IEICE TRANS. COMMUN., VOL.E1–B, NO.1 JANUARY 1918

PAPER

A Low-latency Routing in Ad Hoc Networks for Short-lived TCP Connections

Takayuki YAMAMOTO[†], ***Student Member***, **Masashi SUGANO**[††],
and **Masayuki MURATA**[†††], ***Members***

SUMMARY Wireless ad hoc network is expected to be integrated with wired networks and many applications will communicate with TCP (Transmission Control Protocol). Many studies have been dedicated to improving the TCP performance over ad hoc networks. However, most of these studies assumed a persistent TCP connection. This is clearly inadequate because many TCP connections are actually short-lived. For such connections, the routing latency in ad hoc networks is considerably long. In this paper, we propose a new routing protocol for ad hoc networks, the Low-latency Hybrid Routing protocol (LHR), that is suitable for a network where many TCP connections are short-lived. According to simulation results, LHR can establish and process more connections within a given time period than other existing routing protocols.

key words: *Ad hoc network, Routing, Short-lived TCP connection, Latency*

1. Introduction

Ad hoc wireless networks are self-organized networks built with wireless terminals. They communicate with each other and exchange network structure information. They can also relay data packets for another terminal to construct a wide area multi-hop wireless network. The ad hoc networks need neither a wired backbone network nor a base station. As a result, 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 systems, disaster recovery networks, and sensor networks. Accordingly, many studies have been dedicated to analyze its characteristics and/or propose new routing methods (see, e.g., [1]–[6]). With expectation that it is integrated with wired networks by using TCP, some other studies have considered the use of TCP over ad hoc networks (e.g., [7]–[9]). However, most of them assume that the TCP connection is persistent; i.e., it has an infinite amount of data to transmit, and then they examine the steady-state throughput values. It is apparently inadequate because many TCP connections are short-lived. For example, it is reported in [10] that the average size of Web documents at several Web servers is about 10 [Kbytes]. Furthermore, in the sensor network, the amount of data on each connection is small, and major of the TCP connections would be short-lived. Another approach to enhance the TCP performance over ad hoc networks has been proposed in [11], [12]. They have introduced a feedback-based mechanism into TCP. However, modifying TCP itself is not adequate for protocol migration since TCP is an end-to-end communication protocol including wireless and wired terminals. Instead, we should consider a new routing protocol in the ad hoc network suitable for the short-lived TCP connections.

In Fig. 1, we compare short- and long-lived TCP connections by dividing the overall connection time into two blocks; one is a connection establishment time, which contains routing and TCP three-way handshake latency, and the other is a data transmission time. Comparing them, we can see that the connection establishment time is independent of the transmitting data size. In other words, the connection establishment time in a short-lived connection occupies larger percentage of overall connection time than that in a long-lived connection. When major of connections in a network are short-lived, we need to tackle the following problems, which are not resolved in the existing routing protocols;

- large overhead of exchanging the routing table
- large latency for an initial route search process
- large latency for another route search in the case of link disconnection

If we assume the TCP connection is persistent, the above problems do not affect the performance even in high-mobility and high traffic load environment. However, in actual TCP is never persistent, and most connections are short-lived.

A sensor network is one promising application over wireless ad hoc networks. In the sensor network, small amount of data is collected from many sensor terminals. Such a network model fits the most of characteristics of ad hoc networks, such as a distributed operation, scalability, and ease of maintenance. To ascertain that the reliable transmission by TCP is important also in the network where the data traffic from sensors is intermittent, we measure the packet loss prob-

---

Manuscript received January 15, 1999.
Manuscript revised January 12, 2001.

[†] The author is with Graduate School of Information Science and Technology, Osaka University, Toyonaka-shi, 560-8531, Japan.

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

[†††] The author is with Cyber Media Center, Osaka University, Toyonaka-shi, 560-8531, Japan.

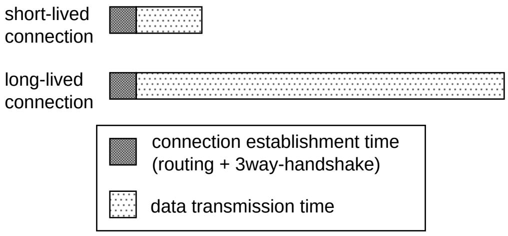


Fig. 1 Comparison of Connection Lengths of Short- and Long-lived TCP Connections

ability of UDP and TCP in the later Section 3. The TCP traffic is routed to a destination node (i.e., data collecting terminal) by a routing protocol. Many researchers have proposed various routing protocols in ad hoc networks. The Destination Sequenced Distance Vector (DSDV) [13] is a proactive protocol that each wireless node exchanges a route table periodically with neighbor nodes. The Ad hoc On-demand Distance Vector (AODV) [14] is a reactive (on-demand) protocol. When a route is unknown, the source node broadcasts a route query packet. The destination node sends a route reply packet to the source node, and all intermediate nodes create a route entry to relay the reply packet. However, most of previously proposed protocols target to achieve high performance in high-mobility and/or high-load network, and they do not consider the traffic behavior of the upper layer protocols.

In this paper, we propose a new routing protocol that resolves the above problems and achieves low latency for the short-lived connections. We call it as the Low-latency Hybrid Routing (LHR) protocol that combines an on-demand route search and proactive route maintenance. LHR adopts a quick route re-search method against a link disconnection and a combination of routing and TCP connection establishment. In addition, LHR integrates a TCP connection establishment process with a route search process to start data transmission quickly. T/TCP [15] equips a similar method to reduce the overhead of connection establishment. It integrates a TCP connection establishment process with data transmission. LHR can also works with T/TCP although we must consider that these methods fatten the broadcasted routing packets. These are advantages of LHR over the existing proactive and on-demand hybrid routing like ADV [1]. As a result LHR can process more TCP connections within a given time period. The remainder of this paper is organized as follows. We first describe the detail of LHR in Section 2. Simulation setup and results are shown in Section 3, and concluding remarks are made in Section 4.

2. Routing Protocol for Short-Lived TCP Connections

2.1 Decreasing the Overhead of Route Table Exchange

In some existing ad hoc routing protocols, more routes are maintained in a routing table than those actually used. One example is DSDV [13] that maintains routes to all nodes. However, such routing strategy unnecessarily increases the network/terminal load because it increases the size of route table with needless routes in current transmission requests. On the other hand, LHR registers the target destination node as an active data receiver like ADV [1]. Nodes maintain only routes to active data receivers and exchange them with neighbor nodes, so that the size of the route table can be much decreased compared to DSDV. While an initial connection to an inactive receiver takes large latency with a table driven routing protocol, LHR also adopts another on-demand route search mechanism to find a route to inactive receivers quickly. We describe it in the next Subsection.

2.2 Decreasing Latency for New Route Search

Proactive routing protocols can search a route promptly if it is cached in the route table [13]. However, it requires a long time to collect routing information from all over the network, and nodes cannot transmit packets to unknown destinations.

LHR uses an on-demand initial route search and proactive route maintenance. A source node broadcasts a Route-Request (RREQ) packet to search a route to an inactive receiver. The target destination node receiving the RREQ packet broadcasts a Route-Reply (RREP) packet. All nodes receiving these two packets register the target destination node as an active receiver. Then, they begin to multicast the HELLO messages periodically to neighboring nodes to update routes. Every RREP packet is given a sequence number that indicates the routes' freshness and is increased by one at every transmission of RREP. Some nodes that could not receive the RREQ or RREP packet register the routes by receiving the HELLO packet including new routes' information. However, in case of the link disconnection, nodes must wait for the neighbors' route update message. To decrease this latency, LHR adopts another route re-search method, which will be described in the next subsection in detail.

2.3 Decreasing Latency for Route Re-search

The main originality of our proposed protocol is its route re-search method. There are several techniques

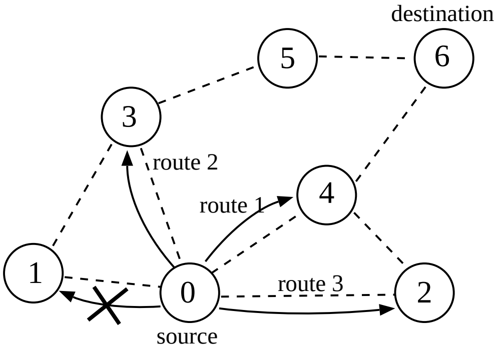


Fig. 2 Multiple Routes' Entry

listed below for recovering routes against the link disconnection, which is caused by node movement and/or changes in the wireless environment.

1. The route is updated by exchanging a route table. Its problem is that it may take long time to get new available route.
2. A route error is acknowledged to the source node to make another route request. It would be effective for long-lived connection because the new route will be short and in good quality between end-hosts. However, in an environment where there are many short-lived connections, this way apparently wastes time.
3. The RREQ packet is broadcasted from the node detecting the link failure. Though this method may make longer route than that of the above method 2, it is not a serious overhead when the connection time is short.
4. Multiple routes are always tried to be maintained beforehand.

We combined methods 3 and 4. In short, nodes suffering a link disconnection first try retransmission through method 4. If no other routes are available, they try method 3 and recover the route quickly. We describe these methods in more detail below.

In an initial route search, described in Subsection 2.2, a node may receive the identical RREP from two or more neighbor nodes. It indicates that there are multiple routes to the destination. The node caches these routes to recover a link disconnection quickly. When a node detects a link disconnection by feedback information from a data-link layer, the node inactivates routes through the link in its route table. The inactive routes are re-activated when a packet is received through the link again. If the packet transmission through any cached route does not succeed, the node initiates a RREQ to find the current available routes to the destination.

With this multiple routes maintenance mechanism, the number of route entries in the route table may increase too much. To avoid this problem, LHR adopts the limitation of route entries for each active receiver. This limitation is that when the shortest route to an active receiver has n hops, the node maintains only n hop routes and $n+1$ hop routes. See Fig. 2 as an example. The shortest route from node 0 to node 6 has two hops. The node 0 maintains only two hops and three hops routes. It is difficult to estimate the appropriate limit on the number of hop counts that the node maintains. However, we have some experiences from our past research about another ad hoc network system [16]. Based on the result in [16], the shorter route is given a higher priority. When node 0 has a packet destined for node 6, node 0 first tries the transmission on route 1, the shortest route to node 6. If the transmission fails (i.e., the link layer detects the link disconnection), node 0 inactivates the route 1, and tries the second shortest route 2. If all transmission trials fail at last, node 0 initiates a RREQ packet to search a fresh route.

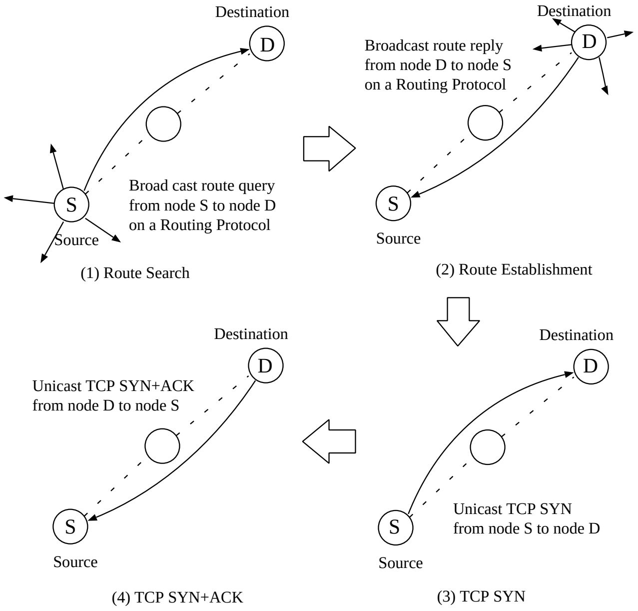
2.4 TCP Connection Establishment Integrated with Routing Protocol

The TCP connection is established by three-way handshake. At first, TCP sender and receiver exchange SYN, SYN+ACK, and ACK packets. Because this negotiation is necessary regardless of the connection time, the time for connection establishment becomes considerable especially in short-lived TCP connections.

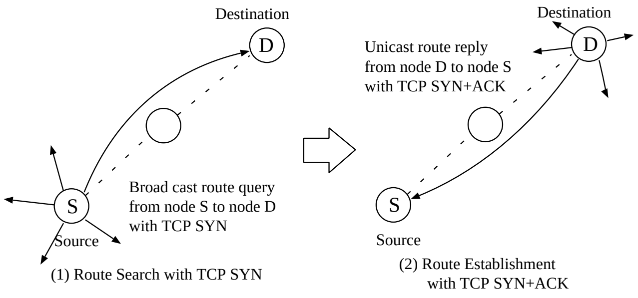
As explained in Subsection 2.2, two message packets are broadcasted at TCP end-hosts when the route to destination is unknown in LHR. Therefore, at the beginning of TCP connection on a new route, the end-hosts must exchange four packets (two routing packets and two TCP packets) as before the source node receives the SYN+ACK packet. It results a considerable latency for short-lived connections. We can decrease it by integrating TCP connection establishment with route search in LHR. See Fig. 3. When the node initiating the SYN packet finds no available route, it broadcasts the RREQ packet carrying the SYN packet together. The RREP packet also carries the SYN+ACK packet. Thus, the connection establishment time can be decreased. It is inevitable that the network load increases since the size of routing packets gains. However, it is acceptable because we now aim at decreasing the latency for short-lived connections at the expense of increased traffic load.

3. Simulation Experiments

We implemented LHR using an ns-2 network simulator [17]. We also used AODV and DSDV implementations of ns-2 for comparison. IEEE 802.11 Wireless LAN was employed at the link- and physical-layer. The radio propagation range was 250 meters and the buffer capacity of each node was 50 packets. We simulated a



(a) Sequential Operation of Route Search and TCP Connection Establishment



(b) Route Search integrated with TCP Connection Establishment

Fig. 3 Connection Establishment Flow

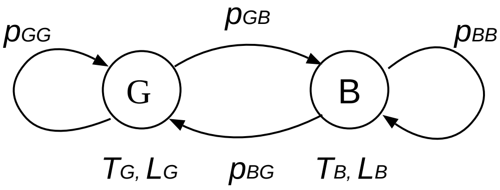


Fig. 4 Two-state Transition Error Model

500 x 2000 m^2 network field that consisted of randomly placed nodes. Their mobility pattern was based on a *random way-point* model [2]. To investigate the effect of node mobility, we used the three mobility patterns listed below.

- Max speed: 0 (m/sec)
- Max speed: 5 (m/sec), mean speed: 2.5 (m/sec), pause time: 0 (sec)
- Max speed: 20 (m/sec), mean speed: 10 (m/sec), pause time: 0 (sec)

Wireless error was modeled by a two-state transition error model (Fig. 4) known as the Gilbert model [18]. For each of the simulation experiments, we calculated the error rate from the expressions derived in [19]. The packet loss probability is 0% in the good state and 100% in the bad state. The unit time (T_G and T_B) of error and error-free states are defined as the time required for transmitting one data packet. In our simulations, this was about 5.84 msec. We employed the 1% error rate for all simulations, that is, the mean length of staying in the error-free state was set to 20,000 slots, and 200 slots in the error state. The load of the network was defined as the total number of connections generated per second in the network. All connections were destined for one data collection terminal and the average of transmission data on each connection was 5 packets.

We first compared the packet loss probability of UDP and TCP in the simulated network. In these simulations, we saw the importance of reliable transmission by TCP in a low-load network like a sensor network. Tables 1 and 2 show the results of the network where 1 and 5 connections have been generated per second respectively. Using UDP, the packet loss is considerably high also in the low-mobility and low-load network. TCP achieves low packet loss rate with any type of routing protocols although TCP increases the total amount of network traffic by connection setup and acknowledging data transfer. This is important for collecting time-dependent data.

Next, we measured TCP connection establishment delay, that is the time since TCP SYN generation until SYN+ACK receipt at a TCP source node, with LHR, AODV, and DSDV routing protocols respectively. Figures 5 and 6 show the cumulative frequency distribution of the number of connections that are established within the latency indicated on the horizontal axis. Before we describe these figures, let us explain Fig. 7 to know what we see in the former figures. In Fig. 7, the maximum value of the x-axis is extended to 20 seconds, and all simulation parameters are the same as in Fig. 5(c). We can see that the number of established connections increases sharply at 6 and 18 seconds. These were caused by retransmissions of TCP SYN packets, that is, the source nodes could not receive SYN+ACK packets until those times. As denoted in Table 1, there were little difference in the number of established connections between LHR and AODV within the simulation time. Figure 7 indicates, however, the connection establishment latency becomes sufficiently long if a routing protocol fails to transmit the first SYN and SYN+ACK packets. Based on the above result, we attended to the first connection establishment trial in Figs. 5 and 6. According to the simulation results,

Routing	LHR				AODV				DSDV			
Mobility (m/s)	0		20		0		20		0		20	
Transport	TCP	UDP	TCP	UDP	TCP	UDP	TCP	UDP	TCP	UDP	TCP	UDP
Packet Loss (%)	0	3.8	0	11	0	9.7	0.3	16.7	0	6.2	0.3	63.9

Table 1 Comparison of UDP and TCP for Intermittent Traffic (1 connection/sec)

Routing	LHR				AODV				DSDV			
Mobility (m/s)	0		20		0		20		0		20	
Transport	TCP	UDP	TCP	UDP	TCP	UDP	TCP	UDP	TCP	UDP	TCP	UDP
Packet Loss (%)	1.4	7.2	0.9	12.4	2.5	17.3	1.3	17.8	1.1	10.4	0.3	63.7

Table 2 Comparison of UDP and TCP for Intermittent Traffic (5 connections/sec)

LHR can establish more TCP connections in shorter time than other protocols. As Figs. 5(a) and 6(a) show, routing protocols with table-driven route updates (LHR and DSDV) can quickly establish many TCP connections in a network with stationary nodes, because these protocols cache necessary routes in the routing tables. However, if the number of link disconnections grows as a result of high node mobility, DSDV cannot search a correct route quickly ((b) and (c) of Figs. 5 and 6), since proactive protocols are less able to accommodate the frequently changing network topology. On-demand routing protocols are adequate for such high-mobility network. LHR outperforms AODV also in such a network, because AODV generates too many route requests broadcasted by every source node and they waste the network resources. Thus, LHR can process more connections than other protocols in a given time period.

When we think of constructing an ad hoc network consisting of many terminals, we also need to consider the node density in the network as an important network parameter. If the node density is low, the number of routes to the other nodes decreases. It causes no route to a destination, no substitute route against a link-disconnection, concentration of traffic to few routes, and increasing the hop counts of routes. In contrast, in high node density network, increase of routing traffic and interference from other nodes will degrade the network performance. Figure 8 shows how many connections are successfully established without TCP SYN retransmissions when the number of nodes placed in the same field changes. If all nodes are stationary, the network performance degrades as the number of nodes increases from 50 to 65, since increasing routing messages block the data traffic. In high-mobility network, the result indicates that the larger node density than in stationary network is acceptable. This is because, when the node density becomes large, the probability that a route broken by node movement is re-constructed increases, while the routing traffic increases.

4. Conclusion

In this paper, we have proposed a new routing protocol that is applicable to the existing Web systems where many TCP connections are short-lived. The sensor network is another important application where the TCP connections for collecting a small amount of data from many sensor terminals are major within the network. In those networks, we need to decrease the connection and transmission latency for the short-lived connections.

Our protocol LHR adopts an on-demand route search and a proactive routing update. Packet receiving nodes are registered as active receivers, and only routes to them are exchanged. Against the link disconnections due to wireless error or node mobility, LHR maintains multiple routes for each destination to decrease the route re-search latency. In addition, to decrease initial connection establishment latency, LHR route request and route reply packet can carry the TCP connection establishment packets at the same time. These features are capable of decreasing the latency of connection establishment and improving the performance for short-lived TCP connections.

Acknowledgements

This work was partly supported by Special Coordination Funds for promoting Science and Technology of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

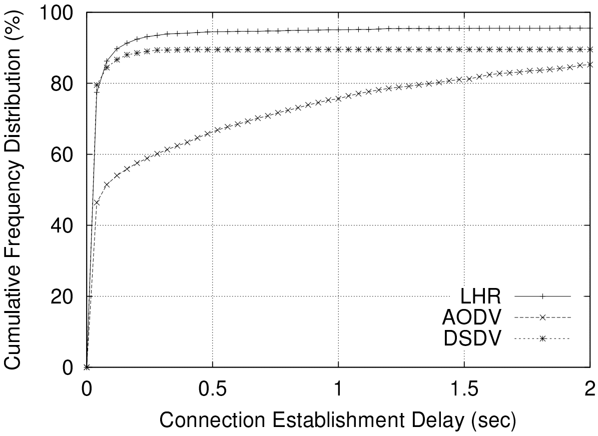
References

[1] R.V. Boppana, and S.P. Konduru, “An adaptive distance vector routing algorithm for mobile, ad hoc networks,” Proc. IEEE INFOCOM 2001, March 2001.

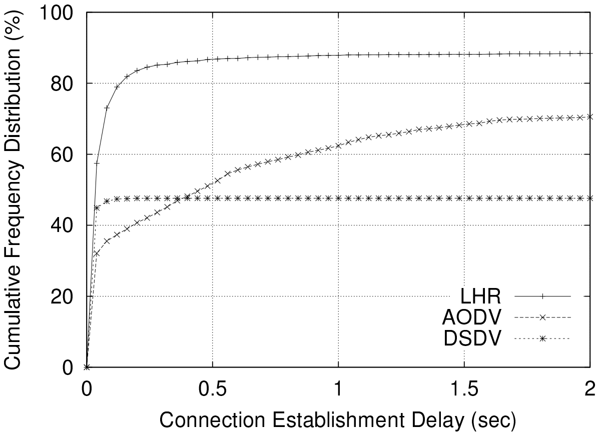
[2] J. Broch, D.A. Maltz, D. Johnson, Y.C. Hu, and J. Jetcheva, “A performance comparison of multi-hop wireless ad hoc network routing protocols,” Proc. ACM/IEEE MobiCom ’98, pp.85–97, Oct. 1998.

[3] T. Goff, N. Abu-Ghazaleh, D.S. Phatak, and R. Kahvecioglu, “Preemptive routing in ad hoc networks,” Proc. ACM SIGMOBILE 2001, pp.43–52, July 2001.

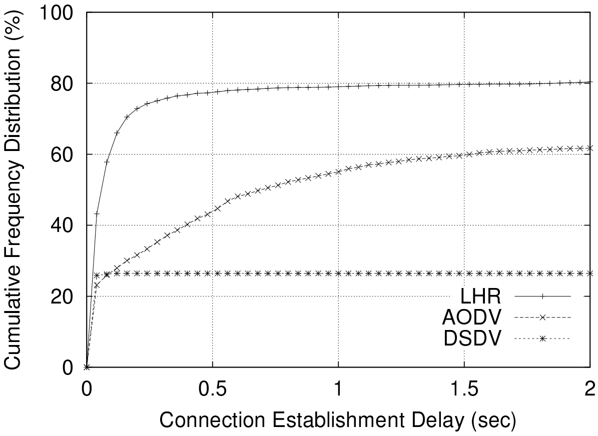
[4] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, “Scenario-based performance analysis of routing protocols for mobile ad-hoc networks,” Proc.



(a) Max Speed 0 m/sec

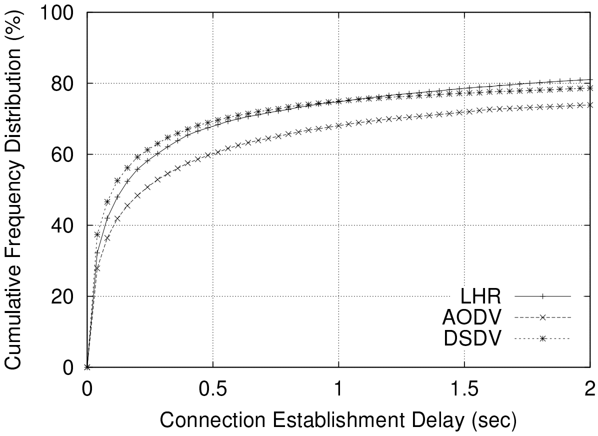


(b) Max Speed 5 m/sec

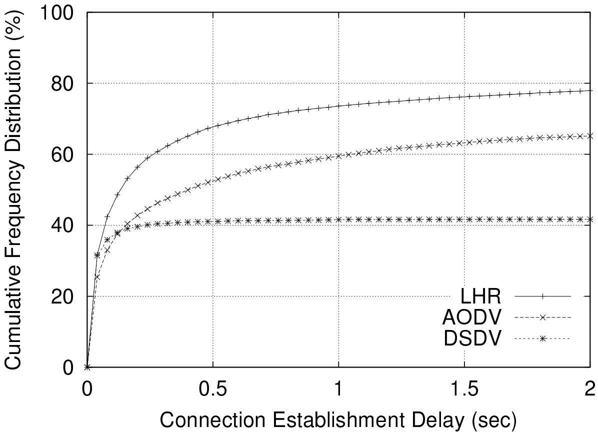


(c) Max Speed 20 m/sec

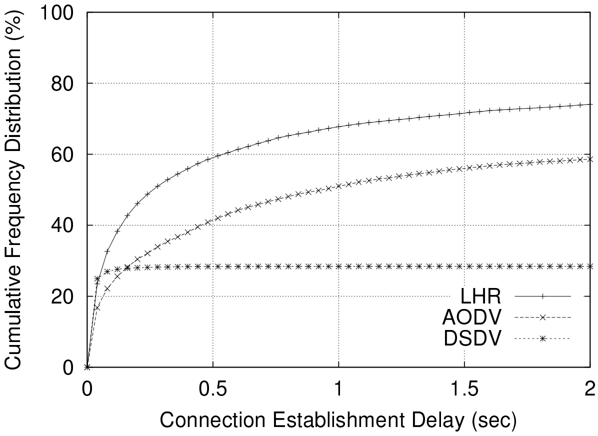
Fig. 5 Connection Establishment Delay (1 connections/sec)



(a) Max Speed 0 m/sec



(b) Max Speed 5 m/sec



(c) Max Speed 20 m/sec

Fig. 6 Connection Establishment Delay (5 connections/sec)

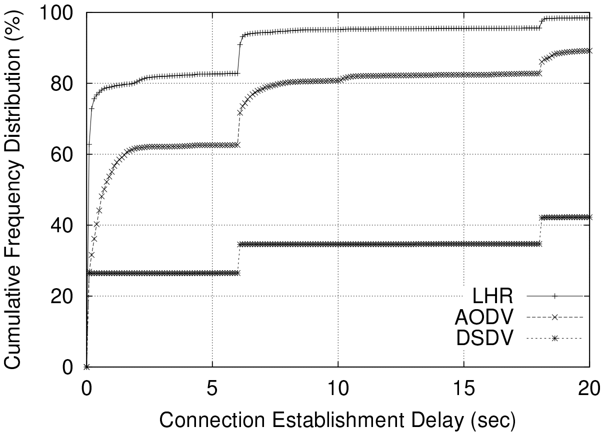


Fig. 7 Connection Establishment Retrials by SYN Retransmissions

ACM/IEEE MOBICOM'99, pp.195–206, Seattle, Aug. 1999.

[5] N. Nikaein, H. Labiod, and C. Bonnet, "DDR – distributed dynamic routing algorithm for mobile ad hoc networks," Proc. ACM/IEEE MobiHoc 2000, Aug. 2000.

[6] M.R. Pearlman, and Z.J. Haas, "Determining the optimal conguration for the zone routing protocol," IEEE J. Select. Areas Commun., vol.17, no.8, pp.1395–1414, Aug. 1999.

[7] A. Ahuja, S. Agarwal, J.P. Singh, and R. Shorey, "Performance of TCP over different routing protocols in mobile ad-hoc networks," Proc. IEEE VTC 2000-Spring, May 2000.

[8] T.D. Dyer, and R.V. Boppana, "A comparison of TCP performance over three routing protocols for mobile ad hoc networks," Proc. ACM/IEEE MobiHoc 2001, Oct. 2001.

[9] G. Holland, and N.H. Vaidya, "Analysis of TCP performance over mobile ad hoc networks," Proc. ACM/IEEE MOBICOM '99, pp.219–230, Seattle, Aug. 1999.

[10] M. Nabe, M. Murata, and H. Miyahara, "Analysis and modeling of world wide web traffic for capacity dimensioning of internet access lines," Performance Evaluation, vol.34, no.4, pp.249–271, Dec. 1999.

[11] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, "A feedback based scheme for improving TCP performance in ad-hoc wireless networks," Proc. IEEE ICDCS '98, pp.472–479, Amsterdam, May 1998.

[12] D. Kim, C.K. Toh, and Y. Choi, "TCP-BuS : Improving TCP performance in wireless ad hoc networks," Proc. IEEE ICC 2000, New Orleans, June 2000.

[13] C.E. Perkins, Ad Hoc Networking, Addison-Wesley, New York, Dec. 2000.

[14] C.E. Perkins, E.M. Belding-Royer, and S.R. Das, "Ad hoc on-demand distance vector (AODV) routing," IETF Internet Draft. `http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-10.txt`, Jan. 2002.

[15] B. Braden, "T/TCP – TCP extensions for transactions functional specifications," RFC 1644, July 1994.

[16] T. Yamamoto, M. Sugano, M. Murata, T. Hatauchi, and Y. Hosooka, "Performance improvement in ad hoc wireless networks with consideration to packet duplication," Proc. APCC 2001, pp.348–351, Sept. 2001.

[17] "The network simulator – ns-2," available at `http://www.isi.edu/nsnam/ns/`.

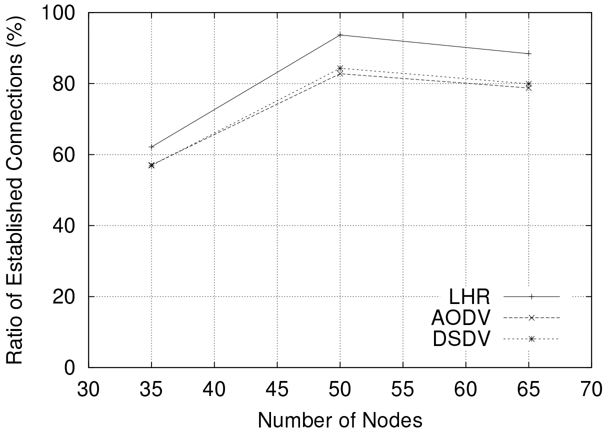
[18] E.N. Gilbert, "Capacity of a burst-noise channel," Bell System Technical Journal, vol.39, pp.1253–1265, Sept. 1960.

[19] J.R. Yee, and E.J. Weldon, "Evaluation of the performance of error-correcting codes on a gilbert channel," IEEE Trans. Commun., vol.43, no.8, pp.2316–2323, Aug. 1995.

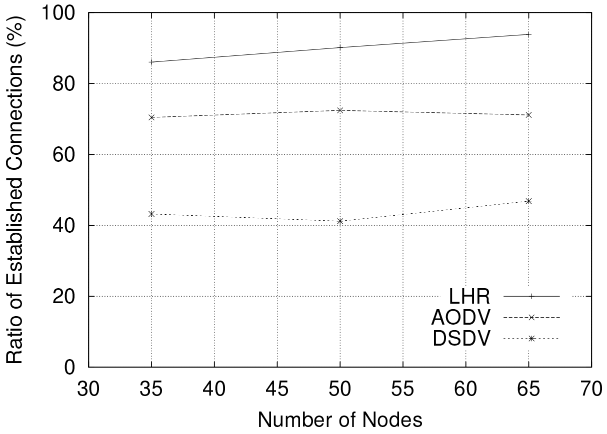
Takayuki Yamamoto received the M.E. degree in Information and Computer Sciences from Osaka University, Japan, in 2002. He is now a doctoral student at the Graduate School of Information Science and Technology, Osaka University. He is also now a research fellow of the Japan Society for the Promotion of Science. His research interests include wireless ad hoc networks and their 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 Researcher. Since September 1996, he had been Associate Professor in Osaka Prefecture College of Health Sciences. He moved to the Faculty of Comprehensive Rehabilitation, Osaka Prefecture College of Nursing in April 2003. 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, IEICE and IPSJ.

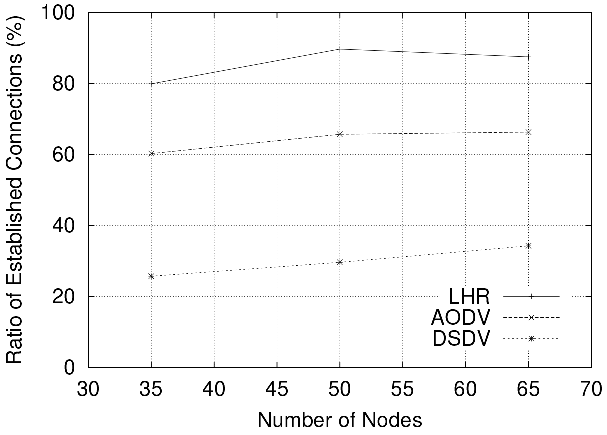
Masayuki Murata received the M.E. and D.E. degrees in Information and Computer Sciences from Osaka University, Japan, in 1984 and 1988, respectively. In April 1984, he joined Tokyo Research Laboratory, IBM Japan, as a Researcher. From September 1987 to January 1989, he was an Assistant Professor with 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.



(a) Max Speed 0 m/sec



(b) Max Speed 5 m/sec



(c) Max Speed 20 m/sec

Fig. 8 Node Density versus Successfully Established Connections without SYN Retransmission (3 connections/sec)