Using channel predictions for improved proportional-fair utility for vehicular users

Thi Thuy Nga Nguyen, Olivier Brun, Balakrishna Prabhu

Using channel predictions for improved proportional-fair utility for vehicular users

Thi Thuy Nga Nguyen * Olivier Brun

Balakrishna J. Prabhu †

Abstract

As the channel conditions experienced by vehicular users in cellular networks vary as they move, we investigate to what extent the quality of channel allocation could be improved by exploiting predictions on future data rates in non-stationary environments. Assuming mean future rates can be computed from Signal-to-Noise Ratio (SNR) maps, we propose an algorithm which predicts future throughputs over a short-term horizon at regular time intervals, and then uses this extra-knowledge for improved online channel allocation. The prediction of future throughputs is obtained by solving a relaxed version of the problem using a projected gradient algorithm. When the transmit powers of the base stations can be varied over time, a straightforward extension of our algorithm can be used for the joint optimization of channel allocation and transmit power control under average and maximum power constraints. Using event-driven simulations, we compare the performance of the proposed algorithms against those of other channel allocation algorithms, including the Proportional Fair (PF) scheduler, which is known to be optimal in stationary environments, and the $(\mathrm{PF})^2\mathrm{S}$ scheduler, which was devised for mobiles nodes in non-stationary environments. The simulated scenarios, which cover the cases with and without power control, include scenarios with multiple base stations and are based on realistic mobility traces generated using the road traffic simulator SUMO. Simulation results show that the proposed algorithms outperform the other algorithms and that exploiting the knowledge of future radio conditions allows a significantly better channel allocation.

1 Introduction

A central and challenging problem in cellular networks is channel allocation, that is, to decide which mobile user the base station (BS) should serve in each time

*T.T. Nga Nguyen is with Torus Actions SAS, Toulouse, France, e-mail: nt-tnga@math.ac.vn

†Olivier Brun and Balakrishna J. Prabhu are with LAAS-CNRS, Université de Toulouse, CNRS, e-mail: brun@laas.fr and balakrishna.prabhu@laas.fr

slot. To this end, the BS gathers the channel state information (CSI) from users in order to know their radio conditions, which are mainly determined by their distances to the BS and by fading effects. As maximizing the overall throughput would lead to the starvation of distant users (those with the worst potential data rates), today cellular networks allocate the channel to the user with the highest potential rate proportionally to its time-average throughput[1]. With this strategy, users with comparatively low allocated throughput are occasionally assigned a higher priority even when they are in worse channel conditions. This scheduling algorithm, which is known as the Proportional Fair (PF) scheduler, provides a fair and efficient sharing of bandwidth between users in the sense that it maximizes the aggregate logarithmic utility of obtained throughput in a fixed population of permanent users [8].

A number of studies have been devoted to the analysis of the performance of PF scheduling in wireless networks [3, 17, 4, 19, 20, 5], assuming either a static population of permanent users, or a dynamic setting in which random finite-size data transfers come and go over time. In both cases, it was shown that PF scheduling strikes a good balance between the overall network throughput and the degree of fairness among users. In contrast to the above references, where it is assumed that the transmit powers of BSs cannot be varied over time, some authors have considered the joint optimization of channel allocation and transmit power control. This problem has been investigated for different multiplexing schemes such as CDMA [16] and OFDM [7]. The proposed algorithms base their decisions on the current channel conditions and on previous decisions.

However, 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 [9]. Thus, even if they take into account the fast channel variations due to multi-path propagation, most studies ignore the variations of the channel conditions on slower time scale dues to user mobility. Taking into account such slow fading effects is particularly important for vehicular users as the mean of the Signal-to-noise ratio (SNR) improves as a vehicle comes closer to a BS and then worsens as it moves far away. Another usual assumption which is not realistic for vehicular users is the assumption of long sojourn time. Indeed, a vehicle typically stays in the scheduling area of a BS for only a few minutes.

In this article, we investigate to which extent the quality of channel allocation could be improved by exploiting information on future radio conditions in non-stationary environments. Our main motivation comes from connected vehicles which will use cellular networks to exchange information related to security and driving conditions with their environment. If the trajectory of a car is known or can be estimated from historical travel data and/or observations of the surrounding environment, then one can obtain good statistical predictions of the SNR that will be experienced by the car in the near future. In turn, these

[1]The throughput is different from the data rate. While the latter is the 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.

predictions could be used by the BSs to achieve a channel allocation with a higher utility than that of the PF algorithm. In this paper, we propose a channel allocation policy exploiting this extra knowledge and evaluate the improvement in utility that it yields in non-stationary environments. Note that such an improvement in utility is not possible under the assumption of a stationary channel as knowing the users' trajectory does not bring any new information on the future data rates, since users are static and hence their trajectory is just one point. We also show how to extend the proposed scheduling policy to address a joint channel allocation and transmit power control problem in which there are average and power constraints on the base-stations.

The idea of using information on future radio conditions for channel allocation was already explored in [2]. It uses 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 BS. In the context of high-speed trains, [18] solves the opportunistic utility maximization problem assuming all the future rates are perfectly known and with average power constraints.

Another closely related work is [11] in which, using SNR maps obtained by measurements, the authors first show that PF scheduling may perform poorly in the presence of slow fading. They then propose a scheduling algorithm which is similar to PF in that the channel is allocated to the user with the highest potential rate proportionally to its total throughput. This new algorithm, which is called $(\mathrm{PF})^2\mathrm{S}$ differs however from PF in that the total throughput includes an estimation of the future throughput whereas PF considers only the already allocated throughput. In order to estimate the future throughput, the authors proposed three methods: round-robin, blind estimation, and a local search heuristic. It was shown that even with a rough estimation of the future throughput, this new index leads to an improved utility compared to the PF algorithm in non-stationary environments. The channel allocation policy proposed in this paper is similar to the $(\mathrm{PF})^2\mathrm{S}$ scheduling policy except that we use a different method for estimating future throughputs of vehicles. For the purposes of numerical comparisons, we shall assume in this paper that $(\mathrm{PF})^2\mathrm{S}$ uses the round-robin policy. It was stated in [11] that, out of the three estimation methods, round-robin is the most robust to prediction errors.

1.1 Contributions

We first consider the case when the transmit powers of BSs cannot be varied over time. We present two heuristic algorithms for non stationary channels that improve the total utility of users compared to the PF and the $(\mathrm{PF})^2\mathrm{S}$ algorithms. Our heuristics are similar to the $(\mathrm{PF})^2\mathrm{S}$ algorithm, except that instead of computing an estimation of future throughput from a round-robin allocation, we compute it as the solution of a utility maximization problem over a short-term horizon assuming that the means of the future data rates are known over this short horizon. The two heuristics differ in the frequency at which they recompute future allocations.

The original utility maximization problem being computationally complex, we employ three techniques to obtain lower complexity heuristics: *(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 that helps the algorithm run in real time. 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.

When the transmit powers of the BSs can be varied over time, a straightforward extension of our heuristics can be used for the joint optimization of channel allocation and transmit power control under average and maximum power constraints.

We compare the performance of the proposed algorithms against those of other channel allocation algorithms using event driven simulations. The simulated scenarios include scenarios with multiple base stations and are based on realistic mobility traces generated using the open-source road traffic simulator SUMO with vehicles moving at either equal or different speeds. Simulation results show that the proposed algorithms outperform other algorithms and that exploiting the knowledge of future radio conditions allows a significantly better channel allocation.

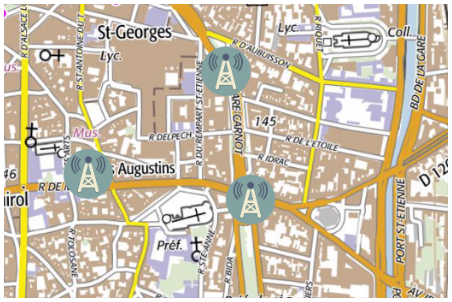
Some of the results presented in this paper appeared in ASMTA 2019 [13] and RAWNET 2020 [15]. These publications were limited to scheduling in a single base station setting and do not include experiments with SUMO.

1.2 Organisation

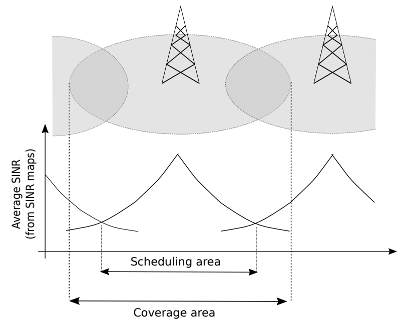
In Section 2, we state the channel allocation problem addressed in this paper. Section 3 briefly describes some existing channel allocation algorithms, which shall be used for comparison purposes. In Section 4, we present two different heuristics for improving the proportional fairness between vehicular users. Section 5 is devoted to the extension of our heuristics for the joint optimization of channel allocation and power control. Section 6 presents the numerical results for some simple scenarios with homogeneous vehicles, as well as for some more advanced scenarios generated with SUMO. Finally, Section 7 summarizes our findings and discusses some future research directions.

2 Problem formulation

We consider a geographical region with a network of roads that is served by a set $\mathcal{B} = \{1, \ldots, B\}$ of BSs. The region is partitioned into B non-overlapping sub-regions, one for each BS. The sub-region assigned to a BS corresponds to the area in which this BS has a better average SINR that the other BSs as



(a) A selected area of Toulouse which is covered by three BSs of the French mobile network operator Free Mobile.



(b) Difference between coverage area and scheduling area

given by their SINR maps. We will call the sub-region of a BS as the *scheduling area* of this BS. This reflects the fact that users in the scheduling area of a BS are assigned to this BS which is responsible for the scheduling decisions for the users that are present there. The scheduling area is in fact a subset of the coverage area of a BS which is the area over which an acceptable signal is received from this BS. See Figure 1b illustrating the difference between the two. The SINR maps could in general vary over time and need not be symmetric. The scheduling area can be updated to reflect the temporal variations.

Users (vehicles, bicycles, pedestrians, etc) enter the network, move along different routes, and leave the network. In the following, we shall denote by $\mathcal{U}(t)$ the set of users in the system at time t. Figure 1a shows an area within the city of Toulouse which will be later used in the numerical experiments. In the figure, the width of the box is approximately 1 km, and the height is around 0.65 km. The data for BS location can be found on the website[2] of the French Frequency Agency (ANFR), which manages all radio frequencies in France.

Every $\delta = 2$ ms each BS has to decide which user to serve in a decentralized fashion. Here, we are assuming that there is one resource (channel, in this case) that a BS can allocate in each slot to the users in its scheduling area. In practice, there can be multiple channels and the ideas presented in this paper can be applied to this case as well. Throughout the paper, we shall also assume that a user can only be served by the closest BS. We define $\mathcal{U}_b(t)$ as the subset of users that are attached to BS b at time t, and $x_i(t)$ as a binary decision variable which is equal to 1 if the channel of BS b is allocated to user $i \in \mathcal{U}_b(t)$ at that time, and 0 otherwise.

The allocation decisions of the BSs are based on the channel conditions of the users. In time slot t, user $i \in \mathcal{U}_b(t)$ has a potential data rate of $r_i(t)$ if it is allocated the channel of BS b. In the numerical experiments (see Section 6), when the user is at distance d from the BS, $r_i(t)$ will have the form

$$r_i(t) = \eta_i(t) \cdot \bar{r}\,(d)\,, \tag{1}$$

[2] `https://data.anfr.fr/anfr/portail`

where $\eta_i(t)$ is a random variable that models time-varying SNR and $\bar{r}\,(d)$ is the average data rate which depends on the distance d. More precisely, we assume that

$$\bar{r}(d) = \begin{cases} 0 & \text{if } d > d_{max}, \\ 1 + \kappa\, e^{-d/\sigma} & \text{otherwise,} \end{cases} \tag{2}$$

where κ and σ are adjustable parameters and d_{max} is a given number standing for the scheduling area of the BS. We emphasize that the scheduling algorithms proposed in this paper do not require this assumption to work.

With the above definitions, the throughput of user i at time t can then be defined as

$$\phi_i(t) = x_i(t) \times r_i(t), \tag{3}$$

which means that the throughput is 0 if the channel of the serving BS is not allocated to user i, and that it is equal to the potential data rate $r_i(t)$ otherwise.

Denote by T the time horizon over which the scheduling decisions need to be determined, and let K be total number of users who pass through the considered region during that time. For simplicity, we assume that T is a multiple of δ. Our objective is to achieve the proportional-fairness between users, which is described by the following optimization problem (see, e.g., [3, 11, 1]):

$$\begin{cases} \text{maximize} \quad U(\mathbf{x}) = \displaystyle\sum_{i=1}^{K} \log\left(\frac{1}{T}\sum_{t=1}^{T}\phi_i(t)\right) \\ \text{s.t.} \\ \qquad\qquad \displaystyle\sum_{i\in\mathcal{U}_b(t)} x_i(t) = 1, \quad b \in \mathcal{B},\ t = 1,\dots,T, \\ \qquad\qquad x_i(t) \in \{0,1\}, \quad i \in \mathcal{U}_b(t), b \in \mathcal{B}, t = 1,\dots,T. \end{cases} \tag{I}$$

The objective function is the sum of the logarithm of the mean throughputs of the users. Constraints $\sum_{i\in\mathcal{U}_b(t)} x_i(t) = 1$ imply that each BS serves exactly one user at each time t. Finally, the last constraints $x_i(t) \in \{0,1\}$ imply that a feasible solution is a binary vector $\mathbf{x}$.

Solving the optimization problem (I) presents several practical difficulties. First and foremost, in time-slot t, the potential data rates $r_i(k)$ in the future (i.e., for $k > t$) are unknown. Thus, the channel allocation has to be done online with the knowledge of the past allocations and the current potential data rates only. The other difficulty is computational due to (I) being NP-hard [11].

Apart from the algorithm in [11], the existing algorithms for channel allocation base their decisions only on the past and current information. It was shown in [11] that by including an estimate of future data rates, the global utility can be improved. We shall present two algorithms in Section 4 that further improve upon the gains shown in [11].

3 Existing Algorithms

Before introducing our algorithms, we first present three existing channel allocation algorithms that will also be used for comparison in our numerical experiments.

3.1 Greedy allocation

The simplest channel allocation scheme is the greedy algorithm in which the BS allocates the channel to the user with maximum current potential rate, that is, in time-slot t, BS b allocates its channel to a user $i_b^* \in \mathrm{argmax}_{i \in \mathcal{U}_b(t)} \{r_i(t)\}$. This algorithm does not use information on the past allocations and greedily maximizes the utility for the local allocation only. In scenarios where users spend similar amounts of time close to a BS (that is, have good channel conditions), greedy can perform well. On the other hand, when these times can vary a lot between users, it can lead to unbalanced allocations and possibly starvation to users who spend more time in poor channel conditions. Moreover, greedy allocation can be sporadic from a user's point of view. A user who is moving away from a BS may have to wait until it gets close to the next BS before it get a new allocation. This may lead to short-term unfairness or starvation.

3.2 Proportional Fair (PF) allocation

The PF-EXP algorithm, versions of which are implemented in cellular networks, improves the fairness by taking into account not only the current potential rate but also the previous allocations. It chooses the user with the highest ratio of the current rate to the observed throughput, that is, BS b chooses the user $i \in \mathcal{U}_b(t)$ who maximizes the ratio $r_i(t)/A_i(t-1)$, where

$$A_i(t) = A_i(0) + \sum_{k=1}^{t} \phi_i(k),$$

is the total allocated rate to user i up to time t ($A_i(0)$ is the initial value for each user). The algorithm takes into account the past allocation but ignoring completely the future one. A nice property that makes this algorithm attractive is that, in the long-run when T goes to ∞, it is optimal for a stationary and ergodic channels and for a fixed number of users [9].

In [11], it was shown using measurements that the potential data rates for vehicular traffic are not necessarily stationary. The mean potential data rate can vary along a road segment as a vehicle gets close or moves away from the BS. The non-stationarity of potential data rate means that PF-EXP may no longer be optimal which opens up the possibility to use future information for improving the global utility. The exact future rate is of course unknown but current allocation can be based on predictions made using SNR maps.

3.3 Predictive Finite-horizon PF Scheduling ($(\mathrm{PF})^2\mathrm{S}$)

Based on the observation of non-stationarity, in [11], the authors proposed a modified PF algorithm that exploits estimated future rate. This algorithm works as follows:

- It predicts future data rates $\hat{r}_i(s)$ of cars in every future slot $s = t+1, t+2, \ldots$, T.
- It estimates future channel allocations $\hat{\mathbf{x}}$ based on the data rate predictions. As mentioned in Section 1, the estimations can be computed using either a round-robin policy, a blind estimation, or a local-search method. It is stated in [11] that, out of these three, round-robin is more robust to prediction errors. Given this, we shall use *Round Robin Estimation* (RRE) as the estimation policy for $(\mathrm{PF})^2\mathrm{S}$ in the numerical comparisons. As a reminder, RRE assumes that future time slots are allocated in a round-robin manner.
- For each time slot t, the BS b chooses the user $i \in \mathcal{U}_b(t)$ who maximizes $M_i(t)$, where

$$M_i(t) = \frac{r_i(t)}{A_i(t-1) + \hat{x}_i(t) r_i(t) + \sum_{k=t+1}^{T} \hat{x}_i(k)\hat{r}_i(k)}. \tag{4}$$

$(\mathrm{PF})^2\mathrm{S}$ is also an index-based algorithm like PF-EXP. The main difference between the two is that, while PF-EXP takes into account only the past rates in the denominator, $(\mathrm{PF})^2\mathrm{S}$ also includes the estimated future allocations. It can be seen that (4) is related to the gradient of (I) with future rates replaced by their estimates, and choosing the user with maximal $M_i(t)$ can be seen as choosing a step in the direction of the maximal gradient. In the case of one BS, provided the future channel allocations $\hat{\mathbf{x}}$ can be predicted correctly, an optimal solution to problem (I) can be obtained, as stated in Proposition 1.

Proposition 1. *If there exists* $\mathbf{x}^*$ *satisfying* $x^*_{i(t)}(t) = 1$ *and* $x^*_i(t) = 0, \forall i \neq i(t)$, *where* $i(t)$ *belongs to*

$$\underset{i \in \mathcal{U}_b(t)}{\arg\max} \frac{r_i(t)}{\sum_{k=1}^{T} x^*_i(k) r_i(k)}, \tag{5}$$

for all t, *then* $\mathbf{x}^*$ *is an optimal solution of problem* (I).

Proof. See B.2. □

Note that condition (5) is a sufficient condition for $\mathbf{x}^*$ to be an optimal solution of problem (I), but not a necessary condition and x^* whose components are integers does not always exist. In chapter 5 of [14], we discussed more detail about other properties of this optimization problem and it's solution. Moreover, in proposition of 5.5.5 of that reference, when $|\mathcal{U}_b(t)| = 1$, $T = 2$, $K = 2$, we solved and illustrated the solutions of the relaxed version which will be defined later in (II).

In the next section, we present our heuristics. The motivation for the heuristics comes from the observation that the formula (4) for (PF)^{2}S looks like one-step of the gradient descent with starting point chosen according to the round robin policy when the RRE is used. We may expect to get a better allocation if we do more iterations instead of only one, ensuring that in every iteration the allocation is in the feasible set. To do this we employ a projected gradient algorithm, as described in the next section.

4 Channel allocation heuristics

We shall assume that each BS allocates the channel independently, that is, in a decentralized manner and without coordination with the other BSs. The channel allocation is done by a BS by taking into account the future data rates of the users currently attached to this BS. Since each BS decides independently, we consider an arbitrary BS, say BS b.

We propose two heuristic algorithms, Short Term Objective 1 (STO1) and Short Term Objective 2 (STO2), which are presented in the following. As explained below, the two heuristics use a different method for estimating the future throughput than the round-robin scheme used in the (PF)^{2}S algorithm. This estimate is based on maximizing the total utility with the future mean channel gains as an estimate for the actual realizations. This is similar in spirit to Stochastic Model Predictive Control [12]. The two heuristics differ in the time-scale at which updated future information is used as well as in the dimension of the optimization problem solved at each decision epoch.

4.1 Projected gradient algorithm

Before describing the two heuristics, we explain the ideas common to them. The first step is to relax the integer constraints on the allocation variables in optimization problem (I), so as to obtain the following convex optimization problem

$$\begin{cases} \text{maximize} & U(\mathbf{x}) = \displaystyle\sum_{i=1}^{K} \log\left(\frac{1}{T}\sum_{t=1}^{T}\phi_i(t)\right) \\ \text{s. t.} & \\ & \sum_{i\in\mathcal{U}_b(t)} x_i(t) = 1, \quad t = 1,\dots,T, \\ & x_i(t) \in [0,1], \quad i \in \mathcal{U}_b(t), t = 1,\dots,T, \end{cases} \tag{II}$$

where K is total number of users passing BS b during T. The relaxed problem (II) is almost the same as the original problem, except the fact that now $x_i(t)$ does not need to be an integer. Let us assume for the moment that all the future arrivals are known. Under this assumption, (II) can be solved efficiently using the projected-gradient algorithm based on the formula for the projection on a simplex given in [6], as described below. Let us denote by

$$D = \left\{ \mathbf{x} \in [0,1]^{K(t)\times T} \ : \ \sum_{i=1}^{K(t)} x_i(t) = 1, \ t = 1, 2, ..., T \right\}$$

the feasible set of the relaxed problem (II) with $K(t) = |\mathcal{U}_b(t)|$. Observe that although D is not a simplex, it is the Cartesian product of T simplexes since for every component $t = 1, 2, \ldots, T$ the feasible set of allocations is indeed a simplex (see details in A). We can therefore obtain a projection on D by projecting independently on the simplexes corresponding to each of the time-steps. The procedure for computing the projection Π_D on the set D is formalized in A.

In the following, we present the projected gradient algorithm. Starting by initializing an arbitrary $\mathbf{x}^{(0)}$ in the feasible set D, the algorithm computes at each iteration $n = 1, 2, \ldots$ a new feasible solution using the formula

$$\mathbf{x}^{(n+1)} = \Pi_D \left(\mathbf{x}^{(n)} + \epsilon_n \nabla U(\mathbf{x}^{(n)}) \right), \tag{6}$$

where $\nabla U(\mathbf{x})$ is the gradient of the utility function at point $\mathbf{x}$ and $\epsilon_n \in (0,1)$ is the step size at iteration n. A new feasible solution is computed until convergence is reached. We have however limited the number of iterations to 20 in the numerical results presented in Section 6.

Proposition 2 below states that if the iteration (6) converges, then the resulting allocation is optimal. In this proposition, we use the notation $\tilde{\nabla} U(\mathbf{x}, \epsilon) := \Pi_D(\mathbf{x} + \epsilon \nabla U(\mathbf{x})) - \mathbf{x}$.

Proposition 2. *If $\mathbf{x}^* \in D$ and if there exists an $\epsilon \in (0,1)$ such that $\tilde{\nabla} U(\mathbf{x}^*, \epsilon) = 0$, then $\mathbf{x}^*$ is an optimal allocation of the relaxed problem* (II).

Proof. See B.1. □

Solving (II) using the projected gradient algorithm (6) requires the knowledge of all the future arrivals which may not be available. Further, the horizon T could be potentially large (tens of minutes giving roughly of the order of $300,000$ small-slots). This means the BS will have to solve a very high dimensional problem every 2 ms.

For the heuristics, we circumvent these two issues as follows. First, we solve (II) only for cars that are actually present in the coverage range and ignore the future arrivals. Second, we reduce the computational complexity in two ways: *(i)* we solve the problem over a shorter horizon; and *(ii)* we compute the future allocations on a larger time-scale rather than the short time-scale of channel allocation slots δ, which is usually in the order of a few milliseconds. The distance travelled in δ ms by a vehicle is typically too small to observe large changes in the mean channel conditions. Therefore, we define the notion of a big-slot over which there is noticeable change in the mean channel conditions. For example, a big-slot can be $500 \times \delta$, giving a value of 1 second for the big-slot when $\delta = 2$ ms. The exact value of a big-slot is an adjustable parameter that can be set by the system designer.

Next, we describe the two heuristics.

4.2 Short term objective algorithm (STO1)

Let Δ be the size of the big-slot in absolute time units and let $m = \Delta/\delta$ be the number of small-slots in a big-slot. If $\bar{r}_i(t)$ is the average rate in slot t for user i, then $\bar{\rho}_i(\tau) = \sum_{t=(\tau-1)m+1}^{\tau m} \bar{r}_i(t)$ is the total average data rate that user i will get in big-slot τ if it is allocated in τ. Define $\bar{x}_i(\tau)$ to be the allocation of user i in future big-slot τ. These allocations can be interpreted as the fraction of small-slots that user i will be allocated in the big-slot τ.

The STO1 heuristic works in two steps. At each small-slot t, it first solves the allocation for the current small-slot and the future big-slots. In the second step, it allocates the channel to the user with the largest fractional allocation for the current slot. These steps are described below:

- **Step 1**– In each small-slot t, solve the following optimization problem over a short-term horizon of J big-slots using the projected gradient algorithm as described above:

$$\begin{cases} \text{maximize} & \displaystyle\sum_{i\in\mathcal{U}_b(t)} \log\left(A_i(t-1) + x_i(t)r_i(t) + \sum_{\tau'=\tau}^{\tau+J-1} \bar{x}_i(\tau')\bar{\rho}_i(\tau')\right) \\ \text{s.t.} & \\ & \sum_{i\in\mathcal{U}_b(t)} x_i(t) = 1, \\ & x_i(t) \in [0,1], \quad i \in \mathcal{U}_b(t), \\ & \sum_{i\in\mathcal{U}_b(t)} \bar{x}_i(\tau') = 1, \quad \tau' = \tau \mathinner{..} \tau + J - 1, \\ & \bar{x}_i(\tau') \in [0,1], \quad \tau' = \tau \mathinner{..} \tau + J - 1, i \in \mathcal{U}_b(t). \end{cases} \tag{III}$$

  where τ is the big-slot counted from $t+1$, i.e., it contains m consecutive small-slots $t+1 \mathinner{..} t+m$. The decision variables in Problem (III) are the allocations $x_i(t)$ in the current small-slot, and the allocations $\bar{x}_i(\tau)$ in the future big-slots. Since the future allocations are only computed on the time-scale of big-slots, there is reduction of factor m in the number of variables in (III).

- **Step 2** – The channel of BS b is allocated to (an arbitrary) user $i \in \text{argmax}_{j\in\mathcal{U}_b(t)} x_j(t)$.

The computational complexity of this heuristic is mainly in Step 1, which requires in the order of $20 \times (J+1) \times |\mathcal{U}_b(t)| \times \log(|\mathcal{U}_b(t)|)$ operations to complete (as we limit the number of iterations of the projected gradient algorithm to 20).

4.3 Short term objective algorithm 2 (STO2)

In STO2, we further reduce the complexity by recomputing the future allocations only at the beginning of a big-slot. The future allocation computed is thus used until the end of this big-slot. If one new user arrives to the system in the middle of a big-slot, we just ignore it for this big-slot and wait until the beginning of next big-slot to update the state. Once the allocations for future big-slots are computed, then in every small-slot of this big-slot, we apply an index-based policy as in (4).

The steps for STO2 are:

- **Step 1** – At the beginning of each big-slot τ, that is at time $t = \tau m + 1$, solve the following problem using the projected gradient algorithm:

$$\begin{cases} \text{maximize} & \displaystyle\sum_{i \in \mathcal{U}_b(t)} \log \left(A_i(t-1) + \sum_{\tau'=\tau}^{\tau+J-1} \bar{x}_i(\tau') \bar{\rho}_i(\tau') \right) \\ \text{s.t.} & \\ & \sum_{i \in \mathcal{U}_b(t)} \bar{x}_i(\tau') = 1, \quad \tau' = \tau, \ldots, \tau + J - 1, \\ & \bar{x}_i(\tau') \in [0,1], \quad i \in \mathcal{U}_b(t), \tau' = \tau, \ldots, \tau + J - 1. \end{cases} \tag{IV}$$

- **Step 2** – Inside a big-slot, in each small-slot s, compute $M_i(s)$ as in (4) where the future allocation $\hat{\mathbf{x}}$ is the solution $\bar{\mathbf{x}}$ of (IV), and then allocate the channel to the user i maximizing $M_i(s)$.

Note that Step 1 in STO2 is computed only once every big-slot unlike m times in every big-slot as in STO1. By doing this, we further reduce the number of computations almost by a factor m.

5 Extension to joint channel allocation and power control

We now show how the heuristics can be extended for joint channel allocation and power control. The throughput of user $i \in \mathcal{U}_b(t)$ is now computed according to the Shannon formula

$$r_i(t) = x_i(t) \log \left(1 + \frac{\gamma_i(t)\, p_i(t)}{x_i(t)} \right), \tag{7}$$

where $x_i(t)$ is the fraction of the channel assigned to user i in slot t by base station b, and $p_i(t)$ is the power with which the BS transmits to user i. In (7), $\gamma_i(t)$ represents the channel gain of user i at time t. As before, we do not make explicit the dependence of $r_i(t)$ on the decision variables $x_i(t)$ and $p_i(t)$ in order to keep the notation light.

We assume that each BS has an average transmit power constraint of $\bar{P}$ over the horizon T, and that the maximum transmit power in each time slot is P_{max}. The objective of each BS is now to choose the power and the channel allocation so as to maximize the total utility of the K users over a horizon of T time slots:

$$\text{maximize} \sum_{i=1}^{K} \log\left(\frac{1}{T}\sum_{t=1}^{T} r_i(t)\right) \tag{V}$$

$$\text{s.t.} \tag{8}$$

$$\sum_{i\in\mathcal{U}_b(t)} x_i(t) = 1, \quad b \in \mathcal{B},\ t = 1,..,T, \tag{9}$$

$$\frac{1}{T}\sum_{t}\sum_{i\in\mathcal{U}_b(t)} p_i(t) \le \bar{P}, \quad b \in \mathcal{B}, \tag{10}$$

$$\sum_{i\in\mathcal{U}_b(t)} p_i(t) \le P_{max}, \quad b \in \mathcal{B},\ t = 1,..,T, \tag{11}$$

$$x_i(t) \in [0,1], \quad i \in \mathcal{U}_b(t), t = 1,\ldots,T. \tag{12}$$

Here (10) is the average power constraint, while (11) is the maximum power constraint. Note that in (V) we have allowed for fractional channel allocations. If the system imposes a binary constraint, that is only one user on one channel in any given slot, then these constraints can be imposed as well as in the algorithms we propose. In our experiments, we observed however that allocations were mostly binary. So, we expect the qualitative conclusions to be valid whether allocations are binary or not.

Remark 3 (Maximum power constraint)**.** *For conciseness, we shall not write the maximum power constraint explicitly in the optimization problems that we will define from now on. This constraint will be implicit and assumed to be applicable in all slots.*

In contrast to the current literature, which is based solely on the channel gains in the current time slot, we can extend our heuristics so as to exploit also the predicted values of the mean channel gains in the future slots.

5.1 Short term objective 2 (STO2) with power control

We will only give the STO2 version of the algorithm since it is computationally less expensive and performs well. The STO1 version can also be formulated along similar lines, if needed.

As before, a big-slot is m consecutive time-slots. For a non-stationary and slowly varying model in which, for each vehicle i, the mean of the channel gains varies on a slow time-scale, the preferred value of m will be the number of time-slots during which mean channel gain remains constant. For example, for a time-slot of 2 ms, the means may vary every 200 ms, which yields $m = 100$ slots. In contrast, in a mobility model where cars can come and leave, the channel gains $\gamma_i(t)$ are non-stationary and their means change every slot. In this case, the value of m can be set by the system designer depending on how fast the mean channel gains vary.

Let $\tau_t \in \{1, \ldots, T\}$ be the big-slot to which time-slot t belongs to. The number of slots remaining in big-slot τ_t after (but not including) time t is then $\theta_t = (\tau_t + 1)m - t$.

We shall use the notation $\hat{x}$ for a quantity that is computed over a big-slot. For example $\hat{p}_i(\tau)$ will denote the power used in all the slots inside big-slot τ. Similarly,

$$\hat{r}_i(\tau) = m \cdot \hat{x}_i(\tau) \log\left(1 + \frac{\bar{\gamma}_i(\tau)\hat{p}_i(\tau)}{\hat{x}_i(\tau)}\right) \tag{13}$$

is the total rate obtained by vehicle i in big-slot τ when it is served a fraction $\hat{x}_i(\tau)$ of time at a transmit power of $\hat{p}_i(\tau)$. Note that here the rate is computed assuming that the channel gain is its mean value in big-slot τ. With a slight abuse of notation,

$$\hat{r}_i(\tau_t) = \theta_t \hat{x}_i(\tau_t) \log\left(1 + \frac{\bar{\gamma}_i(\tau_t)\hat{p}_i(\tau_t)}{\hat{x}_i(\tau_t)}\right), \tag{14}$$

shall denote the total rate in the remaining slots in current big-slot τ_t.

Recall that STO2 recomputes the allocations and powers of the future big-slots only at the beginning of each big-slot and over a short-term horizon of J. Inside a big-slot, it computes only the solution for the current slot assuming the solution for the future big-slots to be the same as that computed at the the start of the current big-slot. The power control version of this algorithm operates in two steps.

- **Step 1** – If $t \equiv 1 \pmod m$, i.e., at the beginning of each big slot, BS b first maximizes

$$\sum_{i \in \mathcal{U}_b(t)} \log\left(\sum_{s=1}^{t-1} r_i(s) + \sum_{\tau=\tau_t}^{\tau_t+J-1} \hat{r}_i(\tau)\right) \tag{STO2-Big}$$

subject to

$$[\hat{x}_i(\tau)] \in \mathcal{S},\ i \in \mathcal{U}_b(t), \tau = \tau_t, \ldots, \tau_t + J - 1, \tag{15}$$

$$m \sum_{i \in \mathcal{U}_b(t)} \hat{p}_i(\tau) \leq \bar{P},\ \tau = \tau_t, \ldots, \tau_t + J - 1, \tag{16}$$

where $\mathcal{S}$ is the set of feasible solutions of problem (IV). The variables in this problem are $[\hat{x}_i(\tau)]$ and $[\hat{p}_i(\tau)], \tau = \tau_t, \ldots, \tau_t + J - 1$, only for the cars in the scheduling area of BS b. The constraint (16) ensures that the average power does not exceed $\bar{P}$ in each big-slot.

Remark 4. *The average power constraint* (16) *is stricter than* (10) *since it is imposed in each big-slot and not over the remaining horizon. This is done because otherwise the algorithm does not necessarily use up all its power budget. Depending upon the scenario, it may leave quite some power budget for the future big-slots and not end up using it if enough cars do not arrive in the future or the channel conditions were not favorable. Therefore, to ensure that STO2 uses all the power budget, constraint* (16) *is imposed in each big-slot.*

- **Step 2** – Next, in each small-slot t, the optimal allocation and transmit power are computed for t as well as for the remaining small-slots in the current big-slot assuming that the allocations and transmit powers in the future big-slots are those computed from solving (STO2-Big). That is, in slot t, STO2 maximizes

$$\sum_{i\in\mathcal{U}_b(t)} \log\left(\sum_{s=1}^{t-1} r_i(s) + r_i(t) + \sum_{\tau=\tau_t}^{\tau_t+J-1} \hat{r}_i(\tau)\right) \tag{STO2-Small}$$

subject to

$$[x_i(t)] \in \mathcal{S},\ [\hat{x}_i(\tau_t)] \in \mathcal{S}; \tag{17}$$

$$\sum_{i\in\mathcal{U}_b(t)} (p_i(t) + \theta_t \hat{p}_i(\tau_t)) \le P_t^{(b)}, \tag{18}$$

where $P_t^{(b)}$ is the remaining power budget of BS b in the current big-slot. The variables in this problem are $[x_i(t)]$, $[p_i(t)]$, $[\hat{x}_i(\tau_t)]$ and $[\hat{p}_i(\tau_t)]$.

In each small-slot, STO2 solves a lower dimensional problem with $4|\mathcal{U}_b(t)|$ variables – $2|\mathcal{U}_b(t)|$ for the current small-slot and $2|\mathcal{U}_b(t)|$ for the remaining small-slots in τ_t. This is twice the number that would be required when not using future information. So, the complexity induced by optimizing over future information is not excessive.

6 Numerical results

This section compares the utility of the proposed heuristics with PF-EXP, $(\text{PF})^2$S and a greedy algorithm. For the $(\text{PF})^2$S the future allocation was done using the round robin algorithm. Two different simulation setups will be used. In the first one in Section 6.1, vehicles move along a single road served by one BS. The simulations for this scenario are performed with Python. The second set of scenarios, presented in Section 6.2, consists of three road networks with multiple BSs. The simulations for these scenarios are done using SUMO which simulates actual vehicle driving behaviour.

Before presenting the scenarios, we first introduce the performance metric. Recall that the proportional-fair utility is defined in (I). Let

$$U^A = \sum_{i=1}^{K} \log\left(\sum_{t=1}^{T} \phi_i(t)\right),$$

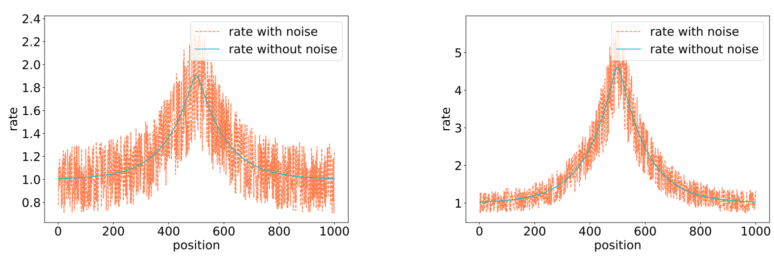
be the total utility or the reward of algorithm A and by $\bar{U}^A = \frac{1}{K}U^A$ its average utility (reward) over K users. A comparison of two algorithms A and B using the relative error is not appropriate because of the logarithm in the utility. Note that when a different unit of measure is used for the data rate (for example, bits instead of megabits), the ratio U^A/U^B changes as well. In order to have a comparison that is independent of the units of measure, it is more appropriate to use the difference $U^A - U^B$, or, for convenience, $\exp(\bar{U}^A - \bar{U}^B)$ which gives a non-negative value with 0 meaning that the utility of B is vastly superior to A, and ∞ meaning that it is the other way round. As the performance measure, we shall use the percentage improvement of algorithm A over B given by $(\exp(\bar{U}^A - \bar{U}^B) - 1) \cdot 100\%$.

6.1 Single road and homogeneous vehicle velocities

In the first set of simulations, there is only one base station and one straight road in the scheduling area. 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 $z = 500$ m. The data rate at position z along the road is given by:

$$r(z) = \eta \cdot (1 + \kappa \exp(|z - 500|/\sigma), \tag{19}$$

where $\kappa \geq 0$ is a real number and η is uniform random variable whose range will be in $[0.7, 1.3]$ unless stated otherwise. A sample path of $r(z)$ is shown in Figure 2 for two different values of κ. This function has the highest mean at the mid-point of the segment and the lowest mean at the two end points. We chose the above rate function for convenience and emphasize that the algorithm itself is independent of the rate function.



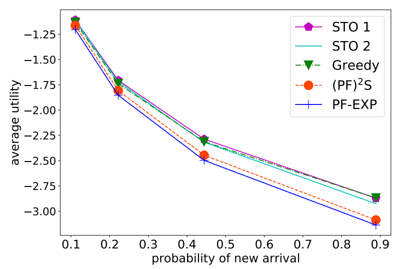
(a) $\kappa = 1$. (b) $\kappa = 4$.

Figure 2: Sample path of data rate at various positions along the road. $\sigma = 100$, and $\eta \sim U[0.7, 1.3]$.

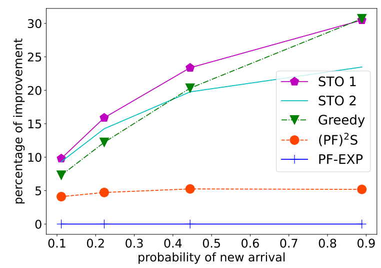
The time horizon T was $4,000,000$ small-slots which corresponds to 8000 seconds (slightly more than two hours). The big-slot length Δ for our algorithms was taken as 1 second or equivalently 500 small-slots.

Vehicles enter the road from the left edge, move with the same velocity $v = 25$ m/s, and leave from the right edge. This gives $N = 20,000$ spatial small-slots in the scheduling area and $J = 40$ big-slots. The arrival of new cars on the left edge is assumed to happen with probability p in every second.

Figure 3a shows the average utility obtained by a vehicle for each of the four algorithms as a function of the arrival probability p. When p is small it is natural for all the algorithms to give similar rewards since there is rarely more than one car at a time in the scheduling area. The difference becomes more apparent at higher values of p when there are more cars competing for the channel. Figure 3b shows the percentage improvement of the three other algorithms compared to PF-EXP. Both STO1 and STO2 do better than PF-EXP and more importantly better than $(\mathrm{PF})^2\mathrm{S}$. The greedy algorithm does well in this scenario mainly because all vehicles move along the same road and observe statistically identical but position-dependent radio conditions during their stay. As mentioned previously, greedy is not practically implemented because it can be very unfair to users that have heterogeneous rates.



(a) Average utility per car.



(b) Percentage of improvement over PF-EXP.

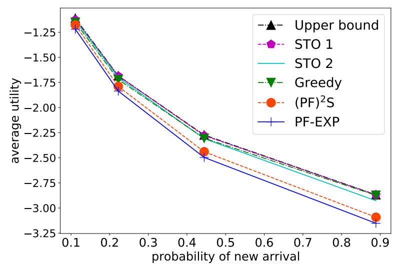
Figure 3: Comparison of algorithms for homogeneous vehicle velocities.

Comparison with the upper bound: before considering more complex scenarios, we compare our algorithms to the upper bound which can be obtained by solving the relaxed problem (II) assuming that all the future arrivals as well as future data rates are known exactly and given as input. Solving with such assumptions is infeasible in practice. However, purely for comparison purposes, the upper bound can show how much we lose due to lack of precise information. Since the problem size of (II) can grows quickly with the parameters, we restrict the road length to $L = 100$ m, and reduce the time-horizon to $T = 500$ s. The big-slot Δ is shortened to 0.1 s so that $J = 40$ big-slots as before. The rate curve remains the same as for the previous setting.

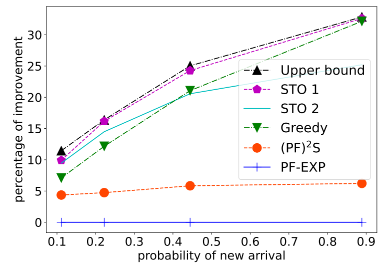
Figures 4a and 4b plot the average utility per car and percentage improvement with respect to PF-EXP for the four algorithms as well as for the upper bound. It can be observed that the proposed algorithm is quite close to the upper bound in this scenario which means that average data rates provide a good estimate of the actual data rates.

6.2 Network simulation with SUMO

Simulation of Urban MObility application (SUMO) [10] is an open source software designed for simulating mobility of vehicles in large traffic networks. One of the features of this simulator is that we can import maps of different cities and simulate realistic mobility traces. We use this application to simulate the



(a) Small setting: average utility per car.



(b) Small setting: percentage of improvement over PF-EXP.

Figure 4: Comparison of algorithms against the upper bound from the solution of (II) for homogeneous vehicle velocities.

complex driving dynamics in a simple scenario and in two specific regions of Toulouse city to have an objective comparison of our heuristics against existing algorithms in realistic scenarios.

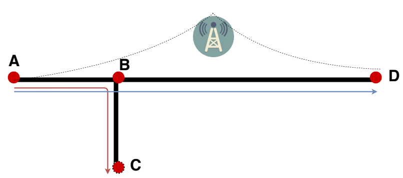
The performance evaluation of heuristics in done in two steps. In the first step SUMO is used for generating the mobility traces of vehicles. These traces are then fed to a Python script which implements the different heuristics and computes the value of the objective function.

In the following, unless otherwise specified, the parameters are chosen as follows: $\kappa = 4, \sigma = 100$, big-slot $\Delta = 1s$ and the short-term horizon J is the maximal remaining sojourn time of the users that are currently inside the system. We calculate the allocation plan every one second. From now on, we do not include STO 1 in the comparison because STO 1 takes much longer time to run and may not be computationally interesting on small time-scales. Also, we also do not show the performance of the greedy algorithm here since some users may starve in a greedy allocation leading to a value of $-\infty$.

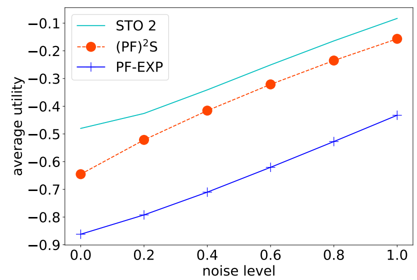
6.2.1 A simple network with 1 BS

Let us consider the network presented in Figure 5a. There are two classes of users: one that arrives from A then moves along the long road to B and D (the blue one), and another one that arrives from A then moves to B and then to C (the red one). The base station is placed at location $(20, 40)$, A is placed at $(-94, 40)$, B is placed at $(-34, 39)$, C is placed at $(-35, -18)$ and D is placed at $(96, 35)$, so the length from A to D in this simple network is 190 m. The simulation duration was 1 hour 9 minutes and 25 seconds with a total of 1598 users passing through during this time.

If we apply the greedy heuristic in this situation, then many users of the second class are never allocated the channel assuming that when the second class leaves the scheduling area it does not get close to the BS of the next cell. This is the reason why we do not evaluate the performance of the greedy algorithm for this scenario.



(a) A simple road network with two classes of users.



(b) Comparison of the utilities obtained with STO2, $(\text{PF})^2$S and PF-EXP for the scenario of Fig. 5a.

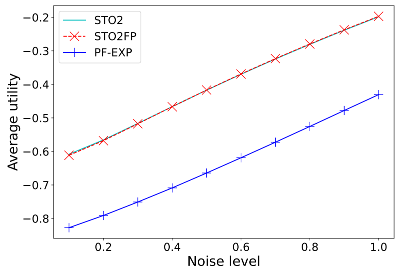
Figure 5: Simple netwokr scenario and its average utility.

Figure 5b shows the numerical results for this case. In this scenario, it was observed that PF-EXP always gives priority to the new arrivals no matter what the initial value is. This leads to a higher sub-optimality of PF-EXP since the other heuristics focus on users that are closer to the base station and have a higher quality channel.

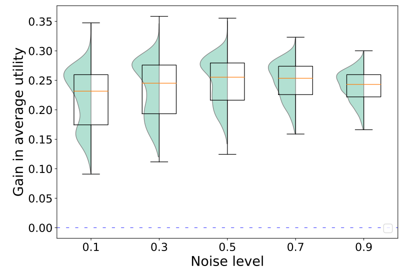
Power control

We redo the above experiment for the algorithms with power control. The average power level, $\bar{P}$ is taken to be 1 and $P_{max} = 5$. The horizon $J = 5$ big-slots and $\Delta = 1$ sec. The rate curve is taken such that the rate when transmitting at $\bar{P}$ gives the rate curve of the scenario without power control. Fig. 6a shows the average utility for STO2, STO2-FP which is STO2 but with a fixed power level of $\bar{P}$ in each small-slot, and PF-EXP again with a fixed power level of $\bar{P}$ in each small-slot. Gains in utility (and the throughput) can be achieved by using prediction of the future average rates. There is no discernible gain achieved by power control compared to the fixed power algorithm. The gains will of course depend upon the parameters of the scenario. One such scenario will be shown later.

Fig. 6b shows the statistics of the gain in average utility obtained by STO2 with respect to that of PF-EXP. Only alternate noise values are shown so as not to clutter the plot. The box plot shows the median and the 25 and 75 percentiles and violin plot shows how the values are distributed. The points above the horizontal line at 0 are for vehicles that improved their utility under STO2 when compared to that under PF-EXP while any point under this line indicates vehicles that see a decrease in utility under STO2. In this scenario, only two vehicles out of 1600 did not benefit by the use of STO2.



(a) Average utility

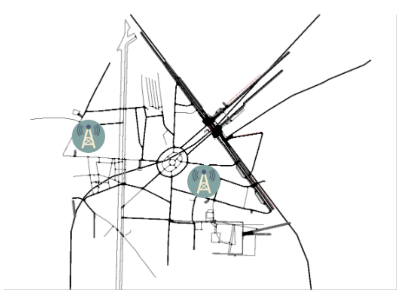


(b) Gain in average utility of STO2 with respect to PFexp. Outliers have been omitted.

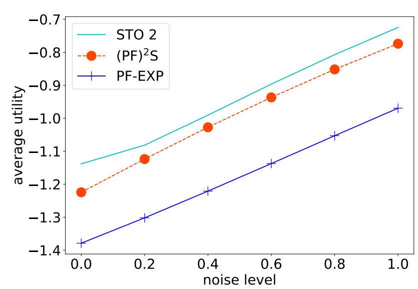
Figure 6: Simple network with power control

6.2.2 Place Wilson (Toulouse) scenario with 2 BSs

In this scenario, we evaluate the algorithms on users moving in the Place Wilson area of Toulouse with two BSs as shown in Fig. 7a. The average utilities of the different heuristics are shown in Fig. 7b. The various parameters for the rate function are the same as those indicated at the beginning of this section. It took 219 seconds, 229 seconds, 433 seconds and 833 seconds respectively to run greedy, PF-EXP, $(PF)^2S$ and STO2 for simulating 1.07 hours of traffic with 483 users (including cars, buses, and bicycles). The staying times of the users varied from $2s$ to $361s$. We do not show greedy in the utility comparison since there were several starving users in this case. As expected, there is a trade-off between the quality of the solution and the computation time. STO2 takes longer to solve but gives a better allocation.

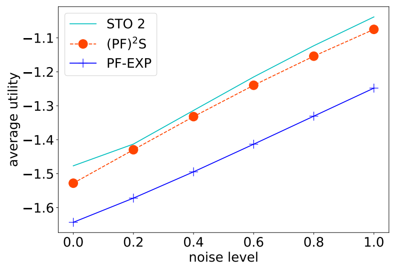


(a) Place Wilson, Toulouse with two 4G BSs.

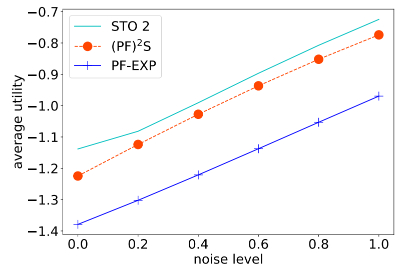


(b) Comparison of the utilities obtained with STO2, $(PF)^2S$ and PF-EXP for the scenario of Fig. 7a.

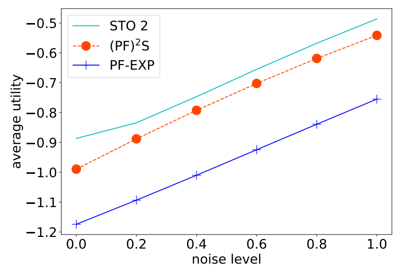
Figure 7: Place Wilson scenario and its average utility.



(a) $\kappa = 2$.



(b) $\kappa = 4$.



(c) $\kappa = 6$.

Figure 8: Comparison of the utilities obtained with STO2, (PF)^{2}S and PF-EXP for the scenario of Fig. 7a and for different values of κ.

Now, we change some of the parameters to see how the performance of the heuristic is influenced by these parameters.

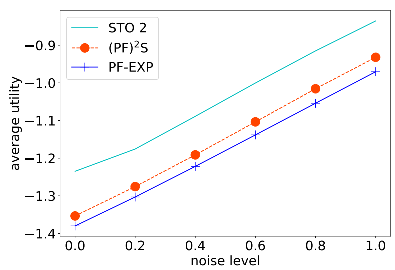
Figure 8a, 8b, 8c plot the average utilities for different values of κ with the same J and Δ. The gap between STO2 and (PF)^{2}S become larger when κ increases.

Figure 9a, 9b, 9c and 9d illustrate the average utilities for different values of J with the same values of κ and Δ. Remark that we assume (PF)^{2}S and STO2 use the same information, so in (PF)^{2}S the future information is estimated until J as well. It is seen that the more information we have, the better (PF)^{2}S and STO2 perform.

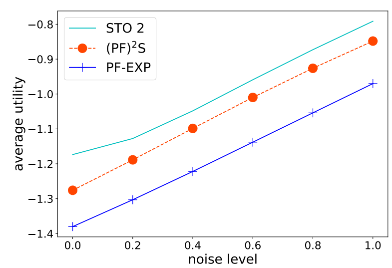
Figure 10a, 10b and 10c show the influence of the size Δ of the big-slot, assuming the same values of J and κ. The performance of STO2 is almost the same for these different values of Δ but the running time is significantly shorter with the largest value.

Power control

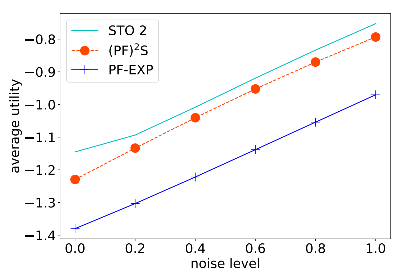
As for the previous scenario, we redo the above experiment for the algorithms with power control. The average power level, $\bar{P}$ is taken to be 1 and $P_{max} = 5$. The horizon $J = 5$ big-slots and $\Delta = 1$ sec. The rate curve is taken such that the



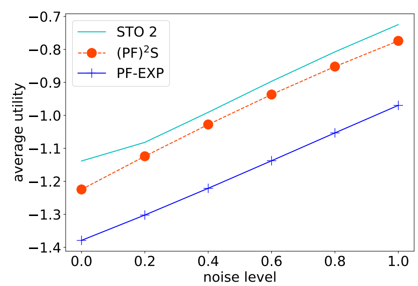
(a) $J = 20s$.



(b) $J = 60s$.

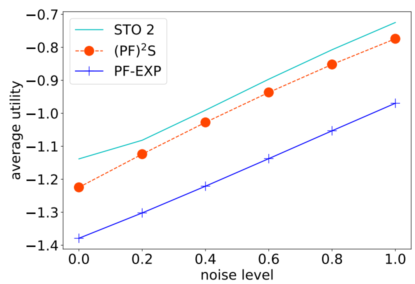


(c) $J = 120s$.

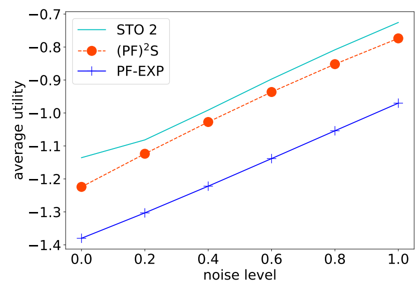


(d) $J =$ maximum sojourn time in term of big-slot of all users inside the system.

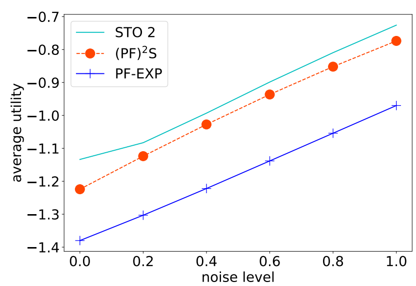
Figure 9: Comparison of the utilities obtained with STO2, $(\text{PF})^2$S and PF-EXP for the scenario of Fig. 7a and for different time horizon J.



(a) $\Delta = 1s$ ($833s$ to run STO2).



(b) $\Delta = 2s$ ($568s$ to run STO2).

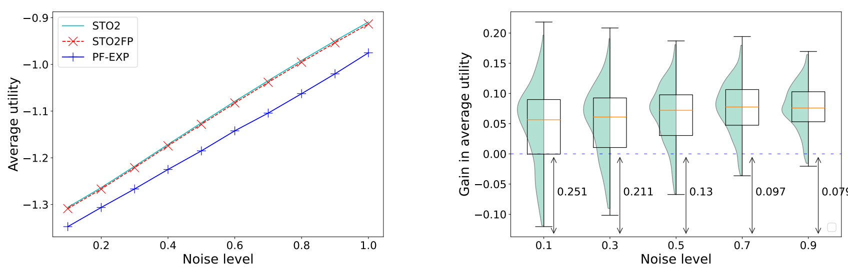


(c) $\Delta = 4s$ ($438s$ to run STO2).

Figure 10: Comparison of the utilities obtained with STO2, $(\mathrm{PF})^2$S and PF-EXP for the scenario of Fig. 7a and for different values of Δ. The performance are almost the same, but the running time is much shorter when increasing the big-slot size.

rate when transmitting at $\bar{P}$ gives the rate curve of the scenario without power control. Fig. 11a shows the average utility for STO2, STO2-FP. As before, gains in utility (and the throughput) can be achieved by using prediction of the future average rates. The gains achieved by power control again are negligible compared to those by prediction.

Fig. 11b shows box plot and the violin plot for the gain in average utility of STO2 with respect to PF-EXP. The number next to the double arrowed line is the fraction of vehicles that see a decrease in their utility under STO2. Unlike in the previous scenario, this time, a non negligible proportion of vehicles see a decrease in their utility. For example, 27% of vehicles saw their utility decrease for noise level of 0.1 while this number was 8% for noise level of 0.9. However, overall STO2 improves the average utility with a median of +0.05 units of utility, that is at least half of the vehicles gained 0.05 units of utility.



(a) Average utility

(b) Gain in average utility of STO2 with respect to PFexp. Outliers have been omitted.

Figure 11: Place Wilson scenario with power control

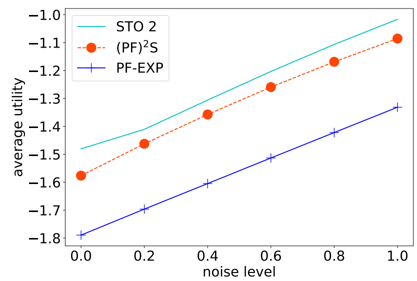
6.2.3 Jardin de Plantes (Toulouse) scenario with 4 BSs

In the final set of simulations, we take another area of Toulouse called Jardin des Plantes with four BSs as shown in Fig. 12a. The average utilities for this scenario are plotted in Fig. 12b. It took 976 seconds, 1009 seconds, 1502 seconds and 3403 seconds to run greedy, PF-EXP, $(\mathrm{PF})^2\mathrm{S}$ and STO2 for 1.05 hours of traffic with 740 users (including cars, buses, motorbikes, bicycles and pedestrian). Again, we do not include the greedy algorithm in the utility comparison since there are several starving users in this case.

Power control

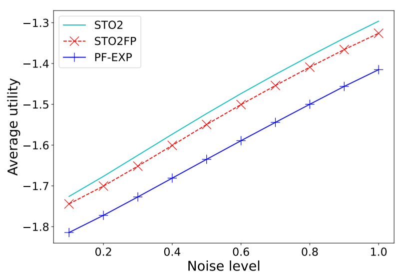
The plots for this scenario with power control are shown in Fig. 13. Here, power control does bring improvements over the fixed-power STO2 with respect to the previous two scenarios. Compared to the Place Wilson scenario, STO2 has a smaller fraction of vehicles that are worse off than with PF-EXP.

(a) Jardin de Plantes, Toulouse with four 4G BSs.

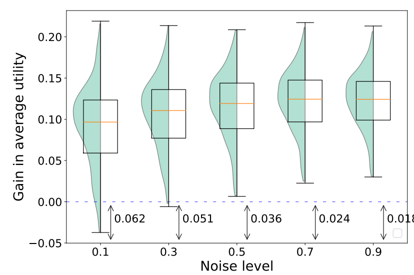


(b) Comparison of the utilities obtained with STO2, (PF)^{2}S and PF-EXP for the scenario of Fig. 12a.

Figure 12: Jardin des Plantes scenario and its average utility.



(a) Average utility



(b) Gain in average utility of STO2 with respect to PFexp. Outliers have been omitted.

Figure 13: Jardin des Plantes with power control

7 Conclusions and future work

We proposed two heuristics that use future mean channel gain information to improve the utility of users in cellular networks. In order to reduce the computational complexity, they solve the problem over a shorter time horizon as well as on a larger time-scale. It was shown on numerical experiments carried out on traces generated from realistic mobility patterns that these heuristics give better utility compared to PF as well as (PF)^{2}S algorithms.

There are several directions in which this work can be taken. One avenue is to implement a centralized and coordinated version of these heuristics. Further, it will be interesting to design heuristics for networks in which there are a fractions of users whose future channel gain information is not known. These could be users who do not share their mobility information or users like pedestrians whose mobility can be random and hence not known exactly.

Other directions of research include designing heuristics for different utility functions and QoS requirements such are latency and jitter. On the analytical side, obtaining heuristics with guaranteed sub-optimality bounds will be worth investigating.

A Projection on feasible set D

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

$$D_t = \left\{ \mathbf{x}_t = (x_{i,t})_{i=1..K} \in [0,1]^K \ : \ \sum_{i=1}^{K} x_{i,t} = 1 \right\},$$

for all $t = 1, 2, ..., J$.

Since the sets D_t are simplexes, we can compute the projection on D_t following [6]. The projection on D can thus be computed with the simple following lemma.

Lemma 5. *If* $\mathbf{y} = (y_{i,t})_{i=1..K,t=1..J} \in \mathbb{R}^{K \times J}$, *then*

$$\Pi_D(\mathbf{y}) = \Pi_{D_1}(\mathbf{y}_1) \times \Pi_{D_2}(\mathbf{y}_2) \times \cdots \times \Pi_{D_J}(\mathbf{y}_J),$$

where $\mathbf{y}_t = (y_{i,t})_{i=1..K}$.

Proof. Denote by $\mathbf{z} = \Pi_{D_1}(\mathbf{y}_1) \times \Pi_{D_2}(\mathbf{y}_2) \times \cdots \times \Pi_{D_J}(\mathbf{y}_J)$. It is easy to check that for any $\mathbf{x} \in D$, we have $\langle \mathbf{y} - \mathbf{z}, \mathbf{x} - \mathbf{z} \rangle \leq 0$. □

As described in [6], the worst-case complexity of computing Π_{D_t} is $O(K^2)$, but is observed to be equal to $K \times \log(K)$ in practice. Therefore the complexity of computing the projection on $D = D_1 \times D_2 \cdots \times D_J$ is expected to be $J \times K \times \log(K)$ in practice.

B Proofs

B.1 Proof of Proposition 2

Proposition 2. The point $\mathbf{x}^*$ is an optimal solution if for any $\mathbf{x} \in D$,

$$\nabla U(\mathbf{x}^*) \cdot (\mathbf{x}^* - \mathbf{x}) \leq 0.$$

From Lemma 5, it follows that it is sufficient to prove the above inequality on D_t for an arbitrary $t \in \{1, .., J\}$. Assuming U is a convex function on D_t, we shall prove that if $\mathbf{x}_t^* = (x_{i,t}^*)_{i=1,..,K} \in D_t$ satisifes

$$\Pi_{D_t}\left(\mathbf{x}_t^* + \epsilon \nabla U(\mathbf{x}_t^*)\right) = \mathbf{x}_t^* \tag{20}$$

for some $\epsilon > 0$, then

$$\nabla U(\mathbf{x}_t^*) \cdot (\mathbf{x}_t^* - \mathbf{x}_t) \geq 0, \text{ for any } \mathbf{x}_t \in D_t,$$

i.e, $\mathbf{x}_t^*$ is global optimum of U. Indeed, without loss of generality, we assume that

$$x_{1,t}^* + \epsilon \frac{\partial U}{\partial x_{1,t}^*} \geq x_{2,t}^* + \epsilon \frac{\partial U}{\partial x_{2,t}^*} \geq \ldots \geq x_{n,t}^* + \epsilon \frac{\partial U}{\partial x_{n,t}^*} \geq x_{K,t}^* + \epsilon \frac{\partial U}{\partial x_{K,t}^*},$$

where n is the largest index such that

$$\frac{1}{n} \sum_{i=1}^{n} \left(x_{i,t}^* + \epsilon \frac{\partial U}{\partial x_{i,t}^*} - 1 \right) \leq x_{n,t}^* + \epsilon \frac{\partial U}{\partial x_{n,t}^*}.$$

Define $\tau = \frac{1}{n} \sum_{i=1}^{n} \left(x_{i,t}^* + \epsilon \frac{\partial U}{\partial x_{i,t}^*} - 1 \right)$. By Proposition 10 in [6] we have:

$$\Pi_{D_t}(\mathbf{x}_t^* + \epsilon \cdot \nabla U(\mathbf{x}_t^*)) = \left(x_{1,t}^* + \epsilon \frac{\partial U}{\partial x_{1,t}^*} - \tau, x_{2,t}^* + \epsilon \frac{\partial U}{\partial x_{2,t}^*} - \tau, ..., x_{n,t}^* + \epsilon \frac{\partial U}{\partial x_{n,t}^*} - \tau, 0, ..., 0 \right).$$

Comparing term by term with (20), we get:

1. $x_{n+1,t}^* = \cdots = x_{K,t}^* = 0$,
2. $x_{n+1,t}^* + \epsilon \frac{\partial U}{\partial x_{n+1,t}^*} \leq \tau, \cdots, x_{K,t}^* + \epsilon \frac{\partial U}{\partial x_{K,t}^*} \leq \tau$. It thus follows from the first item that $\epsilon \frac{\partial U}{\partial x_{n+1,t}^*} \leq \tau, \cdots, \epsilon \frac{\partial U}{\partial x_{K,t}^*} \leq \tau$,
3. $\epsilon \frac{\partial U}{\partial x_{1,t}^*} = \cdots = \epsilon \frac{\partial U}{\partial x_{n,t}^*} = \tau$.

It yields

$$\begin{aligned}
\epsilon \nabla U(\mathbf{x}_t^*) \cdot (\mathbf{x}_t^* - \mathbf{x}_t) &= \sum_{i=1}^{K} \epsilon \frac{\partial U}{\partial x_{i,t}^*} (x_{i,t}^* - x_{i,t}) \\
&= \sum_{i=1}^{n} \epsilon \frac{\partial U}{\partial x_{i,t}^*} (x_{i,t}^* - x_{i,t}) + \sum_{i=n+1}^{K} \epsilon \frac{\partial U}{\partial x_{i,t}^*} (x_{i,t}^* - x_{i,t}), \\
&= \sum_{i=1}^{n} \epsilon \tau (x_{i,t}^* - x_{i,t}) + \sum_{i=n+1}^{K} \epsilon \frac{\partial U}{\partial x_{i,t}^*} (x_{i,t}^* - x_{i,t}), \\
&= \sum_{i=1}^{K} \epsilon \tau x_{i,t}^* - \sum_{i=1}^{K} \epsilon \tau x_{i,t} + \sum_{i=n+1}^{K} \left(\epsilon \frac{\partial U}{\partial x_{i,t}^*} - \tau \right) (x_{i,t}^* - x_{i,t}), \\
&= \tau - \tau + \sum_{i=n+1}^{K} \left(\epsilon \frac{\partial U}{\partial x_{i,t}^*} - \tau \right) (0 - x_{i,t}), \\
&\geq 0.
\end{aligned}$$

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

B.2 Proof of Proposition 1

Proposition 1. In fact the condition (5) implies that $\tilde{\nabla}U(\mathbf{x}^*) = 0$, which yields the proof according to Proposition 2. □

References

[1] H. Abou-zeid, H. S. Hassanein, and N. Zorba. Long-term fairness in multi-cell networks using rate predictions. In *2013 7th IEEE GCC Conference and Exhibition (GCC)*, pages 131–135, Nov 2013.

[2] H. J. Bang, T. Ekman, and D. Gesbert. Channel predictive proportional fair scheduling. *IEEE Transactions on Wireless Communications*, 7(2):482–487, February 2008.

[3] 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.

[4] S. Borst, N. Hegde, and A. Proutiere. Mobility-driven scheduling in wireless networks. In *IEEE INFOCOM 2009*, pages 1260–1268, April 2009.

[5] 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.

[6] Laurent Condat. Fast projection onto the simplex and the l1 ball. *Math. Program.*, 158:575–585, 2016.

[7] Jianwei Huang, Vijay Subramanian, Rajeev Agrawal, and Randall Berry. Downlink scheduling and resource allocation for ofdm systems. *IEEE Transactions on Wireless Communications*, 8:288–296, 01 2009.

[8] Frank Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(1):33–37, 1997.

[9] 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.

[10] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018.

[11] Robert Margolies, Ashwin Sridharan, Vaneet Aggarwal, Rittwik Jana, N. K. Shankaranarayanan, Vinay A. Vaishampayan, and Gil Zussman. Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms. *IEEE/ACM Trans. Netw.*, 24(1):355–367, February 2016.

[12] A. Mesbah. Stochastic model predictive control: An overview and perspectives for future research. *IEEE Control Systems Magazine*, 36(6):30–44, Dec 2016.

[13] N. Nguyen, O. Brun, and B. Prabhu. An algorithm for improved proportional-fair utility for vehicular users. *The 25th International Conference on Analytical and Stochastic Modelling Techniques and Applications ASMTA-2019*, May 2019.

[14] Thi Thuy Nga Nguyen. *Proportional-Fair Scheduling of Mobile Users based on a Partial View of Future Channel Conditions*. Theses, INSA de Toulouse, November 2020.

[15] Thi Thuy Nga Nguyen, Olivier Brun, and Balakrishna Prabhu. Joint downlink power control and channel allocation based on a partial view of future channel conditions. In *The 15th Workshop on Resource Allocation, Cooperation and Competition in Wireless Networks (RAWNET)*, Volos, Greece, June 2020. RAWNET Workshop was part of The 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt 2020).

[16] Vijay G. Subramanian, Randall A. Berry, and Rajeev Agrawal. Joint scheduling and resource allocation in CDMA systems. *IEEE Trans. Inf. Theory*, 56(5):2416–2432, 2010.

[17] 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.

[18] Shengfeng Xu, Gang Zhu, Chao Shen, Yan Lei, and Zhangdui Zhong. Utility-based resource allocation in high-speed railway wireless networks. *EURASIP Journal on Wireless Communications and Networking*, 2014(1):68, Apr 2014.

[19] Yung Yi and Mung 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.

[20] 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.

Acknowledgment

The authors would like to thank Continental Digital Service in France for supporting partially of this work.