REDUCED COMPLEXITY M-ARY HYPOTHESES TESTING IN WIRELESS COMMUNICATIONS

Mohammed Nafie and Ahmed H. Tewfik

Dept. of Electrical Engineering, University of Minnesota
Minneapolis, MN 55455
mnafie,tewfik@ece.umn.edu

ABSTRACT

We present a progressive refinement approach to M-ary detection problems. The approach leads on average to a logarithmic reduction in the complexity of the detector. It relies on designing binary decision trees that trade complexity with probability of error. We also discuss simplified solutions that can be used in several cases of interest in wireless communications such as CDMA multiuser detection and blind equalization.

1. INTRODUCTION

M-ary hypotheses testing problems arise in many applications including wireless communications (e.g., CDMA receivers, blind equalization, etc..), target recognition, surveillance and information filtering. The complexity of the M-ary testing problem is linear in the number M of possible hypotheses. In many cases, the number M of hypotheses to be tested is large and may be related exponentially to another parameter. For example, we will show below that in the blind equalization problem the noiseless data lies in one plane out of M possible planes. By finding that plane we can detect the data and provide a Maximum Likelihood estimate to the channel. This is an M-hypotheses detection problem, where M is exponential in the length of the channel and the length of the data vector that we are considering. A large M effectively precludes the use of optimal Bayesian approaches to solving the M-ary testing problems. This is one reason why an optimal Bayesian approach is not usually used for solving blind equalization problems in wireless communications.

In this paper we provide a progressive refinement approach to the M-ary testing problem. The main advantage of this approach is its almost logarithmic reduction in complexity. Specifically, we use a binary decision tree to arrive at our decision. Each node of the tree represents a subgroup of the M hypotheses. The cardinality of these subgroups decreases with the depth of the tree. The terminal leaves of the tree represent individual hypotheses. Subgroups corresponding to non-leaf nodes at a given level of the tree need not have an equal number of hypotheses. Furthermore, these subgroups are *not* mutually exclusive. The depth of the tree depends on the M-ary hypotheses testing problem under consideration. We propose an approach for constructing such decision trees. The approach is similar to the well-known techniques that are used to design tree vector quantizers with one major difference. Specifically, we split nodes or subgroups of hypotheses by clustering the hypotheses in the subgroup based on the distances between the signals or signal planes that correspond to the different hypotheses. We associate representative signal or signal planes to each subcluster by identifying decision boundaries that minimize the cardinality of the cluster with maximum cardinality at each level of the tree. However, since we allow subgroups of hypotheses to overlap, we must decide at each step whether we should associate to each child node all the hypotheses that cover the subregion of the decision space that is identified with the node. By dropping one or more hypotheses from the subgroup, we effectively associate the subareas of the decision space linked to these hypotheses that fall in the subspace identified with the parent node to other hypotheses. This of course increases the probability of error but decreases the final complexity of the procedure. The final decision tree is obtained by optimally trading probability of error and complexity. Our approach has been observed to yield on average a logarithmic reduction in complexity with tree depths close to $log_2 M$. Note that our approach may be viewed as a non-iterative design technique for classification trees that uses models for the hypotheses rather than training data as in the usual CART scenario [1].

We also describe simple alternative approaches that are easier to design and can be used in several cases of interest in wireless communications such as CDMA multiuser detection and blind equalization.

This paper is organized as follows. In Section 2, we will explain how the problems of blind equalization, subset selection and CDMA multiuser detection can be posed as M-hypotheses problems. In Section 3, we describe our approach to progressive M-ary hypotheses testing. We provide simple approaches in 3 special cases in detail. Case 1 involves known signals observed in white Gaussian noise. Case 2 involves known signals multiplied by unknown scalar gains observed in white Gaussian noise. Finally, case 3 deals with the blind equalization problem where signals are known to lie in certain planes. We conclude with simulation examples that compare the performance of the proposed approach to the optimal exhaustive search.

2. M-ARY HYPOTHESES PROBLEMS

Let us begin by briefly reviewing several problems that may be viewed as M-ary hypotheses testing problems with large values of M. The first two problems arise in communications. The third problem is a generalization of problem 2.

2.1. CDMA multiuser detection

Code division multiple access have been approved as the new standard in cellular phones. Here, each user is assigned a specific sequence, called signature. The transmitted waveform would be this signature multiplied by the transmitted bit stream. The receiver is

This work was partially supported by ONR under grant N00014-92-J-1678 and partially by AFOSR under grant AF/F49620-94-1-0461.

required to obtain the bits of all the different users. This is why the signatures are chosen so as to minimize their correlation [2]. An optimal receivers perform the following operation

$$\max_{\substack{all\ possible\ combinations \\ of\ user\ bits\ x}} p(r|x) \tag{1}$$

where r is the received signal. We form the M hypotheses as all possible combinations of user bits. For example if each of the n users uses a Quadrature Amplitude Modulation (QAM) constellation, then we have $M = 4^n$ possible received vectors forming our hypotheses. Notice that we can always add a '0' to mark an inactive user and therefore we would have a 5^n hypotheses problem instead.

2.2. Blind Equalization

Intersymbol interference (ISI) occurs when data symbols from a particular constellation are sent over communication channels. These channels are generally non ideal, and so they have a certain impulse response with which the input data stream is convolved. A receiver, in order to detect the transmitter symbols, has to equalize the channel, i.e. deconvolve these two signals. Classically, in deriving the optimum equalizer, the channel is assumed to be known. If the channel is not known and/or changes, the transmitter sends an agreed upon transmitting sequence which help the receiver in identifying the channel. Once the channel is identified, it can be inverted. Practically, a training sequence was used to train an adaptive filter so that the combined response of the channel and the equalizer is an impulse response, and hence the stream output from the equalizer is free of ISI. If the channel changes, then the transmitter has to re-send a transmitting sequence on agreed upon intervals. In many communication applications, especially in wireless communications where a number of users share limited resources, and the channel might change often, transmitting a training sequence occupies part of the bandwidth, and, for example, in time division multiple access, occupies a portion of the time assigned to every user. An increase in the capacity of such systems can be achieved if the receiver was able to equalize the channel without using a training sequence. Several methods appeared in the literature for achieving this [7][8] [9][10]. We can write the received signal as

$$y(n) = \sum_{k=0}^{L-1} h(k)x(n-k) + w(n) \tag{2}$$

where $y(n)$ is the sampled output of the matched filter, $x(n)$ are the transmitted symbols, h is the combined transmitting waveform, channel and matched filter, and $w(n)$ is white noise. This equation can be written in a matrix form as

$$X_n \vec{H} + \vec{w}_n = \vec{Y}_n \tag{3}$$

where

$$X_n = \begin{bmatrix} x(n) & x(n-1) & . & . & x(n-L+1) \\ x(n-1) & x(n-2) & . & . & x(n-L) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x(n-N+1) & x(n-N) & . & . & x(n-N+1-L+1) \end{bmatrix} \tag{4}$$

and

$$\vec{Y}_n = \begin{bmatrix} y(n) & y(n-1) & . & .y(n-L+1-N+1) \end{bmatrix}^T \tag{5}$$

and

$$\vec{H} = \begin{bmatrix} h(0) & h(1) & . & . & h(L-1) \end{bmatrix}^T \tag{6}$$

We can find the transmitted data, $x(n)$ by solving a joint maximum likelihood problem on the data and the channel. The components $x(n), x(n-1), ...x(n-N+1-L+1)$ completely specifies the matrix X_n. The length of this vector is $N+L-1$. Therefore there are a^{N+L-1} possible data vectors, where a is the size of the alphabet. By enumerating all these vectors and maximizing the likelihood of the received vector over all these possibilities we can find the best solution.

2.3. Subset Selection

The blind equalization problem as formulated above is an example of subset selection problems. In these problems, one needs to identify a sparse representation of a given signal in terms of elements of an over-complete set of vectors or signals. Such applications include signal coding, chemical analysis of compounds and direction finding. Several algorithms for solving the subset selection problem have been proposed in the literature [3] [4][5]. But, as the problem is NP-complete [6], none of these methods always finds the true global solution. Here we also assume some noise (or model inaccuracies) to be added. Therefore the problem can be written as $\mathbf{\Phi}\vec{\alpha} + noise = \vec{s}$. where $\mathbf{\Phi}$ is the matrix whose column vectors of length P are the over-complete dictionary elements, and $\vec{\alpha}$ has a small number of non-zero elements, n. A true global optimization technique is to use brute force search to search through all linearly independent subsets of a given size from the dictionary and decompose the signal in each such basis. A numerical measure may be applied to select the "optimality" criterion. This is an NP-complete problem, with complexity increasing combinatorially with the size of dictionary. If we enumerate all the possible solutions we will have $M = \frac{N!}{(N-n)!n!}$ possible solutions, where N is the number of vectors in $\mathbf{\Phi}$. If now we maximize the likelihood of getting $\vec{s}$ over all these possible solutions we get a true optimal solution. But this is computationally prohibitive, and hence we suggest applying the progressive refinement algorithm to these M-ary hypotheses problem to get complexity logarithmic in M. Notice that these M hypotheses are n dimensional hyper-planes in P dimensional space, and we are basically trying to find the most likely plane in which $\vec{s}$ lies.

3. PROGRESSIVE REFINEMENT APPROACH TO M-ARY HYPOTHESES TESTING

As explained in the Introduction, our progressive refinement decision approach relies on a binary decision tree. Specifically, we aim to represent the optimal partitioning of the decision space using a sequence of binary partitions. The sets defined by these binary partitions represent the nodes of the decision tree. Each node is associated to a subspace of the decision space and a subset of hypotheses that cover all the subregions of that subspace. The subspace lies entirely on one side of a binary partitioning of the subspace corresponding to the parent of the node.

Ideally, we would like to reduce our uncertainty by a factor of 2 each time we perform a comparison. In other words, we would hope to group the hypotheses in two large groups and subsequently split these two sets in two recursively. Unfortunately, binary partitions cannot always capture the exact boundary between two groups of hypotheses. Therefore, we cannot group the hypotheses in disjoint groupings if we wish to achieve a detection performance close to optimal. On the other hand, we want to minimize overlap between the groupings to minimize the number of comparisons required to make a decision. Hence, our problem then is one of approximating the partitions of the decision plane with the *minimal* number of binary partitions, or equivalently of designing a tree of minimal depth.

Our procedure is reminiscent of the tree vector quantizer design approaches. It consists of three steps

1. Hypotheses clustering or node splitting step: In this step we cluster the hypotheses associated with a given node into 2 subgroups. The clustering is based on distances between the signals or signal planes associated to the hypotheses. Distances are measured using

the notion of principal angles [11]. Note that the signals or signal spaces associated to a given hypothesis at a given node need not be the ones associated to that hypothesis in the original problem. This follows from the fact that the subspace of the decision space associated to the node may involve part of the decision region associated to an interval. In that case, we need to associate a new centroid to that part.

2. Selection of partitioning boundary and representative signal or signal subspaces associated to clusters: This step is equivalent to the centroid evaluation step in vector quantization. It yields the subspace of the decision spaces associated to each child node. The partitioning boundary is always a boundary between 2 subregions linked to 2 hypotheses that appear in only one of the clusters.

3. Selection of the set of hypotheses associated with each child node: Once we have settled on a given partitioning of the subspace associated to the parent node, we need to decide on the exact set of hypotheses that we will associate to the child node. In particular, we must decide whether we would like to associate hypotheses corresponding to subareas of the decision space that were split by the boundary selected in the previous step to one or both of the children nodes. This decision is performed by using a rate-distortion like framework [12]. (See [13] for details.) If these hypotheses are not associated to a given node, the subareas of the decision space linked to these hypotheses are divided optimally between the neighboring subareas linked to other hypotheses. This increases the probability of error but reduces the final complexity of the procedure. We keep track of the increase in the probability of error and use this information in the tree pruning step.

We are still working on an efficient implementation of the above procedure. At this point we use a brute force approach to implement it. We have tested many variations of the above design procedure to reduce its complexity. (Note that its complexity is an off-line problem.) These variations apply to special cases and involve using non- overlapping subgroups of hypotheses. In the remainder of this section, we describe three special cases that are applicable to wireless communications problems. All three cases involve deterministic signals observed in the presence of identically distributed samples of a zero mean white Gaussian noise process. Case 1 deals with known signals. Case 2 discusses known signals with unknown scalar gains. Case 3 discusses the situation where the signals are known to lie in given planes.

3.1. Case 1: Known signals

Assume that we receive $v_i + noise$, $1 \leq i \leq M$, where the noise is white Gaussian noise. The optimal receiver is $\max_i r^T v_i$ where r is the received vector and v_i is the ith possible transmitted vector. We assume that all vectors v_i have equal norm. We want to divide these M vectors into 2 groups and chose a representative for each group. Assume these representatives are g_1 and g_2. To decide which group is more likely we would compare $r^T g_1$ and $r^T g_2$. Or, we can compare $r^T(g_1 - g_2)$ to zero. Let $g = g_1 - g_2$. Therefore, we should select the g that solves

$$\max_{all\ N-dim\ vectors\ g} \min\ abs(g^T v_i) \tag{7}$$

and then divide the vectors into two groups: those that have positive correlation with g^T and those with negative correlation.

A better solution would be to do the following.

1. Chose a vector g
2. Calculate all the values $s_i = g^T v_i$
3. Sort these values
4. Divide the vectors in two groups such as to maximize the difference between the smallest $A = s_i$ value in one group and the largest $B = s_i$ in the other group. Record the distance between the two groups
5. Chose another g and repeat the above procedure
6. Chose the g that gives the maximum $A - B$ over all possible g's.
7. The test between the two hypotheses would then be to compare $r^T g$ to $A + \frac{A+B}{2}$.

The above procedure can also be used if the vectors v_i do not have equal norms. In that case it does not perform as well as the general procedure described above. However, as shown in Section 4, the resulting scheme has a performance that is still very close to optimal.

3.2. Case 2: Known signals with unknown scalar gains

Assume the received signal is $r = av_i + noise$, where a is an unknown scalar factor that can be positive or negative. The optimal receiver is $\max_i abs(r^T v_i)$ We clearly cannot cluster the data vectors using the suboptimal procedure described above. There are two solutions to this problem. We can assume that we have a $2M$ hypotheses problem where the $2M$ hypotheses are the original M vectors and their negatives. Or we can cluster the vectors using the following k-means algorithm

1. Divide the vectors into two groups in any way
2. Chose g_l for group l such that g_l that solves
$$\max_{all\ N-dim\ vectors} \min abs(g_l^T v_i) | v_i \in group\ l \tag{8}$$
3. Calculate $A_i = abs(g_1^T v_i)$ and $B_i = abs(g_2^T v_i)$ for all the vectors v_i, and place vector v_i with g_1 if $A_i > B_i$ and with g_2 otherwise
4. Repeat the above procedure till there is no more change

3.3. Case 3: Planes

For simplicity let us consider 2-dimensional planes only. There are M planes, and the received signal is a point in one of these planes added to noise. Formally, if vectors v_{i1} and v_{i2} span plane i, then we receive:$r = \begin{bmatrix} \vec{v_{i1}} & \vec{v_{i2}} \end{bmatrix}^T * h + noise$ where h is an unknown vector of length 2, and the noise is white Gaussian noise. The optimal receiver for such a setting is to project r on each of the possible planes and chose the plane that has the largest projection. Notice that in the case of 2 dimensional planes in 3 dimensional space, any two such planes must meet in a line. Any point lying on the intersection of any two planes causes ambiguity in the process of identifying in which plane the signal lies. Also points close to that intersection might cause that ambiguity since we have additive noise.

This problem can be solved using the general approach described at the beginning of this section. Here we present an alternative simpler solution. We will chose a plane as a representative of each group of planes. We divide the planes into two groups such that the projection of the matrix representing plane i should have larger projection on the representative of its group. Chose the representative planes so as to maximize the minimum difference between the projection of the matrices representing the planes on the two representatives. Notice that this is reasonable since we are really scanning the planes through multiplying those matrices by a vector. Let's assume that we know the distribution of the elements of the vector h. Let A_l be the norm of the projection of the received vector r on G_l, the representative planes for the two groups. If $A_1 - A_2$ is larger than a certain threshold we would assume that the transmitted plane is one in group 1, while if $A_1 - A_2$ is smaller than a certain threshold, we would assume that the transmitted plane is one in group 2. If $A_1 - A_2$ is a value in between these two threshold we would use the optimal detector instead of our progressive refinement scheme. These thresholds are obtained through our assumed knowledge of the distribution of the elements of h.

4. SIMULATION RESULTS

Fig. 1 shows results of cases 1 and 2. Fig. 1(a) compares the probability of error, at various noise standard deviations, of applying the algorithm discussed in 3.1 on 16 unit norm vectors to the optimal probability of error. Fig. 1(b) is for 16 vectors generated by enumerating all the possible bit sequences of a 4-user CDMA setting. Each user is given a random 3-point signature. Fig. 1(c) is for case 2 applied on 8 vectors. In all these cases the vectors were divided into 2 equal groups of 8 vectors each, and so on, so that the final complexity is proportional to $log_2 16$ instead of to 16. It is clear that our algorithm performs very close to the optimal while providing a large reduction in complexity. Fig. 2 shows results for case 3. Here h elements are assumed to be uniformly distributed between -1 and 1. Fig. 2(a) shows the trade off between complexity and performance. We have 4 2 dimensional planes in 4 dimensional space. We cluster the planes into 2 clusters, with planes 2 and 3 in cluster 1 and plane 4 in cluster 2. Plane 1 is placed in either cluster 2 only or in both clusters. This needs on average 2 and 2.5 operations respectively to arrive at a solution. In both cases plane 3 was a representative of cluster 1 and plane 4 was a representative of cluster 2. In this simple example the tradeoff was only applied in the first step of clustering, as in the second step, we were able to optimally divide both clusters into two disjoint subclusters. When M is larger, this probably won't be the case. Fig. 2(b) is the probability of error for the above example when we use 2.5 operations vs noise standard deviation. Fig. 2(c) is obtained using the simple algorithm in 3.3 applied on 4 2 dimensional planes in 4 dimensional space. The optimal algorithm was used only for about .6-.8% of the time, i.e. when $A_1 - A_2$ is in between the thresholds used. Fig. 2(d) is an example of blind equalization were the channel is assumed to be a 2-tap filter, and the transmitter is assumed to send binary bits. We apply the simple algorithm in 3.3 to a case of 4 two dimensional planes in 3 dimensional space. Here, we found that the optimal detector was applied around 60% of the time, but this might be expected as all the 2 dimensional subspaces in a 3 dimensional space must intersect. This percentage will be lower if we looked at longer sequences as in the case of Fig. 2(c). We can see from all these figures the probability of error of our proposed progressive refinement technique is very close to the optimal probability of error.

5. CONCLUSION

In this paper, we have presented a progressive refinement approach to the problem of M-hypotheses detection. We have illustrated its use in problems that arise in CDMA and blind equalization. We are currently working on an efficient implementation of the search procedure, and on designing signatures for CDMA for increasing its capacity.

6. REFERENCES

[1] L. Breiman et al., *Classification and Regression Trees*, Wadsworth International Group, Calif., 1984.

[2] J. Proakis, *Digital Communications*, McGraw Hill, NY, 1995.

[3] Chen, S., and Donoho D., "Atomic Decomposition by Basis Pursuit", Preprint, 1995.

[4] Mallat, S., and Zhang, Z., "Matching Pursuits with Time-Frequency Dictionaries", IEEE Trans. on Signal Processing, vol. 41, no. 12, pp. 3397-3415, Dec. 1993.

[5] M. Nafie, M. Ali,and A. Tewfik, "Optimal Subset Selection for Adaptive Signal Representation", ICASSP 1996.

[6] B. Natarajan, "Sparse Approximate Solutions to Linear Systems," SIAM J. Comput., vol. 24, pp.227-234, Apr. 1995.

[7] Y. Sato, "A Method of Self-recovering Equalization for Multilevel Amplitude-Modulation Systems," IEEE Trans. on Comm., pp.679-682, June 1975.

[8] L. Tong, G. Xu and T. Kailath, "A New Approach to Blind Identification and Equalization of Multipath Channels," 25th Asilomar Conf., Nov. 1991.

[9] L. Tong, G. Xu and T. Kailath, "Fast Blind Equalization Via Antenna Arrays," ICASSP 1993.

[10] E. Moulines, P. Duhamel, J. Cardoso, and S. Mayrargue, "Subspace Methods for the Blind Identification of Multichannel FIR Filters," IEEE Trans. on Sig. Proc., Vol. 43, pp.516-525, Feb. 1995.

[11] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 1989.

[12] K. Ramchandran, M.Vetterli, "Rate-distortion optimal fast thresholding with complete JPEG/MPEG decoder compatibility," IEEE Trans. on Image Proc., Vol.3 pp.700-704, Sept. 1994.

[13] M. Nafie, PhD Thesis Proposal, Univ. of Minnesota, in preparation.

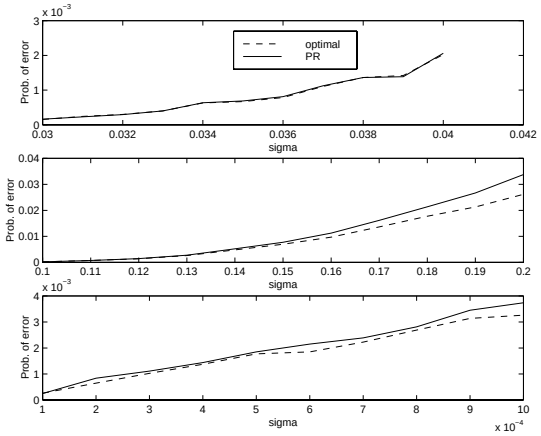

Figure 1: (a)Error for 16 unit norm vectors-(b)Error for 4 users with a 3-point signature-(c)Error for 8 vectors multiplied by an unknown scalar

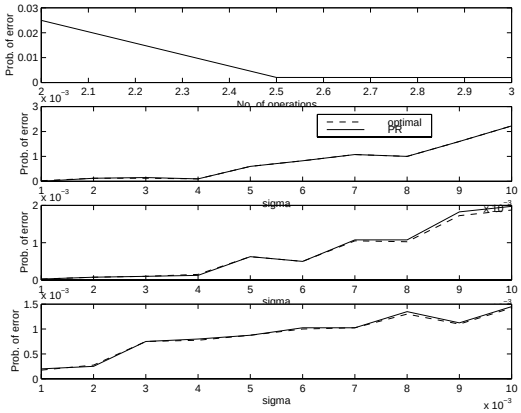

Figure 2: (a)Complexity trade off-(b)(c)Error for 4 2-dimensional planes in 4-dimensional space-(d) A blind equalization example