Distributionally Robust Slow Adaptive OFDMA with Soft QoS via Linear Programming

Anthony Man-Cho So, *Member, IEEE,* and Ying Jun (Angela) Zhang, *Senior Member, IEEE*

***Abstract*—Being the predominant air interface of next-generation wireless standards, orthogonal frequency division multiple access (OFDMA) is well known for its flexibility in allocating subcarriers to different mobile users according to their different fast channel variations. Numerous research studies have demonstrated that OFDMA can bring substantial capacity gain when the subcarriers are *optimally* allocated. Nonetheless, practical systems can hardly afford *optimal* subcarrier allocation, because frequent re-optimization performed at the same timescale as fast fading variation would lead to excessively high computational and signaling costs. As a result, most practical systems settle for low-complexity schemes that operate far from the optimum, thus making them unable to enjoy the large capacity gain predicted by theoretical studies. To address this problem, we propose a novel alternative, termed the *slow* adaptive OFDMA, to drastically reduce the computational and signaling costs. The proposed scheme adapts subcarrier allocation at a much slower timescale than that of channel fading variation, yet achieves similar system capacity and quality of service (QoS) levels as the optimal fast adaptive OFDMA. Moreover, it possesses several attractive features. First, neither prediction of channel state information nor specification of channel fading distribution is needed for subcarrier allocation. As such, the algorithm is robust against any mismatch between actual channel state/distributional information and the one assumed. Secondly, although the optimization problem arising from our proposed scheme is non-convex in general, based on recent advances in chance-constrained optimization, we show that it can be approximated by a certain linear program with provable performance guarantees. In particular, we only need to handle an optimization problem that has the same structure as the fast adaptive OFDMA problem, yet we are able to enjoy lower computational and signaling costs. Last but not the least, instead of relying on standard but abstract linear program solvers such as the interior-point method to solve the aforementioned linear program, we can exploit its special structure and design a provably efficient algorithm for solving it. Consequently, the proposed algorithm not only has a transparent engineering interpretation but is also easy to implement at the base stations of practical systems.**

***Index Terms*—OFDMA, Adaptive resource allocation, Stochastic optimization, Chance-constrained programming**

I. Introduction

A. Motivation

Various next-generation broadband wireless systems, such as IEEE WiMax, 3GPP-LTE and LTE-advanced, have converged to adopt orthogonal frequency division multiple access (OFDMA), the multiuser version of the popular OFDM scheme, as the main air interface. To extract the utmost efficiency out of limited radio resources, tremendous research efforts have been invested in the study of adaptive OFDMA (see, e.g., [14], [16]), which dynamically allocates subcarriers to different users according to their instantaneous channel fading conditions. It has been convincingly demonstrated that adaptive OFDMA can greatly enhance spectrum and energy efficiency if subcarriers are *optimally* allocated to fully exploit the fast variation of wireless channel fading.

To keep track of the fast fading variation, adaptive OFDMA must re-optimize subcarrier allocation at least as often as channel coherence time (which is in the order of milliseconds), thus resulting in excessively high computational complexity. Moreover, control signaling has to be sent frequently between the BS and mobile users in order to keep the users informed of the latest allocation decision. The overhead thus incurred is likely to negate the diversity gain obtained by the adaptation schemes. As a result, most if not all practical systems settle for low-complexity heuristics that operate far from the optimum, which makes them unable to enjoy the large capacity gain predicted by theoretical studies.

To enhance the practicality of adaptive OFDMA without compromising the capacity gain, we propose a novel alternative called the *slow* adaptive OFDMA. As the name suggests, the proposed scheme adapts subcarrier allocation on a much *slower* timescale than that of fading fluctuation, thus substantially reducing the computational cost and signaling overhead. The key challenge here is to achieve similar system capacity and quality of service (QoS) levels as the fast adaptation scheme, now that the channel fading condition can fluctuate drastically in between two successive subcarrier allocations.

B. Advantage over Slow Adaptation Schemes Based on Statistical Averages

The idea of slow adaptation has recently been pursued in different contexts, including slow adaptive modulation [5], [10] and slow adaptive power control [13]. The concept of slow adaptive OFDMA was introduced for the first time in our recent work [8]. In these papers, adaptation decisions are made solely based on the long-term average channel conditions instead of fast channel fading. Specifically, random channel parameters are replaced by their mean values, resulting in a deterministic rather than stochastic optimization problem. However, such approaches fail to exploit the diversities that potentially reside in higher order channel statistics (e.g.,

This work was supported in part by the Hong Kong Research Grants Council (RGC) General Research Fund (GRF) Projects CUHK 416012 and CUHK 419509.

A. M.-C. So is with the Department of Systems Engineering and Engineering Management and, by courtesy, Department of Computer Science and Engineering and the CUHK-BGI Innovation Institute of Trans-omics, The Chinese University of Hong Kong, Hong Kong (email: manchoso@se.cuhk.edu.hk).

Y. J. Zhang is with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong (email: yjzhang@ie.cuhk.edu.hk).

channel variance). Moreover, QoS can only be guaranteed in a long-term average sense, since the short-term fluctuation of channel is completely ignored in the problem formulation. On the other hand, fast adaptation schemes are able to provide short-term QoS (e.g., short-term data rate) guarantee, which is important to delay-sensitive applications, such as wireless multimedia applications.

In contrast to [5], [8], [10], [13], our proposed scheme incorporates statistical properties of the channel into the resource allocation decisions. In particular, it aims at maximizing the long-term system throughput, while satisfying short-term data rate requirements with probability at least $1-\epsilon$, where $\epsilon \in [0,1)$ is a user-defined outage tolerance parameter. In the special case where $\epsilon = 0$, the formulation reduces to the well-studied robust optimization formulation. By allowing ϵ to pass from 0 to a small positive number, one can have a tradeoff between system performance and QoS levels, a flexibility that is absent in previous schemes. This formulation, which contains *probabilistic* constraints on short-term data rate requirements, is particularly well suited for next-generation broadband applications that require high throughput and stringent short-term QoS, yet can typically tolerate occasional outages. Moreover, the formulation can be explicitly expressed as a *probabilistic* or *chance-constrained programming* problem. Solving the chance-constrained programming problem, however, is mathematically challenging, for the problem is non-convex in general [11].

C. Distributional Robustness and Simple Implementation

To deal with probabilistic constraints, one typically would need some information about the probability distribution of the underlying random data. In our recent work on slow adaptive OFDMA with probabilistic short-term QoS [9], we assume that the channel follows the Rayleigh fading model and formulate a convex approximation of the non-convex probabilistic constraint. In practice, however, the precise channel distribution information is difficult to obtain. Very often, channel distributions that arise from simplifying models such as Rayleigh fading may not match the reality perfectly [6], [7]. Such misspecification of distributional information may compromise the optimality of the solution.

In contrast to [9], the scheme proposed in this paper does not require any explicit knowledge of the channel fading distribution. Instead, it only assumes that the channel fading process can be *sampled*. As such, the scheme is robust against any misspecification of the probability distribution of channel fading. Note that most wireless systems are constantly estimating the channel coefficients for coherent detection and adaptive modulation. In other words, the system is already "sampling" channel fading process as part of the operation. Thus, the above sampling assumption can be easily justified.

Another appealing aspect of the proposed scheme is that subcarrier allocation decisions are obtained as solutions to a *linear program*. Such a linear program approximates the original chance-constrained programming formulation, and the approximation quality can be rigorously established. To further simplify the implementation, instead of relying on standard linear program solvers to solve the aforementioned linear program, we present an intuitive yet provably efficient solution algorithm called the "Biggest Bang for the Buck" (BBB). The BBB algorithm involves only simple operations and has a transparent engineering interpretation. As such, it is well suited for implementation at base stations. Our results show that the proposed slow adaptive OFDMA can achieve a significant portion of the fast adaptive OFDMA throughput with drastically lower implementation cost. Depending on the channel fading distribution, slow adaptive OFDMA may even achieve higher throughput than the fast adaptation scheme, thanks to its low control signaling overhead.

The rest of the paper is organized as follows. In Section II, we discuss the channel model and problem formulation. In Section III, we present the proposed linear programming based slow adaptive OFDMA scheme and establish its soundness and performance. In Section IV, we develop the BBB algorithm for solving the linear programs arising from the adaptive schemes. In Section V, we investigate the performance of the proposed scheme through numerical simulations and discuss the possibility of reducing the number of samples. We end with some concluding remarks in Section VI.

II. System Model and Problem Formulation

A. System Model

In this paper, we consider a single-cell multiuser OFDM system with K users and N subcarriers. Let $h_{k,n}^{(t)}$ be the random variable representing the instantaneous channel coefficients of user k on subcarrier n at time t, and $g_{k,n}^{(t)} = \left|h_{k,n}^{(t)}\right|^2$ be the instantaneous channel gain. Depending on the nature of the radio propagation channel, the random variable $h_{k,n}^{(t)}$ may follow different kinds of distributions. Among the commonly adopted channel statistical models are Rayleigh, Rician, Nakagami, and Weibull distributions. In this paper, however, we shall not make any assumption on the channel distribution.

For simple implementation, we assume that the transmission power p_t on each subcarrier is fixed[1]. The transmission rate of user k on subcarrier n is adapted as

$$r_{k,n}^{(t)} = W \log_2 \left(1 + \frac{p_t g_{k,n}^{(t)}}{\Gamma N_0} \right),$$

where W is the bandwidth of a subcarrier, N_0 is the power spectral density of Gaussian noise, and Γ is the capacity gap that is related to the target bit error rate (BER) and coding-modulation schemes. Without loss of generality, we assume that $W = 1$ and $\Gamma = 1$ for the rest of this paper.

B. Problem Formulation

In traditional fast adaptive OFDMA systems, subcarrier allocation (SCA) decisions are made at the base station (BS)

[1]Needless to say, system performance can be further enhanced if transmission power is also variable. However, varying transmission power not only adds to the computational complexity of finding the optimal allocation solution, but also complicates the hardware implementation. Moreover, it has been shown that power adaptation does not bring much additional gain if mobile users are already assigned "good" channels. Thus, we keep the power fixed in this paper.

according to the instantaneous channel gains in order to maximize a system-wide efficiency metric. For simplicity, we adopt system throughput as the efficiency metric in this paper. Since the transmission power (hence the energy consumption per unit time) is fixed in our system, maximizing system throughput also maximizes the energy efficiency (in terms of number of bits transmitted per unit energy consumption) of the system. Nonetheless, the idea of slow adaptive OFDMA can be extended to other objective functions.

Assuming that the BS knows the instantaneous channel gain $g_{k,n}^{(t)}$ (and hence $r_{k,n}^{(t)}$), a fast adaptive OFDMA system solves at time t the following linear programming problem:

$$\mathcal{P}_{\text{fast}}: \quad \max_{\{x_{k,n}^{(t)}\}} \quad \sum_{k=1}^{K}\sum_{n=1}^{N} x_{k,n}^{(t)} r_{k,n}^{(t)} \tag{1a}$$

$$\text{s.t.} \quad \sum_{n=1}^{N} x_{k,n}^{(t)} r_{k,n}^{(t)} \geq q_k, \quad \forall k, \tag{1b}$$

$$\sum_{k=1}^{K} x_{k,n}^{(t)} \leq 1, \quad \forall n,$$

$$x_{k,n}^{(t)} \geq 0, \quad \forall k, n,$$

where $x_{k,n}^{(t)}$ is the fraction of airtime assigned to user k on subcarrier n. The objective function in (1a) represents the total system throughput at time t, and (1b) represents the data rate constraint of user k at time t, where q_k is the minimum required data rate. Note that one has to solve $\mathcal{P}_{\text{fast}}$ every time the channel gain $g_{k,n}^{(t)}$ (and hence $r_{k,n}^{(t)}$) changes in order to keep the SCA decisions optimal. In practice, $g_{k,n}^{(t)}$ varies on the order of channel coherence time, which ranges from a few milliseconds to tens of milliseconds. Thus, fast adaptive OFDMA schemes can be extremely costly in practice.

One effective way to overcome the practicality issue of fast adaptive OFDMA systems is to adopt a slow adaptation scheme, in which SCA decisions are only updated once in every "adaptation window" of length T [9]. More precisely, an SCA decision is made at the beginning of each adaptation window, and the allocation remains unchanged till the next window. In this case, the SCA variable becomes $x_{k,n} \in [0,1]$ (i.e., there is no superscript (t) as opposed to the fast adaptation formulation), because SCA here is no longer a function of t during an adaptation window. Unlike in fast-adaptive systems where the BS needs exact CSI for resource allocation, in a slow adaptive OFDMA system, the BS does not have the luxury of foreseeing the channel realizations over the entire adaptation window when it performs SCA at the beginning of the window. Thus, it has to rely on information about the channel fading distribution to make the optimal subcarrier allocation decision. More precisely, for any given $\{x_{k,n}\}$, the quantity $\sum_{n=1}^{N} x_{k,n} r_{k,n}^{(t)}$, which appears in both the objective and constraint functions, is now a random variable. A fundamental problem here is to understand what constitutes a "good" SCA decision for each window and how such a decision can be computed.

Suppose that the duration T of a window is large compared with the fast fading fluctuation so that the channel fading process over the window is ergodic, yet small compared with the large-scale variation of path loss and shadowing so that the *probability distribution* of the channel remains unchanged during a window. Then, the system throughput during a window $[t_0, t_0+T]$ is given by

$$\frac{1}{T}\int_{t_0}^{t_0+T}\left(\sum_{k=1}^{K}\sum_{n=1}^{N} x_{k,n} r_{k,n}^{(t)}\right)dt = \sum_{k=1}^{K}\sum_{n=1}^{N} x_{k,n}\mathbb{E}\left\{r_{k,n}^{(t)}\right\},$$

where the equality is due to the ergodicity of channel fading over the window, and the expectation is taken over the random channel process $r = \{r_{k,n}^{(t)}\}$, for $t \in [t_0, t_0+T]$. Note that $\mathbb{E}\left\{r_{k,n}^{(t)}\right\}$ does not vary with t for $t \in [t_0, t_0+T]$, because the distribution of the channel remains unchanged during a window.

As in the fast adaptation scenario, we wish to satisfy each user's *short-term* data rate requirement q_k for each time t. To circumvent the randomness in $\sum_{n=1}^{N} x_{k,n} r_{k,n}^{(t)}$ and considering the fact that many wireless applications allow a small outage probability in their QoS requirements, we replace the short-term data rate constraint by the probabilistic constraint

$$\Pr\left\{\sum_{n=1}^{N} x_{k,n} r_{k,n}^{(t)} \geq q_k \ \forall k\right\} \geq 1-\epsilon, \quad \forall t \in [t_0, t_0+T], \tag{2}$$

where $\epsilon \in [0,1)$ is the maximum tolerable system outage probability[2]. In the special case where $\epsilon = 0$, the constraint (2) becomes a robustness constraint, i.e., the instantaneous data rate constraints must be satisfied at all times. In practice, we can choose ϵ depending on the QoS requirement of the particular application.

With the above considerations, the slow adaptive OFDMA problem can be formulated as follows:

$$\mathcal{P}_{\text{slow}}: \quad \max_{\{x_{k,n}\}} \quad \sum_{k=1}^{K}\sum_{n=1}^{N} x_{k,n}\mathbb{E}\left\{r_{k,n}^{(t)}\right\}$$

$$\text{s.t.} \quad \Pr\left\{\sum_{n=1}^{N} x_{k,n} r_{k,n}^{(t)} \geq q_k \ \forall k\right\} \geq 1-\epsilon, \tag{3a}$$

$$\sum_{k=1}^{K} x_{k,n} \leq 1, \quad \forall n,$$

$$x_{k,n} \geq 0, \quad \forall k, n.$$

$\mathcal{P}_{\text{slow}}$ is a so-called chance-constrained program because it contains the chance constraint (3a).

Before leaving this subsection, we should emphasize that many slow adaptation schemes simply replace $\sum_{n=1}^{N} x_{k,n} r_{k,n}^{(t)}$ by its mean $\sum_{n=1}^{N} x_{k,n}\mathbb{E}\{r_{k,n}\}$ in the data rate constraint. Although such an approach still gives *linear* data rate constraints, those constraints are only imposed on *long-term* average data rates. By contrast, the chance constraints in (3a) allow us to work on *short-term* data rate requirements, which are important for many wireless applications.

[2]The system is said to be in outage if there is a user whose instantaneous data rate is below the prescribed threshold q_k.

C. Challenges

Optimization problems with chance constraints are generally computationally intractable, as the feasible set defined by chance constraints is most likely non-convex. In fact, even verifying the feasibility of a solution is hard except for a few special cases. In our previous work [9], we tackled the problem by finding a safe tractable approximation of (3a) using the Bernstein approximation theorem. Specifically, the potentially non-convex constraint (3a) is replaced by a convex constraint $\mathcal{H}$, which has the property that every solution that is feasible for $\mathcal{H}$ is also feasible for (3a). A potential problem with the safe tractable approximation approach in [9] is that it requires the calculation of the cumulant generation function of $r_{k,n}^{(t)}$. Thus, the quality of the approximation depends crucially on the accuracy of our knowledge about the channel fading distribution. Of course, we may adopt simplified channel statistical models such as Rayleigh or Rician fading. However, this may lead to suboptimal or even infeasible resource allocation solutions when there is a mismatch between the actual channel distribution and the one assumed. In the next section, we will develop an alternative approach that does not require explicit information about the channel fading distribution. In particular, our approach is robust against misspecification of channel distribution.

Another challenge concerns the implementation of the solution algorithm at cellular base stations. Although the results in [9] showed that $\mathcal{P}_{\text{slow}}$ can be converted into a convex optimization problem, deploying a convex program solver at base stations could still be non-trivial[3]. With this in mind, we will present a "Biggest Bang for the Buck" algorithm for solving $\mathcal{P}_{\text{slow}}$ in Section IV. As we shall see, the BBB algorithm not only has a transparent engineering interpretation, but is also provably efficient.

III. Linear Programming Based Slow Adaptive OFDMA Scheme

To handle the chance constraints (3a), we adopt the *scenario approximation* approach, which has been extensively studied in the Operations Research community; see, e.g., [12]. In particular, the non-convex chance-constrained problem will be converted into a linear programming problem, which is much easier to solve.

To begin, suppose that we can draw J independent samples of the random vector $(r_{k,n})$, say $(\bar{r}_{k,n}^1), \ldots, (\bar{r}_{k,n}^J)$[4]. Then, we can replace the chance constraint (3a) by the following system of linear constraints:

$$\sum_{n=1}^{N} x_{k,n}\bar{r}_{k,n}^{j} \geq q_k, \quad \forall k, j.$$

In other words, consider the following linear programming problem, which serves as an approximation to the original chance-constrained problem $\mathcal{P}_{\text{slow}}$:

$$\begin{aligned} \hat{\mathcal{P}}_{\text{slow}}^J : \quad \max_{\{x_{k,n}\}} \quad & \sum_{k=1}^{K}\sum_{n=1}^{N} x_{k,n}\mathbb{E}\{r_{k,n}\} \\ \text{s.t.} \quad & \sum_{n=1}^{N} x_{k,n}\bar{r}_{k,n}^{j} \geq q_k, \quad \forall k, j, \quad \text{(4a)} \\ & \sum_{k=1}^{K} x_{k,n} \leq 1, \quad \forall n, \\ & x_{k,n} \geq 0, \quad \forall k, n. \end{aligned}$$

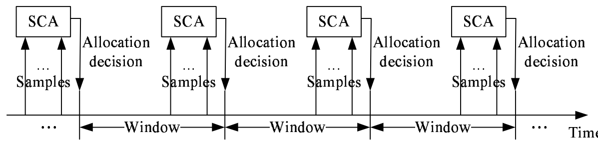

Fig. 1. Sampling procedure in the proposed slow adaptive OFDMA scheme.

Before we study the accuracy of the approximating constraints (4a), we would like to emphasize that in practical wireless systems, the samples $(\bar{r}_{k,n}^1), \ldots, (\bar{r}_{k,n}^J)$ can be obtained almost *at no extra costs*, as long as the channel distribution does not vary much across adjacent adaptation windows[5]. This is because most practical wireless systems constantly estimate the channel coefficients for coherent detection and adaptive modulation. Thus, it is not uncommon for the BS to know the CSI of the current time. In other words, the BS is already "sampling" $(r_{k,n})$ as part of the system operation. Assuming that the distribution of the channel does not vary rapidly, we can always make use of the latest J samples of $(r_{k,n})$ to perform SCA at the beginning of an adaptation window. This is illustrated in Fig. 1.

Now, let $\bar{\mathbf{x}}^J = (\bar{x}_{k,n}^J) \in \mathbb{R}_+^{NK}$ be an optimal solution to $\hat{\mathcal{P}}_{\text{slow}}^J$. Intuitively, if the number of samples J is reasonably large, then the J samples would constitute a set of significant measure. Thus, if $\bar{\mathbf{x}}^J$ satisfies the linear constraints (4a), it should also satisfy the chance constraint (3a) with high confidence.

It turns out that the above intuition can be made precise. For notational simplicity, let us denote by $V : \Delta \to [0,1]$ the left-hand side of the chance constraint (3a):

$$V(\mathbf{x}) = \Pr\left\{\sum_{n=1}^{N} x_{k,n}r_{k,n} \geq q_k \ \forall k\right\},$$

where

$$\Delta = \left\{\mathbf{x} \in \mathbb{R}_+^{NK} : \sum_{k=1}^{K} x_{k,n} \leq 1 \ \forall n\right\}.$$

Suppose for the moment that the optimal solution $\bar{\mathbf{x}}^J$ to $\hat{\mathcal{P}}_{\text{slow}}^J$ is unique. We are now interested in determining $V(\bar{\mathbf{x}}^J)$. By definition, if $V(\bar{\mathbf{x}}^J) \geq 1-\epsilon$, then $\bar{\mathbf{x}}^J$ satisfies the chance constraint (3a). However, note that $V(\bar{\mathbf{x}}^J)$ is a random variable,

[3]Note that most current base stations do not have the ability to solve convex optimization problems.

[4]Since the distribution of the channel remains unchanged for all t belonging to the same window, we may drop the superscript t in $r_{k,n}^{(t)}$.

[5]In general, the distribution of the channel stays nearly constant during a time span of seconds or tens of seconds, although the actual channel realizations fluctuate wildly at a much smaller timescale.

since $\bar{\mathbf{x}}^J$ depends on the J random samples $(\bar{r}^1_{k,n}), \ldots, (\bar{r}^J_{k,n})$. Thus, it is more appropriate to study the quantity $\Pr\{V(\bar{\mathbf{x}}^J) \geq 1-\epsilon\}$, where the probability is computed over all possible realizations of the J-tuple $((\bar{r}^1_{k,n}), \ldots, (\bar{r}^J_{k,n}))$. The following theorem establishes the soundness and performance of the proposed scheme.

Theorem 1. *Let $\bar{\mathbf{x}}^J$ be the unique solution to $\hat{\mathcal{P}}^J_{slow}$, and let $\epsilon \in (0,1)$ be the maximum tolerable system outage probability. Then, we have*

$$\Pr\{V(\bar{\mathbf{x}}^J) \geq 1-\epsilon\} \geq 1 - \sum_{i=0}^{NK-1} \binom{J}{i} \epsilon^i (1-\epsilon)^{J-i}, \quad (5)$$

where the probability is computed over all possible realizations of the J-tuple $((\bar{r}^1_{k,n}), \ldots, (\bar{r}^J_{k,n}))$. In particular, for any $\beta \in (0,1)$, if $J \geq J^*$, where

$$J^* \equiv \left\lceil \frac{1}{\epsilon}\left(NK - 1 + \ln\frac{1}{\beta} + \sqrt{2(NK-1)\ln\frac{1}{\beta} + \ln^2\frac{1}{\beta}} \right) \right\rceil, \quad (6)$$

then $\Pr\{V(\bar{\mathbf{x}}^J) \geq 1-\epsilon\} \geq 1-\beta$.

Proof: The bound in (5) follows from the result in [3]. To derive (6), we first recall the following well-known fact due to Chernoff [4] (cf. [1]):

Fact 1. *For any $\epsilon \in (0,1)$ and $d \leq \epsilon J$, we have*

$$\sum_{i=0}^{d} \binom{J}{i} \epsilon^i (1-\epsilon)^{J-i} \leq \exp\left[-\frac{(\epsilon J - d)^2}{2\epsilon J} \right].$$

Using Fact 1 and (5), we see that if

$$\exp\left[-\frac{(\epsilon J - NK + 1)^2}{2\epsilon J} \right] \leq \beta,$$

then $\Pr\{V(\bar{\mathbf{x}}^J) \geq 1-\epsilon\} \geq 1-\beta$. By taking logarithm on both sides and solving for J, we obtain (6). ■

As it turns out, Theorem 1 holds even when there is more than one solution to $\hat{\mathcal{P}}^J_{\text{slow}}$. We refer the interested reader to [3] for details.

Theorem 1 shows that if we use $J \geq J^*$ samples to form the linear program $\hat{\mathcal{P}}^J_{\text{slow}}$, then with probability at least $1-\beta$, the solution obtained will be feasible for $\mathcal{P}_{\text{slow}}$. In particular, by setting β to a very small value, say 10^{-10}, the solution to $\hat{\mathcal{P}}^J_{\text{slow}}$ will almost surely be feasible for $\mathcal{P}_{\text{slow}}$. Notably, the number of samples needed would not increase too much as β decreases, since J^* only has a logarithmic dependence on $1/\beta$.

The formulation described above provides a tractable way to handle the chance constraint (3a). Moreover, it has several advantages over the convex programming formulation in [9]:

1) We do not need to know the explicit channel distribution model in order to solve $\hat{\mathcal{P}}^J_{\text{slow}}$. All we need are samples of channel realizations, which are readily available as part of the operation of current wireless systems. In particular, our scheme is *more distributionally robust* and *much simpler to describe* than that in [9].
2) Our formulation shows that the optimization problems involved in both fast and slow adaptive OFDMA schemes *can be of the same class* (recall that $\mathcal{P}_{\text{fast}}$ is also a linear programming problem). In particular, our proposed slow adaptive OFDMA scheme $\hat{\mathcal{P}}^J_{\text{slow}}$ can be solved using standard linear program solvers, which already makes its implementation much simpler than that in [9].
3) Although both the fast and slow adaptive systems are based on solving linear programs, it should be noted that the former needs to solve an instance of $\mathcal{P}_{\text{fast}}$ every few milliseconds, while the latter only needs to solve one instance of $\hat{\mathcal{P}}^J_{\text{slow}}$ for each adaptation window. Thus, *the complexity gain of our scheme could be substantial*. To illustrate this, recall that the complexity of solving a general linear program with v variables and m inequality constraints is $\mathcal{O}((m+v)^{3/2}v^2)$ (see, e.g., [2, Section 6.6.1]). Suppose that a fast adaptive scheme needs to solve $\mathcal{P}_{\text{fast}}$ Λ times for each adaptation window. Since $\mathcal{P}_{\text{fast}}$ has NK variables and $N+K+NK$ inequality constraints, it follows that for each adaptation window, the complexity of the fast adaptive scheme is
$$\mathcal{C}_{\text{fast}} = \mathcal{O}(\Lambda(NK)^2(N+K+2NK)^{3/2}).$$
On the other hand, our slow adaptive scheme needs to solve only one $\hat{\mathcal{P}}^J_{\text{slow}}$ for each adaptation window. Since $\hat{\mathcal{P}}^J_{\text{slow}}$ has NK variables and $N+JK+NK$ inequality constraints, the complexity of our slow adaptive scheme is
$$\mathcal{C}_{\text{slow}} = \mathcal{O}((NK)^2(N+JK+2NK)^{3/2}).$$
In particular, for $J = J^*$, we see that if
$$\Lambda \geq \left(1 + \frac{(J-1)K}{N+K+2NK}\right)^{3/2} = \Omega((K/\epsilon)^{3/2}), \quad (7)$$
then $\mathcal{C}_{\text{slow}} \leq \mathcal{C}_{\text{fast}}$ [6]. The approximation in (7) is due to the fact that $\ln(1/\beta)$ is typically much smaller than NK even for β as small as 10^{-10}. In practice, K, the number of users that are grouped together for subcarrier allocation, is usually small (e.g., not exceeding 10). Hence, $\mathcal{C}_{\text{slow}}$ is in general much lower than $\mathcal{C}_{\text{fast}}$ for any practical systems.

In the next section, we will consider some practical issues concerning the implementation of the linear programming based slow adaptive scheme.

IV. "Biggest Bang for the Buck" Algorithm

Although the linear programs $\mathcal{P}_{\text{fast}}$ and $\hat{\mathcal{P}}^J_{\text{slow}}$ can be solved by standard interior-point algorithms, implementing such algorithms at base stations could be a challenging task. Thus, it is desirable to have easy-to-implement and intuitive algorithms to solve the linear programs arising from the fast and slow adaptive schemes. In this section, we will present one such algorithm and provide a theoretical analysis on its performance. The crucial observation underlying our approach is that both $\mathcal{P}_{\text{fast}}$ and $\hat{\mathcal{P}}^J_{\text{slow}}$ are instances of a so-called *mixed packing and covering linear program*, which means that they can be tackled

[6] Recall that $f(n) = \Omega(g(n))$ if there exist constants $c > 0$ and $n_0 \geq 1$ such that $f(n) \geq c \cdot g(n)$ for all $n \geq n_0$.

by the algorithmic framework developed in [15]. To illustrate this approach and fix ideas, let us focus on problem $\hat{\mathcal{P}}^J_{\text{slow}}$ and note that the treatment for problem $\mathcal{P}_{\text{fast}}$ is similar. Consider the following parametric formulation of $\hat{\mathcal{P}}^J_{\text{slow}}$:

$$\hat{\mathcal{P}}^J_{\text{slow}}(\tau): \ \text{find} \quad \mathbf{x} = (x_{k,n}) \in \mathbb{R}^{NK}$$

$$\text{s.t.} \quad \tilde{\gamma}_0(\mathbf{x}) \equiv \sum_{k=1}^{K}\sum_{n=1}^{N} x_{k,n}\mathbb{E}\{r_{k,n}\} \geq \tau, \tag{8a}$$

$$\tilde{\gamma}_{k,j}(\mathbf{x}) \equiv \sum_{n=1}^{N} x_{k,n}\bar{r}^j_{k,n} \geq q_k, \ \forall k,j, \tag{8b}$$

$$\tilde{\pi}_n(\mathbf{x}) \equiv \sum_{k=1}^{K} x_{k,n} \leq 1, \quad \forall n, \tag{8c}$$

$$x_{k,n} \geq 0, \quad \forall k,n, \tag{8d}$$

where $\tau \geq 0$ is a parameter. Observe that an optimal solution to $\hat{\mathcal{P}}^J_{\text{slow}}$ can be found simply by binary searching on τ and solving a sequence of linear feasibility problems $\{\hat{\mathcal{P}}^J_{\text{slow}}(\tau)\}$. Thus, it suffices to develop an efficient method to solve $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$ for any given $\tau \geq 0$. Towards that end, note that the choice of $\mathbf{x}$ is dictated by two opposing criteria. On one hand, we would like to increase the components of $\mathbf{x}$ so that the target system throughtput (8a) and data rates (8b) are achieved. On the other hand, such an increase is limited by the airtime allocation constraint (8c). The above observation suggests the following intuitive strategy for finding a solution to $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$: Starting with an arbitrary $\mathbf{x}$, at each step, identify the variable $x_{k,n}$ whose increment would make the most progress towards satisfying the throughput and data rate constraints ("the biggest bang for the buck"). More concretely, at each step, choose the index $(\bar{k},\bar{n})$ such that the ratio

$$\frac{\tilde{C}(\mathbf{x}+\delta\mathbf{e}_{\bar{k},\bar{n}}) - \tilde{C}(\mathbf{x})}{\tilde{P}(\mathbf{x}+\delta\mathbf{e}_{\bar{k},\bar{n}}) - \tilde{P}(\mathbf{x})} \tag{9}$$

is maximized, where the functions $\tilde{C}, \tilde{P} : \mathbb{R}^{NK}_+ \to \mathbb{R}_+$ are given by

$$\tilde{C}(\mathbf{x}) = \min\left\{\tilde{\gamma}_0(\mathbf{x}), \min_{k,j}\tilde{\gamma}_{k,j}(\mathbf{x})\right\}, \quad \tilde{P}(\mathbf{x}) = \max_n \tilde{\pi}_n(\mathbf{x}),$$

$\delta > 0$ is a step size, and $\mathbf{e}_{\bar{k},\bar{n}} \in \mathbb{R}^{NK}$ is the vector that has an 1 on the $(\bar{k},\bar{n})$-th entry and 0 elsewhere. Unfortunately, despite its intuitive appeal, the above strategy is not easy to analyze, as the piecewise linear nature of $\tilde{C}$ and $\tilde{P}$ makes it difficult to determine an appropriate step size δ in each step. However, not all is lost, as one can apply smoothing to $\tilde{C}$ and $\tilde{P}$, and the resulting smoothed version of the "the biggest bang for the buck" strategy can be analyzed using the techniques developed in [15]. In particular, we will show that the algorithm obtained by smoothing can find an almost feasible solution to $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$ in $\mathcal{O}(m\log m)$ iterations, where $m = N + JK + 1$ is the total number of constraints excluding the non-negativity constraints (8d), and each iteration can be implemented in $\mathcal{O}(m)$ time.

To begin, let us rewrite Problem (8) as follows:

$$\text{find} \quad \mathbf{x} = (x_{k,n}) \in \mathbb{R}^{NK}$$

$$\text{s.t.} \quad \gamma_0(\mathbf{x}) \equiv \sum_{k=1}^{K}\sum_{n=1}^{N} x_{k,n}\left(\frac{\Theta}{\tau}\mathbb{E}\{r_{k,n}\}\right) \geq \Theta, \tag{10a}$$

$$\gamma_{k,j}(\mathbf{x}) \equiv \sum_{n=1}^{N} x_{k,n}\left(\frac{\Theta}{q_k}\bar{r}^j_{k,n}\right) \geq \Theta, \quad \forall k,j, \tag{10b}$$

$$\pi_n(\mathbf{x}) \equiv \sum_{k=1}^{K} x_{k,n}\Theta \leq \Theta, \quad \forall n,$$

$$x_{k,n} \geq 0, \quad \forall k,n,$$

where $\Theta > 0$ is a parameter to be determined. Note that Problem (10) has identical right-hand sides, a fact that will facilitate our analysis later. Since we will only increase the components of $\mathbf{x}$ during the course of our algorithm, once a throughput or data rate constraint in (10) is satisfied, we can drop it from further consideration. To facilitate bookkeeping, we will use the index set $S \subset \bar{S} \equiv \{0\} \cup \{(k,j) : k = 1,\ldots,K;\ j = 1,\ldots,J\}$ to keep track of the unsatisfied throughput and data rate constraints.

Now, for each $S \subset \bar{S}$, define the functions $C_S, P : \mathbb{R}^{NK}_+ \to \mathbb{R}_+$ by

$$C_S(\mathbf{x}) = -\ln\left(\sum_{s\in S}\exp(-\gamma_s(\mathbf{x}))\right),$$

$$P(\mathbf{x}) = \ln\left(\sum_{n}\exp(\pi_n(\mathbf{x}))\right).$$

Note that C_S and P are obtained by applying the standard $\log\exp$ smoothing to the $\min$ and $\max$ functions. In particular, they satisfy

$$C_S(\mathbf{x}) \leq \min_{s\in S}\gamma_s(\mathbf{x}) \quad \text{for each } S \subset \bar{S}, \quad P(\mathbf{x}) \geq \max_n \pi_n(\mathbf{x}).$$

Moreover, the partial derivatives $\partial C_S(\mathbf{x})/\partial x_{k,n}$ and $\partial P(\mathbf{x})/\partial x_{k,n}$, which are essential in the development of our algorithm, admit the following explicit formulae:

$$\frac{\partial C_S(\mathbf{x})}{\partial x_{k,n}} = \frac{\sum_{s\in S}(\partial\gamma_s(\mathbf{x})/\partial x_{k,n})\exp(-\gamma_s(\mathbf{x}))}{\sum_{s\in S}\exp(-\gamma_s(\mathbf{x}))}, \tag{11}$$

$$\frac{\partial P(\mathbf{x})}{\partial x_{k,n}} = \frac{\Theta\exp(\pi_n(\mathbf{x}))}{\sum_n\exp(\pi_n(\mathbf{x}))}, \tag{12}$$

where

$$\frac{\partial\gamma_s(\mathbf{x})}{\partial x_{k,n}} = \begin{cases} \Theta\mathbb{E}\{r_{k,n}\}/\tau & \text{if } s = 0, \\ \Theta\bar{r}^j_{k,n}/q_k & \text{if } s = (k,j), \\ 0 & \text{otherwise.} \end{cases}$$

We are now ready to describe the BBB algorithm for finding an approximate solution to $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$ as formulated in (10); see Algorithm 1. The "biggest bang for the buck" strategy is implemented in line 7, where we increment the variable $x_{k,n}$ that maximizes the ratio $(\partial C_S(\mathbf{x})/\partial x_{k,n})/(\partial P(\mathbf{x})/\partial x_{k,n})$ (cf. (9)), subject to the condition that the rate at which P increases is at most $1+\eta$ times that at which C_S increases. Roughly speaking, this latter condition ensures that the progress towards satisfying the throughput and data rate constraints is not paid by an excessive allocation of airtime. It also ensures that the solution $\mathbf{x}$ output by the algorithm has the stated approximation guarantee.

Algorithm 1 The BBB Algorithm for Finding an Approximate Solution to $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$

Input: data defining $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$, accuracy $\eta \in (0,1]$

Output: a solution $\bar{\mathbf{x}} \geq \mathbf{0}$ that satisfies (10a), (10b) and $\sum_k \bar{x}_{k,n} \leq 1 + \mathcal{O}(\eta)$ for all n; or declare the problem infeasible

1: set $\mathbf{x} \longleftarrow \mathbf{0}$, $S \longleftarrow \bar{S}$ and $\Theta \longleftarrow (\ln m)/\eta$, where $m = N + JK + 1$ is the number of constraints

2: **while** $\min_{s \in S} \gamma_s(\mathbf{x}) < \Theta$ **do**

3: **if** $\min_{k,n} \left\{ (\partial P(\mathbf{x})/\partial x_{k,n}) / (\partial C_S(\mathbf{x})/\partial x_{k,n}) \right\} > 1$ **then**

4: return "infeasible"

5: **end if**

6: set $S \longleftarrow S \setminus \{s \in S : \gamma_s(\mathbf{x}) \geq \Theta\}$ // remove satisfied throughput and data rate constraints

7: let // "biggest bang for the buck", cf. (9)

$$(\bar{k}, \bar{n}) = \arg\max_{k,n} \left\{ \frac{\partial C_S(\mathbf{x})/\partial x_{k,n}}{\partial P(\mathbf{x})/\partial x_{k,n}} : \frac{\partial P(\mathbf{x})}{\partial x_{k,n}} \leq (1+\eta) \frac{\partial C_S(\mathbf{x})}{\partial x_{k,n}} \right\}$$

8: set $\delta > 0$ such that // determine the step size

$$\max \left\{ \max_{s \in S} \gamma_s(\delta \mathbf{e}_{\bar{k},\bar{n}}), \;\; \max_n \pi_n(\delta \mathbf{e}_{\bar{k},\bar{n}}) \right\} = \eta$$

9: set $\mathbf{x} \longleftarrow \mathbf{x} + \delta \mathbf{e}_{\bar{k},\bar{n}}$

10: **end while**

11: return $\mathbf{x}$

To analyze the performance of the BBB algorithm, we adopt the framework developed in [15]. First, let us show that line 3 of the algorithm is well defined:

Proposition 1. *If $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$ is feasible, then for all $\mathbf{x} \in \mathbb{R}^{NK}_+$, there exists (k', n') such that*

$$\frac{\partial P(\mathbf{x})/\partial x_{k',n'}}{\partial C_S(\mathbf{x})/\partial x_{k',n'}} \leq 1.$$

Proof: Let $\mathbf{x} \in \mathbb{R}^{NK}_+$ be arbitrary, and let $\bar{\mathbf{x}} \in \mathbb{R}^{NK}_+$ be feasible for $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$. Then, by (12) and the fact that

$$0 \leq \sum_k \bar{x}_{k,n} \Theta \leq \Theta, \quad \frac{\exp(\pi_n(\mathbf{x}))}{\sum_n \exp(\pi_n(\mathbf{x}))} \geq 0,$$

we have

$$\sum_{k,n} \bar{x}_{k,n} \frac{\partial P(\mathbf{x})}{\partial x_{k,n}} = \sum_n \left(\sum_k \bar{x}_{k,n} \Theta \right) \frac{\exp(\pi_n(\mathbf{x}))}{\sum_n \exp(\pi_n(\mathbf{x}))} \leq \Theta.$$

Similarly, using (11), it can be shown that

$$\sum_{k,n} \bar{x}_{k,n} \frac{\partial C_S(\mathbf{x})}{\partial x_{k,n}} \geq \Theta.$$

Hence, there exists (k', n') such that

$$\frac{\partial P(\mathbf{x})}{\partial x_{k',n'}} \leq \frac{\partial C_S(\mathbf{x})}{\partial x_{k',n'}},$$

as required. ∎

Next, we study the effect of incrementing $\mathbf{x}$ on the functions C_S and P. Towards that end, we need the following result, whose proof is standard and can be found, e.g., in [15, Lemma 1]:

Lemma 1. For any $\mathbf{v}, \zeta \in \mathbb{R}^l_+$ such that $0 \leq \zeta_i \leq \eta \leq 1$ for $i = 1, \ldots, l$, the following inequalities hold:

$$\begin{aligned} & \ln \left(\sum_{i=1}^l \exp(v_i + \zeta_i) \right) \\ \leq\; & \ln \left(\sum_{i=1}^l \exp(v_i) \right) + (1+\eta) \sum_{i=1}^l \zeta_i \frac{\exp(v_i)}{\sum_{j=1}^l \exp(v_j)}, \end{aligned}$$

$$\begin{aligned} & -\ln \left(\sum_{i=1}^l \exp(-(v_i + \zeta_i)) \right) \\ \geq\; & -\ln \left(\sum_{i=1}^l \exp(-v_i) \right) \\ & + \left(1 - \frac{\eta}{2}\right) \sum_{i=1}^l \zeta_i \frac{\exp(-v_i)}{\sum_{j=1}^l \exp(-v_j)}. \end{aligned}$$

Armed with Lemma 1, we can prove the following result:

Proposition 2. *Suppose that in an iteration of the BBB algorithm, we increment $\mathbf{x}$ to $\mathbf{x} + \delta \mathbf{e}_{\bar{k},\bar{n}}$. Then,*

$$P(\mathbf{x} + \delta \mathbf{e}_{\bar{k},\bar{n}}) - P(\mathbf{x}) \leq \frac{(1+\eta)^2}{1 - \eta/2} \left[C_S(\mathbf{x} + \delta \mathbf{e}_{\bar{k},\bar{n}}) - C_S(\mathbf{x}) \right].$$

Proof: By line 8 of the algorithm, we have $\max_n \pi_n(\delta \mathbf{e}_{\bar{k},\bar{n}}) \leq \eta$. Hence, by Lemma 1 and (12),

$$\begin{aligned} P(\mathbf{x} + \delta \mathbf{e}_{\bar{k},\bar{n}}) &= \ln \left(\sum_n \exp(\pi_n(\mathbf{x}) + \pi_n(\delta \mathbf{e}_{\bar{k},\bar{n}})) \right) \\ &\leq P(\mathbf{x}) + (1+\eta) \sum_n \pi_n(\delta \mathbf{e}_{\bar{k},\bar{n}}) \frac{\exp(\pi_n(\mathbf{x}))}{\sum_n \exp(\pi_n(\mathbf{x}))} \\ &= P(\mathbf{x}) + (1+\eta) \delta \frac{\partial P(\mathbf{x})}{\partial x_{\bar{k},\bar{n}}}. \end{aligned}$$

Similarly, using (11), it can be shown that

$$C_S(\mathbf{x} + \delta \mathbf{e}_{\bar{k},\bar{n}}) \geq C_S(\mathbf{x}) + \left(1 - \frac{\eta}{2}\right) \delta \frac{\partial C_S(\mathbf{x})}{\partial x_{\bar{k},\bar{n}}}.$$

Now, by line 7 of the algorithm,

$$\frac{\partial P(\mathbf{x})}{\partial x_{\bar{k},\bar{n}}} \leq (1+\eta) \frac{\partial C_S(\mathbf{x})}{\partial x_{\bar{k},\bar{n}}}.$$

It follows that

$$\begin{aligned} & P(\mathbf{x}+\delta\mathbf{e}_{\bar{k},\bar{n}}) - P(\mathbf{x}) \\ \leq\; & (1+\eta)^2\delta\frac{\partial C_S(\mathbf{x})}{\partial x_{\bar{k},\bar{n}}} \\ \leq\; & \frac{(1+\eta)^2}{1-\eta/2}\left[C_S(\mathbf{x}+\delta\mathbf{e}_{\bar{k},\bar{n}}) - C_S(\mathbf{x})\right], \end{aligned}$$

as required. ■

Finally, using Proposition 2, we can establish the correctness and runtime of the BBB algorithm.

Theorem 2. *Suppose that $\hat{\mathcal{P}}^J_{slow}(\tau)$ is feasible. Then, the BBB algorithm will return a solution $\bar{\mathbf{x}}$ with the stated properties in $\mathcal{O}(\eta^{-2}m\log m)$ iterations. Moreover, each iteration can be implemented in $\mathcal{O}(m)$ time.*

Proof: Let us first show that $\bar{\mathbf{x}}$ has the stated properties. Consider the function $\Phi : \mathbb{R}^{NK}_+ \times 2^{\bar{S}} \to \mathbb{R}$ given by

$$\Phi(\mathbf{x}, S) = P(\mathbf{x}) - \frac{(1+\eta)^2}{1-\eta/2}C_S(\mathbf{x}).$$

Initially, we have $\mathbf{x} = \mathbf{0}$ and $S = \bar{S}$. Hence, the initial value of Φ is given by

$$\Phi(\mathbf{0}, \bar{S}) = \ln N + \frac{(1+\eta)^2}{1-\eta/2}\ln(JK+1) \leq \eta\Theta + \frac{(1+\eta)^2}{1-\eta/2}\eta\Theta.$$

By Proposition 2, all increments in subsequent iterations do not increase the value of Φ. Moreover, since $C_S(\mathbf{x}) \geq C_{S'}(\mathbf{x})$ for all $S \subset S' \subset \bar{S}$ and $\mathbf{x} \in \mathbb{R}^{NK}_+$, removing the satisfied throughput and data rate constraints in line 6 will not increase the value of Φ either. Thus, we have $\Phi(\mathbf{x}, S) \leq \Phi(\mathbf{0}, \bar{S})$ for all $\mathbf{x} \in \mathbb{R}^{NK}_+$ and $S \subset \bar{S}$ that arise during the course of the algorithm. Now, just before the last increment, we have $C_S(\mathbf{x}) \leq \min_{s\in S}\gamma_s(\mathbf{x}) < \Theta$ by line 2. This implies that

$$\begin{aligned} \max_n \pi_n(\mathbf{x}) \;\leq\; P(\mathbf{x}) \;&\leq\; \Phi(\mathbf{0}, \bar{S}) + \frac{(1+\eta)^2}{1-\eta/2}C_S(\mathbf{x}) \\ &\leq\; \eta\Theta + \frac{(1+\eta)^3}{1-\eta/2}\Theta \;=\; (1+\mathcal{O}(\eta))\Theta. \end{aligned}$$

Moreover, by line 8, the last increment increases $\max_n \pi_n(\mathbf{x})$ by at most η. It follows that when the algorithm terminates, the solution $\bar{\mathbf{x}}$ satisfies (10a) and (10b) (because the condition in line 2 no longer holds), and $\sum_n \bar{x}_{k,n} \leq 1 + \mathcal{O}(\eta)$ for all n, as required.

Next, we bound the number of iterations of the algorithm. Define the function $\Psi : \mathbb{R}^{NK}_+ \times 2^{\bar{S}} \to \mathbb{R}$ by

$$\Psi(\mathbf{x}, S) = \sum_n \pi_n(\mathbf{x}) + \sum_{s\in S}(\gamma_s(\mathbf{x}) - \Theta - \eta).$$

Initially, we have $\Psi(\mathbf{0}, \bar{S}) \geq -(JK+1)(\Theta+\eta)$. By line 8 of the algorithm, each increment increases Ψ by at least η. Moreover, if S is the index set of the remaining throughput and data rate constraints before an increment, then for all $s \in S$, we have $\gamma_s(\mathbf{x}) < \Theta$ before the increment and $\gamma_s(\mathbf{x}) < \Theta + \eta$ after the increment. Thus, the constraint removal step in line 6 can only further increase the value of Ψ. Finally, when the algorithm terminates, we have $\Psi(\bar{\mathbf{x}}, S) \leq \Psi(\bar{\mathbf{x}}, \emptyset) \leq (1 + \mathcal{O}(\eta))N\Theta$. Hence, the number of iterations required is at most $(1+\mathcal{O}(\eta))(N+JK+1)\Theta/\eta$, which by definition of Θ is of order $\mathcal{O}(\eta^{-2}m\log m)$.

Finally, it is clear that each iteration can be implemented in $\mathcal{O}(m)$ time. This completes the proof of Theorem 2. ■

Although the BBB algorithm is designed to find an almost feasible solution to $\hat{\mathcal{P}}^J_{\text{slow}}(\tau)$ *for a fixed* $\tau \geq 0$, we can use it to find a good feasible solution to the original chance-constrained program $\mathcal{P}_{\text{slow}}$ (see (3)) by performing a binary search on $\tau \in [0, U]$, where $U = \sum_{k=1}^K\sum_{n=1}^N \mathbb{E}\{r_{k,n}\}$. Specifically, for any given $\alpha > 0$, we call the BBB algorithm $\mathcal{O}(\log(U/\alpha))$ times to obtain an $\bar{\mathbf{x}} \in \mathbb{R}^{NK}_+$ that satisfies

$$\sum_{k=1}^K\sum_{n=1}^N \bar{x}_{k,n}\mathbb{E}\{r_{k,n}\} \geq \mathsf{OPT} - \alpha,$$

$$\sum_{n=1}^N \bar{x}_{k,n}\bar{r}^j_{k,n} \geq (1+\mathcal{O}(\eta))q_k, \quad \forall k, j,$$

$$\sum_{k=1}^K \bar{x}_{k,n} \leq 1 + \mathcal{O}(\eta), \quad \forall n,$$

where OPT is the optimal value of $\hat{\mathcal{P}}^J_{\text{slow}}$ with $\{q_k\}$ replaced by $\{(1+\mathcal{O}(\eta))q_k\}$. Now, by scaling down each component of $\bar{\mathbf{x}}$ by a factor of $1+\mathcal{O}(\eta)$, we obtain a *feasible* airtime allocation vector $\bar{\mathbf{x}}' \in \mathbb{R}^{NK}_+$ that is *approximately optimal*, i.e.,

$$\sum_{k=1}^K\sum_{n=1}^N \bar{x}'_{k,n}\mathbb{E}\{r_{k,n}\} \geq \frac{\mathsf{OPT} - \alpha}{1+\mathcal{O}(\eta)},$$

$$\sum_{n=1}^N \bar{x}'_{k,n}\bar{r}^j_{k,n} \geq q_k, \quad \forall k, j,$$

$$\sum_{k=1}^K \bar{x}'_{k,n} \leq 1, \quad \forall n$$

(recall that $\alpha, \eta > 0$ are parameters that can be chosen to be arbitrarily small). Moreover, by Theorem 1, the vector $\bar{\mathbf{x}}'$ will be feasible for the original chance-constrained program $\mathcal{P}_{\text{slow}}$ with high probability.

Finally, in terms of the dependence on N (number of subcarriers), K (number of users) and ϵ (maximum system tolerable outage probability), the complexity of the BBB algorithm scales like $(N^2K^4/\epsilon^2)\log(NK/\epsilon)$ for each fixed $\alpha, \eta > 0$ when $J = J^*$ (recall that J^* is of order NK/ϵ; see (6)), which is competitive against the $N^{7/2}K^5/\epsilon^{3/2}$ scaling of general interior-point algorithms. Thus, the BBB algorithm can be used to develop an alternative, provably efficient yet intuitive method for implementing our proposed slow adaptive OFDMA scheme.

V. Performance Evaluation

In this section, we investigate the performance of the proposed linear programming based slow adaptive OFDMA schemes through numerical simulations. Throughout this section, we assume that there are 4 users and 256 subcarriers. The users are assumed to be uniformly distributed in a cell of radius $R = 50$m. Each user k has a minimum requirement on its short-term data rate $q_k = 64$ bits per OFDM symbol, and the tolerable system outage probability ϵ is 0.1. Furthermore, suppose that β is set to be 0.01.

Suppose that the path-loss exponent is equal to 4, and the small-scale channel fading varies once every 1ms. Meanwhile, the length of the adaptation window for slow adaptation is chosen to be $T = 1$s, implying that SCA is adapted 1000 times less frequently in slow adaptive OFDMA than in fast adaptive OFDMA. We further suppose that the control signaling overhead consumes 10% of communication bandwidth every time SCA is updated. Note that with our system setting, the control signaling for slow adaptation occupies 1000 times less spectrum resource than that for fast adaptation.

A. Spectrum Efficiency and Comparison with Fast Adaptation

In Fig. 2 and Fig. 3, the wireless channel is assumed to be Rayleigh faded. In Fig. 2, we compare the spectral efficiency of slow-adaptive OFDMA with that of fast adaptive OFDMA over 100 independent adaptation windows, where spectral efficiency is defined as the data rate per user per OFDM symbol. Suppose that the transmission power of the BS on each subcarrier is such that the average received signal-to-noise ratio (SNR) at the boundary of the cell is 6dB. It can be seen that although slow adaptive OFDMA updates subcarrier allocation 1000 times less frequently than fast adaptive OFDMA, it can achieve on average 91% of the spectral efficiency.

In Fig. 3, the average spectrum efficiencies for slow and fast adaptation schemes are plotted against the average SNR at the cell edge. The figures shows that slow adaptive OFDMA achieves around 72% to 95% of the spectral efficiency of fast adaptive OFDMA when the average SNR at the cell edge ranges between 0 to 10dB. Considering the substantially lower computational complexity and signaling overhead, slow adaptive OFDMA holds significant promise for deployment in real-world systems.

For comparison, we also plot in Fig. 3 the performance of the slow adaptive OFDMA system proposed in [9]. Note that the probabilistic data rate constraints in [9] are slightly different from those in this paper. Therein, the constraints are imposed on individual users separately, i.e.,

$$\Pr\left\{\sum_{n=1}^{N} x_{k,n} r_{k,n}^{(t)} \geq q_k\right\} \geq 1 - \epsilon_k, \quad \forall k, t, \qquad (13)$$

while in (2), the outage probability is defined jointly on all users. For fair comparison, we set $\epsilon_k = 0.1/K = 0.025$ in (13), so that the joint system outage probability is close to 0.1 by the union bound. It can be seen that the scheme proposed in this paper outperforms that in [9] by a big margin. This is because the quality of the convex approximation in [9] depends crucially on the shape of the original feasible set and can be quite conservative at times.

In Fig. 4 and Fig. 5, we repeat the simulations by assuming the channel fading is Rician distributed. In Fig. 4, the K-factor, defined as the ratio between the signal power in the line-of-sight (LOS) path and the scattered path, is set to be

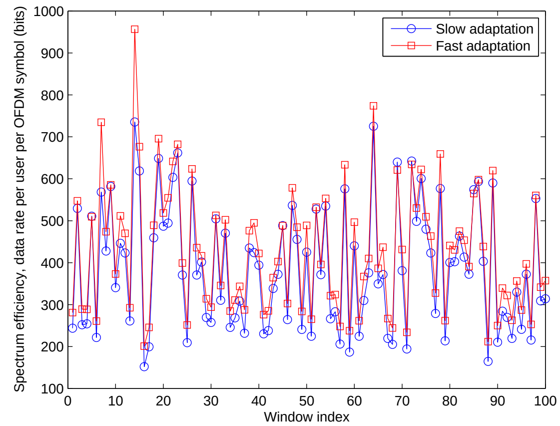

Fig. 2. Comparison of spectrum efficiencies of slow adaptive OFDMA and fast adaptive OFDMA under Rayleigh fading channel.

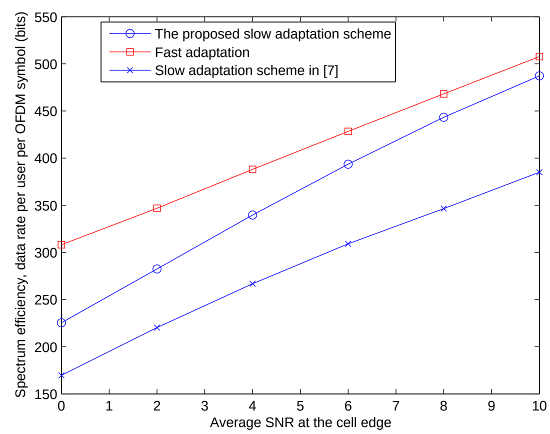

Fig. 3. Comparison of spectrum efficiencies of slow adaptive OFDMA and fast adaptive OFDMA under Rayleigh fading channel.

0.5. It can be seen that the gap between the throughput of slow and fast adaptation systems is slightly narrowed to 26% and 4% when the average SNR at the cell edge is 0 and 10dB, respectively. This is due to the existence of deterministic line-of-sight component in Rician fading, which reduces the need for fast adaptation. As we increase K to 1 in Fig. 5, the gap is further narrowed to 24% and 3% when the cell-edge SNR is 0 and 10dB, respectively.

In Fig. 6, we investigate the system performance when the channel fading is Nakagami distributed with shape parameter $m = 2$. With Nakagami fading, the amplitude of received signal is a result of m-branch maximum ratio combining of Rayleigh-faded signals. From the figure, it can be seen that the gap between the spectrum efficiency of the slow and fast adaptation is much narrower than that in the Rayleigh fading channel. In particular, slow adaptation achieves 82% throughput of the fast adaptation when the cell-edge SNR is 0dB, and 99.6% when the cell-edge SNR is 8dB. This is because with m-branch maximum ratio combining, the fluctuation of the channel gain is reduced compared with Rayleigh fading, thus eliminating the advantage of fast adaptation. Noticeably,

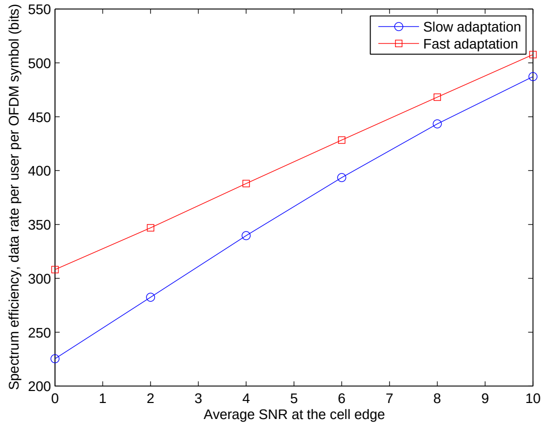

Fig. 4. Comparison of spectrum efficiencies of slow adaptive OFDMA and fast adaptive OFDMA under Rician fading channel with $K = 0.5$.

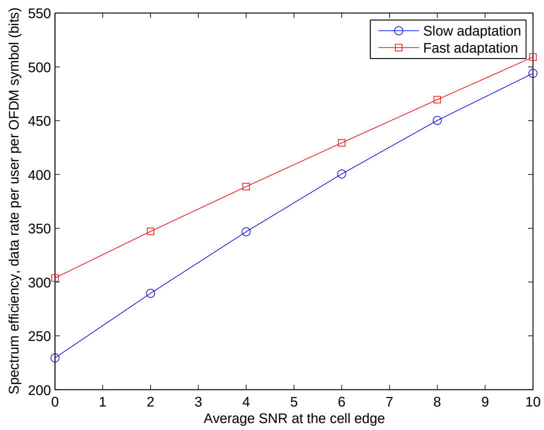

Fig. 5. Comparison of spectrum efficiencies of slow adaptive OFDMA and fast adaptive OFDMA under Rician fading channel with $K = 1$.

slow adaptive OFDMA even starts to outperform fast adaptive OFDMA when the cell-edge SNR is 10dB. This is due to the significant reduction in control signaling overhead in the slow adaptation scheme.

Before leaving this subsection, we would like to note that the reduced computational complexity due to slow adaptation also leads to a reduction in the energy consumption of hardware processing units. Thus, the overall energy efficiency may turn out to be comparable to or even higher than that of fast adaptive OFDMA.

B. Outage Probability and the Potential of Reducing the Sample Size

The proposed slow adaptive OFDMA scheme ensures that the short-term data rate of each user is satisfied with a high probability $1 - \epsilon$, even though SCA is performed on a much larger time scale. Fig. 7 investigates the system outage probabilities of the slow adaptation schemes, where an outage is said to have occurred as long as there is a user whose short-term data rate falls below the requirement q_k. It can be seen

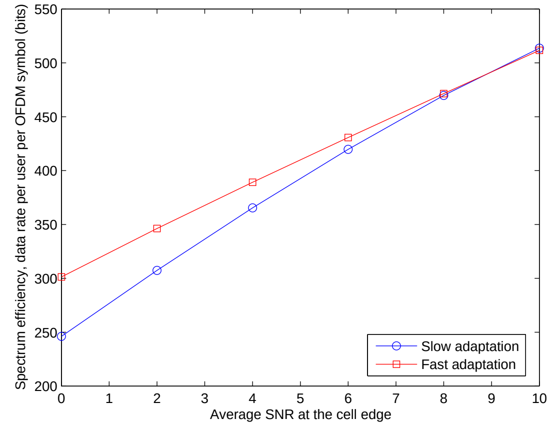

Fig. 6. Comparison of spectrum efficiencies of slow adaptive OFDMA and fast adaptive OFDMA under Nakagami fading channel with $m = 2$.

that the slow adaptive OFDMA scheme leads to SCA solutions that satisfies the outage probability requirement in different channel fading scenarios. The approach, however, seems to be conservative in general, as the outage probability is about 60% of ϵ.

The conservatism observed in Fig. 7 suggests that we may be able to reduce the number of samples J while still fulfilling the outage probability requirement. After all, the a priori bound J^* given in (6) is derived regardless of the probability distribution of the channel[7]. In practice, the wireless fading distributions may possess certain features, such as having light tails, that could reduce the number of samples required to construct a significant measure. In Fig. 8, we plot the outage probability as a function of the number of samples J under Rayleigh fading channel when the cell-edge SNR is 10dB. The figure shows that we can safely reduce the number of samples J to $5J^*/8$ without violating the outage probability requirement. Note that J is directly related to the size of the linear programming problem $\hat{\mathcal{P}}^J_{\text{slow}}$. Thus, we have a strong incentive to reduce this number. It would be an interesting future research topic to derive a tighter bound on J^* by exploiting the specific features of wireless fading distributions.

VI. Conclusion

In this paper, we proposed an efficient and distributionally robust slow adaptive OFDMA scheme that can significantly enhance the practicality of adaptive OFDMA. By exploring a chance-constrained formulation, the proposed scheme can guarantee users' short-term data rate requirements while maximizing long-term system throughput. Through scenario approximation methods, the potentially hard chance-constrained programming problem is converted into a linear program, which can be efficiently solved using off-the-shelf linear program solvers. Moreover, the proposed scheme is distributionally robust, namely, it does not require the precise knowledge

[7] We must emphasize that the dependence on ϵ and β of the bound J^* given in (6) is tight in the worst case. Indeed, it can be proven that in general there does not exist a sample complexity bound that scales better than $\mathcal{O}(\epsilon^{-1} \ln \beta^{-1})$.

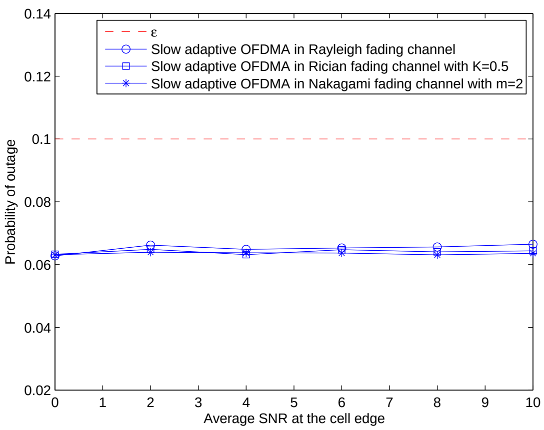

Fig. 7. Probability of outage of slow adaptive OFDMA with $\epsilon = 0.1$.

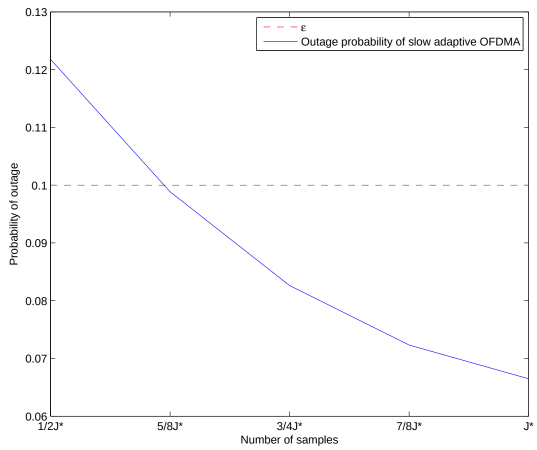

Fig. 8. Probability of outage against the number of samples when $\epsilon = 0.1$.

of channel fading distribution. For practical implementation, we presented a "Biggest Bang for the Buck" (BBB) algorithm, which is provably efficient and has a transparent engineering interpretation.

Our simulation results showed that the proposed slow adaptive OFDMA scheme could achieve a significant portion of the throughput of the fast adaptive OFDMA scheme with much lower computational complexity. Interestingly, in some scenarios, e.g., Nakagami fading with $m = 2$, slow adaptive OFDMA may achieve even higher throughput than its fast adaptation counterpart, thanks to the drastically lower control signaling overhead. Its high performance and low implementation cost make slow adaptive OFDMA a strong competitor of the popular fast adaptive OFDMA scheme.

REFERENCES

[1] D. Angluin and L. G. Valiant, "Fast Probabilistic Algorithms for Hamiltonian Circuits and Matchings," *Journal of Computer and System Sciences*, vol. 18, no. 2, pp. 155–193, 1979.

[2] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*, ser. MPS–SIAM Series on Optimization. Philadelphia, Pennsylvania: Society for Industrial and Applied Mathematics, 2001, vol. 2.

[3] M. C. Campi and S. Garatti, "The Exact Feasibility of Randomized Solutions of Uncertain Convex Programs," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1211–1230, 2008.

[4] H. Chernoff, "A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the Sum of Observations," *The Annals of Mathematical Statistics*, vol. 23, no. 4, pp. 493–507, 1952.

[5] A. Conti, M. Z. Win, and M. Chiani, "Slow Adaptive M–QAM with Diversity in Fast Fading and Shadowing," *IEEE Transactions on Communications*, vol. 55, no. 5, pp. 895–905, 2007.

[6] J. Karedal, A. J. Johansson, F. Tufvesson, and A. F. Molisch, "A Measurement–Based Fading Model for Wireless Personal Area Networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4575–4585, 2008.

[7] W. Kumwilaisak, C.-C. J. Kuo, and D. Wu, "Fading Channel Modeling via Variable–Length Markov Chain Technique," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1338–1358, 2008.

[8] W. L. Li, Y. J. Zhang, and M. Z. Win, "Slow Adaptive OFDMA via Stochastic Programming," in *IEEE International Conference on Communications, 2009 (ICC '09)*, 2009.

[9] W. W.-L. Li, Y. J. Zhang, A. M.-C. So, and M. Z. Win, "Slow Adaptive OFDMA Systems through Chance Constrained Programming," *IEEE Transactions on Signal Processing*, vol. 58, no. 7, pp. 3858–3869, 2010.

[10] Y. Li and S. Kishore, "Slow Adaptive M–QAM under Third–Party Received Signal Constraints in Shadowing Environments," *Research Letters in Communications*, vol. 2008, Article ID 121383, 2008.

[11] A. Nemirovski and A. Shapiro, "Convex Approximations of Chance Constrained Programs," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 969–996, 2007.

[12] ——, "Scenario Approximations of Chance Constraints," in *Probabilistic and Randomized Methods for Design under Uncertainty*, G. Calafiore and F. Dabbene, Eds. London: Springer–Verlag, 2006, pp. 3–47.

[13] T. Q. S. Quek, H. Shin, and M. Z. Win, "Robust Wireless Relay Networks: Slow Power Allocation with Guaranteed QoS," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 700–713, 2007.

[14] C. Y. Wong, R. S. Cheng, K. B. Letaief, and R. D. Murch, "Multiuser OFDM with Adaptive Subcarrier, Bit, and Power Allocation," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 10, pp. 1747–1758, 1999.

[15] N. E. Young, "Sequential and Parallel Algorithms for Mixed Packing and Covering," in *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science (FOCS 2001)*, 2001, pp. 538–546.

[16] Y. J. Zhang and K. B. Letaief, "Multiuser Adaptive Subcarrier–and–Bit Allocation with Adaptive Cell Selection for OFDM Systems," *IEEE Transactions on Wireless Communications*, vol. 3, no. 5, pp. 1566–1575, 2004.

Anthony Man-Cho So (M'12) received his BSE degree in Computer Science from Princeton University in 2000 with minors in Applied and Computational Mathematics, Engineering and Management Systems, and German Language and Culture. He then received his MSc degree in Computer Science in 2002, and his PhD degree in Computer Science with a PhD minor in Mathematics in 2007, all from Stanford University.

Dr. So joined The Chinese University of Hong Kong in 2007, where he is currently an Assistant Professor in the Department of Systems Engineering and Engineering Management, and, by courtesy, the Department of Computer Science and Engineering and the CUHK-BGI Innovation Institute of Trans-omics. His current research focuses on the interplay between optimization theory and various areas of algorithm design, such as computational geometry, signal processing, bioinformatics, stochastic optimization, combinatorial optimization, and algorithmic game theory.

Dr. So currently serves on the editorial boards of IEEE TRANSACTIONS ON SIGNAL PROCESSING, Journal of Global Optimization, Mathematics of Operations Research, and Optimization Methods and Software. He is a recipient of the 2008 Exemplary Teaching Award and 2011 Dean's Exemplary Teaching Award given by the Faculty of Engineering at The Chinese University of Hong Kong, the 2010 Optimization Prize for Young Researchers given by the Optimization Society of the Institute for Operations Research and the Management Sciences (INFORMS), and the Young Researcher Award 2010 given by The Chinese University of Hong Kong.

Ying Jun (Angela) Zhang (S'00-M'05-SM'11) received her PhD degree in Electrical and Electronic Engineering from the Hong Kong University of Science and Technology, Hong Kong in 2004. She received a BEng in Electronic Engineering from Fudan University, Shanghai, China in 2000.

Since 2005, she has been with Department of Information Engineering, The Chinese University of Hong Kong, where she is currently an Associate Professor. She was with Wireless Communications and Network Science Laboratory at Massachusetts Institute of Technology (MIT) during the summers of 2007 and 2009. Her current research topics include resource allocation, convex and non-convex optimization for wireless systems, stochastic optimization, cognitive networks, MIMO systems, etc.

Dr. Zhang is on the editorial Boards of IEEE TRANSACTIONS ON COMMUNICATIONS, and Wiley Security and Communications Networks Journal. She was an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and a Guest Editor of a Feature Topic in IEEE COMMUNICATIONS MAGAZINE. She has served as a Workshop Chair of IEEE ICCC 2013, TPC Vice-Chair of Wireless Communications Track of IEEE CCNC 2013, TPC Co-Chair of Wireless Communications Symposium of IEEE GLOBECOM 2012 Publication Chair of IEEE TTM 2011, TPC Co-Chair of Communication Theory Symposium of IEEE ICC 2009, Track Chair of ICCCN 2007, and Publicity Chair of IEEE MASS 2007. She was a Co-Chair of IEEE ComSoc Multimedia Communications Technical Committee, an IEEE Technical Activity Board GOLD Representative, 2008 IEEE GOLD Technical Conference Program Leader, IEEE Communication Society GOLD Coordinator, and a Member of IEEE Communication Society Member Relations Council (MRC).

Dr. Zhang received the 2011 Young Researcher Award of The Chinese University of Hong Kong. She is a co-recipient of 2011 IEEE Marconi Prize Paper Award on Wireless Communications, the Annual Best Paper Award of IEEE Transactions on Wireless Communications. As the only winner from Engineering Science, she has won the Hong Kong Young Scientist Award 2006, conferred by the Hong Kong Institution of Science.