IETF MANET Working Group
INTERNET-DRAFT

Jorjeta G. Jetcheva
Carnegie Mellon University
David B. Johnson
Rice University
13 July 2001

The Adaptive Demand-Driven Multicast Routing Protocol for Mobile Ad Hoc Networks (ADMR)

<draft-jetcheva-manet-admr-00.txt>

Status of This Memo

This Internet-Draft is a submission to the IETF Mobile Ad Hoc Networks (MANET) Working Group. Comments on this draft may be sent to the Working Group at manet@itd.nrl.navy.mil, or may be sent directly to the authors.

Abstract

The Adaptive Demand-Driven Multicast Routing protocol (ADMR) has been designed specifically for use in the ad hoc network environment. Multicast routing state in ADMR is dynamically established and maintained only for groups that have at least one receiver and one active sender in the network. Each multicast data packet is forwarded along the shortest-delay path with multicast forwarding state, from the sender to the receivers. Senders are not required to announce their intention to start or stop sending data to the group, or to join the group to which they wish to send. Receivers dynamically adapt to the sending pattern of senders and mobility in the network in order to efficiently balance overhead and maintenance of the multicast routing state as nodes in the network move or as wireless transmission conditions in the network change. State for groups whose senders have become inactive or whose receivers have left the group is expired automatically without the need for control signaling or application-level notification at the source. ADMR also detects when mobility in the network is too high to efficiently maintain multicast routing state, and instead reverts to flooding for a short period of time it determines that the high mobility has subsided.

Contents

1. Introduction

The Adaptive Demand-Driven Multicast Routing protocol (ADMR) enables efficient multicast data packet delivery in wireless ad hoc networks. The protocol does not require any existing infrastructure or preconfiguration to operate. It discovers "multi-hop" routes between multicast receivers for a group and senders that have data to send to that group, and maintains connectivity between these senders and receivers in the face of route disconnection caused by effects such as node motion, propagation phenomena, or wireless interference.

ADMR conforms to the standard IP multicast API in which any node can send data to any multicast group without explicitly announcing its intention to send or to stop sending, and any node can join or leave a multicast group at any time. In addition, ADMR supports the source-specific multicast API [3], allowing receivers to join source-specific groups.

The design of ADMR has been guided by the following requirements: low overhead and battery consumption, active link break detection and maintenance, correct and efficient operation in the presence of control packet loss, and adaptiveness to change in network conditions such as mobility or packet loss.

The operation of ADMR is driven by the presence of data packets being sent and by changes in network conditions, rather than by continuous or periodic background activity of the protocol. The protocol tunes its behavior in response to changing mobility in the network without requiring GPS or other external positioning information.

In ADMR, source-based forwarding trees for a group are created whenever there is at least one source and one receiver for the group active in the network. ADMR monitors the traffic pattern of the multicast source application, and based on that can detect link breaks in the tree, as well as sources that have become inactive and will not be sending any more data. In the former case, the protocol initiates local repair procedures, and then global repair if the local repair fails. In the latter case, multicast forwarding state is silently expired without the need to send an explicit shutdown message. To enable monitoring for link breaks in the multicast forwarding tree when the source is not sending data temporarily, ADMR sends a limited number of keep-alives at increasing inter-packet times. To balance the cost of the keep-alives against that of maintaining the multicast routing state, when the source has not sent any data for a period of time that constitutes a significant deviation from its sending pattern, the keep-alives stop and the entire tree silently expires. A significant deviation from a source's sending pattern is an indication that the source is likely to be inactive for a while, in which case it would be wasteful to maintain routing state in the network.

ADMR detects changes in mobility in the network and can adjust the frequency of the keep-alives it sends accordingly. If desired, keep-alives may also optionally be sent in between application data packets in order to speed up detection and repair of link breaks. If mobility in the network becomes too high to allow timely multicast state setup and maintenance, ADMR switches to flooding for some period of time, after which it attempts to operate efficiently again as the mobility in the network may have decreased. Detection of high mobility in the network is based on frequency of link breaks in the multicast forwarding tree and does not require any additional control traffic or GPS or other external positioning information.

2. Assumptions

We assume that all nodes wishing to communicate with other nodes within the ad hoc network are willing to participate fully in the protocols of the network. In particular, each node participating in the network should also be willing to forward packets for other nodes in the network.

We refer to the minimum number of hops necessary for a packet to reach from any node located at one extreme edge of the network to another node located at the opposite extreme, as the diameter of the network. We assume that the diameter of an ad hoc network will be small (e.g., perhaps 5 or 10 hops), but may often be greater than 1.

Packets may be lost or corrupted in transmission on the wireless network. A node receiving a corrupted packet can detect the error and discard the packet.

3. Terminology

3.1. General Terms

link

A communication facility or medium over which nodes can communicate at the link layer, such as an Ethernet (simple or bridged). A link is the layer immediately below IP.

interface

A node's attachment to a link.

prefix

A bit string that consists of some number of initial bits of an address.

link-layer address

A link-layer identifier for an interface, such as IEEE 802 addresses on Ethernet links.

packet

An IP header plus payload.

piggybacking

Including two or more conceptually different types of data in the same packet so that all data elements move through the network together.

network flood

The flood of a packet in which each node in the network forwards the packet if it receives it and has not previously forwarded it.

tree flood

The flood of a packet constrained to the nodes in a multicast forwarding tree. The packet is forwarded by any node in the tree receiving the packet that has not previously forwarded it, and nodes in the tree may accept and forward the packet when received from any other node in the tree.

3.2. Specification Language

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

4. ADMR Protocol Overview

ADMR does not depend on the operation of any particular underlying unicast routing protocol in the ad hoc network, allowing complete flexibility in the set of protocols used. In fact, ADMR can even work in networks with no unicast protocol.

Currently, ADMR operates only over bidirectional links.

4.1. Multicast State Setup

Multicast state is set up when some new multicast sender S starts sending to a group G for which at least one receiver exists in the network, or when a receiver joins a group G for which there is at least one source in the network. Group G may be a source-specific group. State setup following a link break is discussed later in Section 4.3.

4.1.1. Multicast Receiver Discovery

The multicast forwarding state for a given multicast group G and sender S in ADMR is conceptually represented as a loosely-structured multicast forwarding tree rooted at S.

When an application running at source S sends a multicast packet targeted at group G when no routing state yet exists for this sender and group, the routing layer on S adds an ADMR header to the data packet and sends the data packet as a network flood. Each node in the network that receives this packet forwards it unless it has already forwarded a copy of it. In addition, the node records in its Node Table the MAC address of the node from which it received the packet, and the sequence number stored in the packet's ADMR header. This information will be used for duplicate detection and also for forwarding Receiver Join packets back to the source as described below. After forwarding the packet, each node processes the rest of the packet as a normal packet based on its group destination address.

In addition to forwarding and processing the packet, receivers for group G send a Receiver Join packet back towards the source. The Receiver Join is sent automatically along the shortest path traversed by the flood back towards the source. Each node that forwards the Receiver Join creates a forwarding entry in its Membership Table for source S and group G, indicating that it is a forwarder for this sender and group. The collection of paths with forwarding state between S and the receivers for G abstractly constitutes the forwarding tree.

If there are multiple new receivers for a given multicast group G near each other in the network, many Receiver Join packets will traverse the same paths or subpaths on their way to the source S. However, in order to make each node along these paths a forwarder for G and S, as necessary, it is enough for one Receiver Join packet to be received and forwarded by each such node. It would thus be possible to filter all but the first of these multiple Receiver Join packets received by each of these nodes, but doing so would leave the connection of these new receivers to the multicast forwarding tree susceptible to the loss of the single Receiver Join packet that was forwarded. To reduce overhead and yet provide resilience to such packet loss, each node forwards at most RECEIVER_JOIN_COUNT Receiver Join packets for the last sequence number it has recorded in its Node Table entry for G and S. To implement this filtering, when sending a Receiver Join packet, the receiver R copies the sequence number from the received multicast packet from S into its Receiver Join, and each node maintains in its Node Table entry a count of Receiver Join packets forwarded for the sequence number in that Node Table entry.

Once a receiver for group G sends a Receiver Join packet in response to a multicast data flood, it sets a join timer. If this timer expires and the receiver has not received data from the source, it will resend its Receiver Join packet and set the timer again. At the next expiration of the timer, the receiver will flood a Multicast Solicitation (Section 4.1.2) on the assumption that the path the Receiver Join is trying to traverse is no longer connected. The join timer value is set according to a field specified by the source in the ADMR header of the data flood and is computed based on an application-specified or default value of the expected inter-packet time at which the source application will be originating packets.

The source buffers data packets while multicast state is being set up. The source node will start sending packets only after STATE_SETUP time has elapsed and it has received at least one Receiver Join packet. The STATE_SETUP wait time is intended to allow for multicast state to be set up in the network. The source will not send data if there are no receivers for group G in the network as indicated by a lack of Receiver Join packets.

Once the source has received at least one Receiver Join packet and the STATE_SETUP time has elapsed, the source can send the buffered packets for group G; optionally, the source may apply a pacing scheme to avoid sending a large burst of packets at once and creating temporary network congestion along the paths from the source to the receivers.

To deal with partitions, an ADMR source MAY flood (instead of multicasting) a subset of its data packets, selected from the stream of normal data packets generated by the source application. If it does so, the period between such flooded multicast data packets

SHOULD be limited, e.g., on the order of several tens of seconds or more between flooded packets.

4.1.2. Multicast Source Discovery

When an application running at some node R requests to join a group G, the node checks its Membership Table to determine if it is already connected to G. If the table indicates that it is a forwarder for G, it records in the entry for G that it is also a receiver member for the group. If R is neither a forwarder nor a receiver for the group, the ADMR routing layer on R sends a Multicast Solicitation packet as a network flood. The ADMR header MUST include a Multicast Group option which contains the multicast group address that the receiver is attempting to join. If the group is a source-specific multicast group, the specific sender address S requested by the application MUST be included in a Multicast Sender Address option.

Each node in the network forwards the Multicast Solicitation, except that in the case of source-specific multicast, the specified source does not forward this packet. Also in this case, if a node receiving the Multicast Solicitation has a Membership Table entry for this group and source indicating that it is a forwarder, this node will instead unicast (rather than forward as part of the flood) the Multicast Solicitation only to the previous hop address indicated in that Membership Table entry; the packet thus follows the multicast tree towards the source, speeding up and decreasing the overhead of the receiver join.

When any source S for multicast group G receives the Multicast Solicitation packet (or the single source, in the case of a source-specific multicast group join), the source replies to the Multicast Solicitation to advertise to R its existence as a sender for the group. This reply may take one of two forms:

- If the next scheduled network flood of the next multicast data packet from the source application (Section 4.1.1) is to occur soon, S MAY choose to advance the time for this network flood and use this packet as the reply for the Multicast Solicitation from R. This form of reply is appropriate, for example, when many new receivers attempt to join the group at about the same time, since S would then receive a Multicast Solicitation from each of them, but could use the single existing network flood of the next data packet to reply to all of them.

- The other form that this reply may take is for S to send an ADMR keep-alive packet unicast to R, following the path taken by R's Multicast Solicitation packet; each node forwarding this unicast keep-alive packet unicasts it to the address recorded in the previous hop address field of that node's Node Table entry for R, created when it forwarded R's Multicast Solicitation as it traveled toward S. When forwarding this unicast keep-alive packet toward R, each node updates its Node Table entry for S in the same way as it would for a flood from S, recording the path back to S in each entry's previous hop address field.

When node R receives a keep-alive from a source for group G in response to its Multicast Solicitation, R sends a Receiver Join packet back to S in the same manner as described in Section 4.1.1, creating the forwarding state to connect it to the multicast forwarding tree for this group and source.

If node S replies to the Multicast Solicitation from R by sending a unicast keep-alive, as described above, then S also sets a keep-alive retransmission timer and expects to receive the Receiver Join from R within a short time. If S does not receive the Receiver Join, it will retransmit its reply to R's Multicast Solicitation (which again, may be in the form of S's next network flood of an existing multicast data packet or may use a unicast ADMR keep-alive packet). If the timer expires a second time and S has not received a Receiver Join from R, then S assumes that the path that the unicast keep-alive is trying to traverse, created by the forwarding of R's Multicast Solicitation to S, is broken, and S advances its next scheduled network flood of a multicast data packet to reply to R.

A multicast receiver considers itself connected once it receives a data packet that was sent to it via multicast as described in Section 4.2.

4.2. Multicast Packet Forwarding

A node whose Membership Table indicates it is a forwarders for group G and source S forwards non-duplicate multicast packets with a source address of S and destination address of G. Each multicast packet is dynamically forwarded from S along the shortest-delay path through the tree to the receiver members of the multicast group, only by members of the multicast tree.

In this packet forwarding, however, packets are not constrained to follow any particular branches or parent/child links in the tree. In particular, the tree is defined only by the set of nodes, not by particular branches between the nodes; packets being forwarded along the tree may be accepted and forwarded when received from any other node in the tree. Different packets may thus reach a receiver along different paths within the forwarding tree when nodes along these paths acquire the wireless medium in a different order, or when the packet does not get received correctly over some path within the tree due to wireless interference.

We refer to the flood of a packet constrained to the nodes in the multicast forwarding tree as a tree flood, and to the more general type of flood of a packet through all nodes as a network flood.

When a sender using ADMR sends a multicast packet, it floods within the multicast distribution tree only towards the group's receivers, in contrast to other protocols in which the packet also floods back towards any other senders that are not receivers. Although this difference requires maintenance of source-specific state in forwarding nodes, such state is required anyway in order to support the source-specific multicast service model [3]. In addition, source-specific state at each node is required in other protocols, since they must detect duplicate packets during a flood within the forwarding group, and any type of packet identifiers used for this duplicate detection when there may be multiple group senders must be source-specific.

When a node R receives any multicast packet, in addition to forwarding the packet, if it has forwarding state for the group and source of the packet, node R also checks the entry for this sender and group in its Membership Table to determine if it is a receiver member. If so, then R processes it as a multicast packet that it is intended to receive, passing the packet up to the next layer within its receiving protocol stack.

In addition, as part of this processing of the received multicast packet, if the packet was sent as a tree flood (rather than as a network flood), then this indicates that the receiver node R is currently connected to the multicast forwarding tree for this sender and group. The node considers itself to remain connected until detecting that it has become disconnected, as described in Section 4.3.

If the MAC layer in use in the network supports MAC-layer multicast addressing and packet transmission, ADMR takes advantage of it by causing receivers and nodes in the multicast forwarding tree to join the MAC-layer multicast group corresponding to the network-layer multicast group address. For example, the IP multicast model [2] defines a mapping from "Class D" IP addresses (multicast addresses) to multicast MAC addresses for Ethernet and similar wireless media such as IEEE 802.11 wireless LANs [4]. By utilizing MAC-layer multicast when available, ADMR limits the overhead on other nodes in the network due to multicast packet transmissions.

4.3. Multicast State Maintenance

Multicast state maintenance refers to mechanisms within ADMR responsible for monitoring the forwarding tree for link breaks and for repairing these breaks when they occur. Maintenance in ADMR begins as soon as the multicast forwarding state is set up, and continues as long as the source application generates packets and there are receivers in the network interested in receiving these packets.

4.3.1. Link Break Detection

Each multicast packet originated by some node S for multicast group G contains a small ADMR header, including a number of fields used by the protocol in forwarding the packet and in maintaining the multicast distribution tree for S and G. One of the fields in the ADMR header is the inter-packet time (interval) at which new packets should be expected from this sender S for this group G. This field in the ADMR header is initialized by S for each packet originated; it is based on dynamically tracking the average interval at which it originates multicast packets for group G, or can optionally be set based on IP port number information or supplied by the application.

This inter-packet time is used by members of the multicast forwarding tree to adaptively detect disconnection in the forwarding tree, as well as inactive periods during which the source application does not send data temporarily and it will be more resource-efficient to expire the multicast state.

If the application layer at node S originates no new multicast packets for G within some multiple (e.g., 1.5) of this current inter-packet time, the routing layer at S begins originating "keep-alive" packets for G; the keep-alive packet is multicast to the group (not flooded through the network) and is used to maintain the existing forwarding state for the multicast distribution tree for S and G. The inter-packet time between keep-alives SHOULD be multiplied by some factor (e.g., 2) with each successive keep-alive, until reaching a maximum interval; after some further multiple of this interval, S is assumed to no longer be an active sender for G; in this case, the keep-alives are stopped, and all forwarding state for this sender and group in the network is allowed to expire.

The ADMR header includes the multiplicative factor increasing the time between successive keep-alives and a count of keep-alives left to send before the multicast state will expire, allowing all nodes receiving any of these keep-alive packets to know when the tree is scheduled to expire, if the sender application does not begin to send new multicast data packets before that time. The keep-alive count is initialized to EXPIRATION_KEEPALIVE_COUNT (which is based on the inter-packet time) and is decremented by 1 for each successive keep-alive.

Some forwarders or receiver members of a multicast group may become disconnected from the multicast forwarding tree for the group, as nodes in the network move or as wireless transmission conditions change. Each forwarder or receiver for some multicast group G and source S detects that it has become disconnected from the multicast forwarding tree when it fails to receive a number of successive expected multicast data (or keep-alive) packets (e.g., 3) from S for G. Upon detection of disconnection, ADMR begins link repair procedures, as described in Section 4.3.2.

The frequency of the keep-alives can be based on the source's sending pattern or can be specified by the application if it wants to ensure a maximum latency to link break discovery. In the latter case, keep-alives MAY be sent in between data packets as well, to ensure timely link break detection and repair.

The frequency of keep-alives MAY also change with the level of mobility in the network. Each receiver MAY keep track of the packet loss that it experiences based on a sequence number contained in each packet (which is also used for duplicate detection). In the event of receiver-initiated link repair (Section 4.3.2), the receiver can set the Loss Coeff field in the ADMR header (Section 4.1.2) of the Receiver Join packet that it sends to the source as part of the repair. When the source gets some number of such packets that indicate high packet loss at the multicast receivers, it can increase the frequency of the keep-alives that it sends (including ones in between data packets if necessary).

4.3.2. Link Break Repair

Each node maintains a disconnection timer for each group G and sender S for which it is either a forwarder or a receiver member, and resets this timer each time it receives a packet for the group. The timer value is based on the inter-packet time value in the ADMR header of the last received packet, plus a time proportional to the node's hop count from the source S, as determined by the forwarding of the last packet from S that updated the node's Node Table entry for S. This small increase in disconnection timeout value as a function of hop count is intended to generally allow nodes closer to S (i.e., closer to a broken link on the path from S) to detect the disconnection before nodes further from S. This property is not required for correct operation of the protocol, but it improves the efficiency of the repair process.

When some node C detects disconnection, it initiates a local repair of the multicast forwarding tree. Node C first sends a Repair Notification packet to the other nodes in the subtree "below" node C in the multicast distribution tree for group G and sender S. Here, the subtree "below" is defined by the previous hop address recorded in each node's Node Table for sender S, such that any node whose previous hop for S is node C or is some other node below C is defined to be below C in the tree. Although each multicast packet is forwarded through the tree without regard to such relationships, as described in Section 4.2, this relationship represents the set of nodes that received the previous multicast packet through C and who will thus possibly detect the disconnection themselves later, due to the increase in disconnection timer values with hop count from S.

To forward the Repair Notification packet to the nodes in the subtree below C, each node accepts and forwards the Repair Notification packet only if the MAC-layer transmitting source address of the packet matches the previous hop address stored in that node's Membership Table entry for the multicast sender S. In addition, the sequence number and bitmap in each node's Node Table entry (Section 5.3) are used to avoid duplicates in the forwarding of the Repair Notification packet.

After sending the Repair Notification packet, node C waits for a period of time REPAIR_DELAY before proceeding with its local repair. If, during this delay, node C receives a Repair Notification initiated by an upstream node for this same group and source, then C cancels its own local repair, since this other node is closer to the break and will perform the repair itself.

The Repair Notification packet serves two purposes. It is a notification to nodes in the subtree below C that a local repair is in progress and that they should not initiate their own local repair. It is also a chance to double-check that the link to node C's parent is indeed the one that is broken. The Repair Notification will be received by nodes directly below C in the forwarding tree, and if the link from C to its parent B in the tree is actually not broken, may also be received by B. In the Repair Notification packet, C lists the address of the node that is currently its parent, as represented by the previous hop address in its Membership Table entry for the multicast source S and group G. If the Repair Notification is received by this parent node, it recognizes that one of the nodes directly below it in the tree (node C) is performing a local repair. The parent then sends a one-hop Repair Notification to C, causing it to cancel its local repair as described above.

When a receiver member of the group receives a Repair Notification, it SHOULD postpone its disconnection timer for an interval of time determined by an estimate of the amount of time the local repair is expected to take.

After this short delay, if node C has not received a Repair Notification initiated by an upstream node for this group and source, node C sends a hop-limited Reconnect packet as a form of network flood. The Reconnect packet identifies the group and source for which the local repair is being performed. The hop limit for the

Reconnect packet (e.g., 3) limits this flood to only reaching nodes near C.

In addition to the normal handling of a network flood in deciding whether or not to forward the packet, nodes that are forwarders for the group G and source S being repaired treat the Reconnect packet specially. Such a node, if it has not received a Repair Notification for this repair, assumes that it is upstream of the repair node C and that it is therefore still connected to the source S in the tree. Rather than forwarding the Reconnect packet as part of the hop-limited network flood, the node instead reinitializes the packet's hop count limit (TTL) to the default value and unicasts the packet to the node listed as its parent in the previous hop address field in its Node Table entry for S. This packet is no longer treated as a network flood packet, and is instead forwarded by each node in turn to its parent in the same way, until reaching S. If the node is in fact not upstream from the repair node C and its unicast Reconnect reaches C, node C will discard the packet.

Instead, if the Reconnect reaches S (the node is truly upstream of the broken link at C and no other broken links are encountered), node S responds with a Reconnect Reply packet. This Reconnect Reply packet is unicast back to the repair node C along the path the Reconnect took to reach S, as recorded in the Node Table entry at each node for C.Each node through which the Reconnect Reply packet is forwarded on the path to C becomes a forwarder for the multicast group G and source S, and creates an entry in its Membership Table to record this if it is not already a forwarder for it.

If the local repair procedure succeeds, the multicast forwarding tree will be reconnected and the receiver members will continue to receive data as expected. If the disconnection timer expires at some receiver member R for a group G and source S, this is an indication that the local repair has probably failed, perhaps because the amount of mobility in the network has been too great to allow the type of hop-limited repair attempted. In this case, node R performs its own individual repair by rejoining the group and source in the same way as when it originally joined, as described in Section 4.1.2.

4.4. Reaction to Mobility

Each receiver MAY keep track of how many times it had to perform a global repair (Section 4.3.2) to rejoin a group because of disconnection. When this number reaches DISCONNECTION_THRESHOLD, the receiver sets the High Mobility (M) flag in the ADMR header of the Receiver Join packet. When the source receives some number of Receiver Joins with this flag set, it switches to flooding mode in which every multicast packet is flooded. The high number of re-joins indicate that the multicast state setup cannot keep up with the high mobility in the network and only flooding can deliver the data successfully. After flooding for some period of time, the protocol reverts back to its normal mode of operation as mobility in the network may have decreased.

4.5. State Expiration

Each forwarder node in the multicast forwarding tree for some group G and source S automatically expires its own state and leaves the tree when it determines that it is no longer necessary for multicast forwarding. Similarly, the multicast source S automatically expires its state and stops transmitting multicast data packets when it determines that there are no downstream receiver members of the group for this source; the sender continues to send certain of its subsequent multicast packets as infrequent background network flood packets (rather than multicasting them), but otherwise defers sending other multicasts for this group until receiving at least one new Receiver Join packet, as described in Section 4.1.1. This mechanism helps to prune nodes from the forwarding tree that are no longer needed because a downstream receiver has left or crashed or because, as a result of a disconnection and an ensuing repair, some forwarding state may no longer be necessary.

The decision to expire this state is based at each such node on a technique is similar to the use of passive acknowledgements [?]. In particular, each such node attempts to determine whether the multicast packets that it originates (at S) or forwards (at forwarder nodes) are subsequently forwarded by other nodes that received them from this node.

In order to determine this for each multicast packet, a node C expects to hear at least one other node B that received the packet from C forward the packet; as described in Section 4, when node B receives and forwards a packet, B copies the MAC-layer source address of the received packet (i.e., node C's address) into the previous hop address field in the packet's ADMR header, before forwarding the packet. If node C overhears B transmit this packet, C considers this as confirmation that it should continue forwarding subsequent multicast packets, so that nodes such as B can continue to receive them. On the other hand, if S fails to receive such confirmation for a number of consecutive multicast packets that it sends, then C decides that it is no longer necessary in the multicast forwarding tree for this group and source, or in the case of the source S itself, that no receiver members or forwarders remain. Receivers for a multicast sender and group that are not forwarders retransmit each data or keep-alive packet stripped of its data payload so that it serves in the same way as an acknowledgement to upstream nodes.

5. Conceptual Data Structures

The multicast forwarding state for ADMR is maintained locally by each node in a Sender Table (Section 5.1), Membership Table (Section 5.2), and Node Table (Section 5.3). In addition each node maintains a Send Buffer (Section 5.4).

5.1. Sender Table

The Sender Table at a node logically contains one entry for each multicast group address for which this node is an active sender. Each entry in the Sender Table includes the following values:

- The current inter-packet time for this node sending to the group.

- The multiplication factor for the inter-packet time between consecutive keep-alives.

- The inter-keepalive time.

- The value of the keep-alive packet count used for this group. This count is initialized to an EXPIRATION_KEEPALIVE_COUNT value and decremented with each successive keep-alive sent since the last data packet sent to the group by this node. The state for a given group expires when this count reaches 0.

- A mobility counter used to track high mobility and packet loss statistics received from multicast receivers via Receiver Join packets.

- A packet loss field used to track high mobility and packet loss statistics received from multicast receivers via Receiver Join packets.

- A State Setup flag. If set indicates that the STATE_SETUP timer has expired and the multicast sender can start sending if it has received at least one Receiver Join (Section 4.1.1).

- A Flood flag. If set, indicates that the next multicast data packet should be flooded. This flag is set periodically by the flood timer.

- A Flood Mode flag. If set, indicates that the Flood flag should not be cleared, i.e., all multicast data packets are to be flooded.

- A Connected flag. If set, indicates that there is at least one receiver for this sender and group in the network, as determined by this node.

5.2. Membership Table

The Membership Table at a node logically contains one entry for each combination of multicast group address and sender address for which this node is either a receiver member or a forwarder. Each entry in the Membership Table includes the following values:

- A flag to indicate if this node is a receiver.

- A flag to indicate that this node is connected to the multicast tree for this sender and group.

- A flag bit to indicate if this node is a forwarder.

- The current inter-packet time for the sender sending to this group.

- The current value of the keep-alive count from packets received for the group.

- The multiplication factor for the inter-packet time of successive keep-alives.

In addition, if a node is a receiver for this group and sender, the corresponding table entry also contains the following additional values:

- A mobility counter which keeps track of how many times the receiver was disconnected from the multicast tree. This counter is re-initialized to 0 every MOBILITY_ESTIMATION_PERIOD.

- The previous hop address of data packets forwarded by this node for state maintenance purposes (Section 4.3).

5.3. Node Table

The Node Table at a node logically contains one entry for each other node in the network from which this node has received a flooded packet. Each entry in the Node Table includes the following values:

- The sequence number from the ADMR header of the most recent tree flood or network flood packet received from the node represented by this table entry.

- A bitmap representing a number of previous sequence numbers of packets from this sender. The sequence number and bitmap are used to detect and discard duplicate packets during a flood: if the bit corresponding to some sequence number in this bitmap is set, the packet is assumed to be a duplicate; all sequence numbers prior to that corresponding to the first bit in the bitmap are also assumed to be duplicates (or are of no further interest and are discarded); this use of a bitmap is similar to the data structure suggested for anti-replay protection in the IP Security protocols [5].

- The previous hop address for the sender node and sequence number in this entry. This value is taken from the MAC-layer transmitting source address of the flood packet received for this sequence number that contained the minimum hop count in its ADMR header.

To manage space in the Node Table, new entries should be created only as needed, and existing entries should be retained in an LRU fashion.

5.4. Send Buffer

The Send Buffer of a node implementing ADMR is a queue of packets that cannot be sent by that node yet because the node does not yet know of the existence of receivers for a multicast group, or because its STATE_SETUP timer has not yet expired. Each packet in the Send Buffer is logically associated with the time that it was placed into the Buffer, and SHOULD be removed from the Send Buffer and silently discarded SEND_BUFFER_TIMEOUT seconds after initially being placed in the buffer.

6. ADMR Header Format

The Adaptive Demand-Driven Multicast Routing protocol (ADMR) makes use of a special header carrying control information that can be included in any existing IP packet. This ADMR header in a packet contains a small fixed-sized, 4-octet portion, followed by a sequence of zero or more ADMR options carrying optional information. The end of the sequence of ADMR options in the ADMR header is implied by the total length of the ADMR header.

The ADMR header is inserted in the packet following the packet's IP header, before any following header such as a traditional (e.g., TCP or UDP) transport layer header. Specifically, the Protocol field in the IP header is used to indicate that an ADMR header follows the IP header, and the Next Header field in the ADMR header is used to indicate the type of protocol header (such as a transport layer header) following the ADMR header.

The total length of the ADMR header (and thus the combined length of all ADMR options present) MUST be a multiple of 4 octets. This requirement preserves the alignment of any following headers in the packet.

6.1. Fixed Portion of ADMR Header

The fixed portion of the ADMR header is used to carry information that MUST be present in any ADMR header. This fixed portion of the ADMR header has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |    Reserved   |        Payload Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                                                               .
.                            Options                            .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Next Header

8-bit selector. Identifies the type of header immediately following the ADMR header. Uses the same values as the IPv4 Protocol field [6].

Reserved

Sent as 0; ignored on reception.

Payload Length

The length of the ADMR header, excluding the 4-octet fixed portion. The value of the Payload Length field defines the total length of all options carried in the ADMR header.

Options

Variable-length field; the length of the Options field is specified by the Payload Length field in the ADMR header. Contains zero or more pieces of optional information (ADMR options), encoded in type-length-value (TLV) format (with the exception of the Pad1 option, described in Section 6.10).

The placement of ADMR options following the fixed portion of the ADMR header MAY be padded for alignment. However, due to the typically limited available wireless bandwidth in ad hoc networks, this padding is not required, and receiving nodes MUST NOT expect options within an ADMR header to be aligned. A node inserting an ADMR header into a packet MUST set the Don't Fragment (DF) bit in the packet's IP header.

The following types of ADMR options are defined in this document for use within an ADMR header:

- Source Information Option (Section 6.2)

- Receiver Join option (Section 6.3)

- Multicast Solicitation option (Section 6.4)

- Repair Notification option (Section 6.5)

- Reconnect option (Section 6.6)

- Reconnect Reply option (Section 6.7)

- Multicast Group Address option (Section 6.8)

- Multicast Sender Address option (Section 6.9)

- Pad1 option (Section 6.10)

- PadN option (Section 6.11)

6.2. Source Information Option

The Source Information carries information initialized by the multicast sender of the packet, needed by nodes forwarding or receiving the packet. Each multicast data packet MUST contain a Source Information option. A "keep-alive" packet is encoded as a multicast packet containing a Source Information option but without a data payload following the ADMR header.

The Source Information option is encoded as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Option Type  |  Opt Data Len |         Identification        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Hop Count   | Inter-Packet Time |  Keep-Alive Count |  MF   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Address Length|   Previous Hop MAC Address
+-+-+-+-+-+-+-+-+-------------------------------------------------
```

IP fields:

Source Address

MUST be set to the address of the node originating this packet. Intermediate nodes that retransmit the packet to propagate the packet MUST NOT change this field.

Destination Address

MUST be set to the IP limited broadcast address (255.255.255.255) when the packet is sent as a network flood. Otherwise, MUST be set to the address of the multicast group to which this packet is sent.

Source Information option fields:

Option Type

2

Opt Data Len

8-bit unsigned integer, which is the length of the option, in octets, excluding the Option Type and Opt Data Len fields.

Identification

A unique value generated by the initiator (original sender) of the packet. Nodes generate a new Identification value for each multicast packet for a given multicast group, for example on a sequence number counter of all multicast packets sent by the node to the group.

This value allows a receiving node to determine whether it has recently seen a copy of this packet; if so, this receiving node MUST discard the packet. When propagating a packet, this field MUST be copied from the received copy of the packet being propagated.

Hop Count

Contains the number of hops that this packet has traversed. It is initialized to 0 by the originator of the packet and is incremented by 1 each time the packet is forwarded.

Inter-Packet Time

Contains the estimated inter-packet time (in milliseconds) for packets sent by this source application to the destination multicast group. This value MUST be set by the sender and MUST not be changed when the packet is forwarded.

Keep-Alive Count

Initialized to 0 and ignored on reception if the packet carries a data payload. In the event that no data packets are sent by a multicast source, ADMR sends a limited number of keep-alives spread over a period of time. The Keep-Alive Count field in these keep-alive packets indicates how many keep-alives are left to send before the multicast tree is scheduled to expire. This count is copied from the keep-alive count in the corresponding Sender Table entry at the source. The count in the table entry is initialized to EXPIRATION_KEEPALIVE_COUNT by the source when it starts sending keep-alives in the absence of data, and is decremented for each consecutive keep-alive. The keep-alive count in the Source Information option, along with the inter-packet time and multiplication factor there, allows nodes with multicast state for this group and source to determine when they should expire this multicast state even if some of the keep-alives are lost and not received.

MF (Multiplication Factor)

If the packet carries a data payload, this field MUST be initialized to 0 and ignored on reception. Otherwise, the keep-alive packets sent when the source application does not generate any more data packets MAY be sent at increasing inter-packet times as indicated by this field.

Address Length

The length in octets of the Previous Hop MAC Address field in the option.

Previous Hop MAC Address

Variable length field. When forwarding a packet containing a Source Information option, this field contains the MAC address of the node from which this node received the packet being forwarded; the length of this field is given by the Address Length field. When originating (rather than forwarding) this packet, the Address Length field MUST be set to 0 and this field MUST NOT be present in the option.

The Source Information option MUST be followed by a Multicast Group Address option (Section 6.8) when the packet is flooded.

6.3. Receiver Join Option

The Receiver Join option is encoded as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Option Type  |  Opt Data Len |         Identification        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     Multicast Group Address                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Loss Coeff  |M|
   +-+-+-+-+-+-+-+-+
```

IP fields:

Source Address

MUST be set to the address of the node originating this packet. Intermediate nodes that retransmit the packet to propagate the packet MUST NOT change this field.

Destination Address

MUST be set to the IP address of the multicast sender to which this receiver is trying to connect.

Receiver Join option fields:

Option Type

3

Opt Data Len

8-bit unsigned integer. The length of the option, in octets, excluding the Option Type and Opt Data Len fields. For the current definition of this option, this field MUST be set to 7.

Identification

This value MUST be copied from the keep-alive packet in response to which the Receiver Join is sent, in order to enable filtering of Receiver Joins sent by different receivers in response to the same keep-alive, as described in Sections 4.1.1 and 4.1.2.

Multicast Group Address

The address of the multicast group this node is trying to join.

Loss Coeff

Indicates on a discrete scale how much packet loss this receiver is experiencing. Setting different bits in this field can indicate different ranges of packet loss. Based on the packet loss experienced by receivers for a group, a source MAY vary the frequency of the keep-alives that it sends in between data packets in order to be able to detect link breaks faster as described in Section 4.3.1.

High Mobility (M)

The node originating this option sets this bit to indicate that the receiver has been disconnected more than DISCONNECTION_THRESHOLD times during an interval of DISCONNECTION_FREQUENCY.

Reserved

Set to 0; ignored on reception.

6.4. Multicast Solicitation Option

The Multicast Solicitation option is encoded as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Option Type  |  Opt Data Len |         Identification        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Multicast Group Address                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Hop Count   |
   +-+-+-+-+-+-+-+-+
```

IP fields:

Source Address

MUST be set to the address of the node originating this packet. Intermediate nodes that retransmit the packet to propagate the packet MUST NOT change this field.

Destination Address

MUST be set to the IP limited broadcast address (255.255.255.255)

Multicast Solicitation option fields:

Option Type

4

Opt Data Len

8-bit unsigned integer. The length of the option, in octets, excluding the Option Type and Opt Data Len fields. For the current definition of this option, this field MUST be set to 7.

Identification

A unique value generated by the initiator (original sender) of the packet. Nodes generate a new Identification value for each multicast packet for a given multicast group, for example by a sequence number counter of all multicast packets sent by the node to the group.

This value allows a receiving node to determine whether it has recently seen a copy of this packet; if so, this receiving node MUST discard the packet. When propagating a packet, this field MUST be copied from the received copy of the packet being propagated.

Multicast Group Address

The address of the multicast group this node is trying to join.

Hop Count

Contains the number of hops that this packet has traversed. It is initialized to 0 by the originator of the packet and is incremented by 1 each time the packet is forwarded.

If the Multicast Solicitation is for a specific source, a Multicast Sender Address option (Section 6.9) MUST be included after the Multicast Solicitation option.

6.5. Repair Notification Option

The Repair Notification option is encoded as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Option Type  |  Opt Data Len |         Identification        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Multicast Sender Address                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         Parent Address                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IP fields:

Source Address

MUST be set to the address of the node originating this packet. Intermediate nodes that retransmit the packet to propagate the packet MUST NOT change this field.

Destination Address

MUST be set to the IP address of the multicast group to which this repair notification is sent.

Hop Limit (TTL)

SHOULD be set to 1 if this packet is sent in response to another Repair Notification (Section 4.3.2) and to the default otherwise.

Repair Notification option fields:

Option Type

5

Opt Data Len

8-bit unsigned integer. The length of the option, in octets, excluding the Option Type and Opt Data Len fields. For the current definition of this option, this field MUST be set to 10.

Identification

A unique value generated by the initiator (original sender) of the packet. Nodes generate a new Identification value for each multicast packet for a given multicast group, for example on a sequence number counter of all multicast packets sent by the node to the group.

This value allows a receiving node to determine whether it has recently seen a copy of this packet; if so, this receiving node MUST discard the packet. When propagating a packet, this field MUST be copied from the received copy of the packet being propagated.

Multicast Sender Address

The IP address of the sender whose tree this node is trying to repair (Section 4.3.2).

Parent Address

The address of the node that last forwarded a multicast data packet for this group and source to this node. This field is used to determine if a node below this node is performing a repair for a node above this node, to which it is still connected; if so, the parent SHOULD take over the repair as it is closer to the broken link (Section 4.3.2).

6.6. Reconnect Option

The Reconnect option is encoded as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Option Type  |  Opt Data Len |         Identification        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Multicast Group Address                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Multicast Sender Address                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Hop Count   |
   +-+-+-+-+-+-+-+-+
```

IP fields:

Source Address

MUST be set to the address of the node originating this packet. Intermediate nodes that retransmit the packet to propagate the packet MUST NOT change this field.

Destination Address

MUST be set to the IP limited broadcast address (255.255.255.255) by the originator of the packet. When the Reconnect reaches a node which is part of the tree connected to the multicast sender, this node MUST set this field to the address of the multicast sender which is the target of the reconnection as described in Section 4.3.2.

Reconnect option fields:

Option Type

6

Opt Data Len

8-bit unsigned integer. The length of the option, in octets, excluding the Option Type and Opt Data Len fields. For the current definition of this option, this field MUST be set to 11.

Identification

A unique value generated by the initiator (original sender) of the packet. Nodes generate a new Identification value for each multicast packet for a given multicast group, for example on a sequence number counter of all multicast packets sent by the node to the group.

This value allows a receiving node to determine whether it has recently seen a copy of this packet; if so, this receiving node MUST discard the packet. When propagating a packet, this field MUST be copied from the received copy of the packet being propagated.

Multicast Group Address

The IP address of the group for which this link repair is performed.

Multicast Sender Address

The IP address of the sender for which this link repair is performed.

Hop Count

The number of hops that this packet has traversed. Initialized to 0 by the originator of the packet and incremented by 1 each time the packet is forwarded.

Reserved

Sent as 0; ignored on reception.

6.7. Reconnect Reply Option

The Reconnect Reply option is encoded as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Option Type  |  Opt Data Len |           Reserved            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                   Multicast Group Address                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IP fields:

Source Address

MUST be set to the address of the node originating this packet. Intermediate nodes that retransmit the packet to propagate the packet MUST NOT change this field.

Destination Address

MUST be set to the IP address of the originator of the Reconnect packet in response to which this Reconnect Reply is sent.

Reconnect Reply option fields:

Option Type

7

Opt Data Len

8-bit unsigned integer. The length of the option, in octets, excluding the Option Type and Opt Data Len fields. For the current definition of this option, this field MUST be set to 6.

Multicast Group Address

The IP address of the multicast group for which this Reconnect Reply is sent.

Reserved

Set to 0; ignored on reception.

6.8. Multicast Group Address Option

The Multicast Group Address option MUST only appear after a Source Information option (Section 6.2) and is encoded as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Option Type  |  Opt Data Len |          Reserved             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Multicast Group Address                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Multicast Group option fields:

Option Type

8

Opt Data Len

8-bit unsigned integer. The length of the option, in octets, excluding the Option Type and Opt Data Len fields. For the current definition of this option, this field MUST be set to 6.

Reserved

Set to 0; ignored on reception.

Multicast Group Address

A multicast group address.

6.9. Multicast Sender Address Option

The Multicast Sender Address option MUST only appear after a Multicast Solicitation option (Section 6.4) and is encoded as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Option Type  |  Opt Data Len |            Reserved           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Multicast Sender Address                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Multicast Sender option fields:

Option Type

9

Opt Data Len

8-bit unsigned integer. The length of the option, in octets, excluding the Option Type and Opt Data Len fields. For the current definition of this option, this field MUST be set to 6.

Reserved

Set to 0; ignored on reception.

Multicast Sender Address

The IP address of a multicast sender.

6.10. Pad1 Option

The Pad1 ADMR option is encoded as follows:

```
+-+-+-+-+-+-+-+-+
|  Option Type  |
+-+-+-+-+-+-+-+-+
```

Pad1 option fields:

Option Type

0

A Pad1 option MAY be included in the Options field of a ADMR header in order to align subsequent ADMR options, but such alignment is not required and MUST NOT be expected by nodes receiving packets containing a ADMR header.

The total length of a ADMR header, indicated by the Payload Length field in the ADMR header MUST be a multiple of 4 octets. When building a ADMR header in a packet, sufficient Pad1 or PadN options MUST be included in the Options field of the ADMR header to make the total length a multiple of 4 octets.

If more than one consecutive octet of padding is being inserted in the Options field of a ADMR header, the PadN option, described next, SHOULD be used, rather than multiple Pad1 options.

Note that the format of the Pad1 option is a special case; it does not have an Opt Data Len or Option Data field.

6.11. PadN Option

The PadN ADMR option is encoded as follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+- - - - - - - - -
|  Option Type  |  Opt Data Len |   Option Data
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+- - - - - - - - -
```

PadN option fields:

Option Type

1

Opt Data Len

8-bit unsigned integer. Length of the option, in octets, excluding the Option Type and Opt Data Len fields.

Option Data

A number of zero valued octets equal to the Opt Data Len.

A PadN option MAY be included in the Options field of a ADMR header in order to align subsequent ADMR options, but such alignment is not required and MUST NOT be expected by nodes receiving packets containing a ADMR header.

The total length of a ADMR header, indicated by the Payload Length field in the ADMR header MUST be a multiple of 4 octets. When building a ADMR header in a packet, sufficient Pad1 or PadN options MUST be included in the Options field of the ADMR header to make the total length a multiple of 4 octets.

7. Detailed Operation

7.1. General Packet Processing

In addition to the special processing of each packet discussed in Sections 7.2 through 7.4, each packet should be processed as described in this section.

7.1.1. Originating a Packet

When originating any packet, a node using ADMR MUST perform the following sequence of steps:

- If the IP destination of the packet is a multicast address, check the Sender Table for an entry for this multicast group.

- If no entry for the group exists, create an entry initialized as follows:

  * Set inter-packet time to a default or specified value

  * Set the keep-alive inter-packet time to 0 or a default.

  * Set multiplication factor to a default or specified value

  * Set keep-alive count to EXPIRATION_KEEPALIVE_COUNT

  * Set the flood flag.

  * Clear the flood mode flag.

  * Clear the mobility flag.

  Schedule a flood timer for the periodic background flood (Section 4.1.1).

- If the flood flag, the state setup flag, and the connected flag are all cleared, then the packet is placed in the Send Buffer (Section 5.4).

- If the flood flag is set, insert an ADMR header in the packet as described in Section 7.1.2, and add a Source Information option to it. Initialize the Inter-Packet Time in the Source Information option with the value from the Sender Table; the multiplication factor and keep-alive count field in the header are initialized to 0. The IP destination address of the packet is set to the limited IP broadcast address (255.255.255.255), and a Multicast Group option is added initialized to the original IP destination address of the packet. Clear the flood flag in the

Sender Table entry. Reset the keep-alive count to its default value.

- If an entry for the group exists, and the flood flag is not set, insert an ADMR header in the packet as described in Section 7.1.2 and add a Source Information option. Initialize the inter-packet in the Source Information option with the value from the Sender Table; the multiplication factor and keep-alive count field in the header are initialized to 0.

- The Identification field in the option MUST be set to a new value, different from that used for other multicast packets recently originated by this node for a particular multicast group. For example, each node MAY maintain a single counter value for generating a new Identification value for each multicast packet it originates for a given group.

- Reschedule the maintenance timer to POSTPONE_FACTOR times the average inter-packet time in the Sender Table entry.

- Increment the unforwarded packets counter in the Sender Table entry for this group. This counter is used to keep track of whether sent packets are forwarded by a downstream node (Section 4.5). Schedule the silent expiration timer with a timeout multiple of the average interpacket time. When this timer goes off, the sender will stop sending multicast packets, except for the periodic background data flood.

- Set Previous Hop MAC Address to own MAC address and set the Address Length to the length of this address in octets.

- Transmit the packet.

7.1.2. Adding an ADMR Header to a Packet

A node originating a packet adds an ADMR header to the packet, if necessary, to carry information needed by the routing protocol. A packet MUST NOT contain more than one ADMR header. An ADMR header is added to a packet by performing the following sequence of steps (these steps assume that the packet contains no other headers that MUST be located in the packet before the ADMR header):

- Insert an ADMR header after the IP header but before any other header that may be present.

- Set the Next Header field of the ADMR header to the Protocol number field of the packet's IP header.

- Set the Protocol field of the packet's IP header to the Protocol number assigned for an ADMR header.

7.1.3. Receiving a Packet

When a node receives any packet containing an ADMR header, it must process the packet according to the following sequence of steps:

- If the destination address of the packet is a multicast address and the packet carries a data payload, compare the Previous Hop Mac Address in the packet header with own MAC address. If the addresses match, clear the unforwarded packets counter in the Membership Table entry for this multicast group, or if this node is the IP source of the packet, clear the corresponding Sender Table unforwarded packets counter. Reschedule the corresponding silent expiration timer.

- If the destination address of the packet is a multicast address and the packet carries a data payload, the node should lookup the entry in its Membership Table for the IP source and destination addresses in the packet.

- If the node is a forwarder or receiver for the multicast sender and group in the packet header, and the node determines that the packet is a duplicate by checking the sender and group entry in its Node Table, and looking up the Identification field in the packet header, the packet is dropped.

- If no entry exists for this sender and group in the Node Table, such an entry is created for future duplicate detection. The Identification field is set to the Identification field in the packet.

- If the packet is not a duplicate and the node is a forwarder or receiver for this group and source (corresponding flag set in the Node Table entry for the group and source), the Node Table entry for the IP source and destination addresses is updated with the Identification field of the packet and the inter-Packet Time in its Source Information option. In addition, the disconnection timer is scheduled (or rescheduled if already scheduled) and the expiration timer is canceled if it was set.

- If the node is a receiver, the ADMR header, including all options is removed and the rest of the packet is passed to the next layer in the protocol stack.

- If the node is a forwarder, the node increments the hop count filed in the header, and hands the packet to the network layer output routine.

- If the node is a forwarder, set the Previous Hop MAC Address field to the MAC source address and set the Address Length field to the length of the Previous Hop MAC Address in octets.

- If the packet does not have a payload, each option (if any) in the ADMR header is examined and processed in the order in which it occurs in the packet, skipping over any Pad1 or PadN options.

7.2. Multicast Receiver Discovery

Multicast Receiver Discovery refers to the mechanisms within the ADMR protocol by which senders for a multicast group discover paths in the network to receivers for that multicast group. A multicast sender performs receiver discovery when it has data packets to send to a multicast group and has not yet performed a discovery for that group. Receiver Discovery MAY also be performed periodically in order to resolve partitions by sending certain of a source's packets as network floods.

The Receiver Discovery procedure utilizes two types of messages, a Receiver Discovery Keep-Alive (Section 6.2) and a Receiver Join (Section 6.3), to actively search for receivers for the multicast group and to establish routes with multicast state to these receivers in order to be able to deliver multicast data to them. These ADMR messages MAY be carried in any type of IP packet, through use of the ADMR header as described in Section 6.

7.2.1. Originating a Receiver Discovery Keep-Alive

A node initiating a Receiver Discovery for a multicast group, creates and initializes a Source Information option in an ADMR header attached to the data packet that triggered the discovery for this group. The Source Information option MUST be included in an ADMR header in the packet. To initialize the Source Information option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 2.

- The Opt Data Len field in the option MUST be set to the value 6. The total size of the Source Information option when initiated is 8 octets; the Opt Data Len field excludes the size of the Option Type and Opt Data Len fields themselves.

- The Identification field in the option MUST be set to a new value, different from that used for other multicast packets recently originated by this node for a particular multicast group. For example, each node MAY maintain a single counter

value for generating a new Identification value for each multicast packet it originates for a given group.

- The Hop Count field MUST be initialized to 0.

- The Inter-Packet Time field MUST be initialized with the inter-packet time value from the Sender Table entry for this multicast group.

- The Keep-Alive count MUST be initialized to the keep-alive count value from the Sender Table entry for this multicast group.

- The MF (Multiplication Factor) field must be initialized to 0.

- A Multicast Group option MUST be added after the Source Information option and MUST be initialized to the IP address of the group to which the data packet that triggered the discovery is addressed.

The Source Address in the IP header of this packet MUST be the node's own IP address. The Destination Address in the IP header of this packet MUST be the IP "limited broadcast" address (255.255.255.255).

A node MUST create a Sender Table when it first performs a Receiver Discovery for a multicast group (Section 5.1) and SHOULD schedule a flood timer which periodically sets the flood flag in the Sender Table entry. It MUST also schedule a state setup timer with a value of STATE_SETUP (Section 4.1.1). When this timer expires, it sets the state setup flag in the Sender Table entry for the corresponding multicast group.

The Sender Table entry MUST be updated every time a Receiver Discovery is launched by clearing the flood flag unless the flood mode flag is set.

7.2.2. Processing a Received Receiver Discovery Keep-Alive

When a node receives a packet with a Source Information option in it, which has an IP limited broadcast destination address and a Multicast Group option right after the Source Information option, it identifies the packet as a Receiver Discovery Keep-Alive and the node MUST process it according to the following sequence of steps:

- The node checks its Membership Table for an entry for the multicast sender and group in the ADMR header of this packet.

- If such an entry exists and the receiver flag is set but the connected flag is cleared, the node will send a Receiver

Join packet back towards the multicast sender as described in Section 7.2.3.

- If an entry exists with the receiver flag set, copy the multicast group address from the Multicast Group option into the IP destination field, remove all ADMR options, and pass the packet to the next layer in the protocol stack as described in Section 7.1.3.

- Create an entry in the Node Table for the IP source and multicast group of the packet if one does not already exist.

- Update the Node Table entry for the IP source of the packet with the Identification in the packet header.

- Update the Node Table entry's previous hop field with the MAC source address in the MAC header of the packet.

- Increment the Hop Count field in the Source Information option in the ADMR header.

- Transmit packet.

7.2.3. Originating a Receiver Join

A multicast receiver originates a Receiver Join in response to a Receiver Discovery Keep-Alive (Section 7.2.2 or a unicast keep-alive (Section 7.3.4) sent in response to a Multicast Solicitation (Section 7.3.2), or when the disconnection timer at a receiver for a multicast sender and group expires (Section 4.3.2).

The Receiver Join is returned in a Receiver Join option (Section 6.3).

The Receiver Join option MUST be included in an ADMR header in the packet addressed to the multicast sender. To initialize the Receiver Join option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 3.

- The Opt Data Len field in the option MUST be set to the value 7.

- The Identification field must be set to the value of the Identification field in the keep-alive packet received from the multicast sender in order to enable Receiver Join filtering (Section 4.1.1).

- The Multicast Group Address must be set to the address of the multicast group for which the keep-alive was sent by the multicast sender.

- The Loss Coefficient must be set to the value of the loss coefficient in the node's Membership Table entry for this group and source.

- The High Mobility Flag (M) must be set if the mobility flag in the node's Membership Table is set, otherwise it must be set to 0. The mobility counter must be incremented.

The Destination Address field in the IP header of the packet carrying the Receiver Join option MUST be set to the address of the multicast sender which originated the the keep-alive.

Schedule a join timer unless the Receiver Join was sent in response to a disconnection.

7.2.4. Processing a Receiver Join

When a node receives a packet with a Receiver Join option in it, it MUST process it according to the following sequence of steps:

- If the IP destination address of the packet matches the node's IP address and the node does not have a Sender Table entry for the multicast group in the packet, then the packet is dropped.

- Otherwise if the node does have a Sender Table entry, if the connected flag is not set, then the node sets the flag and if the state setup flag is set (indicating that the STATE_SETUP time has elapsed), it sends all packets addressed to this multicast group that are currently in the Send Buffer, optionally pacing them to avoid network congestion.

- The node copies the Packet Loss Coefficient into the loss coefficient field in the Sender Table entry. Based on the value of the coefficient, the node MAY change the frequency of its keep-alives by modifying the inter-keepalive time and multiplication factor in the Sender Table entry, and rescheduling the maintenance timer (Section 4.3).

- If the High Mobility flag in the packet is set, the node MUST increment the mobility counter in the Sender Table entry for the corresponding multicast group contained in the Receiver Join. If the counter exceeds the MOBILITY_HIGH threshold, the flood and flood mode flags in the Sender Table entry are set and subsequent packets MAY be flooded for a period of TEMPORARY_FLOOD.

- If the IP destination address of the packet is not this node's address, then the node MUST check its Node Table for an entry for the IP destination address of the packet.

- If the value of the Identification field in the Receiver Join option matches the value stored in the Node Table entry, the node increments the joins counter in the entry. If the counter is larger than MAX_RECEIVER_JOINS, the packet is dropped.

- Otherwise, the packet is sent after its destination MAC address is set to the address saved in the previous hop field of the Node Table entry for the IP destination address and multicast group address of the Receiver Join.

7.3. Multicast Source Discovery

Multicast Source Discovery refers to mechanisms within the ADMR protocol which allow a receiver interested in joining a multicast group to discover all sources for the group in the network (or a specific source in the case of single-source multicast), and establish paths with multicast state to them. The Multicast Source Discovery procedure uses three types of packets: Multicast Solicitation (Section 6.4), Receiver Join (Section 6.3), and unicast keep-alive (Section 6.2). These ADMR messages MAY be carried in any type of IP packet, through use of the ADMR header as described in Section 6.

7.3.1. Originating a Multicast Solicitation

A receiver node initiating a Multicast Source Discovery, creates and and initializes a Multicast Solicitation option in an ADMR header.

The Multicast Solicitation option MUST be included in an ADMR header and the destination IP address of the packet MUST be set to the limited broadcast address (255.255.255.255). To initialize the Multicast Solicitation option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 4.

- The Opt Data Len field in the option MUST be set to the value 7.

- The Identification field in the option MUST be set to a new value, different from that used for other multicast packets recently originated by this node for a particular multicast group. For example, each node MAY maintain a single counter value for generating a new Identification value for each multicast packet it originates for a given group.

- The Multicast Group Address MUST be set to the address of the multicast group the receiver is interested in joining.

- The Hop Count field must be initialized to 0.

- If the join is source specific, a Multicast Sender Address option MUST be added after the Multicast Solicitation option and initialized as described in Section 7.4.11.

- Transmit the packet.

7.3.2. Processing a Multicast Solicitation

When a node receives a packet with a Multicast Solicitation option, it MUST process the packet according to the following sequence of steps:

- If there is a Multicast Sender Address option in the packet and it matches the IP destination address in the packet, the option is processed first and the packet discarded if the sender address in the option does not match the node's address.

- The node checks its Sender Table for an entry for the multicast group in the Multicast Solicitation option.

- If such an entry exists, this node is a source for the multicast group and:

  * Creates and sends a unicast keep-alive in response to the Multicast Solicitation as described in Section 7.3.3.

  * Creates an entry in the Node Table for the IP source of the packet if one does not already exist.

  * Update the Node Table entry for the IP source of the packet with the Identification in the packet header, and the previous hop address in the entry with the MAC source address in the MAC header of the packet.

  * Increment the Hop Count field in the Multicast Solicitation option in the ADMR header.

  * If this is not a single-source join (i.e., no Multicast Sender Address option is attached), transmit packet.

- If the node does not have an entry in its Sender Table for the multicast group in the Multicast Solicitation option in the packet, then it is not a source for the group, and MUST do the following:

- Create a Node Table entry for the IP source of the Multicast Solicitation.

- Update Node Table entry for the IP source of the packet with the Identification in the packet header, and the previous hop field in the entry with the MAC source address from the MAC header of the packet.

- If the destination IP address is the limited broadcast address and this is a single-source Multicast Solicitation, the node MUST check its Membership Table for an entry for the multicast source and group. If such an entry exists and the node is a receiver or forwarder for the group and source (respective flag set in the Membership Table), it MUST set the IP destination address of the packet to the sender address from the Multicast Sender Address option, and set the MAC destination address to the previous hop field from the Membership Table entry.

- Increment the Hop Count field in the packet.

- Transmit packet.

The node MUST set an expiration time for each Sender Table entry after which the previous hop information will be considered invalid.

7.3.3. Originating a Unicast Keep-Alive

A unicast keep-alive is generated in response to a Multicast Solicitation (Section 7.3.2). The node which received the Multicast Solicitation creates and initializes a Source Information option in an ADMR header. The Source Information option MUST be included in an ADMR header in the packet. To initialize the Source Information option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 2.

- The Opt Data Len field in the option MUST be set to the value 6. The total size of the Source Information option when initiated is 8 octets; the Opt Data Len field excludes the size of the Option Type and Opt Data Len fields themselves.

- The Identification field in the option MUST be set to a new value, different from that used for other multicast packets recently originated by this node for a particular multicast group. For example, each node MAY maintain a single counter value for generating a new Identification value for each multicast packet it originates for a given group.

- The Hop Count field MUST be initialized to 0.

- The Inter-Packet Time field MUST be initialized with the inter-packet time value from the Sender Table entry for this multicast group.

- The Keep-Alive count MUST be initialized to the keep-alive count value from the Sender Table entry for this multicast group.

- The MF (Multiplication Factor) field must be initialized to 0.

- The node MUST schedule a retransmission timer for this packet (Section 4.1.2).

The Source Address in the IP header of this packet MUST be the node's own IP address. The Destination Address in the IP header of this packet MUST be the IP address of the multicast receiver which sent the Multicast Solicitation packet.

7.3.4. Processing a Unicast Keep-Alive

When a node receives a packet with a Source Information option which has a unicast IP destination address and a Multicast Group option right after the Source Information option, this is a unicast keep-alive and the node MUST process it according to the following sequence of steps:

- The node checks its Node Table for an entry for the IP destination address and multicast group from the ADMR header of this packet.

- If no such entry exists, drop the packet.

- Otherwise, update the Node Table entry for the IP source of the packet with the Identification in the packet header.

- Update the MAC next hop address of the packet with the Node Table entry's previous hop field.

- Increment the Hop Count field in the Source Information option in the ADMR header.

- Transmit the packet.

The node MUST set an expiration timer on the Node Table entry after which the previous hop information becomes invalid.

7.4. Multicast State Maintenance

Multicast State Maintenance refers to the mechanisms within the ADMR protocol by which link breaks in the multicast forwarding tree are detected and repaired. All nodes that are receivers or forwarders for a given multicast source and group are involved in multicast state maintenance.

The Multicast State Maintenance procedure utilizes 5 types of packets: Maintenance Keep-Alive (Section 6.2), Repair Notification (Section 6.5), Reconnect (Section 6.6), and Reconnect Reply (Section 6.7), to actively detect links breaks in the multicast tree and repair them incrementally as needed. These ADMR messages MAY be carried in any type of IP packet, through use of the ADMR header as described in Section 6.

7.4.1. Originating a Maintenance Keep-Alive

A Maintenance Keep-Alive is sent when the maintenance timer expires (Section 7.1.1). The Source Information option MUST be included in an ADMR header in the packet. To initialize the Source Information option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 2.

- The Opt Data Len field in the option MUST be set to the value 6. The total size of the Source Information option when initiated is 8 octets; the Opt Data Len field excludes the size of the Option Type and Opt Data Len fields themselves.

- The Identification field in the option MUST be set to a new value, different from that used for other multicast packets recently originated by this node for a particular multicast group. For example, each node MAY maintain a single counter value for generating a new Identification value for each multicast packet it originates for a given group.

- The Hop Count field MUST be initialized to 0.

- The Inter-Packet Time field MUST be initialized with the keep-alive inter-packet time value from the Sender Table entry for this multicast group. If this entry is 0, then the average inter-packet time value should be copied into it and into the Source Information option of the packet.

- The Keep-Alive count MUST be initialized to the keep-alive count value from the Sender Table entry for this multicast group, and the keep-alive count value in the Sender Table must be decremented.

- The MF (Multiplication Factor) field must be initialized to the the value from the corresponding Sender Table entry.

The Source Address in the IP header of this packet MUST be the node's own IP address. The Destination Address in the IP header of this packet MUST be the IP address of the multicast group to which this packet is sent.

7.4.2. Processing a Maintenance Keep-Alive

When a node receives a packet with a Source Information option which has an IP multicast destination address, it identifies this packet as a Maintenance Keep-Alive and the node MUST process it according to the following sequence of steps:

- The node checks its Membership Table for an entry for the multicast sender and group from the ADMR header of this packet.

- If no such entry exists, drop the packet.

- Update the Node Table entry for the IP source of the packet with the Identification in the packet's ADMR header.

- If the previous hop address in the Membership Table entry for the multicast sender and group of the packet does not match the MAC source address, drop the packet (Section 4.3).

- Increment the Hop Count field in the Source Information option in the ADMR header.

- Schedule an expiration timer based on the Inter-Packet Time, Keep-Alive Count, and MF (Multiplication Factor) fields in the Source Information option.

- If node is a receiver, strip data payload from packet if any. A receiver transmits a packet stripped of its data payload so that it serves as a passive acknowledgement to upstream nodes (Section 4.5).

- Transmit packet.

7.4.3. Originating a Repair Notification

A node originates a Repair Notification packet when its disconnection timer expires and its Membership Table has a set forwarding flag for a given multicast source and group. In addition, a node originates a Repair Notification when it receives a Repair Notification from a node in which the Parent Node Address field is set to this node's address. In this case the TTL value in the IP header of the packet is set to 1.

The Repair Notification option MUST be included in an ADMR header and the destination IP address of the packet MUST be the address of the multicast group for which a link break has been detected. To initialize the Repair Notification Option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 5.

- The Opt Data Len field in the option MUST be set to the value 10.

- The Identification field in the option MUST be set to a new value, different from that used for other multicast packets recently originated by this node for a particular multicast group. For example, each node MAY maintain a single counter value for generating a new Identification value for each multicast packet it originates for a given group.

- The Multicast Sender Address field MUST be initialized with the IP address of the multicast sender in whose multicast tree the link break was detected.

- The Parent Address field MUST be set to the previous hop field from the node's Membership Table for the given multicast sender and group.

- If the Repair Notification is sent in response to another Repair Notification, set the TTL field in the packet to 1 (Section 4.3.2).

- The node MUST schedule the repair timer.

- If node is a receiver as indicated by the receiver flag in its Membership Table entry, it MUST reschedule its disconnection timer by LOCAL_REPAIR_TIME.

- Transmit packet.

7.4.4. Processing a Repair Notification

When a node receives a Repair Notification, it MUST process it according to the following sequence of steps:

- If the Parent Address field matches the node's address, it drops the Repair Notification packet, and creates and sends its own Repair Notification with a ttl = 1 as described in Section 7.4.3.

- Otherwise, if the node has an entry in its Membership Table for the multicast group address in the destination IP address of the packet, and the multicast sender in the Multicast Sender Address field in the Repair Notification option, and if the MAC source address of the packet matches the previous hop field in the Membership Table entry:

  * If this entry indicates this node as a forwarder, then it cancels its disconnection timer and the repair timer if scheduled, and transmits the packet.

  * Otherwise, if this entry indicates this node as a receiver, then the node postpones its disconnection timer by LOCAL_REPAIR_TIME.

  * If the node is also a forwarder, it transmits the packet.

- Otherwise, drop the packet.

7.4.5. Originating a Reconnect

A node originates a Reconnect after the repair timer for a given multicast source and group expires. The Reconnect option MUST be included in an ADMR header and the destination IP address of the packet MUST be set to the limited broadcast address (255.255.255.255). To initialize the Reconnect Option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 6.

- The Opt Data Len field in the option MUST be set to the value 11.

- The Identification field in the option MUST be set to a new value, different from that used for other multicast packets recently originated by this node for a particular multicast group. For example, each node MAY maintain a single counter value for generating a new Identification value for each multicast packet it originates for a given group.

- The Multicast Group Address option MUST be initialized to the IP address of the multicast group for which the repair is being performed.

- The Multicast Sender Address field MUST be initialized to the IP address of the multicast sender for which the repair is being performed.

- The Hop Count field MUST be initialized to 0.

- The IP TTL field MUST be set to LOCAL_REPAIR_TTL.

- Transmit the packet.

7.4.6. Processing a Reconnect

When a node receives a Reconnect, it MUST process it according to the following sequence of steps:

- If the source IP address in the packet matches this node's address, drop packet.

- If the address in the Multicast Sender Address field in the Reconnect option matches the address of the node: if the node has an entry in its Sender Table for the multicast group address from the Multicast Group Address field of the Reconnect option, then the node creates and sends a Reconnect Reply packet (Section 7.4.7), otherwise if no Sender Table entry exists, the Reconnect is dropped.

- Create an entry in the Node Table for the IP source of the packet if one does not already exist.

- Update the Node Table entry for the IP source of the packet with the Identification in the packet header, as well as the MAC source address in the MAC header of the packet.

- Check Membership Table for an entry for the multicast source and group address listed in the Reconnect option. If such an entry exists and the forwarder and connected flags are set, then set the IP destination address of the packet to the value of the Multicast Sender Address field and the MAC destination address to the previous hop entry from the Membership table entry.

- Increment the Hop Count field in the Reconnect option in the ADMR header.

- Transmit packet.

7.4.7. Originating a Reconnect Reply

A node originates a Reconnect Reply in response to a Reconnect packet as described in Section 4.3.2. The Reconnect Reply option MUST be included in an ADMR header and the destination IP address of the packet MUST be set to the IP source address (i.e., the address of the node that initiated the repair) from the received Reconnect packet. To initialize the Reconnect Reply option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 7.

- The Opt Data Len field in the option MUST be set to the value 6.

- The Multicast Group Address field MUST be initialized to the address of the multicast group this Reconnect Reply pertains to.

- Transmit packet.

7.4.8. Processing a Reconnect Reply

When a node receives a packet with a Reconnect Reply option in it, it MUST process it according to the following sequence of steps:

- If the IP destination in the packet header matches the address of the node, the node sets the connected flag in its Membership Table entry for the multicast group in the Multicast Group Address field in the Reconnect Reply option and the IP source address of the packet. If no such Membership Table entry exists, the packet MUST be dropped.

- If the IP destination address of the packet does not match the node's IP address, then the node transmits the packet after it sets the destination MAC address to the value saved in the previous hop field in the Node Table entry for the multicast group in the Multicast Group Address field in the Reconnect Reply option and the IP source address of the packet. This entry was created when this node forwarded the Reconnect packet in response to which the Reconnect Reply was sent. If no such entry exists, the packet MUST be dropped.

- If this node is also a receiver, reschedule the disconnection timer.

7.4.9. Originating a Multicast Group Option

The Multicast Group option MUST only appear after a Source Information option with a limited broadcast IP destination address (255.255.255.255). To initialize the Multicast Group option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 8.

- The Opt Data Len field in the option MUST be set to the value 6.

- The Multicast Group Address must be set to the IP address of the multicast group to which the packet is being sent.

7.4.10. Processing a Multicast Group Option

A Multicast Group option is processed as described in Section 7.2.2.

7.4.11. Originating a Multicast Sender Address Option

The Multicast Group option MUST only appear after a Multicast Solicitation option. To initialize the Multicast Sender Address option, the node performs the following sequence of steps:

- The Option Type in the option MUST be set to the value 9.

- The Opt Data Len field in the option MUST be set to the value 6.

- The Multicast Sender Address must be set to the IP address of the multicast sender to which the receiver is interested in sending a single-source Multicast Solicitation.

7.4.12. Processing a Multicast Sender Address Option

A Multicast Sender Address option is processed as described in Section 7.3.2.

8. Constants

STATE_SETUP

A period of time during which multicast senders buffer data packets when they first perform a Receiver Discovery for a group (Section 4.1.1). The STATE_SETUP wait time is intended to allow multicast state to be set up in the network before the data is sent out.

EXPIRATION_KEEPALIVE_COUNT

The number of consecutive keep-alive packets that are sent before multicast state for a given group and source expires (Section 4.5). Value SHOULD be determined based on the inter-packet time for the multicast group.

DISCONNECTION_THRESHOLD

The DISCONNECTION_FREQUENCY that merits setting the High Mobility (M) flag in the Receiver Join option in the ADMR header (Section 7.2.3).

DISCONNECTION_FREQUENCY

The number of times a receiver has gotten disconnected over a period of MOBILITY_ESTIMATION_PERIOD (Section 4.3.1).

MOBILITY_ESTIMATION_PERIOD

Controls the interval over which DISCONNECTION_FREQUENCY is computed.

MOBILITY_HIGH

The threshold used by multicast senders to determine if enough receivers have set the High Mobility flag in their Receiver Joins to merit going into flood mode (Section 4.1.1).

TEMPORARY_FLOOD

The period of time a multicast sender spends in flood mode once the mobility counter has exceeded the MOBILITY_HIGH threshold (Section 4.1.1).

REPAIR_DELAY

The interval of time a node attempting a repair should wait after sending a Repair Notification and before initiating a local repair.

LOCAL_REPAIR_TTL

The number of hops a Reconnect packet is allowed to traverse away from the source of the packet.

POSTPONE_FACTOR

The amount of time a receiver postpones initiating global repair after it gets notification that a local repair is in progress (Section 4.3.2).

LOCAL_REPAIR_TIME

The estimated time that a local repair would take. Used by receivers for a group and source to postpone their disconnection timer (which triggers global repair (Section 4.3.2)).

MAX_RECEIVER_JOINS

The maximum number of Receiver Joins sent in response to a given data flood that a network node should forward (Section 4.1.1).

9. IANA Considerations

This document proposes the use of an ADMR header, which requires an IP Protocol number.

In addition, this document proposes use of the value "No Next Header" originally defined for use in IPv6) within an IPv4 packet, to indicate that no further header follows an ADMR header.

10. Security Considerations

This document does not specifically address security concerns. This document does assume that all nodes participating in the ADMR protocol do so in good faith and without malicious intent to corrupt the routing ability of the network. In mission-oriented environments where all the nodes participating in the ADMR protocol share a common goal that motivates their participation in the protocol, the communications between the nodes can be encrypted at the physical channel or link layer to prevent attack by outsiders.

Acknowledgments

The protocol described in this draft has been designed and developed within the Monarch Project, a research project at Rice University and Carnegie Mellon University which is developing adaptive networking protocols and protocol interfaces to allow truly seamless wireless and mobile node networking.

References

[1] Scott Bradner. Key words for use in RFCs to Indicate Requirement Levels. RFC 2119, March 1997.

[2] Steve Deering. Host extensions for IP multicasting. RFC 1112, August 1989.

[3] Hugh Holbrook and Brad Cain. Source-specific multicast for ip. Internet-Draft, draft-holbrook-ssm-arch-01.txt, November 2000. Work in progress.

[4] IEEE Computer Society LAN MAN Standards Committee. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, New York, 1997.

[5] S. Kent and R. Atkinson. Security architecture for the internet protocol. RFC 2401, November 1998.

[6] Joyce K. Reynolds and Jon Postel. Assigned numbers. Internet Request For Comments RFC 1700, October 1994.

Chair's Address

The Working Group can be contacted via its current chairs:

M. Scott Corson
Institute for Systems Research
University of Maryland
College Park, MD 20742
USA

Phone: +1 301 405-6630
Email: corson@flarion.com

Joseph Macker
Information Technology Division
Naval Research Laboratory
Washington, DC 20375
USA

Phone: +1 202 767-2001
Email: macker@itd.nrl.navy.mil

Authors' Addresses

Questions about this document can also be directed to the authors:

Jorjeta G. Jetcheva
Carnegie Mellon University
Computer Science Department
5000 Forbes Avenue
Pittsburgh, PA 15213-3891
USA

Phone: +1 412 268-3053
Fax: +1 412 268-5576
Email: jorjeta@cs.cmu.edu

David B. Johnson
Rice University
Computer Science Department, MS 132
6100 Main Street
Houston, TX 77005-1892
USA

Phone: +1 713 348-3063
Fax: +1 713 348-5930
Email: dbj@cs.rice.edu