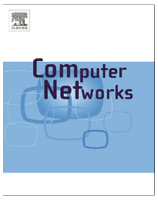
Computer Networks 57 (2013) 2193–2205

Contents lists available at SciVerse ScienceDirect

Computer Networks

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



On the exploitation of CDF based wireless scheduling

Udi Ben-Porat [a,*], Anat Bremler-Barr [b], Hanoch Levy [c]

[a] Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland
[b] Computer Science Dpt., Interdisciplinary Center, Herzliya, Israel
[c] Computer Science Dpt., Tel-Aviv University, Tel-Aviv, Israel

ARTICLE INFO

Article history:
Received 11 June 2012
Accepted 21 November 2012
Available online 18 March 2013

Keywords:
Wireless
CDF
Scheduling
Fairness
DDoS
Attack

ABSTRACT

Channel-aware scheduling strategies – such as the CDF scheduler (CS) algorithm – provide an effective mechanism for utilizing the channel data rate for improving throughput performance in wireless data networks by exploiting channel fluctuations. A highly desired property of such a scheduling strategy is that its algorithm is stable, in the sense that no user has incentive "cheating" the algorithm in order to increase his/hers channel share (on the account of others). Considering a *single user* we show that no such user can increase his/hers channel share by misreporting the channel capacity. In contrast, considering a *group of users*, we present a scheme by which coordination allows them to *gain permanent increase* in both their time slots share and in their throughput at the expense of others, by misreporting their rates. We show that for large populations consisting of regular and coordinated users in equal numbers, the ratio of allocated time slots between a coordinated and a regular user converges to $e - 1 \approx 1.7$. Our scheme targets the very fundamental principle of CS (as opposed to just attacking implementation aspects), which bases its scheduling decisions on the Cumulative Distribution Function (CDF) of the channel rates reported by users. Our scheme works both for the continuous channel spectrum and the discrete channel spectrum versions of the problem. Finally, we outline a modified CDF scheduler immune to such attacks.

1. Introduction

High-speed wireless networks are becoming increasingly common and along with that the strategy of *scheduling* the high-speed data – which is vital to the performance of modern wireless systems – has become the subject of active research. The modern wireless networks standards such as HSPDA [1] and EV-DO [2,3] allow a new generation of channel aware schedulers – such as the Proportional Fairness [4,5] and the CDF scheduler [6] – which improve throughput performance by exploiting channel fluctuations while maintaining fairness between the users.

The CDF scheduler (CS) makes scheduling decisions based on the Cumulative Distribution Function (CDF) functions of the users in such way that in every time slot, the scheduler selects for transmission the user whose rate is the least probable to become higher. An important property of this scheduler is that it statistically allocates all users an equal number of slots while smartly utilizing the knowledge of channel capacity to dynamically select at every moment the more attractive (higher capacity) users. A distinctive feature of this algorithm is that it allows prediction of the exact throughput for each user based on his/her[1] CDF alone, regardless of changes in the channel rate distribution of other users. These features and its simple notion of fairness (equal time share) make CS an attractive alternative to the Proportional Fairness Scheduler (PFS) [4]. Recent studies [7,8] revealed the vulnerability of PFS to delays/jitter and loss of throughput caused by

* Corresponding author. Tel.: +41 764056626.
E-mail addresses: ehudb@tik.ee.ethz.ch (U. Ben-Porat), bremler@idc.ac.il (A. Bremler-Barr), hanoch@post.tau.ac.il (H. Levy).

[1] From now on we use "he" and "his" to mean "he/she" and "his/her" for the sake of reading flow.

1389-1286/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.comnet.2012.11.022

malicious users by providing false channel capacity reports. In this paper the vulnerability of the CDF scheduler to threats of non-conforming opportunistic users as well as malicious users is investigated for the first time.

One of the main roles of a resource allocation mechanism is to ensure fairness of the allocation under the assumption that every user aims at increasing his own allocation. Furthermore, it is highly important that the scheduler will be resilient to users who may try to increase the resources allocated to them by not fully conforming with the protocol rules.

The objective of this work is to study this problem. Namely, whether a user, or a group of users, can mislead the CDF scheduler by providing false channel capacity reports and use it to increase the amount of resources allocated to them. Every modern channel-aware scheduler must allow a temporary state of unfairness in order to utilize a temporary exceptionally good channel condition of one of the users. Nevertheless, it is still expected that in the long run – that is, in the steady state – fairness is enforced. For example, in [7] the authors presented an attack on PFS in which a starved user can suddenly report an exceptionally good channel condition and temporarily be granted high priority, which causes other users to experience jitter. However, in the long run, the fairness that PFS is meant to ensure, is kept. In this work, we show that the CDF scheduler can be attacked by malicious and selfish users who gain a *permanent* advantage over users. That is, the time share fairness that the CDF scheduler is meant to ensure is not kept even in the steady state. We show that this is a fundamental weak point of the CDF scheduler regardless of its exact implementation.

To this end we show that the CDF algorithm is resilient against "attacks" produced by a *single user*. That is, a single user can increase neither the number of slots nor the bandwidth allocated to him by providing misleading information about his channel capacity. We then show, that nonetheless, a *group of coordinated users* which collaborate with each other can *increase both* the *number of slots* and the *bandwidth* allocated to *each of them*. That is, while the scheduler is designed to counter an independent selfish behavior of a single user, its design does not take into account the possibility of a coordinated group of users. The capacity announcement strategy used by the coordinated users is very simple and requires only knowledge of each other's capacity. We conduct the analysis of this strategy and derive its performance gains. The analysis is carried out both for the continuous rate distribution model (Section 3) and the discrete rate distribution model (Section 4). Our results show that the gain that such non-conforming users can achieve may be as high as 28% in a typical system configuration (30 users). Furthermore, the ratio between the slot allocation of a coordinated user and a regular user can reach $e - 1 \approx 1.7$. We further consider coordinated *malicious* users. These aim at reducing the performance the regular users, not caring about their own performance. We show that the channel share loss that the regular innocent users suffer can be as high as 48% in a typical system configuration.

The attack algorithm we show exploits the stochastic worst case traffic pattern of multiple users that can be applied to the system. This type of attack is demonstrated in the Reduction of Quality (RoQ) attacks papers [9–11]. RoQ attacks target the adaptation mechanisms by hindering the adaptive component from converging to steady-state. This is done by sending – from time to time – a very short burst of surge demand imitating many users and thus pushing the system into an overload condition. Using a similar technique, Kuzmanovic and Knightly [12] presented the Shrew Attack which is tailored and designed to exploit TCP's deterministic retransmission timeout mechanism. Another example of an attack exploiting the stochastic worst case is given in [13,14]. There it is shown that Weighted Fair Queueing (WFQ), a commonly deployed mechanism to protect traffic from DDoS attacks, is ineffective in an environment consisting of bursting applications such as the Web client application. The paper [15] shows attack on the SSL handshake, by requesting again and again hard SSL requests.

The rest of the paper is organized as follows: After model and preliminaries given in Section 2, Section 3 analyzes non-conformist users under the continuous rate distribution, and Section 4 does it under the discrete rate distribution. In Section 5 we analyze the loss for regular users by coordinated and malicious users in the practical discrete model. Finally, in Section 6 we outline a modified CDF scheduler immune to selfish or malicious behavior. Note that a short abstract of this work has been presented at [16].

2. Assumptions, model and preliminaries

In the scheduling models discussed in this work, time is slotted to slots $t = 1, 2, \dots$ and the possible channel rates are arbitrary and non negative. The rate at which user k can transmit at time slot t is given by $R_k(t)$. $R_k(t)$ is distributed according to random variable R_k associated with user k, and whose CDF is $F_{R_k}(r) = Pr[R_k \leqslant r]$. $R_k(t)$ is a stationary random process assumed to be independent of $R_k(t')$ for any $t \neq t'$ and of $R_j(t')$ for any $j \neq k$ and any t'.

At each slot t, each user k announces to the scheduler his actual value $R_k(t)$. The scheduler may compute the distribution $F_{R_k}(r)$ from the past reports of user k. Note that we demonstrate the vulnerability of CS without targeting a weak point in the inferring mechanism, therefore throughout the paper we assume the scheduler has the precise CDF functions of the channel rates reported by users. At time t, the scheduler can use both the studied $F_{R_k}(r)$, $k = 1, \dots, K$ and the current user rates $R_k(t)$, $k = 1, \dots, K$ to decide to which user to transmit at slot t. The rate at which the server will transmit to the selected user, say k, is $R_k(t)$.

3. The basic problem: dealing with continuous rate distributions

In this section we assume that all the channel rate distributions are continuous, that is the distribution functions do not contain mass values (i.e. $F_{R_k}(r)$ is differentiable and $Pr[R_k = x]$ equals zero for every x). Later, in Section 4, we will deal with discrete (and mixed) probability functions.