Blind Estimation of Output Labels of SIMO Channels Based On a Novel Clustering Algorithm

F. Daneshgaran, *Member, IEEE,* M. Mondin, *Member, IEEE,* F. Dovis, *Member, IEEE,* and M. S. Roden, *Member, IEEE*

***Abstract*—This paper addresses the problem of data detection for digital communications employing space diversity reception, where the system model contains a single-input multiple-output (SIMO) vector channel. The received vector corrupted by additive white Gaussian noise (AWGN) is modeled as the noisy output of a finite state vector Markov source. Subsequently, a discretely valued basis of minimal dimensionality is identified for the input space. Estimation of the output labels associated with this basis allows labeling of the state transition diagram. For this purpose, certain identifiable characteristics of the output sequences of the Markov source are used to classify its states and generate an initial codebook for a vector quantizer used to restore the output levels.**

***Index Terms*—Blind equalization, channel estimation, clustering, fading.**

I. NTRODUCTION

SPACE diversity reception is known to be an excellent mean of combatting the detrimental effects of the mobile radio channel in data transmission [1]. It is of great interest to investigate the fruitfulness of this approach in blind sequence detection (BSD) [2], [3]. Several key features of the mobile radio channel that are of relevance in the paper are: 1) for typical transmission rates and symbol periods the length of the intersymbol interference (ISI) at the channel output is relatively short (e.g., less than six symbol periods); 2) the channel is rapidly time varying; and 3) the channel may occasionally produce a null in the signal spectrum.

As pointed out by Sato in [2], the mobile radio channel should not necessarily be equalized, rather, it should be identified. This is evident in light of feature 3) noted above. Given feature 1), maximum-likelihood BSD is indeed plausible for such channels. Feature 2) suggests that if channel estimation is to be of any use, it must be accomplished in a short time. Finally, note that, in general it is not even necessary to identify the channel impulse response coefficients. All we really need are to identify the noiseless channel output *labels* and label the state transition diagram of the overall system [4].

System Model

Consider digital transmission over a frequency selective slowly fading channel whereby several receiving antennas are used. We assume that the received signal in different channels undergo uncorrelated fading. We assume that the carrier recovery loop generates a coherent carrier used for the down conversion of the RF signal. Samples of the matched filters outputs taken at the symbol rate generate the matrix of sufficient statistics whereby each column is a vector whose components are the samples of the filter outputs at the receiver. The overall complex baseband equivalent model of the communication system can be modeled as a single input multiple output system. As an example, Fig. 1 depicts a one-input–two-output channel that will be used throughout the paper.

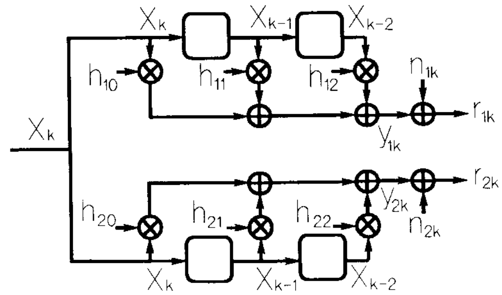

Fig. 1. Example of a one input two output vector channel. The state at time k is defined by $s_k = (x_{k-2}, x_{k-1})$.

In Fig. 1, the noise components n_{1k}, n_{2k} are uncorrelated white Gaussian and each component of the received vector $\vec{r_k} = [r_{1k}, r_{2k}]'$ has an associated signal-to-noise ratio (SNR) that we assume to be constant and equal to 10 dB. For our example, both channels have a memory of two. The upper channel (CH-1) has the impulse response coefficients $[h_{10}, h_{11}, h_{12}] = [0.3, -0.54, 0.23]$, and the lower channel (CH-2) has the impulse response coefficients $[h_{20}, h_{21}, h_{22}] = [-0.57, 0.24, 0.33]$ (these coefficients were chosen arbitrarily). With this setup, the overall system resembles a vector Markov source whose state transition diagram is depicted in Fig 2.

Simultaneous maximum-likelihood estimation of the input vector and the channel coefficients reduces to one of obtaining the values of this pair that minimizes the sum of the Euclidian norms between the received and the estimated output vectors. This can be done using a standard Viterbi algorithm provided the trellis diagram of the underlying vector Markov source can be *labeled* with the noiseless channel outputs. Of course, such noiseless labels are not directly available and must themselves be estimated from the vector channel outputs. In [4], Seshardi proposes the technique of labeling the trellis diagram of the

Manuscript received February 5, 1998. The associate editor coordinating the review of this letter and approving it for publication was Prof. A. Haimovich.

F. Daneshgaran and M. S. Roden are with the Electrical and Computer Engineering Department, California State University, Los Angeles, Los Angeles, CA 90032 USA (e-mail: fdanesh@calstatela.edu).

M. Mondin and F. Dovis are with the Dipartimento di Elettronica, Politecnico di Torino, Turin, Italy (e-mail: mondin@polito.it).

Publisher Item Identifier S 1089-7798(98)09318-1.

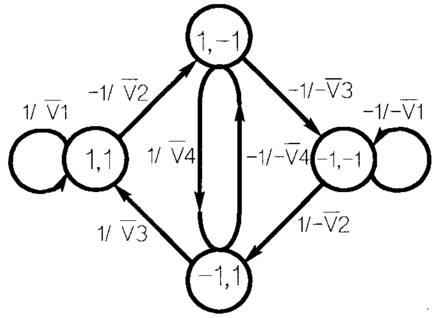

Fig. 2. The state transition diagram of the overall system of Fig. 1.

Markov source in all the permissible ways. Unfortunately, with this technique the number of trellises that must be searched grows in a factorial manner with the size of the branch space. In this letter, we propose a novel trellis labeling technique that avoids this problem. The problem of identification of the noiseless channel output labels is equivalent to designing a vector quantizer (VQ) from a training set corresponding to the observed noisy output vectors. For the trellis labeling problem, we propose a novel classification of the states of the vector Markov source based on the properties of its output sequences.

II. Codebook Initialization and Data Clustering

The fact that the input at a given symbol period comes from a constellation carved out of a lattice, induces a rich algebraic structure on the output labels. For a channel memory of size ν, the components of the output vectors are of the type $\sum_{k=0}^{\nu} x_{n-k}h_{ik}$ where n is the time index, and the value of x_l is assumed to be restricted to a given subset of a lattice (i.e., the signal constellation). Looking at the input in blocks of size ν, the output labels are obtained via a generally one-to-one mapping from the input ν-tuples. Identification of a discrete and possibly *redundant* basis for the input space allows us to identify a set of output labels, from which all the other output labels may be derived via simple algebraic operations.

A standard technique for vector quantizer design is that of the generalized Lloyd algorithm, also referred to as the k-means or the LBG algorithm [5]. A typical distortion measure for such designs is the Euclidian distance, which is indeed appropriate in our problem. In our VQ designs we compute the average distortion for simplicity. The LBG algorithm starts with an initial codebook and at each iteration two steps are performed: 1) the received vectors are classified using the nearest neighbor (NN) search rule relative to the distortion measure employed and 2) after all the members of the training set have been classified, the centroid of each class (or cell) is generated and used to form a new codebook that can be used in the next iteration.

Two basic problems associated with the VQ design are the selection of a suitable initial codebook, and the empty cell problem. For the blind sequence detection problem we generally have relatively high noise levels *and* a small training set dictated by the requirement imposed by a rapidly time varying channel. For this problem, we would like to use as many of the structural properties of the output vectors as possible to eliminate or alleviate the problems noted above.

TABLE I
Length Two and Three Paths in the State Transition Diagram of Fig. 2 and Their Respective Properties

Path Length L	Starting State s_k	Output Sequence	Property
2	$(\pm 1, \pm 1)$	$\pm\vec{v}_1, \pm\vec{v}_1$	p1=0
	$(\pm 1, \mp 1)$	$\pm\vec{v}_4, \mp\vec{v}_4$	p2=0
3	$(\pm 1, \pm 1)$	$\pm\vec{v}_1, \pm\vec{v}_1, \pm\vec{v}_1$	p3=0
		$\pm\vec{v}_2, \mp\vec{v}_3, \mp\vec{v}_2$	p4=0
	$(\pm 1, \mp 1)$	$\pm\vec{v}_4, \mp\vec{v}_4, \pm\vec{v}_4$	p5=0
		$\mp\vec{v}_3, \mp\vec{v}_2, \pm\vec{v}_3$	p4=0

Among the well-known initialization techniques, there are the random coding, pruning, pairwise nearest neighbor, product code, and splitting techniques [5]. None of these methods explicitly use the dynamical structure of the vectors generated at the output of the channels. In this paper, we propose to use the dynamical structure of the output for both the VQ codebook initialization and labeling of the trellis diagram of the vector Markov channel. We demonstrate this for the example used in the paper in order to present the key ideas quickly. For this purpose, consider the labeled state transition diagram of the Markov source depicted in Fig. 2. We assume that the channel input is equiprobable binary and real. The state of the system at time k is the binary two tuple $s_k = (x_{k-2}, x_{k-1})$. The branches of Fig. 2 are labeled by the input that causes the transition, and the corresponding output vector denoted $\pm 1/\vec{vj}$. The following labeling is used: $\vec{v}_1 = [h_{10} + h_{11} + h_{12}, h_{20} + h_{21} + h_{22}]'$, $\vec{v}_2 = [-h_{10} + h_{11} + h_{12}, -h_{20} + h_{21} + h_{22}]'$, $\vec{v}_3 = [h_{10} + h_{11} - h_{12}, h_{20} + h_{21} - h_{22}]'$, and $\vec{v}_4 = [h_{10} - h_{11} + h_{12}, h_{20} - h_{21} + h_{22}]'$. Note that $\vec{v_4} = (\vec{v_1} - \vec{v_2} - \vec{v_3})$.

The significant observations that can be made with regards to Fig. 2 are: 1) there are really only *three* distinct branch labels $\vec{v}_1, \vec{v}_2, \vec{v}_3$, since all the other branch labels can be obtained from these via simple *algebraic* operations and 2) certain state transitions of this Markov source have *identifiable properties* that can be checked from the observation of the *output sequences* only.

To determine these properties, consider any starting state and the corresponding paths of different lengths through the state transition diagram starting from paths of length two. Subsequently, we mark the output label sequences that have specific algebraic properties. For instance, consider the length two and length three paths. Table I lists the paths having specific properties that can be checked.

Index k in Table I is the starting time index of the noted sequences, and the properties $p1$–$p5$ are defined as: $p1 = ||\vec{y}_{k+1} - \vec{y}_k||$, $p2 = ||\vec{y}_{k+1} + \vec{y}_k||$, $p3 = ||\vec{y}_{k+2} - \vec{y}_{k+1}|| + ||\vec{y}_{k+1} - \vec{y}_k|| + ||\vec{y}_{k+2} - \vec{y}_k||$, $p4 = ||\vec{y}_{k+2} + \vec{y}_k||$, and $p5 = ||\vec{y}_{k+2} + \vec{y}_{k+1}|| + ||\vec{y}_{k+1} + \vec{y}_k|| + ||\vec{y}_{k+2} - \vec{y}_k||$.

The entries of Table I suggests that we may be able to detect certain state transitions and consequently *certain labels* of the state transition diagram by using the noted properties as tests. Certain tests may provide *redundant information* that may in fact be used to discard or accept certain labels obtained in this manner. Note that there is always a sign ambiguity in the labels of the state transition diagram even when all are detected correctly.

In the presence of noise, the calculation of the norms suggested by various tests for recognizable properties in Table I produce nonzero results. However, the maximum-likelihood detection rule suggests that we may use the minimum norm criterion to rank the outcomes of the tests conducted at the output.

Codebook Initialization

In our simulations, we used an observation length of 100 symbols and the following codebook initialization technique was used: 1) properties $p1$, $p4$, $p2$ of Table I were used as tests of the output channel symbols and the corresponding vector of test results were generated and 2) for each test vector, the index producing the minimum element was used to identify the initial codebook vectors. In particular, test for $p1$ was used to obtain an initial estimate of $\vec{v_1}$ (the sign was assigned arbitrarily). Test for $p4$ was used to obtain the initial estimates of $\vec{v_2}$ and $\vec{v_3}$; here, we not only have a sign ambiguity, but we also have an order ambiguity. Test $p2$ was used as a check on the results of the previous two tests while simultaneously providing an initial estimate of $\vec{v_4}$. Subsequently, vectors $\vec{v_1}, -\vec{v_1}, \vec{v_2}, -\vec{v_2}, \cdots$ were used to generate the initial VQ codebook C_0. The initial VQ codebook obtained from a sample simulation run is $\vec{v}_1 = [-0.0673, -0.1548]'$, $\vec{v}_2 = [-0.4440, 0.8662]'$, $\vec{v}_3 = [-0.4174, -0.4979]'$, and $\vec{v}_4 = [1.1186, -0.5705]'$.

Data Clustering

Using the initial codebook noted above, the partition of the 100 received vectors generates eight cells. The cells by design contain different elements of the training set. The following merges are performed at each iteration: 1) the cells containing codevectors of C_0 differing in their signs are pairwise merged. This process generates four cells and 2) the centroids of the four cells obtained from step 1) are obtained.

As noted earlier, there are really only three codevectors whose knowledge allows complete determination of all the output labels. Hence, one more merging step is possible that would reduce the total number of cells to three. More specifically $\vec{v_4} = \vec{v_1} - \vec{v_2} - \vec{v_3}$. To eliminate the *relative* sign ambiguity and the order ambiguity in $\vec{v_2}$ and $\vec{v_3}$, the centroids of the respective cells were examined. Subsequently, the elements of the cell associated with vector $\vec{v_4}$ (with the centroids of other cells taken to be the true values of $\vec{v_1}, \vec{v_2}, \vec{v_3}$) were distributed among the remaining cells depending on which assignment produced the least distortion. In almost all of our simulation runs, three iterations of the LBG algorithm with this clustering routine were sufficient for estimation of the noiseless output labels. The results of a sample simulation run are depicted in Fig. 3.

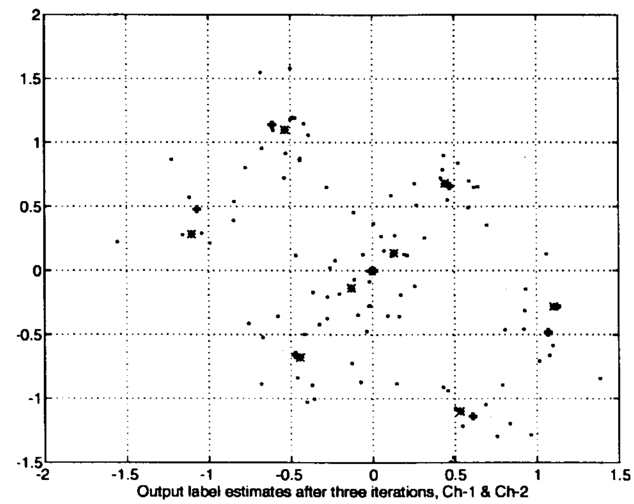

Fig. 3. The observed noisy outputs (marked by "·"), estimated noiseless outputs (marked by "$*$"), and the actual noiseless outputs (marked by "$+$") of the vector quantizer after three iterations.

Trellis Labeling

In the process of data clustering and initial codebook design, we explicitly labeled the state transition diagram. The problem is that the resulting labeling may in fact be incorrect. If we treat the vector quantizer so designed as a nonlinearity placed at the receiver filters outputs to restore the noiseless output vectors, we expect the output of this nonlinearity to maintain the dynamical properties of the overall system. In particular, the sequence of observed outputs of this nonlinearity *should* represent a valid trajectory in the trellis diagram of the underlying Markov source. If in fact, excessive spurious and disallowed state transitions are observed at the VQ output, most likely the trellis labeling is incorrect. In our simulations thus far, we have not encountered a labeling problem.

III. Conclusions

In this paper, we have presented a novel clustering algorithm for blind estimation of the noiseless channel output labels in a SIMO communication system model. The proposed approach utilizes the dynamic properties of the output sequences of the SIMO system which may be viewed as a vector Markov source whose outputs are corrupted by noise.

References

[1] J. G. Proakis, *Digital Communications.* 3rd ed. New York: McGraw-Hill, 1995.
[2] Y. Sato, "A blind sequence detection and its application to digital mobile communication," *IEEE Trans. J. Select. Area Commun.*, vol. 13, pp. 49–58, Jan. 1995.
[3] G. D. Forney, "Maximum likelihood sequence estimation of digital sequence in the presence of intersymbol interference," *IEEE Trans. Inform. Theory* vol. IT-18, pp. 363–378, May 1972.
[4] N. Seshardi, "Joint data and channel estimation using trellis search techniques," *IEEE Trans. Commun.*, vol. 42, pp. 1000–1011, Feb.–Apr. 1994.
[5] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* Amsterdam, The Netherlands: Kluwer Academic, 1992.