A Modified Differential Evolution for Symbol Detection in MIMO-OFDM System

Aritra Sen[1], Subhrajit Roy[1] and Swagatam Das[2]

[1] School of Electrical and Electronic Engineering,
Nanyang Technological University, Singapore 639798
[2] Electronics and Communications Sciences Unit,
Indian Statistical Institute, Kolkata 700 108, India
ARITRA001@e.ntu.edu.sg, subhrajit.roy@ntu.edu.sg,
swagatam.das@isical.ac.in

Abstract. It is essential to estimate the Channel and detect symbol in multiple-input and multiple-output (MIMO)-orthogonal frequency division multiplexing (OFDM) systems. Symbol detection by applying the maximum likelihood (ML) detector gives excellent performance but in systems with higher number of antennas and greater constellation size, the computational complexity of this algorithm becomes quite high. In this paper we apply a recently developed modified Differential Evolution (DE) algorithm with novel mutation, crossover as well as parameter adaptation strategies (MDE_pBX) for reducing the search space of the ML detector and the computational complexity of symbol detection in MIMO-OFDM systems. The performance of MDE_pBX have been compared with two classical symbol detectors namely ML and ZF and two famous evolutionary algorithm namely SaDE and CLPSO.

1 Introduction

Orthogonal frequency division multiplexing (OFDM)[1] is a standard multi-carrier modulation in high data rate wireless as well as wired communication systems. OFDM has the potential to increase spectral efficiency. This attribute has recently attracted much attention to OFDM so that data rate transmission can be increased considerably in modern communication systems.Recent communication systems like WLAN, HIPERMAN and 4G wireless cellular systems [2] have multiple-input, multiple-output (MIMO)technology incorporated in them. Combining OFDM with such MIMO technology has resulted in a significant capacity increase in such systems. For coherent demodulation of the signal, we need channel estimation and symbol detection at the receiver of these systems.A number of algorithms, such as the maximum likelihood (ML) and zero forcing (ZF) algorithms [3,4], have been proposed to detect symbols in OFDM. The implementation of the ZF algorithm is not complex and it is not computationally tedious but it fails to perform satisfactorily in fast-fading and time-varying environments. For these environments we therefore use the ML algorithm which performs excellently in these cases. The primary disadvantage of

B.K. Panigrahi et al. (Eds.): SEMCCO 2013, Part I, LNCS 8297, pp. 236–247, 2013.

the ML algorithm is its extremely high computational complexity. By computing the Euclidean distance between the received and actual symbols for all possible combinations of the transmitted symbols ,an exhaustive search of the candidate symbol vector on each subcarrier is made. As the number of transmitter and receiver antennas increase in the constellation, the search space grows exponentially. So the algorithms computational complexity becomes intensive [5]. There is substantial literature which focuses on the area of reducing the complexity and obtaining an optimal solution from the ML algorithm, which detects symbols.Sphere decoders are used for ML detection of signals at the receiver end for multi fading channels. [6] has used orthogonal matrix triangularization (QR decomposition) with sort and Dijkstras algorithm for decreasing the computational complexity of these decoders. The multistage likelihood was presented in [7] to calculate the Euclidean distance of the candidate symbol. In [8], the sphere detector was proposed to have a polynomial computational reduction, but when the search space is large, it takes much more computational time. So research has focussed particularly in reducing the search space and decreasing the computational complexity. So for channel estimation and symbol detection heuristic approaches such as the genetic algorithm (GA) and particle swarm optimization (PSO) are implemented with the ML principle for their ability to reduce the computational burden by efficiently shrinking the search space.In pulse amplitude modulation-based communication systems (PAM) the GA [9] and PSO [10] strategies were used for channel estimation and data detection. A memetic differential evolution (DE) algorithm was proposed for minimum bit error rate detection in multiuser MIMO systems in [13]. DE [11] was also employed by [12] to reduce the computational complexity of symbol detectors in MIMO-OFDM system. In this paper, we employ a recently developed variant of DE called MDE_pBX [13], which introduced a new group-based mutation strategy, novel schemes for the adaptation of control parameters scale factor (F) and crossover rate (Cr), and also an exploitative crossover strategy (p-best crossover). The results of MDE_pBX has been compared to ML, ZF, SaDE [14] and CLPSO [15]. The results clearly indicate that MDE_pBX can obtain high quality results better than ZF, SaDE and CLPSO under various simulation strategies. Moreover, the results obtained by MDE_pBX are very close to the optimal results provided by ML algorithm, but MDE_pBX is computationally much less intensive than ML especially for larger systems.

2 Theory

2.1 MIMO-OFDM System Model

The simplified block diagram of the MIMO-OFDM system is shown in Figure 1. For this system, we consider the N_{tx} transmit, N_{rx} receive antennas, K subcarriers, n OFDM symbols. Considering the modulation type a vector of the information data is mapped onto complex symbols. The transmitted symbol vector is expressed as:

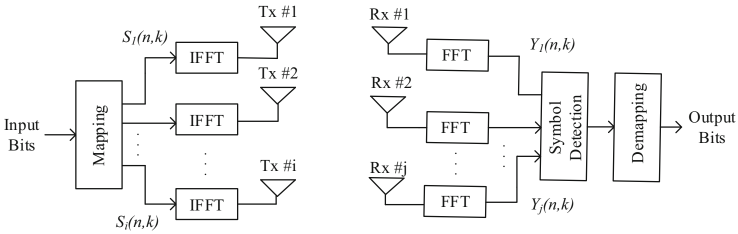

Fig. 1. Basic block diagram of MIMO-OFDM System

$$S[n,k] = [S_1(n,k), ..., S_{N_{tx}}(n,k)]^T, \qquad k = 0, ..., K-1, \tag{1}$$

where $S_i(n,k)$ is the symbol that is transmitted at the n^{th} symbol, k^{th} subcarrier, and i^th antenna, and $[.]^T$ is the transpose operation. By applying inverse fast Fourier transform (IFFT), symbol vectors are turned into the OFDM symbol:

$$s_n[m] = \frac{1}{\sqrt{KN_{tx}}} \sum_{k=0}^{K-1} S[n,k] e^{j2\pi m/k}, \qquad m = 0, ..., K-1 \tag{2}$$

Then we add the cyclic prefix (CP) to avoid inter-symbol interference (ISI). The signal vectors are fed through the i^{th} transmitter antenna. After removing the CP from the received signal vector at the q^{th} receiver antenna, the fast Fourier transform (FFT) is taken as:

$$Y[n,k] = \frac{1}{\sqrt{K}} \sum_{m=0}^{K-1} y[m] e^{-j2\pi m/K}, \qquad n = 0, ..., K-1 \tag{3}$$

Next, the received signal vector can be expressed as:

$$Y_q[n,k] = \sum_{i=1}^{N_{tx}} H_i[n,k] S_i[n,k] + W_q[n,k] \tag{4}$$

where $H_i[n,k]$ is the channel impulse response vector and $W_q[n,k]$ is the additive white Gaussian noise [14].

2.2 Symbol Detection in MIMO-OFDM

The estimations of the data symbols are obtained by maximizing the following metric:

$$S_* = \arg\max P(Y|S) \tag{5}$$

Furthermore,the ML algorithm detects the symbols by minimizing the squared Euclidian distance to target vector Y over the N_{tx} dimensional discrete search set:

$$S_* = \arg\min ||Y - HS||^2 \tag{6}$$

All possible $M^{N_{tx}}$ combinations of the transmitted symbols must be searched for the optimal solution of the ML detection which increases the computational complexity. Hence, we propose heuristic approaches in order to reduce the computational complexity of the symbol detection in the MIMO-OFDM systems.

3 Differential Evolution

DE [11,16] is a simple real-coded evolutionary algorithm. It works through a simple cycle of stages, which are detailed below.

3.1 Parameter Vector Initialization

DE searches for a global optimum point in a D-dimensional continuous hyperspace. It begins with a randomly initiated population of NP D dimensional real-valued parameter vectors. Each vector, also known as chromosome, forms a candidate solution to the multi-dimensional optimization problem. We shall denote subsequent generations in DE by $G = 0, 1, ...G_{max}$. Since the parameter vectors are likely to be changed over different generations, we may adopt the following notation for representing the i^{th} vector of the population at the current generation:

$$\overrightarrow{X}_{i,G} = [x_{1,i,G}, x_{2,i,G}, ...x_{D,i,G}] \tag{7}$$

The initial population (at $G = 0$) should cover the entire search space as much as possible by uniformly randomizing individuals within the search space constrained by the prescribed minimum and maximum bounds: $\overrightarrow{X}_{min} = [x_{1,min}, x_{2,min}, ..., x_{D,min}]$ and $\overrightarrow{X}_{max} = [x_{1,max}, x_{2,max}, ..., x_{D,max}]$. Hence we may initialize the j^{th} component of the i^{th} vector as:

$$x_{j,i,0} = x_{j,min} + rand_{i,j}[0, 1).(x_{j,max} - x_{j,min}) \tag{8}$$

where $rand$ is a uniformly distributed number lying between 0 and 1 and is instantiated independently for each component of the i^th vector.

3.2 Mutation with Difference Vectors

After initialization, DE creates a donor vector V_{in} corresponding to each population member or target vector $X_{i,G}$ in the current generation through mutation. Three most frequently referred mutation strategies implemented in the public-domain DE codes available online at http://www.icsi.berkeley.edu/storn/code.html are listed below:

$$DE/rand/1 \ : \ \overrightarrow{V}_{i,G} = \overrightarrow{X}_{r_1^i,G} + F \cdot (\overrightarrow{X}_{r_2^i,G} - \overrightarrow{X}_{r_3^i,G}) \tag{9}$$

$$DE/best/1 \ : \ \overrightarrow{V}_{i,G} = \overrightarrow{X}_{r_{best}^i,G} + F \cdot (\overrightarrow{X}_{r_1^i,G} - \overrightarrow{X}_{r_2^i,G}) \tag{10}$$

$$DE/target-to-best/1 \;:\; \overrightarrow{V}_{i,G} = \overrightarrow{X}_{i,G} + F\cdot(\overrightarrow{X}_{r^i_{best},G} - \overrightarrow{X}_{i,G} + \overrightarrow{X}_{r^i_1,G} - \overrightarrow{X}_{r^i_2,G}) \quad (11)$$

The indices r^i_1, r^i_2 and r^i_3 are mutually exclusive integers randomly chosen from the range $[1, NP]$, and all are different from the index i. These indices are randomly generated once for each donor vector. The scaling factor F is a positive control parameter for scaling the difference vectors. $\overrightarrow{X}_{r^i_{best},G}$ is the best individual vector with the best fitness in the population at generation G.

3.3 Crossover

To enhance the potential diversity of the population, a crossover operation comes into play after generating the donor vector through mutation. The donor vector exchanges its components with the target vector $\overrightarrow{X}_{i,G}$ under this operation to form the trial vector $\overrightarrow{U}_{i,G} = [u_{1,i,G}, u_{2,i,G}, ...u_{D,i,G}]$. In this article we focus on the widely used binomial crossover that is performed on each of the D variables whenever a randomly generated number between 0 and 1 is less than or equal to the Cr value. The scheme may be outlined as:

$$u_{j,i,G} = \begin{cases} v_{j,i,G} & if \quad rand_{i,j}[0,1) \le Cr or j = j_{rand} \\ x_{j,i,G} & otherwise \end{cases} \quad (12)$$

where, as before, $rand_{i,j}[0,1)$ is a uniformly distributed random number, which is called anew for each j^{th} component of the i^{th} parameter vector. $j_{rand} \epsilon 1, 2, ...D$ is a randomly chosen index, which ensures that $\overrightarrow{U}_{i,G}$ obtains at least one component from $\overrightarrow{V}_{i,G}$.

3.4 Selection

The next step of the algorithm calls for selection to determine whether the target or the trial vector survives to the next generation, i.e., at $G = G+1$. The selection operation is described as:

$$\overrightarrow{X}_{i,G+1} = \begin{cases} \overrightarrow{U}_{i,G} & if \quad f(\overrightarrow{U}_{i,G}) \le f(\overrightarrow{X}_{i,G}) \\ \overrightarrow{X}_{i,G} & if \quad f(\overrightarrow{U}_{i,G}) > f(\overrightarrow{X}_{i,G}) \end{cases} \quad (13)$$

where $f(\overrightarrow{X})$ is the objective function to be minimized. Therefore, if the new trial vector gives an equal or lower value of the objective function, it replaces the corresponding target vector in the next generation; otherwise, the target is retained in the population.

4 The MDE_*p*BX Algorithm

We describe MDE_*p*BX and discuss the various features of the algorithm such as the mutation scheme called DE/current-to-gr_best/1, a *p*-best crossover scheme in this section. We also coin the rules for adapting the control parameters F and Cr in each iteration.

4.1 DE/current-to-gr_best/1

DE/current-to-best/1 is one of the widely used mutation schemes in DE. The useful information of the best solution (with highest objective function value for maximization problems) is incorporated in this algorithm. This results in fast convergence by guiding the evolutionary search towards a specific point in the search space.The algorithm may converge to a locally optimal point in the search space due to its exploitative behavior and lose its global exploration capabilities. To avoid this predicament, in this paper we propose a less greedy and more explorative variant of the DE/current-to-best/1 mutation strategy termed as DE/current-to-gr_best/1.This utilizes the best vector of a dynamic group of $q\%$ of the randomly selected population members for each target vector. Now the population moves towards different points and explores the landscape much better without getting attracted towards a specific point in the search space. The new scheme may be formulated as:

$$\overrightarrow{V}_{i,G} = \overrightarrow{X}_{i,G} + F \cdot (\overrightarrow{X}_{gr_best,G} - \overrightarrow{X}_{i,G} + \overrightarrow{X}_{r_1^i,G} - \overrightarrow{X}_{r_2^i,G}) \tag{14}$$

where $\overrightarrow{X}_{gr_best,G}$ is the best solution of $q\%$ members randomly selected from the present population whereas $\overrightarrow{X}_{r_1^i,G}$ and $\overrightarrow{X}_{r_2^i,G}$ are two randomly selected distinct population vectors. Using the above technique, the target solutions are prevented from being attracted towards the same best position found so far by the entire population. This helps in avoiding premature convergence at local optima.

4.2 The *p*-best Crossover

The crossover operation in MDE_pBX is named p-best crossover where for each donor vector, a vector is randomly chosen from the p top-ranking individuals (in accordance with their objective function values) in the current population and then normal binomial crossover is carried out between the donor vector and the randomly selected $p$best vector to produce the trial vector of same index. Fast convergence is ensured by means of this innovative crossover scheme, where the information contained in the top ranking individuals of the population is incorporated into the trial vector. The parameter p is reduction takes place in a linear fashion in the following way:

$$p = ceil[N_p/2(1 - \frac{G-1}{G_{max}})] \tag{15}$$

where N_p is the population size, G is the current generation number, G_{max} is the maximum number of generations ($G = [1, 2, ..., G_{max}]$) and $ceil(y)$ is the 'ceiling function returning the lowest integer greater than its argument y. p is reduced by a routine which favours exploration at the initial stages of the search and exploitation during the later stages. This is done by gradually reducing the elitist portion of the population, with a randomly selected member from where the component mixing of the donor vector is allowed for generation of the trial vector.

4.3 Parameter Adaptation

The parameter adaptation schemes in MDE_*p*BX are guided by the knowledge of the successful values of F and Cr that were able to generate better offspring (trial vectors) in the last generation.

Scale Factor Adaptation. At every generation, the scale factor F_i of each individual target vector is independently generated as $F_i = Cauchy(F_m, 0.1)$ where $Cauchy(F_m, 0.1)$ is a random number sampled from a Cauchy distribution with location parameter F_m and scale parameter 0.1. The value of F_i is regenerated if $F_i \leq 0$ or $F_i > 1$. Let us denote $F_{success}$ as the set of the successful scale factors, so far, of the current generation generating better trial vectors that are likely to advance to the next generation. Moreover let us say $mean_A(F_{G-1})$ is the simple arithmetic mean of all scale factors associated with population members in generation $G-1$. Location parameter F_m of the Cauchy distribution is initialized to be 0.5 and then updated at the end of each generation in the following manner:

$$F_m = w_F F_m + (1 - w_F) mean_{Pow}(F_{success}) \tag{16}$$

The weight factor though in the original MDE_*p*BX paper was varied randomly, but in our case we have set it as 0.8. $mean_{Pow}$ stands for power mean which is given by:

$$mean_{Pow}(F_{success}) = \sum_{x \epsilon F_{success}} (x^n / |F_{success}|)^(1/n) \tag{17}$$

where $|F_{success}|$ is the cardinality of the set $F_{success}$ and n is taken as 1.5.

Crossover Probability Adaptation. At every generation the crossover probability Cr_i of each individual vector is independently generated as $Cr_i = Gaussian(Cr_m, 0.1)$, where $Gaussian(Cr_m, 0.1)$ is a random number sampled from a Gaussian distribution with mean Cr_m and standard deviation 0.1. Cr_i is truncated if it falls outside the interval $[0, 1]$. Denote $Cr_{success}$ as the set of all successful crossover probabilities Cr_i s at the current generation. The mean of the normal distribution Cr_m is initialized to be 0.6 and then updated at the end of each generation as:

$$Cr_m = w_{Cr} Cr_m + (1 - w_{Cr}) mean_{Pow}(Cr_{success}) \tag{18}$$

with the weight w_{Cr} being kept constant at 0.9. The power mean is calculated as:

$$mean_{Pow}(Cr_{success}) = \sum_{x \epsilon Cr_{success}} (x^n / |Cr_{success}|)^(1/n) \tag{19}$$

where $|Cr_{success}|$ is the cardinality of the set $Cr_{success}$ and n is taken as 1.5.

5 Experiments and Results

For our experimentation we have considered three OFDM systems with 2×4, 4×4 and 8×8 transmitter and receiver antennas. We have compared the performance of MDE_pBX with two classical symbol detectors namely ML and ZF and two other renowned heuristic algorithms SaDE and CLPSO. The simulation parameters of the MIMO-OFDM system are presented in Table 1. The population size was kept as 25 for all the participating evolutionary algorithms. The parameter q in the mutation scheme DE/current-to-gr_best/1 of MDE_pBX is kept as 1/4th of the population size. The reason for setting such a value for the group size q is that if q is on par with population size, the probability that the best of randomly chosen q% vectors is similar to the globally best vector of the entire population will be high and the proposed mutation scheme DE/current-to-gr_best/1 basically becomes identical to the DE/current-to-best/1 scheme. This drives most of the vectors towards a specific point in the search space resulting in premature convergence. The parameter p in p-best crossover is linearly decreased with generations. For the contestant heuristic algorithms, we follow the parameter settings in the original paper of SaDE and CLPSO. The convergence analysis of MDE_pBX, SaDE and CLPSO are done in Fig. 2 where the number of iterations needed by these algorithms for converging to the optimal solution (obtained by ML) are shown for the three OFDM systems. were found out by averaging over 25 runs. As the same algorithm converges to the optimal solution at a different iteration in each simulation so the data presented in Fig. 2 are averaged over 25 runs. It is evident from Fig. 2 that MDE_pBX requires less number of iterations than SaDE and CLPSO for converging to the optimal solution for all the three cases.

Table 1. MIMO-OFDM simulation parameters

Parameters	Values
Number of subcarriers	128
Cyclic prefix size	FFT/4=32
Modulation type	8PSK
Channel type	Rayleigh fading

The BER performance of the symbol detectors for the 2×4 system are shown in Fig. 3. The figure depicts that MDE_pBX outperforms SaDE, CLPSO and ZF and is very close to the optimal solution i.e. the BER provided by ML. For example, at 25 dB SNR the Bit Error Rates for ML, MDE_pBX, SaDE, CLPSO and ZF are 2.15×10^{-4} 4×10^{-4}, 6×10^{-4}, 10^{-3} and 0.05 respectively.

Fig. 4 and Fig. 5 shows the BER performance of the symbol detectors for 4×4 and 8×8 systems respectively. These two figures show the effect of the increasing number of receiver and transmitter antennas on the detection performance. For instance, for the 4×4 system at 15 dB SNR the BERs for ML, MDE_pBX, SaDE, CLPSO and ZF are 5.95×10^{-3}, $8 \times 10^{(} - 3)$, $9 \times 10^{(} - 3)$, $1.3 \times 10^{(} - 2)$ and

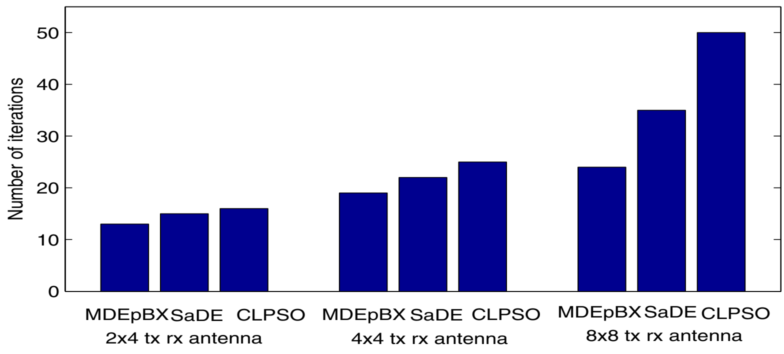

Fig. 2. Convergence analysis of MDE_*p*BX, SaDE and CLPSO

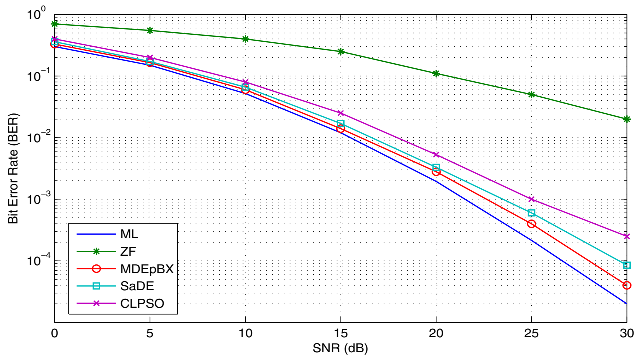

Fig. 3. The BER versus the SNR of the detectors for 2x4 system

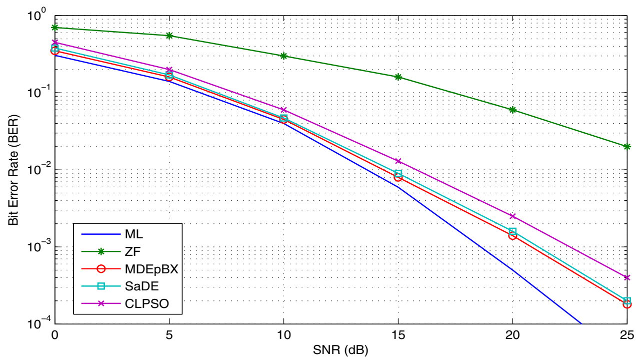

Fig. 4. The BER versus the SNR of the detectors for 4x4 system

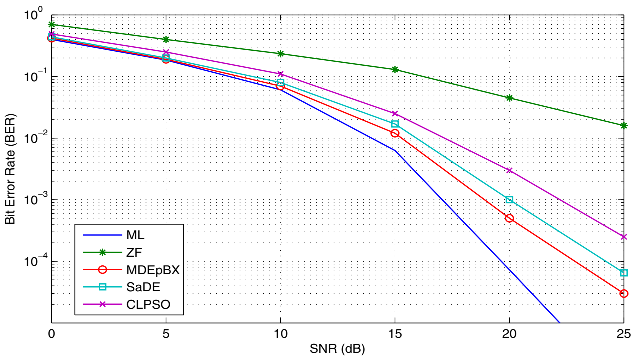

Fig. 5. The BER versus the SNR of the detectors for 8x8 system

0.16 respectively. Again, for the 8×8 system at 15 dB SNR the BERs for ML, MDE_pBX, SaDE, CLPSO and ZF are $6.3 \times 10^(-3)$, $1.2 \times 10^(-2)$, $1.7 \times 10^(-2)$, $2.5 \times 10^(- 2)$ and 0.13 respectively.

6 Computational Complexity

To show that the heuristic algorithms especially MDE_pBX are computationally advantageous than the classical symbol detectors we have done computational complexity analysis. The computational complexity of the symbol detectors have been represented in terms of N_{rx} (number of receiver antenna), N_{tx} (number of transmitter antenna), N_p (Population size), N_{itr} (Number of iterations) and M (Constellation size). The number of operations required for ZF and ML algorithms are $4N_{tx}^3 + 2N_{tx}^2 N_{rx}$ and $N_{rx}(N_{tx} + 1)M^{N_{tx}}$ respectively. On the other hand the heuristic algorithms require $N_p(N_{tx}N_{rx} + \mu)N_{itr}$ operations, where μ is the number of population updating parameters. μ depends on the algorithm and is almost same for the heuristic approaches. But, the computational complexity does not directly depend on μ. It directly depends on the number of iterations that provide convergence in the algorithms. As for all the three systems, MDE_pBX requires the least number of iterations among the heuristic approaches so it has the lowest computational complexity. Moreover, it can be also seen from the above analysis that the computational complexity of the ML algorithm is very high when the number of transmitter antenna, receiver antenna and the constellation size increases. For this reason, the ML algorithm is not a practical solution for symbol detection in MIMO-OFDM systems that have large antenna and constellation sizes. However, the proposed MDE_pBX based detector has significantly less computational complexity than the other algorithms.

7 Conclusion

In this article, we utilized a DE variant, MDE_pBX, for reducing the search space of the ML detector and the computational complexity of symbol detection in MIMO-OFDM systems. We have compared the performance of MDE_pBX to two classical (ML and ZF) and two evolutionary algorithms (SaDE and CLPSO). Our results show that ML has the best performance followed by MDE_pBX, SaDE, CLPSO and ZF. However, though ML produces best results it is computationally very inefficient. Moreover, among the evolutionary algorithms MDE_pBX takes least number of iterations to reach the optimal solution i.e. the solution produced by ML. Thus, we can conclude that MDE_pBX can be a very good trade off between results and computational complexity.

References

1. Cimini, L.J.: Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing. IEEE Transactions on Communications 33(7), 665–675 (1985)
2. van Zelst, V., Schenk, T.C.W.: Implementation of a mimo ofdm-based wireless lan system. IEEE Transactions on Signal Processing 52(2), 483–494 (2004)
3. Kiessling, M., Speidel, J.: Analytical performance of mimo zero-forcing receivers in correlated rayleigh fading environments. In: 4th IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC 2003, pp. 383–387 (2003)
4. Spencer, Q., Swindlehurst, A., Haardt, M.: Zero-forcing methods for downlink spatial multiplexing in multiuser mimo channels. IEEE Transactions on Signal Processing 52(2), 461–471 (2004)
5. Jiang, M., Akhtman, J., Guo, F.C., Hanzo, L.: Iterative joint channel estimation and symbol detection for multi-user mimo ofdm. In: IEEE VTC 2006, May 7-10, vol. 1, pp. 221–225 (Spring 2006)
6. Fukatani, T., Matsumoto, R., Uyematsu, T.: Two methods for decreasing the computational complexity of the mimo ml decoder. CoRR cs.IT/0410040 (2004)
7. Xie, Y., Li, Q., Georghiades, C.: On some near optimal low complexity detectors for mimo fading channels. IEEE Transactions on Wireless Communications 6(4), 1182–1186 (2007)
8. Hassibi, B., Vikalo, H.: On the sphere-decoding algorithm i. expected complexity. IEEE Transactions on Signal Processing 53(8), 2806–2818 (2005)
9. Chen, S., Wu, Y.: Maximum likelihood joint channel and data estimation using genetic algorithms. IEEE Transactions on Signal Processing 46(5), 1469–1473 (1998)
10. Zubair, M., Choudhry, M.A.S., Malik, A.N., Qureshi, I.M.: Joint channel and data estimation using particle swarm optimization. IEICE Transactions 91-B(9), 3033–3036 (2008)
11. Storn, R., Price, K.: Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11(4), 341–359 (1997)
12. Seyman, M.N., Taspinar, N.: Symbol detection using the differential evolution algorithm in mimo-ofdm systems. Turkish Journal of Electrical Engineering & Computer Sciences (2013)

13. Islam, S., Das, S., Ghosh, S., Roy, S., Suganthan, P.: An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 42(2), 482–500 (2012)
14. Qin, A.K., Huang, V.L., Suganthan, P.: Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Transactions on Evolutionary Computation 13(2), 398–417 (2009)
15. Liang, J.J., Qin, A.K., Suganthan, P., Baskar, S.: Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. IEEE Transactions on Evolutionary Computation 10(3), 281–295 (2006)
16. Yang, X.S., Deb, S.: Two-stage eagle strategy with differential evolution. International Journal of Bio-inspired Computation 4(1), 1–5 (2012)