Multicast over Wireless Mobile Ad Hoc Networks: Present and Future Directions

Carlos de Morais Cordeiro, Hrishikesh Gossain, and Dharma P. Agrawal
University of Cincinnati

Abstract

Recent advances in portable computing and wireless technologies are opening up exciting possibilities for the future of wireless mobile networking. A mobile ad hoc network consists of mobile platforms, which are free to move arbitrarily. In a MANET the nodes are mobile, and internode connectivity may change frequently due to mobility of nodes. Recently, there has been an increasing demand for applications like multiplayer online gaming, where players residing at different locations participate in the same gaming session through their handheld portable devices. Such applications are characterized by a close degree of collaboration, typical of ad hoc network scenarios. Multicasting could prove to be an efficient way of providing necessary services for these kinds of applications; hence, it is imperative to determine what is the best way to provide these services in an ad hoc environment. Due to very diverse requirements of the applications and the unpredictable nature of ad hoc networks, it is necessary to investigate and discern the applicability of existing ad hoc multicast protocols and quantify which is more suitable for which type of applications. In this article we provide a detailed description and comparison of ad hoc multicast protocols. We also hope that the discussion presented here will be helpful to application developers in selecting an appropriate multicast protocol and pave the way for further research.

A mobile ad hoc network (MANET) is an autonomous system of mobile hosts (also serving as routers) connected by wireless links, the union of which forms a communication network modeled in the form of an arbitrary communication graph [1]. This is in contrast to the well-known single-hop cellular network model that supports the needs of wireless communication by installing base stations as access points. In cellular networks, communications between two mobile nodes completely rely on the wired backbone and fixed base stations. In a MANET no such infrastructure exists, and the network topology may dynamically change in an unpredictable manner since nodes are free to move.

Multicasting is the transmission of datagrams to a group of hosts identified by a single destination address and hence is intended for group-oriented computing. In MANETs, multicasting can efficiently support a variety of applications that are characterized by close collaborative efforts. Think about the scenario where a user is walking with a handheld device or waiting for a flight in an airport terminal. He/she does not know about his/her neighbor, and switches on the handheld device and tries to scan the network to detect if someone would be interested in playing games or start a similar application of interest. This kind of "community-centric" application is envisioned to draw a lot of attention in the data communication world of the near future. This is a typical ad hoc network application, wherein users are mobile and a community of interest is formed on demand using portable devices. There are many applications of MANETs; email and file transfer can be considered easily deployable within an ad hoc network environment. Web services are also possible if any node in the network can serve as a gateway to the rest of the world. We need not emphasize the wide range of military applications possible with ad hoc networks since the technology was initially developed with them in mind, such as a battlefield in unknown territory wherein an infrastructure network is almost impossible to have or maintain. In such situations, MANETs, having self-organizing capability, can be effectively used where other technologies either fail or cannot be deployed effectively. Advanced features of wireless mobile systems, including data rates compatible with multimedia applications, global roaming capability, and coordination with other network structures, are enabling new applications. Therefore, if we can efficiently combine the features of a MANET with the usefulness of multicasting, it will be possible to realize a number of envisioned group-oriented applications.

With that in mind, this article provides information about the current state of the art in multicast protocols for MANETs, and compares them with respect to several performance metrics. We also attempt to provide insight into future trends in the area and outline the approaches that are likely to play a major role, as well as point out open problems that need careful attention from the research community. It may be noted that there exists a large amount of literature on multicast in

0890-8044/03/$17.00 © 2003 IEEE

wired and infrastructured wireless networks and for detailed insight please refer to [2]. Here, we focus only on multicasting over MANETs.

This article is organized as follows. We initially outline specific characteristics needed to provide multicast in a MANET. Next, we cover the multicast routing protocols in a MANET in detail, classifying them based on the forwarding mechanism. Open problems in the field of multicasting over MANETs that still need careful attention are then described. We conclude this article by providing insight into future directions of this research field.

Issues in Providing Multicast in a MANET

Well established routing protocols do exist to offer efficient multicasting service in conventional wired networks. These protocols, having been designed for fixed networks, may fail to keep up with node movements and frequent topology changes in a MANET. As nodes become increasingly mobile, these protocols need to evolve to provide efficient service in the new environment. Therefore, adopting existing wired multicast protocols as such to a MANET, which completely lacks infrastructure, appears less promising. Host mobility increases the protocol overheads substantially. Rather, new protocols are being proposed and investigated that take issues such as topological changes into consideration. Moreover, the nodes of a MANET rely on batteries; thusm routing protocols must limit the amount of control information passed between nodes.

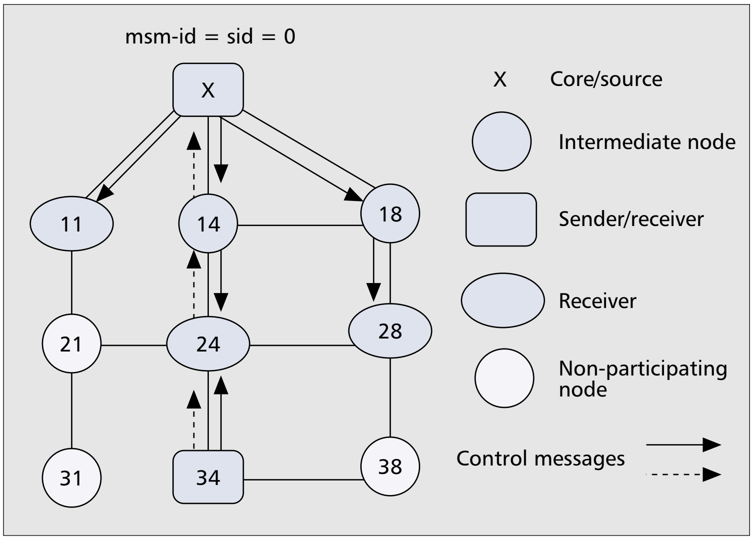
The majority of applications are in areas where rapid deployment and dynamic reconfiguration are necessary and a wireline network is not available. These include military battlefields, emergency search and rescue sites, classrooms, and conventions where participants share information dynamically using their mobile devices. These applications lend themselves well to multicast operation. In addition, within a wireless medium, it is even more crucial to reduce transmission overhead and power consumption. Multicasting can improve the efficiency of the wireless links, when sending multiple copies of messages, by exploiting the inherent broadcast property of the wireless medium when multiple mobile nodes are located within the transmission range of a node. However, besides the issues for any ad hoc routing protocol listed above, wireless mobile multicasting faces several key challenges. Multicast group members can move, thus precluding the use of a fixed multicast topology. Transient loops may form during reconfiguration of distribution structure (e.g., tree) as a result of the mobility. Therefore, the reconfiguration scheme should be kept simple to maintain the channel overhead low.

As we can see, providing efficient multicasting over MANET faces many challenges, including dynamic group membership and constant update of delivery path due to node movement. In the next sections we cover the major protocols proposed so far and compare them under several criteria.

Multicast Routing Protocols

One straightforward way to provide multicast in a MANET is through flooding. With this approach, data packets are sent throughout the MANET, and every node that receives this packet broadcasts it to all its immediate neighbor nodes exactly once. It is suggested that in a highly mobile ad hoc network, flooding of the whole network may be a viable alternative for reliable multicast. However, this approach has considerable overhead since a number of duplicated packets are sent and packet collisions do occur in a multiple-access-based MANET. In this section we discuss multicast routing protocols proposed for MANETs. For simplicity, we can classify these into four categories based on how routes are created to the members of the group:

- Tree-based approaches
- Meshed-based approaches
- Stateless multicast
- Hybrid approaches



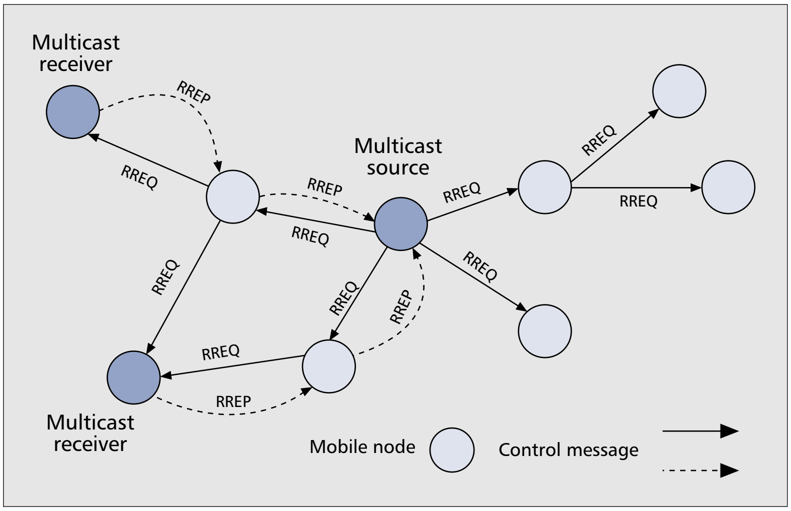
■ Figure 1. *AMRIS packet forwarding (X and 34 are sources; 11, 24, and 28 are recipients).*

Tree-Based Approaches

Tree-based multicast is a very well established concept in wired networks. Most schemes for providing multicast in wired networks are either source- or shared-tree-based. Different researchers have tried to extend the tree-based approach to provide multicast in a MANET environment. This section gives an overview of some of those approaches.

Ad Hoc Multicast Routing Protocol Utilizing Increasing ID Numbers (AMRIS) — AMRIS [3] is an on-demand protocol that constructs a shared multicast delivery tree (Figure 1) to support multiple senders and receivers in a multicast session. AMRIS dynamically assigns an ID number to each node in each multicast session. Based on the ID number, a multicast delivery tree — rooted at a special node with Smallest-ID (Sid) — is created, and the ID number increases as the tree expands from the Sid. Generally, Sid is the source or the node that initiates a multicast session.

The first step in AMRIS protocol operation is the selection of Sid. If there is only one sender for a group, the Sid is generally the source of the group. In case of multiple senders, a Sid is selected among the given set of senders. Once a Sid is identified, it sends a NEW-SESSION message to its neighbors. The contents of this message includes Sid's multicast session member ID (msm-id) and the routing metrics. Nodes receiving the NEW-SESSION message generate their own msm-ids, which are larger than the msm-id of the sender. If a node receives multiple NEW-SESSION messages from different nodes, it keeps the message with the best routing metrics and calculates its msm-ids. To join an ongoing session, a node checks the NEW-SESSION message, determines a parent with the smallest msm-id, and unicasts a JOIN-REQ to its potential parent node. If the parent node is already in the multicast delivery tree, it replies with a JOIN-ACK. Otherwise, the parent itself tries to join the multicast tree by sending a JOIN-REQ to its



■ Figure 2. *Route discovery in the MAODV protocol.*

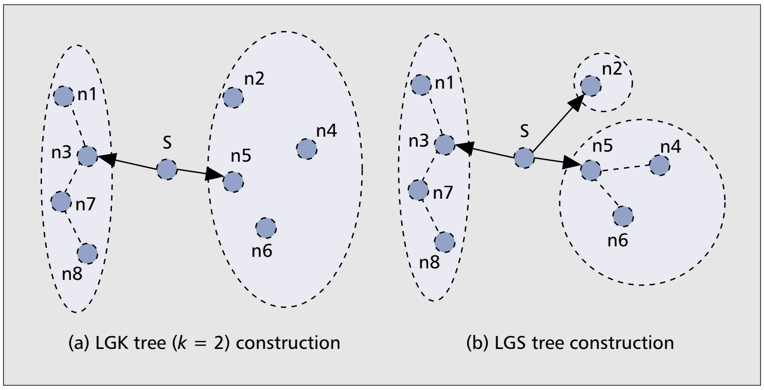
parent. If a node is unable to find any potential parent node, it executes a branch reconstruction (BR) process to rejoin the tree. BR consists of two subroutines. BR1 is executed when a node has a potential parent node for a group (as discussed above). If it does not find any potential parent node, BR2 is executed. In BR2, instead of sending a unicast JOIN_REQ to a potential parent node, the node broadcasts a JOIN-REQ that consists of a range field *R* to specify the nodes till *R* hops. Upon link breakage, the node with the larger msm-id tries to rejoin the tree by executing any of the BR mechanisms. It is to be noted that AMRIS detects link disconnection by a beaconing mechanism. Hence, until the tree is reconstructed, packets could possibly be dropped.

Multicast Ad Hoc On-Demand Distance Vector (MAODV) Protocol — MAODV routing protocol [4] follows directly from unicast AODV, and discovers multicast routes on demand using a broadcast route discovery mechanism employing the same route request (RREQ) and route reply (RREP) messages that exist in the unicast AODV protocol. A mobile node originates an RREQ message when it wishes to join a multicast group, or has data to send to a multicast group but does not have a route to that group. Only a member of the desired multicast group may respond to a join RREQ. If the RREQ is not a join request, any node with a fresh enough route (based on group sequence number) to the multicast group may respond. If an intermediate node receives a join RREQ for a multicast group of which it is not a member, or it receives a RREQ and does not have a route to that group, it rebroadcasts the RREQ to its neighbors.

As the RREQ is broadcast across the network, nodes set up pointers to establish the reverse route in their route tables. A node receiving an RREQ first updates its route table to record the sequence number and the next hop information for the source node. This reverse route entry may later be used to relay a response back to the source. For join RREQs, an additional entry is added to the multicast route table and is not activated unless the route is selected to be part of the multicast tree. If a node receives a join RREQ for a multicast group, it may reply if it is a member of the multicast group's tree and its recorded sequence number for the multicast group is at least as great as that contained in the RREQ. The responding node updates its route and multicast route tables by placing the requesting node's next hop information in the tables, and then unicasts an RREP back to the source. As nodes along the path to the source receive the RREP, they add both a route table and a multicast route table entry for the node from which they received the RREP, thereby creating the forward path (Fig. 2).

When a source node broadcasts an RREQ for a multicast group, it often receives more than one reply. The source node keeps the received route with the greatest sequence number and shortest hop count to the nearest member of the multicast tree for a specified period of time, and disregards other routes. At the end of this period, it enables the selected next hop in its multicast route table, and unicasts an activation message (MACT) to this selected next hop. The next hop, on receiving this message, enables the entry for the source node in its multicast routing table. If this node is a member of the multicast tree, it does not propagate the message any further. However, if this node is not a member of the multicast tree, it would have received one or more RREPs from its neighbors. It keeps the best next hop for its route to the multicast group, unicasts MACT to that next hop, and enables the corresponding entry in its multicast route table. This process continues until the node that originated the chosen RREP (member of tree) is reached. The activation message ensures that the multicast tree does not have multiple paths to any tree node. Note that the nodes forward data packets only along activated routes.

The first member of the multicast group becomes the leader for that group, which also becomes responsible for maintaining the multicast group sequence number and broadcasting this number to the multicast group. This update is done through a Group Hello message. The Group Hello contains extensions that indicate the multicast group IP address and sequence numbers (incremented every Group Hello) of all multicast groups for which the node is the group leader. Since AODV keeps "hard-state" in its routing table, the protocol has to actively track and react to changes in this tree. If a member terminates its membership with the group, the multicast tree requires pruning. Links in the tree are monitored to detect link breakages, and the node that is farther from the multicast group leader (downstream of the break) takes the responsibility to repair the broken link. If the tree cannot be reconnected, a new leader for the disconnected downstream node is chosen as follows. If the node that initiated the route rebuilding is a multicast group member, it becomes the new multicast group leader. On the other hand, if it was not a group member and has only one next hop for the tree, it prunes itself from the tree by sending its next hop a prune message. This continues until a group member is reached.



■ Figure 3. *Location guided tree construction algorithms.*

Once separate partitions reconnect, a node eventually receives a Group Hello for the multicast group that contains group leader information different from the information it already has. If this node is a member of the multicast group, and if it is a member of the partition whose group leader has the lower IP address, it can initiate reconnection of the multicast tree.

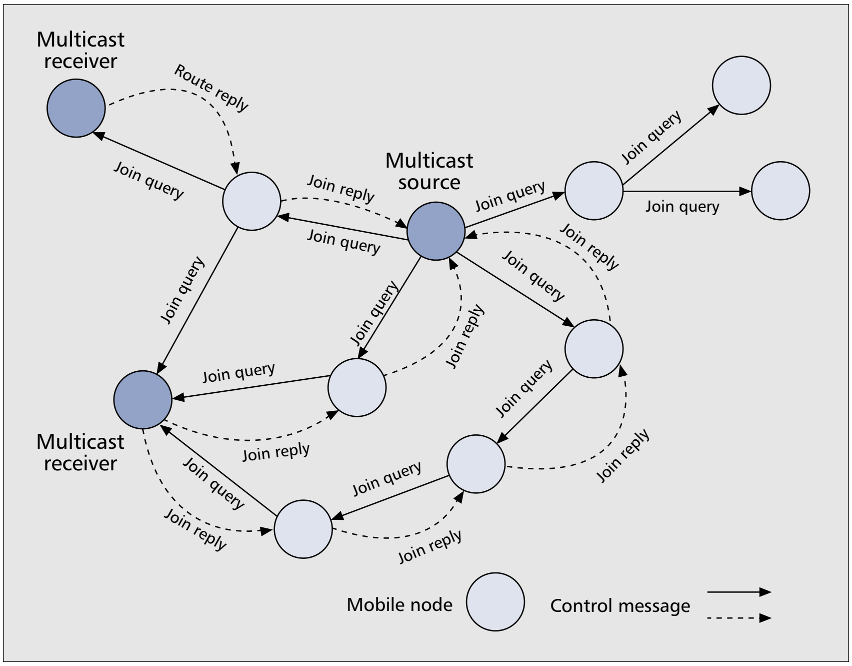
Lightweight Adaptive Multicast — The Lightweight Adaptive Multicast (LAM) [5] protocol draws on the Core-Based Tree (CBT) algorithm[2] and Temporal Ordering Routing Algorithm (TORA) in order to provide multicast services. Similar to CBT, it builds a group-shared multicast routing tree for each multicast group centered at the CORE. Nodes in LAM maintain two variables, POTENTIAL-PARENT and PARENT, and two lists, POTENTIAL-CHILD-LIST and CHILD-LIST. The PARENT variable is used to remember the parent node in the multicast tree. The CHILD-LIST stores identities of one-hop children in the multicasting tree. These potential data objects are used when the node is in a "join" or "rejoin" waiting state.

Since LAM is based on the CBT approach to building the multicast delivery tree, with one CORE to a group, LAM is not very robust, especially in a MANET environment. To address the problem posed by having a single centralized core, Intercore LAM (IC-LAM) is proposed. IC-LAM is a tunnel-based protocol connecting multiple cores. By allowing multiple cores, IC-LAM avoids total group failure due to a single core failure.

Location Guided Tree Construction Algorithm for Small Group Multicast — Location Guided Tree (LGT) [6] is a small group multicast scheme based on packet encapsulation. It builds an overlay multicast packet distribution tree on top of the underlying unicast routing protocol. Multicast data is encapsulated in a unicast packet and transmitted only among the group nodes. It is based on the construction of two types of tree, location-guided k-array (LGK) and location-guided Steiner (LGS). The geometric location information of the destination nodes is utilized to construct the packet distribution tree without knowing the global topology of the network (Fig. 3). It is assumed that the longer the geometric distance is, the longer the network-level hops to reach the destination will be. Therefore, the algorithm attempts to construct a tree with geometrically shorter tree edges. The protocol also supports an optimization mechanism through route caching, wherein a node can cache the computed route and reuse it the next time a new packet comes in with the same set of destinations.

In the LGK tree approach, the sender first selects the nearest k destinations as children nodes. The sender then groups the rest of the nodes to its k children according to close geometric proximity. Once the group nodes are mapped to its corresponding child nodes, the sender forwards a copy of the encapsulated packet to each of the k children with its corresponding subtree (subdestination list of group members) as destinations. The process stops when an incoming packet has an empty destination list. In the LGS scheme, based on the geometric distance as a measurement of closeness, a Steiner tree is constructed that uses the multicast group members as tree nodes.

The protocol uses a hybrid mechanism for location/membership update, which includes in-band and periodic update. In in-band update, a node always includes its geometric location if it has any data packets to send. If a node has no data packet to send for an extended period of time, it sends a periodic update wherein a null packet is sent with its present geometric location.



■ Figure 4. *Mesh creation in ODMRP.*

Mesh-Based Approaches

In contrast to a tree-based approach, mesh-based multicast protocols may have multiple paths between any source and receiver pair. Existing studies show that tree-based protocols are not necessarily best suited for multicast in a MANET where network topology changes frequently. In such an environment, mesh-based protocols seem to outperform tree-based proposals due to the availability of alternative paths, which allow multicast datagrams to be delivered to the receivers even if links fail. This section gives an overview of some of the mesh-based approaches to provide multicast in a MANET.

On-Demand Multicast Routing Protocol (ODMRP) — ODMRP [7] is a mesh-based protocol that uses a forwarding group concept (only a subset of nodes forwards the multicast packets). A soft state approach is taken in ODMRP to maintain multicast group members. No explicit control message is required to leave the group. In ODMRP, group membership and multicast routes are established and updated by the source on demand. When a multicast source has packets to send, but does not have any route to the multicast group, it broadcasts a Join-Query control packet to the entire network. This Join-Query packet is periodically broadcast to refresh the membership information and updates routes as depicted in Fig. 4. When an intermediate node receives a Join-Query packet, it stores the source ID and the sequence number in its message cache to detect any potential duplicate. The routing table is updated with an appropriate node ID (i.e., backward learning) from which the message has been received. If the message is not a duplicate and the TTL is greater than zero, it is rebroadcast.

When a Join-Query packet reaches a multicast receiver, it creates and broadcasts a Join-Reply to its neighbors. When a node receives a Join-Reply, it checks if the next hop node ID of one of the entries matches its own ID. If it does, the node realizes that it is on the path to the source and thus part of the forwarding group, and sets the forwarding group flag (FG_FLAG). It then broadcasts its own Join-Reply built on matched entries. The next hop node ID field contains the information extracted from its routing table. In this way, each forward group member propagates the Join-Reply until it reaches the multicast source via the selected (shortest) path. This whole process constructs (or updates) the routes from sources to receivers and builds a mesh of nodes. After establishing a forwarding group and route construction process, a source can multicast packets to receivers via selected routes and forwarding groups. While a node has data to send, the source periodically sends Join-Query packets to refresh the forwarding group and the routes. When receiving the multicast data packet, a node forwards it only when it is not a duplicate and setting of the FG_FLAG for the multicast group has not expired. This procedure minimizes the traffic overhead and prevents sending packets through stale routes.

In ODMRP, no explicit control packets need to be sent to join or leave the group. If a multicast source wants to leave the group, it simply stops sending Join-Query packets since it does not have any multicast data to send to the group. If a receiver no longer wants to receive from a particular multicast group, it does not send the Join-Reply for that group. Nodes in the forwarding group are demoted to nonforwarding nodes if not refreshed (no Join-Replies received) before they timeout.

Core-Assisted Mesh Protocol — The Core-Assisted Mesh Protocol (CAMP) [8] supports multicasting by creating a shared mesh for each multicast group. Meshes thus created help maintain connectivity to multicast users, even with node mobility. It borrows concepts from CBT, but unlike CBT where all traffic flows through the core node, the core nodes in CAMP are used to limit the control traffic needed for receivers to join multicast groups. The basic operation of CAMP includes building and maintaining the multicast mesh for a multicast group. It assumes a mapping service, which provides routers with the addresses of groups identified by their names. It also implies availability of routing information from a unicast routing protocol. Each router maintains a routing table (RT) built with the unicast routing protocol. This table is modified by CAMP when a multicast group needs to be inserted or removed. Based on RT, a multicast routing table (MRT) is built, which consists of a set of groups known to the router. A router may update its MRT based on topological changes or messages received from its neighbors.

CAMP classifies the nodes in the network in three modes: simplex, duplex, and nonmember. A router joins a group in simplex mode if it intends only to send traffic received from specific nodes or neighbors to the rest of the group, and does not intend to forward packets from the group. A duplex member forwards any multicast packets for the group, whereas nonmember nodes need not be in the multicast delivery mesh. CAMP uses a receiver-initiated method for routers to join a multicast group. If a router wishing to join a group has multiple neighbors that are duplex members of the multicast group, it simply changes its MRT and directly announces to its neighbors that it is a new member of the multicast group using multicast routing update. If it has no neighbors that are members of the multicast group, it either propagates a join request to one of the multicast group "cores" or attempts to reach a member through expanding its ring search. Any router that is a regular member of the multicast group and has received the join request is free to transmit a join acknowledgment (ACK) to the sending router. A router can leave a group if it has no hosts that are members of the group, and also has no neighbors for whom it is an anchor; that is, as long as it is not needed to provide efficient paths for the dissemination of packets in the multicast meshes for the groups. Cores are also allowed to leave multicast group if there are no routers using them as anchors.

CAMP ensures that the mesh contains all reverse shortest paths between a source and the recipients. A receiver node periodically reviews its packet cache in order to determine whether it is receiving data packets from neighbors, which are on the reverse shortest path to the source. Otherwise, a HEARTBEAT message is sent to the successor in the reverse shortest path to the source. This HEARTBEAT message triggers a PUSH JOIN (PJ) message. If the successor is not a mesh member, the PJ forces the specific successor and all the routers in the path to join the mesh. CAMP has the advantage that it does not use flooding, and the requests only propagate to mesh members. On the other hand, CAMP relies on an underlying unicast routing protocol to guarantee correct distances to all destinations within a finite time.

Forwarding Group Multicast Protocol (FGMP) — FGMP [9] can be viewed as flooding with "limited scope," wherein the flooding is contained within selected forwarding group (FG) nodes. FGMP makes innovative use of flags and an associated timer to forward multicast packets. When the forwarding flag is set, each node in FG forwards data packets belonging to a group *G* until the timer expires. When a packet is forwarded, only the nodes with an enabled forwarding flag can accept the packet. This soft state approach of using a timer works well in dynamically changing environments. FGMP uses two approaches to elect and maintain FG of forwarding nodes: FGMP-RA (receiver advertising) and FGMP-SA (sender advertising).

In FGMP-RA, multicast receivers periodically announce their group membership by flooding. Senders maintain a table with all receivers of the group. In FGMP-SA, a sender periodically announces its presence in the network by flooding. The nodes that relay this message store the next hop to the sender. Multicast receivers join the group by sending replies to the sender. FGMP can be seen as a twin method to ODMRP, where their main difference lies in the way group meshes are established. However, both FGMP and ODMRP have scalability problems due to flooding of control packets.

Stateless Multicast

Tree- and mesh-based approaches have an overhead of creating and maintaining the delivery tree/mesh with time. In a MANET environment, frequent movement of mobile nodes considerably increases the overhead in maintaining the delivery tree/mesh. To minimize the effect of such a problem, stateless multicast is proposed wherein a source explicitly mentions the list of destinations in the packet header. Stateless multicast focuses on small group multicast and assumes the underlying routing protocol to take care of forwarding the packet to respective destinations based on the addresses contained in the header.

Differential Destination Multicast — The Differential Destination Multicast (DDM) protocol [10] is meant for small multicast groups operating in dynamic networks of any size. Unlike other MANET routing protocols, DDM lets the source control multicast group membership. The source encodes multicast receiver addresses in multicast data packets using a special DDM data header. This variable length destination list is placed in the packet headers, resulting in packets being self-routed toward their destinations using the underlying unicast routing protocol. It eliminates maintaining per-session multicast forwarding states at intermediate nodes and thus is easily scalable with respect to the number of sessions.

DDM supports two kinds of operation modes: stateless and soft state. In stateless mode, the nodes along the data forwarding paths need not maintain multicast forwarding states. An intermediate node receiving a DDM packet only needs to look at the header to decide how to forward the packet. In soft state mode, based on in-band routing information, each node along the forwarding path remembers the destinations to which the packet was forwarded last time and its next hop information. By caching this routing information at each node, the protocol does not need to list the entire destination in future data packets. If changes occur in the underlying unicast routing, an upstream node only needs to inform its downstream nodes about the differences in the destination forwarding since the last packet; hence the name Differential Destination Multicast.

At each node, there is one forwarding set (FS) for each multicast session that records to which destinations this node forwards the data. The nodes also maintain a direction set (DS) to record the particular next hop to which multicast destination data are forwarded. At the source node, the FS contains the same set of nodes as the multicast member list (ML). In the intermediate nodes, the FS is the union of several subsets based on the data stream received from upstream neighbors. Associated with each set FS_*k*, there is a sequence number SEQ(FS_*k*) used to record the last DDM block sequence number seen in a received DDM data packet from an upstream neighbor *k*. It helps to detect loss of a data packet containing forwarding set updates. At a given node, the FS also needs to be partitioned into subsets according to the next hops for different destinations.

DDM supports two types of packets: control and data packets, where the data packets may also contain control information. There are five types of control packets: JOIN, ACK, LEAVE, RSYNC, and CTRL_DATA. To join a multicast session, a receiver needs to unicast a JOIN message to the source for that session. The source updates its ML and replies with an ACK. In DDM, membership refreshing is source-initiated. After a specified period of time, the source sets a POLL flag in the next outgoing data packet. Multicast members need to unicast a JOIN message again to the source to express their continued interest. A member can also leave the session by sending an explicit LEAVE message. CTRL_DATA is used to encapsulate multicast data to send it to a particular destination by using unicasting, while an RSYNC message is used to synchronize the multicast destination address sets between a pair of neighboring nodes whenever the topology changes.

It is important to discuss the differences between LGT and DDM since both are meant to provide small group multicast. In DDM, the packet distribution tree is uncontrollable by upper transport and application layers, whereas in LGT, the packet distribution tree is constructed explicitly with the flexibility of adding upper layer packet processing and routing. Additionally, DDM requires every node in the network to eventually participate in packet forwarding, while in LGT, only the nodes participating in the session need to cooperate.

Hybrid Approaches

The protocols to provide multicast in MANETs discussed so far address either efficiency or robustness but not both simultaneously. The tree-based approaches provide high data forwarding efficiency at the expense of low robustness, whereas mesh-based approaches provide better robustness (link failure may not trigger a reconfiguration) at the expense of higher forwarding overhead and increased network load. Thus, there is a possibility that a hybrid multicasting solution may achieve better performance by combining the advantages of both tree and meshed-based approaches. In this section we discuss some of the different hybrid approaches to provide ad hoc multicasting.

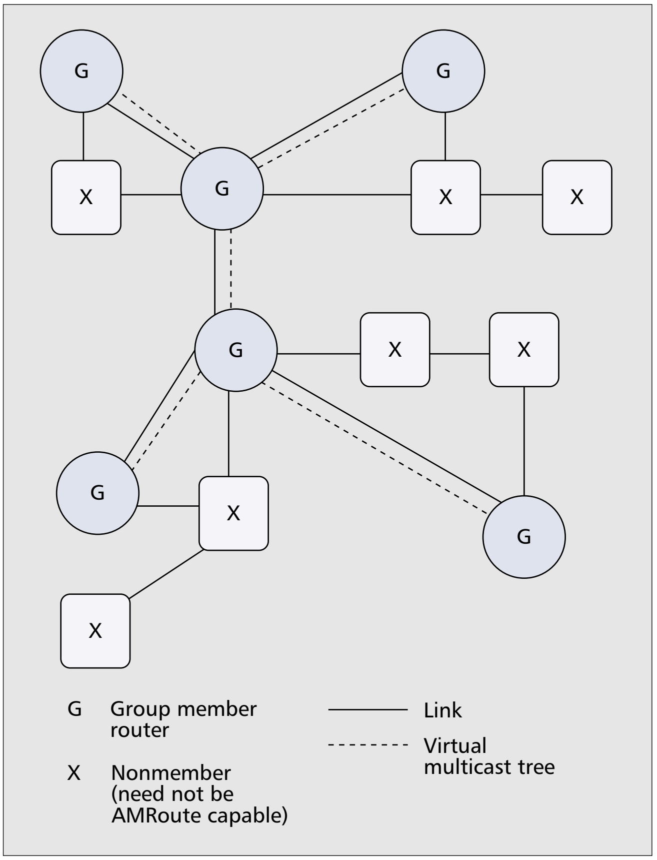


Figure 5. *AMRoute virtual multicast tree.*

Ad Hoc Multicast Routing Protocol — The Ad Hoc Multicast Routing Protocol (AMRoute) [11] creates a bidirectional, shared tree using only group senders and receivers as tree nodes for data distribution. The protocol has two main components: mesh creation and tree setup (Fig. 5). The mesh creation identifies and designates certain nodes as logical cores, and these are responsible for initiating the signaling operation and maintaining the multicast tree to the rest of the group members. A non-core node only responds to messages from the core nodes and serves as a passive agent. The selection of a logical core in AMRoute is dynamic and can migrate to any other member node depending on network dynamics and group membership. AMRoute does not address network dynamics and assumes the underlying unicast protocol to take care of it. To create a mesh, each member begins by identifying itself as a core and broadcasts JOIN_REQ packets with increasing time to live (TTL) to discover other members. When a core receives a JOIN_REQ from a core in a different mesh for the same group, it replies with a JOIN_ACK. A new bidirectional tunnel is created between the two cores, and one of them is selected as a core after the mesh merger. Once the mesh has been established, the core initiates the tree creation process. The core sends out periodic TREE_CREATE messages along all links incident on its mesh. Using unicast tunnels, the TREE_CREATE messages are sent only to group members. Group members receiving a nonduplicate TREE_CREATE message forwards it to all mesh links except the incoming one, and marks the incoming and outgoing links

Protocols	Multicast topology	Loop free	Dependence on unicast protocol	Periodic message	Control packet flooding done/ required
Flooding	Mesh	Yes	No	No	No
AMRoute	Hybrid	No	Yes	Yes	Yes
AMRIS	Tree	Yes	No	Yes	Yes
MAODV	Tree	Yes	Yes	Yes	Yes
LAM	Tree	Yes	Yes	No	No
LGT-based	Tree	Yes	No	Yes	No
ODMRP	Mesh	Yes	No	Yes	Yes
CAMP	Mesh	Yes	Yes	Yes	No
DDM	Stateless tree	Yes	No	Yes	No
FGMP-RA	Mesh	Yes	Yes	Yes	Yes
FGMP-SA	Mesh	Yes	No	Yes	Yes
MCEDAR	Hybrid	Yes	Yes	Yes	Yes

■ Table 1. *Comparison of ad hoc multicast routing protocols.*

as tree links. If a link is not going to be used as part of the tree, the TREE_CREATE is discarded and TREE_CREATE_NAK is sent back to incoming links. A member node that wants to leave a group can do so by sending a JOIN_NAK message to its neighboring nodes.

AMRoute employs the virtual mesh links to establish the multicast tree, which helps keep the multicast delivery tree unchanged with changes of network topology, as long as routes between core nodes and tree members exist via mesh links. The main disadvantage of this protocol is that it may have temporary loops and may create nonoptimal trees with host mobility.

Multicast Core Extraction Distributed Ad Hoc Routing (MCEDAR) — MCEDAR [12] is a multicast extension to the CEDAR architecture. The main idea of MCEDAR is to incorporate the efficiency of tree-based forwarding protocols and robustness of mesh-based protocols by combining the two approaches. It decouples the control infrastructure from the actual data forwarding. It uses a mesh as the underlying infrastructure, so it can tolerate a few link breakages without reconfiguration of the infrastructure. The efficiency is achieved by using a forwarding mechanism on the mesh that creates an implicit route-based forwarding tree. This ensures that the packets need to travel only the minimum distance in the tree.

Comparison of Protocols and Open Problems

The basic idea behind defining multicast routing protocols for MANETs is to form a path to group members with minimal redundancy, and various algorithms described earlier attempt to achieve this goal using different mechanisms. The host mobility also influences the routes selected, and the possibility of loop formation or the paths becoming nonoptimal are all important considerations. It is also critical to know if the paths created are on demand, or if optimal paths are determined once and updated periodically as needed. Another important issue is if control packets are flooded throughout the network or limited to the nodes in the multicast delivery tree. Keeping this in mind, Table 1 compares the prominent proposals to provide multicasting over MANETs under several metrics. A performance study of various multicast routing protocols can be found in [13].

As mentioned earlier, research in the area of multicast over MANETs is far from exhaustive. Much of the effort so far has been on devising routing protocols to support effective and efficient communication between nodes that are part of a multicast group. However, there are still many topics that deserve further investigation, such as:

Scalability: This issue is not only related to multicast in MANETs but also with the MANET itself. An obvious question comes to mind: to what extent can an ad hoc network grow? Can we design a multicast routing protocol for MANETs that is scalable with respect to number of members in the group, their mobility, and other constraints posed by the MANET environment itself?

Address configuration: Due to the infrastructureless nature of MANETs, a different addressing approach may be required. Special care must be taken so that other groups do not reuse a multicast address used by a group at the same time. Node movement and network partitioning makes this task of synchronizing multicast addresses in a MANET really difficult.

Quality of service (QoS): Is it feasible for bandwidth/delay-constrained multicast applications to run well in a MANET? Since MANET itself does not have a well defined framework for QoS support yet, it may be difficult to address this task for some time.

Applications for multicast over MANETs: Have we found a killer application? Although we talk about online gaming and military applications, what are the potential commercial applications of the MANETs; and whether service providers can be convinced to support multicast is still an open issue.

Security: How can the network secure itself from malicious or compromised hosts? Due to the broadcast nature of the wireless medium security provisioning becomes more difficult. Further research is needed to investigate how to stop an intruder from joining an ongoing multicast session or stop a node from receiving packets from other sessions.

Power control: How can battery life be maximized? Both source and core-based approaches concentrate traffic on a single node. For example, in stateless multicast group membership is controlled by the source, which limits the lifetime of its battery. Still to be investigated is how to efficiently distribute traffic from a central node to other member nodes in a MANET.

Conclusions and Future Directions

Within the wired network, well established routing protocols exist to offer efficient multicasting service. Adopting wired multicast protocols to a MANET, which completely lack any infrastructure, appears less promising. These protocols, having been designed for fixed networks, may fail to keep up with node movements and frequent topology changes due to host mobility, and the protocol overheads may increase substantially. Rather, new protocols that operate in an on-demand manner are being proposed and investigated. Existing studies show that tree-based on-demand protocols are not necessarily the best choice for all applications. In a harsh environment, where the network topology changes very frequently, mesh-based protocols seem to outperform tree-based protocols due to the availability of alternate paths, which allow multicast datagrams to be delivered to all or most multicast receivers even if some links fail. Between tree-based and mesh-based approaches, we can find hybrid protocols suitable for medium mobility networks taking advantage of both a tree and a mesh structure. Finally, stateless multicast seems promising for supporting multiple small groups.

In summary, multicasting can efficiently support a wide variety of applications that are characterized by a close degree of collaboration, typical for many MANETs. In this article we have provided a thorough description of the current state of the art in multicast over MANETs, and compared them under several criteria. It is our conclusion that the usefulness of different protocols depends on the application environments. Much work has to be done before we have consistent and adequate solutions to multicasting over MANETs. Issues such as QoS, reliability, and power- and location-aware multicasting need to be investigated in much more detail.

Acknowledgment

This work has been supported by the Ohio Board of Regents Doctoral Enhancements Funds and the National Science Foundation under Grant CCR-0013361.

References

[1] D. P. Agrawal and Q-A. Zeng, *Introduction to Wireless and Mobile Systems*, Brooks/Cole, 2003.
[2] H. Gossain, C. M. Cordeiro, and D. P. Agrawal, "Multicast: Wired to Wireless," *IEEE Commun. Mag.*, vol. 40, no. 6, June 2002, pp. 116–23.
[3] C. W. Wu, Y.C. Tay, and C.-K. Toh, "Ad Hoc Multicast Routing Protocol Utilizing Increasing id-numberS (AMRIS) Functional Specification," Internet draft, Nov. 1998.
[4] E. M. Royer and C. E. Perkins, "Multicast Operation of the Ad Hoc On-Demand Distance Vector Routing Protocol," ACM MOBICOM, Aug. 1999, pp. 207–18.
[5] L. Ji and M. Scott Corson, "A Lightweight Adaptive Multicast Algorithm," *GLOBECOM 1998*, pp. 1036–42.
[6] K. Chen and K. Nahrstedt, "Effective Location-Guided Tree Construction Algorithms for Small Group Multicast in MANET," *Proc. INFOCOM*, 2002, pp. 1180–89.
[7] M. Gerla, S.-J. Lee, and W. Su. "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," Internet draft, draft-ietf-manet-odmrp-02.txt, 2000.
[8] J. J. Garcia-Luna-Aceves and E.L. Madruga, "The Core-Assisted Mesh Protocol," *IEEE JSAC*, Aug. 1999, pp. 1380–94.
[9] C.-C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks," *AJ. Cluster Comp*, Special Issue on Mobile Computing, vol. 1, no. 2, 1998, pp. 187–96.
[10] L. Ji, and M. S. Corson, "Differential Destination Multicast — A MANET Multicast Routing Protocol for Small Groups," *Proc. INFOCOM*, 2001, pp. 1192–02.
[11] E. Bommaiah *et al.*, "AMRoute: Adhoc Multicast Routing Protocol," Internet draft, Aug. 1998.
[12] P. Sinha, R. Sivakumar, and V. Bharghavan, "MCEDAR: Multicast Core-Extraction Distributed Ad hoc Routing," *IEEE Wireless Commun. and Net. Conf.*, Sept. 1999, pp. 1313–17.
[13] S.-J. Lee *et al.*, "A Performance Comparison Study of Ad Hoc Wireless Multicast Protocols," *Proc. INFOCOM 2000*, Mar. 2000, pp. 565–74.

Biography

CARLOS DE MORAIS CORDEIRO (cordeicm@ececs.uc.edu) received his B.Sc. and M.Sc. in computer science in 1998 and 2000, respectively, from the Federal University of Pernambuco, Brazil. He is presently pursuing a Ph.D. degree in computer science and engineering at the University of Cincinnati where he is a research assistant in the Center of Distributed and Mobile Computing. His research interests and papers are mostly in the area of wireless and mobile communications including ad hoc networks, multicast, TCP over wireless, and short-range radio communication such as Bluetooth. He has prior work experience with IBM in San Jose, California.

HRISHIKESH GOSSAIN (hgossain@ececs.uc.edu) is a Ph.D. student in the Department of ECECS, University of Cincinnati. He received a B.E. in electronics engineering from Motilal Nehru Regional Engineering College, Allahabad, India in 1998, where he was undergraduate Gold Medalist. He is co-inventor of four patents in QoS and e-media. His research interests include multicast, mobility management in wireless environments, Mobile IP, QoS, and multimedia communications. He previously worked at Nortel Networks, Richardson, Texas.

DHARMA P. AGRAWAL (dpa@ececs.uc.edu) is the Ohio Board of Regents Distinguished Professor of Computer Science and Computer Engineering and the founding director for the OBR Research Center for Distributed and Mobile Computing in the Department of Electrical and Computer Engineering and Computer Science, University of Cincinnati, Ohio. He has been a faculty member at Wayne State University, Detroit, Michigan (1977–1982), and North Carolina State University, Raleigh (1982–1998). His current research interests include wireless and mobile networks, distributed processing, and scheduling techniques. He has authored a textbook, *Introduction to Wireless and Mobile Systems* (Brooks/Cole, 2002). He is an editor for *Journal of Parallel and Distributed Systems* and *International Journal of High Speed Computing*. He has served as an editor of *IEEE Computer* and *IEEE Transactions on Computers*. He has been Program Chair and General Chair for numerous international conferences and meetings.