Finding a better variant of the CSMA/CD

Soham Sarkar

Computer Science, VIT- Chennai Campus

DOI: 10.29322/IJSRP.9.11.2019.p9590
http://dx.doi.org/10.29322/IJSRP.9.11.2019.p9590

Abstract- This paper is focused on finding a more efficient version of the Carrier Sense Multiple Access with Collision Detection Networking protocol. It analyses where CSMA/CD falls short and provides two possible solutions to better it. These potential solutions are then compared with the Carrier Sense Multiple Access with Collision Avoidance protocol.

I. Introduction to CSMA

Carrier Sense Multiple Access is a media access control method widely used in Ethernet technology/LANs. It falls under the Random-Access Protocols where no station is considered superior to the others. Now, CSMA was designed to minimize the collision between data sent by various stations using the same medium. CSMA has many variants using the persistence methods. In CSMA, the station senses the channel before accessing the medium. However, the possibility of collision still exists because of propagation delay.

II. CSMA/CD

In CSMA/CD, a collision detection mechanism comes into play. Suppose a station has some data to send. It uses one of the persistence methods (1-persistent, p-persistent or non-persistent) to detect if the channel is idle. Sending station has to keep on checking if the transmission link is idle. For this, it continuously detects transmissions from other nodes. Sending station sends dummy data on the link. If it does not receive any collision signal, this means the link is idle at the moment. If it senses that the carrier is free and there are no collisions, it sends the data. Otherwise it refrains from sending data. If no collision occurs with data from another station then it is a success. But, if a collision is detected, a jamming signal is sent and transmission is aborted. It then waits a random amount of time and then tries again, with the assumption that no other station has started transmitting in the meantime.

On a network that uses CSMA/CD method, every node/station has equal access to the network media.

III. Advantages and disadvantages of CSMA/CD

The advantage of CSMA/CD is that it has relatively low overhead, meaning that not much is involved in the workings of the system. The disadvantage is that as more systems are added to the network, there are more collisions, and the network as a whole becomes slower. The performance of a network that uses CSMA/CD method degrades exponentially as more systems/stations are added to it. Its low overhead means that CSMA/CD systems theoretically can achieve greater speeds than high-overhead systems, such as token passing. However, because collisions take place, the chances of all that speed translating into usable bandwidth are relatively low.

Now, as we can see, even though CSMA/CD is an efficient process, there are a few problems associated with it. Firstly, collision degrades the network performance and secondly priorities cannot be assigned to certain nodes. Also, performance degrades exponentially as more devices are added. The basis of these problems lies in the "random" wait time when collision is detected. As more devices are added, the chance of two or more devices ending up taking the same wait time increases. Also, since this is a random-access protocol, so assigning priorities becomes an issue.

IV. Variant of CSMA/CD

Firstly, we need to assign a registration (or token) number to each station. This will help resolve conflict when two stations end up with same "random" time. Now, there are two methods that we can use to benefit from the situation:

A. Putting p-persistence on collision detection

In this method, when a station detects a possibility of collision, it waits for a random amount of time then has 'p' probability to send the data, and 1-p probability to wait another random amount of time before sending.

This method will further decrease the chance of two stations ending up with same moment of data sending. But it still suffers from the drawback of increase in chance of collision upon increasing the number of stations.

B. Using a queue

In this method, whenever two stations have a collision chance, they will be given two options. Either they can continue using the random wait system (the normal CSMA/CD approach) otherwise they can enter the waiting queue.

Now, there will be certain "queue clearing times" in the channel timeframe where no station will be allowed to transmit and the queued stations will be allowed to send data. Queue will follow FIFO (First In First Out) policy and if two stations enter the queue at the same moment, the station with lower token number will be considered first. The "queue clearing time" will depend on number of elements(stations) in the queue, propagation delay, propagation time, etc. If there are too many elements in the queue, the stations may again be prompted to use the random wait time.

The advantage of this method is that it clears a lot of collision chance. But, then again, if the "queue clearing time" is not generated in a proper manner, it can cause a lot of difficulty for the remaining stations as they would have to wait for the channel to be idle.

V. Comparison with CSMA/CA

CSMA/CA was developed as a better variant of CSMA/CD, to be used in wireless networks specially. It uses interframe space (IFS), contention window and acknowledgements to give a better collision avoidance probability as compared to CSMA/CD. Now, IFS can be used to determine the priority of a station which is similar to my idea of assigning token numbers. The contention window divides the timeframe into slots which makes the system more efficient with respect to time consumption as both of my techniques do not consider how much data is being sent by the station. Lastly, acknowledgements also make CSMA/CA superior.

VI. Conclusion

My model shows some of the weaknesses of the CSMA/CD method and how to overcome them. Through further research, this model can be tested to get an accurate analysis of its strengths and weaknesses.

Acknowledgment

I would like to thank my Network and Communication teacher Dr Reshmi TR for guiding me. I would also like to thank my parents and friends for their continual support.

References

[1] http://www.pearsonitcertification.com/articles/article.aspx?p=1358404&seqNum=2

[2] https://www.geeksforgeeks.org/collision-detection-csmacd/

Authors

First Author – Soham Sarkar, Second year B.Tech(Computer Science) Student, VIT – Chennai Campus, email address- sohamsarkar49@gmail.com.