IETF MANET Working Group INTERNET-DRAFT Expiration: December 17, 2002 Mario Gerla, UCLA Xiaoyan Hong, UCLA Guangyu Pei Rockwell Scientific Company June 17, 2002

#### Fisheye State Routing Protocol (FSR) for Ad Hoc Networks

<<u>draft-ietf-manet-fsr-03.txt</u>>

Status of This Memo

This document is an Internet-Draft and is subject to all provisions of <u>Section 10 of RFC2026</u>.

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/lid-abstracts.txt

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

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 Fisheye State Routing (FSR) algorithm for ad hoc networks introduces the notion of multi-level "scope" to reduce routing update overhead in large networks. A node stores the Link State for every destination in the network. It periodically broadcasts the Link State update of a destination to its neighbors with a frequency that depends on the hop distance to that destination

Gerla, Hong and Pei

[Page 1]

(i.e., the "scope" relative to that destination). State updates corresponding to far away destinations are propagated with lower frequency than those for close by destinations. From state updates, nodes construct the topology map of the entire network and compute efficient routes. The route on which the packet travels becomes progressively more accurate as the packet approaches its destination. FSR resembles Link State routing in that it propagates Link State updates. However, the updates are propagated as aggregates, periodically (with period dependent on distance) instead of being flooded individually from each source. FSR leads to major reduction in link O/H caused by routing table updates. It enhances scalability of large, mobile ad hoc networks.

## Contents

Stat	us of This Memo	•	·	·	·	•	•	•	·	·	1
Abst	ract										<u>1</u>
<u>1</u> .	Introduction										<u>3</u>
<u>2</u> .	Changes										<u>4</u>
<u>3</u> .	Terminology										<u>4</u>
	<u>3.1</u> . General Terms	•	•	·	·	·	·	·	·	•	<u>4</u>
	<u>3.2</u> . Specification Language	·	·	·	·	•	·	·	·	·	<u>4</u>
<u>4</u> .	Protocol Applicability										<u>5</u>
	<u>4.1</u> . Networking Context										<u>5</u>
	4.2. Protocol Characteristics and Mechanisms	5			•	•		•			<u>5</u>
<u>5</u> .	Protocol Overview										7
<u>6</u> .	Fisheye Technique										7
	<u>6.1</u> . Fisheye Scope										<u>8</u>
	<u>6.2</u> . Fisheye State Routing Exchange Scheme .	•	·	·	·	·	·	·	·	·	<u>8</u>
<u>7</u> .	Protocol Operation										<u>9</u>
	7.1 Contents of Tables										<u>9</u>
	<u>7.1.1</u> Neighbor List	•				•					<u>9</u>
	<u>7.1.2</u> Topology Table	•				•					<u>10</u>
	7.1.3 Routing Table	•		•	·	•	•		•	•	<u>10</u>
	<u>7.2</u> . Link State Message	•		•	·	•	•		•	•	<u>10</u>
	<u>7.2.1</u> . Description of the fields	•	•	·	·	·	·	·	·	·	<u>11</u>
	<u>7.3</u> . Link State Message Processing	•	•	·	·	·	·	·	·	·	<u>12</u>
	7.3.1 Originating Link State Message										12

7.3.2 Receiving Link State	Message					<u>12</u>
<u>7.4</u> . Link Breakage						<u>13</u>
<u>7.5</u> . Routing Table Calculation .		 •		•		<u>13</u>

Gerla, Hong and Pei

[Page 2]

INTERNET-DRAFT Fisheye State Routing Protocol June 17, 2002

<u>8</u> .	Data Packet Forwarding
<u>9</u> .	Considerations about Routing Inaccuracy
<u>10</u> .	Configuration Parameters
Ackı	nowledgments
Ref	erences
Cha	r's Address
Autl	nors' Addresses

## **1**. Introduction

This document describes the Fisheye State Routing Protocol (FSR) [1, 2] developed by the Wireless Adaptive Mobility (WAM) Laboratory [7] at University of California, Los Angeles. Fisheye State Routing is a table-driven routing protocol which is adapted to the wireless ad hoc environment. It provides an implicit hierarchical routing structure. Through updating link state information with different frequencies depending on the fisheye scope distance, FSR scales well to large network size and keeps overhead low without compromising route computation accuracy when the destination is near. The routing accuracy of FSR is comparable with an ideal Link State scheme. By retaining a routing entry for each destination, FSR avoids the extra work of "finding" the destination (as in on-demand routing) and thus maintains low single packet transmission latency. As mobility increases, routes to remote destinations become less accurate. However, when a packet approaches its destination, it finds increasingly accurate routing instructions as it enters sectors with a higher refresh rate. As a results, FSR is more desirable for large mobile networks where mobility is high and the bandwidth is low. By choosing proper number of scope levels and radius size, FSR proves to be a flexible solution to the challenge of maintaining accurate routes in ad hoc networks.

The following properties of FSR highlight its advantages.

- \* Simplicity
- \* Usage of up-to-date shortest routes
- \* Robustness to host mobility

\* Exchange Partial Routing Update with neighbors

Gerla, Hong and Pei

[Page 3]

\* Reduced Routing Update Traffic

## 2. Changes

Major changes from version 02 to version 03:

- Added descriptions of operations on the lastHeardTime variable in the topology table.
- Editorial changes.

Major changes from version 01 to version 02:

- Added operations to exclude those entries whose propagation will not cause topology update at its neighbors.
- Editorial changes.

Major changes from version 00 to version 01:

- Update of "Status of This Memo".

# 3. Terminology

#### 3.1. General Terms

This section defines terminology used in FSR.

node

A MANET router that implements this Fisheye State Routing protocol.

### neighbor

Nodes that are within the radio transmission range.

fisheye scope

A zone that is bounded by hop distances.

# **<u>3.2</u>**. 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 <u>RFC 2119</u> [8].

Gerla, Hong and Pei

[Page 4]

# **<u>4</u>**. Protocol Applicability

#### **4.1**. Networking Context

FSR is best suited for large scale mobile ad hoc wireless networks, as the scope update scheme has larger advantages in reducing routing update packet size and achieve high date packet to routing packet ratio. Moreover, the fact that the route error is weighted by distance obviously reduces the sensitivity to network size.

FSR is also suited for high mobility ad hoc wireless networks. This is because in a mobility environment, a change on a link far away from the source does not necessarily cause a change in the routing table at the source.

#### **4.2.** Protocol Characteristics and Mechanisms

\* Does the protocol provide support for unidirectional links? (if so, how?)

Yes, since FSR is link state based routing protocol and directional link states can be included in the FSR update messages.

- \* Does the protocol require the use of tunneling? (if so, how?) No.
- \* Does the protocol require using some form of source routing? (if so, how?)

No.

\* Does the protocol require the use of periodic messaging? (if so, how?)

Yes. The FSR periodically exchanges partial link state information with its neighbors

\* Does the protocol require the use of reliable or sequenced packet delivery? (if so, how?)

No. As the packets are sent periodically, they need not be sent reliably.

\* Does the protocol provide support for routing through a multitechnology routing fabric? (if so, how?)

Yes. It is assumed that each node's network interface is assigned a unique IP address.

\* Does the protocol provide support for multiple hosts per router? (if so, how?)

Gerla, Hong and Pei

[Page 5]

Yes. The router that has multiple hosts can use network ID of these hosts as the address to participate FSR.

\* Does the protocol support the IP addressing architecture? (if so, how?)

Yes. Each node is assumed to have a unique IP address (or set of unique IP addresses in the case of multiple interfaces). The FSR references all nodes/interfaces by their IP address.

This version of the FSR also supports IP network addressing (network prefixes) for routers that provide access to a network of non-router hosts.

\* Does the protocol require link or neighbor status sensing (if so, how?)

No.

\* Does the protocol have dependence on a central entity? (if so, how?)

No.

\* Does the protocol function reactively? (if so, how?)

No.

\* Does the protocol function proactively? (if so, how?)

The FSR proactively maintains routing information at each Yes. node.

\* Does the protocol provide loop-free routing? (if so, how?)

Yes. As the protocol uses link state algorithm, so the routing is loop-free in a stable state. In mobile environment, each link state entry is destination sequenced which can prevent loop in routing.

\* Does the protocol provide for sleep period operation?(if so, how?)

Yes. However, this requires TDMA MAC layer support. The router can be scheduled to sleep during the idle period.

\* Does the protocol provide some form of security? (if so, how?)

No.

\* Does the protocol provide support for utilizing multi-channel, link-layer technologies? (if so, how?)

Gerla, Hong and Pei

[Page 6]

Yes, in fact the multi-channel can be used to separate routing messages from user data packets.

## 5. Protocol Overview

Fisheye State Routing is a table-driven or proactive routing protocol. It bases on link state protocol and has the ability of immediately providing route information when needed. The fisheye scope technique allows exchanging link state messages at different intervals for nodes within different fisheye scope distance, which reduces the link state message size. Further optimization allows FSR only broadcast topology message to neighbors [4] in order to reduce the flood overhead. With these optimizations, FSR significantly reduces the topology exchange overhead and scales well to large network size.

FSR routes each data packet according to locally computed routing table. The routing table uses most recent topology information. The fisheye scope message updating scheme will not lose routing accuracy for inner scope nodes. For outer scope nodes, information in routing entries may blur due to longer exchange interval, but the extra work of "finding" the destination (as in on-demand routing) is not necessary. Thus low single packet transmission latency can be maintained. At a mobile environment, this inaccuracy for remote nodes will increase. However, when a packet approaches its destination, it finds increasingly accurate routing instructions as it enters sectors with a higher refresh rate.

FSR does not trigger any control messages when a link failure is reported. Thus it is suitable for high topology change environment. The broken link will not be included in the next fisheve scope link state message exchange. Sequence number and table refreshment enables the FSR to maintain the latest link state information and loop-free in a unreliable propagation media and highly mobile network.

The protocol works independently with the IP format of packets and is a distributed protocol. It can be implemented either in network layer or in application layer. It only deals with routing table management of the network system.

## 6. Fisheye Technique

FSR uses the "fisheye" technique proposed by Kleinrock and Stevens [3], where the technique was used to reduce the size of information required to represent graphical data. The eye of a fish captures with high detail the pixels near the focal point.

The detail decreases as the distance from the focal point increases. In routing, the fisheye approach translates to maintaining accurate distance and path quality information about the immediate

Gerla, Hong and Pei

[Page 7]

neighborhood of a node, with progressively less detail as the distance increases.

### 6.1. Fisheye Scope

The fisheye scope is defined as the set of nodes that can be reached within a given number of hops. An example of a fisheye scope (at node A) of hop 2 is shown below.



Note that in this example node E can be reached through B from A with 2 hops and through F with 3 hops. Since the minimum path length is 2, E are within the fisheye scope of node A. By setting multiple hop radius, multiple level of scopes will cover the entire network.

#### **6.2.** Fisheye State Routing Exchange Scheme

FSR is functionally similar to Link State Routing (LS) in that it maintains a topology map at each node. The key difference is the way in which routing information is disseminated. In LS, the update message could consume considerable amount of bandwidth, which depends on the update period. In order to reduce the size of update messages without seriously affecting routing accuracy, FSR uses the Fisheye technique. Entries in routing table corresponding to nodes within the smallest scope are propagated to the neighbors with the highest frequency. The rest of the entries are sent out with lower frequencies. As a result, a considerable fraction of link state entries is suppressed in a typical update, thus reducing the message size. Thus by using different exchange periods for different entries in routing table, routing update overhead is reduced.

This strategy produces timely updates from near stations, but creates large latencies from stations afar. However the imprecise knowledge of the best path to a distant destination is compensated by the fact that the route becomes progressively more

accurate as the packet gets closer to destination. As the network size grows large, a "graded" frequency update plan must be used across multiple scopes to keep the overhead low.

Gerla, Hong and Pei

[Page 8]

Link state packets are generated and flooded into the network periodically or whenever a node detects a topology change. In FSR, link state packets are not flooded. Instead, they are exchanged periodically with their local neighbors only. The periodic updates act also as neighbor discovering. Each node, after hearing its neighbor's broadcast message, adds/updates its neighbor list and topology table. A node maintains a link state table based on the up-to-date information received from neighboring nodes. Sequence numbers are used for entry replacements. The sequenced periodic table update resembles the vector exchange in Destination-Sequenced Distance-Vector Routing (DSDV [5]) where the distances are updated according to the time stamp or sequence number assigned by the node originating the update. However, in FSR link states rather than distance vectors are propagated. Moreover, like in LS, a full topology map is kept at each node and shortest paths are computed using this map.

In a wireless environment, a radio link between mobile nodes may experience frequent disconnects and reconnects. The LS protocol releases a link state update for each such change, which floods the network and causes excessive overhead. FSR avoids this problem by using periodic, instead of event driven, exchange of the topology map, greatly reducing the control message overhead.

The entries to be included in the periodic broadcast message are selected according to the rule that their propagation should cause an update in topology tables of its neighbors. The benefit of this operation is large in a dense network where a node may find that all its neighbors have the same freshness of the information about another node. By excluding such entries, only the effective entries are included in the link state update message which leads to reduction in the size of the message.

## 7. Protocol Operation

This section discusses the operation of FSR routing protocol. The sending and receiving of routing packets are in the proactive nature. And the routing packets are processed separately from ordinary data packets.

### 7.1 Contents of Tables

Each node running FSR is required to maintain following tables. The required tables and suggested fields are listed below.

### 7.1.1 Neighbor List

When a node receives a link state message, it records/updates the sender's link state information in its neighbor list. If the node

Gerla, Hong and Pei

[Page 9]

does not receive link state updates from a neighbor for a NEIGHBOR\_TIMEOUT interval, the entry for that neighbor will be deleted from the neighbor list. The following information is maintained in the list for each neighbor node:

- Neighbor ID
- Latest time receiving from the neighbor

#### 7.1.2 Topology Table

Topology table records the topology information obtained from the link state message. Each destination has an entry in the table. The entry contains three parts: the destination information, its link state information and variables for selecting effective entries. Based on this table, the routing table is calculated. The distance information is maitained in the routing table and is used to classified the node to a fisheye scope. The topology table has following fields for every link state entry:

- Destination Address
- Destination sequence number
- Last heard time
- A list of its neighbors
- Previous sequence number
- A flag for "NeedToSend"

## 7.1.3 Routing Table

The routing table of FSR provides the next hop information to forward the packets for the other destinations in the network. The entries are updated when topology table is changed. The routing table has the following fields:

- Destination Address
- Next hop address
- Distance

Initially when a FSR router is powered up, it knows nothing about the network except routing information about all interfaces and hosts that are directly attached to the router.

#### <u>7.2</u>. Link State Message

Periodically, nodes broadcast the effective portion of the topology tables to their neighbors. By listening to these broadcastings, each node builds up its neighbor list. As the source address (the address originating this message) and destination address(broadcast address of the network) are already included in the IP header, the link state message given here only contains packet size and topology table entries. At very beginning, this broadcast contains only itself with empty topology table.

The proposed format of a Link State Message (assume the link cost is measured in hop distance and is 1 for each neighbor, therefore

Gerla, Hong and Pei

[Page 10]

the link cost is not included) is as follows.

Θ	1		2	3		
01234	5678901	23456	7890123	45678901		
+ - + - + - + - + - +	-+-+-+-+-+-	+ - + - + - + - + -	+ - + - + - + - + - + - + -	+-+-+-+-+-+-+-+		
P	acket Length		Rese	rved		
+ - + - + - + - + - +	-+-+-+-+-+-	+ - + - + - + - + -	+ - + - + - + - + - + - + -	+ - + - + - + - + - + - + - + - +		
Destination Address 1						
+ - + - + - + - + - +	-+-+-+-+-+-	+-+-+-+-	+ - + - + - + - + - + - + -	+ - + - + - + - + - + - + - + - +		
Destinati	on Sequence N	umber 1		N_neighbors 1		
+ - + - + - + - + - +	-+-+-+-+-+-	+ - + - + - + - + -	+ - + - + - + - + - + - + -	+-+-+-+-+-+-+-+-+		
		Neighbor A	ddress 1			
+ - + - + - + - + - +	-+-+-+-+-+-	+ - + - + - + - + -	+ - + - + - + - + - + - + -	+ - + - + - + - + - + - + - + - +		
:				:		
+ - + - + - + - + - +	-+-+-+-+-+-	+ - + - + - + - + -	+ - + - + - + - + - + - + -	+-+-+-+-+-+-+-+-+		
	Neighb	or Address	N_neighbors 1	.		
+ - + - + - + - + - +	+-					
Destination Address 2						
+-						
Destinati	on Sequence N	umber 2		N_neighbors 2		
+ - + - + - + - + - + - + - + - + - + -						
Neighbor Address 1						
+-						
:				:		
+ - + - + - + - + - +	-+-+-+-+-+-	+ - + - + - + - + -	+ - + - + - + - + - + - + -	+-		
Neighbor Address N_neighbors 2						
+-						
:				:		
:				:		
		<i>.</i>	,			

(etc)

## 7.2.1. Description of the fields

Packet Length

The length (in bytes) of the packet.

# Reserved

The bits are set to '0' and are ignored on reception.

Destination Address 1

The first destination address in the topology table.

Destination Sequence Number 1

The last sequence number received for the first destination in the past by the source of the message. N\_neighbors 1 The number of neighbors of the first destination.

Gerla, Hong and Pei

[Page 11]

Neighbor Address 1, ..., Neighbor Address N\_neighbors 1 The list of neighbors of the first destination. The number is indicated in the field N\_neighbors 1.

Destination Address 2 The second destination address in the topology table.

These fields are repeated for each entry in the topology table. The length of link state message is limited to the maximum IP packet size. In that case, multiple packets may required to broadcast all the topology entries.

#### 7.3. Link State Message Processing

#### 7.3.1 Originating Link State Message

Each node broadcasts the latest link state information to its neighbors. FSR uses different update intervals for different entries in the table according to their distance from the node. To be precise, entries corresponding to nodes that are nearby (within a predefined scope) are propagated to the neighbors more frequently than entries of nodes that are far away. When it is the time for broadcasting a node's topology table (for any broadcast frequencies), the node first times out its topology table. The lastHeardTime variable in the entries corresponding to the frequency is checked, stale entries are timed out. The time-out interval is set to be three times of the update interval. For inner scope broadcasting, timing-out is performed after the node refreshes its own entry. Then at the time for updating fisheye scope-i's topology information, the node scans the topology table to pick up corresponding nodes. The nodes whose distance to the current node is within the range of scope-i will be included in the update message if it is an effective entry. The update interval of fisheye scope level i is UpdateInterval i. If the current node will be included in the link state message, its sequence number is increased by one.

To select the effective portion of the scope-i link state message, an entry corresponding to scope-i is chosen if its flag of "NeedToSend" is true. After the message is sent, all the flags of entries of scope-i are reset to false, and the previous sequence numbers are copied with the current sequence number.

## 7.3.2 Receiving Link State Message

When a node receives a link state message, it first checks its neighbor link state list for the sender. If the sender is a new neighbor, it will insert the sender into the list. Otherwise, it will update the timestamp of the sender in the neighbor list. The node then processes the link state information contained

Gerla, Hong and Pei

[Page 12]

in the update message. For each link state entry in the packet, the following situations are considered.

- (1) If it is a new destination to the current node, a new topology table entry will be created and filled with corresponding information. The "NeedToSend" flag sets to true.
- (2) Otherwise, only the most up to date destination is copied to the local topology table. That is, if any entry in the incoming message has a larger sequence number regarding to destination j comparing with the ones stored in the node's topology table, the local entry will be replaced by the incoming one. The "NeedToSend" flag sets to true.
- (3) For a incoming destination not fitting into the previous two cases, if the information is older than the one stored in current node, i.e., its destination sequence number is less than the previous sequence number of the stored entry, the entry in the current node's topology table needs to be sent in next link state update. The "NeedToSend" flag of the entry sets to true.

In all the cases, when a topology entry is inserted or updated, the time is stamped in the lastHeardTime variable. After all incoming entries are examined, if there are any changes in the topology table, routing table is recomputed.

#### 7.4 Link Breakage

In a mobile ad hoc network, link breakage is a frequent incident. Each node uses soft state to detect link breakage, i.e., each node considers a link to be broken if it does not receive link state messages from the neighbor after a NEIGHBOR\_TIMEOUT interval. The node then deletes the neighbor from neighbor list and remove it from its link state entry in topology table.

FSR does not rely on feedback from MAC. If MAC can provide Feedback in case of link failure, FSR will use the report to update the neighbor table and provide more up-to-date routing information. The operation is the same as above. This faster discovery can result in a better performance.

# 7.5. Routing Table Calculation

Whenever there are changes in the topology table, the routing table will be recomputed. The topology table is checked to remove stale entries prior to the calculation. Based on the latest topology table, the Dijkstra's algorithm [6] is performed to find the shortest paths from current node to all the destinations known to it, i.e., in the topology table. Old routing table is replaced

Gerla, Hong and Pei

[Page 13]

completely by the newly computed next hop information.

FindSP(i) is the algorithm used to create a shortest path tree rooted at i. In this documentation, the procedure is based on the Dijkstra's algorithm with modifications so that the next hop (denoted as table NEXTi()) and the distance (denoted as table Di()) are computed in parallel with the tree reconstruction.

At node i, FindSP(i) initiates with  $P = \{i\}$ , then it iterates until P = V, where V is the set of all nodes. In each iteration, it searches for a node j such that node j minimizes the value of (Di(k) + weight(k,j)), for all j and k, where j belongs to  $\{V - P\}$ , k belongs to nodes in topology table and weight(k,j) does not equal to infinite. Once node j is found, P is augmented with j, D(j) is assigned to D(k) + weight(k,j) and NEXTi(j) is assigned to NEXTi(k). That is, as the shortest path from i to j has to go through k, the successor for i to j is the same successor for i to k.

A weight function, weight(), is used to compute the distance of a link. Since minimum hop shortest path is considered in this stage, this weight function simply returns 1 if two nodes have direct connection, otherwise, it returns 0. This weight function may also be replaced with other functions for routing with different metrics. For instance, a bandwidth function can be used to realize a QoS routing.

#### 8. Data Packet Forwarding

As the routing table is calculated in background, hop by hop data packet forwarding is very straight forward. The source node or any intermediate nodes retrieve the destination from the data packet, and look at their routing tables. If the route is known, i.e., there is an entry for the destination, the data packet is sent to the next hop node. This procedure repeats until the packet finally reaches the destination.

#### 9. Considerations about Routing Inaccuracy

FSR only has inaccurate routing information about remote nodes. This inaccuracy is affected by the route update interval. The longer the update interval, the less accurate the routing information. However, several features of FSR reduces the routing inaccuracy. For large network size, as the route error is weighted by distance, the sensitivity to network size is largely reduced. Thus, receiving updates about far away nodes at low frequency will not significantly affect the routing accuracy. The same mechanism applies to mobility, i.e., the progressive accurating of the routing path reduces the impact from mobility.

Also, increasing the scope radius will improve the accuracy. But

Gerla, Hong and Pei

[Page 14]

using larger radii will increase the routing update packets which causes more control overhead.

## **10.** Configuration Parameters

This section gives default values for some important parameters associated with the FSR protocol operations. Readers should take these values as an example for the design of their own network scenario. Different values of these parameters may affect the performance of the protocol. The following values define a network with two level fisheye scopes. The inner scope is bounded by hop distance 2. Two different update frequencies are used for the scopes, they are called IntraScope\_Interval and InterScope\_Interval for inner and outer scope respectively.

Parameter Name	Value					
Number of Scope	2					
IntraScope_Interval	5 second					
IntreScope_Interval	15 seconds					
NEIGHBOR_TIMEOUT	15 seconds					

#### Acknowledgments

This work was supported in part by NSF under contract ANI-9814675, in part by DARPA under contract DAAB07-97-C-D321.

## References

[1] G. Pei, M. Gerla, and T.-W. Chen, "Fisheye State Routing: A Routing Scheme for Ad Hoc Wireless Networks", Proceedings of ICC 2000, New Orleans, LA, Jun. 2000.

[2] G. Pei, M. Gerla, and T.-W. Chen, "Fisheye State Routing in Mobile Ad Hoc Networks", Proceedings of Workshop on Wireless Networks and Mobile Computing, Taipei, Taiwan, Apr. 2000.

[3] L. Kleinrock and K. Stevens, "Fisheye: A Lenslike Computer Display Transformation," Technical report, UCLA, Computer Science Department, 1971.

[4] T.-W. Chen and M. Gerla, "Global State Routing: A New Routing Scheme for Ad-hoc Wireless Networks," In Proceedings of IEEE ICC'98, Atlanta, GA, Jun. 1998, pp. 171-175.

[5] C.E. Perkins and P. Bhagwat, "Highly Dynamic Destination-

Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," In Proceedings of ACM SIGCOMM'94, London, UK, Sep. 1994, pp. 234-244.

Gerla, Hong and Pei

[Page 15]

- [6] R. Sedgewick, Weighted Graphs, chapter 31, Addision-Wesley, 1983.
- [7] UCLA Wireless Adaptive Mobility (WAM) Laboratory. http://www.cs.ucla.edu/NRL/wireless
- [8] S. Bradner. Key words for use in RFCs to Indicate Requirement Levels. RFC 2119, March 1997.

Chair's Address

The MANET Working Group can be contacted via its current chairs:

M. Scott Corson Flarion Technologies, Inc. Bedminster One 135 Route 202/206 South Bedminster, NJ 07921 USA

Phone: +1 908 947-7033 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:

Mario Gerla 3732F Boelter Hall Computer Science Department University of California Los Angeles, CA 90095-1596 USA Phone: +1 310 825-4367 Fax: +1 310 825-7578 Email: gerla@cs.ucla.edu

Xiaoyan Hong

Gerla, Hong and Pei

[Page 16]

June 17, 2002

3803F Boelter Hall Computer Science Department University of California Los Angeles, CA 90095-1596 USA Phone: +1 310 825-4623

Fax: +1 310 825-7578 Email: hxy@cs.ucla.edu

Guangyu Pei Rockwell Scientific Company 1049 Camino Dos Rios P.O. Box 1085 Thousand Oaks, CA 91358-0085 USA

Phone: +1 805 373-4639 Fax: +1 805 373-4383 Email: gpei@rwsc.com Gerla, Hong and Pei

[Page 17]