

An Improvement over AODV Routing Protocol by Limiting Visited Hop Count

Reza Fotohi

Department of Computer Engineering, Germei branch, Islamic Azad University, Germei, Iran

E-mail: Fotohi.reza@gmail.com

Shahram Jamali

Department of Computer Engineering, Mohaghegh Ardabili University, Ardabil, Iran

E-mail: Jamali@iust.ac.ir

Fateme Sarkohaki and Shahram Behzad

Department of Computer Engineering, Germei branch, Islamic Azad University, Germei, Iran

E-mail: Ahf_1365@yahoo.com, Sh.behzad173@gmail.com

Abstract— The AODV protocol is based on the minimum delay path as its route selection criteria, regardless of the paths load. This issue leads to unbalanced load dissemination in the network and the energy of the nodes on the shortest path deplete earlier than others. We proposed an improved AODV protocol with limited TTL (Time to Live) of RREP packet in which the route reply (RREP) packet of AODV is modified to limit TTL information of nodes. Experiments have been carried out using network simulator software (NS2). Simulation results show that our proposed routing protocol outperforms regular AODV in terms of packet delivery rate, good put, throughput, and jitter.

Index Terms— AODV, Routing Protocols, Multi-Hop

I. Introduction

In Mobile ad-hoc Networks (MANET) mobile nodes communicate with each other without the need for a central structure [1]. In addition, MANETs are a specific type of networks without infrastructure. An ad-hoc network is a collection of mobile nodes that communicate with each other without any centralized control and infrastructure. If the sender and receiver (transceivers) are not in communication range with each other, then the packages can be sent to the destination node using intermediate nodes. Mobile ad-hoc networks can be divided into two categories, structured and unstructured networks. An unstructured network or mobile ad-hoc network consists of mobile nodes and exchanges information without using a fixed station. In this type of network, nodes are not only doing administrative duties, but also act as a host. Often, nodes in these networks are moving dynamically. There is no base station or fixed structure in the network

configuration. Networks are composed of wireless devices, and besides each other, form a network with the ability of self-organization. Since the transmission range of a wireless medium is limited, communications in this type of networks depend on the constituent intermediate nodes. Thus, each node in the network also plays the role of a router. In these types of networks, the network topology is constantly changing because of the mobility of the network nodes. In addition, new nodes may be added to the network at any moment, or be removed from the network, or some nodes may turn themselves off. Some of the important characteristics of ad-hoc networks include: open and shared transmission medium, dynamic topology, unlimited battery power, limited processing power, limited transmission range, self-organization, distributed cooperation, and being temporary [2]. Three basic approaches are taken for routing protocols: proactive protocol, which is the sequential exchange routing information between the nodes (e.g. OLSR); reactive, which builds routes on demand (e.g. AODV), and hybrid combinations of the two protocols (e.g. ZRP). The ad-hoc routing protocol (AODV) [2] is a reactive routing protocol now in the process of being standardized at the IETF (RFC3561) and being implemented [3]. To transmit data over such a network, the AODV protocol enables dynamic, multi-hop routing between mobile nodes. AODV is an on-demand algorithm; a concept that a route discovery mechanism is invoked only when the sender wishes to transmit data. These routes are maintained as long as they are needed by the senders, and are deleted after a certain amount of time has passed; therefore, they do not overload the routing tables. AODV is designed for ad-hoc networks of a wide range of sizes, from the very small ones, to networks of tens to thousands of mobile internet-enabled nodes. Simulation experiments for 80 nodes have been reported, and first implementations are available [4].

The dynamic Features of mobile ad-hoc networks mean that both the topology of such networks and their size vary over time, giving rise to an unbounded execution tree, and infinitely, many states. This scenario is much more complex than the existing network protocols, and as a result, designing protocols that achieve correct delivery of messages is inherently more difficult. Much valuable effort is thus being directed to the formulation of routing protocol standards for MANETs, of which AODV RFC3561 is one example to serve as a characteristic to which a protocol implementation must conform. The implementations are then extended according to the guidelines set by the standard. Unfortunately, the thereupon protocol complexity sometimes results in unintended obscurity introduced into the standard, which, if undetected, can be transferred to the implementation. An analysis of the proposed standards is therefore desirable, resulting in subsequent revisions. Recently, formal confirmation has been successfully employed as an aid to detect obscurity in the improved AODV standards and implementations, resulting in the discovery of routing loop errors in early protocol versions (version 4) that have been addressed in later revisions of the draft standard. Both these approaches do not model in real-time manner, and instead replace the real-valued timer events with non-deterministic time-outs. This can result in false positives (i.e. error traces that do not conform to realistic scenarios), and is undesirable since the AODV protocol uses real-valued timers in an essential way, for example to determine the lifetime of routes. It is important that routing be handled in a timely manner (i.e. route discovery and message delivery happen without unnecessary time delays) [5, 6]. The timing values are determined by formulas dependent on protocol parameters (constants) specified by the standard. Clearly, the choice of constants and route lifetimes will affect the timeliness of protocol actions, especially as the network size and topology change dynamically over time.

In this paper, we modify the AODV routing protocol with limited TTL of the RREP packet. We show that the improved AODV has better than the regular AODV protocol (approximately 20%). We consider the effect of the default protocol parameters on our solution toward the behavior of AODV, and investigate properties such as route discovery and the ability to packet delivery ratio within a specified period. Our study of the AODV draft standard has highlighted a dependency of the lifetime of routes on network size, which may lead to failure in route discovery if it exists, or failure in data delivery to destinations. The observation pertains to the latest version (RFC3561-bis-01 [11]) and, in a simpler form, to earlier versions (13 and RFC3561-bis-00) of the draft standard. Having inspected a recent implementation of AODV [7], we confirm our perception also for this implementation with the ns2 version 2.32 simulation modifications to the standard that alleviates the problem by allowing route timeouts to adapt to network growth.

The rest of the paper is organized as follows. Section 2 review of some routing protocols in MANETs is given. Section 3 describes the AODV routing protocol. In section 4 we describe the proposed protocol. Section 5 describes the simulation setup and considered performance metrics. Section 6 gives the simulation results and. finally section 7 brings concluding remarks.

II. Routing Protocols in Mobile Ad-hoc Networks

Routing in the suggested algorithm is based on routing established upon demand. Each node has a routing table in which the node keeps its own routing information. The routing table contains fields such as destination node address, next node address, sequence number, distance, minimum requested bandwidth, maximum permitted delay, stream type, and route validity period. The destination node address field specifies the address of the destination node. The next node address identifies the next node on the route for sending packages to the destination. The sequence number field is used to avoid routing loops formations and repeated transmissions. As a routing message reaches a node, if the sequence number of the received message for a specific destination node is greater than the sequence number for that specific node in the routing table, the message will be processed. This simple act will prevent from repeatedly sending the routing packages, and avoids creation of routing loops in routing packages transmission. The distance field specifies the route length. The minimum requested bandwidth specifies the minimum amount of bandwidth required by the stream. This field is required only in cases of service quality streams (flows which require the service quality) and will be processed only when the stream type is of quality service. The maximum permitted delay field determines the maximum tolerable delay for the service quality streams. This field is also used only when service quality streams are being sent. The stream type is determined by the stream type field. This field can have the service quality level or the best effort. This field specifies the type of requested service. The validity period field determines the period in which a route is valid. After passing this period, the route will not be valid anymore. If this field receives a package for a destination before end of the validity period, it will be re-initialized [8, 9]. Routing protocols in mobile ad-hoc networks can be divided into two categories, table-based or proactive protocols and need-based protocols table-based or proactive protocols are used for periodic updating of the links. The routes information is kept in a table and is used whenever needed. However, need-based protocols do not require keeping route data, and whenever a route is needed, they start to explore a route based on source location.

2.1 Table Driven Routing Protocols (Proactive Model)

In this category of protocols, each node keeps one or more tables containing routing information to the other nodes of the network, all nodes update their tables to maintain consistency and to have an up-to-date view about the network. As the network topology changes, the nodes broadcast updating messages throughout the network. This category of protocols is distinctive by the manner of distribution of information about topology changes throughout the network, and by the number of tables that are required for routing. WRP, DSDV, FSR, HSR, GSR, ZHLS & CGSR are some examples of table-based protocols.

2.2 On Demand Routing Protocols (Reactive)

In comparison with table-based routing protocols, in this category of protocols, not all updated routes are stored on each node; instead, the routes will be constructed whenever they are needed. When a source node wants to send one message to a destination, it will request the route discovery mechanisms to find a route to the destination (RREQ). The route remains valid until the destination is available or if is not for long-term needs. Once a route to the destination is found, the RREP mechanism sends, in reverse, the route to the source node. CBRP, AODV, DSR, TORA & ABR are some examples of need-based protocols [13].

III. The AODV Routing Protocol

AODV is a reactive routing protocol, where the route between the source and a destination node is created on an on-demand basis [10].

The route discovery process is initiated when a sender (source node) *S* has data to send to a destination (destination node) *D*, but has no valid corresponding routing table entry. In this case, node *S* broadcasts a route request (RREQ) message in the network. The RREQ message is re-broadcasted and forwarded by other mediocre nodes in the network until it achieves the destination node *D* (or a mediocre node that has a valid route to node *D*). Every node that receives the RREQ message will create a routing table entry to create a reverse route back to node *S*. In response to the RREQ message, the destination node *D* (or a mediocre node that has a valid route to node *D*) unicasts a route reply (RREP) message back along the previously established reverse route. At the end of this route discovery action, an end-to-end route between the source node *S* and destination node *D* is established. Usually, all nodes on this route have a routing table entry to both the source node *S* and destination node *D*. In the event a connection in this end-to-end route was to break down (due to mobility or interference), the node that detects the breakdown sends a route error (RERR) message back to the source node. The RERR message will cause the set of affected nodes to dispense their routing table entries using the broken connection.

IV. Our Protocol

In ad-hoc On-demand distance vector (AODV) that is a routing protocol in which each node maintains a routing table, one entry per destination, which records the next hop to the destination and its hop count. AODV also uses a sequence number to ensure freshness of routes. AODV discovers a route through network-wide broadcasting. It does not record the nodes it has passed, but only counts the number of hops. It builds the reversed routes to the source node by looking into the node that the route request has come from. The intermediate nodes check for fresh routes according to the hop count and destination sequence number, and forward the packets they received from their neighbors to the respective destinations. AODV utilizes periodic beaconing (HELLO packets) for route maintenance. If a node does not receive a HELLO packet within a certain time, or it receives a route break signal that is reported by the link layer, it sends a route error packet by either unicast or broadcast, depending on the precursor lists in its routing table (i.e. active nodes toward the destination). AODV avoids the stale route cache problem of DSR and adapts the network topology changes quickly by resuming route discovery from the very beginning.

However, in our proposed method, we have limited the TTL value for the RREQ request path and considered the following two states in which TTL is very low. Considering that in this state the packet (RREQ or any other packet) does not reach a node and remains in the middle nodes, therefore it was discarded. Therefore, we did not consider this state. In the second state, the TTL value is very high, meaning that we allowed a larger number of hops. In this state, the rate of packet delivery was lowered, therefore we disregarded this state too. In our proposed method (improved AODV), we limited the hop counts using the following condition:

```

INFINITY2=#FFFF
If (rt->rt_last_hop_count < INFINITY2)
{
    rt->rt_req_last_ttl = max(rt->rt_req_last_ttl, rt->
rt_last_hop_count);
}

```

This means that if the condition applies, we consider the last hop for the last TTL (i.e. the packet has used its authorized hop). However, if the condition does not apply, meaning that the packet wants to use a larger number of hops to reach its destination, the packet is discarded; because if we increase the number of hops, it will take the packets longer to reach their destination; therefore, resulting traffic will occupy bandwidth. However, we freely allow the packets requiring 255 hops or less to reach their destination. For example, when we see a packet pass the 256 threshold, we limit

the hops using a certain condition. This leads to improvement of packet delivery rate, throughput, good put, and jitter.

V. Simulation Setup and Performance Metrics

Here we emphasize the performance evaluation of regular AODV and improved AODV routing protocols using the NS simulator, that is an object oriented and discrete event driven simulator. The NS software is generally used for simulation of local area networks and wide area networks. We could simulate the regular AODV and improve AODV routing protocols using NS-all-in-one version 2.32. Although, NS can be implemented on different operating systems, but in this paper, we used the Back Track 5 Linux operating system, which is being employed for a number of programming tools with simulation process to run the simulation of NS-all-in-one version 2.32. For simulation and implementation, we should firstly design our network scenario in OTCL language that gives us the output as trace. In addition, we use XGRAPH to show the graphs, and finally, we employed the auxiliary (supporting) program NAM to analyze performance of nodes and how the packages are being sent and eliminated [11].

5.1 Simulation Setup

We brought up a network of nodes placing within a 1000m X 1000m area. The nodes in the network model with random motion, velocity in the 0 – 20 m/s, when the node reaches the target point, residence time, and then randomly select a new target point and a new speed, movement to the new target point, and so on [11], MAC layer uses the 802.11 protocol, the node transmission radius of 250m, the link bandwidth 1Mbps, packet size is 512 bytes, simulation time is 100s. The efficiency of regular AODV and improved AODV routing protocols is evaluated by keeping the network speed and pause time constant and varying the network size (number of mobile nodes). Table 1 shows the simulation parameters used in this evaluation.

Table 1: Values of parameters for AODV Simulation

Simulator	NS2 2.32
Protocol	AODV
Simulation duration	200 Second
Simulation area	1000m X 1000m
Number of nodes	20,40,60,80
Transmission range	250 m
Movement model	Chain topology
MAC Layer Protocol	IEEE 802.11
Pause time	100 sec
Maximum speed	20 m/s
Packet rate	4 packets/sec
Traffic type	CBR (UDP)
Data payload	512 bytes/packet

5.2 Performance Metrics

PDR is the ratio of the number of data packets received by the destination node to the number of data packets sent by the source mobile node. It can be evaluated in terms of percentage (%). This parameter is also called “success rate of the protocols”, and is described as follows:

$$PDR = \left(\frac{\text{Sent Packet No}}{\text{Receive Packet No}} \right) * 100 \quad (1)$$

Throughput is the average rate of successful message delivery over a communication channel. This data may be delivered over a physical or logical link, or pass through a certain network node.

$$X = \frac{C}{T} \quad (2)$$

Where X is the throughput, C is the number of requests that are accomplished by the system, and T denotes the total time of system observation

Good put is the application level throughput, for example the number of useful information bits, which the network delivers to some destinations per unit of time. The value of considered data excludes protocol overhead bits as well as retransmitted data packets. This is related to the value of time, from the time, the first bit of the first packet is sent (or delivered), until the last bit of the last packet is delivered.

TTL is a method that limits the lifetime of data in a computer or network. TTL may be implemented as a counter attached in a packet in the network. When count is reduced to zero or timespan has elapsed, data is dropped or discarded in the data network [12].

In this method, we limit TTL (time to live) value for route request packet (RREP) that broadcasts periodically or when an event occurs in the network for finding the path to the destination node (e.g. limited to 128 or 256; this value varies depending on network size or network topology). This value is not lower than threshold (as the threshold depends on network size or network topology, this value is variable too). Since in this case, the packet (RREP or any other packets) does not receive to the destination node and drops or discards in the middle node. This value is lower than threshold, because packets maintain long time in network and then increase the traffic, packet delay, jitter and decrease packet delivery ratio and throughput in network. We simulated this method in NS-2.32 and saw good improvement in packet delivery rate, good put, throughput, and jitter.

VI. Simulation Results

The simulation results are shown in the following section in the form of line graphs. Performance of regular AODV and improved AODV based on the varying number of nodes in chain topology is done on

parameters like packet delivery ratio, good put and throughput.

Fig .1 shows packet delivery ratio against the number of nodes. It shows that the improved AODV protocol has a better throughput in the time range of 20 to 80 nodes (above 20%).

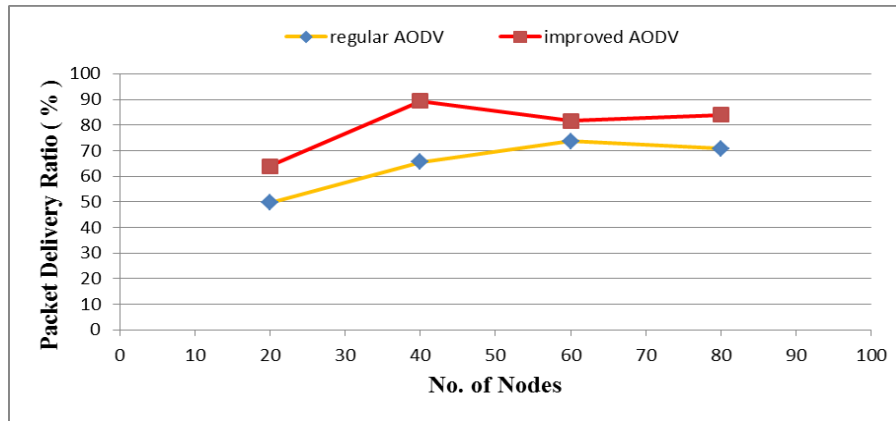


Fig. 1: Packet Delivery Ratio for Regular AODV and Improved AODV

Fig. 2 and Fig. 3 show good put and throughput against the number of nodes. It shows that when the number of nodes is 40, the improved AODV has less

good put and throughput than regular AODV, respectively; however, generally it is better than regular AODV protocol.

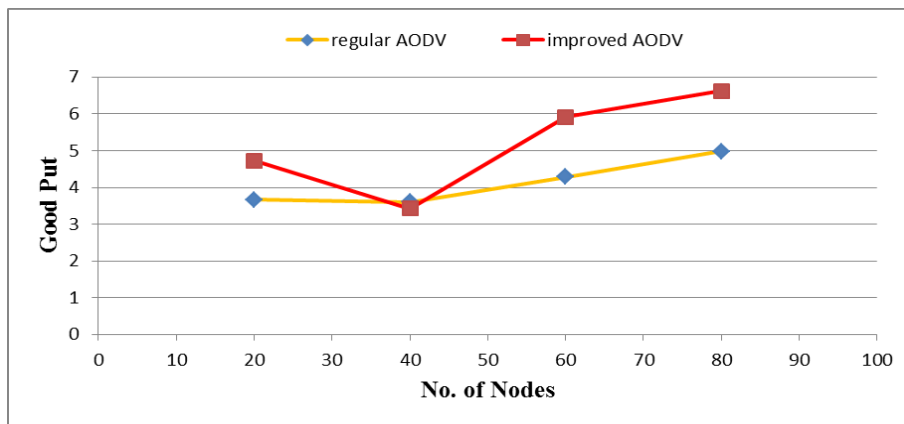


Fig. 2: Good Put Regular AODV and Improved AODV

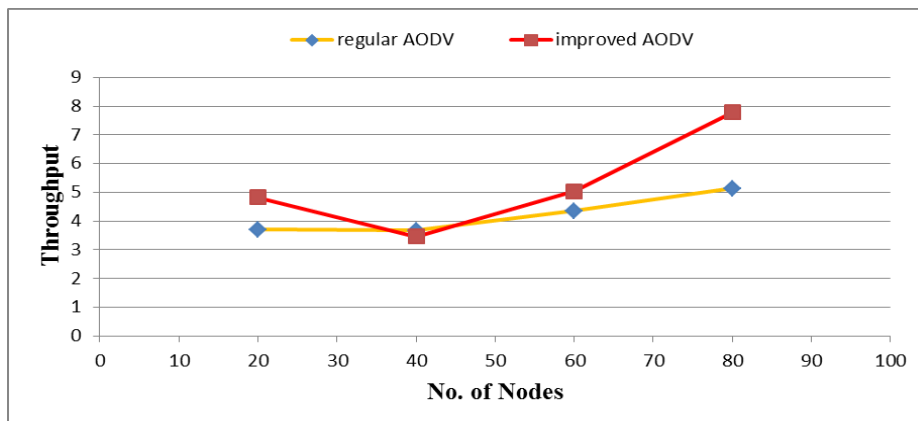


Fig. 3: Throughput Regular AODV and Improved AODV

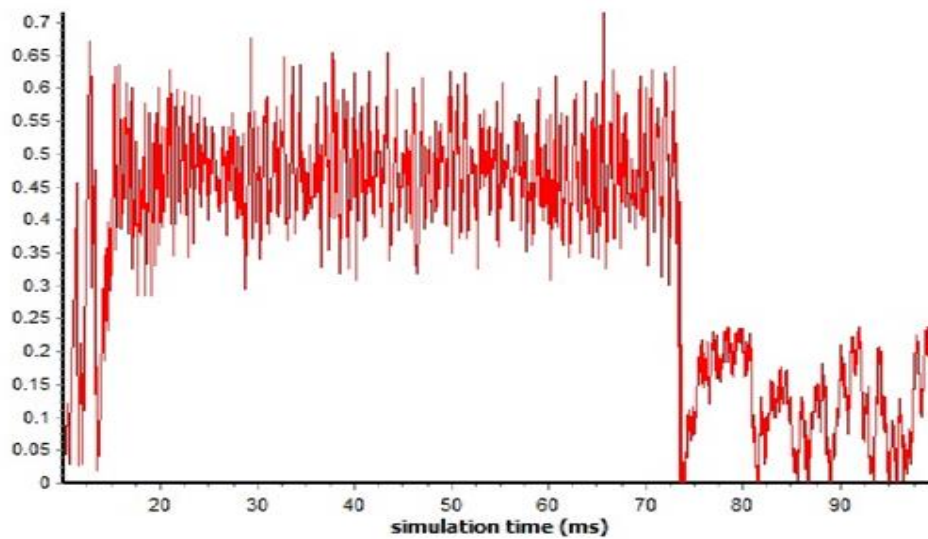


Fig. 4: Jitter of regular AODV

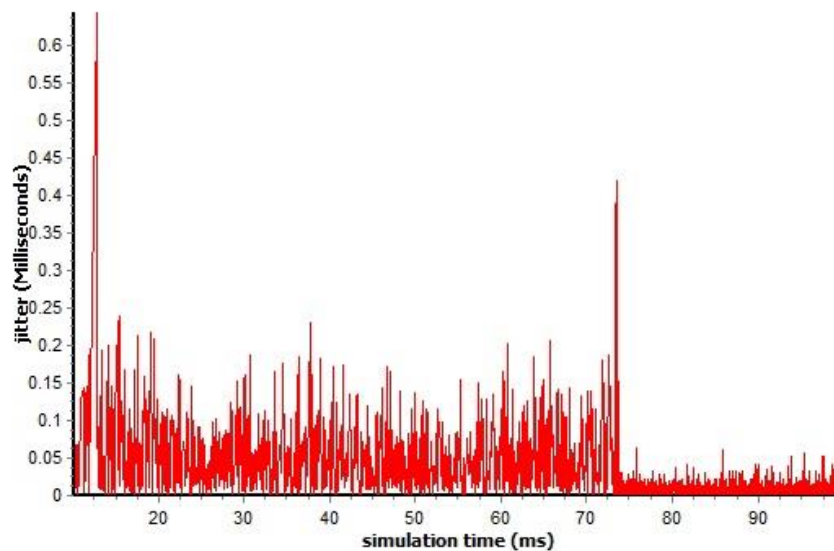


Fig. 5: Jitter of improved AODV

Fig. 4 and Fig. 5 show jitter against time. They show jitter rate in the regular AODV routing protocol compared to Fig. 5 of improved AODV protocol, which has worsened. However, in the improved AODV, this value has significantly improved.

VII. Conclusion

In this paper, we have developed an improved AODV routing protocol for mobile ad-hoc networks. The work was accomplished by limited TTL (Time to Live) of RREP packet that the route reply (RREP) packet of AODV is modified to limited TTL information of nodes, and evaluated the four performance measures (i.e. PDR, throughput, good put and jitter with different number of

nodes). Then we compared performance of our work with regular AODV in one scenario with 20 to 80 nodes. Simulation results show that the improved AODV protocol has a distinct advantage in terms of packet delivery ratio (PDR), throughput, good put and jitter over the regular AODV protocol (approximately 20%). As part of future work, we plan to add the factor of interfere between nodes into the route metric then work for MANETs.

References

- [1] Akshatha, P.S., Khurana, N., Rathi, and A.: Optimal Path For Mobile Ad-Hoc Networks Using Reactive Routing Protocol. International Journal of

Advances in Engineering & Technology, IJAET (2011) ISSN: 2231-1963

- [2] D. Chakeres and E. M. Belding-Royer. AODV routing protocol implementation design. In Proceedings of the International Workshop on Wireless Ad-hoc Networking (WWAN), pp. 698–703, Tokyo, Japan, March 2004.
- [3] E. N. et al. AODV-UU: Ad-hoc On-demand Distance Vector Routing. <http://user.it.uu.se/~henrik/aodv>
- [4] NS -2, The ns Manual (formally known as NS Documentation)
- [5] E. Perkins, E. M. Royer, and S. R. Das, “Ad-hoc On-Demand Distance Vector (AODV) Routing”, Internet Draft, draft-ietf-MANETSaodv -10.txt, work in progress, 2002.
- [6] Li Ting Jun “Study On Airborne Single passive location Technology” Applied Mechanics and Materials Vols.58 (2011) pp. 2006-2009
- [7] Li Ting Jun, Lin Xueyuan, “GPS/SINS Integrated Navigation System Based On Multi-Scale Preprocessing”, Journal of Wuhan University, 2011, Vols 36(1): pp. 6-9
- [8] Li Tingjun “The Phonetic Complex Data Based on FPGA Key Engineering Materials”, Vols 475 (2011) pp. 1156-1160.
- [9] Khan, J., Hyder, S.I., Fakar, S.M.: Modeling and simulation of dynamic intermediate nodes and performance analysis in Manet’s reactive routing protocols. International Journal of Grid and Distributed Computing 4(1) (March 2011).
- [10] Barakovic, S., Kasapovic, S., Barakovic, J.: Comparison of MANETS routing protocols in different traffic and mobility models. Telfor Journal 2(1) (2010).
- [11] E. M. Belding-Royer, I. D. Chakeres, and C. E. Perkins. Ad-hoc On-Demand Distance Vector (AODV) Routing: Work in progress, July 2004. Internet Draft, RFC 3561bis-01, <http://moment.cs.ucsb.edu/pub/draft-perkins-MANETS-aodv-bis-02.txt>.
- [12] Kaur, S.: Performance Comparison of DSR and AODV Routing Protocols with Efficient Mobility Model in Mobile Ad-Hoc Network. IJCST 2(2) (June 2011).
- [13] Shahram Behzad, Reza Fotohi, Shahram Jamali, “Improvement over the OLSR Routing Protocol in Mobile Ad Hoc Networks by Eliminating the Unnecessary Loops”, IJITCS, vol.5, no.6, pp.16-22, 2013. DOI: 10.5815/ijitcs.2013.06.03

Authors’ Profiles



Reza Fotohi received his B.Sc. in computer engineering from Shabestar University of Applied Science And Technology, Tabriz, Iran, in 2009, and his M.Sc. in Computer Engineering from Islamic Azad University, Germei branch, Ardabil, Iran, in 2013. His research interests include wireless networks, mobile computing and combinatorial optimization.



Shahram Jamali received his B.Sc. degree in Computer Engineering from Amirkabir University, Tehran, Iran, in 1999, his M.Sc. degree in Architecture of Computer Systems from Iran University of Science & Technology (IUST), Tehran, Iran, in 2001, and his Ph.D. degree in Architecture of Computer Systems in 2008. He is currently Associate Professor of Computer Engineering at University of Mohaghegh Ardabili (UMA). His research interests include Congestion Control in computer networks, Self-managing Computer Systems, Biologically-inspired Networking and Load balancing in Grid environment.



Fateme Sarkohaki received her B.Sc. in computer engineering from Jahad Daneshgahi, Ahvaz, Iran, in 2009 and her M.Sc. in Computer Engineering from Islamic Azad University, Germei branch, Ardabil, Iran, in 2013.



Shahram Behzad received his B.Sc. in Computer Engineering from Parsabad University, Parsabad, Iran, in 2008, and his M.Sc. in Computer Engineering from Islamic Azad University, Germei branch, Ardabil, Iran, in 2013.

How to cite this paper: Reza Fotohi, Shahram Jamali, Fateme Sarkohaki, Shahram Behzad, "An Improvement over AODV Routing Protocol by Limiting Visited Hop Count", International Journal of Information Technology and Computer Science(IJITCS), vol.5, no.9, pp.87-93, 2013. DOI: 10.5815/ijitcs.2013.09.09