Evolutionary-Algorithm-Assisted Joint Channel Estimation and Turbo Multiuser Detection/Decoding for OFDM/SDMA

Jiankang Zhang, *Member, IEEE*, Sheng Chen, *Fellow, IEEE*, Xiaomin Mu, and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—The development of evolutionary algorithms (EAs), such as genetic algorithms (GAs), repeated weighted boosting search (RWBS), particle swarm optimization (PSO), and differential evolution algorithms (DEAs), have stimulated wide interests in the communication research community. However, the quantitative performance-versus-complexity comparison of GA, RWBS, PSO, and DEA techniques applied to the joint channel estimation (CE) and turbo multiuser detection (MUD)/decoding in the context of orthogonal frequency-division multiplexing/space-division multiple-access systems is a challenging problem, which has to consider both the CE problem formulated over a continuous search space and the MUD optimization problem defined over a discrete search space. We investigate the capability of the GA, RWBS, PSO, and DEA to achieve optimal solutions at an affordable complexity in this challenging application. Our study demonstrates that the EA-assisted joint CE and turbo MUD/decoder is capable of approaching both the Cramér–Rao lower bound of the optimal CE and the bit error ratio (BER) performance of the idealized optimal maximum-likelihood (ML) turbo MUD/decoder associated with perfect channel state information, respectively, despite imposing only a fraction of the idealized turbo ML-MUD/decoder's complexity.**

***Index Terms*—Differential evolution algorithm (DEA), evolutionary algorithms (EAs), genetic algorithm (GA), joint channel estimation (CE) and turbo multiuser detection (MUD)/decoding, orthogonal frequency-division multiplexing (OFDM), particle swarm optimization (PSO), repeated weighted boosting search (RWBS), space-division multiple access (SDMA).**

I. Introduction

THE BEST possible exploitation of the finite available spectrum in light of the increasing demand for wireless services has been at the center of wireless system optimization. In recent years, multiple antennas have been employed both at the transmitter and/or the receiver, which leads to the concept of multiple-input–multiple-output (MIMO) systems. MIMO systems may be designed for achieving various design goals, such as maximizing the achievable diversity gain, the attainable multiplexing gain, or the number of users supported [1], [2]. Orthogonal frequency-division multiplexing (OFDM) [3], [4] has found its way into numerous recent wireless network standards, owing to its virtues of resilience to frequency-selective fading channels. Both the modulation and demodulation operations of an OFDM system facilitate convenient low-complexity hardware implementations with the aid of the inverse fast Fourier transform (IFFT) and fast Fourier transform (FFT) operations. In an effort to further increase the achievable system capacity, space-division multiple-access (SDMA) communication systems were conceived [5], [6], where several users, roaming in different geographical locations and sharing the same bandwidth and time slots (TSs), are differentiated by their unique user-specific "spatial signature," i.e., by their unique channel impulse responses (CIRs). As one of the most widespread MIMO types, OFDM/SDMA systems [7], [8] exploit the advantages of both OFDM and SDMA.

In the uplink (UL) of an OFDM/SDMA system, the transmitted signals of several single-antenna mobile stations (MSs) are simultaneously received by an array of antennas at the base station (BS). Multiuser detection (MUD) techniques are invoked at the BS for separating the signals of the different MSs, based on their unique user-specific CIRs. A state-of-the-art turbo MUD/decoder exploits the error correction capability of the channel code by exchanging extrinsic information between the MUD and the channel decoder [9]. Naturally, for a turbo MUD/decoder to achieve an optimal or near-optimal performance, the CIRs have to be accurately estimated [1], [4]. Intensive research efforts have been devoted to developing efficient approaches for channel estimation (CE) in multiuser OFDM/SDMA systems [1], [8], [10], [11]. To achieve a near-optimal performance, joint CE and turbo MUD/decoding has recently received significant research attention [12]. Naturally, approaching the performance of the optimal solution, namely, that of the maximum-likelihood (ML) joint CE and turbo MUD/decoding solution, is highly desired. However, in practice, one often has to settle for suboptimal solutions due to the excessive computational complexity of the optimal ML solution, particularly for systems with a high number of

Manuscript received February 13, 2013; revised July 5, 2013; accepted September 17, 2013. This work was supported in part by the National Natural Science Foundation of China under Grant 61172086 and Grant 61271421, by the Research Councils UK under the India–UK Advanced Technology Center, and by the European Research Councils under its Advanced Fellow Grant. The review of this paper was coordinated by Dr. H. Lin.

J. Zhang was with the Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K., where this work was carried out. He is now with the School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China (e-mail: jz09v@ecs.soton.ac.uk).

S. Chen and L. Hanzo are with the Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (e-mail: sqc@ecs.soton.ac.uk; lh@ecs.soton.ac.uk).

X. Mu is with the School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China (e-mail: iexmmu@zzu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2013.2283069

users/antennas and employing high-order quadrature amplitude modulation (QAM) signaling [13]. Fortunately, evolutionary algorithms (EAs) offer potentially viable alternatives for achieving optimal or near-optimal joint CE and turbo MUD/decoding at an affordable complexity.

EAs have found ever-increasing applications in communication and signal processing, where creating globally or near-globally optimal designs at affordable computational costs is critical. The family of the most popular EAs[1] includes genetic algorithms (GAs) [16], [17], repeated weighted boosting search (RWBS) [18], [19], particle swarm optimization (PSO) [20], [21], and differential evolution algorithms (DEAs) [22], [23]. Significant advances have been made in applying these EAs in single-user joint channel and data estimation [18], [24]–[26], in CE and MUD for the multiuser code-division multiple-access UL [27]–[30], in the SDMA-aided OFDM UL [31]–[34], in joint CE and data detection for MIMO systems [35]–[37], and in a diverse range of other applications. However, there is paucity of contributions on EA-aided joint CE and turbo MUD/decoding schemes designed for OFDM/SDMA systems. An exception is our previous work [38], which applies a DEA for supporting the joint CE and turbo MUD/decoding process. Iterative joint CE and turbo MUD/decoding for OFDM/SDMA represents an ideal benchmark application for evaluating various EAs. The ML-MUD optimization is NP-hard, and the joint ML CE and turbo MUD/decoding solution is computationally prohibitive in general. Furthermore, within the iterative CE and turbo MUD/decoding optimization, the CE optimization problem is defined over a continuous search space, whereas the MUD optimization problem is defined over a discrete search space. Thus, both discrete-valued and continuous-valued EAs are required. While individual EAs may have been tested in this challenging iterative joint CE and turbo MUD/decoding optimization, to the best of our knowledge, no performance-versus-complexity comparisons of a group of EA techniques have been presented in the literature in the context of joint CE and turbo MUD/decoding.

Against this background, in this paper, we design and characterize four EAs, namely, the GA, RWBS, PSO, and DEA, under the challenging framework of joint CE and turbo MUD/decoding in OFDM/SDMA systems, in terms of their achievable performance, computational complexity, and convergence characteristics. More specifically, continuous-valued EAs are employed in solving the associated CE optimization, whereas the discrete-binary versions of EAs are employed for finding the ML or near-ML solution for the MUD. In the proposed EA-aided iterative scheme conceived for joint blind CE and turbo MUD/decoding, the EA-aided turbo MUD/decoder feeds back ever more reliable detected data to the EA-based channel estimator. Likewise, a more accurate channel estimate will result in an increased-integrity MUD/decoder. We demonstrate the power and efficiency of this EA-aided iterative CE and turbo MUD/decoder in our extensive simulation study. Our obtained results confirm that the channel estimate and the bit error ratio (BER) performance of our EA-assisted iterative CE and turbo MUD/decoder scheme approach the Cramér–Rao lower bound (CRLB) of the optimal CE [39] and the optimal ML turbo MUD/decoding performance associated with perfect channel state information (CSI), respectively, while only imposing a fraction of the complexity of the idealized turbo ML-MUD/decoder.

The remainder of this paper is organized as follows: The multiuser OFDM/SDMA UL model is described in Section II, which provides the necessary notations and defines the associated optimization problems of the joint CE and turbo MUD/decoding. Section III characterizes the four EAs, i.e., the GA, RWBS, PSO, and DEA, which are used for solving the joint CE and turbo MUD/decoding optimization. Both the continuous-valued EAs invoked for solving the CE optimization and their discrete versions used for solving the ML MUD optimization are detailed in this section. Section IV is devoted to the structure of the proposed EA-aided iterative CE and turbo MUD/decoder as well as to its computational complexity analysis. Our simulation results are presented in Section V, whereas our conclusions are offered in Section VI.

II. Multiuser MIMO OFDM/SDMA System

The multiuser MIMO system considered supports U MSs simultaneously transmitting in the UL to the BS, as shown in Fig. 1. Each user is equipped with a single transmit antenna, whereas the BS employs an array of Q antennas. A time-division multiple-access protocol organizes the available time-domain (TD) resources into TSs. All the U MSs are assigned to every TS, and thus, they are allowed to simultaneously transmit their streams of OFDM-modulated symbols to the SDMA-based BS [4], [7] for the sake of exploiting the available resources. Consequently, the users' signals can only be separated with the aid of their unique CIRs.

A. System Model

For the multiuser OFDM/SDMA UL shown in Fig. 1, all the users simultaneously transmit their data streams, which are denoted by $\mathbf{b}^u$ for $1 \le u \le U$. The information bits, i.e., $\mathbf{b}^u$, are first encoded by the user-specific forward error correction (FEC) encoder. The bit stream after the FEC encoder, which is denoted as $\mathbf{b}_C^u$, is passed through an interleaver $\prod$ to yield an output bit stream $\mathbf{b}_I^u$, which is then grouped into blocks of $\log_2 M$ bits as a unit and modulated onto a stream of M-QAM symbols. The modulated data $\tilde{\mathbf{X}}^u$ are serial-to-parallel (S/P) converted, and the pilot symbols are embedded to yield the frequency-domain (FD) OFDM symbol, i.e., $X^u[s,k]$, $1 \le k \le K$, where s denotes the OFDM symbol index, and K is the number of subcarriers. The FD pilot symbols and their allocation are known at the receiver and, hence, can be exploited for initial CE. The parallel modulated data are fed to a K-point IFFT-based modulator to generate the TD-modulated signal $x^u[s,k]$. After concatenating the cyclic prefix (CP) of K_{cp} samples, the resultant sequence is transmitted through the MIMO channel and contaminated by the receiver's additive white Gaussian noise (AWGN). The length of the CP must

[1]There are numerous other EAs, for example, the ant colony optimization [14], [15]; however, given our limited space, we concentrate on only four algorithms in this paper.

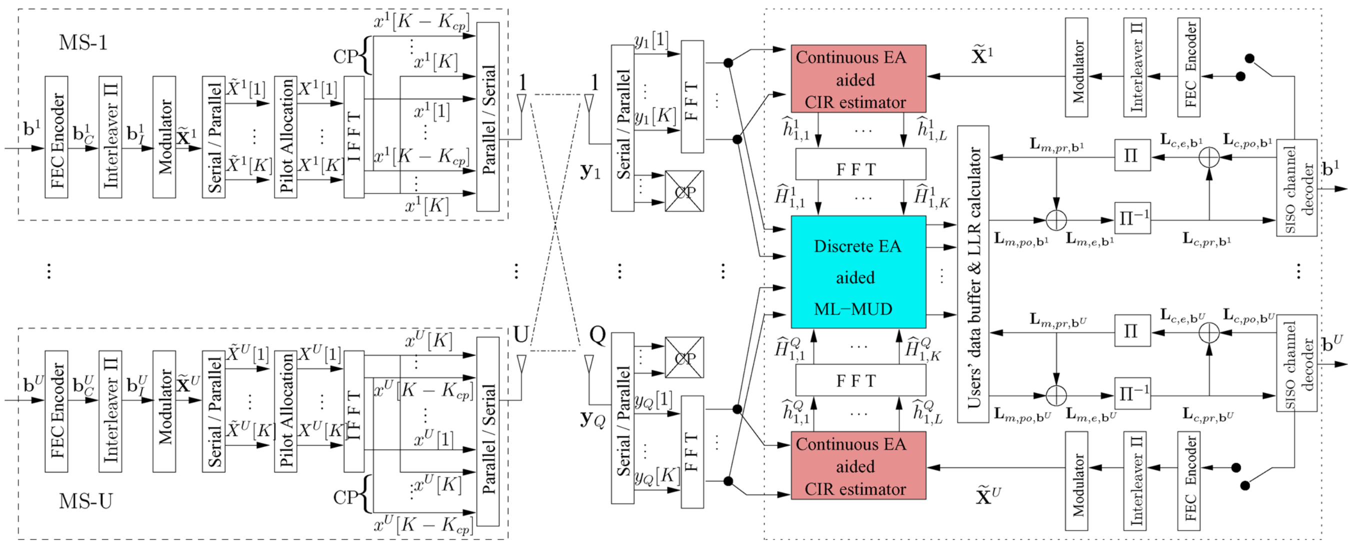


Fig. 1. UL system model for multiuser MIMO OFDM/SDMA. The notation L denotes the log-likelihood ratio. The subscripts m and c of L are associated with the MUD and the channel decoder, respectively, whereas subscripts pr, po, and e are used for representing the *a priori*, *a posteriori*, and extrinsic information, respectively. For notational conciseness, OFDM symbol index s is omitted in $X^u[k]$.

be chosen as $K_{\rm cp} \geq L_{\rm cir}$, where $L_{\rm cir}$ denotes the length of the CIRs.

At the BS, the received signals $\mathbf{y}_q$ for $1 \leq q \leq Q$ are parallel-to-serial (P/S) converted, and the CPs are discarded from every OFDM symbol. The resultant signals are fed into the K-point FFT-based receiver. The signal $Y_q[s,k]$ received by the qth receiver antenna element in the kth subcarrier of the sth OFDM symbol can be expressed as [4]

$$Y_q[s,k] = \sum_{u=1}^{U} H_q^u[s,k] X^u[s,k] + W_q[s,k] \tag{1}$$

where $H_q^u[s,k]$ denotes the FD channel transfer function (FD-CHTF) coefficient of the link between the uth user and the qth receiver antenna in the kth subcarrier of the sth OFDM symbol, whereas $W_q[s,k]$ is the associated FD AWGN having the power of $2\sigma_n^2$. Let $\mathbf{h}_q^u[s] \in \mathbb{C}^{L_{\rm cir}\times 1}$ be the CIR vector of the link between the uth user and the qth receive antenna element during the sth OFDM symbol period, which contains $L_{\rm cir}$ significant CIR coefficients. Then, the FD-CHTF vector $\mathbf{H}_q^u[s] \in \mathbb{C}^{K\times 1}$ is the K-point FFT of $\mathbf{h}_q^u[s]$ defined by

$$\mathbf{H}_q^u[s] = \left[H_q^u[s,1]\; H_q^u[s,2] \cdots H_q^u[s,K]\right]^{\rm T} = \mathbf{F}\mathbf{h}_q^u[s] \tag{2}$$

where $\mathbf{F} \in \mathbb{C}^{K\times L_{\rm cir}}$ denotes the FFT matrix [4]. As a benefit of the CP, the OFDM symbols do not overlap, and SDMA processing can be applied on a per-carrier basis.

Arrange the received data at each receive antenna in a column vector $\mathbf{Y}_q[s] \in \mathbb{C}^{K\times 1}$, i.e.,

$$\mathbf{Y}_q[s] = \left[Y_q[s,1]\; Y_q[s,2] \cdots Y_q[s,K]\right]^{\rm T}, \qquad 1 \leq q \leq Q \tag{3}$$

which hosts the subcarrier-related signals $Y_q[s,k]$, and the transmitted data of each user in a diagonal matrix $\mathbf{X}^u[s] \in \mathbb{C}^{K\times K}$, i.e.,

$$\mathbf{X}^u[s] = \mathrm{diag}\left\{X^u[s,1], X^u[s,2], \ldots, X^u[s,K]\right\} \tag{4}$$

with $X^u[s,k]$ as its diagonal elements, for $1 \leq u \leq U$. Furthermore, let us define the CIR vector $\mathbf{h}_q[s] \in \mathbb{C}^{UL_{\rm cir}\times 1}$ corresponding to the qth receive antenna during the sth OFDM symbol period as

$$\mathbf{h}_q[s] = \left[\left(\mathbf{h}_q^1[s]\right)^{\rm T} \left(\mathbf{h}_q^2[s]\right)^{\rm T} \cdots \left(\mathbf{h}_q^U[s]\right)^{\rm T}\right]^{\rm T}, \qquad 1 \leq q \leq Q. \tag{5}$$

The operations of the BS receiver can be summarized as follows: Given the received data $\{\mathbf{Y}_q[s]\}_{q=1}^{Q}$, find the channels $\{\mathbf{h}_q[s]\}_{q=1}^{Q}$ and the transmitted data $\{\mathbf{X}^u[s]\}_{u=1}^{U}$. Ultimately, the receiver is responsible for recovering the users' transmitted information bit streams $\{\mathbf{b}^u\}_{u=1}^{U}$. The turbo MUD/decoder exchanges soft extrinsic information between the soft-in–soft-out (SISO) MUD and the SISO channel decoder [9], which effectively mitigates both the noise and multiuser interference. As a result, it is capable of achieving an accurate recovery of the users' information bit streams. We defer the discussion on the per-carrier-based turbo MUD/decoder [7] in Fig. 1 to Section IV and concentrate on the basic operations of joint CE and MUD at the BS receiver to highlight our motivation for applying EAs to this challenging application.

B. Optimization Problems in Joint CE and MUD

Denote the overall system's CIR vector by $\mathbf{h}[s] \in \mathbb{C}^{UQL_{\rm cir}\times 1}$ and all the users' transmitted data matrix $\mathbf{X}[s] \in \mathbb{C}^{UK\times K}$, respectively, as

$$\mathbf{h}[s] = \left[\mathbf{h}_1^{\rm T}[s]\mathbf{h}_2^{\rm T}[s] \cdots \mathbf{h}_Q^{\rm T}[s]\right]^{\rm T} \tag{6}$$

$$\mathbf{X}[s] = \left[\mathbf{X}^1[s]\mathbf{X}^2[s] \cdots \mathbf{X}^U[s]\right]^{\rm T}. \tag{7}$$

The optimal solution of the joint CE and MUD problem is achieved by maximizing the probability of all the received data $\{\mathbf{Y}_q[s]\}_{q=1}^{Q}$ conditioned on $\mathbf{h}[s]$ and $\mathbf{X}[s]$. Noting that this conditional distribution is Gaussian, this joint optimization is equivalent to the one that minimizes the log-likelihood cost function (CF) formulated as

$$J\left(\mathbf{h}[s], \mathbf{X}[s]\right) = \sum_{q=1}^{Q} \left\| \mathbf{Y}_q[s] - \mathbf{X}^{\mathrm{T}}[s]\overline{\mathbf{F}}\mathbf{h}_q[s] \right\|^2 \tag{8}$$

where the block diagonal matrix $\overline{\mathbf{F}} \in \mathbb{C}^{UK \times UL_{\mathrm{cir}}}$ is given by

$$\overline{\mathbf{F}} = \mathrm{diag}\{\underbrace{\mathbf{F}, \mathbf{F}, \ldots, \mathbf{F}}_{U}\}. \tag{9}$$

Thus, the joint ML CE and MUD solution is defined as

$$\left(\widehat{\mathbf{h}}[s], \widehat{\mathbf{X}}[s]\right) = \arg \min_{\mathbf{h}[s], \mathbf{X}[s]} J\left(\mathbf{h}[s], \mathbf{X}[s]\right). \tag{10}$$

Joint ML optimization (10) is defined in an extremely high-dimensional space with both discrete- and continuous-valued decision variables, and therefore, it is computationally prohibitive. The complexity of this optimization process may be reduced to a more tractable level by invoking an iterative search loop that is carried out first over the continuous space of the legitimate channels $\mathbf{h}[s]$ and then over the discrete set of all the possible transmitted data $\mathbf{X}[s]$. The iterative loop between the CE and the MUD encapsulates two optimization problems. CE optimization can be performed when the data $\mathbf{X}[s]$ are available, either as the known pilot symbols at the start or, more generally, as the detected data fed back from the MUD and FEC-decoder unit. The MUD can be carried out with the estimated CIRs provided by the channel estimator. The iterative procedure exchanging extrinsic information between the decision-directed channel estimator and the MUD based on the estimated CIRs gradually improves both solutions, and typically, only a few iterations are required for approaching the joint ML CE and MUD solution of (10).

1) ML CE: With the detected data $\widehat{\mathbf{X}}[s]$ fed back from the MUD/decoder, the ML CE solution is obtained by minimizing the CF $J_{\mathrm{ce}}(\mathbf{h}[s]) = J(\mathbf{h}[s], \widehat{\mathbf{X}}[s])$. Since the CIRs $\mathbf{h}_q[s]$, $1 \leq q \leq Q$, are only related to the received signals $\mathbf{Y}_q[s]$ recorded at the qth receiver antenna, the ML CE solution $\widehat{\mathbf{h}}[s]$ is given as the solutions of the following Q smaller minimization problems:

$$\widehat{\mathbf{h}}_q[s] = \arg \min_{\mathbf{h}_q[s]} J_{\mathrm{ce}}\left(\mathbf{h}_q[s]\right), \qquad 1 \leq q \leq Q \tag{11}$$

where the CE CF is expressed as

$$J_{\mathrm{ce}}\left(\mathbf{h}_q[s]\right) = \left\| \mathbf{Y}_q[s] - \widehat{\mathbf{X}}^{\mathrm{T}}[s]\overline{\mathbf{F}}\mathbf{h}_q[s] \right\|^2. \tag{12}$$

Since $\mathbf{h}_q[s] \in \mathbb{C}^{UL_{\mathrm{cir}} \times 1}$, the search space for the CE optimization is a continuous-valued $(2UL_{\mathrm{cir}})$-element space. As the detected data contain erroneous decisions, error propagation imposes a serious problem. The OFDM symbol index $[s]$ will be omitted during our forthcoming discourse.

The standard least squares (LS) channel estimator [40] may provide the solutions of (11), which, however, is computationally very expensive as it requires the inverse of the Q very large $(UL_{\mathrm{cir}}) \times (UL_{\mathrm{cir}})$ complex-valued correlation matrices to obtain $\widehat{\mathbf{h}}_q$ for $1 \leq q \leq Q$. A low-complexity simplified LS channel estimator was provided in [40]. However, this simplified LS estimator only works for optimally designed pilots to ensure all the correlation matrices are diagonal. This simplified LS channel estimator performs poorly even given with the correct error-free transmitted data, and clearly, it cannot be applied in decision-directed mode.

2) ML MUD: As a benefit of the CP, the OFDM symbols do not overlap, and receiver processing can be applied on a per-carrier basis [1], [7]. Let us define the received data vector $\mathbf{Y}[s,k] \in \mathbb{C}^{Q \times 1}$ of Q antennas and the transmitted signal vector $\mathbf{X}[s,k] \in \mathbb{C}^{U \times 1}$ of U users in the kth subcarrier of the sth OFDM symbol, respectively, as

$$\mathbf{Y}[s,k] = \left[Y_1[s,k] Y_2[s,k] \cdots Y_Q[s,k]\right]^{\mathrm{T}} \tag{13}$$

$$\mathbf{X}[s,k] = \left[X^1[s,k] X^2[s,k] \cdots X^U[s,k]\right]^{\mathrm{T}}. \tag{14}$$

Furthermore, denote the FD-CHTF matrix linking $\mathbf{X}[s,k]$ to $\mathbf{Y}[s,k]$ as $\mathbf{H}[s,k] \in \mathbb{C}^{Q \times U}$, whose qth row and uth column element is $H_q^u[s,k]$. Given the FD-CHTF matrix estimate $\widehat{\mathbf{H}}[s,k]$, the MUD recovers the transmitted signals $\mathbf{X}[s,k]$ from the received signals $\mathbf{Y}[s,k]$. Since each element $X^u[s,k]$ of $\mathbf{X}[s,k]$ belongs to the finite M-QAM alphabet $\mathcal{S}$ of size $|\mathcal{S}| = M$, there are M^U possible candidate solutions for $\mathbf{X}[s,k]$, and the optimal ML MUD solution is defined as

$$\widehat{\mathbf{X}}[s,k] = \arg \min_{\mathbf{X}[s,k] \in \mathcal{S}^U} J_{\mathrm{mud}}\left(\mathbf{X}[s,k]\right) \tag{15}$$

with the MUD optimization CF expressed as

$$J_{\mathrm{mud}}\left(\mathbf{X}[s,k]\right) = \left\| \mathbf{Y}[s,k] - \widehat{\mathbf{H}}[s,k]\mathbf{X}[s,k] \right\|^2. \tag{16}$$

Optimization (15) is well known to be NP-hard. Since each $X^u[s,k]$ contains $A = \log_2 M$ bits, the bit-stream representation of $X^u[s,k]$ is $\mathbf{b}^u[s,k] = [b_1^u[s,k] b_2^u[s,k] \cdots b_A^u[s,k]]^{\mathrm{T}}$, where each element or bit $b_i^u[s,k] \in \{0,1\}$. Thus, the bit-stream representation of $\mathbf{X}[s,k]$ is

$$\mathbf{b}[s,k] = \left[b_1^1[s,k] \cdots b_A^1[s,k] b_1^2[s,k] \cdots b_A^2[s,k] \cdots b_1^U[s,k] \cdots b_A^U[s,k]\right]^{\mathrm{T}} \tag{17}$$

and the MUD optimization CE is equivalently denoted as $J_{\mathrm{mud}}(\mathbf{b}[s,k]) = J_{\mathrm{mud}}(\mathbf{X}[s,k])$. The OFDM index and the subcarrier index $[s,k]$ will be omitted in the sequel.

Various alternative solutions to the NP-hard ML solution of optimization (15) are available, which trade off performance with complexity. The examples of low-complexity suboptimal solutions include the minimum-mean-square-error MUD, successive-interference-cancelation MUD, and parallel-interference-cancelation MUD. Sphere-detection-based MUD, on the other hand, offers a near-optimal solution with more affordable computational complexity. Moreover, EAs have been demonstrated to be capable of solving this ML optimization problem with complexity that is a fraction of the full-optimal ML complexity [27]–[30], [33]–[38].

III. EAs for Iterative CE and MUD

The continuous versions of the GA, RWBS, PSO, and DEA are adopted to aid in CE optimization, which are denoted as the continuous-GA-assisted CE (CGA-CE), continuous-RWBS-assisted CE (CRWBS-CE), continuous-PSO-assisted CE (CPSO-CE), and continuous-DEA-assisted CE (CDEA-CE). By contrast, the discrete-binary versions of these four EAs are adopted for MUD optimization, which are referred to as the discrete-binary GA-assisted MUD (DBGA-MUD), discrete-binary RWBS-assisted MUD (DBRWBS-MUD), discrete-binary PSO-assisted MUD (DBPSO-MUD), and discrete-binary DEA-assisted MUD (DBDEA-MUD).

A. GA for Iterative CE and MUD

1) CGA-CE: The CGA-CE evolves the population of the P_s candidate solutions over the entire solution space, where P_s is known as the population size. These candidate solutions represent the estimates of the CIR coefficient vector $\mathbf{h}_q$, where the p_sth individual of the population in the gth generation is readily expressed as

$$\widehat{\mathbf{h}}_{q,g,p_s} = \Big[\widehat{h}^1_{q,g,p_s,1}\cdots\widehat{h}^1_{q,g,p_s,L_{\rm cir}}\widehat{h}^2_{q,g,p_s,1}\cdots\widehat{h}^2_{q,g,p_s,L_{\rm cir}} \cdots\widehat{h}^U_{q,g,p_s,1}\cdots\widehat{h}^U_{q,g,p_s,L_{\rm cir}}\Big]^{\rm T} \tag{18}$$

in which $\widehat{h}^u_{q,g,p_s,l}$ represents an estimate of the lth coefficient in CIR vector $\mathbf{h}^u_q$ for the channel linking user-u to antenna-q. The search space for CE optimization is specified by $(-1-\mathrm{j},+1+\mathrm{j})^{UL_{\rm cir}}$, with $\mathrm{j}=\sqrt{-1}$. Referring to Fig. 2, we now specify this CGA-CE.

Algorithm 1: CGA-CE.

1) **Initialization**. Set the generation index to $g=1$ and randomly generate the initial population, i.e., $\{\widehat{\mathbf{h}}_{q,1,p_s}\}_{p_s=1}^{P_s}$, over the search space $(-1-\mathrm{j},+1+\mathrm{j})^{UL_{\rm cir}}$.
2) **Selection**. The fitness value of an individual $\widehat{\mathbf{h}}_{q,g,p_s}$ is related to its CF value by $f(\widehat{\mathbf{h}}_{q,g,p_s}) = J^{-1}_{\rm ce}(\widehat{\mathbf{h}}_{q,g,p_s})$. The roulette wheel selection operator [17] in Fig. 2 is adopted for selecting high-fitness individuals, where the selection ratio of r_s decides how many individuals are to be selected into the mating pool from the total P_s individuals. The value of r_s is defined by $r_s=(N_{\rm pool}/P_s)$, where $N_{\rm pool}$ is the size of the mating pool.
3) **Crossover**. For each pair of parents randomly chosen from the mating pool, the pair of integers u^* and l^* is randomly generated in the ranges of $\{1,2,\ldots,U\}$ and $\{1,2,\ldots,L_{\rm cir}\}$, respectively. The parents selected for the crossover operation can be expressed as

$$\begin{cases}\widehat{\mathbf{h}}_{q,g,{\rm mum}} = \Big[\widehat{h}^1_{q,g,{\rm mum},1}\cdots\widehat{h}^{u^*}_{q,g,{\rm mum},l^*-1}\widehat{h}^{u^*}_{q,g,{\rm mum},l^*} \\ \qquad\qquad \widehat{h}^{u^*}_{q,g,{\rm mum},l^*+1}\cdots\widehat{h}^{U}_{q,g,{\rm mum},L_{\rm cir}}\Big]^{\rm T} \\ \widehat{\mathbf{h}}_{q,g,{\rm dad}} = \Big[\widehat{h}^1_{q,g,{\rm dad},1}\cdots\widehat{h}^{u^*}_{q,g,{\rm dad},l^*-1}\widehat{h}^{u^*}_{q,g,{\rm dad},l^*} \\ \qquad\qquad \widehat{h}^{u^*}_{q,g,{\rm dad},l^*+1}\cdots\widehat{h}^{U}_{q,g,{\rm dad},L_{\rm cir}}\Big]^{\rm T}. \end{cases} \tag{19}$$

As indicated in Fig. 2, the two new offsprings are produced as

$$\begin{cases}\widehat{\mathbf{h}}_{q,g,{\rm os1}} = \Big[\widehat{h}^1_{q,g,{\rm mum},1}\cdots\widehat{h}^{u^*}_{q,g,{\rm mum},l^*-1}\widehat{h}^{u^*}_{q,g,{\rm os1},l^*} \\ \qquad\qquad \widehat{h}^{u^*}_{q,g,{\rm os1},l^*+1}\cdots\widehat{h}^{U}_{q,g,{\rm os1},L_{\rm cir}}\Big]^{\rm T} \\ \widehat{\mathbf{h}}_{q,g,{\rm os2}} = \Big[\widehat{h}^1_{q,g,{\rm dad},1}\cdots\widehat{h}^{u^*}_{q,g,{\rm dad},l^*-1}\widehat{h}^{u^*}_{q,g,{\rm os2},l^*} \\ \qquad\qquad \widehat{h}^{u^*}_{q,g,{\rm os2},l^*+1}\cdots\widehat{h}^{U}_{q,g,{\rm os2},L_{\rm cir}}\Big]^{\rm T} \end{cases} \tag{20}$$

with

$$\begin{cases}\widehat{h}^{u^*}_{q,g,{\rm os1},l} = \widehat{h}^{u^*}_{q,g,{\rm mum},l} - \beta\left(\widehat{h}^{u^*}_{q,g,{\rm mum},l} - \widehat{h}^{u^*}_{q,g,{\rm dad},l}\right) \\ \widehat{h}^{u^*}_{q,g,{\rm os2},l} = \widehat{h}^{u^*}_{q,g,{\rm dad},l} + \beta\left(\widehat{h}^{u^*}_{q,g,{\rm mum},l} - \widehat{h}^{u^*}_{q,g,{\rm dad},l}\right)\end{cases} \tag{21}$$

for $l^* \le l \le L_{\rm cir}$, where β is a random value uniformly chosen in the range of $(0,1)$.
4. **Mutation**. As shown in the operation of Step 4) Mutation in Fig. 2, an element or gene $\widehat{h}^u_{q,g,p_s,l}$ of the individual $\widehat{\mathbf{h}}^u_{q,g,p_s}$ is mutated according to

$$\breve{h}^u_{q,g,p_s,l} = \widehat{h}^u_{q,g,p_s,l} + \gamma(\alpha_m + \mathrm{j}\beta_m) \tag{22}$$

where both α_m and β_m are randomly generated in the range $(-1, 1)$, whereas γ is a mutation parameter. The number of genes that will mutate is governed by mutation probability M_b.
5. **Termination**. If $g > G_{\max}$, where $G_{\max}$ defines the maximum number of generations, the procedure is curtailed. Otherwise, we set $g=g+1$, and go to 2) **Selection**.

The key algorithmic parameters of this CGA-CE are population size P_s, selection ratio r_s, mutation probability M_b, and mutation parameter γ.

2) DBGA-MUD: A discrete-binary GA has similar basic operations as a continuous GA, which are shown in Fig. 2. This GA evolves a population of the P_s (UA)-element binary-valued candidate vectors, and each individual represents an estimate of the bit sequence $\mathbf{b}$ defined in (17). The p_sth individual of the population in the gth generation is expressed as

$$\widehat{\mathbf{b}}_{g,p_s} = \Big[\widehat{b}^1_{g,p_s,1}\cdots\widehat{b}^1_{g,p_s,A}\widehat{b}^2_{g,p_s,1}\cdots\widehat{b}^2_{g,p_s,A}\cdots\widehat{b}^U_{g,p_s,1}\cdots\widehat{b}^U_{g,p_s,A}\Big]^{\rm T}. \tag{23}$$

Each binary-valued individual $\widehat{\mathbf{b}}_{g,p_s}$ is related to a signal $\widehat{\mathbf{X}}_{g,p_s}$ transmitted by the M-QAM modulator that represents a candidate solution of MUD optimization (15). The CGA-CE is specified as follows.

Algorithm 2: DBGA-MUD.

1) **Initialization**. Set the generation index to $g=1$ and randomly generate the initial population of the P_s binary-valued individuals $\{\widehat{\mathbf{b}}_{1,p_s}\}_{p_s=1}^{P_s}$.
2) **Selection**. The fitness value of an individual $\widehat{\mathbf{b}}_{g,p_s}$ is re lated to its CF value by $f(\widehat{\mathbf{b}}_{g,p_s}) = J^{-1}_{\rm mud}(\widehat{\mathbf{b}}_{g,p_s})$. The selection ratio r_s specifies the percentage of the P_s indi-

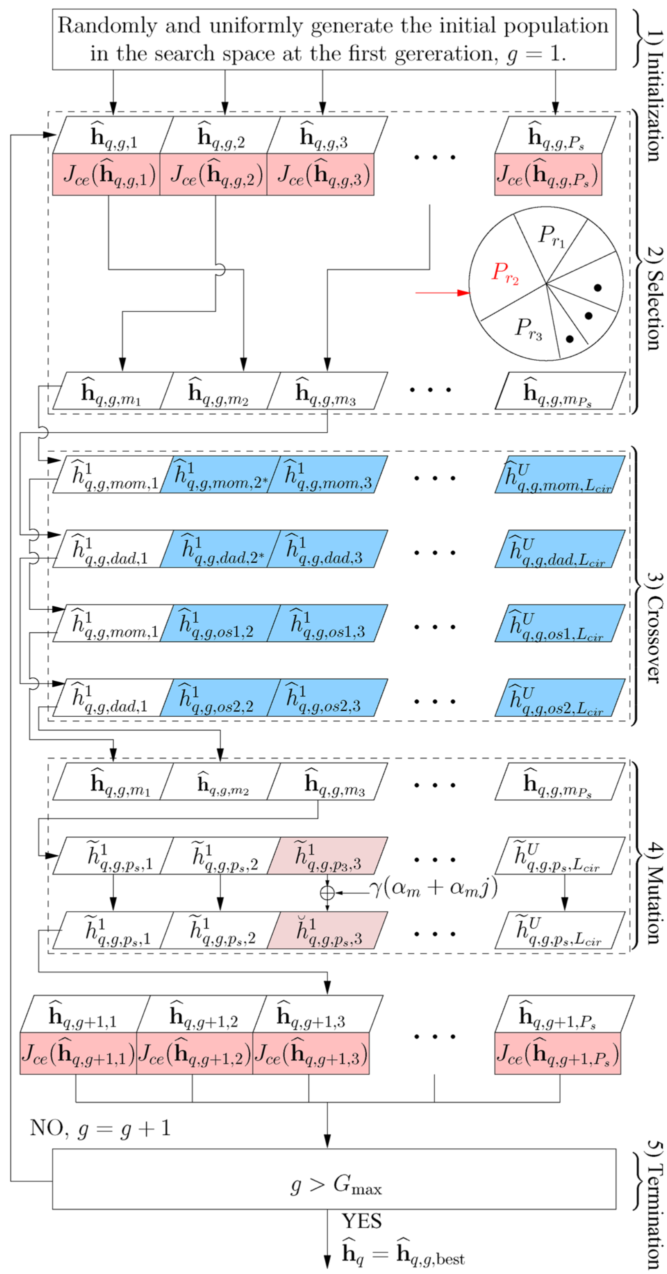


Fig. 2. Flowchart of the continuous-GA-assisted CE.

viduals that are selected to form the mating pool, and we also adopt the roulette wheel selection operator.

3) **Crossover**. We opt for employing the uniform crossover algorithm [17], where a crossover point is randomly selected between the first bit and the last bit of the parent individuals, and the bits are then exchanged between the selected pair of parents.

4) **Mutation**. Given mutation probability M_b, $\lfloor M_b P_s U A \rfloor$ bits are randomly selected from the total number of $(P_s U A)$ bits in the P_s individuals for mutation, where $\lfloor \bullet \rfloor$ denotes the integer floor operator. A bit is mutated by toggling its value from 1 to 0, and vice versa.

5) **Termination**. Optimization is stopped when the predefined maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g + 1$, and go to 2) **Selection**.

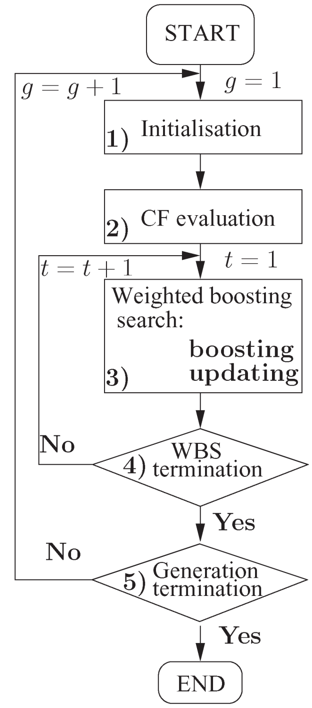


Fig. 3. Flowchart depicting the operations of both the continuous and discrete-binary RWBS algorithms.

The key algorithmic parameters of this DBGA-MUD are population size P_s, selection ratio r_s, and mutation probability M_b.

B. *RWBS for Iterative CE and MUD*

The operations of the RWBS algorithm [18], [19] are shown in Fig. 3, which consists of the generation-based outer loop and the weighted boosting search (WBS) inner loop.

1) CRWBS-CE: Given an initial estimate $\widehat{\mathbf{h}}_{q,0,\text{best}}$, which can be either randomly generated in the search space $(-1-\text{j}, +1+\text{j})^{UL_{\text{cir}}}$ or chosen as the initial-training-based channel estimate with the aid of the simplified LS channel estimator in [40], the CRWBS-CE is initialized by setting the generation index to $g = 1$ and then following the operations given in Algorithm 3.

Algorithm 3: CRWBS-CE.

---

1) **Generation initialization**. The CIRs $\{\widehat{\mathbf{h}}_{q,g,p_s}\}_{p_s=1}^{P_s}$ are initialized according to: $\widehat{\mathbf{h}}_{q,g,1} = \widehat{\mathbf{h}}_{q,g-1,\text{best}}$

$$\widehat{\mathbf{h}}_{q,g,p_s} = \widehat{\mathbf{h}}_{q,g-1,\text{best}} + \gamma \left(\mathbf{Grv}_{UL_{\text{cir}}}(0,1) + \text{j}\mathbf{Grv}_{UL_{\text{cir}}}(0,1) \right), \quad 2 \le p_s \le P_s \quad (24)$$

where $\mathbf{Grv}_{UL_{\text{cir}}}(0,1)$ denotes the (UL_{cir})-element vector, whose elements are drawn from the normal distribution with zero mean and unit variance, $\widehat{\mathbf{h}}_{q,g-1,\text{best}}$ denotes the best individual found in the previous generation, and γ is referred to as the mutation rate.

2) **CF evaluation**. Calculate the CF values associated with the population according to $J_{g,p_s} = J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,p_s})$, $1 \le p_s \le P_s$. Each individual $\widehat{\mathbf{h}}_{q,g,p_s}$ is initially assigned an

equal weight $\delta_{p_s}(0) = (1/P_s)$, where $1 \le p_s \le P_s$. Then, set the WBS iteration index to $t = 1$.

3) **WBS**. This consists of boosting the weights and updating the population.

•*Stage 1. Boosting*. The relative merits of the individuals are used to adapt the weights for guiding the search. Let us define the best and worst individuals, i.e., $\widehat{\mathbf{h}}_{q,g,p_{\text{best}}}$ and $\widehat{\mathbf{h}}_{q,g,p_{\text{worst}}}$, in the population, where we have $p_{\text{best}} = \arg\min_{1 \le p_s \le P_s} J_{g,p_s}$ and $p_{\text{worst}} = \arg\max_{1 \le p_s \le P_s} J_{g,p_s}$.

i) Normalize the CF values $\bar{J}_{g,p_s} = J_{g,p_s} / \sum_{j=1}^{P_s} J_{g,j}$, $1 \le p_s \le P_s$, and compute weighting factor $\beta(t)$ according to

$$\beta(t) = \frac{\eta(t)}{1 - \eta(t)} \text{ with } \eta(t) = \sum_{p_s=1}^{P_s} \delta_{p_s}(t-1) \bar{J}_{g,p_s}. \tag{25}$$

ii) Adapt the weights for $1 \le p_s \le P_s$ as follows:

$$\tilde{\delta}_{p_s}(t) = \begin{cases} \delta_{p_s}(t-1)\left(\beta(t)\right)^{\bar{J}_{g,p_s}}, & \beta(t) \le 1 \\ \delta_{p_s}(t-1)\left(\beta(t)\right)^{1-\bar{J}_{g,p_s}}, & \beta(t) > 1 \end{cases} \tag{26}$$

and normalize them as $\delta_{p_s}(t) = \tilde{\delta}_{p_s}(t) / \sum_{j=1}^{P_s} \tilde{\delta}_j(t)$, $1 \le p_s \le P_s$.

• *Stage 2. Updating*. This population updating stage consists of

i) Convex combination of $\{\widehat{\mathbf{h}}_{q,g,p_s}\}_{p_s=1}^{P_s}$ constructs a new individual as follows:

$$\widehat{\mathbf{h}}_{q,g,P_s+1} = \sum_{p_s=1}^{P_s} \delta_{p_s}(t) \widehat{\mathbf{h}}_{q,g,p_s}. \tag{27}$$

Intuitively, as the individuals of low CF values have high weights, (27) is capable of producing a new individual, which may have an even lower CF value. A "mirror image" of $\widehat{\mathbf{h}}_{q,g,P_s+1}$ is produced as $\widehat{\mathbf{h}}_{q,g,P_s+2} = \widehat{\mathbf{h}}_{q,g,p_{\text{best}}} + (\widehat{\mathbf{h}}_{q,g,p_{\text{best}}} - \widehat{\mathbf{h}}_{q,g,P_s+1})$.

ii) Compute $J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,P_s+1})$ and $J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,P_s+2})$ and find $p_* = \arg\min_{i=P_s+1,P_s+2} J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,i})$. The new individual $\widehat{\mathbf{h}}_{q,g,p_*}$ then replaces $\widehat{\mathbf{h}}_{q,g,p_{\text{worst}}}$ in the population.

4) **WBS termination**. If $t > T_{\text{wbs}}$, where T_{wbs} defines the maximum number of WBS iterations T_{wbs}, exit the WBS inner loop. Otherwise, set $t = t + 1$ and go to 3)**WBS**.

5) **Generation termination**. Stop when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g + 1$, and go to 1) **Generation initialization**.

The key algorithmic parameters of this CRWBS-CE are the population size P_s, the mutation rate γ and the maximum number of WBS iterations T_{wbs}.

2) DBRWBS-MUD: Given a randomly generated initial binary-valued estimate $\widehat{\mathbf{b}}_{0,\text{best}}$, the DBRWBS-MUD commences by setting the generation index to $g = 1$, and it then follows the operations given in Algorithm 4.

Algorithm 4: DBRWBS-MUD.

---

1) **Generation initialization**. Initialize the population $\{\widehat{\mathbf{b}}_{g,p_s}\}_{p_s=1}^{P_s}$ as: set $\widehat{\mathbf{b}}_{g,1} = \widehat{\mathbf{b}}_{g-1,\text{best}}$, while the remaining $P_s - 1$ individuals $\widehat{\mathbf{b}}_{g,p_s}$, $2 \le p_s \le P_s$, are generated by randomly muting a certain percentage of the bits in $\widehat{\mathbf{b}}_{g-1,\text{best}}$, the best individual found in the previous generation. The percentage of bits mutated is governed by the mutation probability M_b.

2) **CF evaluation**. The CF values associated with the population are calculated according to $J_{g,p_s} = J_{\text{mud}}(\widehat{\mathbf{b}}_{g,p_s})$, $1 \le p_s \le P_s$. Each individual $\widehat{\mathbf{b}}_{g,p_s}$ is initially assigned an equal weight $\delta_{p_s}(0) = (1/P_s)$, where $1 \le p_s \le P_s$. Then set the WBS iteration index to $t = 1$.

3) **WBS**. Again, this is composed of the weight boosting and population updating stages.

•*Stage 1. Boosting*. The operations are identical to those of i) and ii) in *Stage 1.* of the CRWBS-CE, which yields the set of weights, $\delta_{p_s}(t)$ for $1 \le p_s \le P_s$.

•*Stage 2. Updating*. Given the P_s individuals' weights $\delta_{p_s}(t)$ for $1 \le p_s \le P_s$, define

$$\begin{cases} \Delta\delta_0(t) = 0 \\ \Delta\delta_{p_s}(t) = \Delta\delta_{p_s-1}(t) + \delta_{p_s}(t), \quad 1 \le p_s \le P_s. \end{cases} \tag{28}$$

Then the four (or a different user-defined number) new individuals $\widehat{\mathbf{b}}_{g,P_s+i}$, $1 \le i \le 4$, are generated as follows: for $1 \le a \le A$ and $1 \le u \le U$,

$$\widehat{b}^u_{g,P_s+i,a} = \widehat{b}^u_{g,p_s,a}, \text{ if } \Delta\delta_{p_s-1}(t) < rand(0,1) \le \Delta\delta_{p_s}(t) \tag{29}$$

where $rand(0,1)$ denotes the random number generator which randomly returns a value from the interval $[0,1)$. The newly generated individuals replace the worst individuals in the population, whose CF values are larger than theirs.

4) **WBS termination**. The WBS iterative procedure is terminated, when the maximum number of WBS iterations T_{wbs} is reached. Otherwise, set $t = t + 1$ and go to 3) **WBS**.

5) **Generation termination**. The procedure is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g + 1$, and go to 1) **Generation initialization**.

The key algorithmic parameters of this DBRWBS-MUD are population size P_s, mutation probability M_b, and the maximum number of WBS iterations T_{wbs}.

C. PSO for Iterative CE and MUD

In a PSO algorithm, individuals of the population are known as particles, and the population is referred to as the swarm. The flowchart of the PSO algorithm adopted is shown in Fig. 4.

1) CPSO-CE: The position of the p_sth particle in the gth generation of the population, i.e., $\widehat{\mathbf{h}}_{q,g,p_s}$, is defined in (18). Associated with each $\widehat{\mathbf{h}}_{q,g,p_s}$, there is a velocity vector $\mathbf{v}_{q,g,p_s} \in (-1 - \mathrm{j}, +1 + \mathrm{j})^{UL_{\text{cir}}}$. Each particle $\widehat{\mathbf{h}}_{q,g,p_s}$ remembers its best position visited so far, denoted by $\widehat{\mathbf{h}}^{\text{ci}}_{q,g,p_s}$, which provides the so-called cognitive information. Every particle also knows the best position visited so far by all particles of the

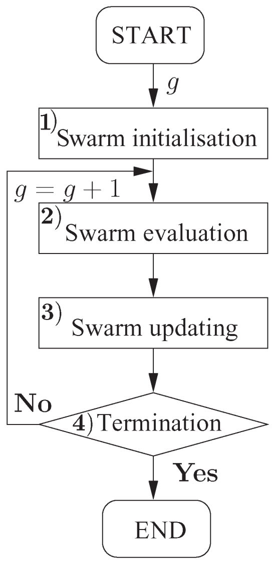


Fig. 4. Flowchart depicting the operations of both the continuous and discrete-binary PSO algorithms.

entire swarm, denoted by $\widehat{\mathbf{h}}^{\rm si}_{q,g}$, which provides the so-called social information. Algorithm 5 details the operations of the CPSO-CE.

Algorithm 5: CPSO-CE.

1) **Initialization**. Set the generation index to $g=1$. Then, randomly generate the initial population, i.e., $\{\widehat{\mathbf{h}}_{q,1,p_s}\}_{p_s=1}^{P_s}$, in the search space $(-1-\mathrm{j},+1+\mathrm{j})^{UL_{\rm cir}}$, and the associated initial velocities, i.e., $\{\mathbf{v}_{q,1,p_s}\}_{p_s=1}^{P_s}$, in the velocity space $(-1-\mathrm{j},+1+\mathrm{j})^{UL_{\rm cir}}$.
2) **Swarm evaluation**. For each particle $\widehat{\mathbf{h}}_{q,g,p_s}$, compute its CF value $J_{\rm ce}(\widehat{\mathbf{h}}_{q,g,p_s})$. For $1\le p_s\le P_s$, update the cognitive information according to the following: If $J_{\rm ce}(\widehat{\mathbf{h}}_{q,g,p_s}) < J_{\rm ce}(\widehat{\mathbf{h}}^{\rm ci}_{q,g-1,p_s})$, set $\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s}=\widehat{\mathbf{h}}_{q,g,p_s}$; otherwise, set $\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s}=\widehat{\mathbf{h}}^{\rm ci}_{q,g-1,p_s}$. Given $p_s^*=\arg\min_{1\le p_s\le P_s} J_{\rm ce}(\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s})$, the swarm's social information is then updated as follows: If $J_{\rm ce}(\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s^*}) < J_{\rm ce}(\widehat{\mathbf{h}}^{\rm si}_{q,g-1})$, set $\widehat{\mathbf{h}}^{\rm si}_{q,g}=\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s^*}$; otherwise, set $\widehat{\mathbf{h}}^{\rm si}_{q,g}=\widehat{\mathbf{h}}^{\rm si}_{q,g-1}$.
3) **Swarm updating**. The individuals' velocities and positions are updated according to

$$\mathbf{v}_{q,g+1,p_s} = \omega\mathbf{v}_{q,g,p_s} + c_1\, rand(0,1)\left(\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s}-\widehat{\mathbf{h}}_{q,g,p_s}\right) + c_2\, rand(0,1)\left(\widehat{\mathbf{h}}^{\rm si}_{q,g}-\widehat{\mathbf{h}}_{q,g,p_s}\right) \quad (30)$$

$$\widehat{\mathbf{h}}_{q,g+1,p_s} = \widehat{\mathbf{h}}_{q,g,p_s}+\mathbf{v}_{q,g+1,p_s} \quad (31)$$

for $1\le p_s\le P_s$, where ω is the inertia weight, whereas c_1 and c_2 are known as the cognitive learning rate and the social learning rate, respectively.
4) **Termination**. Optimization is terminated, when the max imum number of generations $G_{\max}$ is reached. Otherwise, set $g=g+1$, and go to 2) **Swarm evaluation**.

The key algorithmic parameters of this CPSO-CE are population size P_s, cognitive learning rate c_1, and social learning rate c_2.

DBPSO-MUD: In the population of the gth generation, the p_sth individual's position, i.e., $\widehat{\mathbf{b}}_{g,p_s}$, is given by (23), and its associated velocity is expressed as

$$\mathbf{v}_{g,p_s} = \left[v^1_{g,p_s,1}\cdots v^1_{g,p_s,A} v^2_{g,p_s,1}\cdots v^2_{g,p_s,A} \cdots v^U_{g,p_s,1}\cdots v^U_{g,p_s,A}\right]^{\rm T}. \quad (32)$$

The velocity space is defined as $(0,1)^{UA}$, i.e., $\mathbf{v}_{g,p_s}\in(0,1)^{UA}$ [41]. Associated with $\widehat{\mathbf{b}}_{g,p_s}$, there are two bit-toggling probability vectors given, respectively, by

$$\mathbf{v}^0_{g,p_s} = \left[v^{1,0}_{g,p_s,1}\cdots v^{1,0}_{g,p_s,A} b^{2,0}_{g,p_s,1}\cdots v^{2,0}_{g,p_s,A} \cdots\ v^{U,0}_{g,p_s,1}\cdots v^{U,0}_{g,p_s,A}\right]^{\rm T} \quad (33)$$

$$\mathbf{v}^1_{g,p_s} = \left[v^{1,1}_{g,p_s,1}\cdots v^{1,1}_{g,p_s,A} b^{2,1}_{g,p_s,1}\cdots v^{2,1}_{g,p_s,A} \cdots\ v^{U,1}_{g,p_s,1}\cdots v^{U,1}_{g,p_s,A}\right]^{\rm T} \quad (34)$$

where $v^{u,0}_{g,p_s,l}$ represents the probability of the bit $\widehat{b}^u_{g,p_s,l}$ being changed to 0, whereas $v^{u,1}_{g,p_s,l}$ represents the probability of the bit $\widehat{b}^u_{g,p_s,l}$ being changed to 1. The cognitive information on the p_sth individual is denoted as $\widehat{\mathbf{b}}^{\rm ci}_{g,p_s}$, and the social information on the swarm is expressed as $\widehat{\mathbf{b}}^{\rm si}_g$. The DBPSO-MUD algorithm is presented as follows.

Algorithm 6: DBPSO-MUD.

1) **Initialization**. Set the generation index to $g=1$. Ran domly generate the initial population $\{\widehat{\mathbf{b}}_{1,p_s}\}_{p_s=1}^{P_s}$ and randomly generate the two initial sets of the bit-toggling probability vectors, i.e., $\{\mathbf{v}^0_{1,p_s}\}_{p_s=1}^{P_s}$ and $\{\mathbf{v}^1_{1,p_s}\}_{p_s=1}^{P_s}$, over the probability space $[0,1]^{UA}$.
2) **Swarm evaluation**. For each $\widehat{\mathbf{b}}_{g,p_s}$, compute its CF value $J_{\rm mud}(\widehat{\mathbf{b}}_{g,p_s})$. Then, update the cognitive information $\{\widehat{\mathbf{b}}^{\rm ci}_{g,p_s}\}_{p_s=1}^{P_s}$ and the swarm's social information $\widehat{\mathbf{b}}^{\rm si}_g$.
3) **Swarm updating**. The two sets of the bit-toggling probability vectors are updated according to [42]

$$\mathbf{v}^0_{g+1,p_s} = \omega\mathbf{v}^0_{g,p_s} + c_1\, rand(0,1)\left(\mathbf{1}_{UA}-2\widehat{\mathbf{b}}^{\rm ci}_{g,p_s}\right) +\ c_2\, rand(0,1)\left(\mathbf{1}_{UA}-2\widehat{\mathbf{b}}^{\rm si}_g\right) \quad (35)$$

$$\mathbf{v}^1_{g+1,p_s} = \omega\mathbf{v}^1_{g,p_s} + c_1\, rand(0,1)\left(2\widehat{\mathbf{b}}^{\rm ci}_{g,p_s}-\mathbf{1}_{UA}\right) +\ c_2\, rand(0,1)\left(2\widehat{\mathbf{b}}^{\rm si}_g-\mathbf{1}_{UA}\right) \quad (36)$$

for $1\le p_s\le P_s$, where $\mathbf{1}_{UA}$ is the UA-element vector, whose elements are all equal to 1; ω is the inertia weight; and c_1 and c_2 are the cognitive learning rate and the social learning rate, respectively. The velocities associated with $\widehat{\mathbf{b}}_{g,p_s}$, for $1 \le p_s \le P_s$, are calculated as follows. Define the intermediate velocity of the bit $\widehat{b}^u_{g,p_s,l}$, where $1 \le l \le A$ and $1 \le u \le U$, as [42]

$$\tilde{v}^u_{g+1,p_s,l} = \begin{cases} v^{u,1}_{g+1,p_s,l}, & \text{if } \widehat{b}^u_{g,p_s,l} = 0 \\ v^{u,0}_{g+1,p_s,l}, & \text{if } \widehat{b}^u_{g,p_s,l} = 1 \end{cases} \tag{37}$$

which is then used to generate the velocity associated with $\widehat{b}^u_{g,p_s,l}$ according to [41]

$$v^u_{g+1,p_s,l} = \frac{1}{1 + e^{-\tilde{v}^u_{g+1,p_s,l}}}. \tag{38}$$

Next, the individuals are updated as follows:

$$\widehat{b}^u_{g+1,p_s,l} = \begin{cases} \widehat{b}^u_{g,p_s,l}, & \text{if } rand(0,1) \le v^u_{g+1,p_s,l} \\ 1 - \widehat{b}^u_{g,p_s,l}, & \text{if } rand(0,1) > v^u_{g+1,p_s,l} \end{cases} \tag{39}$$

for $1 \le p_s \le P_s$, $1 \le u \le U$, and $1 \le l \le A$.

4) **Termination**. Optimization is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g + 1$, and go to 2) **Swarm evaluation**.

The key algorithmic parameters of this DBPSO-MUD are population size P_s, cognitive learning rate c_1, and social learning rate c_2.

D. DEA for Iterative CE and MUD

1) CDEA-CE: The operations of the CDEA-CE are shown in Fig. 5. More explicitly, the CDEA-CE scheme is elaborated in Algorithm 7.

Algorithm 7: CDEA-CE.

---

1) **Initialization**. Set $g = 1$ and randomly generate the initial $\{\widehat{\mathbf{h}}_{q,g,p_s}\}_{p_s=1}^{P_s}$. The mean of crossover probability C_r is initialized to $\mu_{C_r} = 0.5$, whereas the location parameter of scaling factor λ is initialized to $\mu_\lambda = 0.5$. The archive of the DEA is initialized to be empty.

2) **Population evaluation**. For each $\widehat{\mathbf{h}}_{q,g,p_s}$, where $1 \le p_s \le P_s$, evaluate the CF value $J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,p_s})$. The archive of DEA contains the P_s best solutions that the population has found, and it is updated every generation by adding the $\lfloor P_s \cdot p \rfloor$ parent solutions that are in the top $100{\cdot}p\%$ of high fitness to it, where p is known as the greedy factor. If the archive size exceeds P_s, some solutions are randomly removed from it.

3) **Mutation**. As shown in Step 3) of Fig. 5, the mutation perturbs the candidate solutions by adding randomly selected and appropriately scaled difference-vectors to each base population vector $\widehat{\mathbf{h}}_{q,g,p_s}$ as follows:

$$\widetilde{\mathbf{h}}_{q,g,p_s} = \widehat{\mathbf{h}}_{q,g,p_s} + \lambda_{p_s}(\widehat{\mathbf{h}}^p_{q,g,\text{best},r_1} - \widehat{\mathbf{h}}_{q,g,p_s}) + \lambda_{p_s}(\widehat{\mathbf{h}}_{q,g,r_2} - \widehat{\mathbf{h}}_{q,g,r_3}) \tag{40}$$

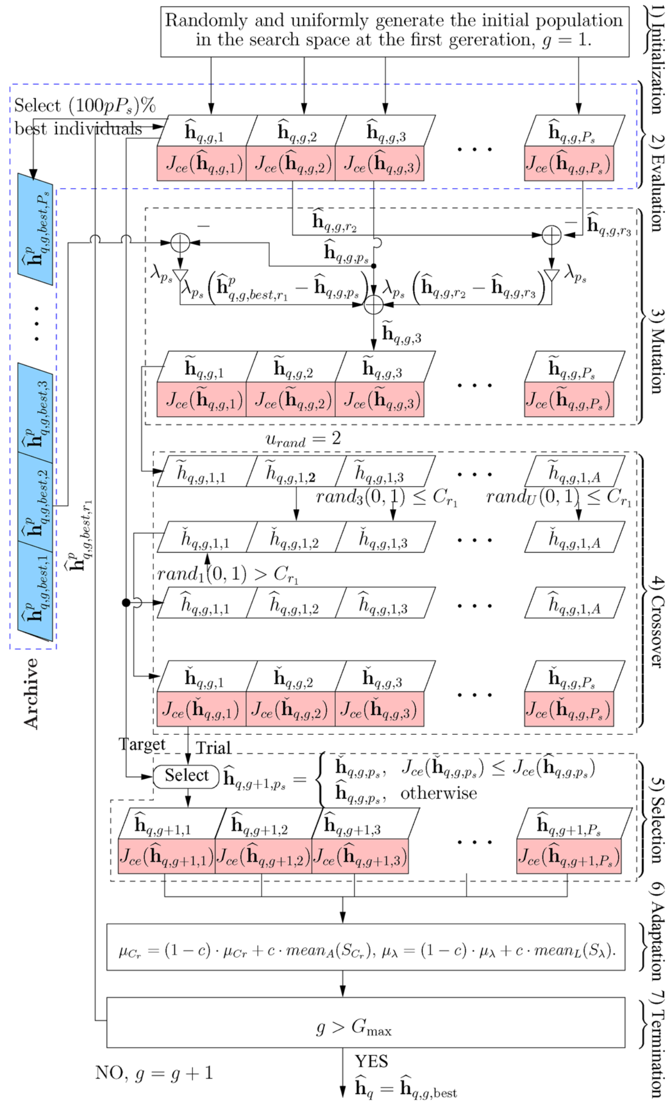


Fig. 5. Flowchart of the continuous-DEA-assisted CE.

where scaling factor $\lambda_{p_s} \in (0, 1]$ is a positive number, which is randomly generated for each individual according to the normal distribution having a mean of μ_λ and a standard deviation of 0.1; $\widehat{\mathbf{h}}^p_{q,g,\text{best},r_1}$ is a randomly selected archive value; and r_2 and r_3 are two random integer values fetched from the set $\{1, 2, \ldots, (p_s - 1), (p_s + 1), \ldots, P_s\}$.

4) **Crossover**. A trial vector $\check{\mathbf{h}}_{q,g,p_s}$ is generated upon replacing certain elements of the target vector $\widehat{\mathbf{h}}_{q,g,p_s}$ by the corresponding elements of the related donor vector $\widetilde{\mathbf{h}}_{q,g,p_s}$, which is illustrated in Step 4) of Fig. 5. Specifically, the (u, l)th element of the p_sth trial vector $\check{\mathbf{h}}_{q,g,p_s}$, $\check{h}^u_{q,g,p_s,l}$, is given by

$$\check{h}_{q,g,p_s,l} = \begin{cases} \widetilde{h}^u_{q,g,p_s,l}, & rand(0,1) \le C_{r_{p_s}} \\ \widehat{h}^u_{q,g,p_s,l}, & \text{otherwise} \end{cases} \tag{41}$$

where $C_{r_{p_s}} \in [0, 1]$ is the randomly generated crossover probability for each individual according to the Cauchy distribution with location parameter μ_{C_r} and scale parameter 0.1.

5) **Selection**. If $J_{\text{ce}}(\check{\mathbf{h}}_{q,g,p_s}) \leq J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,p_s})$, the trial vector survives to the next generation and $\widehat{\mathbf{h}}_{q,(g+1),p_s} = \check{\mathbf{h}}_{q,g,p_s}$. Otherwise, the target vector survives and $\widehat{\mathbf{h}}_{q,(g+1),p_s} = \widehat{\mathbf{h}}_{q,g,p_s}$.

6) **Adaptation**. The mean of crossover probability μ_{C_r} and the location parameter of scaling factor μ_λ are updated according to [23]

$$\mu_{C_r} = (1-c)\cdot\mu_{C_r} + c\cdot mean_A(S_{C_r}) \tag{42}$$

$$\mu_\lambda = (1-c)\cdot\mu_\lambda + c\cdot mean_L(S_\lambda) \tag{43}$$

where $c \in (0,1]$ is the adaptive update factor, $mean_A(\cdot)$ and $mean_L(\cdot)$ denote the arithmetic-mean and Lehmer-mean [23] operators, and S_{C_r} and S_λ denote the sets of successful crossover probabilities C_{r_i} and scaling factors λ_i in generation g.

7) **Termination**. The procedure is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g=g+1$, and go to 2) **Population evaluation**.

The key algorithmic parameters of this CDEA-CE are population size P_s, greedy factor p, and adaptive update factor c.

2) DBDEA-MUD: The DBDEA-MUD is described as follows.

Algorithm 8: DBDEA-MUD.

---

1) **Initialization**. With the generation index set to $g=1$, randomly generate the initial population $\{\widehat{\mathbf{b}}_{g,p_s}\}_{p_s=1}^{P_s}$. Set $\mu_{C_r}=0.5$ and $\mu_\lambda = 0.5$.

2) **Population evaluation**. For each $\widehat{\mathbf{b}}_{g,p_s}$, where $1 \leq p_s \leq P_s$, evaluate the CF value $J_{\text{mud}}(\widehat{\mathbf{b}}_{g,p_s}) = J_{\text{mud}}(\widehat{\mathbf{X}}^b_{g,p_s})$, where $\widehat{\mathbf{X}}^b_{g,p_s}$ is the M-QAM symbol vector generated from $\widehat{\mathbf{b}}_{g,p_s}$. The archive, which contains the P_s best solutions that the population has explored, is updated every generation by adding the $\lfloor P_s \cdot p \rfloor$ parent solutions that are in the top $100{\cdot}p\%$ of high fitness to the archive, where again, p is the greedy factor. If the archive size exceeds P_s, some solutions are randomly removed from it.

3) **Mutation**. The mutant version of base vector $\widehat{\mathbf{b}}_{g,i}$ is created according to

$$\widehat{\mathbf{v}}_{g,i} = \widehat{\mathbf{b}}_{g,i} \oplus \left(\mathbf{z}_i^b \otimes \left(\widehat{\mathbf{b}}^p_{g,\text{best},r_1} \oplus \widehat{\mathbf{b}}_{g,i}\right)\right) \oplus \left(\mathbf{z}_i^b \otimes \left(\widehat{\mathbf{b}}_{g,r_2} \oplus \widehat{\mathbf{b}}_{g,r_3}\right)\right) \tag{44}$$

where $\widehat{\mathbf{b}}^p_{g,\text{best},r_1}$ is randomly chosen from the archive, $\widehat{\mathbf{b}}_{g,r_2}$ and $\widehat{\mathbf{b}}_{g,r_3}$ with $r_2 \neq i$ and $r_3 \neq i$ are randomly selected from the current population, $\mathbf{z}_i^b$ is a randomly generated $(U \times A)$-length binary vector known as the bit-scaling factor, $\oplus$ denotes the bitwise exclusive-OR operator, and $\otimes$ denotes the bitwise exclusive-AND operator.

4) **Crossover**. With the uniform crossover, each element of the trial vector has the same probability of inheriting its value from a given vector. Specifically, the (u,j)th element of the p_sth trial vector $\widehat{\mathbf{t}}_{g,p_s}$ at the gth generation, i.e., $\widehat{t}^u_{g,p_s,j}$, is given by

$$\widehat{t}^u_{g,p_s,j} = \begin{cases} \widehat{v}^u_{g,p_s,j}, & rand(0,1) \leq C_{r_{p_s}} \text{ or } j = j_{\text{rand}} \\ \widehat{b}^u_{g,p_s,j}, & \text{otherwise} \end{cases} \tag{45}$$

where crossover probability $C_{r_{p_s}} \in [0,1]$ is randomly generated according to the normal distribution having a mean of μ_{C_r} and a standard deviation of 0.1, whereas j_{rand} is a randomly chosen integer in the range of $\{1,2,\ldots,P_s\}$.

5) **Selection**. Let $\widehat{\mathbf{X}}^b_{g,p_s}$ and $\widehat{\mathbf{X}}^t_{g,p_s}$ be the M-QAM symbol vectors generated from $\widehat{\mathbf{b}}_{g,p_s}$ and $\widehat{\mathbf{t}}_{g,p_s}$, respectively. If $J_{\text{mud}}(\widehat{\mathbf{X}}^t_{g,p_s}) \leq J_{\text{mud}}(\widehat{\mathbf{X}}^b_{g,p_s})$, then we set $\widehat{\mathbf{b}}_{g+1,p_s} = \widehat{\mathbf{t}}_{g,p_s}$. Otherwise, we set $\widehat{\mathbf{b}}_{g+1,p_s} = \widehat{\mathbf{b}}_{g,p_s}$.

6) **Adaptation**. Given the adaptive update factor $c \in (0,1]$ specified by the designer, μ_{C_r} and μ_λ are adapted according to (42) and (43).

7. **Termination**. Optimization is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g=g+1$, and go to 2) **Population evaluation**.

The key algorithmic parameters of this DBDEA-MUD are population size P_s, greedy factor p, and adaptive update factor c.

IV. EA-Aided Iterative CE and Turbo MUD/Decoder

A. Iterative CE and Turbo MUD/Decoder

The iterative joint CE and turbo MUD/decoder is constituted by the continuous-EA-aided CE and the discrete-binary EA-assisted SISO MUD, followed by U parallel single-user SISO channel decoders, as shown within the dotted-line box at the right-hand side in Fig. 1. The operations of the EA-aided iterative CE and turbo MUD/decoder are outlined as follows.

1) **Initialization**. The training-based channel estimator uses the pilot symbols to provide an initial channel estimate for activating the iterative procedure of joint CE and turbo MUD/decoder. Set the iteration index of the joint CE and turbo MUD/decoder to $loop = 1$.

2) **Iterative CE** and **turbo MUD/decoder**.

   1) **Initialization of turbo MUD/decoder**. Forward the channel estimates provided by the "Continuous-EA-aided CIR estimator" block in Fig. 1 to the MUD, and set the iteration index of the turbo MUD/decoder to $Iter = 1$.

3) **Turbo MUD/decoder**. The discrete-binary EA-aided ML-MUD, which is shown by the central rectangle in Fig. 1, detects the users' data.

   Step-3.1). The SISO MUD delivers the *a posteriori* information on bit $b^u(i)$ expressed in terms of its log-likelihood ratio (LLR) as [2]

$$L_{m,po,b^u(i)} = \ln \frac{\Pr\left\{\widehat{X}^u \middle| b^u(i)=0\right\}}{\Pr\left\{\widehat{X}^u \middle| b^u(i)=1\right\}} + \ln \frac{\Pr\{b^u(i)=0\}}{\Pr\{b^u(i)=1\}} = L_{m,e,b^u(i)} + L_{m,pr,b^u(i)} \tag{46}$$

where $b^u(i)$ is the ith bit in the bit stream that is mapped to the M-QAM symbol stream of user u. The second term in (46), i.e., $L_{m,pr,b^u(i)}$, represents the *a priori* LLR of the interleaved and encoded bits $b^u(i)$, whereas the term $L_{m,e,b^u(i)}$ in (46) is the extrinsic information delivered by the SISO MUD, based on the received signal $\mathbf{Y}$ and the *a priori* information about the encoded bits of all users, except for the ith bit of user u.

Step-3.2). As shown in the receiver in Fig. 1, the extrinsic information output by the SISO MUD is then deinterleaved and fed into the uth user's SISO channel decoder as its *a priori* information, which is denoted as $L_{c,pr,b^u(i)}$. The uth SISO channel decoder then delivers the *a posteriori* information on decoded bits in terms of LLRs $L_{c,po,b^u(i)}$ [9], which can be expressed as $L_{c,po,b^u(i)} = L_{c,e,b^u(i)} + L_{c,pr,b^u(i)}$. The extrinsic information output by the SISO decoder, which is denoted by $L_{c,e,b^u(i)}$, will then be interleaved to provide the *a priori* information for the next iteration of the SISO MUD.

Step-3.3) Turbo MUD/decoder convergence test. If $Iter < I_{\mathrm{tb}}$, where I_{tb} defines the maximum number of turbo iterations,[2] set $Iter = Iter + 1$ and go to **Step-3.1)**. Otherwise, the turbo MUD/decoder has converged, and the detected and decoded bit streams are encoded by the channel encoders, interleaved by the interleavers, and then mapped to the corresponding M-QAM symbol streams, which will be used by the continuous-EA-based CE.

4) **Decision-directed channel estimator**.

Step-4.1) Continuous-EA-aided CE. The "Continuous-EA-aided CIR estimator" blocks in Fig. 1 use the re-encoded and remodulated data $\{\widetilde{\mathbf{X}}^u\}_{u=1}^U$ to perform CIR estimation. The resultant CIR estimate $\widehat{\mathbf{h}}$ is transformed to the FD-CHTF matrix estimate $\widehat{\mathbf{H}}$ by the FFT, which will then be used by the turbo MUD/decoder so that the iterative process can continue.

Step-4.2) CE and **turbo MUD/decoder convergence test.** If $loop < I_{\mathrm{ce}}$, where I_{ce} defines the maximum number of joint CE and turbo MUD/decoder iterations in Fig. 1, set $loop = loop + 1$ and go to **2.1)**. Otherwise, the iterative CE and turbo MUD/decoder has converged.

The *a posteriori* information on the turbo ML-MUD associated with bit $b^u(i)$ is given by [2]

$$\begin{aligned} L^{\mathrm{ML}}_{m,po,b^u(i)} &= \ln \frac{\Pr\{\mathbf{Y}, b^u(i)=0\}}{\Pr\{\mathbf{Y}, b^u(i)=1\}} \\ &= \ln \frac{\sum\limits_{\forall \mathbf{X}\in\mathcal{S}^U: b^u(i)=0} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\mathbf{X}\|^2}{2\sigma_n^2}} \prod\limits_{u=1}^{U}\prod\limits_{j=1}^{A} \Pr\{b^u(j)\}}{\sum\limits_{\forall \mathbf{X}\in\mathcal{S}^U: b^u(i)=1} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\mathbf{X}\|^2}{2\sigma_n^2}} \prod\limits_{u=1}^{U}\prod\limits_{j=1}^{A} \Pr\{b^u(j)\}} \end{aligned} \tag{47}$$

where the probability $\Pr\{b^u(j)\}$ of $b^u(j)$ is given by

$$\Pr\{b^u(j)\} = \frac{1}{2}\left(1+\mathrm{sgn}\left(\frac{1}{2}-b^u(j)\right)\tanh\left(\frac{L^{\mathrm{ML}}_{m,pr,b^u(j)}}{2}\right)\right). \tag{48}$$

Note from (47) that the $M^U = |\mathcal{S}|^U$ legitimate candidate solutions of the U users are partitioned into the two subsets conditioned on $b^u(i) = 0$ and $b^u(i) = 1$, respectively, and the complexity of calculating $L^{\mathrm{ML}}_{m,po,b^u(i)}$ exponentially increases with the size of M-QAM signaling and the number of users U.

By contrast, the discrete-binary EA-aided turbo MUD is capable of reducing the complexity of the *a posteriori* information calculation to that of a near-single-user scenario, once the transmitted data $\mathbf{X}$ are detected by the discrete-binary EA-aided MUD. Specifically, the *a posteriori* information on the discrete-binary EA-aided turbo MUD associated with bit $b^u(i)$ is given as

$$L^{\mathrm{EA}}_{m,po,b^u(i)} = \ln \frac{\sum\limits_{\forall X^u\in\mathcal{S}: b^u(i)=0} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\tilde{\mathbf{X}}\|^2}{2\sigma_n^2}} \prod\limits_{j=1}^{A} \Pr\{b^u(j)\}}{\sum\limits_{\forall X^u\in\mathcal{S}: b^u(i)=1} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\tilde{\mathbf{X}}\|^2}{2\sigma_n^2}} \prod\limits_{j=1}^{A} \Pr\{b^u(j)\}} \tag{49}$$

where $\Pr\{b^u(j)\}$ is also calculated using (48) by replacing $L^{\mathrm{ML}}_{m,po,b^u(i)}$ with $L^{\mathrm{EA}}_{m,po,b^u(i)}$, and $\widetilde{\mathbf{X}} = [\widehat{X}^1 \cdots \widehat{X}^{u-1} X^u \widehat{X}^{u+1} \cdots \widehat{X}^U]^{\mathrm{T}}$, with X^u assuming values from the M-QAM symbol set $\mathcal{S}$ and $\widehat{X}^v, v = 1, \ldots, u-1, u+1, \ldots, U$ being acquired by the discrete-binary EA-aided MUD at the first turbo iteration. Following the first turbo iteration, $\widehat{X}^v$ for $v \neq u$ is given by

$$\hat{X}^v = \max_{X^v\in\mathcal{S}} \Pr\{X^v\} = \max_{X^v\in\mathcal{S}} \prod_{j=1}^{A} \Pr\left[b^v(j)\right]. \tag{50}$$

Observe in (49) that the number of legitimate candidate solutions is $M = |\mathcal{S}|$ for each user, since the transmitted signal of user v ($v \neq u$) is given by (50). Thus, the computational complexity of the *a posteriori* information's calculation has been reduced to $M \cdot U$.

B. Convergence Discussion and Complexity Analysis

To characterize the convergence behavior of the population $\{\widehat{\mathbf{X}}_{g,p_s}\}_{p_s=1}^{P_s}$, as generation g evolves,[3] we may adopt the

[2] A turbo iteration represents one exchange of extrinsic information between the discrete-binary EA-assisted SISO MUD and the SISO channel decoder, as described in **Step 3.1)** and **Step 3.2)** and shown in Fig. 1.

[3] Although the discussion only refers to the discrete-binary EA-assisted MUD, it also makes sense for the continuous-EA-aided CE.

probability of convergence, which is defined as [43]

$$\lim_{g\to+\infty} \Pr\left\{\left\|\hat{\mathbf{X}}_{g,p_s} - \mathbf{X}_{\mathrm{ML}}\right\| > \epsilon\right\} = 0, \quad \forall p_s \tag{51}$$

where $\mathbf{X}_{\mathrm{ML}}$ denotes the optimal ML solution, and ϵ is an arbitrary positive value. The probability of convergence defined in (51) requires that the solutions are located outside the ϵ-neighborhood of $\mathbf{X}_{\mathrm{ML}}$ with a probability of zero, as the population evolves. Generally, there exists a probability $p(g) > 0$ at each generation g that the individuals in the parental population will generate an offspring belonging to the ϵ-neighborhood of $\mathbf{X}_{\mathrm{ML}}$. As a benefit of the elitism, the individuals of the next generation are as good as or better than their counterparts in the current generation, which indicates that sequence $\{p(g)\}$ is monotonically increasing. This leads to [43]

$$\lim_{g\to+\infty} \Pr\left\{\left\|\hat{\mathbf{X}}_{g,p_s} - \mathbf{X}_{\mathrm{ML}}\right\| < \epsilon\right\} = 1, \quad \forall p_s. \tag{52}$$

The given proposition indicates that the population will converge to the ϵ-neighborhood of $\mathbf{X}_{\mathrm{ML}}$ with a probability of 1, but does not address the vital question of convergence speed. As we use an EA to solve an NP-hard optimization problem, whose optimal solution by the "brute force" exhaustive ML search imposes an exponentially increasing complexity in the problem size. Vast amounts of empirical results found in the literature have demonstrated that appropriately tuned EAs are capable of approaching the globally optimal solutions even for the most challenging optimization problems at affordable complexity. Moreover, the theoretical analysis of EAs has made significant progress in the past few years [44]. Specifically, many NP-hard problems can be turned into the so-called EA-easy class [44], implying that they can be solved by a well-tuned EA algorithm at complexity at most polynomial in the problem size.

Given the CSI, i.e., $\mathbf{h}$, the computational complexity of a turbo MUD/decoder is given by

$$C_{\mathrm{turbo}} = I_{\mathrm{tb}} \cdot C_{\mathrm{MUD}} + I_{\mathrm{tb}} \cdot C_{\mathrm{dec}} \tag{53}$$

where C_{MUD} and C_{dec} are the complexity of the turbo MUD and that of the channel decoder, respectively. The second term in (53) remains the same for both the turbo ML-MUD/decoder and the turbo EA-aided MUD/decoder. Furthermore, the second term in (53) is significantly smaller than the first term. The complexity $C_{\mathrm{MUD}}^{\mathrm{ML}}$ of the turbo ML-MUD/decoder imposed by detecting a frame of S OFDM symbols, each having K subcarriers, can be shown to be (54), shown at the bottom of the page, whereas the complexity $C_{\mathrm{MUD}}^{\mathrm{EA}}$ of the turbo EA-aided MUD/decoder can be shown to be (55), shown at the bottom of the page.

The total complexity of the EA-assisted joint CE and turbo MUD/decoder is given by

$$C_{\mathrm{joint}}^{\mathrm{EA}} = I_{\mathrm{ce}} \cdot \left(C_{\mathrm{turbo}}^{\mathrm{EA}} + C_{\mathrm{one-mud}}^{\mathrm{EA}} + C_{\mathrm{ce}}^{\mathrm{EA}}\right). \tag{56}$$

In (56), $C_{\mathrm{ce}}^{\mathrm{EA}}$ denotes the complexity of the continuous-EA-based CE, which is specified by the number $N_{\mathrm{CF-EVs}}^{\mathrm{ce}}$ of $J_{\mathrm{ce}}(\bullet)$ CF evaluations and the complexity per CF evaluation. Given the population size P_s^{ce} and the maximum number of generations $G_{\max}^{\mathrm{ce}}$, we have $N_{\mathrm{CF-EVs}}^{\mathrm{ce}} \approx P_s^{\mathrm{ce}} \cdot G_{\max}^{\mathrm{ce}}$ for all the four continuous-EA-based CEs,[4] whereas the complexity per $J_{\mathrm{ce}}(\bullet)$ CF evaluation may be derived according to (12) as

$$\begin{cases} 4KS(UL_{\mathrm{cir}} + U + 1) & \text{multiplications} \\ KS(5UL_{\mathrm{cir}} + 3U + 3) & \text{additions.} \end{cases} \tag{57}$$

The term $C_{\mathrm{one-mud}}^{\mathrm{EA}}$ represents the complexity imposed by the discrete-binary EA-aided MUD at each outer iteration loop, which is specified by the number of $J_{\mathrm{mud}}(\bullet)$ CF evaluations $N_{\mathrm{CF-EVs}}^{\mathrm{mud}} \approx P_s^{\mathrm{mud}} \cdot G_{\max}^{\mathrm{mud}}$ for all the four discrete-binary EA-aided MUDs,[5] where P_s^{mud} is the population size, and $G_{\max}^{\mathrm{mud}}$ is the maximum number of generations, as well as the complexity per $J_{\mathrm{mud}}(\bullet)$ CF evaluation, which can be determined according to (16) as

$$\begin{cases} 4KSQU & \text{multiplications} \\ KS(3QU + Q + U - 1) & \text{additions.} \end{cases} \tag{58}$$

The ratio of the complexity of the EA-assisted joint CE and turbo MUD/decoder to that of the idealized turbo

[4]For the CRWBS-CE, $N_{\mathrm{CF-EVs}}^{\mathrm{ce}} = ((P_s^{\mathrm{ce}} - 1) + 2T_{\mathrm{wbs}}) \cdot G_{\max}^{\mathrm{ce}}$. The approximation is met by appropriately choosing T_{wbs}.

[5]Again, the approximation holds for the DBRWBS-MUD by appropriately choosing the number of WBS iterations.

$$\begin{cases} KS\left(2UM^U(2Q\log_2 M + 2Q + \log_2 M) + U\log_2 M \right. \\ \quad \left. + MU(4\log_2 M - 1)\right) & \text{multiplications} \\ KS\left(M^U(4QU\log_2 M + 4QU - 2U\log_2 M - Q)\right. \\ \quad \left. + 2U(M-1)\log_2 M\right) & \text{additions} \end{cases} \tag{54}$$

$$\begin{cases} KS\left(MU(4QU(\log_2 M + 1) + 2U\log_2 M \right. \\ \quad \left. + 4\log_2 M - 1) + U\log_2 M\right) & \text{multiplications} \\ KS\left(MU(4QU(\log_2 M + 1) - 2U\log_2 M - Q)\right. \\ \quad \left. + 2\log_2 M)2U\log_2 M\right) & \text{additions} \end{cases} \tag{55}$$

TABLE I
SIMULATION PARAMETERS OF THE MULTIUSER OFDM/SDMA SYSTEM

Encoder	Type	RSC
	Code rate	1/2
	Constraint length	3
	Polynomial	$(g_0, g_1) = (7, 5)$
Channel	Number of paths L_{cir}	4
	Delays	$\{0, 1, 2, 3\}$
	Average path gains	$\{0, -5, -10, -15\}$ (dB)
	Taps: frame to frame	Complex white Gaussian
	Taps: within frame	fading rate $F_D = 10^{-7}$
System	MSs U	4
	Receiver antennas Q	3
	Modulation	16-QAM
	Subcarriers K	64
	Cyclic prefix K_{cp}	16

ML-MUD/decoder associated with perfect CSI is expressed by

$$\begin{aligned}\frac{C^{\mathrm{EA}}_{\mathrm{joint}}}{C^{\mathrm{ML}}_{\mathrm{turbo}}} &= \frac{I_{\mathrm{ce}}\cdot\left(C^{\mathrm{EA}}_{\mathrm{turbo}}+C^{\mathrm{EA}}_{\mathrm{one-mud}}+C^{\mathrm{EA}}_{\mathrm{ce}}\right)}{I_{\mathrm{tb}}\cdot\left(C^{\mathrm{ML}}_{\mathrm{MUD}}+C_{\mathrm{dec}}\right)}\\ &\approx \frac{I_{\mathrm{ce}}\cdot\left(I_{\mathrm{tb}}\cdot C^{\mathrm{EA}}_{\mathrm{MUD}}+C^{\mathrm{EA}}_{\mathrm{one-mud}}+C^{\mathrm{EA}}_{\mathrm{ce}}\right)}{I_{\mathrm{tb}}\cdot C^{\mathrm{ML}}_{\mathrm{MUD}}}\end{aligned} \tag{59}$$

where the approximation is obtained by omitting the second term in (53).

V. EXPERIMENTAL PERFORMANCE RESULTS

The parameters of our simulated multiuser SDMA/OFDM UL are listed in Table I. A four-path Rayleigh fading channel model was employed for each link, and the delays of the paths were normalized to the sample duration. At the beginning of every frame, which contained $S = 100$ OFDM symbols, a new channel tap was generated for each of the four paths according to the complex-valued white Gaussian process with its power specified by the corresponding average path gain. Within the frame, each channel tap experienced independent Rayleigh fading having the same normalized Doppler frequency of $F_D = 10^{-7}$. A half-rate recursive systematic convolutional code was employed as the channel code. The default values of the EAs' algorithmic parameters are listed in Table II. The first OFDM symbol of each frame was populated with pilots for the initial-training-based CE, yielding a training overhead of 1%. The system's signal-to-noise ratio (SNR) was specified by SNR $= E_{\mathrm{b}}/N_{\mathrm{o}}$ in decibels, where E_{b} denotes the energy per bit, and N_{o} is the power spectral density of the channel AWGN.

A. Efficiency, Reliability, and Convergence Investigation

We first quantified the efficiency and reliability of the continuous-EA-aided CEs and the discrete-binary EA-based MUD schemes separately over $N_{\mathrm{tot}} = 1000$ independent simulation runs. Perfect CSI was assumed for evaluating the discrete-binary EA-assisted MUD schemes, while the transmitted data were available, when evaluating the continuous-EA-aided CE schemes. There was no information exchange between the MUD and the decoder, i.e., we had $I_{\mathrm{tb}} = 1$, and the channel's AWGN had $N_{\mathrm{o}} = 0$. For an EA-aided CE scheme, we declared a "successful" run when the algorithm achieved the CF value of $J_{\mathrm{ce}}(\widehat{\mathbf{h}}_{q,G^i_{\max},\mathrm{best}}) < 10^{-4}$ within the set upper limit for the number of CF evaluations $\overline{N}^{\mathrm{lim}}_{\mathrm{CF-EVs}} = P_s \cdot G^{\mathrm{lim}}_{\max} = 100 \times 1000$, where $G^i_{\max}$ denotes the number of generations in the ith simulation run. Otherwise, the run was declared as "failed." Over the $N_{\mathrm{tot}} = 1000$ simulation runs, we collected the statistics of the number of successful runs, denoted as N_{suc}; the number of failed runs, denoted as N_{fail}; the total number of CF evaluations in the N_{suc} successful runs, defined by $N^{\mathrm{suc}}_{\mathrm{CF-EVs}}$; and the total number of CF evaluations in the N_{fail} failed runs, defined by $N^{\mathrm{fail}}_{\mathrm{CF-EVs}}$, using the following.

for run $= 1 : N_{\mathrm{tot}}$
  if $(G^{\mathrm{run}}_{\max} \le G^{\mathrm{lim}}_{\max})$ and $(J_{\mathrm{ce}}(\widehat{\mathbf{h}}_{q,G^{\mathrm{run}}_{\max},\mathrm{best}}) < 10^{-4})$
    $N_{\mathrm{suc}} = N_{\mathrm{suc}} + 1$; $N^{\mathrm{suc}}_{\mathrm{CF-EVs}} = N^{\mathrm{suc}}_{\mathrm{CF-EVs}} + P_s \cdot G^{\mathrm{run}}_{\max}$,
  else
    $N_{\mathrm{fail}} = N_{\mathrm{fail}} + 1$; $N^{\mathrm{fail}}_{\mathrm{CF-EVs}} = N^{\mathrm{fail}}_{\mathrm{CF-EVs}} + P_s \cdot G^{\mathrm{lim}}_{\max}$.

After obtaining these statistics, the average number of CF evaluations per run was given by

$$\overline{N}^{\mathrm{tot}}_{\mathrm{CF-EV}_s} = \left(N^{\mathrm{suc}}_{\mathrm{CF-EV}_s} + N^{\mathrm{fail}}_{\mathrm{CF-EV}_s}\right)/N_{\mathrm{tot}} \tag{60}$$

while the average number of CF evaluations per successful run was defined by

$$\overline{N}^{\mathrm{suc}}_{\mathrm{CF-EV}_s} = N^{\mathrm{suc}}_{\mathrm{CF-EV}_s}/N_{\mathrm{suc}}. \tag{61}$$

Then, the normalized average number of CF evaluations per run was formulated as

$$\overline{R}^{\mathrm{tot}}_{\mathrm{CF-EV}_s} = \overline{N}^{\mathrm{tot}}_{\mathrm{CF-EV}_s}/\overline{N}^{\mathrm{lim}}_{\mathrm{CF-EV}_s} \tag{62}$$

and the normalized average number of CF evaluations per successful run was defined as

$$\overline{R}^{\mathrm{suc}}_{\mathrm{CF-EV}_s} = \overline{N}^{\mathrm{suc}}_{\mathrm{CF-EV}_s}/\overline{N}^{\mathrm{lim}}_{\mathrm{CF-EV}_s} \tag{63}$$

offered the metrics for quantifying the efficiency of the EA-aided CE scheme investigated. The smaller $\overline{R}^{\mathrm{tot}}_{\mathrm{CF-EVs}}$ or $\overline{R}^{\mathrm{suc}}_{\mathrm{CF-EVs}}$, the more efficient the EA-aided CE scheme. On the other hand, the reliability of the EA-aided CE was measured by the failure ratio, i.e.,

$$R_{\mathrm{fail}} = N_{\mathrm{fail}}/N_{\mathrm{tot}}. \tag{64}$$

The lower R_{fail}, the more reliable the EA-aided CE scheme. The efficiency and reliability of the four continuous-EA-assisted CE schemes are shown in Fig. 6, where it can be seen that the CDEA-CE outperformed the other three schemes, and the former always arrived at the target CF value within the average computational complexity of 15 000 CF evaluations. The CRWBS-CE came a close second, and it always attained the target CF value within the average complexity of 22 000 CF evaluations. The CGA-CE was the the worst CE candidate, having the failure rate of $R_{\mathrm{fail}} \approx 7\%$ and imposing an average computational complexity of 90 000 CF evaluations.

A similar procedure was carried out for investigating the efficiency and reliability of the four discrete-binary EA-assisted MUDs by setting $G^{\mathrm{lim}}_{\max} = 500$ and $\overline{N}^{\mathrm{lim}}_{\mathrm{CF-EVs}} = M^U = 16^4$. A successful detection run was confirmed, if $(G^{\mathrm{run}}_{\max} \le G^{\mathrm{lim}}_{\max})$ and

TABLE II
ALGORITHMIC PARAMETERS FOR THE EA-ASSISTED CE AND MUD

Scheme	Parameter	Value	Scheme	Parameter	Value
CGA-CE	Population size P_s	100	DBGA-MUD	Population size P_s	100
	Selection ratio r_s	0.5		Selection ratio r_s	0.5
	Mutation parameter γ	0.01		Mutation probability M_b	0.15
	Mutation probability M_b	0.2			
CRWBS-CE	Population size P_s	100	DBRWBS-MUD	Population size P_s	100
	Mutation parameter γ	0.001		Mutation probability M_b	0.5
	Weighted boosting search T_{wbs}	40		Weighted boosting search T_{wbs}	40
CPSO-C	Population size P_s	100	DBPSO-MUD	Population size P_s	100
	Cognition learning rate c_1	2		Cognition learning rate c_1	0.1
	Social learning rate c_2	2		Social learning rate c_2	0.3
CDEA-CE	Population size P_s	100	DBDEA-MUD	Population size P_s	100
	Greedy factor p	0.1		Greedy factor p	0.7
	Adaptive update factor c	0.1		Adaptive update factor c	0.8

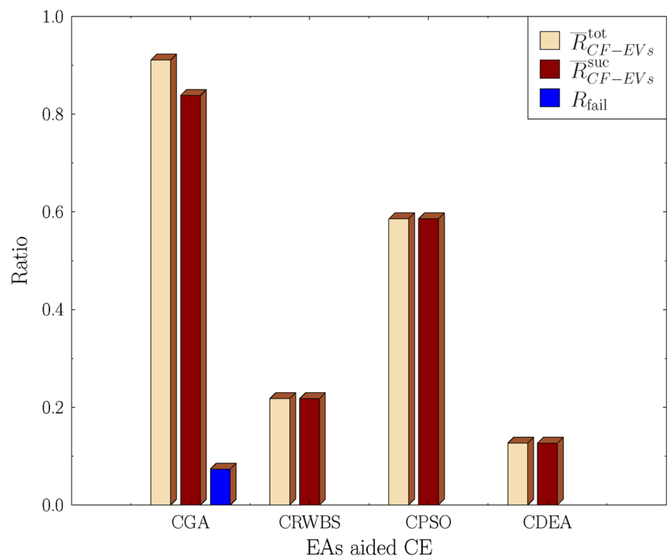


Fig. 6. Histograms of the efficiency and reliability measures, in terms of $\overline{R}^{\rm tot}_{\rm CF-EVs}$, $\overline{R}^{\rm suc}_{\rm CF-EVs}$, and $R_{\rm fail}$, for the four continuous-EA-assisted CE schemes.

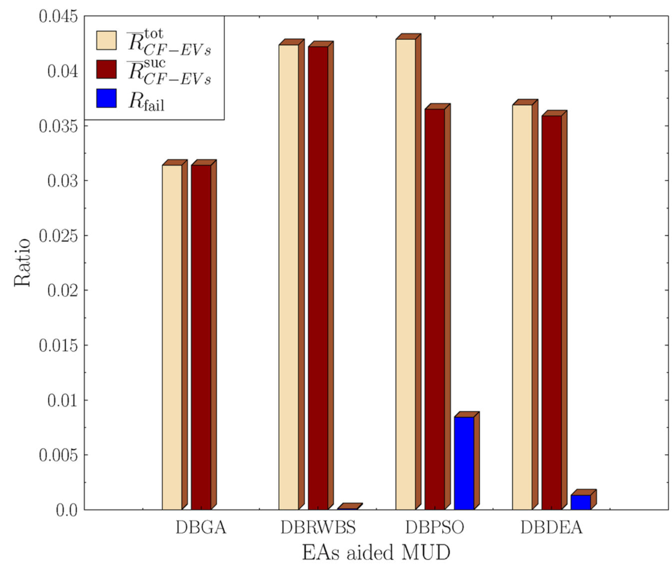


Fig. 7. Histograms of the efficiency and reliability measures, in terms of $\overline{R}^{\rm tot}_{\rm CF-EVs}$, $\overline{R}^{\rm suc}_{\rm CF-EVs}$, and $R_{\rm fail}$, for the four discrete-binary EA-assisted MUDs.

the BER of the best individual $\widehat{\mathbf{X}}_{G^{\rm run}_{\rm max},{\rm best}}$ was infinitesimally low. Otherwise, the run was declared a failure. Note that $\overline{N}^{\rm lim}_{\rm CF-EVs} = M^U$ was the number of CF evaluations required by the full-search ML MUD. Fig. 7 compares the efficiency and reliability of the four discrete-binary EA-assisted MUDs. Observe that the DBGA-MUD was the winner with a zero failure rate and requiring only 3.2% of the ML-MUD's complexity. The DBDEA-MUD came a close second with an extremely low failure rate and an average complexity that was 3.7% of the optimal ML-MUD's complexity.

We then added the channel's AWGN and considered the cases of $E_{\rm b}/N_{\rm o} = 14$ and 20 dB. Fig. 8 compares the convergence behaviors of the four continuous-EA-assisted CE schemes. The approximate number of CF evaluations required for the mean square error (MSE) of a continuous-EA-assisted CE scheme to approach the CRLB[6] [39] was extracted in Fig. 8 and listed in Table III. It can be seen that the CRWBS-CE and the CDEA-CE had the fastest convergence speed, whereas the CGA-CE had the slowest convergence speed. Fig. 9 characterizes the convergence behaviors of the four discrete-binary

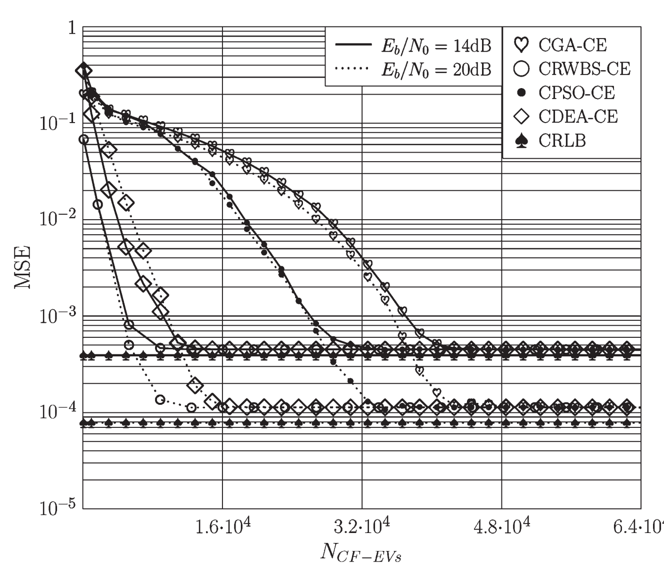


Fig. 8. MSE versus the number of CF evaluations, which characterizes the convergence performance of the different continuous-EA-assisted CE schemes.

[6]The CRLB [39] provides the best attainable MSE performance for the optimal channel estimator based on the optimally designed pilots, and it is given by $\text{CRLB}(\mathbf{h}) = (\sigma_n^2/KE_{\rm s})$ (e.g., [34]), where $E_{\rm s}$ denotes the average symbol energy.

TABLE III
NUMBERS OF CF EVALUATIONS REQUIRED FOR THE MSES OF DIFFERENT CONTINUOUS-EA-ASSISTED CE SCHEMES TO APPROACH THE CRLB

Scheme	$E_b/N_o = 14$ dB	$E_b/N_o = 20$ dB
CGA-CE	43000	44000
CRWBS-CE	10000	13000
CPSO-CE	34000	36000
CDEA-CE	12000	17000

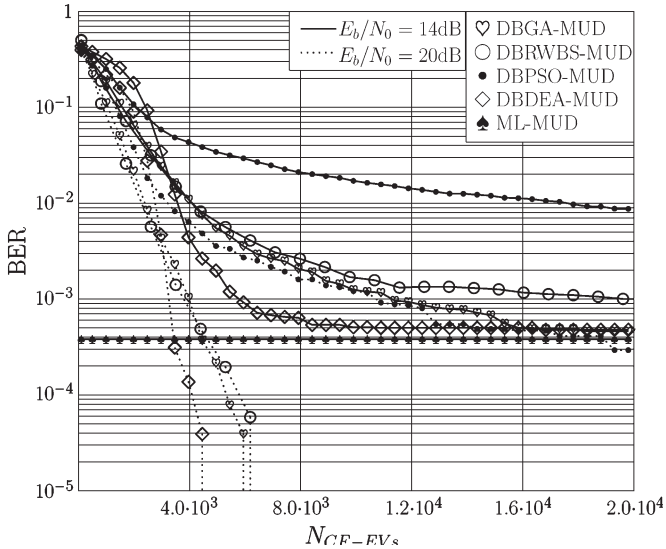


Fig. 9. BER versus the number of CF evaluations, which characterizes the convergence performance of the different discrete-binary EA-assisted MUDs. Note that at $E_b/N_o = 20$ dB, the optimal ML-MUD attains an infinitesimally low BER.

TABLE IV
NUMBERS OF CF EVALUATIONS REQUIRED FOR THE BERS OF DIFFERENT DISCRETE-BINARY EA-ASSISTED MUDS TO ATTAIN THE BER OF THE OPTIMAL ML-MUD

Scheme	$E_b/N_o = 14$ dB	$E_b/N_o = 20$ dB
DBGA-MUD	16000	6000
DBRWBS-MUD	> 20000	6500
DBPSO-MUD	failed	failed
DBDEA-MUD	10000	4500

EA-assisted MUDs. The approximate number of CF evaluations required for the BER of a discrete-binary EA-assisted MUD to approach the BER of the optimal ML-MUD was found in Fig. 9, and it is shown in Table IV. Observe that the DBDEA-MUD and the DBGA-MUD achieved rapid convergence. Although the nonturbo DBPSO-MUD failed to approach the ML-MUD solution in this experiment, by introducing the powerful turbo iterative procedure, the turbo DBPSO-MUD/decoder is capable of attaining the optimal solution of the turbo ML-MUD/decoder, as will be confirmed in Section V-B.

B. Performance of EA-Aided Joint CE and Turbo MUD/Decoder Schemes

Having examined the individual EA-assisted CE schemes and the individual EA-aided MUDs, we investigated the four EA-aided iterative joint CE and turbo MUD/decoder schemes, as outlined in Section IV, namely, the GA-aided joint CE and turbo MUD/decoder, the RWBS-aided joint CE and turbo MUD/decoder, the PSO-aided joint CE and turbo MUD/decoder, and the DEA-aided joint CE and turbo MUD/decoder. In an EA-aided joint CE and turbo MUD/decoder, the information is exchanged $I_{\rm tb}$ times at the inner turbo loop between the EA-assisted MUD and the channel decoder, whereas the information is exchanged $I_{\rm ce}$ times at the outer iterative loop between the EA-assisted CE scheme and the EA-aided turbo MUD/decoder. It is worth emphasizing that the EA-assisted channel estimator is based on the detected data fed back from the EA-assisted MUD/decoder. The MSE of the channel estimate obtained by an EA-aided joint CE and turbo MUD/decoder was compared with the CRLB, whereas the BER achieved by an EA-aided joint CE and turbo MUD/decoder was compared with the BER of the idealized turbo ML-MUD/decoder associated with perfect CSI.

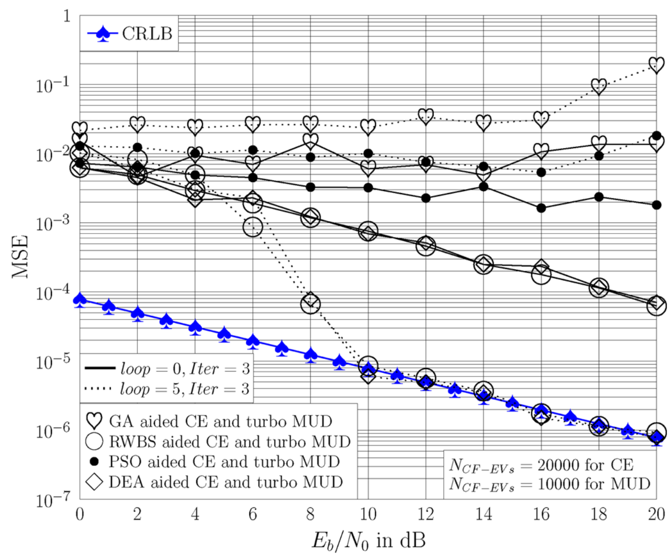


Fig. 10. Comparison of the MSE performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 20 000, and the number of CF evaluations for EA-aided MUD to 10 000.

Figs. 10 and 11 compare the MSE and BER performance, respectively, of the four EA-aided iterative joint CE and turbo MUD/decoder schemes, when fixing the number of the inner turbo iterations to $I_{\rm tb} = 3$, the number of CF evaluations for EA-aided CE to $N^{\rm ce}_{\rm CF-EVs} = 20000$ ($G_{\max} = 200$), and the number of CF evaluations for EA-aided MUD to $N^{\rm mud}_{\rm CF-EVs} = 10000$ ($G_{\max} = 100$). Observe in Fig. 10 that for $loop = 5$ outer iterations, the MSEs of the two channel estimates associated with the RWBS- and DEA-aided joint CE and turbo MUD/decoder schemes approached the CRLB for $E_b/N_o \geq 10$ dB; however, the PSO- and GA-aided joint CE and turbo MUD/decoder schemes exhibited divergence. Similarly, it is shown in Fig. 11 that for five outer iterations, the RWBS- and DEA-aided joint CE and turbo MUD/decoder schemes approached the BER performance of the idealized turbo ML-MUD/decoder; however, the PSO- and GA-aided joint CE and turbo MUD/decoder schemes failed to find the optimal solution.

From the results in Section V-A, we note that the PSO- and GA-aided joint CE and turbo MUD/decoder schemes

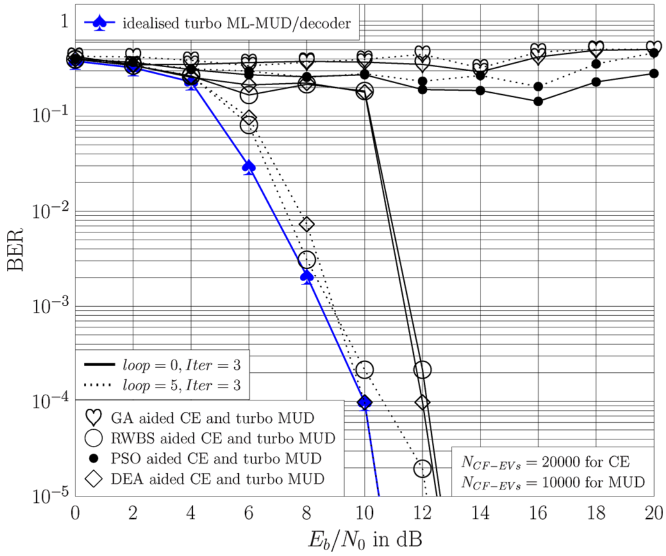


Fig. 11. Comparison of the BER performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 20 000, and the number of CF evaluations for EA-aided MUD to 10 000.

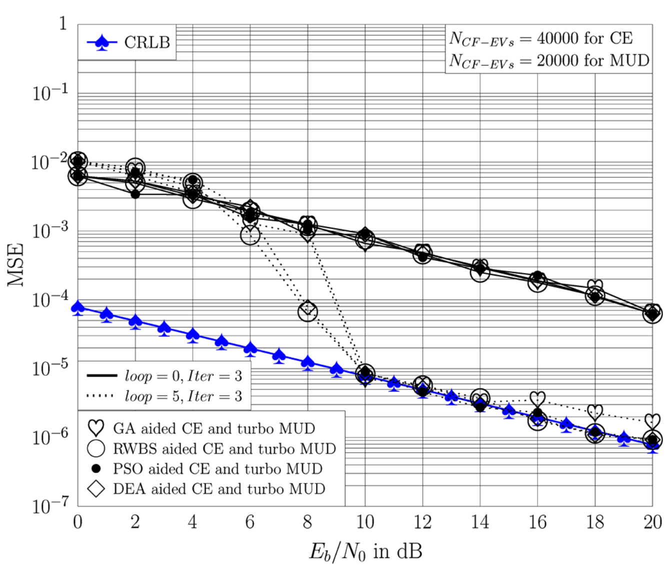


Fig. 12. Comparison of the MSE performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 40 000, and the number of CF evaluations for EA-aided MUD to 20 000.

may be less efficient in comparison to the RWBS- and DEA-aided schemes, and we surmise that $N_{\rm CF-EVs}^{\rm ce} = 20000$ and $N_{\rm CF-EVs}^{\rm mud} = 10000$ may not be sufficient for the PSO- and GA-aided schemes. We then opted for $N_{\rm CF-EVs}^{\rm ce} = 40000$ ($G_{\max} = 400$) and $N_{\rm CF-EVs}^{\rm mud} = 20000$ ($G_{\max} = 200$) and carried out simulations for the four EA-aided joint CE and turbo MUD/decoder schemes again. Figs. 12 and 13 show the achievable MSE and BER performance, respectively, for the four EA-aided joint CE and turbo MUD/decoder schemes. In Fig. 12, it is shown that the MSEs of the four channel estimates associated with the four EA-aided joint CE and turbo MUD/decoder schemes all approached the CRLB with $loop = 5$ outer itera-

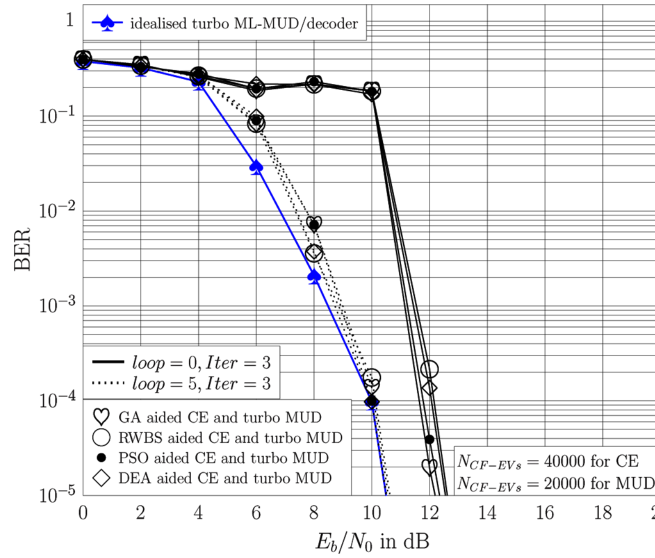


Fig. 13. Comparison of the BER performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 40 000, and the number of CF evaluations for EA-aided MUD to 20 000.

tions for $E_{\rm b}/N_{\rm o} \geq 10$ dB, whereas the BERs of the four EA-aided schemes all approached the optimal BER performance of the idealized turbo ML-MUD/decoder associated with perfect CSI, as shown in Fig. 13.

Our computational complexity comparisons are provided in terms of the three ratios, namely, $C_{\rm MUD}^{\rm EA}/C_{\rm MUD}^{\rm ML}$, $C_{\rm turbo}^{\rm EA}/C_{\rm turbo}^{\rm ML}$, and $C_{\rm joint}^{\rm EA}/C_{\rm turbo}^{\rm ML}$, as shown in Table V. The ratio $C_{\rm MUD}^{\rm EA}/C_{\rm MUD}^{\rm ML}$ characterizes the complexity of an EA-aided MUD in comparison to that of the optimal full-search ML MUD. It can be seen from Table V that all the four EA-aided MUDs impose only 0.1% of the ML MUD's complexity. Given the CSI, the complexity of the RWBS- and DEA-assisted turbo MUD/decoder algorithms is less than 3.5% of the complexity of the turbo ML-MUD/decoder, whereas the complexity of the GA- and PSO-aided turbo MUD/decoder algorithms is less than 6.6% of the turbo ML-MUD/decoder's complexity, as seen in the column $C_{\rm turbo}^{\rm EA}/C_{\rm turbo}^{\rm ML}$ of Table V. An EA-aided joint CE and turbo MUD/decoder involves $I_{\rm ce}$ number of outer iterations between the EA-aided decision-directed channel estimator and the EA-assisted turbo MUD/decoder, and it performs blind joint CE and data detection. Comparing its complexity with that of the idealized turbo ML-MUD/decoder provided with the perfect CSI is really "unfair." Even so, from the column $C_{\rm joint}^{\rm EA}/C_{\rm turbo}^{\rm ML}$ in Table V, we can see that the total complexity of the RWBS- and DEA-assisted joint CE and turbo MUD/decoder schemes is less than 39% of the idealized turbo ML-MUD/decoder's complexity, whereas the GA- and PSO-assisted joint CE and turbo MUD/decoder schemes impose a total complexity that is less than 77% of the idealized turbo ML-MUD/decoder's complexity.

C. Comparing an EA-Aided CE With the Simplified LS CE

In Section II-B, we have pointed out that although the standard LS channel estimator [40] can also provide the optimal

TABLE V
COMPUTATIONAL COMPLEXITY COMPARISON IN TERMS OF THE RATIO OF THE COMPLEXITY OF AN EA-ASSISTED ITERATIVE JOINT CE AND TURBO MUD/DECODER TO THE COMPLEXITY OF THE IDEALIZED TURBO ML-MUD/DECODER ASSOCIATED WITH PERFECT CSI

Scheme	Operation	$C^{EA}_{\mathrm{MUD}}/C^{ML}_{\mathrm{MUD}}$	$C^{EA}_{\mathrm{turbo}}/C^{ML}_{\mathrm{turbo}}$	$C^{EA}_{\mathrm{joint}}/C^{ML}_{\mathrm{turbo}}$
GA assisted joint CE	multiplications	0.10%	5.69%	62.24%
and turbo MUD/decoder	additions	0.10%	7.45%	91.41%
RWBS assisted joint CE	multiplications	0.10%	3.00%	31.27%
and turbo MUD/decoder	additions	0.10%	3.88%	45.86%
PSO assisted joint CE	multiplications	0.10%	5.69%	62.24%
and turbo MUD/decoder	additions	0.10%	7.45%	91.41%
DE assisted joint CE	multiplications	0.10%	3.00%	31.27%
and turbo MUD/decoder	additions	0.10%	3.88%	45.86%

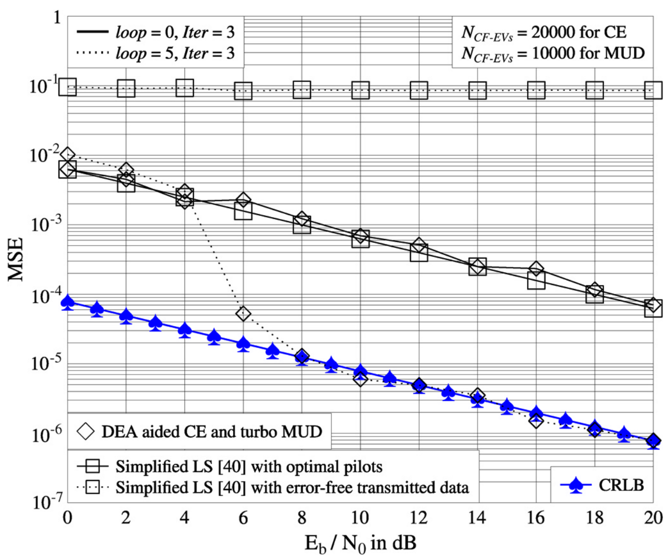


Fig. 14. Comparison of the MSE performance for the DEA-aided joint CE and turbo MUD/decoder scheme with that of the simplified LS channel estimator in [40].

solution for CE optimization (11), it is computationally very expensive. Therefore, it is difficult to combine the standard LS channel estimator with a turbo MUD/decoder to form a joint CE and turbo MUD/decoder scheme, as this approach will impose excessive computational complexity. The simplified LS channel estimator in [40], on the other hand, has low complexity, but it performs poorly even given with the correct error-free transmitted data. We now demonstrate this by investigating the MSE performance of the simplified LS channel estimator using our OFDM/SDMA simulation system. Fig. 14 shows the MSEs attained by the simplified LS CE relying on optimally designed pilots and the true error-free transmitted data, respectively, in comparison with the MSE performance obtained by the DEA-aided joint CE and turbo MUD/decoder recorder at $loop = 0$ and $loop = 5$.

Observe in Fig. 14 that the simplified LS channel estimator, given optimally designed pilots, attains the same MSE as the DEA-aided CE at $loop = 0$. However, this channel estimator performs very poorly even given with the true transmitted data, as shown in Fig. 14. The reason for this poor performance is that this low-complexity channel estimator requires optimal pilots, as discussed in [40, Sec. III], where the relative phases of the training sequences (pilots) for the different users (transmit antennas) must be carefully designed so that each individual CIR (linking the ith transmit antenna to the jth receive antenna) can be separately estimated. However, the users' transmitted data do not meet this requirement of "optimal pilots." Hence, this simplified LS CE cannot benefit from the iterative CE using the detected users' data—it cannot even work adequately using the true users' data. Therefore, the simplified LS channel estimator cannot be combined with a turbo MUD/decoder to form a joint CE and turbo MUD/decoder. By contrast, our proposed EA-aided CE benefits from the iterative joint CE and turbo MUD/decoding process and is capable of approaching the CRLB, as confirmed in Fig. 14.

VI. CONCLUSION

Four EAs, namely, the GA, RWBS, PSO, and DEA, have been applied to the challenging problem of joint semiblind CE and turbo MUD/decoding for ODFM/SDMA communication systems. Extensive results have been provided to demonstrate that by iteratively exchanging information between a continuous-EA-aided decision-directed channel estimator and a discrete-binary EA-assisted turbo MUD/decoder, an EA-aided joint blind CE and turbo MUD/decoder is capable of approaching both the CRLB associated with the optimal channel estimate and the BER of the idealized optimal turbo ML-MUD/decoder associated with perfect CSI, despite imposing only a fraction of the idealized turbo ML-MUD/decoder's complexity.

REFERENCES

[1] M. Jiang and L. Hanzo, "Multiuser MIMO-OFDM for next-generation wireless systems," *Proc. IEEE*, vol. 95, no. 7, pp. 1430–1469, Jul. 2007.
[2] L. Hanzo, Y. Akhtman, L. Wang, and M. Jiang, *MIMO-OFDM for LTE, WIFI and WIMAX: Coherent versus Non-Coherent and Cooperative Turbo-Transceivers.*. Chichester, U.K.: Wiley, 2011.
[3] J. A. C. Bingham, "Multicarrier modulation for data transmission: An idea whose time has come," *IEEE Commun. Mag.*, vol. 28, no. 5, pp. 5–14, May 1990.
[4] L. Hanzo, M. Münster, B. J. Choi, and T. Keller, *OFDM and MC-CDMA for Broadband Multi-User Communications, WLANs, and Broadcasting*. Chichester, U.K.: Wiley, 2003.
[5] P. Vandenameele, L. Van Der Perre, and M. Engels, *Space Division Multiple Access For Wireless Local Area Networks*. Boston, MA, USA: Kluwer, 2001.
[6] S. Chen, L. Hanzo, and A. Livingstone, "MBER space–time decision feedback equalization assisted multiuser detection for multiple antenna aided SDMA systems," *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 3090–3098, Aug. 2006.
[7] P. Vandenameele, L. Van Der Perre, M. Engels, B. Gyselinckx, and H. De Man, "A combined OFDM/SDMA approach," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 11, pp. 2312–2321, Nov. 2000.
[8] J. Zhang, L. Hanzo, and X. Mu, "Joint decision-directed channel and noise-variance estimation for MIMO OFDM/SDMA systems based

on expectation-conditional maximization," *IEEE Trans. Veh. Technol.*, vol. 60, no. 5, pp. 2139–2151, Jun. 2011.

[9] L. Hanzo, O. R. Alamri, M. El-Hajjar, and N. Wu, *Near-Capacity Multi-Functional MIMO Systems: Sphere-Packing, Iterative Detection and Cooperation*. Chichester, U.K.: Wiley, 2009.

[10] S. Thoen, L. Deneire, L. Van Der Perre, M. Engels, and H. De Man, "Constrained least squares detector for OFDM/SDMA-based wireless networks," *IEEE Trans. Wireless Commun.*, vol. 2, no. 1, pp. 129–140, Jan. 2003.

[11] X. Dai, "Pilot-aided OFDM/SDMA channel estimation with unknown timing offset," *Proc. Inst. Elect. Eng.—Commun.*, vol. 153, no. 3, pp. 392–398, Jun. 2006.

[12] J. Ylioinas and M. Juntti, "Iterative joint detection, decoding, and channel estimation in turbo coded MIMO-OFDM," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1784–1796, May 2009.

[13] L. Hanzo, S. Ng, T. Keller, and W. Webb, *Quadrature Amplitude Modulation: From Basics To Adaptive Trellis-Coded Turbo-Equalised and Space–Time Coded OFDM, CDMA and MC-CDMA Systems*. Chichester, U.K.: Wiley, 2004.

[14] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, Apr. 1997.

[15] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006.

[16] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: Univ. of Michigan Press, 1975.

[17] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA, USA: Addison-Wesley, 1989.

[18] S. Chen, X. X. Wang, and C. J. Harris, "Experiments with repeating weighted boosting search for optimization in signal processing applications," *IEEE Trans. Syst., Man, Cybern., B*, vol. 35, no. 4, pp. 682–693, Aug. 2005.

[19] S. F. Page, S. Chen, C. J. Harris, and N. M. White, "Repeated weighted boosting search for discrete or mixed search space and multiple-objective optimisation," *Appl. Soft Comput.*, vol. 12, no. 9, pp. 2740–2755, Sep. 2012.

[20] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Perth, WA, Australia, Nov. 27–Dec., 1, 1995, vol. 4, pp. 1942–1948.

[21] J. Kennedy and R. Eberhart, *Swarm Intelligence*. San Mateo, CA, USA: Morgan Kaufmann, 2001.

[22] K. Price, R. Storn, and J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Berlin, Germany: Springer-Verlag, 2005.

[23] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.

[24] S. Chen and Y. Wu, "Maximum likelihood joint channel and data estimation using genetic algorithms," *IEEE Trans. Signal Process.*, vol. 46, no. 5, pp. 1469–1473, May 1998.

[25] S. Chen and B. L. Luk, "Adaptive simulated annealing for optimization in signal processing applications," *Signal Process.*, vol. 79, no. 1, pp. 117–128, Nov. 1999.

[26] H. Ali, A. Doucet, and D. Amshah, "GSR: A new genetic algorithm for improving source and channel estimates," *IEEE Trans. Circuits Syst. I. Reg. Papers*, vol. 54, no. 5, pp. 1088–1098, May 2007.

[27] K. Yen and L. Hanzo, "Genetic algorithm assisted joint multiuser symbol detection and fading channel estimation for synchronous CDMA systems," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 6, pp. 985–998, Jun. 2001.

[28] K. Yen and L. Hanzo, "Genetic-algorithm-assisted multiuser detection in asynchronous CDMA communications," *IEEE Trans. Veh. Technol.*, vol. 53, no. 5, pp. 1413–1422, Sep. 2004.

[29] X. Wu, T. C. Chuah, B. S. Sharif, and O. R. Hinton, "Adaptive robust detection for CDMA using a genetic algorithm," *Proc. Inst. Elect. Eng.—Commun.*, vol. 150, no. 6, pp. 437–444, Dec. 2003.

[30] K. K. Soo, Y. M. Siu, W. S. Chan, L. Yang, and R. S. Chen, "Particle-swarm-optimization-based multiuser detector for CDMA communications," *IEEE Trans. Veh. Technol.*, vol. 56, no. 5, pp. 3006–3013, Sep. 2007.

[31] M. Y. Alias, S. Chen, and L. Hanzo, "Multiple antenna aided OFDM employing genetic algorithm assisted minimum bit error rate multiuser detection," *IEEE Trans. Veh. Technol.*, vol. 54, no. 5, pp. 1713–1721, Sep. 2005.

[32] M. Jiang, S. X. Ng, and L. Hanzo, "Hybrid iterative multiuser detection for channel coded space division multiple access OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 55, no. 1, pp. 115–127, Jan. 2006.

[33] M. Jiang, J. Akhtman, and L. Hanzo, "Iterative joint channel estimation and multi-user detection for multiple-antenna aided OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 6, no. 8, pp. 2904–2914, Aug. 2007.

[34] J. Zhang, S. Chen, X. Mu, and L. Hanzo, "Joint channel estimation and multi-user detection for SDMA/OFDM based on dual repeated weighted boosting search," *IEEE Trans. Veh. Technol.*, vol. 60, no. 7, pp. 3265–3275, Sep. 2011.

[35] W. Dong, J. Li, and Z. Lu, "Joint frequency offset and channel estimation for MIMO systems based on particle swarm optimization," in *Proc. VTC*, Singapore, May 11–14, 2008, pp. 862–866.

[36] M. Abuthinien, S. Chen, and L. Hanzo, "Semi-blind joint maximum likelihood channel estimation and data detection for MIMO systems," *IEEE Signal Process. Lett.* , vol. 15, pp. 202–205, 2008. **AQ1**

[37] S. Chen, W. Yao, H. Palally, and L. Hanzo, "Particle swarm optimisation aided MIMO transceiver designs," in *Computational Intelligence in Expensive Optimization Problems*, Y. Tenne and C. Goh, Eds. Berlin, Germany: Springer-Verlag, 2010, pp. 487–511.

[38] J. Zhang, S. Chen, X. Mu, and L. Hanzo, "Turbo multi-user detection for OFDM/SDMA systems relying on differential evolution aided iterative channel estimation," *IEEE Trans. Commun.*, vol. 60, no. 6, pp. 1621–1663, Jun. 2012.

[39] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.

[40] Y. Li, "Simplified channel estimation for OFDM systems with multiple transmit antennas," *IEEE Trans. Wireless Commun.*, vol. 1, no. 1, pp. 67–75, Jan. 2002.

[41] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Orlando, FL, USA, Oct. 12–15, 1997, vol. 5, pp. 4104–4108.

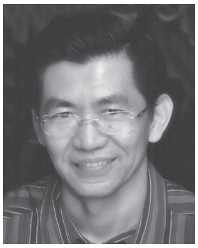
[42] M. A. Khanesar, M. Teshnehlab, and M. A. Shoorehdeli, "A novel binary particle swarm optimization," in *Proc. Mediterranean Conf. Control Autom.*, Athens, Greece, Jul. 27–29, 2007, pp. 1–6.

[43] T. Hanne, "On the convergence of multiobjective evolutionary algorithms," *Eur. J. Oper. Res.*, vol. 117, no. 3, pp. 553–564, Sep. 1999.

[44] X. Yao, "Unpacking and understanding evolutionary algorithms," in *Advances in Computational Intelligence*, J. Liu, C. Alippi, B. Bouchon-Meuniev, G. W. Greenwood, and H. A. Abbass, Eds. Berlin, Germany: Springer-Verlag, 2012, pp. 60–76.

Jiankang Zhang (S'08–M'12) received the B.Sc. degree in mathematics and applied mathematics from Beijing University of Posts and Telecommunications, Beijing, China, in 2006 and the Ph.D. degree in communication and information systems from Zhengzhou University, Zhengzhou, China, in 2012.

Since 2012, he has been a Lecturer with the School of Information Engineering, Zhengzhou University. From September 2009 to December 2011 and from January 2013 to May 2013, he was a Visiting Researcher with the Department of Electronics and Computer Science, University of Southampton, Southampton, U.K. His research interests include wireless communications and signal processing, including channel estimation, multiuser detection, beamforming/precoding, and optimization algorithms.



Sheng Chen (M'90–SM'97–F'08) received the B.Eng. degree in control engineering from the East China Petroleum Institute, Dongying, China, in 1982; the Ph.D. degree in control engineering from City University London, London, U.K., in 1986; and the D.Sc. degree from the University of Southampton, Southampton, U.K., in 2005.

From 1986 to 1999, he held research and academic appointments at the University of Sheffield, Sheffield, U.K.; The University of Edinburgh, Edinburgh, U.K.; and the University of Portsmouth, Portsmouth, U.K. Since 1999, he has been with the Department of Electronics and Computer Science, University of Southampton, where he is currently a Professor of intelligent systems and signal processing. He is also a Distinguished Adjunct Professor with King Abdulaziz University, Jeddah, Saudi Arabia. He has published over 480 research papers. His recent research interests include adaptive signal processing, wireless communications, modeling and identification of nonlinear systems, neural network and machine learning, intelligent control system design, evolutionary computation methods, and optimization.

Dr. Chen is a Chartered Engineer and a Fellow of the Institution of Engineering and Technology. He is an Institute for Scientific Information Highly Cited Researcher in the engineering category (March 2004).

Xiaomin Mu received the B.E. degree from Beijing Institute of Technology, Beijing, China, in 1982.

She is currently a Full Professor with the School of Information Engineering, Zhengzhou University, Zhengzhou, China. She has published many papers in the field of signal processing and coauthored two books. Her research interests include signal processing in communication systems, wireless communications, and cognitive radio.

Lajos Hanzo (M'91–SM'92–F'04) received the M.S. degree in electronics and the Ph.D. degree from the Technical University of Budapest, Budapest, Hungary, in 1976 and 1983, respectively, and the D.Sc. degree from the University of Southampton, Southampton, U.K., in 2004. In 2009, he was awarded the honorary doctorate "Doctor Honoris Causa" by the Technical University of Budapest. AQ2

During his 37-year career in telecommunications, he held various research and academic posts in Hungary, Germany, and the U.K. He was a Chaired Professor with Tsinghua University, Beijing, China. Since 1986, he has been with the Department of Electronics and Computer Science, University of Southampton, where he holds the Chair in Telecommunications. He coauthored 20 John Wiley/IEEE Press books on mobile radio communications totaling in excess of 10 000 pages and published 1356 research entries at IEEE Xplore. He has successfully supervised 83 Ph.D. students. Currently, he is directing a 100-strong academic research team, working on a range of research projects in the field of wireless multimedia communications sponsored by industry, the UK Engineering and Physical Sciences Research Council, the European Research Council, and the Royal Society. He is an enthusiastic supporter of industrial and academic liaison, and he offers a range of industrial courses. He has over 17 000 citations. (For further information on research in progress and associated publications, please refer to http://www-mobile.ecs.soton.ac.uk.)

Dr. Hanzo was the Editor-in-Chief of the IEEE Press from 2008 to 2012. He is also the Governor of the IEEE Vehicular Technology Society. He is a Fellow of the Royal Academy of Engineering, the Institution of Engineering and Technology, and the European Association for Signal Processing. He acted both as Technical Program Committee and General Chair of IEEE conferences and presented keynote lectures. He has been awarded a number of distinctions, including the European Research Council's Senior Research Fellow Grant and the Royal Society's Wolfson Research Merit Award.

AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

AQ1 = Please provide issue number and month of publication in Ref. [36].
AQ2 = Please check if the first paragraph in author L. Hanzo's vitae is captured appropriately.

END OF ALL QUERIES

Evolutionary-Algorithm-Assisted Joint Channel Estimation and Turbo Multiuser Detection/Decoding for OFDM/SDMA

Jiankang Zhang, *Member, IEEE*, Sheng Chen, *Fellow, IEEE*, Xiaomin Mu, and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—The development of evolutionary algorithms (EAs), such as genetic algorithms (GAs), repeated weighted boosting search (RWBS), particle swarm optimization (PSO), and differential evolution algorithms (DEAs), have stimulated wide interests in the communication research community. However, the quantitative performance-versus-complexity comparison of GA, RWBS, PSO, and DEA techniques applied to the joint channel estimation (CE) and turbo multiuser detection (MUD)/decoding in the context of orthogonal frequency-division multiplexing/space-division multiple-access systems is a challenging problem, which has to consider both the CE problem formulated over a continuous search space and the MUD optimization problem defined over a discrete search space. We investigate the capability of the GA, RWBS, PSO, and DEA to achieve optimal solutions at an affordable complexity in this challenging application. Our study demonstrates that the EA-assisted joint CE and turbo MUD/decoder is capable of approaching both the Cramér–Rao lower bound of the optimal CE and the bit error ratio (BER) performance of the idealized optimal maximum-likelihood (ML) turbo MUD/decoder associated with perfect channel state information, respectively, despite imposing only a fraction of the idealized turbo ML-MUD/decoder's complexity.**

***Index Terms*—Differential evolution algorithm (DEA), evolutionary algorithms (EAs), genetic algorithm (GA), joint channel estimation (CE) and turbo multiuser detection (MUD)/decoding, orthogonal frequency-division multiplexing (OFDM), particle swarm optimization (PSO), repeated weighted boosting search (RWBS), space-division multiple access (SDMA).**

I. Introduction

THE BEST possible exploitation of the finite available spectrum in light of the increasing demand for wireless services has been at the center of wireless system optimization. In recent years, multiple antennas have been employed both at the transmitter and/or the receiver, which leads to the concept of multiple-input–multiple-output (MIMO) systems. MIMO systems may be designed for achieving various design goals, such as maximizing the achievable diversity gain, the attainable multiplexing gain, or the number of users supported [1], [2]. Orthogonal frequency-division multiplexing (OFDM) [3], [4] has found its way into numerous recent wireless network standards, owing to its virtues of resilience to frequency-selective fading channels. Both the modulation and demodulation operations of an OFDM system facilitate convenient low-complexity hardware implementations with the aid of the inverse fast Fourier transform (IFFT) and fast Fourier transform (FFT) operations. In an effort to further increase the achievable system capacity, space-division multiple-access (SDMA) communication systems were conceived [5], [6], where several users, roaming in different geographical locations and sharing the same bandwidth and time slots (TSs), are differentiated by their unique user-specific "spatial signature," i.e., by their unique channel impulse responses (CIRs). As one of the most widespread MIMO types, OFDM/SDMA systems [7], [8] exploit the advantages of both OFDM and SDMA.

In the uplink (UL) of an OFDM/SDMA system, the transmitted signals of several single-antenna mobile stations (MSs) are simultaneously received by an array of antennas at the base station (BS). Multiuser detection (MUD) techniques are invoked at the BS for separating the signals of the different MSs, based on their unique user-specific CIRs. A state-of-the-art turbo MUD/decoder exploits the error correction capability of the channel code by exchanging extrinsic information between the MUD and the channel decoder [9]. Naturally, for a turbo MUD/decoder to achieve an optimal or near-optimal performance, the CIRs have to be accurately estimated [1], [4]. Intensive research efforts have been devoted to developing efficient approaches for channel estimation (CE) in multiuser OFDM/SDMA systems [1], [8], [10], [11]. To achieve a near-optimal performance, joint CE and turbo MUD/decoding has recently received significant research attention [12]. Naturally, approaching the performance of the optimal solution, namely, that of the maximum-likelihood (ML) joint CE and turbo MUD/decoding solution, is highly desired. However, in practice, one often has to settle for suboptimal solutions due to the excessive computational complexity of the optimal ML solution, particularly for systems with a high number of

Manuscript received February 13, 2013; revised July 5, 2013; accepted September 17, 2013. This work was supported in part by the National Natural Science Foundation of China under Grant 61172086 and Grant 61271421, by the Research Councils UK under the India–UK Advanced Technology Center, and by the European Research Councils under its Advanced Fellow Grant. The review of this paper was coordinated by Dr. H. Lin.

J. Zhang was with the Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K., where this work was carried out. He is now with the School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China (e-mail: jz09v@ecs.soton.ac.uk).

S. Chen and L. Hanzo are with the Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (e-mail: sqc@ecs.soton.ac.uk; lh@ecs.soton.ac.uk).

X. Mu is with the School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China (e-mail: iexmmu@zzu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2013.2283069

users/antennas and employing high-order quadrature amplitude modulation (QAM) signaling [13]. Fortunately, evolutionary algorithms (EAs) offer potentially viable alternatives for achieving optimal or near-optimal joint CE and turbo MUD/decoding at an affordable complexity.

EAs have found ever-increasing applications in communication and signal processing, where creating globally or near-globally optimal designs at affordable computational costs is critical. The family of the most popular EAs[1] includes genetic algorithms (GAs) [16], [17], repeated weighted boosting search (RWBS) [18], [19], particle swarm optimization (PSO) [20], [21], and differential evolution algorithms (DEAs) [22], [23]. Significant advances have been made in applying these EAs in single-user joint channel and data estimation [18], [24]–[26], in CE and MUD for the multiuser code-division multiple-access UL [27]–[30], in the SDMA-aided OFDM UL [31]–[34], in joint CE and data detection for MIMO systems [35]–[37], and in a diverse range of other applications. However, there is paucity of contributions on EA-aided joint CE and turbo MUD/decoding schemes designed for OFDM/SDMA systems. An exception is our previous work [38], which applies a DEA for supporting the joint CE and turbo MUD/decoding process. Iterative joint CE and turbo MUD/decoding for OFDM/SDMA represents an ideal benchmark application for evaluating various EAs. The ML-MUD optimization is NP-hard, and the joint ML CE and turbo MUD/decoding solution is computationally prohibitive in general. Furthermore, within the iterative CE and turbo MUD/decoding optimization, the CE optimization problem is defined over a continuous search space, whereas the MUD optimization problem is defined over a discrete search space. Thus, both discrete-valued and continuous-valued EAs are required. While individual EAs may have been tested in this challenging iterative joint CE and turbo MUD/decoding optimization, to the best of our knowledge, no performance-versus-complexity comparisons of a group of EA techniques have been presented in the literature in the context of joint CE and turbo MUD/decoding.

Against this background, in this paper, we design and characterize four EAs, namely, the GA, RWBS, PSO, and DEA, under the challenging framework of joint CE and turbo MUD/decoding in OFDM/SDMA systems, in terms of their achievable performance, computational complexity, and convergence characteristics. More specifically, continuous-valued EAs are employed in solving the associated CE optimization, whereas the discrete-binary versions of EAs are employed for finding the ML or near-ML solution for the MUD. In the proposed EA-aided iterative scheme conceived for joint blind CE and turbo MUD/decoding, the EA-aided turbo MUD/decoder feeds back ever more reliable detected data to the EA-based channel estimator. Likewise, a more accurate channel estimate will result in an increased-integrity MUD/decoder. We demonstrate the power and efficiency of this EA-aided iterative CE and turbo MUD/decoder in our extensive simulation study. Our obtained results confirm that the channel estimate and the bit error ratio (BER) performance of our EA-assisted iterative CE and turbo MUD/decoder scheme approach the Cramér–Rao lower bound (CRLB) of the optimal CE [39] and the optimal ML turbo MUD/decoding performance associated with perfect channel state information (CSI), respectively, while only imposing a fraction of the complexity of the idealized turbo ML-MUD/decoder.

The remainder of this paper is organized as follows: The multiuser OFDM/SDMA UL model is described in Section II, which provides the necessary notations and defines the associated optimization problems of the joint CE and turbo MUD/decoding. Section III characterizes the four EAs, i.e., the GA, RWBS, PSO, and DEA, which are used for solving the joint CE and turbo MUD/decoding optimization. Both the continuous-valued EAs invoked for solving the CE optimization and their discrete versions used for solving the ML MUD optimization are detailed in this section. Section IV is devoted to the structure of the proposed EA-aided iterative CE and turbo MUD/decoder as well as to its computational complexity analysis. Our simulation results are presented in Section V, whereas our conclusions are offered in Section VI.

II. Multiuser MIMO OFDM/SDMA System

The multiuser MIMO system considered supports U MSs simultaneously transmitting in the UL to the BS, as shown in Fig. 1. Each user is equipped with a single transmit antenna, whereas the BS employs an array of Q antennas. A time-division multiple-access protocol organizes the available time-domain (TD) resources into TSs. All the U MSs are assigned to every TS, and thus, they are allowed to simultaneously transmit their streams of OFDM-modulated symbols to the SDMA-based BS [4], [7] for the sake of exploiting the available resources. Consequently, the users' signals can only be separated with the aid of their unique CIRs.

A. System Model

For the multiuser OFDM/SDMA UL shown in Fig. 1, all the users simultaneously transmit their data streams, which are denoted by $\mathbf{b}^u$ for $1 \leq u \leq U$. The information bits, i.e., $\mathbf{b}^u$, are first encoded by the user-specific forward error correction (FEC) encoder. The bit stream after the FEC encoder, which is denoted as $\mathbf{b}_C^u$, is passed through an interleaver $\prod$ to yield an output bit stream $\mathbf{b}_I^u$, which is then grouped into blocks of $\log_2 M$ bits as a unit and modulated onto a stream of M-QAM symbols. The modulated data $\tilde{\mathbf{X}}^u$ are serial-to-parallel (S/P) converted, and the pilot symbols are embedded to yield the frequency-domain (FD) OFDM symbol, i.e., $X^u[s,k]$, $1 \leq k \leq K$, where s denotes the OFDM symbol index, and K is the number of subcarriers. The FD pilot symbols and their allocation are known at the receiver and, hence, can be exploited for initial CE. The parallel modulated data are fed to a K-point IFFT-based modulator to generate the TD-modulated signal $x^u[s,k]$. After concatenating the cyclic prefix (CP) of $K_{\rm cp}$ samples, the resultant sequence is transmitted through the MIMO channel and contaminated by the receiver's additive white Gaussian noise (AWGN). The length of the CP must

[1]There are numerous other EAs, for example, the ant colony optimization [14], [15]; however, given our limited space, we concentrate on only four algorithms in this paper.

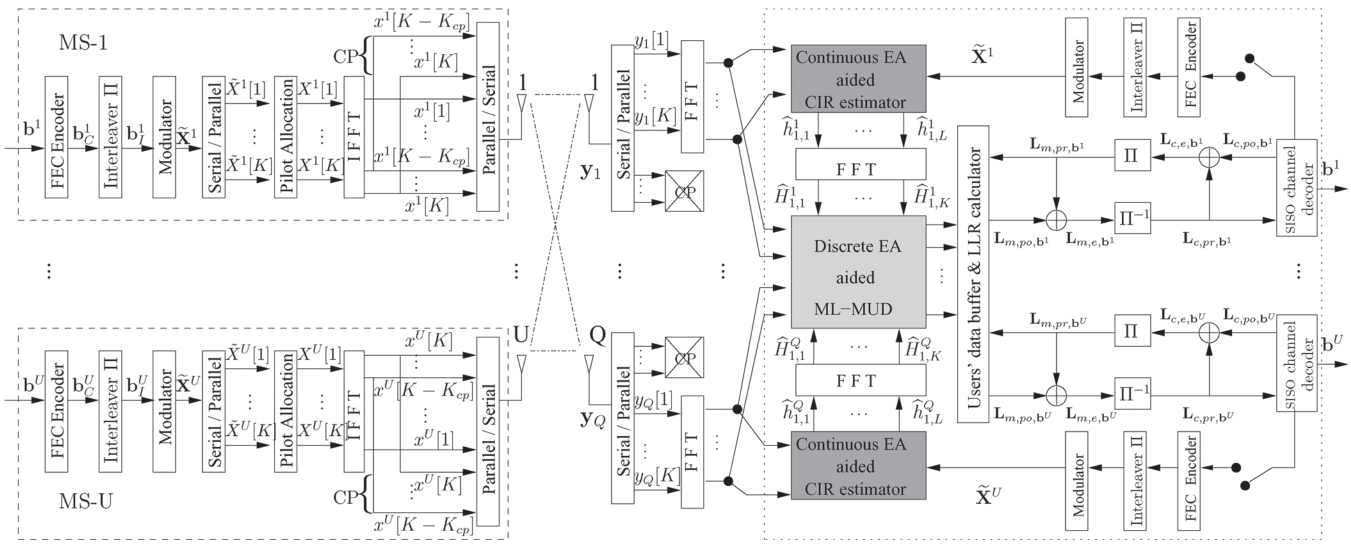


Fig. 1. UL system model for multiuser MIMO OFDM/SDMA. The notation L denotes the log-likelihood ratio. The subscripts m and c of L are associated with the MUD and the channel decoder, respectively, whereas subscripts pr, po, and e are used for representing the *a priori*, *a posteriori*, and extrinsic information, respectively. For notational conciseness, OFDM symbol index s is omitted in $X^u[k]$.

be chosen as $K_{\mathrm{cp}} \geq L_{\mathrm{cir}}$, where L_{cir} denotes the length of the CIRs.

At the BS, the received signals $\mathbf{y}_q$ for $1 \leq q \leq Q$ are parallel-to-serial (P/S) converted, and the CPs are discarded from every OFDM symbol. The resultant signals are fed into the K-point FFT-based receiver. The signal $Y_q[s,k]$ received by the qth receiver antenna element in the kth subcarrier of the sth OFDM symbol can be expressed as [4]

$$Y_q[s,k] = \sum_{u=1}^{U} H_q^u[s,k] X^u[s,k] + W_q[s,k] \tag{1}$$

where $H_q^u[s,k]$ denotes the FD channel transfer function (FD-CHTF) coefficient of the link between the uth user and the qth receiver antenna in the kth subcarrier of the sth OFDM symbol, whereas $W_q[s,k]$ is the associated FD AWGN having the power of $2\sigma_n^2$. Let $\mathbf{h}_q^u[s] \in \mathbb{C}^{L_{\mathrm{cir}} \times 1}$ be the CIR vector of the link between the uth user and the qth receive antenna element during the sth OFDM symbol period, which contains L_{cir} significant CIR coefficients. Then, the FD-CHTF vector $\mathbf{H}_q^u[s] \in \mathbb{C}^{K \times 1}$ is the K-point FFT of $\mathbf{h}_q^u[s]$ defined by

$$\mathbf{H}_q^u[s] = \left[H_q^u[s,1]\ H_q^u[s,2] \cdots H_q^u[s,K]\right]^{\mathrm{T}} = \mathbf{F}\mathbf{h}_q^u[s] \tag{2}$$

where $\mathbf{F} \in \mathbb{C}^{K \times L_{\mathrm{cir}}}$ denotes the FFT matrix [4]. As a benefit of the CP, the OFDM symbols do not overlap, and SDMA processing can be applied on a per-carrier basis.

Arrange the received data at each receive antenna in a column vector $\mathbf{Y}_q[s] \in \mathbb{C}^{K \times 1}$, i.e.,

$$\mathbf{Y}_q[s] = \left[Y_q[s,1]\ Y_q[s,2] \cdots Y_q[s,K]\right]^{\mathrm{T}}, \qquad 1 \leq q \leq Q \tag{3}$$

which hosts the subcarrier-related signals $Y_q[s,k]$, and the transmitted data of each user in a diagonal matrix $\mathbf{X}^u[s] \in \mathbb{C}^{K \times K}$, i.e.,

$$\mathbf{X}^u[s] = \mathrm{diag}\left\{X^u[s,1], X^u[s,2], \ldots, X^u[s,K]\right\} \tag{4}$$

with $X^u[s,k]$ as its diagonal elements, for $1 \leq u \leq U$. Furthermore, let us define the CIR vector $\mathbf{h}_q[s] \in \mathbb{C}^{UL_{\mathrm{cir}} \times 1}$ corresponding to the qth receive antenna during the sth OFDM symbol period as

$$\mathbf{h}_q[s] = \left[\left(\mathbf{h}_q^1[s]\right)^{\mathrm{T}} \left(\mathbf{h}_q^2[s]\right)^{\mathrm{T}} \cdots \left(\mathbf{h}_q^U[s]\right)^{\mathrm{T}}\right]^{\mathrm{T}}, \qquad 1 \leq q \leq Q. \tag{5}$$

The operations of the BS receiver can be summarized as follows: Given the received data $\{\mathbf{Y}_q[s]\}_{q=1}^{Q}$, find the channels $\{\mathbf{h}_q[s]\}_{q=1}^{Q}$ and the transmitted data $\{\mathbf{X}^u[s]\}_{u=1}^{U}$. Ultimately, the receiver is responsible for recovering the users' transmitted information bit streams $\{\mathbf{b}^u\}_{u=1}^{U}$. The turbo MUD/decoder exchanges soft extrinsic information between the soft-in–soft-out (SISO) MUD and the SISO channel decoder [9], which effectively mitigates both the noise and multiuser interference. As a result, it is capable of achieving an accurate recovery of the users' information bit streams. We defer the discussion on the per-carrier-based turbo MUD/decoder [7] in Fig. 1 to Section IV and concentrate on the basic operations of joint CE and MUD at the BS receiver to highlight our motivation for applying EAs to this challenging application.

B. Optimization Problems in Joint CE and MUD

Denote the overall system's CIR vector by $\mathbf{h}[s] \in \mathbb{C}^{UQL_{\mathrm{cir}} \times 1}$ and all the users' transmitted data matrix $\mathbf{X}[s] \in \mathbb{C}^{UK \times K}$, respectively, as

$$\mathbf{h}[s] = \left[\mathbf{h}_1^{\mathrm{T}}[s]\mathbf{h}_2^{\mathrm{T}}[s] \cdots \mathbf{h}_Q^{\mathrm{T}}[s]\right]^{\mathrm{T}} \tag{6}$$

$$\mathbf{X}[s] = \left[\mathbf{X}^1[s]\mathbf{X}^2[s] \cdots \mathbf{X}^U[s]\right]^{\mathrm{T}}. \tag{7}$$

The optimal solution of the joint CE and MUD problem is achieved by maximizing the probability of all the received data $\{\mathbf{Y}_q[s]\}_{q=1}^{Q}$ conditioned on $\mathbf{h}[s]$ and $\mathbf{X}[s]$. Noting that this conditional distribution is Gaussian, this joint optimization is equivalent to the one that minimizes the log-likelihood cost function (CF) formulated as

$$J\left(\mathbf{h}[s], \mathbf{X}[s]\right) = \sum_{q=1}^{Q} \left\| \mathbf{Y}_q[s] - \mathbf{X}^{\mathrm{T}}[s]\overline{\mathbf{F}}\mathbf{h}_q[s] \right\|^2 \tag{8}$$

where the block diagonal matrix $\overline{\mathbf{F}} \in \mathbb{C}^{UK \times UL_{\mathrm{cir}}}$ is given by

$$\overline{\mathbf{F}} = \mathrm{diag}\{\underbrace{\mathbf{F}, \mathbf{F}, \ldots, \mathbf{F}}_{U}\}. \tag{9}$$

Thus, the joint ML CE and MUD solution is defined as

$$\left(\widehat{\mathbf{h}}[s], \widehat{\mathbf{X}}[s]\right) = \arg \min_{\mathbf{h}[s], \mathbf{X}[s]} J\left(\mathbf{h}[s], \mathbf{X}[s]\right). \tag{10}$$

Joint ML optimization (10) is defined in an extremely high-dimensional space with both discrete- and continuous-valued decision variables, and therefore, it is computationally prohibitive. The complexity of this optimization process may be reduced to a more tractable level by invoking an iterative search loop that is carried out first over the continuous space of the legitimate channels $\mathbf{h}[s]$ and then over the discrete set of all the possible transmitted data $\mathbf{X}[s]$. The iterative loop between the CE and the MUD encapsulates two optimization problems. CE optimization can be performed when the data $\mathbf{X}[s]$ are available, either as the known pilot symbols at the start or, more generally, as the detected data fed back from the MUD and FEC-decoder unit. The MUD can be carried out with the estimated CIRs provided by the channel estimator. The iterative procedure exchanging extrinsic information between the decision-directed channel estimator and the MUD based on the estimated CIRs gradually improves both solutions, and typically, only a few iterations are required for approaching the joint ML CE and MUD solution of (10).

1) ML CE: With the detected data $\widehat{\mathbf{X}}[s]$ fed back from the MUD/decoder, the ML CE solution is obtained by minimizing the CF $J_{\mathrm{ce}}(\mathbf{h}[s]) = J(\mathbf{h}[s], \widehat{\mathbf{X}}[s])$. Since the CIRs $\mathbf{h}_q[s]$, $1 \leq q \leq Q$, are only related to the received signals $\mathbf{Y}_q[s]$ recorded at the qth receiver antenna, the ML CE solution $\widehat{\mathbf{h}}[s]$ is given as the solutions of the following Q smaller minimization problems:

$$\widehat{\mathbf{h}}_q[s] = \arg \min_{\mathbf{h}_q[s]} J_{\mathrm{ce}}\left(\mathbf{h}_q[s]\right), \qquad 1 \leq q \leq Q \tag{11}$$

where the CE CF is expressed as

$$J_{\mathrm{ce}}\left(\mathbf{h}_q[s]\right) = \left\| \mathbf{Y}_q[s] - \widehat{\mathbf{X}}^{\mathrm{T}}[s]\overline{\mathbf{F}}\mathbf{h}_q[s] \right\|^2. \tag{12}$$

Since $\mathbf{h}_q[s] \in \mathbb{C}^{UL_{\mathrm{cir}} \times 1}$, the search space for the CE optimization is a continuous-valued $(2UL_{\mathrm{cir}})$-element space. As the detected data contain erroneous decisions, error propagation imposes a serious problem. The OFDM symbol index $[s]$ will be omitted during our forthcoming discourse.

The standard least squares (LS) channel estimator [40] may provide the solutions of (11), which, however, is computationally very expensive as it requires the inverse of the Q very large $(UL_{\mathrm{cir}}) \times (UL_{\mathrm{cir}})$ complex-valued correlation matrices to obtain $\widehat{\mathbf{h}}_q$ for $1 \leq q \leq Q$. A low-complexity simplified LS channel estimator was provided in [40]. However, this simplified LS estimator only works for optimally designed pilots to ensure all the correlation matrices are diagonal. This simplified LS channel estimator performs poorly even given with the correct error-free transmitted data, and clearly, it cannot be applied in decision-directed mode.

2) ML MUD: As a benefit of the CP, the OFDM symbols do not overlap, and receiver processing can be applied on a per-carrier basis [1], [7]. Let us define the received data vector $\mathbf{Y}[s,k] \in \mathbb{C}^{Q \times 1}$ of Q antennas and the transmitted signal vector $\mathbf{X}[s,k] \in \mathbb{C}^{U \times 1}$ of U users in the kth subcarrier of the sth OFDM symbol, respectively, as

$$\mathbf{Y}[s,k] = \left[Y_1[s,k] Y_2[s,k] \cdots Y_Q[s,k]\right]^{\mathrm{T}} \tag{13}$$

$$\mathbf{X}[s,k] = \left[X^1[s,k] X^2[s,k] \cdots X^U[s,k]\right]^{\mathrm{T}}. \tag{14}$$

Furthermore, denote the FD-CHTF matrix linking $\mathbf{X}[s,k]$ to $\mathbf{Y}[s,k]$ as $\mathbf{H}[s,k] \in \mathbb{C}^{Q \times U}$, whose qth row and uth column element is $H_q^u[s,k]$. Given the FD-CHTF matrix estimate $\widehat{\mathbf{H}}[s,k]$, the MUD recovers the transmitted signals $\mathbf{X}[s,k]$ from the received signals $\mathbf{Y}[s,k]$. Since each element $X^u[s,k]$ of $\mathbf{X}[s,k]$ belongs to the finite M-QAM alphabet $\mathcal{S}$ of size $|\mathcal{S}| = M$, there are M^U possible candidate solutions for $\mathbf{X}[s,k]$, and the optimal ML MUD solution is defined as

$$\widehat{\mathbf{X}}[s,k] = \arg \min_{\mathbf{X}[s,k] \in \mathcal{S}^U} J_{\mathrm{mud}}\left(\mathbf{X}[s,k]\right) \tag{15}$$

with the MUD optimization CF expressed as

$$J_{\mathrm{mud}}\left(\mathbf{X}[s,k]\right) = \left\| \mathbf{Y}[s,k] - \widehat{\mathbf{H}}[s,k]\mathbf{X}[s,k] \right\|^2. \tag{16}$$

Optimization (15) is well known to be NP-hard. Since each $X^u[s,k]$ contains $A = \log_2 M$ bits, the bit-stream representation of $X^u[s,k]$ is $\mathbf{b}^u[s,k] = [b_1^u[s,k] b_2^u[s,k] \cdots b_A^u[s,k]]^{\mathrm{T}}$, where each element or bit $b_i^u[s,k] \in \{0,1\}$. Thus, the bit-stream representation of $\mathbf{X}[s,k]$ is

$$\mathbf{b}[s,k] = \left[b_1^1[s,k] \cdots b_A^1[s,k] b_1^2[s,k] \cdots b_A^2[s,k] \cdots b_1^U[s,k] \cdots b_A^U[s,k]\right]^{\mathrm{T}} \tag{17}$$

and the MUD optimization CE is equivalently denoted as $J_{\mathrm{mud}}(\mathbf{b}[s,k]) = J_{\mathrm{mud}}(\mathbf{X}[s,k])$. The OFDM index and the subcarrier index $[s,k]$ will be omitted in the sequel.

Various alternative solutions to the NP-hard ML solution of optimization (15) are available, which trade off performance with complexity. The examples of low-complexity suboptimal solutions include the minimum-mean-square-error MUD, successive-interference-cancelation MUD, and parallel-interference-cancelation MUD. Sphere-detection-based MUD, on the other hand, offers a near-optimal solution with more affordable computational complexity. Moreover, EAs have been demonstrated to be capable of solving this ML optimization problem with complexity that is a fraction of the full-optimal ML complexity [27]–[30], [33]–[38].

III. EAs for Iterative CE and MUD

The continuous versions of the GA, RWBS, PSO, and DEA are adopted to aid in CE optimization, which are denoted as the continuous-GA-assisted CE (CGA-CE), continuous-RWBS-assisted CE (CRWBS-CE), continuous-PSO-assisted CE (CPSO-CE), and continuous-DEA-assisted CE (CDEA-CE). By contrast, the discrete-binary versions of these four EAs are adopted for MUD optimization, which are referred to as the discrete-binary GA-assisted MUD (DBGA-MUD), discrete-binary RWBS-assisted MUD (DBRWBS-MUD), discrete-binary PSO-assisted MUD (DBPSO-MUD), and discrete-binary DEA-assisted MUD (DBDEA-MUD).

A. GA for Iterative CE and MUD

1) CGA-CE: The CGA-CE evolves the population of the P_s candidate solutions over the entire solution space, where P_s is known as the population size. These candidate solutions represent the estimates of the CIR coefficient vector $\mathbf{h}_q$, where the p_sth individual of the population in the gth generation is readily expressed as

$$\widehat{\mathbf{h}}_{q,g,p_s} = \Big[\widehat{h}^1_{q,g,p_s,1} \cdots \widehat{h}^1_{q,g,p_s,L_{\mathrm{cir}}} \widehat{h}^2_{q,g,p_s,1} \cdots \widehat{h}^2_{q,g,p_s,L_{\mathrm{cir}}} \cdots \widehat{h}^U_{q,g,p_s,1} \cdots \widehat{h}^U_{q,g,p_s,L_{\mathrm{cir}}}\Big]^{\mathrm{T}} \tag{18}$$

in which $\widehat{h}^u_{q,g,p_s,l}$ represents an estimate of the lth coefficient in CIR vector $\mathbf{h}^u_q$ for the channel linking user-u to antenna-q. The search space for CE optimization is specified by $(-1-\mathrm{j},+1+\mathrm{j})^{UL_{\mathrm{cir}}}$, with $\mathrm{j}=\sqrt{-1}$. Referring to Fig. 2, we now specify this CGA-CE.

Algorithm 1: CGA-CE.

---

1) **Initialization**. Set the generation index to $g=1$ and randomly generate the initial population, i.e., $\{\widehat{\mathbf{h}}_{q,1,p_s}\}_{p_s=1}^{P_s}$, over the search space $(-1-\mathrm{j},+1+\mathrm{j})^{UL_{\mathrm{cir}}}$.
2) **Selection**. The fitness value of an individual $\widehat{\mathbf{h}}_{q,g,p_s}$ is related to its CF value by $f(\widehat{\mathbf{h}}_{q,g,p_s}) = J^{-1}_{\mathrm{ce}}(\widehat{\mathbf{h}}_{q,g,p_s})$. The roulette wheel selection operator [17] in Fig. 2 is adopted for selecting high-fitness individuals, where the selection ratio of r_s decides how many individuals are to be selected into the mating pool from the total P_s individuals. The value of r_s is defined by $r_s = (N_{\mathrm{pool}}/P_s)$, where N_{pool} is the size of the mating pool.
3) **Crossover**. For each pair of parents randomly chosen from the mating pool, the pair of integers u^* and l^* is randomly generated in the ranges of $\{1,2,\ldots,U\}$ and $\{1,2,\ldots,L_{\mathrm{cir}}\}$, respectively. The parents selected for the crossover operation can be expressed as

$$\begin{cases} \widehat{\mathbf{h}}_{q,g,\mathrm{mum}} = \Big[\widehat{h}^1_{q,g,\mathrm{mum},1} \cdots \widehat{h}^{u^*}_{q,g,\mathrm{mum},l^*-1} \widehat{h}^{u^*}_{q,g,\mathrm{mum},l^*} \\ \qquad \widehat{h}^{u^*}_{q,g,\mathrm{mum},l^*+1} \cdots \widehat{h}^U_{q,g,\mathrm{mum},L_{\mathrm{cir}}}\Big]^{\mathrm{T}} \\ \widehat{\mathbf{h}}_{q,g,\mathrm{dad}} = \Big[\widehat{h}^1_{q,g,\mathrm{dad},1} \cdots \widehat{h}^{u^*}_{q,g,\mathrm{dad},l^*-1} \widehat{h}^{u^*}_{q,g,\mathrm{dad},l^*} \\ \qquad \widehat{h}^{u^*}_{q,g,\mathrm{dad},l^*+1} \cdots \widehat{h}^U_{q,g,\mathrm{dad},L_{\mathrm{cir}}}\Big]^{\mathrm{T}}. \end{cases} \tag{19}$$

As indicated in Fig. 2, the two new offsprings are produced as

$$\begin{cases} \widehat{\mathbf{h}}_{q,g,\mathrm{os}1} = \Big[\widehat{h}^1_{q,g,\mathrm{mum},1} \cdots \widehat{h}^{u^*}_{q,g,\mathrm{mum},l^*-1} \widehat{h}^{u^*}_{q,g,\mathrm{os}1,l^*} \\ \qquad \widehat{h}^{u^*}_{q,g,\mathrm{os}1,l^*+1} \cdots \widehat{h}^U_{q,g,\mathrm{os}1,L_{\mathrm{cir}}}\Big]^{\mathrm{T}} \\ \widehat{\mathbf{h}}_{q,g,\mathrm{os}2} = \Big[\widehat{h}^1_{q,g,\mathrm{dad},1} \cdots \widehat{h}^{u^*}_{q,g,\mathrm{dad},l^*-1} \widehat{h}^{u^*}_{q,g,\mathrm{os}2,l^*} \\ \qquad \widehat{h}^{u^*}_{q,g,\mathrm{os}2,l^*+1} \cdots \widehat{h}^U_{q,g,\mathrm{os}2,L_{\mathrm{cir}}}\Big]^{\mathrm{T}} \end{cases} \tag{20}$$

with

$$\begin{cases} \widehat{h}^{u^*}_{q,g,\mathrm{os}1,l} = \widehat{h}^{u^*}_{q,g,\mathrm{mum},l} - \beta\left(\widehat{h}^{u^*}_{q,g,\mathrm{mum},l} - \widehat{h}^{u^*}_{q,g,\mathrm{dad},l}\right) \\ \widehat{h}^{u^*}_{q,g,\mathrm{os}2,l} = \widehat{h}^{u^*}_{q,g,\mathrm{dad},l} + \beta\left(\widehat{h}^{u^*}_{q,g,\mathrm{mum},l} - \widehat{h}^{u^*}_{q,g,\mathrm{dad},l}\right) \end{cases} \tag{21}$$

for $l^* \le l \le L_{\mathrm{cir}}$, where β is a random value uniformly chosen in the range of $(0,1)$.
4. **Mutation**. As shown in the operation of Step 4) Mutation in Fig. 2, an element or gene $\widehat{h}^u_{q,g,p_s,l}$ of the individual $\widehat{\mathbf{h}}^u_{q,g,p_s}$ is mutated according to

$$\breve{h}^u_{q,g,p_s,l} = \widehat{h}^u_{q,g,p_s,l} + \gamma(\alpha_m + \mathrm{j}\beta_m) \tag{22}$$

where both α_m and β_m are randomly generated in the range $(-1,\ 1)$, whereas γ is a mutation parameter. The number of genes that will mutate is governed by mutation probability M_b.
5. **Termination**. If $g > G_{\max}$, where $G_{\max}$ defines the maximum number of generations, the procedure is curtailed. Otherwise, we set $g=g+1$, and go to 2) **Selection**.

The key algorithmic parameters of this CGA-CE are population size P_s, selection ratio r_s, mutation probability M_b, and mutation parameter γ.

2) DBGA-MUD: A discrete-binary GA has similar basic operations as a continuous GA, which are shown in Fig. 2. This GA evolves a population of the P_s (UA)-element binary-valued candidate vectors, and each individual represents an estimate of the bit sequence $\mathbf{b}$ defined in (17). The p_sth individual of the population in the gth generation is expressed as

$$\widehat{\mathbf{b}}_{g,p_s} = \Big[\widehat{b}^1_{g,p_s,1} \cdots \widehat{b}^1_{g,p_s,A} \widehat{b}^2_{g,p_s,1} \cdots \widehat{b}^2_{g,p_s,A} \cdots \widehat{b}^U_{g,p_s,1} \cdots \widehat{b}^U_{g,p_s,A}\Big]^{\mathrm{T}}. \tag{23}$$

Each binary-valued individual $\widehat{\mathbf{b}}_{g,p_s}$ is related to a signal $\widehat{\mathbf{X}}_{g,p_s}$ transmitted by the M-QAM modulator that represents a candidate solution of MUD optimization (15). The CGA-CE is specified as follows.

Algorithm 2: DBGA-MUD.

---

1) **Initialization**. Set the generation index to $g=1$ and randomly generate the initial population of the P_s binary-valued individuals $\{\widehat{\mathbf{b}}_{1,p_s}\}_{p_s=1}^{P_s}$.
2) **Selection**. The fitness value of an individual $\widehat{\mathbf{b}}_{g,p_s}$ is re lated to its CF value by $f(\widehat{\mathbf{b}}_{g,p_s}) = J^{-1}_{\mathrm{mud}}(\widehat{\mathbf{b}}_{g,p_s})$. The selection ratio r_s specifies the percentage of the P_s indi-

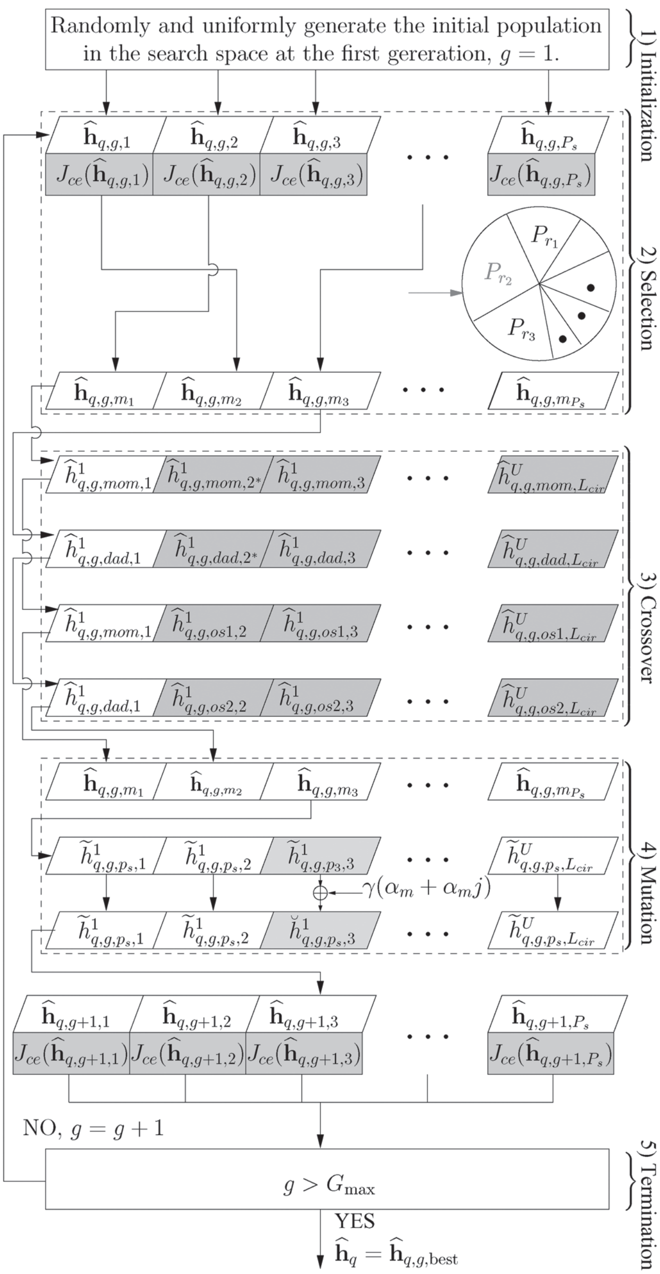


Fig. 2. Flowchart of the continuous-GA-assisted CE.

viduals that are selected to form the mating pool, and we also adopt the roulette wheel selection operator.

3) **Crossover**. We opt for employing the uniform crossover algorithm [17], where a crossover point is randomly selected between the first bit and the last bit of the parent individuals, and the bits are then exchanged between the selected pair of parents.

4) **Mutation**. Given mutation probability M_b, $\lfloor M_b P_s U A \rfloor$ bits are randomly selected from the total number of $(P_s U A)$ bits in the P_s individuals for mutation, where $\lfloor \bullet \rfloor$ denotes the integer floor operator. A bit is mutated by toggling its value from 1 to 0, and vice versa.

5) **Termination**. Optimization is stopped when the predefined maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g + 1$, and go to 2) **Selection**.

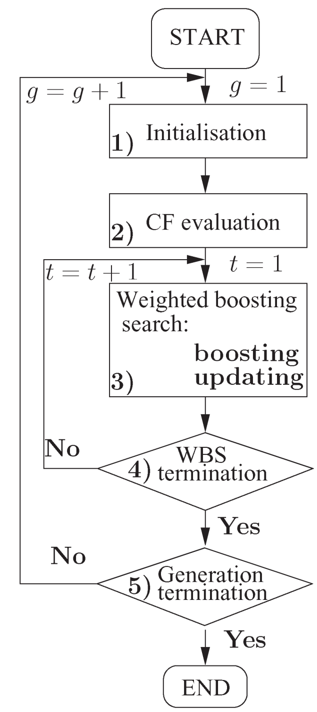


Fig. 3. Flowchart depicting the operations of both the continuous and discrete-binary RWBS algorithms.

The key algorithmic parameters of this DBGA-MUD are population size P_s, selection ratio r_s, and mutation probability M_b.

B. RWBS for Iterative CE and MUD

The operations of the RWBS algorithm [18], [19] are shown in Fig. 3, which consists of the generation-based outer loop and the weighted boosting search (WBS) inner loop.

1) CRWBS-CE: Given an initial estimate $\widehat{\mathbf{h}}_{q,0,\text{best}}$, which can be either randomly generated in the search space $(-1-\text{j}, +1+\text{j})^{UL_{\text{cir}}}$ or chosen as the initial-training-based channel estimate with the aid of the simplified LS channel estimator in [40], the CRWBS-CE is initialized by setting the generation index to $g = 1$ and then following the operations given in Algorithm 3.

Algorithm 3: CRWBS-CE.

---

1) **Generation initialization**. The CIRs $\{\widehat{\mathbf{h}}_{q,g,p_s}\}_{p_s=1}^{P_s}$ are initialized according to: $\widehat{\mathbf{h}}_{q,g,1} = \widehat{\mathbf{h}}_{q,g-1,\text{best}}$

$$\begin{aligned}\widehat{\mathbf{h}}_{q,g,p_s} = \widehat{\mathbf{h}}_{q,g-1,\text{best}} + \gamma\,(&\mathbf{Grv}_{UL_{\text{cir}}}(0,1) \\ &+ \text{j}\mathbf{Grv}_{UL_{\text{cir}}}(0,1)), \qquad 2 \le p_s \le P_s \quad (24)\end{aligned}$$

where $\mathbf{Grv}_{UL_{\text{cir}}}(0,1)$ denotes the (UL_{cir})-element vector, whose elements are drawn from the normal distribution with zero mean and unit variance, $\widehat{\mathbf{h}}_{q,g-1,\text{best}}$ denotes the best individual found in the previous generation, and γ is referred to as the mutation rate.

2) **CF evaluation**. Calculate the CF values associated with the population according to $J_{g,p_s} = J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,p_s})$, $1 \le p_s \le P_s$. Each individual $\widehat{\mathbf{h}}_{q,g,p_s}$ is initially assigned an

equal weight $\delta_{p_s}(0)=(1/P_s)$, where $1\le p_s\le P_s$. Then, set the WBS iteration index to $t=1$.

3) **WBS**. This consists of boosting the weights and updating the population.

•*Stage 1. Boosting*. The relative merits of the individuals are used to adapt the weights for guiding the search. Let us define the best and worst individuals, i.e., $\widehat{\mathbf{h}}_{q,g,p_{\text{best}}}$ and $\widehat{\mathbf{h}}_{q,g,p_{\text{worst}}}$, in the population, where we have $p_{\text{best}}=\arg\min_{1\le p_s\le P_s} J_{g,p_s}$ and $p_{\text{worst}}=\arg\max_{1\le p_s\le P_s} J_{g,p_s}$.

i) Normalize the CF values $\bar{J}_{g,p_s}=J_{g,p_s}/\sum_{j=1}^{P_s} J_{g,j}$, $1\le p_s\le P_s$, and compute weighting factor $\beta(t)$ according to

$$\beta(t)=\frac{\eta(t)}{1-\eta(t)} \text{ with } \eta(t)=\sum_{p_s=1}^{P_s}\delta_{p_s}(t-1)\bar{J}_{g,p_s}. \quad (25)$$

ii) Adapt the weights for $1\le p_s\le P_s$ as follows:

$$\tilde{\delta}_{p_s}(t)=\begin{cases}\delta_{p_s}(t-1)\left(\beta(t)\right)^{\bar{J}_{g,p_s}}, & \beta(t)\le 1\\ \delta_{p_s}(t-1)\left(\beta(t)\right)^{1-\bar{J}_{g,p_s}}, & \beta(t)>1\end{cases} \quad (26)$$

and normalize them as $\delta_{p_s}(t)=\tilde{\delta}_{p_s}(t)/\sum_{j=1}^{P_s}\tilde{\delta}_j(t)$, $1\le p_s\le P_s$.

• *Stage 2. Updating*. This population updating stage consists of

i) Convex combination of $\{\widehat{\mathbf{h}}_{q,g,p_s}\}_{p_s=1}^{P_s}$ constructs a new individual as follows:

$$\widehat{\mathbf{h}}_{q,g,P_s+1}=\sum_{p_s=1}^{P_s}\delta_{p_s}(t)\widehat{\mathbf{h}}_{q,g,p_s}. \quad (27)$$

Intuitively, as the individuals of low CF values have high weights, (27) is capable of producing a new individual, which may have an even lower CF value. A "mirror image" of $\widehat{\mathbf{h}}_{q,g,P_s+1}$ is produced as $\widehat{\mathbf{h}}_{q,g,P_s+2}=\widehat{\mathbf{h}}_{q,g,p_{\text{best}}}+(\widehat{\mathbf{h}}_{q,g,p_{\text{best}}}-\widehat{\mathbf{h}}_{q,g,P_s+1})$.

ii) Compute $J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,P_s+1})$ and $J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,P_s+2})$ and find $p_*=\arg\min_{i=P_s+1,P_s+2} J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,i})$. The new individual $\widehat{\mathbf{h}}_{q,g,p_*}$ then replaces $\widehat{\mathbf{h}}_{q,g,p_{\text{worst}}}$ in the population.

4) **WBS termination**. If $t>T_{\text{wbs}}$, where T_{wbs} defines the maximum number of WBS iterations T_{wbs}, exit the WBS inner loop. Otherwise, set $t=t+1$ and go to 3)**WBS**.

5) **Generation termination**. Stop when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g=g+1$, and go to 1) **Generation initialization**.

The key algorithmic parameters of this CRWBS-CE are the population size P_s, the mutation rate γ and the maximum number of WBS iterations T_{wbs}.

2) DBRWBS-MUD: Given a randomly generated initial binary-valued estimate $\widehat{\mathbf{b}}_{0,\text{best}}$, the DBRWBS-MUD commences by setting the generation index to $g=1$, and it then follows the operations given in Algorithm 4.

Algorithm 4: DBRWBS-MUD.

---

1) **Generation initialization**. Initialize the population $\{\widehat{\mathbf{b}}_{g,p_s}\}_{p_s=1}^{P_s}$ as: set $\widehat{\mathbf{b}}_{g,1}=\widehat{\mathbf{b}}_{g-1,\text{best}}$, while the remaining P_s-1 individuals $\widehat{\mathbf{b}}_{g,p_s}$, $2\le p_s\le P_s$, are generated by randomly muting a certain percentage of the bits in $\widehat{\mathbf{b}}_{g-1,\text{best}}$, the best individual found in the previous generation. The percentage of bits mutated is governed by the mutation probability M_b.

2) **CF evaluation**. The CF values associated with the population are calculated according to $J_{g,p_s}=J_{\text{mud}}(\widehat{\mathbf{b}}_{g,p_s})$, $1\le p_s\le P_s$. Each individual $\widehat{\mathbf{b}}_{g,p_s}$ is initially assigned an equal weight $\delta_{p_s}(0)=(1/P_s)$, where $1\le p_s\le P_s$. Then set the WBS iteration index to $t=1$.

3) **WBS**. Again, this is composed of the weight boosting and population updating stages.

•*Stage 1. Boosting*. The operations are identical to those of i) and ii) in *Stage 1.* of the CRWBS-CE, which yields the set of weights, $\delta_{p_s}(t)$ for $1\le p_s\le P_s$.

•*Stage 2. Updating*. Given the P_s individuals' weights $\delta_{p_s}(t)$ for $1\le p_s\le P_s$, define

$$\begin{cases}\Delta\delta_0(t)=0\\ \Delta\delta_{p_s}(t)=\Delta\delta_{p_s-1}(t)+\delta_{p_s}(t), \quad 1\le p_s\le P_s.\end{cases} \quad (28)$$

Then the four (or a different user-defined number) new individuals $\widehat{\mathbf{b}}_{g,P_s+i}$, $1\le i\le 4$, are generated as follows: for $1\le a\le A$ and $1\le u\le U$,

$$\widehat{b}^u_{g,P_s+i,a}=\widehat{b}^u_{g,p_s,a}, \text{ if } \Delta\delta_{p_s-1}(t) < rand(0,1)\le\Delta\delta_{p_s}(t) \quad (29)$$

where $rand(0,1)$ denotes the random number generator which randomly returns a value from the interval $[0,1)$. The newly generated individuals replace the worst individuals in the population, whose CF values are larger than theirs.

4) **WBS termination**. The WBS iterative procedure is terminated, when the maximum number of WBS iterations T_{wbs} is reached. Otherwise, set $t=t+1$ and go to 3) **WBS**.

5) **Generation termination**. The procedure is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g=g+1$, and go to 1) **Generation initialization**.

The key algorithmic parameters of this DBRWBS-MUD are population size P_s, mutation probability M_b, and the maximum number of WBS iterations T_{wbs}.

C. PSO for Iterative CE and MUD

In a PSO algorithm, individuals of the population are known as particles, and the population is referred to as the swarm. The flowchart of the PSO algorithm adopted is shown in Fig. 4.

1) CPSO-CE: The position of the p_sth particle in the gth generation of the population, i.e., $\widehat{\mathbf{h}}_{q,g,p_s}$, is defined in (18). Associated with each $\widehat{\mathbf{h}}_{q,g,p_s}$, there is a velocity vector $\mathbf{v}_{q,g,p_s}\in(-1-\mathrm{j},+1+\mathrm{j})^{UL_{\text{cir}}}$. Each particle $\widehat{\mathbf{h}}_{q,g,p_s}$ remembers its best position visited so far, denoted by $\widehat{\mathbf{h}}^{\text{ci}}_{q,g,p_s}$, which provides the so-called cognitive information. Every particle also knows the best position visited so far by all particles of the

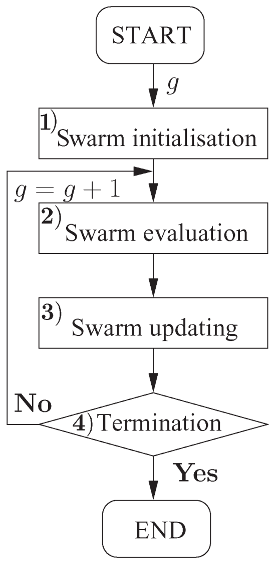


Fig. 4. Flowchart depicting the operations of both the continuous and discrete-binary PSO algorithms.

entire swarm, denoted by $\widehat{\mathbf{h}}^{\rm si}_{q,g}$, which provides the so-called social information. Algorithm 5 details the operations of the CPSO-CE.

Algorithm 5: CPSO-CE.

---

1) **Initialization**. Set the generation index to $g = 1$. Then, randomly generate the initial population, i.e., $\{\widehat{\mathbf{h}}_{q,1,p_s}\}_{p_s=1}^{P_s}$, in the search space $(-1-\mathrm{j}, +1+\mathrm{j})^{UL_{\rm cir}}$, and the associated initial velocities, i.e., $\{\mathbf{v}_{q,1,p_s}\}_{p_s=1}^{P_s}$, in the velocity space $(-1-\mathrm{j}, +1+\mathrm{j})^{UL_{\rm cir}}$.
2) **Swarm evaluation**. For each particle $\widehat{\mathbf{h}}_{q,g,p_s}$, compute its CF value $J_{\rm ce}(\widehat{\mathbf{h}}_{q,g,p_s})$. For $1 \le p_s \le P_s$, update the cognitive information according to the following: If $J_{\rm ce}(\widehat{\mathbf{h}}_{q,g,p_s}) < J_{\rm ce}(\widehat{\mathbf{h}}^{\rm ci}_{q,g-1,p_s})$, set $\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s} = \widehat{\mathbf{h}}_{q,g,p_s}$; otherwise, set $\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s} = \widehat{\mathbf{h}}^{\rm ci}_{q,g-1,p_s}$. Given $p_s^* = \arg\min_{1\le p_s\le P_s} J_{\rm ce}(\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s})$, the swarm's social information is then updated as follows: If $J_{\rm ce}(\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s^*}) < J_{\rm ce}(\widehat{\mathbf{h}}^{\rm si}_{q,g-1})$, set $\widehat{\mathbf{h}}^{\rm si}_{q,g} = \widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s^*}$; otherwise, set $\widehat{\mathbf{h}}^{\rm si}_{q,g} = \widehat{\mathbf{h}}^{\rm si}_{q,g-1}$.
3) **Swarm updating**. The individuals' velocities and positions are updated according to

$$\mathbf{v}_{q,g+1,p_s} = \omega \mathbf{v}_{q,g,p_s} + c_1\, rand(0,1) \left(\widehat{\mathbf{h}}^{\rm ci}_{q,g,p_s} - \widehat{\mathbf{h}}_{q,g,p_s}\right) + c_2\, rand(0,1) \left(\widehat{\mathbf{h}}^{\rm si}_{q,g} - \widehat{\mathbf{h}}_{q,g,p_s}\right) \tag{30}$$

$$\widehat{\mathbf{h}}_{q,g+1,p_s} = \widehat{\mathbf{h}}_{q,g,p_s} + \mathbf{v}_{q,g+1,p_s} \tag{31}$$

for $1 \le p_s \le P_s$, where ω is the inertia weight, whereas c_1 and c_2 are known as the cognitive learning rate and the social learning rate, respectively.
4) **Termination**. Optimization is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g + 1$, and go to 2) **Swarm evaluation**.

The key algorithmic parameters of this CPSO-CE are population size P_s, cognitive learning rate c_1, and social learning rate c_2.

DBPSO-MUD: In the population of the gth generation, the p_sth individual's position, i.e., $\widehat{\mathbf{b}}_{g,p_s}$, is given by (23), and its associated velocity is expressed as

$$\mathbf{v}_{g,p_s} = \left[v^1_{g,p_s,1} \cdots v^1_{g,p_s,A} v^2_{g,p_s,1} \cdots v^2_{g,p_s,A} \cdots v^U_{g,p_s,1} \cdots v^U_{g,p_s,A}\right]^{\rm T}. \tag{32}$$

The velocity space is defined as $(0,1)^{UA}$, i.e., $\mathbf{v}_{g,p_s} \in (0,1)^{UA}$ [41]. Associated with $\widehat{\mathbf{b}}_{g,p_s}$, there are two bit-toggling probability vectors given, respectively, by

$$\mathbf{v}^0_{g,p_s} = \left[v^{1,0}_{g,p_s,1} \cdots v^{1,0}_{g,p_s,A} b^{2,0}_{g,p_s,1} \cdots v^{2,0}_{g,p_s,A} \cdots\ v^{U,0}_{g,p_s,1} \cdots v^{U,0}_{g,p_s,A}\right]^{\rm T} \tag{33}$$

$$\mathbf{v}^1_{g,p_s} = \left[v^{1,1}_{g,p_s,1} \cdots v^{1,1}_{g,p_s,A} b^{2,1}_{g,p_s,1} \cdots v^{2,1}_{g,p_s,A} \cdots\ v^{U,1}_{g,p_s,1} \cdots v^{U,1}_{g,p_s,A}\right]^{\rm T} \tag{34}$$

where $v^{u,0}_{g,p_s,l}$ represents the probability of the bit $\widehat{b}^u_{g,p_s,l}$ being changed to 0, whereas $v^{u,1}_{g,p_s,l}$ represents the probability of the bit $\widehat{b}^u_{g,p_s,l}$ being changed to 1. The cognitive information on the p_sth individual is denoted as $\widehat{\mathbf{b}}^{\rm ci}_{g,p_s}$, and the social information on the swarm is expressed as $\widehat{\mathbf{b}}^{\rm si}_g$. The DBPSO-MUD algorithm is presented as follows.

Algorithm 6: DBPSO-MUD.

---

1) **Initialization**. Set the generation index to $g = 1$. Randomly generate the initial population $\{\widehat{\mathbf{b}}_{1,p_s}\}_{p_s=1}^{P_s}$ and randomly generate the two initial sets of the bit-toggling probability vectors, i.e., $\{\mathbf{v}^0_{1,p_s}\}_{p_s=1}^{P_s}$ and $\{\mathbf{v}^1_{1,p_s}\}_{p_s=1}^{P_s}$, over the probability space $[0,1]^{UA}$.
2) **Swarm evaluation**. For each $\widehat{\mathbf{b}}_{g,p_s}$, compute its CF value $J_{\rm mud}(\widehat{\mathbf{b}}_{g,p_s})$. Then, update the cognitive information $\{\widehat{\mathbf{b}}^{\rm ci}_{g,p_s}\}_{p_s=1}^{P_s}$ and the swarm's social information $\widehat{\mathbf{b}}^{\rm si}_g$.
3) **Swarm updating**. The two sets of the bit-toggling probability vectors are updated according to [42]

$$\mathbf{v}^0_{g+1,p_s} = \omega \mathbf{v}^0_{g,p_s} + c_1\, rand(0,1) \left(\mathbf{1}_{UA} - 2\widehat{\mathbf{b}}^{\rm ci}_{g,p_s}\right) + c_2\, rand(0,1) \left(\mathbf{1}_{UA} - 2\widehat{\mathbf{b}}^{\rm si}_g\right) \tag{35}$$

$$\mathbf{v}^1_{g+1,p_s} = \omega \mathbf{v}^1_{g,p_s} + c_1\, rand(0,1) \left(2\widehat{\mathbf{b}}^{\rm ci}_{g,p_s} - \mathbf{1}_{UA}\right) + c_2\, rand(0,1) \left(2\widehat{\mathbf{b}}^{\rm si}_g - \mathbf{1}_{UA}\right) \tag{36}$$

for $1 \le p_s \le P_s$, where $\mathbf{1}_{UA}$ is the UA-element vector, whose elements are all equal to 1; ω is the inertia weight; and c_1 and c_2 are the cognitive learning rate and the social learning rate, respectively. The velocities associated with $\widehat{\mathbf{b}}_{g,p_s}$, for $1 \le p_s \le P_s$, are calculated as follows. Define the intermediate velocity of the bit $\widehat{b}^u_{g,p_s,l}$, where $1 \le l \le A$ and $1 \le u \le U$, as [42]

$$\tilde{v}^u_{g+1,p_s,l} = \begin{cases} v^{u,1}_{g+1,p_s,l}, & \text{if } \widehat{b}^u_{g,p_s,l} = 0 \\ v^{u,0}_{g+1,p_s,l}, & \text{if } \widehat{b}^u_{g,p_s,l} = 1 \end{cases} \tag{37}$$

which is then used to generate the velocity associated with $\widehat{b}^u_{g,p_s,l}$ according to [41]

$$v^u_{g+1,p_s,l} = \frac{1}{1 + e^{-\tilde{v}^u_{g+1,p_s,l}}}. \tag{38}$$

Next, the individuals are updated as follows:

$$\widehat{b}^u_{g+1,p_s,l} = \begin{cases} \widehat{b}^u_{g,p_s,l}, & \text{if } rand(0,1) \le v^u_{g+1,p_s,l} \\ 1 - \widehat{b}^u_{g,p_s,l}, & \text{if } rand(0,1) > v^u_{g+1,p_s,l} \end{cases} \tag{39}$$

for $1 \le p_s \le P_s$, $1 \le u \le U$, and $1 \le l \le A$.

4) **Termination**. Optimization is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g + 1$, and go to 2) **Swarm evaluation**.

The key algorithmic parameters of this DBPSO-MUD are population size P_s, cognitive learning rate c_1, and social learning rate c_2.

D. DEA for Iterative CE and MUD

1) CDEA-CE: The operations of the CDEA-CE are shown in Fig. 5. More explicitly, the CDEA-CE scheme is elaborated in Algorithm 7.

Algorithm 7: CDEA-CE.

---

1) **Initialization**. Set $g = 1$ and randomly generate the initial $\{\widehat{\mathbf{h}}_{q,g,p_s}\}_{p_s=1}^{P_s}$. The mean of crossover probability C_r is initialized to $\mu_{C_r} = 0.5$, whereas the location parameter of scaling factor λ is initialized to $\mu_\lambda = 0.5$. The archive of the DEA is initialized to be empty.

2) **Population evaluation**. For each $\widehat{\mathbf{h}}_{q,g,p_s}$, where $1 \le p_s \le P_s$, evaluate the CF value $J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,p_s})$. The archive of DEA contains the P_s best solutions that the population has found, and it is updated every generation by adding the $\lfloor P_s \cdot p \rfloor$ parent solutions that are in the top $100{\cdot}p\%$ of high fitness to it, where p is known as the greedy factor. If the archive size exceeds P_s, some solutions are randomly removed from it.

3) **Mutation**. As shown in Step 3) of Fig. 5, the mutation perturbs the candidate solutions by adding randomly selected and appropriately scaled difference-vectors to each base population vector $\widehat{\mathbf{h}}_{q,g,p_s}$ as follows:

$$\widetilde{\mathbf{h}}_{q,g,p_s} = \widehat{\mathbf{h}}_{q,g,p_s} + \lambda_{p_s}(\widehat{\mathbf{h}}^p_{q,g,\text{best},r_1} - \widehat{\mathbf{h}}_{q,g,p_s}) + \lambda_{p_s}(\widehat{\mathbf{h}}_{q,g,r_2} - \widehat{\mathbf{h}}_{q,g,r_3}) \tag{40}$$

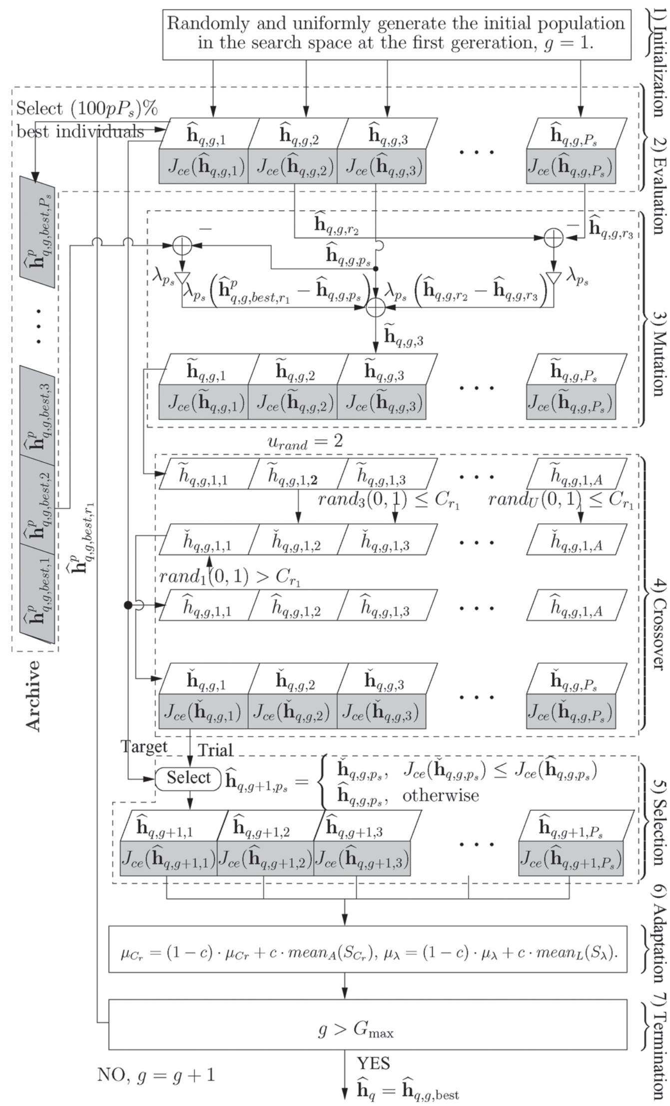


Fig. 5. Flowchart of the continuous-DEA-assisted CE.

where scaling factor $\lambda_{p_s} \in (0,1]$ is a positive number, which is randomly generated for each individual according to the normal distribution having a mean of μ_λ and a standard deviation of 0.1; $\widehat{\mathbf{h}}^p_{q,g,\text{best},r_1}$ is a randomly selected archive value; and r_2 and r_3 are two random integer values fetched from the set $\{1, 2, \ldots, (p_s - 1), (p_s + 1), \ldots, P_s\}$.

4) **Crossover**. A trial vector $\check{\mathbf{h}}_{q,g,p_s}$ is generated upon replacing certain elements of the target vector $\widehat{\mathbf{h}}_{q,g,p_s}$ by the corresponding elements of the related donor vector $\widetilde{\mathbf{h}}_{q,g,p_s}$, which is illustrated in Step 4) of Fig. 5. Specifically, the (u,l)th element of the p_sth trial vector $\check{\mathbf{h}}_{q,g,p_s}$, $\check{h}^u_{q,g,p_s,l}$, is given by

$$\check{h}_{q,g,p_s,l} = \begin{cases} \widetilde{h}^u_{q,g,p_s,l}, & rand(0,1) \le C_{r_{p_s}} \\ \widehat{h}^u_{q,g,p_s,l}, & \text{otherwise} \end{cases} \tag{41}$$

where $C_{r_{p_s}} \in [0,1]$ is the randomly generated crossover probability for each individual according to the Cauchy distribution with location parameter μ_{C_r} and scale parameter 0.1.

5) **Selection**. If $J_{\text{ce}}(\check{\mathbf{h}}_{q,g,p_s}) \leq J_{\text{ce}}(\widehat{\mathbf{h}}_{q,g,p_s})$, the trial vector survives to the next generation and $\widehat{\mathbf{h}}_{q,(g+1),p_s} = \check{\mathbf{h}}_{q,g,p_s}$. Otherwise, the target vector survives and $\widehat{\mathbf{h}}_{q,(g+1),p_s} = \widehat{\mathbf{h}}_{q,g,p_s}$.

6) **Adaptation**. The mean of crossover probability μ_{C_r} and the location parameter of scaling factor μ_λ are updated according to [23]

$$\mu_{C_r} = (1-c)\cdot\mu_{C_r} + c\cdot mean_A(S_{C_r}) \tag{42}$$

$$\mu_\lambda = (1-c)\cdot\mu_\lambda + c\cdot mean_L(S_\lambda) \tag{43}$$

where $c \in (0,1]$ is the adaptive update factor, $mean_A(\cdot)$ and $mean_L(\cdot)$ denote the arithmetic-mean and Lehmer-mean [23] operators, and S_{C_r} and S_λ denote the sets of successful crossover probabilities C_{r_i} and scaling factors λ_i in generation g.

7) **Termination**. The procedure is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g+1$, and go to 2) **Population evaluation**.

The key algorithmic parameters of this CDEA-CE are population size P_s, greedy factor p, and adaptive update factor c.

2) DBDEA-MUD: The DBDEA-MUD is described as follows.

Algorithm 8: DBDEA-MUD.

---

1) **Initialization**. With the generation index set to $g = 1$, randomly generate the initial population $\{\widehat{\mathbf{b}}_{g,p_s}\}_{p_s=1}^{P_s}$. Set $\mu_{C_r} = 0.5$ and $\mu_\lambda = 0.5$.

2) **Population evaluation**. For each $\widehat{\mathbf{b}}_{g,p_s}$, where $1 \leq p_s \leq P_s$, evaluate the CF value $J_{\text{mud}}(\widehat{\mathbf{b}}_{g,p_s}) = J_{\text{mud}}(\widehat{\mathbf{X}}^b_{g,p_s})$, where $\widehat{\mathbf{X}}^b_{g,p_s}$ is the M-QAM symbol vector generated from $\widehat{\mathbf{b}}_{g,p_s}$. The archive, which contains the P_s best solutions that the population has explored, is updated every generation by adding the $\lfloor P_s \cdot p \rfloor$ parent solutions that are in the top $100{\cdot}p\%$ of high fitness to the archive, where again, p is the greedy factor. If the archive size exceeds P_s, some solutions are randomly removed from it.

3) **Mutation**. The mutant version of base vector $\widehat{\mathbf{b}}_{g,i}$ is created according to

$$\widehat{\mathbf{v}}_{g,i} = \widehat{\mathbf{b}}_{g,i} \oplus \left(\mathbf{z}_i^b \otimes \left(\widehat{\mathbf{b}}^p_{g,\text{best},r_1} \oplus \widehat{\mathbf{b}}_{g,i}\right)\right) \oplus \left(\mathbf{z}_i^b \otimes \left(\widehat{\mathbf{b}}_{g,r_2} \oplus \widehat{\mathbf{b}}_{g,r_3}\right)\right) \tag{44}$$

where $\widehat{\mathbf{b}}^p_{g,\text{best},r_1}$ is randomly chosen from the archive, $\widehat{\mathbf{b}}_{g,r_2}$ and $\widehat{\mathbf{b}}_{g,r_3}$ with $r_2 \neq i$ and $r_3 \neq i$ are randomly selected from the current population, $\mathbf{z}_i^b$ is a randomly generated $(U \times A)$-length binary vector known as the bit-scaling factor, $\oplus$ denotes the bitwise exclusive-OR operator, and $\otimes$ denotes the bitwise exclusive-AND operator.

4) **Crossover**. With the uniform crossover, each element of the trial vector has the same probability of inheriting its value from a given vector. Specifically, the (u,j)th element of the p_sth trial vector $\widehat{\mathbf{t}}_{g,p_s}$ at the gth generation, i.e., $\widehat{t}^u_{g,p_s,j}$, is given by

$$\widehat{t}^u_{g,p_s,j} = \begin{cases} \widehat{v}^u_{g,p_s,j}, & rand(0,1) \leq C_{r_{p_s}} \text{ or } j = j_{\text{rand}} \\ \widehat{b}^u_{g,p_s,j}, & \text{otherwise} \end{cases} \tag{45}$$

where crossover probability $C_{r_{p_s}} \in [0,1]$ is randomly generated according to the normal distribution having a mean of μ_{C_r} and a standard deviation of 0.1, whereas j_{rand} is a randomly chosen integer in the range of $\{1, 2, \ldots, P_s\}$.

5) **Selection**. Let $\widehat{\mathbf{X}}^b_{g,p_s}$ and $\widehat{\mathbf{X}}^t_{g,p_s}$ be the M-QAM symbol vectors generated from $\widehat{\mathbf{b}}_{g,p_s}$ and $\widehat{\mathbf{t}}_{g,p_s}$, respectively. If $J_{\text{mud}}(\widehat{\mathbf{X}}^t_{g,p_s}) \leq J_{\text{mud}}(\widehat{\mathbf{X}}^b_{g,p_s})$, then we set $\widehat{\mathbf{b}}_{g+1,p_s} = \widehat{\mathbf{t}}_{g,p_s}$. Otherwise, we set $\widehat{\mathbf{b}}_{g+1,p_s} = \widehat{\mathbf{b}}_{g,p_s}$.

6) **Adaptation**. Given the adaptive update factor $c \in (0,1]$ specified by the designer, μ_{C_r} and μ_λ are adapted according to (42) and (43).

7. **Termination**. Optimization is terminated, when the maximum number of generations $G_{\max}$ is reached. Otherwise, set $g = g+1$, and go to 2) **Population evaluation**.

The key algorithmic parameters of this DBDEA-MUD are population size P_s, greedy factor p, and adaptive update factor c.

IV. EA-Aided Iterative CE and Turbo MUD/Decoder

A. Iterative CE and Turbo MUD/Decoder

The iterative joint CE and turbo MUD/decoder is constituted by the continuous-EA-aided CE and the discrete-binary EA-assisted SISO MUD, followed by U parallel single-user SISO channel decoders, as shown within the dotted-line box at the right-hand side in Fig. 1. The operations of the EA-aided iterative CE and turbo MUD/decoder are outlined as follows.

1) **Initialization**. The training-based channel estimator uses the pilot symbols to provide an initial channel estimate for activating the iterative procedure of joint CE and turbo MUD/decoder. Set the iteration index of the joint CE and turbo MUD/decoder to *loop* $= 1$.

2) **Iterative CE** and **turbo MUD/decoder**.

   1) **Initialization of turbo MUD/decoder**. Forward the channel estimates provided by the "Continuous-EA-aided CIR estimator" block in Fig. 1 to the MUD, and set the iteration index of the turbo MUD/decoder to *Iter* $= 1$.

3) **Turbo MUD/decoder**. The discrete-binary EA-aided ML-MUD, which is shown by the central rectangle in Fig. 1, detects the users' data.

   Step-3.1). The SISO MUD delivers the *a posteriori* information on bit $b^u(i)$ expressed in terms of its log-likelihood ratio (LLR) as [2]

$$\begin{aligned} L_{m,po,b^u(i)} &= \ln\frac{\Pr\left\{\widehat{X}^u \middle| b^u(i)=0\right\}}{\Pr\left\{\widehat{X}^u \middle| b^u(i)=1\right\}} + \ln\frac{\Pr\{b^u(i)=0\}}{\Pr\{b^u(i)=1\}} \\ &= L_{m,e,b^u(i)} + L_{m,pr,b^u(i)} \end{aligned} \tag{46}$$

where $b^u(i)$ is the ith bit in the bit stream that is mapped to the M-QAM symbol stream of user u. The second term in (46), i.e., $L_{m,pr,b^u(i)}$, represents the *a priori* LLR of the interleaved and encoded bits $b^u(i)$, whereas the term $L_{m,e,b^u(i)}$ in (46) is the extrinsic information delivered by the SISO MUD, based on the received signal $\mathbf{Y}$ and the *a priori* information about the encoded bits of all users, except for the ith bit of user u.

Step-3.2). As shown in the receiver in Fig. 1, the extrinsic information output by the SISO MUD is then deinterleaved and fed into the uth user's SISO channel decoder as its *a priori* information, which is denoted as $L_{c,pr,b^u(i)}$. The uth SISO channel decoder then delivers the *a posteriori* information on decoded bits in terms of LLRs $L_{c,po,b^u(i)}$ [9], which can be expressed as $L_{c,po,b^u(i)} = L_{c,e,b^u(i)} + L_{c,pr,b^u(i)}$. The extrinsic information output by the SISO decoder, which is denoted by $L_{c,e,b^u(i)}$, will then be interleaved to provide the *a priori* information for the next iteration of the SISO MUD.

Step-3.3) Turbo MUD/decoder convergence test. If $Iter < I_{\mathrm{tb}}$, where I_{tb} defines the maximum number of turbo iterations,[2] set $Iter = Iter + 1$ and go to **Step-3.1)**. Otherwise, the turbo MUD/decoder has converged, and the detected and decoded bit streams are encoded by the channel encoders, interleaved by the interleavers, and then mapped to the corresponding M-QAM symbol streams, which will be used by the continuous-EA-based CE.

4) **Decision-directed channel estimator**.

Step-4.1) Continuous-EA-aided CE. The "Continuous-EA-aided CIR estimator" blocks in Fig. 1 use the re-encoded and remodulated data $\{\widetilde{\mathbf{X}}^u\}_{u=1}^U$ to perform CIR estimation. The resultant CIR estimate $\widehat{\mathbf{h}}$ is transformed to the FD-CHTF matrix estimate $\widehat{\mathbf{H}}$ by the FFT, which will then be used by the turbo MUD/decoder so that the iterative process can continue.

Step-4.2) CE and **turbo MUD/decoder convergence test.** If $loop < I_{\mathrm{ce}}$, where I_{ce} defines the maximum number of joint CE and turbo MUD/decoder iterations in Fig. 1, set $loop = loop + 1$ and go to **2.1)**. Otherwise, the iterative CE and turbo MUD/decoder has converged.

The *a posteriori* information on the turbo ML-MUD associated with bit $b^u(i)$ is given by [2]

$$L^{\mathrm{ML}}_{m,po,b^u(i)} = \ln \frac{\Pr\{\mathbf{Y}, b^u(i)=0\}}{\Pr\{\mathbf{Y}, b^u(i)=1\}}$$

$$= \ln \frac{\sum\limits_{\forall \mathbf{X}\in\mathcal{S}^U : b^u(i)=0} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\mathbf{X}\|^2}{2\sigma_n^2}} \prod\limits_{u=1}^{U}\prod\limits_{j=1}^{A} \Pr\{b^u(j)\}}{\sum\limits_{\forall \mathbf{X}\in\mathcal{S}^U : b^u(i)=1} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\mathbf{X}\|^2}{2\sigma_n^2}} \prod\limits_{u=1}^{U}\prod\limits_{j=1}^{A} \Pr\{b^u(j)\}} \tag{47}$$

where the probability $\Pr\{b^u(j)\}$ of $b^u(j)$ is given by

$$\Pr\{b^u(j)\} = \frac{1}{2}\left(1 + \mathrm{sgn}\left(\frac{1}{2} - b^u(j)\right)\tanh\left(\frac{L^{\mathrm{ML}}_{m,pr,b^u(j)}}{2}\right)\right). \tag{48}$$

Note from (47) that the $M^U = |\mathcal{S}|^U$ legitimate candidate solutions of the U users are partitioned into the two subsets conditioned on $b^u(i) = 0$ and $b^u(i) = 1$, respectively, and the complexity of calculating $L^{\mathrm{ML}}_{m,po,b^u(i)}$ exponentially increases with the size of M-QAM signaling and the number of users U.

By contrast, the discrete-binary EA-aided turbo MUD is capable of reducing the complexity of the *a posteriori* information calculation to that of a near-single-user scenario, once the transmitted data $\mathbf{X}$ are detected by the discrete-binary EA-aided MUD. Specifically, the *a posteriori* information on the discrete-binary EA-aided turbo MUD associated with bit $b^u(i)$ is given as

$$L^{\mathrm{EA}}_{m,po,b^u(i)} = \ln \frac{\sum\limits_{\forall X^u\in\mathcal{S}: b^u(i)=0} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\tilde{\mathbf{X}}\|^2}{2\sigma_n^2}} \prod\limits_{j=1}^{A} \Pr\{b^u(j)\}}{\sum\limits_{\forall X^u\in\mathcal{S}: b^u(i)=1} e^{-\frac{\|\mathbf{Y}-\mathbf{H}\tilde{\mathbf{X}}\|^2}{2\sigma_n^2}} \prod\limits_{j=1}^{A} \Pr\{b^u(j)\}} \tag{49}$$

where $\Pr\{b^u(j)\}$ is also calculated using (48) by replacing $L^{\mathrm{ML}}_{m,po,b^u(i)}$ with $L^{\mathrm{EA}}_{m,po,b^u(i)}$, and $\widetilde{\mathbf{X}} = [\widehat{X}^1 \cdots \widehat{X}^{u-1} X^u \widehat{X}^{u+1} \cdots \widehat{X}^U]^{\mathrm{T}}$, with X^u assuming values from the M-QAM symbol set $\mathcal{S}$ and $\widehat{X}^v, v = 1, \ldots, u-1, u+1, \ldots, U$ being acquired by the discrete-binary EA-aided MUD at the first turbo iteration. Following the first turbo iteration, $\widehat{X}^v$ for $v \neq u$ is given by

$$\hat{X}^v = \max_{X^v\in\mathcal{S}} \Pr\{X^v\} = \max_{X^v\in\mathcal{S}} \prod_{j=1}^{A} \Pr\left[b^v(j)\right]. \tag{50}$$

Observe in (49) that the number of legitimate candidate solutions is $M = |\mathcal{S}|$ for each user, since the transmitted signal of user v ($v \neq u$) is given by (50). Thus, the computational complexity of the *a posteriori* information's calculation has been reduced to $M \cdot U$.

B. Convergence Discussion and Complexity Analysis

To characterize the convergence behavior of the population $\{\widehat{\mathbf{X}}_{g,p_s}\}_{p_s=1}^{P_s}$, as generation g evolves,[3] we may adopt the

[2]A turbo iteration represents one exchange of extrinsic information between the discrete-binary EA-assisted SISO MUD and the SISO channel decoder, as described in **Step 3.1)** and **Step 3.2)** and shown in Fig. 1.

[3]Although the discussion only refers to the discrete-binary EA-assisted MUD, it also makes sense for the continuous-EA-aided CE.

probability of convergence, which is defined as [43]

$$\lim_{g\to+\infty} \Pr\left\{\left\|\hat{\mathbf{X}}_{g,p_s} - \mathbf{X}_{\mathrm{ML}}\right\| > \epsilon\right\} = 0, \quad \forall p_s \tag{51}$$

where $\mathbf{X}_{\mathrm{ML}}$ denotes the optimal ML solution, and ϵ is an arbitrary positive value. The probability of convergence defined in (51) requires that the solutions are located outside the ϵ-neighborhood of $\mathbf{X}_{\mathrm{ML}}$ with a probability of zero, as the population evolves. Generally, there exists a probability $p(g) > 0$ at each generation g that the individuals in the parental population will generate an offspring belonging to the ϵ-neighborhood of $\mathbf{X}_{\mathrm{ML}}$. As a benefit of the elitism, the individuals of the next generation are as good as or better than their counterparts in the current generation, which indicates that sequence $\{p(g)\}$ is monotonically increasing. This leads to [43]

$$\lim_{g\to+\infty} \Pr\left\{\left\|\hat{\mathbf{X}}_{g,p_s} - \mathbf{X}_{\mathrm{ML}}\right\| < \epsilon\right\} = 1, \quad \forall p_s. \tag{52}$$

The given proposition indicates that the population will converge to the ϵ-neighborhood of $\mathbf{X}_{\mathrm{ML}}$ with a probability of 1, but does not address the vital question of convergence speed. As we use an EA to solve an NP-hard optimization problem, whose optimal solution by the "brute force" exhaustive ML search imposes an exponentially increasing complexity in the problem size. Vast amounts of empirical results found in the literature have demonstrated that appropriately tuned EAs are capable of approaching the globally optimal solutions even for the most challenging optimization problems at affordable complexity. Moreover, the theoretical analysis of EAs has made significant progress in the past few years [44]. Specifically, many NP-hard problems can be turned into the so-called EA-easy class [44], implying that they can be solved by a well-tuned EA algorithm at complexity at most polynomial in the problem size.

Given the CSI, i.e., $\mathbf{h}$, the computational complexity of a turbo MUD/decoder is given by

$$C_{\mathrm{turbo}} = I_{\mathrm{tb}} \cdot C_{\mathrm{MUD}} + I_{\mathrm{tb}} \cdot C_{\mathrm{dec}} \tag{53}$$

where C_{MUD} and C_{dec} are the complexity of the turbo MUD and that of the channel decoder, respectively. The second term in (53) remains the same for both the turbo ML-MUD/decoder and the turbo EA-aided MUD/decoder. Furthermore, the second term in (53) is significantly smaller than the first term. The complexity $C_{\mathrm{MUD}}^{\mathrm{ML}}$ of the turbo ML-MUD/decoder imposed by detecting a frame of S OFDM symbols, each having K subcarriers, can be shown to be (54), shown at the bottom of the page, whereas the complexity $C_{\mathrm{MUD}}^{\mathrm{EA}}$ of the turbo EA-aided MUD/decoder can be shown to be (55), shown at the bottom of the page.

The total complexity of the EA-assisted joint CE and turbo MUD/decoder is given by

$$C_{\mathrm{joint}}^{\mathrm{EA}} = I_{\mathrm{ce}} \cdot (C_{\mathrm{turbo}}^{\mathrm{EA}} + C_{\mathrm{one-mud}}^{\mathrm{EA}} + C_{\mathrm{ce}}^{\mathrm{EA}}). \tag{56}$$

In (56), $C_{\mathrm{ce}}^{\mathrm{EA}}$ denotes the complexity of the continuous-EA-based CE, which is specified by the number $N_{\mathrm{CF-EVs}}^{\mathrm{ce}}$ of $J_{\mathrm{ce}}(\bullet)$ CF evaluations and the complexity per CF evaluation. Given the population size P_s^{ce} and the maximum number of generations $G_{\max}^{\mathrm{ce}}$, we have $N_{\mathrm{CF-EVs}}^{\mathrm{ce}} \approx P_s^{\mathrm{ce}} \cdot G_{\max}^{\mathrm{ce}}$ for all the four continuous-EA-based CEs,[4] whereas the complexity per $J_{\mathrm{ce}}(\bullet)$ CF evaluation may be derived according to (12) as

$$\begin{cases} 4KS(UL_{\mathrm{cir}} + U + 1) & \text{multiplications} \\ KS(5UL_{\mathrm{cir}} + 3U + 3) & \text{additions.} \end{cases} \tag{57}$$

The term $C_{\mathrm{one-mud}}^{\mathrm{EA}}$ represents the complexity imposed by the discrete-binary EA-aided MUD at each outer iteration loop, which is specified by the number of $J_{\mathrm{mud}}(\bullet)$ CF evaluations $N_{\mathrm{CF-EVs}}^{\mathrm{mud}} \approx P_s^{\mathrm{mud}} \cdot G_{\max}^{\mathrm{mud}}$ for all the four discrete-binary EA-aided MUDs,[5] where P_s^{mud} is the population size, and $G_{\max}^{\mathrm{mud}}$ is the maximum number of generations, as well as the complexity per $J_{\mathrm{mud}}(\bullet)$ CF evaluation, which can be determined according to (16) as

$$\begin{cases} 4KSQU & \text{multiplications} \\ KS(3QU + Q + U - 1) & \text{additions.} \end{cases} \tag{58}$$

The ratio of the complexity of the EA-assisted joint CE and turbo MUD/decoder to that of the idealized turbo

[4]For the CRWBS-CE, $N_{\mathrm{CF-EVs}}^{\mathrm{ce}} = ((P_s^{\mathrm{ce}} - 1) + 2T_{\mathrm{wbs}}) \cdot G_{\max}^{\mathrm{ce}}$. The approximation is met by appropriately choosing T_{wbs}.

[5]Again, the approximation holds for the DBRWBS-MUD by appropriately choosing the number of WBS iterations.

$$\begin{cases} KS\left(2UM^U(2Q\log_2 M + 2Q + \log_2 M) + U\log_2 M \right. \\ \qquad \left. + MU(4\log_2 M - 1)\right) & \text{multiplications} \\ KS\left(M^U(4QU\log_2 M + 4QU - 2U\log_2 M - Q) \right. \\ \qquad \left. + 2U(M-1)\log_2 M\right) & \text{additions} \end{cases} \tag{54}$$

$$\begin{cases} KS\,(MU(4QU(\log_2 M + 1) + 2U\log_2 M \\ \qquad +4\log_2 M - 1) + U\log_2 M) & \text{multiplications} \\ KS\big(MU(4QU(\log_2 M + 1) - 2U\log_2 M - Q) \\ \qquad +2\log_2 M)2U\log_2 M\big) & \text{additions} \end{cases} \tag{55}$$

TABLE I
SIMULATION PARAMETERS OF THE MULTIUSER OFDM/SDMA SYSTEM

Encoder	Type	RSC
	Code rate	1/2
	Constraint length	3
	Polynomial	$(g_0, g_1) = (7, 5)$
Channel	Number of paths L_{cir}	4
	Delays	$\{0, 1, 2, 3\}$
	Average path gains	$\{0, -5, -10, -15\}$ (dB)
	Taps: frame to frame	Complex white Gaussian
	Taps: within frame	fading rate $F_D = 10^{-7}$
System	MSs U	4
	Receiver antennas Q	3
	Modulation	16-QAM
	Subcarriers K	64
	Cyclic prefix K_{cp}	16

ML-MUD/decoder associated with perfect CSI is expressed by

$$\frac{C_{\text{joint}}^{\text{EA}}}{C_{\text{turbo}}^{\text{ML}}} = \frac{I_{\text{ce}} \cdot \left(C_{\text{turbo}}^{\text{EA}} + C_{\text{one-mud}}^{\text{EA}} + C_{\text{ce}}^{\text{EA}}\right)}{I_{\text{tb}} \cdot \left(C_{\text{MUD}}^{\text{ML}} + C_{\text{dec}}\right)} \approx \frac{I_{\text{ce}} \cdot \left(I_{\text{tb}} \cdot C_{\text{MUD}}^{\text{EA}} + C_{\text{one-mud}}^{\text{EA}} + C_{\text{ce}}^{\text{EA}}\right)}{I_{\text{tb}} \cdot C_{\text{MUD}}^{\text{ML}}} \tag{59}$$

where the approximation is obtained by omitting the second term in (53).

V. EXPERIMENTAL PERFORMANCE RESULTS

The parameters of our simulated multiuser SDMA/OFDM UL are listed in Table I. A four-path Rayleigh fading channel model was employed for each link, and the delays of the paths were normalized to the sample duration. At the beginning of every frame, which contained $S = 100$ OFDM symbols, a new channel tap was generated for each of the four paths according to the complex-valued white Gaussian process with its power specified by the corresponding average path gain. Within the frame, each channel tap experienced independent Rayleigh fading having the same normalized Doppler frequency of $F_D = 10^{-7}$. A half-rate recursive systematic convolutional code was employed as the channel code. The default values of the EAs' algorithmic parameters are listed in Table II. The first OFDM symbol of each frame was populated with pilots for the initial-training-based CE, yielding a training overhead of 1%. The system's signal-to-noise ratio (SNR) was specified by SNR $= E_{\text{b}}/N_{\text{o}}$ in decibels, where E_{b} denotes the energy per bit, and N_{o} is the power spectral density of the channel AWGN.

A. *Efficiency, Reliability, and Convergence Investigation*

We first quantified the efficiency and reliability of the continuous-EA-aided CEs and the discrete-binary EA-based MUD schemes separately over $N_{\text{tot}} = 1000$ independent simulation runs. Perfect CSI was assumed for evaluating the discrete-binary EA-assisted MUD schemes, while the transmitted data were available, when evaluating the continuous-EA-aided CE schemes. There was no information exchange between the MUD and the decoder, i.e., we had $I_{\text{tb}} = 1$, and the channel's AWGN had $N_{\text{o}} = 0$. For an EA-aided CE scheme, we declared a "successful" run when the algorithm achieved the CF value of $J_{\text{ce}}(\widehat{\mathbf{h}}_{q,G_{\max}^i,\text{best}}) < 10^{-4}$ within the set upper limit for the number of CF evaluations $\overline{N}_{\text{CF-EVs}}^{\text{lim}} = P_s \cdot G_{\max}^{\text{lim}} = 100 \times 1000$, where $G_{\max}^i$ denotes the number of generations in the ith simulation run. Otherwise, the run was declared as "failed." Over the $N_{\text{tot}} = 1000$ simulation runs, we collected the statistics of the number of successful runs, denoted as N_{suc}; the number of failed runs, denoted as N_{fail}; the total number of CF evaluations in the N_{suc} successful runs, defined by $N_{\text{CF-EVs}}^{\text{suc}}$; and the total number of CF evaluations in the N_{fail} failed runs, defined by $N_{\text{CF-EVs}}^{\text{fail}}$, using the following.

for run $= 1 : N_{\text{tot}}$
  if $(G_{\max}^{\text{run}} \leq G_{\max}^{\text{lim}})$ and $(J_{\text{ce}}(\widehat{\mathbf{h}}_{q,G_{\max}^{\text{run}},\text{best}}) < 10^{-4})$
    $N_{\text{suc}} = N_{\text{suc}} + 1$; $N_{\text{CF-EVs}}^{\text{suc}} = N_{\text{CF-EVs}}^{\text{suc}} + P_s \cdot G_{\max}^{\text{run}}$,
  else
    $N_{\text{fail}} = N_{\text{fail}} + 1$; $N_{\text{CF-EVs}}^{\text{fail}} = N_{\text{CF-EVs}}^{\text{fail}} + P_s \cdot G_{\max}^{\text{lim}}$.

After obtaining these statistics, the average number of CF evaluations per run was given by

$$\overline{N}_{\text{CF-EV}_s}^{\text{tot}} = \left(N_{\text{CF-EV}_s}^{\text{suc}} + N_{\text{CF-EV}_s}^{\text{fail}}\right) / N_{\text{tot}} \tag{60}$$

while the average number of CF evaluations per successful run was defined by

$$\overline{N}_{\text{CF-EV}_s}^{\text{suc}} = N_{\text{CF-EV}_s}^{\text{suc}} / N_{\text{suc}}. \tag{61}$$

Then, the normalized average number of CF evaluations per run was formulated as

$$\overline{R}_{\text{CF-EV}_s}^{\text{tot}} = \overline{N}_{\text{CF-EV}_s}^{\text{tot}} / \overline{N}_{\text{CF-EV}_s}^{\text{lim}} \tag{62}$$

and the normalized average number of CF evaluations per successful run was defined as

$$\overline{R}_{\text{CF-EV}_s}^{\text{suc}} = \overline{N}_{\text{CF-EV}_s}^{\text{suc}} / \overline{N}_{\text{CF-EV}_s}^{\text{lim}} \tag{63}$$

offered the metrics for quantifying the efficiency of the EA-aided CE scheme investigated. The smaller $\overline{R}_{\text{CF-EVs}}^{\text{tot}}$ or $\overline{R}_{\text{CF-EVs}}^{\text{suc}}$, the more efficient the EA-aided CE scheme. On the other hand, the reliability of the EA-aided CE was measured by the failure ratio, i.e.,

$$R_{\text{fail}} = N_{\text{fail}} / N_{\text{tot}}. \tag{64}$$

The lower R_{fail}, the more reliable the EA-aided CE scheme. The efficiency and reliability of the four continuous-EA-assisted CE schemes are shown in Fig. 6, where it can be seen that the CDEA-CE outperformed the other three schemes, and the former always arrived at the target CF value within the average computational complexity of 15 000 CF evaluations. The CRWBS-CE came a close second, and it always attained the target CF value within the average complexity of 22 000 CF evaluations. The CGA-CE was the the worst CE candidate, having the failure rate of $R_{\text{fail}} \approx 7\%$ and imposing an average computational complexity of 90 000 CF evaluations.

A similar procedure was carried out for investigating the efficiency and reliability of the four discrete-binary EA-assisted MUDs by setting $G_{\max}^{\text{lim}} = 500$ and $\overline{N}_{\text{CF-EVs}}^{\text{lim}} = M^U = 16^4$. A successful detection run was confirmed, if $(G_{\max}^{\text{run}} \leq G_{\max}^{\text{lim}})$ and

TABLE II
ALGORITHMIC PARAMETERS FOR THE EA-ASSISTED CE AND MUD

Scheme	Parameter	Value	Scheme	Parameter	Value
CGA-CE	Population size P_s	100	DBGA-MUD	Population size P_s	100
	Selection ratio r_s	0.5		Selection ratio r_s	0.5
	Mutation parameter γ	0.01		Mutation probability M_b	0.15
	Mutation probability M_b	0.2			
CRWBS-CE	Population size P_s	100	DBRWBS-MUD	Population size P_s	100
	Mutation parameter γ	0.001		Mutation probability M_b	0.5
	Weighted boosting search T_{wbs}	40		Weighted boosting search T_{wbs}	40
CPSO-C	Population size P_s	100	DBPSO-MUD	Population size P_s	100
	Cognition learning rate c_1	2		Cognition learning rate c_1	0.1
	Social learning rate c_2	2		Social learning rate c_2	0.3
CDEA-CE	Population size P_s	100	DBDEA-MUD	Population size P_s	100
	Greedy factor p	0.1		Greedy factor p	0.7
	Adaptive update factor c	0.1		Adaptive update factor c	0.8

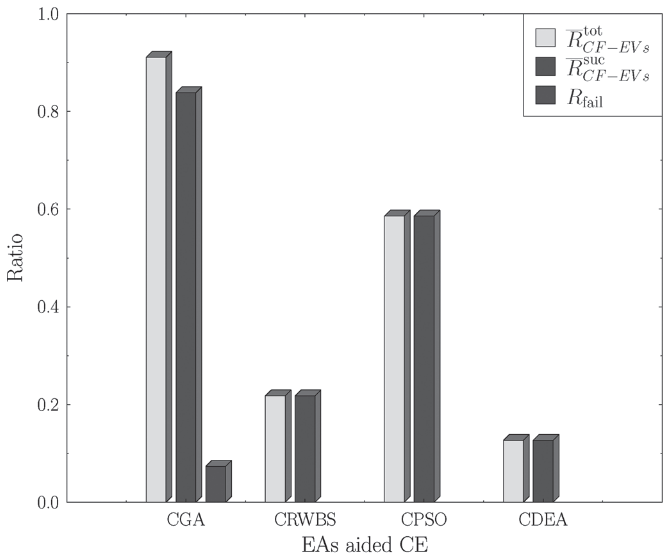


Fig. 6. Histograms of the efficiency and reliability measures, in terms of $\overline{R}^{\text{tot}}_{\text{CF}-\text{EVs}}$, $\overline{R}^{\text{suc}}_{\text{CF}-\text{EVs}}$, and R_{fail}, for the four continuous-EA-assisted CE schemes.

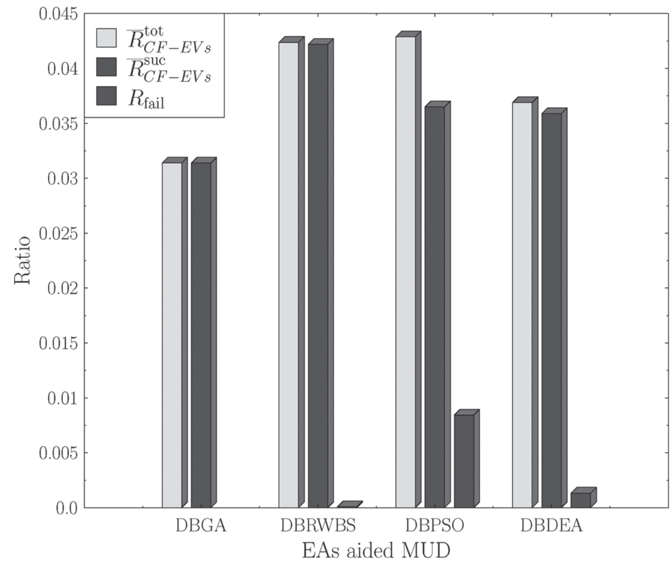


Fig. 7. Histograms of the efficiency and reliability measures, in terms of $\overline{R}^{\text{tot}}_{\text{CF}-\text{EVs}}$, $\overline{R}^{\text{suc}}_{\text{CF}-\text{EVs}}$, and R_{fail}, for the four discrete-binary EA-assisted MUDs.

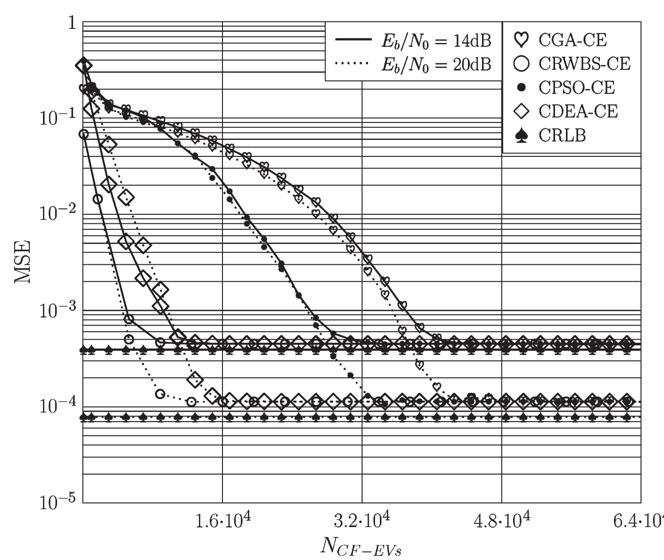


Fig. 8. MSE versus the number of CF evaluations, which characterizes the convergence performance of the different continuous-EA-assisted CE schemes.

the BER of the best individual $\widehat{\mathbf{X}}_{G^{\text{run}}_{\max},\text{best}}$ was infinitesimally low. Otherwise, the run was declared a failure. Note that $\overline{N}^{\text{lim}}_{\text{CF}-\text{EVs}} = M^U$ was the number of CF evaluations required by the full-search ML MUD. Fig. 7 compares the efficiency and reliability of the four discrete-binary EA-assisted MUDs. Observe that the DBGA-MUD was the winner with a zero failure rate and requiring only 3.2% of the ML-MUD's complexity. The DBDEA-MUD came a close second with an extremely low failure rate and an average complexity that was 3.7% of the optimal ML-MUD's complexity.

We then added the channel's AWGN and considered the cases of $E_{\text{b}}/N_{\text{o}} = 14$ and 20 dB. Fig. 8 compares the convergence behaviors of the four continuous-EA-assisted CE schemes. The approximate number of CF evaluations required for the mean square error (MSE) of a continuous-EA-assisted CE scheme to approach the CRLB[6] [39] was extracted in Fig. 8 and listed in Table III. It can be seen that the CRWBS-CE and the CDEA-CE had the fastest convergence speed, whereas the CGA-CE had the slowest convergence speed. Fig. 9 characterizes the convergence behaviors of the four discrete-binary

[6]The CRLB [39] provides the best attainable MSE performance for the optimal channel estimator based on the optimally designed pilots, and it is given by $\text{CRLB}(\mathbf{h}) = (\sigma_n^2/KE_{\text{s}})$ (e.g., [34]), where E_{s} denotes the average symbol energy.

TABLE III
NUMBERS OF CF EVALUATIONS REQUIRED FOR THE MSEs OF DIFFERENT CONTINUOUS-EA-ASSISTED CE SCHEMES TO APPROACH THE CRLB

Scheme	$E_{\rm b}/N_{\rm o} = 14$ dB	$E_{\rm b}/N_{\rm o} = 20$ dB
CGA-CE	43000	44000
CRWBS-CE	10000	13000
CPSO-CE	34000	36000
CDEA-CE	12000	17000

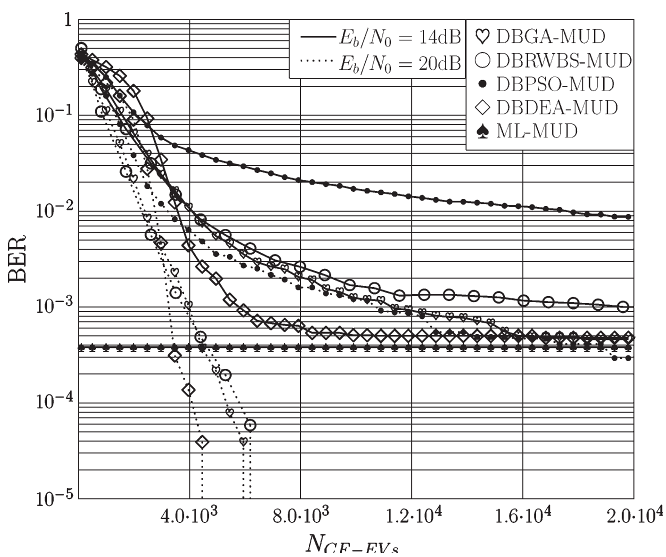


Fig. 9. BER versus the number of CF evaluations, which characterizes the convergence performance of the different discrete-binary EA-assisted MUDs. Note that at $E_{\rm b}/N_{\rm o} = 20$ dB, the optimal ML-MUD attains an infinitesimally low BER.

TABLE IV
NUMBERS OF CF EVALUATIONS REQUIRED FOR THE BERs OF DIFFERENT DISCRETE-BINARY EA-ASSISTED MUDs TO ATTAIN THE BER OF THE OPTIMAL ML-MUD

Scheme	$E_{\rm b}/N_{\rm o} = 14$ dB	$E_{\rm b}/N_{\rm o} = 20$ dB
DBGA-MUD	16000	6000
DBRWBS-MUD	> 20000	6500
DBPSO-MUD	failed	failed
DBDEA-MUD	10000	4500

EA-assisted MUDs. The approximate number of CF evaluations required for the BER of a discrete-binary EA-assisted MUD to approach the BER of the optimal ML-MUD was found in Fig. 9, and it is shown in Table IV. Observe that the DBDEA-MUD and the DBGA-MUD achieved rapid convergence. Although the nonturbo DBPSO-MUD failed to approach the ML-MUD solution in this experiment, by introducing the powerful turbo iterative procedure, the turbo DBPSO-MUD/decoder is capable of attaining the optimal solution of the turbo ML-MUD/decoder, as will be confirmed in Section V-B.

B. Performance of EA-Aided Joint CE and Turbo MUD/Decoder Schemes

Having examined the individual EA-assisted CE schemes and the individual EA-aided MUDs, we investigated the four EA-aided iterative joint CE and turbo MUD/decoder schemes, as outlined in Section IV, namely, the GA-aided

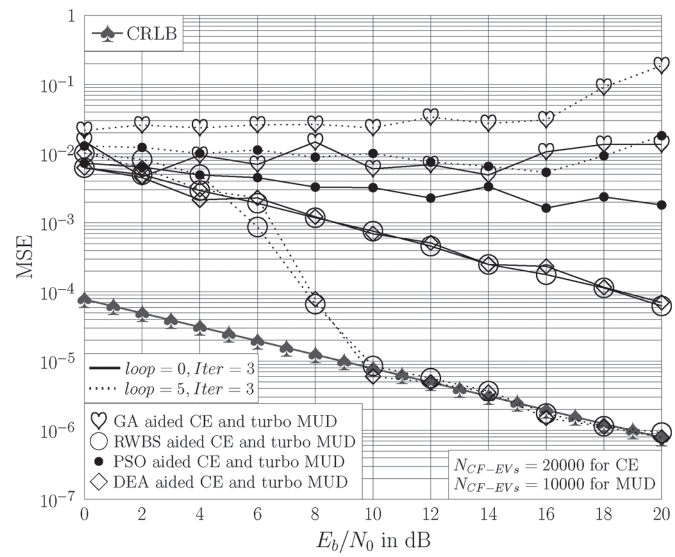


Fig. 10. Comparison of the MSE performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 20 000, and the number of CF evaluations for EA-aided MUD to 10 000.

joint CE and turbo MUD/decoder, the RWBS-aided joint CE and turbo MUD/decoder, the PSO-aided joint CE and turbo MUD/decoder, and the DEA-aided joint CE and turbo MUD/decoder. In an EA-aided joint CE and turbo MUD/decoder, the information is exchanged $I_{\rm tb}$ times at the inner turbo loop between the EA-assisted MUD and the channel decoder, whereas the information is exchanged $I_{\rm ce}$ times at the outer iterative loop between the EA-assisted CE scheme and the EA-aided turbo MUD/decoder. It is worth emphasizing that the EA-assisted channel estimator is based on the detected data fed back from the EA-assisted MUD/decoder. The MSE of the channel estimate obtained by an EA-aided joint CE and turbo MUD/decoder was compared with the CRLB, whereas the BER achieved by an EA-aided joint CE and turbo MUD/decoder was compared with the BER of the idealized turbo ML-MUD/decoder associated with perfect CSI.

Figs. 10 and 11 compare the MSE and BER performance, respectively, of the four EA-aided iterative joint CE and turbo MUD/decoder schemes, when fixing the number of the inner turbo iterations to $I_{\rm tb} = 3$, the number of CF evaluations for EA-aided CE to $N^{\rm ce}_{\rm CF-EVs} = 20000$ $(G_{\max} = 200)$, and the number of CF evaluations for EA-aided MUD to $N^{\rm mud}_{\rm CF-EVs} = 10000$ $(G_{\max} = 100)$. Observe in Fig. 10 that for $loop = 5$ outer iterations, the MSEs of the two channel estimates associated with the RWBS- and DEA-aided joint CE and turbo MUD/decoder schemes approached the CRLB for $E_{\rm b}/N_{\rm o} \geq 10$ dB; however, the PSO- and GA-aided joint CE and turbo MUD/decoder schemes exhibited divergence. Similarly, it is shown in Fig. 11 that for five outer iterations, the RWBS- and DEA-aided joint CE and turbo MUD/decoder schemes approached the BER performance of the idealized turbo ML-MUD/decoder; however, the PSO- and GA-aided joint CE and turbo MUD/decoder schemes failed to find the optimal solution.

From the results in Section V-A, we note that the PSO- and GA-aided joint CE and turbo MUD/decoder schemes

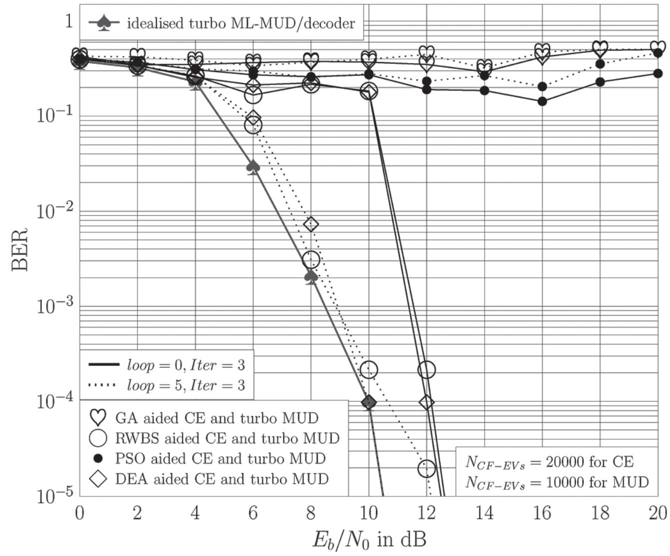


Fig. 11. Comparison of the BER performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 20 000, and the number of CF evaluations for EA-aided MUD to 10 000.

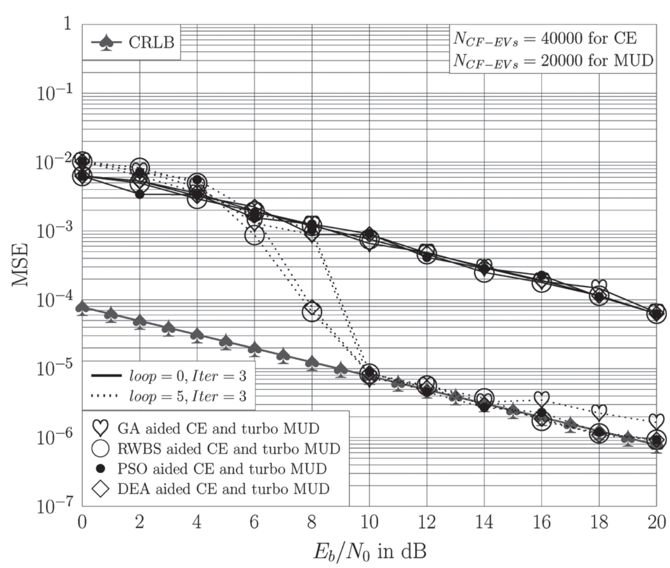


Fig. 12. Comparison of the MSE performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 40 000, and the number of CF evaluations for EA-aided MUD to 20 000.

may be less efficient in comparison to the RWBS- and DEA-aided schemes, and we surmise that $N_{\mathrm{CF-EVs}}^{\mathrm{ce}} = 20000$ and $N_{\mathrm{CF-EVs}}^{\mathrm{mud}} = 10000$ may not be sufficient for the PSO- and GA-aided schemes. We then opted for $N_{\mathrm{CF-EVs}}^{\mathrm{ce}} = 40000$ ($G_{\max} = 400$) and $N_{\mathrm{CF-EVs}}^{\mathrm{mud}} = 20000$ ($G_{\max} = 200$) and carried out simulations for the four EA-aided joint CE and turbo MUD/decoder schemes again. Figs. 12 and 13 show the achievable MSE and BER performance, respectively, for the four EA-aided joint CE and turbo MUD/decoder schemes. In Fig. 12, it is shown that the MSEs of the four channel estimates associated with the four EA-aided joint CE and turbo MUD/decoder schemes all approached the CRLB with $loop = 5$ outer iterations for $E_{\mathrm{b}}/N_{\mathrm{o}} \geq 10$ dB, whereas the BERs of the four EA-aided schemes all approached the optimal BER performance of the idealized turbo ML-MUD/decoder associated with perfect CSI, as shown in Fig. 13.

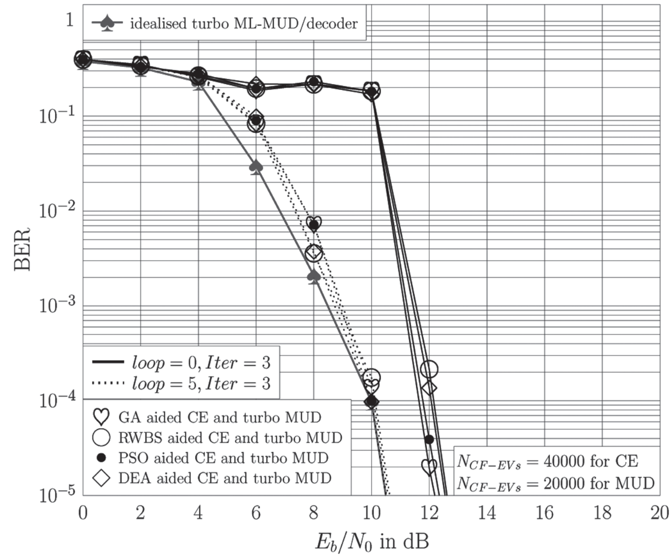


Fig. 13. Comparison of the BER performance for the four EA-aided joint CE and turbo MUD/decoder schemes recorded at the outer iterations $loop = 0$ and $loop = 5$, respectively, when fixing the number of the inner turbo iterations to $Iter = 3$, the number of CF evaluations for EA-aided CE to 40 000, and the number of CF evaluations for EA-aided MUD to 20 000.

Our computational complexity comparisons are provided in terms of the three ratios, namely, $C_{\mathrm{MUD}}^{\mathrm{EA}}/C_{\mathrm{MUD}}^{\mathrm{ML}}$, $C_{\mathrm{turbo}}^{\mathrm{EA}}/C_{\mathrm{turbo}}^{\mathrm{ML}}$, and $C_{\mathrm{joint}}^{\mathrm{EA}}/C_{\mathrm{turbo}}^{\mathrm{ML}}$, as shown in Table V. The ratio $C_{\mathrm{MUD}}^{\mathrm{EA}}/C_{\mathrm{MUD}}^{\mathrm{ML}}$ characterizes the complexity of an EA-aided MUD in comparison to that of the optimal full-search ML MUD. It can be seen from Table V that all the four EA-aided MUDs impose only 0.1% of the ML MUD's complexity. Given the CSI, the complexity of the RWBS- and DEA-assisted turbo MUD/decoder algorithms is less than 3.5% of the complexity of the turbo ML-MUD/decoder, whereas the complexity of the GA- and PSO-aided turbo MUD/decoder algorithms is less than 6.6% of the turbo ML-MUD/decoder's complexity, as seen in the column $C_{\mathrm{turbo}}^{\mathrm{EA}}/C_{\mathrm{turbo}}^{\mathrm{ML}}$ of Table V. An EA-aided joint CE and turbo MUD/decoder involves I_{ce} number of outer iterations between the EA-aided decision-directed channel estimator and the EA-assisted turbo MUD/decoder, and it performs blind joint CE and data detection. Comparing its complexity with that of the idealized turbo ML-MUD/decoder provided with the perfect CSI is really "unfair." Even so, from the column $C_{\mathrm{joint}}^{\mathrm{EA}}/C_{\mathrm{turbo}}^{\mathrm{ML}}$ in Table V, we can see that the total complexity of the RWBS- and DEA-assisted joint CE and turbo MUD/decoder schemes is less than 39% of the idealized turbo ML-MUD/decoder's complexity, whereas the GA- and PSO-assisted joint CE and turbo MUD/decoder schemes impose a total complexity that is less than 77% of the idealized turbo ML-MUD/decoder's complexity.

C. Comparing an EA-Aided CE With the Simplified LS CE

In Section II-B, we have pointed out that although the standard LS channel estimator [40] can also provide the optimal

TABLE V
COMPUTATIONAL COMPLEXITY COMPARISON IN TERMS OF THE RATIO OF THE COMPLEXITY OF AN EA-ASSISTED ITERATIVE JOINT CE AND TURBO MUD/DECODER TO THE COMPLEXITY OF THE IDEALIZED TURBO ML-MUD/DECODER ASSOCIATED WITH PERFECT CSI

Scheme	Operation	$C_{\mathrm{MUD}}^{EA}/C_{\mathrm{MUD}}^{ML}$	$C_{\mathrm{turbo}}^{EA}/C_{\mathrm{turbo}}^{ML}$	$C_{\mathrm{joint}}^{EA}/C_{\mathrm{turbo}}^{ML}$
GA assisted joint CE	multiplications	0.10%	5.69%	62.24%
and turbo MUD/decoder	additions	0.10%	7.45%	91.41%
RWBS assisted joint CE	multiplications	0.10%	3.00%	31.27%
and turbo MUD/decoder	additions	0.10%	3.88%	45.86%
PSO assisted joint CE	multiplications	0.10%	5.69%	62.24%
and turbo MUD/decoder	additions	0.10%	7.45%	91.41%
DE assisted joint CE	multiplications	0.10%	3.00%	31.27%
and turbo MUD/decoder	additions	0.10%	3.88%	45.86%

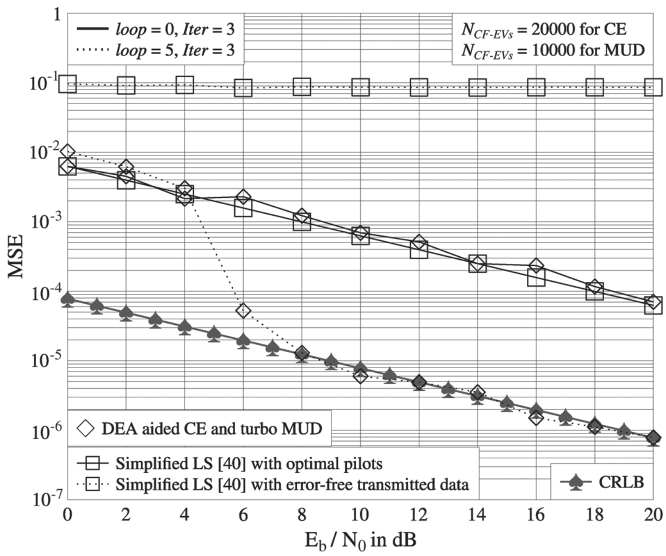


Fig. 14. Comparison of the MSE performance for the DEA-aided joint CE and turbo MUD/decoder scheme with that of the simplified LS channel estimator in [40].

solution for CE optimization (11), it is computationally very expensive. Therefore, it is difficult to combine the standard LS channel estimator with a turbo MUD/decoder to form a joint CE and turbo MUD/decoder scheme, as this approach will impose excessive computational complexity. The simplified LS channel estimator in [40], on the other hand, has low complexity, but it performs poorly even given with the correct error-free transmitted data. We now demonstrate this by investigating the MSE performance of the simplified LS channel estimator using our OFDM/SDMA simulation system. Fig. 14 shows the MSEs attained by the simplified LS CE relying on optimally designed pilots and the true error-free transmitted data, respectively, in comparison with the MSE performance obtained by the DEA-aided joint CE and turbo MUD/decoder recorder at $loop = 0$ and $loop = 5$.

Observe in Fig. 14 that the simplified LS channel estimator, given optimally designed pilots, attains the same MSE as the DEA-aided CE at $loop = 0$. However, this channel estimator performs very poorly even given with the true transmitted data, as shown in Fig. 14. The reason for this poor performance is that this low-complexity channel estimator requires optimal pilots, as discussed in [40, Sec. III], where the relative phases of the training sequences (pilots) for the different users (transmit antennas) must be carefully designed so that each individual CIR (linking the ith transmit antenna to the jth receive antenna) can be separately estimated. However, the users' transmitted data do not meet this requirement of "optimal pilots." Hence, this simplified LS CE cannot benefit from the iterative CE using the detected users' data—it cannot even work adequately using the true users' data. Therefore, the simplified LS channel estimator cannot be combined with a turbo MUD/decoder to form a joint CE and turbo MUD/decoder. By contrast, our proposed EA-aided CE benefits from the iterative joint CE and turbo MUD/decoding process and is capable of approaching the CRLB, as confirmed in Fig. 14.

VI. CONCLUSION

Four EAs, namely, the GA, RWBS, PSO, and DEA, have been applied to the challenging problem of joint semiblind CE and turbo MUD/decoding for ODFM/SDMA communication systems. Extensive results have been provided to demonstrate that by iteratively exchanging information between a continuous-EA-aided decision-directed channel estimator and a discrete-binary EA-assisted turbo MUD/decoder, an EA-aided joint blind CE and turbo MUD/decoder is capable of approaching both the CRLB associated with the optimal channel estimate and the BER of the idealized optimal turbo ML-MUD/decoder associated with perfect CSI, despite imposing only a fraction of the idealized turbo ML-MUD/decoder's complexity.

REFERENCES

[1] M. Jiang and L. Hanzo, "Multiuser MIMO-OFDM for next-generation wireless systems," *Proc. IEEE*, vol. 95, no. 7, pp. 1430–1469, Jul. 2007.

[2] L. Hanzo, Y. Akhtman, L. Wang, and M. Jiang, *MIMO-OFDM for LTE, WIFI and WIMAX: Coherent versus Non-Coherent and Cooperative Turbo-Transceivers.*. Chichester, U.K.: Wiley, 2011.

[3] J. A. C. Bingham, "Multicarrier modulation for data transmission: An idea whose time has come," *IEEE Commun. Mag.*, vol. 28, no. 5, pp. 5–14, May 1990.

[4] L. Hanzo, M. Münster, B. J. Choi, and T. Keller, *OFDM and MC-CDMA for Broadband Multi-User Communications, WLANs, and Broadcasting*. Chichester, U.K.: Wiley, 2003.

[5] P. Vandenameele, L. Van Der Perre, and M. Engels, *Space Division Multiple Access For Wireless Local Area Networks*. Boston, MA, USA: Kluwer, 2001.

[6] S. Chen, L. Hanzo, and A. Livingstone, "MBER space–time decision feedback equalization assisted multiuser detection for multiple antenna aided SDMA systems," *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 3090–3098, Aug. 2006.

[7] P. Vandenameele, L. Van Der Perre, M. Engels, B. Gyselinckx, and H. De Man, "A combined OFDM/SDMA approach," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 11, pp. 2312–2321, Nov. 2000.

[8] J. Zhang, L. Hanzo, and X. Mu, "Joint decision-directed channel and noise-variance estimation for MIMO OFDM/SDMA systems based

on expectation-conditional maximization," *IEEE Trans. Veh. Technol.*, vol. 60, no. 5, pp. 2139–2151, Jun. 2011.

[9] L. Hanzo, O. R. Alamri, M. El-Hajjar, and N. Wu, *Near-Capacity Multi-Functional MIMO Systems: Sphere-Packing, Iterative Detection and Cooperation*. Chichester, U.K.: Wiley, 2009.

[10] S. Thoen, L. Deneire, L. Van Der Perre, M. Engels, and H. De Man, "Constrained least squares detector for OFDM/SDMA-based wireless networks," *IEEE Trans. Wireless Commun.*, vol. 2, no. 1, pp. 129–140, Jan. 2003.

[11] X. Dai, "Pilot-aided OFDM/SDMA channel estimation with unknown timing offset," *Proc. Inst. Elect. Eng.—Commun.*, vol. 153, no. 3, pp. 392–398, Jun. 2006.

[12] J. Ylioinas and M. Juntti, "Iterative joint detection, decoding, and channel estimation in turbo coded MIMO-OFDM," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1784–1796, May 2009.

[13] L. Hanzo, S. Ng, T. Keller, and W. Webb, *Quadrature Amplitude Modulation: From Basics To Adaptive Trellis-Coded Turbo-Equalised and Space–Time Coded OFDM, CDMA and MC-CDMA Systems*. Chichester, U.K.: Wiley, 2004.

[14] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, Apr. 1997.

[15] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006.

[16] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: Univ. of Michigan Press, 1975.

[17] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA, USA: Addison-Wesley, 1989.

[18] S. Chen, X. X. Wang, and C. J. Harris, "Experiments with repeating weighted boosting search for optimization in signal processing applications," *IEEE Trans. Syst., Man, Cybern., B*, vol. 35, no. 4, pp. 682–693, Aug. 2005.

[19] S. F. Page, S. Chen, C. J. Harris, and N. M. White, "Repeated weighted boosting search for discrete or mixed search space and multiple-objective optimisation," *Appl. Soft Comput.*, vol. 12, no. 9, pp. 2740–2755, Sep. 2012.

[20] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Perth, WA, Australia, Nov. 27–Dec., 1, 1995, vol. 4, pp. 1942–1948.

[21] J. Kennedy and R. Eberhart, *Swarm Intelligence*. San Mateo, CA, USA: Morgan Kaufmann, 2001.

[22] K. Price, R. Storn, and J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Berlin, Germany: Springer-Verlag, 2005.

[23] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.

[24] S. Chen and Y. Wu, "Maximum likelihood joint channel and data estimation using genetic algorithms," *IEEE Trans. Signal Process.*, vol. 46, no. 5, pp. 1469–1473, May 1998.

[25] S. Chen and B. L. Luk, "Adaptive simulated annealing for optimization in signal processing applications," *Signal Process.*, vol. 79, no. 1, pp. 117–128, Nov. 1999.

[26] H. Ali, A. Doucet, and D. Amshah, "GSR: A new genetic algorithm for improving source and channel estimates," *IEEE Trans. Circuits Syst. I. Reg. Papers*, vol. 54, no. 5, pp. 1088–1098, May 2007.

[27] K. Yen and L. Hanzo, "Genetic algorithm assisted joint multiuser symbol detection and fading channel estimation for synchronous CDMA systems," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 6, pp. 985–998, Jun. 2001.

[28] K. Yen and L. Hanzo, "Genetic-algorithm-assisted multiuser detection in asynchronous CDMA communications," *IEEE Trans. Veh. Technol.*, vol. 53, no. 5, pp. 1413–1422, Sep. 2004.

[29] X. Wu, T. C. Chuah, B. S. Sharif, and O. R. Hinton, "Adaptive robust detection for CDMA using a genetic algorithm," *Proc. Inst. Elect. Eng.—Commun.*, vol. 150, no. 6, pp. 437–444, Dec. 2003.

[30] K. K. Soo, Y. M. Siu, W. S. Chan, L. Yang, and R. S. Chen, "Particle-swarm-optimization-based multiuser detector for CDMA communications," *IEEE Trans. Veh. Technol.*, vol. 56, no. 5, pp. 3006–3013, Sep. 2007.

[31] M. Y. Alias, S. Chen, and L. Hanzo, "Multiple antenna aided OFDM employing genetic algorithm assisted minimum bit error rate multiuser detection," *IEEE Trans. Veh. Technol.*, vol. 54, no. 5, pp. 1713–1721, Sep. 2005.

[32] M. Jiang, S. X. Ng, and L. Hanzo, "Hybrid iterative multiuser detection for channel coded space division multiple access OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 55, no. 1, pp. 115–127, Jan. 2006.

[33] M. Jiang, J. Akhtman, and L. Hanzo, "Iterative joint channel estimation and multi-user detection for multiple-antenna aided OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 6, no. 8, pp. 2904–2914, Aug. 2007.

[34] J. Zhang, S. Chen, X. Mu, and L. Hanzo, "Joint channel estimation and multi-user detection for SDMA/OFDM based on dual repeated weighted boosting search," *IEEE Trans. Veh. Technol.*, vol. 60, no. 7, pp. 3265–3275, Sep. 2011.

[35] W. Dong, J. Li, and Z. Lu, "Joint frequency offset and channel estimation for MIMO systems based on particle swarm optimization," in *Proc. VTC*, Singapore, May 11–14, 2008, pp. 862–866.

[36] M. Abuthinien, S. Chen, and L. Hanzo, "Semi-blind joint maximum likelihood channel estimation and data detection for MIMO systems," *IEEE Signal Process. Lett.* , vol. 15, pp. 202–205, 2008. **AQ1**

[37] S. Chen, W. Yao, H. Palally, and L. Hanzo, "Particle swarm optimisation aided MIMO transceiver designs," in *Computational Intelligence in Expensive Optimization Problems*, Y. Tenne and C. Goh, Eds. Berlin, Germany: Springer-Verlag, 2010, pp. 487–511.

[38] J. Zhang, S. Chen, X. Mu, and L. Hanzo, "Turbo multi-user detection for OFDM/SDMA systems relying on differential evolution aided iterative channel estimation," *IEEE Trans. Commun.*, vol. 60, no. 6, pp. 1621–1663, Jun. 2012.

[39] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.

[40] Y. Li, "Simplified channel estimation for OFDM systems with multiple transmit antennas," *IEEE Trans. Wireless Commun.*, vol. 1, no. 1, pp. 67–75, Jan. 2002.

[41] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Orlando, FL, USA, Oct. 12–15, 1997, vol. 5, pp. 4104–4108.

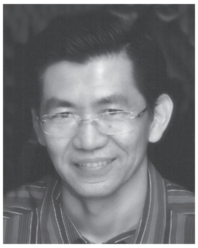
[42] M. A. Khanesar, M. Teshnehlab, and M. A. Shoorehdeli, "A novel binary particle swarm optimization," in *Proc. Mediterranean Conf. Control Autom.*, Athens, Greece, Jul. 27–29, 2007, pp. 1–6.

[43] T. Hanne, "On the convergence of multiobjective evolutionary algorithms," *Eur. J. Oper. Res.*, vol. 117, no. 3, pp. 553–564, Sep. 1999.

[44] X. Yao, "Unpacking and understanding evolutionary algorithms," in *Advances in Computational Intelligence*, J. Liu, C. Alippi, B. Bouchon-Meuniev, G. W. Greenwood, and H. A. Abbass, Eds. Berlin, Germany: Springer-Verlag, 2012, pp. 60–76.

Jiankang Zhang (S'08–M'12) received the B.Sc. degree in mathematics and applied mathematics from Beijing University of Posts and Telecommunications, Beijing, China, in 2006 and the Ph.D. degree in communication and information systems from Zhengzhou University, Zhengzhou, China, in 2012.

Since 2012, he has been a Lecturer with the School of Information Engineering, Zhengzhou University. From September 2009 to December 2011 and from January 2013 to May 2013, he was a Visiting Researcher with the Department of Electronics and Computer Science, University of Southampton, Southampton, U.K. His research interests include wireless communications and signal processing, including channel estimation, multiuser detection, beamforming/precoding, and optimization algorithms.



Sheng Chen (M'90–SM'97–F'08) received the B.Eng. degree in control engineering from the East China Petroleum Institute, Dongying, China, in 1982; the Ph.D. degree in control engineering from City University London, London, U.K., in 1986; and the D.Sc. degree from the University of Southampton, Southampton, U.K., in 2005.

From 1986 to 1999, he held research and academic appointments at the University of Sheffield, Sheffield, U.K.; The University of Edinburgh, Edinburgh, U.K.; and the University of Portsmouth, Portsmouth, U.K. Since 1999, he has been with the Department of Electronics and Computer Science, University of Southampton, where he is currently a Professor of intelligent systems and signal processing. He is also a Distinguished Adjunct Professor with King Abdulaziz University, Jeddah, Saudi Arabia. He has published over 480 research papers. His recent research interests include adaptive signal processing, wireless communications, modeling and identification of nonlinear systems, neural network and machine learning, intelligent control system design, evolutionary computation methods, and optimization.

Dr. Chen is a Chartered Engineer and a Fellow of the Institution of Engineering and Technology. He is an Institute for Scientific Information Highly Cited Researcher in the engineering category (March 2004).

Xiaomin Mu received the B.E. degree from Beijing Institute of Technology, Beijing, China, in 1982.

She is currently a Full Professor with the School of Information Engineering, Zhengzhou University, Zhengzhou, China. She has published many papers in the field of signal processing and coauthored two books. Her research interests include signal processing in communication systems, wireless communications, and cognitive radio.

Lajos Hanzo (M'91–SM'92–F'04) received the M.S. degree in electronics and the Ph.D. degree from the Technical University of Budapest, Budapest, Hungary, in 1976 and 1983, respectively, and the D.Sc. degree from the University of Southampton, Southampton, U.K., in 2004. In 2009, he was awarded the honorary doctorate "Doctor Honoris Causa" by the Technical University of Budapest. AQ2

During his 37-year career in telecommunications, he held various research and academic posts in Hungary, Germany, and the U.K. He was a Chaired Professor with Tsinghua University, Beijing, China. Since 1986, he has been with the Department of Electronics and Computer Science, University of Southampton, where he holds the Chair in Telecommunications. He coauthored 20 John Wiley/IEEE Press books on mobile radio communications totaling in excess of 10 000 pages and published 1356 research entries at IEEE Xplore. He has successfully supervised 83 Ph.D. students. Currently, he is directing a 100-strong academic research team, working on a range of research projects in the field of wireless multimedia communications sponsored by industry, the UK Engineering and Physical Sciences Research Council, the European Research Council, and the Royal Society. He is an enthusiastic supporter of industrial and academic liaison, and he offers a range of industrial courses. He has over 17 000 citations. (For further information on research in progress and associated publications, please refer to http://www-mobile.ecs.soton.ac.uk.)

Dr. Hanzo was the Editor-in-Chief of the IEEE Press from 2008 to 2012. He is also the Governor of the IEEE Vehicular Technology Society. He is a Fellow of the Royal Academy of Engineering, the Institution of Engineering and Technology, and the European Association for Signal Processing. He acted both as Technical Program Committee and General Chair of IEEE conferences and presented keynote lectures. He has been awarded a number of distinctions, including the European Research Council's Senior Research Fellow Grant and the Royal Society's Wolfson Research Merit Award.

AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

AQ1 = Please provide issue number and month of publication in Ref. [36].
AQ2 = Please check if the first paragraph in author L. Hanzo's vitae is captured appropriately.

END OF ALL QUERIES