A GENETIC-ALGORITHM-BASED MUTIUSER DETECTOR FOR MULTIPLE-ACCESS COMMUNICATIONS

X. F. Wang, W.-S. Lu, and A. Antoniou
Department of Electrical and Computer Engineering
University of Victoria
Victoria, B.C., Canada
Email: {xfwang, wslu, andreas}@ece.uvic.ca

ABSTRACT
We propose a new detector for multiuser communications, which is based on the maximum-likelihood decision rule and uses a genetic algorithm in conjunction with a modified Viterbi algorithm to detect the user bits sequentially. The major advantage of the proposed detector is that it provides a trade-off between performance and computational complexity. Simulation results show that the proposed detector achieves similar performance to that of two well-known sub-optimal linear detectors with a lower computational complexity.

1. INTRODUCTION
The optimal multiuser detector [1] maximizes the joint a posteriori probability with a computational complexity that grows exponentially with the number of users. The high computational complexity involved has motivated the search for suboptimal multiuser detectors with much reduced computational complexity. An important family of suboptimal multiuser detectors, namely, linear multiuser detectors, were proposed and extensively analyzed in [2]-[4]. The lower computational complexity of these detectors is at the cost of performance degradation primarily due to the suboptimal decision rules adopted.

In order to maximize the joint a posteriori probability, the optimal multiuser detector must evaluate a log likelihood function over the set of all possible information sequences. This combinatorial optimization problem was shown to be NP-complete [5], and traditional optimization algorithms are inefficient for its solution. However, earlier works have indicated that combinatorial optimization problems can be solved efficiently by using genetic algorithms (GAs) [6]. Motivated by this observation, we propose a new multiuser detector which, unlike the suboptimal detectors found in the literature, is based on the maximum-likelihood decision rule and uses a GA to maximize the joint a posteriori probability to achieve much improved efficiency. GAs were first introduced by Holland [7] and have received considerable attention in DSP research [8]. As a guided random technique, a GA is naturally suited for solving the multiuser detection problem as it fully exploits the information contained in the output of the matched filters. It is also interesting to note that by appropriately modifying the termination criterion used in the GA, the proposed GA-based detector is inherently capable of making a trade-off between performance and implementation complexity.

In this paper, we first briefly review the optimal multiuser detector as well as two types of linear detectors. The basic idea of GAs is then introduced and applied to solve the multiuser detection problem. For the sake of generality, asynchronous transmission is assumed. In such a case, the optimal detector is of infinite memory length, and a direct implementation is impractical due to its high computation burden. This difficulty is overcome by using a dynamic programming technique (a modified Viterbi algorithm) in conjunction with a GA to transform the problem into a sequential optimization problem. The paper concludes with a comparison of the proposed detector with the linear decorrelating detector [3] and the linear maximum mean-square-error (MMSE) detector [4].

2. MULTIUSER DETECTION
We consider binary phase-shift-keying (BPSK) transmission through an additive white Gaussian noise (AWGN) channel shared by K asynchronous users in a direct-sequence code-division multiple-access (DS-CDMA) system. Let the message length be N and denote the kth information bit of the ith user, the corresponding signal energy, and the output of the matched filter by bi(k) ∈ {−1, 1},  e(k), and r(k), respectively. For coherent reception, the concatenation of N successive output vectors of the K matched filters, rN, can be modeled as [9]

\[ r_N = R_N E_N b_N + n \]  

(1)

where

\[ r_i = [r^T(1) \ r^T(2) \ \cdots \ r^T(i)]^T \]
\[ r(k) = [r_1(k) r_2(k) \ \cdots \ r_K(k)]^T \]
\[ b_i = [b^T(1) \ b^T(2) \ \cdots \ b^T(i)]^T \]
\[ b(k) = [b_1(k) b_2(k) \ \cdots \ b_K(k)]^T \]
\[ E_i = \text{diag}(E(1) \ \cdots \ E(i)) \]
\[ E(k) = \text{diag}(\sqrt{e_1(k)} \ \sqrt{e_2(k)} \ \cdots \ \sqrt{e_K(k)}) \]
In (1), \( n \) is the AWGN vector with zero mean and covariance matrix \( \sigma^2 R_N \), and
\[
R_i = \begin{bmatrix}
R(0) R(1)^T & 0 & \cdots & 0 \\
R(1) & R(0) & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & R(1) R(0) R(1)^T \\
0 & 0 & \cdots & 0 & R(1) R(0)
\end{bmatrix}
\tag{2}
\]

is an \((i \times i)\)-block 3-band matrix. Each \( R(m) \) for \( m = 0, 1 \) in (2) is a \( K \times K \) matrix whose \((k, l)\)th entry is given by
\[
R_{kl}(m) = \int_{-\infty}^{\infty} g_k(t - \tau_k) g_l(t + mT - \tau_l) dt \quad \text{for} \quad m = 0, 1
\]

where \( g_k(t) \) and \( \tau_k \) denote the normalized signature signal and the transmission delay for the \( k \)th user, respectively. For the sake of convenience, the transmission delays are arranged in ascending order, i.e., \( 0 = \tau_1 \leq \cdots \leq \tau_K < T \), so as to obtain an \( R(1) \) which is an upper triangular matrix with zero diagonal entries. In the rest of the paper, we drop the subscript \( N \) in \( b_N, E_N \) for simplicity.

It can be shown [9] that maximizing the joint a posteriori probability for the optimal multiuser detection is equivalent to minimizing the correlation metric
\[
C(b) = b^T E R_N E b - 2r^T E b
\tag{3}
\]

Notice that the optimal detector must evaluate metric \( C(b) \) over all \( 2^{NK} \) possible \( b \) vectors in order to identify the \( b \) that minimizes \( C(b) \). This approach involves a great deal of computation and causes a large decision delay in implementation. The Viterbi algorithm is usually adopted to solve the problem sequentially thereby resulting in a computational complexity which is exponential in \( K \) rather than exponential in \( NK \) [2].

In contrast to the optimal multiuser detector, a linear multiuser detector can be viewed as a linear mapping, \( T \), as applied to the output of the matched-filter bank, \( r \). The linear mappings for the decorrelating detector and the MMSE detector are
\[
T_L = R_N^{-1}
\tag{4}
\]
and
\[
T_M = (R_N + \sigma^2 E^{-2})^{-1}
\tag{5}
\]
respectively. These two detectors have shown a number of advantages. However, due to the changes in the correlations \( R_N(m) \), in practice the linear mappings have to be updated from time to time, which requires a considerable amount of computation.

3. GENETIC ALGORITHMS

A GA is a guided random technique which mimics the process of natural evolution. A simple GA contains the following steps:

1. Generate an initial population \( P(0) \).
2. Evaluate the objective function at points in \( P(0) \), and map these objective values into fitness values.
3. Create a new population \( P(r) \) by applying crossover and mutation.
4. Evaluate the objective function at points in \( P(r) \), and map these objective values into fitness values.
5. If the termination criterion is satisfied, output the solution and stop; otherwise, set \( r := r + 1 \) and repeat from Step 3.

The population \( P(r) \) comprises a set of chromosomes which are encoded representations of possible solutions. The encoding is for the convenience of the process of genetic operations in Step 3. For the problem at hand, however, there is no need for encoding since the possible solutions (user information sequences) are vectors with binary elements.

The fitness value of a chromosome can be obtained by applying a mapping operation to its objective value, e.g., \( C(b) \) for the multiuser detection problem. Based on the fitness values, parent chromosomes are selected to generate the next population through crossover and mutation. In the proposed GA-based detector, windowing mapping and proportionate selection are adopted. In window mapping, each chromosome is assigned a fitness value \( f_i \) as
\[
f_i = c + |V_i - V_w|
\tag{6}
\]
where \( V_i \) and \( V_w \) are the objective values of the \( i \)th chromosome and the worst chromosome, respectively, and \( c \) is a constant. Once all the chromosomes are assigned fitness values, the parent selection can be carried out by the survival-of-the-fittest law. If we let \( f_{total} \) denote the sum of the fitness values of chromosomes in a population, then the probability of a chromosome being selected as a parent is assigned by proportionate selection as
\[
p_i = \frac{f_i}{f_{total}}
\]
Crossover and mutation are two types of genetic operations. The crossover operation combines substrings of two parent chromosomes to create two new offspring chromosomes. A number of variations on crossover operations have been proposed [6]. For multiuser detection, we select uniform crossover whereby every locus in the chromosomes is a possible crossover site. A randomly generated binary string with the same length as the chromosome determines which parent the corresponding bit of the offspring chromosomes comes from. For example, two offspring chromosomes can be created by exchanging the bits of the parents if the corresponding bit of the random string is 0 as depicted in Fig. 1.

\[\text{Parent Chromosomes} \quad \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \]
\[\text{Offspring Chromosomes} \quad \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \]
\[\text{The random string is 010001}\]

Figure 1. Example of uniform crossover.

Mutation introduces variations into the chromosome with a given probability \( p_m \). For each bit of a chromosome, a probability test is first applied. If the test is passed, then the
sign of the bit will be changed. Typically, the probability \( p_m \) is very small. Therefore, in comparison with the crossover operation, the mutation operation plays a minor role in the GA.

The scheme for the replacement of the old generation is another important factor which affects the performance of the GA. An old generation \( P(r) \) may be entirely replaced by a new generation \( P(r+1) \) which is created by applying genetic operations to \( P(r) \). This is called generation replacement. Notice that an optimization procedure using this scheme might retrograde if the best chromosomes fail to reproduce. A safer scheme is elitist replacement whereby some of the best chromosomes are copied into the next generation. This scheme guarantees the progress of the optimization and is adopted in our proposed detector.

4. A GA-BASED MULTIUSER DETECTOR

Pre-decision If the number of generations is fixed, the performance of a GA is proportional to the rate of the population size to the dimension of the chromosomes. Therefore, if the number of bits needed to be detected can be reduced, the amount of computation will be reduced and the performance of the algorithm will be much improved given fixed computing resources.

A simple way to reduce the number of unknown bits is to utilize the information given by the output of the matched-filter bank, \( r \), as follows. A lower bound for the bit-error probability (BEP) of the \( k \)th user of the optimal multiuser detector can be found as the BEP of a single-user receiver in the absence of other users \([1]\), i.e.,

\[
p_e(k) = Q(\sqrt{\gamma_k})
\]

where \( \gamma_k = c_k/\sigma^2 \), and \( Q(x) \) is given by

\[
Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt
\]

Thus a confidence measure \( \text{conf} \) can be derived as a fraction of \( p_e(k) \), say \( \text{conf} = 0.1p_e(k) \). The probability of \(-\text{sign}(r_k)\) being transmitted given that \( r_k \) was received is smaller than \( \text{conf} \) or, equivalently, if \( |r_k| \) is larger than a certain threshold \( t_k \), \( \text{sign}(r_k) \) can be deemed to be the true transmitted information bit. The threshold \( t_k \) can be computed by treating the combination of multiple-access interference as well as the background noise as AWGN, which leads to

\[
t_k = \sqrt{2(\sigma^2 + \sigma_1^2)} \text{erfinv}(2 \text{conf} - 1) - \sqrt{\sigma_1}
\]

where \( \text{erfinv} \) is the inverse error function and \( \sigma_1^2 \) is the variance of the Gaussian noise translated from the multiple-access interference. In practice, \( \sigma_1^2 \) can be simply approximated by the largest off-diagonal magnitude in the correlation matrix \( R_s \).

Sequential detection A sequential implementation of the GA-based detector is derived by exploiting the block tridiagonal structure of \( R_s \). From (2), it is clear that

\[
R_k = \begin{bmatrix} R_{k-1} & L \\ L^T & R(0) \end{bmatrix}
\]

where

\[
L = [0 \ R(1)^T]
\]

It then follows from (3), (9), and (10) that

\[
C(b_k) = C(b_{k-1}) + 2b^T(k - 1)E_1R(1)^TE_1b(k) + b^T(k)E_1R(0)E_1b(k) - 2r^2(k)E_1b(k) \tag{11}
\]

After performing the pre-decision step, the \( k \)th user bits \( b(k) \) can be divided into two groups: the group of detected information bits \( b_1(k) \) and the group of unknown information bits \( b_2(k) \). At the end of the \( k \)th transmission interval, \( b_1(j) \) for \( j \leq i \) are known to the sequential multiuser detector; hence we can drop all the terms in \( C(b_i) \) that depend only on \( b_1(j) \) and denote the resulting metric by \( \hat{C}(b_2) \), where \( b_2 = [b_2(1)^T \ b_2(2)^T \ \cdots \ b_2(i)^T]^T \). For the sake of convenience, we assume that \( b_1(k) \), which is of length \( m \), contains exactly the information bits of the first \( m \) users. In accordance with the partition of vector \([b_1(k)^T \ b_2(k)^T]^T \), we partition \( r(k) \), \( E_1 \), and \( R(0) \) to obtain

\[
r(k) = [r_1^T \ r_2^T]^T
\]

\[
E_1 = \text{diag}[E_{11} \ E_{12}]
\]

\[
R(0) = \begin{bmatrix} H_1 & H_1^T \\ H_2 & H_2 \end{bmatrix}
\]

Then (11) leads to

\[
\hat{C}(b_2) = \hat{C}(b_{2k-1}) + 2b^T(k - 1)E_1R(1)^TE_1b(k) + 2b_1(k)^TE_1H_{12}E_2b_2(k) + b_2(k)^TE_2H_2E_2b_2(k) - 2r^2TE_2b_2(k) \tag{12}
\]

From (12), we see that the Viterbi algorithm can be used to detect the user information bits sequentially. At the \( k \)th stage, \( 2^n(k-1)+n(k) \) partial metrics \( \hat{C}(b_{2k}) \) have to be computed for searching the \( 2^n(k) \) surviving states, where \( n(k) \) denotes the length of \( b_2(k) \). Evidently, this still requires a considerable amount of computation. It is observed that only a small number of states at each stage have chance to survive in the later decision, namely, the states whose partial metrics are significantly larger than the others. Therefore, a small number of states, say \( N_s \), is sufficient to give virtually the same performance as the optimal sequential detector. This observation enables one to employ the GA to search for the \( N_s \) surviving states with the largest partial metrics at each stage rather than computing the partial metrics over all the possible \( b_2(k) \). In our simulations, we found that an \( N_s = 6 \) is sufficient for a system with 20 users.

5. SIMULATIONS AND COMPARISONS

In order to examine the proposed detector, extensive Monte-Carlo simulations were carried out for the decorrelating detector, the MMSE detector, and the GA-based detector. In all simulations, a 31-chip Gold code family with a rectangular chip waveform was used for the spreading signals. For cases with the near-far problem, the received user energies were randomly generated to be uniformly distributed in the interval \([e(1), e(k)]\). The average BEP versus signal-to-noise ratio \((e_i/N_0)\) with \( N_0 = 2\sigma^2 \) for
different detectors is plotted in Fig. 2. As can be seen, the performance of the GA-based detector is very close to that of the MMSE detector and compares favorably with that of the decorrelating detector.

![Graphs showing BEP of different detectors](image)

Figure 2. Average BEP of different detectors for asynchronous systems: (a) with equal received energies, (b) with near-far problem.

Denote the number of generations and the size of a population by \( N_g \) and \( N_p \), respectively. In the \( k \)th stage, the GA-based detector computes \( C(b_{2k}) \) in \((12) N_g N_p \) times and each time about \( n(k)^2 \) flops\(^1\) are required. Since usually about half of the bits can be detected in the pre-decision stage, the computational complexity of the proposed detector is about \( N_g N_p K / 4 \) flops per bit. For a system with 20 users, which is typical in present micro-cell CDMA mobile communications, \( N_g = 6 \) and \( N_p = 20 \). Hence, the proposed detector requires about \( 30K \) flops per bit. On the other hand, the two linear detectors have the same computational complexity, and each of them requires \( 6K \) flops per bit given that the linear mappings are known. However, the linear mappings have to be updated from time to time. By fully taking advantage of the tridiagonal structure of the cross-correlation matrix, the updating requires about \( 2K^2 \) flops per bit. Hence the overall number of flops required by the linear detectors is about \( 2K^2 + 6K \) flops, which is larger than that of the proposed detector.

6. CONCLUSIONS

We have proposed a GA-based multiuser detector which uses a genetic algorithm in conjunction with a modified Viterbi algorithm to detect the user bits sequentially. A pre-decision scheme has been proposed and integrated into the new detector to enhance the performance and computational efficiency. The computer simulations have shown that the proposed detector achieves almost the same performance as that of two well-known linear detectors with a lower computational complexity.

REFERENCES


---

\(^1\) One floating-point multiplication plus one floating-point addition are counted as one flop.