

DEVELOPMENT AND EVALUATION OF REACTIVE ROUTING PROTOCOLS FOR WIRELESS SENSOR NETWORKS

M.Anitha¹, M.V.Sudhakar²

^{1,2} ECE Department

¹ M.Tech Student

² Assistant Professor

¹ anithaece777@gmail.com

ABSTARCT:

This paper evaluates the performance of prominent on demand routing protocols, which are Ad Hoc On Demand Distance vector Routing(AODV),Dynamic Source Routing(DSR)for wireless sensor networks Results obtained from simulations demonstrates that even though AODV and DSR share a similar on demand behavior, the differences in protocol mechanics can lead to significant performance differentials. The performance differentials are analyzed using varying network load and network size using PHENOM ATTRIBUTES NS-2 is used as a platform for simulatingAODV&DSR under various conditions.PHENOM routing protocol is designed especially for wireless sensor networks. Wireless Sensor Networks (WSNs) are characterized by multi-hop wireless connectivity, frequently changing network topology and need for efficient routing protocols.

Key words: phenom, AODV, DSR, Sensor networks, Ns-2.

1.INTRODUCTION

Latest advances in processing, storage, and communication technologies have advanced the capabilities of small scale and cost effective sensor systems to support numerous applications[8]. Much of the research in sensor networks is funded for military tasks, but applications such as forest fire detection and rush-hour traffic monitoring exemplify the versality envisioned for this rapidly expanding technology[11]. In this Context We define a sensor network as an autonomous, multi-hop,wireless network with nondeterministic routes over a set of possibly heterogeneous physical layers. In other words, routing will occur throughout the network at nodes configured in ad hoc mode.

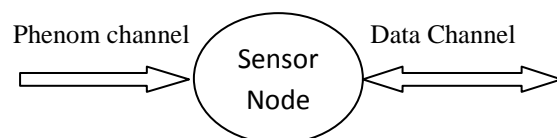


Fig 1: Sensor network Model

We are including the ns-2 simulation environment in these evaluations. The primary purpose of this project is to establish a foundation in ns-2 for simulating sensor networks. This foundation, illustrated in figure1, consists of dual homed sensor nodes that are tapped into an 802.11 channel for communicating with other networks\ stations and into a phenomenon channel for detecting some physical phenomenon. It is an effort to aid the analysis of various sensor network configurations under the demands of specific sensor applications..

2. NS-2 OVERVIEW

The ns-2 simulation environment [7] offer great flexibility in investigating the characteristics of sensor networks because it already contains flexible models for energy constrained wireless ad hoc networks. The wireless model also includes support for node movements and energy constraints

3..SETTING UP A SENSOR NETWORK IN NS-2.

Setting up a sensor network in ns -2 follows the same format as mobile node simulations. The best way to create your own simulations is to modify one of the examples distributed with our code.

1) Configure a phenomenon channel and data channel: phenomenon nodes should emanate in a different channel than sensor nodes in order to avoid contention at the physical layer. all phenomenon nodes should be configured on the same channel, even if they are emanating different types of phenomena[13].

```
Set chan-1-[new $val(chal)]
Set chan-2-[new $val(chan)]
```

2) Configure a MAC protocol for the phenomenon channel. Choose a MAC layer to use for emanating phenomena over the phenomenon channel using 802.11 is not appropriate ,since phenomena should be emanating without regard to collisions or congestion control .we suggest using the basic “Mac” class instead ,shown as follows:

```
set val(mac) mac/802.11
set val (PHENOMmac) mac
```

3) Configure phenomenon nodes with the PHENOM”routing”protocol: use node-config, just like with mobile nodes, but specify PHENOM as the routing protocol so the phenomenon is emanated according to the methods defined in phenom\phenomena .cc. also be sure to configure in the channel and MAC layer previously specified for phenomena broadcast. A sample node configuration statement is shown below.

```
$ns- node-config\
-adhocrouting PHENOM\
-channel $chan-1-\
-ll type LL\
-macType $val(PHENOM mac)\
-if qtype Queue/droptail/priqueue\
-ifqlen 50\
-anttype Antenna/omniAntenna\
-phytype phy/wirelessphy\
-topoinstance $ topo \
-agentTrace ON\
-routerTrace ON\
-macTrace on\
-movement trace on\[2]
```

```
-propType propagation/TwoRayGround
```

4) Configure the phenomenon node’s pulse rate and type. The two parameters that can be used to customize phenomena are listed below. They are both optional.

a) pulse rate FLOAT

- FLOAT must be a real number.
- Describes how frequently a phenomenon node broadcasts its presence
- Defaults to 1 broadcast.

b) phenomenon PATTERN

- PATTERN must be any one of the following keywords: CO, HEAVY-GEO, LIGHT-GEO, SOUND, TEST-PHENOMENON corresponding to carbon Monoxide, heavy seismic activity, light seismic activity, audible sound, and some other generic phenomenon.
- This option is mostly useful for simulations involving multiple phenomenon nodes, so that it is easier to distinguish who a sensor node is detecting by looking at the ns trace file.
- Defaults to TEST-PHENOMENON

The following source code illustrates how these phenomenon parameters can be set to emanate carbon Monoxide 10 times per second:

```
[$node-(0) set ragent-] \
pulse rate 0.1 ;
[$node-(0) set ragent-] \
phenomenon CO ;
```

5) Configure sensor nodes. Sensor nodes must be configured with the-PHENOMchannel attribute and the -channel attribute. PHENOMchannel must be the same as the channel you configured the phenomenon node with. The other

channel is the channel that will be used communicating sensor reports. Sensor node configurations must also specify a MAC protocol (such as Mac/802-11) for the channel shared with other wireless node. This is done with the –

PHENOMmacTYPE and –macTYPE attributes. – PHENOMmacTYPE should be the same as the macType used in other nodes participating in the IP network. For example

```
$ns-node-config \
  -adhocrouting $val(rp) \
  -channel $chan-2- \
  -mactype $val (mac) \
  - PHENOMchannel $chan-1- \
  - PHENOMmacType $val (PHENOMmac)
```

If desired, a sensor node can be configured so that a specified amount of energy will be deducted from its energy reserve each time it receives a phenomenon broadcast[5]. To set this up, include the following parameters in the sensor node's node-config routine:

```
-energy model Energy Model \
-rxpower 0.175 \
-txpower 0.175 \
-sense power 0.0 \
-initial Energy 0.5
```

Where

- RxPower .175 indicates 175mw consumed for receiving a packet of arbitrary size,
- txPower .175 indicates 175mw consumed for transmitting a packet of arbitrary size,
- sense Power .00000175 1.75μw consumed for receiving PHENOM broadcast packet, and
- initial Energy 5 indicates a total energy reserve of 5j.

IMPORTANT CAVEAT:

Ns-2's energy consumption model utilizes color to illustrate when a node is about to exhaust its energy. In order to avoid confusion in the nam visualization, the node coloring that is part of sensor application should be disabled with the DIS-ABLE-COLORS.

These parameter are listed below:

- SILENT-PHENOMENON is the seconds of quiescence required for a sensor to go off it's alarming state. Example:
#define SILENT-PHENOMENON .2
- MWSG-SIZE is the size (in bytes) of the messages to send to the gateway, or data collection point, or whatever you want to call the sink node attached to this sensor node (over UDP,for example). Example:
#define MWSG-SIZE 256
- TRANSMIT-FREQ is the frequency with which a sensor node triggered by PHENOM packets will send a message to the sink node. Units are in seconds, so a message of size MWSG-SIZE bytes will be transmitted to the gateway node once for ever TRANSMIT_FREQ seconds in which the sensor node has received one or more PHENOM packets
Example:
#define TRANSMIT_FREQ 0.1

6) *Configure non-sensor nodes, such as data collection points, or gateways for the sensor network.* Nodes that are not sensor nodes or phenomenon nodes should not be configured with a PHENOM channel, since their only interface is to the MANET network. This is done with the – PHENOM channel “OFF” attribute, as follows:

```
$ns_ node-config \
  -adhocRouting $val(rp) \
  -channel $chan-2- \
  -PHENOMchannel “off”[7]
```

7) *Attach sensor agents.* Create a sensor agent for each sensor node, and attach agent to its respective node. Also, specify that all packets coming in from the PHENOM channel should be received by the sensor agent. In the following example, \$i would represent the node number for the sensor node currently being configured [13].

```
Set sensor_($i) [new \ Agent /sensor Agent]
$ns_ attach-agent $node_($i) \ $sensor_($i)
# specify the sensor agent
# as the up-target for the
# sensor node's link layer
# configured on the PHENOM
# interface, so that the
# sensor agent handles the
# received PHENOM packets
# instead of any other agent
# attached to the node.
[$node_($i) set 11_(1) \
up-target $sensor_($i)
```

8) *Attach a UDP agent and sensor application to each node(optional).* How the sensor nodes react once they detect their target phenomenon is a behavior that should be defined in the sensor application. One such application might involve sensor nodes alerting a data collection point via UDP with information about the phenomenon. The following example illustrates how an application like that could be setup. Again, \$i represents the node number for the sensor node currently being configured.

```
Set src_($i) [new Agent/UDP]
$ns_ attach_agent $node_($i) \
    $src_($i)
$ns_ connect $src_($i)$sink
Set app_($i) [new \
Application/sensorapp]
$app_($i) attach_agent $src_($i)
```

9) *Start the sensor application.* The sensor node can receive PHENOM packets as the sensor agent is attached to the node. since the sensor agent does nothing but notify the sensor application of received phenomenon broadcasts, the sensor node does not visibly react to PHENOM packets until the sensor application has been attached and started. The following example shows how to start a sensor application:

```
$ns_ at 5.0 "$app_($i) start /
$sensor_($i)"
```

4.PROOF OF CONCEPT:MANET ROUTING WITHIN A DYNAMIC SENSOR NETWORK

This experiment begins to show the types of results one can achieve from sensor network simulations in ns-2. suppose we would like to characterize how well AODV, DSR scales with the size of a sensor network running the sensor application, we will look at networks of stationary sensors with infinite energy placed in a grid with d units of distance between adjacent nodes. the network size will vary between 50 and 2000 sensor nodes, we will limit the broadcast range of 802.11 radios and the range of the phenomenon to $\sqrt{2}d^2$, as shown in figure 8. since we are using the the two_ray ground radio propagation model, nodes within this boundary always receive the broadcast and nodes outside never receive the broadcast. We will exit the network with a single phenomenon node that slowly travels near the perimeter of the network. As the grid density increases, the phenomenon will encounter sensor nodes more frequently, thus as the grid density increases, AODV will flood more route requests through the network. as the network becomes more congested, we should observe higher latency and higher loss rates in sensor reports delivered to the stationary data rate and loss fraction statistics.

This experiment's purpose as a proof of concept for our ns-2 extensions is complete. we have captured details of the AODV routing protocol through multiple sensor network simulations, and those results follow our expectations. A more useful result would involve classifying AODV routing as better or worse than some other routing protocol, but this work is left for future research, As it stands, we have demonstrated AODV performance in large networks of up to 2000 sensors exited by a mobile phenomenon. Reproducing the traffic patterns exhibited in these simulations would be extremely difficult without using similar extensions to ns-2.

5. SIMULATED RESULTS

After simulation, we obtained results for AODV, DSR of their packet rate delivery ratio, Rate-drop, rate-delay and nam outputs as shown in the following fig's 1, fig 2, fig 3, we will compare the performance of results.

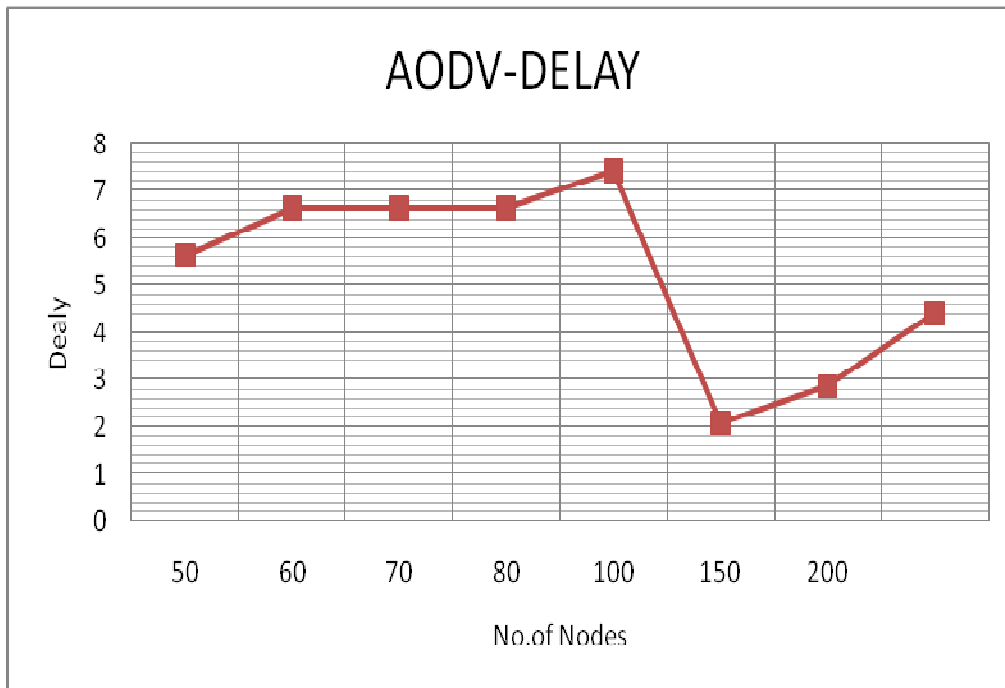


Fig 1.: Performance of AODV

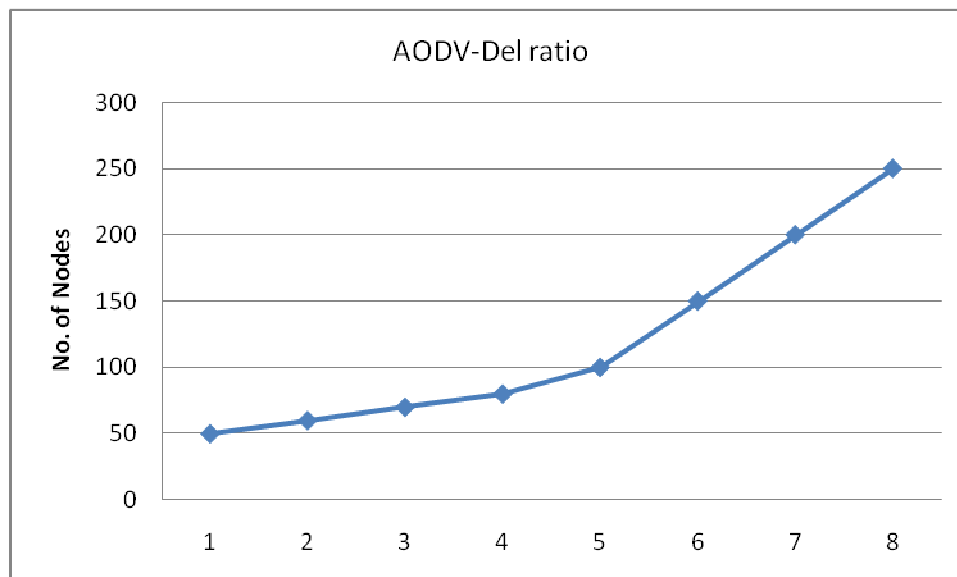


Fig2 Del ratio of AODV

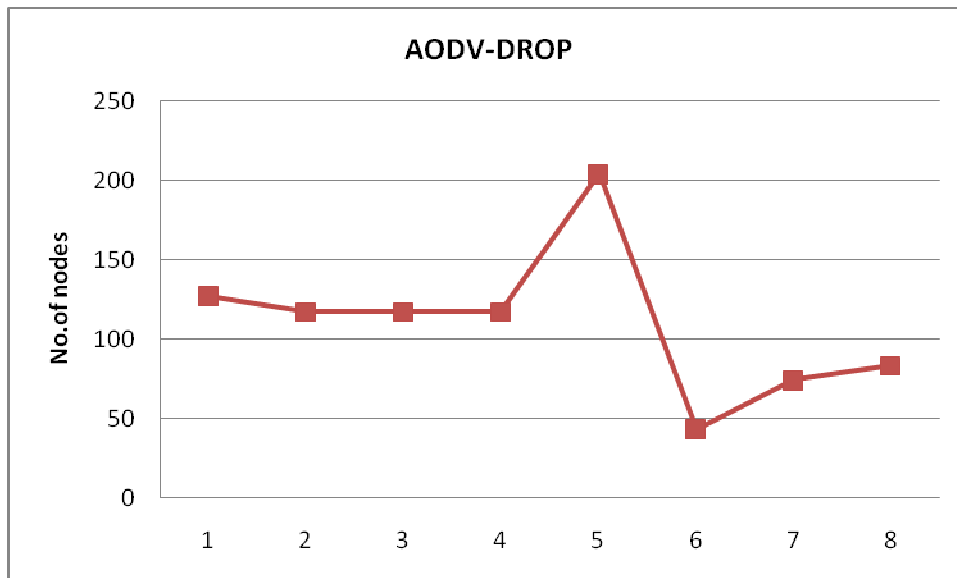


Fig 3 Drop of AODV

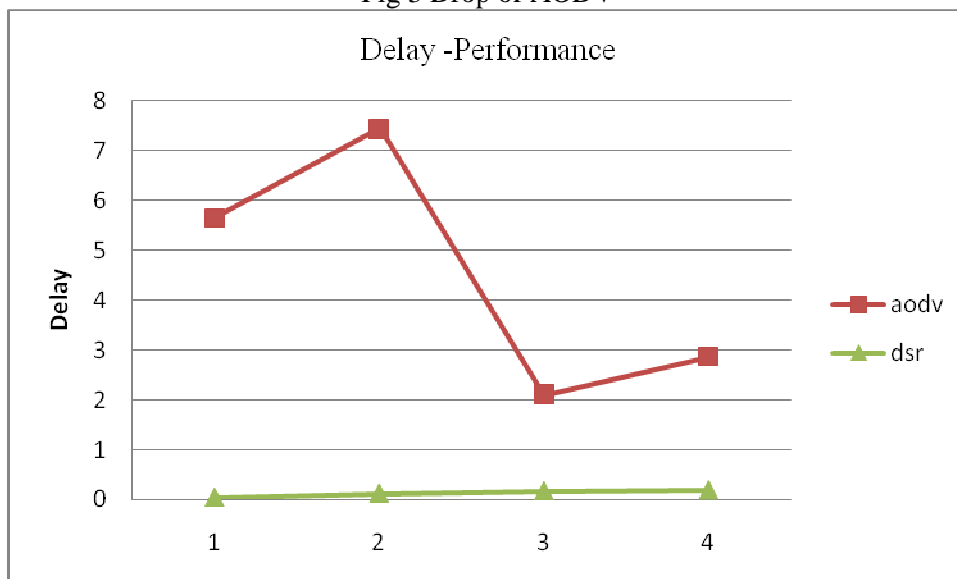


Fig 4 Delay -Perofrmance of Aodv and DSR

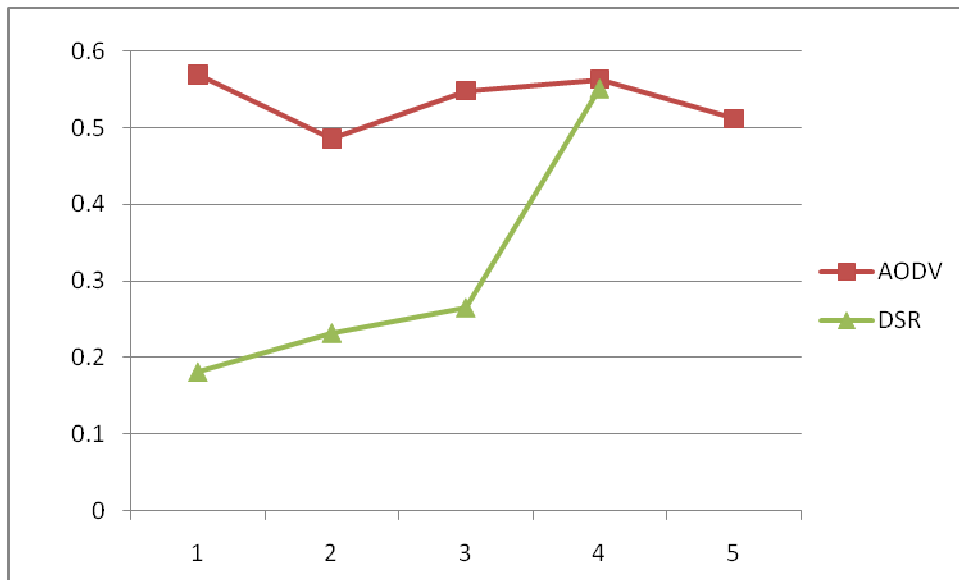


Fig5 Del ratio performance of AODV and DSR

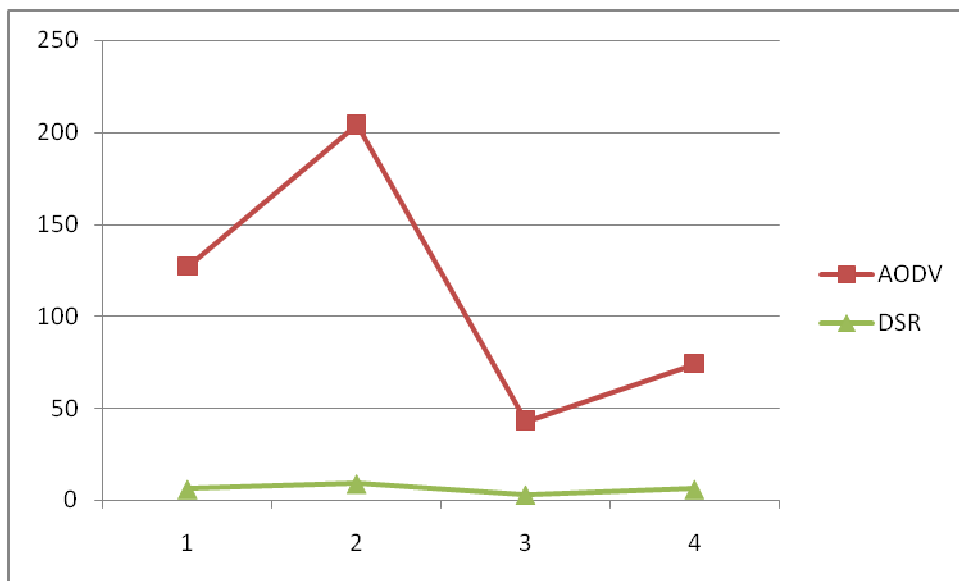


Fig 6.Drop Performance of AODV and DSR

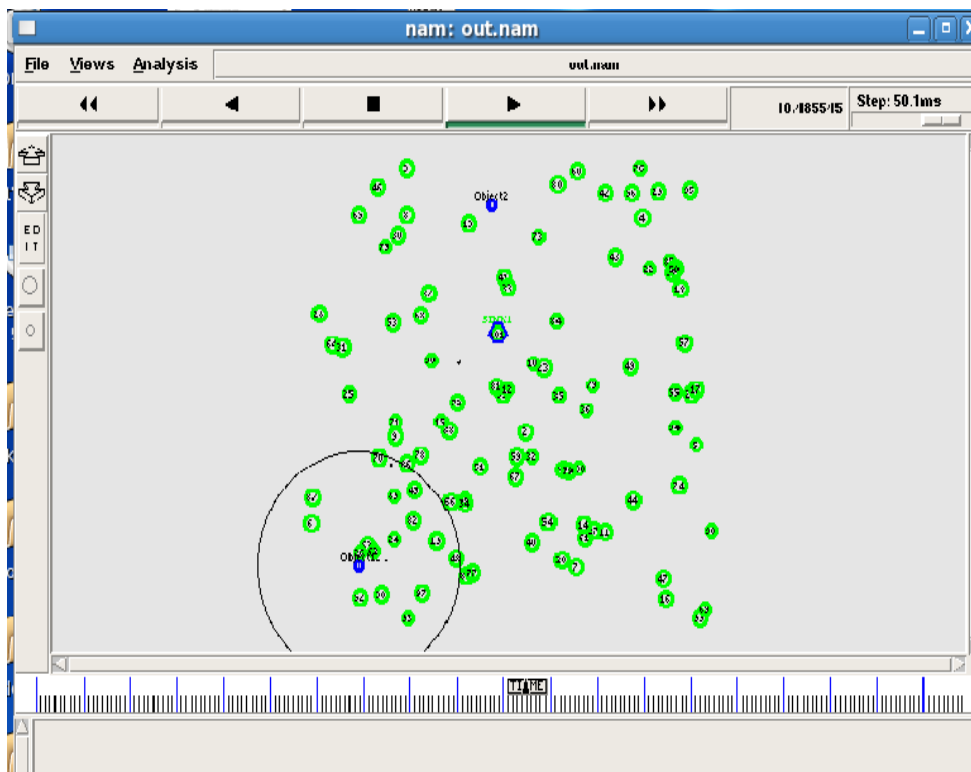


Fig7. AODV -NAM output

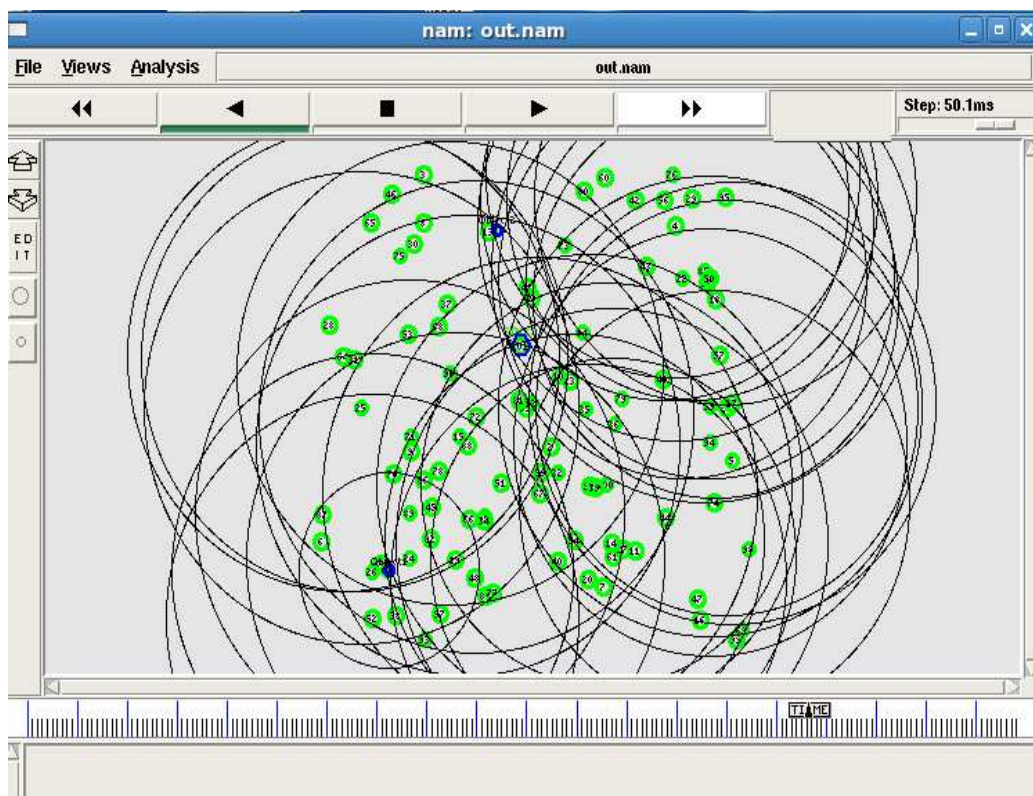


Fig.8 DSR- NAM Output

6. CONCLUSION

This paper is being simulated and solutions are realized in Ns-2.26(Network Simulator) to prove assumptions considered in current work. Ns-2 is an object-oriented event-driven simulator with extensive support for simulation of MAC protocol. This work facilitates our objective to evaluate how well current routing protocols support the requirements of various sensor network applications. The reactive routing protocol significantly reduces the total number of route request packets, increase the packet delivery ratio, decreasing end- to-end delays for the data packets, lower control overhead, fewer collisions of packets, supporting reliability and decreasing the power consumption.

REFERENCES

- [1] The Network Simulator-ns-2, <http://www.isi.edu/nsam/ns/>
- [2] NRL's OLSR implementation for ns-2. <http://pf.itd.nrl.navy.mil/projects/olsr/>
- [3] Wei Ye John Heidemann, Deborah Estrin. "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," in the proceedings of the IEEE INFOCOM, 2002.
- [4] H. Yang, B. Sikdar. "A Protocol for the Proceedings of the First IEEE International Workshop on Sensor Network Protocol and Applications, pp. 71-81, Anchorage, Alaska, May 2003.
- [5] The ns Manual.<http://www.isi.edu/nsnm/ns/ns-documentation.html>
- [6] The Doxygen documentation system. <http://www.doxygen.org>
- [7] nRL's Sensor Network Extension to ns-2. <http://nrlsensorsim.pf.itd.nrl.navy.mil/>
- [8] J. M. Khan, R. H. Katz and K. S. J. Pister. "Mobile Networking for Smart Dust," in the ACM/IEEE International Computing and Networking (Mobicom 99), Seattle, WA, August 1999.
- [9] μ -Adaptive multi-domain power aware sensors at MIT. <http://www-mtl.mit.edu/reaserch/icsystems/uamps/>
- [10] wireless Integrated Sensor Networks at UCLA. <http://www.janet.ucla.edu/wins/>
- [11] "h-MAS; a Heterogeneous, mobile, ad-hoc Sensor-Network Simulation Environment," in the Seventh Annual Swarm Users/researchers Conference, Notre Dame, Indiana, April 2003.
- [12] Park, Savvides, Srivastava. "SensorSim: A Simulation Framework for Sensor Networks." <http://nes1.ee.ucla.edu/projects/sensorsim/>
- [13] Ian Downard " Simulating Sensor Networks in NS-2", NRL CODE 5523, Washington DC,20375-5337.