An algorithm for improved proportional-fair utility for vehicular users

Thi Thuy Nga Nguyen, Olivier Brun, Balakrishna Prabhu

An algorithm for improved proportional-fair utility for vehicular users*

Thi Thuy Nga Nguyen[1,2], Olivier Brun[2], and Balakrishna J. Prabhu[2]

[1] Continental Digital Service in France,Toulouse, France
[2] LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
Thi.Thuy.Nga.Nguyen@continental-corporation.com
{brun,Balakrishna.Prabhu}@laas.fr

Abstract. The Proportional Fair (PF) scheduler currently implemented in cellular networks is optimal when the channel conditions are stationary. Using measurements, a recent work shows that the conditions for moving cars can be non stationary and vary along the route. Based on these observations, the authors of [8] devise an algorithm called $(\text{PF})^2\text{S}$ that exploits Signal-to-Noise Ratio (SNR) maps and rate predictions to improve the utility over the standard PF algorithm. We propose an algorithm which gives a better prediction of the future rate allocation and has a better utility compared to both the PF and $(\text{PF})^2\text{S}$ algorithms. The proposed algorithm employs projected gradient on a relaxed version of the problem to predict the future allocations. Simulation results show that non negligible gains in utility over $(\text{PF})^2\text{S}$ can be achieved by this algorithm.

Keywords: Scheduling, proportional fairness, projected gradient

1 Introduction

Connected vehicles are expected to have more stringent QoS requirements than that of typical mobile users of today. These will include a higher data rate, reduced latency, and very low outage. In order to meet these requirements, 5G and future cellular technologies will operate in a range of spectrum with sufficiently large bandwidth to accommodate the requests. While increasing the available wireless resources is indeed necessary, it is also important to design resource sharing algorithms that can maximize the utilization of these resources.

The scheduling algorithm in 3G systems is based on the Proportional Fair (PF) scheduler which is obtained as the solution of a utility maximization problem [6]. It was designed with the objective of being fair to users with different channel conditions. Indeed, due to the various wireless effects such as shadowing and fading, data rates of mobile users can vary widely within a zone. By giving the channel to the user with the highest current data rate, the mobile operator

* This work was partially funded by a contract with Continental Digital Services France.

will be unfair to users who find themselves in unfavorable channel conditions over long periods of time. The PF scheduler alleviates this problem by allocating the channel to the user with the highest ratio of current data rate to the previous throughput[3] (we will call this ratio the index of a user). Thus, users with comparatively low throughput are assigned a higher priority even when they are in worse channel conditions. The performance and design of the proportional fair scheduler has been widely investigated for wireless networks [2, 11].

Proportional Fair scheduling algorithms for wireless networks have been widely investigated in various settings [3, 12, 13, 4]. Most of the literature is based on the assumption that users experience stationary channel conditions. This was partly motivated by the fact that a simple index-based allocation algorithm had been shown to be optimal for stationary channels [7]. This assumptions is not necessarily true for vehicular traffic moving along a given path, as was shown in [8] using SNR maps obtained by measurements. Indeed, as a car move along a road, the SNR improves as it moves closer to a base station and then worsens as it moves away. This implies that SNR is not stationary since its mean varies with time. The long sojourn time assumption is also not realistic for mobile users, since vehicles pass the coverage range of the base station in more or less than one minute. Moreover, if the trajectory of a car is known, one can obtain good statistical predictions on the SNR that will be experienced by the car. Knowing the future channel conditions, one can hope to design scheduling algorithms that can obtain a higher utility compared to the PF algorithm. This improvement in utility was not possible for stationary channel conditions as knowing the current position in the trajectory did not bring any new information on the future data rates.

Several variations of opportunistic scheduling algorithms have been studied recently using future information [1], [8]. In [1], the authors use a different kind of proportional fair objective function to ours, and they use future information by looking at channel state of users in a few small time-slots. Different from their approach, we do not look at the predicted channel state in few time slots which may be different between users and difficult to predict correctly due to fast fading. Instead, we base our allocation on average rate the user will experience during the time interval this user stays inside the coverage range of the base station. The average rate in the future is easier to estimate, has a lower error in prediction, and gives useful information for how much data the user can receive. In [8], the authors investigate the same objective function to ours, and propose an improved scheduling algorithm, called $(PF)^2S$ (to be explained later), based on prediction of future rates. In brief, in every time slot the channel is allocated to the user with highest index which is the ratio of the current data rate and the total throughput. The main difference with the PF algorithm is that the total throughput includes the future predicted throughput whereas in the PF algorithm only the past throughput was used. It was shown that this new

[3] The throughput is different from the data rate. While the latter is potential rate at which a user can be served, the former can be smaller since in some slots a user may not be served due to the presence of other users.

index led to improved utility compared to the PF algorithm in non-stationary environments.

The $(\mathrm{PF})^2\mathrm{S}$ algorithm would have been optimal if the future throughput could be computed optimally. For this, one requires the exact knowledge of all the channel conditions in the future. Since this information is not available, $(\mathrm{PS})^2\mathrm{S}$ uses allocations like round-robin (and some other heuristics) for the computation of the future throughput. That is, it is assumed that cars would be served in a round-robin fashion in future time-slots. Clearly, this may not be true since cars are actually served using the index policy of $(\mathrm{PF})^2\mathrm{S}$. Nevertheless, it was shown that even without the knowledge of the future throughput, improvements in the utility were made compared PF.

1.1 Contributions

We present a heuristic algorithm for non stationary channels that improves the total utility of users compared to the PF and the $(\mathrm{PF})^2\mathrm{S}$ algorithms. The original utility maximization problem being computationally complex, we employ three techniques to obtain a lower complexity heuristic: *(i)* we relax the integer constraints of the original problem; *(ii)*, we shorten the time horizon over which the problem is solved; and *(iii)* we compute the solution over macroscopic time slots instead of microscopic ones.

The relaxation turns the problem into a convex one and allows for its efficient resolution. Shortening of the time horizon and solving over macroscopic slots reduces the number of variables in the problem and decreases the computation time.

Simulation results for a single base station scenario show that improvements are achieved in different scenarios with vehicles moving at either equal or different speeds.

1.2 Organisation

In Section 2, we state the assumptions, define the objective function, and give some background on PF and $(\mathrm{PF})^2\mathrm{S}$ algorithms. In Section 3, we present the our heuristic for improving the utility based on estimations of future average data rate. Section 4 contains the numerical results for scenarios with homogeneous as well as heterogeneous vehicles. Finally, we end the paper in Section 5 with a few open problems.

2 Problem formulation

We consider a single base station (BS) that covers a linear stretch of road of length L (for example $L = 1$ km) along which vehicles move in one direction. The users enter the coverage range of that BS at left edge, move at different velocities, and leave when they arrive at the right edge. Every $\delta = 2$ ms the

BS[4] has to decide which user to serve. Let v be the velocity of the users. The coverage range can be chopped into N small spatial slots with each spatial slot corresponding to distance moved in δ. (See Figure 1). We also define a big time slot of length $\Delta = 1$ s in which a new car enters the coverage range from the left with some probability.

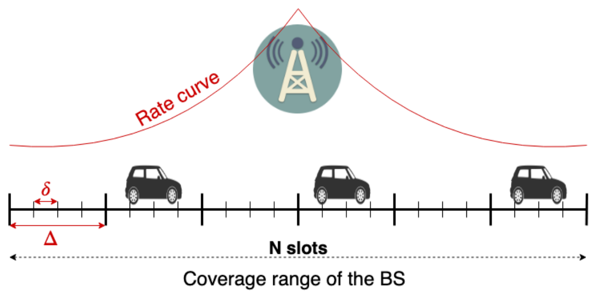

Fig. 1. Coverage model for moving vehicles

We shall assume that the data rate received by a user in spatial-slot s depends on the distance between the BS and s. The data rate depends upon the SNR which itself can vary along the road. In our numerical experiments, we assume that the data rate decays exponentially as shown in Fig. 1. The scheduling algorithm we propose does not require this assumption to work.

Denote by T (T should be large compared to δ) the time horizon over which the scheduling decisions are made. Let K be number of total users who pass by the BS in T. Let $r_{i,j}$ be the feasible data rate of user i in time-slot j, and let $\alpha_{i,j} \in \{0, 1\}$ denote whether user i is served in slot j or not. Denote by $\alpha = \{\alpha_{i,j}\}_{i,j}$ allocation matrix, and $r = \{r_{i,j}\}_{i,j}$ rate matrix. Our objective is to achieve the proportional-fairness between users, which is described by the following optimization problem (see, e.g., [2, 8]):

$$(I) \begin{cases} \max O(\alpha) = \sum_{i=1}^{K} \log\left(\sum_{j=1}^{T} \alpha_{ij} r_{ij}\right) \\ \text{subject to } \sum_{i=1}^{K} \alpha_{ij} = 1,\ j = 1, \ldots, T \\ \text{and } \alpha_{ij} \in \{0, 1\} \end{cases}$$

Proportional fairness is a resource allocation algorithm which is a special case of the general framework of utility maximization problem [10]. It was first introduced for wired in [6] and has been applied in various wireless settings [2, 11]. In brief, each user stays in the system for a certain duration during which it receives certain throughput which depends on the allocation decisions of the scheduler. This allocation can either be continuous in time or in blocks. The

[4] For simplicity, the BS will also be called the scheduler. In practice, the scheduler could be situated further inside the radio network.

utility of the user is defined as a function its throughput. The network utility maximization problem is to assign bandwidth or to schedule users in such a way so as to maximize the sum of the utilities of all the users. For proportional fairness, the utility is assumed to be logarithmic.

2.1 PF-EXP

Remark that the above problem is a discrete problem, which is not convex yet. Even though the number of options is finite, it is NP-hard to find the optimal solution (see, e.g., [8]). Nevertheless, a simple heuristic, called PF-EXP [7], can be obtained by assuming that the number of users is fixed and that the data rates $r_{i,j}$ are time stationary and ergodic, that is, there is no correlation between $r_{i,j}$ and $r_{i,j+1}$.

The PF-EXP algorithm chooses the user with the highest ratio of the current rate to the observed throughput, that is it chooses the user who is

$$\arg\max_{i} \frac{r_{i,j}}{\theta_{i,j}}, \tag{1}$$

where $\theta_{i,j} = \beta_j \theta_{i,j-1} + (1 - \beta_j)\alpha_{i,j} r_{i,j}$ the weighted sum allocation of user i until slot j. In the long-run when T goes to ∞, this algorithm was shown to be optimal for stationary and ergodic channel and fixed number of users [7].

The stationarity assumption is not necessarily true for road traffic when all users always move along a given path. As can be seen in Fig. 1, when users move along the line from left to right, their rate first increases and then decreases. Knowing the position of a vehicle, we can predict the future rate which is not possible when the rate process is stationary and ergodic. Thus, the PF-EXP algorithm need not be optimal for a rate process observed by vehicles.

2.2 Predictive Finite-horizon PF Scheduling ($(\mathbf{PF})^2\mathbf{S}$)

In [8], a modified PF algorithm based on predicted future rate was proposed. This algorithms works as follows:

- Predict future rate $\hat{r}_{i,j}$ of cars in every future slot.
- Estimate future allocation $\hat{\alpha}$. They propose three ways to chose and update this $\hat{\alpha}$: (i) *round-robin*, (ii) *blind search* and (iii) *local search*. In our simulations, we use only round-robin because blind search is similar to PF-EXP. Finally, in each time slot, local search iteratively computes until T and then allocates according to (2), making it a computationally expensive method. It is shown in [8] for local search to be effective, the prediction error has to be low for the whole horizon. If the prediction error is high then local search can be worse than round robin.
- For each time slot j, compute:

$$M_{i,j} = \frac{r_{i,j}}{\sum_{t=1}^{j-1} \alpha_{i,t} r_{i,t} + \hat{\alpha}_{i,j} r_{i,j} + \sum_{t=j+1}^{T} \hat{\alpha}_{i,t} \hat{r}_{i,t}}. \tag{2}$$

– Choose $i^* = \arg\max_{i\in\{1,2,\dots K\}} M_{i,j}$.

The index $M_{i,j}$ looks similar to that of the PF-EXP algorithm but includes the future allocation. It is related to the gradient of the utility function in (I). It can be shown that if we can predict correctly $\hat{\alpha}$, then the optimal solution can be obtained.

Proposition 1 *If there exist α^* satisfying: $\alpha^*_{i^*,j} = 1$ and $\alpha^*_{i,j} = 0, \forall i \neq i^*_j$, where i^*_j is*

$$\arg \max_{i\in\{1,2,\dots K\}} \frac{r_{i,j}}{\sum_{t=1}^{j-1} \alpha^*_{i,t} r_{i,t} + \alpha^*_{i,j} r_{i,j} + \sum_{t=j+1} \alpha^*_{i,t} r_{i,t}}. \tag{3}$$

then α^* is the optimal solution of (I).

The condition (3) need not always be satisfied. However, when it is, it is sufficient for α^* to be optimal solution of problem (I) as stated in Proposition 1.

For this approach of using the future rates to be efficient, one needs a good estimate of the optimal future throughput. This is not easy because the future throughput is computed from the optimal solution which itself is hard to compute.

3 Projected gradient approach

We now present a method for estimating the future throughput which improves the utility compared to the (PF)^{2}S algorithm. The main difficulty of solving (I) is that the problem is discrete. To simplify it, we relax the integer constraints to get a convex optimization problem, which is called relaxed problem as described in (II). The relaxed problem can be solved efficiently using projected gradient based on the projection on simplex formula given in [5].

Consider the following relaxed problem:

$$(II) \begin{cases} \max O(\alpha) = \sum_{i=1}^{K} \log \left(\sum_{j=1}^{T} \alpha_{ij} r_{ij} \right) \\ \text{subject to } \sum_{i=1}^{K} \alpha_{ij} = 1, j = 1, \dots, T \\ \text{and } \alpha_{ij} \in [0,1] \end{cases}$$

which is very similar to the original problem except that α_{ij} can be non-integer in $[0,1]$. Below, we describe the projected gradient formula for the above relaxed problem.

Denote by $D = \{\alpha \in [0,1]^{K\times T}, \sum_{i=1}^{K} \alpha_{ij} = 1 \, \forall j = 1,2,...,T\}$ the feasible set of the relaxed problem. D is not a simplex yet, therefore we cannot apply directly the algorithm in [5] and we need to modify it.

Denote Π_D is the projection on D (see Appendix A). The projected gradient algorithm follows the below steps:

– Initialize $\alpha^0 \in D$ arbitrary.

- From $n = 1, 2, 3, ...$ compute: $\alpha^{n+1} = \Pi_D(\alpha^n + \epsilon_n \nabla O(\alpha^n))$, where $\epsilon_n \in (0, 1)$ is step size at step n.
- Until α^n converges. In our numerical examples, we limited the number of iterations to 20.

Denote by $\tilde{\nabla} O(\alpha) = \Pi_D(\alpha + \epsilon \nabla O(\alpha)) - \alpha$ with the step size $\epsilon \in (0, 1)$ small enough. If there is a positive step size ϵ such that the following condition happens, we have an optimal guarantee of this algorithm, as described the following proposition.

Proposition 2 *If $\alpha^* \in D$ and $\tilde{\nabla} O(\alpha^*) = 0$ then α^* is the optimal value of the relaxed problem.*

3.1 Projected gradient short term objective algorithm (STO1)

Based on the above relaxation, we propose a heuristic which computes the optimal solution for the relaxed problem but at a shorter horizon (Step 1 in the algorithm described below). This is done in order to reduce the computation time of the solution. Further, instead of computing the allocation for each future time slot, we compute the average rate allocated over a larger time slot which corresponds to the time scale at which cars enter and leave the coverage area of the base station (in Step 1). Note that in $\delta = 2$ ms, a car hardly moves any perceivable distance. So, we expect the average channel conditions to change over a much larger time scale (around 1 second) instead of every 2 ms. This larger time scale is also the one in which cars leave and enter the coverage range of the BS. That, is number of cars in the coverage range changes state at this time scale rather than every 2 ms.

At each small time slot t, let $a_i(t) = \sum_{j=1}^{t} \alpha_{ij} r_{ij}$ be the cumulative rate of user i until time slot t, and $K(t)$ be the number of users inside the coverage range.

Our heuristic algorithm follows the steps:

Step 1: In each small slot t, we reduce the dimension of variable α and solve the following problem using projected gradient:

$$(III) \begin{cases} \max \sum_{i=1}^{K(t)} U_i \\ \text{subject to } \sum_{i=1}^{K(t)} \alpha_{it} = 1, \\ \text{and } \sum_{i=1}^{K(t)} \bar{\alpha}_{i\tau} = 1, \\ \alpha_{it}, \bar{\alpha}_{i\tau} \in [0, 1] \end{cases}$$

where

$$U_i = \log \left(a_i(t-1) + \alpha_{it} r_{it} + \sum_{\tau=1}^{J} \bar{\alpha}_{i\tau} \bar{\rho}_{i\tau} \right),$$

τ is big slot, $m = \Delta/\delta$, and $\bar{\alpha}_{i\tau}$ is the future allocation in big slot τ. Note that α_{it} and $\bar{\alpha}_{i\tau}$ are the decision variables in problem (III). Also, $\bar{r}_{ij}$ is the average rate in slot j for user i, and $\bar{\rho}_{i\tau} = \sum_{j=(\tau-1)m+t+1}^{\tau m+t} \bar{r}_{ij}$, is the total average data rate that user i will experience in big slot τ. The value of $\bar{r}_{ij}$ can be predicted using measurements. We remark that the advantage of using the average value $\bar{r}_{ij}$ instead of the exact value $\hat{r}_{ij}$ as is done in (2) is that the prediction error of an average value will be smaller than that of the exact value.

Since the noise is unpredictable in the future, we assume that only the current rate, r_{it}, and the average rate in the future are known.

Step 2: In each small slot t, give full allocation for the user who has the *largest allocation computed by* $(\alpha_{it})_{i=\overline{1,K(t)}}$. We observe that: when number of slots is large enough, the optimal is 0-1 almost everywhere, we can show that when time is continuous the solution is 0-1 everywhere (proof omitted due to lack of space). So even in this step round over $(\alpha_{it})_i$, we can hope that we do not go far from the optimal solution of (III).

The complexity of numerically optimal α computation in step 2 is equal to $20(J+1)\bar{K}\log(\bar{K})$ where 20 is the number of iteration steps of projected gradient in Step 1, $\bar{K}$ is average number of users inside the coverage range, J is the number of big slots.

The proposed algorithm is the similar in spirit to Stochastic Model Predictive Control [9].

3.2 Projected gradient short term objective algorithm 2 (STO2)

The STO1 algorithm recomputes the future allocation in every small time slot. In order to reduce the computational complexity, in STO2, we propose to recompute the future allocation in every big time slot instead.

Step 1: In each big slot τ, we reduce the dimension of variable α and solve the following problem in each big slot by using projected gradient:

$$(III)\begin{cases}\max \sum_{i=1}^{K(\tau)} U_i \\ \text{and } \sum_{i=1}^{K(\tau)} \bar{\alpha}_{i\tau} = 1, \\ \bar{\alpha}_{i\tau} \in [0,1]\end{cases}$$

where

$$U_i = \log\left(a_i((\tau-1)m) + \sum_{\tau=1}^{J} \bar{\alpha}_{i\tau}\bar{\rho}_{i\tau}\right).$$

Here $a_i((\tau-1)m)$ is the total received rate by user i just before the start of big slot τ. The other quantities are same as for algorithm STO1.

Step 2: In each small slot j we shall compute M_{ij} as in (2) where the future allocation $\hat{\alpha}$ is the solution $\bar{\alpha}$ of problem (III).

By doing this, we reduce the computation almost Δ/δ times since we calculate $\bar{\alpha}$ in each big slot only.

4 Numerical results

We now compare the utility of the proposed heuristic with the PF-EXP, $(\text{PF})^2\text{S}$ and a greedy algorithm which allocates to user who has the highest current rate, that is in each time slot j we choose:

$$i^* = \arg \max_{i \in \{1,2,\dots K\}} r_{ij}.$$

For the $(\text{PF})^2\text{S}$ the future allocation was done using the round robin algorithm.

Denote by

$$O^A = \sum_{i=1}^{K} \log \big(\sum_{j=1}^{T} \alpha_{ij}^A r_{ij} \big),$$

the total reward of algorithm A and by $\bar{O}^A = \frac{1}{K} O^A$ its average reward over K users. Given A, B two algorithms, then the ratio between A, B equals $\exp(\bar{O}^A - \bar{O}^B)$. The percentage of improvement of algorithm A over B is computed equal to $(\exp(\bar{O}^A - \bar{O}^B) - 1) \cdot 100\%$.

Due to the logarithm in the objective function, taking a different unit of measure for the rate will give a different percentage improvement between algorithms. Although logarithm is an increasing function, we can know which algorithm is better than the other, but we will not get a consistent percentage improvement across different units of measure. Therefore, by taking the difference as above we construct a consistent criterion for comparison.

The road length is taken to be $L = 1000$ m with 0 at the leftmost edge. The closest point on the road to the BS is at $x = 500$ m. The data rate at position x along the road is given by:

$$r(x) = \eta \cdot (1 + \kappa \exp(|x - 500|/\sigma), \tag{4}$$

where $\kappa \geq 0$ is a real number and η is uniform random variable whose range will be in $[0.8, 1.2]$ unless stated otherwise. A sample path of $r(x)$ is shown in Fig. 2. This function has the highest mean at the mid-point of the segment and the lowest mean at the two end points.

We emphasize the algorithm itself is independent of the rate function. We chose the above rate function for convenience.

The time horizon T was 4000000 small time slots which is 8000 secs or a little over two hours. The big slot length Δ for our projected gradient short term objective algorithm was taken as 1 sec or equivalently 500 small time slots.

4.1 Homogeneous vehicle velocities

First, we show the results when all vehicles move with the same velocity which is taken to be $v = 25$ m/s. That is, there are $N = 20000$ spatial small slots in the coverage range and $J = 40$ big slots. A new car enters through the left edge in every big slot (i.e., every second) with probability p.

Figure 3 shows the average utility obtained by a vehicle for each of the four algorithms as a function of the probability of arrival of car in each big slot. Figure 4 shows the percentage improvement of three other algorithms compared to PF-EXP. The proposed algorithm does better than PF-EXP and more importantly better than $(\text{PF})^2\text{S}$. Although, we have shown the greedy algorithm for

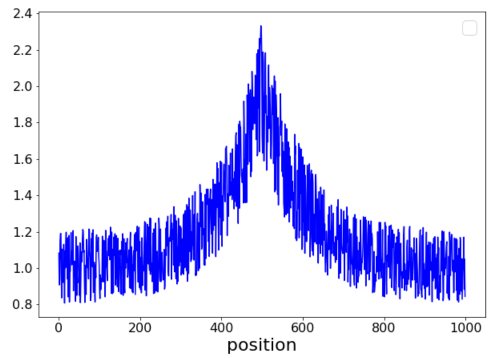

Fig. 2. Sample path of data rate at various positions along the road. $\sigma = 100, \kappa = 1$ and $\eta \in [0.8, 1.2]$.

comparison, we emphasize that greedy is not practically implemented because it can be very unfair to users that have heterogeneous rates. In the simulated scenario, all vehicles move along the same road and observe statistically identical but position dependent conditions during their stay. These conditions are rather favorable for the greedy algorithm.

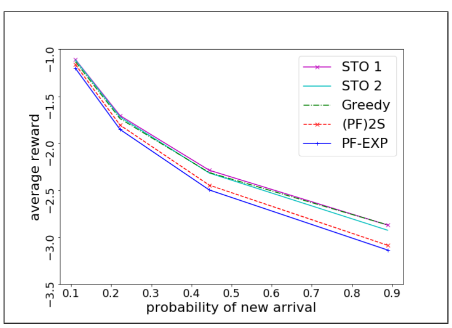

Fig. 3. Average reward per car. Homogeneous velocities.

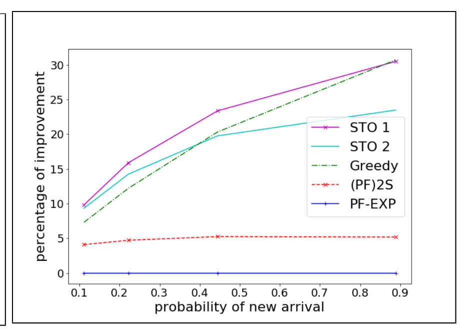

Fig. 4. Percentage improvement over PF-EXP. Homogeneous velocities.

4.2 Comparison with the upper bound

Next, again for homogeneous velocities, we also include the solution of the relaxed problem (II) but for a smaller road length and shorter horizon because it is computationally expensive. The parameters for this setting are: $L = 100$ m, $J = 40$ big slots, $T = 500$ s, and the other parameters are the same as in the homogeneous case. We assume that the relaxed algorithm knows the future arrivals and the future rate exactly whereas the other algorithms do not know this information. The solution to the relaxed problem gives an upper bound to the optimal solution of the original problem in (I).

Figures 5 and 6 plot the average reward per car and percentage improvement for the five algorithms with respect to PF-EXP. It is seen that the proposed algorithm is quite close to the upper bound in this scenario.

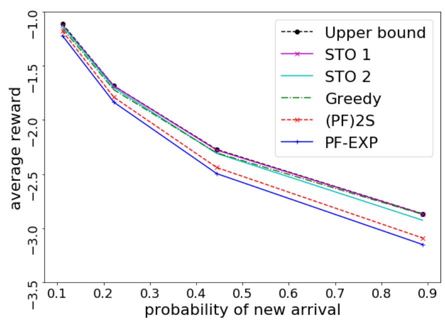

Fig. 5. Average reward per car. Includes the upper bound from the solution of (II). Small setting of homogeneous velocities.

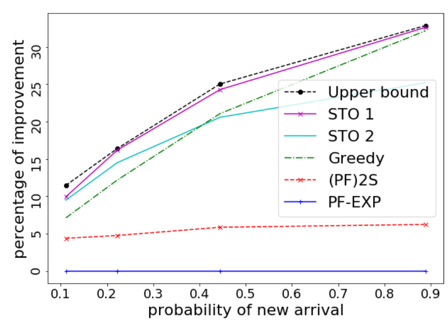

Fig. 6. Percentage improvement over PF-EXP. Small setting of homogeneous velocities.

4.3 Heterogeneous vehicle velocities

Finally, we show the results when a fraction q of vehicles move with $v_0 = 25$ m/s and the other fraction with $v_1 = 12.5$ m/s. Note that the proposed algorithm takes the larger of the two values of N computed with the two velocities. The faster class of cars will be called class 0. Here, the horizon, T, is a little over 2 hours.

For this scenario, the probability of new a arrival is fixed at $p = 2/9$ in figure 7, 8 and at $p = 4/9$ in figure 9, 10. Thus a new car of class 0 arrives with probability $p \cdot q$ and with probability $p(1-q)$ a new car of class 1 arrives. A new car enters through the left edge in every big slot (i.e., every second) with probability p.

Figure 7 (resp. Fig. 9) shows the average utility obtained by a vehicle for each of the four algorithms as a function q for probability of new arrival $p = 2/9$ (resp. $p = 4/9$). Figure 8 (resp. Fig. 10) shows the percentage improvement of three other algorithms compared to PF-EXP for the two probabilities of arrival as before. As before, the proposed algorithm does better than both PF and $(\text{PF})^2$S.

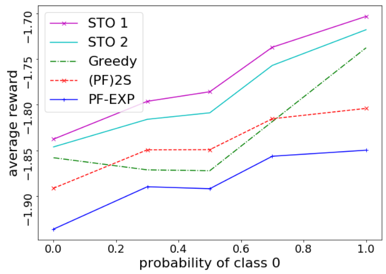

Fig. 7. Average reward per car. Heterogeneous velocities. $p = 2/9$.

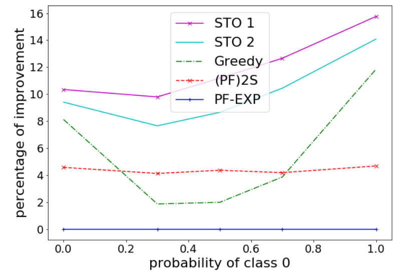

Fig. 8. Percentage improvement over PF-EXP. Heterogeneous velocities. $p = 2/9$.

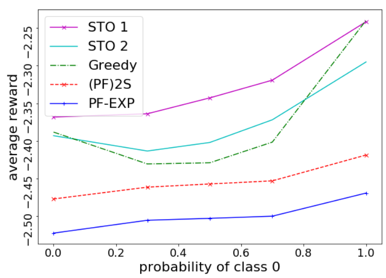

Fig. 9. Average reward per car. Heterogeneous velocities. $p = 4/9$.

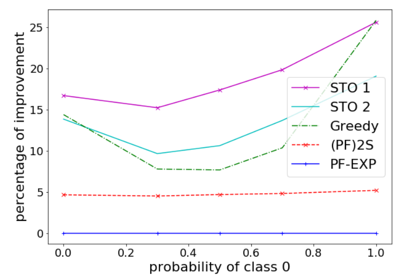

Fig. 10. Percentage improvement. Heterogeneous velocities. $p = 4/9$.

5 Future work

The results in this paper were obtained on a simplified scenario of one base station serving only cars traveling on one road. Our immediate goal is to evaluate the performance of the proposed algorithm in more complex scenarios with a network of multiple base stations and different classes or mobile (e.g., pedestrians) and stationary users. The algorithm will be designed to be run in either a coordinated or a distributed manner.

Other directions of research include: investigation of the tradeoff between the reduction of complexity (e.g., by changing the size of the horizon and the big slots) and the quality of the heuristic; integration of users with different classes of QoS requirements including latency, jitter, or periodic communication constraints; and computation of analytical bounds on the sub optimality of the proposed algorithm.

References

1. H. J. Bang, T. Ekman, and D. Gesbert. Channel predictive proportional fair scheduling. *IEEE Transactions on Wireless Communications*, 7(2):482–487, February 2008.
2. S. Borst. User-level performance of channel-aware scheduling algorithms in wireless data networks. *IEEE/ACM Transactions on Networking*, 13(3):636–647, June 2005.
3. S. Borst, N. Hegde, and A. Proutiere. Mobility-driven scheduling in wireless networks. In *IEEE INFOCOM 2009*, pages 1260–1268, April 2009.
4. P. Chandur, R. M. Karthik, and K. M. Sivalingam. Performance evaluation of scheduling algorithms for mobile wimax networks. In *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 764–769, March 2012.
5. L. Condat. Fast projection onto the simplex and the l1 ball. *Math. Program.*, 158:575–585, 2016.
6. F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(1):33–37, 1997.
7. H. J. Kushner and P. A. Whiting. Convergence of proportional-fair sharing algorithms under general conditions. *IEEE Transactions on Wireless Communications*, 3(4):1250–1259, July 2004.
8. R. Margolies, A. Sridharan, V. Aggarwal, R. Jana, N. K. Shankaranarayanan, V. A. Vaishampayan, and G. Zussman. Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms. *IEEE/ACM Trans. Netw.*, 24(1):355–367, Feb. 2016.
9. A. Mesbah. Stochastic model predictive control: An overview and perspectives for future research. *IEEE Control Systems Magazine*, 36(6):30–44, Dec 2016.
10. J. Mo and J. Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5):556–567, Oct 2000.
11. L. Tan, Z. Zhu, F. Ge, and N. Xiong. Utility maximization resource allocation in wireless networks: Methods and algorithms. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(7):1018–1034, July 2015.
12. Y. Yi and M. Chiang. Stochastic network utility maximisation—a tribute to kelly's paper published in this journal a decade ago. *European Transactions on Telecommunications*, 19(4):421–442, 2008.
13. H. Zhou, P. Fan, and J. Li. Global proportional fair scheduling for networks with multiple base stations. *IEEE Transactions on Vehicular Technology*, 60(4):1867–1879, May 2011.

A Projection on feasible set D

In fact D is a Cartesian product of J simplexes: $D = D_1 \times D_2 \cdots \times D_J$ where

$$D_j = \{a_j = (\alpha_{ij})_{i=\overline{1,K}} \in [0,1]^K, \sum_{i=1}^{K} \alpha_{ij} = 1\}$$

for all $j = 1, 2, ..., J$.
Then $(D_j)_j$ are simplexes, so we can compute the projection on D_j following [5]. The projection on D can be computed by the simple following lemma:

Lemma 1 *If* $Y = (y_{ij})_{i=\overline{1,K}, j=\overline{1,J}} \in \mathbb{R}^{K \times J}$, *then*

$$\Pi_D(Y) = \Pi_{D_1}(Y_1) \times \Pi_{D_2}(Y_2) \times \cdots \times \Pi_{D_J}(Y_J),$$

where $Y_j = (y_{ij})_{i=\overline{1,K}}$.

Proof. (of the lemma 1) Denote by $Z = \Pi_{D_1}(Y_1) \times \Pi_{D_2}(Y_2) \times \cdots \times \Pi_{D_J}(Y_J)$. It is obvious to check that for any $X \in D^{K \times J}$ then $\langle Y - Z, X - Z \rangle \leq 0$.

As describe in [5], the complexity of finding Π_{D_j} is equal to $K \log(K)$ by observation in practice, and equal to $O(K^2)$ in the worst case. Therefore the complexity of finding projection on $D = D_1 \times D_2 \cdots \times D_J$ is equal to $JK \log(K)$ in practice.

B Modeling as an optimal control problem

As aforementioned, in this part we prove 0-1 everywhere property of the solution of the continuous time for the original problem.

When user i moves in the road, his position changes continuously, denote by $x_i(t)$ position of that user. The rate changes in the way we explain above according to position of the user, denoted by $r_i(x_i(t))$. Assume that the BS can allocate in continuous time, we denote by $a_i(t)$ is the allocation for user i at time t, we relaxed the integer constraint so that $a_i(t) \in [0, 1]$ then we get a continuous control optimization:

$$(VI) \begin{cases} \max \sum_{i=1}^{K} \log\left(\frac{1}{T}\int_0^T \alpha_i(t) r(t, x_i(t)) dt\right) \\ \text{such that } \alpha_i \in \mathcal{S}^K \forall t, \forall i, \end{cases}$$

On other hand we have Mayer form with the *terminal cost function* as follows:

$$\begin{cases} \max \sum_{i=1}^{K} \log\left(y_i(T)\right), \\ \dot{y}_i(t) = \alpha_i(t) r(t, x_i(t)) dt, y_i(0) = 0 \forall i, \\ \dot{x}(t) = v(t) \forall i, \end{cases}$$

We can solve (VI) to get the optimal solution by using maximum principle. The solution of (VI) can be not unique, but one of its solution has the integer form as described in the following proposition:

Proposition 3 *The solution of the problem (I) is of the form: for every* t, $\alpha_{i^*}(t) = 1$ *and* $\alpha_j(t) = 0$ *for all* $j \neq i^*(t)$, *where*

$$i^*(t) = \arg\max_i \frac{r(t, x_i(t))}{y_i(T)}.$$

C Proofs

Proof. (proof of proposition 2) The optimal is obtained by proving that for any $\alpha \in D$,

$$\nabla O(\alpha^*)(\alpha^* - \alpha) \leq 0.$$

Because of lemma 1, we will reduce the proof on D_1. Assuming O is convex function on D_1, we shall prove that if $\alpha^* = (\alpha_i^*)_{i=1,...,K} \in D_1$ satisifes

$$\Pi_{D_1}(\alpha^* + \epsilon \nabla(\alpha^*)) = \alpha^* \tag{5}$$

where ϵ positive, then

$$\nabla O(\alpha^*)(\alpha^* - \alpha) \geq 0, \text{ for any } \alpha \in D_1$$

i.e, α^* is global optimal of O. Indeed, without loss of generality, we assume that

$$\alpha_1^* + \epsilon \frac{\partial O}{\partial \alpha_1^*} \geq \alpha_2^* + \epsilon \frac{\partial O}{\partial \alpha_2^*} \geq ... \geq +\alpha_M^* + \epsilon \frac{\partial O}{\partial \alpha_M^*} \geq ... \geq \alpha_K^* + \epsilon \frac{\partial O}{\partial \alpha_K^*}$$

where M is the largest index such that

$$\frac{1}{M} \sum_{i=1}^{M} \big(\alpha_i^* + \epsilon \frac{\partial O}{\partial \alpha_i^*} - 1 \big) \leq \alpha_M^* + \epsilon \frac{\partial O}{\partial \alpha_M^*}.$$

Denote by $\tau = \frac{1}{M} \sum_{i=1}^{M} \big(\alpha_i^* + \epsilon \frac{\partial O}{\partial \alpha_i^*} - 1 \big)$, by proposition 10 in [5] we have:
$\Pi_{D_1}(\alpha^* + \epsilon \cdot \nabla(\alpha^*)) = \big(\alpha_1^* + \epsilon \frac{\partial O}{\partial \alpha_1^*} - \tau, \alpha_2^* + \epsilon \frac{\partial O}{\partial \alpha_2^*} - \tau, ..., \alpha_M^* + \epsilon \frac{\partial O}{\partial \alpha_M^*} - \tau, 0, ..., 0 \big)$.
Using (5) to compare term by term we get:

1. $\alpha_{M+1}^* = \cdots = \alpha_K^* = 0$,
2. $\alpha_{M+1}^* + \epsilon \frac{\partial O}{\partial \alpha_{M+1}^*} \leq \tau, \cdots, \alpha_K^* + \epsilon \frac{\partial O}{\partial \alpha_K^*} \leq \tau$. Now, from the first item we have $\alpha_{M+1}^* = \cdots = \alpha_K^* = 0$. It implies $\epsilon \frac{\partial O}{\partial \alpha_{M+1}^*} \leq \tau, \cdots, \epsilon \frac{\partial O}{\partial \alpha_K^*} \leq \tau$,
3. $\epsilon \frac{\partial O}{\partial \alpha_1^*} = \cdots = \epsilon \frac{\partial O}{\partial \alpha_M^*} = \tau$.

Thus,

$$\begin{aligned}
\epsilon \nabla O(\alpha^*)(\alpha^* - \alpha) &= \sum_{i=1}^{K} \epsilon \frac{\partial O}{\partial \alpha_i^*} (\alpha_i^* - \alpha_i) \\
&= \sum_{i=1}^{M} \epsilon \frac{\partial O}{\partial \alpha_i^*} (\alpha_i^* - \alpha_i) + \sum_{i=M+1}^{K} \epsilon \frac{\partial O}{\partial \alpha_i^*} (\alpha_i^* - \alpha_i), \\
&= \sum_{i=1}^{M} \tau (\alpha_i^* - \alpha_i) + \sum_{i=M+1}^{K} \epsilon \frac{\partial O}{\partial \alpha_i^*} (\alpha_i^* - \alpha_i), \\
&= \sum_{i=1}^{K} \tau \alpha_i^* - \sum_{i=1}^{K} \tau \alpha_i + \sum_{i=M+1}^{K} \big(\epsilon \frac{\partial O}{\partial \alpha_i^*} - \tau \big) (\alpha_i^* - \alpha_i), \\
&= \tau - \tau + \sum_{i=M+1}^{K} \big(\epsilon \frac{\partial O}{\partial \alpha_i^*} - \tau \big) (0 - \alpha_i) \\
&\geq 0.
\end{aligned}$$

The last sum less than 0 since all its terms are greater than or equal to 0.

Proof. (proof of proposition 1) In fact the condition (3) implies that $\tilde{\nabla} O(\alpha^*) = 0$ and from proposition 2 we have conclusion.