International Journal of Information & Computation Technology.
ISSN 0974-2239 Volume 2, Number 2 (2012), pp. 125-132

http://www. ripublication.com

A Distributed CSMA Algorithm for Maximizing Throughput in Wireless Networks

[1]K K S Harika, [2]Dr P Harini, [3]M Kiran Kumar and [4]K Kondaiah

[1&4]II M. Tech, [2]Professor, [3]Assitant Professor
[1,2&3]CSE, Sacet, Chirala, Prakasam
[4]CSE, IITT, SIDDIPET
E-mail: harika.kollipara@gmail.com, hpogadanda@yahoo.com, kiran8kumar@gmail.com, kondaya.kuppala@gmail.com

Abstract

In multihop wireless networks, designing distributed scheduling algorithms to achieve the maximal throughput is a challenging problem because of the complex interference constraints among different links. Traditional maximal-weight scheduling (MWS), although throughput-optimal, is difficult to implement in distributed networks. On the other hand, a distributed greedy protocol similar to IEEE 802.11 does not guarantee the maximal throughput. In this paper, we introduce an adaptive carrier sense multiple access (CSMA) scheduling algorithm that can achieve the maximal throughput distributively. Some of the major advantages of the algorithm are that it applies to a very general interference model and that it is simple, distributed, and asynchronous. Furthermore, the algorithm is combined with congestion control to achieve the optimal utility and fairness of competing flows. Simulations verify the effectiveness of the algorithm. Also, the adaptive CSMA scheduling is a modular MAC-layer algorithm that can be combined with various protocols in the transport layer and network layer. Finally, the paper explores some implementation issues in the setting of 802.11 networks.

Introduction

In Multihop wireless networks, it is important to efficiently utilize the network resources and provide fairness to competing data flows. These objectives require the cooperation of different network layers. The transport layer needs to inject the right amount of traffic into the network based on the congestion level, and the MAC layer needs to serve the traffic efficiently to achieve high throughput. Through a utility optimization framework [1], this problem can be naturally decomposed into congestion control at the transport layer and scheduling at the MAC layer. It turns out that MAC-layer scheduling is the bottleneck of the problem [1]. In particular, it is not easy to achieve the maximal throughput through distributed scheduling, which in turn prevents full utilization of the wireless network. Scheduling is challenging since the conflicting relationships between different links can be complicated.

It is well known that maximal-weight scheduling (MWS) [22] is *throughput-optimal*. That is, that scheduling can support any incoming rates within the capacity region. In MWS, time is assumed to be slotted. In each slot, a set of non conflicting links (called an "independent set," or "IS") that have the maximal weight are scheduled, where the "weight" of a set of links is the summation of their queue lengths. (This algorithm has also been applied to achieve 100% throughput in input-queued switches [23].) However, finding such a maximal-weighted IS is NP-complete in general and is hard even for centralized algorithms. Therefore, its distributed implementation is not trivial in wireless networks.

A few recent works proposed throughput-optimal algorithms for certain interference models. For example, Eryilmaz *et al.* [3] proposed a polynomial-complexity algorithm for the "two-hop interference model".1 Modiano *et al.* [4] introduced a gossip algorithm for the "node-exclusive model".2 The extensions to more general interference models, as discussed in [3] and [4], involve extra challenges. Sanghavi *et al.* [5] introduced an algorithm that can approach the throughput capacity (with increasing overhead) for the node-exclusive model.

On the other hand, a number of low-complexity but suboptimal scheduling algorithms have been proposed in the literature. By using a distributed greedy protocol similar to IEEE 802.11, [8] shows that only a fraction of the throughput region can be achieved (after ignoring collisions). The fraction depends on the network topology and interference relationships. The algorithm is related to Maximal Scheduling [9], which chooses a maximal schedule among the nonempty queues in each slot. Different from Maximal Scheduling, the Longest-Queue-First (LQF) algorithm [10]–[13] takes into account the queue lengths of the nonempty queues. It shows good throughput performance in simulations. In fact, LQF is proven to be throughput-optimal if the network topology satisfies a "local pooling" condition [10], [12] or if the network is small [13]. In general topologies, however, LQF is not throughput-optimal, and the achievable fraction of the capacity region can be characterized as in [11]. Reference [14] studied the impact of such imperfect scheduling on utility maximization in wireless networks. In [16], Proutiere *et al.* developed asynchronous random-access-based scheduling algorithms that can achieve throughput performance similar to that of the Maximum Size scheduling algorithm.

Our first contribution in this paper is to introduce a *distributed* adaptive carrier sense multiple access (CSMA) algorithm for a general interference model. It is inspired by CSMA, but may be applied to more general resource sharing problems (i.e., not limited to wireless networks). We show that if packet collisions are ignored (as in some of the mentioned references), the algorithm can achieve maximal throughput. The optimality in the presence of collisions is studied in [30] and [31] (and also in [35] with a different algorithm). The algorithm may not be directly comparable to those throughput-optimal algorithms we have mentioned since it utilizes the carrier-sensing capability. However, it does have a few distinct features:

- Each node only uses its local information (e.g., its backlog). No explicit control messages are required among the nodes.
- It is based on CSMA random access, which is similar to the IEEE 802.11 protocol and is easy to implement.
- Time is not divided into synchronous slots. Thus, no synchronization of transmissions is needed.

Our second contribution is to combine the proposed scheduling algorithm with congestion control using a novel technique to achieve fairness among competing flows as well as maximal throughput (Sections III and IV). The performance is evaluated by simulations (Section VI).We show that the proposed CSMA scheduling is a modular MAC-layer algorithm and demonstrate its combination with optimal routing, any cast, and multicast with network coding [40]. Finally, we considered some practical issues.

There is extensive research in joint MAC and transport-layer optimization, for example [6] and [7]. Their studies have assumed the slotted-Aloha random access protocol in the MAC layer instead of the CSMA protocol we consider here. Slotted-Aloha does not need to consume power in carrier sensing. On the other hand, CSMA has a larger capacity region.(In this paper, we are primarily interested in the throughput performance.) Other related works assume physical-layer models which are quite different from ours. For example, [18] considered the CDMA interference model, and [19] focused on time-varying wireless channels.

Adaptive CSMA for Maximal Throughput

Interference Model

First, we describe the general interference model we will consider in this paper. Assume there are K *links* in the network, where each *link* is an (ordered) transmitter–receiver pair. The network is associated with a conflict graph (or "CG"). G= {V, ε} where V is the set of vertices and ε is the set of edges. Two links cannot transmit at the same time (i.e., "conflict") iff there is an edge between them. Note that this framework includes the "node-exclusive model" and "two-hop interference model" mentioned as two special cases.

An Idealized CSMA Protocol and the Average Throughput

There are two reasons for using this model in our context, although it makes the above simplifying assumptions about collisions and the HN problem: 1) The model is simple, tractable, and captures the essence of CSMA/CA. It is also an easier starting point before analyzing the case with collisions. Indeed, in [30], [31], we have developed a more general model that explicitly considers collisions in wireless network and extended the distributed algorithms in this paper to that case to achieve throughput-optimality. This will be further discussed in Section VII. 2) The algorithms we propose here were inspired by CSMA, but they can be applied to more general resource-sharing problems4 that does not have the issues of collisions and HN (i.e., not limited to wireless networks).

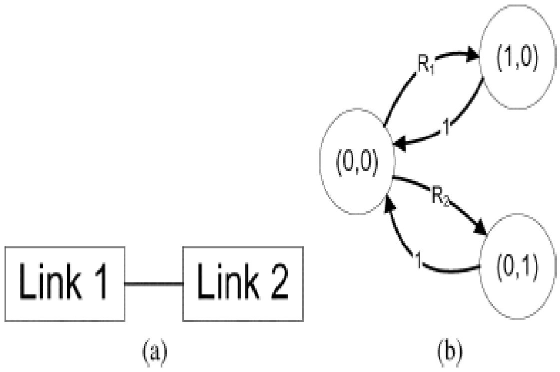

Figure 1: Example A conflict graph and corresponding CSMA Markov chain (a) Conffict graph (b) CSMA Markov chain

Fig. 1 gives an example network whose CG is shown in (a). There are two links with an edge between them, which means that they cannot transmit together. Fig. 1(b) shows the corresponding CSMA Markov chain. State (0,0) means that no link is transmitting, state (1,0) means that only link 1 is transmitting, and (0,1) means that only link 2 is transmitting. The state (1,1) is not feasible.

Lemma 1: ([25]–[27]) The stationary distribution of the CSMA Markov chain has the following product-form:

$$p(x^i; \mathbf{r}) = \frac{\exp\left(\sum_{k=1}^{K} x_k^i r_k\right)}{C(\mathbf{r})}$$

Where

$$C(\mathbf{r}) = \sum_j \exp\left(\sum_{k=1}^{K} x_k^j r_k\right).$$

Note that the summation is over all feasible states

Proof: We verify that the distribution (1)

$$\frac{p(x^i + \mathbf{e}_k; \mathbf{r})}{p(x^i; \mathbf{r})} = \exp(r_k) = R_k$$

satisfies the detailed balance equations [24]. Consider

Which is exactly the detailed balance equation between state x^j. and $x^i + \mathbf{e}_k$. Such relations hold for any two states that differ in only one element, which are the only pairs that correspond to nonzero transition rates. Thus, the distribution is invariant.

Since the detailed balance equations hold, the CSMA Markov chain is time-reversible. In fact, the Markov chain is a reversible "spatial process," and its stationary distribution (1) is a Markov random field

It follows from Lemma 1 that $s_k(\mathbf{r})$, , the probability that link transmits, is given by

$$s_k(\mathbf{r}) = \sum_i [x_k^i \cdot p(x^i; \mathbf{r})].$$

Without loss of generality, assume that each link K has a capacity of 1. That is, if link K transmits data all the time (without contention from other links), then its service rate is 1 (unit of data per unit time). Then, $s_k(\mathbf{r})$ is also the *normalized* throughput (or service rate) with respect to the link capacity.

Even if the waiting time and transmission time are not exponential distributed but have the same means and 1 (in fact, as long as the ratio of their means is), [27] shows that the stationary distribution (1) still holds. That is, the stationary distribution is insensitive.

Convergence and Stability

The intuition is that one can make change slowly (i.e., "quasi-static") to allow the CSMA Markov chain to approach its stationary distribution (and thus obtaining good estimation of). This allows the separation of time scales of the dynamics of and the CSMA Markov chain. The extended algorithm is

$$r_k(j+1) = [r_k(j) + \alpha(j) \cdot (\lambda'_k(j) + h(r_k(j)) - s'_k(j))]_D$$

Where $D := [0, r_{\max}]$ and the function $h(\cdot) \geq 0$. If $h(\cdot) = 0$, . If then algorithm (9) reduces to Algorithm 1. If , then algorithm (9) "pretends" to serve some arrival rates higher than the actual ones. In Appendix B, we state some results in [38] (which includes the detailed proofs). In summary: 1) with properly chosen decreasing step sizes and increasing update intervals

In a related work [21], Liu *et al.* carried out a convergence analysis, using a differential-equation method, of a utility maximization algorithm extended from [2] (see also Section IV for the algorithm), although queueing stability was not considered in [21].

The Primal-Dual Relationship

In the previous section, we have described the adaptive CSMA algorithm to support any strictly feasible arrival rates. For joint scheduling and congestion control, however, directly using the expression of service rate (3) will lead to a on convex problem. This section takes another look at the problem and also helps to avoid the difficulty.

Rewrite (4) as

$$\max_{\mathbf{r},\mathbf{h}} \quad \left\{\sum_k \lambda_k r_k - \log\left(\sum_j \exp(h_j)\right)\right\}$$
$$\text{s.t.} \quad h_j = \sum_{k=1}^{K} x_k^j r_k,\ \forall j$$
$$r_k \geq 0,\ \forall k.$$

For each $j = 1, 2, \ldots, N,$ associate a dual variable to the constraint $h_j = \sum_{k=1}^{K} x_k^j r_k.$ Write the vector of dual variables as . Then, it is not difficult to find the dual problem of (10) as follows. (The computation was given in [41], but is omitted here due to the limit of space.)

$$\max_{\mathbf{u}} \quad -\sum_i u_i \log(u_i)$$
$$\text{s.t.} \quad \sum_i (u_i \cdot x_k^i) \geq \lambda_k,\ \forall k$$
$$u_i \geq 0, \sum_i u_i = 1.$$

where the objective function is the entropy of the distribution U,

$H(\mathbf{u}) := -\sum_i u_i \log(u_i).$7

Also, if for each K, we associate a dual variable to the constraint $\sum_i (u_i \cdot x_k^i) \geq \lambda_k$ in problem (11), then one can compute that the dual problem of (11) is the original problem $\max_{\mathbf{r}\geq 0} F(\mathbf{r}; \boldsymbol{\lambda}).$ (This is shown in Appendix A as a by-product of the proof of Proposition 2.) This is not surprising since, in convex optimization, the dual problem of dual problem is often the original problem.

What is interesting is that both and have concrete physical meanings. We have seen that is the TA of link . Also, can be regarded as the probability of state. This observation will be useful in later sections. A convenient way to guess this is by observing the constraint. If is the probability of state , then the constraint simply means that the service rate of link , , is larger than the arrival rate.

Conclusion

In this paper, we have proposed a distributed CSMA scheduling algorithm and showed that, under the idealized CSMA, it is throughput-optimal in wireless networks with a general interference model. We have utilized the product-form stationary distribution of CSMA networks in order to obtain the distributed algorithm and the maximal throughput. Furthermore, we have combined that algorithm with congestion control to approach the maximal utility and showed the connection with back-pressure scheduling. The algorithm is easy to implement, and the simulation results are encouraging.

The adaptive CSMA algorithm is a modular MAC-layer component that can work with other algorithms in the transport layer and network layer. In [40], for example, it is combined with optimal routing, any cast, and multicast with network coding. We also considered some practical issues when implementing the algorithm in an 802.11 setting. Since collisions occur in actual 802.11 networks, we discussed a few recent algorithms that explicitly consider collisions and can still approach throughput optimality. Our current performance analysis of Algorithms 1–3 is based on a separation of time scales, i.e., the vector is adapted slowly to allow the CSMA Markov chain to closely track the stationary distribution. The simulations, however, indicate that such slow adaptations are not always necessary. In the future, we are interested to understand more about the case without time-scale separation.

References

[1] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," IEEE J. Sel. Areas Commun., vol. 24, no. 8, pp. 1452–1463, Aug. 2006.

[2] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," in Proc. 46th Annu. Allerton Conf. Commun., Control, Comput., Sep. 23–26, 2008, pp. 1511–1519.

[3] A. Eryilmaz, A. Ozdaglar, and E. Modiano, "Polynomial complexity algorithms for full utilization of multi-hop wireless networks," in Proc. IEEE INFOCOM, Anchorage, AK, May 2007, pp. 499–507.

[4] E. Modiano, D. Shah, and G. Zussman, "Maximizing throughput in wireless networks via gossiping," ACM SIGMETRICS Perform. Eval. Rev., vol. 34, no. 1, pp. 27–38, Jun. 2006.

[5] S. Sanghavi, L. Bui, and R. Srikant, "Distributed link scheduling with constant overhead," in Proc. ACM SIGMETRICS, Jun. 2007, pp. 313–324.

[6] J. W. Lee, M. Chiang, and R. A. Calderbank, "Utility-optimal random-access control," IEEE Trans. Wireless Commun., vol. 6, no. 7, pp. 2741–2751, Jul. 2007.

[7] P. Gupta and A. L. Stolyar, "Optimal throughput allocation in general random access networks," in Proc. Conf. Inf. Sci. Syst., Princeton, NJ, Mar. 2006, pp. 1254–1259.

[8] X. Wu and R. Srikant, "Scheduling efficiency of distributed greedy scheduling algorithms in wireless networks," in Proc. IEEE INFOCOM, Barcelona, Spain, Apr. 2006, pp. 1–12.

[9] P. Chaporkar, K. Kar, and S. Sarkar, "Throughput guarantees in maximal scheduling in wireless networks," in Proc. 43rd Annu. Allerton Conf. Commun., Control, Comput., Sep. 2005, pp. 1557–1567.

[10] A. Dimakis and J. Walrand, "Sufficient conditions for stability of longest-queue-first scheduling: Second-order properties using fluid limits," Adv. Appl. Probab., vol. 38, no. 2, pp. 505–521, 2006.

[11] C. Joo, X. Lin, and N. Shroff, "Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop wireless networks," in Proc. IEEE INFOCOM, Phoenix, AZ, Apr. 2008, pp. 1103–1111.

[12] G. Zussman, A. Brzezinski, and E. Modiano, "Multihop local pooling for distributed throughput maximization in wireless networks," in Proc. IEEE INFOCOM, Phoenix, AZ, Apr. 2008, pp. 1139–1147.

[13] M. Leconte, J. Ni, and R. Srikant, "Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks," in Proc. ACM MobiHoc, May 2009, pp. 165–174.

[14] X. Lin and N. Shroff, "The impact of imperfect scheduling on crosslayer rate control in multihop wireless networks," in Proc. IEEE INFOCOM, Miami, Florida, Mar. 2005, vol. 3, pp. 1804–1814.