

INTERNET-DRAFT

Zygmunt J. Haas, Cornell University
Marc R. Pearlman, Cornell University
Prince Samar, Cornell University

Expires in six months on January 2003

July 2002

The Bordercast Resolution Protocol (BRP) for Ad Hoc Networks
<[draft-ietf-manet-zone-brp-02.txt](#)>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC 2026](#), except the right to produce derivative works is not granted.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

Distribution of this memo is unlimited.

Abstract

The Bordercast Resolution Protocol (BRP) provides the bordercasting packet delivery service used to support network querying applications. The BRP uses a map of an extended routing zone, provided by the local proactive Intrazone Routing Protocol (IARP), to construct bordercast (multicast) trees, along which query packets are directed. Within the context of the hybrid ZRP, the BRP is used to guide the route requests of the global reactive Interzone Routing Protocol (IERP). The BRP employs special query control mechanisms to steer route requests away from areas of the network that have already been covered by the query. The combination of multicasting and zone based query control makes bordercasting an efficient and tunable service that is more suitable than flood searching for network probing applications like route discovery.

Contents

Status of this Memo	i
Abstract	i
Applicability Statement	iii
A. Networking Context	iii
B. Protocol Characteristics and Mechanisms	iii
1. Introduction	1
2. Bordercasting (BRP) Terminology	2
3. Bordercasting -- Routing Zone Based Querying	3
4. Bordercast Resolution Protocol (BRP) Implementation	6
A. Packet Format	7
B. Data Structures	8
C. Interfaces	8
D. State Machine	9
E. Pseudocode Implementations	10
5. References	13
6. Draft Modifications	14
Authors' Information	15
MANET Contact Information	15

Applicability Statement

A. Networking Context

Bordercasting is an efficient multicast packet delivery service used for guiding queries through the network. When each node proactively tracks the topology of its surrounding extended routing zone, queries can be directed to the edge of the node's routing zone rather than blindly being relayed to **all** neighbors. Special routing zone based query control mechanisms steer query packets away from regions of the network that have already been covered by the query.

Within the context of ad hoc network routing, bordercasting is proposed as a more efficient and tunable alternative to broadcasting of route request messages for reactive (on-demand) routing protocols. We refer to any reactive routing protocol that bordercasts route requests as an Interzone Routing Protocol (IERP). The link state information needed to support bordercasting is provided by a local proactive Intrazone Routing Protocol (IARP). Thus, a routing protocol based on bordercasting is actually hybrid reactive/proactive. For a properly chosen routing zone radius, IARP's cost of tracking routing zone topology is more than justified by the resulting savings in route discovery overhead through bordercasting.

B. Protocol Characteristics and Mechanisms

The Bordercast Resolution Protocol (BRP) is a packet delivery service, not a full featured routing protocol. Bordercasting is enabled by a local proactive Intrazone Routing Protocol (IARP) and supports a global reactive Interzone Routing Protocol (IERP). The characteristics of the IARP [2] and IERP [3] are summarized in their corresponding Internet drafts.

Haas, Pearlman, Samar

Expires January 2003

[Page iii]

1. Introduction

The design of ad hoc network routing protocols is influenced by link instability (due to node mobility) and limitations in available bandwidth and transmission power. Traditional wired networks use proactive routing protocols, like OSPF [7] and RIP [15], to maintain up-to-date routes to all networks nodes. More efficient proactive routing protocols have been developed for ad hoc networks [1][5][8][9][14]. However, continuously tracking the frequent topology changes in a practical ad hoc network can still produce an overwhelming amount of control traffic. Even worse, most of the acquired route information expires before it is ever used, making the proactive control traffic a poor investment of bandwidth. In contrast, reactive routing protocols [6][10][13] only initiate a global, query-based, route discovery as routes are needed. While some delay is incurred for route acquisition, the amount of overhead traffic is generally much less than proactive routing protocols, because routing information is not wasted. For this reason, reactive protocols are generally viewed as being more suitable than proactive routing protocols for the power/bandwidth limited mobile ad hoc network.

Although a global proactive routing protocol may overwhelm an ad hoc network's resources, a LIMITED SCOPE proactive routing protocol can be used to benefit a global reactive routing protocol. This hybrid routing approach forms the basis for the Zone Routing Protocol (ZRP) framework [4].

The cost for each node to proactively track the topology of its surrounding R-hop neighborhood (routing zone) can be justified by improved route discovery efficiency and more effective route maintenance [11][12]. Routes to local destinations are immediately available, avoiding route discoveries. When the global route discovery is needed, the routing zones can be used to efficiently guide route queries outward through bordercasting, rather than blindly relaying queries from neighbor to neighbor. The proactive maintenance of routing zones also helps improve the quality and survivability of discovered routes, by making them more robust to changes in network topology. Once routes have been discovered, routing zone offers enhanced, real-time, route maintenance. Link failures can be bypassed by multiple hop paths within the routing zone. Similarly, sub-optimal route segments can be identified and traffic re-routed along shorter paths.

The bordercasting packet delivery service is provided by the Bordercast Resolution Protocol (BRP). The BRP uses a map of an extended routing zone, provided by the local proactive Intrazone Routing Protocol (IARP) [2], to construct bordercast (multicast) trees along which query packets are directed. (Within the context of the hybrid ZRP, the BRP is used to guide the route requests of the global reactive Interzone Routing Protocol (IERP) [3]). The BRP uses special query control mechanisms to steer route requests away from areas of the network that have already been covered by the query. The combination of multicasting and zone based query control makes bordercasting an efficient and tunable service that is more suitable than flood searching for network probing applications like route discovery.

2. Bordercasting (BRP) Terminology

bordercast

A packet delivery service, based on routing zones, which distributes a message through a network in such a way that all reachable network nodes belong to the routing zone of at least one node that has relayed the message. Bordercasting uses the known structure of routing zones to efficiently relay messages away from regions of the network where the message has already appeared.

This type of delivery is intended for efficient network querying, where nodes proactively distribute queriable information throughout their routing zones.

bordercasting node

A node that is relaying / has relayed a message as part of a bordercast

bordercast tree

A multicast tree, rooted at a bordercasting node X, which spans the uncovered peripheral nodes of node X.

covered node

A node that belongs to the routing zone of a bordercasting node.

interior node

A node which lies inside of a routing zone. A routing zone member is either an interior node or a peripheral node.

peripheral node

A node which lies at the edge of a routing zone

Haas, Pearlman, Samar

Expires January 2003

[Page 2]

routing zone

With respect to a given node X, the collection of nodes whose connectivity can be monitored by X through a localized proactive routing protocol (ie. an Intrazone Routing Protocol (IARP)).

routing zone radius

The distance (in hops) from a node to the peripheral nodes of its routing zone

uncovered node

A node that does not belong to the routing zone of a bordercasting node.

zone radius

see routing zone radius

3. Bordercasting -- Routing Zone Based Querying

In this section, we describe the basic operation of route discovery based on bordercasting. A node, in need of a route to a destination, first checks whether the destination lies within its routing zone. If a path to the destination is known, no further route discovery processing is required. On the other hand, if the destination is not within the source's routing zone, the source node constructs a "bordercast tree" spanning all of its peripheral nodes (ie. the nodes on the edge of its routing zone). It then forwards a route query to the neighbors in this tree.

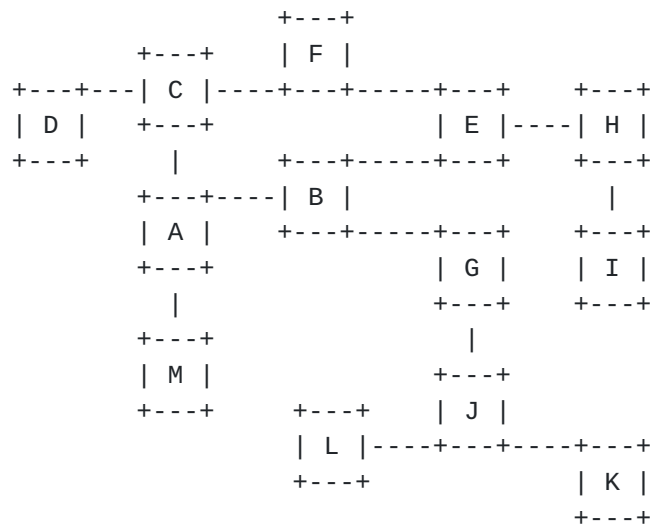
The first time that a node receives a copy of the route query, the node determines whether it belongs to the bordercast tree of the neighbor that forwarded the query, because only tree members need to actively process the route query. If the query was forwarded via layer 2 unicast, tree membership is implied by receipt of the query. If the query was forwarded via layer 2 broadcast, the node must reconstruct the bordercast tree of its forwarding neighbor. If the node determines that it does not belong to the bordercast tree, it simply notes reception of the query and then discards the message.

Haas, Pearlman, Samar

Expires January 2003

[Page 3]

If the node does belong to the forwarding neighbors bordercast tree, it proceeds to process the route query. If the query destination lies in the node's routing zone, it sends a route reply back to the query source, indicating a route to the destination. Otherwise, the node constructs its own bordercast tree, which spans the subset of its peripheral nodes that have not been covered by the query. (After a node processes a route query, the nodes in its routing zone are considered covered. The objective of bordercasting is to forward the route query toward peripheral nodes that have not been covered by the query.) The node then forwards the query to the neighbors in its bordercast tree. After forwarding the query, the node's routing becomes covered, thereby making it unnecessary for the node to forward subsequent copies of the route query.



In the example illustrated above, node A has data to send to node L. Assuming each node's zone radius is 2 hops, node L does not lie within node A's routing zone (which includes nodes B,C,D,E,F,G). Therefore, node A constructs a bordercast tree spanning its peripheral nodes: D,E,F and G. Node A then sends the route query to neighbors in this multicast tree: B and C. Each of these neighbors check if L belongs to its routing zone. Since node L is not found in any of these nodes' routing zones, the nodes construct bordercast trees spanning their uncovered peripheral nodes and send the query to neighbors in their bordercast trees. In particular, node B constructs a bordercast tree spanning its uncovered peripheral nodes F,H and J. Nodes C and M are excluded because they belong to the routing zone of A (the node that relayed the query to B). Node B sends the query to its bordercast tree neighbors: E and G. Likewise, node C identifies its uncovered peripheral nodes (node E) and forwards the route query to its neighbor, node F.

Haas, Pearlman, Samar

Expires January 2003

[Page 4]

The full trace of this bordercast route discovery example is summarized in the following table.

+-----+				
Rcv'd		Peripheral Nodes		Relays to
From	Node	Covered	Uncovered	(Tree Neighbors)
+-----+				
---	A		E,D,F,G	B,C
A	B	C,M	F,H,J	E,G
A	C	B,M	E	F
B	E	A,G	I,C	H,F
B	G	destination discovered -- reply sent		
C	F	A,D	B,H	E
E	H	F,B	---	---
E	F	A,D,B,H	---	---
F	E	A,G,C,I	---	---
+-----+				

Eventually, a route query is received by node G, which has the destination L in its routing zone. Node G does not forward the query, but sends a route reply back to A, indicating the discovered path: A--B--G--J--L.

This example demonstrates the efficiency of bordercasting as compared with flood searching. If the network consists of point-to-point links, bordercasting generates 8 query transmissions. If the network consists of a single, shared channel, then bordercasting generates 5 query broadcasts. In contrast, flood searching would generate 13 point-to-point query transmissions, or 12 query broadcasts.

4. Bordercast Resolution Protocol (BRP) Implementation

When a node's BRP receives a bordercast query packet, it marks the routing zone of the previous bordercasting node as having been "covered". If the node is an intended recipient of the query, it proceeds to deliver the query message up to the querying application (eg. IERP). To enhance bordercasting efficiency, this delivery is scheduled with some random delay. This provides more opportunity for the node to acquire query coverage information prior to forwarding the query (see below). If the node is not an intended recipient of the query, it is implied that the node's own routing zone has been covered by other bordercasting nodes. As such, the node marks its entire routing zone as covered and discards the query.

After processing the query, the querying application may return the query to BRP. If the node knows the route to the destination, it forwards the query to the destination. Otherwise, the node forwards the query to neighbors that span its uncovered peripheral nodes in its bordercast tree. In either case, after forwarding the query packet, the node marks all nodes in its routing zone as covered.

INTERNET DRAFT

Bordercast Resolution Protocol

July 2002

A. Packet Format

```

      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
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               Query Source Address                      |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               Query Destination Address                  |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|               Query ID          |Query Extension|   RESERVED            |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               Prev Bordercast Address                    |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|
|               E N C A P S U L A T E D           P A C K E T             |
|
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

- * Query Source Address (node_id) (32 bits)
IP address of the node that initiates the query.
 - * Query Destination Address (node_id) (32 bits)
IP address of the node that is the ultimate query destination.
 - * Query ID (unsigned int) (16 bits)
Sequence number which, along with the Query Source Address uniquely identifies any BRP query in the network.
 - * Query Extension (char) (8 bits)
Indicates whether query should be forwarded to query destination.
 - * Prev Bordercast Address (node_id) (32 bits)
IP address of the most recent bordercasting node.
 - * Encapsulated Packet (packet)
- *** note: Within the context of the BRP, the Query Source Address, Query Destination Address and Query ID can assume the same values as corresponding fields in the encapsulated query packet.

These BRP fields can be eliminated AS LONG AS:

- (a) The BRP has read access to the contents of the encapsulated packet.
- (b) The BRP knows the format of the query packet.

Haas, Pearlman, Samar

Expires January 2003

[Page 7]

INTERNET DRAFT

Bordercast Resolution Protocol

July 2002

B. Data Structures

B.1 IARP Routing Table (see IARP specification [\[2\]](#))

B.2 IARP Link State Table (see IARP specification [\[2\]](#))

B.3 Query Coverage

+-----+-----+-----+			
Query	Query ID	BRP_cache_ID	graph
Source			
(node_id)	(unsigned int)	(unsigned int)	(net graph)*
+-----+		+-----+	+-----+
+-----+		+-----+	+-----+
+-----+		+-----+	+-----+

* net_graph is a data structure that represents routing zone connectivity and whether each routing zone member has been covered by the query.

C. Interfaces

C.1 Higher Layer (IERP)

C.2.1 Send(packet, BRP_cache_ID)

Used by the IERP to request packet transmission.

C.2 Lower Layer (IP)

C.2.1 Deliver(packet)

Used by IP to deliver packet to BRP

D. State Machine

The BRP protocol consists of only one state (IDLE). Therefore, no state transitions need to be specified. The BRP immediately acts upon an event and then returns back to the IDLE state.

Notes: 1) X is used as a label for the node running this state machine.

D.1

Event: A query is received from the higher layer (IERP).
An intrazone route to the query destination exists.

Action: If X has not already relayed the query to the destination, X sends the query packet to the next hop to the query destination.

D.2

Event: A query is received from the higher layer (IERP).
An intrazone route to the query destination
DOES NOT exist.

Action: X constructs the bordercast tree spanning its "uncovered" peripheral nodes. The query packet is forwarded to the node's tree neighbors. Once the query is forwarded, the node marks all of its routing zone nodes as covered

D.3

Event: A query is received from IP.

Action: Mark the routing zone nodes of the previous bordercaster as covered.

If X is an intended recipient for the query, it records its BRP state for this query and schedules (with a random delay) delivery of the encapsulated query to the higher layer (i.e. IERP).

If X is not an intended recipient for the query, it is implied that X's routing zone is covered by the routing zones of other relaying nodes. Thus, X marks its own routing zone as covered and discards the packet

E. Pseudocode Implementation

We define two complimentary operations on packets:
extract(packet) and load(packet)

```
extract (packet)
    extracts the fields of the BRP packet to the following
    variables: {source, query_id, prev_bordercast,
               encap_packet}
```

```
load (packet)
    loads the values of the aforementioned variables into
    the fields of the BRP packet.
```

E.1 Send(encap_packet, BRP_cache_ID)

```
// If BRP_cache_ID is not NULL, then this is an existing
// query that is being relayed and BRP state is extracted
// from the Query Coverage Table. Otherwise, this is a
// new query and BRP state is initialized.
if(BRP_cache_ID)
{
    source      = Query_Coverage[BRP_cache_ID].source;
    query_id    = Query_Coverage[BRP_cache_ID].query_id;
}
else
{
    source      = MY_ID;
    query_id    = MY_BORDERCAST_ID++;
}
```

```

        Query_Coverage[source,query_id] =
            new_net_graph(IARP_link_state_table);
    }
    coverage = Query_Coverage[source,query_id].graph;

```

Haas, Pearlman, Samar

Expires January 2003

[Page 10]

INTERNET DRAFT

Bordercast Resolution Protocol

July 2002

```

if((EXIST)IARP_route_table[query_dest])
{
    // Route to destination is KNOWN

    // If the query destination is not already covered,
    // select next hop to query destination as the
    // outgoing neighbor.
    if(!covered(coverage, query_dest))
        out_neighbors =
            IARP_Route_Table[query_dest].next_hop;
    else
        out_neighbors = {};
}
else
{
    // Route to destination is UNKNOWN

    // Construct the node's bordercast tree, spanning
    // all remaining "uncovered" peripheral nodes.
    bordercast_tree =
        construct_bordercast_tree(coverage, MY_ID);
    out_neighbors =
        bordercast_tree.my_dowstream_neighbors();
}

// Relay query packet to outgoing neighbors.

```

```

prev_bcast = MY_ID;
load(packet);
send(packet,out_neighbors, IP);

// After relaying the route query, the node can mark its
// entire routing zone as covered.
my_zone_nodes = construct_routing_zone(coverage, MY_ID);
record_query_coverage(coverage, my_zone_nodes);

```

E.2 Deliver(packet)

```

extract(packet);

// Load the known coverage of this query
if(!(EXISTS) Query_Coverage[source, query_id])
{
    Query_Coverage[source,query_id].graph =
        new_net_graph(IARP_link_state_table);
    Query_Coverage[source,query_id].BRP_cache_ID =
        BRP_cache_ID++;
}
coverage = Query_Coverage[source, query_id].graph;

// Mark the previous bordercaster's routing zone nodes as
// covered.
prev_bcast_zone_nodes =
    construct_routing_zone(coverage, prev_bcast);
record_query_coverage(coverage, prev_bcast_zone_nodes);

// If this node is the previous bordercaster's outgoing
// neighbor, then this node becomes a bordercasting node

```

```

if(is_out_neighbor(prev_bcast, MY_ID, coverage))
{
    // Deliver query to querying application (eg. IERP)
    // with a randomly generated delay.
    schedule(deliver(encap_packet, BRP_cache_ID),
              RELAY_JITTER);

    // Schedule deletion of Query_Coverage entry for some
    // future time after route query has died off in the
    // network
    schedule(remove(Query_Coverage[BRP_cache_ID]),
              MAX_QUERY_LIFETIME);
}
else
{
    // This node does not need to relay the query, because
    // its entire routing zone is covered by prev_bcast and
    // prev_bcast's bordercast tree neighbors.
    my_zone_nodes =
        construct_routing_zone(coverage, MY_ID);
    record_query_coverage(coverage, my_zone_nodes);

    discard(encap_packet);
}

```

5. References

- [1] Garcia-Luna-Aceves, J.J. and Spohn, M., "Efficient Routing in Packet-Radio Networks Using Link-State Information," WCNC '99, September 1999.
- [2] Haas, Z.J., Pearlman, M.R. and Samar, P., "Intrazone Routing Protocol (IARP)," IETF Internet Draft, [draft-ietf-manet-iarp-02.txt](#), July 2002.
- [3] Haas, Z.J., Pearlman, M.R. and Samar, P., "Interzone Routing Protocol (IERP)," IETF Internet Draft, [draft-ietf-manet-ierp-02.txt](#), July 2002.
- [4] Haas, Z.J., Pearlman, M.R. and Samar, P., "Zone Routing Protocol (ZRP)," IETF Internet Draft, [draft-ietf-manet-zrp-04.txt](#), July 2002.

- [5] Jacquet, P., Muhlethaler, P., Qayyum A., Laouiti A., Viennot L., and Clausen T., "Optimized Link State Routing Protocol," IETF Internet Draft, [draft-ietf-manet-olsr-03.txt](#), November 2000.
- [6] Johnson, D.B. and Maltz, D.A., "Dynamic Source Routing in Ad-Hoc Wireless Networks," in Mobile Computing, edited by T. Imielinski and H. Korth, chapter 5, pp. 153-181, Kluwer, 1996.
- [7] Moy, J., "OSPF Version 2," INTERNET DRAFT STANDARD, [RFC 2178](#), July 1997.
- [8] Murthy, S. and Garcia-Luna-Aceves, J.J., "An Efficient Routing Protocol for Wireless Networks," MONET, vol. 1, no. 2, pp. 183-197, October 1996.
- [9] Ogier, R. "Efficient Routing Protocols for Packet-Radio Networks Based on Tree Sharing," MoMUC '99, November 1999.
- [10] Park, V.D. and Corson, M.S., "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," IEEE INFOCOM'97, Kobe, Japan, 1997.
- [11] Pearlman, M.R. and Haas, Z.J., "Determining the Optimal Configuration for the Zone Routing Protocol," IEEE JSAC, August, 1999.
- [12] Pearlman, M.R., Haas, Z.J. and S.I. Mir, "Using Routing Zones to Support Route Maintenance in Ad Hoc Networks," IEEE WCNC 2000, Chicago, IL, Sept. 2000.

Haas, Pearlman, Samar Expires January 2003 [Page 13]

INTERNET DRAFT Bordercast Resolution Protocol July 2002

- [13] Perkins, C.E. and Royer, E.M., "Ad Hoc On-Demand Distance Vector Routing," IEEE WMCSA'99, New Orleans, LA, Feb. 1999
- [14] Perkins, C.E. and Bhagwat, P., "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," ACM SIGCOMM, vol.24, no.4, October 1994.
- [15] Postel, J., "Internet Protocol," [RFC 791](#), Sept. 1981.

6. Draft Modifications

The following are major changes between this version (02) of the BRP draft and the previous version (01):

- The role of "intermediate node" has been eliminated from this version of bordercasting. All nodes that relay bordercast messages are now considered bordercasting nodes and therefore execute a common, simplified algorithm. The BRP description, example and implementation have been updated accordingly.
- A terminology section has been added.

Haas, Pearlman, Samar

Expires January 2003

[Page 14]

INTERNET DRAFT

Bordercast Resolution Protocol

July 2002

Authors' Information

Zygmunt J. Haas
Wireless Networks Laboratory
323 Frank Rhodes Hall
School of Electrical & Computer Engineering
Cornell University
Ithaca, NY 14853

United States of America
tel: (607) 255-3454, fax: (607) 255-9072
Email: haas@ece.cornell.edu

Marc R. Pearlman
389 Frank Rhodes Hall
School of Electrical & Computer Engineering
Cornell University
Ithaca, NY 14853
United States of America
tel: (607) 255-0236, fax: (607) 255-9072
Email: pearlman@ece.cornell.edu

Prince Samar
374 Frank Rhodes Hall
School of Electrical & Computer Engineering
Cornell University
Ithaca, NY 14853
United States of America
tel: (607) 255-9068, fax: (607) 255-9072
Email: samar@ece.cornell.edu

The MANET Working Group contacted through its chairs:

M. Scott Corson
Flarion Technologies, Inc.
Bedminster One
135 Route 202/206 South
Bedminster, NJ 07921
(908) 947-7033
corson@isr.umd.edu

Joseph Macker
Information Technology Division
Naval Research Laboratory
Washington, DC 20375
(202) 767-2001
macker@itd.nrl.navy.mil