

# An Iterative List-Based Multiuser Detector for Overloaded Receivers in a Rayleigh Fading Channel

Michael Krause, Desmond P. Taylor and Philippa A. Martin

Department of Electrical and Computer Engineering  
University of Canterbury, Christchurch, New Zealand  
Email: {michael.krause, taylor, p.martin}@elec.canterbury.ac.nz

**Abstract**—We consider a wireless communication system where multiple co-channel users transmit data via a synchronous, frequency-flat Rayleigh fading channel. The receiver employs an antenna array. The paper develops an iterative list multiuser detector for overloaded applications where the number of transmitted signals exceeds the number of receive antennas. The receiver uses a linear preprocessor to reduce co-channel interference followed by a multiuser detector with an iterative groupwise symbol detection algorithm that extracts a list of the most likely user symbols. Simulation results show that the proposed detector provides good complexity-performance trade offs.

## I. INTRODUCTION

In a wireless communication system with multiple transmitters and multiple receive antennas, the detection of co-channel signals is key to achieving capacity and ensuring reliable data transfer. We consider the uplink mode of such a system where multiple single-antenna users independently but synchronously<sup>1</sup> transmit data to a receiver with an antenna array. The multiple co-channel signals cause severe co-channel interference (CCI) at the receiver. We assume an overloaded receiver system where the number of transmitted signals exceeds the number of receive antennas. Under overload, the receiver operates in under-determined conditions which cause linear detection techniques to perform poorly and the demodulation of the signals of interests becomes a challenging task.

Recently, multiuser detection (MUD) for these overloaded receivers has attracted considerable attention. While comprehensive fundamental work on MUD is available in [1], we restrict ourselves to the overloaded case.

Joint detection of multiple co-channel signals transmitted over a frequency-flat Rayleigh fading channel was studied in [2]. It showed that the response differences among the received co-channel signals can be exploited to separate and detect the user symbols. Joint MUD techniques such as joint maximum likelihood (JML) [2] and joint maximum a posteriori probability are optimum and perform well under overload. They require an exhaustive search over all user symbols. Complexity is exponential in the number of co-channel users. This prohibits its use in most practical systems and motivates the design of suboptimum, reduced complexity MUD.

<sup>1</sup>The extension to the non-synchronous case requires a synchronization process.

The iterative MUD scheme in [3] appears to be the first MUD algorithm for overloaded multiple input multiple output systems. The algorithm searches over groups of user symbols and makes soft decisions. Complexity is exponential in group size but increases only linearly with the number of groups.

Many approaches split the MUD process into linear preprocessing followed by detection stages, e.g. [4]–[8]. Even though a linear preprocessor fails to cancel all CCI under overload, it reduces its effect. This allows complexity reductions in the detection stage. Reduced complexity MUD in [5]–[7] relies on minimum mean square error MUD with interference cancellation [5], [6] or the formation of a spatially reduced-state search trellis<sup>2</sup> [7]. In [8], we proposed a unified list-based MUD algorithm referred to as parallel detection with interference estimation (PD-IE). It employs either a special purpose receive beamformer or maximum-ratio combining (MRC) as a preprocessing technique. PD-IE estimates residual CCI and searches over subsets of the user symbols. It can be used with arbitrary antenna array geometries. PD-IE has exponential complexity in the size of the subsets making it best suited to scenarios with modest subset sizes, e.g. line-of-sight channels with sufficient spatial separation between users.

In this paper we propose the List Group Search MUD (LGS-MUD) algorithm for frequency-flat Rayleigh fading channels and arbitrary user spacing. This channel results in large subsets, making the PD-IE too complex. As with the PD-IE, parallel processing branches are used, one for each user. Symbol candidates are then computed for each group. These are used to find an ordered list of the most likely user symbols. Results show that our approach achieves good performance at lower complexity than JML MUD.

In Section II the system model and the preprocessor are described. Section III presents the LGS-MUD algorithm. Performance results are shown in Section IV. Conclusions are drawn in Section V.

## II. SYSTEM MODEL AND PREPROCESSOR

We consider a single-input multiple-output system where the receiver has an  $M$ -element arbitrary receive antenna array and uncorrelated antennas. There are  $D$  independent single antenna

<sup>2</sup>The algorithm requires a diagonally-banded channel matrix as can be achieved by a uniform circular array and a receive beamformer preprocessor. It is restricted to line-of-sight channels.

transmitters whose signals impinge on each receive antenna. The receiver load factor is  $f = D/M$  and it is overloaded for  $f > 1$ . The  $D$  users are assumed to transmit symbol synchronous equal energy quadrature amplitude modulation (QAM) signals over a frequency-flat Rayleigh fading channel. At each antenna, the received signal is passed through a filter matched to the transmitted pulse shape and then sampled at symbol rate to give the  $M \times 1$  received signal vector

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{z} \quad (1)$$

where  $\mathbf{A}$  is the  $M \times D$  channel response matrix and  $\mathbf{s} = [s_1, s_2, \dots, s_D]^T$  is the  $D \times 1$  symbol vector of user symbols  $s_d$ . Each  $s_d$  is independent and uniformly drawn from an alphabet  $\mathcal{A}$ . The elements of  $\mathbf{A}$  are independent, complex Gaussian random variables with zero mean and variance  $1/2$  in each dimension. The  $M \times 1$  vector  $\mathbf{z}$  denotes temporally uncorrelated noise with zero mean and autocorrelation  $\Phi_{zz} = E[\mathbf{z}\mathbf{z}^H]$ , where  $E[\cdot]$  is the expectation operator. The noise is spatially uncorrelated so that  $\Phi_{zz} = \sigma_z^2 \mathbf{I}$ , where  $\sigma_z^2$  denotes the noise variance and  $\mathbf{I}$  is the  $M \times M$  identity matrix. Any time dependence in equations has been dropped throughout the paper for convenience.

The received signal vector  $\mathbf{x}$  and the channel response matrix  $\mathbf{A}$  are input to a preprocessor. See [8] for a description of various linear preprocessors. Assuming perfect channel state information, here we consider MRC diversity combining. This is the optimum diversity combining technique. It maps the  $M \times 1$  receive vector  $\mathbf{x}$  into a  $D \times 1$  vector  $\tilde{\mathbf{y}}$  such that each of the  $D$  users has maximum SNR in one of the components of  $\tilde{\mathbf{y}}$ . The preprocessor output is a weighted linear combination of the different signal replicas observed at the receive antennas. The MRC weight matrix is given by  $\tilde{\mathbf{W}} = \mathbf{A}^H$  [9]. We then obtain the MRC receive vector as

$$\tilde{\mathbf{y}} = \tilde{\mathbf{W}}\mathbf{x} = \tilde{\mathbf{W}}\mathbf{A}\mathbf{s} + \tilde{\mathbf{W}}\mathbf{z}. \quad (2)$$

The resulting correlation in the colored noise term  $\tilde{\mathbf{W}}\mathbf{z}$  degrades detection performance. As a result, a noise whitening filter is used to regain optimum performance [10].

We define the  $D \times D$  matrix  $\tilde{\mathbf{H}} = \mathbf{A}^H \mathbf{A}$  and compute its square root inverse, denoted  $\tilde{\mathbf{H}}^{-1/2}$ . Letting  $\mathbf{H} = \tilde{\mathbf{H}}^{-1/2}$ , the noise whitened MRC output is obtained as

$$\mathbf{y} = \tilde{\mathbf{H}}^{-1/2} \tilde{\mathbf{y}} = \mathbf{H}\mathbf{s} + \mathbf{n} \quad (3)$$

where  $\mathbf{n} = \mathbf{H}^{-1} \mathbf{A}^H \mathbf{z}$  is the  $D \times 1$  whitened noise vector. The JML decision metric applied to (3) becomes

$$\hat{s} = \arg \min_{s \in \mathcal{A}^D} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \quad (4)$$

where  $\hat{(\cdot)}$  denotes estimate and  $\mathcal{A}^D$  is the combined  $2D$ -dimensional alphabet of the  $D$  users.

Fig. 1 provides an example of the energy of the preprocessor output matrix  $\mathbf{H}$  after MRC combining and noise whitening  $M = 6$  receive antennas and  $D = 9$  equal energy users.

A corresponding  $D \times D$  sparsity matrix  $\mathbf{P}$  is shown in Fig. 2. It is formed by denoting ‘‘low’’ and ‘‘high’’ energy elements

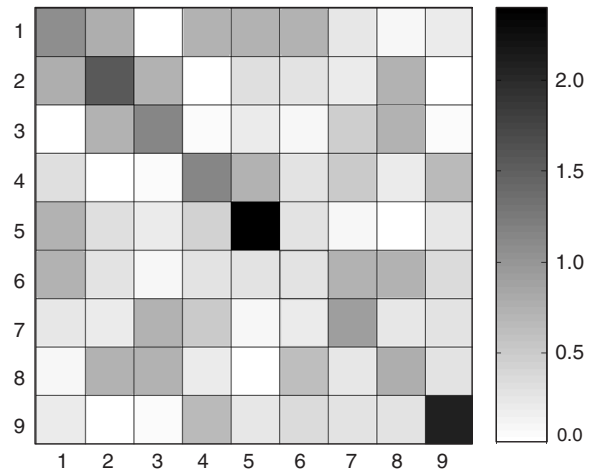


Fig. 1. Example of the energy of the preprocessor output matrix  $\mathbf{H}$ ,  $(\mathbf{H}^H \mathbf{H})^{(1/2)}$ , for a receiver with  $M = 6$  antennas and  $D = 9$  single antenna users in a frequency-flat Rayleigh fading channel.

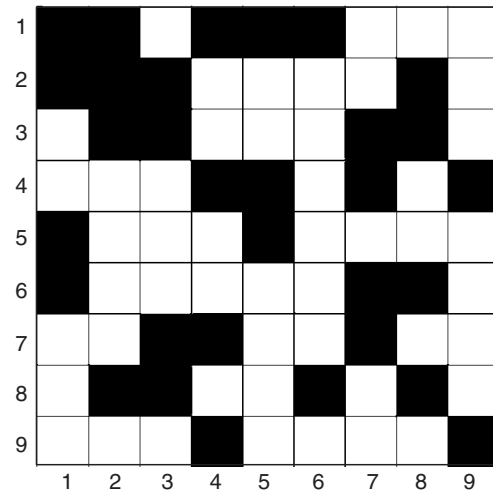


Fig. 2. Sparsity matrix  $\mathbf{P}$  for the example in Fig. 2 using the SEAIR and SSSER criteria in [8].

in  $\mathbf{H}$  as ‘0’ and ‘1’ values, respectively. We use the signal-energy-to-average-interference-ratio (SEAIR) and the signal-to-strongest-signal-energy-ratio-criteria (SSSER) proposed in [8] to specify  $\mathbf{P}$  from  $\mathbf{H}$ . We denote the  $d$ -th row vectors of  $\mathbf{H}$  and  $\mathbf{P}$  as  $\mathbf{h}[d] \in \mathbf{H}$  and  $\mathbf{p}[d] \in \mathbf{P}$ . The matrix element in the  $d$ -th row and  $u$ -th column of  $\mathbf{H}$  and  $\mathbf{P}$  is denoted  $h_{du}$  and  $p_{du}$ , respectively. We further define column group matrices  $\mathbf{H}_j$ , each containing one or more columns of  $\mathbf{H}$ .

We next obtain enumeration sets  $U_e[d] = \{u | p_{du} > 0, p_{du} \in \mathbf{p}[d]\}$  and complementary sets  $\bar{U}_e[d] = \{v | p_{dv} = 0, p_{dv} \in \mathbf{p}[d]\}$  for  $d = 1, 2, \dots, D$ . These allow us to write the sets of high and low energy user symbols as

$$\tau[d] = \{s_u | u \in U_e[d]\}, \quad \omega[d] = \{s_v | v \in \bar{U}_e[d]\}, \quad (5)$$

respectively. Correspondingly, each row vector  $\mathbf{h}[d]$  can be split into two subset vectors  $\mathbf{h}_\tau[d]$  and  $\mathbf{h}_\omega[d]$ . As an example, in row  $d = 3$  of Fig. 2, we have the sets  $\tau[3] =$

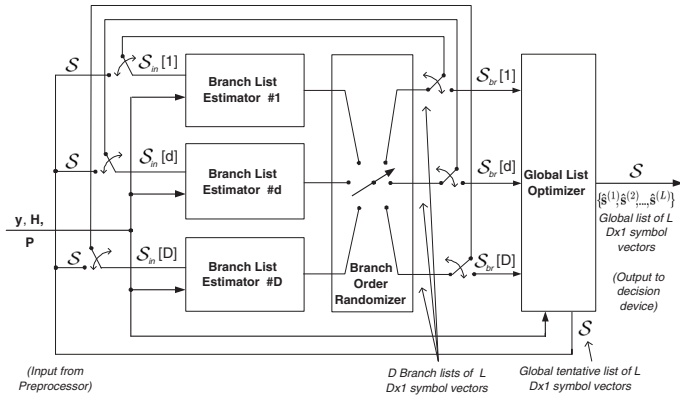


Fig. 3. Block diagram of the LGS-MUD detector.

$\{s_2, s_3, s_7, s_8\}$  and  $\omega[3] = \{s_1, s_4, s_5, s_6, s_9\}$  with corresponding subset vectors  $\mathbf{h}_\tau[3] = \{h_{32}, h_{33}, h_{37}, h_{38}\}$  and  $\mathbf{h}_\omega[3] = \{h_{31}, h_{34}, h_{35}, h_{36}, h_{39}\}$ , respectively. See [8] for more details.

### III. THE LGS-MUD ALGORITHM

The proposed LGS-MUD algorithm is illustrated in Fig. 3. The detector takes the quantities  $\mathbf{y}$ ,  $\mathbf{H}$  and  $\mathbf{P}$  from the preprocessor as inputs and outputs the ordered  $(D \times 1)$  list of  $L$  symbol vectors,  $\mathcal{S} = \{\hat{\mathbf{s}}^{(1)}, \hat{\mathbf{s}}^{(2)}, \dots, \hat{\mathbf{s}}^{(L)}\}$ , where the ordering is from most to least likely.

LGS-MUD uses  $D$  parallel processing branches each containing a Branch List Estimator (BLE). The  $d$ -th BLE has  $\mathbf{y}$ ,  $\mathbf{H}$ ,  $\mathbf{P}$  and the list  $\mathcal{S}_{in}[d]$  as inputs and outputs the branch list  $\mathcal{S}_{br}[d] = \{\hat{\mathbf{s}}_{br}^{(1)}[d], \hat{\mathbf{s}}_{br}^{(2)}[d], \dots, \hat{\mathbf{s}}_{br}^{(L)}[d]\}$ . The elements  $\hat{\mathbf{s}}_{br}^{(k)}[d] \in \mathcal{S}_{br}[d]$  are the estimated  $(D \times 1)$  branch symbol vectors. LGS-MUD starts by initializing the list  $\mathcal{S}$  with random symbol values drawn from the alphabet  $\mathcal{A}$ . It then performs  $Q$  global iterations in which the lists  $\mathcal{S}_{in}[d]$  are initialized with symbol values from the tentative list  $\mathcal{S}$ .

LGS-MUD exchanges tentative branch lists  $\mathcal{S}_{br}[d]$  between BLE blocks to update estimates of the symbols with low energy contribution. Therefore, a switch at the input of each BLE is required as shown in Fig. 3. The order in which the branch lists are passed to the next branch is randomized to break up statistical dependencies. After  $Q_{BLE}$  iterations, the  $D$  lists  $\mathcal{S}_{br}[d]$  are input to an optimizer stage. Here, the list  $\mathcal{S}$  is updated by searching over the symbol groups. The improved list  $\mathcal{S}$  is fed back to the  $D$  branches and after  $q = Q$  iterations LGS-MUD outputs  $\mathcal{S}$ . A decision device selects the first element  $\hat{\mathbf{s}}_1 \in \mathcal{S}$  as the best estimate  $\mathbf{s}$ . Alternatively,  $\mathcal{S}$  can be used to compute soft information for subsequent stages such as error control decoders.

We describe branch list estimation in Subsection III-A and global list optimization in Subsection III-B.

#### A. Branch List Estimation

The  $d$ -th BLE searches only over the symbols grouped in the set  $\tau[d]$ . Using the subset vectors  $\mathbf{h}_\tau[d]$  and  $\mathbf{h}_\omega[d]$ , the  $d$ -th receive component can be written as

$$y[d] = \mathbf{h}_\tau[d]\mathbf{s}_\tau + \mathbf{h}_\omega[d]\mathbf{s}_\omega + n[d] \quad (6)$$

where  $\mathbf{s}_\tau$ ,  $\mathbf{s}_\omega$  are symbol vectors corresponding to the subsets  $\tau[d]$ ,  $\omega[d]$  and  $n[d] \in \mathbf{n}$  is the  $d$ -th noise element. In a fading channel, the size of the high energy subsets,  $|\tau[d]|$ , is a random variable with  $0 \leq |\tau[d]| \leq D$ . The resulting search complexity often exceeds the limit of the receiver. In LGS-MUD we solve this problem by splitting the high energy subset  $\tau[d]$  into  $G$  groups,  $\tau_j[d]$ , such that  $\tau[d] = \{\tau_1[d], \tau_2[d], \dots, \tau_G[d]\}$ . The size of the  $j$ -th group is denoted  $|\tau_j[d]|$ , where  $j = 1, 2, \dots, G$ . We also split the corresponding subset vector  $\mathbf{h}_\tau[d]$  into  $G$  group vectors,  $\mathbf{h}_j[d]$ . Using group notation,  $\mathbf{h}_\tau[d]$  is decomposed into  $\mathbf{h}_\tau[d] = [\mathbf{h}_1[d], \mathbf{h}_2[d], \dots, \mathbf{h}_G[d]]$ .

In order to search over the  $j = 1$  group, we can rewrite (6) as

$$y[d] = \mathbf{h}_1[d]\mathbf{s}_1 + \left[ \sum_{i=2}^G \mathbf{h}_i[d]\mathbf{s}_i + \mathbf{h}_\omega[d]\mathbf{s}_\omega + n[d] \right] \quad (7)$$

where  $\mathbf{s}_i$  denotes the group vector of the symbols in  $\tau_i[d]$ . Note that for the detection of the  $j = 1$  group all terms within the brackets are undesired components defined as  $y_u[d]$ . Hence, (7) may be written as

$$y[d] = \mathbf{h}_1[d]\mathbf{s}_1 + y_u[d]. \quad (8)$$

The exhaustive search of the  $d$ -th BLE over the symbols of the  $j$ -th group,  $\tau_j[d]$ , is described by

$$\hat{\mathbf{s}}_j = \arg \min_{\mathbf{s}_j \in \tau_j[d]} \|y_j[d] - \mathbf{h}_j[d]\mathbf{s}_j\|^2, j = 1, 2, \dots, G \quad (9)$$

where the received component for the  $j$ -th group is

$$y_j[d] = y_0[d] - \sum_{i=1, i \neq j}^G \mathbf{h}_i[d]\mathbf{s}_i \quad (10)$$

and the received signal after cancelling energy contributions from low energy users is given by

$$y_0[d] = y[d] - \mathbf{h}_\omega[d]\mathbf{s}_\omega. \quad (11)$$

This is done before processing the first group  $\tau_1[d]$ .

We use (9), (10) and (11) to obtain branch symbol lists  $\mathcal{S}_{br}[d]$  as described in Table I. For a given candidate symbol vector  $\hat{\mathbf{s}}$ , we use the squared Euclidean distance metric

$$e[d] = |y[d] - \mathbf{h}[d]\hat{\mathbf{s}}|^2 \quad (12)$$

to order the symbol list. The branch list estimation algorithm in Table I is employed by each BLE during  $Q_{BLE}$  iterations.

#### B. Global List Optimizer

After the branch lists  $\mathcal{S}_{br}[d]$  have been computed they are passed to a Global List Optimizer (GLO). Here, the lists are searched again to find symbol vectors  $\hat{\mathbf{s}}^{(l)} \in \mathcal{S}$  that minimize (4). The global list optimization algorithm is also based on a groupwise search to reduce the computational complexity. We define the set  $v$  which contains symbols for all  $D$  user signals. This is in contrast to the BLEs, where the symbols are divided into sets with low and high energy contribution,  $\tau$  and  $\omega$ , respectively. The set  $v$  is given by

$$v = \{s_1, s_2, \dots, s_D\}. \quad (13)$$

TABLE I

ITERATIVE GROUPWISE BRANCH LIST ESTIMATION ALGORITHM

- 1) Initialize the  $d$ -th branch list  $\mathcal{S}_{br}[d]$  of  $L$  ( $D \times 1$ ) symbol vectors and the corresponding list of  $L$  branch error metrics  $\mathcal{E}_{br}[d]$ .
- 2) Find the elements  $s_{br}^{(l)}[d] \in \mathcal{S}_{br}[d]$  and  $e_{br}^{(l)}[d] \in \mathcal{E}_{br}[d]$  by computing (9) and (12) for all  $L$  elements  $s_{in}^{(l)}[d]$  from the input list  $\mathcal{S}_{in}[d]$ .
- 3) Apply  $\mathbf{p}[d] \in \mathbf{P}$  to obtain symbol values for  $\mathbf{s}_w$  from the first element  $s_{in}^{(1)}[d] \in \mathcal{S}_{in}[d]$ . Cancel residual CCI using (11).
- 4) For  $\theta = 1 : \Theta$  do
  - a) Find the members of the  $G$  groups  $\tau_j$ ,  $j = 1, 2, \dots, G$ , by choosing  $|\tau_j|$  column indices  $u$  from  $\mathbf{p}[d] \in \mathbf{P}$  with  $p_{du} = 1$ . If  $\theta = 1$  select the indices  $u$  according to decreasing value of  $\|h_{du}\|^2$  otherwise use random selection.
  - b) For  $j = 1 : G$  do
    - i) Perform an exhaustive search over all symbol values of the  $j$ -th group  $\tau_j$  using (9) and (12). Symbol values for all elements  $\mathbf{s}_i$  from the undesired groups  $i$  are drawn from  $s_{in}^{(1)}[d] \in \mathcal{S}_{in}[d]$ . Update the lists  $\mathcal{S}_{in}[d]$  and  $\mathcal{E}_{br}[d]$ .
    - ii) Calculate  $y_{j+1}[d]$  using (10).
    - iii) Increase the group index  $j$ ,  $j = j + 1$ .
  - c) Increase the iteration index  $\theta$ ,  $\theta = \theta + 1$ .
- 5) Output the list  $\mathcal{S}_{br}[d]$ .

Defining the subsets  $v_j$  for  $F$  groups, we have  $v = \{v_1, v_2, \dots, v_F\}$ . Similar to (7), we can now write (3) using column group matrices as

$$\mathbf{y} = \mathbf{H}_j \mathbf{s}_j + \left[ \sum_{i=1, i \neq j}^F \mathbf{H}_i \mathbf{s}_i + \mathbf{n} \right] \quad (14)$$

$$= \mathbf{H}_j \mathbf{s}_j + \mathbf{y}_u \quad (15)$$

where, for the detection of the  $j$ -th group, the terms within the brackets are the undesired components  $\mathbf{y}_u$ . The group symbol vectors  $\mathbf{s}_i$  and  $\mathbf{s}_j$  are drawn from their corresponding sets  $v_i \in v$  and  $v_j \in v$ , respectively. Note that the columns of  $\mathbf{H}_i$  and  $\mathbf{H}_j$  are the permuted column vectors of  $\mathbf{H}$ . MUD decisions are based on

$$\hat{\mathbf{s}}_j = \arg \min_{\mathbf{s}_j \in v_j} \|\mathbf{y}_j - \mathbf{H}_j \mathbf{s}_j\|^2, j = 1, 2, \dots, F \quad (16)$$

where the received component for the  $j$ -th group is

$$\mathbf{y}_j = \mathbf{y} - \sum_{i=1, i \neq j}^F \mathbf{H}_i \mathbf{s}_i. \quad (17)$$

All candidate symbol vectors  $\hat{\mathbf{s}}$  are evaluated by their Euclidean distance

$$e = \|\mathbf{y} - \mathbf{H} \hat{\mathbf{s}}\|^2. \quad (18)$$

The GLO stage stores the tentative global list  $\mathcal{S}$  and the corresponding list  $\mathcal{E}$  that holds the  $L$  error metrics  $e^{(l)} \in \mathcal{E}$ . The list  $\mathcal{S}$  is fed back to the  $D$  branches and output by the LGS-MUD detector after  $Q$  global iterations. Table II summarizes the global list optimization algorithm.

#### IV. PERFORMANCE EVALUATION

Performance results for both LGS-MUD and JML are obtained by simulation. We assume that  $D$  equal energy single antenna users transmit QPSK (4-QAM) signals through

TABLE II

ITERATIVE GROUPWISE GLOBAL LIST OPTIMIZATION ALGORITHM

- 1) Compute (18) for all non-redundant input vectors  $s_{br}^{(k)}[d] \in \mathcal{S}_{br}[d]$ . There are  $1 \leq K \leq (LD)$  non-redundant input vectors. Update the tentative list  $\mathcal{S}$  by storing the  $L$  vectors with minimum error metric in  $\mathcal{S}$ . The corresponding error metrics are stored in  $\mathcal{E}$ .
- 2) Allocate a  $L$ -member list of  $(D \times 1)$  search symbol vectors,  $\mathcal{S}_{search}$ .
- 3) For  $\phi = 1 : \Phi$  do
  - a) Find the members of the  $F$  groups  $v_j$ ,  $j = 1, 2, \dots, F$ , by choosing  $|v_j|$  column indices  $u$ . If  $\phi = 1$  select the indices  $u$  according to decreasing value of  $\|\mathbf{h}[u]\|^2$  otherwise use random selection. Note that here  $\mathbf{h}[u]$  is the  $u$ -th column vector of  $\mathbf{H}$ .
  - b) For  $j = 1 : F$  do
    - i) Copy the  $L$  symbol vectors  $\hat{\mathbf{s}}^{(l)} \in \mathcal{S}$  into  $\mathcal{S}_{search}$ . Delete the column entries of the symbols in the  $j$ -th group and remove all redundant elements from  $\mathcal{S}_{search}$ . The resulting list for the  $j$ -th group is denoted  $\tilde{\mathcal{S}}_{search}$  and has size  $1 \leq K_j \leq L$ .
    - ii) For  $k = 1 : K_j$  do
      - A) Use (17) to calculate  $\mathbf{y}_j$  for the  $k$ -th list member. This is denoted as  $\mathbf{y}_j^{(k)}$ . Symbol values for the elements  $\mathbf{s}_i$  from all groups  $i$  are drawn from  $\hat{\mathbf{s}}^{(k)} \in \tilde{\mathcal{S}}_{search}$ .
      - B) Perform an exhaustive search over all symbol values of the  $j$ -th group  $v_j$  using (16) and (18). Update the lists  $\mathcal{S}$  and  $\mathcal{E}$ .
    - iii) Increase the group index  $j$ ,  $j = j + 1$ .
  - c) Increase the iteration index  $\phi$ ,  $\phi = \phi + 1$ .
- 4) Store the lists  $\mathcal{S}$  and  $\mathcal{E}$ . Output  $\mathcal{S}$ .

TABLE III

LGS-MUD SIMULATION PARAMETERS.

SEAIR Threshold <sup>3</sup>	2	List size $L$	$2D, 4D, 8D$
SSSER Threshold <sup>3</sup>	0.001	BLE group iterations $\Theta$	2
Receive Antennas $M$	4, 6	BLE iterations $Q_{BLE}$	2
Co-channel users $D$	$> M$	GLO group iterations $\Phi$	2
BLE group size $ \tau_j $	3, 4, 5	Overall iterations $Q$	2, 3
GLO group size $ v_j $	2, 3, 4	Symbol errors	80

a frequency-flat Rayleigh fading channel to a receiver with  $M$  uncorrelated antennas. The receiver has perfect CSI. The SNR at each receive antenna is defined as  $SNR = 10 \log_{10} (\sigma_s^2 / \sigma_z^2)$ , where  $\sigma_s^2$  denotes the average received signal power. Performance is evaluated in terms of the symbol error rate (SER) of the worst user. Simulations were stopped after one user experienced 80 symbol errors. Table III provides a summary of the LGS-MUD parameters used in simulation.

In Fig. 4, the SER performance of a  $M = 4$  antenna receiver is shown for different numbers of co-channel users. JML is optimum and provides the lower bound on LGS-MUD performance. We use  $Q = 2$  overall iterations for LGS-MUD. For list size  $L = 8D$  and group sizes  $(|\tau_j|, |v_j|) = (5, 3)$ , it achieves near JML performance for up to  $D = 8$  users (load factor  $f \leq 2$ ). Under heavy overload, i.e.  $f > 2$ , performance is slightly impaired due to limitations in list size, the number of iterations and the use of groups. It can further be seen that decreasing the list size to  $L = 4D$ , degrades the performance of LGS-MUD over all  $D$ . This is caused by an increased probability that the correct symbols are not

<sup>3</sup>See [8] for details.



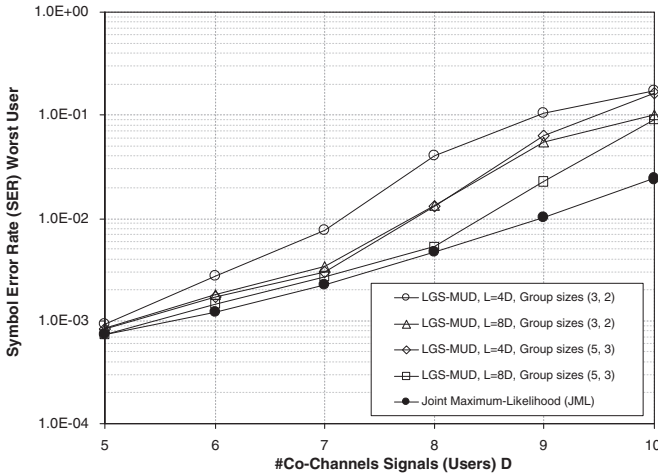


Fig. 4. SER performance versus the number of co-channel users for a  $M = 4$  antenna receiver at  $10dB$ .

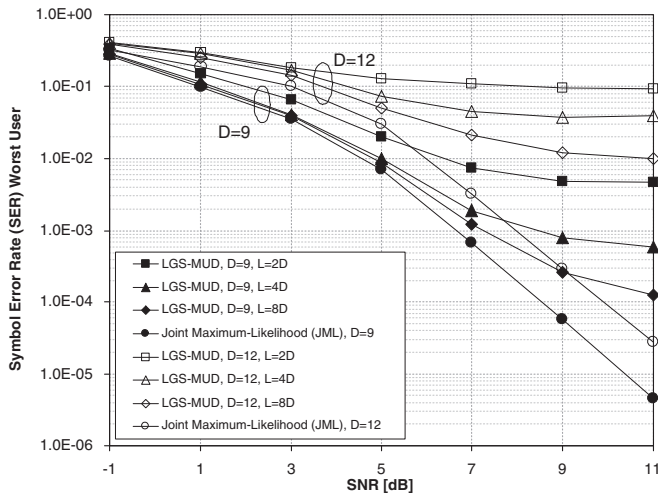


Fig. 5. SER performance versus SNR for a  $M = 6$  antenna receiver with  $D = 9$  and  $12$  co-channel users.

included in the lists  $\mathcal{S}_{br}[d]$  and  $\mathcal{S}$ . For smaller group sizes, i.e.  $(|\tau_j|, |v_j|) = (3, 2)$ , LGS-MUD loses performance if the receiver is heavily overloaded, i.e.  $D > 6$  users ( $f > 1.5$ ).

Fig. 5 illustrates SER performance for a  $M = 6$  antenna receiver for the cases of  $D = 9$  and  $12$  co-channel users at different SNR. LGS-MUD uses group sizes  $(|\tau_j|, |v_j|) = (5, 4)$ . We use  $Q = 2$  overall iterations for  $D = 9$  users and  $Q = 3$  for  $D = 12$ . JML provides the lower bound on SER performance. Performance curves for JML were obtained by simulation for  $SNR \leq 5dB$  and by computing the bound in [2] for  $SNR > 5dB$ .

The results show that LGS-MUD approximates JML in the low SNR region but introduces an error floor for small sizes of  $L$  (i.e.  $L = 2D$  and  $4D$ ) and under extreme overload (i.e.  $D = 12$ ). Near JML performance over a wide range of SNR and under heavy overload is achieved by increasing the list size  $L$ . This is at the cost of an increase in complexity.

The computational complexity of LGS-MUD is measured

TABLE IV  
COMPARISON OF COMPUTATIONAL COMPLEXITY FOR THE  $M = 6$  ANTENNA RECEIVER IN FIG. 5

Detector	Co-channel Users	
	D=9	D=12
LGS-MUD, $L = 2D$	$7.1E5$	$3.0E6$
LGS-MUD, $L = 4D$	$1.0E6$	$4.8E6$
LGS-MUD, $L = 8D$	$1.6E6$	$8.2E6$
JML MUD	$3.1E6$	$2.0E8$

in the number of required real squaring operations in the computation of the Euclidean distance. This is usually the most hardware intensive operation. Table IV provides results for JML and LGS-MUD. The choices of the group sizes  $(|\tau_j|, |v_j|)$  and the list size  $L$  determine the performance-complexity trade off. Small values of  $(|\tau_j|, |v_j|)$  and  $L$  result in lower complexity, while larger values achieve better performance. Table IV shows that for  $L = 8D$  LGS-MUD requires approximately half of the JML complexity for  $D = 9$  users and saves more than 95% for  $D = 12$  users. Complexity savings are more significant for higher numbers of receive antennas and transmitters.

## V. CONCLUSIONS

In this paper we propose LGS-MUD for the detection of multiple co-channel signals in an overloaded receiver. LGS-MUD employs parallel branch list estimators which independently search over subsets of the user symbols. If the size of a subset exceeds a certain maximum, this subset is split up into independent groups. We then search over the symbol groups thereby reducing the computational complexity. Simulation results show that LGS-MUD achieves good performance at lower complexity than JML MUD for up to 50% overload but degrades at higher overload factors.

## REFERENCES

- [1] S. Verdú, *Multuser Detection*, Cambridge: Cambridge University Press, 1998.
- [2] S. J. Grant and J. K. Cavers, "Performance enhancement through joint detection of cochannel signals using diversity arrays," *IEEE Trans. Commun.*, vol. 46, pp. 1038-1049, Aug. 1998.
- [3] B. W. Zirikoff, J. K. Cavers and S. Bavarian, "An iterative groupwise multiuser detector for overloaded MIMO applications," *IEEE Trans. on Wireless Commun.*, vol. 6, pp. 443-447, Feb. 2007.
- [4] J. Tsai, J. Hicks, and B. D. Woerner, "Joint MMSE beamforming with SIC for an overloaded array system," in *Proc. MILCOM*, pp. 1261-1265, Washington D.C., Oct. 2001.
- [5] J. E. Hicks, J. Tsai, J. H. Reed, W. H. Tranter, and B. D. Woerner, "Overloaded array processing with MMSE-SIC," in *Proc. VTC*, pp. 542-546, Birmingham, AL, May 2002.
- [6] J. Tsai and B. D. Woerner, "Performance of combined MMSE beamforming with parallel interference cancellation for overloaded OFDM-CDMA systems," in *Proc. MILCOM*, pp. 748-752, Anaheim, CA, Oct. 2002.
- [7] J. Hicks, S. Bayram, W. H. Tranter, R. J. Boyle and J. H. Reed, "Overloaded array processing with spatially reduced search joint detection," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 1584-1593, Aug. 2001.
- [8] M. Krause, D. P. Taylor, and P. A. Martin, "A unified approach to list-based multiuser detection in overloaded receivers," *EURASIP Journal on Wireless Communications and Networking*, vol. 2008.
- [9] J. G. Proakis, *Digital Communications*, 3rd ed., New York: McGraw-Hill, 1995.
- [10] E. W. Jang, J. Lee, H.-L. Lou and J. M. Cioffi, "Optimal combining schemes for MIMO systems with hybrid ARQ," in *Proc. ISIT*, pp. 2286-2290, Nice, France, June 2007.