with AntNet, our proposal constructs shorter paths. Whereas exploration ants with AntNet search for a destination node making a random walk across the whole network and often make an indirect path as shown in Fig. 2, our proposal gradually expands the scope of exploration by pushing the front line consisting of border nodes from which exploration ants perform local search. Furthermore, our proposal does not allow exploration ants to go back to a source node. This mechanism also contributes in making a shorter path than AntNet. However, a chain of border nodes does not always form the shortest path from a source node to a destination node. As a result, the path length with our proposal is slightly longer than that of HOPNET. Figures 3 and 4 show that HOPNET is the best among three, but it is at the sacrifice of considerable overhead as will be shown in the next.

Finally, Fig. 5 shows the average overhead, i.e. the total number of hops of control messages before convergence. A reason why HOPNET incurs significant overhead is that it conducts a greedy and exhaustive search by multicasting control messages. The control overhead of our proposal is larger than that of AntNet because of zone-based search and introduction of two types of ants. We also evaluate the amount of overhead per unit time, which is derived by dividing the amount of overhead by the convergence time for each of $scales$, in Fig. 6. As shown in the figure, HOPNET puts the considerable control overhead on a network during

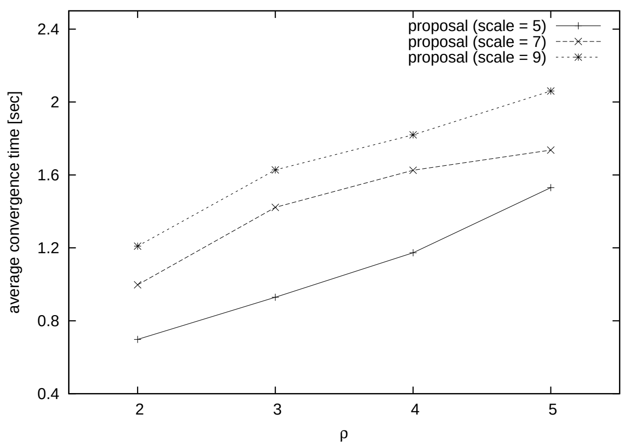

Figure 7. Parameter ρ and average convergence time

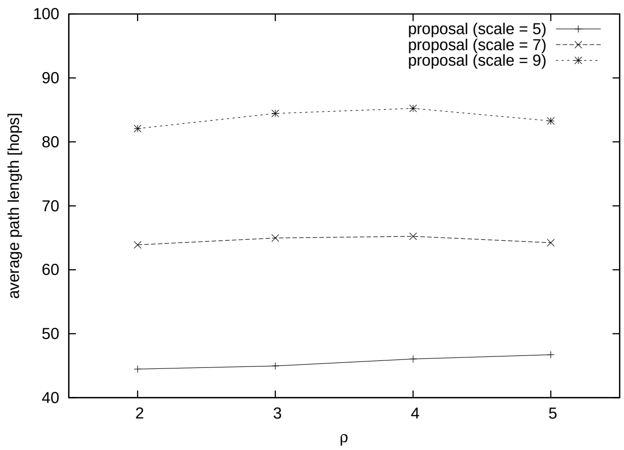

Figure 8. Parameter ρ and average path length

the short period of time of path establishment. It easily causes heavy congestion and both of path establishment and data communication would fail. In conclusion HOPNET is not scalable to a large network with many nodes and many sessions. On the contrary, the instantaneous overhead is small with our proposal. A reason that the overhead per unit time is quite small with AntNet is that it takes long to converge.

C. Influence of parameter ρ

In our proposal, convergence is accelerated by limiting the area of exploration with parameter ρ. We conduct simulation experiments by changing ρ from 2 to 5 in networks of $scale$ 5, 7, and 9 to investigate the influence of ρ. Figure 7 shows that the average convergence time increases with a larger ρ independently of the size of a network as can be presumed from a curve of AntNet in Fig. 3. Furthermore, the average path length is not affected by parameter ρ as shown in Fig. 8 and the average overhead is the smallest with the smallest ρ as shown in Fig. 9 independently of the

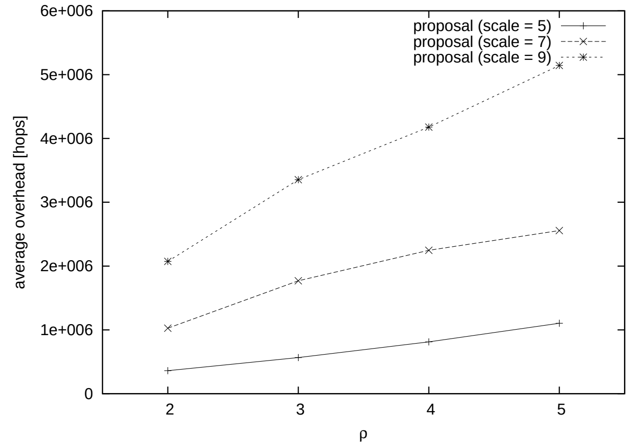

Figure 9. Parameter ρ and average overhead

$scale$ of a network. In conclusion, $\rho = 2$ provides the best performance particularly with the current setting.

A reason that parameter ρ has little influence on the average path length is as follows. Each exploration ant explores the area avoiding pheromones based on the probability given by Eq. (5) to scatter border nodes. Eventually border nodes are densely distributed in a network independently of the value of ρ. The numbers of border nodes at convergence are 155.5, 147.4, 146.3, and 146.0 with $\rho = 2$, 3, 4, and 5 in a network $scale = 5$, respectively. As a result, the lengths of paths constructed by connecting border nodes become similar to each other. Finally since the time required for path establishment to converge increases with a larger ρ as shown in Fig. 7 and control messages are generated throughout the time, the average control overhead increases with a larger value of ρ.

V. Conclusion

In a self-organizing system, the global pattern emerges as a consequence of mutual interaction among individuals. In a case of ant-based routing, a path is constructed through interaction among ants mediated by pheromones. In this paper, as an example of controlled self-organization, we propose and evaluate a mechanism to accelerate path convergence of AntNet by limiting the search space. Simulation results show that our proposal can facilitate path establishment and make ant-based routing more scalable to the size of network.

As discussed in [21] self-organization can also be accelerated by prediction. We now consider controlled self-organization with prediction by extending our proposal, where each ant determines its behavior by predicting future direction of self-organization from locally available information.

Acknowledgements

This research was supported in part by “International Collaborative Research Grant” of National Institute of In-

formation and Communications Technology, Japan.

REFERENCES

[1] *Future internet design (FIND)*, National Science Foundation, http://www.nets-find.net/.

[2] *Global environment for network innovations (GENI)*, National Science Foundation, http://www.geni.net.

[3] *Network of excellence EuroNGI and EuroFGI design and engineering of the next generation internet*, European Commission, http://eurongi.enst.fr.

[4] *Future Internet Research & Experimentation*, European Commission, http://cordis.europa.eu/fp7/ict/fire/.

[5] *New generation network architecture AKARI conceptual design*, AKARI project, October 2007.

[6] F. Dressler, *Self-organization in sensor and actor networks*. Wiley, January 2008.

[7] C. Müller-Schloer, H. Schmeck, and T. Ungerer, *Organic Computing-A Paradigm Shift for Complex Systems*. Birkhauser Verlag AG, May 2011.

[8] M. Prokopenko, "Guided self-organization," *HFSP Journal*, vol. 3, pp. 287–289, October 2009.

[9] S. Arakawa, Y. Minami, Y. Koizumi, T. Miyamura, K. Shiomoto, and M. Murata, "A managed self-organization of virtual network topology controls in WDM-based optical networks," *Journal of Optical Communications*, vol. 32, no. 4, pp. 233–242, December 2011.

[10] D. Kominami, M. Sugano, M. Murata, and T. Hatauchi, "Controlled potential-based routing for large-scale wireless sensor networks," in *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems 2011*, December 2011, pp. 187–196.

[11] F. Dressler and O. Akan, "A survey on bio-inspired networking," *Computer Networks*, vol. 54, no. 6, pp. 881–900, April 2010.

[12] M. Meisel, V. Pappas, and L. Zhang, "A taxonomy of biologically inspired research in computer networking," *Computer Networks*, vol. 54, no. 6, pp. 901–916, April 2010.

[13] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial systems*. Oxford University Press, USA, October 1999.

[14] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz, "Ant-based load balancing in telecommunications networks," *Adaptive behavior*, vol. 5, no. 2, pp. 169–207, January 1997.

[15] G. Di Caro and M. Dorigo, "AntNet: Distributed stigmergetic control for communications networks," *Arxiv preprint arXiv:1105.5449*, vol. 9, pp. 317–365, December 1998.

[16] G. Di Caro, F. Ducatelle, and L. Gambardella, "AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks," *European Transactions on Telecommunications*, vol. 16, no. 5, pp. 443–455, September 2005.

[17] S. Dhillon and P. Van Mieghem, "Performance analysis of the antnet algorithm," *Computer Networks*, vol. 51, no. 8, pp. 2104–2125, June 2007.

[18] L. Carvelli and G. Sebastiani, *Some Issues of ACO Algorithm Convergence*. InTech, February 2011, ch. 4.

[19] J. Wang, E. Osagie, P. Thulasiraman, and R. Thulasiram, "HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network," *Ad Hoc Networks*, vol. 7, no. 4, pp. 690–705, June 2009.

[20] Y. Ohtaki, N. Wakamiya, M. Murata, and M. Imase, "Scalable and efficient ant-based routing algorithm for ad-hoc networks," *IEICE Transactions on Communications*, vol. 89, no. 4, pp. 1231–1238, 2006.

[21] H. Zhang, M. Chen, G. Stan, T. Zhou, and J. Maciejowski, "Collective behavior coordination with predictive mechanisms," *IEEE, Circuits and Systems Magazine*, vol. 8, no. 3, pp. 67–85, August 2008.