Active Route-Maintenance Protocol for Signal-Based Communication Path in Ad-Hoc Networks

Chih-Yung Chang*, Chao-Tsun Chang+, Shin-Chih Tu* and Tsung-Tien Hsieh*
*Department of Computer and Information Science,
Aletheia University
changcy@email.au.edu.tw
+Department of Computer Science and Information Engineering,
National Central University

Abstract

Ad-hoc wireless networks consist of mobile hosts in a network without of base stations and characterized by a highly dynamic network topology. The network topology changes frequently due to host migration, signal interference and power outages. This makes routing maintenance challenging for routing protocol design. In this paper, we present an active route maintenance protocol to prevent the existing route from disconnecting. By monitoring the status of signal strength and stability of individual hosts, an active node that causes the route to have a weak connection will actively issue a route reestablishing process in which the active host selects one of its neighbors as the candidate to partially reconstruct the weak connection. An enhanced active route maintenance protocol is also proposed to reduce the probability of route breakage and enhance the route efficiency. Performance simulation results show that the proposed route maintenance protocol effectively reduces both the overhead and route maintenance, as well as alleviating the route-break phenomenon.

1. Introduction

Ad-hoc wireless networks consist of mobile hosts with high mobility, enabling users to exchange information in any location. Mobile hosts in MANET (Mobile Ad-Hoc Networks) can communicate directly with neighboring hosts through the shared wireless media, but communication with non-neighboring hosts requires distributed routing protocols. Unlike static networks, ad-hoc networks have no spatial hierarchy and suffer from frequent link failures. Several recent papers have developed Ad-Hoc networks routing protocol based have developed Ad-Hoc networks routing protocol based on on-demand route discovery philosophy. Routes to a destination are sought only if the node has data to send to that destination. The Dynamic Source Routing (DSR) proposed in [1] uses broadcasts to propagate the route-search. When a host other than the destination receives a route-search packet, it appends its ID and rebroadcasts the packet. As soon as the request reaches the destination, a complete route is listed in the packet. The AODV[6] protocol broadcasts the RREQ packet from source host to destination. When the destination host receives the RREQ packet, it returns a RREP to source host. Hosts on the route record the host ID on the next hop and send a "Hello" message to check the status of neighboring host.

The Associativity-Based Routing (ABR) protocol [4] uses beacon to measure the stability of the neighboring host. The stability measurement will be appended to search packet so that the destination host can construct a stable route according to the stability measurement. The SSA routing protocol proposed in [2] utilizes the information available at the link level to choose routes. However, message flooding issued by source or an intermediator is needed to reestablish a new route after the original route becomes disconnected.

A route in Ad-hoc networks may suffer from route break due to host migration, signal interference or power outages. Thus, most of the previous research[1][2][4] performed a route reconstruction process at the occurrence of route disconnection. The reconstruction process establishes another route by flooding messages from source to destination[5], which causes not only heavy traffic but also long delays for route recovering. In contras to previous works, the route maintenance protocol proposed in this paper determines an active node that is predicted to cause a weak connection in the future. By monitoring the signal strength and stability of neighboring hosts, two end hosts of a weak link (or unreliable link) will perform an *active node determination process* to determine whether or not the host itself is an active node. The active node then actively issues a *local route*

The authors would like to thank the Ministry of Education, ROC, for partially supporting this research under Contract No. 89-H-FA07-1-4 (Learning Technology).

1531-2216/01 $10.00 © 2001 IEEE

reestablish process to recover the weak link before the route is broken.

Compared to previous methods, the *local route reestablishing process* is automatically performed by a small set of hosts before the occurrence of route disconnection. A neighboring host that has stable links will be selected to establish another local route instead of the unreliable link. The proposed *Active Routing Maintenance Protocol* (*ARMP*) has the advantages of small set of message exchange, fast reconstruction of another local and stable route, and no flooding. To enhance the efficiency of routing path maintained by *ARMP* protocol, we develop an *Enhanced Active Routing Maintenance Protocol* (*EARMP*) approach. Performance simulation results show that the proposed ARMP and EARMP protocols reduce the probability of route break and the flooding overhead for route reconstruction.

2. Overview of Active Routing Maintenance Protocol(*ARMP*)

The proposed *ARMP* establishes another route in an active manner before the occurrence of route disconnection. The *ARMP* mainly consists of two phases, the *Active Node Determination Process* (*ANDP*) phase and the *Local Route Reestablishing Process* (*LRRP*) phase. As soon as the state of a link is changed from reliable to unreliable, one of the end hosts of this link will be determined as the active node in the *ANDP* phase. This end host detects whether or not it will cause a route disconnection in the near future by measuring its link status every tick. The active node then creates an *LRRP* process to recover the weak link.

In the *LRRP* phase, a neighboring host will be selected to replace the active node so that the route can be recovered to a reliable state. The host node selected in *LRRP* should have a stable link status on links connecting to the previous and the successive hosts of the active node. This ensures that the selected node can take over the function of the active node and thus the new local route can be established before the original route is disconnected. *ARMP* reduces the opportunity of route disconnection and thus eliminates message flooding, as found in most of route maintenance protocols in previous works.

Fig. 1 shows an ad-hoc network containing six mobile hosts. In Fig. 1, the dotted line with an arrow represents the moving direction of hosts. Host *A* has a routing path $p=ABCEF$ and is transmitting a message to destination host *F*. On the route, each host records the previous and successive hosts in its cache table. For example, host *B* knows that the previous and successive hosts on the route are hosts *A* and *C*, respectively. Assume that host *C* moves and causes link $\overline{BC}$ to be unstable. By measuring the signal strength of link $\overline{BC}$, both hosts *B* and *C* will detect that link $\overline{BC}$ becomes unreliable and may break in the near future. Both host *B* and C will perform the *Active Node Determination Process* to determine an active node. By evaluating the variation of signal strength, host *C*, for example, identifies that it is an active host.

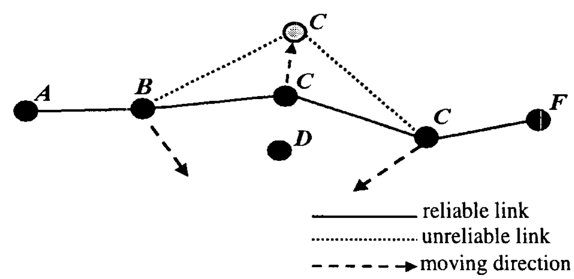

Figure 1. An example of route from source host *A* to destination host *F*.

After the active host is determined, the next step of *ARMP* is to establish a new route before the route is break. Active host *C* creates a *Local Route Reestablishing Process*(***LRRP***) which determines another host *D*, that has reliable links $\overline{BD}$ and $\overline{DE}$, to replace host *C* and establish a new local route. In the ***LRRP*** phase, active node *C* sends LRRP packet (*previous host, successive host, my ID*)=(*B, E, C*), which indicates the host name of its previous and successive nodes, to all its neighboring hosts. As the neighboring hosts of active node receive the *LRRP* packet, they measure the status of links that they connect to the previous host and successive host of active node. For instance, host *D* receives *LRRP* packet (*B, E, C*), it measures the strength of links $\overline{BD}$ and $\overline{DE}$ and replies with the signal strength of these two links to active node *C*, if these two links are reliable. As soon as the active node receives the reply messages from its neighboring hosts, it compares the received values and determines a neighboring host, say *D*, which has the best link condition to communicate with hosts *B* and *E*. The original local route $p_{old}=BCE$ will be changed to a new local route $p_{new}=BDE$.

3. Operations of Active Routing Maintenance Protocol

1. On the route, each host monitors the signal strength of each link. It creates an *Active Node Determination Process* (*ANDP*) whenever the signal strength of one of its links is weaker than a system-defined value. This indicates that the link is in an unstable state of transmission and may cause route disconnection in the future. Any link which

has a signal strength weaker than a specific value is called an *unreliable link*.

2. In the *ANDP* phase, a host whose migration causes an unreliable link will be selected as an active node. Each end host of the unreliable link will transmit the variation of signal strength of its link to another end host. Then the two end hosts compare the variation and determine which host is the active node. The host that has a larger variation value is considered as the active node. Since the unreliable link is caused by active node, the next step is to find a candidate that has a stable links to the previous and successive hosts of active node. The candidate host thus can take over the role of active host.
3. If the active node is neither source nor destination, a candidate host will be found to replace the active node. The active node transmits the related information, stored in its cache, to the candidate and then is replaced by the candidate node. If the active node is either source or destination, a new host will be selected to add to the route, such that the unreliable link can be changed to two reliable links.

In this paper, we use signal strength as the measurement of the migration degree. Hosts with higher speeds will have large variations value in signal strength. In addition, we believe that the mobility of hosts has the locality property. That is, mobile hosts that move with a high speed at time t are also expected to move with a high speed in time t+1. This implies that the active node that causes an unreliable link will have a high probability to cause the link break in future.

To determine which host is in the active node, each end host of the unreliable link totals the variation of signal strength of all its links and compares the result with another end host. We denote the summation of variation of signal strength for all links connected to one host as *strength variation*. The host that has a larger *strength variation* value is considered as the active node.

The proposed *ARMP* technique utilizes signal strength information which is stored in a cache of host and could be obtained by applying an SSA routing protocol. Two tables stored in the cache of each host are illustrated in follows.

Table I. Connectivity information stored in host *C*

Source ID	Destination ID	Previous ID	Successive ID
A	*D*	*B*	*D*
E	*G*	*E*	*F*
...	...	...	...

In Table I, connectivity information of each route such as source host, destination host, previous host, and successive host are stored in a cache table of each host. Whenever this host becomes an active node, it can send message (*previous node, successive node, my ID*) to all neighbors and request the status of links from each of its neighbors to both the previous and successive hosts.

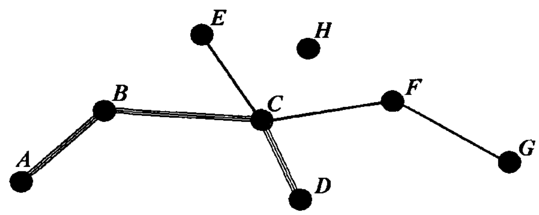

Figure 2. The Ad-hoc network topology

Table II records the signal strength of links. For each neighboring host, information such as the neighboring host ID and the signal strength of link connecting host and its neighbor is recorded. For example, in Fig. 2, host C has neighbors B, D, E, F, and H. For each neighbor of host C, say host B, we record the signal strength of link $\overline{BC}$, as denoted by S_{BC} in Table II. Since the mobility causes frequent changes in the ad-hoc network topology, we need to monitor the variation of signal strength in every tick. This value is varied every tick so that the up-to-date value denotes the speed relative to its neighbors. Whenever a link of host C becomes unreliable, node C issues an *active node determination process* so that both ends of the unreliable link can evaluate the sum of variation of signal strength for all links recorded in Table II. The host with maximum value of the sum of variation of signal strength will be considered as the active node which is moving fastest and causing the link unreliable. The active node will send its previous and successive hosts IDs to all its neighbors to find a candidate for establishing another local route before that the unreliable link breaks.

Table II. Signal strength stored in host *C*.

Neighboring host ID	Signal Strength	Variation per tick
B	S_{BC}	V_{BC}
D	S_{BC}	V_{BC}
...	...	...

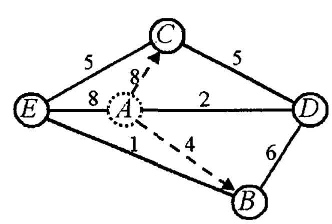

Figure 3. Using signal strength to measure the speed of a mobile host and determine a host as active node.

By applying SSA [3] routing protocol to establish a route, information listed in Tables I and II can be obtained and kept in a cache of each host. Each host computes the value of *variation per tick* at every tick by evaluating the difference of signal strength between ticks t and $t+1$. This value is considered as the degree of mobility. The mobility of the end host that has a larger value of *variation per tick* is considered as the source reason for causing a link to be unreliable. Thus, this end host will actively prevent the route disconnection by issuing a *local route reestablishing process*, and thus it is called an *active node*. We use Fig. 3 to show the computation of *variation per tick* performed in each host. Fig. 3 shows a snapshot at tick $t+1$. At tick t, assume that the location of hosts B and C is position A, as represented by a dotted circle in Fig. 3. The value labeled on the edge denotes the signal strength of this link. The neighboring hosts of B and C are hosts D and E. The *variation per tick* of link $\overline{BE}$ is evaluated by

$$V_{BE} = \left|S^{t}_{BE} - S^{t+1}_{BE}\right| = |8-1| = 7$$

where S^{t}_{BE} denotes the signal strength of link $\overline{BE}$ at tick t. The mobility of host B is thus can be evaluated by

$$\sum_{n\in neighbors} V_{BN} \text{ /number of neighbors} = (V_{BE} + V_{BD})/2 = (|8-1| + |2-6|)/2 = 5.5.$$

Similarly, the mobility of host C is evaluated by

$$(|8-5| + |2-5|)/2 = 3.$$

This indicates that host B has a larger mobility than host C. The same argument can be applied to two end hosts of a link. That is, whenever link $\overline{BE}$ becomes unreliable, hosts B and E can evaluate their mobility and then compare it with another one. The end host with larger mobility will be considered as the active node.

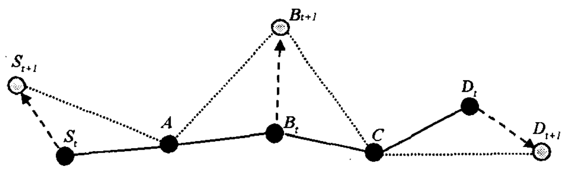

Figure 4. Three main reasons causing an unreliable link.

Fig. 4 illustrates the three main reasons for causing an unreliable link. Unreliable links may be classified into three types, as shown by the dotted lines in Fig. 4. The source and destination hosts are S and D, respectively. According to the role of the active node is source host, intermediate host, or the destination host, we consider the unreliable link as three different types: source type, intermediate type, and destination type, respectively. For instance, host B is in position B_t at tick t. At tick $t+1$, it moves to position B_{t+1} along the direction of dotted arrow. The movement causes that links $\overline{AB}$ and $\overline{BC}$ become unreliable links. Thus, unreliable links $\overline{AB}$ and $\overline{BC}$ is belonging to intermediate type. The other two types of unreliable links, $\overline{SA}$ and $\overline{CD}$, also can be found in Fig.4.

For preventing the source type of unreliable link from breaking, a source node issues the *local route reestablish process* to find a new host as the intermediator of source and the successive host of source. If the unreliable link is caused by the high mobility of an intermediate host, as classified by intermediate type, the intermediate host will find another host to take its place. Two reliable links are needed in place of the unreliable link. Similarly, to prevent an unreliable link of the destination type from breaking, our procedure will find a new host as the intermediator of destination and previous host of destination.

Whenever a host detects that there is a need to choose a candidate for establishing new local connection for the unreliable link, it utilizes information stored in Tables I and II to prevent the route status from deteriorating. There is a standard value, denoted by s, of signal strength initially given by ad-hoc network system and stored in each host. A link whose signal strength is less than s is considered as the unreliable link.

Whenever a host detects that the signal strength of one of its links is lower than s, it will issue the *active node determination process* to determine whether or not it is an active node. By totaling the variation signal strength stored in Table II, a host can measure the speed or mobility and send this value to another end host of the unreliable link. Of these two ends, the one that has a larger mobility will be considered as the active node. Since the unreliable link is caused by the high mobility of the active node, there is a need to find another candidate for active node. Before the link breaks, the active node is responsible for actively finding a candidate node which has reliable links to communicate with the previous and successive hosts of active node.

4. Enhanced-ARMP

In this section, we introduce the ***EARMP*** (Enhanced-ARMP) protocol which enhance the successful probability in *LRRP(local route reestablishing process)*.

Finding Candidates

There are some unknown factors such as the host density, the average host speed and obstacles. These factors may cause difficulty in finding an active node candidate when a weak link appears. Here, we propose an enhanced *ARMP* protocol to improve the successful probability for finding the candidate. Thus, the route is break-resistant.

The main idea of improving the probability of finding the candidate is to allow two or more candidates to replace the active node. As shown in Fig. 5, when the active host B moves from location B_t to location B_{t+1}, then link $\overline{B_{t+1}C}$ is in the unreliable state. However, none of neighbors E, F, and G satisfy the candidate constraint that the candidate host should have two reliable links, one connecting to the predecessor of active node and another connecting to the successor of active node. This would cause the route to break. However, links $\overline{AE}$ and $\overline{FC}$ are both reliable. These two reliable link connects the predecessor and successor of active node. Since link $\overline{EF}$ is reliable, the active node can choose two hosts, E and F, as its candidate. Thus, the local route ABC can be replaced by $AEFC$.

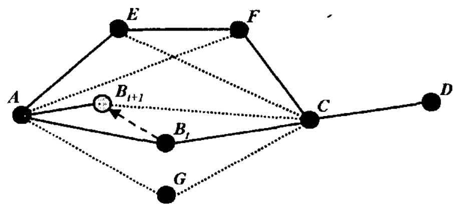

Figure 5. Finding Candidates.

Path Reduction

In the following conditions, the length of the existing route will be lengthened.

(1) The active node is either the source host or destination host.

(2) Allowing two hosts to replace the active node.

Under these two conditions, the active node will select one more nodes as the relaying nodes to locally reconstruct a reliable route. The increasing of path length will create maintenance overhead and increase the route break probability. To improve the route performance, we propose the path reduction technique in this section.

As shown in Fig. 6, when hosts F and B move closer, then the route $ABCDEFG$ will have a cycle $BCDEF$. A signal transmitted by host B would interfere with transmission of host F. By applying the path reduction process, the route could be changed to $ABFG$, so the original path length is thus reduced from 6 to 3. To achieve this path reduction, each relaying host should record the ID of all hosts on the route. An integration of DSR-based[1] and SSA-based[5] routing protocols will help each relaying host to collect the route information. Whenever a forwarding host receives the packet, it checks whether or not it can listen to another host that is recorded in its cache. As soon as the forwarding host detects a new neighbor that is recorded in its cache, it will perform the path reduction process. As shown in Figure 6, when host B detects that host F is a new neighbor and is recorded in its cache, host B will perform the path reduction process. Host B first issues a path reduction packet to host C, which was the successive host on the old route. The path reduction packet indicates that host F is its new successive host on the route. Whenever host C receives the path reduction packet, it transfers all data packets received before the path reduction packet to its next hop, say host D, relaying the path reduction packet to host D, and then releasing the link allocated to the old route. As soon as host D receives the data packets and path reduction packet, it performs the same operation as does in host C. The path reduction packet thus could be transmitted to host F, step by step. Whenever host F receives the path reduction packet from its predecessor, say host E, it sends a replying packet to host B indicating that the path reduction process has been completed. Thus host B can transfer the packets in its buffer to host F, which is a successor on the new route.

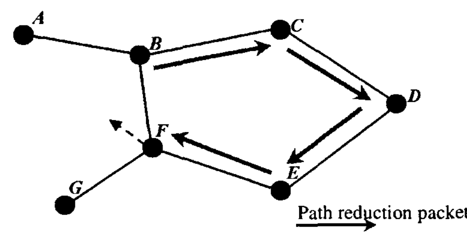

Figure 6. Path Reduction.

5. Performance Study

In this section, we use simulation to evaluate performance, in terms of route break probability and the traffic overhead for maintaining a route. The simulation environment is as follows. The MANET size is 1000*1000 basic units. The numbers of randomly generated hosts is set at 50, 200, and 400. To simulate the performance of ARMP route maintenance protocol, the simulator randomly selects a pair of source and destination hosts and then applies the SSA routing protocol to create a route from source to destination. The simulator automatically sets the host with a fixed speed 5, 10 or 20 units per second. The threshold of weak signal is changed varying from 0 to 50 units, to evaluate the performance of *ARMP* and *EARMP* protocols. All measurements are average values performed by 100 ticks of time.

Fig. 7 shows the traffic overhead and probability of route breakage as the host average speed and the threshold of weak signal is varied. For instance, as the host speed is set at 20 and the threshold value s of weak signal is set at 50, almost all hosts consider themselves to be in an unstable state. This is because that the link connecting two forwarding hosts with distance less than 50 units will be considered as a weak link and the hosts will perform the *ANDP* process, which creates packet transmissions and thus causes constant heavy traffic. However, since hosts on the route frequently find a candidate host and always replace the unreliable link with a reliable link, the probability of route break is close to zero. As the threshold is set by a smaller value, the traffic overhead is reduced and the probability of route breakage is higher. A similar phenomenon can be found in cases of speeds 5 and 10.

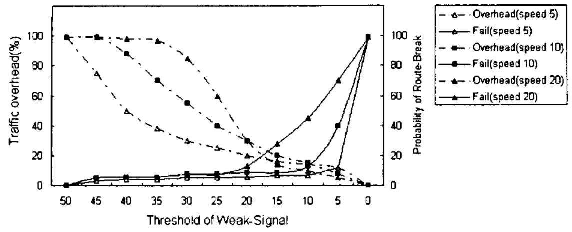

Figure 7. Effect on the value of threshold of *s* defined by ad hoc networks.

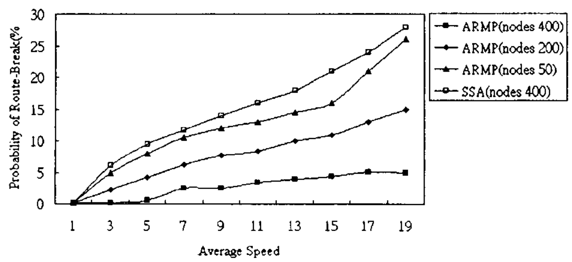

Figure 8. The effect of mobility.

Fig. 8 shows the relation of successful rate of ANDP process, the average host speed and the number of hosts in MANET. When the average speed of hosts is set to 10, a MANET having 400 hosts will have a higher successful rate for finding the candidate than 200 hosts. The more mobile hosts there are in MANET, the larger the success rate for finding candidate to replace the active node. In addition, consider the case the number of host is 400, the higher speed will cause a lower successful rate for finding a candidate in executing the *ANDP* process. Compared to pure SSA, the *ARMP* maintenance protocol will have low probability of route breakage.

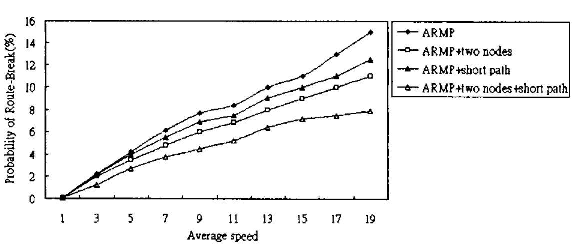

Figure 9. Comparison of *ARMP* and *EARMP* in probability of route break.

To evaluate the performance of enhanced ARMP, EARMP mechanisms are applied to maintain the existing route. Four cases are measured, as shown in Fig. 9: Case 1: pure ARMP; Case 2: allow two nodes to be the candidate to replace the active node; Case 3: perform the path reduction process whenever the route contains a cycle; Case 4: apply mechanisms of Cases 2 and 3.

As shown in Fig. 9, Cases 2, 3, 4 have a lower probability of route break. Thus, compared to pure *ARMP*, *EARMP* reduces the route break probability.

6. Conclusions and Future works

In this paper, we propose protocols for maintaining an existing route. Different from the previous approaches, an active node that has a higher degree of mobility actively issues the route maintenance process. Another significant difference is that this method maintains the route in a preventive manner so that active node finds a candidate before a route disconnection. Techniques for maintaining the route are developed based on these mechanisms, thus reducing the number of message paging or flooding. Disadvantages such as propagation delay for reconstructing a route, collision, and contention caused by performing message flooding thus can be avoided. In addition to propose the ARMP protocol, we propose the EARMP protocol to reduce the route break probability. Experimental study for performance improvement shows that the proposed ARMP and EARMP techniques improve the overhead of control packets and the probability of route break.

References

[1] D. B. Johnson, and D. A. Maltz, "Dynamic Source Routing in Ad-Hoc Networks," Mobile Computing, pp. 153-181, 1996.

[2] Z. J. Haas and M. R. Pearlman. The Zone Routing Protocol (ZRP) for Ad-Hoc Networks(internet draft). August, 1998.

[3] Rohit Dube, Cynthia D. Rais, Kuang-Yeh Wang, and Satish K. Tripathi, "Signal Stability-Based Adaptive Routing(SSA) for Ad-Hoc Mobile Networks," *IEEE Personal Communications*, Vol. 4, No. 1, pp. 36-45, February 1997.

[4] Chai-Keong Toh, "Associativity-Based Routing For Ad-Hoc Mobile Networks," *Wireless Personal Communication Journal*, Vol. 4, No. 2, pp. 103-139, March 1997..

[5] S. Y. Ni, Y. C. Tseng, Y. S. Chen, and J. P. Sheu, "The broadcast storm problem in a Mobile Ad-Hoc network," *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 151-162, 1999.

[6] C. E. Perkins and E. M. Royer "Ad-hoc On-Demand Distance Vector Routing ," *Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*, 1999.