Comparative Study of Performance Evaluation for Mobile Ad hoc networks using a proxy node

G. E. RIZOS  
ATEI of Epirus  
GR-471 00 Arta  
GREECE  
georizos@teiep.gr

D. C. VASILIADIS  
ATEI of Epirus  
GR-471 00 Arta  
GREECE  
dvas@teiep.gr

E. STERGIOU  
ATEI of Epirus  
GR-471 00 Arta  
GREECE  
ster@teiep.gr

Abstract: In this paper, we present a model that incorporates nodes with role of proxy in a MANET environment. MANET is specifically characterized by high mobility of network nodes and frequent changes of direct visibility. In partially connected ad hoc networks, the destination is not always reachable. In our model we need a proxy node to relay messages to the destination. Proxies are nodes that have high probability of reaching the destination. In our scheme, when a source sends a request for a destination and the destination is not reachable, then some of the nodes in the network will choose to become the proxy for the destination. The model has been evaluated via simulation in order to confirm the feasibility of his application in real conditions. In our measurements we use one proxy node in order to broadcast the messages in the destination.

Key words Mobile Ad Hoc Networks (MANET), Routing Protocols, proxy node, ftp protocol.

1. Introduction

Mobile Ad hoc networks (MANET) are considered as promising communication networks in situations where rapid deployment and self-configuration is essential. In ad hoc net-works, nodes are allowed to communicate with each other without any existing infrastructure. Typically every node should also play the role of a router. This kind of networking can be applied to scenarios like conference room, disaster management, battle field communication and places where deployment of infrastructure is either difficult or costly [1, 2].

A classification of routing protocols distinguishes them into source routing and hop-by-hop routing protocols. In source routing, the sources compute the complete path towards the destinations, leading to a loop-free routing protocol. On the other hand, in hop-by-hop routing, each intermediate node computes the next hop itself. Thus, the hop-by-hop routing protocols reduce the chance of failed routes, a parameter of crucial importance especially in mobile networks, which are more prone to the specific error type due to the fact that their topology changes much faster as compared to wired networks. Consequently, source routing protocols - such as the Dynamic Source Routing (DSR) [4] - allow intermediate nodes (and even overhearing nodes) to modify the route, adapting thus better to the nature of mobile networks. Most MANET routing protocols such as Optimized Link State Routing (OLSR) [7] and Ad-hoc On-demand Distance Vector (AODV) protocols [5] have adopted the strategy of hop-by-hop routing.

As we see many routing protocols exist to enable communication in ad hoc networks like, AODV [5], DSR [4], DSDV [3], etc. All these protocols assume that the source and destination nodes can reach each other using a single or multi-hop path. But, there exist situations when connectivity between source and destination cannot be guaranteed always.

In this paper we study the performance analysis of a wireless ad hoc network model using Distance Vector routing protocols, and the FTP protocol.

The remainder of this paper is organized as follows: in section 2 a quickly overview of routing protocols for Ad Hoc Networks is presented. In section 3 we analyse a test-bed wireless ad hoc network scheme using the Distance Vector routing protocols. Section 4 we present the simulations results of our experimental analysis, which has been conducted through simulation experiments, while section 5 provides the concluding remarks.
2. Routing Protocols for Ad Hoc Networks

We distinguished three basic categories of protocols:

- Table-driven
- Demand-driven
- Hybrid

Table-driven Routing Protocols (proactive) Guided from table of routing (dynamic):
In this category of protocols the nodes that constitute the network allocate tables which contain elements for the routing of messages in any other node, and which are informed permanently for each change in the topology of network.

Examples of proactive protocols include the Destination Sequence Distance Vector (DSDV) [3], the Fisheye State Routing (FSR), the Global State Routing (GSR), the Wireless Routing Protocol (WRP), the Clusterhead Gateway Switch Routing (CGSR) and the STAR.

On Demand-driven Protocols (reactive) Guided at requirement (counteractive):
In the particular protocols are not maintained the all ways of each node in tables. On the contrary when exists application for mission of some message, the sender calls an algorithm for the finding of suitable path, which is maintained until the message reaches in his destination, or until is created such change in the topology of network, that automatically will cancel him because henceforth it will not be in effect [11].

Examples of reactive protocols include the Ad hoc On Demand Distance Vector (AODV) [5], the Dynamic Source Routing (DSR) [4], the Signal Stability Routing (SSR), and the Temporally Ordered Routing Algorithm (TORA).

3. The model Analysis (a01)

The following (Fig.1) test-bed configuration was chosen to represent the behaviour of a wireless ad hoc network scheme [8, 9, 10, 11], where all services are roughly categorized with data rate requirements and delay sensitivity.

```plaintext
set val(nn) 10 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set val(x) 1000 ;# X dimension of topography
set val(y) 500 ;# Y dimension of topography
set val(stop) 200 ;# time of simulation end
```

We have chosen AODV routing protocol of ad hoc network as the base of our measurements because of some simple reasons:

- It is a "popular" protocol for ad hoc networks
- Our approach needs least modification in AODV as compare to that in other schemes like DSR, DSDV.

The modelled network (Fig.2) consists of 10 nodes, one source (node 0) and one sink (node 1).

```plaintext
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;#radio-propagation model
set val(setif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) DropTail ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
```

When our nodes start moves, the model tries to exploits client/server location visibility to reorganize the locality via the dynamic election of a proxy agent [12].

```plaintext
$ns at 5.0 "$node_(1) setdest 100.0 100.0 5.0"
$ns at 5.0 "$node_(0) setdest 400.0 300.0 5.0"
$ns at 25.0 "$node_(4) setdest 330.0 50.0 5.0"
$ns at 35.0 "$node_(6) setdest 120.0 350.0 5.0"
$ns at 35.0 "$node_(5) setdest 250.0 350.0 5.0"
$ns at 45.0 "$node_(8) setdest 250.0 350.0 5.0"
$ns at 40.0 "$node_(9) setdest 450.0 235.0 5.0"
$ns at 50.0 "$node_(2) setdest 380.0 380.0 5.0"
```

Recent Researches in Communications and IT


284
The proxy takes care of searching servers by need, of forwarding client requests, and of performing multi hop routing it permits to organize solutions for client/server rebinding and service reestablishment that are scalable and mobility-transparent [7].

Once the proxy finds the server, the proxy starts forwarding service requests/responses from/to interested clients. In other words, all service messages are automatically and transparently sent through the proxy, acting as a bridge between the clients and the server.

![Network topology after some movements](image)

Figure 4. Network topology after some movements

Of course all these activities (how the information of which connections are open are developed by the lower network levels, how the neighbours informed, how the neighbours inform the computer that have the activity, the possibilities of communication of each node with the source) involve some costs in level of network messages (or snooping network mechanisms for communication mechanisms). And of course all these parameters depend from the mobility rates of model that are examined each time [6].

4. Experimental Analysis and Simulations Results

We use the file transfer protocol to measure the performance of our model. To measure the performances, we used the following metrics:

Throughput: The total bytes received by the destination node per second (Data packets and Overhead).

Goodput:

i. Goodput (In terms of Number of packets): The ratio of the total number of data packets that are sent from the source to the total number of packets that are transmitted within the network to reach the destination.

ii. Goodput (In terms of Packet Size in Bytes): The ratio of the total bytes of data that are sent from the source to the total bytes that are transmitted within the network to reach the destination.

![Throughput vs Time](image)

Figure 5. Throughput vs Time

Figure 5 shows ‘Throughput vs. Time’ where we analyzed the total bytes received by the destination node per second (Data packet and Overhead). According to this figure the AODV protocol starts off quickly and the data rate is more stable. The DSR protocol starts off quickly as the AODV however as you can see there are lots of fluctuations in the data rate. Finally the DSDV protocol takes time to start off but the data rate has lesser fluctuations.

![Goodput (Number of data packets)](image)

Figure 6a. Goodput (Number of data packets)
In Figure 6a and Figure 6b we calculated Goodput in terms of number of packets and the packet size in bytes. We can see that on an average, packets are transmitted in the network, 17% packets would be data packets for AODV, 17% for DSR, and 28% for DSDV. In term of bytes, on an average; 32% bytes would be data packets for AODV, 28% bytes for DSR, and 54% bytes for DSDV. From these data, we could deduce that; though DSDV takes time to converge, it actually is sending more data packets in number as well as in bytes than that of AODV and DSR. The rest of the percentage of each individual graph will be the overheads that contain routing packets and acknowledgements.

In the second stage of our measurements we use only the AODV routing protocol (red color line). To measure the performances we used an extension of this protocol (green color line) when source and destination are not connected. In situations where the network is connected our protocol behaves like normal AODV. When there is no connected path, we exploit mobility of nodes and use “store and forward” approach to deliver the data. Then some of the nodes called proxy nodes are selected by source to hold the data on behalf of destination. Proxy nodes acts as source and try to deliver data to the destination.

Figure 7 shows ‘Throughput vs. Time’ where we analysed the total bytes received by the destination node per second (Data packet and Overhead). According to this figure the AODV protocol starts off quickly and the data rate is stable. The protocol with the proxy starts off quickly as the AODV, the data rate is stable enough however as you can see there is a gap between 25 and 100 seconds.

In Figure 8a and Figure 8b we calculated Goodput in terms of number of packets and the packet size in bytes. We can see that on an average, packets are transmitted in the network, 17% packets would be data packets for AODV, and 98% for the extension proxy scenario. In term of bytes, on an
average; 32% bytes would be data packets for AODV, and 97% bytes for the extension.

5. Conclusions

In this work, we have compared performance characteristics of on demand driven routing protocols (reactive) for ad-hoc network routing.

In our first measurements the general observation is that the AODV behaves more effectively from the DSR as long as increase the time of discovery of source. For AODV, we can see that it adapts quickly to the change of the network and has a relatively stable throughput with a moderate goodput. So, in an application where there is a fast change in the network topology and a requirement of stable date rate, AODV is more preferable.

DSR though has a very high throughput, it actually contain less data packets and we can see that there are lots of fluctuations on the throughput curve which are not preferred in a wireless network.

DSDV turns out to have the best goodput; however, it takes time to converge. So if there is relatively lesser number of nodes in the network and the mobility is somewhat steady or slow, DSDV will work more efficiently.

In the second stage of our measurements the general observation is that the AODV routing protocol and the proxy extension have the same behave and relative stable throughput. However our proxy scenario has better goodput in both of our measurements.

From these data, we could deduce that our proxy scenario sending more data packets in number as well as in bytes than that of AODV. The gain for downloading time is greater for the AODV protocol under the used of proxy scenario; when the client informs his neighbours for the session and they start watching the server and the project.

References: