ECE 478/578: Fundamentals of Computer Networks

Project # 1

Due Tuesday March 6
Electronically via D2L at 11:59 pm

February 20, 2012

1 Preliminaries

- You must form a group of (strictly) two people. If you do not have a partner please contact the instructor.
- Due date: Tuesday March 6th, 2012 11:59 pm via D2L.
- You must submit your code and a project report. Develop your code in C, but you can plot graphs in Matlab or Excel. If your code requires some special instructions to run, please include a readme file with your code.

2 Project Description

You are to study the performance of various multiple access protocols in a wireless setting, by developing a *discrete event simulator*. Consider the network shown in Figure 1. The circles denote the communication range R of each node. We are interested in the following two scenarios:

A. Concurrent Communications: Nodes A, B, C, and D of Figure 1(a) are within the same collision domain (any transmission is received by all). Communication takes place between pairs $A \rightarrow B$ and $C \rightarrow D$. Traffic is generated at A and C according to a Poisson distribution with parameters λ_A and λ_C, respectively.

B. Hidden Terminals: Nodes A, B, C, and D of Figure 1(b), belong to separate collision domains. Communication takes place between pairs $A \rightarrow B$ and $C \rightarrow D$. Traffic is generated at A and C according to a Poisson distribution with parameters λ_A and λ_C, respectively.

For each scenario, compute relevant performance metrics for the following multiple access protocols. A time-slotted system is assumed.

1. *CSMA with Collision Avoidance (CSMA/CA):* In this protocol, a node ready to transmit (when a packet has arrived for transmission from the upper layers of the network stack), senses the channel for an initial period of DIFS time.
   - If the channel is idle, it starts to transmit.
   - If the channel is busy, it waits until the medium becomes idle. Once idle, it selects a random backoff value in $[1, CW_0]$. It then waits for DIFS time and starts the countdown of its timer to zero, pausing any time a frame transmission occurs. When the counter reaches zero, it attempts to transmit.
   - If the packet is successfully received (no collision), the node repeats the transmission process by selecting a random backoff value in $[1, CW_0]$. For successive transmissions, the node has to sense for DIFS time before starting the countdown.
   - If a collision occurs, it doubles its contention window CW and repeats the backoff process by selecting a backoff value in $[1, CW]$. The CW value cannot become larger than a threshold value $CW_{\max}$.

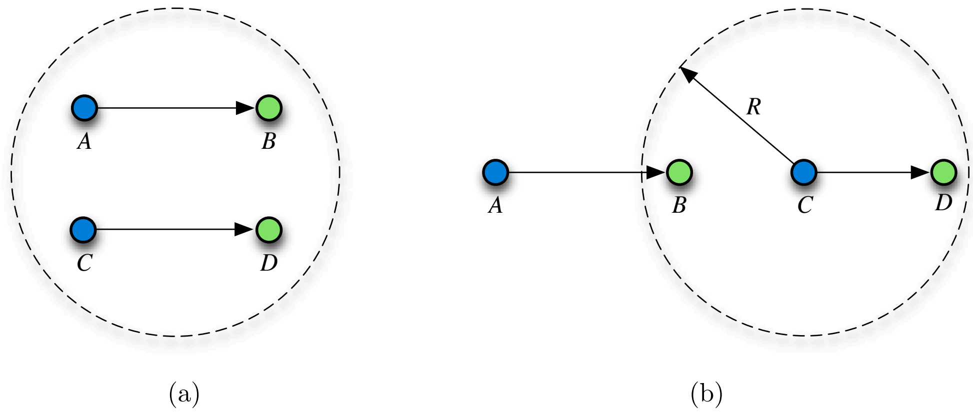

Figure 1: (a) Topology for parallel transmissions within the same collision domain, (b) topology for parallel transmissions when A and C are hidden terminals.

2. *CSMA/CA with virtual carrier sensing enabled:* RTS and CTS frames are exchanged before the transmission of a frame. If RTS transmissions collide, nodes interpret this as a collision and invoke the exponential backoff mechanism. Otherwise, nodes that overhear an RTS/CTS message defer from transmission for the time indicated in the NAV vector.

3 Simulation parameters

Parameter	Value	Parameter	Value
Data packet size	1,500 bytes	ACK,RTS, CTS size	30 bytes
Slot duration	20 μs	DIFS duration	40 μs
SIFS duration	10 μs	Transmission rate	6 Mbps
CW_0	8 slots	$CW_{\max}$	1024 slots
λ	$\{50, 100, 200, 300, 400, 500\}$ pkts/sec	Simulation time	10 sec

4 Performance Metrics

For each of the simulated scenarios you must compare the protocol performance with respect to the following metrics:

1. Throughput: The aggregate and individual throughput as a function of λ.

2. Collisions: The number of collisions (data and RTS/CTS) as a function of λ.

3. Link Utilization $U(\%)$: The percent fraction of time that the medium is used for the transmission of useful data as a function of λ.

4. Fairness Index: The fraction of time that the channel is occupied by pair $A \rightarrow B$ over the fraction of time that the channel is occupied by pair $C \rightarrow D$ as a function of λ.

Your experiments must be repeated when for two different scenarios: (a) $\lambda_A = \lambda_C = \lambda$ and (b) $\lambda_A = 2\lambda, \lambda_C = \lambda$.

Assume no losses due to the imperfections of the wireless medium. All results must be plotted in comparative graphs.

5 Project Report

Include with your report:

1. A brief introduction describing the project. In this section include the responsibilities of each team member with references to which parts/steps he/she completed.
2. Graphs for each of the simulated scenarios.
3. Justification for the results shown in your graphs.
4. Your code (you will not be graded on your code efficiency).

Appendix

Generating Poisson-distributed traffic: To generate Poisson-distributed traffic, it is sufficient to generate a series of exponentially-distributed inter-arrival times. Such times can be generated using the inverse CDF transformation method.

Step 1: Generate a series of uniformly distributed numbers $U = \{u_1, u_2, \ldots, u_n\}$ with $u_i \in (0, 1), \forall i$.
Step 2: Compute a series of numbers $X = \{x_1, x_2, \ldots, x_n\}$, exponentially distributed with parameter λ, as

$$X = -\frac{1}{\lambda} \ln(1 - U) \tag{1}$$

Using X, you can determine the time of each frame arrival at each node. For instance, frame 1 arrives at time x_1, frame 2 arrives at time $x_1 + x_2$, etc. The inter-arrival time generation process has to be repeated for each transmitting node.