DISI – University of Trento

Master in Computer Science AA 2016/2017

Simulation and Performance Evaluation

Assignment 2

Python Simulator Extension

Renato Lo Cigno, Michele Segata

November 17, 2016

We start from the simple simulator framework that is available through classroom. The assignment is to select one of the following extensions, design it (flow chart, modifications, ...) implement it, run proper simulations and write a report presenting and commenting the simulation results. The modified code must be delivered with the report, properly commented, with all the scripts needed to compile and run it to obtain the results presented in the report, including the scripts to analyze and plot results. The report should be concise and clear, typically no more than 4 pages in the given format.

Not all modifications have the same complexity, some are really trivial. Thus we propose you different extensions, but some of them limit the maximum grade you can achieve: Less work, limited insight in the concepts of DES, limited grade, looks fair. In any case each of you is assigned a network topology (see the `topologies.zip` file in Classroom to find your `<surname>.csv` topology description) that must be used to obtain results, so that even if some of you implement the same extension results will be different.

The following list reports the proposed modifications with the maximum grade available. No max. grade reported means no limit. You do not have to declare in advance what modification you choose, so you can also start with an easy one, then move on to a more complex one which is based on the simpler one. However, you have to state explicitly the modification you choose in the report. If none of these modifications satisfy your curiosity or thirst of knowledge, feel free to propose another one, but do it before starting to implement and discuss it with us, to avoid entering a dead end or too complex affair.

Trivial Carrier Sensing (Max: 24/30). Extend the protocol to include a simple carrier sensing function that prevents transmission if there are packets being transmitted within the communication range of the node. The packet is transmitted when the channel becomes free.

Compare results with the original simulator.

Simple Carrier Sensing (Max: 27/30). As the **Trivial Carrier Sensing**, but a p-persistent version. A station before transmitting must sense the channel. If the channel is free, it proceeds to transmit; if it is occupied, it re-schedules the transmission with persistency p. p is a probability that defines if the transmission is scheduled immediately after the channel becomes free, or if it is re-scheduled after an exponential random time with average $T_r = 10T_t$, where T_t is the time needed to transmit the maximum size packet. If the transmission is scheduled immediately after the channel becomes free, no carrier sensing is performed and the packet is transmitted. If the transmission is instead re-scheduled, carrier sensing is performed again before transmitting: if at the moment of transmission the channel is busy, the procedure is repeated.

Performance evaluation must include a sensitive analysis on p.

Carrier Sensing with Collision Avoidance (Max: 27/30). Extend the **Trivial Carrier Sensing** protocol, but including a collision avoidance procedure with a slotted contention window when the channel becomes free. The slot duration is T_s and the contention window size is W_c.

If the channel is busy, wait until it becomes free, then extract an integer random variable B_k uniformly distributed in $[0, W_c - 1]$ and count-down B_k slots. When $B_k = 0$ transmit; if the channel becomes occupied again, then re-schedule the transmission when the channel becomes free again extracting a new value for B_k. After a transmission the station must behave as if the channel were occupied.

Compare the results for different values of W_c. Set T_s to a value that is "justified" by all the other parameters of your protocol.

Realistic Propagation (Max. 24/30 or + 3 on any modification above). Change the packet reception model from a disk model to a probabilistic model where the probability of reception is inversely proportional to the distance. More formally, the reception probability for a packet so far is modeled as

$$\Pr(\text{correct reception}|d) = \begin{cases} 1 & \text{if } d < \text{RX}_\text{range} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

In this extension, change the model to implement the following law

$$\Pr(\text{correct reception}|d) = \begin{cases} 1 - \frac{d}{\text{RX}_\text{range}} & \text{if } d < \text{RX}_\text{range} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

This models the reception probability when no collision occur. The probability of reception for colliding packets is still zero.

Unicast Traffic. Implement destinations and acknowledgment of packets, i.e., stations must be identifiable and packets are sent from one specific station to another one, which, upon correct reception, sends an acknowledgement back. Packets that are not acknowledged must be re-sent after an exponential random time with average $T_r = 10T_t$, where T_t is the time needed to transmit the maximum size packet. Two timing parameters must be defined: SIFS and DIFS (we take WiFi terminology here). SIFS (Short Inter-Frame Space) is $10\,\mu$s and separates a frame (packet) from its ACK. DIFS (Distributed Inter-Frame Space) is $50\,\mu$s and separates two different frames, or better, the end of a transmission (reception) from the attempt to send a new frame.

Carrier sensing must be implemented with $T_s = \text{DIFS}$ to protect ACKs from collisions. Collided packets are re-scheduled after $T_r = 10T_t$, where T_t is the time needed to transmit the maximum size packet.

802.11. Implement the full (well a bit simplified!!) MAC of WiFi as follows.

You have to start from the **Unicast Traffic** extension and change the channel access mechanism like in the **Carrier Sensing with Collision Avoidance**. The slot duration is $T_s = 20\,\mu$s, and the channel must be sensed free for an entire DIFS before starting to contend for the channel, i.e., before the transmission begins if the channel is free or the Collision Avoidance procedure is run.

When a packet is lost (the ACK is not received), a node should double its contention window for extracting the B_k value up to a maximum value W_max. Let's suppose the contention window is set to the initial value $W_c = W_\text{min}$. When a packet is lost, the node schedules the re-transmission and doubles the contention window size, i.e., $W_c = 2 \times W_c$. The "game" is repeated until the packet finally gets through or a maximum number of attempts N_a is reached. When successfully sending a packet (or when giving up after N_a attempts) the contention window is reset to the minimum before trying to send the next packet (i.e., $W_c = W_\text{min}$ whenever a new packet is sent). In basic, standard WiFi we have $W_\text{min} = 32$, $W_\text{max} = 1024$, $N_a = 7$.

Further, when the channel turns busy while performing the countdown for channel access, the value of the countdown should be "frozen", and the node should continue with the countdown when the channel turns free again, without extracting a new value for B_k.

To write your report use the LaTeX template we give you and do not write more than 4 pages. Deliver the PDF file of the report and all the code and scripts you used developed as a single .zip or .tar file through Classroom; if the simulator does not compile and run or the scripts do not run on a standard Linux box, we simply notify you that it does not work, and we will not attempt correction. Keep your code **CLEAN, ORGANIZED, and COMMENTED**. Do not send us your source code with unused portions commented out, blocks of code with no comments, or with monolithic pieces of code. Split your code in functions. DO NOT use absolute folders like

```
ds <- read.csv('/home/john.doe/Documents/spe/doe.csv')
```

but rather

```
ds <- read.csv('./doe.csv')
```

The deadline to have a correct-and-redo chance for this assignment is February 15, 2016. If you deliver the assignment within this date, we will correct it and give you the chance to refine it before the oral discussion, otherwise we will consider the work "as is" before the oral discussion is agreed upon. However, if the quality of the delivery is unacceptable (e.g., no methodology is described, plots are meaningless and not explained, etc.) we will not correct it, but simply reject it, so you lose the privilege of a pre-correction.

If you have some doubts, just write us an email or ask in class.

Have Fun!