

SIMULATION AND COMPARISON OF PURE AND SLOTTED ALOHA

1. Introduction:

In this report we will present the comparison of Pure ALOHA and Slotted ALOHA protocols, through simulation results under different traffic loads. We will then compare the simulation results with theoretical results and analyze why they have different system performances.

Classical Aloha

The pure ALOHA protocol is a random access protocol used for data transfer. In Pure or Classical Aloha each source node (transmitter) in a network sends out a data frame whenever there is a frame to send. Aloha neither co-ordinates medium access nor does it resolve contention on the MAC layer. Instead, each station can access the medium at any time. After a transmission, the user waits for an acknowledgment on either the same channel or a separate feedback channel. In case of collisions, (i.e., when a NACK is received), the terminal waits for a random period of time and retransmits the message. As the number of users increase, a greater delay occurs because the probability of collision increases.

This is a random access scheme, without a central arbiter controlling access and without co-ordination among the nodes. So, there is no operator intervention, and any number of computers could be involved. If the frame successfully reaches the destination node (receiver), the next frame is sent. But if two systems or nodes on the network send at the same time, both signals will be ruined, as a result of "collision" of data frames. Resolving this problem is left to higher layers (e.g., retransmission of data). Hence, if the frame fails to be received at the destination, it is sent again.

In order to determine the throughput, it is important to determine the vulnerable period V_p , which is defined as the time interval during which the packets are susceptible to collisions with transmissions from other users. The Packet A will suffer a collision if other terminals transmit packets during the period t_1 to $t_1 + 2\tau$. Even if only a small portion of packet A sustains a collision, the interference may render the message useless. The rate of arrival which maximizes the throughput for ALOHA is found when the offered load is equal to $G_{\max} = \frac{1}{2}$ which results in maximum throughput $S_{\max} = \frac{1}{2} e^{-1} = 0.1839$. Thus the best traffic utilization one can hope for using ALOHA is 0.184 Erlangs.

Slotted ALOHA

This is a refinement of Classical Aloha scheme by the introduction of time slots. All senders have to be synchronized; transmission can only start at the beginning of a time slot. For this, a centralized clock sends out small clock tick packets to the outlying stations, thus resulting in a discrete distribution of packets. This prevents partial collisions, where one packet collides with a portion of another. Outlying stations were only allowed to send their packets immediately after receiving a clock tick.

Access is still not coordinated. If there is only one station with a packet to send, this guarantees that there will never be a collision for that packet. On the other hand if there are two stations with packets to send, this algorithm guarantees that there will be a collision, and the whole of the slot period up to the next clock tick is wasted. As the number of users increase, a greater delay will occur due to complete collisions and the resulting repeated transmissions of those packets originally lost. The number of slots which a transmitter waits prior to retransmitting also determines the delay characteristics of the traffic. The vulnerable period for slotted ALOHA is only one packet duration, since partial collisions are prevented through synchronization.

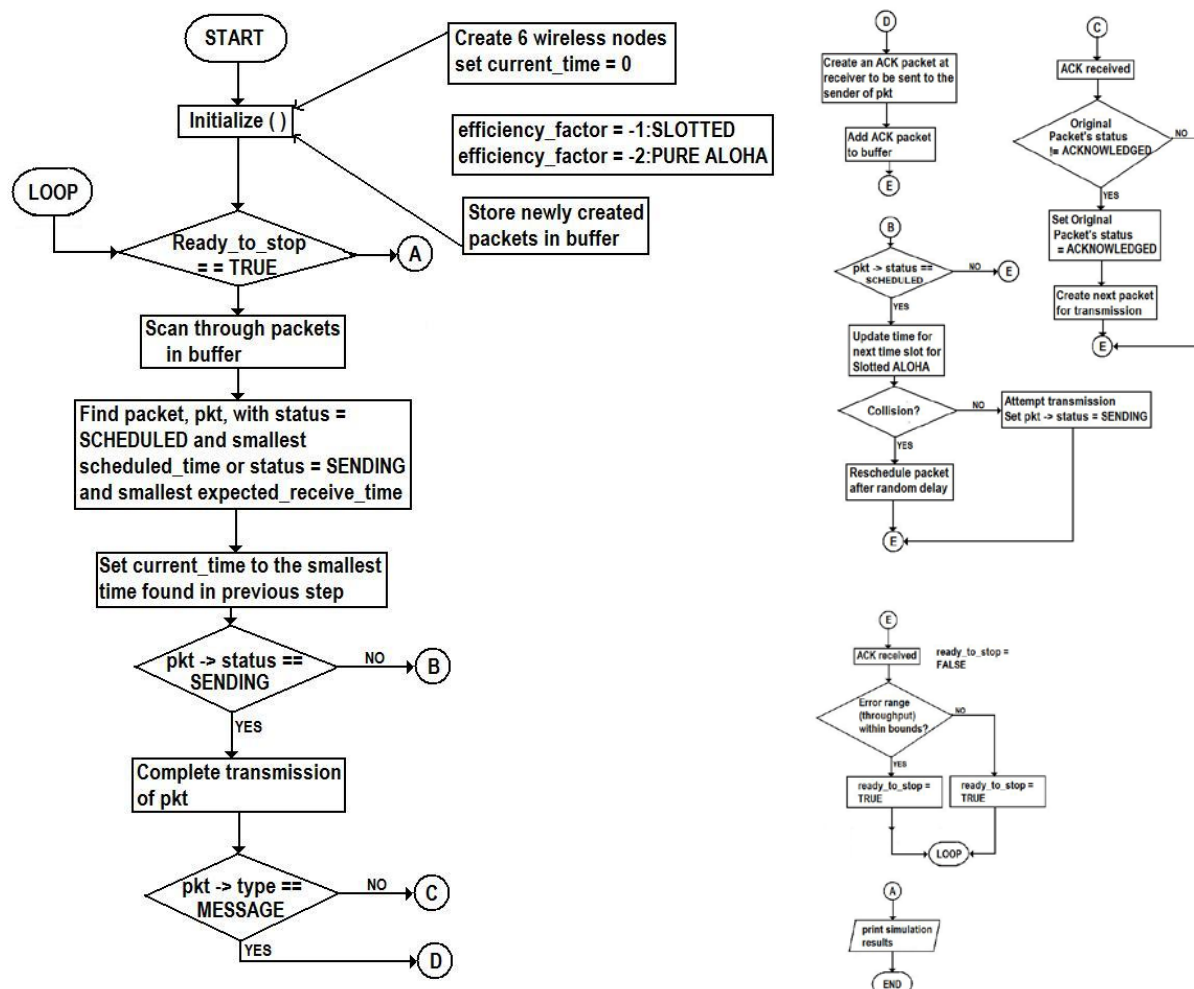
The rate of arrival which maximizes the throughput for Slotted ALOHA is found when $G_{\max} = 1$ and maximum throughput is then that of $S_{\max} = e^{-1} = 0.3679$. Hence, slotted ALOHA provides a maximum channel utilization of 0.368 Erlangs, double that of ALOHA.

2. Simulation:

Description:

Our simulation program has a well defined OO structure and we used C++ as the programming language. We consider, for the purpose of our simulation, six wireless nodes say, nodes 1,2,3,4,5,6 such that the communication (of packets) takes place between nodes 1 & 2, 3 & 4, and, 5 & 6. All frames are assumed to be of equal lengths. In our simulation, the arrival of packets at the wireless nodes follows a Poisson distribution and hence the packet inter-arrival times follow the Exponential distribution. The random number and variate generator is instanced for each simulated node to reduce correlation. Uncorrelated seeds are used by the generator for this purpose. There are sender and receiver buffers which store the packets that are ready for transmission or have been received respectively.

Flow Chart:



As shown in the above flowchat, the simulation begins at current_time=0. The sender buffer is scanned for that packet, pkt, which can either complete its transmission at the earliest or is scheduled for transmission at the earliest. The current_time is updated to this earliest time when another packet can be transmitted such that it does not collide with a packet that is already being transmitted. Now, if pkt is

already being transmitted then it is allowed to complete its transmission without a collision since the current_time has been updated suitably. If pkt is a message/data packet then an ACK packet is generated from the receiver of pkt. Otherwise, if pkt is an ACK packet itself then the sending node is notified of the received ACK and continues with the transmission of its other packets. If the packet pkt chosen from the sender buffer is scheduled for transmission at current_time (In slotted ALOHA, it is scheduled to the beginning of the next time slot), then a transmission attempt is made. A collision check is performed to check pkt collides with some other packet already being transmitted or which begins transmission while pkt is being transmitted (In slotted ALOHA, another packet scheduled in the same slot as pkt). If a collision is detected, pkt is scheduled for retransmission after a random delay. Otherwise, it is allowed to complete its transmission as described above.

Stopping Criteria:

As the stopping criteria, we use the known theoretical throughput for a given offered load and compare that with the calculated simulation throughput. When the absolute value of the difference of the throughput calculated for the above simulation and the theoretical throughput for the ALOHA protocols (the error) lies within a predefined acceptable value (0.03 for pure ALOHA and 0.12 for slotted ALOHA), then the simulation is stopped.

The Random Number and Variate Generator:

For the random number generator, a multiplicative LCG based on an equivalent expression of Lehmer's original proposal¹ was used. It takes the following form: $x_n = (8 * 8192 + 3) x_{n-1} \bmod (2^{31})$. For efficiency reasons, the modulus m takes the form 2^k . To guarantee a full period, the multiplier a is in the form $8i \pm 3$. The period is then equal to $2^{31-2} = 2^{29} = 536870912$, as long as the initial seed is odd. Seed selection was done in such a way as to guarantee non-overlapping streams.

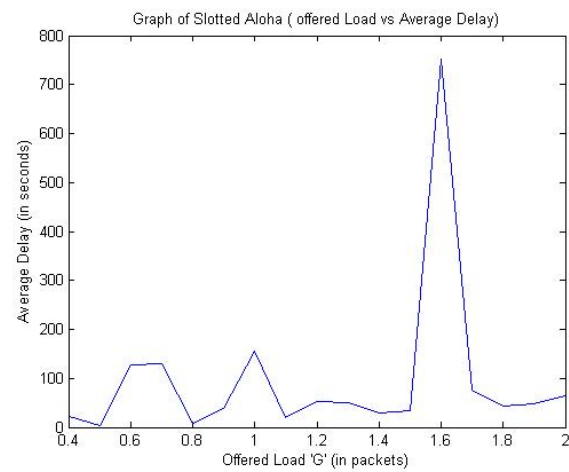
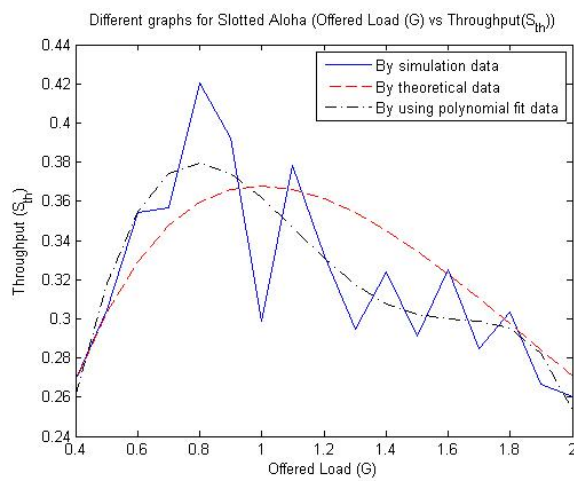
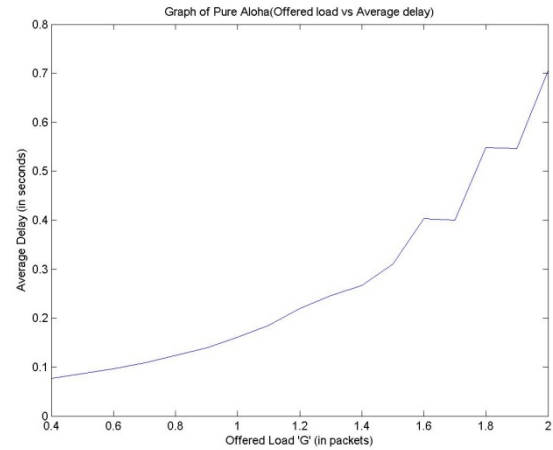
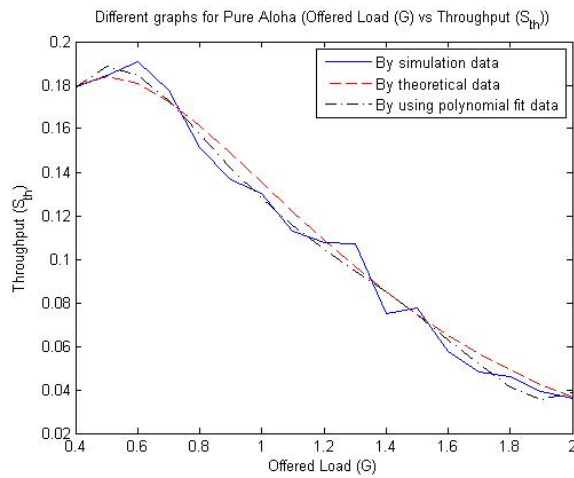
For the exponential random variate generator, we relied on the inverse transformation. Since the *pdf* is $f(x) = \lambda e^{-\lambda x}$ we can generate exponential random values from an uniformly distributed random number u through the CDF $F(x) = 1 - e^{-\lambda x}$ and discover that $x = \ln \frac{(1-u)}{-\lambda}$.

Simulation Results:

We will now present the simulation results. In the following graphs we show the theoretical and simulated throughputs of both Pure and Slotted ALOHA. As expected, there occur variations from the simulations when compared to theoretical values, however they are small. Furthermore we utilized a curve fitting for the data using a fifth order polynomial to get a slightly smoother graph. However, original data is still maintained. Finally, we also plot the graphs of the delays.

In our textbook, neither of the ALOHA schemes states anything about the delays observed in the transmission of frames, so no theoretical comparison has been done in this report. However, it is interesting to see how each protocol behaves. An these results can be seen as the additional work that our group undertook to analyze the average delay observed in transmission of frames for pure as well as slotted ALOHA.

Graphs:



References:

- [1] Jain, R., [The Art of Computer Systems Performance Analysis: techniques for experimental design, measurement, simulation, and modeling], Wiley (1991).
- [2] Agrawal, Dharma Prakash; Zeng , Qing-An, [Introduction to Wireless & Mobile Systems], Thomson (2006), 2nd Ed.