Power Control with Random Delays: Robust Feedback Averaging

Andrew Ward, Daniel Miller, Zhengyuan Zhou, Panayotis Mertikopoulos, and Nicholas Bambos

Abstract— **Distributed power control schemes in wireless networks have been well-examined, but standard methods rarely consider the effect of potentially random delays, which occur in almost every real-world network. We present Robust Feedback Averaging, a novel power control algorithm that is capable of operating in delay-ridden and noisy environments. We prove optimal convergence of this algorithm in the presence of random, time-varying delays, and present numerical simulations that indicate that Robust Feedback Averaging outperforms the ubiquitous Foschini-Miljanic algorithm in several regimes.**

I. Introduction

Proper management of wireless communication networks relies on efficient and robust power control strategies. Such strategies may be broadly categorized into either centralized or distributed paradigms. While centralized methods may allow for more globally optimal theoretical results, this is counterbalanced by an increased communication requirement for implementation, and a reduction in system robustness due to the presence of single total failure point. In contrast, divesting the control into a distributed scheme alleviates these concerns and opens up the study of both cooperative and competitive optimal decentralized control strategies [1], [2]. Such strategies have been well-examined, and solutions for the fundamental problems have been established.

One foremost power control method in such a distributed wireless network was developed by Foschini and Miljanic (FM) [3]. Since its development, the FM algorithm has proved applicable and stable in far more general environments than originally considered [4]. As a clean and polished control scheme which possesses several strong and useful convergence properties, the FM algorithm has heavily inspired subsequent research in the field [5], [6]. Extensive studies have examined the power control problem from game-theoretic and utility based perspectives, allowing the consideration of cooperative and competitive network participants [7], [8]. These control methods generally focus on "memoryless" implementations, which only consider the current power signal vector when deciding the optimal power for a subsequent iteration. This restriction results in well-known stability issues.

In a 2017 study which addresses these stability issues, Zhou et al. presented an alternate power control algorithm known as dual averaging (DA) [9]. This algorithm uses all past iterates in the decision in a way that prevents the current transmitted power from being dominated by any particular past received signal. Particular care is taken to enforce fully distributed control and eliminate communication requirements. Furthermore, the total historical information is not stored explicitly. Rather, to reduce memory requirements, such information is carried forward implicitly and efficiently in rolling summaries.

A further concern beyond memory requirements is the delays inherent in any physically realizable system. These delays can result from either signal propagation time or from the computational requirements to implement such power control algorithms in any real-world device. Previous literature has studied the effect of relatively simple constant and time-varying delays on the FM algorithm and adaptions thereof [10], [11], [12], [13]. However, modern mobile and mesh wireless networks may consist of a wide range of distinct devices, each attempting to send different types of information. This may lead to a spread of delays culminating in either no power signals in a time slot, or multiple signals being delayed to the same subsequent time slot. Furthermore, it could also lead to power signals arriving out of order. Since the presence of delay is so ubiquitous in real-world systems, any new power control algorithm should be shown to handle the concerns that delayed information introduces.

Our goal is to present a new power control algorithm that maintains the desirable properties of the dual averaging algorithm [9], and is capable of handling arbitrary delays. We must first address the situation in which multiple power iterates are delayed such that they arrive within a single time step. In this case, the control algorithm must either choose from or combine all available data without prior knowledge of the order in which the iterates were generated. We must also allow for the situation in which the delays cause a power iterate to be considered out of order. These considerations allow our new algorithm to address a wide range of both deterministic and random delays, a highly desirable capability which greatly increases the robustness of the system.

Delays have been studied in wireless networks in the context of the FM algorithm [10], [11], [12], [13], though none have addressed the case in which multiple SINR measurements arrive simultaneously or the case where SINR measurements arrive out of order. So far, properties of the DA algorithm in a network with delays have not been studied.

A. Our Contributions

Our contributions are threefold: we first present Robust Feedback Averaging (RFA), a novel algorithm which utilizes the core idea of the dual averaging distributed power control

A. Ward, D. Miller, Z. Zhou, and N. Bambos are with the Department of Electrical Engineering, Stanford University, Stanford, CA, 94305, USA. {atward, danielrm, zyzhou, bambos}@stanford.edu P. Mertikopoulos is with the French National Center for Scientific Research (CNRS) and with Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, F-38000, Grenoble, France.

Preprint submitted to 57th IEEE Conference on Decision and Control.
Received March 20, 2018.

algorithm. This new algorithm expands the DA algorithm's capabilities, and allows it to address a wide range of delay types. RFA is a fully asynchronous control algorithm, and this asynchronicity is central in RFA's ability to handle arbitrary delays. This property relaxes the need for update synchronization, and even allows different links to perform power updates at different rates. Furthermore, the RFA algorithm maintains both the constant memory property and the baseline power convergence guarantee of the original DA algorithm in this more general environment.

Second, we prove that the RFA algorithm maintains the original performance guarantees: if the channel is feasible, the RFA algorithm converges to the optimal power vector. Furthermore, we prove a new guarantee for the RFA algorithm: even subject to random delays, the algorithm converges almost surely to the optimal power vector. These guarantees are not matched by the FM algorithm, or by parallel adaptations of such. While the FM algorithm has been shown to converge under certain time-varying delays, it may fail to converge when said delays are arbitrarily random. Additionally, the FM algorithm as it stands cannot address the situation of multiple iterates within a time slot, and even if augmented in a manner similar to the RFA algorithm, does not maintain its convergence guarantee.

Thirdly, we demonstrate the robustness of the RFA algorithm in the presence of both random channels and delays via a series of numerical simulations. These simulations demonstrate that the RFA algorithm achieves convergence in a variety of scenarios, both ones that we prove will converge and ones where the RFA algorithm is not proven to converge. We also show that our algorithm empirically outperforms a corresponding augmentation of the FM algorithm.

II. Background, Model, Motivation

A. Power Control in Wireless Networks

We consider a typical wireless network model consisting of N communication links, each comprised of a transmitter and an intended receiver. The transmitted power vector is denoted by $\mathbf{p} = (p_1, \ldots, p_N)$ where p_i is the power used by the transmitter of link i. For this paper, we consider only non-negative real transmission powers, i.e. $\mathbf{p} \in \mathbb{R}_+^N$.

The most common service quality measure used in power control systems is signal to interference and noise ratio (SINR). Given a power vector $\mathbf{p}$, link i's SINR $r_i(\mathbf{p})$ is given by the following ratio:

$$r_i(\mathbf{p}) = \frac{G_{ii} p_i}{\sum_{j \neq i} G_{ij} p_j + \eta_i} \tag{1}$$

where η_i is the thermal noise associated with the receiver of link i and $G_{ij} \geq 0$ is the power gain between transmitter j and receiver i; for $i \neq j$, G_{ij} represents the interference receiver i experiences due to transmitter j per unit transmission power used. Here, we represent all the power gains G_{ij} with the gain matrix $\mathbf{G}$ and all the thermal noises with the vector η.

Additionally, each link i has a threshold $r_i^* > 0$, which denotes the minimum SINR necessary to achieve acceptable service quality for link i. The standard power control problem [14], [15] is to find a power vector $\mathbf{p}$ such that the quality of service constraints hold:

$$r_i(\mathbf{p}) \geq r_i^*, \forall\ i \tag{2}$$

where inequality is component-wise.

Finding a solution to this problem assumes that such a power vector exists; to characterize the existence of such power vectors, we shall define the notion of *channel feasibility*.

Definition 1. *A channel, given by $\mathbf{G}, \eta$ is said to be feasible with respect to a target SINR vector $\mathbf{r}^* = (r_1^*, \ldots, r_N^*)$ if there exists a $\mathbf{p}$ satisfying (2). The channel is otherwise said to be infeasible.*

We can easily check for channel feasibility by constructing the re-weighted noise vector $\boldsymbol{\gamma}$ and the re-weighted gain matrix $\mathbf{W}$:

$$\gamma_i = r_i^* \frac{\eta_i}{G_{ij}} \quad W_{ij} = \begin{cases} 0 & , i = j \\ r_i^* \frac{G_{ii}}{G_{ij}} & , i \neq j. \end{cases} \tag{3}$$

Based on [3], we have the following definition.

Definition 2. *A channel $\{\mathbf{G}, \eta\}$ is feasible with respect to a target SINR vector $\mathbf{r}^*$ if and only if the largest eigenvalue $\lambda_{max}(\mathbf{W}) < 1$.*

This gives us a convenient way to check channel feasibility.

A standard practical constraint in wireless communications is to restrict the allowed transmitter powers to lie in a specified interval [9] [16], effectively bounding the maximum power for each link by p_i^{max}. The feasible support set, then, is

$$\mathcal{P} = \prod_{i=1}^{N} [0, p_i^{max}] \tag{4}$$

We also define the optimal power vector $\mathbf{p}^* \in \mathcal{P}$ to be the smallest component-wise $\mathbf{p}$ that satisfies (2) and (4).

B. Foschini-Miljanic Algorithm

The standard algorithm used in distributed power control systems is the Foschini Miljanic (FM) algorithm [3]. The FM power update rule, modified slightly to accommodate the maximum power bounds, is

$$p_i^{t+1} = \min\left(p_i^t \frac{r_i^*}{r_i(\mathbf{p}^t)}, p_i^{\max} \right) \tag{5}$$

The FM algorithm, as stated in (5), requires a number of conditions to be satisfied in order to converge to the optimal power $\mathbf{p}^*$:

1) The channel must be feasible w.r.t. r^* (2).
2) The optimal power vector must be within the power bounds (4).
3) The channel $\mathbf{G}, \eta$ must be constant and time-invariant[1].

[1] While the existence of a constant, time-invariant channel is required for convergence to $\mathbf{p}^*$, the FM algorithm may still converge to a steady-state distribution if the channel varies in time.

4) At time t, the previous SINR measurement $r_i(\mathbf{p}^{t-1})$ must be available at transmitter i to calculate p_i^t.

If these conditions are satisfied, then the power iterate $\mathbf{p}^t$ converges to the optimal power vector $\mathbf{p}^*$ with the FM update rule, irrespective of the initial point $\mathbf{p}^0$.

C. Dual Averaging Algorithm

A novel alternative update to the FM algorithm is the dual averaging (DA) power control algorithm [9]. This algorithm performs updates on a dual power vector, $\mathbf{y}$, and then transforms these dual power vectors into the transmitted power vector $\mathbf{p}$ by finding the closest $\mathbf{p}$ to $\mathbf{y}$ that satisfies the constraints given in (2) and (4). The dual averaging update equation is:

$$y_i^{t+1} = y_i^t - \frac{1}{t+1}\big(G_{ii}p_i^t - r_i^*(\sum_{j\neq i} G_{ij}p_j^t + \eta_i)\big), \tag{6}$$

and these dual power vectors are simply clipped to satisfy (2) and (4), i.e.

$$p_i^t = \max(0, \min(y_i^t, p_i^{max})). \tag{7}$$

It is important to note that all of the elements of the second term in (7) can be extracted from the SINR measurement $r_i(\mathbf{p}^t)$ that is returned by the receiver. This algorithm, like the FM algorithm, will converge to the optimal power vector for feasible, time-constant channels with no delay; however, the DA algorithm has also been proven to converge to $\mathbf{p}^*$ in the presence of a stochastic, time-varying channel environment [9]. Therefore, the third condition required by the FM algorithm above is not necessary for DA.

D. Delay in Wireless Networks

While both the FM algorithm and the DA algorithm enjoy optimal convergence properties in a delay-free network, real-world networks include multiple sources of latency. As they stand, neither the FM update presented in 5 nor the DA update shown in 7 is able to accommodate delays; both of these update equations assume the SINR $r_i(\mathbf{p}^{t-1})$ from the previous iteration, measured at receiver i, is immediately available to the transmitter at time step t. Furthermore, the power updates are assumed to be synchronized across all links in the system.

We aim to explore the properties of the dual averaging algorithm in the presence of delayed SINR measurements. Specifically, we set out to answer the following questions:

- How can we modify the DA algorithm to allow the algorithm to operate with general and random delays?
- Is the updated algorithm robust to delays, as well as time-varying channels?
- Can we obtain performance guarantees for this updated algorithm?

Before we present our Robust Feedback Averaging algorithm, we must formally introduce the notion of delay in our model.

We now consider a channel in which, for each time step and for each link, the SINR signal is delayed by $D_i^t \in \mathbb{Z}^+$ units. Notice that these delay variables are distinct for every transmitter i and every time step t. Therefore, if transmitter i sends a power vector at time t' ($p_i^{t'}$), that transmitter will receive the SINR measurement $r_i(\mathbf{p}^{t'})$ from receiver i at time $t' + D_i^{t'}$.

This is a more comprehensive model of delay than those previously considered in the context of the FM algorithm [10], [11], [12], [13]. In these works, the delay was either constant or varying with respect to time, but not also varying with respect to each receiver. This conveniently avoids the aforementioned issues we have highlighted in this paper: simultaneous SINRs and out-of-order SINRs.

The notation presented here allows for a general model for delay, and can account for the both of these cases. Additionally, by making D_i^t a random variable, we can explore the effect of random delays on our algorithm.

III. Robust Feedback Averaging with Delayed and Asynchronous Updates

We now discuss the Robust Feedback Averaging algorithm. As discussed, In the traditional formulation of both the FM algorithm and the DA algorithm, the SINR $r_i(\mathbf{p}^t)$, is assumed to be immediately available to the transmitter at every time step, and updates are assumed to be synchronized across all links in the system. By adjusting the DA algorithm for delayed, asynchronous updates, we can relax these two assumptions.

Algorithm 1 Robust Feedback Averaging Algorithm

```
Each link $i$ arbitrarily chooses an initial $y_i^0$.
for $t = 0, 1, 2, \dots$ do
  for $i = 1, \dots, N$ do
    $p_i^t = \max(0, \min(y_i^t, p_i^{max}))$
    $n = \sum_{t'=0}^{t} \mathbb{I}\left(t' + D_i^{t'} = t\right)$
    if $n > 0$ then
      $y_i^{t+1} = y_i^t - \frac{1}{n(t+1)} \sum_{t'=0}^{t} \Big(G_{ii}p_i^{t'} - r_i^*\Big(\sum_{j\neq i} G_{ij}p_j^{t'} + \eta_i\Big)\Big)\mathbb{I}\left(t' + D_i^{t'} = t\right)$
    else
      $y_i^{t+1} = y_i^t$
    end if
  end for
end for
```

The RFA algorithm, shown in Algorithm 1, is able to accommodate these delayed SINR measurements. Since the delays D_i^t can all be different, it is possible for a link to receive no signals at all on a given time step. Conversely, it is also possible for a link to receive multiple SINR measurements in a given time step. When no new SINR signals are received by transmitter i, the value of y_i (and the transmitted power p_i) do not change. When multiple SINR measurements are received, the received signals are simply averaged.

To perform these two actions, we calculate n, which represents the number of SINR measurements received by a transmitter in a given time step. If the transmitter has received any measurements, then an average is taken and the dual power vector is updated as shown in Line 7 of Algorithm 1. Noticeably, the algorithm doesn't know which SINRs were received, nor if the SINRs arrive out of order. Because of the running average implicit in the dual power vectors y_i, the algorithm can dynamically adapt to whatever SINR measurements it receives, regardless of the order. This implicit storing of all historical power and update information is a necessary practical detail in the context of memory-constrained wireless transmitters.

In addition to the minimal memory requirements, the RFA algorithm as presented satisfies a fundamental requirement: it converges to the optimal power vector $\mathbf{p}^*$ in the presence of random delays. We now establish this theoretical guarantee of the RFA algorithm.

We start by designing an energy function that will play an important role in measuring progress of convergence.

Definition 3. *Define the energy function $L : \mathcal{P} \times \mathcal{Y} \to \mathbb{R}$:*

$$E(\mathbf{y}) = \frac{1}{2}\|\mathbf{p}^*\|_2^2 + \sum_{i=1}^{N} h_i^*(y_i) - \langle \mathbf{y}, \mathbf{p}^* \rangle,$$

where $h_i^*(y_i) = \max_{p_i \in [0, p_i^{\max}]}\{p_i y_i - \frac{1}{2}p_i^2\}$ and $\langle \cdot, \cdot \rangle$ denotes inner product.

Two important properties of the energy function are (proof omitted due space limitation):

Lemma 1. *Let $\mathbf{y}^t, \mathbf{p}^t$ be iterates from Algorithm 1.*

1) $E(\mathbf{y}) \geq 0, \forall \mathbf{y} \in \mathbf{R}^N$.
2) *If $E(\mathbf{y}^t) \to 0$, then $\mathbf{p}^t \to \mathbf{p}^*$.*

Utilizing this energy function, we can then establish the main convergence result:

Theorem 1. *Let $\mathbf{p}^t$ be the iterates generated by Algorithm 1. If the random delays are all bounded: $D_i^t < \infty, \forall i, t$, then $\mathbf{p}^t \to \mathbf{p}^*$,* ***a.s.***.

Brief Proof Sketch:

There are two main ingredients to the proof, which we provide a brief sketch here due to space limitation. First, using the energy function $E(\mathbf{y}^t)$, we show the recurrence of the power iterate $\mathbf{p}^t$. Specifically, we consider $E(\mathbf{y}^t)$ and show that unless the iterate $\mathbf{p}^t$ visits $\mathbf{p}^*$ infinitely often almost surely, then $E(\mathbf{y}^t)$ will diverge to $-\infty$. However, this generates an immediate contradiction because $E(\mathbf{y}^t)$ is always positive (one of its important properties).

Second, having established the above recurrence result, we then perform a fine-grained analysis combining bounded delays and further properties of the energy function to show that, path-by-path, for any fixed neighborhood around $\mathbf{p}^*$, after a sufficiently long time, once the iterate $\mathbf{p}^t$ enters into the neighborhood, it will be forever trapped inside the neighborhood. This trapping property (on a path-by-path basis), together with the recurrence property establishes the claim. ■

This proof of almost sure convergence to the optimal vector makes the RFA algorithm a powerful tool in delay-ridden networks. Furthermore, the only condition required of the proof is that the delays be finite. In real-world systems, this is always true, so this is an extremely weak assumption. And, in the presence of no delay, RFA reduces to the dual averaging algorithm and enjoys almost sure convergence in both deterministic and stochastic time-varying channels.

IV. Simulation Results

To validate the convergence and performance guarantees of Robust Feedback Averaging, we provide several numerical experiments that illustrate the algorithm's various properties. We consider two sets of experiments in this section: first, we introduce two different forms of delay into a deterministic, time invariant channel; second, we examine the effect of adding delays in a more complex stochastic and time-varying channel.

A. Foschini-Miljanic Algorithm with Asynchronous Delay

As an algorithmic comparison, we also alter the Foschini-Miljanic algorithm to account for transmitter-distinct, time-varying delays. We further modify the FM update presented in [10] to allow the FM algorithm to handle delays which are specific for each link. Our modification is similar to the modification in Algorithm 1: all of the SINR values returned in a given time step are averaged, and the average value is used for the FM update. Our modified FM algorithm is shown in Algorithm 2.

Algorithm 2 Foschini-Miljanic Algorithm: Random Delays

```
Each link $i$ arbitrarily chooses an initial $p_i^0$.
for $t = 0, 1, 2, \dots$ do
    for $i = 1, \dots, N$ do
        $n = \sum_{t'=0}^{t} \mathbb{I}(t' + D_i^{t'} = t)$
        if $n > 0$ then
            $p_i^{t+1} = \frac{1}{n} \sum_{t'=0}^{t} \big(p_i^{t'} \frac{r_i^*}{r_i(\mathbf{p}^{t'})}\big) \mathbb{I}(t' + D_i^{t'} = t)$
        else
            $p_i^{t+1} = p_i^t$
        end if
        $p_i^{t+1} = \min(p_i^{t+1}, p_i^{max})$
    end for
end for
```

As expected, our modified FM algorithm reduces to the original FM algorithm when $D_i^t = 0 \quad \forall\, i, t$. We compare the performance of this modified FM algorithm and our RFA algorithm in the following experiments.

B. Deterministic and Time-Invariant Channel

We begin with a deterministic, time-invariant channel with 4 links. The channel properties $\mathbf{G}, \eta$ and optimal power vector $\mathbf{p}^*$ are

$$\mathbf{G} = \begin{bmatrix} 3 & 1 & 2 & 1 \\ 1 & 3 & 1 & 2 \\ 2 & 1 & 3 & 1 \\ 1 & 2 & 1 & 3 \end{bmatrix} \eta = \begin{bmatrix} 0.1 \\ 0.2 \\ 0.3 \\ 0.1 \end{bmatrix} \mathbf{p}^* = \begin{bmatrix} 0.079167 \\ 0.089583 \\ 0.104167 \\ 0.077083 \end{bmatrix}$$

For these simulations, we select the target SINR $r_i^* = 0.5$ and the maximum power $p_i^{max} = 1.0$ for all links in the channel. We chose these parameters to present a relatively challenging environment for the algorithms; as the $\mathbf{G}$ matrix indicates, each link experiences a lot of interference from every other link in the system. Due to this challenging environment, we only require the links to achieve a SINR of 0.5. The maximum power $p_i^{max} = 1.0$ was chosen by considering the optimal power vector $\mathbf{p}^*$; any algorithm which finds the optimal transmitter powers will not be affected by this upper bound in the long run.

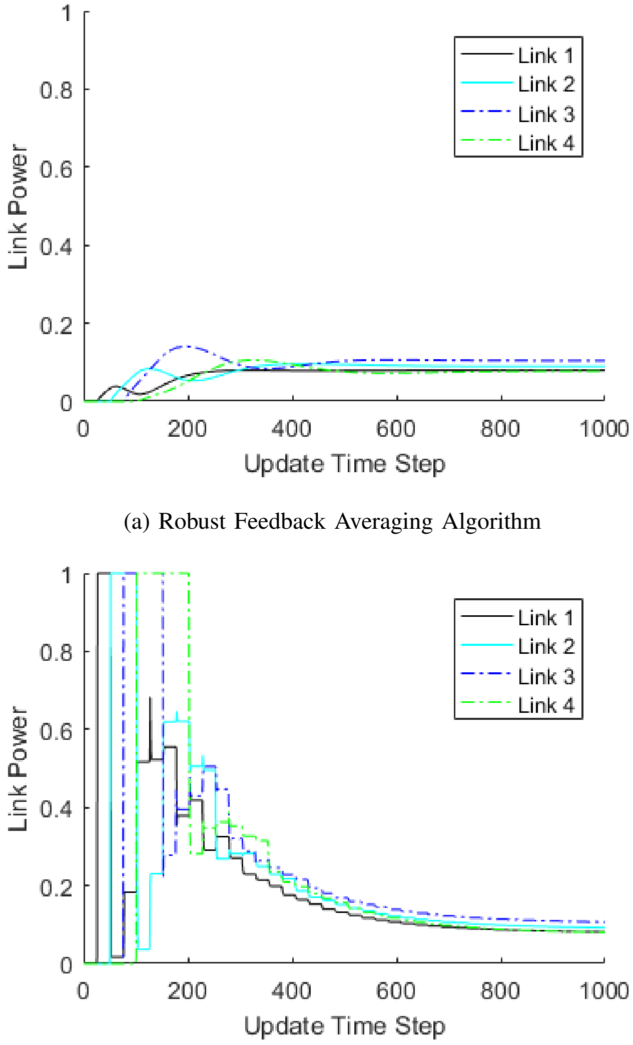
We can calculate the re-weighted gain matrix and re-weighted noise vector $\mathbf{W}$, γ, and verify that $\lambda_{max}(\mathbf{W}) = 0.67 < 1$. Therefore, based on Definition 2, the channel is stable. This means that, for both the FM algorithm and the RFA algorithm, the transmitted link powers will converge to the optimal values in the absence of delay.

We will examine algorithm performance on this channel in several different delay regimes. For the simulations shown, all powers p_i^0 (FM) and dual powers y_i^0 (RFA) were initialized to zero.

Experiment 1: We first examine the algorithmic performance under the regime of constant delay. However, we make the delay distinct for each channel. For this case, we set $\mathbf{D} = [D_1 \ D_2 \ D_3 \ D_4]^T = [25 \ 50 \ 75 \ 100]^T \quad \forall t$. This is a reasonable experiment, as in a practical setting, each transmitter-receiver pair could have different computational capabilities or could be separated by a greater physical distance, causing the delays for each link to be distinct. Even this simple scenario is not covered by the works which studied the FM algorithm in the presence of delay [10], [11], [12], [13], so the FM algorithm is not guaranteed to converge. However, as we proved in Theorem 1, RFA will eventually return the optimal power vector $\mathbf{p}^*$.

Figure 1 shows the power for each link with this constant delay. Specifically, Figure 1(a) shows the transmitted power for each link plotted against time when using the RFA algorithm, and Figure 1(b) does so using the modified FM algorithm. As expected, neither algorithm updates until the 25th time step, when the first SINR is received. At $t = 25$, only Link 1 updates, as it is the only one to receive a SINR measurement; 25 time steps later, Link 2 begins updating, then Link 3, and finally Link 4.

The FM algorithm exhibits interesting, step-wise behavior in Figure 1(b). This is due to the fact that the FM algorithm only depends on the most recent SINR measurement to set its next transmission power. Because the transmission powers at time $t = [0, 25)$ were all the same, the SINRs received during $t = [25, 50)$ were all the same, so the transmission powers in that interval remained constant.



(a) Robust Feedback Averaging Algorithm

(b) Foschini-Miljanic Algorithm

Fig. 1: Results of **Experiment 1**: power marginal convergence behavior of the RFA algorithm (a) and the FM algorithm (b) with a fixed, constant delay for each channel: $\mathbf{D} = [25 \ 50 \ 75 \ 100]^T \quad \forall t$ for channels 1-4, respectively. Both algorithms converge to the optimal power vector, and RFA uses less overall power.

Regardless of the stepwise behavior, the power appears to converge to the optimal values both while using the FM algorithm and while using the RFA algorithm. however, the transient behavior is quite different. Noticeably, the FM algorithm encourages all of the transmitted powers to rise to the maximum power at the beginning of the simulation. This is an effect of the FM algorithm only considering the most recent SINR measurement when performing an update; at the beginning, all SINRs are below the target values, so they are increased drastically. It then takes a considerable amount of time (200 time steps) for the values to eventually stabilize, and it takes another 400 time steps for all of the powers to simultaneously decrease to eventually settle on the optimal powers.

Meanwhile, the RFA algorithm encourages a more gradual approach. Because of the role of averaging, when a transmitter receives a SINR measurement that is far below the target SINR, it slowly increases its transmitted power. With all transmitters doing this in tandem, this causes the trajectories of the transmitted powers to approach the optimal power vector from below, rather than from above. This is significant, as the area under each curve represents the total power used by each transmitter. As seen in Figure 1, there is about an order of magnitude difference in the area under the curves in Figure 1(a) and Figure 1(b), which represents an order of magnitude power savings caused by RFA.

Experiment 2: The advantage of RFA is even more apparent when we allow the delays to be random. In our second simulation, we sample each delay D_i^t individually from a uniform distribution over the integers, i.e. $D_i^t \sim U(0, 100)$. This is a more challenging environment than in the previous experiment, because the delay in the previous experiment, though it varied across channels, was constant with respect to a single channel. That meant that (after $t = 100$) every transmitter received a single SINR measurement. In this experiment, we examine the impact of multiple SINR measurements per time step, and out-of-order SINR measurements.

Figure 2 shows the transmission power of each transmitter vs time. In Figure 2(a), we see a much more noisy convergence curve for the FM algorithm, with frequent spikes in the transmitted powers. Each spike in the plot after the initial transient behavior is due to an out-of-order SINR measurement; here, again, the "memoryless" property of the FM algorithm hinders its performance. The initial spike in power, which is similar to the behavior observed in Experiment 1, is again an unnecessary waste of power.

The RFA algorithm, on the other hand, does not cause any of the transmitters to send spikes in power when an out-of-order SINR is received. Similarly to Experiment 1, it converges from below to the optimal power vector in an energy-efficient manner.

C. *Stochastic and Time-Varying Channel*

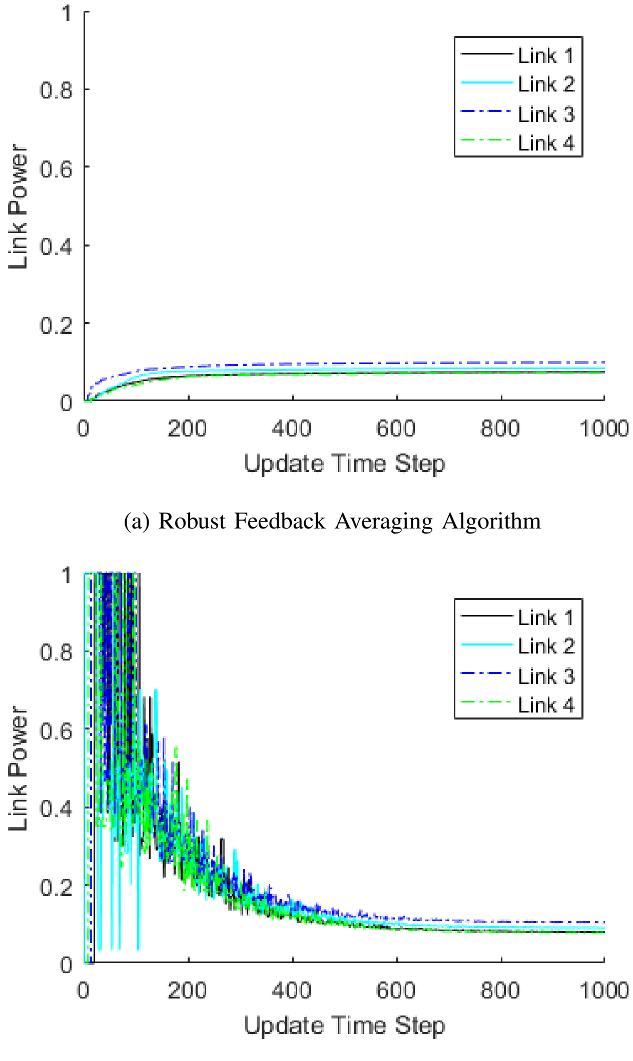
We now turn to the case of a stochastic, time-varying channel with delay. We define a stochastic channel as one in which the gain matrices and noise evolve stochastically over time: $\{\mathbf{G}^t\}_{t=0}^{\infty}, \{\eta^t\}_{t=0}^{\infty}$. For these experiments, we consider a stochastic channel that is drawn **i.i.d.** from a set of 2 deterministic channels, each with equal probability:

$$(\mathbf{G}^t, \eta^t) = \begin{cases} (\mathbf{G_1}, \eta_1), & w.p. \quad 0.5 \\ (\mathbf{G_2}, \eta_2), & w.p. \quad 0.5, \end{cases}$$

$$\mathbf{G_1} = \begin{bmatrix} 2 & 5 \\ 6 & 3 \end{bmatrix} \eta_1 = \begin{bmatrix} 0.1 \\ 0.2 \end{bmatrix}$$

$$\mathbf{G_2} = \begin{bmatrix} 5 & 3 \\ 3 & 3 \end{bmatrix} \eta_2 = \begin{bmatrix} 0.15 \\ 0.05 \end{bmatrix}$$

Again, we chose these matrices to present a challenging environment for the algorithms; even though there are only two channels, the $\mathbf{G}$ matrices dictate that the signal in each link will heavily interfere with the signals from the other link.



(a) Robust Feedback Averaging Algorithm

(b) Foschini-Miljanic Algorithm

Fig. 2: Results of **Experiment 2**: Power marginal convergence behavior of the RFA algorithm (a) and FM algorithm (b) with a uniform random delay of $[0, 100]$ time steps. Due to the out-of-order nature of the SINR measurements, the FM algorithm frequently unnecessarily increases the transmission power. Both algorithms appear to converge.

Experiment 3: First, we set $r_i^* = 0.5$ and $p_{max} = 1.0$ as before. Using this r_i^*, we check channel feasibility by constructing $\mathbf{W}_1$, γ_1, $\mathbf{W}_2$, and γ_2. We see that $\lambda_{max}(\mathbf{W}_1) = 1.34$ and $\lambda_{max}(\mathbf{W}_2) = 0.46$. This indicates that the channel G_2, η_2 is feasible (by Definition 2, but the channel G_1, η_1 is infeasible. This means that, half of the time, the algorithms will be operating on an infeasible channel; the FM algorithm will not converge to the optimal power vector in this scenario.

However, the average channel,

$$\overline{\mathbf{G}} = \begin{bmatrix} 3.5 & 4 \\ 4.5 & 3 \end{bmatrix} \overline{\eta} = \begin{bmatrix} 0.125 \\ 0.125 \end{bmatrix},$$

is feasible (with $\lambda_{max}(\overline{\mathbf{W}}) = 0.79$). So, by the results shown in [9], the RFA algorithm allows all the link powers in this channel to converge to their optimal value (with no delay).

Now, we add delay to this stochastic channel. We again add uniform independent delay for each channel for each time step, i.e. $D_i^t \sim U(0, 100)$. Figure 3 shows the results of this simulation, with Figures 3(a) and 3(b) illustrating the performance of RFA and FM, respectively.

We see that, again, the RFA algorithm converges smoothly to the optimal power vector from below, just as in previous experiments. However, the FM algorithm, as expected, does not converge. Since FM only relies on the most recently returned SINR measurement, whenever the channel is infeasible (which occurs in this experiment with probability 0.5), the transmitted powers will diverge. Of course, when the channel is feasible, the power vector decreases, so the power will only reach the maximum value if a long sequence of SINR measurements from the infeasible channel were returned sequentially, which will only happen with very low probability.

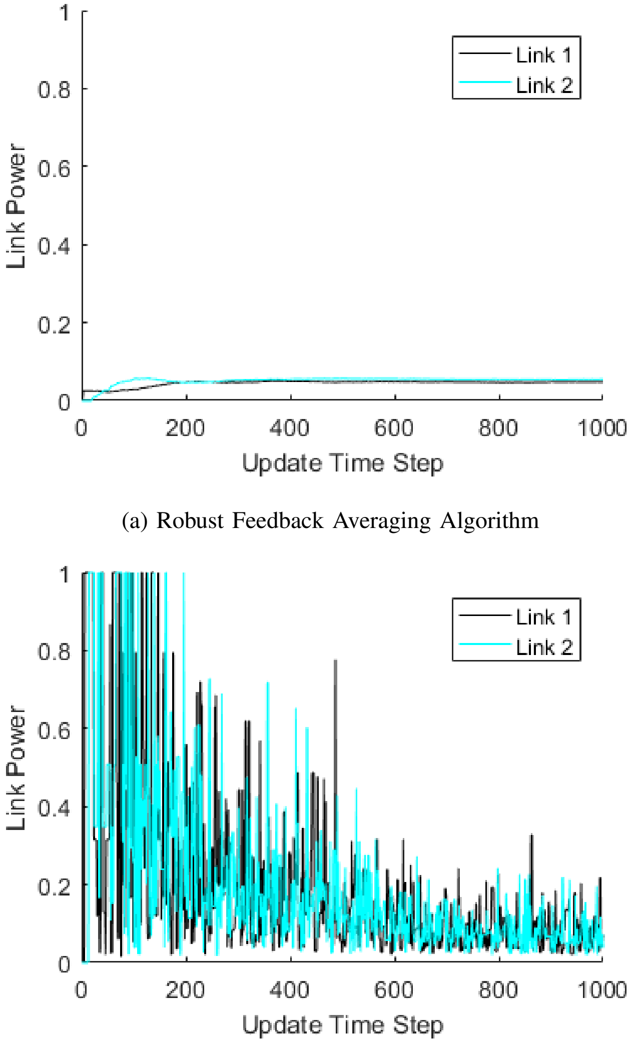
Experiment 4: For this experiment, we did not examine the performance of FM, but instead focused on the performance of RFA. We use the same stochastic channel as in Experiment 3, but here we vary r_i^* between 0.5 and 0.9 (where we always set $r_1^* = r_2^*$).

As in Experiment 3, setting $r_i^* = 0.5$ causes the average channel, $\overline{\mathbf{G}} = \left[\begin{smallmatrix} 3.5 & 4 \\ 4.5 & 3 \end{smallmatrix}\right]$ $\overline{\eta} = \left[\begin{smallmatrix} 0.125 \\ 0.125 \end{smallmatrix}\right]$ to be feasible (with $\lambda_{max}(\overline{\mathbf{W}}) = 0.79$). However, when $r_i^* = 0.9$, we reconstruct $\mathbf{W}_i$ and γ_i and see that $\lambda_{max}(\overline{\mathbf{W}}) = 1.31$ in this case. Therefore, the channel is on average infeasible, and we do not expect the link powers to converge when using RFA.

Figure 4 shows the results of this experiment. Unlike in the other figures, here Figure 4(a) shows the power transmitted by Link 1, and Figure 4(b) shows the power transmitted by Link 2. Each individual trace represents a different target SINR r_i^*. As we expect, when the convergence condition in Definition 2 holds, then the power vector converges to the optimal $\mathbf{p}^*$. But, when $\lambda_{max}(\overline{\mathbf{W}}) > 1$, the powers diverge, and the powers diverge faster with larger $\lambda_{max}(\overline{\mathbf{W}})$.

V. Conclusions

In this paper, we have presented a novel distributed power control update algorithm: Robust Feedback Averaging. We have shown that this algorithm enjoys all of the performance guarantees of the original dual averaging algorithm, namely almost sure convergence to the optimal power vector in both deterministic and stochastic channel environments. Additionally, we have proven that, in the presence of delayed SINR measurements, the RFA algorithm converges almost surely to the optimal power vector in all cases in which the delay is bounded. We have also provided numerical simulations that demonstrate that RFA is robust in both deterministic and stochastic scenarios with delay. We have shown that the RFA algorithm converges in cases where the Foschini-Miljanic algorithm does not converge, and often uses less total power than the FM algorithm on the path to convergence.



(a) Robust Feedback Averaging Algorithm

(b) Foschini-Miljanic Algorithm

Fig. 3: Results of **Experiment 3**: power marginal convergence behavior in the presence of a stochastic, time-varying channel and uniformly random delay. The delays are sampled from uniform distribution: $D_i^t \sim U(0, 100)$. Again, RFA (a) converges to the optimal average power, but the FM algorithm (b) does not converge, since the channel is infeasible half of the time.

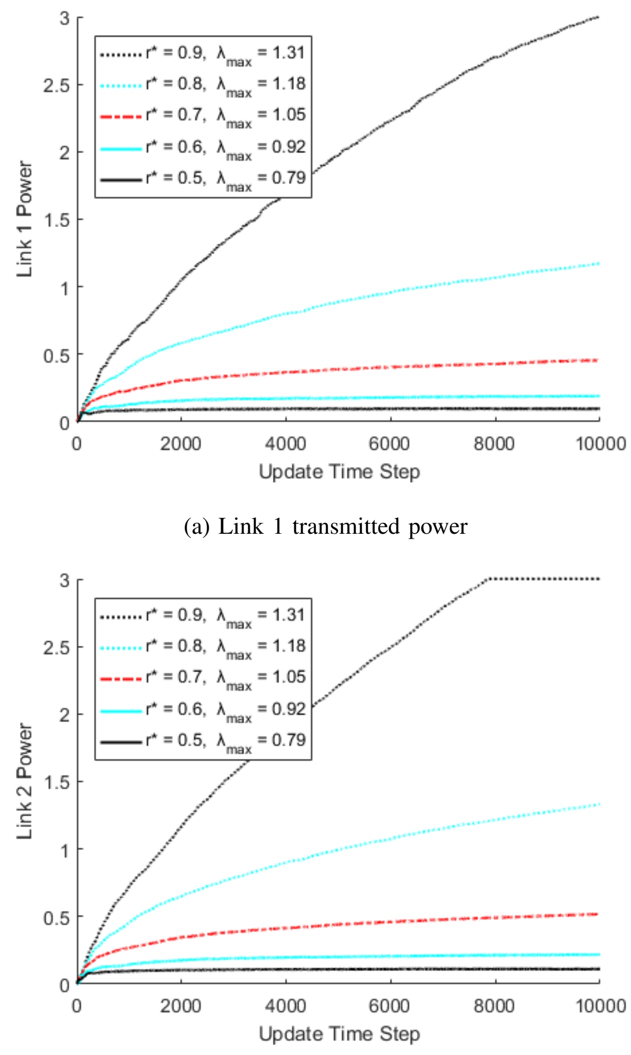
References

[1] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[2] T. S. Rappaport *et al.*, *Wireless communications: principles and practice*. prentice hall PTR New Jersey, 1996, vol. 2.

[3] G. J. Foschini and Z. Miljanic, "A simple distributed autonomous power control algorithm and its convergence," *IEEE transactions on vehicular Technology*, vol. 42, no. 4, pp. 641–646, 1993.

[4] Z. Zhou, D. Miller, N. Bambos, and P. Glynn, "A stochastic stability characterization of the foschini-miljanic algorithm in random wireless networks," in *Global Communications Conference (GLOBECOM), 2016 IEEE*. IEEE, 2016, pp. 1–6.



(a) Link 1 transmitted power

(b) Link 2 transmitted power

Fig. 4: Results of **Experiment 4**: power marginal convergence behavior of the RFA algorithm in the presence of a stochastic, time-varying channel and uniformly random delay ($D_i^t \sim U(0, 100)$). Five different values of r_i^* are plotted here, and the corresponding $\lambda_{max}(\overline{\mathbf{W}})$ for each is shown. The two links in the channel are plotted separately, with the transmitted power of Link 1 (a) above, and the transmitted power of Link 2 (b) below.

[5] R. D. Yates, "A framework for uplink power control in cellular radio systems," *IEEE Journal on selected areas in communications*, vol. 13, no. 7, pp. 1341–1347, 1995.

[6] S. Ulukus and R. D. Yates, "Stochastic power control for cellular radio systems," *IEEE transactions on Communications*, vol. 46, no. 6, pp. 784–798, 1998.

[7] Z. Han, *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge University Press, 2012.

[8] I. Menache and A. Ozdaglar, "Network games: Theory, models, and dynamics," *Synthesis Lectures on Communication Networks*, vol. 4, no. 1, pp. 1–159, 2011.

[9] Z. Zhou, P. Mertikopoulos, A. Moustakas, S. Mehdian, N. Bambos, and P. Glynn, "Stable power control in wireless networks via dual averaging," in *2017 IEEE Global Communications Conference (GLOBECOM)*, Dec 2017, pp. 1–6.

[10] T. Charalambous, I. Lestas, and G. Vinnicombe, "On the stability of the foschini-miljanic algorithm with time-delays," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*. IEEE, 2008, pp. 2991–2996.

[11] T. Charalambous and Y. Ariba, "On the stability of a power control algorithm for wireless networks in the presence of time-varying delays," in *Control Conference (ECC), 2009 European*. IEEE, 2009, pp. 2936–2941.

[12] D. U. Campos-Delgado and M. Luna-Rivera, "Characterization of stability margins of the foschini-miljanic power allocation strategy under constant and time-varying delays," in *Electrical Engineering Computing Science and Automatic Control (CCE), 2011 8th International Conference on*. IEEE, 2011, pp. 1–6.

[13] D. Mitra, "An asynchronous distributed algorithm for power control in cellular radio systems," *KLUWER INTERNATIONAL SERIES IN ENGINEERING AND COMPUTER SCIENCE*, pp. 177–177, 1994.

[14] P. C. Weeraddana, M. Codreanu, M. Latva-aho, A. Ephremides, C. Fischione *et al.*, "Weighted sum-rate maximization in wireless networks: A review," *Foundations and Trends® in Networking*, vol. 6, no. 1–2, pp. 1–163, 2012.

[15] C. W. Tan, "Wireless network optimization by perron-frobenius theory," in *Information Sciences and Systems (CISS), 2014 48th Annual Conference on*. IEEE, 2014, pp. 1–6.

[16] Z. Han, D. Niyato, W. Saad, T. Başar, and A. Hjørungnes, *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*. Cambridge University Press, 2011. [Online]. Available: https://books.google.com/books?id=mvaUAwAAQBAJ