Improving Path Discovery Performance
in Dense Mobile Ad Hoc Networks

Yujin Noishiki,[†] Hidetoshi Yokota[†] and Akira Idoue[†]

In on-demand ad-hoc routing protocols, control packets are flooded into a network during path discovery. Especially in dense ad-hoc networks, these protocols generate a large number of broadcast packets that cause contention, packet collisions and battery power wastage in mobile nodes. We propose an efficient route establishment method that adaptively lowers re-broadcasting overhead based on the number of adjacent nodes and the number of routes that these nodes accommodate. Through simulation, we demonstrate that our proposed method is especially efficient in densely populated areas. It achieves efficiency by lowering the number of control packets for path discovery without causing a drop in the path discovery success ratio. Also, by taking path concentration into account, our method further improves packet delivery. We provide useful guidelines based on simulation results.

1. Introduction

With the advent of wireless access technologies such as WiFi, Bluetooth, and RF-ID, a number of devices can be connected to each other to exchange information. These are not only stationary devices, but can also be mounted on mobile entities such as motor vehicles, thereby establishing networks and making advanced telematics a reality. Connectionless packet delivery may be an ideal match for these networks in terms of ubiquity and mobility, but packets must be routed in a network that dynamically changes its form without the presence of servers. These distributed and self-organized networks can be viewed as mobile ad-hoc networks [1] and various types of routing protocols have been designed and analyzed to support them.

There are two categories of ad-hoc routing schemes. One is table-driven routing, which maintains routing tables as conventional fixed networks, and the other is source-initiated on-demand routing, which attempts to establish a route to a destination when a communication request occurs. Between these two categories, on-demand routing algorithms are more suited to devices with less memory since this type of mechanism initiates path discovery upon receiving communication requests. Therefore, the total number of nodes is less important for these protocols. The Ad-hoc On-demand Distance Vector [2] (AODV) has been thoroughly investigated and is considered a standard by

the Internet Engineering Task Force (IETF). While many studies on this protocol assume the size of the network to be several dozen mobile nodes, the number of nodes may actually exceed one hundred, such as when many cars in a metropolitan area are using wireless devices or when many people gather and communicate within a small place. For this reason, performance in a highly populated ad-hoc network requires further study.

One challenging issue is how to reduce flooding during path discovery, since wireless bandwidth is limited. We propose an ad-hoc routing scheme that stochastically adapts the number of packets for path discovery based on the number of adjacent nodes. From the number of control packets transmitted by other nodes, a node infers how many nodes are adjacent to it. Then, when a node broadcasts a control packet, some of the adjacent nodes can skip re-broadcasting on the assumption that other nodes will re-broadcast. This means that fewer nodes have to re-broadcast as the number of adjacent nodes increases.

If certain nodes hold a greater number of routes than others, then neighboring nodes may also experience congestion due to traffic on those routes. Our method incorporates the number of routes already established on candidate nodes into the path discovery mechanism. Intermediate nodes reflect the number of currently established routes in terms of the probability for re-broadcasting. Routes on those nodes can be distributed based on an average, thereby reducing delay and loss fluctuations. We demonstrate that broadcast packets can

---

† KDDI R&D Laboratories, Inc.

be controlled without lowering the success ratio for establishing routes. We also demonstrate the effectiveness of our proposed algorithms by comparing them with AODV.

The rest of this paper is organized as follows. In Section 2, we describe related work. In Section 3, we present the characteristics of on-demand routing protocols and problems in dense ad-hoc networks. We then propose an efficient method for establishing routes in dense ad-hoc networks. In Section 4, we evaluate our proposed method with simulations and compare its performance with AODV. Finally, Section 5 concludes this paper.

2. Related Work

Several protocols have been proposed for use in on-demand ad-hoc routing. AODV and Dynamic Source Routing[3] (DSR) have been studied by the IETF. In the path discovery process, nodes broadcast control packets and flood them into the network. This can be a problem because nodes generate a large number of broadcast packets in dense ad-hoc networks.

Many approaches have been proposed to make the path discovery process more efficient[4]. One approach is to apply hybrid routing protocols that combine both table-driven routing and on-demand routing protocols. Zone Routing Protocol[5] (ZRP) is one such hybrid routing protocol that defines the zone of each node the number of hops from the node. The basic idea is that a node uses a table-driven routing protocol inside its zone and an on-demand routing protocol outside its zone. This hybrid approach reduces the overhead of the table-driven scheme that holds the routing information of the entire network. However, if the number of nodes in the ad-hoc networks increases, then the overhead of the information update will increase because of the high node density in the zone.

Clustering routing protocols such as Clusterhead Gateway Switch Routing[6] or Passive clustering[7] have been proposed to lower broadcast overhead. These protocols impose a hierarchical topology on ad-hoc networks. Some nodes are chosen as cluster leaders and serve as cluster representatives. The cluster leaders control routing and, therefore, reduce the number of control packets. However, this means that routes are centralized through cluster leaders, which causes problems such as route vulnerability and battery drainage in the cluster leaders.

Some routing protocols use location information to establish routes. Location Aided Routing[8] (LAR) limits the search for a new route to a smaller request zone of the network by using location information. However, in these routing protocols, all mobile nodes have to implement some sort of positioning system.

Ni, et al.[9] advocates the use of probabilistic broadcasting in path discoveries. Nodes that receive control packets decide to re-broadcast or not based on a suitable probability. The same paper also proposes counter-based schemes for efficient broadcasting. In these schemes, when a node receives a previously unseen packet, the node starts counting the number of control packets that have the same ID for a randomly chosen period. If the number of packets that the node receives is below a certain threshold, the node re-broadcasts the control packet. Otherwise, the node discards it. Using this criterion, this scheme can reduce the number of broadcast packets.

Cartigny, et al.[10] also utilizes probabilistic flooding, where the probability of re-broadcasting flooding messages is obtained from HELLO messages, which are assumed to be sent from each node. However, according to RFC3561 for example, HELLO messages are optional and should only be used if the node is part of an active route. This means that such messages cannot be expected if nodes are not on the active routes. In contrast, control messages for path discovery are flooded in almost all on-demand protocols. To improve path discovery, scheme should use messages that are not dependent on protocol specifications.

To improve performance, some routing protocols select a route according to metrics other than a hop number. Associativity-Based Routing[11] (ABR) selects routes based on nodes with associativity states that are periods of stability. In this scheme, the routes are likely to be long-lived, so there is no need to restart frequently, and performance improves. Our scheme selects a route according to the path concentration, which indicates the number of routes that each node accommodates. A node that already supports many routes does not join in the selection of a new route. The routes are then distributed in an ad-hoc network, and communication performance is improved.

Power aware routing is one of the active issues on ad-hoc networks, and Xu, et al.[12] proposes algorithms that aim to conserve power by us-

ing an approach similar to ours. One of the proposed algorithms turns off the radio by using node density. Each node makes a list of adjacent nodes when it receives packets from them. Then, each node increases the time the radio is turned off in proportion to the number of nodes in its neighborhood. By turning off the radio based on the node density, the proposed algorithm would reduce power consumption. However, when destination nodes turn off the radio, it causes delay or packet loss, which is critical to initiating applications on the destination nodes. From the viewpoint of improving performance, which is our goal, these properties are as important as reducing power consumption.

How the number of adjacent nodes relates to wireless communication performance is discussed by Royer, et al. [13]. If the node density is low, some nodes become isolated and cannot send or receive packets. On the other hand, if the node density is high, then the broadcast packets cause data transmission overhead and congestion. Royer, et al. discussed the use of optimum node density to improve communication performance. However, it is not clear whether the node density is optimum in ad-hoc networks where nodes move dynamically. In our paper, we optimize node density virtually by stochastically broadcasting control packets in our routing scheme.

3. Proposal for Improving Path Discovery Performance

3.1 The Scalability Issue in On-demand Routing Protocols

To establish a route to a destination node, on-demand ad-hoc routing protocols broadcast control packets and flood them into the entire network. When the number of nodes in the ad-hoc network increases, many nodes participate in path discovery and generate a large number of broadcast packets. This causes an overhead due to generation of many control packets and contention, both of which block data transmissions. Furthermore, intermediate nodes suffer from battery power drainage due to transmission of redundant control packets.

If a node discards only redundant control packets when establishing routes, it will improve communication performance. However, if a node is too rigid about eliminating broadcast packets, then some routes cannot be found because there are not enough broadcast pack-

ets to establish the routes. In contrast, when a node does not eliminate redundant packets, the broadcast overhead cannot be prevented. To solve the problem of broadcast overhead, we must take this trade-off into account.

3.2 Adaptive Control of Broadcasting

To overcome the scalability issue in on-demand ad-hoc routing protocols, we propose a new on-demand route establishment method. Our proposed method improves path discovery performance by controlling broadcast packets based on network conditions. We focus on two metrics that reflect network conditions. One metric is node density or, in other words, the number of adjacent nodes. Nodes achieve efficient communication by reducing the number of redundant control packets when the node density metric is high. The other metric is path concentration. Taking the path concentration metric into account, we prevent specific nodes from supporting too many routes and reduce the loss and delay variances. By adaptively controlling broadcast packets based on these two metrics, we create a virtual optimum node density that improves communication performance in dense ad-hoc networks.

Our proposed method deduces node density from the number of control packets sent by adjacent nodes. In the ad-hoc network, each node can move around, changing node density as it moves. Our method infers node density by counting the number of control packets that change based on the number of adjacent nodes. When specific nodes hold many route entries, data packets will converge on their routes. Our method therefore infers the path concentration from the number of routes that the node accommodates. Note that our method does not generate any additional control packets to obtain these two metrics.

3.3 Proposed Method

In the path discovery mechanism, each node counts the number of control packets to deduce the node density. To take dynamic topology changes into account, the metric of node density should be calculated adaptively based on the changes. Then, the mechanism uses the exponential weighted moving average to determine the metric of node density as follows. Let T_j denote a time interval $[jT, (j + 1)T]$, where $j = 0, 1, 2, \ldots$, and T is a unit time interval. Node i receives $N_i(j)$ control packets from adjacent nodes during a time interval T_j. We define the exponential weighted moving average

(EWMA) of the number of control packets that Node i receives as

$$\bar{N}_i(j) = (1 - \omega)\bar{N}_i(j - 1) + \omega N_i(j), \quad (1)$$

where ω is the parameter of the EWMA and $\omega \in [0, 1]$. By using this EWMA, a node can detect changes in node density due to node mobility. Our proposed method then uses this EWMA as the metric to deduce the node density. When Node i has r_i active routes and the total route capacity that the node can hold is denoted as R_i, we define the path accommodation ratio of Node i L_i as $L_i = r_i/R_i$. Our method infers the degree of path concentration of Node i using this metric.

Our proposed method is based on AODV, as an example, and to deduce the node density, it uses RREQ packets that are one of the control packets. At the beginning of path discovery, the source node broadcasts an RREQ packet to adjacent nodes. When Node i that receives an RREQ packet is not the destination node, it re-broadcasts the RREQ packet with the probability determined by the two metrics, $\bar{N}_i(j)$ and L_i.

The criteria used for determining the probability of re-broadcasting are as follows. When the metric of node density is high (low), the probability should be low (high). According to these criteria, nodes can drop the redundant packets when the ad-hoc networks are dense. On the other hand, when the networks are sparse, nodes will re-broadcast control packets. In a similar way, we use the criteria that the probability should be low (high) when the metric of path concentration is high (low). These criteria will keep the specific nodes from holding many route entries. In addition to following these criteria, we consider simplicity and ease of performance in determining the parameters and the equation of re-broadcasting probability.

Let $p_i(k)$ denote the probability of re-broadcasting by Node i in the time interval T_k. We define two policies for determining re-broadcasting probability as follows:

- Policy 1: This policy determines probability by taking only the node density metric into account.

$$p_i(j) = \begin{cases} 1 - \beta & \bar{N}_i(j) > \theta, \\ 1 & \bar{N}_i(j) \le \theta. \end{cases} \quad (2)$$

- Policy 2: This policy determines probability by taking both the node density and path concentration metrics into account.

$$p_i(j) = \begin{cases} 1 - f(L_i) & \bar{N}_i(j) > \theta, \\ 1 & \bar{N}_i(j) \le \theta. \end{cases} \quad (3)$$

β is a non negative parameter and $\beta \in [0, 1]$. θ is a threshold parameter and $\theta \ge 0$. $f(L_i)$ is a monotonically increasing function at range $[0, 1]$. In this paper, we define

$$f(L_i) = \min(\alpha L_i, 1), \quad (4)$$

where α is a positive parameter.

4. Performance Evaluation

4.1 Simulation Environment

In this section, we discuss the results of the simulations used to evaluate the qualitative performance of our proposed method. We implement our method on ns2 [14] and compare its performance with AODV.

Each node uses IEEE 802.11 as a wireless interface. The following parameters are also used. The radio transmission radius is 250 meters, the packet rate is 64 Kbps (CBR), the packet size is 512 bytes, and the session time is 1 second. We defined a connection node ratio F as the ratio of nodes transmitting traffic data to all nodes in the network. Pairs of communicating nodes were chosen randomly based on the connection node ratio. In this paper, we fixed the value of F at 8%. In both our proposed method and AODV, the nodes do not retransmit packets if a communication error occurs. In the simulation environment, each node was randomly located in a rectangular field of 1,000 m by 250 m. The overall simulation time was set to 100 seconds.

To investigate the relation between the node density and the path discovery performance, we show the preliminary results in AODV in **Fig. 1**. The figure shows the path discovery success ratio and the average number of adjacent nodes with the number of nodes in the ad-
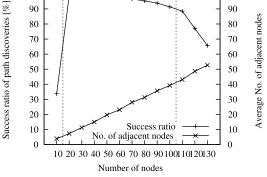

Fig. 1 Path discovery performance (AODV).

hoc network. Here, M denotes the number of nodes. When M is 20, 30, and 40, AODV has nearly a 100% path discovery success ratio. As M increases from 40 to 100, the success ratio decreases gradually to about 90%. At 10 nodes and over 100 nodes, performance falls significantly. However, there are different reasons for the lower performance when $M = 10$ and $M > 100$. The reason for lower performance when M equals 10 is that there are a few isolated nodes that cannot connect to any of the other nodes. In contrast, when $M > 100$, the increase in the number of adjacent nodes results in an excessive re-broadcasting overhead, which leads to a decrease in the path discovery success ratio. The issue of isolated nodes is beyond the scope of this paper. We consider only dense and non-dense node situations, which are also shown in Fig. 1. From this figure, let 120, 60 represent dense and non-dense node situations, respectively.

Here, we focus on the path discovery success ratio and the number of RREQs per path discovery as the definition of path discovery performance. We define the data delivery ratio as the ratio of data packets successfully received by the destination node. Each simulation runs for 100 simulation seconds. Simulation results are averaged over 5 node position patterns, and 5 traffic patterns are set for each parameter.

4.2   Results in Stationary Situations

In this section, to evaluate the fundamental characteristics of our method, we first consider static nodes where all nodes are fixed. We fix the parameter $\omega = 2^{-4}$ and $T = 0.1$.

● Node Density

The node density metric is defined in Eq. (1). The relation between the probability of re-broadcasting and this metric is shown in Eq. (2) (Policy 1) and Eq. (3) (Policy 2). To focus just on the effects of the node density metric, we use Policy 1, and we change the parameters β and θ. In Policy 1, our proposed method re-broadcasts RREQs with a probability $1 - \beta$ when the node density metric is larger than θ. Otherwise, the probability is 1, that is, the nodes always re-broadcast RREQs. **Figure 2** shows path discovery performance at high node density (120 nodes), and **Fig. 3** shows performance at low node density (60 nodes).

At high node density, AODV has about a 77% path discovery success ratio and broadcasts about 120 RREQs per path discovery. This re-broadcasting causes the success ratio
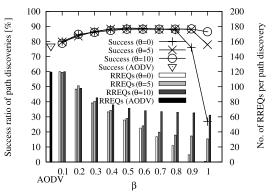

Fig. 2   Path discovery performance under Policy 1 at high node density.
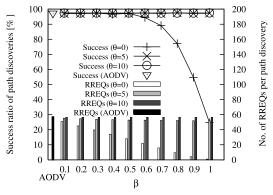

Fig. 3   Path discovery performance under Policy 1 at low node density.

to decline. Our proposed method decreases the number of RREQs compared to AODV when $\theta = 0$ and $\beta < 0.8$. This decrease results in a higher success ratio. However, when $\theta = 0$ and $\beta \geq 0.8$, the success ratio is equal to or lower than AODV. When $\theta = 0$, the probability of re-broadcasting is $1 - \beta$. The node, therefore, re-broadcasts RREQs with a constant probability $1 - \beta$ regardless of node density. So, if the β is too large ($\beta \geq 0.8$ in this experiment), then our method drops not only the redundant RREQs but also the RREQs needed for path discovery.

To prevent losing the RREQs needed for path discovery, we apply Policy 1 with the threshold parameter θ. Under Policy 1, the nodes lower the number of RREQs stochastically only if this metric exceeds the threshold parameter θ. We believe this policy will succeed in both forwarding the RREQs needed for path discovery and discarding redundant RREQs. In Fig. 2, when $\theta = 10$, our method improves path discovery performance even though the β is large. In particular, when $\beta = 0.9$, our method has an 89%

success ratio (12% more than AODV) and decreases the RREQs by 50% ($\theta = 10$) compared to AODV. As β increases (except $\beta = 1$), the number of RREQs respectively reach 35 ($\theta = 5$) and 60 ($\theta = 10$). The reason for this is that the threshold parameter θ establishes a lower limit on dropping RREQs. However, the success ratio falls when $\beta = 1$. When the metric exceeds θ, our method with $\beta = 1$ discards all RREQs. This result indicates that control of re-broadcasting without a stochastic factor tends to cause a drastic drop in RREQs.

At low node density, when AODV has about 60 RREQs per path discovery, it succeeds in almost all path discoveries. That is, there is less re-broadcasting overhead than at high node density. When $\theta = 0$, the success ratio of our method is lower than AODV's. As mentioned for high node density, our method with $\theta = 0$ tends to discard the RREQs needed for path discovery. In contrast, when $\theta = 5$ or 10, our method lowers the number of RREQs without lowering the success ratio. However, the decrease in RREQs is smaller than at high node density.

Here, we demonstrated that when our proposed method has appropriate parameters, its path discovery performance is better at both high and low node density. In this experiment, the parameters $\beta = 0.9$, and $\theta = 10$ were optimized to achieve the best improvement in performance. Also note that $\beta = 1$ should not be selected because the method proposed here drops too many RREQs. Since the proposed method is more effective at high node density, we will now further examine the method in these conditions.

- Path Concentration

We next examine data delivery performance. **Figure 4** illustrates the path discovery success and data delivery ratios at high node density. In this experiment, we use Policy 1 with $\theta = 10$. While AODV had a 77% path discovery success ratio, it delivered 47% of data traffic to destination nodes. Our method, on the other hand, improves both the path discovery and the data delivery success ratios when there is a β increase. This demonstrates that our method under Policy 1 improves data delivery performance by lowering the re-broadcasting overhead. Note, however, that there is a difference between the success ratio and the data delivery ratio. For example, when $\beta = 0.9$, this difference is about 7%, that is, a further 7% of data traffic is lost
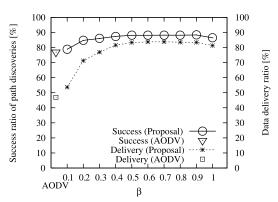

Fig. 4 Path discovery success and data delivery ratios under Policy 1.
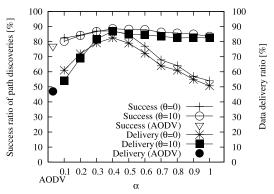

Fig. 5 Path discovery success and data delivery ratios under Policy 2.

for a reason other than path discovery failure. One possible cause of this loss may be path concentration. If the path discovery process causes some nodes to hold too many paths, then the data transmission along such paths would fail due to traffic congestion.

We examine the effects of the path concentration metric using Policy 2. **Figure 5** shows the path discovery success and data delivery ratios at high node density. In Policy 2, the re-broadcasting probability is $1 - \alpha L_i$, where L_i denotes the path concentration metric. Moreover, if the node density metric is less than parameter θ, then the probability is 1. In this paper, the path capacity parameter in Node i is set to $R_i = 10$.

When $\theta = 0$, the data delivery ratio is greater than that for AODV. However, when α exceeds 0.5, both the path discovery success and data delivery ratios decline. When $\alpha = 1$, the success ratio is 53% and the data delivery ratio is 50%. In this experiment, the average number of paths was about 30 or more. The probability of re-broadcasting is very small when α is

large. Our proposed method, therefore, cannot forward the RREQs needed for path discoveries when there is a large α.

When $\theta = 10$, the performance with a large α is better than when $\theta = 0$. This is because our method improves performance based on the node density as well as path concentration. However, when $\alpha \geq 0.7$, the success ratio and the data delivery ratio decrease slightly. At a large α, the probability of re-broadcasting is 0 in many nodes. As previously shown, controlling re-broadcasting without a stochastic factor causes a drastic drop in the RREQs. So in this case, our method causes a drop in performance at an α greater than or equal to 0.7. When $\theta = 10$ and $\alpha = 0.4$, the data delivery ratio is about 86%. This ratio is better than that under Policy 1 where $\beta = 0.9$ and $\theta = 10$. This shows that our method prevents some nodes from holding too many paths, which leads to an improvement in data delivery performance.

To summarize this section, evaluation results for static nodes show that our method improves path discovery performance more than AODV does. We used two policies to determine re-broadcasting probability. Under Policy 1, the threshold parameter θ enables performance to be controlled based on the node density. The simulation results show that Policy 1 with $\beta = 0.9$ and $\theta = 10$ improves path discovery performance the most. Under Policy 2, our method improves data delivery performance as well as path discovery performance by taking path concentration into account. These simulation results show that Policy 2 with $\alpha = 0.4$ and $\theta = 10$ achieves the best performance.

4.3   Results in Mobile Situations

In Section 4.2, we demonstrated that our proposed method improves path discovery performance for static nodes. If the nodes move around, then node density and paths change dynamically. Here, to examine the effects caused by node mobility, we evaluate performance results for mobile nodes. The Random Waypoint Model [3] is used to simulate node mobility. In this paper, the maximum node speed is $10\,\mathrm{m/s}$ ($36\,\mathrm{km/h}$), and the pause time is 10 seconds.

In our method, each node counts the number of received broadcast packets to deduce node density. The node density metric is then calculated from the exponential weighted moving average (EWMA) of that number. The weighting parameter ω will therefore cause changes in the metric.
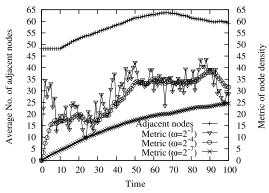

Fig. 6   Average number of adjacent nodes and the node density metric.

Figure 6 illustrates the average number of adjacent nodes and the node density metric in the dense node case. In this figure, we show a typical simulation result. We used Policy 1 and fix the parameters $\beta = \theta = 0$ to focus on the effects of ω. We used ω to 2^{-1}, 2^{-4} and 2^{-7}.

The average number of adjacent nodes does not change in 10 seconds because of the pause time in the Random Waypoint Model. After this pause time, nodes begin to move, and the average number of adjacent nodes then changes. The metric also increases based on these changes. At $\omega = 2^{-1}$, the metric changes quickly, however, it fluctuates even when there is hardly any change in the number of adjacent nodes. In contrast, when $\omega = 2^{-7}$, the metric varies very slowly. The metric continues increasing during the simulation time and does not stabilize at a particular number. In this experiment, when $\omega = 2^{-4}$, the metric correctly follows up on the average number of adjacent nodes. These results show that the parameter ω offers a trade-off effect between response speed and stability.

Next, we show the relation between the parameter ω and performance. **Figure 7** shows the success ratio and the number of RREQs under Policy 1. **Figure 8** shows the success and data delivery ratios under Policy 2. In these figures, we fix the θ value at 10.

In these results with mobile nodes, AODV had a 70% success ratio and a 40% data delivery ratio. Performance with mobile nodes is not as good as that with static nodes, meaning that node mobility is related to lower performance as well as re-broadcasting overhead. With mobile nodes, Fig. 7 and Fig. 8 show that our proposed method improves performance. However, the performance improvement depends on the pa-
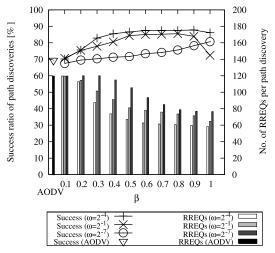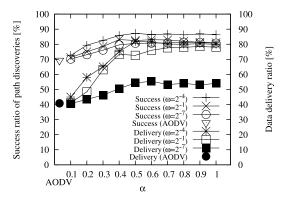
Fig. 7　Path discovery performance under Policy 1.

Fig. 8　Path discovery success and data delivery ratios under Policy 2.

rameter ω. When $\omega = 2^{-7}$, Policy 1 and Policy 2 yield the worst performance for all ω. When $\omega = 2^{-7}$, the node density metric does not stabilize on a value from which the node density can be deduced. Therefore, when $\omega = 2^{-7}$, Policy 1 cannot lower the number of RREQs based on node density. Similarly, Policy 2 does not prevent path concentration when $\omega = 2^{-7}$. In contrast, when $\omega = 2^{-1}$, our method performs better than when $\omega = 2^{-7}$. However, for this ω value, the node density metric frequently fluctuates. Our method, therefore, sometimes lowers the number of RREQs either too much or too little. This means there is no significant difference in performance between $\omega = 2^{-1}$ and $\omega = 2^{-4}$.

In this experiment, our proposed method using Policy 1 performance best when $\omega = 2^{-4}$, $\beta = 0.9$, and $\theta = 10$. In these parameter sets, the success ratio reaches 87% (17% more than

AODV) and the number of RREQs is about 50% compared to AODV. In Policy 2, the data delivery ratio reaches approximately 82% when $\omega = 2^{-4}$, $\alpha = 0.5$, and $\theta = 10$. Our method improves the data delivery ratio about 40% compared to AODV.

Section 4.2 and Section 4.3 focus on dense ad-hoc networks and discuss quantitative performance evaluation. In Section 4.3, we evaluate performance under dynamic node density changing condition and confirm that our method using the best parameter sets improves the performance of path discoveries in these conditions. In non-dense ad-hoc networks by contrast, our method and AODV achieve the same level of performance, as shown in Fig. 3. As qualitative performance evaluation, our proposed method, using the best parameter set, is applicable to ad-hoc networks where the maximum number of nodes is about 120, and maximum moving speed is about 10 m/s. To further apply our method to more varied environments, the following approach is considered. Ad-hoc nodes prepare in advance candidate parameter sets that are suitable for multiple sets of network environments. Then, each node recognizes the network environments and chooses the best parameter set.

5. Conclusion

In this paper, we proposed an efficient ad-hoc route establishment method. This method controls re-broadcasting by taking the node density and the path concentration into account. Through computer simulation, we examined the relation between the path discovery performance and the parameters of our proposed method. We then showed that our method improves path discovery performance and data transmission by choosing appropriate parameter sets. With static nodes, our method achieves a path discovery success ratio of 12% or more and reduces the RREQs 50% more than AODV. Moreover, taking the path concentration into account, our method achieves about a 50% or better data delivery ratio than AODV. With mobile nodes, our method improves performance by following up on changes in the network topology. While the method proposed in this paper is based on AODV, it can also be used with other on-demand protocols.

References

1) Corson, S. and Macker, J.: Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, RFC 2501, IETF (1999).

2) Perkins, C., Royer, E. and Das, S.: Ad hoc On-demand Distance Vector (AODV) Routing, RFC 3561, IETF (2003).

3) Johnson, D. and Maltz, D.: Dynamic Source Routing in Ad Hoc Wireless Networks, *Mobile Computing*, Imielinski and Korth (Eds.), pp.153–181, Kluwer, MA (1996).

4) Williams, B. and Camp, T.: Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks, *Mobihoc 2002 Proceedings*, pp.194–205, ACM (2002).

5) Haas, Z. and Pearlman, M.: The Performance of Query Control Schemes for the Zone Routing Protocol, *Trans. IEEE/ACM Networking*, Vol.9, No.4, pp.427–438 (2001).

6) Chiang, C., Wu, H., Liu, W. and Gerla, M.: Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel, *SICON 1997 Proceedings*, pp.197–211, IEEE (1997).

7) Yi, Y. and Gerla, M.: Scalable AODV with efficient flooding based on on-demand clustering, *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol.6, No.3, pp.98–99 (2002).

8) Ko, Y. and Vaidya, N.: Location-Aided Routing (LAR) in mobile ad hoc networks, *Mobicom 1998 Proceedings*, pp.66–75, ACM/IEEE (1998).

9) Ni, S., Tseng, Y., Chen, Y. and Sheu, J.: The Broadcast Storm Problem in a Mobile Ad Hoc Network, *Mobicom 1999 Proceedings*, pp.151–162, ACM (1999).

10) Cartigny, J. and Simplot, D.: Border Node Retransmission Based Probabilistic Broadcast protocols in Ad-Hoc Networks, *Telecommunication Systems*, Vol.22, No.1–4, pp.189–204 (2003).

11) Toh, C.-K.: A Novel Distributed Routing Protocol to Support Ad-Hoc Mobile Computing, *IPCCC 1996 Proceedings*, pp.480–486, IEEE (1996).

12) Xu, Y., Heidemann, J. and Estrin, D.: Adaptive Energy-Conserving Routing for Multihop Ad Hoc Networks, Research Report 527, USC/Information Sciences Institute (2000).

13) Royer, E., Melliar-Smith, P. and Moser, L.: An Analysis of the Optimum Node Density for Ad hoc Mobile Networks, *ICC 2001 Proceedings*, pp.857–861, IEEE (2001).

14) Network Simulator 2. available as http://www.isi.idu/nsnam/ns/

Yujin Noishiki received his B.E. and M.Informatics degrees from Kyoto University, Japan, in 2000 and 2002, respectively. He began working for KDDI R&D Laboratories, Inc., Japan, in 2002. His current research interests include mobile computing and mobile ad-hoc networking. He is a member of IEICE.

Hidetoshi Yokota received his B.E., M.E. and Ph.D. degrees from Waseda University, Tokyo, in 1990, 1992, and 2003, respectively. He joined KDDI R&D Laboratories, Japan, in 1992. From 1995 to 1996 he was with SRI International, in Menlo Park, CA as an International Fellow. He received the IEICE Young Engineer Award in 1998 and the IPSJ Yamashita SIG Research Award in 2005. His current research interests include mobile communications and ad-hoc networks. He is a member of IPSJ, IEICE and IEEE.

Akira Idoue received his B.E and M.E. degrees from Kobe University in 1984 and 1986, respectively. Since joining KDD (now KDDI) in 1986, he has worked in the field of computer communication. His current research interests include high performance communication protocols, protocol testing, and mobile computing. He is currently a senior manager of the Mobile Network Lab. at KDDI R&D Laboratories, Inc. He received the IPSJ Convention Award in 1992 and the Best Paper Award of the IPSJ National Convention in 1998.