Project Proposal

Shujun Bian Weidong Guo Yue Li Yuxiang Liu Chiyao Shen

Introduction

A wireless ad-hoc network (WANET) is a particular type of computer network in which hosts are interconnected by routers using wireless communication links. Due to the increasing trend of mobile devices usage in recent years, many applications emerged by taking use of the decentralized structure of wireless ad-hoc network. In our project, we plan to study the one of the wireless ad-hoc network application: mobile ad-hoc networks (MANET). We mainly focus on four MANET routing protocols: Ad hoc On-Demand Distance Vector (AODV) Routing, Dynamic Source Routing (DSR), Optimized Link State Routing (OLSR) Protocol, Destination -Sequenced Distance-Vector (DSDV) Routing. The four protocols have some advantages and disadvantages under particular circumstances and we can analyze their performance by simulation. Specifically, we use NS-2 to do the simulation and compare the difference between each protocol.

1. Ad hoc On-Demand Distance Vector
AODV is a reactive routing protocol, combining with some features of proactive routing. It's reactive because routes are found by an on-demand approach that AODV only establishes and maintains a route when it's needed by a source. The advantage is that it's simple and traffic along the links remains low all the time. However, the disadvantage is that the time to establish a route on-demand could be costly.

2. Dynamic Source Routing
DSR is also a reactive routing protocol, and is similar to AODV as it also uses on-demand approach. The difference is DSR uses source routing the path from source to destination is stored in data packets. The advantage is similar to AODV. The disadvantage is that it's not scalable in the environment with high mobility.

3. Optimized Link State Routing
OLSR is a proactive optimized link state routing protocol. The proactive feature in OLSR is that it uses 'hello' and 'topology control' (TC) messages to broadcast the information of the link within the network. Then source can use shortest path to find the route to destination. The advantage is that the topology is stored locally and it's beneficial for computation before use. However, the disadvantage is that it needs more bandwidth because of the link state protocol nature.

4. Destination-Sequenced Distance-Vector
DSDV is also a proactive routing protocol. Each node has a routing table that contains the all destinations this node can reach, and “next hop” of each destination, ‘’number of hops’’ to each destination and the sequence number. The route is based on the information in the route. The advantage is similar to OLSR as some topology information is stored locally. The disadvantage is that it always needs resource to update the routing table, which also causes scalability problem.

General Overview/Motivation

In the process of the development of the Internet, PC Internet has become almost saturated and increase slowly while the mobile Internet is growing increasingly. With the more powerful mobile devices on the market, people start to do a lot of work on the mobile platform instead of PC platform. So we believe the future belongs to mobile devices and the mobile internet, which raises a new question that is current wireless network protocol strong enough to deal with the much more complex situation in mobile network today? Does current wireless network protocol still have good performance in today’s mobile network?

These bunch of questions motivate us to do some deep look on the current wireless network protocol especially AODV, DSR, OLSR, and DSDV.

We would repeat the experiment in [1], while create some new scenario and use different test data to measure all 4 protocols. We would evaluate the performance of AODV, DSR, OLSR and DSDV, which help us to get a conclusion such that which protocol is better for today’s mobile network, which protocol has a better performance in average PDR. Through this kind of conclusion, we might get some ideas about the design of the future mobile network protocol.

Implementation Details

NS2 is an open-source event-driven simulator designed specifically for research in computer communication networks [2]. NS2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTcl). While the C++ defines the internal mechanism (e.g. network protocol stack) of the simulation, the OTcl sets up simulation by assembling and configuring the objects as well as scheduling discrete events.

For each protocol, we implement it by inheriting a class called Agent which creates, transmits, receives, processes, and destroys routing packets. Basically, we create .cc and .h file for main definition of the protocol, packet header, route entry and routing table and the buffer to store data. Then, we will configure the network in OTcl scripts. Basically, we specify the number of nodes, how to deploy these nodes, protocols used in different layers, randomness, transmission and receiving power, antenna type, etc. In addition, we will turn on the ‘trace’ program to trace the packets in different layers. This ‘trace’ program will automatically generate data files in standard format and we will use ‘gawk’ to extract the useful information(e.g. packets drop rate, traceroute RTT, throughput, etc.) from these files. Moreover, we will use ‘Nam’ (an animation tool) to view network simulation traces and real world packet traces.

The following table describes the detailed simulation parameters:

Parameter	Description
Number of nodes	10, 20, 30, 40, 50
Topology	Random
Deployment area	1000(sq. m.)
Channel	WirelessChannel
Model of radio propagation	TwoRayGround
MAC	IEEE 802.11
Antenna	OmniAntenna
Routing protocol	AODV/DSR/OLSR/DSDV
-rxPower	0.3w
-txPower	0.6w
Pause time	0s, 30s, 90s, 120s, 150s
Connection Type	CBR/UDP

Plan of Action

We plan to complete our project in the following steps:

1. Preparation (2 weeks)
    - Read literatures and resources about the routing protocols

- Learn basic techniques of NS2 simulator and AWK scripts
2. Implementation of routing protocols (4 weeks)
    - Implement AODV/DSR/OLSR/DSDV
    - Debug and test
3. Analyzation of performance (2 weeks)
    - Generate data sets for test
    - Get performance of each protocol and compare the results
    - Analyze the results
4. Conclusion and summarization (1 weeks)
    - Summary the analyzed results and get conclusions
    - Consider about the future work
5. Presentation (1 week)
    - Prepare for the in-class presentation

Evaluation and Testing Methodology

Based on the outputs of above experiment, performance evaluation is carried out by varying two parameters, i.e. density(**Number of nodes** divided by **Deployment area**) or pause time, while keeping the other parameters constant. The performance of a certain protocol is judged by **Packet Drop Rate**, **End-to-End Delay** and **Throughput**. For each graph, four protocols, AODV, DSR, OLSR, and DSDV will be compared.

Network Load Analysis:

X-Axis	Y-Axis
Density	Packet Drop Rate
Density	End-to-End Delay
Density	Throughput

Mobility Analysis:

X-Axis	Y-Axis
Pause time	Packet Drop Rate
Density time	End-to-End Delay
Density time	Throughput

What we’ve done in the experiment is basically replicating a previously conducted one, while adding some new elements. One parameter is adjusted to be the density, attempting to combine the two variations, number of nodes and deployment area. We aim at having a comprehensive view of the work presented in the paper. Furthermore, we would like to verify if similar experiment outputs can be achieved, and make deep analysis if otherwise.

Reference

[1] Mohapatra, S., and P. Kanungo. "Performance analysis of AODV, DSR, OLSR and DSDV routing protocols using NS2 Simulator." Procedia Engineering 30 (2012): 69-76.
[2] Teerawat Issariyakul and Ekram Hossain. 2012. *Introduction to Network Simulator Ns2* (2nd ed.). Springer Publishing Company, Incorporated.