PAPER *Special Section on Multi-dimensional Mobile Information Networks*

Performance Improvement of an Ad Hoc Network System for Wireless Data Service

Takayuki YAMAMOTO[†], ***Student Member***, **Masashi SUGANO**[††], **Masayuki MURATA**[†], **Takaaki HATAUCHI**[†††], ***Members***, ***and*** **Yohei HOSOOKA**[†††], ***Nonmember***

SUMMARY In an ad hoc wireless network system, wireless terminals can autonomously construct and maintain a network. They communicate with some neighbor terminals, exchange information about the network and decide routes for packets on the multi-hop wireless network. Flexible Radio Network (FRN), one of the ad hoc wireless network systems, adopts an original protocol that provide a multiple routes management and a packet retransmission mechanism against packet transmission errors. This system has been in use in a recent few years. In this paper, we first evaluate the performance through simulations for data-link protocol and routing protocol of FRN to clarify its basic properties, and then we discover some problems that degrade the performance. Furthermore, we propose some performance improvement techniques. They are methods to set an adaptive system parameter or to improve its protocol statically. We simulate those techniques and show how they improve the system performance.

key words: *ad hoc wireless network, routing protocol, simulation*

1. Introduction

An ad hoc wireless network is a self-organized network built with wireless terminals that communicate with each other and exchange the network information. Those terminals have a capability to relay packets for another terminal, so they can construct a wide area multi-hop wireless network. The ad hoc network needs neither a wired backbone network nor base stations, and therefore network installation, expansion and removal can be performed easily and quickly. Such a wireless infrastructure covers a wide range of applications, e.g., distributed computing, disaster recovery, and military operation. Accordingly, many studies have been dedicated to analyze its characteristics and/or propose new routing methods (see, e.g., [1]–[10]).

Flexible Radio Network (FRN) is one of commercially available products based on ad hoc wireless network system [11]. A large-scale network with stationary terminals can be installed easily into existing facilities by FRN. In addition, the network can be extended only by adding the radio terminal if needed. This system is now utilized for collecting usage information of ski lifts, a sales account and monitoring information of vending machines, and for electric energy control in factories. FRN adopts a proprietary protocol that can efficiently adapt to terminal failures or a change of network configuration. We have introduced this system and investigated the performance of network in [12]. However, it is not clear enough how system parameters affect the performance such as throughput and packet loss rate. In the current system, these are decided by trial and error. In order to clarify the scope of this system, it needs to be evaluated in detail.

Next, we suggest three techniques for improving the performance of FRN. These techniques are based on some problems found through a process of simulated performance evaluation and/or an experience in the real environment.

One of them is a change of system parameter setting. All data packets in FRN have a parameter that is called *maximum lifetime*. This is used to erase some long-living packets. In an original system, the same value that is large enough for the network is selected for all data packets. However, maximum lifetime of packet has close connections in the necessary hop count to the destination node. The system will be able to achieve better performance by setting an adaptive lifetime for each packets.

The rest two of our suggestions are techniques to decrease packet collision. We have studied about FRN system and attended to a problem that packets were duplicated unnecessarily and the duplicated packets degraded the network performance. In FRN, wireless terminals check packet transmission errors at every hop. When a transmission error is detected, they retransmit the packet after selecting the next available route. This error can be detected when the terminal does not receive a corresponding ACK from the neighbor terminal within a pre-specified time. The problem is that the packet sender terminal recognizes the transmission error in the case of only the ACK packet is lost after the data packet is successfully transmitted. In such a case, the terminal will re-transmit the packet although the first packet is not lost. We call this re-transmitted packet duplicated packet. The unnecessarily duplicated packet is put into the network, because in FRN, no ter-

Manuscript received January 1, 1999.

[†]The authors are with the Department of Informatics and Mathematical science, Graduate School of Engineering Science, Osaka University, Toyonaka-shi, 560-8531, Japan.

[††]The author is with Osaka Prefecture College of Health Sciences, Habikino-shi, 583-8555, Japan.

[†††]The authors are with Energy and Electrical Systems Company, Fuji Electric Co. Ltd., Hino-shi, 191-8502, Japan.

Table 1 Configuration Table

	Dest. Node 0	Dest. Node 1	...
Route 1	Neighbor Info.	Neighbor Info.	...
Route 2	Neighbor Info.	Neighbor Info.	...
⋮	⋮	⋮	⋱

minals manage the history of packet transmission to reduce the complexity of wireless terminals. The packet duplication leads to a higher traffic load than an actual one. To make the matter worse, as the traffic load becomes higher, more duplicated packets are generated because also ACK transmission error probability becomes larger. Thus, the network performance degrades rapidly in such a condition. We propose two techniques to prevent packet collisions and ACK losses.

In this paper, we first describe about the Flexible Radio Network system and evaluate the relation between its performance and the system parameter setting. Next, we examine a packet duplication process in detail. We show that the network performance degrades rapidly as the number of duplicated packets increases. We suggest performance improvement techniques with consideration to packet duplication. Through simulation, we show that those techniques can decrease the number of duplicated packets and improve the network performance.

2. System Description of FRN

2.1 Network Configuration

In FRN, every wireless terminal is called a *node*. Nodes with which the node can communicate directly are called *neighbor nodes*. Every node has the ability to select a route of packet and relay it to one of the neighbor nodes. In more detail, a *host node* generates and receives data packets, and other nodes are called *relay nodes* that construct a multi-hop network. Every node maintains the network information in a *configuration table* that contains the routing information from the node itself to each destination node. The routing information consists of the list of neighbor nodes on the route to the destination node i, and hop count of the route. Every node exchanges a configuration control packet periodically, and updates its configuration table by the packets from neighbor nodes.

2.2 Data-link Protocol

A radio channel is divided into fixed-length time slots. In a wireless network, every neighbor node of a certain node can receive packets from the node even when it is not the source/destination of the packet. In FRN, this property is utilized for the hop-level acknowledgment to enhance its reliability. See Fig. 1 as an example. Fig. 1(a) shows a case where packet transmission and acknowledgment at node A is successful. Node B receives the packet from node A and relays it to another node successfully. At the same time, this relayed packet is received by node A because it is a neighbor node of node B. This acknowledgment is called *relay echo* (or simply *echo*). If the relay echo is successfully received by node A, it can recognize that the transmission to node B succeeded. The case where node A fails the first transmission is shown in Fig. 1(b). In this case, node A detects a failure because no echo from node B is received. Then, node A retransmits the packet after a pre-specified time. When a packet reaches its destination node, the destination node no longer transmit it and the previous node cannot get an echo. To delete the buffered packet in the previous node, the destination node create an ACK packet exceptionally (see Fig. 1(c)).

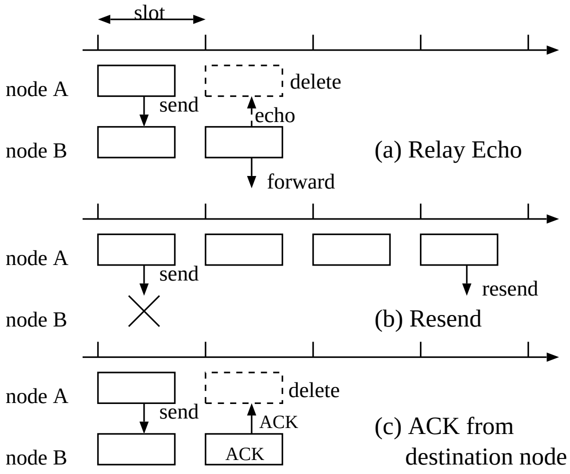

Fig. 1 Packet Transmission Timing

A maximum lifetime is predefined for every data packet. For every time slot, it is decreased by one. When the value reaches zero, the packet is discarded due to lifetime expiration. It is an important configuration parameter since short lifetime gives a chance to effectively remove long-living packets from the network or long lifetime gives a chance to try another route to the destination. It is necessary for a reliable network to set a large enough value for the longest route while too large value causes network congestion. We need to decide a proper value of this parameter for the network.

2.3 Routing Protocol

In FRN, each node collects network information from its neighbor nodes and decides the direction of relaying packets. Since the radio environment changes frequently, a routing protocol must select an appropriate route adaptively. Furthermore, if a node fails to transmit a packet on the first trial, another route should be selected immediately (or should try the same route

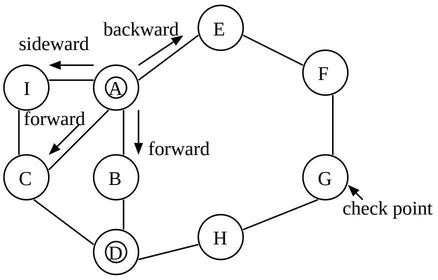

Fig. 2 Multiple Route Selection

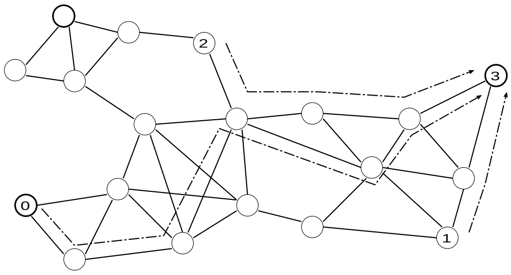

Fig. 3 Simulated Network Model

again). In FRN, every node maintains multiple route information for each destination node in the configuration table as Table 1. For each destination, routes are divided into three groups by their hop count:

- Forward route: The route(s) on which the hop counts to the destination is the shortest.
- Sideward route: The route(s) on which the hop counts to the destination is the shortest hop count plus one.
- Backward route: The route(s) on which the hop counts to the destination is the shortest plus two or more.

Fig. 2 is an example of these route classifications. When node A is source node and node D is destination node, the shortest hop counts are two, and routes through node B and C are forward routes. The route through node I requires three hops, so this route is sideward route. There is a detour, a backward route, to node D that pass through node E. On the backward route, a check point is defined at the source node to avoid back tracking. It is detected for each backward route with its configuration table. We do not describe the check point detection method in this paper.

In this routing protocol, shorter route has higher priority. If the shortest route is unavailable due to some reason such as obstacles on the route, radio noise or packet collision, the node looks up the configuration table again and selects the second shortest route. Each node set a next hop node ID, a temporal destination, to a packet header after it decides the route. When neighbor nodes receive the packet, they check the next hop field and recognize they are a temporal destination node or not. When they are, they decide new next hop of the packet and transmit it. When they are not, they check whether it is a relayed echo (see Section 2.2).

3. Basic Properties of FRN

3.1 Simulation Environment

In this section, we first investigate basic properties of FRN. We attend to the maximum lifetime by which the network performance is greatly affected. When the maximum lifetime is too long for the network, some packets stay long in the network and cause the degradation of performance. When it is too short contrarily, it may expire before relay nodes on the route try another route for the transmission and the system cannot fulfill its potential. We make some simulations where maximum lifetimes are different, and evaluate a relationship between the setting of lifetime and system performance. For the simulations, ns-2 [13] is used to utilize its radio propagation model extended by [14].

We use a network model shown in Fig. 3. A circle represents a node. A line connecting two nodes means that they can communicate directly. In this model, packet losses are assumed to occur only by collision of radio wave. The numbered nodes (node 0, 1, 2, 3) are host nodes that can transmit and receive data packets. Arrow from each source node to the destination is an example of route that packets are actually passed through. In all simulations, node 0, 1 and 2 go on sending constant bit rate UDP packets to node 3. Arrows among host nodes are examples of the shortest route. This network model is based on the application, collecting information from decentralized host nodes. The packet generation rate of each host node is assumed identical. The traffic load is defined as the number of packets generated per one slot in the whole network (i.e., the sum of the packet generation rate at three sender nodes). We simulate in the range of load from 0.01 to 0.1 which is practical in actual environment.

We use throughput and packet loss rate (PLR) to measure the performance. Throughput is an average number of successfully transmitted packets per one time slot. PLR is a ratio of the packet not reaching the destination.

3.2 Network Performance with Various Maximum Lifetime Value

The network performance is largely affected by the maximum lifetime as described in previous section. When it is large, the packets can be relayed to the destination node even after some retransmission trials. As the traffic load becomes higher, on the contrary, setting a smaller value is an efficient way to drop long-living packets and prevent a congestion. There are three results of simulations where we set the maximum lifetime

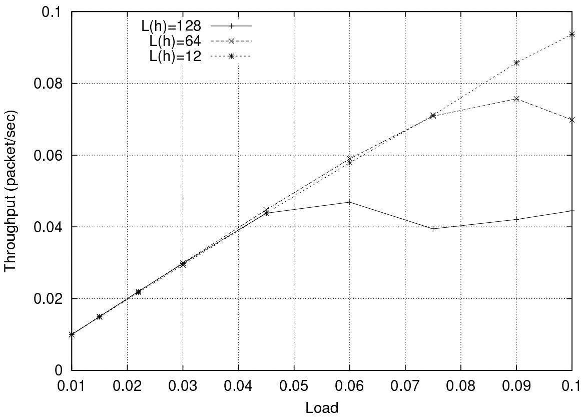

Fig. 4 Original System Performance: Throughput

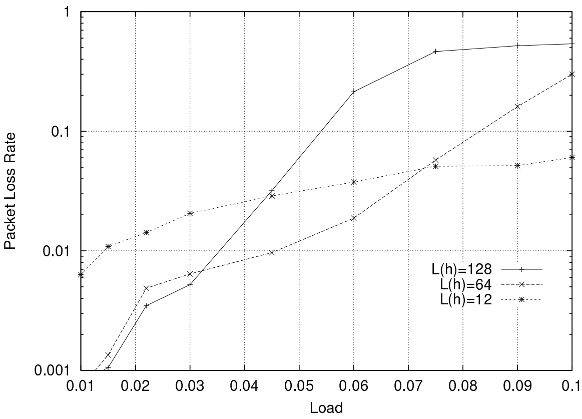

Fig. 5 Original System Performance: PLR

value to 12, 64 and 128.

Fig. 4 shows throughput transition of each simulation. The label "L(h)" means the value of maximum lifetime. The load in simulations is defined as the total of packet generating rate per one slot of all host nodes. We can see that the throughput peaks at the smaller load as the maximum lifetime becomes longer. This is because the number of long-living packets increases and then congestion occurs in the network. We can achieve high throughput in the high load environment with a small lifetime value. It is difficult to clarify a performance difference when the load is low. It can be seen in the next Fig. 5, the graph of PLR. When the load is low, the system with long lifetime shows better performance than one with short lifetime. Every node in FRN maintains multiple routes in its configuration table (see Section 2). In the long lifetime system, nodes can try these variable routes to relay packets to their destination if once a transmission fails. However, as the load of the system increases, the network performance degrades rapidly because of the network congestion caused by many long-living packets.

According to these results, it becomes clear to be important to select an adaptive value for the maximum lifetime. The best value is difficult to be determined because it depends on a scale, structure and current load of the network. We propose a method for calculating the adaptive lifetime with the route length.

4. Performance Improvement Techniques

4.1 Adaptive Maximum Lifetime

A maximum lifetime value of packet has much effect on the network performance as described in Section 3. All packets used to have the same maximum lifetime that is large enough for the network scale in an original FRN system. However, a necessary lifetime for each packet is different because a hop count against each route is various. Generally, the shorter route needs the shorter time to pass through. There are several lengths of routes in a network. Then it is not a good idea to set the same maximum lifetime to all packets. In this subsection, we show a method for setting an adaptive maximum lifetime to each packet.

In FRN, every node maintains the configuration table that includes the shortest hop path to the destination. This hop count has a close relationship with a necessary hop count because a source node first try the shortest route. When a packet is created at the source node, the node can dynamically calculate an adaptive maximum lifetime with the shortest hop count in its configuration table.

According to the simulation results in Section 3.2, the best maximum lifetime may be between 12 and 64 in this network. We use, for example, $L(h) = 6h + 12$ (the maximum lifetime L is a function of the shortest hop length h) for an adaptive maximum lifetime calculation in this paper. This function is derived experimentally through simulations. In a real environment, many factors will affect the necessary lifetime, therefore the function may be derived for each environment.

We show the performance example of adaptive maximum lifetime in Fig. 6. It is difficult to compare the results by their throughput because it shows little difference when the load is low, so we show only the graph of packet loss rate (PLR). The results of static lifetime values of 12 and 64 are shown for purpose of comparison. The simulation environment is the same as in Section 3.1. The function $L(h) = 6h + 12$ is applied to all packets at their source node to calculate their lifetime. Looking at Fig. 6, when the load is low, the modified system shows better performance of static lifetime 12. This is because the packets on longer routes can get longer maximum lifetime in the modified system. When the load is high, it does not show a rapid degradation that is shown in long lifetime system. This result means, in the modified system, a mechanism to drop long-living packets by maximum lifetime works well while it does not work in the static long maximum

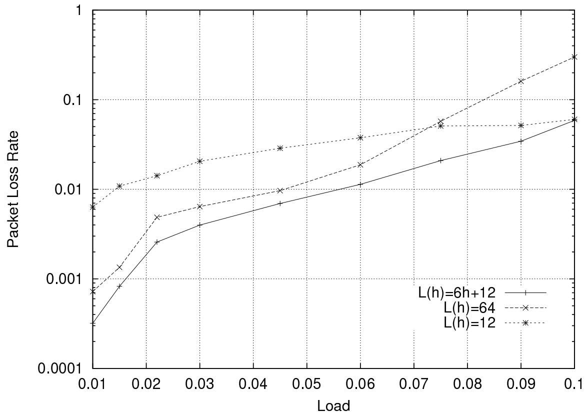

Fig. 6 PLR for the Adaptive Maximum Lifetime System

lifetime system. Therefore, setting an adaptive lifetime for each packet has a good ability to control the trade off between network congestion and reliability that are caused by long lifetime.

4.2 Decrease Packet Collisions

4.2.1 Packet Duplication Problem

As mentioned in Section 2, FRN has a packet retransmission mechanism against transmission errors. While this mechanism is likely to contribute to packet reachability, it sometimes causes unnecessary packet duplications. A duplicated packet is a buffered packet in the intermediate node which will be sent out by the retransmission mechanism although an original packet is being transmitted orderly. Duplicated packets increase a possibility of collisions, and occupy a node buffer. They increase the network load more then an actually given one, and then, the network performance degrades greatly. A packet duplication process is illustrated in Fig. 7. Node A transmits a packet to node B successfully at slot 0. Node B relays the packet to node C at slot 1. At the same time, it is expected that this relayed packet be received by node A as relayed echo. However, it sometimes happens that the echo is lost because of an obstacle (e.g., a person in the case of the skiing ground) or a collision with another packet. Then, node A cannot receive the echo successfully. In such a case, the copied packet in the buffer of node A should not be removed at the end of slot 1. In other words, the same two packets exist in node A and node C. Node A retransmits the packet later. It may be retransmitted through different routes to the destination because of multiple routes information in the configuration table and affects a wide area.

Wireless terminals in an actual system have a feature called *relay stop by eavesdropping*. When a node receives a packet and it is the same as one of the packets in the buffer, the node stops to relay the packet in the buffer. Fig. 8 is an example of this feature. Node A transmits a packet named P1 to node B. At the same time, it can be received by node C, the neighbor node of node A. In other words, node C eavesdrops the radio wave to node B. Node C looks up in its buffer and finds the same packet. It considers that there is a duplicated packet in the same network, and then, erases the packet in its buffer. This feature has a capability to erase some duplicated packets, however, it cannot erase all of duplicated packets and cannot prevent the duplication itself.

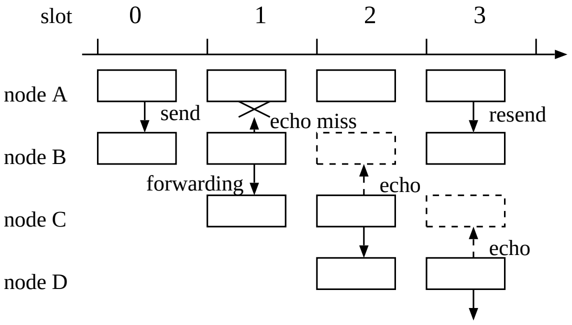

Fig. 7 Packet Duplicating Process

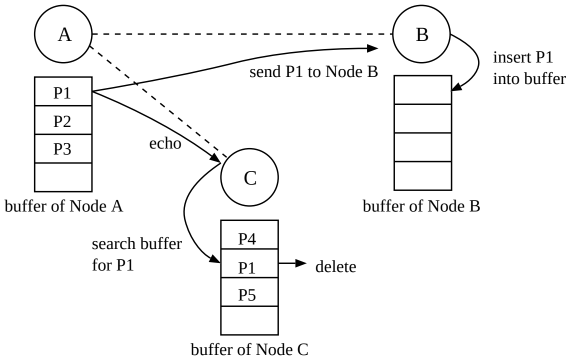

Fig. 8 Relay stop by eavesdropping

4.2.2 Collision Preventing Methods

Network performance degrades rapidly as the number of packets in the network increases. This is clear when we see the results in Fig. 5 again that shows a rapid performance degradation of long maximum lifetime system when a network load becomes large. This is caused by a wireless characteristic that the radio wave has the influence on all nodes within the reachable range. Moreover, the duplicated packet is unnecessary if its original packet reaches the destination properly. Therefore, decreasing a probability of packet duplication leads to much improvement on the network performance. In this subsection, we suggest two techniques that can appropriately control the number of duplicated packets. As described in previous subsection, packet duplication is caused by a loss of relay echo, which can be

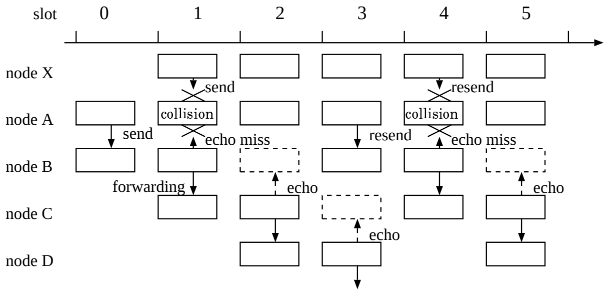

Fig. 9 Repeated Echo Loss

reduced by the following two methods: (1) to introduce the random delay before packet retransmissions (RANDOM_DELAY), and (2) to drop the packets previously which cannot reach the destination host within rest of their lifetime (EARLY_DROP).

The change RANDOM_DELAY above is an improvement of the packet retransmission method described in Section 2.2. As shown in Fig. 1(b), when a node cannot receive an echo within a specified time, it selects another route for retransmission. This waiting duration was a constant length in the original system, and therefore once the packet experiences a collision, it tend to repeat collisions until transmission through another route succeeds. By introducing the random waiting time before retransmission, the possibility of collisions would be reduced. We explain why it can decrease the number of duplicated packets. See Fig. 9. Suppose that the waiting time before retransmission is fixed, as an original system, at three slots. Node A expects to receive an echo from node B at slot 1. Node X, however, sends a packet at the same slot and it collides with the echo. Therefore, node A cannot receive the echo and retransmits the packet at slot 3. At slot 4, the echo from node B collides again with the packet from node X because of the fixed waiting time. Collisions repeat until the transmission through another route succeeds. The change RANDOM_DELAY can inhibit the repeated losses of relay echoes, and it can be expected to reduce the number of packet duplications. In the simulations, nodes select a random time from 3 to 5, which was fixed at 3 previously. To make it 2 is bad because it is too short for an echo based system.

The change EARLY_DROP is a technique for decreasing unnecessary packets within the network. In FRN, the packet lifetime is reduced by one for every time slot. When no collisions occur, a packet can be passed through one hop by one time slot. That is, the lifetime should be ideally the maximum allowable hop count of the packet. The hop count to the destination is maintained at each node. Thus, if the residual lifetime of the packet becomes less than the hop count of the shortest path to the destination, that packet of no use should be discarded. This technique prevents unnecessary packet transmissions and decrease the number of packet collisions that lead to packet duplications.

4.3 Simulation Results of Proposed Techniques

We investigate the effect of these three improvement techniques through simulations. The simulation environment and network model is the same as that used in Section 3. It is difficult to compare the results by their throughput because it shows little difference when the load is low, and then we show only the graph of Packet Loss Rate.

The results of change RANDOM_DELAY, to determine a waiting time before packet retransmissions randomly, are shown in Fig. 10 and Fig. 11. The label “normal” is a result of original system with no modification, “random” is RANDOM_DELAY modified system. See Fig. 10, the packet loss rate. At long maximum lifetime system, such as 64, this method is efficient. This is because long lifetime packets can try many routes after a quick recovery from packet collisions. It shows the same or a little worse performance when the maximum lifetime is short. In this simulation, nodes select the waiting time before retransmission occurs between 3 and 5, whose average is longer than fixed 3 in an original system. This sometimes causes more expiration of packets when the lifetime is short, then this technique has less ability in short lifetime system. Fig. 11 shows the number of duplication occurrences per second. At long maximum lifetime and high load environment, this technique can decrease the number of duplication and improve the performance.

Fig. 12 and Fig. 13 are the results of change EARLY_DROP, to remove packets which lack the lifetime to reach their destinations. The result of modified system is labeled “drop”. On the contrary as shown the result of change RANDOM_DELAY, it shows better performance improvement when the maximum lifetime is short. This technique has an effect on such packets that will be dropped forcibly because of expiration of lifetime. In a long lifetime system, there is less probability of packet time-out, and this modification shows little improvement. This technique has a little efficient on packet duplications. However, in the short lifetime system, the number of duplication is little from the beginning. Therefore, this technique cannot show a great improvement on packet duplication.

At last, we show a result with all modifications. Three techniques we have proposed are independent of one another. It is expected that the network performance improves more by applying all suggestions at the same time than by applying each of them separately. Fig. 14 shows the PLR transitions. In this figure, we add the result of one modified system where the maximum lifetime is calculated dynamically. We can see the network achieves the best performance with all modifications. Through these simulations, our suggestions are proved to have an ability to improve the

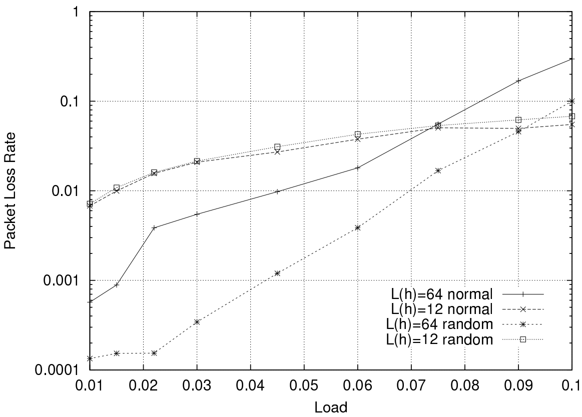

Fig. 10 PLR for the RANDOM_DELAY Modified System

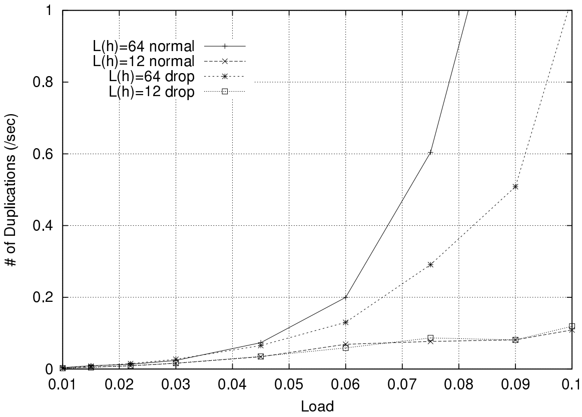

Fig. 11 The Number of Duplications (/sec) for the RANDOM_DELAY Modified System

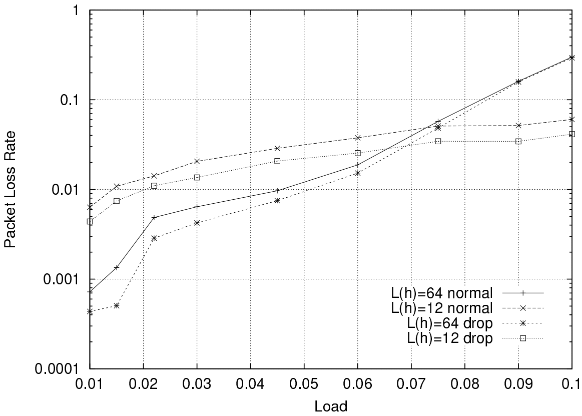

Fig. 12 PLR for the EARLY_DROP Modified System

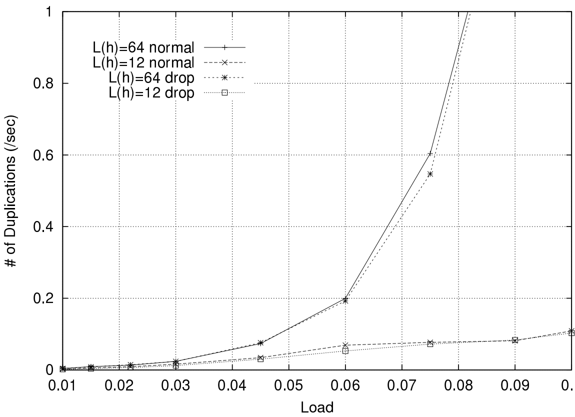

Fig. 13 The Number of Duplications (/sec) for the EARLY_DROP Modified System

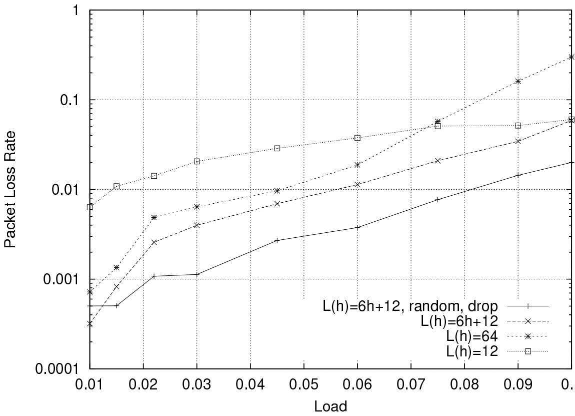

Fig. 14 PLR for the ALL Modified System

network performance of FRN system, and shows much more performance by applying them at the same time.

5. Conclusion

In this paper, we have investigated performance characteristics of Flexible Radio Network (FRN), a product of commercial use. Because of its application, collecting an information from many distributed terminals, FRN has an original protocol to construct a reliable network. However, it causes high packet loss rate because long-living packets disturb other packets and sometimes the retransmission mechanism makes duplicated packets. We have introduced a dynamic adaptive maximum lifetime setting technique to control the lifetime of packet. Next, we have described a packet duplication process in FRN and suggested two techniques with consideration to packet duplication. We have shown those three techniques can improve the network performance through simulation experiments. In addition, we have also shown that they can be applied to the system at the same time to achieve the best performance.

As a future topic, we should develop a method to determine the best expression for adaptive maximum lifetime that considers the network load or current packet loss rate. End-to-end performance by using, e.g., TCP, as an upper layer protocol should also be investigated.

References

[1] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, "A Feedback Based Scheme For Improving TCP Performance In Ad-Hoc Wireless Networks," in Proc. of the ICDCS '98, pp. 472–479, May 1998.

[2] T. Goff, J. Moronski, and D. S. Phatak, "Freeze-TCP – A true end-to-end TCP enhancement mechanism for mobile environments," in Proc. of the INFOCOM 2000, March 2000.

[3] G. Holland and N. H. Vaidya, "Analysis of TCP Performance over Mobile Ad Hoc Networks," in Proc. of the ACM/IEEE MOBICOM'99, pp. 219–230, Aug. 1999.

[4] A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T. Chen, "Scalable Routing Strategies for Ad Hoc Wireless Networks," IEEE J. Select. Areas Commun., vol. 17, no. 8, pp. 1369–1379, Aug. 1999.

[5] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks," in Proc. of the ACM/IEEE MOBICOM'99, pp. 195–206, Aug. 1999.

[6] D. Kim, C.-K. Toh, and Y. Choi, "TCP-BuS : Improving TCP Performance in Wireless Ad Hoc Networks," in Proc. of the IEEE ICC 2000, June 2000.

[7] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," in Proc. of the ACM/IEEE MobiCom 2000, pp. 255–265, Aug. 2000.

[8] N. Nikaein, H. Labiod, and C. Bonnet, "DDR – Distributed Dynamic Routing Algorithm for Mobile Ad Hoc Networks," in Proc. of the MobiHoc 2000, Aug. 2000.

[9] M. R. Pearlman and Z. J. Haas, "Determining the Optimal Configuration for the Zone Routing Protocol," IEEE J. Select. Areas Commun., vol. 17, no. 8, pp. 1395–1414, Aug. 1999.

[10] K. Takasugi, Y. Suzuki, and S. Kubota, "Multicast Routing Protocol for Avoiding Congestion in Ad Hoc Wireless Netowork," IEICE Trans. B (in Japanese), vol. J83–B, no. 7, pp. 991–998, July 2000.

[11] "Flexible Radio Network, Fuji Electric Co. Ltd." available at `http://www.fujielectric.co.jp/denki/p26/ecop_contents2.html`.

[12] M. Sugano, T. Araki, M. Murata, T. Hatauchi, and Y. Hosooka, "Performance Evaluation of a Wireless Ad Hoc Network: Flexible Radio Network (FRN)," in Proc. of the IEEE ICPWC 2000, pp. 350–354, Dec. 2000.

[13] "The Network Simulator - ns-2." available at `http://www.isi.edu/nsnam/ns/`.

[14] "The CMU Monarch Project." available at `http://www.monarch.cs.cmu.edu/`.

Takayuki Yamamoto was born at Hyogo, Japan, in 1979. Now he is a master course student of Graduate School of Engineering Science, Osaka University. His research interests are in wireless ad hoc network, and its performance evaluation and simulation. He is a student member of IEICE.

Masashi Sugano received the B.E., M.E., and D.E. degrees in Information and Computer Sciences from Osaka University, Japan, in 1986, 1988, and 1993, respectively. In 1988,he joined Mita Industrial Co. Ltd. (currently, Kyocera Mita Corporation) as a Researcher. From September 1996, he has been an Associate Professor of Osaka Prefecture College of Health Sciences. His research interests include design and performance evaluation of computer communication networks, network reliability, and wireless network systems. He is a member of IEEE, ACM and IEICE.

Masayuki Murata received the M.E. and D.E. degrees in Information and Computer Sciences from Osaka University, Japan, in 1984 and 1988, respectively. In April 1984, he joined Tokyo Research Laboratory, IBM Japan, as a Researcher. From September 1987 to January 1989, he was an Assistant Professor with Computation Center, Osaka University. In February 1989, he moved to the Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University. From 1992 to 1999, he was an Associate Professor in the Graduate School of Engineering Science, Osaka University, and from April 1999, he has been a Professor of Osaka University. He moved to Advanced Networked Environment Division, Cybermedia Center, Osaka University in April 2000. He has more than two hundred papers of international and domestic journals and conferences. His research interests include computer communication networks, performance modeling and evaluation. He is a member of IEEE, ACM, The Internet Society, IEICE and IPSJ.

Takaaki Hatauchi was born in Hiroshima, Japan, in 1959. He received the B.E. degree from Kinki University in 1982. He joined Fuji Electric. His current research interests are communication protocols for wireless system.

Yohei Hosooka was born in Tochigi, Japan, in 1976. He received the B.E. degree in Faculty of Engineering from Utsunomiya University, Utsunomiya, Japan, in 1999. In 1999, he joined Fuji Electric in Japan, and he is a research engineering of wireless application. His current research interests are in wireless network communication architecture.