Applied Soft Computing 49 (2016) 561–569

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Three Natural Computation methods for joint channel estimation and symbol detection in multiuser communications

Luis M. San-José-Revuelta*, Juan Ignacio Arribas

Department of Signal Theory and Communications, University of Valladolid, Spain

ARTICLE INFO

Article history:
Received 14 May 2015
Received in revised form 16 August 2016
Accepted 18 August 2016
Available online 23 August 2016

Keywords:
Natural Computation
Genetic algorithm
Tabu search
Simulated quenching
Multiuser detection
DS/CDMA
Population diversity

ABSTRACT

This paper studies three of the most important optimization algorithms belonging to Natural Computation (NC): genetic algorithm (GA), tabu search (TS) and simulated quenching (SQ). A concise overview of these methods, including their fundamentals, drawbacks and comparison, is described in the first half of the paper. Our work is particularized and focused on a specific application: joint channel estimation and symbol detection in a Direct-Sequence/Code-Division Multiple-Access (DS/CDMA) multiuser communications scenario; therefore, its channel model is described and the three methods are explained and particularized for solving this. Important issues such as suboptimal convergence, cycling search or control of the population diversity have deserved special attention. Several numerical simulations analyze the performance of these three methods, showing, as well, comparative results with well-known classical algorithms such as the Minimum Mean Square Error estimator (MMSE), the Matched Filter (MF) or Radial Basis Function (RBF)-based detection schemes. As a consequence, the three proposed methods would allow transmission at higher data rates over channels under more severe fading and interference conditions. Simulations show that our proposals require less computational load in most cases. For instance, the proposed GA saves about 73% of time with respect to the standard GA. Besides, when the number of active users doubles from 10 to 20, the complexity of the proposed GA increases by a factor of 8.33, in contrast to 32 for the optimum maximum likelihood detector. The load of TS and SQ is around 15–25% higher than that of the proposed GA.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

Natural Computation-based methods (NCMs, in the following) have been extensively studied during last decades. Since the seminar work of Holland [1], many scientists have studied conceptually, heuristically and quantitatively a wide range of algorithms. Nowadays, there exist many families or branches of NCMs. Frequently, many methods cannot be uniquely assigned to a single class of NCM since they gather interesting properties proper of several different procedures. On the other hand, many applications of these methods have been described in technical literature: communications, control, image processing or electronics, just to name a few. Theoretical descriptions of these algorithms, and even the comparative study and the application of one (most often) or a couple of them to a specific problem, have been studied in many articles (see [2] for instance). Our main aim focuses on comparatively describe the application of three different NCMs in fair terms: Genetic algorithm (GA), Tabu search (TS) and Simulated Quenching (SQ) when applied to a key communications problem, multiuser detection (MUD) in a digital communication channel, where a mobile digital radio channel is shared by U users who simultaneously transmit digital symbols belonging to an M-ary symbol source alphabet.

The task of choosing the different methods to be compared is an open issue, since there exist many possibilities as previously mentioned. We have chosen three classical and well-known methods (GAs, TS and SQ) so that comparisons with other papers can be made easier.

Let us briefly describe these methods. SQ belongs to the branch of Simulated Annealing (SA) related methods. SA is a common algorithm for approaching complex optimization tasks. It was first proposed in [3] and [4], and it has been successfully used for solving several technical problems. The original SA scheme is a generalization of the local search algorithm, where in each iteration of the algorithm a neighbour of each current solution is selected randomly. The new potential solution will replace the current one if cost remains equal or decreases. Its main drawbacks are: (i) possibility of convergence to suboptimal solutions, and (ii) large

* Corresponding author.
E-mail addresses: lsanjose@tel.uva.es (L.M. San-José-Revuelta), jarribas@tel.uva.es (J.I. Arribas).

http://dx.doi.org/10.1016/j.asoc.2016.08.034
1568-4946/© 2016 Elsevier B.V. All rights reserved.

processing time that leads to too slow implementations. SQ reduces computational load by decreasing a parameter called *system temperature*, though, in the process, the convergence to the global optimum is notably limited [5]. Besides, the proposed SQ method, selectively allows some jumps to potential solutions with a higher cost in accordance to the Metropolis rule [6]. However, [7] points out that the main handicap they found was to avoid local minima (by a misplaced transition). This problem can be efficiently addressed with another NCM: GAs. These strategies constitute an efficient alternative for solving highly nonlinear optimization problems since they successfully use exploitative and explorative search in order to avoid local convergence. The particularities of the GA presented in this work are, principally: a reduced computational load, the ability of convergence to quasi-optimal solutions, and the on-line fine-tuning of the genetic operators depending on the diversity of the population, which is quantified using the Shannon entropy. This strategy leads to a robust and flexible algorithm in accordance to the philosophy of GAs.

We have chosen the multiuser detector in DS/CDMA communications as our problem of interest since it is a well-known and widely studied problem, making easier the comparisons with other standard and deterministic receivers. In fact, DS/CDMA is used in communications systems such as IS-95, CDMA2000, FOMA, UMTS, LTE (4G) and is expected to be used also in 5G at least in combination with other techniques, in WiFi data transmission and in some GPS systems.

As interesting previous approaches of NCMs to solve the MUD problem, we can cite the following: a multiuser detector based on a GA was presented in [8]. The scenario is a synchronous DS/CDMA scheme and the algorithm demands good initial estimates of the transmitted data symbols. The asynchronous case was studied in [9], where the interference of the surrounding bits of the remaining users is considered. This GA estimates both the desired bits as well as the edge bits. In 2000, [10] used an algorithm of local search before executing the GA; the performance was close to the single-user limit. The work by Ergun et al. incorporates a multistage detector as a segment of the GA-based detector for improving the convergence rate [11]. More recent approaches include [12,13], that consider a GA-based MUD in frequency selective Rayleigh fading channels, and the method proposed in [14], that studies the properties of a multiuser detector in terms of probability of error and near-far effect resistance. On the other hand, the channel response estimation is specifically addressed in [15], and more computationally efficient algorithms are proposed, for instance, in [16], where a redundancy saving strategy is suggested. Other interesting recent approaches for multiuser detection with GAs include [17–23]. Several earlier references are cited in [24]. In the last decade, tabu search methods for MUD problems have been proposed in [25–27] or in [28], which proposes a user selection scheme for MIMO-CDMA systems. Relevant SQ-SA strategies for MUD have been proposed in [27,29–31], the later is an GA-SA hybrid proposal, while in [30] an SA algorithm is proposed for solving the MUD problem in CDMA, considering both the synchronous and asynchronous cases.

The remaining of the paper is structured as follows:

- Section 2 is devoted to the description of the application problem: joint channel estimation and symbol detection in digital synchronous multiuser communications. Since this is a crucial and well-known problem, it allows to analyze the performance of the proposed NCMs, not only in a comparative analysis among them, but also in contrast to several traditional methods.
- Section 3 presents the basic principles of the NCMs, mainly GA, TS and SQ. In order to facilitate comparison, concise comments to simulated annealing (SA) are outlined, as well. For the sake of brevity, and in order to make the paper more readable, TS and SQ will be explained more concisely, only detailing the main differences with respect to the GA case. Each subsection also shows the specific implementations for the multiuser detection application.
- Finally, Section 4 describes the numerical results, underlining the comparison between GA, TS and SQ in fair terms: their drawbacks and advantages are here explained. For the sake of comparison, some traditional detectors – matched filter, correlator, Radial Basis Function (RBF) networks – have also been included here since these methods have been frequently used in papers approaching this problem, thus allowing an easy comparison with many existing MUD methods.

2. Multiuser detection for DS/CDMA communications

2.1. Problem description

During the last two decades, wireless communications have shown large growth rates per year in several countries. DS/CDMA has been widely studied as telecommunication companies wish to exploit to the maximum the available frequency bands [32,33]. Since DS/CDMA offers high transmission rates, intersymbol interference (ISI) effects become notable and, considered simultaneously with multi-access interference (MAI), they constitute the two most important handicaps to system specifications [32]. If these interferences are not properly controlled, they can lead to drastic degradation of reception quality. Several strategies have been analyzed in order to minimize these interferences, such as power control and optimization of the users' codewords. Traditional schemes relying on a bank of matched filters lead to good performance only when all received codewords are orthogonal. However, real systems rarely present this property and their behaviour is degraded. This degradation becomes more significant if the scenario presents near-far effects. Verdú showed that this drawback was solved if all the users data sequences were jointly-extracted [33]. However, the complexity of this optimum detector based on the maximum likelihood criterion increases exponentially with the number of active users. As a consequence, suboptimal detectors have been widely developed. A considerable amount of these procedures uses Natural Computation methods as previously mentioned in Section 1.

2.2. Description of the DS/CDMA scenario

The following discrete-time signal and channel model has been considered: a BPSK (Binary Phase Shift Keying) communication system with U active users, each of them using a normalized modulation codeword from set $\{s_i(t)\}_{i=1}^{U}$, and transmitting through a flat-fading frequency-nonselective Rayleigh channel, with a zero-mean white Gaussian noise. Perfect synchronization of all signals is assumed. Suppose that the ith user of the system transmits a sequence $d_i(n)$ of F (frame duration) statistically-independent symbols that modulates waveform $s_i(t)$, with the result that the spectrum width is spread by a factor N (known as *processing gain*). Consequently, the signal transmitted by user i is

$$x_i(t) = \sum_{n=0}^{F-1} d_i(n) s_i(t - nT) \quad (1)$$

where T is the data symbol period, $d_i(n)$ are the users' data symbols, and the codewords waveforms are generated as

$$s_i(t) = \sum_{\ell=0}^{N-1} s_{i,\ell}\,\psi(t - \ell T_c) \quad (2)$$

where $s_i = (s_{i,0}, \ldots, s_{i,N-1})^T$ represents the codeword of the ith user, $T_c = T/N$ is the chip period and $\psi(t)$ denotes a chip pulse of normalized energy. Since synchronous transmission is assumed, the signal at the input antenna of the receiver is

$$r(t) = \sum_{i=1}^{U} r_i(t) + g(t) \qquad 0 \le t \le T_f \tag{3}$$

with

$$r_i(t) = \sqrt{E_i} \sum_{n=0}^{F-1} b_i(n) d_i(n) s_i(t - nT) \tag{4}$$

where T_f stands for the duration of the frame, $g(t)$ denotes a zero-mean complex, additive, white and Gaussian (AWG) noise uncorrelated with data symbols $d_i(n)$, E_i stands for the bit energy of user i, and $b_i(n)$ is the complex channel impulse response (CIR) coefficient for user i. We have considered a time-variant channel, whose model is defined in [18], where channel coefficients $b_i(n)$ vary depending on a predefined Doppler frequency, f_d, as

$$b_i(n+1) = a \cdot b_i(n) + \nu \tag{5}$$

with $a = \exp(-2\pi f_d T)$ and ν being a zero-mean white Gaussian variable. The parameters to be estimated in Eq. (4) are the users' data symbols as well as the CIR coefficients of user i. At the receiver, a first block consisting in a bank of filters that are matched to the users' codewords samples the signal every T seconds (see Fig. 1).

The multiuser detector must estimate the transmitted data vector $\mathbf{d}(n) = [d_1(n), \ldots, d_U(n)]^T$, which contains the U transmitted symbols in current period. Following ideas in [18] this task can be seen as a maximization problem, where the matched filters' output is obtained as

$$\mathbf{z}(n) = [z_1(n), \ldots, z_U(n)]^T = \mathbf{RB}(n)\mathbf{Ed}(n) + \mathbf{g} \tag{6}$$

where $\mathbf{R}$ is the $U \times U$ user codeword cross-correlation matrix, $\mathbf{B}(n) = \mathrm{diag}(b_1(n), \ldots, b_U(n))$, $\mathbf{E} = \mathrm{diag}(\sqrt{E_1}, \ldots, \sqrt{E_U})$, $\mathbf{d}(n) = [d_1(n), \ldots, d_U(n)]$ and $\mathbf{g} = [g_1(n), \ldots, g_U(n)]$.

Considering vector $\mathbf{z}$, it is not difficult to show that the log-likelihood pdf conditioned on both the channel impulse response matrix $\mathbf{B}(n)$ and the transmitted symbols vector $\mathbf{d}(n)$, is [34]

$$\begin{aligned} &\mathcal{L}(\mathbf{B}(n), \mathbf{d}(n)) \\ &\quad = 2\Re\left\{\mathbf{d}(n)^T \mathbf{E}[\mathbf{B}(n)]^* \mathbf{z}(n)\right\} - \mathbf{d}(n)^T \mathbf{EB}(n)\mathbf{R}[\mathbf{B}(n)]^* \mathbf{Ed}(n) \end{aligned} \tag{7}$$

where "*" stands for the complex conjugate operator. Therefore, the optimal estimates of the matrix of channel gains and the vector of data symbols are obtained as

$$(\hat{\mathbf{B}(n)}, \hat{\mathbf{d}(n)}) = arg \max_{\mathbf{B}(n), \mathbf{d}(n)} \left\{\mathcal{L}(\mathbf{B}(n), \mathbf{d}(n))\right\} \tag{8}$$

subject to the specific time variation model of the channel impulse response coefficients $\mathbf{B}$. The channel fading will be considered to be independent between users, as well as to have a sufficiently low time variation that allows to consider the channel impulse response coefficients to be constant during each symbol period.

In Section 3, three different multiuser detectors based on the GA/TS/SQ algorithms are described.

3. NCMs for multiuser detection

Since most readers of this journal are familiarized with NCMs, the description of these methods is concise and only the particular specific characteristics of our proposals are next described.

3.1. Genetic algorithms

3.1.1. Basic concepts, genetic operators and elitism

Genetic algorithms are inspired by the Evolution Theory of Darwin. Potential solutions to the problem being solved are encoded on a data structure (known as *individual* or *chromosome*) and a set of genetic operators is applied to a selected group of chromosomes in order to preserve critical information. Members of the new population will probably be better solution estimates than the individuals of the previous generation (the Schema Theory explains how the overall fitness of the population increases along iterations [35]). Those individuals selected to form new chromosomes are stochastically selected according to their fitness (or *aptitude*).

The description of the standard GA is quite general and can be found in many references (see [36] for instance). First, an initial population must be generated. Then, every individual is evaluated according to its fitness and the genetic operators, mainly crossover and mutation, are applied to those chromosomes selected with an aptitude-based stochastic selection scheme. Thus, the probability of selecting individual i (which has fitness λ_i) at iteration k is, $P_i(k) = \lambda_i(k)/\sum_{j=1}^{n_p} \lambda_i(k)$. This concept can be easily implemented using the *roulette wheel* selection method [36]. The basic genetic operators (mutation and crossover) are next applied to selected individuals.

In our application, mutation is applied with an initial probability of $p_m(0) \in [0.02 - 0.05]$, a value that has proven to be a nice trade-off between explorative and exploitative search [37,38].

The proposed GA also implements an elitism strategy, which means that the fittest individuals are directly selected, in order to improve performance. In our simulations, the elite is mutated with probability $p_{m,e} = 0.25 p_m$, and the crossover operator is not applied. The algorithm is repeated a fixed number of generations. When the GA finishes, the chromosome with the highest fitness represents the problem solution. In our case, the estimate of both the channel coefficients $\mathbf{B}$, and the symbols vector $\mathbf{d}$.

3.1.2. Diversity control using entropy dependant genetic operators

An important particularity of the proposed GA is the introduction of schemes to fine-tune the genetic operators for achieving and maintaining diversity within the population. Standard genetic algorithms tend to converge to suboptimal solutions, mainly due to a selection strongly dependent on fitness [39]. When this situation occurs the population is filled with the best individuals, resulting in suboptimal estimates. On the other hand, the most important drawback of standard GAs is their large computational burden, mainly due to selection, mutation and crossover application, as well as fitness evaluation. Standard population sizes n_p are normally about 1–5 hundreds. For instance, [39] uses populations sizes in the order of 400 individuals. Following the ideas shown in [24] the method used in this work requires a much smaller population size (10–25 individuals). As already developed in [24,40] the mutation and crossover operators depend on the Shannon entropy of the population fitness, which is obtained as

$$\mathcal{H}(\mathcal{P}[k]) = -\sum_{i=1}^{n_p} \lambda_i^*(k) \log \lambda_i^*(k) \tag{9}$$

with $\lambda_i^*(k)$ being the normalized fitness of individual u_i, i.e.,

$$\lambda_i^*(k) = \frac{\lambda_i(k)}{\sum_{i=1}^{n_p} \lambda_i(k)} \tag{10}$$

The purpose of this is to on-line adapt the sense of the search (explorative *vs* exploitative) by making use of the diversity of the population found at each phase of the convergence cycle.

Computational requirements of this GA are considerably lower than those of the standard GA since the crossover operator is scarcely applied and the diversity control mechanism allows to operate efficiently with reduced population sizes.

Notice that entropy defined in Eq. (9) will have a large value in two situations: (i) during the initial iterations of the GA, since all

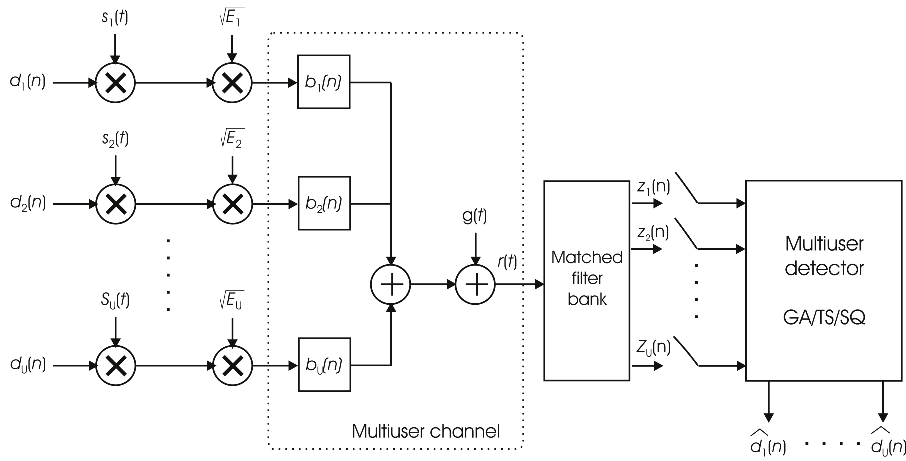

Fig. 1. Multiuser DS/CDMA system model with U active users.

the individuals represent poor estimates of the real values, and (ii) during the last iterations of the GA, since most of the individuals are supposed to be close to the optimum solution (of course, if the GA has converged properly). In order to discriminate between these two situations, the mean aptitude of the population is evaluated along 5 iterations and this parameter is monitored during the GA execution.

3.1.3. Coding and fitness computation

In the proposed algorithm, the fitness of each individual is obtained using the log-likelihood pdf $\mathcal{L}$ defined in Eq. (7), since both users data and channel coefficients are encoded into $\mathbf{u}_i$ as:

$$\begin{aligned}\mathbf{u}_i &= [\mathbf{B}_i(n,k), \mathbf{d}_i(n,k)] \\ &= [(b_{i,1}(n,k), b_{i,2}(n,k), \ldots, b_{i,U}(n,k)), (d_{i,1}(n,k), d_{i,2}(n,k), \ldots, d_{i,U}(n,k))]\end{aligned} \quad (11)$$

where parameters n, i, k and U denote the nth symbol period, the ith individual of the population, the kth GA iteration and the number of system users U, respectively. On the right-most part of the CIR (channel impulse response) coding, vector $\mathbf{d}_i(n, k)$ contains the symbols transmitted by the U users. On the other hand, each channel impulse response coefficient is binary encoded into a vector of 22 bits whose encoding is as follows: both the real and the imaginary part of each coefficient $b_i(n, k)$ are represented with a vector of 1 + 10 bits. The first bit determines the sign of the coefficient and the other 10 bits encode the magnitude. Since 10 bits represent numbers from 0 to 1023, a mapping to the interval [0,1] is required (normalization step). On the other hand, mutation is applied over $U+1$ randomly generated positions: one mutation within the 22 bits of each coefficient $b_{i,j}(n, k)$, $1 \le j \le U$, in Eq. (10), and another one within the users' symbols estimate $d_i(n, k)$ in Eq. (10). Note that one GA is run per symbol interval. Therefore, the impulse response of the channel estimate is kept from one symbol period to the next. This is in accordance to the assumption of low fading channel, where minor variations of the CIR coefficients are experimented within each symbol period. The part corresponding to $\mathbf{d}_i(n, k)$ of $\mathbf{u}_i$ in Eq. (11) is randomly initialized at $k = 0$, while $\mathbf{B}_i(n, k)$ is initialized using the final estimates from previous iteration since the assumption of low fading allows to consider the CIR to be almost constant between consecutive symbol periods.

3.2. Tabu search

3.2.1. Introduction

Tabu search was first developed and studied in [41], although the main concepts were simultaneously outlined in [42], as well. Important posterior formalization tasks can be found in [43,44]. TS was originally proposed to provide an intelligent search so as to avoid the frequently too slow and complex existing metaheuristic algorithms. We will just briefly describe its main concepts and properties, so that the reader can appreciate the differences with respect to GAs. Detailed descriptions can be found in [45,46], for instance.

TS is quite similar to descent methods, since it selects the best solution estimate at each iteration, but TS also allows jumps to a near solution even when its cost is higher. TS performs a systematic and intelligent search, that is specially guided under difficult situations by making use of *memory* structures that allow to efficiently use the past information of the search. TS restrictions can be relaxed under specific circumstances (this is known as *aspiration criteria*).

Memory works as follows: suppose that the TS algorithm keeps in memory a selective history H of previously visited points of the solutions space. TS substitutes the classical neighborhood structure $N(s)$ with a modified one denoted as $N(H, s)$. This way, history determines the set of solutions that can be reached from current position. Besides, when the fitness evaluation is complex, or when $N(s)$ has a huge number of points, it is important to select the most interesting subset of $N(s)$, which will depend on history H, as well.

Frequently, TS uses a memory that is based on *attributes*. An attribute is associated to a move and can be related to any parameter that changes as a consequence of that move. A single move can have several attributes associated to it. Sometimes, attributes corresponding to points in memory are used to set *Tabu restrictions* that avoid moves that could destroy recent changes that are represented by those attributes, avoiding, this way, revisiting solutions.

At a specific solution candidate, an attribute can be either *active* or *inactive*, and the condition to be verified for the attribute is known as *Tabu state*. A certain move can be forbidden –Tabu– due to a restriction defined over any set of its attributes (of course, whenever these attributes are active).

The size of the Tabu list and the time during which an attribute is Tabu are important parameters of the algorithm. Taillard has proved that a random size of the Tabu list gives more robust search schemes than those with a constant size [47]. Later on, Battiti improved this idea with an intelligent strategy for varying the size of the list [48]. The goal of this size adjustment is to reinforce the search in its weakest aspects, i.e., suboptimal solutions, cycling, and chaotic attractors.

The aspiration criteria parameters are introduced in the TS algorithm for determining when to readmit a move classified as Tabu. A new solution will be accepted if its *aspiration level* is better than a predefined threshold.

For the comparative simulations carried out in this work, a standard Tabu search algorithm was implemented. Every point of the solutions space is encoded as described in Section 3.1.3 and the corresponding fitness value is obtained using Eq. (7). When a cost measure is required, an inversely proportional value is taken.

3.3. *Simulated quenching*

As previously mentioned, SQ is a methodology proposed to speed up standard SA algorithms when applied to solve complex optimization problems. The original SA method pretends to simulate the physical annealing process found in nature. To avoid convergence to suboptimal solutions, SQ sporadically allows moves to potential solutions of higher cost using the Metropolis rule [6]. This criterion states that, if α_1 and α_2 are the two solution candidates to choose from, then the SQ method will select α_2 with probability min{1,exp(-[$C(\alpha_2)$-$C(\alpha_1)$]/t)}, with $t>0$ and $t\to 0$ during the execution of the algorithm, and $C(\alpha)$ represents the cost of the solution estimate α. Note that probability decreases as $C(\alpha_2)-C(\alpha_1)$ increases and/or time t decreases, and that transitions that imply a reduction of the cost will always be accepted.

The description of the configuration space alone requires an exponential time, and, in real implementations, simplifications are required. The solution is to define a *cooling schedule* that must define the four following algorithm parameters: (i) the start condition, i.e. temperature at iteration $t=0$, (ii) the scheme for decreasing the temperature, (iii) the condition for equilibrium, and (iv) the termination criterion, i.e. the final temperature.

The initial temperature must be set high enough so that most of the proposed position changes satisfy the Metropolis rule. Thus, during the first iterations of the procedure, an explorative search of the solutions space is performed. Afterwards, the number of accepted transitions decreases as temperature tends to 0. When $t\approx 0$, no more transitions will be performed and the algorithm may finish. The solution of the problem is given by the final configuration of the algorithm.

For the experiments carried out in next section, the standard SQ algorithm is implemented. Every point of the solutions space is encoded as described in Section 3.1.3 and the corresponding fitness value is obtained using Eq. (7).

Our numerical experiments worked fine with acceptance ratios around 0.55–0.65. Temperature decreases with the following restriction [49,50]: the decrease in the mean cost between two consecutive temperatures t_1 and t_2 should not be larger than the standard deviation, σ of the cost (on level t_1). This rule can be expressed as

$$t_2 = t_1 \exp\left(-\frac{\lambda t}{\sigma}\right) \tag{12}$$

with $\lambda \in [0, 1]$.

Temperature is initially set in order to allow a predefined transitions' acceptance ratio: First, temperature is set to 0, and then it is successively increased until the specified acceptance ratio is achieved.

The same simple strategies for generating the neighborhood used in [7] are used in our work, but with probabilities adjusted/optimized to our problem. These strategies are: (i) *single flip*, that mimics the GAs mutation operator, and (ii) *flip-flop*. See [7] for a detailed description.

4. Numerical simulations

For the simulations carried out in this section, synchronous transmission, AWG noise, a rectangular chip pulse $\psi(t)$ – see Eq. (2), Gold-type codewords and a Doppler frequency $f_d=1000$ rad/s have been considered. Simulations are run using BPSK modulation,

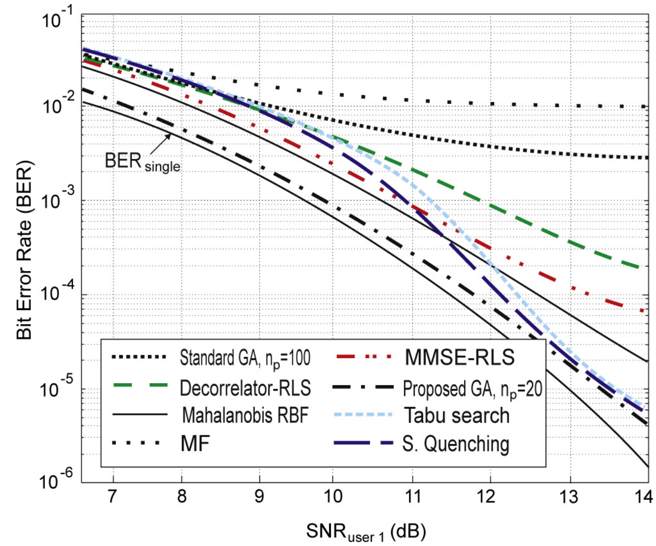

Fig. 2. Bit Error Rate performance as a function of the SNR (dB) for several multiuser detectors. Near-far distortion: $E_k/E_1 = 4$ dB for $k\geq 1$.

U=10 active users and a processing gain with $N=31$ chips. The GA implements a binary tournament algorithm for selection, uniform crossover and elitism. The initial probabilities of the genetic operators are: $p_c(0)=0.05$ and $p_m(0)=0.1$, which have been heuristically determined. Memory and control of tabu moves are implemented in the TS algorithm, as well. On the other hand, SQ uses a cooling scheme where the rule for decreasing temperature is given by Eq. (11), i.e. the criterion developed in [49,50]. In this case, new points are accepted using the Boltzmann probability distribution as acceptance function [51]. 50 runs were averaged to obtain the plots.

4.1. *Bit Error Rate* vs *Signal-to-Noise Ratio (BER* vs *SNR)*

Fig. 2 plots the probability of error performance of user No. 1 (user of interest) versus its signal quality, measured in terms of SNR=E_1/N_0, with a near-far effect of 4 dB (i.e. $E_k/E_1=4$ dB, $2\leq k\leq U$), for different detection schemes. In this case, the GA uses a population size of $n_p=20$ individuals. Table 1 shows mean and standard deviation (stdev) values corresponding to Fig. 2 plots. In general, it can be observed how stdev decreases as convergence improves, i.e., as SNR becomes higher, and, on the other hand, methods with better convergence – i.e. lower BER for a fixed SNR – show slightly lower stdev.

Let us have a look first to the comparison of the proposed GA to the standard GA. Figure plots show how the performance of our GA is close to the optimum single user detector limit – a lower limit for the multiuser case is estimated by simulating the bit error rate of the single user detector without interfering users [32] – while the standard genetic algorithm saturates at high signal-to-noise ratios (SNRs) even when both the population size and the number of iterations are increased – this fact is in accordance to [17].

Numerical experiments show that, for equivalent BERs, our GA requires about 12–15% of the population used in the basic algorithm, and a number of generations about 15–20%, for achieving comparable results, saving, this way, about 73% of time.

This conclusion is similar to that obtained in [17], where the GA only estimates the vector of transmitted data symbols (see Section 3.1.3).

With the aim of comparing the GA with TS and SQ, we have implemented TS and SQ keeping a similar computational load (which is measured in terms of basic float operations) to that of the

Table 1
Mean ± stdev values of the Bit Error Rate as a function of the SNR (dB) for several multiuser detectors. 50 runs. Near-far distortion: $E_k/E_1 = 4$ dB for $k \geq 1$. Notation: Xe-Y $= X \times 10^{-Y}$.

	SNR_{user1} (dB)							
	7	8	9	10	11	12	13	14
Standard GA	0.037 ± 0.013	0.018 ± 0.004	0.011 ± 0.003	0.007 ± 0.002	0.005 ± 0.002	0.0038 ± 0.002	0.003 ± 0.0015	0.0028 ± 0.0010
Decorrelator RLS	0.028 ± 0.006	0.018 ± 0.003	0.009 ± 0.002	0.005 ± 0.001	0.002 ± 0.0008	0.00085 ± 0.00015	0.00037 ± 0.00011	$0.0002 \pm 8e-5$
Mahalanobis RBF	0.020 ± 0.005	0.011 ± 0.002	0.0048 ± 0.0018	0.0018 ± 0.0012	0.0006 ± 0.0002	$0.0002 \pm 1e-5$	$5e-5 \pm 1e-5$	$2e-5 \pm 0.2e-5$
Matched filter	0.028 ± 0.011	0.023 ± 0.011	0.017 ± 0.008	0.012 ± 0.005	0.011 ± 0.002	0.011 ± 0.0015	0.010 ± 0.0013	$0.098 \pm 1.2e-3$
MMSE-RLS	0.025 ± 0.011	0.012 ± 0.006	0.006 ± 0.001	0.0026 ± 0.0008	0.0009 ± 0.0002	$0.0003 \pm 8e-5$	$0.0001 \pm 7e-5$	$6.5e-5 \pm 2.1e-5$
Proposed GA	0.0105 ± 0.010	0.0056 ± 0.0012	0.00205 ± 0.0008	0.0004 ± 0.0007	$2.7e-4 \pm 1.0e-4$	$7.5e-5 \pm 1.5e-5$	$1.7e-5 \pm 0.9e-6$	$4e-6 \pm 1e-6$
Tabu search	0.023 ± 0.005	0.020 ± 0.003	0.0098 ± 0.002	0.005 ± 0.0008	0.0013 ± 0.0006	$0.0002 \pm 2e-5$	$2.8e-5 \pm 2.3e-6$	$6.1e-6 \pm 1.5e-6$
Simulated quenching	0.032 ± 0.015	0.0175 ± 0.011	0.009 ± 0.0015	0.0038 ± 0.0007	0.00085 ± 0.0001	$0.00012 \pm 1.4e-4$	$2e-5 \pm 8e-6$	$5.6e-6 \pm 1e-6$

GA. This is achieved basically by means of adjusting the number of iterations of TS and SQ. As a first simple and general rule, it can be said that TS and SQ require a number of iterations that is increased by a factor of n_p with respect to those required by the simple GA, with n_p being the population size of the GA. Under these conditions, it can be seen that the behaviours of GA/TS/SQ are similar when the energy of user 1 is higher enough than those of the remaining users. This situation is related to low and medium complexity problems, and it can be considered for $\text{SNR}_{\text{user1}} > 12$ dB. Below this threshold, the GA is capable of extracting the information concerning the user of interest, while both TS and SQ begin to have problems for satisfactory estimation. This is due to the more efficient search performed by the GA thanks to the control of the population diversity using the fitness entropy described in Section 3.1.2

Error probabilities corresponding to other well-known detectors, such as the Minimum Mean Square Error (MMSE) detector, the decorrelator [52], the one based on matched filters (MF), and a Radial Basis Function (RBF) network have been plotted for comparison purposes, as well. The RBF-based detector is taken from [53] and has a kernel function implemented with the Mahalanobis distance (the Euclidean distance was also implemented but led to slightly worse results). Those experiments carried out with both the decorrelator and the MMSE receivers, were obtained with a channel impulse response estimation algorithm whose description can be found in [54]. Specifically, the numerical simulations here shown use the Recursive Least Squares (RLS) method.

Analyzing the BER plots shown in Fig. 2, we see that simple methods, such as the standard GA or the MF detector, are not capable of proper estimation even when $\text{SNR}_{\text{user1}}$ is high. On the other hand, nonlinear detectors implemented with RLS estimation (decorrelator, MMSE and RBF) show an intermediate performance between the standard GA or the MF detectors, and that obtained with TS, SQ and the proposed GA.

4.2. Comparison with other GA-based strategies

In this section, the characteristics of the proposed GA are compared to those of other GA-based methods, specifically, the algorithm proposed in [17] – where the channel coefficients are considered to be known, the GA in [18], and the standard GA. Numerical results are summarized in Table 2.

In case the channel response is known, [17] shows that his method requires about 33% of the population size needed by the standard GA (with an *a priori* known channel response), but extremely fewer generations. In our experiments, the standard GA required about 130–250 generations for proper convergence, while our proposed GA uses only 18–25. The number of fitness evaluations depends on both the number of generations and the population size. Considering this parameter it can be seen how the proposed GA requires only about 5% of the operations required by the standard GA. On the other hand, the approaches in [17,18] show intermediate results.

Table 2
Comparison between the main parameters and capabilities of several GA-based multiuser detectors ($U = 10$). Proposed GA implements dynamic genetic operators.

	Standard GA	GA by [17]	GA by [18]	Proposed GA
Population size, n_p	60–80	25	40	15
No. generations, n_g	130–250	20	10	18–25
Fitness evaluation	5000	520	400	260
Convergence	Poor (∼75%)	Fair (∼90%)	Fair (∼90%)	Good (∼97%)
Channel tracking	Poor	Poor	Poor	Good
Comput. load	High	Moderate	Moderate	Low

Table 3
Population size (n_p) and number of generations (n_g) required for several values of parameter U (active users) for the proposed GA, TS and SQ algorithms. Values corresponding to [18] are included for comparison purposes, as well.

No. of users (U)	Population size (n_p)			Generations (n_g)		
	GA	GA-H [18]	TS-SQ	GA	GA-H [18]	TS-SQ
10	15	40	1	20	10	250
15	45	75	1	20	10	650
20	125	160	1	20	10	2000

TS and SQ have not been included in this table since they do not maintain concurrently a population of potential solutions but a unique solution estimate at each iteration.

4.3. Computational complexity *vs* number of active users

In this section we analyze how the number of active system users U modifies the efficiency of the analyzed detectors. Specifically, the computational load as a function of U is analyzed in Table 3. The GA developed in [18] is included for comparison purposes, as well ("GA-H", hereafter). The two most important parameters of a GA, i.e. the number of generations, n_g, and the population size, n_p, are shown in the table for three different values of U (10, 15 and 20 users).

First, notice that a comparable performance is obtained with our GA and GA-H when the number of generations of GA-H is half the generations required by the proposed GA, while the number of individuals in our approach is 37.5%, 64% and 78% the values required with GA-H, respectively. This fact reveals that similar performance is obtained when the number of generations is doubled and the number of individuals is reduced about 50%. This result agrees with previous work where the increment of active users is solved by setting a larger number of iterations n_g and maintaining constant the population size. This interplay between parameters n_p and n_g was first studied in [17]. However, it is not obvious to quantify the interplay between both parameters since it notably depends on other factors such as U, E_k/N_0, etc. For instance, in Table 3, the configuration with $U = 20$, $n_p = 125$ and $n_g = 20$ yields similar results than the implementation with $U = 20$, $n_p = 75$ and, $n_g = 40$.

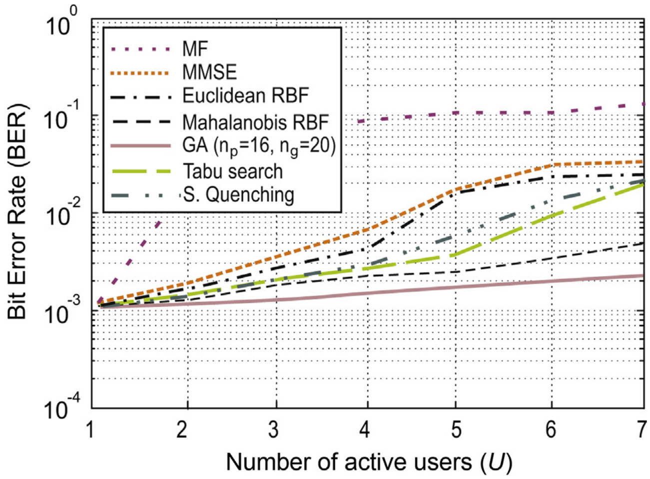

Fig. 3. Bit Error Rate *vs* number of active users U. $E_i = E_j$, $1 \le i, j \le U$, $i \neq j$.

Besides, we can compare the computational load with that of the optimum maximum likelihood detector. For $U = 10$, the ML-MUD requires $2^U = 2^{10} = 1024$ evaluations of the fitness function, while our GA is notably less complex, with a requirement of just $n_p \times n_g = 15 \times 20 = 300$ evaluations, showing a performance that is similar to that of the optimum ML detector. This result evidences that the GA complexity is not an exponential function of U. In fact, when $U = 20$ and $n_p = 125$ an almost optimal performance is achieved. This involves a load increment by 125/15=8.33. Moreover, in contrast to reduced tree-search type methods, the GA does not need any storage resources, since all the information regarding previous iterations can be removed.

Table 3 includes two additional columns for TS-SQ. Values in this case are obtained forcing a similar performance than that of the GA. This is achieved by tuning the n_g parameter as already explained in Section 4.1. Beware that, though the number of iterations in TS-SQ is really higher, the computational load does not increase proportionally, since TS and SQ execute much less operations per generation than the proposed GA. When TS and SQ implement the characteristics explained in Section 3 in order to avoid premature convergence and cycling, the final load is about 15–25% higher than that of the proposed GA. Besides, GA shows better convergence in difficult problems due to the implementation of a more efficient search scheme.

As a final remark, notice that the increment in complexity of the three methods (GA/TS/SQ) is really low in contrast to the complexity of the traditional optimum MUD, whose computational load will now be increased by a factor of 32, while the proposed methods' performance still remain close to the optimum.

4.4. Bit Error Rate vs number of active users

Next, in Fig. 3, the Bit Error Rate for various detection schemes (GA-TS-SQ, RBF, MMSE and MF) as a function of the number of active users is shown. This experiment uses identical transmitted energies: $E_i = E_j$, $1 \le i, j \le U$, $i \neq j$. The MMSE detector and the Matched Filter scheme are clearly outperformed by nonlinear RBF and our detectors. The MF detector is very sensible to multi-access interference while the MMSE detector is not stable enough for defining a good decision boundary. Like in previous section, the Mahalanobis measure in the RBF kernel function leads to some superiority over the Euclidean distance due to the asymmetric configuration of the clusters.

In Fig. 3 it can also be seen that both the GA and the Mahalanobis RBF scheme have similar behaviour. Also, notice that the RBF detector complexity is much higher. This detector uses an

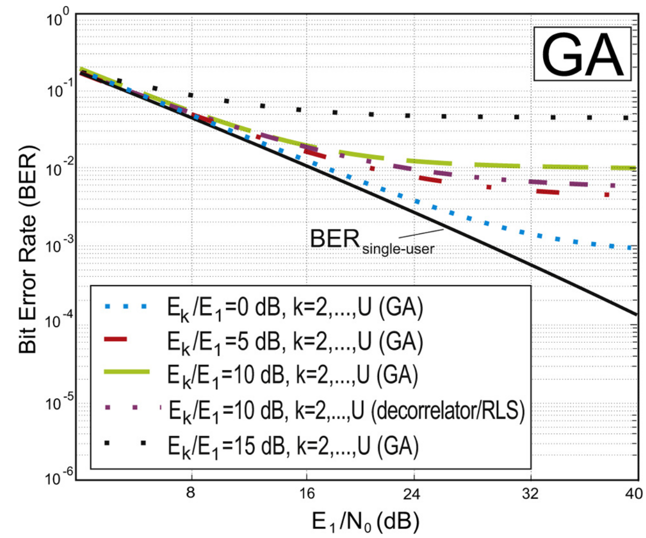

Fig. 4. Bit Error Rate versus SNR (dB) for $U = 10$ users with $E_k/E_1 = 0$, 5, 10 and 15 dB for $k = 2, \ldots, U$. User No. 1: user of interest. Proposed GA-based detector.

initial supervised period if the channel impulse response is not known and its computational load grows exponentially with U [53].

According to previous experiments, the performance of TS and SQ is close to that of the GA and RBF based detectors for a moderate number of system users. Performance degrades towards that of the MMSE detector when the problem complexity increases and multi-access interference becomes larger.

Quantitatively, the computational load is very similar when the system has few active users ($U \le 3$) for the three proposed methods. On the other hand, for higher values of U, the GA MUD certainly outperforms the other detection schemes in terms of computational complexity.

4.5. Near-far resistance

Finally, the near-far properties of the studied detectors as a function of the bit error rate (BER) of the user of interest (user No. 1) are analyzed.

The GA used in the simulations was configured with $n_p = 25$ individuals and $n_g = 30$ iterations. In order to test the near-far effect, the received energies of users $i = 2, 3, \ldots, U$, are forced to be 0, 5, 10 and 15 dB higher. In the latter case, we have also shown the BER of the decorrelator detector implemented with an RLS algorithm for parameters estimation, so as to allow direct comparisons.

The three metaheuristic detectors jointly estimate the channel impulse response and the users' data symbols. Fig. 4 shows the simulation results for the proposed GA detector. It can be seen that for $E_k/E_1 \le 10$ dB, the detector shows good near-far resistance for values $E_1/N_0 \lesssim 15$ dB. For $E_1/N_0 > 15$ dB a substantial performance downfall in terms of the BER is appreciated.

Figs. 5 and 6 show the corresponding BER plots for the TS and SQ-based multiuser detectors, respectively. In order to make simulations comparable, the three methods were implemented with cycling control as well as with the possibility of "high cost jumps" in TS/SQ so as to prevent from convergence to local minima. In the GA, this is achieved through diversity monitoring and dynamic genetic operators. Besides, the number of iterations (in GA/TS/SQ) and the population size (in the GA) were adjusted to obtain algorithms with similar computational load.

BER plots in Figs. 4–6 show that GA and SQ offer very similar near-far resistance values. Both methods are close to the theoretical limit ("BER-single-user" in the figure) up to 14 dB of near-far effect.

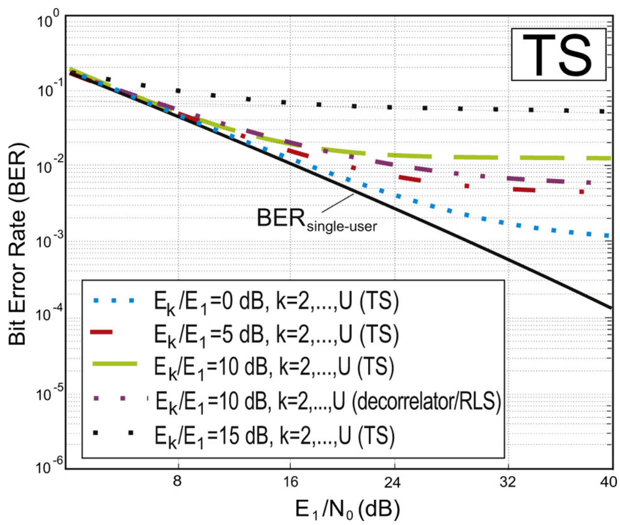

Fig. 5. Bit Error Rate versus SNR (dB) for $U = 10$ users with $E_k/E_1 = 0, 5, 10$ and 15 dB for $k = 2, \ldots, U$. User No. 1: user of interest. TS-based detector.

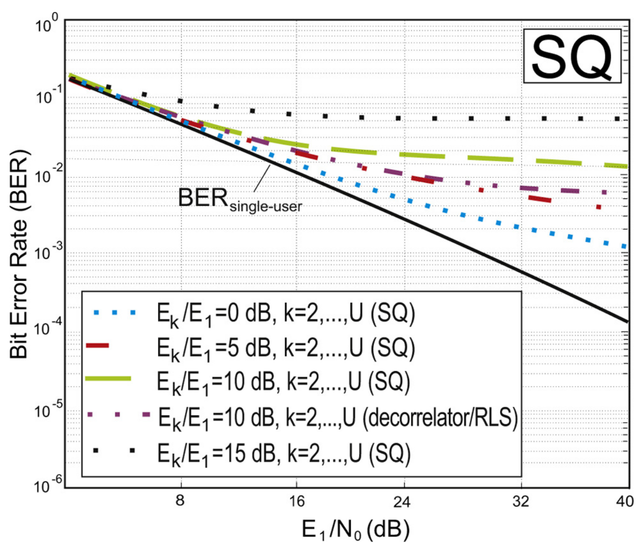

Fig. 6. Bit Error Rate versus SNR (dB) for $U = 10$ users with $E_k/E_1 = 0, 5, 10$ and 15 dB for $k = 2, \ldots, U$. User No. 1: user of interest. SQ-based detector.

On contrast, TS shows a bit poorer behaviour, since it remains near-far resistant only up to 10 dB.

Notice that, in order to allow direct comparison, in the three plots shown in this subsection, the Bit Error Rate performance of the decorrelator detector is shown for $E_k/E_1 = 10$ dB. The decorrelator detector shows some performance improvement, but a much greater computational load, as well.

5. Conclusions

This paper shows a comparative description of three important optimization methods inspired in the Natural Computation paradigm. Specifically, we have studied three metaheuristic methods: genetic algorithms, tabu search and a particular case of simulated annealing known as simulated quenching. Our work summarizes their basic concepts, differences, drawbacks and advantages. These methods are developed in order to solve one of the most common and important problems in digital communications: the joint channel estimation and symbol detection in a multiuser environment. Specific aspects such as the coding of potential solutions or the description of strategies for avoiding local minima convergence or cycling, have been addressed.

The numerical results section shows several simulations that compare the proposed GA, TS and SQ in our best knowledge fair terms, and also includes comparisons with well-known classical detectors such as the decorrelator, the matched filter or the RBF-based multiuser detectors. We have seen how these Natural Computation based methods are efficient tools for solving a complex optimization problem. The GA offers a notably better performance, specially when scenario conditions become harder (low SNR of the user of interest, a large number of interfering users, or the presence of near-far degradation). When these interferences are low or moderate, both TS and SQ exhibit a similar behaviour, which is close to the GA and to the optimum receiver. When interferences are harder or the signal energy of interest is poorer, the proposed GA shows a better performance due to its powerful search scheme that controls the population diversity by monitoring the entropy of the population fitness. Besides, genetic operators are on-line fine-tuned using this information.

The three proposed methods allow to have a cost-effective operation of existing channels at higher data rates – even in cases with more severe fading, more active users and stronger interference conditions – than previously existing detectors with, in worst case, limited additional cost increments.

References

[1] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, 1975.
[2] A. Vasan, K.S. Raju, Comparative analysis of simulated annealing, simulated quenching and genetic algorithms for optimal reservoir operation, Appl. Soft Comput. 9 (2009) 274–281.
[3] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science 220 (1983) 671–680.
[4] V. Cerny, Thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm, J. Opt. Theory Appl. 45 (1985) 41–51.
[5] L. Ingber, Simulated annealing: practice and theory, Math. Comput. Modell. 18 (1993) 29–57.
[6] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (1953) 1087–1092.
[7] M. Duque-Antón, D. Kunz, B. Ruber, Channel assignment for cellular radio using simulated annealing, IEEE Trans. Veh. Technol. 42 (1993) 14–21.
[8] M.J. Juntti, T. Schlosser, J.O. Lilleberg, GAs for multiuser detection in synchronous CDMA, in: Proceedings of the IEEE Int. Symp. on Information Theory, 1997, p. 492.
[9] K. Yen, L. Hanzo, Genetic-algorithm-assisted multiuser detection in asynchronous CDMA communications, IEEE Trans. Veh. Technol. 53 (2004) 1413–1422.
[10] K. Yen, L. Hanzo, Hybrid GA-based multiuser detection schemes for synchronous CDMA systems, in: Proceedings Vehicular Technology Conference, Tokyo, Japan, 2000, pp. 1400–1404.
[11] C. Ergun, K. Hacioglu, Multiuser detection using a GA in CDMA communications systems, IEEE Trans. Commun. 48 (2000) 1374–1383.
[12] H. Wei, L. Hanzo, Reduced-complexity near-optimum GA assisted multiuser detection for synchronous multicarrier CDMA, in: Proceedings of the 59th IEEE Conference on Vehicular Technology, vol. 3, 2004, pp. 1717–1721.
[13] K.G. Maradia, S.M. Joshi, J.S. Patel, Genetic algorithm for CDMA-based MUD techniques under Rayleigh fading environment, IUP J. Telecommun. I (2) (2009) 7–23.
[14] L. Dong, X. Youyun, S. Wentao, L. Hanwen, L. Xingzhao, Genetic algorithm based multiuser detection for CDMA systems, in: Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies, vol. 1, 2004, pp. 321–324.
[15] T.-H. Tan, Y.-F. Huang, L.-C. Hsu, C.-H. Wu, Joint channel estimation and multiuser detection for MC-CDMA system using genetic algorithm and simulated annealing, in: Proceedings of the 2010 IEEE Int. Conference on Systems Man and Cybernetics, 2010, pp. 249–256.
[16] T.-H. Tan, Y.-F. Huang, F.-T. Liu, Multi-user detection in DS-CDMA systems using a genetic algorithm with redundancy saving strategy, Int. J. Innov. Comput. Inform. Control 6 (2010) 3347–3364.
[17] M.G. Shayesteh, M.B. Menhaj, B.G. Nobary, A modified Genetic Algorithm for multiuser detection in DS/CDMA systems, IEICE Trans. Commun. E86-B (2003) 2377–2388.
[18] K. Yen, L. Hanzo, Genetic Algorithm assisted joint multiuser symbol detection and fading channel estimation for synchronous CDMA systems, IEEE J. Sel. Areas Commun. 19 (2001) 985–997.

[19] A. Rashid, F.M. Khan, I.M. Qureshi, Genetic algorithm based multiuser detection in DS-CDMA: a comparative analysis, in: Proceedings of the 2013 IEEE Int. Conference on Machine Learning and Cybernetics, Tianjin, 2013, pp. 728–734.
[20] Y.-F. Huang, T.-H. Tan, C.-H. Cheng, W.C. -Lai, Performance of a novel evolutionary genetic-based multiuser detector for multi-carrier CDMA communication system, Soft Comput. (2015) 1–9.
[21] R.G. Nooka, B.P. Rao, Adaptive genetic algorithm assisted multi user detection of FD-MC-CDMA in frequency selective fading channels, Comput. Sci. Telecommun. 39 (2013) 30–39.
[22] M.A. Khan, M. Umair, M.A.S. Choudhry, GA based adaptive receiver for MC-CDMA system, Turk. J. Electr. Comput. Sci. 23 (2015) 2267–2277.
[23] P.K. Pradhan, Performance evaluation of genetic algorithm assisted synchronous DS/CDMA systems, in: Proceedings of the Int. Conference on Industrial and Information Systems, 2010, pp. 151–154.
[24] L.M. San-José-Revuelta, Entropy-guided micro-genetic algorithm for multiuser detection in CDMA communications, Signal Process. 85 (2005) 1572–1587.
[25] T. Datta, N. Srinidhi, A. Chockalingam, B.S. Rajan, Random-restart reactive tabu search algorithm for detection in large-MIMO Systems, IEEE Commun. Lett. 14 (2010) 1107–1109.
[26] N. Srinidhi, T. Datta, A. Chockalingam, B. Rajan, Layered tabu search algorithm for large-MIMO detection and a lower bound on ML performance, IEEE Trans. Commun. 59 (2011) 2955–2963.
[27] P.H. Tan, L.K. Rasmussen, Multiuser detection in CDMA – a comparison of relaxations, exact, and heuristic search methods, IEEE Trans. Wireless Commun. 3 (2004) 1802–1809.
[28] E. Driouch, W. Ajib, M. Gaha, A tabu search scheduling algorithm for MIMO CDMA systems, in: Proceedings of the 2010 IEEE Global Telecommunications Conference, 2010, pp. 1–5.
[29] H. Gao, M. Diao, Multiuser detection using the novel particle swarm optimization with simulated annealing, in: Proceedings of the 5th Int. Conference on Wireless Communications, Networking and Mobile Computing, 2009, pp. 1–5.
[30] Y.C. Yao, C.H. Cheng, G.J. Wen, J.H. Wen, Multiuser detection using simulated annealing Hopfield Neural Network for DS-UWB systems, in: Proceedings of the IEEE Int. Conference on Machine Learning and Cybernetics, 2011, pp. 763–768.
[31] T.-H. Tan, C.-C. -Chang, F.-R. Jean, J.-Y. Chiang, Y.C. Lu, Joint channel estimation and multi-user detection for OFDMA systems using a genetic algorithm with simmulated annealing-based mutation, in: Proceedings of the IEEE Int, Conference on Syst. Man and Cybern., 2013, pp. 162–167.
[32] J.G. Proakis, Digital Communications, Prentice-Hall, Englewood Cliffs, 2004.
[33] S. Verdú, Multiuser Detection, Cambridge University Press, 1998.
[34] U. Fawer, B. Aazhang, A multiuser receiver for CDMA communications over multipath channels, IEEE Trans. Commun. 43 (1995) 1556–1565.
[35] S.L. Bern, Gender schema theory: a cognitive account for sex typing, Psychol. Rev. 88 (1981) 345–364.
[36] M. Mitchell, An Introduction to Genetic Algorithms, The MIT Press, 1996.
[37] W.K. Lai, G. Coghill, Channel assignment through evolutionary optimization, IEEE Trans. Veh. Technol. 45 (1996) 91–96.
[38] J.J. Grefenstette, Optimization of control parameters for GAs, IEEE Trans. Syst. Man Cybern. 16 (1986) 122–128.
[39] R.K. Ursem, Diversity-guided evolutionary algorithms, in: Proceedings of the Int. Conference on Parallel Problem Solving from Nature VII, Granada, Spain, 2002, pp. 462–471.
[40] L.M. San-José-Revuelta, A new adaptive genetic algorithm for fixed channel assignment, Inf. Sci. 177 (2007) 2655–2678.
[41] F. Glover, Future paths for integer programming and links to artificial intelligence, Comput. Oper. Res. 13 (1986) 533–549.
[42] P. Hansen, The steepest ascent mildest descent heuristic for combinatorial programming, in: Proceedings of the Congress on Numerical Methods in Combinatorial Optimization, Capri, Italy, 1986.
[43] F. Glover, Tabu Search, ORSA J. Comput. 1 (1989) 190–206.
[44] F. Glover, Tabu Search II, ORSA J. Comput. 2 (1990) 4–32.
[45] W.C. Chiang, R.A. Russel, A reactive Tabu search metaheuristic for the vehicle routing problem with time windows, INFORMS J. Comput., Fall 1997 9 (1997) 417–443.
[46] Z. Michalewicz, D.B. Fogel, How to Solve it: Modern Heuristics, Springer Verlag, 2000.
[47] E. Taillard, Robust taboo search for the quadratic assignment problem, Parallel Comput. 17 (1991) 443–455.
[48] R. Battiti, G. Tecchiolli, The reactive Tabu search, INFORMS J. Comput., Spring 1994 6 (1994) 126–140.
[49] M.D. Huang, F. Romeo, S. Vicentelli, An efficient general cooling schedule for simulated annealing, in: Proceedings of the IEEE Int. Conference on Computer-Aided Design (ICCAD), Santa Clara, CA, 1986, pp. 381–384.
[50] F.I. Romeo, Simulated Annealing: Theory and Applications to Layout Problems, Memo UCB/ERL M89/29, Univ, Berkeley, CA, 1989.
[51] G.K. Shojaee, G.H. Shakouri, M.B. Taghadosi, Importance of the initial conditions and the time schedule in the simulated annealing, in: P.N. Rui Chibante (Ed.), Simulated Annealing, Theory with Applications, InTech, 2010.
[52] A.M. Bravo, J. Monera, V. Bhargava, Implementation of the decorrelating receiver for asynchronous DS-CDMA systems over multipath fading channels, Wireless Pers. Commun. 15 (2000) 79–95.
[53] L.M. San-José-Revuelta, J. Cid-Sueiro, Bayesian and RBF structures for wireless communications detection, in: Proceedings of the IEEE Workshop on Neural Networks and Signal Processing, Toulouse, France, 2003, pp. 749–758.
[54] N. Cañibano, L.M. San-José-Revuelta, Analysis of a Bayesian multiuser detector for non-data-aided CDMA communications, in: Proceedings of the IEEE Workshop on Machine Learning and Signal Processing, Sao Luis, Brazil, 2004.