

Efficient Wireless Packet Scheduling in a Non-Cooperative Environment: Game Theoretic Analysis and Algorithms

Zhen Kong[‡] and Yu-Kwong Kwok^{†*}

[†]The University of Hong Kong, Pokfulam Road, Hong Kong

[‡]Wayne State University, Detroit, MI 48201, USA

Abstract

In many practical scenarios, wireless devices are autonomous and thus, may exhibit non-cooperative behaviors due to self-interests. For instance, a wireless cellular device may be programmed to report bogus channel information to gain resource allocation advantages. Such non-cooperative behaviors are highly probable as the device's software can be modified by the user. In this paper, we first analyze the impact of these rationally selfish behaviors on the performance of packet scheduling algorithms in time-slotted wireless networks. Using a mixed strategy game model, we show that the traditional maximum rate packet scheduling algorithm can cause non-cooperative devices to converge to highly inefficient Nash equilibria, in which the wireless channel resources are significantly wasted. By using a repeated game to enforce cooperation, we further propose a novel game theoretic algorithm that can lead to an efficient equilibrium.

Keywords: game theory, incentives, repeated game, Nash equilibrium, packet scheduling, wireless networks.

*Corresponding Author: Yu-Kwong Kwok (Email: ykwok@hku.hk)

1 Introduction

In a centralized infrastructure based wireless network, packet scheduling is a very important issue to manage the precious radio resource while satisfying users' Quality-of-Service (QoS) requirements. Specifically, in a traditional downlink packet scheduling protocol, wireless users are required to report their channel conditions, such as signal-to-noise-ratio (SNR) or maximal achievable data rate, to the scheduler located at the base station (BS) or access point (AP). Then the scheduler can select some users' packets and allocate radio resources, such as power and frequency bandwidth, to these users for transmission according to some scheduling policies, such as maximum rate (MR) [17] and proportional fairness (PF) [6].

Usually, these scheduling algorithms are based on the assumption that the wireless users in the system will cooperate with each other, comply with the predefined scheduling algorithm, and honestly report their real channel conditions to the scheduler [7]. Then the wireless user will accept the scheduling results passively, and it will not affect the scheduling policy employed by the scheduler at all. However, in many practical scenarios, the wireless users are autonomous and thus, may exhibit non-cooperative behaviors due to self-interests. For instance, in a public WLAN hot spot, individual users may attempt to deviate from the standard protocols or algorithms and behave in a rational but selfish manner so as to gain advantages in radio resource allocation, without regard to the overall system performance [16].

Indeed, the main reason that some wireless users exhibit selfish misbehavior is to gain extra benefit in terms of performance. Consequently, such selfish behaviors are usually highly related with its objective or utility. Currently, most wireless applications require a higher data rate (e.g., wireless video streaming). Thus, it is reasonable to assume that a selfish user's main interest is to increase its data rate as much as possible regardless of the whole system's performance.

In a wireless environment, a user's rate-based utility is influenced by many factors, such as the maximal achieved data rate determined by the modulation and coding schemes, and the bit error rate (BER) occurred at the physical layer during wireless transmission. In particular, in a typical wireless packet scheduling process, the scheduling policy is one of the major factors governing a user's data rate because the scheduler determines which user can be selected for transmission. Therefore, in a non-cooperative environment, a wireless user experiencing a bad channel condition might find out that if it honestly reports its channel condition to the scheduler, it may not be scheduled or just be assigned with a low data rate. Consequently, with a rationally selfish motivation, such user might report a bogus channel condition so as to get a higher probability to be scheduled for transmission or get a higher data rate. Though this non-cooperative behavior could increase the data rate for this selfish user, it may lead to inefficient resource utilization for the whole system.

Currently with the increased computing capability of smart phones and

the proliferation of open-source technologies, wireless devices can be easily programmed by users to behave in a selfish manner [1, 14]. On the other hand, with the popularity of wireless Internet services and availability of system virtualization on mobile devices [10], e.g., Xen on ARM architecture based mobile phones, unauthorized software downloaded from the Internet also has the potential to be malicious or selfish and consume more than enough radio resource at the expense of others. Thus, selfish misbehavior is truly a critical problem in wireless environment and whether the traditional packet scheduling algorithms are still effective in allocating resource in non-cooperative environment is in doubt and needs to be scrutinized carefully.

In this paper, we focus on the issue of downlink packet scheduling for selfish and rational wireless users in a non-cooperative TDMA wireless network, in which each user attempts to locally and selfishly choose its transmission rate so as to maximize its utility. By selfish, we mean the wireless users care essentially about their own benefits, without any consideration about the overall system data rate. By rational, we mean a user may not report its SNR or maximal achievable data rate solely based on its real channel condition. Instead, such a rationally selfish user will choose the strategy to maximize its own benefit. However, the strategy chosen by a user may also affect the performances of other users in the network through the related packet scheduling algorithm. Therefore, in practice, a rational user will evaluate its achieved data rate, average packet transmission success rate (PTSR), the probability of being scheduled for transmitting to determine the

best strategy for reporting a "best" channel condition to the scheduler, as so to improve its gain.

In the competitive resource allocation scenario considered in our study, there are several research questions need to be answered. First of all, what kind of strategy should a user choose in order to maximize its utility? Secondly, what is the consequential impact of this kind of selfish behavior on the system performance and its own benefit? Thirdly, if every user in the network selfishly and locally picks its utility-maximizing strategy, will there be a stable state at which no user can unilaterally improve its utility, e.g., a Nash equilibrium? Fourthly, if this NE leads to an inefficient radio resource utilization, are there any other methods that can be proposed to improve the performance?

Based on game theory [12], we formulate this competitive wireless resource allocation problem as a non-cooperative packet scheduling (NPS) game, and then design a novel mixed strategy game model, in which each player has a continuum of actions and seeks to choose its transmission rate to maximize its utility, which is defined as the expected data rate of this user. With our proposed utility function, we deduce the NE for this NPS game and determine that the corresponding NE strategy leads to an allocation that is Pareto inefficient, where the system average data rate is much lower than that in normal cooperative situation. Motivated by this observation, we further formulate a repeated game to enforce cooperation among wireless users, and propose an efficient algorithm to increase the data rate performance in

a non-cooperative environment.

The remainder of this paper is organized as follows. Section 2 discusses related work. In Section 3, we describe the system model, and discuss the impact of selfish behavior on system performance. The NPS game is analyzed in Section 4, where the Nash equilibrium and its property are also discussed. In Section 5, we show that the performance can be improved via a repeated game strategy. We then describe our proposed repeated game theoretic algorithm to optimize scheduling in Section 6. Finally, we give some concluding remarks in Section 7.

2 Related Work

Due to the increasing interests and capabilities for wireless users to deviate from legitimated protocols and algorithms so as to increase their own benefits, the research for the impact of selfish behavior has received considerable attention for various aspects of wireless networks in recent years [1], such as power allocation [3, 11], wireless routing for selfish devices [15, 18], MAC misbehavior in WLAN [2, 9, 16].

In terms of selfish behaviors at the physical layer, Meshkati et al. [11] have modeled power control for multicarrier CDMA systems as a non-cooperative game in which each user tries to selfishly maximize its overall utility. They have also proposed an iterative and distributed algorithm for reaching an equilibrium with a significant improvement in the total network utility. Etkin et al. [3] have studied a spectrum sharing problem in an unlicensed band with

multiple wireless providers and proposed a self-enforcing power allocation strategy to achieve efficient and fair spectrum sharing. As to routing at the network layer, Stephen et al. [15] focus on designing a routing algorithm with the feature of individually rationality, truthfulness and energy-efficiency for selfish devices in ad hoc networks based on an auction model. Wang and Li [18] have proposed a pricing mechanism to stimulate cooperation among all rational wireless devices by paying them for relaying data to and from the access point. Moreover, several approaches have been proposed for addressing selfish misbehavior at the MAC layer in wireless networks. For instance, Tan and Gutttag [16] have examined a kind of selfish behavior under the IEEE 802.11 distributed coordination function (DCF) mode, where selfish devices intentionally transmit at a lower data rate so as to achieve a higher channel share and a higher data rate. With the observation that the resulting Nash equilibrium (NE) [12] is undesirable and the wireless channel is inefficiently utilized, they have also shown that by guaranteeing the allocation of long-term shares of channel time to competing devices, the MAC protocol can enforce rational devices to efficiently use the shared medium, thereby improving the achieved data rates of all competing devices.

Besides, various approaches have been proposed in literatures to enforce cooperation or thwart selfish behaviors. For example, Kyasanur and Vaidya [9] have proposed some modifications on IEEE 802.11 DCF protocol to detect and penalize misbehaving devices, which purposely wait for smaller backoff intervals than well-behaved devices so as to obtain an unfair advan-

tage. By contrast, to avoid changing current protocols in tackling this kind of backoff misbehavior, Cagalj et al. [2] apply a dynamic game model to derive the Pareto-optimal NE point of operation of a network with multiple cheaters and show that the smart selfish devices can collectively determine this point so as to solve the selfish behavior problem in the system. Felegyhazi and Hubaux [4] have theoretically analyzed iterative power control in a spectrum sharing system based on repeated game theory; however, they didn't discuss how to detect selfish behavior in reality. In our previous work [7,8], we have also proposed a repeated game theoretic algorithm to enforce cooperation in non-cooperative scheduling system. However, the condition in the proposed algorithm is too strong, where each user will punish the selfish user by transmitting at the highest rate; thus, the whole system is also penalized too much at the same time. In this paper, we let a user punish the selfish user with a probability p when detecting the selfish behavior. In this situation, the selfish user will also be punished while the whole system throughput will be less affected.

3 Model

3.1 System Model

We consider a time-slotted system with one BS serving N wireless users. The BS transmits in slots with fixed duration, and only one user can be scheduled in one time slot. All users are assumed to be either static or moving slowly. We assume users always have packets to transmit. At the beginning of time

slot t , each user i measures the downlink channel condition, and returns, via a feedback channel, a measured data rate $r_i(t)$ to the BS. Based on this information, the BS then determines which user to transmit its packet in this time slot.

In our analysis, we assume adaptive modulation and ideal phase detection are used in a Rayleigh fading channel with bandwidth W , and no retransmission is considered. Typically, at time slot t , the maximal achievable symbol rate $c_i^a(t)$ (bit/symbol) for user i can be decided by its current channel SNR $\gamma_i(t)$ and the required BER P_{ber} , then $c_i^a(t)$ can be expressed as [13]:

$$c_i^a(t) = \log_2\left(1 + \frac{-1.5}{\ln(5 \cdot P_{ber})} \cdot \gamma_i(t)\right) \quad (1)$$

We assume that the symbol rate belongs to the symbol rate set $C = \{c_i : 0 \leq c_i \leq M\}$, i.e., $c_i^a(t) \in C$, where M can be interpreted as the maximal modulation mode. Then the corresponding maximal achievable data rate is given by:

$$r_i^a(t) = c_i^a(t) \cdot W \quad (2)$$

Correspondingly we define the set of data rate as $R = \{r_i : 0 \leq r_i \leq r^{max}\}$, where $r^{max} = M \cdot W$ and $r_i^a(t) \in R$.

We assume that each $r_i(t)$ is an independent and stationary random variable, and let the reported $r_i(t)$ also belong to the set R , i.e., $r_i(t) \in R$. Within this framework, in the literature there are several well-known packet scheduling algorithms, such as the MR algorithm [1], which is designed to

maximize the data rate at each slot, as well as the system aggregate data rate, by scheduling the user with the largest $r_i(t)$ for transmission, i.e.,

$$n^*(t) = \arg \max_i (r_i(t)) \quad (3)$$

where $n^*(t)$ is the user scheduled for transmission at time slot t .

In cooperative environments, these algorithms are optimal under their respective objectives with the assumption that the wireless users will comply with the algorithms and report their channel conditions to the BS honestly. However, they do not have any consideration of the impact of selfish behavior on their predefined performance and objectives. For example, if a user knows it will not be scheduled for transmission if it reports its real channel condition, it could tell a bogus $r_i(t)$ to the BS so as to increase its chance for transmission. This is possible in wireless environments because a wireless user usually has the ability to overhear the packets of other users in the vicinity of its radio transceiver, and then capture and sniff them [1]. In some cases, it's also not necessary for a selfish user to know the exact transmission rate of others. It can just estimate the scope of transmission rate and then choose to report the maximal possible rate r^{max} to maximize its scheduling opportunity. Then the corresponding scheduling results may be totally different with what will be realized in cooperative environment. In the following, we focus on the performance of the MR algorithm under a non-cooperative wireless environment.

3.2 Selfish Behavior and its Impact

In a cooperative situation, the feedback rate $r_i(t)$ from user i is just the maximum feasible rate $r_i^a(t)$ that the downlink can support under its current SNR $\gamma_i(t)$ to maintain a certain BER or PTSR requirement. On the other hand, in a non-cooperative environment, a selfish but rational device may report a bogus data rate $r_i(t) > r_i^a(t)$, so as to enhance its opportunity to be chosen for transmission as governed by the scheduling algorithm. Of course, a higher assigned data rate $r_i(t)$ may result in a higher BER or lower PTSR under the same SNR $\gamma_i(t)$, making its actual achievable rate lower than the intended value. Nevertheless, a rationally selfish consideration is that the smaller realized rate may be compensated by the rate improvement induced by the increased transmission probability. Consequently, a non-cooperative user could still have the incentive to report a different data rate to the BS so long as its own potential payoff can be increased.

Let $BER(r_i, r_i^a)$ and $\alpha(r_i, r_i^a)$ be the BER and PTSR respectively for device i when its maximum feasible rate is r_i^a and feedback data rate is r_i , where $r_i^a \in R$ and $r_i \in R$. When there are L bits in one packet, we have:

$$BER_i(r_i, r_i^a) = 0.2 \cdot \exp(\log_2(5 \cdot P_{ber}) \cdot \frac{2^{r_i^a} - 1}{2^{r_i} - 1}) \quad (4)$$

and

$$\alpha(r_i, r_i^a) = (1 - BER(r_i, r_i^a))^L \quad (5)$$

Since the probability that user i can transmit at time slot t is $Pr(r_i(t) >$

$r_j(t), \forall j \neq i$), its expected rate $R_i^{cop}(r_i(t))$ in cooperative situations and $R_i^{ncop}(r_i(t))$ in non-cooperative situations can be expressed as the followings, respectively,

$$R_i^{cop}(r_i(t)) = r_i^a \cdot \alpha(r_i^a, r_i^a) \cdot Pr(r_i(t) = r_i^a > r_j(t), \forall j \neq i) \quad (6)$$

$$R_i^{ncop}(r_i(t)) = r_i \cdot \alpha(r_i, r_i^a) \cdot Pr(r_i(t) = r_i > r_j(t), \forall j \neq i) \quad (7)$$

Thus, provided that $R_i^{ncop}(r_i(t))$ is larger than $R_i^{cop}(r_i(t))$, the selfish user i will have the intention to report a bogus rate so as to increase its own benefit.

To empirically quantify the performance impact of this kind of selfish behavior, we conduct simulations based on the above model. In the simulation, the entire system bandwidth is 10 MHz and the wireless channel is modeled as a frequency-flat Rayleigh fading channel with the maximum Doppler frequency of 10 Hz. Moreover, the additive white Gaussian noise (AWGN) power density is -80 dBW/Hz. All of N users are uniformly distributed in the cell. We set the time slot duration as 100ms, and let $c_i \in [0, 10]$. We also assume the packet length is 100 bits/packet and the maximum BER requirement is 10^{-5} . In the simulation, each test case has been repeated for 100 trials and then we plot the figures with average value.

Fig.1 shows the impact of selfish behavior on the performance of traditional MR algorithm, where $N = 8$ and $S = 1$. In a cooperative environment, all N users report their maximum feasible data rate to the BS. Whereas in

non-cooperative situation, user 1 (U1) deliberately fails to adhere to the algorithm and tries to misbehave by reporting bogus rate r^{max} to the BS, while all other users still report their maximum feasible rates honestly. This is a reasonable setting to simulate selfish behavior because r^{max} is larger than U1's true rate $r_1(t)$ such that U1 can get unfair share of the resource according to (7)(??). In this simulation, we can see in Fig.1 that U1 can achieve about 100% increase in average data rate by behaving selfishly. However, as a result of U1 using this selfish strategy, the system average data rate decreases by 30%. Therefore, the existence of selfish behavior in non-cooperative wireless networks could significantly degrade the data rate performance of packet scheduling algorithm. Most importantly, this observation still holds in the network with different sizes, such as those from $N = 2$ to $N = 20$ as shown in Fig.2, where S is also set to be 1.

4 Static Game

4.1 Game Theoretic Model

The above preliminary analysis clearly shows that a selfish user has an incentive to act in a non-cooperative way so as to improve its own data rate. Such selfish behavior can be modeled as a non-cooperative game in which each player has a continuum of actions [12]. In this game, we denote wireless users $\Omega = \{1, \dots, N\}$ as the set of players. When player i 's maximal feasible rate is r_i^a , its action is its reported rate r_i and the set of its possible actions is the interval from r_i^a to r^{max} , i.e., $A_i = \{r_i \mid r_i^a \leq r_i \leq r^{max}\}$. The action

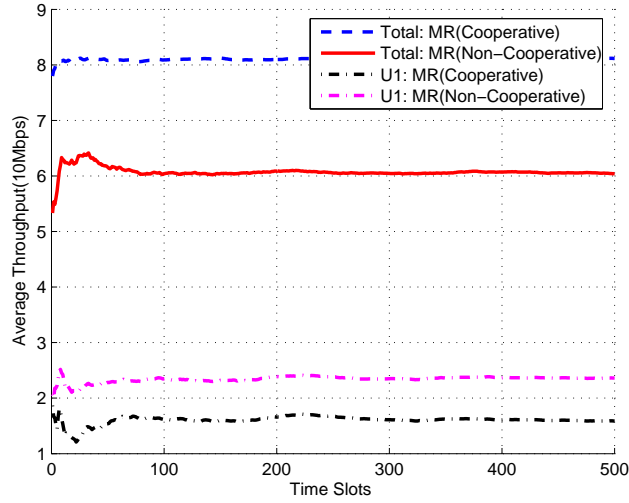


Figure 1: The impact of selfish behavior on the average data rate of packet scheduling algorithm.

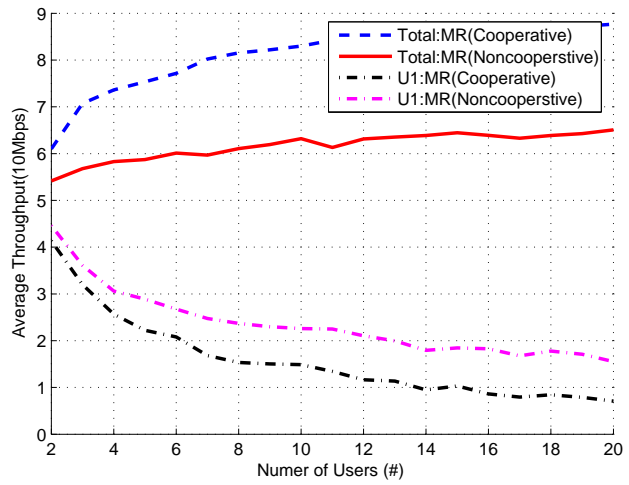


Figure 2: The impact of selfish behavior on the average data rate with different users in the system.

combination is denoted as $r = (r_1, r_2, \dots, r_N) \in A$, where $A = \times_{i \in \Omega} A_i$ is the Cartesian product of the N players' action profile. We identify each player's mixed strategy with a cumulative distribution function (CDF) $F_i(r_i)$ on this interval, for which $0 \leq F_i(r_i) \leq 1$ for every action r_i , and thus, the number $F_i(r_i)$ is the probability that player i 's data rate is at most r_i . Each player i 's preference is represented by the expected value of the data rate:

$$u_i(r) = \begin{cases} r_i \cdot \alpha(r_i, r_i^a) & r_i > r_j \\ \frac{1}{N} \cdot r_i \cdot \alpha(r_i, r_i^a) & r_i = r_j \\ 0 & r_i < r_j \end{cases} \quad (8)$$

where the parameter $\frac{1}{N}$ captures the situation that when these players have the same data rate, they will be chosen with the equal probability. We assume all user rates r_i 's are independent and identically distributed with the same CDF, and then the expected payoff can be expressed as:

$$\begin{aligned} U_i(r, F) &= r_i \cdot \alpha(r_i, r_i^a) \cdot Pr(r_i > r_j) + \frac{1}{N} \cdot r_i \cdot \alpha(r_i, r_i^a) \cdot Pr(r_i = r_j) \\ &\quad + 0 \cdot Pr(r_i < r_j) \\ &= r_i \cdot \alpha(r_i, r_i^a) \cdot Pr(r_i > r_j, \forall j \neq i) \\ &= r_i \cdot (F_j(r_i))^{N-1} \end{aligned} \quad (9)$$

The second term in the first line of (9) equals to zero because r_i is a continuous random variable, then $Pr(r_i = r_j)$ is 0. Within the above model, a NE can be defined as follows.

Definition 1. *An action combination $r^* \in A$ and the corresponding mixed strategy $F^* \in [0, 1]$ are said to achieve the state of Nash equilibrium if for*

every player $i \in \Omega$, we have:

$$U_i(r_i^*, r_{-i}^*; F_i^*, F_{-i}^*) \geq U_i(r'_i, r_{-i}^*; F_i^*, F_{-i}^*) \quad \forall r'_i \in A_i, F'_i \in [0, 1] \quad (10)$$

where r_{-i} and F_{-i} denote the actions and the corresponding mixed strategies chosen by everyone else other than i , r'_i is a reported rate other than Nash equilibrium rate r_i^* , $U_i(r_i, r_{-i}; F_i, F_{-i})$ is user i 's utility under rate r_i , r_{-i} and CDF F_i, F_{-i} .

4.2 Nash Equilibrium

It is known that for a game in which each player has finitely many actions, when a mixed strategy profile is a mixed strategy NE, the expected payoff to every action assigned with positive probability is the same. Correspondingly, as described in the Proposition 142.2 in [12], for the game in which each user has infinitely many actions $A_i = \{r_i \mid r_i^a \leq r_i \leq r^{max}\}$, the mixed strategy is determined by the probabilities assigned to sets of actions; and the expected payoff should be constant from r_i^a to r^{max} in NE. Moreover, because $F_j(r_i = r^{max}) = Pr(r_i(t) \leq r_i = r^{max}) = 1$, we have:

$$U_i(r, F) = \begin{cases} r_i \cdot \alpha(r_i, r_i^a) \cdot (F_j(r_i))^{N-1} = C & r_i^a \leq r_i \leq r^{max} \\ r^{max} \cdot \alpha(r^{max}, r_i^a) \cdot (F_j(r_i))^{N-1} = C & r_i = r^{max} \end{cases} \quad (11)$$

where C is a constant. Thus, for $0 \leq r_i^a \leq r_i \leq r^{max}$, we get:

$$F_j(r_i) = \left(\frac{r^{max} \cdot \alpha(r^{max}, r_i^a)}{r_i \cdot \alpha(r_i, r_i^a)} \right)^{\frac{1}{N-1}} \quad (12)$$

When the maximum achievable rate for player i is r_i^a , it will choose its feedback rate r_i according to the above NE strategy. Because in practice r_i is

a discrete random variable, then if $r_i^a \leq r^A \leq r_i \leq r^B \leq r^{max}$, the probability that it reports r_i to BS is expressed as:

$$Pr(r_i) = \frac{1}{2} \cdot (F_i(r^A) + F_i(r^B)) \quad (13)$$

where $r^A = \lfloor r_i \rfloor$ and $r^B = \lceil r_i \rceil$, $\lfloor x \rfloor$ express the largest integer smaller than x , $\lceil x \rceil$ express the least integer larger than x .

4.3 Performance Analysis

In a cooperative environment, the average data rate of the system is maximized by the MR algorithm. Furthermore, it is also *Pareto efficient* [12] because it is impossible to make one user get a higher rate without adversely affecting other users. While in a non-cooperative environment, selfish behavior breaks this property. With the same simulation environment in Section II, the system average data rate is plotted for a network with 8 users in Fig.3. We can see that in this 8-player game, the system average data rate in NE state is much smaller than that in a cooperative environment. This phenomenon follows readily from Fig.4, where the network size varies from 2 users to 20 users. Thus, the achieved NE data rate is *Pareto inefficient* [12], which is a common characteristic for a non-cooperative game.

5 Repeated Game

The main reason for the data rate decrease in a non-cooperative environment is that a selfish user intends to report a higher data rate. Thus, if the user

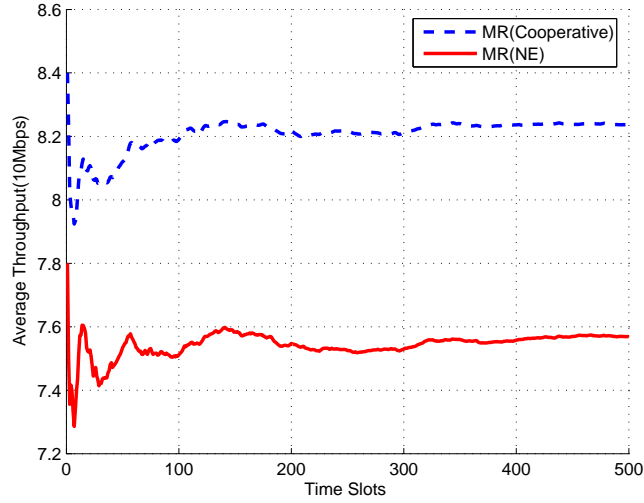


Figure 3: System average data rate comparison with 8 users under cooperative and NE conditions.

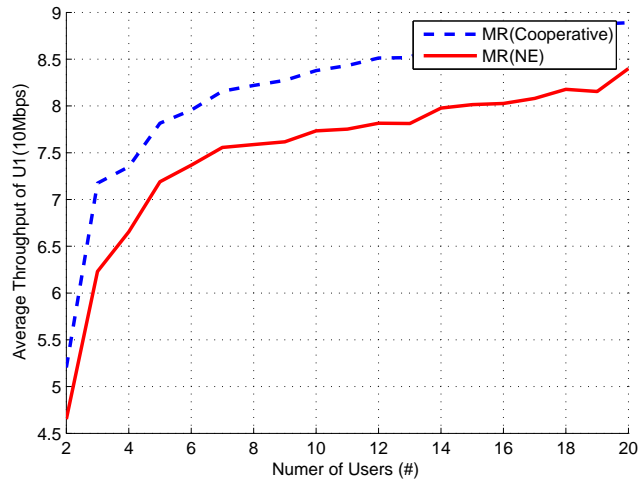


Figure 4: System average data rate comparison with different users.

in a bad channel condition gives up the competition and lets the others to transmit, the average data rate may be increased. However, in a non-

cooperative environment, a rational user has little incentive to give up its channel if there is no mechanism to enforce cooperation.

In this section, we propose a repeated game to enforce cooperation. We assume that the wireless users do not know the end of the game; hence we study the problem in an infinite repeated game model with discounting [12]. We show that cooperation (i.e., all users will report real r_i^a instead of others) can be enforced so as to improve the performance.

5.1 Repeated Game Model

We extend the NPS game as follows: we assume that the game is split up steps denoted by h . In each step, user $i \in \Omega$ adjusts the rate according to its strategy. Furthermore, let us define the discounted average utility in $H_i < +\infty$ time steps as:

$$\bar{U}_i(H_i) = (1 - \omega) \cdot \sum_{h=0}^{H_i} U_i(h) \cdot \omega^h \quad (14)$$

where $0 < \omega < 1$ is the discounting factor, which can be interpreted as the probability that the game ends in the next step. We assume ω is identical for all users in our model.

We have found that the selfish users are in an inefficient equilibrium when they all play NE strategy, whereas the maximal data rate can be achieved by using a cooperative strategy. From Folk Theorem [12], we know that in an infinitely repeated game, any feasible outcome that gives each player better payoff than the NE can be obtained. We can now determine the conditions

that enable the users to enforce cooperation, and prove that they can do better by applying a strategy called *Striker*, as detailed below.

5.2 Striker Strategy

Definition 2. *A wireless user i is said to employ the Striker strategy if it plays r_i^a in the first time step, and for any subsequent time steps, it plays:*

- r_i^a in the next time step if the other player j played r_j^a in the previous time step, or
- r_i^{max} with probability p_i for the next H_i time steps, if the other played anything else.

The punishment interval H_i defines the number of time steps for which a player punishes the selfish player [4]. To simplify our analysis, we assume that the overall channel conditions remain relatively unchanged. Then, r_i^a over each step in the repeated game is similar. However, our simulation results show that our analysis still holds in wireless fading situations. Consequently, cooperation can be enforced using the *Striker* strategy as formalized in the following proposition.

Proposition 1. *An efficient Nash equilibrium can be enforced by the Striker strategy.*

Proof. We set user i 's utility in terms of cooperative and non-cooperative situations as U_i^{COP} and U_i^{NCOP} , respectively. We consider the *Striker* strategy, and suppose user i adhere to it and choose r_i^a . If user $j \neq i$ uses

the same strategy, then the outcome is (U_i^{COP}, U_j^{COP}) in every step, so that it obtains the stream of payoffs, which gives a discounted average of $(1 - \omega) \cdot \sum_{h=0}^{H_i} U_i(h) \cdot \omega^h = (1 - \omega^{H_i+1}) \cdot U_i^{COP}$.

If user j adopts a rate r_j^X so as to get a larger utility $U_j^{NCOP} > U_i^{NCOP}$ in all subsequent steps, user i will chooses r^{max} with probability p_i since user j 's choice of r_j^X triggers the punishment. Then, in an attempt to maintain its own benefit, the selfish user j will have to choose r^{max} in every subsequent step with payoff U_j^M . Consequently, it obtains the stream of payoffs with discounted average utility:

$$\begin{aligned} & (1 - \omega) \cdot (U_j^{NCOP} + U_j^M + \omega \cdot (U_j^M) + \dots + \omega_i^H \cdot (U_j^M)) \\ & = (1 - \omega) \cdot U_j^{NCOP} + (1 - \omega^{H_i+1}) \cdot U_j^M \end{aligned} \quad (15)$$

Thus, user j can not increase its utility by deviating if and only if:

$$(1 - \omega) \cdot U_j^{NCOP} + (1 - \omega^{H_i+1}) \cdot U_j^M < (1 - \omega^{H_i+1}) \cdot U_j^{COP} \quad (16)$$

Thus,

$$\omega^{H_i+1} < 1 - (1 - \omega) \cdot \frac{U_j^{NCOP}}{U_j^{COP} - U_j^M} \quad (17)$$

The inequality cannot be fulfilled if the right side is negative, and therefore:

$$(1 - \omega) \cdot \frac{U_j^{NCOP}}{U_j^{COP} - U_j^M} < 1 \quad (18)$$

When this condition holds, since $\omega < 1$, we have:

$$H_i \geq \log_{\omega}((1 - \omega) \cdot \frac{U_j^{NCOP}}{U_j^{COP} - U_j^M}) - 1 \quad (19)$$

Because the user's utility is highly related to its average data rate, we rewrite the above H_i as

$$H_i \geq \log_{\omega}((1 - \omega) \cdot \frac{R_j^{NCOP}}{R_j^{COP} - R_j^M}) - 1 \quad (20)$$

Because when the selfish behavior is detected, each user i other than j will chooses r^{max} with probability p_i to punish user j . We can derive R_j^M as

$$R_j^M = \frac{1}{1 + \prod_{i \neq j} (p_i)} \cdot R_0^M \quad (21)$$

Here R_j^{NCOP} and R_j^{COP} are the average data rates in terms of non-cooperative and cooperative situations, respectively. R_0^M is the average data rate when all users in the network report r^{max} to BS, we simply estimated it by $R_0^M = \frac{1}{N} \sum_{t=1}^{t=T_H} r^{max} \cdot \alpha(r^{max}, r_j^a(t))$. Specifically, when all $p_i = 100\%$, the scheduler is just Round-Robin, and $R_j^M = \frac{1}{N} \cdot R_0^M$. Thus when the discounting factor is chosen as in (18) and punishment interval is set according to (20), the selfish user j will be forced to cooperate with user i such as to get higher payoff by *Striker* strategy. Correspondingly, an efficient NE is achieved.

□

6 Striker Algorithm

6.1 Implementation Issues

Proposition 1 indicates the feasibility of using a repeated game to enforce cooperation among selfish users and achieve efficient performance. However, to implement this *Striker* strategy in a practical wireless environment, there are still some problems to be tackled. For one thing, the basic assumption used in the game that a wireless user can know other users' actions and utilities instantaneously so as to detect the deviation, and decide its own strategy is impractical in reality [5]. Secondly, in a wireless environment, a user's utility and data rate will be different at different time, which makes the detection of selfish behavior and the decision of punishment interval less efficient. Furthermore, in a practical network, the deviation of users may be found by others with some different delays. Consequently, after the punishment interval H_i of user i is over, user i will directly switch to detection phase again. But if other users are still in punishment phase, it may lead user i to misjudge the punishment behavior of others as selfish behavior and punish them again, thereby making the network fluctuating and unstable.

To tackle the first two issues, we propose to detect the deviation by measuring wireless user's average data rate instead of its utility function because data rate is relatively easier to monitor and is highly related to our proposed utility function. The basic idea is that each user i sees whether its own average data rate \overline{R}_i is satisfied with a threshold condition, which is defined

as:

$$\frac{|V_i - \overline{R}_i|}{V_i} > \beta \quad (22)$$

where β is a predefined threshold coefficient and V_i is the threshold. If the threshold condition is satisfied, there may have some selfish behaviors in the network, and then this user will also play non-cooperatively to punish the deviation. To overcome the problems of misjudging the punishment behavior as selfish behavior and therefore making network unstable, we can let each user broadcast its punishment interval in the network. Thus, with such a broadcast message, it appears that all users can synchronize their punishment processes and the mentioned problem is solved in a practical manner efficiently.

However, the broadcasted messages also introduce additional overhead to the network. Furthermore, this method implies that there could have coordination or cooperation among users, which may be an invalid assumption in such a non-cooperative environment. Thus, we propose to use another method described below to solve it. Generally, after the punishment interval is over, the user i will directly switch to detection phase again. But if other users are still in punishment phase, this process may lead to misdetection of selfish behavior. Thus, we can change the *Striker* strategy slightly. When a user leaves the punishment phase and behaves cooperatively again, it will wait for extra steps H'_i and then enter detection phase so as to let other users have enough time to finish punishment and return to cooperation.

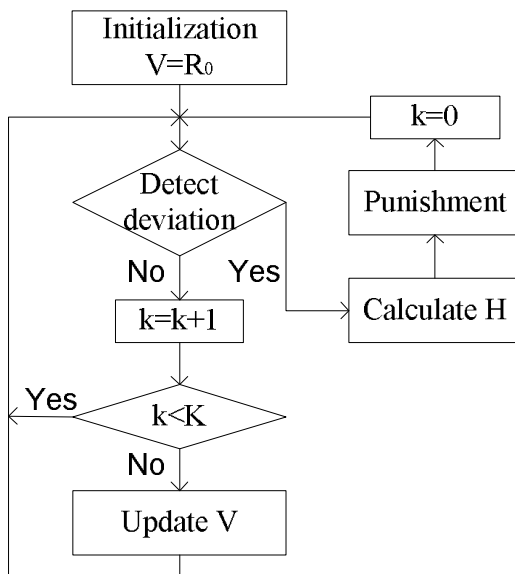


Figure 5: The block diagram of *Striker* algorithm.

6.2 Algorithm Design

Based on the above consideration, we design and develop a distributed *Striker* algorithm to implement the *Striker* strategy in practice so as to enforce cooperation among distributed non-cooperative users efficiently. As shown in the algorithm block diagram in Fig.5, this algorithm mainly consists of four parts: initialization, detection, punishment and update. The algorithm is explained in detail as follows.

- Initialization

In this stage, all users enter the network and begin to play the game. We initialize the deviation threshold V_i as the average data rate \overline{R}_i in the first step, i.e. $V_i = \overline{R}_i$. Let each step consist of T_H time slots, then $\overline{R}_i =$

$\frac{1}{T_H} \cdot \sum_{t=1}^{t=T_H} r_i(t)$. We also set step counter $k = 0$, which will be used for threshold updating later.

- Detection

In this stage, each user measures its average data rate and tests the threshold condition according to (22). The main reason we do not directly use the simple rule that whether \overline{R}_i is smaller than V_i to judge the deviation is that the wireless environment is dynamic, then \overline{R}_i may be varied at different steps. If the initialized or currently used V_i is not accurate, the misjudgement of deviation may occur.

Then in any following step, if a user finds its condition (22) is satisfied, it will begin to perform punishment action according to *Striker* strategy. By doing this, the short-term benefit of deviation will be eliminated by long-term punishment gradually.

- Punishment

When selfish deviation is detected by one user i , it will decide the punishment interval according to the expression (19). Because the average data rate is used instead of user's utility to detect deviation and determine punishment interval as analyzed above, we will rewrite H_i as:

$$H_i = \lceil \log_{\omega}(1 - (1 - \omega) \cdot \frac{R_j^{NCOP}}{R_j^{COP} - R_j^M}) - 1 \rceil \quad (23)$$

Here R_j^M is the average data rate when all users in the network report r^{max} to BS and can be estimated as (21). R_j^{NCOP} is the selfish user's average data rates in non-cooperative situation. Because in non-cooperative situation, the selfish user may get all transmission opportunities by reporting its rate as r^{max} , we can get $R_j^{NCOP} \leq r^{max}$. While in cooperative situation, a user's average rate is at most $\frac{r^{max}}{N}$. Thus, we can simply estimate H_i as

$$H_i = \lceil \log_{\omega} (1 - (1 - \omega) \cdot \frac{r^{max}}{\frac{r^{max}}{N} - R_j^M}) - 1 \rceil \quad (24)$$

With these parameters, a wireless user can locally decide its punishment interval. Then it can play r^{max} for next $H_i + H'_i$ time steps, where $H'_i < H_i$ is used to overcome the problem of misjudging as mentioned before.

- Update

The threshold V_i is initialized to be the average data rate at first step. If condition (22) is not satisfied, we can assume there is no deviation and the step counter k is increased by one. When there is no deviation detected for a period of continuous K steps, where K is a predefined constant, this may mean that cooperation is enforced so that we can update the threshold to improve current performance, or the current used threshold V_i is too small to check the deviation correctly. Thus, in any case, the process of updating threshold V_i is important and essential. Here, we use the average data rate in the previous K steps as the new threshold, i.e., $V_i = \frac{1}{T_H \cdot K} \sum_{t=1}^{T_H \cdot K} r_i(t)$.

6.3 Performance Analysis

To demonstrate the effect of *Striker* algorithm, we set up simulations with the same configuration as in Section 3. We first analyze the data rate performance under *Striker* strategies with different Striker probability p . In this test, $N = 20$ and $S = 1$. We also let $\beta = 20\%$, $K = 10$ and $T_H = 25$ here. In the beginning, all wireless users perform honestly, and the individual average data rate is about $8Mbps$, which is also used as a threshold to detect deviation. At time slot 100, U1 begins to deviate from the cooperative action by reporting a bogus rate r^{max} . As shown in Fig. 6, its average rate increases up to $14.8Mbps$ dramatically. This abnormal deviation is soon detected by others when they notice the threshold condition (22) is satisfied, and then punished by other users, which then make the data rate of this selfish user decrease dramatically. Then the data rate returns to the cooperative state. Thus, a more efficient equilibrium is achieved as well as the scheduling performance is optimized in a non-cooperative wireless network.

In the mean time, we notice that the punishment interval decreases with the Striker probability p . For example, when $p = 40\%$, the punishment interval is about 90 slots; while it becomes 60 slots when $p = 100\%$. This can be explained by the fact that when p is larger, more users would likely use *Striker* strategy to punish the selfish user, which then could more quickly realize that it has been penalized. As a result, the selfish user will return to the cooperation state soon. This observation is also consistent with our analysis in Proposition 1.

Another important observation is that a lower Striker probability p will make less decrease of system rate during the period of punishment, as shown in Fig.7. This is because the system aggregate rate will suffer more loss when there are more users who increase their rate to penalize the selfish user. However, when p is larger, the punishment becomes much severer and the system will return to cooperative situation more quickly. Thus, the Striker probability p can be used to strike a balance between punishment interval and system rate. If we want to punish the selfish behavior more greatly and force the selfish user to behave cooperatively soon, a higher p is preferable.

We further show the average individual throughput over the whole 500 time slots for 4 users to investigate the long-term fairness in Fig. 9, where only user 1 is selfish and $p = 70\%$. The results illustrate that the fairness performance is much worse in a non-cooperative situation because selfish user 1 consumes large part of radio resources by lying to the BS. With the introduction of *Striker* algorithm, the impact of selfish behavior is restricted and then the users are more likely to report their channel conditions honestly. Consequently, the users are scheduled in a fairer manner.

Besides throughput and fairness, we are also interested to see if the honest wireless users who experience dynamic channel changes may be misjudged as selfish users and be wrongly punished. To this end, we assume all $N = 20$ users are honest and measure the misjudging probability, which is defined as the relative frequency of occurrence of misjudgments in a long run of simulation trials with the same configuration as above. As shown in Fig. 9,

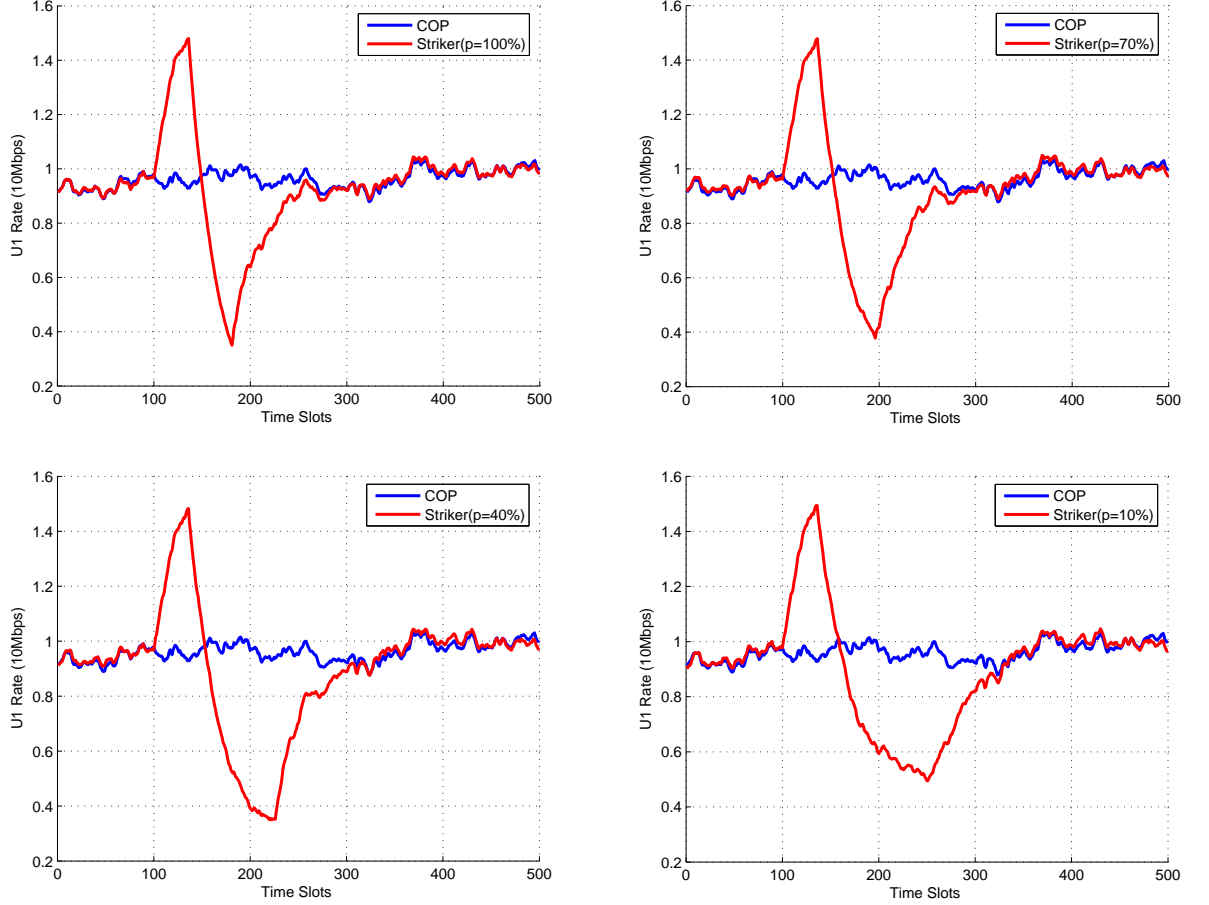


Figure 6: U1 date rate under Striker strategies with different Striker probability p . (a) $p = 100\%$; (b) $p = 70\%$; (c) $p = 40\%$; (d) $p = 10\%$;

the misjudging probability decreases with T_H , the time slots in one step; and when T_H is larger than 20, the misjudging probability is smaller than 1%. This is because T_H is related to the average window size to calculate average data rate \bar{R}_i and threshold V_i , then with a higher T_H , they can average out at more stable values, which are less affected by the dynamic changes in data rate. Thus, the misbehavior can be detected more accurately based

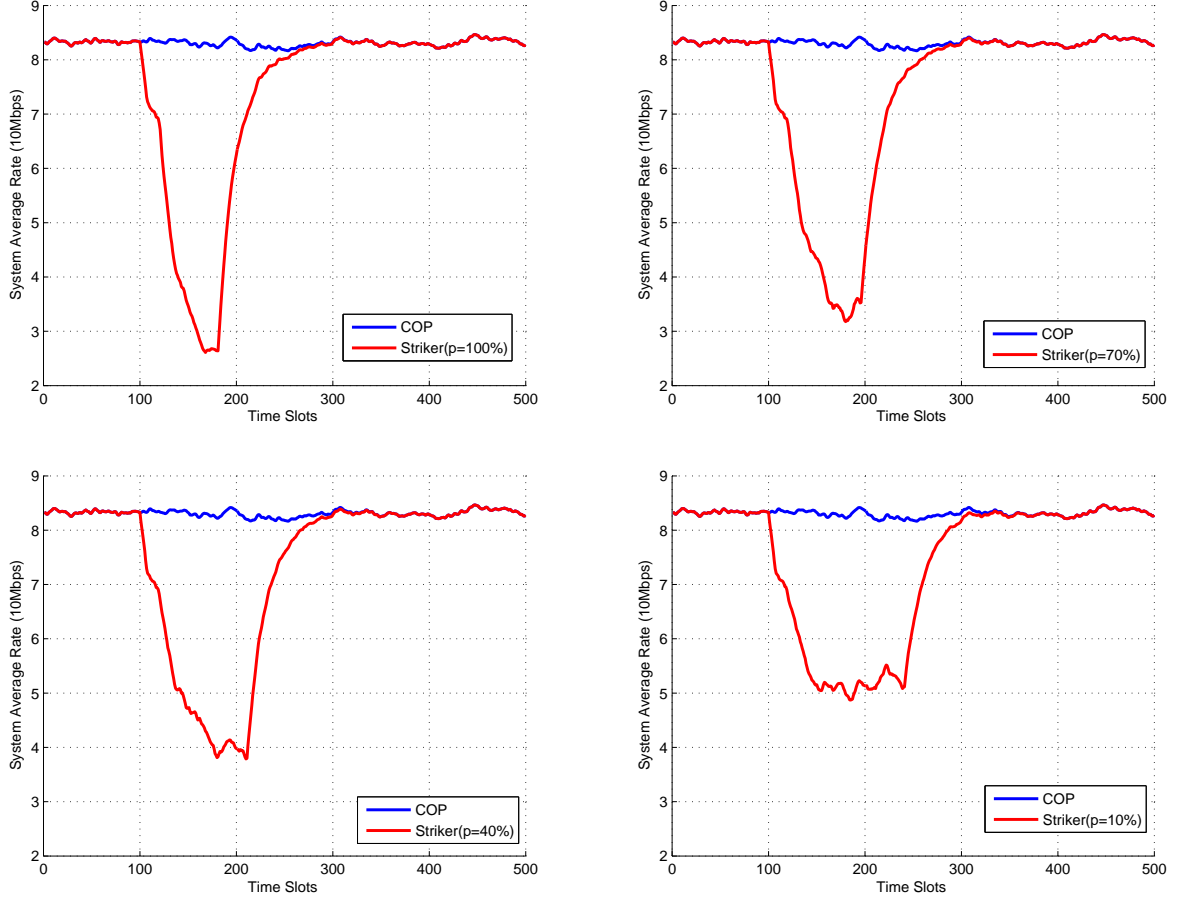


Figure 7: System data rate under Striker strategies with different Striker probability p . (a) $p = 100\%$; (b) $p = 70\%$; (c) $p = 40\%$; (d) $p = 10\%$;

on (22) and the misjudging probability decreases accordingly. Besides, the effect of β upon misjudging probability is not as distinct as T_H , whereas the misjudging probability is usually higher in the case of a smaller β because in this situation (22) is relatively more sensitive to temporal fluctuation of data rate, therefore it is prone to bring about misjudgment.

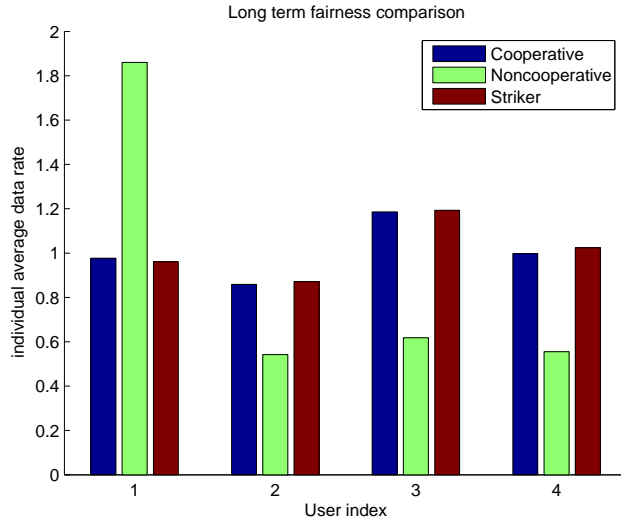


Figure 8: Fairness comparison.

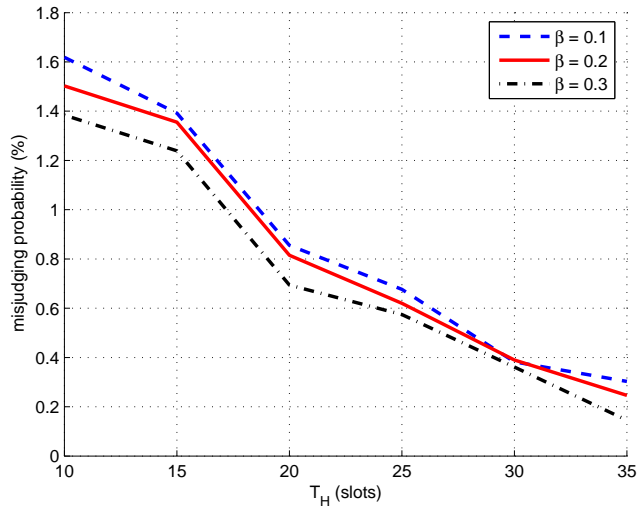


Figure 9: Misjudging probability.

7 Conclusions

In this paper we investigate the impact of rationally selfish behaviors on wireless packet scheduling algorithm in a non-cooperative wireless network.

We find that the existence of selfish behavior indeed makes the data rate performance of MR packet scheduling algorithm decrease greatly. Based on this observation, we further set up a novel mixed strategy NPS (non-cooperative packet scheduling) game to analyze this problem and derive a corresponding Nash equilibrium, in which the system average data rate is found to be significantly reduced. Then we further propose a distributed *Striker* algorithm based on a repeated game to enforce cooperation among users and achieve a more desirable Nash equilibrium, in which the data rate performance can be increased efficiently. Furthermore, users can be scheduled in a fairer manner with our proposed *Striker* algorithm. As part of our on-going work, we are now analyzing the impact of selfishness on fairness-oriented packet scheduling algorithms in a non-cooperative environment.

References

- [1] L. Buttyan and J. P. Hubaux, *Security and Cooperation in Wireless Networks*, Cambridge University Press, 2007.
- [2] M. Cagalj, S. Ganeriwal, I. Aad, and J. P. Hubaux, “On Selfish Behavior in CSMA/CA Networks,” *Proceedings of the 24th Annual IEEE Conference on Computer Communications (INFOCOM 2005)*, vol. 4, pp. 2513–2524, Miami, Florida, USA, March 2005.
- [3] R. Etkin, A. Parekh, and D. Tse, “Spectrum Sharing for Unlicensed Bands,” *IEEE Journal on Selected Areas in Communications*, vol. 25,

no. 3, pp. 517–528, Apr. 2007.

- [4] M. Felegyhazi and J. P. Hubaux, “Wireless Operators in Shared Spectrum,” *Proceedings of the 25th Annual IEEE Conference on Computer Communications (INFOCOM 2006)*, pp. 1–11, Barcelona, Catalunya, Spain, Apr. 2006.
- [5] Z. Han, C. Pandana, and K. J. R. Liu, “A Self-Learning Repeated Game Framework for Optimizing Packet Forward Networks,” *Proceedings of the 6th IEEE Wireless Communications and Networking Conference (WCNC2005)*, pp. 2131–2136, New Orleans, LA, USA, Mar. 2005.
- [6] A. Jalali, R. Padovani, and R. Pankai, “Data Throughput of CDMA-HDR a High Efficiency-High Data Rate Personal Communication Wireless System,” *Proceeding of the 51th IEEE Vehicular Technology Conference (VTC2000-Spring)*, vol. 3, pp.1854–1858, Tokyo, Japan, Jan. 2001.
- [7] Z. Kong, Y.-K. Kwok, and J. Wang, “On the Impact of Selfish Behaviors in Wireless Packet Scheduling,” *Proceedings of the IEEE International Conference on Communications (ICC 2008)*, Beijing, China, May 2008.
- [8] Z. Kong, Y. K. Kwok, and J. Wang, “Game Theoretic Packet Scheduling to Combat Non-Cooperativeness in Wireless Mesh Networks,” in *Proceedings of The Second IEEE International Workshop on Wireless Mesh and Ad Hoc Networks (WiMAN 2008)*, in conjunction with the 28th

- International Conference on Distributed Computing Systems (ICDCS 2008)*, Beijing, China, June 2008.
- [9] P. Kyasanur and N. H. Vaidya, “Selfish MAC Layer Misbehavior in Wireless Networks,” *IEEE Transactions on Mobile Computing*, vol. 4, pp. 502-516, Sept.-Oct. 2005.
- [10] S-M. Lee, S-B. Suh, B. Jeong, S. Mo, B. M. Jung, J-H. Yoo, J-M. Ryu and D-H. Lee, “Fine-Grained I/O Access Control of the Mobile Devices Based on the Xen Architecture,” *Proceedings of the 15th ACM International Conference on Mobile Computing and Networking (MobiCom 2009)*, Beijing, China, Sept. 2009.
- [11] F. Meshkati, M. Chiang, H. V. Poor, and S. C. Schwartz, “A Game-Theoretic Approach to Energy-Efficient Power Control in Multicarrier CDMA Systems,” *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 1115–1129, June 2006.
- [12] M. J. Osborne, *An Introduction to Game Theory*, New York: Oxford University Press, 2004.
- [13] X. Qiu and K. Chawla, “On the Performance of Adaptive Modulation in Cellular Systems,” *IEEE Transactions on Communications*, vol. 47, no. 6, pp. 884-895, June 1999.
- [14] S. Radosavac, J. S. Baras, and I. Koutsopoulos, “A Framework for MAC Protocol Misbehavior Detection in Wireless Networks,” *Proceedings of*

- the 4th ACM workshop on Wireless security (ACM WiSe'05)*, pp. 33–42, Cologne, Germany, Sept. 2005.
- [15] E. Stephan, R. Giovanni, and S. Paolo, “The COMMIT Protocol for Truthful and Cost-Efficient Routing in Ad Hoc Networks with Selfish Nodes,” *IEEE Transactions on Mobile Computing*, vol. 8, pp. 19–33, Jan. 2008.
- [16] G. Tan and J. Gutttag, “The 802.11 MAC Protocol Leads to Inefficient Equilibria,” *Proceedings of the 24th Annual IEEE Conference on Computer Communications (INFOCOM 2005)*, vol. 1, pp.1-11, Miami, Florida, USA, March 2005.
- [17] B. S. Tsybakov, “File Transmission Over Wireless Fast Fading Downlink,” *IEEE Transactions on Information Theory*, vol. 48, pp. 2323-2337, Aug. 2002.
- [18] W. Wang and X. Y. Li, “Low-cost Routing in Selfish and Rational Wireless Ad Hoc Networks,” *IEEE Transactions on Mobile Computing*, vol. 5, pp. 596–607, May 2006.