A SPECTRAL CLUSTERING ALGORITHM FOR DECODING FAST TIME-VARYING BPSK MIMO CHANNELS

Steven Van Vaerenbergh, Emilio Estébanez and Ignacio Santamaría

Dept. of Communications Engineering, University of Cantabria, Spain
E-mail: {steven,emilioea,nacho}@gtas.dicom.unican.es

ABSTRACT

Clustering techniques for equalization have been proposed by a number of authors in the last decade. However, most of these approaches focus only on time-invariant single-input single-output (SISO) channels. In this paper we consider the case of fast time-varying multiple-input multiple-output (MIMO) channels. The varying nature of the mixing matrix poses new problems that cannot be solved by conventional clustering techniques. By introducing the time scale into the clustering process we are able to untangle the clusters, which in this way behave like intertwined threads. Then, a spectral clustering algorithm is applied. Finally, the identified clusters are assigned to the transmitted symbols using only a few pilots. The geometry of the transmitted constellation is exploited within the spectral clustering algorithm in order to reduce the number of clusters. As shown in the paper, the proposed procedure saves a considerable amount of pilot symbols in comparison to other recently proposed techniques.

1. INTRODUCTION

In recent years, multiple-input multiple-output (MIMO) wireless communication technology has gained considerable attention due to its potential to significantly increase spectral efficiency compared to traditional single-input single-output (SISO) technology. A number of computationally efficient algorithms for reliable symbol detection in time-invariant flat-fading MIMO systems have been developed, such as the V-BLAST architecture [1].

A direct application of these algorithms to time-varying channels is difficult, however, due to the need of perfect channel state information at the receiver side. A number of adaptive algorithms have been proposed to resolve this issue, such as an adaptive receiver based on the V-BLAST algorithm with a generalized decision feedback equalizer (GDFE) [2], a numerically more robust version of this algorithm [3], and a channel tracking algorithm based on decision-directed recursive least squares[4]. All of these are *supervised* equalization algorithms, requiring an initialization phase in which a number of pilot symbol slots are sent.

An approach for *blind* equalization in communication problems can be based on clustering techniques, which have mainly been applied in time-invariant SISO systems, using for instance radial basis function networks [5] or a cluster-based MLSE equalizer [6]. Some of these algorithms maintain their performance under slowly time-varying channels. Extensions to MIMO systems have also been proposed [7, 8].

Most algorithms for equalization of fast time-varying channels are supervised adaptive algorithms. However, thanks to the recently proposed advances in the field of machine learning, some efficient clustering techniques can also be extended to tackle this problem. In particular, it appears that the non-convex clustering capabilities of the recently proposed spectral clustering technique [9] can be applied to this end.

The main contributions of this work are twofold. First, it is shown that by incorporating the temporal variable in the clustering process, it is possible to untangle the different clusters representing different input symbol slots. The major limitation of the method is that the number of clusters increases exponentially as a function of the number of transmitting antennas, N_t. To relax this limitation, the second contribution of this paper is to exploit the geometry of the transmitted constellation within the spectral clustering algorithm in order to reduce the number of clusters. Once these different clusters have been identified, a simple decoding process is applied to relate each cluster with a symbol slot. It is shown that this can be done by sending as few as N_t pilot symbol slots.

This paper organized as follows: In Section 2 a detailed formulation of the problem is given. In Section 3 spectral clustering is introduced, followed by a description of how it can be applied directly to this equalization problem. Its performance can then be enhanced by exploiting the constellation's geometry, as presented in Section 4, and in Section 5 a decoding stage concludes the algorithm. Test results and comparisons with the GDFE algorithm are given in Section 6, followed by the conclusions of this work in Section 7.

2. PROBLEM FORMULATION

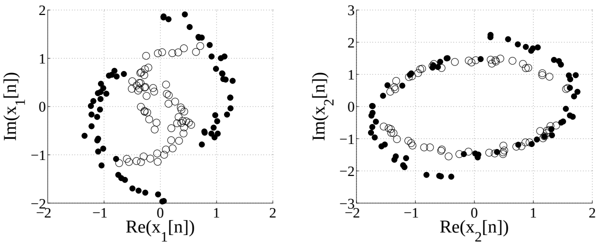
MIMO systems are used in wireless communications to enhance signal diversity, spectral efficiency, or both. In a typical MIMO flat-fading system with N_t transmit and N_r receive antennas, the $N_r \times 1$ received vector $\mathbf{x}[n]$ at time n is expressed as

$$\mathbf{x}[n] = \mathbf{H}[n]\mathbf{d}[n] + \mathbf{v}[n] \qquad (1)$$

where $\mathbf{H}[n]$ is the $N_r \times N_t$ channel matrix whose elements represent independent flat-fading SISO channels, $\mathbf{d}[n]$ contains the N_t (in general, complex) symbols transmitted by the N_t antennas at time n, and $\mathbf{v}[n]$ represents both spatially and temporally white complex zero-mean Gaussian noise.

In MIMO systems with *block fading* channels, variations of the channel during the transmission of one block of sym-

This work was supported by MEC (Ministerio de Educación y Ciencia) under grant TEC2004-06451-C05-02 and FPU grant AP2005-5366.



(a) (b)

Fig. 1. (a) and (b): Scatter plots of the data received by the two antennas of a 2×2 BPSK MIMO system with fast time-varying channels. The points corresponding to the symbols $[+1,+1]^T$ and $[-1,-1]^T$ are represented by circles and the points corresponding to $[+1,-1]^T$ and $[-1,+1]^T$ by black dots, emphasizing the symmetry of the used constellation. Note that no such information is available at receiver side.

bols are so small that they can be ignored. Hence the channel matrix $\mathbf{H}[n] = \mathbf{H}$ is considered constant during transmission of one block of symbols. This is not the case for MIMO systems with *fast time-varying* channels, where the channel matrix changes from symbol to symbol due to the Doppler spread caused by the movement of the transmitter and/or receiver. In time-varying MIMO systems, depending on the Doppler spread, the channel matrices $\mathbf{H}[n]$ are temporally correlated. The variations can be modeled for instance by the Clarke-Gans model [10] which states that if a vertical $\lambda/4$ antenna with uniform power distribution is used to transmit a single tone, the received spectrum is

$$S_{E_z}(f) = \frac{1.5}{\pi f_m \sqrt{1 - \left(\frac{f-f_c}{f_m}\right)^2}}, \qquad (2)$$

where f_c and f_m are the carrier frequency and the maximum Doppler shift, respectively.

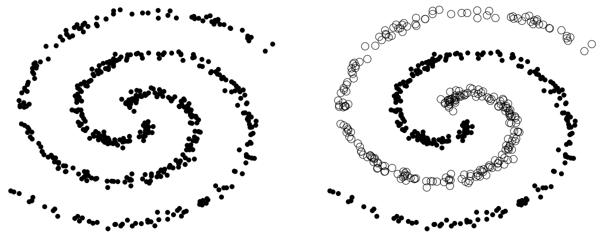
The proposed method aims to estimate the symbols $\mathbf{d}[n]$ given the received data points $\mathbf{x}[n]$. This problem is illustrated in Fig. 1, which shows typical scatter plots of the complex data $\mathbf{x}_1[n]$ and $\mathbf{x}_2[n]$ received by the two antennas in a time-varying 2×2 MIMO system with binary phase-shift keying (BPSK) modulation, for which the basic constellation points are $d \in \{+1,-1\}$. Classical clustering algorithms that operate directly on the data of these scatter plots will fail due to overlapping of the clusters. In the next section we propose a solution to these problems that combines a spectral clustering approach with the incorporation of the temporal dimension into the clustering process.

3. CLUSTERING FOR TIME-VARYING CHANNELS

3.1 Spectral clustering

Spectral clustering [9] is a recently proposed successful method rooted in graph theory [11], capable of clustering data based on point-to-point similarities. For most problems, "similarity" is measured as the distance between data points, defining clusters as connected zones of points and making it possible to easily cluster non-convex data sets, as illustrated by Fig. 2.

The similarity between two points $\mathbf{x}[i]$ and $\mathbf{x}[j]$ is mea-



(a) (b)

Fig. 2. (a) Two sets of intertwined data points, difficult or impossible to cluster with conventional clustering algorithms. (b) Spectral clustering easily divides the points in two separate groups, based on the principle that two points should be in the same group if they are close to each other.

sured through a kernel function $\kappa(.)$ such as

$$\kappa(\mathbf{x}[i],\mathbf{x}[j]) = \exp\left(-\frac{d^2(\mathbf{x}[i],\mathbf{x}[j])}{\sigma^2}\right) \qquad (3)$$

where $d(\mathbf{x}[i],\mathbf{x}[j])$ is some distance measure between points $\mathbf{x}[i]$ and $\mathbf{x}[j]$ and σ is the kernel size. This kernel function is almost 1 for points that are close to each other, and lowers as the distance rises.

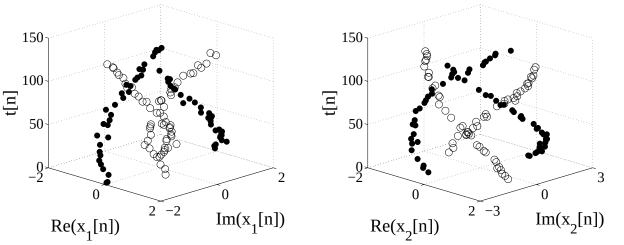
Given a set of N points $\{\mathbf{x}[1],\mathbf{x}[2],\ldots,\mathbf{x}[N]\}$, a similarity matrix (also called "affinity" or *kernel* matrix) can be defined as $A_{ij} = \kappa(\mathbf{x}[i],\mathbf{x}[j])$. Clustering is performed by analyzing the spectrum of that matrix. One of the most successful spectral clustering algorithms is the Ng-Jordan-Weiss (NJW) algorithm, introduced in [9]. It can be summarized as follows:

1. Calculate the affinity matrix A using (3), and set $A_{ii} = 0$ for $i = 1,\ldots,N$.
2. Obtain $L = D^{-1/2}AD^{-1/2}$, where D is a diagonal matrix with $D_{ii} = \sum_{j=1}^{N} A_{ij}$. This normalization will assure that all clusters have more or less equal size.
3. Form the matrix $V = [v_1, v_2, \ldots, v_k]$ where $v_1, v_2, \ldots, v_k$ are the k largest eigenvectors of L and k is the number of subsets to retrieve.
4. Treat the rows of V as points in $\mathbb{R}^k$, and normalize them to unit length. These points correspond to the original points $\mathbf{x}[i]$ but form compact clusters now. Cluster them with an algorithm such as k-means.
5. Assign the original point $\mathbf{x}[i]$ to cluster j if and only if row i of the matrix V was assigned to cluster j.

3.2 Fine-tuning Spectral Clustering

The choice of the kernel size σ in (3) has a high impact on the clustering quality. It is a measure of when two points are considered similar, and should be of the same order of the distance between similar points. Some rules of thumb have been proposed to set a value for σ, whereas in other cases this value is set manually.

When the data contains clusters with different local statistics, there may not be a single value of σ that works well for all the data. In [12] a "local" scaling parameter σ_i is proposed instead of this global parameter. It allows self-tuning of the point-to-point distances by studying the local statistics of the neightboring points of every point $\mathbf{x}_i$. This leads to the



(a) (b)

Fig. 3. Scatter plots of the data of Fig. 1 to which the temporal dimension was added. Threads of data points are now distinguishable in both figures.

following extension of (3):

$$\tilde{\kappa}(\mathbf{x}[i],\mathbf{x}[j]) = \exp\left(-\frac{d^2(\mathbf{x}[i],\mathbf{x}[j])}{\sigma_i\sigma_j}\right). \quad (4)$$

If σ_i is chosen as

$$\sigma_i = d(\mathbf{x}_i,\mathbf{x}_L) \quad (5)$$

where $\mathbf{x}_L$ is the L'th nearest neighbor of point $\mathbf{x}_i$, then spectral clustering will group together all points with their closest neighbors that have similar σ_i. The selection of L only depends on the data dimension of the embedding space.

3.3 Incorporating the temporal dimension into the clustering problem

The received data $\mathbf{x}[n]$ in a fast time-varying MIMO system can be preprocessed for spectral clustering by simply adding the temporal dimension. If the temporal index is $t[n]$, the combined vector of data points and temporal indices, $\mathbf{x}^+[n] = [\mathbf{x}[n]^T, t[n]^T]^T$, is an $(N_r+1)\times 1$ complex vector. When this extra dimension is added to the scatter plots of Fig. 1, threads appear due to the temporal correlation between consecutive channel matrices (see Fig. 3). Given the non-convex clustering capabilities of spectral clustering algorithms, they should be capable of retrieving the different threads from $\mathbf{x}^+[n]$.

The performance of a suitable spectral clustering algorithm depends mainly on two factors. In the first place, the number of data points N in one block must be larger than the number of clusters (a rule of thumb is to have at least 10 samples per cluster). Since spectral clustering is a computationally costly procedure, the number of clusters to detect should be limited. For constellations with alphabet size M (the *cardinality*) this number of clusters is M^{N_t}, which is exponential in N_t. Taking into account that most commercial MIMO systems use up to $N_t = 4$ transmit antennas, we will only treat BPSK systems ($M = 2$) in this work. Extensions to more complex modulations will be considered for future investigation.

In the second place, clusters should be well connected, i.e., the distance between neighboring points of the same thread should not be larger than the distance between points of different threads. This requires a rescaling of the temporal dimension to match the scale of the spatial dimensions, for instance $t[n] = n/256$, $n = 0,\ldots,255$ for blocks of 256 symbols. Moreover, this means that if a symbol is not sent during a considerable time, a thread might be incorrectly identified as two separate threads. However, as will be shown in the next section, both difficulties can be reduced by using information derived from the geometric properties of the constellation.

4. EXPLOITING THE CONSTELLATION GEOMETRY

In this section we show that the geometrical symmetries of the transmitted constellation can be used to reduce the number of clusters.

4.1 Example: 2×2 BPSK MIMO

In the noiseless case ($\mathbf{v}[n] = \mathbf{0}$), Eq. (1) can be written as

$$\mathbf{x}[n] = \mathbf{H}[n]\mathbf{d}[n] \quad (6)$$

For a 2×2 BPSK MIMO system, there will be 4 symbol clusters to detect in the data $\mathbf{x}[n]$, corresponding to the transmitted symbol vectors $[+1,+1]$, $[+1,-1]$, $[-1,-1]$ and $[-1,-1]$. In Fig. 1 and Fig. 3 we can observe that for any cluster following a certain trajectory, there is always another cluster following a trajectory symmetric with respect to the origin. This observation is confirmed by (6): since a BPSK system can emit both $\mathbf{d}[n]$ and $-\mathbf{d}[n]$, the data point $\mathbf{x}[n]$ as well as its opposite $-\mathbf{x}[n]$ can be received. These data points lie in clusters that follow symmetric trajectories. This property can be exploited to improve the spectral clustering stage, by first grouping together the data points that follow symmetric trajectories, as will be shown in Section 4.2. Although this work is limited to BPSK systems, the extension of the described property and procedure to other M-PSK constellations is straightforward.

4.2 Clustering procedure for $N_t \times N_r$ BPSK MIMO systems

The geometrical property indicated in the previous section is not limited to 2×2 systems. In a general $N_t\times N_r$ BPSK MIMO system with fast time-varying channels, for any cluster following a certain trajectory in time, there is always another cluster following the symmetric trajectory. Combining the data of two such clusters might provide a more robust clustering problem. This observation leads to the following two-phase algorithm: In the first phase, groups of symmetric clusters are detected. One clustering problem needs to be solved here to find 2^{N_t-1} clusters. In the second phase, each group of symmetric clusters is separated into two different clusters, representing N_t independent problems.

4.2.1 Phase 1: Grouping of symmetric clusters.

Spectral clustering is extended to find clusters consisting of two symmetric clusters at a time. Consider the distance measure

$$d(\mathbf{x}[i],\mathbf{x}[j]) = \min\left(\left|\mathbf{x}^+[i]-\mathbf{x}^+[j]\right|^2, \left|\mathbf{x}^+[i]-\mathbf{x}^-[j]\right|^2\right), \quad (7)$$

where $\mathbf{x}^-[j] = [-\mathbf{x}[j]^T, t[j]^T]^T$ is the point symmetric to $\mathbf{x}^+[j]$. As can be seen, this measure is small in two cases: Firstly for points that are very close to each other, and secondly for points that are very close to opposite of each other. If the the modified Gaussian kernel (4) is used with this distance measure for spectral clustering, neighboring points as well as opposite points will be grouped together, leading to

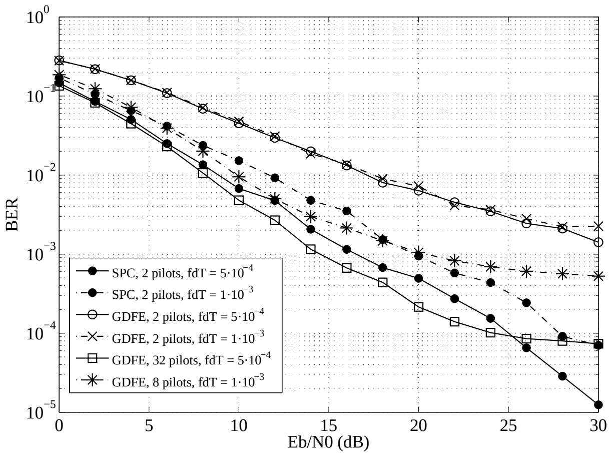


Fig. 4. BER curves for a 2×2 BPSK MIMO system.

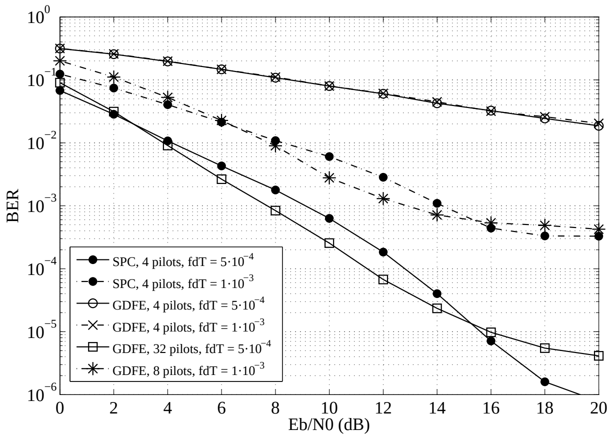


Fig. 5. BER curves for a 4×4 BPSK MIMO system.

2^{N_t-1} clusters. This first phase avoids the incorrect clustering that might occur when some of the threads have a low number of data points, by combining the information of symmetric threads.

4.2.2 Phase 2: Retrieving the individual clusters.

After having identified the 2^{N_t-1} groups of symmetric clusters, the two individual threads for each group need to be retrieved. Since only two clusters have to be retrieved in each group in this phase, the inter-cluster distances will be much larger than the distances between neighboring points of the same cluster and therefore performing spectral clustering using conventional Euclidian distances on each group is sufficient to obtain the 2^{N_t} final symbol clusters.

5. SYMBOL DECODING

Once the symbol clusters have been successfully retrieved, the original time-varying problem has been reduced to a simpler decoding problem, which is the only supervised part of the proposed algorithm. Symbols need to be assigned to each cluster, and to this end a small number of pilot symbol slots $\mathbf{d}[i]$ is transmitted at the start of the symbol block, specifically N_t. Not that these pilot symbol slots are not needed for the clustering process.

Defining the matrix of pilot symbols $\mathbf{D}_p = [\mathbf{d}[1], \mathbf{d}[2], \ldots, \mathbf{d}[N_t]]$ and the matrix of corresponding received data $\mathbf{X}_p = [\mathbf{x}[1], \mathbf{x}[2], \ldots, \mathbf{x}[N_t]]$, an approximation of the initial channel matrix $\mathbf{H}$ can be obtained as

$$\hat{\mathbf{H}} = \mathbf{X}_p \mathbf{D}_p^{-1}. \tag{8}$$

The algorithm concludes by assigning the symbol slot $\mathbf{d}$ to the cluster whose first data point in time is closest to the vector $\hat{\mathbf{H}}\mathbf{d}$.

6. TEST RESULTS AND COMPARISON

In this section the performance of the proposed spectral clustering is compared to that of the GDFE algorithm.

6.1 An overview of the GDFE algorithm

The generalized decision feedback equalizer (GDFE) algorithm from [2] is an adaptive algorithm for decoding fast time-varying MIMO systems, based on the V-BLAST architecture. For each time instant, the symbols are successively detected and canceled from the received data vector via decision feedback filtering. The filter tap weights and symbol detection order are updated using an RLS-based time- and order-update algorithm. Its complexity is $O(N_t^3)$ but it provides some savings compared to V-BLAST. During its training period, it needs to send a number of pilot symbols to initialize the algorithm.

6.2 Comparison

A number of simulations were carried out to illustrate the performance of the proposed algorithm. The following parameters were assumed: a BPSK signal was used, the channels were independent Rayleigh flat-fading and the temporal variation of each channel between a transmit and receive antenna pair was based on the Clarke-Gans model [10]. The symbols $\mathbf{d}[n]$ were grouped into frames consisting of $N = 256$ slots. In the first setup a MIMO system with $N_t = N_r = 2$ antennas was used, in the second setup $N_t = N_r = 4$, and in the last setup $N_t = 2$ and $N_r = 4$. In all cases the normalized Doppler frequencies $f_d T = 5 \cdot 10^{-4}$ and $f_d T = 10^{-3}$ were considered, where $f_d = f_c \cdot v/c$ with receiver velocity v and c is the speed of light. For a GSM symbol period $T = 3.7 \cdot 10^{-6}$ seconds and a carrier frequency $f_c = 900$ MHz, these normalized Doppler frequencies correspond to receivers moving at 162 km/h and 324 km/h, respectively. For a carrier frequency $f_c = 1800$ MHz, the normalized Doppler frequencies correspond to 81 km/h and 162 km/h, respectively.

The bit error rate (BER) curves of two algorithms were compared. In the first place the proposed spectral clustering method (referred to as SPC) was tested, in which fine-tuning spectral clustering was applied with $L = 5$. The number of pilot symbol slots used was fixed as N_t for this method. In the second place, the GDFE algorithm from [2] was applied, with forgetting factor $\lambda = 0.95$.

The BER against Eb/N_0 for the 2×2, a 4×4 and a 2×4 setup are shown in Fig. 4, Fig. 5 and Fig. 6, respectively. Because the GDFE algorithm is essentially a supervised method, it requires transmitting more pilot symbols. Therefore, apart from its BER curves for N_t pilots (both of which coincide, in all figures), a second set of BER curves was also displayed for a higher number of pilots, to achieve

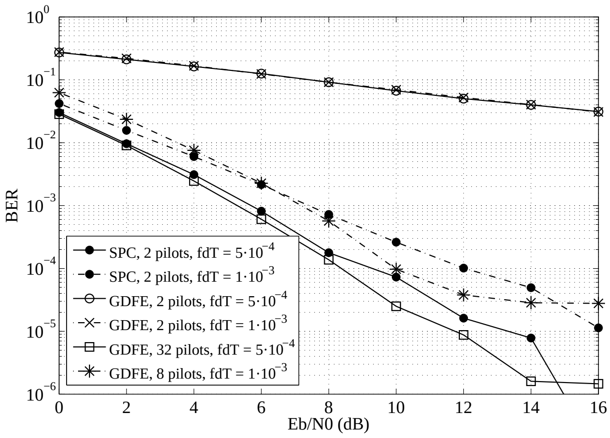


Fig. 6. BER curves for a 2×4 BPSK MIMO system.

the same performance as the spectral clustering algorithm. For the three cases, the GDFE algorithm needs 32 pilot symbols to achieve similar performance as the presented method when $f_d T = 5 \cdot 10^{-4}$, and 8 pilots when $f_d T = 1 \cdot 10^{-3}$. Comparing Fig. 4 and Fig. 6 shows that the presented algorithm performs significantly better when receiver antennas are added. This can be explained by observing that the clusters will be more separated in space when dimensions are added to the data points.

In cases where only a few pilot symbols can be sent, the spectral clustering algorithm obtains superior performance for the tested MIMO systems. However, it requires the calculation of the eigenvectors of its affinity matrix, which generally requires $O(N^3)$ operations. In most cases this can be lowered to $O(N^2)$ [13] taking into account that the affinity matrix is symmetric and can be approximated by a tridiagonal matrix.

This analysis suggests that spectral clustering could be used as an initialization for the GDFE or any other supervised algorithm. Specifically, given only N_t pilot symbol slots it can estimate a short symbol vector sequence which can be used as a pilot sequence for a (computationally more efficient) supervised algorithm.

7. CONCLUSIONS

We presented a novel clustering algorithm that is capable of decoding fast flat-fading time-varying BPSK MIMO channels. This algorithm operates on the received data to which a temporal dimension is added, and it exploits constellation geometry to retrieve symbol threads from it. The only supervised part of the presented clustering method is the final decoding phase.

For moderate numbers of transmitting antennas, the presented method only needs transmission of very few pilot symbols to achieve superior BER performance compared to the tested supervised adaptive algorithm. Its results can further be improved by adding more receiver antennas to the MIMO system.

Future research lines include the extension of this algorithm to communication systems with more clusters, which are encountered if more transmit antennas are used or if the constellation alphabet is extended. To that end other system characteristics could be exploited, apart from the constellation geometry, such as space-time coding and in particular orthogonal block coding, for instance the popular Alamouti coding.

REFERENCES

[1] G. J. Foschini, G. D. Golden, R. A. Valenzuela, and P. W. Wolniansky, "Simplified processing for high spectral efficiency wireless communication employing multi-element arrays," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 9, pp. 1841–1852, 1999.

[2] J. Choi, H. Yu, and Y.H. Lee, "Adaptive MIMO decision feedback equalization for receivers with time-varying channels," *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4295–4303, Nov. 2005.

[3] A. A. Rontogiannis, V. Kekatos, and K. Berberidis, "A square-root adaptive V-BLAST algorithm for fast time-varying MIMO channels," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 265–268, May 2006.

[4] E. Karami and M. Shiva, "Decision-directed recursive least squares MIMO channels tracking," *EURASIP Journal on Wireless Communications and Networking*, vol. 2006, pp. Article ID 43275, 10 pages, 2006.

[5] S. Chen, B. Mulgrew, and P.M. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," *IEEE Transactions on Neural Networks*, vol. 4, no. 4, pp. 570–590, July 1993.

[6] Y. Kopsinis and S. Theodoridis, "A novel cluster based MLSE equalizer for M-PAM signaling schemes," *Signal Processing*, vol. 83, no. 9, pp. 1905–1918, 2003.

[7] K. I. Diamantaras, "A clustering approach for the blind separation of multiple finite alphabet sequences from a single linear mixture," *Signal Processing*, vol. 86, no. 4, pp. 877–891, 2006.

[8] K. I. Diamantaras and T. Papadimitriou, "Blind deconvolution of multi-input single-output systems with binary sources," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3270–3731, October 2006.

[9] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds., Cambridge, MA, 2002, pp. 849–856, MIT Press.

[10] T. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall PTR, 2001.

[11] F. Chung, *Spectral Graph Theory*, Number 92 in CBMS Regional Conference Series in Mathematics. American Mathematical Society, 1997.

[12] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou, Eds., Cambridge, MA, 2005, pp. 1601–1608, MIT Press.

[13] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, October 1996.