Demo v(t) CSMA: A Link Scheduling Algorithm in Wireless Networks with Improved Delay Characteristics

Ahmed Elbagoury, Moustafa Youssef
Wireless Research Center, Alexandria University and E-JUST, Egypt
{ahmed.elbagoury, moustafa.youssef}@ejust.edu.eg

Dongyue Xue, Eylem Ekici
Department of Electrical and Computer Engineering, the Ohio State University, USA
{xued, ekici}@ece.osu.edu

ABSTRACT

Efficient scheduling of wireless resources has always been one of the most challenging tasks for wireless networks. To achieve throughput-optimality, traditional back-pressure algorithms calculate a maximal weight matching at each time slot. However, these algorithms need centralized scheduling with high complexity, and thus are not suitable for practical distributed implementations. A class of distributed queue-length-based CSMA algorithms have been proposed that achieve throughput optimality, which we refer to as regular throughput-optimal. These algorithms suffer from two problems: large delays, and temporal starvation. In this demo we demonstrate the operation of the v(t)-regulated CSMA algorithm that mitigates these two problems while provably retaining throughput optimality. The demo allows the participants to see the the performance advantage of v(t)-regulated CSMA over queue-length-based CSMA algorithms and change the different system parameters.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Distributed networks, Wireless communication*

General Terms

Algorithms, Experimentation, Performance

Keywords

Optimal wireless scheduling; CSMA-based scheduling; Delay-based scheduling

1. INTRODUCTION

Efficient scheduling of wireless resources has always been one of the most challenging tasks for wireless networks. To achieve throughput-optimality, traditional back-pressure algorithms calculate a maximal weight matching at each time slot. However, these algorithms need centralized scheduling with high complexity, and thus are not suitable for practical distributed implementations. A class of distributed queue-length-based CSMA algorithms have been proposed in literature [2] that achieve throughput optimality (CSMA algorithms). Although these CSMA algorithms have been proved to be throughput-optimal they suffer from the following problems: (1) Temporal starvation, where links usually undergo prolonged periods of inactivity followed by a prolonged period of activity, which leads to bursty service and undesirable jitter performance. (2) Undesirable delay performance. This behavior of regular throughput-optimal CSMA algorithms leads to the scheduling of links with short queues while there exist unscheduled links with longer queues in the network, resulting in long average packet delays.

2. GOALS

In this demo, we consider a time-slotted wireless network, where wireless nodes are implemented on TelosB motes. Each link is backlogged with a constant arrival rate λ. We demonstrate v(t)-regulated CSMA [1] which solves both the temporal starvation and undesirable delay performance problems ***while provably retaining throughput optimality***. In v(t)-regulated CSMA, only links with weights above a certain threshold qualify to be scheduled. Since link weights are chosen as increasing functions of packet queue lengths, only links with sufficiently large queue lengths are scheduled.

The demo will present two algorithms v(t)-regulated CSMA and QCSMA, which is a state-of-art scheduling algorithm [2]. The rich user interface will allow the attendees to compare the two algorithms and change the different system parameters.

2.1 Basic Idea of v(t)-regulated CSMA

The v(t)-regulated CSMA algorithm establishes a vector of thresholds $(\eta_l)_{l \in L}$, such that, if the packet queue of an active link l has a link weight below threshold η_l, the active link l relinquishes the wireless resource and becomes idle. Since link weights are chosen as increasing functions of packet queue lengths, only links with sufficiently large queue lengths are scheduled [1]. In comparison, under regular throughput-optimal CSMA algorithms, when a link occupies the channel, even if it has few packets (or even no packets) in its queue, it is highly likely that this link will remain scheduled for a considerably long period of time [2]. By always scheduling links with sufficiently large queue lengths, the v(t)-regulated CSMA algorithm potentially results in a reduction of packet delays in these scheduled queues, which

MobiCom'13, September 30–October 4, Miami, FL, USA.
ACM 978-1-4503-1999-7/13/09.
http://dx.doi.org/10.1145/2500423.2505298.

outweighs the increase in delay of the packets in the other unscheduled queues (which have fewer packets). In addition, under the proposed algorithm, the switch between schedules becomes more frequent than under regular throughput-optimal CSMA algorithms, mitigating the temporal starvation problem.

The v(t)-regulated CSMA algorithm starts by choosing a set of non-conflicting links $x(t)$. For any link $l \in x(t)$, if its neighboring links in the interference set N_l are not scheduled in the previous time slot, or do not have a link weight above the threshold η, then link l is scheduled for service ($\mu_l(t) = 1$) with a link activation probability of

$$p_l \triangleq \frac{e^{w_l(t)}}{1+e^{w_l(t)}}$$

Where $w_l(t)$ is the link weight chosen as an increasing function of queue length $Q_l(t)$. Therefore, the v(t)-regulated CSMA algorithm ensures that only links with sufficiently large queues (such that the corresponding link weights $w_l(t)$ are larger than the thresholds) can be scheduled.

2.2 Implementation

Figure 2 shows the the hardware testbed setup of the demo. Five TelosB motes are connected to one laptop through a USB hub. The demo implements a time-slotted wireless network, where all nodes are within the communication range of each other, leading to a 10-link topology shown in Figure 3. The interference model employed is the one-hop interference model.

The system works as follows: On each mote, each of the two algorithms (v(t)-regulated CSMA and Q-CSMA) alternately runs for a a time slot then the other algorithm runs. This makes the two algorithms experience almost the same channel conditions.

The application running on the laptop collects the information from all the motes at the end of each time slot. After each cycle, i.e. running one slot for each algorithm, the application calculates the different statistics for both algorithms and updates on the GUI. This is performed in parallel while performing the next cycles on the physical motes.

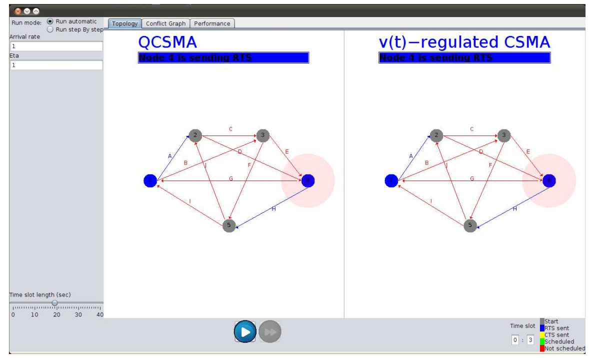
The application also shows all the control messages sent from both algorithms and their internal state (Figure 1(a)). Each node is represented by a circle that is colored according to the node state (sent RTS, sent CTS, scheduled or not scheduled) during each time slot. The program also allows the user to change different system parameters including the packets arrival rate and the of value of η used by v(t)-regulated CSMA.

The demo has two modes of operation: (a) Automatic mode: where the cycles are advanced automatically with a speed controlled using a slider. The user can pause the demo at any time and (b) Manual mode: where the user can single step the demo to see specific events.

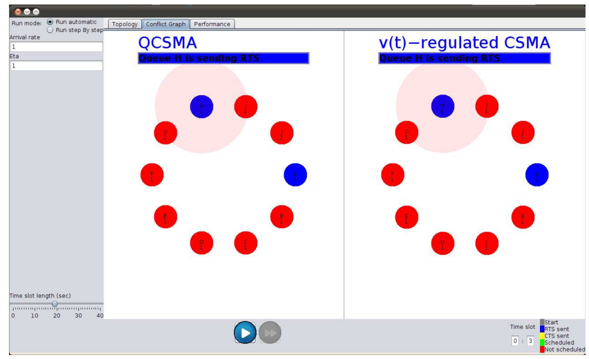
3. DEMO EQUIPMENT

The demo uses the following equipment:

- Five TelosB motes.
- USB cables used to connect the motes to the laptop.
- One laptop as an application server.
- Standard power supply for the equipment above.



(a) Network topology.



(b) Conflict graph.

(c) Performance metrics.

Figure 1: The applications GUI.

4. SPACE NEEDED

The demo requires minimum space for spreading the trusted and un-trusted sensors. Typical required space is 6×6 feet.

5. SETUP TIME

The system setup requires less than 30 minutes to connect the components.

6. DEMO VIDEO

A video showing the system in action and its characteristics can be found at:
`http://youtu.be/fetuUs_pCpY`.

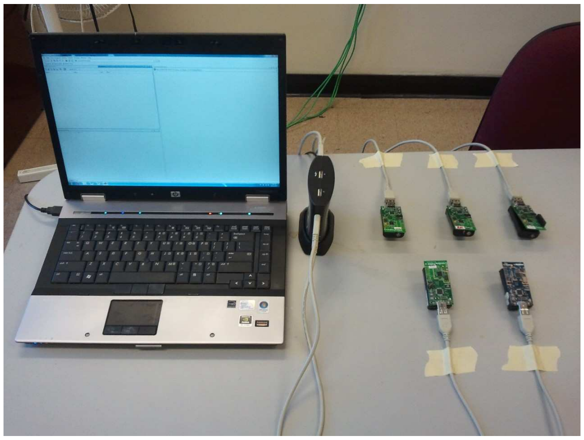


Figure 2: Hardware testbed setup.

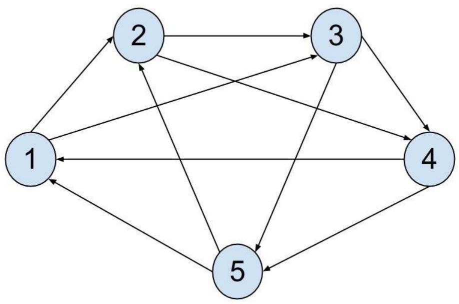


Figure 3: Network topology for implementation.

7. CONCLUSIONS

This demo allows the attendees to experience with wireless scheduling algorithms showing how to achieve better delay characteristics on actual motes. Attendees will have the opportunity to change the different system parameters and see their effect in real times.

8. REFERENCES

[1] Xue Dongyue and Ekici Eylem. On reducing delay and temporal starvation of queue-length-based CSMA algorithms. In *Communication, Control, and Computing (Allerton), 50th Annual Allerton Conference*, pages 754–761. IEEE, 2012.

[2] R. Srikant Jian Ni, Bo Tan. Q-CSMA: queue-length based CSMA/CA algorithms for achieving maximum throughput and low delay in wireless networks. In *INFOCOM'10 proceedings of the 29th conference on Information Communications*, pages 271–275. ACM, 2010.