Research Journal of Applied Sciences, Engineering and Technology 8(19): 2092-2096, 2014
DOI:10.19026/rjaset.8.1202
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: May 31, 2014 Accepted: July 13, 2014 Published: November 20, 2014

Research Article

Improved LDSRDV-localized Dynamic Source Routing for Distance Vector Performance in MANET

[1]Deepa Dasarathan and [2]P. Nirmal Kumar
[1]Department of Electronics and Communication Engineering, Sathyabama University,
[2]Department of Electronics and Communication Engineering, College of Engineering, Guindy,
Anna University, Chennai, India

Abstract: A considerable amount of research within the field of Quality of Service (QoS) support for distributed environment. In this study, we propose Improved LDSRDV-localized Dynamic Source Routing with aim to improve the Distance Vector Performance in MANET. One of the key issues in providing end-to-end QoS guarantees in packet networks is determining a feasible path that satisfies a number of QoS constraints. Nodes mobility brings flinty challenges to multicast in Mobile Ad hoc Networks (MANETs). To track nodes mobility, flooding messages are widely used for data delivery structure construction and maintenance in many multicast protocols. These periodic flooding messages significantly consume network resources, such as energy and bandwidth and result in network. To release data delivery structure maintenance onus, profited from location based service, a number of stateless location based multicast protocol were proposed, where a destination list is encapsulated into each data packet for data packet orientation. However, due to data packet capability limitation, the way of encapsulating a destination list in each data packet header restricts the protocol scalability. Simulation is performed using network simulator to demonstrate the efficiency of Improved LDSRDV.

Keywords: Bandwidth, energy, flooding messages, mobile ad hoc networks, multicast protocol, quality of service

INTRODUCTION

Most of the routing algorithms have ignored the QoS issues and their main focus is on establishing a path between a source and a destination. This is typically done using the number of hops or the administrative distance as the sole measure of quality. In today’s networks, QoS requirements have been considered as secondary to routing, primarily assuming a best-effort delivery scheme. With the growing diversification of networking applications and explorations, networks are being forced to cater for a variety of traffic demands with definite and often critical QoS requirements. There are three predominantly used routing protocols present for finding routes from a source to destination which are Ad hoc On-demand Distance Vector routing (AODV) by Perkins and Royer (1999), Destination Sequenced Distance Vector routing (DSDV) by Perkins and Bhagwat (1994) and Dynamic Source Routing (DSR). The DSR was proven to have higher throughput when compared to the other two protocols by Johnson and Maltz (2001).

The DSR protocol allows nodes to dynamically discover a source route across multiple network hops to any destination in the ad hoc network. Each data packet sent then carries in its header the complete, ordered list of nodes through which the packet must pass, allowing packet routing to be trivially loop-free and avoiding the need for up-to-date routing information in the intermediate nodes through which the packet is forwarded. By including this source route in the header of each data packet, other nodes forwarding or overhearing any of these packets may also easily cache this routing information for future use.

Owing to the advantages that the DSR protocol portrays, in this study, we provide a means to improve the Localized Dynamic Source Routing in MANETs to result in high network performance. The remaining sections of this study contain the works related to this paper and the problem identified; the proposed solution to the problem; the simulation of the Improved LDSRDV with discussion of results; and the conclusion.

LITERATURE REVIEW

Most of the above mentioned works focus on the design and development of MAC protocols and ignore the effect of co-channel interference along an active path from source to destination. To the best of our

Corresponding Author: Deepa Dasarathan, Department of Electronics and Communication Engineering, Sathyabama University, Chennai, India

knowledge, there has been little work on the design of routing protocols for wireless ad-hoc networks using directional antennas. A directional routing strategy has been proposed, mainly to suppress interferences from neighboring relay nodes while attaining power effectiveness. The approach has been applied to two popular on-demand routing protocols, namely DSR and AODV, which will be referred to as the Directional DSR (DDSR) protocol by Gharavi and Hu (2009) and the Directional AODV (DAODV) protocol by Le *et al.* (2007).

The wireless ad hoc network routing protocols and the issues that affect the design, deployment and performance of an ad hoc wireless system network, four routing protocols were chosen for analysis and evaluation: AODV, DSR, OLSR (Hauser *et al.*, 1999) and TORA. The analyses of the secure versions of these protocols are presented. The above four routing protocols have been chosen for two main reasons. First, they are considered the most popular ad hoc routing protocols. Second, many secure versions have been derived from their basic implementation. Routing protocols for ad hoc wireless networks can be classified into three types based on the underlying routing information update mechanism employed. An ad hoc routing protocol could be reactive (on demand), proactive (table driven) or hybrid. The three types of ad hoc routing protocols and list the available routing protocols for that category as well as some of their secure versions. Reactive routing protocols obtain the necessary path when it is required, by using a connection establishment process. They do not maintain the network topology information and they do not exchange routing information periodically. Hu and Johonson (2004), Hu *et al.* (2002) and Hu and Perrig (2004) have discussed about some of the reactive protocols. The authors Gracia-Luna-Aceves and Spohn (1999), Liu and Ramakrishnan (2001), Mieghem *et al.* (2001) and Nicholson (1966) have also provided some of the works related to the improvement of LDSRDV made in this study.

Problem statement: The topology of a mobile ad hoc network as a dynamic un-directed graph $G = (V, E)$, where V is the set of mobile nodes in the network and E is the set of wireless links between neighboring nodes. A subset of the network nodes in V forms a multicast group. Those nodes are called group nodes, denoted by VG. Between the group nodes, unicast routing is available as part of the underlying unicast network routing protocol, which forms a set of paths EG between any two of the group nodes. The overlay virtual network is thus denoted by an un-directed complete graph $GG = (VG, EG)$.

PROPOSED METHODOLOGY

General view of tree construction algorithm: The tree construction algorithm that is based on this circular order C_{min}. Since it will only be using C_{min} from now on, then refer to that order with C only. The input for the algorithm is the PAM (Point Accepted Mutation) distances of the pair wise alignments plus the circular order C of the optimal tree T_{min}.

Given is a set of sequences $S = \{s_1, \ldots, s_n\}$. First, the optimal circular order C is calculated. The sequences are renamed with respect to that order. Starting with the optimal circular order C, each of the n-1 pairs of neighboring leaves are swapped (Algorithm) and the path difference δ (s_i) is calculated for $i = 1..n$. To save computation time, initially calculate all n δ (s_i), sort them and store them in a list D. The best connection is the one with the smallest δ (s_i): $\in = \min_{1 \le i \le n} \delta\ (s_i)$. The leaves are connected and one of the connected leaves is chosen as a representative for the next steps. For the next connection step only two path differences δ have to be recalculated, since nothing else has changed. Since there are n-2 internal nodes (without the root), the total computation needs O (n log (n)) for the sorting and linear time in n for the actual tree construction. Once the tree structure is known the exact places of the nodes can be obtained with any least squares method, which takes in the order of n^2 time. So given a circular order C, the tree topology can be determined in O (n log (n)) time. If the edge lengths are to be computed as well, then the computation takes O (n^2) time. The extended nodes can be added to the nearest neighbor node. The algorithm for the proposed method is shown below:

```
Algorithm Tree Construction ($L$)→$T_{opt}$ is
    $C$: = $N$ ($L$)
    $L$: = rcorder ($S$, $C$)
    $F_{min}$: = $\sum_{i=1}^{n} OPA$ ($s_i$, $s_{i+1}$)
    $T$: = $L$
    "The initial tree only consists of leaves"
    $D$: = {}
    for $i$ = 1.. |$L$| do
        $D$: = $D \cup \{\delta\ (i), i\}$
    od
    $D$: = sort ($D$)
    for $k$ = 1 … $n$-3 do
        {$c_1$ $best$}: = $min$ ($D$)
        $T$: = join ($T$, $L_{best}$, $L_{best+1}$)
        $L$: = $L \setminus L_{best}$
        "represent the two leaves with one leaf"
        Recalculate $\delta$ ($best$) and put it into the sorted list $D$
    od
    return T
end
```

Construct a tree T from a set of sequences $S = \{s_1, \ldots, s_n\}$ that has the minimum score

The node point of calculation: The tree construction algorithm is the following the leaves in order of C. Starting from the leaves we want to find the internal nodes. Since we consider binary trees only, we know that at least two leaves are connected. As shown in the

Fig. 1, when two leaves that are connected are swapped, the result a tree with the same topology, if we traverse the tree with the swapped leaves in the same circular order as before, we get the same path length. But when two leaves are swapped that are not connected, some edges of the tree are traversed too often, which increases the score. So all we have to do is to swap each of the neighboring pair of leaves and to calculate the resulting total path length or score. If the score stays the same, when the two leaves are connected, if the score increases, the leaves are not connected.

Construction tree algorithm based path way: A general algorithm of constructing a tree with degree *k* is similar to the binary tree example above, except that each time *k* children are selected and *k* sub trees are clustered. In the case when a packet cannot be forwarded to a children node (no route), the packet is lost and the destination nodes in that sub tree will not receive the packet. This situation may happen in MANET because mobile nodes can shut it down abruptly or move out of reach. Therefore, the sender should include only the "active" nodes as the packet's destination nodes. This requires that each node periodically refreshes the membership of itself to the rest of the group. If an update message has not been received over a timeout period from a node, the node is purged from the destination list of other nodes. Because of the periodic membership refreshment, the destination nodes are very likely to be reachable from the source node.

SIMULATION RESULTS

The responsibility for assessing the status of a route falls to each node in the route. Each must insure that packets successfully cross the link to the next node.

If it doesn't receive an acknowledgement, it reports the error back to the source and leaves it to the source to establish a new route. While this process could use up a lot of bandwidth, DSR gives each node a route

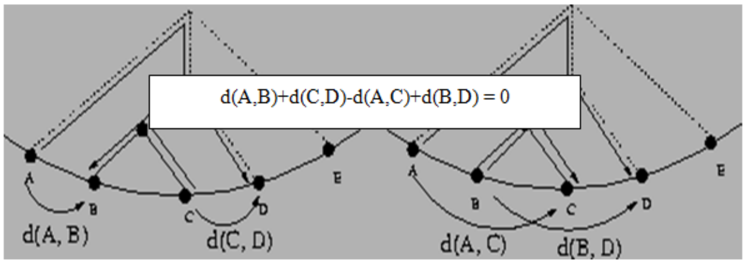

Fig. 1: The connection and swapping in tree construction algorithm

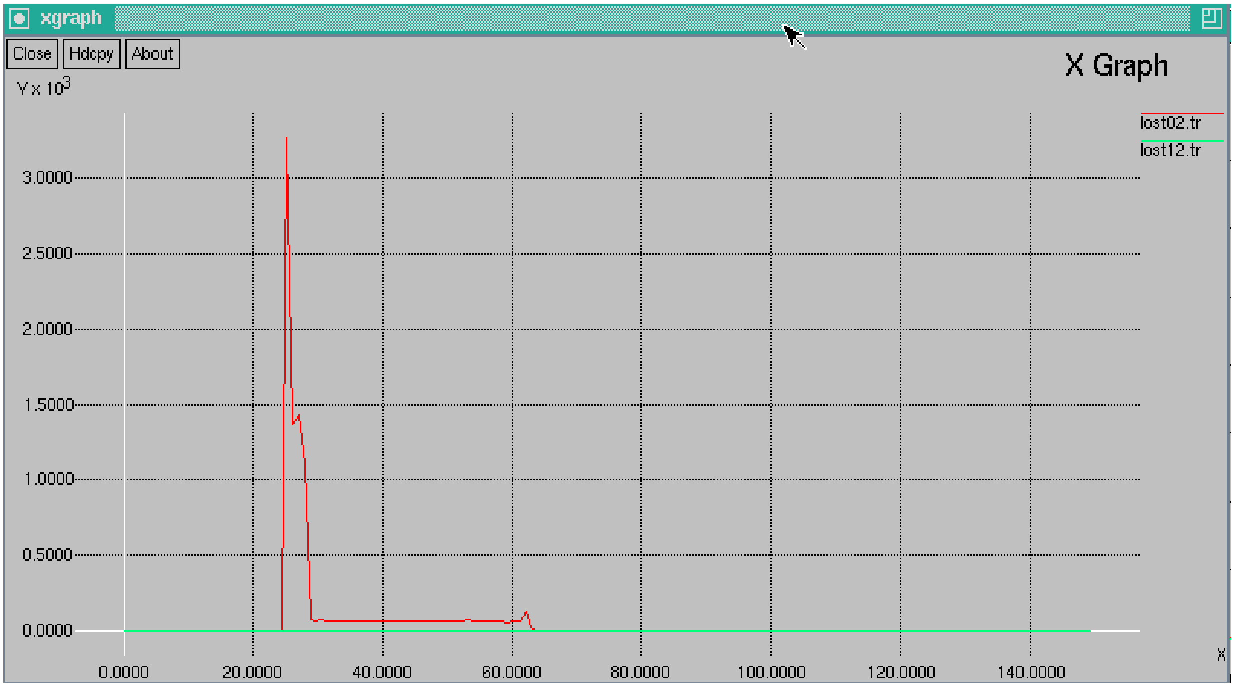

Fig. 2: Lost packet rate

Fig. 3: Data packets received

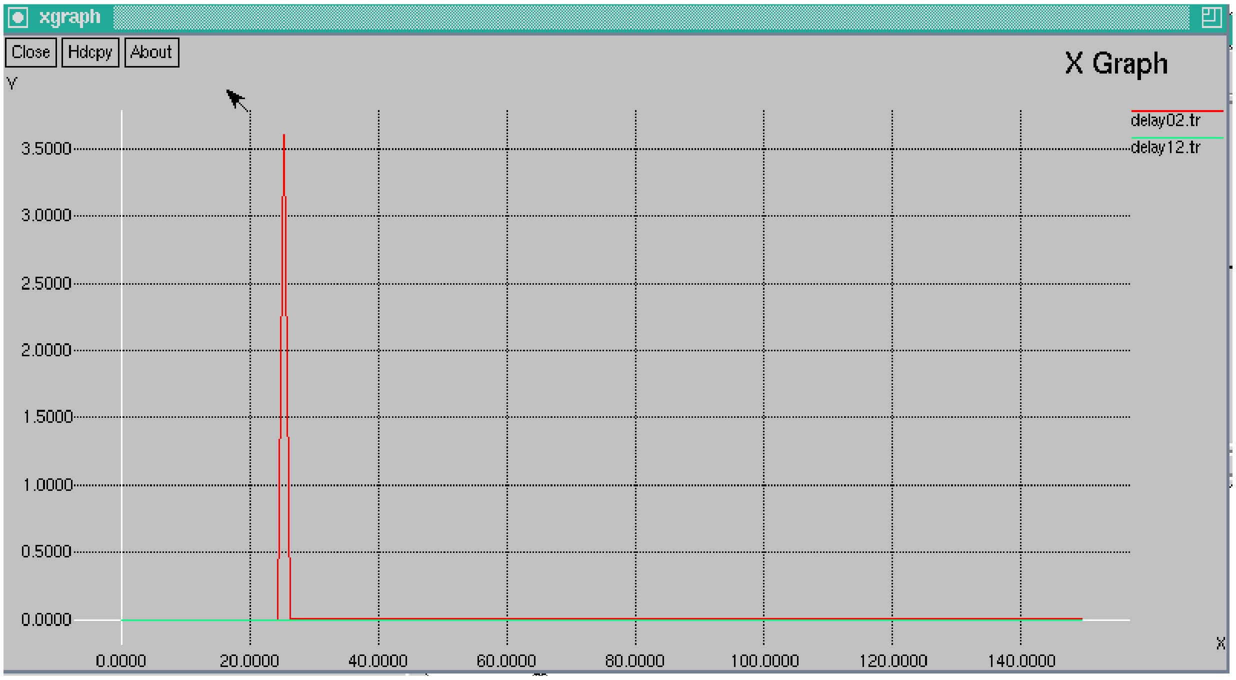

Fig. 4: Packet delay range

cache for them to use aggressively to reduce the number of control messages sent. If it has a cache entry for any destination request received, it uses the cached copy rather than forward the request. In addition, it promiscuously listens to other control messages for additional routing data to add to the cache. Figure 2 to 4 show the lost packet rate, data packets received and packet delay range, respectively.

CONCLUSION

The multicast routing protocols mentioned above all belong to the traditional multicast routing model, which can support very large multicast groups in the network. The overhead of maintaining many multicast trees or meshes in MANET is large especially when the nodes are constantly moving. Each packet covers a subset of the destination nodes within a cone area centered at the sending node. Another future direction is the interaction between the tree construction algorithms and the underlying unicast routing protocol, to obtain the actual data throughput in the ns-2 network simulator.

REFERENCES

Gharavi, H. and B. Hu, 2009. Directional antenna for multipath ad hoc routing. Proceeding of 6th IEEE Consumer Communications and Networking Conference (CCNC, 2009), pp: 1-5.

Gracia-Luna-Aceves, J.J. and M. Spohn, 1999. Source-tree routing in wireless networks. Proceeding of 7th International Conference on Network Protocols (ICNP '99), pp: 273-282.

Hauser, R., T. Przygienda and G. Tsudik, 1999. Lowering security overhead in link state routing. Comput. Netw., 31(8): 885-894.

Hu, Y.C. and A. Perrig, 2004. A survey of secure wireless ad hoc routing. IEEE Secur. Priv., 2(03): 28-39.

Hu, Y.C. and D.B. Johonson, 2004. Securing quality-of-service route discovery in on-demand routing for ad hoc networks. Proceeding of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN ’04), pp: 106-117.

Hu, Y.C., D.B. Johnson and A. Perrig, 2002. SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks. Proceeding of the 4th IEEE Workshop on Mobile Computing Systems and Applications, pp: 3.

Johnson, D.B. and D.A. Maltz, 2001. Dynamic source routing in ad hoc wireless networks. Ad hoc Networking, pp: 139-172.

Le, A.N., D.W. Kum, S.H. Lee, Y.Z. Cho and I.S. Lee, 2007. Directional AODV routing protocol for wireless mesh networks. Proceeding of IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC, 2007), pp: 1-5.

Liu, G. and K.G. Ramakrishnan, 2001. A*Prune: An algorithm for finding k shortest paths subject to multiple constraints. Proceeding of 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM, 2001), pp: 743-749.

Mieghem, P.V., H. De Neve and F.A. Kuipers, 2001. Hop-by-hop quality of service routing. Comput. Netw., 37(3-4): 407-423.

Nicholson, T.A.J., 1966. Finding the shortest route between two points in a network. Comput. J., 9: 275-280.

Perkins, C.E. and P. Bhagwat 1994. Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers. Proceeding of the ACM Special Interest Group on Data Communication. London, UK, pp: 234-244.

Perkins, C.E. and E. Royer, 1999. Ad-hoc on-demand distance vector routing. Proceeding of the 2nd IEEE Workshop on Mobile Computing Systems and Applications. New Orleans, LA, USA, pp: 90-100.