

INTERNET-DRAFT  
[draft-ietf-manet-cbrp-spec-00.txt](#)

Mingliang Jiang  
National University of S'pore  
Jinyang Li  
National University of S'pore  
Yong Chiang Tay  
National University of S'pore  
August 1998

## Cluster Based Routing Protocol(CBRP) Functional Specification

### Status of this Memo

This document is an Internet-Draft. 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."

To view the entire list of current Internet-Drafts, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Northern Europe), ftp.nis.garr.it (Southern Europe), munnari.oz.au (Pacific Rim), ftp.ietf.org (US East Coast), or ftp.isi.edu (US West Coast).

### Abstract

Cluster Based Routing Protocol (CBRP) is a routing protocol designed for use in mobile ad hoc networks. The protocol divides the nodes of the ad hoc network into a number of overlapping or disjoint clusters in a distributed manner. A cluster head is elected for each cluster to maintain cluster membership information. Inter-cluster routes are discovered dynamically using the cluster membership information kept at each cluster head. By clustering nodes into groups, the protocol efficiently minimizes the flooding traffic during route discovery and speeds up this process as well. Furthermore, the protocol takes into consideration of the existence of uni-directional links and uses these links for both intra-cluster and inter-cluster routing.

## 1 Introduction

There are several major difficulties for designing a routing protocol for MANET. Firstly and most importantly, MANET has a dynamically changing topology due to the movement of mobile nodes which favors routing protocols that dynamically discover routes over conventional distant vector routing protocols [6], e.g. Dynamic Source Routing[4], TORA[3], ABR[5] etc. Secondly, the fact that MANET lacks any structure makes IP subnetting impossible. However, routing protocols that are flat, i.e. only one level of hierarchy, might suffer from excessive overhead when scaled up. Thirdly, links in mobile networks could be asymmetric at times. Routing protocols could make use of the uni-directional links to improve its overall performance.

CBRP has the following features:

- \* fully distributed operation.
- \* less flooding traffic during the dynamic route discovery process.
- \* explicit exploitation of uni-directional links that would otherwise be unused.

The idea of using clusters for routing in Ad hoc networks has appeared in [7],[8]. In these protocols clusters are introduced to minimize updating overhead during topology change. However, the overhead for letting each and every node maintain up-to-date information about the whole network's cluster membership and inter-cluster routing information in order to route a packet is considerable. In comparison, simpler and smaller clusters are found in [9] and [10], however, the use of these clusters is mainly for the task of channel assignment; how they can help in the routing process is not discussed.

CBRP adopts the cluster formation algorithm as proposed in [9], but unlike [9], CBRP mainly concentrates on the use of clusters in the routing process. CBRP is different from previously proposed cluster-based routing algorithms and it uses a dynamic route discovery strategy.

## 2. CBRP terminology

This section defines terms used in CBRP that do not appear in [2]:

#### \* Node ID

Node ID is a string that uniquely identifies a particular mobile node. Node IDs must be totally ordered. In CBRP, we use a node's IP

address as its ID for purposes of routing and interoperability with fixed networks.

#### \* Cluster

A cluster consists of a group of nodes with one of them elected as a cluster head. The election procedure is described in [section 5.1](#) and 5.2. A cluster is identified by its Cluster Head ID. Clusters are either overlapping or disjoint. Each node in the network knows its corresponding Cluster Head(s) and therefore knows which cluster(s) it belongs to.

#### \* Host Cluster

A node regards itself as in cluster X if it has a bi-directional link to the head of cluster X. In such a case, cluster X is a host cluster for this node.

#### \* Cluster Head

A cluster head is elected in the cluster formation process for each cluster. Each cluster should have one and only one cluster head. A cluster head has complete knowledge about group membership and link state information in the cluster within a bounded time once the topology within a cluster stabilizes. In CBRP, each node in the cluster has a bi-directional link to the cluster head.

#### \* Cluster Member

All nodes within a cluster EXCEPT the cluster head are called members of this cluster.

#### \* Gateway Node

A node is called a gateway node of a cluster if it KNOWS that it has a bi-directional or uni-directional link to a node from another

cluster. As shown in Figure 1, node 3 and 4 are gateway nodes of cluster 1, node 6 is a gateway node of cluster 8.

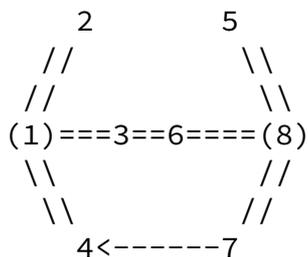


Fig. 1

\* Neighbor Table

The neighbor table is a conceptual data structure that we employ for link status sensing and cluster formation. Each entry contains the ID of a neighbor that it has connectivity with and the status of that link and the role of the neighbor (a leader or a member).

3. Physical and Link Layer Assumptions

This section lists the assumptions we made about the underlying physical and link layers when designing CBRP. Each node in MANET is equipped with a transceiver. In general, CBRP works best with omnidirectional antennas. Each packet that a node sends is broadcast into the region of its radio coverage.

4. Link/Connection Status Sensing Mechanism

In CBRP, each node knows its bi-directional links to its neighbors as well as unidirectional links from its neighbors to itself. For this purpose, each node maintains a Neighbor Table as follows:

NEIGHBOR_ID	LINK_STATUS	ROLE
neighbor 1	bi/unidirectional link to me?	is 1 a cluster head?
neighbor 2	bi/unidirectional link to me?	is 2 a cluster head?

```

+-----+-----+-----+
| ...
+-----+-----+-----+
| neighbor N | bi/unidirectional link to me? | is N a cluster head?|
+-----+-----+-----+

```

Each node broadcasts its Neighbor Table periodically in a HELLO message as shown below every HELLO\_INTERVAL.

```

+-----+-----+-----+
| MY_OWN_ID | MY_MEMBERSHIP_STATUS |
|           | (CLUSTER_HEAD/CLUSTER_MEMBER/UNDECIDED) |
+-----+-----+-----+
| Neighbor Table |
+-----+-----+-----+

```

The first field specifies the ID of the sender. The second field tells whether the sender is a cluster head, cluster member or undecided. ("undecided" means a node is still in search of its host

cluster)

Upon receiving a HELLO message from its neighbor B, node A modifies its own Neighbor Table as follows:

1. It checks if B is already in the Neighbor Table, if not, it adds one entry for B.
2. If B's Neighbor Table contains A, A marks the link to B as bi-directional in the relevant entry.
3. If B is cluster head, marks B as a cluster head in the entry.

Each entry in the Neighbor Table is associated with a timer. A table entry will be removed if a HELLO message from the entry's node is not received for a period of  $(HELLO\_LOSS+1)*HELLO\_INTERVAL$ , allowing HELLO\_LOSS consecutive HELLO messages to be lost from that node.

When a nodes' neighborhood topology stabilizes, each node will have complete knowledge of all the nodes that it has a bi-directional link to and all the nodes that have a uni-directional link to it within a bounded time. However, a node would not know to whom it has a uni-directional link.

Note that the 2nd and 3rd column of the neighbor table need not be sent out for the purposes of link sensing, however, they are there mainly for cluster formation and other functionality of the protocol.

## 5. Protocol Operation

The operations of CBRP are entirely distributed. According to the functionality, CBRP could be viewed as a combination of the three following sub-functions which will be discussed below.

### 5.1 Cluster Formation

The Cluster Formation algorithm is a simple "lowest ID" clustering algorithm in which the node with a lowest ID is elected as the Cluster Head [9].

A node uses the information obtained from the HELLO message for Cluster Formation. A node that has the lowest ID among all its bi-directionally linked neighbors (a node that has given up the role as a Cluster Head is not counted, i.e. a cluster member node). The new Cluster Head will change the first field in its subsequently broadcast HELLO messages from "undecided" to "cluster head" thereafter.

A cluster head regards all the nodes it has bi-directional links to as its member nodes. A node regards itself as a member node for a particular cluster if it has a bi-directional link to the corresponding cluster head. Note that a member node may hear from several cluster heads and therefore have several host clusters.

### 5.2 Cluster Maintenance

As clusters are identified by their respective cluster heads, we would like to have the cluster heads change as infrequently as possible. We use the following rules for changing cluster head, as described in [10].

1. A non-cluster head never challenges the status of an existing cluster head, i.e. if X is a non-cluster head node with a

- bi-directional link to cluster head Y, X does not become a cluster head even if it has an ID lower than Y's.
2. Only when two cluster heads move next to each other (i.e. there is a bi-directional link between them), one of them loses the role of cluster head.

Therefore, our exact algorithm is not a strict "lowest ID" clustering algorithm. The cluster maintenance procedure are discussed under three subsections:

#### \* Node Removal

A node X is removed from a host cluster either because it loses the bi-directional links to the cluster head, or because of node failures. In either case, the cluster head and node X will time out the corresponding entries in their Neighbor Table so that the cluster information will be updated.

#### \* Node Addition

A member node is added to a new cluster when it discovers a bi-directional link to the respective cluster head even if the cluster head has a higher ID, which is consistent with rule 1. The node will know its new host cluster and the new host cluster head will know its new member through the updated Neighbor Table.

When a node is switched on, its MY\_MEMBERSHIP\_STATUS field in the HELLO message is initially UNDECIDED for a period of UNDECIDED\_PD. If it discovers that it has a bi-directional link to a cluster head, it modifies this field as CLUSTER\_MEMBER. If there is no such bi-directional links to any cluster head, it takes the role as a cluster head and indicates this in the subsequent HELLO messages it sends.

#### \* Cluster Head Contention

When two Cluster Heads move next to each other (till they have a bi-directional link in-between), one of them must give up its role as cluster Head. As a result, whenever a cluster head hears the HELLO message from another cluster head indicating a bi-directional link in between, it will compare its own ID with that of the other cluster head's. The one with a smaller ID will continue to act as cluster

head. The one with a bigger ID gives up its role as cluster head and changes from "cluster head" to "member" in its subsequent HELLO messages. This might trigger the formation of other new clusters.

### [5.3](#) Routing Considerations

Routing in the CBRP is based on source routing, which is similar to [4]. Therefore, it can be viewed as consisting of 3 phases: route discovery, packets routing and stale route removal, of which packet routing is trivial.

Cluster structure is exploited to minimize the flooding traffic during route discovery phase. Furthermore, some uni-directional links are discovered and used which increases network connectivity. This is discussed in the Gateway Discovery section.

#### [5.3.1](#) Gateway Discovery

Cluster X and cluster Y are said to be bi-directionally linked, if any node in cluster X is bi-directionally linked to another node in cluster Y, or if there is a pair of opposite uni-directional links between any 2 nodes in cluster X and cluster Y respectively.

Cluster X is said to be uni-directionally linked to cluster Y if any node in cluster X is uni-directionally linked to any other node in cluster Y. Y is called X's upstream uni-directionally linked neighboring cluster.

By Gateway Discovery, a cluster head for cluster X will obtain the information about cluster X's bi-directionally and upstream uni-directionally linked neighboring clusters.

For this purpose, a Cluster Adjacency table is kept at each node which is formatted as follows:



### 5.3.2 Route Discovery

Route Discovery is the mechanism whereby a node S wishing to send a packet to a destination D obtains a source route to D. Similar to other MANET protocols [4] [1], the way S finds a route(or multiple routes) to D is also done by flooding, however, because of the clustering approach the number of nodes that are disturbed are much less in general.

Essentially in Route Discovery, cluster heads are flooded in search for a source route. To perform Route Discovery, the source node S sends out a Route Request Packet (RREQ), with a recorded source route listing only itself initially. Any node that forwards this packet will append its own ID in this RREQ. Each node forwards a RREQ packet only once and it never forwards it to a node that has already appeared in the recorded route. In CBRP, the RREQ will always follow a route with the following pattern to reach destination D:

S,CH1,G1,CH2,G2,G3,CH3 ..... D

A detailed description of how this is achieved is presented below. Source always unicasts RREQ to its cluster head, say CH1. Each cluster head will unicast RREQ to each of its bi-directionally linked neighbors which have not yet appeared in the recorded route through the corresponding gateway. This process continues until the target is found or until another node that can supply a route to the target is found.

When the target of the Request, node D, receives the RREQ, D may choose to memorize the reversed source route to S. Node D then copies the recorded source route into a Route Reply packet(RREP) which it then sends back to the initiator of the Route Request (e.g., node S) by reversing the recorded route and putting it in the IP header of the Route Reply packet. The recorded route gives the complete information about the SEQUENCE OF CLUSTERS source should traverse in order to reach destination D. While forwarding the Route Reply, intermediate cluster heads modifies the IP header of the packets, substitute the inter-cluster incoming links to inter-cluster outgoing links. Each intermediate cluster head also modifies the recorded route in the Route Reply packet to optimize the recorded route as much as possible using its knowledge of the cluster topology and inter-cluster gateway information. An example of such optimization is to connect two gateway nodes by an intra-cluster link that does not go through the cluster head.

All source routes learned by a node are kept in a Route Cache, which is used to further reduce the cost of Route Discovery. When a node

wishes to send a packet, it examines its own Route Cache and performs Route Discovery only if no suitable source route is found in its cache.

### [5.3.3](#) Route Removal

A source route is removed from S if the network topology has changed such that S can no longer use the route to D because a hop along the route no longer works. If S still wants to communicate with D, it can invoke Route Discovery again to find a new route.

## [6.](#) Discussions and Implementation Considerations

This section discusses some of the problems that we have encountered during the design of CBRP. In particular, they are related to the use of uni-directional links in routing.

### [6.1](#) ARP and Uni-directional links

In a MANET context, links between 2 nodes can be bi-directional and uni-directional. However, when the link layer does MAC-layer address based packet filtering, (which most current technologies do), special care has to be taken for ARP for uni-directional links. For example, when there is a uni-directional link from node A to node B as shown in Figure 2, node A should not rely on the conventional ARP protocol to resolve node B's MAC layer address, because node B's ARP reply will never reach node A directly.

A---->B

Fig. 3

CBRP is able to use uni-directional links. For an intra-cluster uni-directional link, the upstream node can be informed by its cluster head of the MAC-layer address of the downstream node. For a inter-cluster uni-directional link, as shown in Figure 1, node 3(5) will know node 4(6)'s MAC-layer address by the process of gateway

discovery.

## 6.2 Rate of Stale Route Discovery and Uni-directional Links

In general (not pertaining to CBRP), when uni-directional links are used, discovery of stale routes can be slow. As shown in Figure 3, when the link between 1 and 2 breaks, node 1 will not be aware until a message comes from 2 by way of 3,4,5,6. This observation justifies CBRP's use of inter-cluster uni-directional links only between 2

clusters.

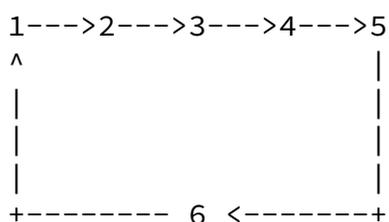


Fig. 4

## 7. Future Directions

Cluster based routing protocols (CBRP) is the first step towards a more scalable solution using clustering approach. It also suggests how uni-directional links in Ad hoc networks can be used in routing and the reveals problems resulting from such use.

Several questions remain to be answered in CBRP,

1. How collisions can be avoided or handled effectively. Shall we have spatial reuse of the channels among different clusters?
2. Should clusters have native support for QoS as in [10]?
3. Should clusters be made bigger than two hops diameter? If so, what criteria should be used to form a bigger cluster? Or, should we use a hierarchy of clusters? Will the resulted more complex cluster formation and maintenance procedure offset the advantage of having a bigger size?

To give satisfactory answers to these interesting questions will be our future directions in refining CBRP.

Jiang, Li, Tay

[Page 11]

---

INTERNET-DRAFT

CBRP Functional Specification

August 1998

#### References

- [1] Perkins, C.E., "Ad-Hoc On-Demand Distance Vector Routing", MILCOM'97 panel on Ad-Hoc Networks, Monterey, CA, November 3, 1997.
- [2] Perkins, C.E., "Mobile Ad Hoc Networking Terminology", [draft-ietf-manet-term-00.txt](#), work in progress.
- [3] Corson, M.S., and Ephremides, A., "A Distributed Routing Algorithm for Mobile Wireless Networks", ACM/Baltzer Journal on Wireless Networks, January 1995.
- [4] Johnson, D.B., and Maltz, D.A., "Dynamic Source Routing in Ad-Hoc Wireless Networks", Mobile Computing, T.Imielinski and H.Korth, editors, Kluwer, 1996.
- [5] Toh, C.K., "A Novel Distributed Routing Protocol To Support Ad-Hoc Mobile Computing", International Phoenix Conference on Computers and Communications (IPCCC'96), pg 480-486, March 1996.
- [6] Hedrick, C., "Routing Information Protocol", [RFC 1058](#), June 1988.
- [7] Das, B., Raghupathy, S, Vaduvur, B, "Routing in Ad Hoc Networks

Using a Spine", Proceedings of 6th International Conference on Computer Communications and Networks, Las Vegas, USA, September, 1997.

- [8] Krishna, P., Vaidya, N.H., Chatterjee, M., Pradhan, D.K., " A Cluster-based Approach for Routing in Dynamic Networks", Proceedings of the Second USENIX Symposium on Mobile and Location-Independent Computing, 1995.
- [9] Gerla, M., Tsai, T.C., "Multiculster, mobile, multimedia radio network", ACM/Balzer Journal of Wireless Networks, 1995.
- [10] Chiang, C.C., Wu, H.K., Liu, W., Gerla, M., "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel", The Next Millennium, the IEEE SICON, 1997.

This work was supported in part by National University of Singapore ARF grant RP960683.

#### Authors' Information

Jiang Mingliang  
Mobile Computing Lab  
School of Computing  
National University of Singapore  
Singapore 119260  
Email: jiangmin@comp.nus.edu.sg

Li Jinyang  
Mobile Computing Lab  
School of Computing  
National University of Singapore  
Singapore 119260  
Email: lijinyan@comp.nus.edu.sg

Tay Yong Chiang  
Department of Mathematics

National University of Singapore  
Singapore 11920  
Email: mattyc@leonis.nus.edu.sg