University of Victoria

CSC 579: Advanced Computer Networks: Overlay and Peer-to-Peer Networking

Topic: Ad-hoc Network Simulation for disaster management

PROJECT PROPOSAL

GROUP MEMBERS

Name: Alice Irankunda, Student ID: V00902687
Name: Noel Khaemba, Student ID: V00902707

Chapter 1

1.1 Abstract

In this research project, the various ad hoc network categories are explored as a way of coming up with a viable solution of alerting the community/rescue team in the occurrence of a disaster. The project involves clearly understanding ad hoc network functionality, then coming up with OMNET++ simulation for various networks. It will have a distributed network architecture for rescue officers. It will be based on some recent published paper on the same topic by focussing on improving what has been proposed.

In case of disaster, the reporting and rescue is one of the urgent action to be taken rapidly and carefully. To locate where the disaster has taken place which is one useful information for rescue officers. This project will implement a prototype that works towards reducing the number of people who suffer during disaster through a quicker communication. It will be simulation in OMNET++ which will be used to analyse the performance metrics such as packet loss ratio and delay.

1.2 Background information

Wireless network is a way computers, phones and other telecommunication devices get connected for sharing data and communicate using wireless. Wireless ad-hoc network(WANET) is known as a wireless mesh network based on its topology. MANET is a type of WANET which is able to be routed, this behaviour allows MANET to move independently in any direction and to be able to change the links many times. It is noted to have many advantages for mobile users for example, it is self configured without requiring infrastructure based on its capability of delivering connection wirelessly [**?**].

In this paper, ad hoc network (infrastructure less network) is studied in depth with reference to the previous research work. In Infrastructure less network, the mobile devices dynamically create their network as they move, the nodes act as routers and they are not controlled by base stations. For infrastructure network on the other hand, the mobile nodes are fixed to base stations and as they move around they get into different base station ranges [**?**].

In the previous work done under this area, there are different disaster/emergency scenarios of which the researchers suggested solutions using the mobile ad-hoc network (MANET) [**?**]. According to [**?**], Ad hoc wireless network is divided into three branches; Wireless mesh network (WMN), Mobile ad-hoc network (MANET) and Wireless sensor network. Much research has been done on MANET unlike the other two. There's need to explore other options under Ad hoc network for instance using the wireless sensor mechanism or Wireless mesh network.

1.3 Problem statement

The problem to be solved is when there's emergency/disaster and the communication network is ruined/damaged. The infrastructure less network is important because it acts as a solution especially when there is a disaster like fire or air-bone disease outbreak and the existing network infrastructure is destroyed. It helps provide services like search and rescue and recovery from disasters. This can be useful particularly in rural areas and in informal settlements.

1.4 Proposed solution and approach

The study seeks to consider different Ad-hoc networks and findout how disasters can be quickly published through the networks and reach out to rescuers.

- OMNET++: Network simulation tool with MN

The goal is to provide needed disaster information to help rescue officers. All nodes in the network will have equal priority (peer-to-peer networking).

1.5 Project schedule

‘

Date	**Tasks**
14th - 21st Feb	Environment setup
22nd - 28th Feb	Introduction/ Literature writing
01st - 14th March	Implemetation of MANET function in OMNET++
15th - 21st March	Testing and debuging
22nd -28th March	Methodology/ Results/ Documentatation report writting
29th - 31st March	Compulation and future work

Chapter 2

2.1 How does MANET works

MANET nodes have capability to work as host as well as routers. each host has routing information of its connected neighbours. When a node want to send a message, it broadcasts the destination address through its transmission network range. If none of its neighbours matches to that destination address, all nodes in neighbourhood broadcast that message until it reaches to the destination. [REF1]

There are protocols designed to manage communication between nodes. AODV is one of routing protocol which hepls nodes to communicate evenif there are not directly connected i.e not in close approximation. Nodes which are not neighbours dont have each others information but they rather have network information. They use Route Request (RREQ) messages from source node to destination. RREQ message carring the following information:

- Source/destination address
- Unique message Id
- Lifespam

When RREQ has been sent, it can only be received by nodes in neighbourhood of the sender. They cross check it to make sure if it doesnt belong to them. if not, they broadcast RREQ message to their transmission network range. whenever the message reaches to its destination, the receiver node sends a reply until it reaches to the sender. The sender has a time out period called lifespam and if the receiver reply doesnt reaches to the sender within that time, it retransmit RREQ message. Lifespam keeps on decreasing and the message is retransmitted when it reaches zero[REF1].

During retransmission, the RREQ get a new ID and the lifespam is set to a longer period compared to the previous.

2.1 WHY OMNET++

Compared to other researchers on disaster rescue for MANET, OMNET++ has been widely used after ns-2. OMNET++ is a network simulation tool which is flexible to be used for either setting up new network simulations or changing existing models. It is flexible and deliver performance checking, assessing routing quality and delay calculations. In this project we will set up networks and perform simulation by considering different metrics.

In this part, we will be emphasizing on existing applications of disaster alert, discuss on their weaknesses and make some improvement suggestions .

Chapter 3 : Tools and Implementation

As discussed in previous sections, this project will be implemented in OMNET++. It is an open source free for research / academic use. OMNET++ network simulation tool has a framework known as INET which has built in models and libraries as examples simulations. However, we will build our Ad hoc network composed by mobile nodes. Our network will compare the performance using cluster network as well as independants nodes networks.

3.1 Communication methods

The communication between nodes can be done into two different ways. Nodes start by their neighbours or can be grouped into clusters. The direct node communication, node starts by its neighbours until its message is spread into the whole network as explained in the prvious chapter. Communication by Cluster based MANET has advantage of a dynamic resource allocation and time saving due to the use of CBRP(Cluster Based Routing Protocol) and Dynamic Source Routing (DSR) protocol.

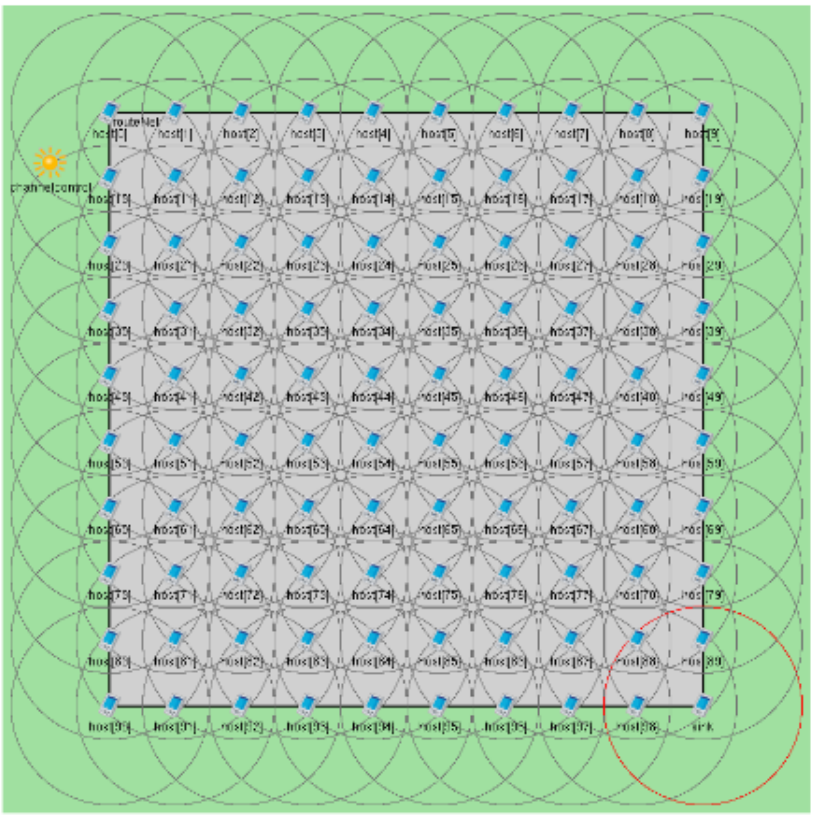


Figure 1: Cluster based MANET nodes

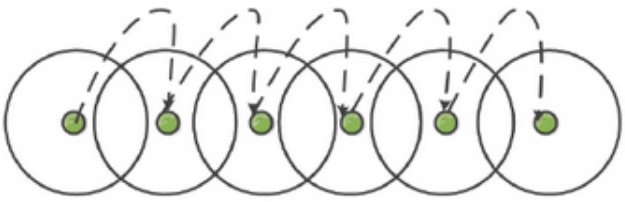


Figure 2: Direct communication

For our project, we will compare performance of both architectures. We will be taking care on the spread of the message and packet loss ration relating to each method. This speedy the communication when the disaster occur.

3.2 Protocols behaviours

MANET for disaster management through OMNET++ can be performed using different protocols such as OADV, DYMO, DSR etc. These protocols are supported by INET platform. For our project, we will be imphasizing on OADV, CBRP and DRS.

3.2.1 OADV

Ad hoc On-Demand Distance Vector (OADV) is a one of routing protocol used in wireless networks and MANET which has been developed in July 2003. AODV uses four main states such as sleep, idle, send and receive. The diagram below demonstrate the transitions between states.

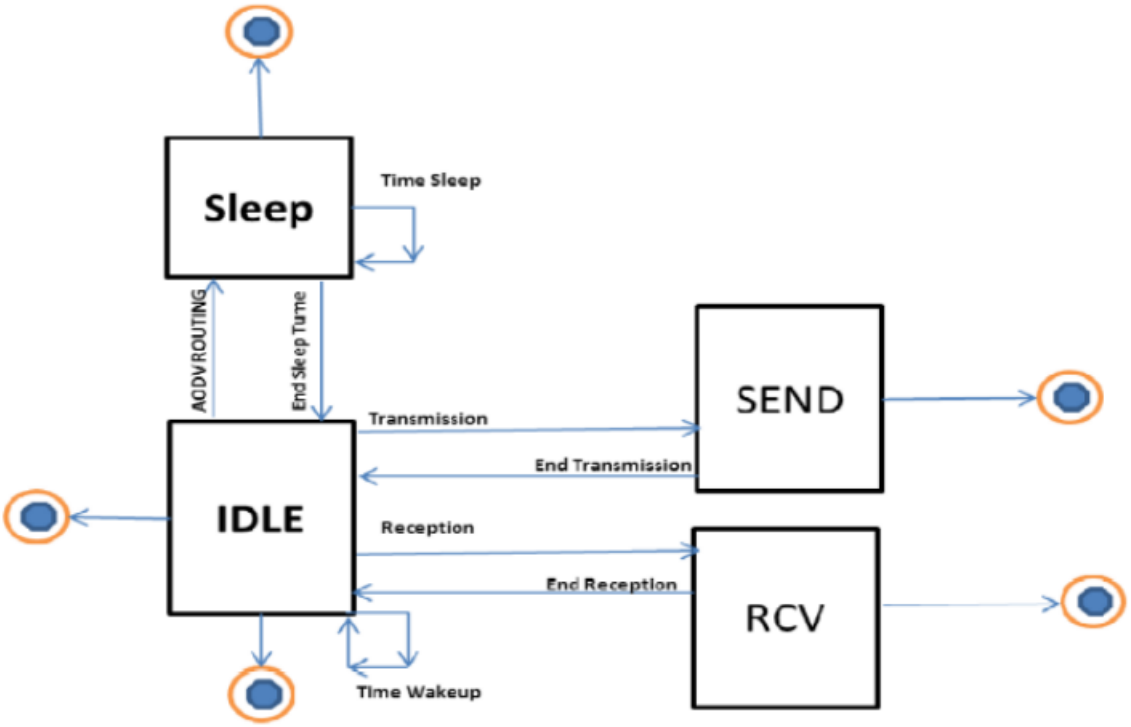


Figure 3: State stransitions of OADV

- Sleep state: when the station is off. Nodes receive signals from the cluster header hence communication is not possible.
- Idle state: is also called listen to idle state, it is when the node is ready to send/ receive.
- Send: when a node is sending to the node in idle state
- Receive: when a node is receiving from a node in idle state.

When two or more nodes want to send data to their neighbors simultaneously cause collision. To avoid collision, OADV uses the same attitude as Distributed Coordinated Function (DCF).

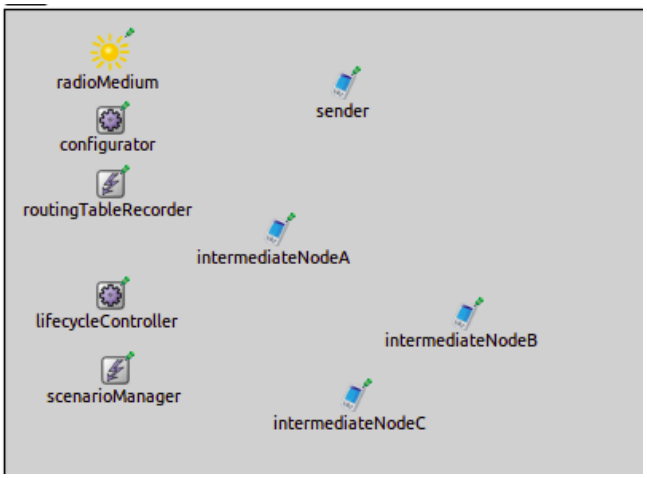
One of the challenges when using OADV is the battery capacity. However OADV is mainly used in emergency conditions, the users face the problem of keeping batteries with enough capacity.

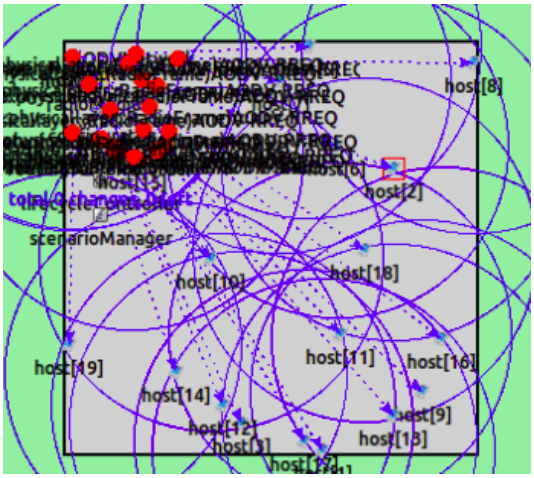
3.3 Implementation

The communication is initiated by the station by sending the ”hello” message to nodes for checking if there are active. when the hello message finds a node in idle state, the node replies and the communication starts. The station is considered as a rescue office where disaster might be reported.This senario can be explained in the following

sample xml codes

```
<scenario>
<at t="30">
<tell module="lifecycleController" target="host[16]" operation="NodeShutdownOperation"/>
</at>
<at t="30">
<tell module="lifecycleController" target="host[7]" operation="NodeShutdownOperation"/>
</at>
<at t="30">
<tell module="lifecycleController" target="host[10]" operation="NodeShutdownOperation"/>
</at>
<at t="30">
<tell module="lifecycleController" target="host[17]" operation="NodeShutdownOperation"/>
</at>
<at t="30">
<tell module="lifecycleController" target="host[14]" operation="NodeShutdownOperation"/>
</at>
<at t="30">
<tell module="lifecycleController" target="host[19]" operation="NodeShutdownOperation"/>
</at>
<at t="30">
<tell module="lifecycleController" target="host[10]" operation="NodeShutdownOperation"/>
</at>
</scenario>
```


radioMedium
sender
configurator
routingTableRecorder
intermediateNodeA
lifecycleController
intermediateNodeB
scenarioManager
intermediateNodeC


host[8]
host[2]
scenarioManager
host[10]
host[18]
host[19]
host[11]
host[16]
host[14]
host[9]
host[12]
host[13]
host[3]