

Computer Engineering 4DK4

Lab 4

ALOHA Medium Access Control and Packet Reservation

This lab investigates the **performance of the ALOHA** media access control protocol. In the provided ALOHA simulation code, the **random backoff** used by the stations is **exponentially distributed with a fixed mean**. Performance results are obtained using this simulation for various combinations of numbers of stations and mean backoff. Later in the lab, this is changed so that the mean backoff is chosen dynamically for a given packet using a binary exponential backoff algorithm. The code is then modified so that one station functions differently than the others, which results in preferential treatment. A Slotted-ALOHA version of the protocol is then designed and its performance is characterized.

1 Preparation

1. An electronic copy of the simulation program must first be obtained. The program consists of a number of C code and header files. You must also include the `simlib.h`, `simlib.c` and `trace.h` Simlib library files when you compile and link the simulation. A zip file of everything you need is available on the course web site.

2 Experiments

1. As in the other labs first familiarize yourself with the code and with running the simulation. Make sure that you understand how this simulation works. Set **MEAN_PACKET_DURATION to 1**. *In all the experiments make sure you include runs using your McMaster student ID number as the random number generator seed.*
2. Set the `NUMBER_OF_STATIONS` to 10, and **generate mean delay versus arrival rate curves for** `MEAN_BACKOFF_DURATION` values of 5, 10, and 20. Repeat these experiments when the `NUMBER_OF_STATIONS` is set to 5 using `MEAN_BACKOFF_DURATION` values of 3, 5 and 7.
3. Modify the simulation so that the stations use a **binary exponential backoff**, such as that used in Ethernet. The **backoff is chosen uniformly in the range $[0, 2^{N_c})$** packet transmission times, where **N_c is the number of collisions that the packet has suffered**. Note that in the simulation code there is a packet struct member that already keeps track of the number of collisions that a packet has had. Generate a **mean delay versus throughput curve** for the **10 and 5 station** cases used in Part 2. Compare the results that you obtained using fixed mean backoffs.

4. Modify the simulation from Part 3 so that **one particular station always re-transmits an unsuccessful packet in the very next slot (i.e., it persists and does not backoff at all)**. Compare the mean delay performance of this station versus the mean delay performance of the other stations.
5. Change the simulation so that it operates using Slotted-ALOHA. The best way to do this is to implement a guard time where transmissions start and end ϵ seconds after and before the actual time slot boundaries as shown in Figure 1. Compare the results you get with S-ALOHA to those generated in Part 3. In your results use values of ϵ that are small enough that they have only a negligible affect on the overhead.

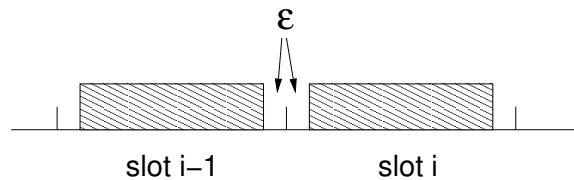


Figure 1: Slotted ALOHA with Guard Times

Explain the results that you obtain in all the experiments.

3 Writeup

Submit a writeup for the lab. Each group (of 3 maximum) is responsible for their own experiments and writeup. Include in your writeup a description of everything that you did including all data (and random number generator seeds) that were used to obtain the graphs. Include the plots and a listing of the modified program with your writeup.