Differential Evolution Algorithm Aided MBER Beamforming Receiver for Quadrature Amplitude Modulation Systems

Jiankang Zhang, Sheng Chen, Xiaomin Mu and Lajos Hanzo

Abstract—Evolutionary computational intelligence methods have found wide-ranging application in communication and other walks of engineering. The main attraction of adopting evolutionary computational intelligence algorithms is that they may facilitate global or near global optimal designs with affordable computational costs. This contribution considers the beamforming assisted multiple-antenna receiver for multi-user quadrature amplitude modulation systems. The bit error rate (BER) expression as the function of the beamformer’s weight vector is derived explicitly. The minimum BER (MBER) beamforming receiver can then be obtained as the solution of the resulting optimisation problem that minimises the MBER criterion. We propose to employ a differential evolution algorithm to solve the MBER optimisation by its virtue of computational efficiency and ability to locate a global minimum quickly.

I. INTRODUCTION

Evolutionary or bio-inspired computational intelligence methods, such as the generic algorithm (GA), ant colony optimisation (ACO), particle swarm optimisation (PSO), and differential evolution (DE) algorithm have found ever-increasing applications in all walks of engineering, especially communication signal processing, where attaining global or near global optimal solutions at affordable computational costs are critical. The Communication Research Group at the University of Southampton has a long and successful record in applying the GA, ACO and PSO in communication system design applications [1]–[12]. Recently, DE has become popular and has been applied to a variety of engineering applications. The DE algorithm [13]–[16] constitutes a random guided population-based search method, which employs repeated cycles of candidate-solution re-combination and selection operations for guiding the population towards the vicinity of a global optimum. Supported by extensive empirical results, it is believed that the DE algorithm is capable of arriving at a globally optimal solution very efficiently. The effectiveness of DE in tackling challenging optimisation problems have now widely been recognised by the computational intelligence community. This contribution reports an application of the DE algorithm to multiple-antenna communication receiver design.

The ever-increasing demand for mobile communication capacity has motivated the employment of space-division multiple access (SDMA) for the sake of improving the achievable spectral efficiency. A particular approach that has shown real promise in achieving substantial capacity enhancements is the use of adaptive beamforming receiver with antenna arrays [17]–[19]. Classically, beamforming design is based on minimising the mean square error (MSE) criterion. For a communication system, however, it is the bit error rate (BER), not the MSE, that really matters, the minimum BER (MBER) beamforming has been derived for binary phase shift keying (BPSK) systems [20], which has a bandwidth efficiency of 1 bit per symbol, and quadrature phase shift keying (QPSK) systems [21], which enables a bandwidth efficiency of 2 bits per symbol. Note that a QPSK system basically consists of two BPSK systems. Quadrature amplitude modulation (QAM) schemes [22], which offers much higher bandwidth efficiency, have become popular in numerous wireless standards by virtue of providing a high throughput. Minimum symbol error rate (MSER) beamforming receiver has been conceived for QAM systems [23].

To the best of our knowledge, however, no direct MBER solution has been derived for QAM systems to date. We explicitly derive the BER expression as the function of the beamformer’s weight vector, and formulate the MBER beamforming for QAM systems as the solution of the resulting optimisation problem that minimises the MBER criterion. In principle, the MBER beamforming design for the QAM system can be obtained by minimising the BER criterion based on a gradient-descent algorithm [20], [21], [24]. However, there are some potential drawbacks associated with the gradient-descent approach. The calculation of the gradient for the BER cost function of the QAM beamforming design imposes high computational complexity, and the gradient-descent algorithm converges slowly owing to the fact that the BER cost function is a highly complex nonlinear function of the beamformer’s weight vector. The initial choice of the weight vector also significantly influences the solution obtained by the gradient-based optimisation. To overcome these difficulties, we adopt a DE algorithm [13]–[16] to this challenging MBER optimisation. The DE algorithm has the capability to arrive at a globally optimal solution very efficiently, and this makes the DE algorithm aided MBER design computationally attractive.

II. SYSTEM MODEL

Consider the SDMA system that employs the \( L \)-element receive antenna array to support \( M \) QAM users. The receive signal vector \( \mathbf{x}(k) = \begin{bmatrix} x_1(k) \ x_2(k) \cdots x_L(k) \end{bmatrix}^T \) can be expressed as [19], [23]

\[
\mathbf{x}(k) = \mathbf{Pb}(k) + \mathbf{n}(k) = \bar{\mathbf{x}}(k) + \mathbf{n}(k),
\]
where $k$ denotes the symbol index, the Gaussian white noise vector $\mathbf{n}(k) = [n_1(k) \ n_2(k) \ldots n_L(k)]^T$ has a covariance matrix $E[\mathbf{n}(k)\mathbf{n}^H(k)] = 2\sigma_n^2 I_L$ with $I_L$ denoting the $L \times L$ identity matrix, $\mathbf{b}(k) = [b_1(k) \ b_2(k) \ldots b_M(k)]^T$ is the transmitted symbol vector of the $M$ users, and the system matrix $\mathbf{P}$ is given by
\[
\mathbf{P} = [A_1 \mathbf{s}_1 \ A_2 \mathbf{s}_2 \ldots A_M \mathbf{s}_M] = [\mathbf{p}_1 \ \mathbf{p}_2 \ldots \mathbf{p}_M],
\]
with $A_i$ being the $i$th channel coefficient and the steering vector for user $i$ given by
\[
\mathbf{s}_i = [e^{j\omega_i t_i(\theta_i)} \ e^{j\omega_i t_x(\theta_i)} \ldots e^{j\omega_i t_L(\theta_i)}]^T.
\]
In (3), $t_i(\theta_i)$ is the relative time delay at array element $l$ for user $i$, $\theta_i$ is the direction of arrival for user $i$, and $\omega_i = 2\pi f_c$ is the angular carrier frequency. We define the system’s signal to noise ratio as $\text{SNR} = E_b/N_0 = E_b/2\sigma_n^2$, where $E_b$ is the average energy per bit of the QAM symbol.

For notational simplicity, we assume the 16-QAM modulation. Therefore, the $4$th transmitted symbol of user $i$
\[
b_i(k) \in \{\pm 1 \pm j, \ \pm 1 \pm 3j, \ \pm 3 \pm j, \ \pm 3 \pm 3j\},
\]
where $j = \sqrt{-1}$. The approach adopted in this study, however, can be extended to higher-order QAM schemes. Without loss of generality, user 1 is assumed to be the desired user and the rest of the sources are the interfering users. A linear beamformer is employed, whose output is given by
\[
y(k) = \mathbf{w}^H \mathbf{x}(k) = \mathbf{w}^H \mathbf{x}(k) + \mathbf{w}^H \mathbf{n}(k) = \bar{y}(k) + e(k),
\]
where $\mathbf{w} = [w_1 \ w_2 \ldots w_L]^T$ is the complex-valued beamformer weight vector, and $e(k)$ is Gaussian distributed with zero mean and $E[|e(k)|^2] = 2\sigma_n^2\mathbf{w}^H\mathbf{w}$. Define the combined impulse response of the beamformer and the system as $\mathbf{w}^H \mathbf{P} = [c_1 \ c_2 \cdots c_M]$. The beamformer’s output can alternatively be expressed as
\[
y(k) = c_1b_1(k) + \sum_{i=2}^{M} c_ib_i(k) + e(k),
\]
where the first term is the desired signal and the second term represents the residual interfering signal. Denote $y_R(k) = \mathbb{R}[y(k)]$ and $y_I(k) = \mathbb{I}[y(k)]$. Provided that $c_1$ is real-valued and positive, the decisions regarding the in-phase and two quadrature bits can be made separately based on $y_R(k)$ and $y_I(k)$, respectively. If $c_1 = \mathbf{w}^H \mathbf{p}_1$ is complex-valued, the rotating operation $\mathbf{w}^\text{new} = \frac{c_1\mathbf{w}^\text{old}}{|c_1|}$ can be used to ensure a real and positive $c_1$. This rotation is a linear transformation and does not alter the system’s BER [23].

The in-phase bit combinations that map to the symbols $b_{R_1}(k) = \mathbb{R}[b_1(k)] = -3, -1, 1, 3$ are 11, 10, 00, 01. The two in-phase bits that form the real part of 16-QAM symbol are known as the class 1 (C1) and class 2 (C2) bits, respectively [22]. The decision for the in-phase C1 bit is given by
\[
\begin{align*}
C1 \text{ bit} = 0, & \quad \text{if } y_R(k) > 0, \\
C1 \text{ bit} = 1, & \quad \text{if } y_R(k) \leq 0.
\end{align*}
\]
and the decision regarding the in-phase C2 bit is given by
\[
\begin{align*}
C2 \text{ bit} = 0, & \quad \text{if } -2c_1 < y_R(k) < 2c_1, \\
C2 \text{ bit} = 1, & \quad \text{if } y_R(k) \leq -2c_1 \text{ or } y_R(k) \geq 2c_1.
\end{align*}
\]
(8)
The decision rules for the quadrature C1 and C2 bits are given similarly based on $y_I(k)$.

Traditionally, the beamformer’s weight vector is determined by minimising the MSE metric of $E[|b_1(k) - y(k)|^2]$, which leads to the following minimum MSE (MMSE) solution [25]
\[
\mathbf{w}_{\text{MMSE}} = \left( \mathbf{P} \mathbf{P}^H + 2\sigma_n^2 \mathbf{I}_L \right)^{-1} \mathbf{p}_1,
\]
where $\sigma_n^2$ is the average QAM symbol energy. The previous work [23] has derived the beamforming solution based on minimising the system’s achievable SER. One of the contributions of this study is to derive the MBER beamforming solution for SDMA based QAM systems. As we will see, the derivation of the BER expression as a function of the beamformer’s weight vector is much more complicated than the SER expression given in [23].

III. Minimum Bit Error Rate Beamforming

The noise-free part of the beamformer input takes values from the finite set given by $\bar{y}(k) \in \mathcal{X} \triangleq \{\bar{y}(q) = \mathbf{P}\mathbf{b}(q), 1 \leq q \leq N_b\}$, where $N_b = 16^M$ and $\mathbf{b}(q), 1 \leq q \leq N_b$ are all the legitimate equiprobable sequences of $\mathbf{b}(k)$. Thus, the noise-free part of the beamformer output only takes values from the finite set given by $\bar{y}(k) \in \mathcal{Y} \triangleq \{\bar{y}(q) = \mathbf{w}^H\bar{y}(q), 1 \leq q \leq N_b\}$. The real and imaginary parts of the set $\mathcal{Y}$ are respectively
\[
\begin{align*}
\mathcal{Y}_R & \triangleq \{\bar{y}_R(q) = \mathbb{R}[\bar{y}(q)], \bar{y}_R(q) \in \mathcal{Y}\}, \\
\mathcal{Y}_I & \triangleq \{\bar{y}_I(q) = \mathbb{I}[\bar{y}(q)], \bar{y}_I(q) \in \mathcal{Y}\}.
\end{align*}
\]

The set $\mathcal{Y}_R$ can be divided into the four conditioned subsets
\[
\begin{align*}
\mathcal{Y}_R^{(\pm 1)} & \triangleq \{\bar{y}_R(q) \in \mathcal{Y}_R : b_{R_1}(k) = \pm 1\}, \\
\mathcal{Y}_R^{(\pm 3)} & \triangleq \{\bar{y}_R(q) \in \mathcal{Y}_R : b_{R_1}(k) = \pm 3\}.
\end{align*}
\]
Similarly, $\mathcal{Y}_I$ can be partitioned into the four subsets $\mathcal{Y}_I^{(\pm 1)}$ and $\mathcal{Y}_I^{(\pm 3)}$, depending on the values of $b_{I_1}(k)$. The number of the points in each of these subsets is $N_{sb} = N_b/4$.

The conditional PDF of $y_R(k)$ given $b_{R_1}(k) = +1$ is
\[
p(y_R+i) = \frac{1}{N_{sb}} \sum_{y_R(q) \in \mathcal{Y}_R^{(i)}} \frac{1}{2\pi \sigma_n^2 \mathbf{w}^H \mathbf{w}} e^{-\frac{(y_R(q)-y_R)^2}{2\pi \sigma_n^2 \mathbf{w}^H \mathbf{w}}},
\]
where $i = 1, 3$. Taking into account the symmetric distribution of $\mathcal{Y}_R^{(i)}$ and $\mathcal{Y}_R^{(\bar{i})}$ with respect to the decision boundary $y_R = 0$ [23], the in-phase C1 bit error probability can be found to be
\[
P_{E_R,C1}(w) = \frac{1}{2N_{sb}} \sum_{y_R(q) \in \mathcal{Y}_R^{(+)}} Q \left( y_R(q)/2 \left| E_R,C1(w) \right. \right),
\]
(13)
where \( Y_R^{(+) \!} = Y_R^{(1)} \cup Y_R^{(3)} \), \( Q(u) = \int_{-\infty}^{\infty} e^{(-u^2)} \),
\[
g^{(q)}_{R,C1}(w) = \frac{\text{sgn}(\Re\{b_1^{(q)}(w)\})\overline{g}_R^{(q)}}{\sigma_n\sqrt{W^H w}} = \frac{\text{sgn}(b_1^{(q)}(w))\Re\{w^H\overline{g}^{(q)}(w)\}}{\sigma_n\sqrt{W^H w}}, \tag{14}
\]
and \( b_1^{(q)} \) denotes the first element of \( b^{(q)} \), corresponding to the desired user’s symbol \( b_1^{(k)} \). Similarly, the quadrature C1 bit error probability is given by
\[
P_{E_1,C1}(w) = \frac{1}{2N^s_b} \sum_{\overline{g}_R^{(q)}} Q\left(g^{(q)}_{R,C1}(w)\right)
\text{with } Y_I^{(1)} = Y_I^{(1)} \cup Y_I^{(3)}
\]
and
\[
g^{(q)}_{I,C1}(w) = \frac{\text{sgn}(\Im\{b_1^{(q)}(w)\})\overline{g}_I^{(q)}}{\sigma_n\sqrt{W^H w}} = \frac{\text{sgn}(b_1^{(q)}(w))\Im\{w^H\overline{g}^{(q)}(w)\}}{\sigma_n\sqrt{W^H w}}, \tag{16}
\]
with \( g^{(q)}_{R,C1}(w) \), \( g^{(q)}_{R,C2}(w) \), \( g^{(q)}_{I,C1}(w) \), \( g^{(q)}_{I,C2}(w) \). The C2 bit error rate is much more involved. After some lengthy derivation, which we again omit here for space economy, the conditional in-phase C2 BER given \( b_{R_1}(k) = +1 \) can be shown to be
\[
P_{E_r,C2}^{(1)}(w) = \frac{1}{N_s b} \sum_{\overline{g}_R^{(q)}} \left( Q\left(g^{(q,a)}_{R,C2}(w)\right) + Q\left(g^{(q,b)}_{R,C2}(w)\right) \right),
\]
with
\[
g^{(q,a)}_{R,C2}(w) = \frac{2c_1 + \text{sgn}(b_{R_1}(w))\overline{g}_R^{(q)}}{\sigma_n\sqrt{W^H w}} \tag{18}
\]
and
\[
g^{(q,b)}_{R,C2}(w) = \frac{2c_1 - \text{sgn}(b_{R_1}(w))\overline{g}_R^{(q)}}{\sigma_n\sqrt{W^H w}} \tag{19}
\]
After a lengthy simplification, which we again omit here for space economy, the conditional in-phase C2 BER given \( b_{R_1}(k) = +3 \) can be expressed as
\[
P_{E_r,C2}^{(3)}(w) = \frac{1}{N_s b} \sum_{\overline{g}_R^{(q)}} \left( Q\left(g^{(q,c)}_{R,C2}(w)\right) - Q\left(g^{(q,a)}_{R,C2}(w)\right) \right),
\]
where
\[
g^{(q,c)}_{R,C2}(w) = \frac{\text{sgn}(b_{R_1}(w))\overline{g}_R^{(q)}}{\sigma_n\sqrt{W^H w}} - 2c_1 \tag{21}
\]
Thus, the in-phase C2 bit error probability is given by
\[
P_{E_r,C2}(w) = \frac{1}{2} \left( P_{E_r,C2}^{(1)}(w) + P_{E_r,C2}^{(3)}(w) \right). \tag{22}
\]
Similarly, the quadrature C2 bit error probability is given by
\[
P_{E_1,C2}(w) = \frac{1}{2} \left( P_{E_1,C2}^{(1)}(w) + P_{E_1,C2}^{(3)}(w) \right), \tag{23}
\]
where
\[
P_{E_1,C2}^{(1)}(w) = \frac{1}{N_s b} \sum_{\overline{g}_R^{(q)}} \left( Q\left(g^{(q,a)}_{I,C2}(w)\right) + Q\left(g^{(q,b)}_{I,C2}(w)\right) \right),
\]
\[
P_{E_1,C2}^{(3)}(w) = \frac{1}{N_s b} \sum_{\overline{g}_R^{(q)}} \left( Q\left(g^{(q,c)}_{I,C2}(w)\right) - Q\left(g^{(q,b)}_{I,C2}(w)\right) \right)
\]
while \( g^{(q,a)}_{I,C1}(w) \), \( g^{(q,b)}_{I,C1}(w) \), \( g^{(q,c)}_{I,C1}(w) \) and \( g^{(q,a)}_{I,C2}(w) \), \( g^{(q,b)}_{I,C2}(w) \), \( g^{(q,c)}_{I,C2}(w) \) are obtained by substituting \( \text{sgn}(b_{R_1}^{(q)}) \) and \( \text{sgn}(\overline{g}_R^{(q)}) \) with \( \text{sgn}(b_{R_1}^{(q)}) \) and \( \overline{g}_R^{(q)} \) in (18), (19) and (21), respectively.

The BER of the 16-QAM beamformer with weight vector \( w \) is therefore given by
\[
P_E(w) = \frac{1}{4} \left( P_{E_{R,C1}}(w) + P_{E_{I,C1}}(w) + P_{E_{R,C2}}(w) + P_{E_{I,C2}}(w) \right), \tag{26}
\]
and the MBER beamformer solution is defined as
\[
\mathbf{w}_{\text{MBER}} = \arg \min_w P_E(w). \tag{27}
\]
A MBER beamformer design may be obtained based on a gradient-descent numerical optimisation. However, calculating the gradient of \( P_E(w) \) may require extensive computation and, therefore, a gradient-based algorithm may not be computationally efficient. Furthermore, the choice of the initial weight value can significantly affect the convergence speed and the quality of the final solution obtained.

IV. DIFFERENTIAL EVOLUTION ALGORITHM

As a relatively new member in the family of evolutionary algorithms, the DE algorithm has its distinctive feature in that it mutates candidate-solution vectors by adding weighted random difference-vector to them, which makes it more powerful and efficient in arriving at the globally optimal solution. A typical DE algorithm [13]–[16] is characterised with its initialisation, mutation, re-combination and selection operations invoked for exploring the search space in an iterative procedure, until some termination criteria are met. We employ the DE algorithm, as illustrated in Fig. 1, to find an MBER solution of the optimisation problem defined in (27). The algorithm is detailed as follows.

1) Initialisation. DE algorithm commences its search from a population of \( P_s \) \( L \)-dimensional complex-valued solution vectors. The \( p_{th} \) vector of the population in the first generation of \( g = 1 \) may be readily expressed as
\[
\mathbf{w}_{1,p} = [\tilde{w}_{1,p,1} \; \tilde{w}_{1,p,2} \cdots \tilde{w}_{1,p,L}]^T \tag{28}
\]
where \( L \) is the number of antenna elements. The initial population \( \{\mathbf{w}_{1,p}\}_{p=1}^{P_s} \) is randomly generated within the search space, and one of the initial candidate-solution vectors may be set to the MMSE solution given in (9).

2) Mutation. The mutation operation allows DE to maintain the diversity of the population, while insightfully steering the optimisation. The appropriate choice of the mutation parameters allows DE to prevent “premature convergence” to a local minimum without thoroughly exploring the entire solution space. Mutation is one of the distinctive features of the DE algorithm, which does not use a predefined probability density function for generating the perturbed solutions. Instead, it relies upon the population itself in perturbing the candidate solutions by adding an appropriately scaled and randomly selected difference-vector to a base population vector. More
Randomly and uniformly generate the initial population in the search space at the first generation, i.e., $g = 1$.

- **Mutation**: The mutation operation creates a mutant vector by combining three different, randomly chosen vectors according to

  $$ \tilde{v}_{g,1} = \tilde{w}_{g,1} \cdot \gamma (\tilde{w}_{g,2} - \tilde{w}_{g,3}) $$

  where $\gamma \in (0, 1]$ is a real positive number that controls the rate at which the population evolves. A larger value of $\gamma$ results in a higher diversity in the population, while a lower value promotes faster convergence.

- **Crossover**: The crossover generates a trial vector by replacing certain parameters of the target vector with the corresponding parameters of a randomly selected donor vector. As a significant complementarity to the above-mentioned differential mutation, the crossover operation increases the potential diversity of the population vectors. There exist diverse crossover mechanisms [13]–[16]. We opted for employing the uniform crossover algorithm, where each parameter, regardless of its location in the trial vector, has the same probability of inheriting its value from a given vector.

$$
\begin{align*}
\text{Target} & \quad \text{Trial} \\
\text{Select trial or target} & \quad w_{g+1,p} = \begin{cases} 
\tilde{u}_{g,p} & P_E(\tilde{u}_{g,p}) \leq P_E(\tilde{w}_{g,p}) \\
\tilde{w}_{g,p} & \text{otherwise}
\end{cases}
\end{align*}
$$

Specifically, a mutant vector is created by combining three different, randomly chosen vectors according to

$$
\tilde{v}_{g,1} = \tilde{w}_{g,1} + \gamma (\tilde{w}_{g,2} - \tilde{w}_{g,3}), \quad (29)
$$

where $\gamma \in (0, 1]$ is a real positive number that controls the rate at which the population evolves. A larger value of $\gamma$ results in a higher diversity in the population, while a lower value promotes faster convergence.

3) **Crossover**: The crossover generates a trial vector by replacing certain parameters of the target vector with the corresponding parameters of a randomly selected donor vector. As a significant complementarity to the above-mentioned differential mutation, the crossover operation increases the potential diversity of the population vectors. There exist diverse crossover mechanisms [13]–[16]. We opted for employing the uniform crossover algorithm, where each parameter, regardless of its location in the trial vector, has the same probability of inheriting its value from a given vector.

**Fig. 1.** Flowchart of the differential evolution algorithm for solving the MBER beamforming design.
More specifically, the \(l\)-th element of the \(i\)-th vector in the population at the \(g\)-th generation, namely, \(\hat{u}_{g,i,l}\), is given by

\[
\hat{u}_{g,i,l} = \begin{cases} 
\hat{v}_{g,i,l}, & \text{rand}(0, 1) \leq C_r \text{ or } l = l_{\text{rand}}, \\
\hat{w}_{g,i,l}, & \text{otherwise},
\end{cases}
\]

where \(C_r \in [0, 1]\) is the crossover probability, which represents the specific weight applied to the parameter values that are copied from a previous vector to the mutant, and \(\text{rand}(0, 1)\) denotes a random number uniformly distributed in the range \([0, 1]\), while \(l_{\text{rand}}\) is a randomly and uniformly drawn integer from the integer set \(\{1, 2, \cdots, L\}\). Note that the crossover is guaranteed at least at one element, i.e. \(\hat{u}_{g,i}\) differs from \(\hat{w}_{g,i}\) at least at the (random) position \(l = l_{\text{rand}}\).

4) Selection. The selection operator determines whether the target vector \(\hat{w}_{g,p,s}\) or the trial vector \(\hat{u}_{g,p,s}\) survives to the next generation. Unlike the GA, the DE algorithm does not use fitness-based selection for the next generation. Instead, the cost function (CF) of the trial vector \(\hat{u}_{g,p,s}\), namely, \(P_E(\hat{u}_{g,p,s})\), is compared to that of \(\hat{w}_{g,p,s}\). If the trial vector has lower or equal CF value in comparison to the corresponding target vector, the trial vector replaces the target vector to proceed to the next generation. Otherwise, the target vector remains in the population for the next generation. Specifically, the selection procedure is described by

\[
\hat{w}_{g+1,p,s} = \begin{cases} 
\hat{u}_{g,p,s}, & P_E(\hat{u}_{g,p,s}) \leq P_E(\hat{w}_{g,p,s}), \\
\hat{w}_{g,p,s}, & \text{otherwise},
\end{cases}
\]

The selection operation also maintains a constant population size \(P_s\) over all generations.

Once the new population is created, the above-mentioned three processes, the mutation, crossover and selection, are repeated, until the termination criterion described below is met.

5) Termination. The ultimate stopping criterion would be that the optimal MBER solution has indeed been found. However, it is impossible in practice to glean any proof of evidence to confirm this. Therefore, we opt to halt the optimisation procedure, when any of the following two stopping criteria are satisfied:

- The pre-defined maximum affordable number of generations \(G_{\text{max}}\) has been exhausted.
- \(\Delta g_{\text{max}}\) generations have been explored without a trial vector being accepted.

V. SIMULATION STUDY

The simulated beamforming systems consisted of four 16-QAM user sources with a four-element linear uniform antenna array, which represents a full-rank system of \(L \geq M\), and a three-element linear uniform antenna array, which renders the system into a rank-deficient one of \(L < M\). Conventional beamforming receiver based on the MMSE design of (9) requires the system to be full rank. Previous study [23] has demonstrated that the MSER design significantly outperforms the MMSE solution, particularly in the rank-deficient case. The locations of the desired user and the interfering users were graphically illustrated in Fig. 2. All the four users were assumed to have an equal transmit power, and the four channel taps, \(A_i\) for \(1 \leq i \leq 4\), were identical. The interfering user 4 was assumed to be the “heaviest” interferer, which had the minimum angular separation with the desired user, that is, \(\theta < 65^\circ\). The three basic algorithmic parameters of the DE algorithm, namely, the population size, the scaling factor and the crossover probability were empirically set to \(P_s = 100\), \(\gamma = 0.4\) and \(C_r = 0.4\), respectively. The maximum affordable number of generations was set to \(G_{\text{max}} = 200\). In our simulation, the perfect channel knowledge was assumed at the receiver. The proposed MBER beamforming receiver was compared with the classical MMSE beamforming receiver (9) as well as the MSER beamforming receiver of [23]. It is expected that the MBER and MSER beamforming solutions should achieve the same performance in terms of BER.

Specifically, the BER performance of the MBER-based beamforming for the full-rank scenario was shown in Fig. 3, in comparison with those of the MSER-based beamforming and MMSE-based beamforming. As expected, the MBER beamforming had the same BER performance as the MSER beamforming, and they outperformed the MMSE beamforming with about 2 dB, 2.5 dB and 3.5 dB when the minimum
separation angular $\theta = 40^\circ$, $30^\circ$ and $20^\circ$, respectively.

For the challenging rank-deficient scenario of using three-element receive antenna array to support the four 16-QAM users, the BER performance comparison of the three beamforming solutions is depicted in Fig. 4, where it can be seen again that the MBER and MSER solutions attained the same performance which was significantly better than the performance achievable by the MMSE solution.

VI. CONCLUSIONS

We have proposed an DE assisted MBER beamforming receiver for multi-user SDMA based QAM systems. The BER formula has been derived explicitly, and the optimal MBER solution has been obtained by minimising the BER cost function using an DE algorithm. Simulation results have demonstrated the effectiveness of the DE assisted MBER-based beamforming receiver. As expected, the results obtained have confirmed that the MBER beamforming receiver attains the same BER performance as the MSER beamforming receiver, and the both solutions significantly outperform the standard MMSE-based beamforming receiver, regardless of the full-rank and rank-deficient scenarios.

This study has demonstrated the effectiveness of the DE algorithm as a design tool for communication signal processing applications. Our further work will compare a range of evolutionary computational intelligence methods, including the GA, ACO, PSA and DE algorithm, in benchmark communication system design problems.

REFERENCES


