User Scheduling in Massive MIMO

Hong Yang
Mathematics of Networks and Communications Research Department
Bell Laboratories, Nokia, Murray Hill, NJ 07974 USA
h.yang@research.bell-labs.com

SPAWC 2018
www.spawc2018.org
19th IEEE INTERNATIONAL WORKSHOP ON SIGNAL PROCESSING ADVANCES IN WIRELESS COMMUNICATIONS
June 25-28, 2018 ELITE CITY RESORT HOTEL KALAMATA GREECE

Abstract

Massive MIMO relies on nearly orthogonal user channels to achieve unprecedented spectral efficiency. However, in LoS (line-of-sight) environment, some users can be subjected to similar channel vectors. Serving users with similar channel vectors simultaneously can severely compromise the throughput performance to all users. We propose a scheduler that identifies users with similar channels and serves them in separate time slots with properly assigned data rates, while aiming to provide fair service to all users and maximize the system spectral efficiency at the same time. Simulation results show the effectiveness of the scheduler on both downlink and uplink of a single cell Massive MIMO with MR (maximum ratio) processing or ZF (zero-forcing) processing, and that channel correlation threshold for scheduling users is an important design parameter that can be fine-tuned to optimize the user throughput performance.

Maximal Feasible Grouping

The algorithm is to divide the K users into groups such that correlation between users in each group is upper bounded by a threshold and that each group has a maximum number of users (i.e., adding any users into the group will cause the maximum correlation in that group to exceed the given threshold).

More precisely, given a $K \times K$ hollow symmetric matrix $\Gamma = [\gamma_{i,j}]$, let $\mathcal{K} = \{1, \cdots, K\}$. We want to find J distinct subsets $\mathcal{K}_j \subset \mathcal{K}$ such that $\gamma_{k_1,k_2} \le \gamma_{\mathrm{h}}$ for any $k_1, k_2 \in \mathcal{K}_j, j = 1, \cdots, J$ and

$$\mathcal{K} = \bigcup_{j=1}^{J} \mathcal{K}_j.$$

We call $(\Gamma, K, \{\mathcal{K}_1, \cdots, \mathcal{K}_J\}, \gamma_{\mathrm{h}})$ a *feasible grouping*.

To achieve maximum multi-user multiplexing gain, we would like each subset $\mathcal{K}_j$ to contain as many elements as possible. Thus, we make the following definition.

Definition. A *maximal feasible grouping* is a feasible grouping such that $\mathcal{K}_1 = \mathcal{K}$ or for any $k \in \mathcal{K} \backslash \mathcal{K}_j$, there exists a $k_1 \in \mathcal{K}_j$ such that $\gamma_{k,k_1} > \gamma_{\mathrm{h}}$.

One-Parameter SINRs

If it is a one-parameter SINR target, i.e., the SINR target for the K users is parametrized by a single parameter: $s_k = s_k(\delta), \breve{s}_k = \breve{s}_k(\breve{\delta}), k = 1, \cdots, K$, then bisection searches can be carried out to find the optimal parameter $\delta^\star$ and $\breve{\delta}^\star$ such that $\min_k\{s_k(\delta)\}$ and $\min_k\{\breve{s}_k(\breve{\delta})\}$ are maximized. We propose some one-parameter SINR targets for scheduling purposes.

Algorithm

Construction of a maximal feasible grouping:

Given a $K \times K$ hollow symmetric matrix $\Gamma = [\gamma_{i,j}]$ with $\gamma_{i,j} \in [0,1], \forall i, j$, and a correlation threshold γ_{h}, let $\gamma_{\max} = \max_{i,j}\{\gamma_{i,j}\}$.

1. If $\gamma_{\max} \le \gamma_{\mathrm{h}}$, then we have a unique maximal feasible grouping, which has only one element: $\mathcal{K}_1 = \mathcal{K}$. Done.
2. If $\gamma_{\max} > \gamma_{\mathrm{h}}$, find all the columns in Γ with all its entries $\le \gamma_{\mathrm{h}}$. Denote the set of the column indices for these columns as $\mathcal{K}_0$. Let $\mathcal{K}_0^c = \mathcal{K} \backslash \mathcal{K}_0 \neq \emptyset$. Continue.
3. Randomly pick an $i_1 \in \mathcal{K}_0^c$. Construct a user set $\mathcal{K}_{(i_1)}$ that includes i_1 as follows:
   (a) Let $\{i_1\} \to \mathcal{K}_{(i_1)}$, i.e., add i_1 to $\mathcal{K}_{(i_1)}$.
   (b) Let $\mathcal{L}_{(i_1)} = \{k \in \mathcal{K} : \gamma_{k,j} \le \gamma_{\mathrm{h}}, \forall j \in \mathcal{K}_{(i_1)}\}$. If $\mathcal{L}_{(i_1)} \neq \emptyset$, randomly pick $k \in \mathcal{L}_{(i_1)}$ and add it to $\mathcal{K}_{(i_1)}$. Repeat until $\mathcal{L}_{(i_1)} = \emptyset$.
   (c) Let $\mathcal{K}_{(i_1)} \cup \mathcal{K}_0 \to \mathcal{K}_{(i_1)}$, i.e., set $\mathcal{K}_{(i_1)}$ as $\mathcal{K}_{(i_1)} \cup \mathcal{K}_0$.
4. Randomly pick $i_2 \in \mathcal{K} \backslash \mathcal{K}_{(i_1)}$, and construct a user set $\mathcal{K}_{(i_2)}$ as in Step 3. If $\mathcal{K}_{(i_1)} \cup \mathcal{K}_{(i_2)} = \mathcal{K}$, done. Otherwise, randomly pick $i_3 \in \mathcal{K} \backslash [\mathcal{K}_{(i_1)} \cup \mathcal{K}_{(i_2)}]$. Construct a new user set: $\mathcal{K}_{(i_3)}$ as in Step 3.
5. In general, if $\mathcal{K} \backslash \left(\cup_{j=1}^{n-1} \mathcal{K}_{(i_j)}\right) \neq \emptyset$, randomly pick $i_n \in \mathcal{K} \backslash \left(\cup_{j=1}^{n-1} \mathcal{K}_{(i_j)}\right) \neq \emptyset$ and construct a new user set $\mathcal{K}_{(i_n)}$ as in Step 3. Repeat until the union of user sets equals $\mathcal{K}$.

Numerical Examples

We assume:

- Number of service antennas $M = 256$
- Number of users $K = 18$

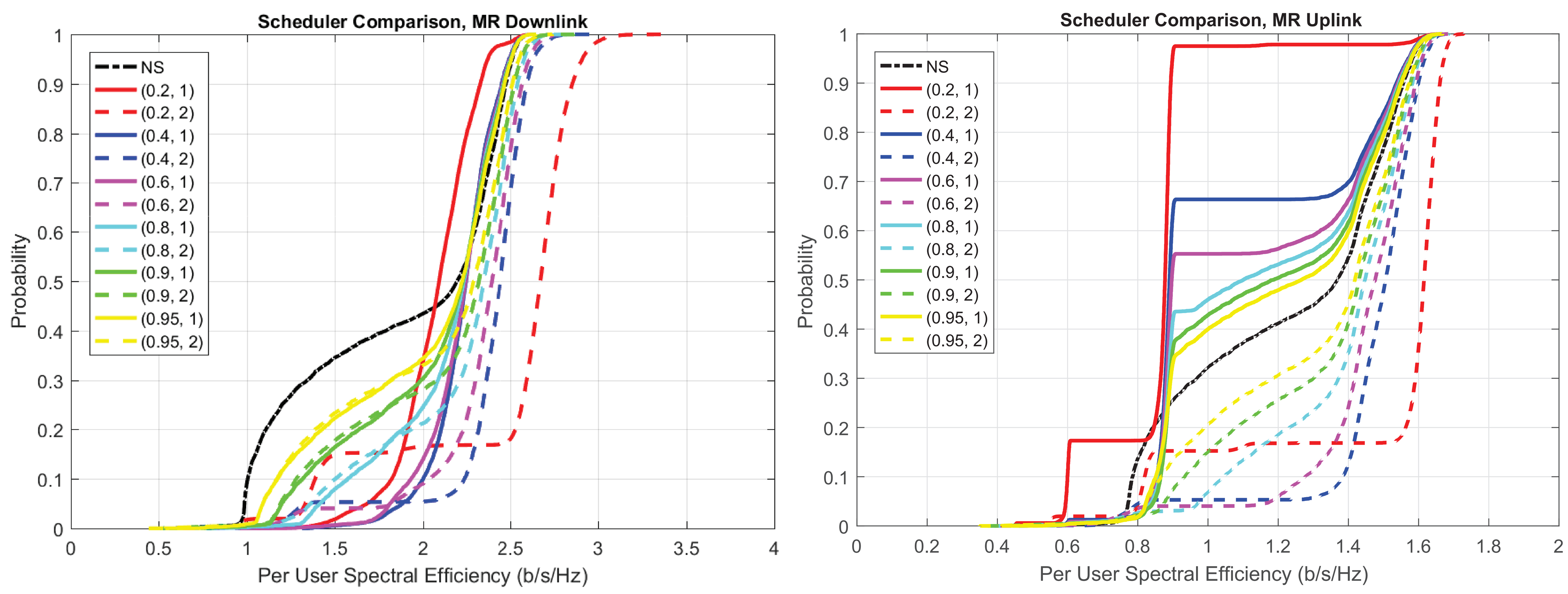

Figure 1: Maximum Ratio downlink (left); Maximum Ratio uplink (right)

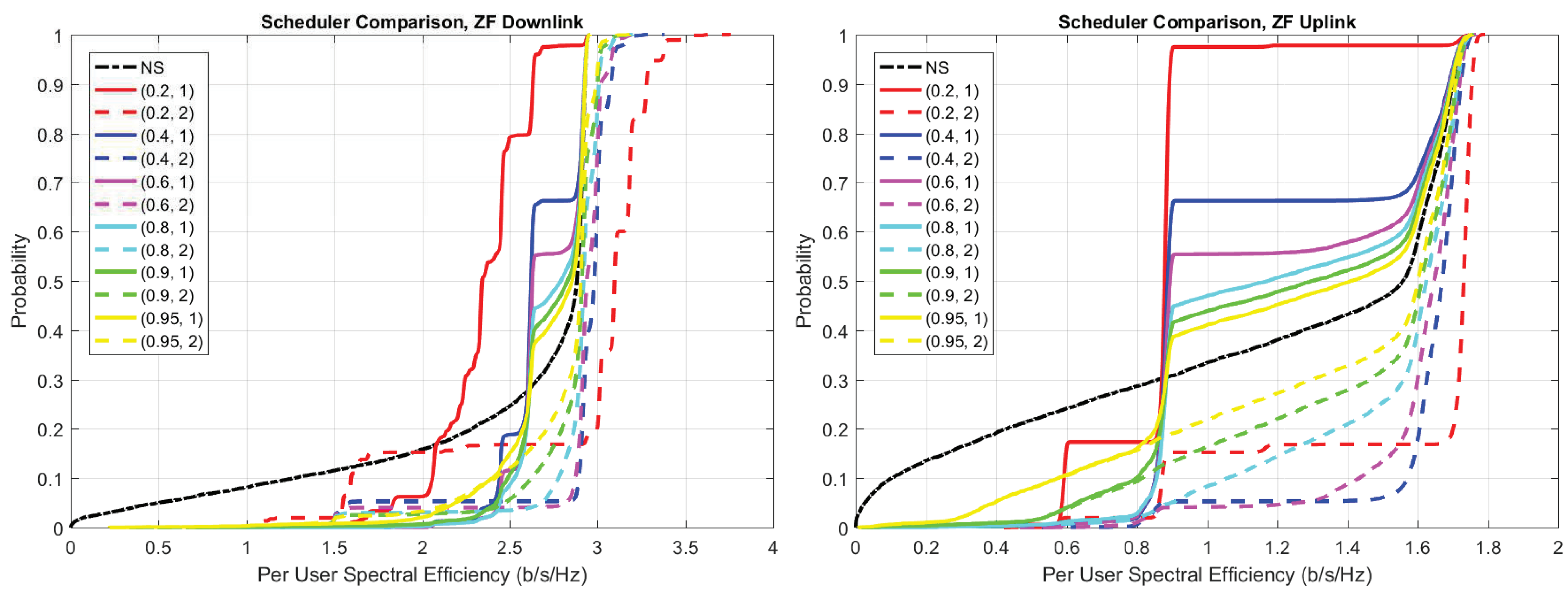

Figure 2: Zero-Forcing downlink (left); Zero-Forcing uplink (right)