

GENETIC ALGORITHMS FOR BLIND MAXIMUM- LIKELIHOOD RECEIVERS

Romis R. de F. Attux¹, Renato R. Lopes¹, Leandro N. de Castro², Fernando J. Von Zuben¹, João Marcos T. Romano¹

¹FEEC/Unicamp, Caixa Postal 6101, CEP: 13083-970, Campinas/SP, Brazil

²Unisantos, Catholic University of Santos, R. Dr. Carvalho de Mendonça, 144, 11070-906, Santos/SP, Brazil

{rlopes, romisri, romano}@decom.fee.unicamp.br, lnunes@unisantos.edu.br, vonzuben@dca.fee.unicamp.br

ABSTRACT

The ultimate receiver in a communications system is one that minimizes the bit-error rate (BER) or, equivalently, that maximizes the likelihood function. Unfortunately, a maximum-likelihood (ML) receiver can be prohibitively complex in some cases. For instance, in a blind system, where neither the channel nor any part of the transmitted sequence are known, an ML receiver would have to test all possible transmitted sequences to determine the one that minimizes the BER. In this paper, we derive a likelihood function for blind communications, and we use a genetic algorithm as the optimization strategy, at a reasonable computational cost. The performance of the resulting algorithm can be improved by exploiting structural aspects of the transmitted sequence that are normally neglected by blind techniques, such as the presence of some known symbols or of an error-control code. Simulation results are presented to validate the proposal.

1. INTRODUCTION

Intersymbol interference (ISI) is one of the most common impairments introduced by communications systems, and it is normally compensated for by an equalizer. Given a block of received symbols, the best equalizer, in the sense that it minimizes the bit-error rate (BER), is the maximum-likelihood (ML) equalizer, implemented via the Viterbi algorithm [1]. The Viterbi algorithm has two main disadvantages. First, its complexity grows exponentially with the length of the channel impulse response and the size of the modulation, rendering it impractical for long channels or large constellations. Second, it requires the estimation of the communications channel impulse response.

Channel estimation is traditionally based on the transmission of a training sequence, also known to the receiver. However, the transmission of a training sequence in some systems is undesirable or impossible. For instance, in wireless systems the training sequence increases the system overhead, thus decreasing the number of users that the system can accommodate and/or the rate available for

each user. Furthermore, traditional trained systems are suboptimal, since only the received samples corresponding to the training symbols are used for estimation. The size of the training sequence can be reduced if all the received samples are used for estimation, leading to *semi-blind* systems [2]. Interestingly, channel estimation is possible even when a training sequence is not available. In this case, estimation is based solely on the statistics of the transmitted symbols, resulting in *blind* systems [3]. These systems are the focus of this work.

In a blind context, the ML detector also minimizes the BER. The blind likelihood function depends on the channel parameters and the transmitted sequence; therefore, the blind ML detector jointly and optimally performs channel estimation and equalization. However, finding the blind ML solution is prohibitively complex, because, *in abstracto*, it would be necessary to compute and compare the likelihood of virtually every possible transmitted sequence to determine the ML sequence. Even in simple problems, the resulting search space can have more than 10^{40} elements.

Problems such as blind ML detection, which exhibit an explosive increase in the computational burden, are well-suited to efficient and relatively parsimonious heuristic search techniques, such as genetic algorithms (GA) [4]. This assumption is encouraged by the work of Chen and Wu [5], wherein this kind of optimization tool is used to solve the blind ML problem. The GA proposed in [5] is based on explicit channel estimates and on the Viterbi algorithm.

In this work, we propose an alternative receiver based on a simple formulation of the ML problem and on a specially tailored GA. We first derive in Section 2 a likelihood function that does not depend on the channel estimate. The computation of this likelihood function is simple, and its complexity grows only linearly with the channel length and the length of the transmitted sequence. Based on this likelihood, we propose in Section 3 a GA that works directly on the transmitted sequence, obviating the need for the Viterbi algorithm and direct channel estimates. Section 4 is dedicated to applicability issues, including a deeper comparison of the present proposal with the one presented in [5]. Simulation results are shown in Section 5 to illustrate the search capability of the proposed algorithm. Concluding remarks and future perspectives are briefly outlined in Section 6.

2. CHANNEL MODEL AND THE LIKELIHOOD FUNCTION

In this work, we consider the transmission of a sequence of symbols $\mathbf{a} = [a_1 \dots a_N]$ through a channel whose impulse response is given by $\mathbf{h} = [h_0 \dots h_M]$. For ease of presentation, we assume a binary alphabet, where $a_k \in \{-1, +1\}$, although generalization to higher-order constellations is straightforward. The received sequence is $\mathbf{r} = [r_1 \dots r_L]$, where

$$r_k = \sum_{m=0}^M h_m a_{k-m} + n_k, \quad (1)$$

$L = M + N$, n_k is an independent and identically distributed additive-white-Gaussian noise with variance σ^2 , and we assume that $a_k = 0$ if $k \notin \{1, \dots, N\}$.

Now, consider the likelihood function $p(\mathbf{r}|\hat{\mathbf{a}}, \hat{\mathbf{h}})$, defined as the conditional probability of \mathbf{r} assuming that the sequence $\hat{\mathbf{a}}$ was transmitted through the channel $\hat{\mathbf{h}}$. In particular, consider the ML vectors \mathbf{a}_{ML} and \mathbf{h}_{ML} that maximize this function: $(\mathbf{a}_{\text{ML}}, \mathbf{h}_{\text{ML}}) = \text{argmax } p(\mathbf{r}|\hat{\mathbf{a}}, \hat{\mathbf{h}})$. These are the parameters that best “explain” the received sequence, in the sense that we are less likely to observe \mathbf{r} if we assume any other set of parameters, i.e., $p(\mathbf{r}|\hat{\mathbf{a}}, \hat{\mathbf{h}}) \leq p(\mathbf{r}|\mathbf{a}_{\text{ML}}, \mathbf{h}_{\text{ML}}) \forall \hat{\mathbf{a}}, \hat{\mathbf{h}}$. More importantly, it can be shown that \mathbf{a}_{ML} minimizes the BER [1]. Since the BER is the ultimate performance measure of a communication system, \mathbf{a}_{ML} is the best possible estimate of the transmitted sequence.

Note that maximizing a positive function is equivalent to maximizing its logarithm. Recalling that the noise samples are independent and Gaussian, it can be shown [1] that

$$\log(p(\mathbf{r}|\hat{\mathbf{a}}, \hat{\mathbf{h}})) = -\|\mathbf{r} - \hat{\mathbf{r}}\|^2 + C, \quad (2)$$

where C is a constant that does not depend on $\hat{\mathbf{a}}$ and $\hat{\mathbf{h}}$, and $\hat{\mathbf{r}} = [\hat{r}_1 \dots \hat{r}_L]$, where

$$\hat{r}_k = \sum_{m=0}^M \hat{h}_m \hat{a}_{k-m}. \quad (3)$$

In matrix notation, we may write $\hat{\mathbf{r}} = \hat{\mathbf{h}}\hat{\mathbf{A}}$, where $\hat{\mathbf{A}}$ is an $(M+1) \times L$ matrix given by

$$\hat{\mathbf{A}} = \begin{bmatrix} a_1 & a_2 & \dots & a_k & \dots & 0 \\ 0 & a_1 & \dots & a_{k-1} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & a_{k-M} & \dots & a_N \end{bmatrix}. \quad (4)$$

From (2), it is clear that finding the ML solution is equivalent to finding the estimated received sequence $\hat{\mathbf{r}}$ that is closest to the actual received sequence \mathbf{r} . In other words, $(\mathbf{a}_{\text{ML}}, \mathbf{h}_{\text{ML}}) = \text{argmin } \|\mathbf{r} - \hat{\mathbf{r}}\|^2$. Since we are only interested in estimating the transmitted sequence, we may write

$$\mathbf{a}_{\text{ML}} = \text{argmin}_{\hat{\mathbf{a}}} (\min_{\hat{\mathbf{h}}} \|\mathbf{r} - \hat{\mathbf{r}}\|^2). \quad (5)$$

The minimization problem within parenthesis should be solved for each given $\hat{\mathbf{a}}$. Considering the matrix form of $\hat{\mathbf{r}}$, we see that this minimization is a traditional least-squares problem [6], leading to

$$\begin{aligned} \mathbf{a}_{\text{ML}} &= \text{argmin}_{\hat{\mathbf{a}}} (\|\mathbf{r}\|^2 - \mathbf{r}\hat{\mathbf{A}}^T (\hat{\mathbf{A}}\hat{\mathbf{A}}^T)^{-1} \hat{\mathbf{A}}\mathbf{r}^T), \\ &= \text{argmax}_{\hat{\mathbf{a}}} \mathbf{r}\hat{\mathbf{A}}^T (\hat{\mathbf{A}}\hat{\mathbf{A}}^T)^{-1} \hat{\mathbf{A}}\mathbf{r}^T \end{aligned}, \quad (6)$$

where the second equality follows from the elimination of the terms of the first equality that do not depend on $\hat{\mathbf{a}}$. Thus, (6) gives a simple function of the transmitted sequence whose maximization is equivalent to the ML problem. Computing this function requires $(M+1)N$ multiplications and $(M+1)(N-1)$ additions, as

well as the inversion of an $(M+1) \times (M+1)$ matrix. Also, note that $\hat{\mathbf{h}}$ does not appear in (6).

The term being maximized in (6) is equal to the square of the norm of the projection of \mathbf{r} onto the space spanned by the rows of $\hat{\mathbf{A}}$. This observation yields an interesting interpretation to the blind ML problem: find the matrix $\hat{\mathbf{A}}$ such that the projection of \mathbf{r} onto the range space of the rows of $\hat{\mathbf{A}}$ has maximum norm. Interestingly, note that the spaces spanned by $\hat{\mathbf{A}}$ and $-\hat{\mathbf{A}}$ are the same. Thus, if $\hat{\mathbf{a}}$ is an ML sequence, so is $-\hat{\mathbf{a}}$. This type of ambiguity is standard and acceptable in blind systems [3].

Finally, note that $\|\mathbf{r} - \hat{\mathbf{r}}\|^2/L$ is the estimated noise variance assuming that $\hat{\mathbf{a}}$ was transmitted. Thus, if \mathbf{a}_{ML} is known, we can obtain the ML noise estimate as $\|\mathbf{r} - \mathbf{r}_{\text{ML}}\|^2/L$. This observation yields an interesting test for a given sequence $\hat{\mathbf{a}}$ when the noise variance is known: if the estimated noise variance is close to the actual noise variance, we may have some degree of confidence that $\hat{\mathbf{a}}$ is the ML solution. The difference between the actual and the estimated noise variance can be used as a stopping criterion for any detection algorithm.

3. A GENETIC ALGORITHM FOR ML DETECTION

The task of finding a sequence that fulfils the requirements established in (6) is by no means simple. In fact, an exhaustive search is the only known exact algorithm. However, the associated computational cost is prohibitive even for a modest value of N , since the relation between N and the number of possible sequences is governed by an exponential law. This undesirable feature renders imperative the use of techniques capable of allying the parsimony required by a practical application to a remarkable search capability. Furthermore, due to the lack of expert information, it is necessary that the proposed algorithm be based on a framework as general as possible.

Genetic Algorithms (GA) [4] belong to the distinct class of techniques that proved themselves capable of meeting the aforementioned demands in countless occasions. In very simple terms, a GA can be understood as a population-based stochastic search procedure inspired by elements of the modern evolution theory, such as *selection*, *recombination (crossover)* and *mutation*. Its remarkable search potential derives from the synergy among these elements, which implicitly combines the ideas of local and global search.

The promising characteristics of GAs, together with their reported success in problems similar to the one at hand [5], justify our proposal: to use an evolutionary technique as the basis of a paradigm for approximate blind ML sequence estimation. In other words, we will attempt to solve the problem posed by (6) with a genetic algorithm. Since solving (6) is equivalent to maximizing the likelihood,

the proposed solution is equivalent to a GA having the likelihood function as the fitness measure. The proposed GA, heuristically built to accomplish this particular task, is presented in Table 1.

Table 1: Proposed Genetic Algorithm

<p>1. Initialization</p> <ul style="list-style-type: none"> • Randomly create the N_{ind} individuals of the initial population. Each individual consists of a candidate for the transmitted sequence, \hat{a}. <p>2. While Stopping Criterion is not Achieved</p> <p>2.1 – Create an Intermediate Population with $2N_{ind}$ Individuals</p> <ul style="list-style-type: none"> • Select $1.8N_{ind}$ individuals by means of a binary tournament • Insert $0.2N_{ind}$ randomly generated individuals <p>2.2 – Uniform Crossover</p> <ul style="list-style-type: none"> • Divide the individuals of the intermediate population into pairs (parents) • Randomly choose $N/2$ positions of the chromosome • Switch the bits associated with the chosen positions, thereby producing two offsprings • Replace the parents by the offsprings <p>2.3 – Mutation</p> <ul style="list-style-type: none"> • Change, with probability p_m, bits belonging to the individuals of the intermediate population <p>2.4 – Final Selection</p> <ul style="list-style-type: none"> • Select 50% of the individuals from the best 10% of the intermediate population • Select 50% of the individuals from the worst 40% of the intermediate population • Select 50% of the individuals from the remaining 50% of the intermediate population <p>2.5 – Elitism</p> <ul style="list-style-type: none"> • Reintroduce the best individual of the population if it has been lost.

4. APPLICABILITY ISSUES

In this section, we discuss some practical aspects of the GA presented in Table 1. We begin by observing that the computational cost of the proposed GA stems mostly from the computation of the fitness of the individuals in the population; the other steps in the algorithm have negligible cost. Since the fitness is computed $2N_{\text{ind}}$ times per iteration, the number of individuals in the population directly affects the cost of each iteration of the proposed GA.

As seen in Table 1, the individuals in the population of the proposed GA consist of candidates for the transmitted sequence, $\hat{\mathbf{a}}$. This is a very attractive feature, since it allows the GA to incorporate any knowledge about the transmitted sequence. For instance, the proposed GA can be easily adapted to semi-blind systems, where a short training sequence is transmitted. In this case, the entries of $\hat{\mathbf{a}}$ corresponding to the training symbols remain constant at their known values, and are not subject to mutation and crossover. Other important structures of the transmitted sequence arise in systems where error-control coding is present and/or where a higher-order modulation is used [1]. In these cases, we are actually interested in recovering the message bits \mathbf{m} that are fed to the encoder or the modulator, whereas the channel input \mathbf{a} is, to some extent, of no interest other than aiding in the estimation of \mathbf{m} . The proposed GA can easily exploit the structure induced by encoding and modulation by working directly on the message bits. The computation of the likelihood of a given \mathbf{m} is normally straightforward, since, in most cases, the encoding and the modulation functions are easy to compute; therefore, the channel input \mathbf{a} corresponding to a given \mathbf{m} can be easily determined and used in (6) to compute the likelihood of \mathbf{m} .

In contrast to the current proposal, the individuals in the population of the GA proposed in [5] consist of channel estimates. These estimates are used in a Viterbi algorithm to determine the most likely transmitted sequence assuming that the channel estimates are correct. The likelihood function is then computed using (2). However, since the complexity of the Viterbi algorithm grows exponentially with the channel length M and the size of the modulation, the computation of the fitness in [5] is impractical for systems with long impulse responses or high modulation orders. Furthermore, determining the most likely transmitted sequence for a given channel estimate is normally very hard in coded systems. In fact, in most cases the only known exact solution is also exhaustive search. Thus, the system proposed in [5] cannot be easily used in coded systems.

Finally, it should be noted that GAs, in spite of their remarkable search capability, cannot provide successful convergence at a guaranteed cost. In other words, each execution of a GA may require a varying number of iterations to converge. This observation may limit the applicability of a GA in real-time systems. However, practical applications of the proposed system can still be envisioned. For instance, consider an application in which the channel impulse response changes slowly with time. In this case, the channel estimates produced by the proposed GA may be used to periodically and blindly initialize other blind ML algorithms such as

the expectation-maximization algorithm [7], which are plagued by convergence to local maxima of the ML function.

5. SIMULATION RESULTS

In order to test the proposal, we considered two channels: C1, whose coefficient vector is $h_1 = [0.8575 \ 0.5145]$, and C2, a more complex channel with two zeros on the unit circle and coefficient vector $h_2 = [0.5 \ 0.7 \ 0.5]$. In both cases, we assumed that $\text{SNR} \rightarrow \infty$, which implies that the ML solution is the actual transmitted sequence. We chose a 2-PAM modulation with unit variance and $N = 150$ symbols as the size of the transmitted sequence. Notice that, under these assumptions, the number of possible solutions is $N_{\text{total}} = 2^{(150-1)} \approx 7.136 \times 10^{44}$. We performed 20 runs of this experiment for each channel, which allowed us to evaluate the average convergence rate and other relevant averages with a satisfactory degree of precision. Based on the discussions about noise variance estimation in Section 2, an infinite SNR implies that \mathbf{r}_{ML} should be close to \mathbf{r} . Thus, in all cases, the stopping criterion was a fitness threshold to be reached within a maximum number of iterations. In other words, since the noise variance is zero, the proposed GA would stop if $\|\mathbf{r} - \hat{\mathbf{r}}\|^2 > 10^{-6}$, or if a maximum number of iterations were reached.

5.1 – Channel C1

The parameters of the GA for this case are presented in Table 2. The optimal solution was attained in all simulations, which demonstrates the efficiency of the proposal. The average number of iterations was 118.9, and the maximum number was 269, both noticeably small. In Fig. 1, we present a typical time evolution of the fitness associated with the best individual in the population. The rapid convergence confirms the high exploration capability inherent to the adopted search procedure. We also show in Fig. 1 the evolution of the average fitness of the population. As we can see, there is a significant difference between the average fitness and the maximum fitness, indicating that the diversity in the population is maintained along generations.

Table 2: Parameters – First Test

Parameter	Value
Number of Individuals N_{ind}	180
Probability of Mutation (p_m)	0.002
Individuals per Tournament	2

The average number of evaluations of the fitness function was 42,984, a number that is very small in comparison with N_{total} . This confirms our best expectations that the GA would be able to detect the transmitted sequence after considering just few candidate sequences.

5.2 – Channel C2

Despite the higher complexity of the problem engendered by this channel, we decided to use the same number of individuals as in the previous example, thus keeping the computational cost per iteration of both cases at similar levels. The parameters of the GA for channel C2 are shown in Table 3.

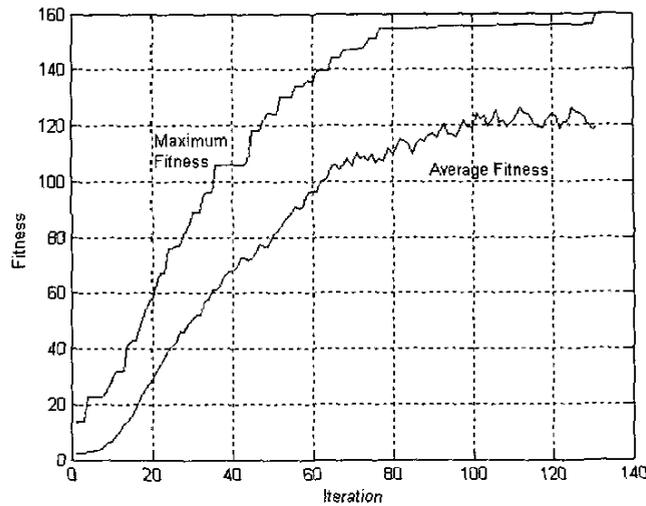


Figure 1. Evolution of the average fitness and of the fitness of the best individual for channel C1.

Table 3: Parameters – Second Test

Parameter	Value
Number of Individuals N_{ind}	180
Probability of Mutation (p_m)	0.005
Individuals per Tournament	2

Once more, the optimal solution was attained in all twenty trials, which strongly reasserts the feasibility of the use of GAs in this particular problem. The average number of iterations was 1,802.8, and the maximum number was 5,098, being the increase justified by the more demanding channel structure. The average number of cost evaluations was 649,188, a value once more much smaller than N_{total} .

In Fig. 2, we present the evolution of the average fitness, as well as a typical time evolution of the fitness associated with the best individual in the population. Again, we see a significant difference between the average fitness and the maximum fitness, indicating that there is no loss of diversity in the population. Two other aspects deserve special attention: 1) again, the convergence was quite fast at

the beginning; 2) the “refinement” stage was more arduous due to the peculiar character of the second channel. Note that in the refinement stage, the value of the fitness remains constant for over 2,000 iterations. A possible explanation for this undesirable behavior is an insufficient selective pressure associated with the loss of good individuals.

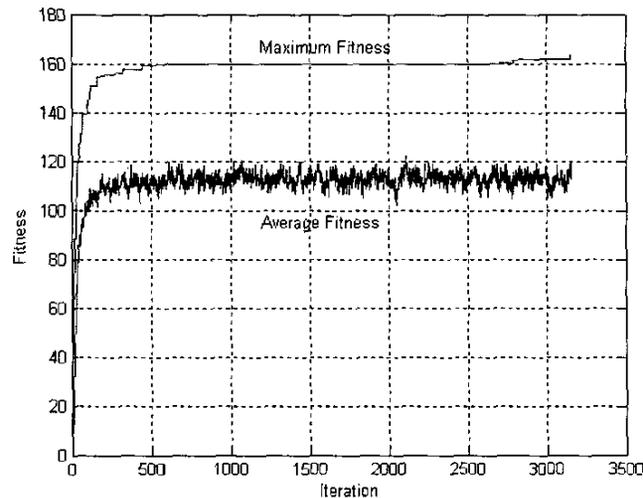


Figure 2. Evolution of the average fitness and of the fitness of the best individual for channel C2.

6. CONCLUSIONS

In this paper, we proposed a GA for blind ML detection. The proposed algorithm works directly on the transmitted sequence, bypassing the Viterbi algorithm and direct channel estimation. Thus, the search strategy is based on a low-complexity structure that can be used even for channels with long impulse responses. The receiver presented a remarkable performance under the simulation scenarios considered in this paper, where it was able to correctly detect all transmitted sequences when no noise was present. Furthermore, the proposed algorithm was able to detect the transmitted sequence in a scenario where a very short sequence was transmitted. This is in contrast to other traditional blind equalization techniques, which require hundreds or even thousands of symbols before they can accurately equalize the channel.

In spite of its remarkable performance, the proposed GA can still be improved. In particular, the GA showed a fast improvement of the fitness in the initial stages, but a very slow improvement in the final stages. Overall, the convergence of the algorithm is rather slow. Devising a better strategy for the refinement of the popu-

lation in the final stages, as well as a strategy to speed up convergence, are the object of current investigation. Finally, the use of the proposed GA for systems with long impulse responses, a short training sequence, higher-order modulation or error-control coding is also being investigated.

ACKNOWLEDGEMENTS

The authors thank FAPESP and CNPq for their financial support.

REFERENCES

- [1] J. R. Barry, E. A. Lee and D. G. Messerschmitt, *Digital Communication*, Third Edition, Kluwer Academic Publishers, 2003.
- [2] J. Ayadi, E. de Carvalho and D. T. M. Slock, "Blind and Semi-Blind Maximum-Likelihood Methods for FIR Multichannel Identification," *IEEE ICASSP*, vol. 6, pp. 3185 -3188, 1998.
- [3] S. Haykin, ed, *Blind Deconvolution*, Prentice-Hall, 1994.
- [4] Davis, L. Handbook of Genetic Algorithms. Van Nostrand Reinhold, 1991.
- [5] S. Chen and Y. Wu, "Maximum Likelihood Joint Channel and Data Estimation Using Genetic Algorithms," *IEEE Trans. Signal Proc.*, vol. 46, no. 5, pp. 1469-1473, May 1998.
- [6] R.A. Horn and C.R. Johnson *Matrix Analysis*, Cambridge University Press, 1985.
- [7] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum Likelihood from Incomplete Data Via the EM Algorithm", *Journal of the Royal Statistics Society*, vol. 39, no. 1, pp 1-38, 1977.