INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

Implementation of Token Based Distributed DCF for 802.11 MAC

Manas Manjunath Shastri
4th Sem, M.Tech (DCE) Acharya Institute of Technology, Bangalore, India
Nataraju A. B.
Assistant Professor, ECE Department, Acharya Institute of Technology, Bangalore, India

Abstract:
A distributed and dynamically adaptive MAC protocol for wireless networks, called Token-DCF. Main focus of this approach is on reducing idle and collision times by introducing an implicit token passing algorithm. In Token-DCF, a transmitting station schedules one of its neighboring stations for the next transmission epoch using a distributed opportunistic algorithm. Furthermore, packet overhearing is employed to exchange scheduling information across the network. IEEE 802.11 DCF is the MAC protocol currently used in wireless LANs. However, due to idle and collision times, 802.11 DCF performs poorly when it comes to channel utilization, system throughput, and channel access time.

1. Introduction

IEEE 802.11 defines the distributed coordination function (DCF) to share the wireless medium among multiple stations. DCF employs CSMA/CA with a binary exponential backoff algorithm to resolve channel contention. DCF specifies random backoff, which forces a station to defer its access to the channel for a random period of time. This backoff period corresponds to the number of idle slots a station has to wait before its transmission attempt. If multiple stations choose the same backoff, they will attempt to transmit at the same time and collisions will occur. Two types of overhead are associated with random access protocols. One is channel idle time (i.e., backoff time) which is the time when contending stations are waiting to transmit. Another is collision which happens when multiple stations transmit simultaneously. If there are few contending stations, idle time is the dominant overhead. If there are many contending stations, collision probability increases and becomes the main source of low channel utilization.

The purpose of designing a distributed MAC protocol, called Token-DCF 1 is to reduce idle time; collision time and network throughput is improved significantly. In Token-DCF, when a station transmits on the channel, it might give a privilege (i.e., a token) to one of its neighbors. When a transmission ends, the privileged station starts transmitting after a short period of time Non-privileged stations follow the backoff procedure of 802.11 to access the channel. In this way, the privileged station does not go through the contention resolution phase and grabs the channel immediately. A distributed scheduling algorithm is used for choosing the privileged stations.

- Distributed MAC protocols to improve the efficiency of 802.11 DCF
- Token passing MAC protocols
- Scheduling algorithms of wireless networks

Token-DCF 1 is fully distributed and does not require any centralized point of coordination. In Token-DCF, a station might schedule one of its neighbors for transmission on the channel. In this way, each network station acts as a scheduler. Token-DCF uses an opportunistic approach based on packet overhearing for exchanging scheduling information as well as token passing. In Token-DCF, queue length of a station is included in the MAC header of the transmitted packets and is overheard by the neighboring stations. Each station keeps track of queue length of its neighbors. Queue length information is used in the scheduling component of the protocol, where a neighbor of the transmitting station is selected as the privileged station. No extra control packet is needed for giving a privilege to a station. Instead, the next privileged station (i.e., the scheduled station) is specified in the MAC header of data packets being transmitted on the channel. The probability of giving a privilege is always less than 1 to cope with newly arrived traffic as well as imperfections in traffic estimation. This probability is adjusted based on the accuracy of the neighbors' traffic estimation. Token-DCF is an opportunistic MAC protocol which behaves similar to 802.11 DCF when packets are not overheard by the neighboring stations. However, when the opportunistic overhearing is feasible, we eliminate the backoff procedure of 802.11 DCF to improve efficiency.

2. Literature Review

From the previous section it is clear that token DCF helps to increase the efficiency of the network by reducing idle collision time and network throughput is improved significantly. Following are some of the already completed work in this area which forms the basis for current proposed work.

Xue Yang observed that idle overhead and collision overhead are the two overhead which are associated with distributed MAC protocols. Either idle or collision overhead being large, contention resolution will decrease. Hence the authors proposed to apply “pipelining” techniques to design better MAC protocol. Traditional 802.11 DCF protocol uses a binary exponential backoff (BEB) algorithm, which corresponds to idle slots and collision, while this paper uses pipelined scheduling, an implicitly pipelined dual-stage contention resolution (DSCR) MAC protocol. The key mechanism used in DSCR is that the total task is divided into subtasks, and introduces parallelism by allowing subtasks of different tasks to proceed simultaneously, where the task is to schedule the channel access without overlapping to avoid collisions. Hence improves channel utilization and reduces idle time and collision. It is found that without dependence on extensive feedback, without relying on elimination of burst or other signaling, DSCR provides an improved performance for 802.11 DCF and hence the pipelining technique can be useful in improving the performance of MAC protocol.

Zheng zeng observed that to increase the efficiency of 802.11 DCF a distributed random media access protocol called Coordinated Heavy-traffic efficient Access scheme (CHAIN) is introduced. CHAIN mainly uses *overhearing to neighboring nodes* and a *novel piggyback transmission opportunity*. In CHAIN, clients maintain a precedence relation among each neighbor node and client can immediately transmit a new packet after it overhears a successful transmission of its predecessor, without going through the regular contending process. When the network load is low, CHAIN behaves similar to DCF but when the network becomes congested, clients automatically start chains of transmissions to improve efficiency. CHAIN is derived from DCF and co-exists friendly with it. The author has analytically proved the correctness and fairness of CHAIN by extensive simulations on J-SIM and verified the analytical results, and demonstrated the significant performance gain of CHAIN over DCF.

Frederico Calì observed that medium access control (MAC) protocol is the main element that determines the efficiency in sharing the limited communication bandwidth of the wireless channel. To increase the efficiency the author analytically derived average size of the contention window that maximizes the throughput and showed that depending on the network configuration, the standard can operate very fast from theoretical throughput limit and an opportunity tuning back of algorithm can derive IEEE 802.11 protocol close to theoretical throughput limit. Hence a distributed algorithm enables each station tune its back of algorithm at runtime. The author investigated criteria to improve the protocol capacity of IEEE 802.11 network by adopting the *p*-persistence back of algorithm.

Martin Heusse considered wireless LANs such as IEEE 802.11 operating in the unlicensed radio spectrum. The author observed that most proposals for tuning the access method focus on a single aspect and disregard others. The main objective of the author is to define an access method optimized for throughput and fairness, able to dynamically adapt to physical channel conditions, to operate near optimum for a wide range of error rates, and to provide equal time shares when hosts use different bit rates. The author proposed a novel access method derived from 802.11 DCF (*Distributed Coordination Function*) in which all hosts use similar values of the contention window (*CW)* to benefit from good short-term access fairness. This method is called *Idle Sense*, because each host observes the mean number of idle slots between transmission attempts to dynamically control its contention window. *Idle Sense* enables each host to estimate its frame error rate, which can be used for switching to the right bit rate. We present simulations showing how the method leads to high throughput, low collision overhead, and low delay. The method also features fast reactivity and time-fair channel allocation.

Ghazale Hosseinabadi IEEE 802.11 DCF is the MAC protocol currently used in wireless LANs. 802.11 DCF is inefficient due to two types of overhead they are channel idle time and collision time. The author presents the design and performance evaluation of an efficient MAC protocol for wireless networks, called Token-DCF. Token-DCF decreases both idle time and collision time. In Token-DCF, each station keeps track of neighboring links, queue length by overhearing of transmitted packets on the wireless medium. Simulation results show that Token-DCF can significantly improved channel utilization, system throughput and channel access delay over 802.11 DCF.

3. Outcome of Literature Review

From the literature review, it is very clear that in 802.11 DCF the main drawbacks are *idle overhead, collision overhead,* efficiency in sharing the limited communication bandwidth system throughput and channel access delay. Some of these parameters must be considered together to make the system well efficient and significantly improved channel utilization with elimination of idle time and collision. The idle time and collision can be partially addressed using the Token DCF opportunistic MAC protocol for Wireless Networks.

4. Token-DCF Design

In this section, we first provide a high-level overview of Token-DCF and then detail the scheduler signaling and scheduling algorithm.

4.1. Overview

At a high level, the operation of Token-DCF is described as follows. Token-DCF runs a distributed scheduling protocol, where a privilege might be assigned by a transmitting station to one of its neighbors. In each transmission, the transmitting station might select one of its neighbors to have a higher priority for the next transmission. Selection mechanism is based on flow queue lengths. When a transmission finishes, the station with a privilege, called privileged, starts transmission after a short period of time, SIFS (Short Inter Frame Space), if the channel is sensed idle.

Token-DCF is implemented in the MAC layer of the protocol stack. Scheduling information is embedded in the MAC header of the data packets and is transferred to the stations via overhearing. Token-DCF reduces signaling overhead in its scheduling component compared to central scheduling algorithms. Each station maintains queue length of the neighboring stations. The queue lengths are used in the scheduling component to select the privileged station for the next transmission. Transmitting station announces the privileged station in the privileged field of the MAC header of the data packets it transmits. By overhearing of these packets, the privileged station is informed that it has a higher priority for the next transmission. When a transmission finishes, the privileged station can start transmission after SIFS, if the channel is sensed idle. Note that in multi-hop networks, at Fig. 1: Access method of IEEE 802.11 DCF each time instance, several privileged stations might be present in the network, since in multi-hop networks, non-interfering transmitters transmit at the same time and each of them assigns a privilege to one of their neighbors.

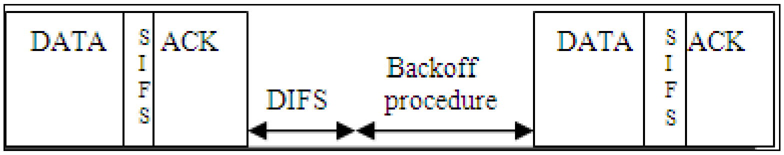

Figure 1: Access method of IEEE 802.11 DCF

Token-DCF is done via embedding the scheduling information in the header of data packets by the source station and overhearing of the packets to retrieve such information by the neighboring stations. When a packet is transmitted, the station that will have higher priority for the next transmission, the privileged station, and the queue length of the transmitter are embedded in the MAC header of the packet. Once a packet is received or overheard, queue length of the source of the packet is saved by the receiving or overhearing station. Furthermore, a station that receives or overhears a packet, checks the privileged field of the MAC header of the packet to find if it is chosen to be the next privileged station.

In Token-DCF, no central scheduler is deployed in the network and no extra control messages are transmitted to find and disseminate a schedule. Collecting the information needed for scheduling, assigning a privilege to one of the neighbors and obtaining the privilege by the privileged station are all done via receiving or overhearing of data packets.

Token-DCF has two major components: (1) A method to reduce the idle time of the backoff mechanism. (2) A scheduling algorithm to determine which neighbor is chosen as the privileged station.

4.2. Reducing Idle Time

Token-DCF reduces the idle time of the backoff mechanism by assigning privileges to network stations. When a station transmits data packets, it might give higher priority for the next transmission to one of its neighbors. A transmitting station gives a high priority to one of its neighbors with probability p. With probability p, no station is given a higher priority. The scheduling algorithm of Token-DCF determines which neighbor is chosen as the privileged station, i.e., the station with a higher priority. When a transmission finishes, a station that has a privilege starts transmission after short period of time, SIFS, if the channel is sensed idle. Non-privileged stations follow the backoff mechanism of IEEE 802.11 to access the wireless medium.

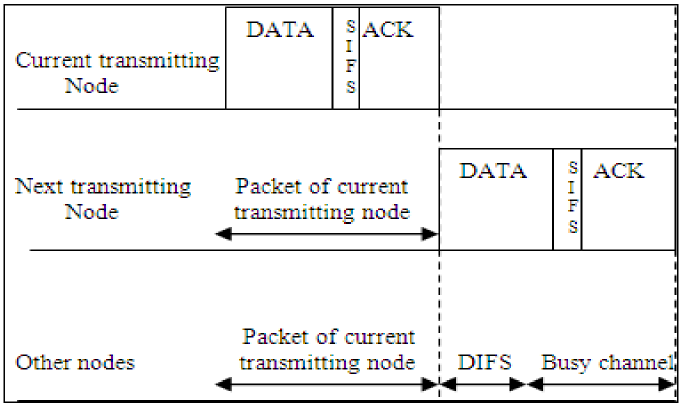

Figure 2

Backoff mechanism of 802.11 DCF is shown in Figure 1. In this mechanism, after a transmission finishes, the station senses the channel after DIFS interval and if the channel is sensed idle, it waits for a random contention time: it chooses backoff b, an integer distributed uniformly in the window [0;CW] and waits for b time slots before attempting to transmit. Channel access mechanism of our protocol, Token-DCF, Access method of Token-DCF protocol shown in Figure 2. In Token-DCF, when the channel becomes idle, the privileged station, if there is any, starts transmission on the channel immediately, and non-privileged stations have to defer backoff countdown till when transmission of the privileged station finishes. The process of giving a privilege by a transmitting station to one of its neighbors repeats in each transmission. Whenever a privileged station transmits on the channel, the idle time of the channel is only SIFS. On the other hand, in IEEE 802.11 protocol, the channel idle time between two consecutive transmissions is equal to DIFS plus random backoff duration.

4.3. Scheduling algorithm

The scheduling algorithm of Token-DCF provides a mechanism for choosing the privileged stations. Information needed in the scheduling component of the protocol is embedded in the MAC header of data packets. Such information includes queue length of transmitter of the packet and the next privileged station. Each station keeps track of neighbor’s queue length in order to enable neighbor scheduling. In Token- DCF, when a station transmits, it acts as a scheduler as well and with probability p gives a higher priority for the next transmission to one of its neighbors. This technique removes the need for a separate scheduler as well as transmission of control messages between the scheduler and network stations.

In central scheduling algorithms, scheduler component and network hosts must exchange control information to coordinate the schedule. As a trade off, our approach is opportunistic and uses message overhearing to exchange the information needed in the scheduling component. Different mechanisms can be used to choose the privileged station. In this paper, we presently consider only single hop flows (i.e., sender and receiver are adjacent nodes), but our ideas can be extended to multi-hop flows as well. A station might choose the neighbor with the largest queue length as the next privileged station. If this policy is implemented as the scheduling component of the protocol, a transmitting node should announce its queue length, qi, as well as its link capacity, cij , in the packets it transmits.

In single hop networks, if every station overhears packets of every other station, this policy implements backpressure scheduler, explained in Section II-C. In single hop networks, data transmission of any two links interferes with each other and as a result, at each time instance, at most one link can be scheduled for transmission. If network is single hop and every station overhears every other transmission, each station knows queue length qi and capacity cij of other network links and schedules the link with the largest qicij as the privileged link. In single hop networks, the link with largest qicij is the one that maximizes probability. In practice, cij may be approximated by the transmission rate used by the MAC-layer rate control algorithm.

Another scheduling policy is to pick the link with the longest queue. In single hop networks, when every station overhears every transmission, this policy implements longest queue first (LQF) as the scheduling component of Token- DCF. LQF is throughput optimal if the so called local pooling condition is satisfied [12]. In our simulations, we have used LQF as the scheduling component of Token-DCF, and all transmissions occur at a fixed rate.

4.4. Protocol details

A station that is going to transmit on the channel, with probability p, chooses one of its neighbors to have a higher priority for transmission. With probability p, no station is chosen to have a privilege. p is initially set to zero and changes during the protocol execution in order to adapt the probability of giving a privilege to neighbors. Active denotes the set of neighbors of a station that has transmitted on the channel during the current scheduling period and the transmission is overheard by the station. The station itself, myId, is also included in the set active. When a station transmits, it might give a privilege to one of the stations in the set active. By including myId in the set active, a station might choose itself as the privileged station. Each station keeps track of the transmissions on the channel by overhearing of the packets. Success denotes the number of transmissions from the set active. Fail denotes the number of transmissions in which the sender of the packet is not in the set active. Protocol parameters are reset to initial values each period seconds.

Protocol parameters are reset periodically in order to prevent stale information making the protocol unfair. An alternative to this method (i.e., resetting the initial values) is to use moving average for adapting parameter values during the protocol execution. If the packet is a MAC data packet, the station might give a privilege to one of its neighbors. The mechanism of assigning a privilege or transmitting as the privileged station is not used when control packets are transmitted. In this way, the transmissions of non-data packets such as ARP packets or routing packets are not affected by our protocol. The station chooses a privileged station with probability p, where privileged is the station in the set active with the longest queue. With probability p, no station is given a privilege. If a station chooses itself as the privileged, it sets its flag to 1. Otherwise, flag is set to 0. Flag equals to 1 means that the station has a privilege for the next transmission on the channel.

4.4.1. Procedure for Coding in MAC Protocol

- p = 0
- active = fmyIdg
- success = 0
- fail = 0
- call Initialization after period
- if it is a MAC data packet then
- with probability p
- privileged = station with the longest queue in
- active
- if privileged == myId then

- flag = 1
- else
- flag = 0
- Adapt
- else
- privileged = null

5. Testing and Results

We simulate Token-DCF and 802.11 DCF in NS2 to measure and compare performance of these two MAC protocols. The network is a wireless ad hoc network in which transmitting stations are placed uniformly at random in a square area. Flows are single hop, and the receiver of each flow is placed at a distance of 100m from the transmitter of the flow. We run the simulations for different network sizes, including single-hop and multi-hop networks. The effective transmission range in the simulations is limited to 250 meters and carrier sense range is limited to 550 meters. IEEE 802.11 RTS/CTS mechanism is turned off. Two-ray ground radio propagation model is assumed. Packet payload size is 1500 bytes. Each simulation lasts for 30 seconds and the presented results are averaged over 20 runs. In each run, a different random network topology is considered. In simulations, the scheduling algorithm presented in Token-DCF is used as the scheduling component. We measure the performance of Token-DCF and 802.11 DCF in terms of aggregate *throughput, average access delay, channel idle time and collision frequency.*

6. Conclusion

Token-DCF is a distributed MAC protocol that uses an opportunistic overhearing mechanism to schedule network stations for transmission on the channel. The main design goal of Token-DCF is to reduce both idle time and collision time by introducing an implicit token passing algorithm. The simulation results of Token- DCF should achieve a better throughput and access delay due to its dynamic and distributed nature.

7. Acknowledgment

I would express my gratitude and appreciation to all those who gave me the possibility to initiate the review paper. A special thanks to my final year project coordinator, Dr. Arvind. I wish to express my deepest gratitude and thanks to Mr. Nataraju A.B, Assistant professor, Department of Electronics and Communication for helping me throughout and guiding me from time to time.

8. References

1. F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the ieee 802.11 protocol to achieve a theoretical throughput limit," IEEE/ACM Trans. On Networking, vol. 8, no. 6, December 2000.
2. H. Kim and J. Hou, "Improving protocol capacity with model-based frame scheduling in ieee 802.11-operated wlans," in Mobicom, 2003.
3. Y. C. Tay, K. Jamieson, and H. Balakrishnan, "Collision-minimizing csma and its applications to wireless sensor networks," IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, vol. 22, 2004.
4. M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless lans," SIGCOMM Comput. Commun. Rev., vol. 35, no. 4, 2005.
5. X. Yang and N. H. Vaidya, "A wireless mac protocol using implicit pipelining," IEEE Trans. on Mobile Computing, vol. 5, no. 3, 2006.
6. Z. Zeng, Y. Gao, K. Tan and P. R. Kumar, "CHAIN: Introducing minimum controlled coordination into random access MAC," In Proceedings of INFOCOM 2011, pp. 2669-2677, 2011.
7. ANSI/IEEE Standard 802.4, "Token-passing bus access method and physical layer specifications," IEEE, 1985.
8. "IEEE Standards: P802.5 Working Group Area," Ieee802.org, E. E. Johnson, Z. Tang, M. Balakrishnan, J. Rubio, H. Zhang.
9. S. Sreepuram, "Robust token management for unreliable networks," In Proceedings of the 2003 IEEE conference on Military communications, MILCOM'03, vol. 1, 2003.
10. M. Ergen, D. Lee, A. Puri, P. Varaiya, R. Attias, R. Sengupta, S.Tripakis, "Wireless Token Ring Protocol," Proceedings of the Eighth IEEE Symposium on Computers and Communications (ISCC 2003) pp. 710-715, 2003.
11. L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," IEEE Transactions on Automatic Control, 37(12): pp. 1936-1948, 1992.
12. Dimakis and J. Walrand. "Sufficient Conditions for Stability of Longest-Queue-First Scheduling: Second-order Properties using Fluid Limits." Advances in Applied Probability, 38(2):505521, 2006.
13. L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," 46th Annual Allerton Conference on Communication, Control, and Computing, Page(s):1511 - 1519, September 2008.
14. J. Ni and R. Srikant, "Distributed CSMA/CA algorithms for achieving maximum throughput in wireless networks", Information Theory and Applications Workshop, Page(s):250 - 261, February 2009.
15. T. H. Kim, J. Ni, R. Srikant and N. H. Vaidya, "On the achievable throughput of CSMA under imperfect carrier sensing", In Proc. IEEE INFOCOM, April 2011.
16. Ghazale Hosseinabadi and Nitin Vaidya "Token-DCF: An Opportunistic MAC protocol for Wireless Networks", 2013