Object-Oriented Programming and Design - Monsoon 2024

Project – Modeling and Simulation of WiFi Communication

Arani Bhattacharya

The goal of this project is to design a simple WiFi simulator. The use of object-oriented features such as inheritance, data abstraction, data hiding and polymorphism is essential.

1. **WiFi 4 Communication** WiFi communication happens via a single access point and a set of users. A single transmission consists of sniffing of the channel, and transmission of the data packet if the channel is free. If the channel is not free, the transmission is deferred by a random unit of time (subject to a maximum limit of backoff time), followed by repetition of the above process. Simulate this above process of transmission to find out the throughput, average and maximum latency if a total of (i) 1 user and 1 AP is communicating, (ii) 10 users and 1 AP is communicating, and (iii) 100 users and 1 AP are communicating. You may assume that a total of 20 MHz being used as bandwidth, modulation of 256-QAM and coding rate of 5/6 is used for transmission and that each packet has size of 1 KB. You may ignore the other periods of time wastage such as DIFS, CIFS and guard intervals.

2. **WiFi 5 Communication** WiFi 5 allowed a new type of communication, using multi-user MIMO. In this case, parallel transmission was allowed. However, parallel communication could be done only after (i) a single broadcast packet was sent by the access point, and (ii) each user would sequentially send their channel state information, which would be a single packet of 200 bytes. Once this was done, each user could communicate in parallel for a total of 15 ms. After 15 ms, the above process is repeated. Find out the throughput, average and maximum latency in each of the above cases once again, assuming round-robin scheduling of the users.

3. **WiFi 6 Communication** WiFi 6 allowed a yet new type of communication, using OFDMA. In this case, a second type of parallel transmission was allowed, where the 20 MHz channel could be further subdivided into units of 2 MHz, 4 MHz or 10 MHz, and each of these sub-channels could be utilized in parallel for a total of 5ms. After 5ms, the channel allocation is done once again. Again, find out the throughput, average and maximum latency in each of the above cases, using a process of round-robin scheduling of the users.

Grade Components

0

1. Design of class structure, modeling of access point, user, packet, frequency channel is a must – 20 (4 x 5)
2. Proper functioning of each class – 40 (4 x 5)
3. Use of appropriate data structure – 30 (3 x 10)
4. Modeling of CSMA/CA, MU-MIMO and OFDMA – 30 (3 x 10)
5. Computation of throughput, average and maximum latency in each case – 45 (3 x 15)
6. Additional cases of code cleanliness, such as use of Makefile (both for building library and test program), git commits and proper function/variable names – 15
7. Extra functionality based on the student's understanding – 20

What and How To Submit

- The C++ program sources. Templates and exception handling must be utilized.
- `Makefile` to compile the sources and generate the running binary for the shell. The Makefile should generate two versions of the binary – one for debugging and another for optimized execution.
- A readme text file, explaining the commands needed to build the file, and the format of the input files. If code is copied from anywhere else (not that copying from any other student is plagiarism, but using textbook or open-source code is allowed), that should be mentioned here.
- At least 4 significant commits on a **private** github repository, with proper descriptions of the commits. You may have as many commits as you wish.
- Make the assigned TA the admin of the github repository, **and** submit the same code in zipped form on Google Classroom by the due date.

Late Submission Policy

- Late submissions would only be allowed under exceptional circumstances. Please ensure that you submit the project on time.