Exploiting Mobility in Proportional Fair Cellular Scheduling: Measurements and Algorithms

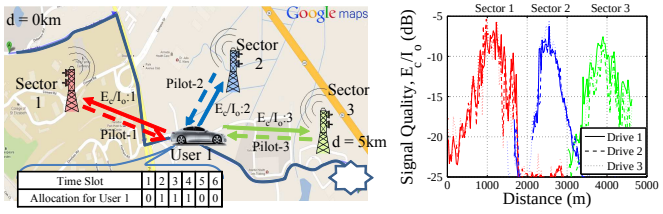
Robert Margolies, *Student Member, IEEE,* Ashwin Sridharan, *Member, IEEE,* Vaneet Aggarwal, *Member, IEEE,* Rittwik Jana, *Member, IEEE,* N. K. Shankaranarayanan, *Senior Member, IEEE,* Vinay A. Vaishampayan, *Fellow, IEEE,* and Gil Zussman, *Senior Member, IEEE*

***Abstract*—Proportional Fair (PF) scheduling algorithms are the de-facto standard in cellular networks. They exploit the users' channel state diversity (induced by fast-fading), and are optimal for stationary channel state distributions and an infinite time-horizon. However, *mobile users* experience a non-stationary channel, due to *slow-fading* (on the order of seconds), and are associated with basestations for short periods. Hence, *we develop the Predictive Finite-horizon PF Scheduling ((PF)^{2}S) Framework that exploits mobility*. We present *extensive channel measurement results* from a 3G network and characterize mobility-induced channel state trends. We show that a user's channel state is highly reproducible and leverage that to develop a *data rate prediction* mechanism. We then present a few *channel allocation estimation algorithms* that exploit the prediction mechanism. Our *trace-based simulations* consider instances of the (PF)^{2}S Framework composed of combinations of prediction and channel allocation estimation algorithms. They indicate that the framework can increase the throughput by 15%–55% compared to traditional PF schedulers, while improving fairness.**

***Keywords*—Cellular networks, Mobility, Proportional fairness, Measurements, Channel state prediction, Slow-fading.**

I. Introduction

3G and 4G (LTE) cellular networks incorporate opportunistic schedulers [11]. These schedulers allocate resources to users with good channel conditions by leveraging channel state variations, due to fast-fading,[1] as well as multi-user diversity. *Proportional Fair* (PF) scheduling algorithms are the de-facto standard for opportunistic schedulers in cellular networks [16]. They aim to provide high throughput while maintaining fairness among the users. PF scheduling algorithms have been extensively studied in the past (e.g., [6], [10], [18]). These algorithms are optimal under the assumptions that the wireless channel state is a stationary process (i.e., it is subject only to fast-fading) and the users' association times are long (e.g., static users or pedestrians) [20], [28]. However, when these assumptions do not hold (which is the case for mobile users), the performance of these algorithms is suboptimal [5].



(a) (b)

Fig. 1. Mobile user trajectory along a road through 3 cellular sectors: (a) map outline and interactions with the cellular network and (b) measured values of channel quality (E_c/I_o) during 3 different drives.

For example, Fig. 1 illustrates a trajectory of a car along a 5km path, and the signal quality (E_c/I_o) to 3 different sectors (we collected the E_c/I_o values during 3 drives on the path). As can be seen, the channel has a dominant *slow-fading* component[2] on which the fast-fading component is overlaid. Since E_c/I_o has noticeable trends over several seconds, the channel state distribution is non-stationary. Additionally, movement along the path initiates hand-offs between the sectors, and therefore, the association periods are short.

Since PF schedulers are not optimized for mobility, we design the Predictive Finite-horizon PF Scheduling ((PF)^{2}S) Framework *which is tailored for mobile nodes and that takes advantage of both slow- and fast-fading*. It includes three components: (i) data rate prediction, (ii) estimation of future channel allocations, and (iii) slow-fading aware scheduling.

To characterize slow-fading, to provide input to the design of the rate prediction mechanism, and to obtain traces for the evaluation of the framework and algorithms, we conducted an extensive measurement campaign. In particular, we discuss fine-grained (i.e., millisecond resolution) measurements, col-

This work began while R. Margolies was a summer intern at AT&T Labs-Research in 2012. The contributions of R. Margolies (in part) and G. Zussman were supported by NSF grants CNS-10-54856 and NSF CIAN ERC under grant EEC-0812072.

R. Margolies and G. Zussman are with the Department of Electrical Engineering, Columbia University, New York, NY, USA email: {robm, gil}@ee.columbia.edu.

A. Sridharan, V. Aggarwal, R. Jana, N. K. Shankaranarayanan, are with AT&T Labs-Research in NJ, USA email: {asridharan, vaneet, rjana, shankar}@research.att.com.

V. A. Vaishampayan is with the Department of Engineering Science and Physics, City University of New York, College of Staten Island, New York, NY, USA email: vavaishampayan@icloud.com.

A partial and preliminary version of this paper was presented at the IEEE INFOCOM 2014 Conference [23].

[1]Fast-fading is characterized by rapid fluctuations in the received signal strength (due mainly to multipath) [25].

[2]Slow-fading is characterized by slow (on the order of seconds) changes of the received signal strength (e.g., due to path loss and shadowing) [25].

lected from a 3G network.[3] Specifically, we measured wireless channel attributes in *drives spanning* 810km *and during a period of over* 1,300 *minutes*. Unlike a few previous studies (e.g., [27]) which measured the Received Signal Strength Indicator (RSSI) which is the total received power in a frequency band, *we measured the signal quality to each sector (E_c/I_o).* This allows us to obtain important insights, since E_c/I_o is the most relevant predictor of a user's data rate.

We analyze the traces and show that mobile users experience pronounced slow-fading. However, the slow-fading trends cannot be simply tied to line-of-sight metrics, and therefore, developing simple channel state predictions is infeasible. Yet, the slow-fading component of E_c/I_o is remarkably reproducible for multiple drives on the same path (e.g., Fig. 1(b)), lending itself to data-driven prediction approaches.

Based on these observations, we develop a 2-phase rate prediction mechanism (referred to as the Coverage Map Prediction Mechanism (CMPM)). In an offline phase, measurement traces are processed to construct channel quality maps. The online phase is conducted by the sector and includes determination of the user's location and velocity, and thereby the predicted data rate. The localization can be simply done by querying the user's GPS. However, since this imposes energy and computation burdens on the user, we also develop the Channel History Localization Scheme (CHLS) which requires some knowledge of the user's trajectory.[4] CHLS uses a variation of the Dynamic Time Warping (DTW) algorithm (originally developed for speech recognition [24]).

The $(PF)^2S$ Framework also requires algorithms that estimate the future channel allocations based on the rate predictions. We propose three such heuristic algorithms with different degrees of robustness to prediction errors and different performance levels for relatively accurate predictions. Using test cases generated from the collected traces, we perform an extensive simulation evaluation of the $(PF)^2S$ Framework. We consider 9 framework instances, representing combinations of rate prediction mechanisms and channel allocation estimation algorithms. We show that various instances of the framework consistently outperform the PF scheduler. Specifically, throughput improvements in realistic mobile scenarios range from 15% to 55% (with maintained or improved fairness levels). Finally, we study the sensitivity of the framework and algorithms to various network parameters and assumptions, including number of users and delay constraints.

The main contributions of this paper are 3-fold: (i) it demonstrates, based on an extensive measurement campaign, that mobile users experience a reproducible but non-stationary slow-fading channel; (ii) it provides a cellular scheduling framework (and corresponding algorithms), tailored for mobile users; and (iii) it shows (using trace-based simulations) that the framework can significantly improve performance.

The paper is organized as follows. Section II discusses related work and Section III reviews channel state metrics and formulates the problem. Section IV presents the scheduling framework. Section V discusses the measurements and characterization of slow-fading. A rate prediction mechanism is presented in Section VI and algorithms to estimate future channel allocations are presented in Section VII. The framework and algorithms are evaluated in Section VIII. We conclude and discuss future work in Section IX.

[3]The measurements were collected from a 3G network, due to lack of ubiquity of LTE networks. Yet, our observations regarding slow-fading apply to 4G networks, as they operate at similar time-scales.

[4]As such, it is highly applicable to users on highways and major roads.

II. Related Work

Opportunistic Scheduling: As mentioned, opportunistic and PF scheduling have been extensively studied (e.g., [6], [11], [18], [20], [28]). PF scheduling algorithms using *fast-fading* channel state predictions appear in [7], [14] (without a prediction mechanism). Scheduling for *mobile users* is considered in [3], [10], [27], where the underlying assumption in [3], [10] is that the user's mobility patterns induce a stationary (and known) *slow-fading* channel. The algorithm of [27] schedules *a single user* using an RSSI-based prediction method *at time scales on the order of minutes*. On the other hand, we solve a *multi-user* scheduling problem at *finer time scales (tens of seconds)* using an E_c/I_o-based prediction mechanism.

Channel Measurements and Predictions: Wireless channel measurement studies have been conducted for decades [4], [15]. Recently, [21] studied the interaction of applications and the physical layer attributes in the 1x-EVDO network (using a predecessor to our measurement tool). Slow-fading is studied in *controlled environments* in [29]. Methods for *short-term* (over a few milliseconds) prediction of non-stationary wireless channel states appear in [22]. The measurements in [30], [31] focus on the repeatability of achieved *bandwidth* in a 3G network. Unlike previous works, we conduct measurements of wireless channel quality in a 3G network to characterize and predict slow-fading patterns over tens of seconds.

Localization and Mobility Prediction: Localization in cellular networks includes approaches that utilize time of arrival, time-difference of arrival, angle-of-arrival, cell-ID, and received signal strength (see [17] and references therein). Mobility prediction schemes that utilize pattern tracking and learning algorithm are reviewed in [19]. The method in [12] uses the DTW algorithm, albeit for velocity estimation. The closest related works are [17], [27] that utilize RSSI in GSM networks to localize users via fingerprinting. On the other hand, our scheme uses multiple channel attributes (i.e., E_c/I_o and RSSI) as well as recent history and is evaluated via trace-based simulations.

III. Model and Problem Formulation

In this section, we review the channel state estimation process in 3G networks and formulate the scheduling problem.

A. Channel States in 3G Networks

In a 3G network [16], each basestation covers a cell which is divided into (typically 3) sectors. As illustrated in Fig. 1(a), for data scheduling and hand-off purposes, users estimate the wireless channel quality to each nearby sector. It is estimated as the ratio between the power of a sector-specific pilot signal

and the total in-band power (including interference and noise), and is denoted by E_c/I_o. In Section V, we will consider these values in our measurement study.

A user associates (connects) with the strongest neighboring sector, termed the *serving sector*, and is assigned a dedicated buffer at the sector. When the serving sector E_c/I_o value drops below a threshold (e.g., due to mobility), the user *hands-off* wherein it disassociates from the serving sector and connects to a new sector with a higher E_c/I_o value.

The downlink channel from the sector to the users is time-slotted. We will denote by $E_c/I_o[j]$ the value in time slot j. The users periodically report their E_c/I_o to the sector. Then, an appropriate channelization code is selected and mapped to a feasible *data rate*.[5] The feasible data rate of user i in slot j is denoted r_{ij}. An *opportunistic* scheduler implemented in the sector utilizes the multiuser diversity of the data rates to allocate downlink slots to users (see Fig. 1(a)).[6]

B. Scheduling Problem Formulation

The common 3G scheduler solves a *Proportional Fair* (PF) Scheduling Problem [16], [18] and aims to achieve high overall throughput while maintaining fairness among the users. The common assumptions regarding stationary channels and long association times do not hold in mobile scenarios (as will be shown in Section V). Hence, we formulate the downlink scheduling problem as a variant of the PF Scheduling Problem while utilizing a formulation similar to [5] (which studied adversarial channels). Unlike previous work, (e.g., [10], [20], [28]), we *do not make assumptions regarding the channel state distributions* and optimize over a *finite time horizon*.

We assume that a sector has K associated users with backlogged downlink buffers.[7] Denote by α_{ij} the scheduler allocation ($\alpha_{ij}=1$, if user i is allocated slot j, and $\alpha_{ij}=0$, otherwise). We denote the feasible data rate and the scheduler allocation matrices by $\mathbf{R}=\{r_{ij}\}_{K\times T}$ and $\boldsymbol{\alpha}=\{\alpha_{ij}\}_{K\times T}$, respectively. The nomenclature can be found in Table I. We assume a finite time horizon of T slots that corresponds to the users' association times. By the end of slot T, user i accrues a *cumulative* service $\sum_{j=1}^{T}\alpha_{ij}r_{ij}$. Hence, we formulate the following problem where the objective is to maximize a *proportional fair* cost function.[8]

Finite-horizon Proportional Fair (FPF) Scheduling:

$$\max_{\alpha} \quad C=\sum\nolimits_{i=1}^{K}\log\Big(\sum\nolimits_{j=1}^{T}\alpha_{ij}r_{ij}\Big) \tag{1}$$

$$\text{subject to} \quad \sum\nolimits_{i=1}^{K}\alpha_{ij}=1 \quad \forall j=1\ldots T \tag{2}$$

$$\alpha_{ij}\in\{0,1\}. \tag{3}$$

[5]The mapping from E_c/I_o to data rates is described in Appendix A. The mapping is phone specific and for our phones, the maximum data rate is 20Mbps.

[6]Multiple users (typically, no more than 4) may share a slot. Practically, it is uncommon, and we assume that *exactly* one user is allocated a slot.

[7]While in practice the number of associated users varies with time, we focus on a specific time-period with a given number of users.

[8]Although we focus on proportional fairness, the general approach can be applied to other concave cost functions (e.g., the α-fairness class).

TABLE I. NOMENCLATURE

$E_c/I_o[j]$	The pilot SINR in time slot j
T	Duration of the time horizon (in time slots)
$\tilde{T}$	Duration of the time horizon (in seconds)
K	Number of users
r_{ij}	Feasible data rate for user i in time slot j
$\mathbf{R}=\{r_{ij}\}_{K\times T}$	Feasible data rates matrix
$\hat{\mathbf{R}}=\{\hat{r}_{ij}\}_{K\times T}$	*Predicted* feasible data rate matrix
α_{ij}	Fraction of time slot j allocated to user i
$\alpha=\{\alpha_{ij}\}_{K\times T}$	Allocation matrix
$\hat{\alpha}=\{\hat{\alpha}_{ij}\}_{K\times T}$	*Estimated* allocation matrix
d_i	User i's accumulated delay since last service (number of time slots)
D_{starved}	Delay threshold at which a user is considered *starved*

Even with full knowledge of $\mathbf{R}$, this problem is NP-hard (the proof is provided in Appendix B). In practice, this problem has to be solved in an online (causal) manner. Users are scheduled slot-by-slot, based only on *knowledge* of the history and without full knowledge of $\mathbf{R}$. While the objective in the FPF Scheduling Problem is to maximize the proportional fairness metric (1), when evaluating the framework (Section VIII), we also consider the following metrics.

Definition 1 (Throughput): The average data rate allocated to all users, $\sum_{i=1}^{K}\sum_{j=1}^{T}\alpha_{ij}r_{ij}/T$ is referred to as *throughput*.

Definition 2 (Delay): The number of consecutive time slots in which a user i does not receive an allocation is referred to as the *delay* and is denoted d_i. User i is *starved* if $d_i \geq D_{\text{starved}}$, where D_{starved} is a delay threshold.

We note that in Section VIII, the time horizon is sometimes considered in seconds, and is denoted by $\tilde{T}$. [9]

IV. PREDICTIVE FPF SCHEDULING (PF)^{2}S FRAMEWORK

In this section, we review the widely deployed PF scheduling algorithm and present an online scheduling framework for solving the FPF Scheduling problem which combines two components: (i) data rate predictions and (ii) an estimation of future channel allocations. The design of these components will be presented in Sections VI and VII, respectively. We first describe the PF scheduler deployed in 3G networks [16] which is used in later sections as a benchmark.

Definition 3 (PF-EXP [20], [28]): The scheduler which sets $\alpha_{i^*j}=1$ where $i^*=\arg\max_{i\in K} r_{ij}/R_i[j]$, and $R_i[j]=(1-\epsilon)R_i[j-1]+\epsilon\alpha_{ij}r_{ij}$, is referred to as PF-EXP.

In the definition of the PF-EXP scheduler, ϵ determines the tradeoff between throughput and delay. With large values of ϵ (≈ 1), the scheduler puts more weight on the users' current feasible rates, thereby improving throughput at the expense of delay performance. With small values of ϵ (≈ 0) the users allocation history has more weight, and therefore, the delay performance improves at the expense of throughput. The PF-EXP scheduler approaches *optimal* proportional fairness [20], [28] when the wireless channel state is a stationary process *and* users have long association times (i.e., $T\to\infty$).

Our Predictive FPF Scheduling (PF)^{2}S Framework follows a similar approach as the PF-EXP scheduler to make slot-by-slot allocations. It utilizes a gradient ascent approach [9] to

[9]In HSDPA, which is the 3G technology used in our measurement campaign, the slot length is 2ms and hence, $\tilde{T}=T\cdot 2$ms.

Predictive FPF Scheduling (PF)^{2}S Framework

1: Predict future data rates $\hat{\mathbf{R}} = \{\hat{r}_{ij}\}_{K\times T}$.
2: Estimate future allocations $\hat{\alpha} = \{\hat{\alpha}_{ij}\}_{K\times T}$.
3: **for** slot $j = 1$ to T **do**
4: Compute $M_{ij} = \frac{r_{ij}}{\sum_{t=1}^{j-1}\alpha_{it}r_{it}+\hat{\alpha}_{ij}r_{ij}+\sum_{t=j+1}^{T}\hat{\alpha}_{it}\hat{r}_{it}} \forall\, i \in K$
5: **if** $\exists\, i \in K$ with $d_i \geq D_{\text{starved}}$ **then**
6: $i^* = \arg\max_{\{i\in K: d_i \geq D_{\text{starved}}\}} M_{ij}$
7: **else** $i^* = \arg\max_{i\in K} M_{ij}$
8: $\alpha_{i^*,j} = 1, \alpha_{i,j} = 0 \quad \forall i \neq i^*$

maximize the objective function (1). In each time slot, the channel is allocated to the user corresponding to the largest objective function increase. Temporarily relaxing the integer constraints in (3), the gradient for user i in time slot j is:

$$\frac{\partial C}{\partial \alpha_{ij}} = \frac{r_{ij}}{\sum_{t=1}^{T}\alpha_{it}r_{it}} = \frac{r_{ij}}{\sum_{t=1}^{j-1}\alpha_{it}r_{it} + \alpha_{ij}r_{ij} + \sum_{t=j+1}^{T}\alpha_{it}r_{it}}. \quad (4)$$

Computing the above gradient *requires knowledge of the entire data rate matrix* $\mathbf{R}$ *and is not feasible for an online algorithm*, which only has knowledge of the past. Hence, the denominator of (4) is broken up into three components (from left to right): *past*, *present*, and *future*. From the perspective of an online scheduler, the first two components are known in any time slot. To enable slot-by-slot scheduling, the *future* component of (4) is computed as part of the (PF)^{2}S Framework, which is described in pseudo-code above.

Predictions of future data rates (r_{ij}) and estimates of future channel allocations (α_{ij}) are denoted by $\hat{r}_{ij}$ and $\hat{\alpha}_{ij}$, respectively, with matrix representations denoted by $\hat{\mathbf{R}}$ and $\hat{\boldsymbol{\alpha}}$. At time 0, predictions of $\hat{\mathbf{R}}$ and $\hat{\boldsymbol{\alpha}}$ are *pre-computed* for the entire horizon (next T slots). These matrices can be generated using the methods described in Sections VI and VII but the framework can support other methods. For each user i in each slot j, a ranking M_{ij} which corresponds to (4), is computed using $\hat{\mathbf{R}}$ and $\hat{\boldsymbol{\alpha}}$. The user with the highest ranking is selected.

For a stationary channel, the future channel statistics are captured in the past component of the denominator (4). Hence, algorithms that rely only on past information (i.e., PF-EXP) are optimal. However, for non-stationary channel distributions, this does not hold. Hence, unlike in PF-EXP, step 4 in the framework considers the future channel component. By incorporating the predicted future, the (PF)^{2}S Framework can leverage *slow-fading* trends. In addition, by making slot-by-slot decisions, the framework also leverages *fast-fading* components, similar to PF-EXP.

Since the (PF)^{2}S Framework aims to schedule users during slow-fading peaks (which may occur at several second intervals), it is essential to ensure that this does not result in long delays. Hence, in each slot, the framework first considers the set of *starved* users whose wait time d_i (from the last slot of service) exceeds D_{starved} (defined in Defn. 2) and selects one. If no user is starved, it selects among all users. Thereby, the framework can handle delay constraints. Note that delay considerations can be ignored by setting $D_{\text{starved}} = \infty$.

Fig. 2. Measurement setup and the routes on which measurements were collected (see also Table. II).

V. Slow-fading Measurements

We now describe the measurements collected from a 3G network. Our analysis demonstrates that mobile users experience pronounced and reproducible slow-fading. The observation regarding reproducibility provides insights into the design of the data rate prediction ($\hat{\mathbf{R}}$) mechanism (Section VI).

A. Measurement Setup and Test Drives

The measurement campaign was conducted with Samsung Galaxy S II (GSII) Skyrocket phones [26] using the setup shown in Fig. 2. The phone was connected via USB to a laptop running the Qualcomm eXtensible Diagnostic Monitor (QXDM) software. QXDM queries the phone in real-time and captures various physical layer attributes (described below) as well as GPS reports of location and velocity. QXDM records these measurements every 20ms, capturing the fast-fading and slow-fading components.

For the mobile measurements, the setup was placed in a car which traversed 4 different routes[10] that span both highways and suburban roads (see Table II). Each drive followed the entirety of a given route and several drives were conducted. For control purposes, we also performed measurements with a *static* (immobile) setup. During the measurements, a continuous download was conducted to ensure a sustained network connection. In summary, we measured wireless channel attributes during drives spanning 810km and during a period of over 1,300 minutes.

B. Channel State Metrics and Dynamics

QXDM stores three physical layer attributes: the total in-band power (including interference and noise), termed RSSI, the received pilot-power (RSCP), and the ratio between the pilot power and the total interference (E_c/I_o).[11] These key attributes characterize the channel quality and are periodically reported by the user to the serving sector [16]. While the latter two are *specific* to each nearby sector's pilot channel, the former (RSSI) is not. Moreover, while RSSI was commonly logged and used in previous work (e.g., [27]), from a scheduling perspective, E_c/I_o is the most relevant indicator of a user's channel quality [16].

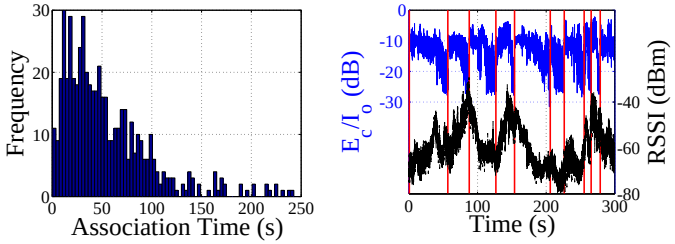
We highlight the slow-fading phenomenon with an example. Fig. 3(a) shows a histogram of the users' association times for 27 drives on all routes, demonstrating that association

[10]Note that routes R1 and R3 have the same start and end locations but are oriented in opposite directions. Hence, we treat them separately.

[11]$E_c/I_o(dB) = RSCP(dB) - RSSI(dB)$.

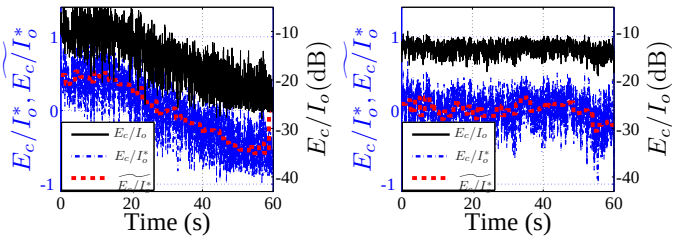
TABLE II. SUMMARY OF COLLECTED MEASUREMENTS

Label	Num Logs	Time Logged (min)	Total Dist. (km)	Av. Dist. (km)	Av. Velocity (m/s)	Total Sectors	Total Serving Sectors	Total Data (MB)
R1	7	305.4	205.5	29.4	6.6	282	67	246
R2	4	85.2	33.0	8.2	10.7	245	42	254
R3	6	252.4	220.8	36.8	21.9	210	68	251
R4	10	359.5	351.0	35.1	16.8	963	336	1538
Static	5	383.4	-	-	-	58	9	895
Total	32	1386.1	810.8	-	-	1758	522	3184



(a) (b)

Fig. 3. (a) Distribution of sector association times for 27 drives along routes R1–R4 and (b) measured values of the RSSI and the serving sector E_c/I_o for a drive on part of route R4 (vertical bars indicate hand-offs).



(a) Mobile Trace (b) Static (Immobile) Trace

Fig. 4. Comparison of E_c/I_o (measured), E_c/I_o^* (normalized), and $\widetilde{E_c/I_o^*}$ (smoothed): (a) a mobile trace from route R4 and (b) a static trace.

times are on the order of tens of seconds. As a specific example, Fig. 3(b) shows measured traces of the RSSI and the serving sector E_c/I_o for part of a single drive along route R4. Clearly, RSSI does not always reflect the same trend as E_c/I_o. Additionally, the E_c/I_o experiences slow-fading on the order of several seconds. Since in most cases, the user's association times are tens of seconds, the slow-fading peaks and troughs occur *within each sector*. Therefore, *we focus in the next two subsections on E_c/I_o slow-fading trends*, which are leveraged by the (PF)^{2}S Framework.

C. Slow-Fading and Mobility

We first demonstrate that the slow-fading phenomenon is closely tied to user mobility. We then characterize the correlation between slow-fading and mobility metrics and show (in contrast to assumptions in past work, e.g., [3]) that slow-fading trends cannot be tied to simple line-of-sight metrics.

To quantify the slow-fading in a user's E_c/I_o trace of T slots, we define a *slow-fading metric* as described below. First, the mean is removed and the trace is normalized to obtain:

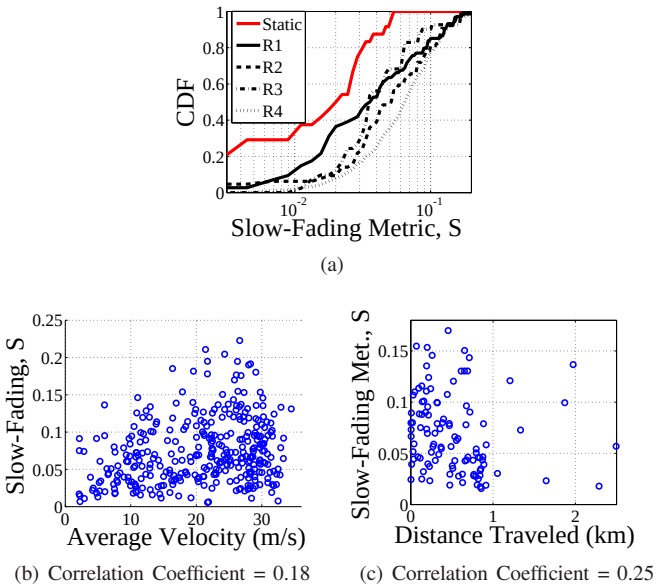
$$E_c/I_o^*[j] = \frac{E_c/I_o[j] - \overline{E_c/I_o}}{\max_{1\le j\le T} \mid E_c/I_o[j] - \overline{E_c/I_o} \mid}. \quad (5)$$

The operation does not affect E_c/I_o *trends*, but removes the amplitude which can vary depending on the sector, thereby enabling a comparison of E_c/I_o traces from different sectors. Then, E_c/I_o^* is *smoothed* by using wavelet transforms to remove the fast-fading components (with frequencies greater than 1Hz). The smoothed version of E_c/I_o^* is denoted by $\widetilde{E_c/I_o^*}$ (more details regarding the smoothing operation appear in Appendix C). Fig. 4 provides visual examples of E_c/I_o^* and $\widetilde{E_c/I_o^*}$ for a mobile user and a static user. Using $\widetilde{E_c/I_o^*}$ clearly illustrates the presence (absence) of a trend in the values of E_c/I_o over the time-period of observation.

Finally, we define the *slow-fading metric* as $S(\widetilde{E_c/I_o^*}) = \sum_{j=1}^{T} \widetilde{E_c/I_o^*}[j]^2/T$. E_c/I_o traces with no appreciable trends will have $\widetilde{E_c/I_o^*}$ values close to zero (Fig. 4(b)) and hence S will be small. On the other hand, E_c/I_o traces with noticeable trends (Fig. 4(a)) will have large values of $\widetilde{E_c/I_o^*}$ (positive or negative) and hence larger values of S. As it is normalized by T, S is used to compare E_c/I_o traces with varying association times from different sectors.

We computed S for every E_c/I_o trace collected from every serving sector (see Table II). Fig. 5(a) shows the Cumulative Distribution Function (CDF) of the slow-fading metric values for routes R1-R4, as well as for the static traces. The mobile routes have much larger values of the slow-fading metric, confirming empirically that S accurately distinguishes mobile traces with slow-fading from immobile traces.

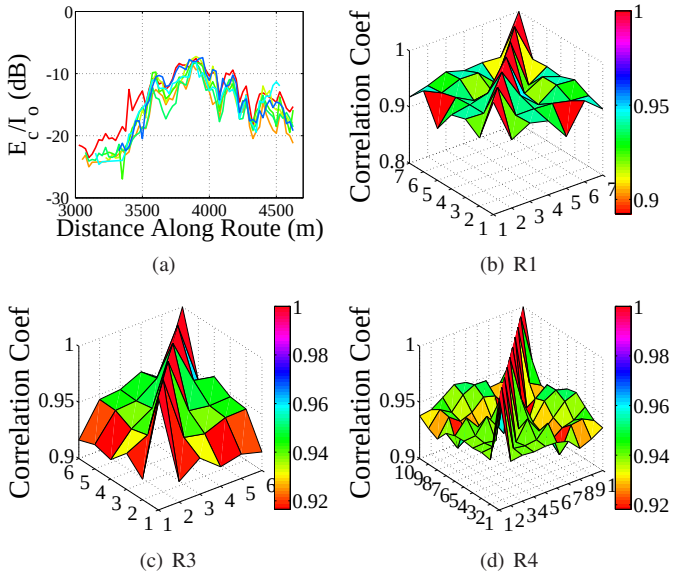
Previous work assumed a strong correlation between slow-fading and line-of-sight parameters (i.e., distance or velocity) [3], [11], supporting a functional prediction of the channel quality. However, our analysis indicates weak correlation between slow-fading (S) and line-of-sight metrics. For example, scatter plots of S and the average velocity or distance traveled of the user are shown in Figures 5(b) and 5(c), respectively. The correlation coefficient between S and the velocity or the distance is 0.18 and 0.25, respectively. Instead, slow-fading is governed by factors such as hand-offs, landscape, and movement-induced shadowing, which are complex to model even in controlled scenarios [29].



(a)

(b) Correlation Coefficient = 0.18 (c) Correlation Coefficient = 0.25

Fig. 5. Characterization of slow-fading: (a) the CDF of the slow-fading metric (S) for mobile and static traces and (b)–(c) scatter plots for all mobile traces of the average velocity or distance traveled while connected to a sector vs. S.



(a) (b) R1

(c) R3 (d) R4

Fig. 6. Reproducibility of measured E_c/I_o values: (a) measurements from 7 drives through a sector on route R1 (aligned in 25m segments) and (b)–(d) the correlation coefficient of E_c/I_o across all drives (with sufficient data) on routes R1, R3, and R4.

D. Slow-Fading Reproducibility

As described above, the slow-fading trend is not directly associated with line-of-sight factors, and therefore, simple functional predictions are infeasible. Yet, the slow-fading component of E_c/I_o is remarkably reproducible, enabling a data-driven prediction approach. Specifically, we observed that the E_c/I_o from multiple measurements (from separate drives) is predictable with an error of 1–3dB (a similar result appears in [27] for RSSI). To illustrate the reproducibility, we divide part of route R4 into 25m segments[12] and show in Fig. 1(b) the E_c/I_o observed across a subset of segments for 3 of the drives on the route through 3 sectors. The overlap of the curves indicates the similarity across all drives.

We strengthen this observation by computing the cross-correlation of E_c/I_o across *all* drives for each route, as follows. Each route is divided into 25m segments and each drive on this route is then represented by a *vector* of E_c/I_o values, one for each segment (e.g., if a route includes n segments, each drive is represented by a n-length vector, with multiple observations in the same segment represented by their average). We then compute the correlation coefficients of all the vectors (drives). Figures 6(b)(c)(d) show the correlation between all drives on routes R1, R3, and R4. Correlation coefficients are between 0.9–0.98, indicating a very high degree of correlation. The high correlation across all repeated drives implies that location-tagged historical measurements of E_c/I_o can be used to accurately predict future slow-fading.

[12]25m is the minimum guaranteed GPS resolution.

VI. Feasible Data Rate Prediction ($\hat{\mathbf{R}}$)

The (PF)^{2}S Framework requires a mechanism to predict the users' feasible data rates for T slots ($\hat{\mathbf{R}}$). We design such a mechanism, based on the observation that the slow-fading component of E_c/I_o is highly reproducible, and refer to it as the **Coverage Map Prediction Mechanism (CMPM)**. In an offline phase, measurement traces are processed to construct geographic coverage maps. The online phase is conducted by the sector and is composed of two steps. First, the user's location and velocity are determined. Then, this information is used in conjunction with the coverage map to predict user i's feasible rates $\hat{r}_{ij}\ \forall\ 1 \leq j \leq T$.

The first step can be implemented by querying the user's GPS. However, since this imposes energy and computation burdens on the user, we also develop the Channel History Localization Scheme (CHLS). The scheme assumes that knowledge of the user's overall trajectory exists. In Section VIII, we evaluate the framework using both alternatives.

A. Coverage Map Construction

The coverage map is constructed offline (once for each route) by placing a lattice over the geographic plane, and dividing it into square *segments* (see Fig. 7). Each segment, denoted by b, is covered by a set of sectors to which a user residing in it can associate, denoted by U_b. Cellular carriers routinely measure the channel quality on major routes. These measurements can be used to compute, for each segment b, an *average* RSSI value as well as average values of E_c/I_o and RSCP for every nearby sector $u \in U_b$. These are denoted by $\overline{\text{RSSI}}\langle b\rangle$, $\overline{E_c/I_{o_u}}\langle b\rangle$, $\overline{\text{RSCP}_u}\langle b\rangle$. To compute these values for our evaluations, each sample measurement was tagged with a GPS location and tied to the appropriate segment.

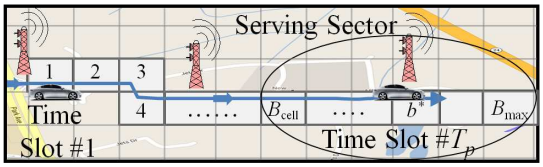


Fig. 7. Illustration of the CHLS: the coverage map segments are labeled starting from 1 (at time slot 1, the user is in segment 1). At the present time slot (T_p), the user is located in one of the segments between B_{cell} and B_{max}, which fall within the coverage area of the serving sector.

B. Channel History Localization Scheme (CHLS)

The first step of the CMPM online phase localizes the user in the coverage map. To do this without GPS, we develop the CHLS. It matches the user's historical channel quality to coverage map segments on the user's trajectory, based on the differences between the channel metric values. Then, the user's location is estimated as the segment paired with its current channel quality value. Matching the channel quality history (i.e., a time-series) to segments (i.e., locations) depends on the user's velocity, which can vary. Hence, we utilize the Dynamic Time Warping (DTW) Algorithm[13] to 'unwarp' the user's historical channel qualities to best fit the coverage map.

The CHLS requires knowledge of the user's trajectory and the user's location at a time slot in the recent history.[14] The user's historical channel measurements are available at no extra cost as they are periodically reported to the network for scheduling and hand-off purposes.

The notation used to describe the scheme is defined below (see also Fig. 7). The sector keeps a history of the user's $E_c/I_{o_u}[j]$, $\text{RSCP}_u[j]$, and $\text{RSSI}[j]$ for the past T_p time slots, which are numbered sequentially from 1 to T_p (present slot). The coverage map segments are sequentially numbered, starting with the segment which is the user's estimated location at slot 1. The segments are numbered up to B_{max}, which is the furthest segment in which the user could reside within the sector coverage area. The serving sector covers a range of segments $\mathbf{B} = \{b : B_{\text{cell}} \leq b \leq B_{\text{max}}\}$.

The DTW Algorithm is applied to identify the cost of selecting each $b \in \mathbf{B}$ as the location estimate for the user. It constructs H, a matrix of size $B_{\text{max}} \times T_p$. The value of entry $h_{b,j}$ represents the minimum cost of pairing time slots from 1 to j with segments 1 to b. The constraint is that segment 1 is paired with slot 1 and segment b is paired with slot j (e.g., the end points are paired). The entries in the first row and column are, $h_{1,j}, h_{b,1} = \infty \, \forall \, b, j$, and the rest of the matrix is computed using $h_{b,j} = \text{c}(b,j) + \min(h_{b-1,j}, h_{b,j-1}, h_{b-1,j-1})$,

[13]A similar dynamic programming algorithm is used in speech recognition [24] to align two phrases which are offset (in time, amplitude, etc.).

[14]For mobile users on highways and major roads, the trajectory can be estimated using mobility prediction techniques (e.g., [19]). The historical location can be reported based on past localization.

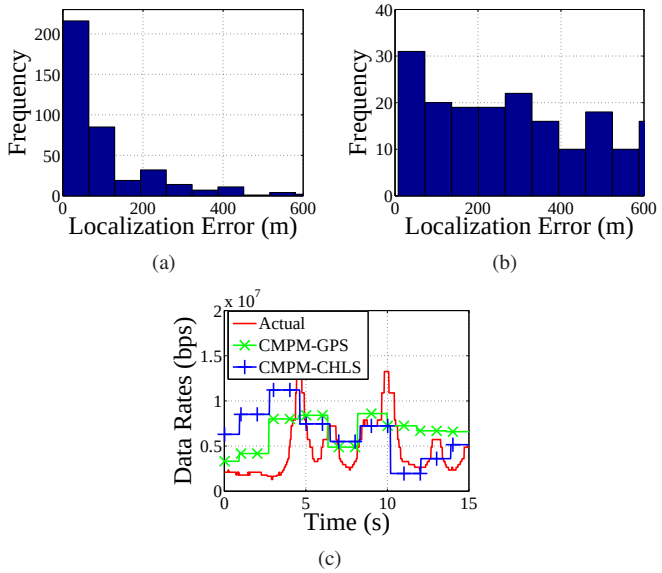


Fig. 8. Evaluation of the Coverage Map Prediction Mechanism (CMPM): (a) CHLS error distribution for 500 tests compared to (b) the error distribution for 500 tests of the localization scheme provided in [27]; (c) An example CMPM data rate prediction when location and velocity are determined using GPS or CHLS.

where the cost of matching segment b to time slot j is

$$\text{c}(b,j) = (\text{RSSI}[j] - \overline{\text{RSSI}}\langle b \rangle)^2 + \sum_{u \in U_b} (E_c/I_{o_u}[j] - \overline{E_c/I_{o_u}}\langle b \rangle)^2 + (\text{RSCP}_u[j] - \overline{\text{RSCP}}_u\langle b \rangle)^2.$$

If channel quality history does not exist for $u \in U_b$ at slot j, then $c(b,j) = \infty$. Note that the CHLS uses all three channel quality attributes to increase accuracy. Moreover, for each time slot, it utilizes channel quality attributes corresponding to several sectors. The scheme concludes by estimating that the user resides in $b^* = \text{argmin}_{b \in \mathbf{B}} h_{b,T_p}$. To complete step one, the user's velocity is estimated, using training data to compute an average of past velocities near the estimated location.

The CHLS was evaluated via simulations. We set $T_{\text{p}} = 3{,}000$ slots (which corresponds to a horizon of 60s), set the segment size to 25m×25m, and assumed that the serving sector coverage radius is 1,000m. We created coverage maps using half of the traces reported in Table II. From the remaining traces, we selected 500 random instances of 60s-length. The distribution of localization errors is shown in Fig. 8(a). The scheme has a median error of 23m and average error of 123m. For comparison, our evaluation of the RSSI-based localization scheme of [27] is shown in Fig. 8(b) and resulted in a median error over 300m.

C. Feasible Data Rate Prediction

Recall that the FPF Scheduling Problem formulation is based on feasible data rates. Hence, we now transition to using data rates. The relation between E_c/I_o and data rates (provided

in Appendix A) is monotonic, and therefore, the reproducibility conclusions from Section V also apply to data rates.

A simple online algorithm that operates in the sector estimates the user's future data rates using a coverage map and an estimate of the user's current location and velocity (either from GPS or the CHLS). First, future locations are predicted assuming that the velocity is constant for future time slots. Each location is then mapped to a segment in the coverage map which in turn yields a data rate.

Fig. 8(c) shows an example data rate prediction for the CMPM using the two variations, to which we refer as CMPM-GPS and CMPM-CHLS. In Section VIII, we demonstrate that the CMPM-CHLS captures enough of the slow-fading effects when integrated into the (PF)^{2}S Framework to improve scheduling performance.

VII. Allocation Estimation ($\hat{\boldsymbol{\alpha}}$)

The (PF)^{2}S Framework (described in Section IV) requires a channel allocation ($\hat{\boldsymbol{\alpha}}$) estimation algorithm based on the data rate predictions. This can be viewed as obtaining a solution to the FPF Scheduling Problem using the *predicted data rate matrix* $\hat{\mathbf{R}}$. As the framework operates in an online manner, the main design considerations are simplicity and robustness to prediction errors. We now introduce three algorithms which trade fairness and throughput performance for robustness to prediction errors. These algorithms will be evaluated with the rest of the framework in Section VIII.

Round Robin Estimation (RRE): This simple heuristic assumes that future time slots are allocated in a *round-robin* manner and each user receives an *equal* number of slots, resulting in an estimated allocation of $\hat{\alpha}_{ij} = 1/K \, \forall \, i, j$.

Blind Gradient Estimation (BGE): This heuristic utilizes (4) to select a user in each slot, but *without* the future component (since it is not known). Specifically, starting from $j = 1$, it sets $\hat{\alpha}_{i^* j} = 1$ where $i^* = \text{argmax}_{i \in K}(\hat{r}_{ij}) / \sum_{t=1}^{j} \hat{\alpha}_{it} \hat{r}_{it}$. The expression contains only slot indices $\leq j$, and is similar to PF-EXP. BGE requires $O(KT)$ operations.

Local Search Estimation (LSE): This greedy algorithm, described below, initiates with an objective function value C based on $\hat{\boldsymbol{\alpha}}$ composed of random values. It iterates slot-by-slot, greedily allocating slot j to the user with the largest gradient value, assuming all other time slots are fixed. The algorithm proceeds cyclically (returning to slot 1 after T) until reaching a local-maxima (i.e., no change in T iterations). Termination is guaranteed as the objective value is bounded from above. Each cycle of LSE takes $O(KT)$ computations. Practically, it usually terminates after a few cycles.

An example of $\hat{\boldsymbol{\alpha}}$ values obtained by each of the algorithms appears in Fig. 9(a). LSE's estimates are tightly clustered near the predicted slow-fading peaks. The estimates from BGE are more diffused and those of RRE are uniform. Consequently, if the rate predictions are accurate, the framework using LSE provides the best performance, since it correctly allocates slots near the peak rates. The framework using BGE allocates slots around the peak rates, resulting in moderately good performance. The framework with RRE allocates slots uniformly, occasionally occurring during the peaks. On the other hand, if the prediction is erroneous, LSE would suffer, since it pushes the framework to schedule the user's slots at the predicted slow-fading peaks. BGE provides some robustness to prediction errors, and RRE is the most robust.

Local Search Estimation (LSE) Algorithm

Input: Predicted data rates $\hat{\mathbf{R}} = \{\hat{r}_{ij}\}_{K \times T}$.
Output: Estimated allocations $\hat{\boldsymbol{\alpha}} = \{\hat{\alpha}_{ij}\}_{K \times T}$.
1: Choose an initial random $\hat{\boldsymbol{\alpha}}$.
2: $j = 1$, $LastChange = 1$, $C = \sum_{i=1}^{K} \log(\sum_{j=1}^{T} (\hat{r}_{ij} \hat{\alpha}_{ij}))$
3: **repeat** $i^* = \arg\max_{i \in K} r_{ij} / \sum_{t \in \{1,T\} \setminus j} (\hat{r}_{it} \hat{\alpha}_{it})$
4: $\hat{\alpha}_{i^*,j} = 1, \hat{\alpha}_{i,j} = 0 \quad \forall i \neq i^*$
5: $C' = \sum_{i=1}^{K} \log(\sum_{j=1}^{T} (\hat{r}_{ij} \hat{\alpha}_{ij}))$
6: **if** $(C' > C)$ **then** $C = C'$, $LastChange = j$
7: $j = (j \bmod T) + 1$
8: **until** $j \neq LastChange$

VIII. Performance Evaluation

We now use trace-based simulations to evaluate the performance of the (PF)^{2}S Framework described in Section IV. Framework *instances* use combinations of CMPM implementations and channel allocation estimation algorithms. The test cases are generated using measurement traces and the performance metrics are *proportional fairness* (1) and *throughput* (Defn. 1). We show that various instances of the (PF)^{2}S framework consistently outperform the deployed scheduler (PF-EXP), with throughput improvements in realistic scenarios ranging from 15% to 55%, while maintaining similar delay performance. We then study the framework's sensitivity to the time horizon, number of users, mobility, $\hat{\mathbf{R}}$ accuracy, delay threshold, and coverage map resolution.

A. Generation of Coverage Map and Test Cases

From the dataset presented in Section V, half of the drives on each route were used as training measurements for coverage map construction (using a 25m×25m segment size). The remaining measurements were used for test case generation. A single test case was generated for every sector that had enough measurement data. Each test case is comprised of K users and T time slots and it emulates users starting at different locations within the sector coverage area and traveling with varying velocities in both directions along a route.

For each user i, the data rates r_{ij}, $1 \leq j \leq T$ were generated by selecting a segment of T random contiguous slots from part of the trace where the user was associated with the sector. In half of the cases, the vector was time-reversed, emulating travel in the opposite direction. Fig. 9(b) shows the data rates (r_{ij}) for an example test case with $K = 7$ and $\tilde{T} = 30$s (recall that $\tilde{T}$ is the time horizon in seconds).

Finally, for each generated rate matrix $\mathbf{R}$, we consider 3 approaches for obtaining the predicted rate matrix $\hat{\mathbf{R}}$: clairvoyant (a.k.a., complete knowledge, $\hat{\mathbf{R}} = \mathbf{R}$), the CMPM which uses GPS information for location estimation (referred to as CMPM-GPS), and the CMPM which uses the CHLS (referred to as CMPM-CHLS). Using these approaches enables evaluating the framework with different qualities of $\hat{\mathbf{R}}$ prediction.

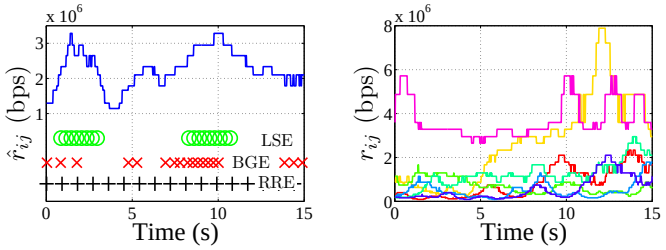


Fig. 9. (a) An example of predicted data rates for a user with $\tilde{T} = 15$s and the corresponding $\hat{\boldsymbol{\alpha}}$ estimations computed by the LSE, BGE, and RRE algorithms. (b) The data rates (r_{ij}) for a test case with $K = 7$ and $\tilde{T} = 15$s.

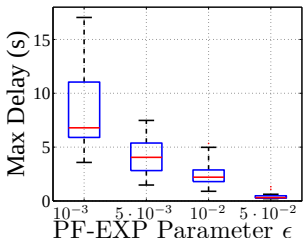


Fig. 10. Statistical characterization of the maximum delay for the PF-EXP algorithm for 30 test cases with $K = 5$, $\tilde{T} = 30$s, and varying ϵ.

B. Baseline Comparison and Upper Bound

The (PF)^{2}S Framework is compared to the deployed scheduler, PF-EXP (see Defn. 3), by normalizing the throughput and fairness values by the corresponding values obtained by PF-EXP. Hence, metric values greater than 1 show improvements over PF-EXP. As an upperbound, the optimal solution to the FPF problem (referred to as OPT) is obtained using CVX, a MATLAB solver [1]. Note that OPT is obtained ignoring the integer constraints (3), using $\mathbf{R}$, and without delay constraints.

As mentioned in Section IV, the parameter ϵ *implicitly* controls the throughput-delay tradeoff for the PF-EXP scheduler. The (PF)^{2}S Framework *explicitly* controls the throughput-delay tradeoff using the parameter D_{starved}. Unless otherwise specified, we fix $D_{\text{starved}} = 0.5$s. To ensure a fair comparison to our framework, we select a value of ϵ which results in the PF-EXP algorithm having similar delay performance. Correspondingly, based on box plots[15] of the PF-EXP maximum delay shown in Fig. 10, we select $\epsilon = 0.01$. We discuss the sensitivity to these parameters in Section VIII-D.

C. Throughput and Fairness Gains

We evaluate the throughput and fairness performance for various (PF)^{2}S Framework instances and confirm experimentally that the $\hat{\boldsymbol{\alpha}}$ estimation algorithms provide different degrees of robustness to rate prediction errors.

Figures 11(a) and 11(b) present box plots of the framework's fairness and throughput performance gains for 22 randomly generated test cases with $K = 7$ and $\tilde{T} = 30$s (gains greater than 1 indicate improvements over PF-EXP). Since the objective function is logarithmic, the fairness gains are at the order of a few percent. The throughput gains over PF-EXP for all framework instances are significant (up to 70%). Clearly, the performance of a framework instance depends on the rate prediction accuracy ($\hat{\mathbf{R}}$) and the channel allocation estimation ($\hat{\boldsymbol{\alpha}}$) algorithm. Hence, we consider framework instances, categorized by the $\hat{\mathbf{R}}$ prediction mechanism:

$\hat{\mathbf{R}}$ Clairvoyant: The throughput gains are substantial (20% to 70%). As expected, based on the framework instance performance, the estimation algorithms are ranked by LSE $>$ BGE $>$ RRE. In general, the LSE performance with complete knowledge was near optimal.[16]

$\hat{\mathbf{R}}$ from CMPM-GPS: Fig. 11(a) and 11(b) shows that the ranking between the $\hat{\boldsymbol{\alpha}}$ estimation algorithms is BGE $>$ RRE $>$ LSE. As described in Section VII, BGE provides relative robustness to prediction errors, and hence with CMPM-GPS it often outperforms LSE with throughput gains of 20% to 55%.

$\hat{\mathbf{R}}$ from CMPM-CHLS: The instances using LSE and BGE show the largest performance decrease (compared to using complete knowledge). Yet, they still result in gains over PF-EXP. In general, we found that RRE is most resilient to errors and results in *significant* throughput gains of 15% to 50%.

To confirm that the gains of the (PF)^{2}S Framework over the PF-EXP algorithm do not come at the expense of degraded user delay performance, Figures 11(c) and 11(d) illustrate the delay experienced by the users. Recall that the delay (see Defn. 2) refers to the duration in which a user receives no allocations. Fig. 11(c) shows the CDF of the delay experienced by users for the (PF)^{2}S Framework using BGE as well as the PF-EXP scheduler. Clearly, the delay performance of the framework is comparable to that of the PF-EXP scheduler, although the framework results in users having a slightly higher variation in the delay as indicated in Fig. 11(d). These results confirm that the (PF)^{2}S Framework provides similar delay performance as the PF-EXP scheduler.

In summary, the evaluations with *real-world measurements* show that practical (PF)^{2}S Framework instances match the PF-EXP scheduler in terms of delay performance, but consistently provide higher performance with throughput gains typically between 15% and 55%. The fairness gains, despite being logarithmic, are on the order of a few percent.

D. Sensitivity Analysis

The results below are for the framework using RRE and CMPM-CHLS (similar results for other framework instances provide equivalent conclusions and are therefore omitted).

Time Horizon ($\tilde{T}$): Fig. 12(a) shows the fairness and throughput gains for test cases with varying time-horizons $\tilde{T}$. Intuitively, larger $\tilde{T}$ provides the framework additional opportunities to benefit over PF-EXP, which does not account for future data rates. For small to moderate values of $\tilde{T}$ (5s,15s),

[15]Box plots include a whisker at maximum and minimum samples, a box at the 25th and 75th sample quantile, and a line at the sample median.

[16]For all test cases, when $D_{\text{starved}} = \infty$, the throughput when using LSE with complete knowledge is within 0.05% of the OPT throughput.

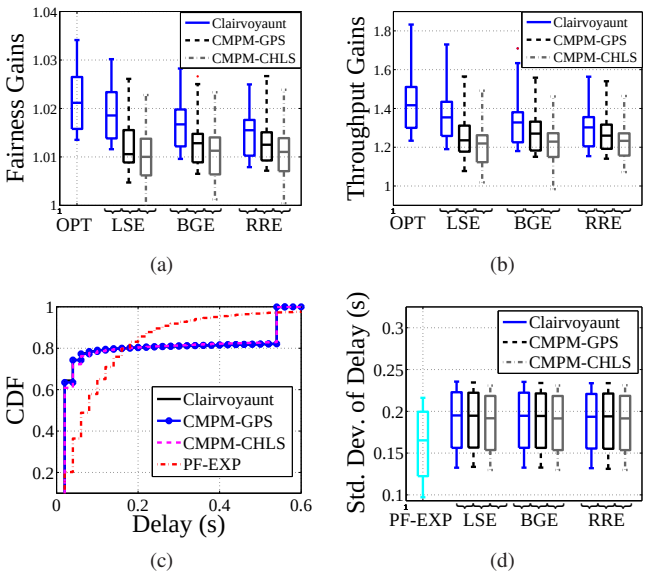


Fig. 11. $(PF)^2S$ Framework performance for various framework instances (combinations of $\hat{\mathbf{R}}$ and $\hat{\alpha}$ prediction algorithms): statistical evaluation of 22 test cases with $K = 7$ and $\tilde{T} = 30$s, and the resulting (a) fairness gains over PF-EXP, (b) throughput gains over PF-EXP, (c) cumulative distribution function of the users' delay for the BGE algorithm, and (d) standard deviation of the users' delay.

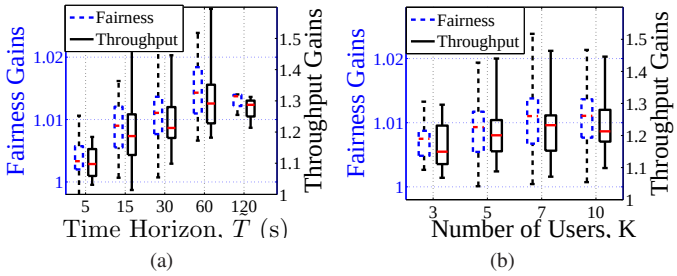


Fig. 12. $(PF)^2S$ Framework performance gains when using RRE with CMPM-CHLS: statistical evaluation of (a) 10 test cases with $K = 10$, varying the time horizon ($\tilde{T}$) and (b) 20 test cases of $\tilde{T} = 30$s, varying the number of users (K).

the framework shows 10%–30% throughput improvements. The performance gain for $\tilde{T} = 60$s increases to 20%–60%. Eventually, as $\tilde{T}$ grows, the framework becomes limited by the accuracy of the prediction, which decays with time.

Number of Users (K): Fig. 12(b) shows the fairness and throughput gains for 20 test cases with $\tilde{T} = 30$s, and varying number of users. With additional mobile users, each experiencing their own slow-fading channel, multi-user diversity increases and the performance improves. The throughput gains increase from up to 25% with 3 users to up to 45% with 10 users.

Effect of Mobility: To ascertain the affect of static users, we evaluate test cases created from the mobile and the *static (immobile)* measurements. With static measurements, shown in Fig. 13(a), the wireless channel state distribution is stationary. Fig. 13(b) considers the framework performance for all algorithms (with complete knowledge, as predictions are irrelevant in this case) in a test case with $K = 5$ and $\tilde{T} = 30$s. The framework performance is very similar to PF-EXP (with throughput gains within 6%) and is very close to OPT. Fig. 14 shows gains for 10 test cases of $\tilde{T} = 30$s with 10 mobile users *and a varying number of static users*. With the addition of static users, PF-EXP performance improves (approaches optimal), and therefore, the gains decrease. Yet, due to the 10 mobile users, the gains are still quite *significant*, with throughput gains of over 30% in some cases.

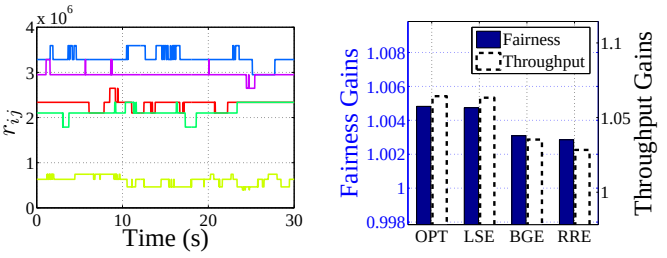


Fig. 13. $(PF)^2S$ Framework evaluation of a test case generated using static (immobile) measurements with $\tilde{T} = 30$s and $K = 5$: (a) the data rates r_{ij} and corresponding (b) fairness and throughput gains over PF-EXP using the BGE algorithm.

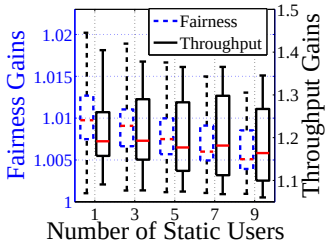


Fig. 14. $(PF)^2S$ Framework performance gains when using RRE with CMPM-CHLS: statistical evaluation of 10 test cases with $\tilde{T} = 30$s, 10 mobile users, and varying number of static users.

Slow-fading Peak Prediction: As indicated, the accuracy of the $\hat{\mathbf{R}}$ prediction impacts the framework performance. Through careful inspection, we found that a key factor in prediction accuracy is the location of the slow-fading peaks. Hence, we now consider the impact of a simulated $\hat{\mathbf{R}}$ on the framework performance.

Specifically, the predicted rate vector for a subset of users is shifted by a certain amount, e.g., $\hat{r}_{ij} = r_{i(j-\text{offset})}$. Fig. 15 shows the fairness and throughput gains of the framework using RRE with subsets of the $K = 5$ users having an offset rate prediction. As expected, the performance decreases with large values of offset (both lead and lag). However, they are still quite significant ($\approx$ 20% throughput gains). This experiment suggests that predicting the slow-fading peak within a few seconds of the actual peak will result in significant performance improvements. Additional tests confirm that this holds in more general scenarios.

Delay Parameters ($D_{\text{starved}}, \epsilon$): As described in Section IV,

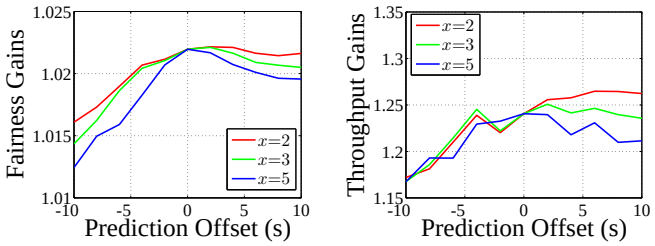


Fig. 15. (PF)^{2}S Framework evaluation for an example test case with $K = 5$ users, $\tilde{T} = 30$s: (a) fairness and (b) throughput gains over PF-EXP using the BGE algorithm for varying the number of users, x, with simulated data rate prediction offsets.

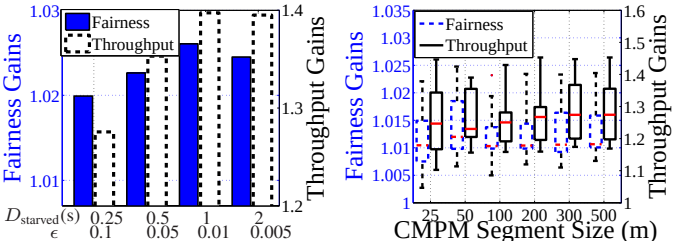


Fig. 16. (PF)^{2}S Framework performance gains when using RRE with the CMPM-CHLS: (a) varying the delay parameters for the (PF)^{2}S Framework (D_{starved}) and the PF-EXP scheduler (ϵ) for the test case given in Fig. 9(b), and (b) statistical evaluation of 15 test cases with $K = 10$ and $\tilde{T} = 30$s for varying coverage map resolution values.

the framework uses the delay threshold D_{starved} to prioritize 'starved' users. The PF-EXP scheduler uses ϵ to implicitly control the delay. We now explore the performance gains achieved as we relax the delay parameters on both the PF-EXP scheduler (ϵ) and the (PF)^{2}S Framework (D_{starved}). In Fig. 16(a), we vary these parameters (matching them based on Fig. 10). The figure shows, for comparable delay parameters over a range of delay tolerances (0.25–2s), that the (PF)^{2}S Framework continues to outperform the PF-EXP scheduler, with the gains only mildly varying.

Coverage Map Resolution: In the results above, the coverage map segment size is 25m×25m. Fig. 16(b) shows the framework gains for 15 test cases with $K = 10$ and $\tilde{T} = 30$s as a function of the map segment size. The performance does not degrade significantly as the segment size becomes reasonably large, since larger segments result in averaging of channel quality attributes over a larger area. This indicates that coarse channel measurements are useful for the framework.

IX. Conclusions and Future Work

We described an extensive wireless measurement study as well as the design and trace-based performance evaluation of the (PF)^{2}S Framework. We showed that by leveraging slow-fading, the framework (composed of various algorithms) can provide significant throughput gains while improving or maintaining fairness levels. Finally, we investigated the sensitivity of the results to different parameters and assumptions.

Future work will focus on relaxing some of the assumptions. Particularly, we plan to consider dynamic user populations handing-off between sectors. Additionally, we plan to extend the evaluations to consider policies that select appropriate $\hat{\boldsymbol{\alpha}}$ estimation algorithms in different scenarios.Moreover, we will extend the localization scheme for cases in which trajectory information is unavailable or limited. Finally, as 4G networks become ubiquitous, we will conduct a corresponding measurement study to tailor the resource allocation algorithms.

Acknowledgements

We thank Dr. H. Karloff for discussions regarding the (PF)^{2}S Framework. We also thank the anonymous reviewers for their insightful comments.

References

[1] CVX: Matlab Software for Disciplined Convex Programming, v2.0 beta.

[2] 3rd Generation Partnership Project, "3GPP specification detail: Physical layer procedures (FDD)," http://www.3gpp.org/ftp/Specs/html-info/25214.htm.

[3] S. H. Ali, V. Krishnamurthy, and V. C. Leung, "Optimal and approximate mobility-assisted opportunistic scheduling in cellular networks," *IEEE Trans. Mobile Comput.*, vol. 6, no. 6, pp. 633–648, June 2007.

[4] J. Andersen, T. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communications channels," *IEEE Commun. Mag.*, vol. 33, no. 1, pp. 42–49, 1995.

[5] M. Andrews and L. Zhang, "Scheduling over nonstationary wireless channels with finite rate sets," *IEEE/ACM Trans. Networking*, vol. 14, no. 5, pp. 1067–1077, Oct. 2006.

[6] M. Andrews, "A survey of scheduling theory in wireless data networks," in *Wireless Communications*, ser. The IMA Volumes in Mathematics and its Applications. Springer, 2007, vol. 143, pp. 1–17.

[7] H. Bang, T. Ekman, and D. Gesbert, "Channel predictive proportional fair scheduling," *IEEE Trans. Wireless Commun.*, vol. 7, no. 2, pp. 482–487, Feb. 2008.

[8] P. Barford, J. Kline, D. Plonka, and A. Ron, "A signal analysis of network traffic anomalies," in *Proc. ACM SIGCOMM'02*, 2002.

[9] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[10] S. Borst, N. Hegde, and A. Proutiere, "Mobility-driven scheduling in wireless networks," in *Proc. IEEE INFOCOM'09*, Apr. 2009.

[11] T. Bu, L. Li, and R. Ramjee, "Generalized proportional fair scheduling in third generation wireless data networks," in *Proc. INFOCOM'06*, 2006.

[12] G. Chandrasekaran, T. Vu, A. Varshavsky, M. Gruteser, R. Martin, J. Yang, and Y. Chen, "Tracking vehicular speed variations by warping mobile phone signal strengths," in *Proc. IEEE PerCom'11*, Mar. 2011.

[13] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990.

[14] J. Hajipour and V. C. M. Leung, "Proportional fair scheduling in multi-carrier networks using channel predictions," in *Proc. IEEE ICC'10*, May 2010.

[15] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Trans. Veh. Technol.*, vol. 29, pp. 317–325, 1980.

[16] H. Holma and A. Toskala, Eds., *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications*, 1st ed. John Wiley & Sons, Inc., 2001.

[17] M. Ibrahim and M. Youssef, "CellSense: An accurate energy-efficient GSM positioning system," *IEEE Trans Vehic. Techn.*, vol. 61, no. 1, pp. 286–296, Jan. 2012.

[18] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, pp. 237–252, 1998.

[19] J. Krumm and E. Horvitz, "Predestination: Where do you want to go today?" *IEEE Computer*, vol. 40, no. 4, pp. 105–107, 2007.

[20] H. Kushner and P. Whiting, "Convergence of proportional-fair sharing algorithms under general conditions," *IEEE Trans. Wireless Commun.*, vol. 3, no. 4, pp. 1250–1259, July 2004.

[21] X. Liu, A. Sridharan, S. Machiraju, M. Seshadri, and H. Zang, "Experiences in a 3G network: interplay between the wireless channel and applications," in *Proc. ACM MobiCom'08*, Sept. 2008.

[22] X. Long and B. Sikdar, "A wavelet based long range signal strength prediction in wireless networks," in *Proc. IEEE ICC'08*, May 2008.

[23] R. Margolies, A. Sridharan, V. Aggarwal, R. Jana, N. Shankaranarayanan, V. Vaishampayan, and G. Zussman, "Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms," in *Proc. IEEE INFOCOM'14, to appear*, Apr. 2014.

[24] L. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. Prentice Hall, 1993.

[25] T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, 2001.

[26] Samsung Electronics Co., "Samsung Galaxy S II," http://www.samsung.com/global/microsite/galaxys2/html/specification.html.

[27] A. Schulman, V. Navda, R. Ramjee, N. Spring, P. Deshpande, C. Grunewald, K. Jain, and V. N. Padmanabhan, "Bartendr: a practical approach to energy-aware cellular data scheduling," in *Proc. ACM MobiCom'10*, Sept. 2010.

[28] A. Stolyar, "On the asymptotic optimality of the gradient scheduling algorithm for multiuser throughput allocation," *Operations Research*, vol. 53, pp. 12–25, 2005.

[29] T. Su, H. Ling, and W. Vogel, "Markov modeling of slow fading in wireless mobile channels at 1.9 GHz," *IEEE Trans. on Antennas and Propag.*, vol. 46, no. 6, pp. 947–948, June 1998.

[30] J. Yao, S. S. Kanhere, and M. Hassan, "An empirical study of bandwidth predictability in mobile computing," in *Proc. ACM WiNTECH'08*, Sept. 2008.

[31] ——, "Geo-intelligent traffic scheduling for multi-homed on-board networks," in *Proc. ACM MobiArch'09*, June 2009.

APPENDIX A
E_c/I_o TO DATA RATE CALCULATION

The Samsung Galaxy S II (GSII) phones [26] compute the E_c/I_o value for each serving sector. Based on empirical measurements, we mapped the E_c/I_o values to Channel-Quality-Indicator (CQI) values; measurements of E_c/I_o and CQI were taken simultaneously and, using a best-fit line, we computed a linear relationship between the two parameters. In conjunction with the phone's category, the CQI metric, which is an integer between 0-30, determines the available data rate to the phone. The GSII phones are category 14, and therefore, we used the mapping supplied in the 3GPP specifications [2] and shown in Fig. 17.

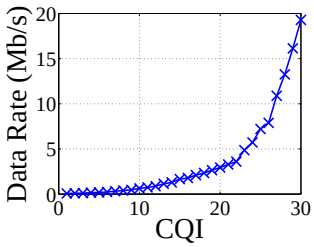


Fig. 17. Mapping of Channel-Quality-Indicator (CQI) to feasible data rate for Galaxy S II phones, obtained from the 3GPP specifications [2].

APPENDIX B
THE FPF SCHEDULING PROBLEM IS NP-HARD

We show that the FPF Scheduling Problem is NP-hard even for the case of 2 users by a reduction from the Set Partition Problem, a classic NP-hard problem [13].

Theorem 1: The FPF Scheduling Problem is NP-hard.

Proof: Given a set of positive integers $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$, let $\mathcal{X}, \mathcal{A}\backslash\mathcal{X}$ be a partition of $\mathcal{A}$ into two disjoint sub-sets. Let $S_{\mathcal{B}} = \sum_{i\in\mathcal{B}} a_i$. The decision version of the Set Partition problem seeks to determine if there exists a set $\mathcal{X}$ such that $S_{\mathcal{X}} = S_{\mathcal{A}\backslash\mathcal{X}}$. The decision version of the FPF Scheduling Problem asks if there exists an allocation such that the value of the objective function, (1), is at least C.

We construct a special case of the decision version of the FPF Scheduling Problem by setting $K = 2$ users and $C = 2\log(S_{\mathcal{A}}/2)$. Each user's channel has n slots with *identical* rates equal to $\mathcal{A}$. Specifically, in slot 1, both users have rate a_1, in slot 2, users have rate a_2 and so on. The order of the rates is not critical, only that each user has identical rates in a slot, and as many rates as there are integers in $\mathcal{A}$.

We first show that the forward direction that a yes-instance of the Set Partition Problem results in a yes-instance of the FPF problem. Clearly, the existence of a set partition $\mathcal{X}^*$ implies that $S_{\mathcal{X}^*} = S_{\mathcal{A}\backslash\mathcal{X}^*} = S_{\mathcal{A}}/2$. In the corresponding schedule from the FPF problem, let user 1 be allocated every slot corresponding to the items in $\mathcal{X}^*$ and user 2 be allocated the remaining slots corresponding to the items in $\mathcal{A}\backslash\mathcal{X}^*$. The value of (1) in this scenario is $\log(S_{\mathcal{X}^*}) + \log(S_{\mathcal{A}\backslash\mathcal{X}^*}) = 2\log(S_{\mathcal{A}}/2)$. Therefore, this is a yes-instance of the FPF Problem.

Now consider the reverse direction. Assume that there does not exist a set partition. We argue by contradiction that this corresponds to a no-instance in the FPF Scheduling Problem. Assume there exists a schedule in the FPF problem with value $C = 2\log(S_{\mathcal{A}}/2)$. Let S_1 represent the sum of the rates allocated to user 1 and S_2 represent the sum of the rates allocated to user 2. As $S_1 + S_2 = S_{\mathcal{A}}$, it is clear that to achieve the optimal value of $C = 2\log(S_{\mathcal{A}}/2)$, $S_1 = S_2 = S_{\mathcal{A}}/2$. However, this is a contradiction of the non-existence of a set partition. Therefore, a no-instance of the Set Partition Problem corresponds to a no-instance of the FPF Problem.

■

APPENDIX C
E_c/I_o SMOOTHING - REVIEW OF WAVELET TRANSFORMS

The E_c/I_o signal is smoothed by applying a Discrete Wavelet Transform (DWT) which is ideally suited for finite, non-stationary signals. The approach follows the standard methodology (see [8] for an excellent description), which we briefly illustrate here. A DWT can be viewed as successive applications of a low-pass filter L and its mirror high-pass filter H, along with sampling to construct successively 'coarser'

signal levels in terms of both time and frequency. For our purposes, a Haar wavelet is used for the L and H filters.

Specifically, consider a signal x of length N sampled at rate f Hz. In an intermediate step, both the L and H filters are applied to obtain the coarse and high frequency signals, denoted by $L(x), H(x)$, respectively. Each element of $L(x)$ and $H(x)$ is a wavelet coefficient and the two sequences are called the approximation and detail coefficients, respectively. $L(x)$ captures the 'lower' half of the frequency components, while $H(x)$ captures the 'upper' half of the frequency components. As each of the sequences are filtered to half of the bandwidth $(f/2)$, they are sampled so that only $N/2$ samples are retained (as required by the Nyquist-Shannon sampling theorem).

Following this, the process continues by generating the sequence of wavelet vectors, $H(X), LH(x), L^2H(x), L^3H(x), \ldots, L^{\log_2 N}H(x)$. At level j, the wavelet $L^jH(x)$ has length of $N \times 2^{-j}$ and a bandwidth of $f/2^j$. Equivalently, since the samples in the original signal were sampled at rate f Hz, and hence are $\tau = 1/f$ seconds apart, the values at level j are $2^{j-1} \times \tau$ seconds apart. Hence, the signals at level j are 'smoothed', with variations at time-scales smaller than $2^{j-1} \times \tau$ removed.

The wavelet vectors across all the $\log_2 N$ levels form a complete orthonormal basis of the original signal.[17] The measurements were collected at a time-interval of $\tau = 20$ milliseconds. To remove variations less than one-second, the wavelet vectors below level $\lceil \log_2 \frac{1}{\tau} \rceil \approx 6$ were removed. The remaining vectors $LH^7, \ldots, LH^{\log_2 N}$ were recombined to obtain the smoothed signal.

Robert Margolies is a Ph.D. student in Electrical Engineering at Columbia University. He received the B.S. degree (Summa Cum Laude) from Rensselaer Polytechnic Institute in 2010 and the M.S. degree from Columbia University in 2011, both in Electrical Engineering. His research interests are in cross-layer PHY and MAC layer design for ultra-low-power networks, networks of energy harvesting nodes, and opportunistic cellular scheduling. Robert has received a number of awards including the NSF Graduate Research Fellowship.

Ashwin Sridharan is a Principal Member of Technical Staff at AT&T Labs-Research where he works on wireless performance analysis and network data mining. He obtained his Masters in Electrical Engineering from the Indian Institute of Science in 1999 followed by a summer stint as a Nokia Grant Student. In 2004, he obtained his Ph.D. in Electrical Engineering from the University of Pennsylvania. Ashwin has received the Deans Fellowship at the University of Pennsylvania, the Nokia Grant Scholarship at the Indian Institute of Science, and is a member of IEEE and ACM.

[17]The original signal can be re-constructed by appropriately weighted combinations of the approximation and detail coefficients.

Vaneet Aggarwal (S'08 - M'11) received the B.Tech. degree in 2005 from the Indian Institute of Technology, Kanpur, India, and the M.A. and Ph.D. degrees in 2007 and 2010, respectively from Princeton University, all in Electrical Engineering. He is currently a Senior Member of Technical Staff at AT&T Labs-Research and an Adjunct Assistant Professor at Columbia University. His research interests are in applications of information and coding theory to wireless systems and distributed storage systems. Dr. Aggarwal was the recipient of Princeton University's Porter Ogden Jacobus Honorific Fellowship in 2009.

Rittwik Jana is a Lead member of Technical Staff at AT&T Labs-Research. He received a Ph.D. degree from the Australian National University in 1999. He worked as an engineer at the Defense Science and Technology Organization (DSTO), Australia from 1996 to 1999 and as a member of technical staff at AT&T, New Jersey from 1999 to date. His research expertise falls in the areas of wireless channel modeling, IPTV, P2P and mobile middleware.

N. K. Shankaranarayanan (S'83-M'92-SM'98) received the B.Tech degree from IIT Bombay in 1985, the M.S. degree from Virginia Tech in 1987, and the Ph.D degree from Columbia University in 1992, all in electrical engineering. Since 1992, he has worked at AT&T (Bell) Labs on research projects that have spanned service and technology aspects of cellular, WiFi, fixed wireless, cable and optical networks. Shankar has over 20 patents in his name and is currently serving as a Principal Member of Technical Staff-Research with AT&T Labs-Research.

Vinay Vaishampayan obtained a B. Tech. degree from the Indian Institute of Technology, Delhi, India and M.S. and Ph.D. degrees from the University of Maryland, College Park, USA, all in electrical engineering. His research interests are in communications, signal processing, information theory and statistical modeling. He was with AT&T Labs-Research (1996-2013), where he served as the head of the communication sciences research department and as Distinguished Member of Technical Staff. He is now Professor, Engineering Science and Physics, at the City University of New York, College of Staten Island. He was elected IEEE Fellow for his contributions to data compression.

Gil Zussman (S'02-M'05-SM'07) received the Ph.D. degree in electrical engineering from the Technion in 2004 and was a postdoctoral associate at MIT in 2004–2007. He is currently an Associate Professor of Electrical Engineering at Columbia University. He is a corecipient of 5 paper awards including the ACM SIGMETRICS'06 Best Paper Award and the 2011 IEEE Communications Society Award for Outstanding Paper on New Communication Topics. He received the Fulbright Fellowship, the DTRA Young Investigator Award, and the NSF CAREER Award, and was a member of a team that won first place in the 2009 Vodafone Foundation Wireless Innovation Project competition.