

Autonomous networks - REPORT DTN1

andrea.zanni8@studio.unibo.it or andrea.zanni@etu.sorbonne-universite.fr

duc_huy.do@etu.sorbonne-univerite.fr

****Question 1:**** looking at the result of the simulation and at the implementation of the WayPointNode, describe the random waypoint mobility model.

In this random waypoint mobility model every node sets a random destination on the map and then moves toward it. When the node reaches the destination, it computes a new destination and moves toward it.

****Question 2:**** Implement the simplest solution that consists in transmitting the data only to the sink node.

The node sends the data when it is near the sink node. Every iteration, a node checks whether or not there is a sink among its neighbors to send it its data.

```
private boolean doWeTransmitToNode(Node node) {
    return node.hasProperty("sink") && (boolean)node.getProperty("sink");
}
```

****Question 3:**** look at the result of the previous solution. Propose and implement a better solution.

In the simulation I ran the sink had gathered all the data in about 12 058, 14 105 time units that averages to about 13 000 time units. A better solution consists in sending the data to the nodes that are already within range, the sender node is the one with the lowest id. This will allow some aggregation of data before the nodes reach the sink. As a result the whole system will converge a lot faster to the state when there are just a few nodes left to deliver the data. 4709

****Question 4:**** How you can use the property `distanceToSink` to improve the data aggregation time.

We could use such property by sending the data to the nodes whose distanceToSink tends to decrease (i.e., probably the node is going toward the sink node).

****Question 5:**** how can you improve again your algorithm.

A better solution consists in asking the neighbour node whether it is going (or it crosses) to the direction of the sink (or anyway in a direction that will enter in the communication range of the sink node) and if so the node passes to it its data.

With the aforementioned implementation the sink node gathers all data in 9262, 7961, 7253, 7999, 3477, 6160, 15122, 7331 time units that is an average of 8070 time units, about 5 000 time units less than the solution that only sends to the sink.

Cons: with this solution, the nodes which are going toward the sink will drain their battery faster since they will make plenty of connections and will receive a great deal of data..

****Question 6:**** use the Stats class to record the evolution of the number of nodes that have transmitted (i.e. the number of aggregated data) over the time.

Stats class is simply implemented with a clock counter which prints how many nodes have transmitted their data in each round.