ISSN:0975-9646

T.Kalaiselvi et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (2) , 2012,3557-3562

Distributed Algorithm for Improving Performance through Swap Link in Wireless Networks

[1]T.Kalaiselvi [2]P.S.Balamurugan [3].K.Sudhakar

1. PG Scholar, Faculty of Engineering, Karpagam University, Coimbatore.
2. Research Scholar, Anna University, Coimbatore.
3. Assistant Professor (CSE), Hindusthan College of Engineering and Technology, Coimbatore

***Abstract*— In the distributed scheduling algorithms to achieve the maximal throughput is a challenging problem because of the complex interference constraints among different links. This can be done by with scheduling and without scheduling. We calculate the delay, Travel time and we trace the path of the packets sent from source to destination nodes. In addition, simulation results indicate that the delay performance of these algorithms can be quite good in with Scheduling. In this paper we apply an Adaptive carrier sense multiple access (CSMA) scheduling algorithm [2] that can achieve the maximal throughput distributively. Some of the major disadvantages of the algorithm are that it does not consider the packet loss and frequency range of the nodes. Furthermore, the algorithm is combined with congestion control to achieve the optimal utility and fairness of competing flows. So what we show that while reducing the packet loss and increasing the frequency range of the nodes applied with the above algorithm greatly improves the performance of transmission.**

***Keywords*—**Adaptive CSMA, node range, travel time, delay, packet loss

I. INTRODUCTION

IMPROVING performance and throughput 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, this problem can be naturally decomposed into congestion control at the transport layer and scheduling at the MAC layer. Design of Distributed Algorithm to achieve maximal throughput in multi-hop wireless network, That is, that scheduling can support any incoming rates within the capacity region. In this process we are considering the delay, packet loss and range to improve throughput. 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. We use an idealized model of CSMA [2]. This model makes two simplifying assumptions. First, it assumes that if two links conflict, because their simultaneous transmissions would result in incorrectly received packets, then each of the two links hears when the other one transmits. Second, the model assumes that this sensing is instantaneous. Consequently, collisions can be avoided, as we will further explain. The first assumption implies that there are no hidden nodes (HN). This is possible if the range of carrier-sensing is large enough. The second assumption is violated in actual systems because of the finite speed of light and of the time needed to detect a received power. 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, 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. 2) The algorithms we propose here were inspired by CSMA, but they can be applied to more general resource-sharing problems that does not have the issues of collisions and HN (i.e., not limited to wireless networks). Design of distributed scheduling algorithms in wireless networks has been extensively studied under various metrics of efficiency and fairness and for different types of traffic and interference models.

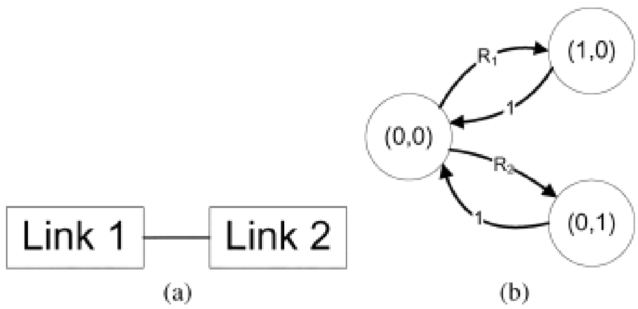


Fig.1. Example: A conflict graph and the corresponding CSMA Markov chain.
(a) Conflict graph. (b) CSMA Markov chain.

We consider a wireless network composed by a set L of L links. Interference is modeled by a symmetric, Boolean matrix

A! {0, 1}L! L, where Akl =1 if link k interferes with link l, and Akl =0 otherwise. Denote by N " {0, 1} L the set of the N feasible link activation profiles, or schedules. A schedule m! N is a subset of non-interfering active links (i.e., for any m! N, k, l! m, Akl =0). We assume that the transmitters can transmit at a fixed unit rate when active. These assumptions on what constitutes "transmission" and "interference", together with several others later in the section, lead to mathematical tractability but also give rise to the theory-practice gaps that will be discussed.

II. Scheduling And Utility Maximization

A. CSMA Interference Model

The network is assumed to handle single-hop data connections. However, the results presented here can be readily extended to multi-hop connections (e.g., using the classical back-pressure ideas). The transmitter of each link is saturated, i.e., it always has packets to send. A scheduling algorithm decides at each time which links are activated. Denote by !s =(!s L ,l ! L) the long-term throughputs achieved by scheduling algorithm s. The throughput vector of any scheduling algorithm has to belong to the rate region! defined by

$$\Gamma = \{\gamma \in \mathbb{R}_+^L : \exists \pi \in \mathbb{R}_+^N, \\ \forall l \in \mathcal{L}, \gamma_l \leq \sum_{m \in \mathcal{N}: m_l = 1} \pi_m, \quad \sum_{m \in \mathcal{N}} \pi_m = 1\}.$$

"N!" N In the above, for any schedule m! N, "m can be interpreted as the proportion of time schedule m is activated. As is a standard in problems with saturated arrivals, the objective is to design a scheduling algorithm maximizing the total network wide utility. Specifically let U: R+ & R is an increasing, strictly concave, differentiable objective function. We wish to design an algorithm to solve the following optimization problem: use of a dual decomposition of the problem into a rate control and a scheduling problem: A virtual queue is associated with each link; a rate control algorithm defines the rate at which packets are sent to the virtual queues, and a scheduling algorithm decides, depending on the level of the virtual queues, which schedule to use with the aim of stabilizing all virtual queues. The main challenge reduces to developing a distributed and efficient scheduling algorithm. Many solutions proposed so far are semi-distributed and require information about the queues to be passed around among the nodes or links. This signaling overhead increases communication complexity and reduces effective throughput. More importantly, for management and security reasons, practical deployment of scheduling algorithms is unlikely going to allow such message passing.

Please submit your manuscript electronically for review as e-mail attachments.

B. Distributed Adaptive carrier sense multiple access (CSMA) algorithm

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 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.

In a related work, we studied a model of CSMA with collisions. It was shown that under the "node-exclusive" interference model, CSMA can be made asymptotically throughput-optimal in the limiting regime of large networks with a small sensing delay. The authors Rajagopalan and Shah independently proposed a randomized algorithm similar to ours in the context of optical networks.

When you submit your final version, after your paper has been accepted, prepare it in two-column format, including figures and tables.

C. ***Formulation Algorithm***

Assume there are flows, and let be their index (). Define if flow uses link, and otherwise. Let be the rate of flow, and be the "utility function" of this flow, which is assumed to be increasing and strictly concave. Assume all links have the same PHY data rates (it is easy to extend the algorithm to different PHY rates). Assume that each link maintains a separate queue for each flow that traverses it. Then, the service rate of flow by link should be not less than the incoming rate of flow to link. For flow, if link is its first link (i.e., the source link), we say.

$$\begin{aligned} \max_{\mathbf{u},\mathbf{s},\mathbf{f}} \quad & -\sum_i u_i \log(u_i) + \beta \sum_{m=1}^{M} v_m(f_m) \\ \text{s.t.} \quad & s_{km} \geq 0, \ \forall k, m : a_{mk} = 1 \\ & s_{km} \geq s_{up(k,m),m}, \ \forall m, k : a_{mk} = 1, k \neq \delta(m) \\ & s_{km} \geq f_m, \ \forall m, k : k = \delta(m) \\ \sum_i u_i x_k^i \geq & \sum_{m: a_{mk}=1} s_{km}, \ \forall k \\ & u_i \geq 0, \sum_i u_i = 1 \end{aligned} \quad ($$

Notice that the objective function is not exactly the total utility, but it has an extra term. As will be further explained in when is large, the “importance” of the total utility dominates the objective function of. (This is similar in spirit to the weighting factor used in as a result, the solution of approximately achieves the maximal utility.

III. Related Work

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.

In reference paper [2], they proposed a discrete-time delay version of the CSMA algorithm. Central to our results is a discrete-time distributed randomized algorithm which is based on a generalization of the so-called Glauber dynamics from statistical physics, where multiple links are allowed to update their states in a single time slot. More importantly, the algorithm allows incorporating delay-reduction mechanisms which lead to very good delay performance while retaining the throughput-optimality property.

Further, a repeated CSMA/CA game is defined, where a station can attack toggle between standard and nonstandard back off configurations with a view of maximizing a long-term utility. While achieving throughput and utility optimization, the delay and packet loss is not concentrated. Identification of misconfigured nodes is not employed which leads to the failure of nodes in the network.

IV. Our Contribution

We have shown that the existing system use CSMA-type algorithms that can achieve the maximum possible throughput in wireless ad hoc networks.

In proposed system we calculate the delay and packet loss. 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 distributed approximations of greedy and Attack optimization has recently been found vulnerable to selfish back-off attacks consisting in nonstandard configuration of the constituent back-off scheme can be formularize in this manuscripts. We calculate the following

a. Node frequency range setting
b. Travel Time
c. Packet loss

In addition to that, here we are using swap link concept. In this concept, if the network is broken it will automatically form a network and will send the data and also find the problem in that broken network and rectify it.

V. Simulation Setup And Results

A. *Setups of Simulation and of Implementation*

To evaluate performance, we compare Adaptive CSMA [2] with the optimal benchmark and the standard 802.11 DCF in both simulation and implementation. In simulation, we implement Adaptive CSMA [2] by changing the CSMA in NS2, where we mainly modified the part that concentrate on packet loss and frequency of the nodes. We disabled ACK operation, so when collision occurs, it lasts for the corresponding holding time. The network is slotted with 1.6ms timeslot and the packet size is set to be 1MB. Through CCA, we then use the same NS2 code to experiment Adaptive CSMA in real hardware after a series of hacking that induces the underlying 802.11 drivers to effectively execute the Adaptive CSMA algorithm. We modify the mechanism of setting CW appropriately so that 802.11 drivers can be turned into a basis for implementing Adaptive CSMA. (a) Per-link CW. In 802.11, CW is maintained at each node, not each link, i.e., one contention window per one interface card. In Adaptive CSMA, back-off counters should be installed per link. We implemented per-link, denoted by nodes N1 and N2, at NS2. In 802.11, there exist two back-off related values. The transfer rate R is first set to be Rmin and then doubles whenever there is a collision. The doubled R value is used when a collided packet is retransmitted. We disable this feature by setting Rmin = Rmax, so that retransmitted packets are not treated in a special way. TCL script is written for network setup at creation of nodes.We applies both with scheduling and without scheduling to show the performance by using graph. Crystal report is generated and viewed as graph. Initially we consider the single-hop transmission and after the successful of this setup it is expanded to multi-hop transmission. Multiple transmissions from number of nodes at a time are allowed. We set device range from 120Hz to 350Hz which is manually increased or decreased by means of the number of nodes present in the network. The packets are made available in the queue until any neighboring node is reachable.

B. *Results of sending packets*

Now we present the preliminary results in a simple transferring of packets, as shown in Figure 2 and3. First we consider 50 nodes in the network we create a packet which is to be transmitted from node 1 to node 36.Initially the device range of all nodes is to set to 120Hz .Now the packets travel through the path N1, N8, N2, N10.The network gets broken from N2 and N10.The packets are in queue. We increase the device range to 200Hz.

After increasing, the nearest node gets the shortest path and the transmission continues and reaches the node N36 by the way of N17, N24, and N30. The travel time of the packets from N1 to N36 is 36second.so far we did not apply scheduling algorithm. When scheduling is done the travel time of the packets is 30 seconds.

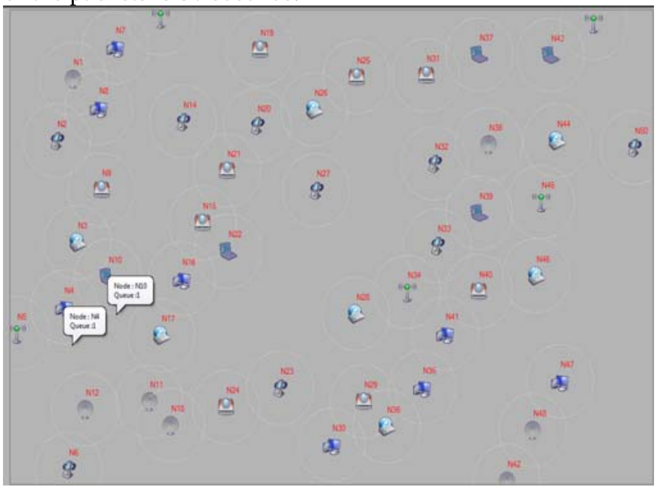
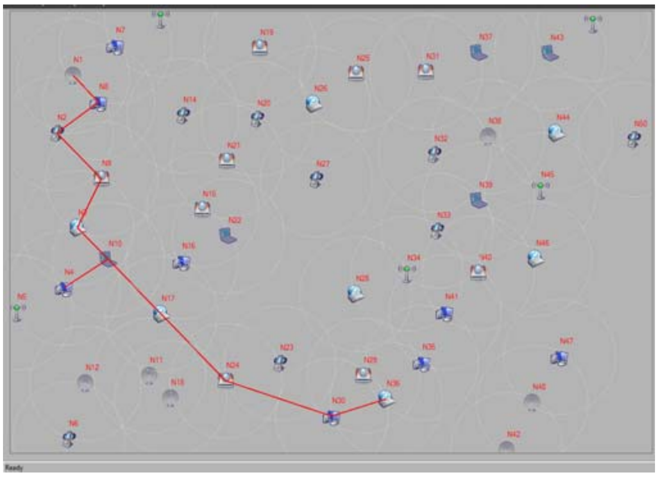


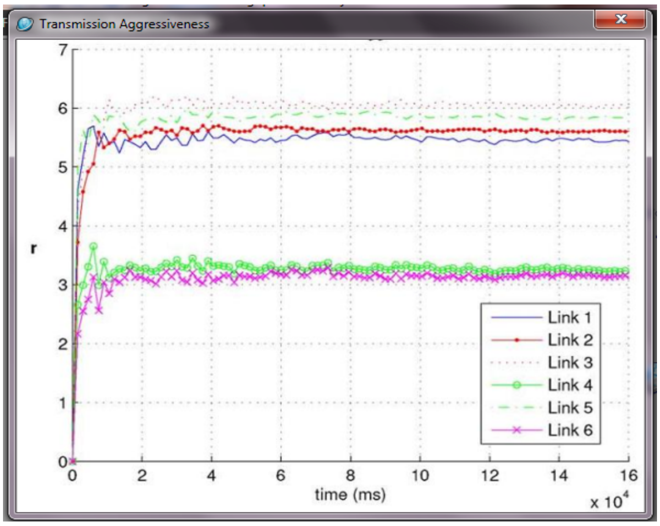
Figure 2-Packets queued in link between two nodes when network is broken.



.Figure 3-Packets resent from node10 when device range is increased

If device range was increased to 240Hz, the path from N1 to N36 is N1, N14, N22, N27, N28, N36 and travel time is 20 seconds and 3seconds without scheduling and with scheduling respectively. It shows the range of the devices and scheduling play a major role in increasing the throughput of the network. So the delay occurs due to the broken nodes and if concentrated on the issue the delay is reduced.

However, in the implementation of adaptive CSMA, we install per- link intermediate buffers to whom packets are injected at congestion controlled rate. The maximum size of the packets that can be sent and queue length is permitted up to 1MB.Each packet has bit stream of length 5 bits. The average length of such intermediate buffers indirectly measures the average queuing delay that a packet experiences.



.Figure 4-Performance Graph showing the transmission rate of the links between the nodes.

In this section, we consider this practical issue and discuss alternative algorithms (for 802.11 networks) that are related to the above algorithms with idealized CSMA. Moreover, similar algorithms (with probe packets RTS/CTS) have been proposed there to approach the maximal throughput and utility by adjusting the mean transmission times with fixed mean back-off times. In it was noted that by using small transmission probability in each mini-slot (which increases the back-off times), and correspondingly increasing the transmission times, the collision probability becomes small, in which case the actual CSMA with collisions can be approximated by the idealized CSMA.

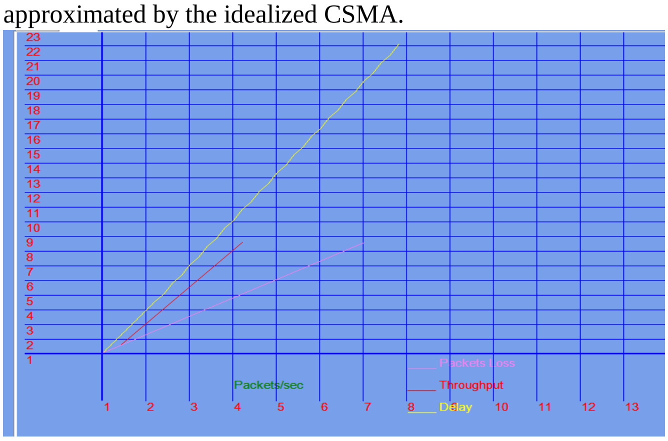


Figure 5. Graph showing packet loss and delay without scheduling

The graph shown in figure 5 indicates there is a maximal delay and packet loss. Eventhough the throughput is maximum packet loss is not reduced. In figure 6, throughput is maximum and the packet loss and delay is very much reduced while we applied scheduling.If the node uses the swaplink mechanism, the delay occurred while misconfiguration of nodes can be avoided and throughput performance is achieved.

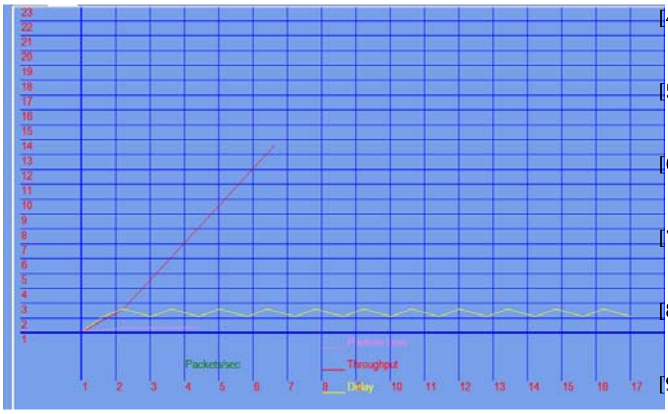


Figure 6. Graph showing packet loss and delay with scheduling

In [3], another protocol was proposed to deal with collisions. The protocol has control phases and data phases. Collisions only occur in the control phase, but not in the data phase. The same product-form distribution (1) can be obtained for the data phase, which is then used to achieve the maximal throughput. In the following, we discussed how to use algorithms in this paper with collisions in mind.

VI. Conclusion

We have used a distributed CSMA scheduling algorithm for showing it is throughput-optimal in wireless networks with a general interference model. We have also shown that with the use of scheduling there is a great reduction in delay and packet loss. We use two techniques of simulation of the nodes that is with scheduling and without scheduling. It clearly identifies the packet loss and delay in the network. Moreover swap link concept reduces the misconfigured nodes by means of using the minimum diameter frequency range of its nearest nodes. If the network is broken it will automatically form a network and will send the data and also find the problem in that broken network and rectify it. So, what we have done clearly states that concentrating in packet loss and nodes frequency settings will greatly achieve maximal throughput.

Acknowledgment

The authors of this paper would thank the reviewers for their valuable comments and suggestions, which help us to improve the quality of this paper.

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, andN. Shroff, "Understanding the capacity region of the greedymaximal scheduling algorithminmulti-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 throughputmaximization inwireless 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 cross-layer rate control in multihop wireless networks," in Proc. IEEE INFOCOM, Miami, Florida, Mar. 2005, vol. 3, pp. 1804–1814.

[15] P. Marbach, A. Eryilmaz, and A. Ozdaglar, "Achievable rate region of CSMA schedulers in wireless networks with primary interference constraints," in Proc. IEEE Conf. Decision Control, 2007, pp. 1156–1161.

[16] A. Proutiere, Y. Yi, and M. Chiang, "Throughput of random access without message passing," in Proc. Conf. Inf. Sci. Syst., Princeton, NJ, Mar. 2008, pp. 509–514.

[17] S. Rajagopalan and D. Shah, "Distributed algorithm and reversible network," in Proc. Conf. Inf. Sci. Syst., Princeton, NJ, Mar. 2008, pp. 498–502.

[18] Y. Xi and E. M. Yeh, "Throughput optimal distributed control of stochastic wireless networks," in Proc. WiOpt, 2006, pp. 1–10. [19] M. J. Neely, E.Modiano, and C. P. Li, "Fairness and optimal stochastic control for heterogeneous networks," IEEE/ACMTrans. Netw., vol. 16, no. 2, pp. 396–409, Apr. 2008.

[19] B.Hajek, "Cooling schedules for optimal annealing,"Math.Oper. Res., vol. 13, no. 2, pp. 311–329, 1988.

[20] J. Liu, Y. Yi, A. Proutiere, M. Chiang, and H. V. Poor, "Conver-gence and tradeoff of utility-optimal CSMA," [Online]. Available:http://arxiv.org/abs/0902.1996

[21] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," IEEE Trans. Autom. Control, vol. 37, no. 12,pp. 1936–1948, Dec. 1992.

[22] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," IEEE Trans. Commun., vol. 47, no. 8, pp. 1260–1267, Aug. 1999.

[23] F. P. Kelly, Reversibility and Stochastic Networks. New York:Wiley, 1979.

[24] R. R. Boorstyn, A. Kershenbaum, B. Maglaris, and V. Sahin, "Throughput analysis in multihop CSMA packet radio networks," IEEE Trans. Commun., vol. COMM-35, no. 3, pp. 267–274, Mar.1987.

[25] X. Wang and K. Kar, "Throughput modelling and fairness issues in CSMA/CAbased ad-hoc networks," in Proc. IEEE INFOCOM,Miami, FL, Mar. 2005, vol. 1, pp. 23–34.

[26] S. C. Liew, C.Kai, J. Leung, and B.Wong, "Back-of-the-envelope computation of throughput distributions in CSMA wireless networks," in Proc. IEEE ICC, 2009, pp. 1–6.

[27] M. Durvy, O. Dousse, and P. Thiran, "Border effects, fairness, and phase transition in large wireless networks," in Proc. IEEE INFOCOM, Phoenix, AZ, Apr. 2008, pp. 601–609.

[28] L. Jiang and S. C. Liew, "Improving throughput and fairness by reducing exposed and hidden nodes in 802.11 networks," IEEE Trans. Mobile Comput., vol. 7, no. 1, pp. 34–49, Jan. 2008.

[29] L. Jiang and J. Walrand, "Approaching throughput-optimality in a distributed CSMA algorithm: Collisions and stability," in Proc. ACM Mobihoc S3 Workshop, May 2009, pp. 5–8.

[30] L. Jiang and J. Walrand, "Approaching throughput-optimality in a distributed CSMA algorithm with contention resolution," UC Berkeley, Tech. Rep., 2009 [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-37.html

[31] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[32] M. J. Wainwright and M. I. Jordan, "Graphical models, exponentiall families, and variational inference," Found. Trends Mach. Learn., vol.1, no. 1-2, pp. 1–305, 2008.

[33] P. Whittle, Systems in Stochastic Equilibrium. New York: Wiley,1986.

[34] J. Ni and R. Srikant, "Distributed CSMA/CA algorithms for achieving maximum throughput in wireless networks," in Proc. Inf. Theory Appl. Workshop, Feb. 2009, p. 250.

[35] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE J. Sel. Areas Commun., vol. 18, no. 3, pp. 535–547, Mar. 2000.

[36] M. J. Neely and R. Urgaonkar, "Cross layer Adaptive control for wireless mesh networks," Ad Hoc Netw., vol. 5, no. 6, pp. 719–743, Aug. 2007.

[37] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling for maximum throughput in multihop radio networks," IEEE Transactions on Automatic Control, vol. 37, no. 12, pp. 1936–1949, 1992.

[38] L. Tassiulas, "Linear complexity algorithms for maximum throughput in radionetworks and input queued switches," in Proceedings of IEEE Infocom, San Francisco, CA, 1998.

Ms.T.Kalaiselvi received MCA degree in Bharathiyar University,Coimbatore and she is currently pursuing M.E(CSE) in karpagam University,Coimbatore.

Prof. P.S.Balamurugan received B.E degree in 2003 from Bharathiyar University Coimbatore and ME in 2005 from Anna University. Currently he is pursuing his Doctoral degree at Anna University, Coimbatore.

Prof.K.Sudhakar received B.E degree in Computer Science & Engineering from Anna University Chennai and M.Tech (IT) from Anna University Coimbatore. Currently he is working in Hindusthan College of Engineering and Technology, Coimbatore.