Wireless and Mobile Computing for Entertainment Applications

Assignment 06:
Mobile Computing Algorithms for Wi-Fi 802.11

Important

- This is assignment 6: Mobile Computing Algorithms for Wi-Fi 802.11
- Submit your results via e-mail: **schmist@inf.ethz.ch**
- Submission deadline: **Wednesday, 2015-Dec-10, 11:59pm**
- As XML scenario file, create one with three stations, for example based on the one from earlier assignments
  - Use two access categories (AC01 and AC02)
  - TCP should be disabled, instead use data or saturation without TCP
  - A simple model for the medium is sufficient: set attribute useInterference="false".

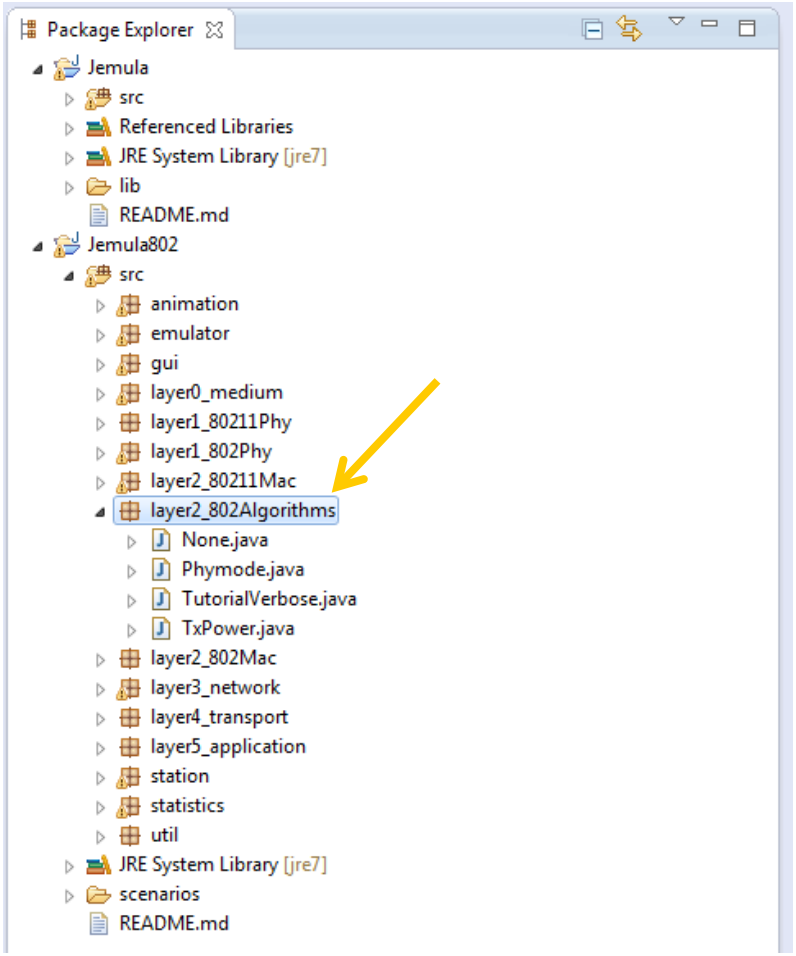
Motivation

- Learn about WLAN IEEE 802.11e QoS support
  - Isolated, single basic service set with perfect radio channel condition
  - Multiple competing stations, each with two streams (AC01 and AC02)
- Develop and test a mobile computing algorithm for modifying the contention window parameters
  - Design and present algorithm
  - Implement algorithm into the Java emulation tool Jemula802
  - Evaluate and illustrate performance with the tool (documentation)
  - Provide confidence information with all results
  - Test your algorithm against other algorithms
- We will organize a final tournament for all algorithms to identify the best performer!

Step 1: Understand IEEE 802.11e & update the tool

- For more detailed information on Wi-Fi, read and understand the 802.11 course chapter
- Read the full paper of assignment 2
- Update tool from repository (GitHub)

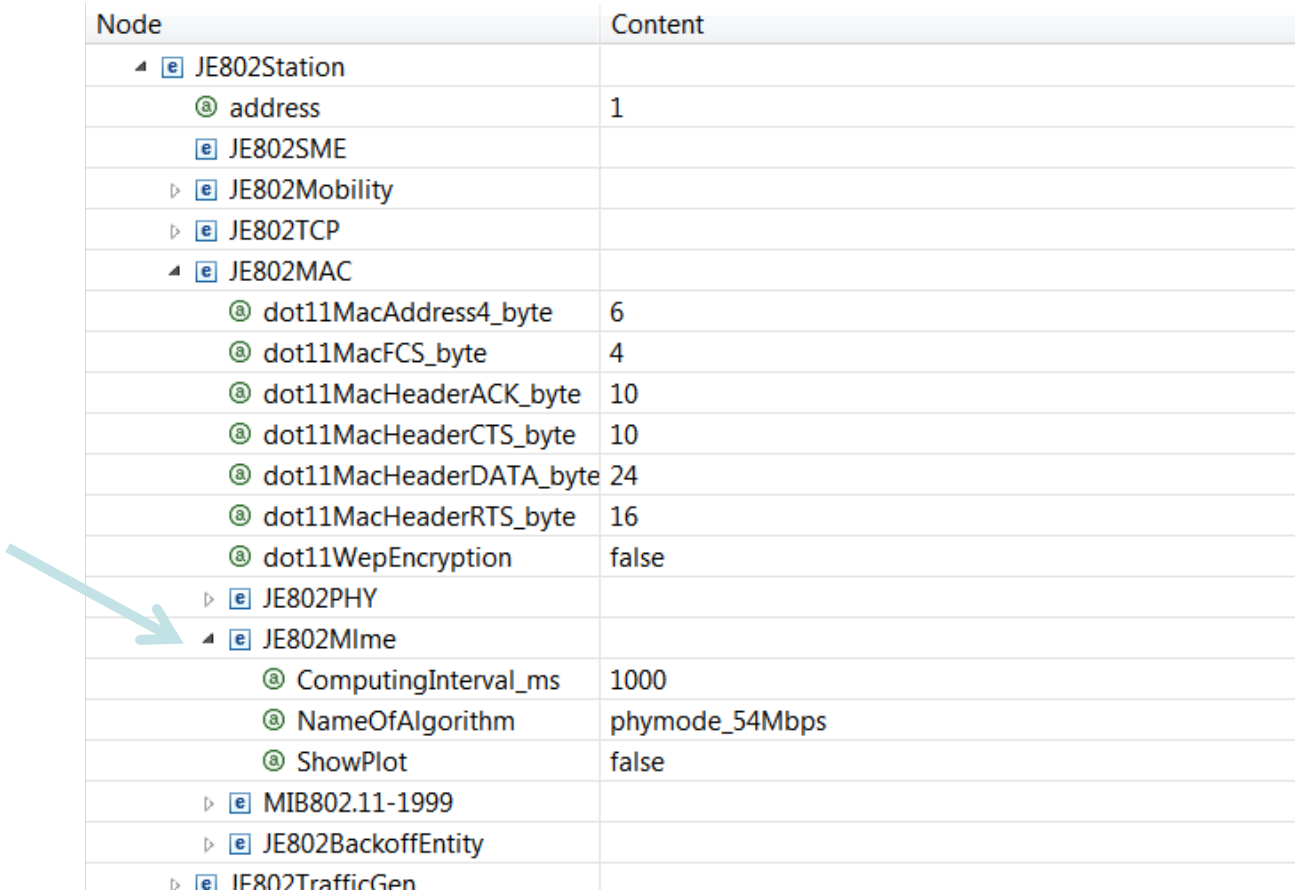
Step 2: Set the algorithm, e.g. “TutorialVerbose“




```java
package layer2_802Algorithms;

import layer2_80211Mac.JE802_11Mac;

public class None extends JE802_11MacAlgorithm {

    public None(String name, JE802_11Mac mac) {
        super(name, mac);
    }

    @Override
    public void compute() {
        // TODO Auto-generated method stub
    }

    @Override
    public void plot() {
        // TODO Auto-generated method stub

    }

}
```

Algorithm name in MLME =
Name of the algorithm Java file without .java

Step 3: Find the XML tag in the scenario file



Node	Content
JE802Station	
address	1
JE802SME	
JE802Mobility	
JE802TCP	
JE802MAC	
dot11MacAddress4_byte	6
dot11MacFCS_byte	4
dot11MacHeaderACK_byte	10
dot11MacHeaderCTS_byte	10
dot11MacHeaderDATA_byte	24
dot11MacHeaderRTS_byte	16
dot11WepEncryption	false
JE802PHY	
JE802MIme	
ComputingInterval_ms	1000
NameOfAlgorithm	phymode_54Mbps
ShowPlot	false
MIB802.11-1999	
JE802BackoffEntity	
JE802TrafficGen	

Step 4: Tutorial algorithm (1/2)

- There should be one receiving station (station 01) and two transmitting stations (station 02 and station 03)
  - Start with this scenario, and scale up to more complex where needed
- Stations 02 and 03 carry two traffic generators each
  - AC01 and AC02 transmitting data to the single receiving station 01
  - As result, station 01 receives four streams of data
- All stations comprise two backoff entities
  - The two backoff entities inside the two transmitting stations will modify their backoff parameters determined by the mobile computing algorithm
- The mobile computing algorithm resides in the MLME
  - MLME = MAC Layer Management Entity
  - Package layer2_802Algorithms for different implementations

Step 4: Tutorial algorithm (2/2)

- Check the tutorial algorithm “TutorialVerbose”
  - <JE802Mlme NameOfAlgorithm=*"TutorialVerbose”* ComputingInterval_ms=*"100" ShowPlot="true"*>
- Receiving station does not send any data
  - Must be using algorithm “none”
- In the initial example, station 02 uses algorithm “none”
  - The algorithm “none” means that that station will not change any parameters
- In the initial example, station 03 should be set to algorithm “TutorialVerbose”
  - Run the simulation, and check the shell logging output (set debuglevel in Jemula.java smaller than the log level you use)
  - Don’t forget to set the ShowPlot=“true”

Step 5: Design your own algorithm

- Station 2 and station 3 should both follow your algorithm
- Give your algorithm a name that can be used in XML scenario
- The objective is to provide maximum QoS to AC1 and (with less priority) AC2 - **in competition to other stations with unknown algorithms**
- Next week (**Monday 8th**), present your design idea to the class
  - 8 min + 2 min questions & answers; use projector, white-board, or PPT
  - What need do you address? What is your approach? What benefit do you expect for whom? Any alternatives / competition?
- The algorithm should perform even against any unknown algorithm
  - What does “perform” mean? what are the metrics, how about fairness?

Step 6: Implement and test algorithm

- Simulate algorithm against itself and against other algorithms
- The algorithm should perform even against other (currently unknown) algorithms
- Create appropriate graphical representation of the simulation result(s), and interpret the results
  - Illustrating benefits, drawbacks, and limitations of the algorithm: How would it perform with more than two transmitting stations?
- Provide confidence information
- We will organize a tournament to evaluate the overall performance

Expected results

- Please submit the following results (step 5 next week, the rest by due date, as a report in PDF format):
  - Step 5: 10 min presentation on algorithm design, submit PPT or PDF after your presentation (during the lecture next week)
  - Step 6: Your algorithm including documentation
  - Step 6: Simulation results to illustrate performance (graphs), interpretation
  - Submit XML scenario file and source code for competition tournament
  - OPTIONAL: feedback on the assignment: difficulty, clarity, time
- Send your report to: **schmist@inf.ethz.ch**

Assignment 06:
Mobile Computing Algorithms for Wi-Fi 802.11
– END OF ASSIGNMENT –