Computer Networks 55 (2011) 343–355

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

On the queue-overflow probabilities of a class of distributed scheduling algorithms ☆,☆☆

Can Zhao, Xiaojun Lin *

School of Electrical and Computer Engineering, Electrical Engineering Building, Purdue University, West Lafayette, IN 47906, United States

ARTICLE INFO

Article history:
Received 22 September 2009
Received in revised form 15 July 2010
Accepted 25 August 2010
Available online 8 September 2010
Responsible Editor: R. LoCigno

Keywords:
Wireless networks
Quality of service
Large deviations
Scheduling algorithms

ABSTRACT

In this paper, we are interested in using large-deviations theory to characterize the asymptotic decay-rate of the queue-overflow probability for distributed wireless scheduling algorithms, as the overflow threshold approaches infinity. We consider ad hoc wireless networks where each link interferes with a given set of other links, and we focus on a distributed scheduling algorithm called Q-SCHED, which is introduced by Gupta et al. First, we derive a lower bound on the asymptotic decay rate of the queue-overflow probability for Q-SCHED. We then present an upper bound on the decay rate for all possible algorithms operating on the same network. Finally, using these bounds, we are able to conclude that, subject to a given constraint on the asymptotic decay rate of the queue-overflow probability, Q-SCHED can support a provable fraction of the offered loads achievable by any algorithms.

1. Introduction

Link scheduling is an important problem for ad hoc wireless networks. In wireless networks, the transmissions at neighboring links can interfere with each other. Hence, in order to maximize the capacity of the system, it is critical to schedule only a subset of non-interfering links at each time. There have been many studies on designing and analyzing scheduling algorithms for wireless network. A notable result is the well-known maximum-weight scheduling algorithm, which has been shown to be throughput-optimal, i.e., it can stabilize the network at the largest set of offered loads [2]. However, this algorithm is centralized and with high computational complexity. Therefore, many researchers have proposed low-complexity and distributed scheduling algorithms (see, e.g., [3–8]). Often, the goal is to be able to stabilize the network for a provable fraction of the capacity region. For example, the low-complexity algorithm in [5] has been shown to sustain close to 1/2 of the capacity region under the node-exclusive model.

To date most studies of wireless scheduling algorithms have mainly focused on stability. In other words, they ensure that the queues do not grow to infinity. Although stability is an important criterion, for many real-time applications stability is far from being sufficient. For example, when watching streaming video or listening to streaming audio, the user would expect that the delay of every packet can be upper bounded with high probability. As stability only ensures that the queue-length of each link remains finite, it cannot guarantee such type of stringent quality-of-service (QoS) requirements.

In certain cases, the probability of delay-violation can be mapped to the probability of queue overflow. Unfortunately, both problems have been known to be very difficult. First, the exact probability distribution is usually mathematically intractable. Hence, one often has to turn to asymptotic techniques, such as large-deviations. For wireline networks, many results have been obtained using large-deviations techniques [9], based on the assumption

☆ An earlier version of this paper has been presented at the 48th IEEE Conference on Decision and Control, Shanghai, China, December 2009 [1].

☆☆ This work has been partially supported by NSF grants CNS-0626703, CNS-0813000, CNS-0643145, CNS-0721484, CCF-0635202 and a grant from Purdue Research Foundation.

* Corresponding author. Tel.: +1 765 494 0626; fax: +1 765 494 3358.
E-mail address: linx@ecn.purdue.edu (X. Lin).

1389-1286/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2010.08.007

that the packet arrival process is known and the service rate of each link is time-invariant. However, in wireless networks, the service rate process is time-varying. Some progress has been made for the case when the scheduling decision is based only on the channel state, which means that the service rate process has known statistics [10,11]. However, for many wireless scheduling algorithms, even the statistics of the service rate process are unknown.

Recently, the queue-overflow probability for a number of queue-length based scheduling algorithms, for which the statistics of the service rate process are unknown, have been studied in [12–16] using sample-path large-deviations. In these works, the algorithms are centralized and are for a single cell. Further, the algorithms are deterministic in the sense that the scheduling decision is a deterministic function of the system state.

In this paper, we will develop techniques to estimate and control the QoS of distributed scheduling algorithms for ad hoc networks. We will focus on a random access algorithm for ad hoc wireless networks called Q-SCHED [5]. Note that due to the distributed and random nature of Q-SCHED, the techniques in prior works [12–16] do not apply directly. This is because the sample-path large-deviations techniques in these prior works require the cost of each sample-path to be known. Note that for the scheduling algorithm in these prior works, since the scheduling decision is a deterministic function of the system state, the statistics of the service rate given the system state are known. Hence, the cost of each sample-path can be written down explicitly. However, for the Q-SCHED algorithm, due to the randomness and distributive nature of the algorithm, the statistics of the service rate given the current system state are not precisely known. (In fact, only bounds on its statistics are known, as can be seen from Lemma 1.) Hence, we can not specify the cost of a sample-path explicitly, and thus, we can not use the methodologies from [12–16] directly.

As in [12–16] the questions that we are interested in are: (a) how to estimate the decay rate of the queue-overflow probability of this algorithm, and (b) given an overflow constraint, how to calculate the set of offered-load vectors that this algorithm can support. To answer these questions, we will first obtain a lower bound on the decay rate of the overflow probability for Q-SCHED. Then, based on this bound, we provide a lower bound on the set of offered-load vectors that this algorithm could support at a given queue-overflow constraint. To the best of our knowledge, this is the first work that characterizes the queue-overflow probability of distributed scheduling algorithms for ad hoc networks in a large-deviations setting. Finally, we show that subject to a given queue-overflow constraint, the offered-load supported by Q-SCHED is at least a provable fraction of the offered-load supported by any other algorithms.

2. System model

We use the model from [5]. We consider a wireless network of N nodes. Let V be the set of nodes, E be the set of directed links between nodes, and $G(V,E)$ be the directed connectivity graph of the network. Each link $l \in E$ interferes with a set of other links in E, which we denote as $\mathcal{E}_l$. We assume that if $k \in \mathcal{E}_l$ then $l \in \mathcal{E}_k$, i.e., the interference relationship is symmetric. We also let $l \in \mathcal{E}_l$, i.e., $\mathcal{E}_l = \{l\} \cup \{l' \in E : l' \text{ interferes with } l\}$. This interference set varies when different communication techniques are used. For example, for Bluetooth, we use the node-exclusive interference model, also known as the primary interference model or the one-hop interference model, where $\mathcal{E}_l$ is the set of all links that are connected to either end-point of l. In IEEE 802.11 WLAN, the interference set $\mathcal{E}_l$ will be the two-hop neighbors of l, including l.

We assume a slotted system. Let $a_l(n)$ denotes the number of packets that arrive at link l in time-slot n. We assume that for each l, $a_l(1)$, $a_l(2), \ldots$ are *i.i.d.* and $\lambda_l = \mathbf{E}[a_l(1)]$. Moreover, we assume that $a_l(n)$ is upper bounded by A_M for all $n > 0$ and all $l \in E$, i.e., $0 \leqslant a_l(n) < A_M$, which means that the number of arrival packets is finite in each time-slot.

Let $d_l(n)$ denote the number of packets that can be served by link l in time-slot n. Assume that the capacity of each link is a fixed number c_l. Let $s_l(n) = 1$ indicates that link l is scheduled in time-slot n, $s_l(n) = 0$ otherwise. Clearly, $d_l(n) = c_l s_l(n)$. We assume a single-hop system, i.e., packets served at link l immediately leave the system. Let $q_l(n)$ denote the backlog of link l in slot n, and $\vec{q}(n) = (q_1(n), q_2(n), \ldots, q_{|E|}(n))$. Then the evolution of each $q_l(n)$ is given by $q_l(n+1) = [q_l(n) + a_l(n) - d_l(n)]^+$, where $[\cdot]^+$ denote the projection to $[0,\infty)$. We also define:

$$A_i(n) \triangleq \sum_{l \in \mathcal{E}_i} \frac{a_l(n)}{c_l}, \quad D_i(n) \triangleq \sum_{l \in \mathcal{E}_i} \frac{d_l(n)}{c_l}. \tag{1}$$

These two variables are the total normalized arrivals and service, respectively, in each interference set at time slot n, and they will be used frequently in Section 3.

We consider the algorithm Q-SCHED that was introduced in [5]. In this algorithm, it is assumed that at the beginning of each time-slot every link l knows the queue-lengths of all links in its interference set $\mathcal{E}_l$ and also the queue-lengths of all links in the interference set $\mathcal{E}_k$ for every $k \in \mathcal{E}_l$. Each time slot is divided into two parts: a scheduling slot and a data transmission slot. Links that are chosen in the scheduling slot will transmit their packets in the data transmission slot. The scheduling slot is further divided into M minislots. At the beginning of each time-slot n, each link l first computes:

$$P_l(n) = \alpha \frac{\frac{q_l(n)}{c_l}}{\max_{i \in \mathcal{E}_l} \sum_{k \in \mathcal{E}_i} \frac{q_k(n)}{c_k}},$$

where $\alpha = \log(M)$. Then, each link l picks a backoff time $Y_l(n)$ from $\{1, 2, \ldots, M+1\}$ according to the following probabilities:

$$\mathbf{P}(Y_l(n) = M+1) = e^{-P_l(n)};$$
$$\mathbf{P}(Y_l(n) = m) = e^{-P_l(n)\frac{m-1}{M}} - e^{-P_l(n)\frac{m}{M}}, \quad m = 1, 2, \ldots, M.$$

A link that chooses backoff time $Y_l(n) = k \leqslant M$ will start transmission at the kth mini-slot unless it has already heard a transmission from one of its interfering links. If a link chooses a backoff time equal to $M+1$, it will not