Fast Cost EfFective Q-CSMA In Wireless Networks

Prof N.Senthilkumar [1], Rinu Elizabeth Kurian [2]

[1] Associate professor, [2] PG student

Vivekananda College of Engineering for Women, Anna University, Chennai.

***Abstract*- The various research focuses in wireless networks are cluster management, Quality of Service and medium access. In this paper we focus on a method to improve the quality of Service. The quality of Service can be improved by improving throughput and delay. Recent paper shows that carrier-sense multiple access (CSMA)-type random access algorithms can achieve the maximum possible throughput in ad hoc wireless networks. However, these algorithms assume an idealized continuous-time CSMA protocol where collisions can never occur. In addition, simulation results indicate that the delay performance of these algorithms can be quite bad. On the other hand, although some simple heuristics (such as greedy maximal scheduling) can yield much better delay performance for a large set of arrival rates, in general they may only achieve a fraction of the capacity region. This paper addresses a distributed, throughput-optimal CSMA/CA for wireless ad hoc networks, which is named as the pre-emptive CSMA/CA. It is completely distributed in a sense that it only requires carrier sense results from the outside of a link (or a node) and it runs with low complexity. At the same time, our CSMA/CA is practical enough to achieve the optimality even with discrete back off time, nonzero carrier sense delay and data packet collisions.**

***Index Terms*-** *Pre*-emptive CSMA, QCSMA.

I.INTRODUCTION

For wireless networks with limited resources, efficient resource allocation plays a crucial role in achieving high performance and providing satisfactory quality of service (QoS). In this paper, we study *link scheduling* (or *medium access control*, MAC) for wireless networks, where the links in the network may not be able to transmit simultaneously due to interference. A *scheduling algorithm* (or *MAC protocol*) decides which links can transmit data at each time instant so that no two active links interfere with each other.

stable. The delay performance of a scheduling algorithm can be characterized by the average delay experienced by the packets transmitted in the network. Since many wireless applications nowadays have stringent bandwidth and delay requirements, designing high-performance scheduling algorithms to achieve maximum throughput and low delay is of great importance, which is the focus of this paper.We also want the scheduling algorithms to be distributed and have low complexity/overhead since, in wireless scenarios, normally there is no central entity and the resources at the nodes are very limited.

Even though the QCSMA [8] is throughput optimal the delay performance of Q-CSMA can be quite bad when the traffic intensity is high. [1] is also throughput optimal but it is assumed that t the sensing time is negligible, so that there is no collision. *Maximal scheduling* is a low-complexity alternative to MWS, but it may only achieve a small fraction of the capacity region[3] *Greedy Maximal Scheduling* (GMS), also known as *Longest-Queue-First* (LQF), is another natural low-complexity alternative to MWS that has been observed to achieve good throughput and delay performance [7]. However, for networks with general topology, GMS may only achieve a fraction of the capacity region.Pre-emptive CSMA in an optimum way to improve both the throughput and delay performance.

II. MODEL

We model a single channel wireless ad hoc network by a graph G = (V,E) where V is the set of nodes and E is the set of links. Nodes are wireless transmitters/receivers. There exists a

directed link (n,m) ∈ E if node m can hear the transmission of node n. We assume that if(n,m) ∈ E Then (m,n) ∈ E. For any link i ∈ E, we use C(i) to denote the set of conflicting links (called *conflict set*) of i , i.e., C(i) is the set of links such that if any one of them is active, then link cannot be active. The conflict set C(i) may include the following:

• links that share a common node with link i:This models the *node-exclusive constraint* where two links sharing a common node cannot be active simultaneously;

• links that will cause interference to link I when transmitting: This models the *radio interference constraint* where two links that are close to each other cannot be active simultaneously.

We assume symmetry in the conflict set so that if i ∈ $c(j)$, Then $j \in c(i)$, . We consider a time-slotted system. A *feasible* (*collision-free*) *schedule* of G=(V,E) is a set of links that can be active at the same time with respect to the conflict set constraint, i.e., no two links in a feasible schedule conflict with each other. Without loss of generality, we assume that all links have unit capacity, i.e., an active link can transmit one packet in one timeslot under a feasible schedule. A schedule is represented by a vector $\in \{0,1\}^E$. The i th element of is equal to 1 (i.e, $x_i = 1$) if link i is included in the schedule; $x_i = 0$otherwise. With a little abuse of notation, we also treat x as a set $i \in x$ and write if $x_i = 1$. Note that a feasible schedule x satisfies

$x_i + x_j \leq 1$ for all $i \in E$ and$j \in c(i)$ (1)

Let M be the set of all feasible schedules of the network. A *scheduling algorithm* is a procedure to decide which schedule to be used (which set of links to be activated) in every timeslot for data transmission. In this paper, we focus on the MAC layer so we only consider single-hop traffic. The *capacity region* of the network is the set of all arrival rates λ for which there exists a scheduling algorithm that can stabilize the queues, i.e., the queues are bounded in some appropriate stochastic sense depending on the arrival model used. For the purposes of this paper, we will assume that if the arrival process is stochastic, then the resulting queue length process admits a Markovian description, in which case stability refers to the positive recurrence of this Markov chain. It is known that the capacity region is given by

$$\Lambda = \{\lambda \mid \exists \mu \in Co(M), \lambda < \mu$$

where Co(M) is the *convex hull* of the set of feasible schedules in M . When dealing with vectors, inequalities are interpreted component wise. We say that a scheduling algorithm is *throughput-optimal*, or achieves the *maximum throughput*, if it can keep the network stable for all arrival rates in Λ.

III.BASIC SCHEDULING ALGORITHM

We divide each timeslot into a *control* slot and a *data* slot. The purpose of the control slot is to generate a collision- free *transmission schedule x(t)∈ M* used for data transmission in the data slot. To achieve this, the network first selects a set of links that do not conflict with each other, denoted by m(t) . Note that these links also form a feasible schedule, but it is not the schedule used for data transmission. We call the *decision schedule* in timeslot t.

Let Mo ⊆ M be the set of possible decision schedules. The network selects a decision schedule according to a randomized procedure. Then, the transmission schedule is determined as follows. For any link i in m(t) in , if no links in C(i) were active in the previous data slot, then link is chosen to be *active* with an *activation probability* p_i and *inactive* with probability 1-p_i in the current data slot. If at least one link in C (i) was active in the previous data slot, then will be inactive in the current data slot. Any link not selected by m (t) will maintain its state (active or inactive) from the previous data slot.

Algorithm 1: Basic Scheduling Algorithm (in Timeslot t)

1. In the control slot, randomly select a decision schedule m(t) ϵ Mo with probability $\alpha(m(t))$

∀i ∈ m(t) :

If no links in C(i) were active in the previous data slot,

i.e ,$\sum_{j \in c(i)} x_j \quad (t-1) = 0$

(a) x_i (t) =1 with probability p_i,0<p_i<1

(b) x_i (t) =0 with probability $\bar{p}_i$=1- p_i.

Else

(c) x_i (t) =0.

$\forall i \ni m(t)$:

(d) $x_i(t) = x_i(t-1)$

2. In the data slot, use x (t) as the transmission schedule.

IV. PREEMPTIVE CSMA/CA

We introduce the preemptive CSMA/CA in this section. It is called the preemptive CSMA/CA as the defining feature is that some links preempt the medium access of others. The preemption for the medium access is to the extent of one link's conflict link set, Ci by our carrier sense model. The preemptive CSMA/CA can be implemented in a completely distributed way. The only information that needs to come from the outside of a link is the result of the carrier sense operation.

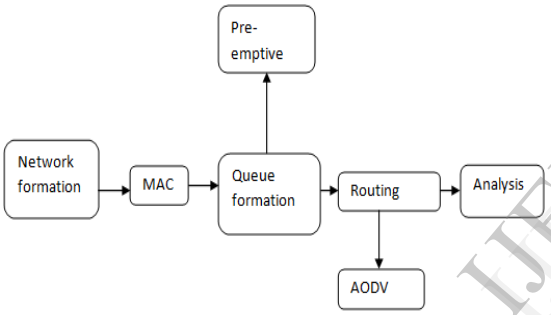

Figure 2: Block diagram of the pre-emptive CSMA

In the network formation, wireless sensor nodes are created in 50 numbers. We construct a multi hop sensor network by using IEEE 802.11 MAC layer where several terminals communicate within a multiple access network that incorporates a shared medium. We use OMNI type antenna which radiates power uniformly in all directions. The omni-directional antenna radiates or receives equally well in all directions. It is also called the "non directional" antenna because it does not favour any particular direction. We use AODV routing protocol here. AODV is a method of routing messages between mobile computers. It allows these mobile computers, or nodes, to pass messages through their neighbours to nodes with which they cannot directly communicate.

Algorithm 2: Pre-emptive CSMA/CA for link i at time t

/* Ber(p): Bernoulli trial with prob. p */

$$If \sum_{j \epsilon ci} uj(t-1) = 0$$

$$If\ ui(t-1) = 1$$

$$xi(t) \leftarrow Ber(pi)$$

Else

$$xi(t) \leftarrow Ber(ai)$$

Else

$$xi(t) = 0$$

Transmit packet by schedule x(t)

Update

$$ui(t) = \begin{cases} 1 & if\ xi = 1\ and\ successful \\ 0 & otherwise \end{cases}$$

The details of the proposed CSMA/CA are shown in Algorithm 2. The function Ber(p) returns 1 with probability p and 0 with 1 - p. By the pre-emptive CSMA/CA in Algorithm 2 a link maintains two internal state variables: transmission schedule xi(t) and pre-emption ui(t). At the beginning of a time slot, a link refers to the report from the carrier sense at the previous slot. If there was no pre-emptive transmission from other conflict neighbours, the link observes ui(t-1) to see if it has been pre-empting the medium access of its conflict neighbours. If so, it selects with probability pi that it continue to pre-empt the access of other links by another transmission. Thus, the pre emptive transmission will be finished only when the link i decides so, and no interference can be generated to the pre-emptive transmission under the ideal carrier sense assumption. When the link that has pre empted others decides not to pre-empt them any more by drawing 0 on Line xi(t) ← Ber(pi), the slot t is left idle since all neighbours will not access the medium due to its carrier sense busy status at time t-1. Essentially, the idle slot is used to signal other neighbour links in conflict that the medium is released from the pre-emption. By Line xi(t)← Ber(ai), the links that have not pre-empted the neighbours (ui(t - 1) = 0) compete only when the medium I the previous slot was free from the pre-emptive transmission If free, the links set xi(t) = 1

with probability ai. After xi(t)'s are updated, the links in the networks transmit a packet and/or observe the medium according to xi(t). If xi(t) = 1, a packet is transmitted, and the transmission result is monitored to update ui(t). If the transmission is successful, which is typically informed by an ACK packet reception, it obtains the pre-emption for the medium access by having ui(t) = 1. Otherwise, it observes the medium to see if the medium access to the next slot is pre-empted by one of others.

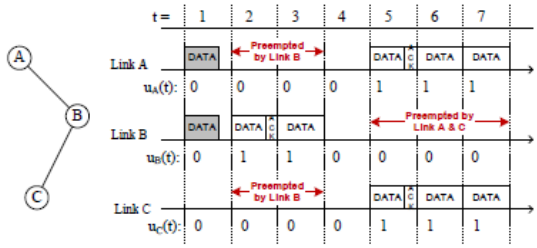

Figure 1:An example of the progress of internal variables by the preemptive CSMA/CA. The left is the conflict graph of the considered network. The gray boxes indicate the packets in collision.

Figure 1 shows one example of the schedules by the pre-emptive CSMA/CA. The network has three links, A, B and C where A and B conflict and so do B and C. In the first slot link A and B transmit at the same time, resulting in a collision. In the second slot, link B succeeds in a transmission and thus, pre-empts the medium access of links A and C at the third slot. In the fourth slot, link B decides not to pre-empt others any more, leaving that slot idle. As discussed, that idle slot cannot be used by other links, either. In the fifth slot, link A and C transmit and succeed together as they do not conflict with each other.

The key strategy for the throughput-optimality of the pre-emptive CSMA/CA is to lengthen the duration of the pre-emptive transmission proportional to the queue lengths. As the network gets more congested, such duration becomes longer and longer, and the loss by CSMA/CA such as collision and backoff is relatively small. Thus, in the extremely congested state, the loss is virtually zero and the optimality is achieved.

Our pre-emptive CSMA/CA explicitly takes into account the loss by random back off and data packet collisions. The loss by non-zero time for the carrier sense, which corresponds to α out of each pre-emptive transmission, is not considered for establishing the optimality. However, as the pre-emptive transmission gets longer, the fraction α to the length of the pre-emptive transmission becomes negligible. Moreover, the time may be used to receive an ACK packet from the receiver as shown in Figure1 in practice.

V. CONCLUSION

We have proposed the preemptive CSMA/CA, which is completely distributed and throughput-optimal. The algorithm is obtained by extending Q-CSMA which has no data collision. the key for the optimality is to have the preemptive transmission of which duration is proportional to the queue lengths, and it gives the throughput-optimality even with the loss by the random nature of CSMA/CA.

REFERENCES

[1]. Gil Zussman, Andrew Brzezinski, and Eytan modiano " multihop local pooling for distributed throughput maximization in wireless networks" MIT/LIDS Technical Report 2719, July 2008

[2]. Libin Jiang and Jean Walrand "A distributed csma algorithm for throughput and utility maximization in wireless networks" forty-sixth annual Allerton conference Allerton house, UIUC, Illinois, USA September 23-26, 2008

[3]. Prasanna Chaporkar, Koushik Kar, Member, Ieee, Xiang Luo, And Saswati Sarkar, Member, IEEE,"Throughput And Fairness Guarantees Through Maximal Scheduling In Wireless Networks". IEEE Transactions On Information Theory, Vol. 54, No. 2, February 2008

[4]. Ajit Warrier, Sankararaman Janakiraman, Sangtae Ha And Injong Rhee IEEE. "Diffq: Practical Differential Backlog Congestion Control For Wireless Networks" IEEE/ACM Transactions On Networking, Vol. 14, January 2009

[5]. Changhee Joo, Member, IEEE, And Ness B. Shroff, Fellow, IEEE. "Performance Of Random Access Scheduling Schemes In Multi-Hop Wireless Networks" IEEE/AM Transactions On Networking, Vol. 17, No. 5, October 2009

[6]. Umut Akyol, Winlab, Rutgers University Nj, Matthew Andrews Bell Labs, Murray Hill Nj, "Joint Scheduling And Congestion Control In Mobile Ad-Hoc Networks" IEEE/ACM Transactions On Networking, February 2010

[7]. Mathieu Leconte, Jian Ni, Member, IEEE, And R. Srikant, Fellow, IEEE, "Improved Bounds On The Throughput Efficiency Of Greedy Maximal Scheduling In Wireless Networks" IEEE/ACM Transactions On Networking, Vol. 19, No. 3, June 2011

[8] Jian Ni, Member, IEEE, Bo (Rambo) Tan, Student Member, IEEE, And R. Srikant, Fellow, IEEE,"Q-CSMA: Queue-Length-Based CSMA/CA Algorithms For Achieving Maximum Throughput And Low Delay In Wireless Networks" IEEE/ACM Transactions On Networking, Vol. 20, No. 3, June 2012