A Route Recovery Scheme in DYMO for Efficient Ad-hoc

Junho Chung, Yonghwan Kwon, Bosung Kim, Hakkwan Kim, Kyungmin Lee, Dowon Hyun and Juwook Jang
Department of Electronic Engineering, Sogang University.
Email: {mywingcat, crusader, snugness, hiver, killoop21, snatcher}@eeca1.sogang.ac.kr, jjang@sogang.ac.kr

Abstract

In this paper, we propose a new route recovery scheme in DYMO routing protocol. DYMO is the routing algorithm that was designed appropriately to mobile Ad-Hoc network. It was proposed recently with added functionality of path accumulation based on AODV. However, data delivery rate of DYMO is smaller than that of AODV in mobile environment. Because DYMO has an overhead by messages that are accumulated the path information. Therefore, we devise a method of more prompt recovery to enhance a performance of DYMO when each node becomes more distant and loses routing path. To make this happen, we propose a new route recovery scheme which involves support nodes. The support node, which is assigned among the neighbors of existing route, is responsible to participate the operation and to transmit a data packet. In addition, We evaluate the performance with number of RREQ messages, an average of route recovery time, an average of packets delivery rate with distance in hops and an average of packets delivery rate by speed of nodes. In the simulation, the proposed method has reduced the route recovery time by 50%.

1. Introduction

Ad-hoc is a network seemly composed of mobile hosts, rather than a wired network, and is suitable when wired network configuration is not desired or network configuration is required in a short period of time. Ad-hoc network, as it does not have restriction in host relocation and does not require a base station nor wired network, has an advantage of faster network configuration and economic costs.

"This study was supported by the Seoul Research and Business Development Program, Seoul, Korea"

AODV[1], a wireless Ad-hoc routing protocol which has been the main subject of research is known to be suitable in movable nodes. However when nodes have high tendency to be mobile AODV shows a limitation in stable data transmission and this circumstance asks for a more appropriate algorithm.

DYMO[3] is a derived from AODV algorithm. In DYMO, the destination node is not a single node, but contains routing path information of nodes within the routing path, enabling more effective transmission.

However, this algorithm also shows an weakness in the time to maintain the routing information when mobility is high, thus a need arises for reinforcing DYMO algorithm by maintaining the path information.

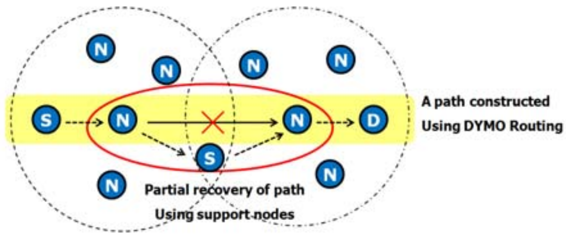


Figure 1. Partial recovery of a path constructed by DYMO routing

As shown in Figure 1, the recovery scheme proposed in this paper, utilizes not only nodes designated for routing but also the support nodes where support nodes assist in reconstruction of route when an increased distance between nodes resulting route breakage. Using simulation, we have tested the performance of the proposed scheme against the conventional algorithm.

This paper is composed in the following manner. Chapter 2, using an analysis of AODV and DYMO, refers to the related studies of conventional algorithms. Based on the contents of Chapter 2, Chapter 3 explains routing path recovery method proposed in this paper. Chapter 4 for the purpose of performance validation, compares the performance using NS-2 against the conventional algorithm. The conclusion is placed at the end of the document.

2. Related works

2.1. AODV

AODV is the most representative on-demand routing protocol. In case when a source node requires a path to transmit data to a destination node, AODV sends RREQ messages, with sequence number to countermeasure loops, to the neighboring nodes via broadcast. RREQ messages are sent throughout the network within the predefined time period until it reaches to the destination node, and all nodes receiving RREQ messages creates reverse path to the source node based on the sequence number of the message.

When RREQ reaches to a node which has the routing path to the destination node, RREP message is generated. RREP message contains the required number of hops and the most recent sequence number at the time of RREP generation, among the messages to the destination node. Nodes receiving RREP messages directed to the origin node of RREQ message creates a path to the destination node.

2.2. DYMO

DYMO routing protocol is a reactive algorithm developed for mobile Ad-hoc network. DYMO is similar to AODV but with a significant difference that DYMO accumulates the routing information of all nodes in the path.

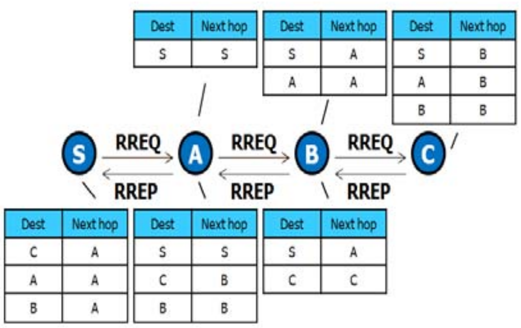


Figure 2. Routing table in nodes configured as DYMO routing algorithm

As shown in Figure 2, in nodes configured with DYMO routing method, all nodes between the source and destination exchanges routing information via routing information accumulation. Consequently, when to receive re-sent packets, newly searching of routing path is not required for packet transmission which decreases the RREQ message overhead as RREQ messages for routing search are not saved from sending.

However, when communication of nodes is broken from relocation of a node(s), existing routing table information within DYMO nodes are no longer valid. Thus, DYMO's advantage of routing path accumulation is shunned by increased RREQ message size from RREQ accumulation, and may result in worse performance against conventional AODV algorithm

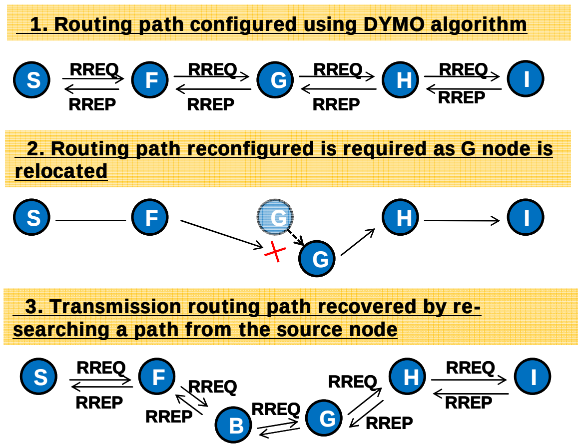


Figure 3. Problems with conventional DYMO

Figure 3 demonstrates how nodes constructed using conventional DYMO routing react to a broken path. Using RREQ and RREP messages, when a path between S, and I is broken by relocation of G node, disabling communication from F to G, S node again generates RREQ message to reconstruct path so that B node can participate in transmission. A single node relocation in G causes path search for all nodes in the path, and causes possible performance loss in overall Ad-hoc network with packet overhead from RREQ messages.

3. Proposed path recovery scheme

DYMO routing protocol is more effective in message packet transmission against conventional AODV with a fixed path. However, with DYMO when a path is lost due to a relocation of a node, a new routing path search must be performed causing higher packet overhead. DYMO has a limitation in guaranteeing performance in wireless networks with high mobility. Consequently, there is a need to increase the performance while guaranteeing path accumulation of DYMO routing protocol.

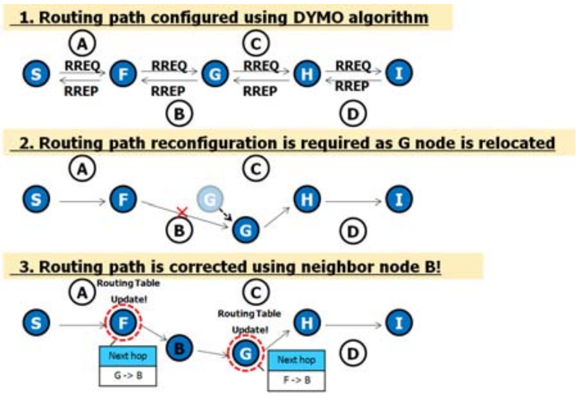


Figure 4. A recovery in a lost path caused by a relocated node

Figure 4 represents a recovery method proposed in this paper. To prevent path re-searching from a node relocation, support nodes are pre-assigned in neighboring nodes of a constructed path in the first search. Nodes A, B, C are pre-assigned as support node in the path between node S and node I, and when node G is moved, disabling communication between F and G, preassigned node B resumes connection between F and G, and updates its own address to the routing tables of the two to ultimately resume the transmission operation. This mechanism can reduce the packet overhead caused by possible path re-searching, and introduce more effective transmission.

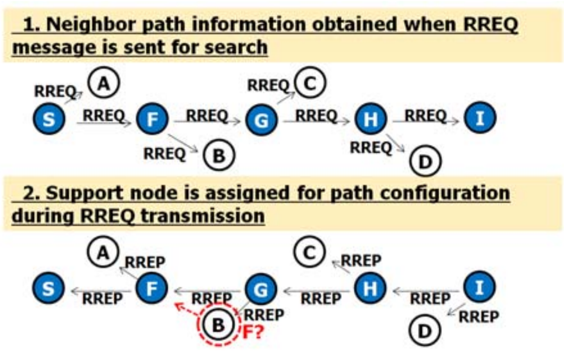


Figure 5. Support node assignment

Figure 5 demonstrates a procedure to pre-assign support node for path recovery. When RREQ messages are broadcasted from S node to I node in order to construct original path, support nodes A, B, C, D also receives these information and following the normal procedure of path search, stores RREQ messages to their own routing tables. Finally, neighboring nodes are assigned as support nodes when RREP messages are delivered from destination node.

Unlike RREQ, RREP messages are unicast messages, thus are sent only to the neighboring nodes within the fixed path and neighboring nodes receiving RREQ messages compares node address of previous hop in RREQ with destination address of RREP addressing the next hop. and self assigns when the two addresses are identical. In figure 5, node B recognizes that node F is the previous hop node from RREQ message. When RREP message is received from G, node B recognizes that next hop destination is node F and assigns itself as support node.

When self assignment of support node is done, node B can set the support flag to 1 to maintain the status of support node between F and G. In this status, node B listens to ACK signals from G to F to check whether packet is received normally.

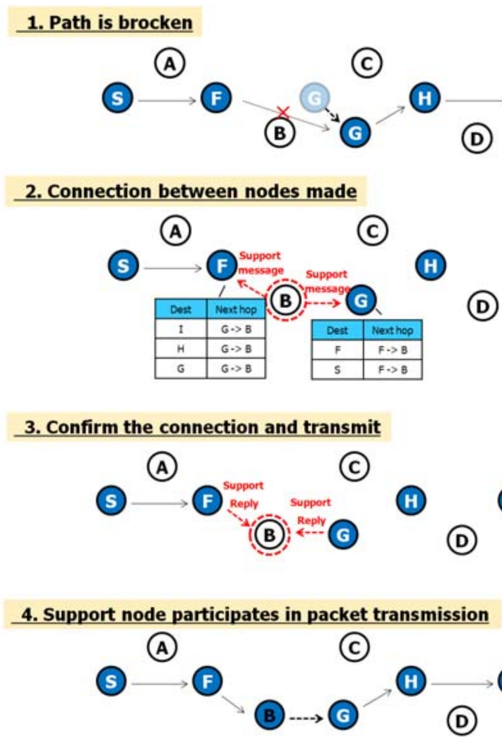


Figure 6. The behavior of support node when path is broken

Figure 6 illustrates proposed recovery scheme when path is broken. When path between F and G is broken during transmission, and packet confirmation signal of

ACK is not received from node G, node B which was also listening for ACK signals along with node F begins base operation to resume connection. Firstly, node B sends support message to node F and node G to notify routing table update, and creates connection with the two nodes.

When routing table update is completed by support node B, node F and G sends a Support Reply to notify that node B can participate in packet transmission. node B checks for reply for the support message sent previously to F and G and when the replies are confirmed from the two nodes, node B may receive data from node F and participate in transmission of data to node G. Along with node F and G with the updated routing table, node B can join in the path to perform data transmission

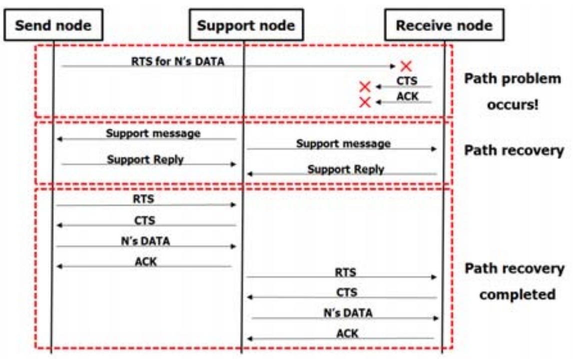


Figure 7. Message delivery overview with support node operations

Figure 7 explains the sequences of support node operation in message delivery. Based on the basic transmission operation of IEEE 802.11 wireless network, when data transmission is erroneous between sending node and receiving node, receiving node does not receive CTS signal and therefore cannot receive ACK signal as well. Support node in between sending node and receiving node receives support messages and support replies and participate in the data transmission as well to resume the original data transmission.

Table 1 lists the necessary elements to implement path recovery scheme proposed in this paper. Table 1 displays pseudo code for necessary operations of each type of nodes. As shown in the table, path recovery method can be easily implemented by addition of several signals and operations.

4. Simulation and result analysis

In this paper, simulation is performed with two scenarios of transmission distance and mobility. In the first scenario, to evaluate the performance for

Node	Action
Intermediate	When receiving a RREP message If Previous Hop Address = RREP's Next Destination Hop Address - Set node to Support node If this node Address = RREP's Next Destination Hop Address - Set node to Forward node
Forward	When receiving a Support message - Send Support reply to Support node - Update a Routing table
Support	When don't listening a ack from Next Hop Node - Send Support message to Forward Node When receiving a Support Reply from forward node - Operating

Table 1. Path recovery algorithm

transmission distance, the range of number of hops are preassigned, to relocate nodes to random directions and in a fixed scalar velocity within the predefined time and the range of movement, in order to determine the average path recovery time, average number of RREQ message occurrences, and average delivery rate of packets according to the message request intervals. In the second scenario, we have evaluated the average data receiving rate in data transmission involving various destination and various relocation velocity. This is to conduct evaluation of the proposed method against conventional method in more various transmission environment, by assigning multiple number of final destination randomly, and by moving nodes to various speed in random speed within the limited range.

4.1 Simulation scenario 1 and result

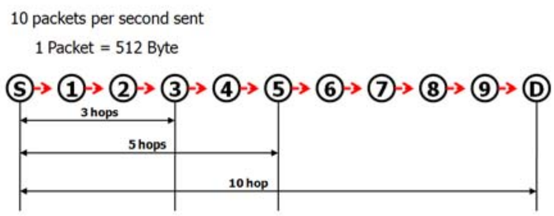


Figure 8. Transmission environment of scenario 1

Scenario 1 is to validate the performance of the proposed algorithm according to the distances between receiving and sending nodes. This is performed by counting the distance of the conventional routing algorithm in hops to construct appropriate path. Because wireless node communications require other nodes for data transmission, we have determined that it is more realistic to count in hops rather than the scalar distance.

Consequently, in this paper, in the transmission operation between moving nodes, packets are sent from the source node sequentially to the predefined multiple destination nodes, and counted the number of hops needed to reach the destination to check for the performance. Figure 8 illustrates the transmission environment of scenario 1, and is tested with 3 hops, 5 hops and 10 hops.

The dimension of the simulation space is 800m width, and 600m height. The wireless communication range of each node is set to 250m, and 50 nodes are to move in 5ms for 300 seconds. Node relocation model is based on random waypoints, with 2 seconds stops in every waypoint.

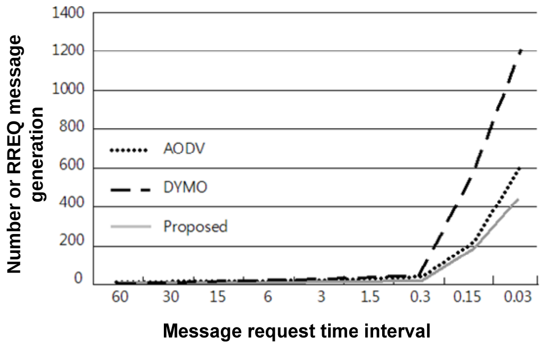
Algorithm	Average time to path recovery
AODV	375 msec
DYMO	417 msec
Proposed	182 msec

Table 2. Average time to path recovery

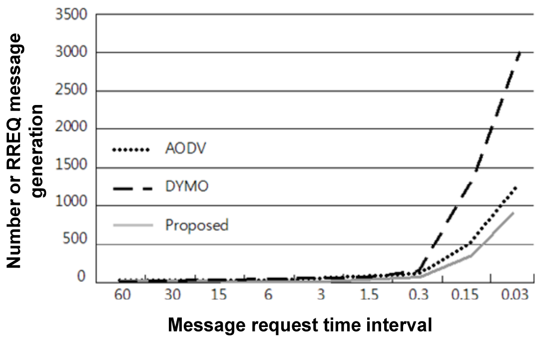
The average time to recover path from the broken connection between nodes shows different results against conventional algorithm in different number of node hops needed for transmission. However, in this paper, the test is performed with the objective of comparing the average recovery time of proposed algorithm against that of the conventional algorithm, and is not subjected to evaluate the recovery time for different number of hops, and have selected 5, the mid value as the number of hops to compare the different algorithms.

Looking at the test results with Table 2, DYMO showed increased search time for new path from the broken path against AODV. This is because RREQ message includes the address information of nodes included in the path, resulting in increased message size, and have caused more loads to the network in the process of path re-searching operation. Proposed algorithm does not initiate a new path search, and performs immediate path recovery using nodes in the existing path, allowing it to perform resuming of data transmission in faster time, and has demonstrated faster recovery time overall.

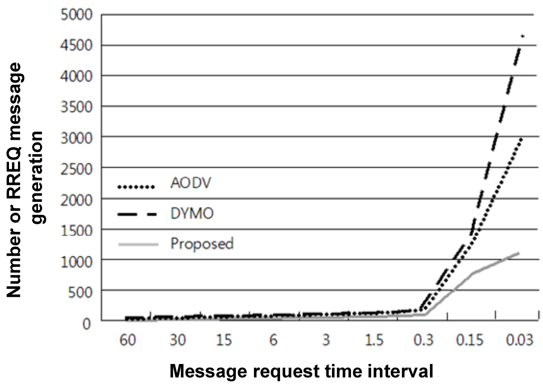
For data transmission between nodes, when destination node data is requested from the source, RREQ message is generated to search for the



(a) 3 Hops



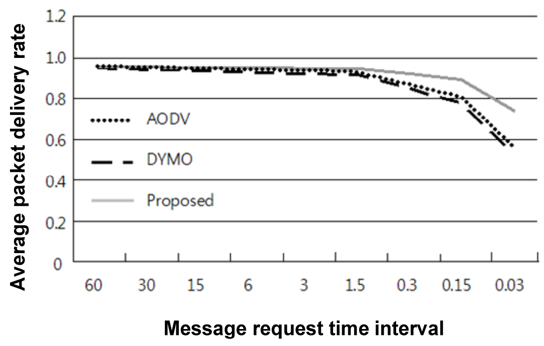
(b) 5 Hops



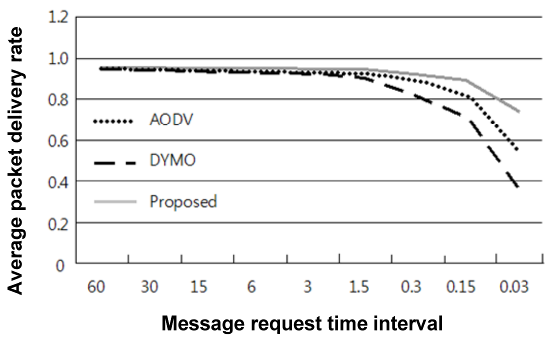
(c) 10 Hops

Figure 9. Number of RREQ message generation according to the message request time intervals

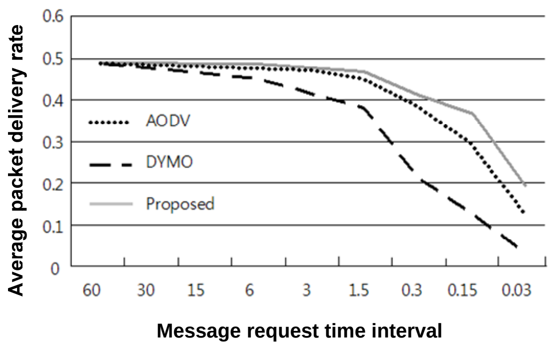
destination. As the message request time interval was smaller, more routing search is performed, prompting more load to the network. Figure 9 depicts the number of actual RREQ message generations with different message request time interval. The simulation result shows that with reduced message request interval, each



(a) 3 Hops



(b) 5 Hops



(c) 10 Hops

Figure 10. Average packet delivery rate with the various message request intervals

algorithm has shown different number of RREQ message generations. With DYMO algorithm, when a request is made within the node of destination path, RREQ message is not generated, and showed relatively small RREQ message generation than that of AODV. The reduced algorithm, as it recovers path recovery between moving nodes, preventing further re-searching of path and showed even more reduced number of RREQ occurrence.

In this paper, in order to evaluate the effects of the proposed algorithm to the entire network, we have measured the average packet delivery rate according to the different message request time interval. Packet delivery rate is calculated by comparing the overall packet size of sent and received packets within the 300 seconds.

Looking at the test results with Figure 10, in the path requests, DYMO had higher delivery rate than that of AODV when destination node is included in the previous routing path. This is a result of reduced packet overhead from the network with reduced number of RREQ message occurrence for message searching. The proposed recovery algorithm in this paper using such DYMO method, has decreased RREQ message occurrence and has shown enhanced packet delivery rate than of the conventional DYMO algorithm when message request time intervals is smaller.

4.2 Simulation scenario 2 and result

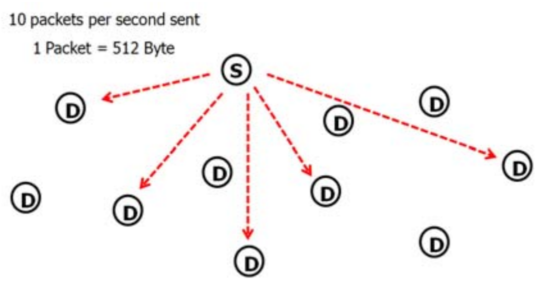
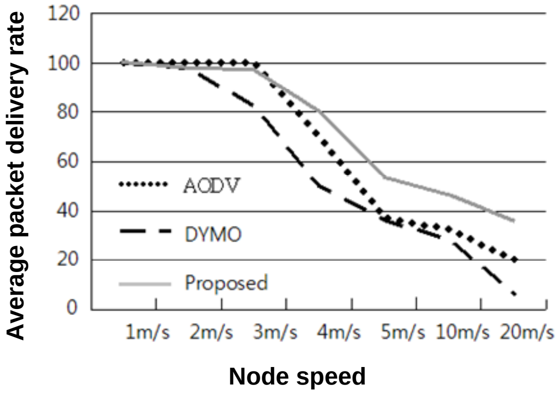


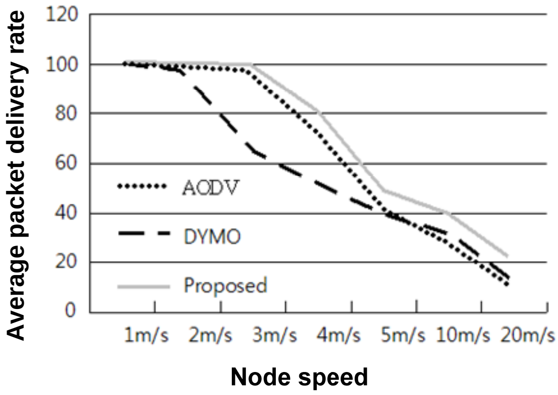
Figure 11. Transmission environment in scenario 2

In scenario 2, multiple nodes are assigned to move in different moving speed to check for performance in simultaneous transmission. As shown in Figure 11, our test included a single source node which transmits the packet and with 3 or 5 destination nodes for simultaneous transmission. This is to simulate an environment where multiple users requesting the information at the same time and to evaluate the performance of the proposed algorithm.

And we have measured performance of each algorithms when multiple nodes with various moving speed need to transmit simultaneously. Figure 12 refers to the average packet delivery rate with different moving speed, and is performed with 3 destination nodes and 5 destination nodes. The purpose of such segregation is to evaluate the performance in different network complexities.



(a) With 3 destination nodes



(b) With 5 destination nodes

Figure 12. Average packet delivery rate according to the node speed

Test result shows that with the increased node speed, average packet receiving rate decreases, This can be evaluated as that the rapid relocation between nodes caused transmission breakage and the corresponding recovery and re-sending tasks have constituted the result. RREQ message used for re-searching task for the purpose of data re-transmission maintains accumulated routing information and causes relatively more loads than AODV. Our proposed method has demonstrated decreased performance drop due to the increased speed. This is because utilization of support node for accumulation of routing information, which is the major advantage of DYMO, helps to maintain the communication longer. Likewise, when destination node is increased to 5 causing more load to the network, nodes applied with the proposed scheme has showed reduced recovery time even in the increased relocation speed, and has illustrated more effective transmission task operation against the other algorithms.

5. Conclusion

In this paper, with purpose of reinforcing DYMO algorithm in high occurrence of relocations, we propose a method to recover a path as prompt as possible in case of lost path due to increased distances between nodes. Our objective is to utilize the neighboring nodes of the existing routing path as support nodes for transmission task in lost path so that transmit rate would not drop below to the transmit rate of AODV when routing message overheads occur from routing path information overflows. The routing path recovery scheme proposed in this paper is based on DYMO protocol, and is effectively implemented by improvements in several additional signals and operations. When certain nodes in Ad-hoc network relocate in the middle of transmission task, in order to maintain the transmission operation, operation of searching for destination is restarted, thus creates loads for the network with transmit time and corresponding messages. Consequently, in order to optimize the time to recover transmission between nodes, using neighboring nodes are assigned as medium to reconstruct path in case of disconnected transmission, and to resume data packet communication.

Testing results with NS-2 simulator shows that with the proposed routing path scheme, path recovery time is reduced to 50% against the conventional DYMO protocol, and also shows improvements in average packet delivery rate and the number of RREQ message occurrences.

6. References

[1] PERKINS, C,. BELDING ROYER, E, S. Das: Ad hoc On Demand Distance Vector (AODV) Routing, frp://ftp.isi.edu/innotes/rfc3561.txt

[2] Stefano. B,. et. al., "Mobile ad hoc Networking", IEEE Press, July 2004.

[3] Chakeres I, et. al., "Dynamic MANET On-demand (DYMO) Routing", IETF, June 2006

[4] Das R., et.al., "Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks", IEEE Infocom 2000, March 2000.
[5] Elizabeth M., et. al., "An Implementation Study of the AODV Routing Protocol", IEEE, 2000.
[6] Sato, F. et, al., "A Route Reconstruction Method Based on Support Group Concept for Mobile Ad hoc Networks", IEEE, 2005.
[7] Marco Fiore code - Webpage, http://www.tlc-networks.polito.it/fiore/,Accessed on September 2007.