

Research Article

Fully Distributed Multicast Routing Protocol for IEEE 802.15.8 Peer-Aware Communication

Woongsoo Na, Yunseong Lee, Jongha Yoon, Junho Park, and Sungrae Cho

School of Computer Science and Engineering, Chung-Ang University, 221 Heukseok, Dongjak, Seoul 156-756, Republic of Korea

Correspondence should be addressed to Sungrae Cho; srcho@cau.ac.kr

Received 5 March 2015; Revised 17 June 2015; Accepted 26 July 2015

Academic Editor: Noel Crespi

Copyright © 2015 Woongsoo Na et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The IEEE 802.15.8 provides peer-aware communication (PAC) protocol for peer-to-peer infrastructureless service with fully distributed coordination. One of the most promising services in IEEE 802.15.8 is group multicast communication with simultaneous membership in multiple groups, typically up to 10 groups, in a dense network topology. Most of the existing multicast techniques in mobile ad hoc networks (MANET) have significant overhead for managing the multicast group and thus cannot be used for fully distributed PAC networks. In this paper, we propose a light-weight multicast routing protocol referred to as a fully distributed multicast routing protocol (FDMRP). The FDMRP minimizes routing table entries and thus reduces control message overhead for its multicast group management. To balance the control message, all nodes in the network have a similar number of routing entries to manage nodes in the same multicast group. To measure the effectiveness of the proposed FDMRP against the existing schemes, we evaluated performance by OPNET simulator. Performance evaluation shows that the FDMRP can reduce the number of routing entries and control message overhead by up to 85% and 95%, respectively, when the number of nodes is more than 500.

1. Introduction

Due to the proliferation of smartphones, cellular networks have experienced a 5,000 percent surge in mobile data traffic in the last three years [1]. Mobile data offloading is the use of complementary network technologies to deliver mobile data traffic originally planned for transmission over cellular networks by using device-to-device direct communication. With this requirement, the IEEE 802.15 standards group has established Task Group 8 (IEEE 802.15 TG8). The IEEE 802.15.8 standard defines PHY and MAC mechanism for wireless personal area networks (WPAN) peer-aware communications (PAC) optimized for peer-to-peer and infrastructureless communications with fully distributed coordination.

PAC is a special type of MANET techniques and its features include (1) discovering peer information without association, (2) discovery signaling rate, typically 100 kbps, (3) being scalable with respect to the number of devices and data rate, (4) group communication with simultaneous membership in multiple groups, typically up to 10, (5) relative positioning, (6) multihop relay, (7) security, and (8) being

operational in selected globally available unlicensed/licensed bands below 11 GHz capable of supporting these requirements [2]. The applications for PAC will be diverse and may include social network services (SNS), local advertisements, games, streaming applications, proximity device control, P2P services, and emergency services as in Table 1. In Table 1, most of the applications require multicast technique that supports multihop transmission. For these reasons, the PAC necessitate a fully distributed multihop multicast protocol.

In terms of MANET, there are a large variety of multicast protocols in the literature. We can classify them into two approaches: tree-based and mesh-based ad hoc multicast routing protocols. The tree-based ad hoc multicast routing [4–10] is a technique where a root node manages a multicast group based on a tree. Since the root node manages the multicast tree, it consumes resources such as memory, power, and control messages. Also, management cost of tree-based multicast routing is high because it is hard to change the structure of the tree to support mobile nodes. Moreover, if the root fails, then the multicast group has to elect a new root node to reconstruct the tree.

TABLE 1: The PAC application matrix [3].

Category	Service type
Social networking	Group communication
Entertaining	Commercial broadcast, advertisement
User-centric services	GroupCast, group gaming
Smart environment	GroupCast, group management
Health/energy monitoring	Multicast of emergency situation
Public safety	Hazard and emergency broadcast
Traffic transportation	Road state broadcast
Network offloading	Extend network coverage (multihop)
Navigation service	User finding in a group

In order to support a high mobility environment, more recent multicast routing protocols use a mesh network [11–17]. Multicast protocols using mesh networks usually provide multipath links to nodes within a multicast group. For this reason, mesh-based multicast routing protocols can respond quickly to link breakage due to mobility. However, a mesh-based protocol imposes two major problems. First, to support multipath links, each node has to store a significant amount of information for neighboring nodes and routing information for the multicast group. Second, the computation overhead of a node in a mesh-based protocol is larger than in a tree-based protocol because there are multiple transmission paths.

These multicast protocols are not suitable for PAC because existing multicast routing protocols for MANET have some problem. In tree-based protocols, node has to manage a multicast group tree. Since one of the PAC features is fully distributed coordination, tree-based multicast routing is improper for PAC. In case of mesh-based multicast protocols, there are two problems as mentioned. Moreover, mesh-based multicast routing protocols consume a lot of power from control messages to manage most of the nodes within a multicast group. Since there is a lack of memory and battery-based devices, mesh-based multicast routing protocols are not suitable for PAC.

To overcome these problems, in this paper, we propose a light-weight multicast routing protocol referred to as fully distributed multicast routing protocol (FDMRP). In FDMRP, we assume that all nodes are deployed in fully distributed manner and they maintain routing table to support multihop. Under the assumption, the FDMRP tries to minimize the number of routing entries and optimize memory space and reduce power consumption for managing a multicast group. By using the above approach, we can reduce periodic control messages to support mobile nodes.

The remainder of this paper is organized as follows. We survey the existing related work in Section 2. In Section 3, we describe the proposed FDMRP in detail. In Section 4, we describe theoretical analysis with discrete Markov chain model. Section 5 evaluates performance of the FDMRP compared with existing schemes. Finally, we draw conclusions and suggest future directions in Section 6.

2. Related Work

To facilitate the analysis of multicast protocols, we can classify them into three approaches: tree-based [4–10, 18], mesh-based [11–17], and hybrid-based ad hoc multicast routing protocols [19, 20]. Ad hoc multicast routing protocol utilizing increasing ID-numbers (AMRIS) [8] falls under the tree-based approach. AMRIS assigns every node in a multicast group with an ID-number and makes multicasting tree. This ordering between ID-numbers is used to direct multicast traffic and quick local repair for the multicast tree. Another tree-based multicast protocol by a cross-layer designed approach [5] uses multipoint relays (MRPs) for MANET. The source node selects some MPRs among its 1-hop neighbor nodes and these MPRs can cover all 2-hop neighbor nodes of the source node. Continuously, each MPR selects the next MPRs. As a result, a protocol has a tree formation. Multicast ad hoc on demand distance vector (MAODV) [6] has a multicast group leader for managing a group. The multicast group leader periodically broadcasts group hello (GRPH) message to the multicast group members to forward information. Ad hoc multicast routing protocol (AMRoute) [9] uses logical cores. Specific tree nodes are designated by AMRoute as logical cores and are responsible for initiating and managing the signaling component of AMRoute, such as detection of group members and tree setup. Extendable multicast routing protocol (EMRP) [7] is used for hierarchical multicasting in MANET environments. This protocol uses the subsource to reduce the paths among receivers. The aforementioned tree-based multicast routing protocols are very simple and their network structure is easy to set up. However, these protocols have to reconstruct the entire tree when the link failures occur. This situation will become severer when nodes are mobile. Additionally, maintaining a shared tree loses path optimality. On the other hands, maintaining multiple trees to support many source nodes imposes significant overhead such as memory, power, and control messages.

Unlike the former, there is no root node in the mesh-based multicast routing protocol. Mobility-based hybrid multicast routing (MHMR) [11] joins a cluster communication mesh. In their scheme, several nodes formed a cluster and they elect a cluster head node. In the cluster, all nodes are connected with mesh link and the cluster head node manages its own cluster members. If a cluster member wants to communicate with the other cluster member, it can communicate through its cluster head node. Core-assisted mesh protocol (CAMP) [13] is another mesh-based protocol. This protocol is proposed to solve the weakness in a tree-based multicast routing protocol. CAMP provides the multipath and the core maintains the mesh network. In network sender multicast routing protocol (NSMRP) [17], a sender transmits its multicast data frames to a special node which is called the mesh sender (MS). In turn, this MS takes the responsibility of delivering the messages to all group members in the network. On demand multicast routing protocol (ODMRP) [16] is distributed mesh-based protocol. Since the ODMRP provides the multipath for receivers, the ODMRP is robust for the link breakage and channel error. However, the ODMRP has a significant weakness if all the source nodes flood the Join-Q

message for maintaining the routing group. Therefore, the control overheads are larger than tree-based or other mesh-based multicast routing protocols. Moreover, since all receiver IDs are saved in the routing table, more memory is used for the routing table. As observed, mesh-based multicast routing protocol requires more control messages than the tree-based approach. As a result, it imposes power inefficiency, network load, and control overhead.

Hybrid-based multicast routing protocols combine with the advantages of both tree- and mesh-based approaches. Efficient hybrid multicast routing protocol (EHMRP) [19] separates data forwarding path from join query forwarding path by incorporating low overhead local clustering technique to solve the scalability issue. However, this protocol has significant overhead for clustering when the nodes have mobility. Additionally, there have been a variety of multicast routing protocols based on application dependence. These multicast routing protocols can be divided into QoS [21], energy-efficiency [22–24], network-coding [25], and reliable multicast [26]. However, this type of research is not applicable to PAC scenario since the PAC is targeting an application where the PAC nodes have light-weight routing entries and thus reduce control message overhead under the mobile environment.

As observed, the existing techniques are not appropriate for PAC networks especially because of bad scalability in densely deployed distributed topology. In order to consider the routing overhead, in this paper, we proposed light-weight multicast routing protocol for fully distributed networks. Our scheme minimizes the number of routing entries to reduce control overhead under the mobile environment and unnecessary control messages.

3. Fully Distributed Multicast Routing Protocol

3.1. System Model and Basic Assumption. In the FDMRP, we assume a multihop mesh topology consisting of N PAC devices (PDs) in R^2 area. The PDs in the networks are deployed according to two-dimensional Poisson point process with density λ , that is, probability of finding i PDs in an area of size R^2 as follows: $(\lambda R^2)^i e^{-\lambda R^2} / i!$. PDs are assumed to have simultaneous communication sessions for same or different applications. Also, we assume that a PD can join or leave a multicast group by its request and it can be either multicast sender or receiver. The transmission time or the packet length is the same for all PDs, which is denoted by T . The time axis is divided into slots with duration equal to δ . For the communication between two nodes within one-hop, the slotted CSMA protocol is assumed. Therefore, PDs are allowed to transmit at the start of each slot. Therefore, the number of spending slots for transmitting a packet is $\tau = T/\delta$. All PDs always have multicast packets waiting to be sent except during their transmissions. At the beginning of every slot, a nontransmitting PD tries to transmit according to a Bernoulli process with parameter p , where $0 < p < 1$. All the PDs use the same and fixed transmission power. Therefore, the range of transmission of each PDs is defined to be r .

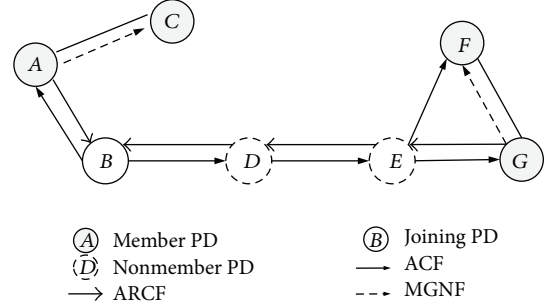


FIGURE 1: The multicast group joining procedure of the FDMRP.

3.2. Multicast Group Management

3.2.1. Finding/Joining Multicast Group. A multicast group consists of two or more PDs with the same application type ID (I_{at}), application specific ID (I_{as}), application specific group ID (I_{ag}), and device group ID (I_{dg}). It can be formed only if two or more PDs can recognize themselves. Before a PD joins a multicast group, it has to find a multicast group within K -hop coverage. If the PD cannot find the group, then it retries to find the group periodically. In order to find a multicast group, a PD broadcasts an advertisement command frame (ACF) after a random time T_j where the maximum time to live (TTL) is set to K . Range of T_j is $[0, T_{jmax}]$. If a PD receives the ACF, it stores the ACF in order to forward it to other PDs and saves a backward path in the routing table during expiration timer T_w . T_w is determined by the one-hop round trip time (RTT) and K . If the PD is relay-enabled PD, it compares the receiving frames I_{at} , I_{as} , I_{ag} , and I_{dg} with its own. If they are all the same, it deletes the ACF and replies with an advertisement reply command frame (ARCF) to the PD who transmits the ACF (reply of the ARCF depends upon the multicast group notification frame (MGNF) explained as follows). If any of them does not match the same, the PD receiving the ACF decreases the TTL of the ACF and forwards the ACF. In order to limit the duplicate ARCF, PDs which have to respond to the ARCF transmits a multicast group notification frame (MGNF) to its members after a random time (MGNF is explained in detail later). Then, the PD multicasting the MGNF replies to the source PD with an ARCF by using a backward path. A PD receiving both ACF and MGNF does not reply to an ARCF because the PD knows that there is another PD in the same multicast group which will reply to the ACF. A PD receiving the ARCF whose destination is not itself saves the route information for the ID of the source PD, ID of the one-hop PD sending the ARCF, I_{at} , I_{as} , I_{ag} , and I_{dg} . Then, this PD is referred to as a *forwarding PD*. A PD receiving the ARCF whose destination is itself saves the route information as a forwarding PD. Figure 1 shows an example of joining procedure of the FDMRP. Assume the topology that A and C are within the same group and F and G are within the same group. If B wants to join a group, B broadcasts an ACF. Then, nodes A and G receive the ACF and they multicast a MGNF to their group. Then, C and F do not react from the ACF. After multicasting the MGNF, A and G

TABLE 2: Type of multicast group notification frame (MGNF).

Type	Description
0	Limiting duplicate ARCF
1	Management of routing table
2	Leaving notification to multicast group
3	Device group ID creation
4	Request for unicast routing
5	Reply for unicast routing
6	Mobility support
7	Local repair
8	Notification of removed routing entry

unicast a ARCF to B , and joining procedure of B is end. The joining algorithm is shown in Algorithm 1.

By proposed joining procedure, we can reduce the number of routing entries since each PD maintains entries only for PDs that has exchanged ACFs and ARCFs in its routing table. Figure 2 shows an example of the number of routing entries when we use our proposed joining procedure. In Figure 2(a), all member PDs maintain entries for other member PDs. Additionally, forwarding PDs maintain entries for member PDs as well as forwarding PDs. Therefore, the total number of entries is 40. On the contrary, in Figure 2(b), each of the member PDs A and D only maintains entry for PDs D and A , respectively, because they exchange ACF and ARCF. Since they do not exchange ACF and ARCF with other member PDs (e.g., PDs E , G , and I), they do not maintain entries for other member PDs. Similarly, since nodes E , G , and I exchange their ACF and ARCF, node E only maintains entries for PDs G and I . Then, the proposed technique can reduce the routing table size as 16.

3.3.2. Device Group ID Creation. A multicast group is determined by I_{at} , I_{as} , I_{ag} , and I_{dg} as mentioned earlier. Therefore, it is inefficient to transmit all IDs in a frame. For this reason, we propose a device group ID creation scheme. The device group ID should be unique and distributed by a PD that sends the first ARCF in the group. The PD generates a device group ID based on its unicast ID. Since a PDs unicast ID is unique, prefix concatenated by PDs unicast ID is also unique. When two or more multicast groups are merged, the device group ID should be the same.

3.3. Multicast Group Notification Frame (MGNF). The purposes of MGNF are (1) limiting duplicate ARCFs, (2) managing the routing table, (3) leaving notification to multicast group, (4) device group ID creation, (5) request for unicast routing, (6) reply for unicast routing, (7) mobility support, (8) local repairs, and (9) notification of removed routing entries. These are listed in Table 2.

3.3.1. Limiting Duplicate ARCFs. A PD receiving an ACF multicasts an MGNF with notification type set to 0 with random timer T_j . The range of T_j is $[0, T_{jmax}]$. Since a missing MGNF can increase duplicate ARCFs, MGNFs are not retransmitted to avoid flooding. In this case, the payload of MGNF contains source information on the ACF.

3.3.2. Management of Routing Table. In order to reduce entries in the routing table, each PD maintains entries only for PDs that have exchanged ACFs and ARCFs in their routing tables. Each PD in a multicast group multicasts MGNFs periodically such as $(T_p + T_j)$ where T_p is period of the MGNFs, with notification type set to 1. Upon receiving a MGNF, a forwarding PD updates the entries of the originator of the MGNF; the one-hop PD sends the MGNF in its routing table and forwards the MGNF. Upon receiving a MGNF, a nonforwarding PD updates the entries of the originator of the MGNF and the one-hop PD sends the MGNF in its routing table but does not forward the MGNF. Also, it can be used for detection and routing table updates if link breakage occurs. When a link between two nodes is broken, the other member nodes in the group which are aware of the link breakage remove the entry for the node which has a broken link in its routing table.

3.3.3. Leaving Notification to Multicast Group. There are several reasons for a PD to leave the network: (i) *by its intention*, (ii) *by mobility*, and (iii) *by limited resources*. If a PD wants to leave from a multicast group, it multicasts (within K -hops) a MGNF with notification type set to 2. Upon receiving the MGNF, a forwarding PD deletes the entry of the originator of the MGNF and forwards it. On the other hand, the nonforwarding PD does not forward the MGNF.

3.3.4. Device Group ID Unification. When two or more multicast groups or nodes are merged, the device group ID should be identical. The PD that recognizes the existence of two or more multicast groups determines the device group ID for those groups randomly. Then, the PD sends MGNF (notification type: 3 and TTL: ∞) to the group that does not have the selected group ID to update the multicast group ID.

3.3.5. Requests for Unicast Routing. When a PD wants to unicast data frame, the PD searches routing entry of destination address in its routing table. If the PD finds the routing entry of destination address, it starts to unicast immediately. If it does not find routing entry of destination address, it multicasts a MGNF (notification type: 4) to group.

3.3.6. Reply for Unicast Routing. When the other PD receives the MGNF (notification type: 4), it saves backward route information in its routing table. Then, it starts to find a routing entry of destination address in its routing table. If a PD receiving MGNF finds routing entry of destination address, it unicasts a MGNF (notification type: 5) to the PD which wants to unicast.

3.3.7. Mobility Support. When a node moves within the one-hop coverage of another node, both of the nodes are aware of each other within the one-hop by TTL of MGNFs and they update their routing table. Then one of them sends a MGNF (notification type: 6) to another neighbor node. Therefore, the notifying node becomes aware that the nodes became closer. However, if the node is a multicast group member, it does not do anything. If it is a forwarding PD, it deletes the routing entry whose destination field is for both of those

```

(1) if a node wants to join a multicast group then
(2)   Broadcasts an ACF
(3) end if
(4) if a frame is received then
(5)   if received frame is ACF then
(6)      $\mathbb{R}_{at} \leftarrow$  application type ID of the ACF;
(7)      $\mathbb{R}_{as} \leftarrow$  application specific ID of the ACF;
(8)      $\mathbb{R}_{ag} \leftarrow$  application specific group ID of the ACF;
(9)      $\mathbb{R}_{dg} \leftarrow$  Device group ID of the ACF;
(10)    if  $(\mathbb{R}_{at} = I_{at}) \ \& \ (\mathbb{R}_{as} = I_{as}) \ \& \ (\mathbb{R}_{ag} = I_{ag}) \ \& \ (\mathbb{R}_{dg} = I_{dg})$  then
(11)      if originator of ACF is not in a routing table then
(12)        Save routing entry for the originator of the ACF;
(13)        Multicast MGNF (Type: 0);
(14)        Reply ARCF through backward path;
(15)      end if
(16)    else if
(17)      Decrease the TTL of the ACF and forward the ACF to neighbors;
(18)    end if
(19)  else if received frame is ARCF then
(20)    if destination of ARCF = my address then
(21)      Save routing information for originator of the ARCF;
(22)      Become a multicast group member;
(23)    else
(24)      Save routing information for originator of the ARCF;
(25)      Become a forwarding PD;
(26)      Forward the ARCF to the destination of the ARCF;
(27)    end if
(28)  end if
(29) end if
  
```

ALGORITHM 1: Joining algorithm of the FDMRP.

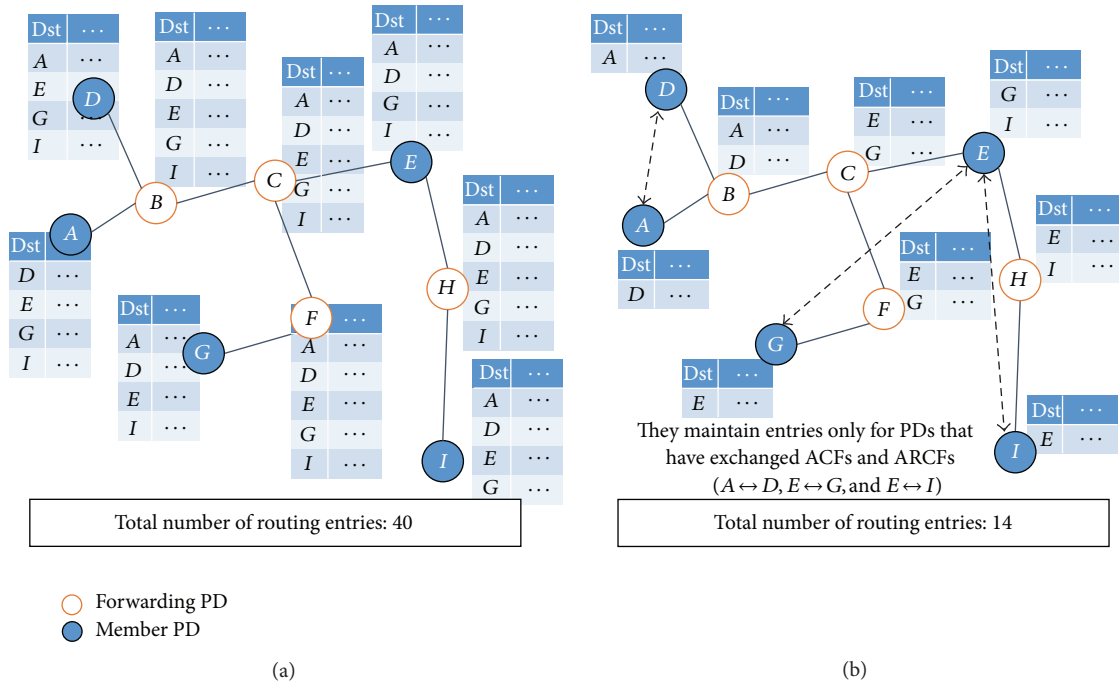


FIGURE 2: The comparison between heavy-weight routing table and light-weight routing table.

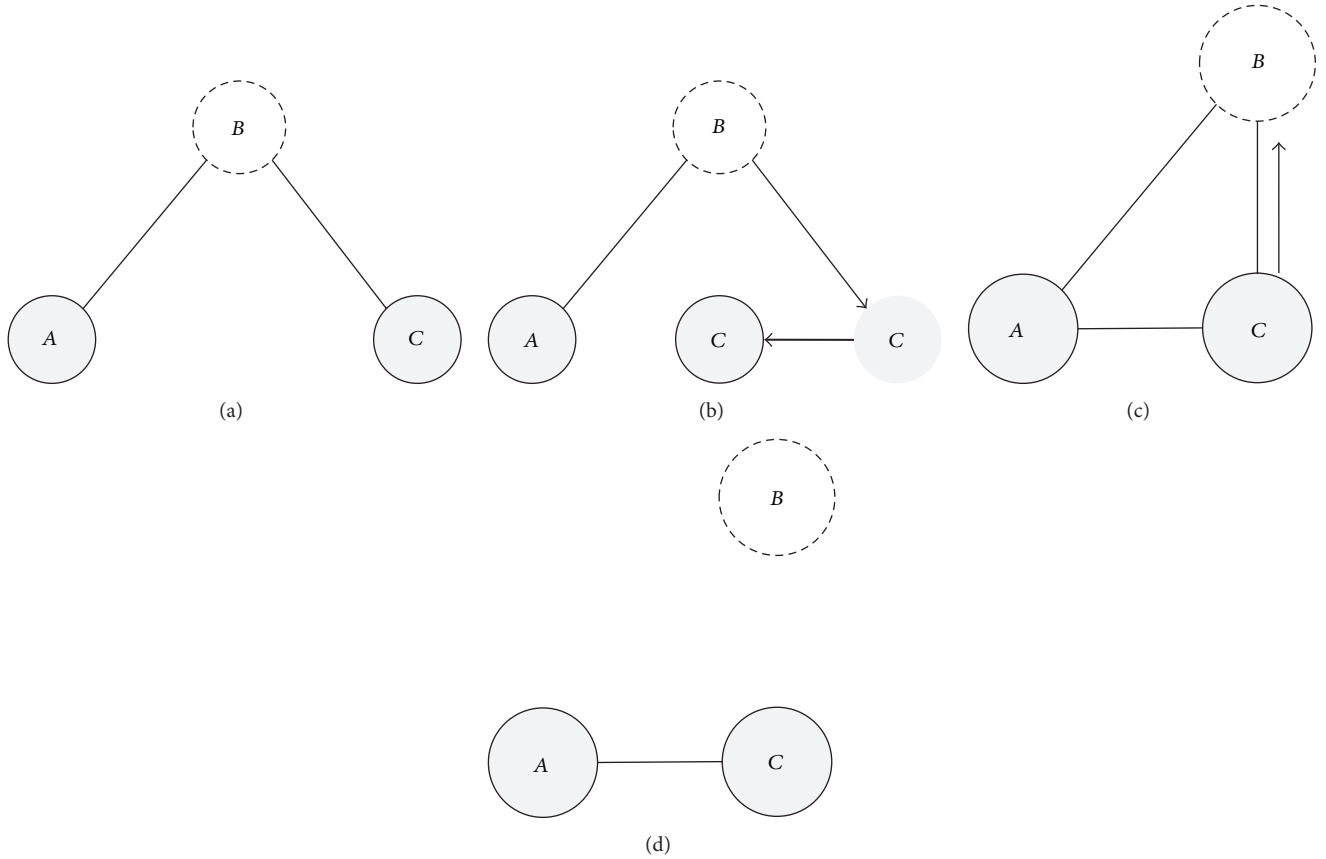


FIGURE 3: An example of mobility support (all nodes A, B, and C are separated by 1-hop).

nodes. Assume the topology that Figure 3(a). If C moves near to A (Figure 3(b)), A and C recognize each other because A directly listens to C's MGNF frame. When C recognizes A, C unicasts a MGNF to B to notice that A and C become neighbor to each other. Then, B deletes routing entries of A and C.

3.3.8. Local Repair. Since an expiration timer is proportional to the number of hops from the impaired PD, the closest multicast group member detects link breakage. A PD starts local repair if it detects link breakage between multicast group members. Then, the PD multicasts a MGNF (notification type: 7). If PDs receiving the MGNF do not have an existing routing entry for originator of the MGNE, a routing entry is created and an expiration timer T_M is set to where the threshold time for an ACF decision is set. Subsequently, the PD performing local repair broadcasts an ACF within K -hops of coverage. If a PD receives the ACF, it compares the receiving frames I_{at} , I_{as} , I_{ag} , and I_{dg} with its own. If it does not match its own, it decreases the TTL of the ACF and forwards the ACF. If it matches its own, the PD finds an entry for the originator of the ACF in its routing table. When it finds the entry, the PD receiving the ACF does not reply. In other cases, the PD replies with an ARCF to the originator of the ACF.

3.3.9. Notification of Removed Routing Entry. When routing table of a node is full of routing entries, it chooses another

node from its routing table with shortest distance (hop-based) firstly. It sends a MGNF (notification type: 8) to the chosen node and sets timer. When the node receives the MGNF, it breaks the link between the node and itself and sends an ACF. The node which has timer ignores the ACF and another node which receives the ACF sends an ARCF to it and creates a new link between the node which sends the ACF and itself.

3.4. Creation and Management of Routing Table

3.4.1. Creation of Routing Table. Whenever a relay-enabled PD receives an ACF or an ARCF, a routing entry is created in the routing table. Also, the routing table is updated by a receiving MGNF. A routing table contains a destination address, next-hop address, expiration timer, number of hops, current sequence number (*current_SN*), device group ID, and last ACF reception time (T_0).

3.4.2. Management of Routing Table. Since MGNF (notification type: 1) is sent periodically, a MGNF implosion problem can occur in the network. To prevent this problem, we propose an adaptive MGNF transmission technique based on the distance between nodes. The distance can be measured from the received signal strength indicator (RSSI). This technique is used when the distance between nodes becomes longer or shorter. If the node moves away from the network, the MGNF

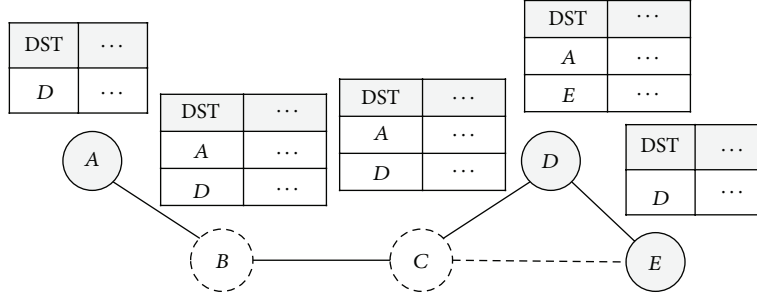


FIGURE 4: The routing table of the FDMRP.

transmissions become more frequent since there can be a link breakage. In contrast, if the node moves within one-hop coverage (e.g., mobility support), the MGNF transmissions become less to prevent redundant MGNF traffic.

3.4.3. Merging Multicast Groups. The finding/joining procedure using ACFs and ARCFs can help to merge disjoint groups. If a PD in a group receives an ACF from different disjoint groups with the same I_{at} , I_{as} , I_{ag} , and I_{dg} , it can initiate a merging process. Then the PD replies to the ACF originator with an ARCF. Then, these two disjoint groups are merged by a device ID creation scheme. Local repair using ACFs and ARCFs can help merge disjoint groups. Each group member performs local repair periodically during T_L (long duty cycles) in order to merge disjoint groups.

3.5. Multicast Data Transmission. If a PD receives a multicast data frame, it has to decide whether to forward the frame or not. The PD receiving the multicast data frame compares the source address of the data frame and next-hop address entries of its routing table. The PD checks the next-hop addresses in its routing table. If it finds one or more next-hop entries which do not overlap with the source address of the received frame and those next-hop entries have the same device group ID as the received frame, then, the PD forwards the incoming data frame to other PDs. Otherwise, the PD does not forward the incoming frame. For the multicast data transmission, we have to know the destination address, source address, originator address, sequence number, and TTL from the multicast data frame. This information should be included in all multicast data frames. In Figure 4, there are five nodes in the topology. A, D, and E are in the same group; B and C are a forwarding PD. Although C and E are in one-hop range, routing table creation policy of the FDMRP does not make cyclic links. If A sends multicast data, B compares source address and routing table. Since B has one next-hop entry which is not overlapped with A, B forwards the data. In the same way, C and D also forward the data. Then, a group including A successfully receives the multicast data from A.

3.6. Prevention Loopback Problem. PDs can prevent multicast/broadcast loopback problems by using the sequence number (SN) of data frames. When a PD forwards data with SN = n , the PD sets the *current_SN* field in its routing

table to n . If the receiving frames SN is not greater than the *current_SN*, the PD discards the frame.

4. Theoretical Analysis of FDMRP

As mentioned earlier, the ARCF is delivered by a PD who transmits a MGNF (type: 0). However, in the special topology, this procedure causes overheads as in Figure 5. We can observe that multicast creation overheads of Figure 5(a) are less than those of Figure 5(b). To resolve this problem, PDs have to negotiate other PDs to decide a PD that will send ARCF to a PD sending ACF. The negotiation process is initiated by the PD who receives an ACF. The MGNF for negotiation contains the number of routing entries. If a PD receives an ACF, the PD multicasts MGNF with the number of routing entries of the PD. When transmission of MGNFs is finished, the PD which has the largest number of entries replies with an ARCF to the PD sending ACF.

Although multicast transmission has advantage, negotiation based joining scheme has a lot of overheads for joining process. This is because all PDs receiving ACF have to multicast MGNFs. To determine appropriate joining scheme, we analyze the control overhead of joining procedure and multicast.

4.1. Success Probability of CSMA in Multihop Networks. The success probability of transmission depends on neighboring PDs. Therefore, the precise computation of the probability is difficult. We use the Markov model to evaluate the average success probability of a transmission. In each slot, a nontransmitting PD may become ready to transmit with probability p . When a PD senses the channel is idle, the PD transmits its frame. We assume that, in each slot, a nontransmitting PD transmits a frame with probability p' where $p' = p \cdot P_I$, where P_I is the probability of the channel sensed to be idle in a slot.

To determine a probability of the channel which is sensed to be idle in a slot, we design the channel process by a two-state Markov chain shown in Figure 6. According to the system model in Section 3.1, the probability that there are i PDs in a disc area where radius is R is given by

$$p(i) = \frac{(\lambda\pi R^2)^i}{i!} e^{-\lambda\pi R^2}. \quad (1)$$

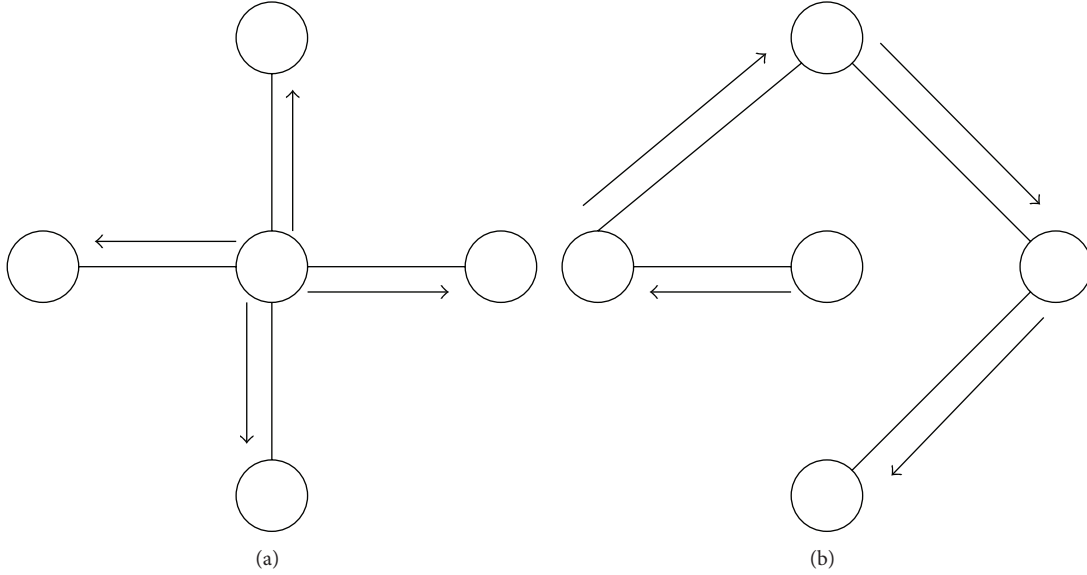


FIGURE 5: (a) Multicast is done with 1 transmission; (b) multicast is done with 4 transmissions.

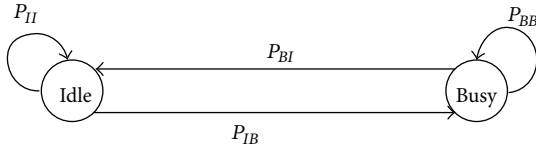


FIGURE 6: The Markov chain for a channel.

Therefore, the transition probability that channel remains idle, denoted by P_{II} , can be expressed as

$$\begin{aligned}
 P_{II} &= \sum_{i=0}^{\infty} (1-p')^i \frac{(\lambda\pi R^2)^i}{i!} e^{-\lambda\pi R^2} \\
 &= \sum_{i=0}^{\infty} \frac{((1-p')\lambda\pi R^2)^i}{i!} e^{-\lambda\pi R^2(1-p')} e^{-p'N} = e^{-p'N},
 \end{aligned} \quad (2)$$

where N is the expected number of PDs in πR^2 area where $N = \lambda\pi R^2$. Additionally, we assume that $P_{BI} = 1 - P_{IB}$.

Denote the steady-state probability of Idle and Busy states as $P(I)$ and $P(B)$, respectively. From Figure 6, we have $P(I) = P(I)P_{II} + P(B)P_{BI}$. From the property that $P(B) = 1 - P(I)$, we have

$$P(I) = \frac{P_{BI}}{1 - P_{II} + P_{BI}} = e^{-p'N}. \quad (3)$$

The limiting probability P_I that a channel is in Idle state can be obtained by

$$\begin{aligned}
 P_I &= \frac{D_I P(I)}{D_B (1 - P(I)) + D_I P(I)} \\
 &= \frac{\delta}{(1 - e^{-p'N})T + \delta e^{-p'N}},
 \end{aligned} \quad (4)$$

where $D_I = \delta$ is the duration of Idle state and $D_B = T$ is the duration of Busy state, respectively.

From (4), the transmission rate p' can be obtained by

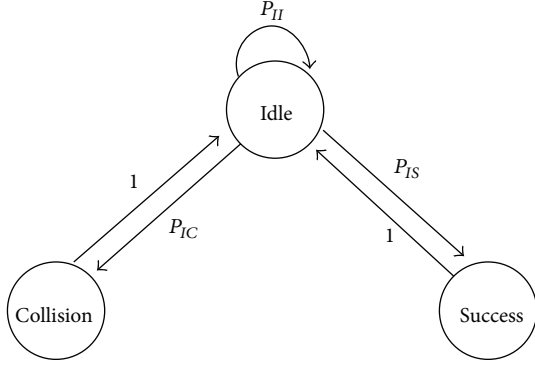
$$p' = \frac{\delta p}{(1 - e^{-p'N})T + \delta e^{-p'N}}. \quad (5)$$

Now, we determine the probability of a successful transmission from PD in a slot. The transmission stages of PD can be represented by a Markov chain which is shown in Figure 7. The Markov chain consists of three states: the Idle state, the Successful transmission state, and the Collision state. At the beginning of a slot, if a PD is in the Idle state, it leaves Idle state with probability p' . Therefore, the transition probability P_{II} which is shown in Figure 7 is given by $P_{II} = 1 - p'$. If the transmission is successful, a PD enters Success state. Otherwise, it enters Collision state. The duration of time of Success and Collision states is equal to the duration of a packet transmission time T . From Figure 7, we can get $P(I)$ obtained by

$$\begin{aligned}
 P(I) &= P(I)P_{II} + P(S) + P(C) \\
 &= P(I)P_{II} + P(I)P_{IS} + P(I)P_{IC} \\
 &= \frac{1}{1 - p' - P_{IS} - P_{IC}}.
 \end{aligned} \quad (6)$$

We can determine the transition probability P_{IS} . Let $P_{IS}(r)$ denote the transition probability when a PD is sending a packet to another PD. The distance between the sender and receiver is less than r . From the condition for successful transmission, we have

$$\begin{aligned}
 P_{IS}(r) &= P[\text{a sender transmits in a slot}] \\
 &\quad \cdot P[\text{a receiver does not transmit in the same slot}]
 \end{aligned}$$

FIGURE 7: The Markov chain for a PD x .

- P [PDs in $C(r)$ do not transmit in the same slot]
 - P [PDs in $M(r)$ do not transmit for 2τ + 1 slots],
- (7)

where $C(r)$ is the intersection of regions between a transmission region of the sender and a transmission region of the receiver. $M(r)$ denotes the transmission region which means the following: (a transmission region of the sender) – (a transmission region of the receiver). To obtain $C(r)$ and $M(r)$, we assume d is the distance between the sender and the receiver. Then, $C(r)$ is obtained by

$$C(r) = 4 \int_{d/2}^r \sqrt{r^2 - x^2} dx. \quad (8)$$

From (8), we can derive $M(r)$ as

$$M(r) = \pi r^2 - C(r). \quad (9)$$

Now, we can obtain the probability that there are i PDs in $C(r)$ given by

$$p(i) = \frac{(\lambda C(r))^i}{i!} e^{-\lambda C(r)}. \quad (10)$$

Therefore, we can obtain the probability that there are no transmitting PDs during one slot in $C(r)$ which is

$$p_c(r) = \sum_{i=0}^{\infty} (1-p')^i \frac{(\lambda C(r))^i}{i!} e^{-\lambda C(r)} = e^{-p' \lambda C(r)}. \quad (11)$$

Similarly, the probability that there are no transmitting PDs in $M(r)$ during one slot is given by

$$p_m(r) = \sum_{i=0}^{\infty} (1-p')^i \frac{(\lambda M(r))^i}{i!} e^{-\lambda M(r)} = e^{-p' \lambda M(r)}. \quad (12)$$

Therefore, $P_{IS}(r)$ can be obtained by

$$\begin{aligned} P_{IS}(r) &= p' (1-p') p_c(r) (p_m(r))^{2\tau+1} \\ &= p' (1-p') e^{-p' \lambda C(r)} e^{-p' \lambda M(r)(2\tau+1)}. \end{aligned} \quad (13)$$

Since the PDs are uniformly distributed in region πr^2 , the probability density function of the distance d between x and its receiver y is given by

$$f(d) = \frac{2d}{r^2}, \quad 0 < d < r. \quad (14)$$

Therefore, the transition probability P_{IS} for x can be obtained by

$$P_{IS} = \int_0^r f(d) P_{IS}(r) dd. \quad (15)$$

Finally, from Figure 7, the steady-state probability of state S can be expressed as

$$P(S) = P(I) P_{IS} = \frac{P_{IS}}{P_{II} - p'}. \quad (16)$$

4.2. *Joining Overhead.* In non-negotiation-based joining scheme, the joining overhead O_J can be obtained by

$$O_J = T + (N_c T + T) P_f C, \quad (17)$$

where N_c is the number of PDs in one multicast group, P_f is the probability of failure when transmit is equal to $1 - P(S) - P(I)$, and C is the number of multicasting groups. The sizes of MGNF, ACF, and ARCF are slightly different but for the sake of simplicity, we assume that sizes are the same. In the other case, the joining overhead of negotiation based joining scheme O_N is given by

$$O_N = T + \left(\frac{P_f N r^2 T N_c}{R^2} + T \right) C. \quad (18)$$

In Figure 8, we compare O_J and O_N versus the number of nodes in the entire network. From limited calculation of N_c , we assume N_c is equal to $N/10$. Also, the parameter r is equal to 100; R is equal to 1000. We can observe that O_J is less than O_N in most of the part. However, O_J is greater than O_N in lower λ . This tendency is observed at other parameters. This is because O_J is influenced by C while O_N is seriously influenced by N_c . This result helps us to decide what scheme is better under the network environment.

4.3. *Multicasting Overhead.* In this subsection, we evaluate the multicasting overhead with mobility. The multicasting overhead is correlated with the number of the forwarding data and the number of the rejoining data frames (local repair) by mobility. First, the overhead with one multicast data O_D can be obtained by

$$O_D = \frac{T_d (N_c - N_x)}{1 - P_f}, \quad (19)$$

where T_d and N_x are the size of data and the number of PDs which have only one routing entry, respectively. This is because if a PD transmits a multicast data, the neighbor PDs which have two or more routing entries forward the multicast data. However, if a PD which has only one routing entry

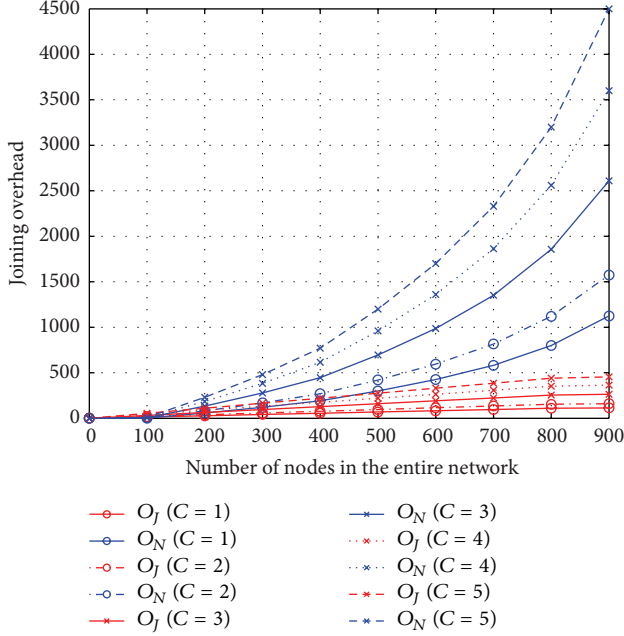


FIGURE 8: The joining overhead versus the number of nodes in the entire network.

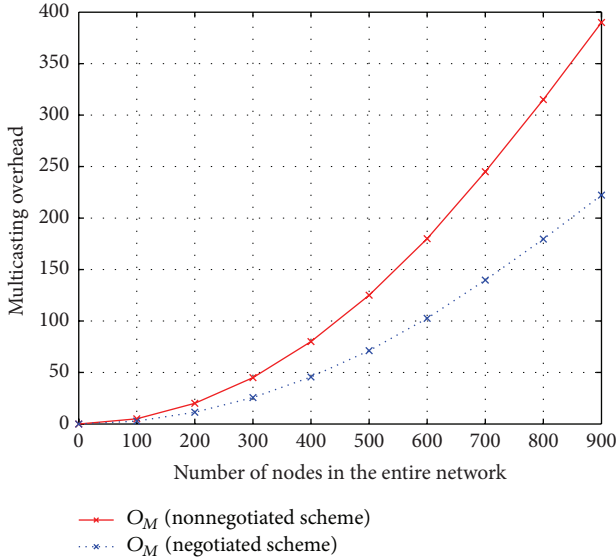


FIGURE 9: The multicasting overhead versus the number of nodes in the entire network.

receives a multicast data, the PD does not need to forward the multicast data. Also, the overhead of local repair O_L can be obtained by

$$O_L = \frac{(T \cdot C)}{1 - P_f} + \text{joining overhead } (O_J \text{ or } O_N). \quad (20)$$

If a PD starts local repair, the PD multicasts a MGNF (notification type: 7) and broadcasts an ACF subsequently. The first term of (20) means that overhead correlates with C . Also, second term is the overhead with joining process. We

TABLE 3: Simulation parameters.

Parameter	Values
ACF size	32.75 bytes
MGNF size	48 bytes
# of groups	4
# of nodes per group	2~128
Topology size	500 m \times 500 m
Transmission range	100 m
Drop model	Two-step uniform random drop model
Data rate	10 Mbps
Number of channels	1

assume that P_m is the probability of link break due to mobility per unit time δ . Finally, we can obtain a multicasting overhead O_M per δ given by

$$O_M = pNO_D + O_L P_m. \quad (21)$$

Figure 9 shows the multicasting overhead per δ . We assume the parameters $\delta = 0.002$, $P_m = 0.001$, and $p = 0.001$, N_x of negotiation based scheme is $N_C/2$, and N_x of non-negotiation-based scheme is $N_C/10$. In this figure, we can observe that the overhead using negotiation based scheme is lower than the overhead using non-negotiation-based scheme. The mobility of network and the joining scheme had a big impact on the multicasting overhead since the mobility affects overhead of local repair, and the joining scheme affects overhead of multicasting.

Through theoretical approach to get a joining overhead and a multicasting overhead, we can decide which approach is appropriate for the network environment.

5. Performance Evaluation

In this section, we evaluate the performance of the proposed protocol compared with the ODMRP [16] and path-aggregation-tree multicast routing scheme (PAT) [22]. For performance evaluation, we use the OPNET modeler [27] and we measure the number of routing entries, control overhead, joining latency, and areal sum goodput during a 1 hour simulation time. In the simulation, we form a two-step uniform random drop model where the nodes are located within 500 m of each other. MAC and PHY layer system models follow the TG8 technical guidance document [28]. The simulation parameters are listed in Table 3.

Figure 10 shows the control frame overhead versus the number of nodes. In the figure, control overhead of the FDMRP is lower than the ODMRP and PAT. This is because in the ODMRP the join request message floods frequently and periodically for managing the multicast group. Also, when a PD receives join request message, it continuously relays to its neighboring nodes until maximum TTL is reached. In the PAT scheme, since they try to maintain multiple trees for transmission, the control overhead increases as the number of nodes increases. Additionally, when a new node joins a multicast group, all nodes should reconstruct all of multicast trees. As a result, it causes significant control overhead. On

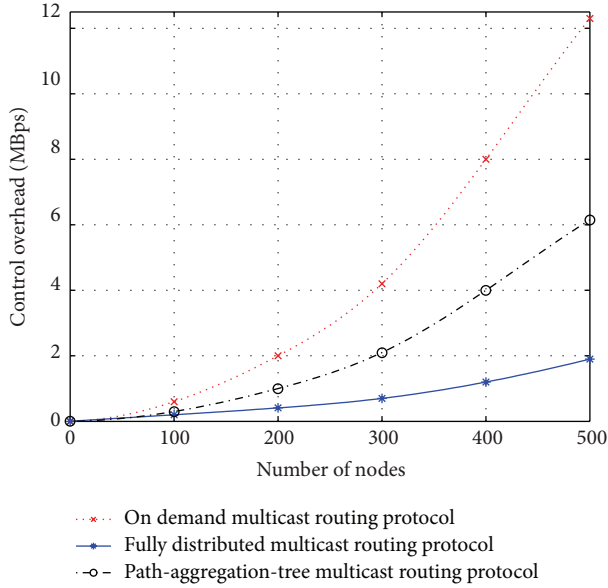


FIGURE 10: Control frame overhead versus # of nodes.

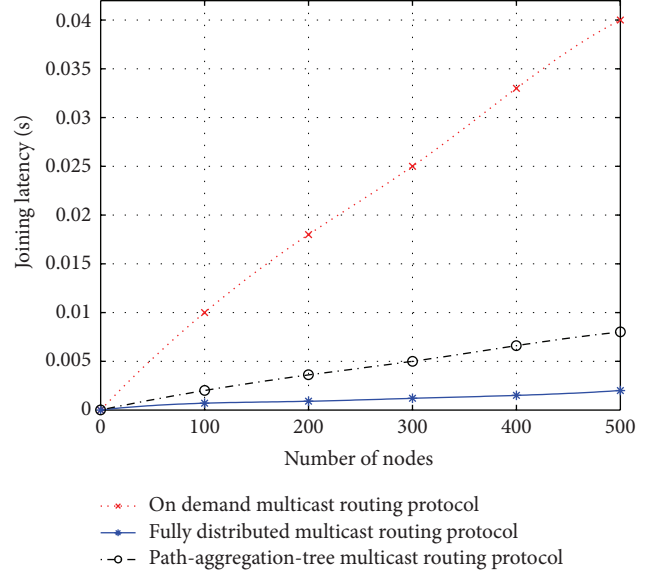


FIGURE 12: Joining latency versus # of nodes.

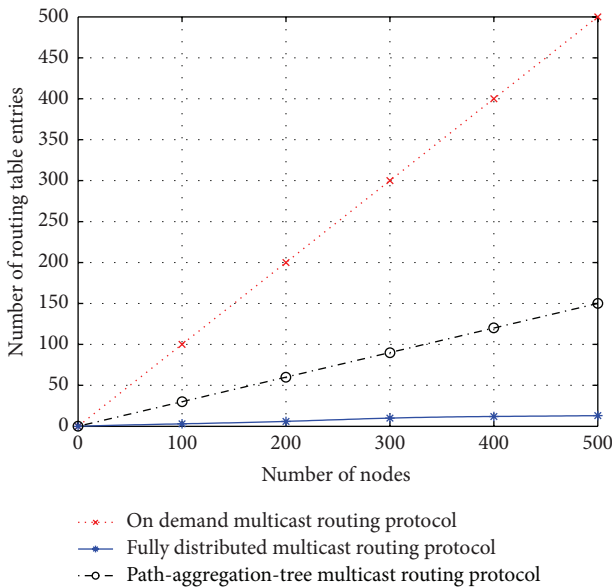


FIGURE 11: # of routing entries per node versus # of nodes.

the other hand, in the FDMRP, ACF flooding does not occur because when a node replies with ARCF, the ACF does not flood because a node sending the ACF is already joined to the same group as the node sending the ARCF.

Figure 11 shows the number of routing table entries versus the number of nodes. From the figure, we observe that the number of routing table entries of the FDMRP is lower than other schemes. This is because both of the schemes store an amount of information for neighbor nodes and routing information for nodes within a multicast group. Particularly, the number of entries significantly increases as the number of nodes increases for ODMRP scheme because all of nodes maintain routing entries for all of destinations. However,

the FDMRP stores minimum information for routing. That means the FDMRP can perform multicast routing with few routing table entries than the other schemes.

Figure 12 shows the time of joining latency versus the number of nodes. In the figure, joining latency of the FDMRP is significantly lower than the other schemes. Since a node which uses the other schemes has to communicate with each receiver in a multicast group, the joining latency is higher in a large network. Particularly, if a receiver is far away from the sender node, joining latency also increases. However, in the FDMRP, a node can join a multicast group when there is at least one peer node in proximity to the node.

Figure 13 shows the areal sum goodput versus the number of nodes. The areal sum goodput is calculated by throughput of all nodes and area of topology. In the figure, goodput of the FDMRP is lower than the other schemes when a low number of nodes are located. Otherwise, goodput of the FDMRP is higher than the other schemes. This is because the other schemes have control messages with broadcasting while the FDMRP has multicasted one. Although the FDMRP performs more forwarding than the other schemes, the control packet overhead is another serious factor in throughput. The result shows that the FDMRP is suitable with large scale network.

In these figures, we observed that a control overhead, routing entries per node, joining latency of the FDMRP are extremely lower than the existing schemes. Although the number of routing tables is less than the existing scheme, the FDMRP shows significant throughput in the large scale network. If the network scale becomes larger, the difference of performance also becomes larger.

6. Conclusion

In this paper, we addressed the problem of coordination of a multicast group. Also we addressed the number of routing

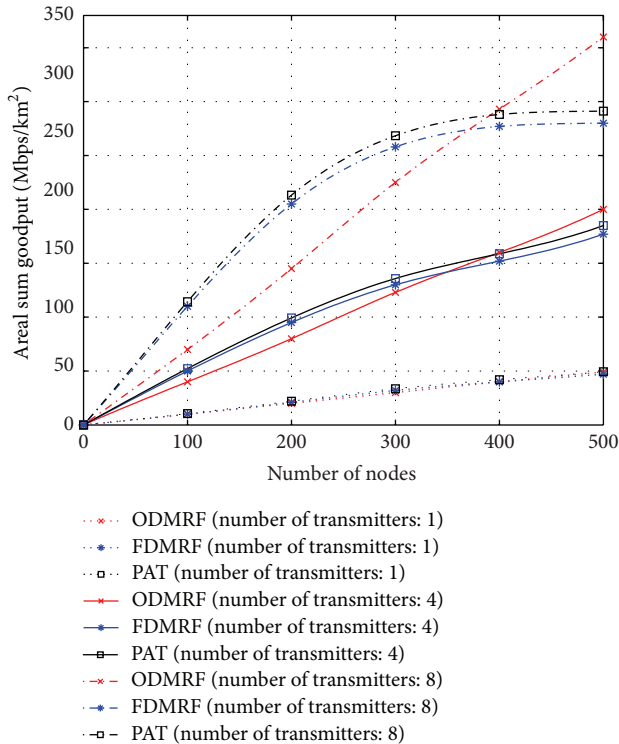


FIGURE 13: Areal sum goodput versus # of nodes.

table entries and time cost for a joining procedure. Existing MANET multicast routing protocols cannot properly resolve the above problems and do not cope with environments which numerous nodes. To solve these problems, we proposed the fully distributed multicast routing protocol for PAC networks. FDMRF exploits the group joining scheme to minimize routing table entries. To minimize routing table entries and reduce join latency, replies to the join message are sent by only one node in each group. To reduce control overhead, group management messages are transmitted to nodes in the routing table. The simulation result shows that our protocol is very effective for reducing control overhead and provides minimized routing table entries. Furthermore, our protocol provides reduced group joining latency compared with existing protocols. As future work, some problems such as reliability in high density environments should be investigated. Enhanced reliability ought to be considered to improve routing efficiency.

Disclosure

Part of this work was presented at ICOIN 2014.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research was supported by the Chung-Ang University Excellent Student Scholarship and National Research

Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013RIA1A2012443).

References

- [1] AT&T Launches Pilot Wi-Fi Project in Times Square, 2010, <http://www.att.com/gen/press-room?pid=17961&cdvn=news&newsarticleid=30838>.
- [2] IEEE Standard for Local and metropolitan area networks Part 15.8: Peer-Aware Communications (PAC), <http://www.ieee802.org/15/pub/TG8.html>.
- [3] The PAC application matrix (DCN: IEEE 802.15-12-0684-00-0008), <https://mentor.ieee.org/802.15/dcn/12/15-12-0684-00-0008-the-pacapplication-matrix.xls>.
- [4] M. B. Krishna and M. N. Doja, "Analysis of tree-based multicast routing in wireless sensor networks with varying network metrics," *International Journal of Communication Systems*, vol. 26, no. 10, pp. 1327–1340, 2013.
- [5] C.-Y. Oh, J. H. Park, J. H. Ahn et al., "Tree-based multicast protocol using multi-point relays for mobile ad hoc networks," in *Proceedings of the 3rd International Conference on Ubiquitous and Future Networks (ICUFN '11)*, pp. 174–178, June 2011.
- [6] E. M. Royer and C. E. Perkins, "Multicast operation of the ad-hoc on-demand distance vector routing protocol," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99)*, pp. 207–218, ACM, Seattle, Wash, USA, August 1999.
- [7] M. S. Rahman, S. Mondal, S. K. Ghosh, and M. M. Rahman, "A new approach of extendable multicast routing protocol in mobile ad hoc networks," in *Proceedings of the 13th International Conference on Computer and Information Technology (ICCIT '10)*, pp. 120–124, December 2010.
- [8] C. W. Wu and Y. C. Tay, "AMRIS: a multicast protocol for ad hoc wireless networks," in *Proceedings of the IEEE Military Communications Conference (MILCOM '99)*, pp. 25–29, November 1999.
- [9] J. Xie, R. R. Talpade, A. Mcauley, and M. Liu, "AMRoute Ad hoc multicast routing protocol," *Mobile Networks and Applications*, vol. 7, no. 6, pp. 429–439, 2002.
- [10] M. Yigit, Ö. D. Incel, and V. Ç. Güngör, "On the interdependency between multi-channel scheduling and tree-based routing for WSNs in smart grid environments," *Computer Networks*, vol. 65, pp. 1–20, 2014.
- [11] B. An and S. Papavassiliou, "A mobility-based hybrid multicast routing in mobile ad-hoc wireless networks," in *Proceedings of the IEEE Military Communications Conference (MILCOM '01)*, pp. 316–320, October 2001.
- [12] E. Astier, A. Hafid, and S. Aljahdali, "An efficient mesh-based multicast routing protocol in mobile ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 9, pp. 822–836, 2012.
- [13] J. J. Garcia-Luna-Aceves and E. L. Madruga, "The core-assisted mesh protocol," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1380–1394, 1999.
- [14] M. Ghasemi and M. Bag-Mohammadi, "Level-based on-demand multicast routing protocol for mobile Ad hoc networks," in *Proceedings of the 6th Wireless and Mobile Networking Conference (WMNC '13)*, pp. 1–7, April 2013.
- [15] N. Kumar, N. Chilamkurti, and J.-H. Lee, "UBMR-CA: utility-based multicast routing and channel assignment with varying traffic demands in multi-radio multi-channel wireless mesh networks," *Mathematical and Computer Modelling*, vol. 57, no. 11–12, pp. 2750–2763, 2013.

- [16] S.-J. Lee, W. Su, and M. Gerla, "On-demand multicast routing protocol in multihop wireless mobile networks," *Mobile Networks and Applications*, vol. 7, no. 6, pp. 441–453, 2002.
- [17] V. Nagar, Y. P. Singh, and D. C. Dhubkarya, "A mesh based multicast routing protocol for network sender," in *Proceedings of the IEEE Recent Advances in Intelligent Computational Systems (RAICS '11)*, pp. 569–574, September 2011.
- [18] L. Canourgues, J. Lephay, L. Soyer, and A.-L. Beylot, "STAMP: shared-tree ad hoc multicast protocol," in *Proceedings of the Military Communications Conference (MILCOM '06)*, pp. 1–7, October 2006.
- [19] J. Biswas, M. Barai, and S. K. Nandy, "Efficient hybrid multicast routing protocol for ad-hoc wireless networks," in *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN '04)*, pp. 180–187, IEEE, November 2004.
- [20] L. Junhai, Y. Danxia, X. Liu, and F. Mingyu, "A survey of multicast routing protocols for mobile Ad-hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 78–91, 2009.
- [21] L. Layuan and L. Chunlin, "A QoS multicast routing protocol for clustering mobile ad hoc networks," *Computer Communications*, vol. 30, no. 7, pp. 1641–1654, 2007.
- [22] W. J. Chang, Y. Y. Hou, and M. C. Chen, "A multicast routing scheme for many-to-many applications in WSNs," in *Proceedings of the IEEE 12th International Conference on Networking, Sensing and Control (ICNSC '15)*, pp. 99–104, April 2015.
- [23] W.-H. Cheng, C.-Y. Wen, and K.-T. Feng, "Power-controlled hybrid multicast routing protocol for mobile ad hoc networks," in *Proceedings of the IEEE 63rd Vehicular Technology Conference (VTC '06)*, pp. 1087–1091, July 2006.
- [24] J. Li, D. Cordes, and J. Zhang, "Power-aware routing protocols in ad hoc wireless networks," *IEEE Wireless Communications*, vol. 12, no. 6, pp. 69–81, 2005.
- [25] Y. C. Wu, P. A. Chou, and S.-Y. Kung, "Minimum-energy multicast in mobile ad hoc networks using network coding," *IEEE Transactions on Communications*, vol. 53, no. 11, pp. 1906–1918, 2005.
- [26] B. Sun and L. Li, "Reliable adaptive multicast protocol in wireless ad hoc networks," *Journal of Systems Engineering and Electronics*, vol. 17, no. 1, pp. 187–192, 2006.
- [27] <http://www.opnet.com>.
- [28] TG8 technical guidance document (DCN: IEEE 802.15-12-0568-07-0008), <https://mentor.ieee.org/802.15/dcn/12/15-12-0568-07-0008-tg8-technical-guidance-document.docx>.