

Assignment on NS-3

AODV (Ad hoc On-demand Distance Vector) is a routing protocol designed for mobile ad hoc networks (MANETs) and other wireless ad hoc networks. In this assignment, we will experiment on this routing protocol.

Task 1

In this task, we will simulate Wireless High Rate Mobile networks (e.g. 802.11), with AODV as the routing protocol. The parameters that need to be varied in the simulations are:

- Number of nodes: 20, 40, 70, 100
- Number of packets per second: 100, 200, 300, 400
- Speed of nodes: 5 m/s, 10 m/s, 15 m/s, 20 m/s

In all cases, you need to measure the following metrics and plot graphs –

- Network throughput
- End-to-end delay
- Packet delivery ratio (total number of packets delivered to end destination / total number of packets sent)
- Packet drop ratio (total number of packets dropped / total number of packets sent)

Note that each node must have an application with a flow. Here, flow means packet exchange between two applications. So, in the case of increasing nodes, you need to increase flows too. You can make reasonable expectations about all other parameters.

Task 2

In this task, you will modify the existing AODV protocol to implement a modified AODV protocol. In [this](#) review paper, you can find many modifications suggested in the literature. You are free to implement any of them (or a tweak of your own). However, we would suggest implementing the [R-AODV](#) protocol.

A really bold move would be to implement the AODV protocol for IPv6 (check [this](#) for details). However, it has not been done before. If you are interested to pursue this direction, [email me](#) beforehand. If we could successfully merge it to NS-3, it will be a meaningful contribution.

Task 3

Repeat Task 1 for your modified AODV protocol.

Implementation Details

1. For task 1 and 3, ensure that all input parameters can be specified through the command line. Use bash scripts to run simulations. You can use [GNUplot](#) to generate the plots. Or you can take output in a csv/xml file and use your favorite plotter (like matplotlib, seaborn, Google sheets etc).
2. You can use FlowMonitor for task 1 and 3.

3. You can use `examples/routing/manet-routing-compare.cc` or `src/aodv/examples/aodv-example.cc` as a starter file of Task 1. If you write reproducible code, Task 3 should just be about experimentation.
4. For task 2, copy the entire `src/aodv` directory to a new `src/raodv` (or pick a suitable name) directory. Rename all references to AODV therein. Then incrementally start to tweak AODV to implement your intended protocol. The bunch of tasks should be to modify the `src/aodv/model/aodv-routing-protocol.cc` file.
5. If you implement the R-AODV protocol, a modified file for R-AODV packets is provided in the MOODLE.

Submission Format

2005xyz

```
| -- report.pdf (contains all the plots with the simulation
                    setting mentioned in the caption;
                    no description required)

| -- code/
| ----- your modified files
| ----- a txt file containing the relative paths of the modified files
```

Marks Distribution

Task	Marks
Task 1	30
Task 2	50
Task 3	20

Any novel modification will carry bonus. It is NOT needed that you will come up with improvement in performance with your modified mechanism. Rather it is needed to make sure that whatever changes you would have done come with an intuition.

Deadline: November 25, 2024, 11:55 PM