OPPORTUNISTIC MULTIUSER SCHEDULING IN DOWNLINK TDMA MISO SYSTEMS

Chuxiang Li and Xiaodong Wang

Department of Electrical Engineering
Columbia University
New York, NY 10027, U.S.A.

ABSTRACT

We consider the problem of multiuser opportunistic fair scheduling (OFS) in downlink MISO systems employing beamforming. OFS is performed on a per scheduling interval basis to achieve fair bandwidth allocation. Transmit beamforming provides TDMA systems with the capability of supporting multiple concurrent transmissions. Since the optimal beamforming scheme can be calculated for a given subset of users, the scheduling problem then refers to the optimal user subset selection to maximize the system throughput subject to certain constraints. We propose two practical multiuser schedulers, and then present discrete stochastic approximation algorithms to adaptively select a better user subset. We also consider scenarios of time-varying channels where the algorithm can track the time-varying optimum. We present results to show the performance of the proposed algorithms in terms of the fast convergence, the time-varying tracking capability and the fairness.

1. INTRODUCTION

Opportunistic fair scheduling (OFS) is an important technique in wireless networks, and it aims at balancing two conflicting goals, fairness and resource utilization [1, 2]. Several approaches have been proposed in current literature. Based on the utility-based single-user scheduler [3], a general methodology and an multiuser scheduler are proposed in [2] for CDMA systems. Note that, instead of incorporating the physical-layer constraints and implementation details, perfect channel knowledge and simplified models for physical-layer are assumed in existing works [2, 3].

Also note that only single-user scheduler has been considered for TDMA systems in current literature [3, 4], which was motivated from an information-theoretic result on the single transmit antenna model [4]. It has been pointed out in [4] that multiple transmit antennas provide the potential for multiuser scheduling in downlink MISO broadcasting systems, which is an interesting and open problem. In [4], a multiuser scheduler is also discussed from the information-theoretic point of view. In this paper, we consider the multiuser scheduling problem for downlink MISO systems employing transmit beamforming from the viewpoint of practical systems. Specifically, the downlink multiuser beamforming is treated as the physical-layer implementation, which has been extensively studied in [5, 6]. We can also draw analogy between the multiuser schedulers treated in this paper and that in [2] for CDMA systems. As pointed out in [6], many results found in synchronous CDMA systems can be transformed to the scenario of the downlink multiuser beamforming. Then the multiuser OFS with beamforming treated in this paper is analogous to that for CDMA systems.

Since the optimal beamforming scheme can be calculated for a given user subset, the multiuser OFS problem then refers to the optimal subset selection at each interval to maximize the system throughput under certain constraints. Straightforward implementation of the selection suffers from several practical problems [7], e.g., the high complexity, the availability of perfect CSI, and the time-varying optimum tracking. In this paper, we propose discrete stochastic approximation (DSA) algorithms to achieve the subset selection, based on the stochastic optimization techniques in recent literature [8, 9]. The rational behind these algorithms is exactly the same as that behind traditional adaptive filtering algorithms, where computation is distributed over time. Specifically, we only make simple update at each time, and the performance gets improved as the algorithm iterates. Note that the solutions here take discrete values. The algorithm is computational efficient due to its self-learning property, i.e., it spends more time at the optimum than at any others. Moreover, the algorithm can adopt a fixed step-size which acts as a forgetting factor, so that it can track the time-varying optimum. The motivation here is again the same as that in the continuous adaptive filtering algorithms in non-static environments, where slow varying dynamics can be tracked.

The remainder of this paper is organized as follows. In Section 2, the multiuser OFS framework with transmit beamforming is described, and the corresponding discrete stochastic optimization problem is formulated. Section 3 presents two practical schedulers, and then proposes the DSA algorithms. Simulation results are given in Section 4; and Section 5 contains the conclusions.

2. SYSTEM DESCRIPTIONS

2.1. Multiuser Downlink Beamforming

In this paper, the downlink MISO system employing transmit beamforming is treated as the physical-layer implementation. Suppose that there are K active users and T transmit antennas at the base station. Then the received SINR at mobile k is given by [5, 6] $\gamma_k = \frac{P_k \boldsymbol{u}_k^H \boldsymbol{Q}_k \boldsymbol{u}_k}{\sum_{j\neq k} P_j \boldsymbol{u}_j^H \boldsymbol{Q}_k \boldsymbol{u}_j + \eta}$, where P_k, $\boldsymbol{u}_k$ and $\boldsymbol{Q}_k \triangleq \mathbb{E}\{\boldsymbol{h}_k^H \boldsymbol{h}_k\}$ is the transmit power, the beamformer and the channel covarince matrix of user k, respectively; η is the AWGN power level. Define $\boldsymbol{U} \triangleq [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_K]$ and $\boldsymbol{p} \triangleq [P_1, \cdots, P_K]$. Denote ρ as the total transmit power constraint, and $\{\gamma_k^{\min}\}_k$ as the minimum SINR requests. Then the SINR balancing problem is formulated as [5, 6]:

$$\max_{\boldsymbol{U},\boldsymbol{p}} \min_{1\leq k\leq K} \frac{\gamma_k(\boldsymbol{U},\boldsymbol{p})}{\gamma_k^{\min}}, \quad \text{subject to} \quad \sum_{k=1}^{K} P_k \leq \rho. \tag{1}$$

The optimal solution for (1) has been well studied in [5, 6].

Remark: In this paper, the throughput evaluation depends on the physical-layer implementation, i.e., the downlink multiuser beamforming scheme [5, 6] which is the state-of-the-art for this scenario. However, our proposed multiuser scheduling framework does NOT depend on any particular physical-layer scheme.

2.2. Multiuser Opportunistic Fair Scheduling Framework

Fig. 1 shows the general OFS framework [2], where the scheduler chooses a user subset to maximize the weighted system throughput; the controller guarantees the fairness by adjusting the weights.

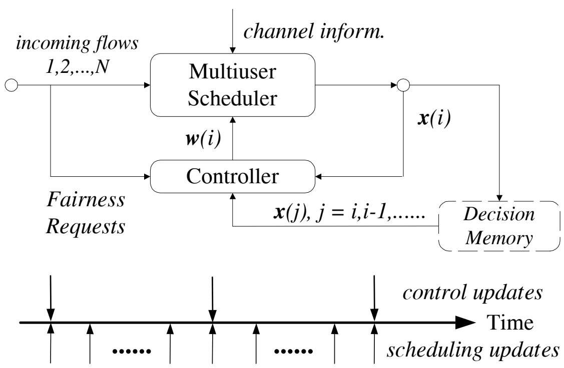

Fig. 1. Generalized architecture of multiuser OFS.

Suppose that there are totally N users in the system. At each scheduling interval i, the inputs to the scheduler are the data flows, the channel states, and the weights $\boldsymbol{w}(i) = [w_1(i), \cdots, w_N(i)]^T$. Denote θ as a user subset, $|\theta|$ as the number of users in θ, Θ be the set of all possible subsets, $\boldsymbol{Q}_\theta(i)$ as the channel set of the users in θ, and $\boldsymbol{x}(i) = [X_1(i), \cdots, X_N(i)]^T \geq 0$ as the rates of the users. Define the objective function $\Phi(\boldsymbol{Q}_\theta(i)) \triangleq \sum_{n\in\theta} w_n(i)X_n(i)$ as the weighted system throughput for θ. Then the user subset selection is formulated as the following discrete optimization problem

$$\theta^*(i) = \arg\max_{\theta\in\Theta} \ \Phi(\boldsymbol{Q}_\theta(i)) = \max_{\theta\in\Theta} \sum_{n\in\theta} w_n(i)X_n(i), \qquad (2)$$

where $\theta^*(i)$ denotes the optimal user subset for the interval i.

For the controller, the inputs at each control interval are the users' throughput priorities $\overrightarrow{\phi} = [\phi_1, \cdots, \phi_N]^T$ and $\boldsymbol{x}(i)$. The deterministic fairness constraint is given by [2] $\frac{\phi_1}{\mathbb{E}\{X_1(i)\}} = \cdots = \frac{\phi_N}{\mathbb{E}\{X_N(i)\}}$. At each control interval, the controller updates $\boldsymbol{w}(i)$ using $\overrightarrow{\phi}$ and $\boldsymbol{x}(i)$ [2, 3]. Specifically, define $\boldsymbol{y}(i, \boldsymbol{w}(i)) \triangleq \frac{\overrightarrow{\phi}}{\sum_{j=1}^N \phi_j} - \frac{\boldsymbol{x}(i,\boldsymbol{w}(i))}{\sum_{j=1}^N X_j(i,\boldsymbol{w}(i))}$, where $\boldsymbol{x}(i, \boldsymbol{w}(i))$ is the decision $\boldsymbol{x}(i)$ for given $\boldsymbol{w}(i)$. Let $\nu(i) = 1/i$ be the step-size. To guarantee the fairness constraint, $\boldsymbol{w}(i)$ should then be updated as [2]

$$\boldsymbol{w}(i + 1) = \boldsymbol{w}(i) + \nu(i)\boldsymbol{y}(i, \boldsymbol{w}(i)). \qquad (3)$$

Remark: The scheduling interval is much smaller than the control interval, i.e., there are many scheduling updates [given by (2)] between each two consecutive control updates [(3)]. The scheduling process is an iterative one (cf. Sec. 3), during which $\boldsymbol{w}(i)$ is fixed so that the algorithm can converge to the optimum.

2.3. Discrete Stochastic Optimization Formulation

Assume that $\boldsymbol{Q}_n(i)$ keeps invariant during each scheduling interval. For the scenario of fixed channels, $\boldsymbol{Q}_n(i)$ is fixed for different scheduling intervals; for the scenario of time-varying channels, the AR model will be used later to describe the dynamic of $\boldsymbol{Q}_n(i)$. Since $\boldsymbol{Q}_\theta(i)$ and $\boldsymbol{w}(i)$ remain fixed within each interval i, hereafter we drop the index i. Note that in practice, $\boldsymbol{Q}_\theta$ is estimated and therefore noisy. Denote $\{\hat{\boldsymbol{Q}}_\theta(m),\ m = 1, 2, \cdots\}$ as a sequence of the estimates for $\boldsymbol{Q}_\theta$. For each $\hat{\boldsymbol{Q}}_\theta(m)$, we can compute the optimal $\boldsymbol{U}$ and $\boldsymbol{p}$ as discussed in Section 2.1, and the corresponding noisy estimate of $\Phi(\boldsymbol{Q}_\theta)$ at the m-th iteration, denoted as $\phi(m, \theta)$. Then, we obtain the sequence $\{\phi(m,\theta),\ m = 1, 2, \cdots\}$ for the fixed $\boldsymbol{w}$. If each $\phi(m, \theta)$ is an unbiased estimate of $\Phi(\boldsymbol{Q}_\theta)$, the discrete optimization problem (2) can then be reformulated as the following discrete stochastic optimization problem

$$\theta^* = \arg\max_{\theta\in\Theta} \Phi(\boldsymbol{Q}_\theta) = \arg\max_{\theta\in\Theta} \mathbb{E}\{\phi(m, \theta)\}. \qquad (4)$$

3. ADAPTIVE MULTIUSER SCHEDULING PROCESS

3.1. Practical Multiuser Schedulers

Given a particular user subset θ and $\boldsymbol{Q}_\theta$, the optimal beamforming scheme can be calculated [5, 6]. Denote $\alpha(\theta)$ as the corresponding SINR ratio. Then the achievable rate for user $n \in \theta$ is given by $X_n(\theta) = \log(1 + \gamma_n^{\min}\alpha(\theta))$. We next define two practical multiuser schedulers, i.e., two forms of the objective function $\Phi(\boldsymbol{Q}_\theta)$.

Scheduler I: fixed-size user subset without rate requirement

Scheduler I treats the candidate θ with fixed size K and without SINR threshold requirement. Then the size of the whole solution space is $|\Theta| = \binom{N}{K}$, and the $\Phi(\boldsymbol{Q}_\theta)$ is defined as

$$\Phi(\boldsymbol{Q}_\theta) \triangleq \sum_{n\in\theta} w_n \log(1 + \gamma_n^{\min}\alpha(\theta)), \ \ \theta \in \Theta. \qquad (5)$$

Scheduler I is defined from the practical concerns as follows. Note that if certain SINR thresholds are required, then only the subsets satisfying $\alpha(\theta) \geq 1$ can be viewed as the possible candidate in Θ; if no SINR threshold exists, any user subset is the possible candidate. Scheduler I is suitable for applications which have no strict rate requirement for individual users, e.g., image communications. Also note that to achieve all the degrees of freedom of the channel, the size of the user subset should be variable, and thus, $|\Theta| = \sum_{|\theta|}\binom{N}{|\theta|}$ is tremendous. The fixed-size constraint can evidently reduce the implementation complexity. Therefore, Scheduler I can be viewed as a sub-optimal multiuser scheduler to approximately achieve all degree of freedom of MISO channels.

Scheduler II: variable-size user subset with rate requirement

Scheduler II treats the candidate θ with certain SINR threshold requirement and variable size which is only upper bound by K, i.e., $|\Theta| = \sum_{k=1}^K \binom{N}{k}$. Scheduler II maximizes the total throughput and the number of simultaneous transmissions, subject to certain $\{\gamma_n^{\min}\}$. Define $\Psi(\theta) \triangleq \sum_{n\in\theta} w_n \log(1 + \gamma_n^{\min}\alpha(\theta))$. Scheduler II is then defined via the difference $\Delta\Phi = \Phi(\boldsymbol{Q}_{\theta_1}) - \Phi(\boldsymbol{Q}_{\theta_2})$ between the two subsets θ_1 and θ_2 as follows:

$$\begin{aligned}\Delta\Phi = (\Psi(\theta_1) - \Psi(\theta_2))\, &\mathbb{I}\{|\theta_1| = |\theta_2|, \alpha(\theta_1) \geq 1, \alpha(\theta_2) \geq 1\} \\ +(|\theta_1| - |\theta_2|)\, &\mathbb{I}\{|\theta_1| \neq |\theta_2|, \alpha(\theta_1) \geq 1, \alpha(\theta_2) \geq 1\} \\ +\mathbb{I}\{\alpha(\theta_1) \geq 1, \alpha(\theta_2) < 1\} &- \mathbb{I}\{\alpha(\theta_1) < 1, \alpha(\theta_2) \geq 1\}. \qquad (6)\end{aligned}$$

The first two items in (6) indicate the cases that both of the two user subsets satisfy $\{\gamma_n^{\min}\}$, where the first one implies that the subset with higher sum throughput is selected when the two subsets have the same size; and the second one implies that the scheduler prefers to select the subset with larger sizes. The latter two items in (6) indicate the cases of only one subset satisfying $\{\gamma_n^{\min}\}$.

Scheduler II also arises from some practical concerns. Certain SINR threshold is required in cellular voice systems, where the throughput can be roughly denoted by the number of active users.

3.2. Discrete Stochastic Approximation Algorithm

One method for solving (4) is the exhaustive search of all possible user subsets, which can in principle find the optimum solution. However, it is highly inefficient in the sense that most computations are useless and only those corresponding to the optimal one are eventually useful. Moreover, if the channels are time-varying, such scheme cannot track the time-varying optimal user subset. We now present the discrete stochastic approximation algorithm for solving (4) [9]. which has high computational efficiency in the sense that most of the computational cost is spent close to θ^*. We use the unit vectors $\{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_{|\Theta|}\}$ to denote all $|\Theta|$ possible subsets. Denote $\theta^{(m)}$ as the subset visited at the m-th iteration. We map the subset sequence $\{\theta^{(m)},\ m = 1, 2, \cdots\}$ to the unit vector sequence $\{\boldsymbol{D}(m),\ m = 1, 2, \cdots\}$, where $\boldsymbol{D}(m) = \boldsymbol{e}_j$ if $\theta^{(m)} = \theta_j$. At each iteration m, the algorithm updates the state occupation probability $\vec{\pi}(m) = [\pi(m, 1), \cdots, \pi(m, |\Theta|)]^T$, where $\pi(m, j) \in [0, 1]$ and $\sum_{j=1}^{|\Theta|} \pi(m, j) = 1$. The discrete stochastic approximation algorithm is then summarized as follows.

Algorithm 1 [User subset selection]

(a) Initialization: $m \Leftarrow 1$; randomly select $\theta^{(m)} \in \Theta$, and $\hat{\theta}^{(m)} \Leftarrow \theta^{(m)}$; set $\vec{\pi}(m)$ by $\pi(m, \theta^{(m)}) = 1$, and $\pi(m, \theta) = 0$ for all $\theta \neq \theta^{(m)}$.

(b) Sampling and evaluation: Given $\theta^{(m)}$, obtain $\hat{\boldsymbol{Q}}_{\theta^{(m)}}(m)$; calculate $\phi(m, \theta^{(m)})$; uniformly choose $\tilde{\theta}^{(m)} \in \Theta \setminus \theta^{(m)}$; compute $\phi(m, \tilde{\theta}^{(m)})$.

(c) Acceptance: If $\phi(m, \tilde{\theta}^{(m)}) > \phi(m, \theta^{(m)})$, then set $\theta^{(m+1)} = \tilde{\theta}^{(m)}$; otherwise set $\theta^{(m+1)} = \theta^{(m)}$.

(d) Update the state occupation probabilities: $\vec{\pi}(m+1) = \vec{\pi}(m) + \mu(m+1)[\boldsymbol{D}(m+1) - \vec{\pi}(m)]$, where $\mu(m) = \frac{1}{m}$.

(e) Update the estimate of the optimizer: If $\pi(m+1, \theta^{(m+1)}) > \pi(m+1, \hat{\theta}^{(m)})$, then set $\hat{\theta}^{(m+1)} = \theta^{(m+1)}$; otherwise set $\hat{\theta}^{(m+1)} = \hat{\theta}^{(m)}$.

(f) $m \Leftarrow m + 1$, and go to step (b).

The sequence $\{\theta^{(m)},\ m = 1, 2, \cdots\}$ is a Markov chain on the state space Θ, and in general is not expected to converge. In Step (d), $\vec{\pi}(m) = [\pi(m, 1), \cdots, \pi(m, |\Theta|)]^T$ denotes the empirical state occupation probability of the Markov chain at the m-th iteration, and thus, Step (e) is equivalent to $\hat{\theta}^{(m)} = \arg\max_\theta\ \vec{\pi}(m, \theta)$. Hence the algorithm essentially chooses the state most frequently visited by the Markov chain. The sequence $\{\hat{\theta}^{(m)},\ m = 1, 2, \cdots\}$ contains the estimates of θ^*. Under certain conditions, $\hat{\theta}^{(m)} \to \theta^*$ almost surely as $m \to \infty$, or equivalently, the Markov chain spends more time in θ^* than in any other state.

Remark: It remains an open problem to analytically verify the convergence of user subset selection algorithms treated in this paper, though numerical results indicate that it seems to hold.

3.3. Adaptive Algorithm for Time-varying Channels

So far we have assumed that the channels are static and therefore for fixed $\boldsymbol{w}$, the optimal user subset θ^* is time-invariant. Under the static channel condition, a decreasing step-size is employed in Algorithm 1 and (3). With such an approach, the method gradually becomes more and more conservative as the iteration number increases. Whereas, in the time-varying channel case, we need such a step-size that moving away from a state is permitted when the optimal user subset changes. Hence, Step (d) in Algorithm 1 is replaced by $\vec{\pi}(m+1) = \vec{\pi}(m) + \mu[\boldsymbol{D}(m+1) - \vec{\pi}(m)]$, where μ is a fixed step-size satisfying $0 < \mu \leq 1$. The fixed step-size introduces an exponential forgetting factor of the past occupation probabilities and allows to track the slowly time-varying optimum.

4. SIMULATION RESULTS

The simulation conditions are as follows. The base station employs $T = 4$ transmit antennas; there are $N = 8$ users in the system; the power constraint is $\rho = 1$; the noise level is $\eta = 0.05$. For Scheduler I, $|\theta| = 4$; for Scheduler II, $|\theta| \leq 3$ and $\gamma_n^{\min} = 6$. The channel estimate is generated by $\hat{\boldsymbol{Q}}_\theta(m) = \boldsymbol{Q}_\theta + \triangle\boldsymbol{Q}_\theta(m)$, where $\triangle\boldsymbol{Q}_\theta(m)$ contains i.i.d. $\triangle\Omega_{i,j} \sim \mathcal{N}(0, 0.05)$.

Optimal User Subset Selection in Static Channels

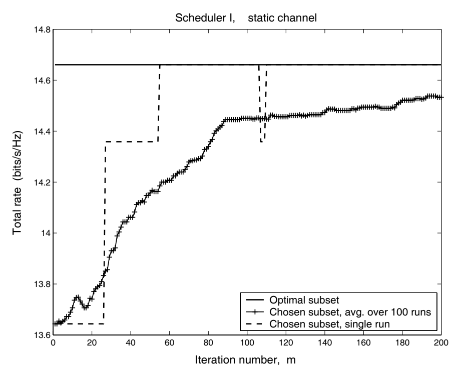

Fig. 2. The total rate of the chosen user subsets versus the iteration number: Scheduler I, fixed channel case.

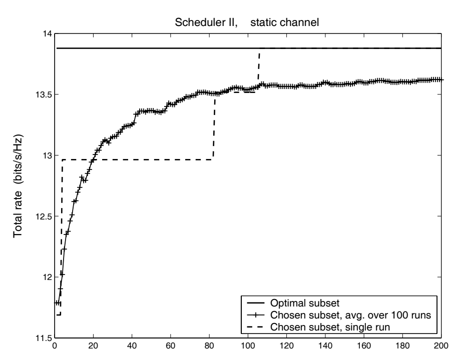

Fig. 3. The total rate of the chosen user subsets versus the iteration number: Scheduler II, fixed channel case.

We first show the effectiveness of Algorithm 1 in terms of throughput maximization with fast convergence. The weights $\boldsymbol{w}$ are all set as 1. The channels are randomly generated and fixed for all simulation runs. Figure 2 shows the total throughput versus the iteration number for Scheduler I. The result in a single simulation run and that averaged over 100 runs, together with the optimal throughput obtained via an exhaustive search, are all shown. Figure 3 shows the similar results for Scheduler II. It is seen that Algorithm 1 can effectively find the optimum, and it can quickly lock on a subset with the performance close to the optimal one.

Tracking Capability of Time-varying Optimal User Subset

We next show the tracking capability of the algorithm in time-varying channels. Suppose the channels keep fixed within $\tau = 200$ slots. The first order AR model over τ is adopted to describe the channel dynamic: $\boldsymbol{Q}_n(t) = \beta_1 \boldsymbol{Q}_n(t-1) + \beta_2 \tilde{\epsilon}_n(t)$, where $\tilde{\epsilon}_n(t)$ contains i.i.d. $\tilde{\epsilon}_{i,j}(t) \sim \mathcal{N}(0,1)$; $\beta_2 = (1-\beta_1^2)^{1/2}$ and $\beta_1 = 0.9$. The step size is fixed as $\mu = 0.02$. $\boldsymbol{w}$ are all set as 1. Figure 4 shows the results for Scheduler I over a single run, where the algorithm can closely track the time-varying optimum.

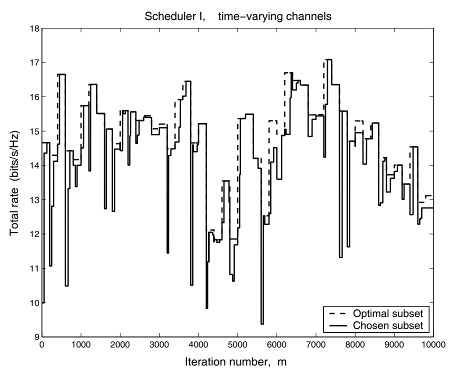

Fig. 4. The total rate of the chosen user subsets versus the iteration number: Scheduler I, time-varying channel case.

Fairness Guarantee

Finally, we show the system performance in terms of the fairness. Assume that there are totally $N = 5$ users, which have the normalized throughput priorities $\overrightarrow{\phi} = [\frac{1}{8}, \frac{1}{8}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}]$. The time-varying channel case is treated, where the same AR model as that in Fig. 4 is used. Note that there are 400 iterations within each interval, during which $\boldsymbol{w}(i)$ is kept fixed. Figure 5 shows the normalized rates of all users versus the interval indexes for Scheduler II. It is seen that although the rate requests for all users are not satisfied within a short time scale, the long term fairness can be well guaranteed.

5. CONCLUSIONS

We have developed a multiuser scheduling framework for downlink MISO beamforming systems. Multiuser downlink beamforming is treated as the physical-layer implementation. The optimal beamforming scheme can be calculated for a given user subset. Then the multiuser scheduling problem refers to the optimum subset selection at each scheduling interval to maximize the weighted system throughput. We have developed the DSA algorithm to efficiently achieve optimal subset selection. The algorithm is also able to track the time-varying optimum when the channels vary. The algorithm is iterative in nature, and we have presented two practical

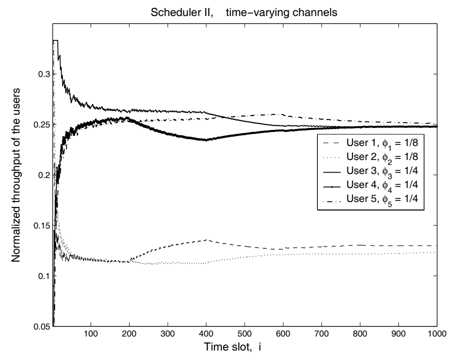

Fig. 5. The normalized throughput of all users versus the time slot number: Scheduler II, time-varying channel case.

multiuser schedulers, with or without minimum rate constraints on individual users. We present simulation results to demonstrate that the algorithms can effectively find the optimal user subset with good convergence performances, and adaptively track the time-varying optimum in the nonstationary environments. The system can also achieve fairness among all users over large time scales.

6. REFERENCES

[1] V. Bharghavan, S. Lu, and T. Nandagopal, "Fair scheduling in wireless networks: Issues and approaches," *IEEE Person. Commun.*, vol. 6, no. 1, pp. 44–53, Feb. 1999.

[2] Y. Liu and E.W. Knightly, "Opportunistic fair scheduling over multiple wireless channels," in *Proc. 2003 IEEE INFOCOM*, San Francisco, CA, Mar. 2003, pp. 1106–1115.

[3] X. Liu, E.K.P. Chong, and N.B. Shroff, "Opportunistic transmission scheduling with resource-sharing constraints in wireless networks," *IEEE J. Select. Areas Commun.*, vol. 19, no. 10, pp. 2053–2064, Oct. 2001.

[4] P. Viswanath, D.N.C. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Trans. Inform. Theory*, vol. 48, no. 6, pp. 1277–1294, June 2002.

[5] H. Boche and M. Schubert, "Optimal multi-user interference balancing using transmit beamforming," *Wireless Person. Commun.*, vol. 26, no. 4, pp. 305–324, Sep. 2003.

[6] M. Schubert and H. Boche, "Solution of the multiuser downlink beamforming problem with individual SINR constraints," *IEEE Trans. Veh. Tech.*, vol. 53, no. 1, pp. 18–28, Jan. 2004.

[7] C. Li and X. Wang, "Adaptive opportunistic fair scheduling for utility-based power control systems," in *Proc. Conf. Inform. Sci. Sys. (CISS'04)*, Princeton, NJ, Mar. 2004.

[8] S. Andradóttir, "Accelerating the convergence of random search methods for discrete stochastic optimization," *ACM Trans. Model. Comput. Simu.*, vol. 9, no. 4, pp. 349–380, Oct. 1999.

[9] W.B. Gong, Y.C. Ho, and W. Zhai, "Stochastic comparison algorithm for discrete optimization with estimation," *SIAM J. Control and Optim.*, vol. 10, no. 2, pp. 384–404, Feb. 1999.