Şimşir *J Wireless Com Network* (2024) 2024:80
https://doi.org/10.1186/s13638-024-02410-3

EURASIP Journal on Wireless
Communications and Networking

RESEARCH **Open Access**

An advanced symbol detection approach for MIMO-FBMC/OQAM scheme based on migrating birds optimization algorithm

Şakir Şimşir[1*]

*Correspondence:
sakirsimsir@nevsehir.edu.tr

[1] Department of Electrical and Electronics Engineering, Nevşehir Hacı Bektaş Veli University, Nevşehir, Türkiye

Abstract

Application of multiple-input multiple-output (MIMO) technology to the filter bank multicarrier/offset quadrate amplitude modulation (FBMC/OQAM) system provides great improvements on its robustness to channel fading effects. Nevertheless, the unique qualities of MIMO-FBMC/OQAM scheme do not make any sense without solving the symbol detection problem on its receiver part. To this end, an efficient symbol detecting strategy having the capability of recovering the transmitted symbol sequences with the highest accuracy is needed for the related system. Maximum likelihood (ML) detector is known with its excellent symbol detection performance in the literature. However, due to the usage of exhaustive search procedure, the computational complexity of the ML detector reaches extremely high levels with the increase of antenna size and modulation order. On the other hand, in the case that the exhaustive search procedure is substituted with the symbol optimization process, it becomes possible to achieve a large amount of complexity reduction in the conventional ML scheme without compromising too much on its symbol detection performance. In order to carry out an efficient symbol optimization in discrete space, we propose migrating birds optimization (MBO) algorithm based on cyclic bit flipping procedure in this paper. By virtue of employing the proposed MBO algorithm reinforced by the cyclic bit flipping mechanism for optimizing the symbol sequences, the computational complexity of the conventional ML is reduced to quite low levels. The percentages of complexity reduction achieved by the proposed scheme over the classical ML for 4×4, 6×6 and 8×8 MIMO configurations are equal to 29.688%, 87.188% and 98.299%, respectively. In addition to these huge complexity gains, an efficient symbol detection performance quite near to that of conventional ML is obtained thanks to the usage of aforementioned MBO-based symbol optimization procedure.

Keywords: FBMC, MIMO, Zero forcing, Maximum likelihood, Migrating birds optimization

1 Introduction

In the last three decades, we have witnessed a tremendous improvement in the wireless communication technology. While the first-generation (1G) systems could support only the voice communication in analog form, now it has become possible with the fourth-generation (4G) wireless technology to benefit from numerous mobile internet-based services such as health care, automated highways, smart homes, disaster management, video conferencing, etc. Nevertheless, supporting the aforementioned types of services and applications will never be sufficient in future due to the rapidly increasing new requirements expected to be met by the fifth generation (5G) and beyond telecommunication technologies. It is expected from the future communication systems to provide inter-connectivity among not only the people, but also the things and machines with the entrance of technologies like internet of things (IoT) and machine-type communication (MTC) to our lives. Herewith, the primary needs for the future wireless systems will be the higher data rates, better quality of service, lower latency and higher mobility [1–4].

In the 4G long-term evolution (LTE) wireless standard, orthogonal frequency division multiplexing (OFDM) is employed as the main waveform [5, 6]. However, due to its rectangular pulse shape, OFDM suffers from the problem of high spectral leakage, which leads to spectral inefficiency. Besides this, the usage of cyclic prefix in the OFDM signals causes a considerable reduction in the data transmission rate. Synchronization requirement to maintain the orthogonality is another major deficiency of the OFDM waveform [5, 6]. All these shortcomings of the OFDM waveform have pushed the researchers to develop alternative waveforms capable of meeting the new requirements of the future communication systems. One of the primary waveforms developed for eliminating the aforementioned drawbacks of the OFDM waveform is the filter bank multicarrier/offset quadrate amplitude modulation (FBMC/OQAM). It has various superiorities over the conventional OFDM, which makes it suitable for the new generation wireless technologies. In the FBMC/OQAM system, the orthogonality doesn't need to be maintained among the subcarriers. Thanks to this advantage, it becomes possible to minimize the power loss. Apart from this, the spectral leakage can be substantially mitigated in the FBMC/OQAM system by performing a nonrectangular pulse shaping process. Besides, filtering operation that provides the subcarrier isolation makes the FBMC/OQAM more robust to synchronization errors compared to the OFDM system. Furthermore, unlike the OFDM, FBMC/OQAM system doesn't need to perform the process of cyclic prefix addition, which causes a certain decrement in the data transmission rate [7–10].

In case of combining all these above-mentioned superior features with the benefits of multiple-input multiple-output (MIMO) technology by performing the data transmission over the multiple antennas, the capability of FBMC/OQAM system reaches a much higher level [11–14]. The usage of multiantenna structure to send and receive the transmission signals has great benefits on the system performance. One of the important benefits provided by the related multiantenna usage is that the fading effects of the multipath channels are significantly alleviated. As a result of this, the data transmission is carried out at lower bit error rates (BERs). Another important benefit of MIMO technology is the capacity increase. The increase of antennas brings about an additional system capacity. In consequence of this, the communication at higher data speeds becomes possible. By the way, in this study, the spatial multiplexing MIMO model, in which the multiple data streams are transmitted in parallel within the same frequency band by using a different transmit antenna for each data stream, is utilized in the proposed system. On the other hand, symbol detection is a crucial operation that needs to be performed in any of the transmission technologies. Without the implementation of symbol detection, coherent recovery of the transmitted symbols will be impossible in the MIMO-FBMC/OQAM scheme. In this case, all its superior features will not make any sense. Maximum likelihood (ML) and zero forcing (ZF) are the most well-known and frequently used strategies developed to fulfill the operation of symbol detection. It is expected from a symbol detector to show a reasonable symbol detection performance without leading to an excessive complexity increment in the transmission system. To put it more clearly, high performance or low computational complexity alone is not sufficient for an ideal symbol detector. Both the performance and complexity criteria must be satisfied, simultaneously. Neither ZF nor ML provides these two criteria at the same time. Along with some advantages, they also have various disadvantages. For instance, while possessing a low computational complexity and easy to apply nature stand out as the positive aspects of the ZF symbol detector, having a poor performance in relatively tough channel conditions puts it at a disadvantageous position. Apart from this, the performance of ZF symbol detector gets worse even more with the increase in the number of antennas [15]. ML is the only symbol detector that is capable of reaching the optimal symbol sequences. In other words, ML is a method that guarantees to achieve the absolute best results. For this reason, its symbol detection performance is generally considered as the upper bound for the symbol detectors developed in the literature. However, the flawless performance of the ML detector comes with an extremely high computational cost. Exhaustive search procedure is the main reason why the ML strategy is too complex to be practically employed in any transmission scheme. In the related searching procedure, when recovering the transmitted symbol vector from the received one that is subjected to a certain amount of degradation by the fading channel, the received symbol sequence is compared with each of the symbol combinations, which have the possibility of being transmitted to the receiver, by using the Euclidean distance. Note that the length of symbol vectors is equal to the number of antennas and the number of possible symbol combinations becomes greater with the increase of antenna number and modulation order. After calculating the Euclidean distance of each possible symbol combination to the symbol vector taken by the receiver, the symbol combination with minimal Euclidean distance is chosen as the optimum symbol vector likely to be sent [16, 17]. Calculation of Euclidean distance for each possible symbol combination causes the computational complexity of ML to be very high. Furthermore, since the number of possible symbol combinations corresponding to search space is determined by the number of antennas and modulation order, the increase in the values of related parameters will expand the search space, exponentially and depending on this, the computational cost of achieving the optimum symbol sequence via the ML scheme will reach extreme levels due to its exhaustive search strategy.

On the other hand, by disabling the strategy of exhaustive search and subjecting the symbol vectors to the optimization process as a replacement, we can reduce the ML complexity in a great proportion with a little bit deterioration in its symbol detection performance. By doing so, rather than trying each of the possible symbol combinations to find the optimum one that provides minimum Euclidean distance to the signal reaching the receiver, an optimization algorithm is utilized to achieve a near-optimal solution with much less trials. In order to fulfill the symbol optimization task in the ML detector, meta-heuristic algorithms will be the best choice due to their capability of reaching better solutions with less computational cost, which is the main reason of why they are widely used in many fields. In our symbol detection problem, it is feasible to perform an optimization on the symbol sequences, each of which consists of the combination of complex quadrate amplitude modulation (QAM) symbols, simply in discrete space. For that purpose, every QAM symbol represented by a unique complex number is converted to its equivalent binary number. Thereby, each symbol vector turns into a binary bit sequence to be optimized in discrete search space. To be able to optimize the resulted binary bit sequences, we need the efficient discrete variants of the meta-heuristic algorithms. By considering this requirement, we propose cyclic bit flipping-based migrating birds optimization (MBO) algorithm [18], which has an efficient searching capability in binary search space due to its cyclic bit flipping mechanism. In this paper, after integrating the related MBO variant to the ML detector as a symbol optimizer, a novel symbol detection strategy called MBO-ML was developed to resolve the symbol detecting issue of the MIMO-FBMC/OQAM scheme. Then, we evaluated our newly developed MBO-ML strategy with regard to both symbol detection performance and computational complexity by comparing it to not only the conventional symbol detection methods like ML and ZF, but also three different modern ML schemes based on meta-heuristic approaches such as discrete binary harmony search [19]-ML (DBHS-ML), discrete artificial bee colony [20]-ML (disABC-ML) and binary particle swarm optimization [21]-ML (BPSO-ML). Exhibiting a near-optimal symbol detection performance and achieving this with a very low computational cost puts the suggested MBO-ML scheme in a superior position than other methods considered in this paper.

5G wireless technology is separated into three categories named as ultra-reliable low-latency communications (URLLC), massive machine-type communications (mMTC) and enhanced mobile broadband (eMBB). Each category has its own requirements. For example, in the URLLC applications, very high reliability and stringent latency are needed while the mMTC and eMBB applications require massive connectivity and high data rates, respectively [22]. In the aforementioned 5G categories, URLLC is the most suitable one for the proposed MBO-ML strategy. In order to provide high reliability in any transmission system, it is necessary to employ a high-performance symbol detector at the receiver side to accurately detect the symbol vectors with minimum error. Due to its near-optimal symbol detection performance, the MBO-ML can easily meet this requirement. Apart from this, it is possible to enable the low latency by the proposed scheme since it can achieve this high symbol detection performance with minimal computational cost. Note that the low latency can only be obtained with low computational complexity. Herewith, the MBO-ML strategy is feasible for the URLLC applications needing both high reliability and low latency such as autonomous driving, healthcare and industrial automation.

To the best of our knowledge, there is no study in which the MBO algorithm is exploited for the detection of symbol sequences in MIMO-FBMC/OQAM or any of the other transmission schemes. Nevertheless, it is possible to find some studies in the literature based on symbol detection by using meta-heuristic algorithms in various transmission procedures [23–28]. In [23], ant colony optimization (ACO) and PSO algorithms were hybridized to develop a low-complexity symbol detector for the large-MIMO system. In this hybrid algorithm, a new probability metric was designed by utilizing the concept of velocity metric from the PSO and distance metric from the ACO. In [24], the back-tracking search algorithm (BSA)-based ML detector was integrated to the receiver of non-orthogonal multiple access combined with MIMO (MIMO-NOMA) system. After that, while the symbol detection capability of the suggested strategy was measured through the BER graphs, its complexity improvement over the classical ML was put forth via the computational complexity analysis. In [25], ABC algorithm was utilized to seek for the optimum combination of QAM symbols at the receiver of massive MIMO transmission scheme. In the corresponding study, the authors used the linear detection results as the initial solution vectors of the ABC algorithm. In [26], minimum bit error rate (MBER) detector, in which the weight vectors are optimized by the PSO algorithm, was combined with Bell Laboratories layered space time (BLAST) to obtain a new detection algorithm named MBER-BLAST for the space division multiple access merged with OFDM (OFDM-SDMA) scheme. In [27], an improved ML detector based on differential evolution (DE) algorithm was developed to detect the symbol vectors in the MIMO-OFDM system. In [28], several symbol detection algorithms assisted by the continuous and binary PSO variants were proposed for MIMO communication systems. A thorough examination of the literature will reveal that there is no work that uses MBO algorithm for symbol detection in any transmission scheme. On the other hand, there exist many papers that include the successful applications of the MBO algorithm to various problems in different fields such as job scheduling, routing, assembly line balancing, exudate classification, land distribution, etc. [29–36].

The list of major contributions provided by this paper are given below:

(1) By integrating the cyclic bit flipping-based MBO algorithm to the conventional ML detector, a novel symbol detecting procedure called MBO-ML, which has the capability of satisfying the high reliability and low latency requirements of the URLLC applications has been created.

(2) This study is the first in the literature in point of using the MBO algorithm to resolve the symbol detecting issue in the telecommunication field.

(3) Complexity of the classical ML has been significantly reduced by using MBO-assisted symbol optimizing approach as a substitute for its exhaustive search procedure. We have reached 29.688%, 87.188% and 98.299% complexity reduction rates in the conventional ML for 4×4, 6×6 and 8×8 antenna structures, respectively.

(4) The suggested strategy named MBO-ML provides better BER achievement in comparison to both the ZF symbol detector and its advanced competitors called disABC-ML, DBHS-ML and BPSO-ML. To give an example, for 4×4 antenna configuration, the BER improvements achieved by the MBO-ML over the DBHS-ML, disABC-ML, BPSO-ML and ZF at 10 dB signal-to-noise ratio (SNR) are equal to 0.51×10^{-2}, 1.55×10^{-2}, 2.66×10^{-2} and 7.27×10^{-2}, respectively.

(5) MBO-ML strategy accomplishes considerable complexity improvements over the disABC-ML, DBHS-ML and BPSO-ML schemes. For instance, it manages to attain 67.9545%, 63.8636% and 74.7273% complexity gains in these considered symbol detection schemes, respectively, for 8×8 MIMO structure.

Layout of the paper is as follows: In Sect. 2, the system description is carried out for the MIMO-FBMC/OQAM, the problem of symbol detecting in the related system is defined via the mathematical expressions, and our proposed MBO-ML strategy is explained in a quite comprehensive manner, respectively. Section 3 includes the results and discussion together with a detailed complexity analysis. Finally, the paper is completed by giving the conclusions in Sect. 4.

2 Methods

2.1 Description of MIMO-FBMC/OQAM system

Even though the FBMC/OQAM system can successfully cope with various problems encountered in wireless communication by itself, it is possible to provide further improvements on the related system by supporting it with the MIMO technology, through which not only the data transmission can be performed at greater speeds, but also the channel fading effects are minimized. In this section, a simple explanation of how the data transmission is carried out over the multiantenna structure of MIMO-FBMC/OQAM scheme is provided [11–13]. It is important to note that the proposed system is intended for a MIMO model of spatial multiplexing.

If a single antenna transmission is carried out for the symbol $a_{m,n}$ with real value, the expression of the demodulated signal on the receiving end of the FBMC/OQAM system becomes as follows:

$$y_{\mathrm{m,n}} \approx h_{\mathrm{m,n}} \cdot \left(a_{\mathrm{m,n}} + j \cdot u_{\mathrm{m,n}}\right) + n_{\mathrm{m,n}} \tag{1}$$

In Eq. (1), while $h_{\mathrm{m,n}}$ and $u_{\mathrm{m,n}}$ signify the channel coefficients and intrinsic interference, respectively, $n_{\mathrm{m,n}}$ corresponds to the noise component. m is the subcarrier index and n is the time index. When it comes to the MIMO transmission in the FBMC/OQAM system by using N_t transmit antennas and N_r receive antennas, the expression of the demodulated signal takes the following form:

$$y_{\mathrm{m,n}}^{(j)} = \sum_{i=1}^{N_t} h_{\mathrm{m,n}}^{(ji)} \cdot \left(a_{\mathrm{m,n}}^{(i)} + j \cdot u_{\mathrm{m,n}}^{(i)}\right) + n_{\mathrm{m,n}}^{(j)} \tag{2}$$

where i is the index of transmit antenna and j is the index of receive antenna. For instance, while $h_{\mathrm{m,n}}^{(ji)}$ represents the channel coefficient affecting the symbols transmitted from the ith transmit antenna to the jth receive antenna, $y_{\mathrm{m,n}}^{(j)}$ indicates the demodulated signal at the jth receive antenna. It is possible to express Eq. (2) in the form of matrix multiplication as shown below:

$$\begin{bmatrix} y_{\mathrm{m,n}}^{(1)} \\ \vdots \\ y_{\mathrm{m,n}}^{(N_r)} \end{bmatrix} = \begin{bmatrix} h_{\mathrm{m,n}}^{(11)} & \cdots & h_{\mathrm{m,n}}^{(1N_t)} \\ \vdots & \ddots & \vdots \\ h_{\mathrm{m,n}}^{(N_r1)} & \cdots & h_{\mathrm{m,n}}^{(N_rN_t)} \end{bmatrix} \begin{bmatrix} a_{\mathrm{m,n}}^{(1)} + j \cdot u_{\mathrm{m,n}}^{(1)} \\ \vdots \\ a_{\mathrm{m,n}}^{(N_t)} + j \cdot u_{\mathrm{m,n}}^{(N_t)} \end{bmatrix} + \begin{bmatrix} n_{\mathrm{m,n}}^{(1)} \\ \vdots \\ n_{\mathrm{m,n}}^{(N_r)} \end{bmatrix} \tag{3}$$

$$\mathbf{y}_{\mathrm{m,n}} = \mathbf{H}_{\mathrm{m,n}} \cdot \left(\mathbf{a}_{\mathrm{m,n}} + j \cdot \mathbf{u}_{\mathrm{m,n}}\right) + \mathbf{n}_{\mathrm{m,n}} \tag{4}$$

where the demodulated signals, channel coefficients, real-valued symbols with intrinsic interferences and noise components are represented by $N_r \times 1$, $N_r \times N_t$, $N_t \times 1$ and $N_r \times 1$ matrixes, respectively.

2.2 Matrix representation for the FBMC/OQAM scheme

For the purpose of simplification, matrix representation can be utilized for the FBMC/OQAM system as in [10, 14]. In the related matrix-based representation, the prototype filter is represented by a transmit matrix $\mathbf{G} \in \mathbb{C}^{D \times MN}$, which is defined in the following way:

$$\mathbf{G} = [\mathbf{g}_{1,1} \;\; \mathbf{g}_{2,1} \;\; . \;\; . \;\; . \;\; \mathbf{g}_{\mathrm{M},1} \;\; \mathbf{g}_{1,2} \;\; . \;\; . \;\; . \;\; \mathbf{g}_{\mathrm{M,N}}] \tag{5}$$

where $\mathbf{g}_{m,n} \in \mathbb{C}^{\mathrm{D} \times 1}$ are the transmit vectors each of which includes D time samples. In the equation given above, while N denotes the number of symbols, M signifies the subcarrier number. Aside from this, the real-valued symbols to be transmitted can be defined as follows:

$$\mathbf{a} = \mathrm{vec}\left\{ \begin{bmatrix} a_{1,1} & \cdots & a_{1,N} \\ \vdots & \ddots & \vdots \\ a_{M,1} & \cdots & a_{M,N} \end{bmatrix} \right\} = [a_{1,1} \;\; a_{2,1} \;\; . \;\; . \;\; . \;\; a_{M,1} \;\; a_{1,2} \;\; . \;\; . \;\; . \;\; a_{M,N}]^{\mathrm{T}} \tag{6}$$

where vec{} transforms the matrix in its parenthesis into a vector. In the equation above, $\mathbf{a} \in \mathbb{C}^{MN \times 1}$ represents the symbol vector, which is multiplied by the transmit matrix defined in Eq. (5) to obtain the transmission signal $\mathbf{s} \in \mathbb{C}^{D \times 1}$ in the following manner:

$$\mathbf{s} = \mathbf{Ga} \tag{7}$$

In case of modeling the multipath channel as a convolution matrix symbolized by $\mathbf{H} \in \mathbb{C}^{D \times D}$, the signal that arrives to the input of receiving end is formulated as shown below:

$$\mathbf{r} = \mathbf{Hs} + \mathbf{n} = \mathbf{HGa} + \mathbf{n} \tag{8}$$

where the aforementioned arriving signal is denoted by $\mathbf{r} \in \mathbb{C}^{D \times 1}$. Apart from this, white Gaussian noise added to the received signal is represented by $\mathbf{n} \approx CN(0,\ P_n\,\mathbf{I}_D)$ where P_n is the noise power and $\mathbf{I}_D$ is the $D \times D$ identity matrix. The notation $CN(0,\ P_n\,\mathbf{I}_D)$ expresses the complex Gaussian distribution with zero mean and $P_n\,\mathbf{I}_D$ variance. Herewith, the received symbols are attained by using the following formulation:

$$\mathbf{y} = \mathbf{G}^{\mathrm{H}}\mathbf{r} = \mathbf{G}^{\mathrm{H}}\mathbf{HGa} + \mathbf{G}^{\mathrm{H}}\mathbf{n} \tag{9}$$

2.3 FBMC/OQAM with block frequency spreading approach

Unlike the conventional OFDM, MIMO methods cannot be directly applied to the FBMC/OQAM system due to its orthogonality constraint which leads to the intrinsic interference. On the other hand, it has become possible with the block frequency spreading approach proposed in [14] to restore the complex orthogonality and make the FBMC/OQAM system suitable for the direct application of all MIMO methods that are compatible with the classical OFDM system. In our study, for the purpose of making the

MIMO-FBMC/OQAM system suitable for the straightforward implementation of the MIMO detection methods, FBMC/OQAM scheme was combined with the block frequency spreading procedure as follows [14]:

In the block frequency spreading approach, a precoding matrix $\mathbf{C} \in \mathbb{C}^{MN \times \frac{MN}{2}}$ is utilized at the transmitter side for the spreading operation in which the QAM symbol sequence $\mathbf{x} \in \mathbb{C}^{\frac{MN}{2} \times 1}$ is multiplied by the related precoding matrix to acquire the real-valued symbol sequence $\mathbf{a} \in \mathbb{C}^{MN \times 1}$ as follows:

$$\mathbf{a} = \mathbf{C}\mathbf{x} \tag{10}$$

After multiplying the received symbols by the $\mathbf{C}^{\mathrm{H}}$, the de-spread symbols are obtained at the receiver side in the following way:

$$\hat{\mathbf{y}} = \mathbf{C}^{\mathrm{H}}\mathbf{y} \tag{11}$$

The expanded version of Eq. (11) will become as follows:

$$\hat{\mathbf{y}} = \mathbf{C}^{\mathrm{H}}\mathbf{G}^{\mathrm{H}}\mathbf{H}\mathbf{G}\mathbf{C}\mathbf{x} + \mathbf{C}^{\mathrm{H}}\mathbf{G}^{\mathrm{H}}\mathbf{n} \tag{12}$$

2.4 Formulizing the symbol detecting problem

When the data transmission is carried out via the MIMO antenna structure with the N_t transmit and N_r receive antennas in the block frequency spreading-based FBMC/OQAM transceiver, each component in the QAM symbol sequence $\mathbf{x} \in \mathbb{C}^{\frac{MN}{2} \times 1}$ to be transmitted will turn into an $N_t \times 1$ sized symbol vector that is defined as $x_{m,n} = \left[x_{m,n}^{(1)}\ ,\ x_{m,n}^{(2)}\ ,\ \ldots\ ,\ x_{m,n}^{(N_t)}\right]^{\mathrm{T}}$, where the subscript m specifies the subcarrier index and subscript n indicates the symbol index. In a similar way, the symbol vector received by the N_r receive antennas will be expressed as $\hat{y}_{m,n} = \left[\hat{y}_{m,n}^{(1)}\ ,\ \hat{y}_{m,n}^{(2)}\ ,\ \ldots\ ,\ \hat{y}_{m,n}^{(N_r)}\right]^{\mathrm{T}}$ in the related system. As it is obvious from the transmitted and received symbol vectors, the numbers of transmit and receive antennas denoted by N_t and N_r determine the numbers of transmitted and received symbols, respectively. To put it another way, while the transmitted symbol vector $x_{m,n}$ consists of N_t QAM symbols, the received symbol vector symbolized by $\hat{y}_{m,n}$ includes N_r number of symbols. If the parameter Z specifies modulation order of QAM, every single QAM symbol included in the symbol sequence $x_{m,n}$ has the possibility of taking Z distinct values. Since the vector $x_{m,n}$ contains N_t QAM symbols and each one can take Z different values, the total count of symbol combinations that can occur, which corresponds to the number of alternative symbol vectors likely to be transmitted, will be equal to Z^{N_t}. In the process of symbol detection based on ML scheme, the symbol combination that has the maximum likelihood to be transmitted among the Z^{N_t} alternatives is tried to find via an exhaustive search procedure. For this purpose, the following equation is utilized to evaluate each alternative:

$$x_{m,n}^{*} = \arg\ \min_{x_{m,n}} \left\{ \left\| \hat{y}_{m,n} - H_{m,n} \cdot x_{m,n} \right\|^{2} \right\} \tag{13}$$

By using Eq. (13), the optimum symbol vector $x_{m,n}^{*}$, which makes the Euclidean distance in the arg min { } operator minimal, is found by trying Z^{N_t} alternative $x_{m,n}$

combinations, each of which has the possibility of being transmitted. For instance, the multiplication of optimal symbol combination $x^{*}_{m,n}$ with the corresponding $N_r \times N_t$ – sized channel coefficient matrix $H_{m,n}$ will have the minimal Euclidean distance to the $\hat{y}_{m,n}$ symbolizing the received signal, which means that the symbol combination owned by the vector $x^{*}_{m,n}$ has the maximal likelihood to have been sent among the Z^{N_t} alternatives. Due to the aforementioned exhaustive search procedure in which each of the Z^{N_t} symbol combinations is tested to find the optimal one providing the minimum Euclidean distance value, an exponential growth will be seen in the ML complexity in case of expanding the search space by increasing the number of transmit antennas N_t or modulation order Z. On the other hand, it is possible to obtain near ML performance by providing a substantial reduction in its computational complexity. For the achievement of this goal, the only thing that needs to be done is to optimize the N_t-length $x_{m,n}$ symbol sequences by using a powerful optimizer as a replacement for the impractical and highly complex exhaustive search procedure. In this study, after the symbol detection problem was transformed into a binary optimization problem, cyclic bit flipping-based MBO algorithm has been incorporated into the traditional ML detector to optimize the symbol sequences. Therefore, with the integration of MBO algorithm, ML detector has gained the ability of reaching near-optimal performance with largely reduced processing load.

In Eq. (13) that formulates the ML symbol detection procedure, it is accepted that the perfect channel coefficients denoted by the matrix $H_{m,n}$ are known in advance by the receiver. On the other hand, in real applications, the actual channel coefficients cannot be estimated, perfectly at the receiver without any estimation error. So, it is impossible for the receivers to have the knowledge of perfect channel coefficients in practical systems as assumed in Eq. (13). Considering this situation, when modeling the MIMO-FBMC/OQAM system, the probability of incorrect channel estimation has been taken into account in this paper. For this reason, the symbol detection process based on ML strategy is reformulated as follows:

$$x^{*}_{m,n} = \arg \min_{x_{m,n}} \left\{ \left\| \hat{y}_{m,n} - \hat{H}_{m,n} \cdot x_{m,n} \right\|^{2} \right\} \tag{14}$$

where $\hat{H}_{m,n}$ represents the inaccurate channel coefficients estimated at the receiver. It is possible to express the related imperfectly estimated channel coefficients as follows [37]:

$$\hat{H}_{m,n} = H_{m,n} + e \cdot \theta \tag{15}$$

where θ denotes the complex Gaussian variable with unit variance and zero mean, while e specifies the rate of estimation error. The multiplication of e by θ gives the estimation error, which is added to the real channel coefficients $H_{m,n}$ to obtain the imperfect channel coefficient matrix $\hat{H}_{m,n}$. The symbol detection performance of any strategy depends on how accurate the channel coefficients are estimated at the receiver side. So, the parameter e representing the rate of estimation error directly affects the BER performance of the symbol detection strategies. As the value of parameter e is increased, the accuracy of channel coefficients is damaged. Depending on this, the symbol detection performance and accordingly the BER achievement of any strategy gets worse. So, it is crucial to take the impact of imperfect channel coefficients into account to be able to evaluate the robustness and consistency of the symbol detection strategies under practical conditions where the channel state information is often imperfect.

2.5 Migrating birds optimization-based ML strategy

V-shaped flight strategy used by the migratory birds to cover long distances with efficient use of energy inspired Duman et al. to develop a new swarm intelligence-based meta-heuristic algorithm called migrating birds optimization (MBO) [38]. The related flight shape formed by the bird flock during the migration is given in Fig. 1.

In the case that the bird flock takes the shape of V as demonstrated in Fig. 1, the birds in the fore positions have to withstand greater wind resistance than those in the back rows. For this reason, the front row birds have to spend more energy compared to their followers in the back rows. The leader bird is the one that consumes the highest energy in the flock. After a period of time, it becomes impossible for the leader bird to fly in the first position anymore. At that moment, it goes back to the left or right end of the bird flock and one of its followers (i.e., the second or third bird) takes the lead. So, each bird in the flock gets the chance to save energy for a certain period of time by benefiting from the wind breaking effects of the front birds. The related wind breaking benefits provided by the front birds to their followers are modeled in the MBO algorithm as the information-sharing mechanism starting from the leader bird and spreading toward the far ends of the left and right tails of the V-shaped bird flock as demonstrated in Fig. 1 by using the arrows.

In this paper, an advanced MBO variant [18] reinforced by the cyclic bit flipping mechanism was proposed for optimizing the symbol sequences. The related version of MBO developed in [18] belongs to the class of binary optimization algorithms. It has an efficient searching capability in binary search space due to its dynamic neighboring mechanism based on cyclic bit flipping procedure. In order for the cyclic bit flipping-based MBO algorithm to be integrated into the conventional ML scheme, the QAM symbol vectors represented by $x_{m,n} = \left[x_{m,n}^{(1)}\ ,\ x_{m,n}^{(2)}\ ,\ .\ .\ .\ ,\ x_{m,n}^{(N_t)}\right]^{\mathrm{T}}$ need to be made suitable for being optimized by a binary optimization algorithm like the proposed one. To this end, the sequence of QAM symbols each of which is represented by a unique complex number can be converted

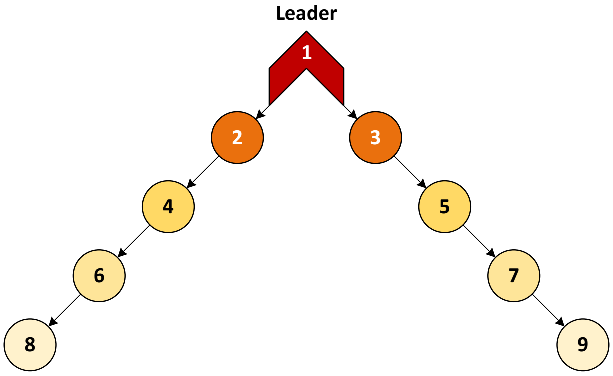

Fig. 1 V-shaped flight pattern for the flock of birds consisting of 9 individuals

to the sequence of binary numbers. By doing so, it becomes possible to use the MBO algorithm based on cyclic bit flipping procedure for the symbol optimization process. Each of the complex-valued QAM symbols in $x_{m,n}$ corresponds to a specific binary number with $\log_2^Z = k$ digits. According to this, the binary equivalent of the complex-valued QAM symbol sequence with the length N_t will have $k \cdot N_t$ elements since each of the N_t complex-valued symbols carries k digit binary information. In the proposed MBO-ML strategy, the complex-valued QAM symbol sequences and their equivalent binary bit strings are represented by the bird positions as expressed in Eqs. (16) and (17), respectively:

$$x_f^{(i)} = \left[x_f^{(1)}\ ,\ x_f^{(2)}\ ,\ \ldots\ ,\ x_f^{(N_t)}\right] \quad ,\quad f = 1\ ,\ 2\ ,\ \ldots\ ,\ F \tag{16}$$

$$b_f^{(j)} = \left[b_f^{(1)}\ ,\ b_f^{(2)}\ ,\ \ldots\ ,\ b_f^{(k\cdot N_t)}\right] \quad ,\quad f = 1\ ,\ 2\ ,\ \ldots\ ,\ F \tag{17}$$

where F represents the aggregate number of birds existing in the bird flock. In Eq. (17), each dimension of the bird position can take binary values, i.e., $b_f^{(j)} \in \{0,\ 1\}$. In the MBO-ML strategy, a certain number of neighboring solutions are generated from each of the bird positions expressed in Eq. (17). These neighbor solutions are defined in the following way:

$$p_{f,w}^{(j)} = \left[p_{f,w}^{(1)}\ ,\ p_{f,w}^{(2)}\ ,\ \ldots\ ,\ p_{f,w}^{(k\cdot N_t)}\right] \quad ,\quad f = 1\ ,\ 2\ ,\ \ldots\ ,\ F\ ;\quad w = 1\ ,\ 2\ ,\ \ldots\ ,\ W \tag{18}$$

In the equation above, $p_{f,w}^{(j)}$ represents the wth neighbor solution vector belonging to the fth population member in the bird flock while the aggregate number of neighbor solutions determined for each of the population members is denoted by W.

The operation steps of the MBO-based ML scheme named MBO-ML in our paper are as follows:

Step 1: At the initial phase, an extra dimension is added to both $b_f^{(j)}$ and $p_{f,w}^{(j)}$ vectors to keep the fitness values of the bird positions and their neighbor solutions, respectively. So, the birds and their neighbors can now be redefined in the following manner:

$$G_f^{(j)} = \left[b_f^{(1)}\ ,\ b_f^{(2)}\ ,\ \ldots\ ,\ b_f^{(k\cdot N_t)}\ ,\ fit\left(b_f^{(j)}\right)\right] \tag{19}$$

$$Q_{f,w}^{(j)} = \left[p_{f,w}^{(1)}\ ,\ p_{f,w}^{(2)}\ ,\ \ldots\ ,\ p_{f,w}^{(k\cdot N_t)}\ ,\ fit\left(p_{f,w}^{(j)}\right)\right] \tag{20}$$

where the fitness values of bird and neighbor position vectors are represented by $fit\left(b_f^{(j)}\right)$ and $fit\left(p_{f,w}^{(j)}\right)$, respectively. Once the predefinition stage is completed, initial bird population is created. To this end, a certain number of birds represented by the vector $G_f^{(j)}$ are generated at random positions of the binary search space. Note that the binary position vectors of the population members correspond to the binary equivalents of the complex QAM symbol sequences that need to be optimized. Each individual in the initial bird flock is then subjected to the operation of fitness calculation. For each population member specified by $G_f^{(j)}$, fitness calculation is carried out in two stages. In the first stage, the binary position vector $b_f^{(j)}$ of the population member $G_f^{(j)}$ is transformed into its equivalent complex-valued QAM symbol vector denoted by $x_f^{(i)}$. In the second stage, the resulted symbol vector $x_f^{(i)}$ is used in the following fitness function to calculate the fitness value of the related bird position denoted by $fit\left(b_f^{(j)}\right)$:

$$fit\left(b_f^{(j)}\right) = \left\| \hat{y}_{m,n} - \hat{H}_{m,n} \cdot x_f^{(i)} \right\|^2 \quad , \quad f = 1\ ,\ 2\ ,\ \ldots\ ,\ F\ ;\quad i = 1\ ,\ 2\ ,\ \ldots\ ,\ N_t \tag{21}$$

Note that the fitness values of the population members are kept at the last dimension of the vector $G_f^{(j)}$ as demonstrated in Eq. (19). After that, the initial bird population is organized to form the shape of V as shown in Fig. 2.

Step 2: After fulfilling the initialization phase in Step 1, W neighboring solutions are generated from the leader bird by using the cyclic bit flipping mechanism, which was incorporated into the conventional MBO technique as a neighbor solution generator in [18] for the first time. In order to carry out cyclic bit flipping-based neighboring generation process, one more dimension is required in both $G_f^{(j)}$ and $Q_{f,w}^{(j)}$ to keep the flipping index value for each of the birds and their neighboring solutions as expressed below:

$$\begin{aligned} G_f^{(j)} &= \left[b_f^{(1)}\ ,\ b_f^{(2)}\ ,\ \ldots\ ,\ b_f^{(k\cdot N_t)}\ ,\ \mathrm{fit}\left(b_f^{(j)}\right)\ ,\ \mu_f \right] \\ &= \left[b_f^{(j)}\ ,\ \mathrm{fit}\left(b_f^{(j)}\right)\ ,\ \mu_f \right] \end{aligned} \tag{22}$$

$$\begin{aligned} Q_{f,w}^{(j)} &= \left[p_{f,w}^{(1)}\ ,\ p_{f,w}^{(2)}\ ,\ \ldots\ ,\ p_{f,w}^{(k\cdot N_t)}\ ,\ fit\left(p_{f,w}^{(j)}\right)\ ,\ \mu_{f,w} \right] \\ &= \left[p_{f,w}^{(j)}\ ,\ fit\left(p_{f,w}^{(j)}\right)\ ,\ \mu_{f,w} \right] \end{aligned} \tag{23}$$

In the equations given above, while the flipping index of the *f*th bird is specified by μ_f, the flipping index belonging to the *w*th neighbor solution of the *f*th bird is symbolized by $\mu_{f,w}$. It is important to note that the initial value of the flipping index is appointed as $\mu_f = 1$ for each individual existing in the bird flock. After the last modifications carried out in the vector definitions of the birds and their neighbors, it becomes possible to generate neighbor solutions by complying with the cyclic bit flipping mechanism. With the intention of generating the 1st neighboring solution of the

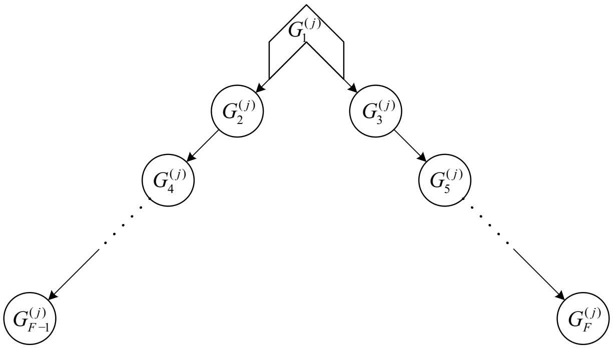

Fig. 2 V-shaped organization of the initial population members

1st bird, which is also called as the leader bird, the bit flipping operation expressed in the following equation is performed by equalizing the variables f and w to the value of 1 (for $f=1$ and $w=1$):

$$Q_{f,w}^{(j)} = flip\left(G_f^{(j)}\right)_{\mu_f} \tag{24}$$

where the operator $flip\left(G_f^{(j)}\right)_{\mu_f}$ carries out the flipping operation on the element that exists in the μ_f th dimension of the vector $G_f^{(j)}$ by changing its value from 0 to 1 or vice versa. Remember that each of the elements existing in the dimensions 1 to $k \cdot N_t$ of the vector $G_f^{(j)}$ can take the value of either 0 or 1. The same goes for the vector $Q_{f,w}^{(j)}$. Namely, both $b_f^{(j)} \in \{0,\ 1\}$ and $p_{f,w}^{(j)} \in \{0,\ 1\}$. Subsequent to the aforementioned bit flipping operation, the flipping index μ_f, the value of which was made equal to 1 for each population member at the beginning, is updated in the following manner.

$$\mu_f = \mu_f + 1 \tag{25}$$

Thanks to increasing the flipping index by 1 right after each flipping operation, it is ensured that the next flipping operation is carried on from the vector element next to the previous flipped one. For each generation of neighbor solution, one bit flipping is performed. In the cyclic bit flipping procedure, the bit flipping is required to move in cyclic fashion from the 1st to the $(k \cdot N_t)$ th dimension of the vector $G_f^{(j)}$. To achieve this goal, it is desired from μ_f to become equal to 1 each time it exceeds the value of $k \cdot N_t$, which is the last vector dimension that keeps the last bit to be flipped in the vector $G_f^{(j)}$. In order to guarantee that the flipping index μ_f takes values cyclically from 1 to $k \cdot N_t$, the following operation is applied as a control mechanism each time the value of μ_f is increased by one:

$$\mu_f = \mathrm{mod}\left(\mu_f - 1,\ \ k \cdot N_t\right) + 1 \tag{26}$$

where the operator mod( ) performs the modulo operation by finding the remainder of dividing $\mu_f - 1$ by $k \cdot N_t$. Thanks to the usage of cyclic bit flipping mechanism as a neighbor generator in the MBO algorithm, the binary search space can be scanned more effectively by minimizing the unvisited neighbor solutions for each of the population members. Subsequent to fulfilling the generation of neighbor solution $Q_{f,w}^{(j)}$ from the $G_f^{(j)}$ (for $f=1$ and $w=1$) and updating the flipping index μ_f via Eqs. (24–26), $\mu_{f,w}$ corresponding to the flipping index of the newly generated neighbor solution $Q_{f,w}^{(j)}$ is made equal to μ_f:

$$\mu_{f,w} = \mu_f \tag{27}$$

Note that $Q_{f,w}^{(j)}$ is produced by flipping the μ_f th element in the vector $G_f^{(j)}$. So, if the neighbor solution $Q_{f,w}^{(j)}$ takes the place of $G_f^{(j)}$ due to its better fitness quality, it is required that the subsequent flipping operation for the $Q_{f,w}^{(j)}$ moves on from the component existing next to the μ_f th dimension to keep the flipping of vector components moving in cyclic mode throughout the optimization process. To this end, after producing the $Q_{f,w}^{(j)}$ from the $G_f^{(j)}$, the value of $\mu_{f,w}$ needs to be equalized with that of μ_f updated by Eqs. (25) and (26).

The operations performed by using Eqs. (24–27) to produce the 1st neighbor solution of the leader bird are repeated W times in total to obtain W different neighbor solutions for the leader bird. Later on, the fitness qualities of the generated neighbor solutions $Q_{f,w}^{(j)}$ are evaluated. For this purpose, the binary bit sequences $p_{f,w}^{(j)}$ that exist in the vectors $Q_{f,w}^{(j)}$ are converted to their equivalent QAM symbol sequences $x_{f,w}^{(i)}$, each of which consists of complex numbers, and the resulted $x_{f,w}^{(i)}$ symbol vectors are then utilized in the fitness function given below to compute the fitness values of the binary positions $p_{f,w}^{(j)}$ belonging to the neighbor solutions represented by $Q_{f,w}^{(j)}$:

$$fit\left(p_{f,w}^{(j)}\right) = \left\| \hat{y}_{m,n} - \hat{H}_{m,n} \cdot x_{f,w}^{(i)} \right\|^2 \quad , \quad f = 1\ ,\ 2\ ,\ \ldots\ ,\ F\ ; \quad i = 1\ ,\ 2\ ,\ \ldots\ ,\ N_t \tag{28}$$

After that, the neighbor solutions are sorted by considering their fitness qualities from best to worst. Since the symbol optimization is a kind of minimization problem, the neighboring solution with the lowest fitness value will take the top place in the group of neighbor solutions. After the sorting process, the first-ranked neighbor solution is compared with the leader bird in point of fitness quality. If the related neighbor solution has better fitness quality compared to the leader bird, it takes the place of that leader bird. Subsequently, starting from the second-ranked neighbor solution, H number of neighbor solutions with even index (i.e., 2, 4, 6, ...) in the sorted list are shared with the second bird while H number of neighbor solutions having odd index (i.e., 3, 5, 7, ...) are shared with the third bird. H is the parameter that determines the number of neighbor solutions to be shared with a single follower bird. As the leader bird has two followers (i.e., second and third bird) unlike the other ones, it shares $2H$ neighbor solutions in total.

Step 3: The operations given below are performed for each of the remaining individuals forming the left and right tails of the V-shaped bird flock:

- (W – H) number of neighbor solutions are generated via Eqs. (24–27).
- The fitness qualities of the generated solutions are calculated via Eq. (28).
- The aggregate number of neighboring solutions is completed to W by adding H more neighbor solutions taken from the bird in front to the current neighbor group consisting of (W – H) members.
- The members of the resulted neighbor group are put in order in point of fitness quality.
- If the best neighbor solution that exists in the first place of the neighbor group leaves behind the corresponding bird with regard to fitness quality, the related bird is replaced by this neighbor solution.
- Starting from the second solution in the neighbor group, H number of solutions are shared with the bird in the back.
- Once the Step 3 operations are fulfilled for each population member excluding the leader bird, one tour is completed.

Step 4: After reaching a certain number of tours (T) predetermined before the optimization process, the leader bird moves to the bottom of the left tail in the V-shaped bird flock and the second bird existing at the top of the same tail takes the lead by shifting to the first position. The operation of leader changing is performed alternately on the left and right tails. For example, in the next leader changing operation, this time the leader bird will move to the bottom of the right tail and the third bird existing at the top of the right tail will take the leadership by passing to the first place in the bird flock with V-shape. Once the operation of leader replacement is carried out after T number of tours, one cycle is completed for the MBO-ML strategy. The operations from Step 2 to Step 4 are reiterated until reaching the maximum number of cycles (C) specified as the terminating criterion at the beginning of optimization. Subsequent to the algorithm's termination, the binary bird position possessing the best fitness value is determined as the optimum solution. As the final operation of the MBO-ML strategy, the optimum symbol vector is achieved by converting the resulted binary bit sequence to its corresponding QAM symbol sequence. Together with this last operation, the process of symbol detection is completed. The flowchart of the MBO-ML scheme is demonstrated in Fig. 3.

3 Results and discussion

In this section, with the intention to put forth the capability of MBO-ML strategy in detecting the QAM symbol vectors at the receiver of MIMO-FBMC/OQAM system, the BER achievement of the proposed strategy is compared to those of the other considered symbol detection methods for 4×4, 6×6 and 8×8 antenna configurations in the related system. Apart from this, in order to visualize how fast the MBO-ML is in approaching the near-optimal solution regardless of antenna structure, convergence analysis is carried out for each of the MIMO configurations considered in this paper. Furthermore, a fairly extensive complexity analysis including elaborated mathematical expressions, complexity graphs and comparison tables is made at the end of this section. In the simulations, the signal transmission is carried out over a widely used standard channel model called "ITU Vehicular B", which has [0, 300, 8900, 12,900, 17,100, 20000] ns relative delays and $[-2.5, 0, -12.8, -10, -25.2, -16]$ dB power paths. Table 1 shows the parameters used for the simulation of MIMO-FBMC/OQAM scheme.

In Table 2, the existing ML-oriented symbol detection schemes including the conventional ML by itself are analyzed with regard to their search complexities for 4×4, 6×6 and 8×8 MIMO configurations. The search complexity should not be confused with the overall computational complexity, which will be obtained for each technique at the end of this section. In the ML-based symbol detectors, in which the optimum symbol vector is tried to find among all possible symbol combinations called as search space, the expression of search complexity forms the main component in their overall computational complexities. Without the search complexity component, the ML-based symbol detectors would have the same computational complexity expressions since the search complexity is the only part that separates the related ML-based strategies from each other with regard to their computational complexities. For this reason, prior to the computational complexity analysis, the search complexity of each technique needs to be obtained.

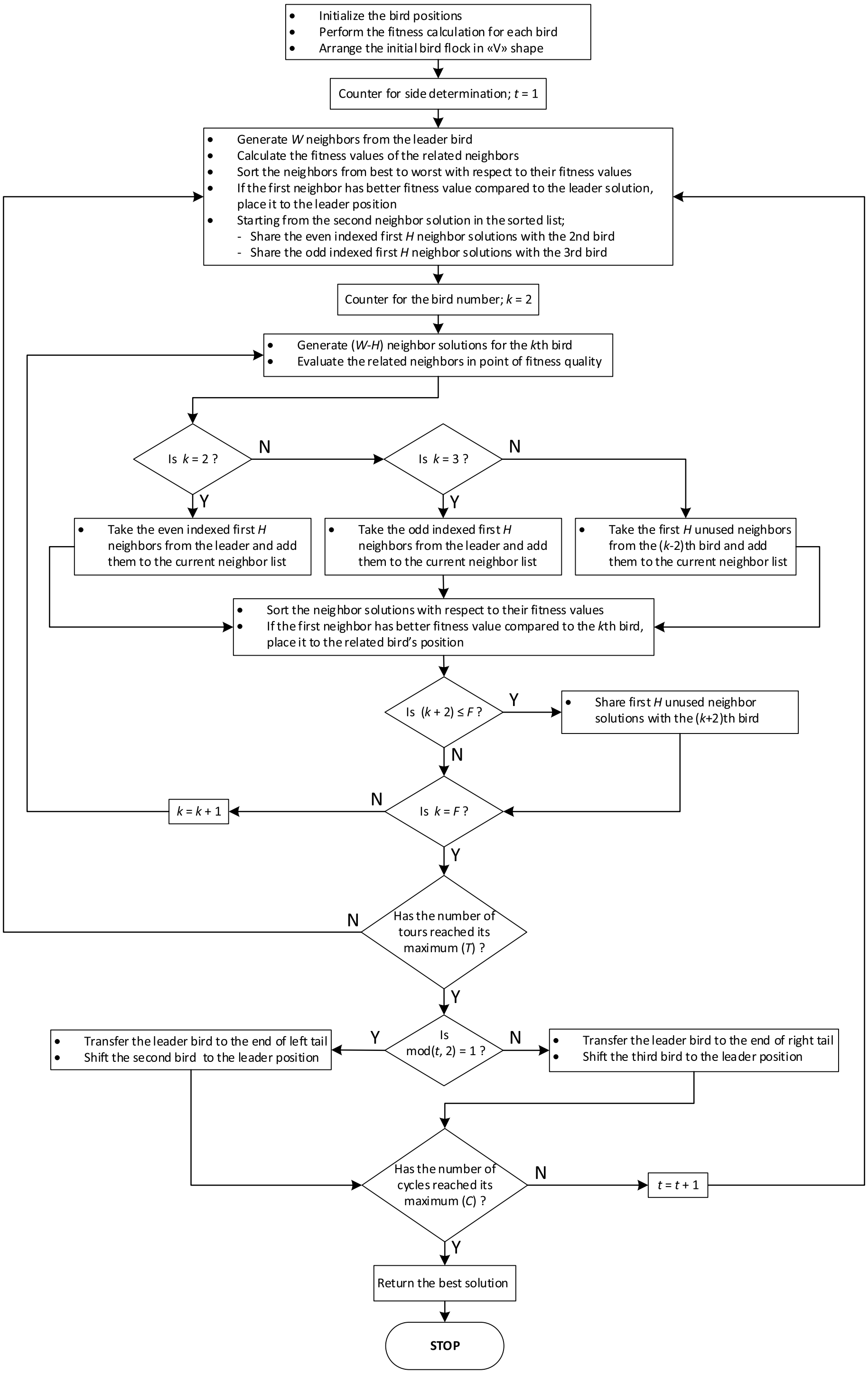
Initialize the bird positions
Perform the fitness calculation for each bird
Arrange the initial bird flock in «V» shape
Counter for side determination; t = 1
Generate W neighbors from the leader bird
Calculate the fitness values of the related neighbors
Sort the neighbors from best to worst with respect to their fitness values
If the first neighbor has better fitness value compared to the leader solution, place it to the leader position
Starting from the second neighbor solution in the sorted list;
- Share the even indexed first H neighbor solutions with the 2nd bird
- Share the odd indexed first H neighbor solutions with the 3rd bird
Counter for the bird number; k = 2
Generate (W-H) neighbor solutions for the kth bird
Evaluate the related neighbors in point of fitness quality
Is k = 2 ?
N
Is k = 3 ?
N
Y
Y
Take the even indexed first H neighbors from the leader and add them to the current neighbor list
Take the odd indexed first H neighbors from the leader and add them to the current neighbor list
Take the first H unused neighbors from the (k-2)th bird and add them to the current neighbor list
Sort the neighbor solutions with respect to their fitness values
If the first neighbor has better fitness value compared to the kth bird, place it to the related bird's position
Is (k + 2) ≤ F ?
Y
Share first H unused neighbor solutions with the (k+2)th bird
N
N
k = k + 1
Is k = F ?
Y
N
Has the number of tours reached its maximum (T) ?
Y
Transfer the leader bird to the end of left tail
Shift the second bird to the leader position
Y
Is mod(t, 2) = 1 ?
N
Transfer the leader bird to the end of right tail
Shift the third bird to the leader position
Has the number of cycles reached its maximum (C) ?
N
t = t + 1
Y
Return the best solution
STOP

Fig. 3 Flowchart of the MBO-ML scheme

When we consider the classical ML scheme to determine its search complexity, the aggregate count of Euclidean distance computations performed for each of the possible QAM symbol combinations during the search for the optimal one is taken into account. Note that the way of calculating the Euclidean distance for each candidate symbol

Table 1 System parameters for the MIMO-FBMC/OQAM

Frequency of the carrier	2.5 GHz
The value of overlapping factor	4
Model of prototype filter	PHYDYAS
Subcarrier number	$M=64$
Order of QAM modulation	$Z=4$
Number of FBMC symbols	$N=30$
MIMO configurations	$4 \times 4, 6 \times 6, 8 \times 8$
The rate of estimation error	$e=25\%$
Spacing for the subcarriers	15 kHz
Type of wireless channel	ITU Vehicular B
Additional multiplications parameter	$\beta=2$

combination is given in Eq. (13). Due to the fact that the total number of possible symbol combinations is calculated by Z^{N_t} as mentioned before, the search complexity of the conventional ML will be equal to Z^{N_t} as well, where N_t and Z represent the number of antennas at the transmitter and modulation order for QAM, respectively.

As for the modern ML strategies that rely on the meta-heuristic approaches, each fitness evaluation will lead to one Euclidean calculation. So, for each meta-heuristic-based ML technique, it is sufficient to find the aggregate number of fitness computations carried out by the relevant strategy throughout the optimization process to acquire its search complexity. In the line with this, BPSO-ML's search complexity can be obtained by finding how many fitness calculations are carried out until the end of iterations. To this end, we multiply the swarm size with the maximum number of iterations due to the fact that all of the particles existing in the swarm are subjected to fitness calculation for each iteration. Likewise, we can obtain the search complexity belonging to the disABC-ML by multiplying the number of bees with the maximum number of loops since the entire bee population is evaluated one by one in point of fitness quality at each loop. DBHS-ML technique separates from the other considered methods in point of search complexity acquisition. In this technique, each search leads to the generation of one new solution. Because of this, one fitness evaluation is required for a single search. According to this, regardless of the population size, maximum number of searches will directly give the aggregate count of fitness evaluations, which will be equal to the DBHS-ML's search complexity at the same time as demonstrated in Table 2. In our proposed MBO-ML strategy, at each tour, while the leader bird generates W neighbor solutions, each of the remaining population members ($F-1$ number of birds) generate $W-H$ neighbor solutions. Since each new solution needs one fitness evaluation, the number of fitness evaluations needed for one tour will be equal to the aggregate count of neighbor solutions generated in a single tour as $(F-1)\cdot(W-H)+W$. For T number of tours, the count of fitness calculations becomes as $[(F-1)\cdot(W-H)+W]\cdot T$. Since T number of tours corresponds to a single cycle, the entire quantity of fitness computations, which

Table 2 The examination of search complexity for each of the ML-based symbol detection strategies

Symbol detection procedures		**4 × 4**	**6 × 6**	**8 × 8**
BPSO-ML	Swarm size	$S_{size}=20$	$S_{size}=30$	$S_{size}=40$
	Maximum number of iterations	$I_{\max}=13$	$I_{\max}=30$	$I_{\max}=55$
	Search complexity	$C_{search}=S_{size}\cdot I_{\max}=260$	$C_{search}=S_{size}\cdot I_{\max}=900$	$C_{search}=S_{size}\cdot I_{\max}=2200$
	BER at 12 dB SNR	0.04564	0.03083	0.03255
disABC-ML	Number of bees	$N_B=20$	$N_B=30$	$N_B=40$
	Maximum number of loops	$L_{\max}=13$	$L_{\max}=30$	$L_{\max}=55$
	Search complexity	$C_{search}=N_B\cdot L_{\max}=260$	$C_{search}=N_B\cdot L_{\max}=900$	$C_{search}=N_B\cdot L_{\max}=2200$
	BER at 12 dB SNR	0.02939	0.01451	0.01143
DBHS-ML	Size of population	$P_{size}=20$	$P_{size}=30$	$P_{size}=40$
	Maximum number of searches	$S_{\max}=260$	$S_{\max}=900$	$S_{\max}=2200$
	Search complexity	$C_{search}=S_{\max}=260$	$C_{search}=S_{\max}=900$	$C_{search}=S_{\max}=2200$
	BER at 12 dB SNR	0.01755	0.007153	0.006172
MBO-ML	Number of birds	$F=13$	$F=19$	$F=25$
	Number of neighbor solutions for each bird	$W=4$	$W=4$	$W=4$
	Number of neighbor solutions to be shared	$H=1$	$H=1$	$H=1$
	Maximum number of tours	$T=1$	$T=2$	$T=2$
	Maximum number of cycles	$C=5$	$C=5$	$C=8$
	Search Complexity	$C_{search}=[(F-1)\cdot(W-H)+W]\cdot C\cdot T=200$	$C_{search}=[(F-1)\cdot(W-H)+W]\cdot C\cdot T=580$	$C_{search}=[(F-1)\cdot(W-H)+W]\cdot C\cdot T=1216$
	BER at 12 dB SNR	0.01199	0.003926	0.002212
ML	Search Complexity	$C_{search}=Z^{N_t}=4^4=256$	$C_{search}=Z^{N_t}=4^6=4096$	$C_{search}=Z^{N_t}=4^8=65536$
	BER at 12 dB SNR	0.009193	0.002682	0.001432

also expresses the eventual search complexity, is going to be $[(F-1)\cdot(W-H)+W]\cdot C\cdot T$ for C number of cycles.

In case of escalating the antenna quantity, which determines the number of dimensions in the QAM symbol vectors to be optimized, the search space determined by the total number of vector variations likely to be transmitted becomes greater as well. For this reason, every symbol detecting scheme on the basis of intelligent optimization method needs expanded population sizes and increased number of searches for higher MIMO configurations as indicated in Table 2. For instance, the population sizes determined for the benchmark techniques are equal to 20, 30 and 40 for 4×4, 6×6 and 8×8 MIMO structures while those of the suggested MBO-ML scheme are appointed as 13, 19 and 25, respectively, for the relevant antenna structures. In a similar way, while the other comparative strategies' search complexities are set to 260, 900 and 2200, those of the MBO-ML strategy developed in this paper are determined as 200, 580 and 1216, respectively, for the aforementioned MIMO configurations. As it seems evident in Table 2, our proposed MBO-ML scheme manages to achieve the lowest BER levels among the existing meta-heuristic-based ML detectors for each antenna configuration in spite of having smaller population sizes and search complexities in comparison to the related benchmark ML strategies modified by meta-heuristic algorithms. It should be noted that when determining the values of search complexities for the suggested MBO-ML scheme, we take the trade-off between the BER achievement and search complexity into consideration. 200, 580 and 1216 numbers of fitness computations become enough for MBO-ML to achieve satisfactory BER levels for 4×4, 6×6 and 8×8 antenna configurations. Increasing the quantity of fitness calculations beyond the related values doesn't contribute much to BER improvement but brings about a significant elevation in the complexity of MBO-ML. Apart from this, the fitness evaluation quantities belonging to the other considered methods are equalized to much bigger values like 260, 900 and 2200 to make the conditions more difficult for our proposed MBO-ML detector by giving more searching opportunity to its opponents. Despite being given considerably less searching opportunity for each antenna configuration, our proposed symbol detection strategy clearly predominates the other benchmark meta-heuristic-based methods by acquiring better BER results. Table 3 provides the remaining parameter values of the meta-heuristic-based ML techniques.

In Fig. 4, the convergence performances of the meta-heuristic-based symbol detection strategies are analyzed for SNR = 12 dB. In this simulation, 4×4 antenna configuration

Table 3 The parameter values of the benchmark techniques based on intelligent optimization algorithms

Technique	Parameter name	Value
BPSO-ML	Inertia weight (w)	0.9 ~ 0.4
	Maximum velocity (V_{max})	6
	Cognitive component (c_1)	2
	Social component (c_2)	1
disABC-ML	Maximum number of trials	30
DBHS-ML	The rate of harmony memory consideration (*HMCR*)	0.6
	The rate of pitch adjustment (*PAR*)	0.05

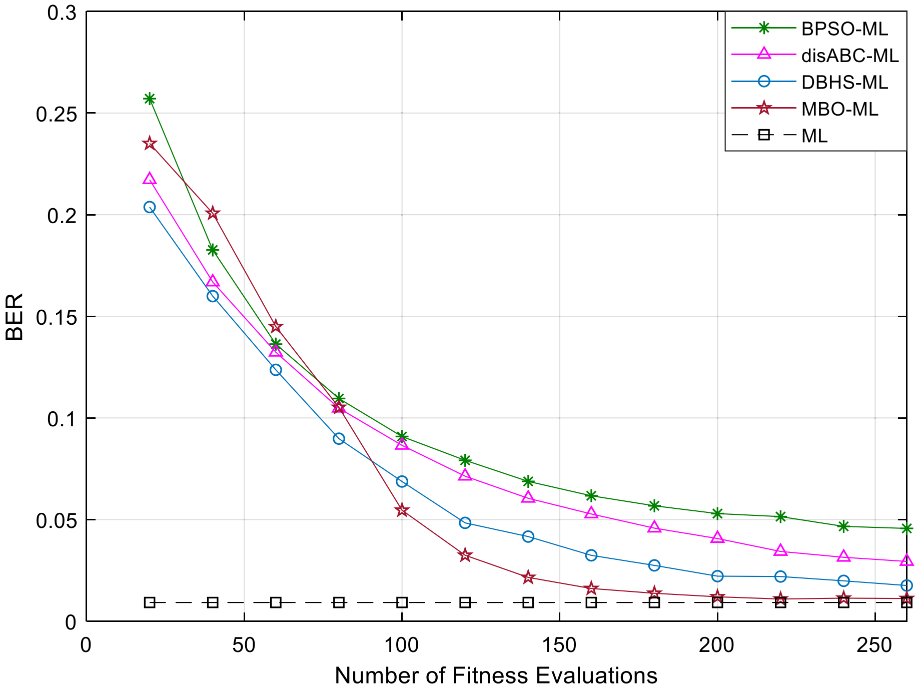

Fig. 4 Converging performances of the meta-heuristic-based ML procedures for 4 × 4 antenna structure in MIMO-FBMC/OQAM scheme

is used for the MIMO-FBCM/OQAM system. The convergence graphs belonging to the considered meta-heuristic-based ML schemes are acquired in the following way: For each number of fitness evaluations given in the horizontal axis of the convergence graph, 10 different BER values are obtained by using the corresponding symbol detection strategy. To put it another way, the number of fitness evaluations belonging to the related meta-heuristic-based ML strategy is set to the values existing in the horizontal axis one by one and subsequent to each setting, BER simulation is carried out for 10 times to obtain 10 different BER levels. Finally, the convergence graph is obtained for the relevant scheme by averaging these BER levels attained for each number of fitness evaluations. The straight line existing at the bottom of Fig. 4 indicates the level of BER acquired by using the conventional ML scheme in MIMO-FBMC/OQAM system with 4 × 4 antenna configuration. It is impossible for any of the ML variants to go beyond that level since the exhaustive search procedure, in which each of the possible symbol combinations available in the search space is tested one by one, always ensures the conventional ML to get the optimum symbol vector at the end of each detecting operation. So, the quality of an intelligent optimization-based ML strategy can be determined by looking at not only how much it gets near to the optimum BER level attained by the classical ML scheme, but also how many fitness evaluations it needs to converge that level. Namely, it is expected from a symbol detector based on a meta-heuristic algorithm to converge the optimal BER level as much as possible by performing minimum number of fitness evaluations. In the convergence graph given in Fig. 4, it is clearly seen that our proposed MBO-ML strategy meets this expectation. It rapidly gets really close to the optimal BER level by performing just 200 fitness evaluations while the other three

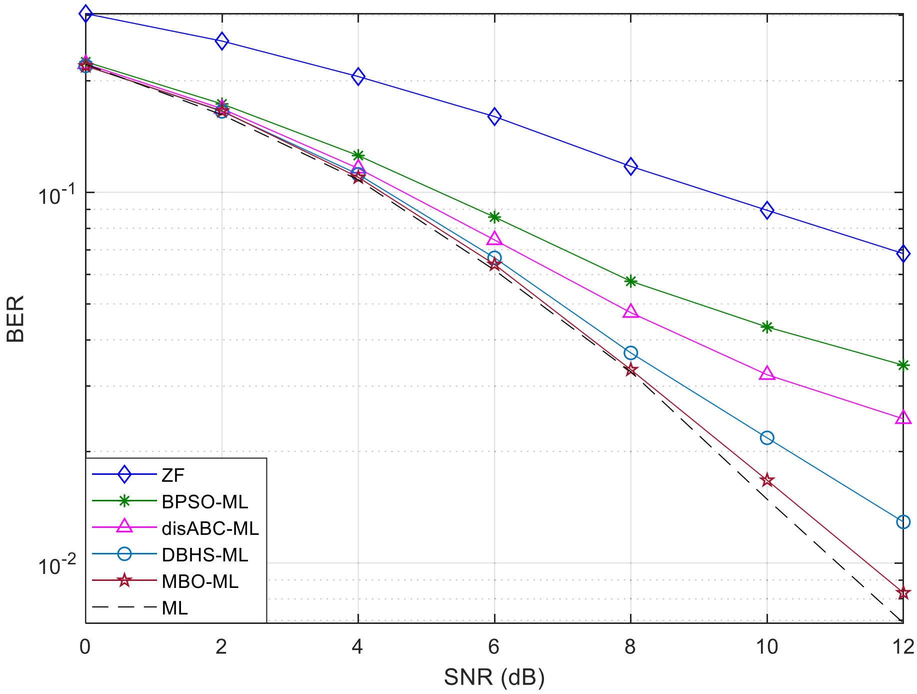

Fig. 5 The BER performances of the existing detection procedures in MIMO-FBMC/OQAM scheme for 4 × 4 antenna configuration

methods still appear to be far away from that level even after 260 fitness evaluations. For instance, while 260 fitness computations are needed by the DBHS-ML, disABC-ML and BPSO-ML techniques to achieve 0.01755, 0.02939 and 0.04564 BERs, respectively, it will be adequate for our proposed MBO-ML scheme to carry out 108, 126 and 155 fitness evaluations to reach the same BER levels. As previously mentioned, we should consider the trade-off between the search complexity and BER performance to determine the optimal number of fitness evaluations. As it is evident from Fig. 4, the number of fitness evaluations greater than 200 is unnecessary for the proposed MBO-ML strategy as very tiny amount of BER improvement is achieved beyond that value. In other words, the convergence of MBO-ML is nearly completed at 200 fitness evaluations. So, it is logical to use 200 fitness evaluations for the MBO-ML scheme to ensure that it achieves near-optimal solution with minimum search complexity.

In Fig. 5, the considered symbol detection strategies are compared with each other in point of their BER achievements in MIMO-FBMC/OQAM scheme employing 4 × 4 antenna configuration. As can be clearly seen from Fig. 5, the conventional ML and ZF stand out as the best and worst performing methods, respectively, while the state-of-the-art symbol detection strategies based on meta-heuristic algorithms perform somewhere between these two methods. It is not a surprise that ML symbol detector reaches the lowest BER level because of its searching procedure in which each of the possible symbol combinations are tested to find the optimal one. The related searching procedure guarantees to reach the optimal solution for each symbol detection process. However, it causes extremely high complexity in the transmission systems especially for higher antenna configurations. Herewith, the performance level belonging to the ML scheme

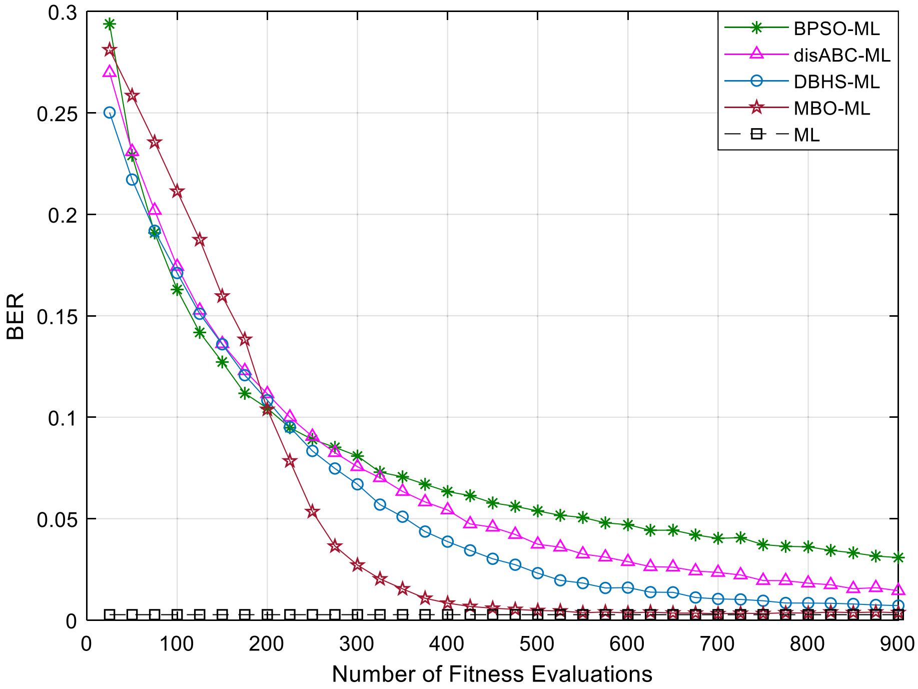

Fig. 6 Converging performances of the meta-heuristic-based ML procedures for 6 × 6 antenna configuration in MIMO-FBMC/OQAM system

can be considered as the upper bound for the remaining symbol detectors. As obviously seen from Fig. 5, MBO-ML shows a very close BER performance to that of conventional ML method. The other three meta-heuristic-based ML schemes cannot even get close to the performance of our proposed strategy even though they are allowed to carry out significantly higher quantity of fitness calculations. As an example, in case of taking the 10 dB SNR as a reference point in the horizontal axis, it will be seen that our proposed MBO-ML strategy manages to obtain 1.67×10^{-2} BER value, which is very close to the value of 1.49×10^{-2} achieved by the ML scheme, by carrying out only 200 number of fitness computations, while the DBHS-ML, disABC-ML and BPSO-ML symbol detection techniques barely reach 2.18×10^{-2}, 3.22×10^{-2} and 4.33×10^{-2} BER levels, respectively, with 260 fitness evaluations.

In Fig. 6, the suggested MBO-ML detector is analyzed with regard to its convergence performance in FBMC/QAM system with 6×6 MIMO configuration by comparing its convergence curve with those of the other meta-heuristic-based ML techniques. The related comparison is carried out for SNR = 12 dB as in the previous convergence analysis made in Fig. 4. The number of feasible vector variations computed by Z^{N_t} is directly determined by the modulation order and the number of transmit antennas. For a fixed modulation order, the increase of $N_t \times N_r$ antenna structure from 4×4 to 6×6 will result in an enhancement in the value of Z^{N_t} and accordingly the expansion of the search space. For this reason, each symbol detector based on an intelligent optimization algorithm will need more fitness evaluations with the relevant increment in the antenna number to be able to converge the optimum BER level as we can realize by comparing Fig. 6 with Fig. 4. The convergence graph given in Fig. 6 clearly demonstrates how quickly

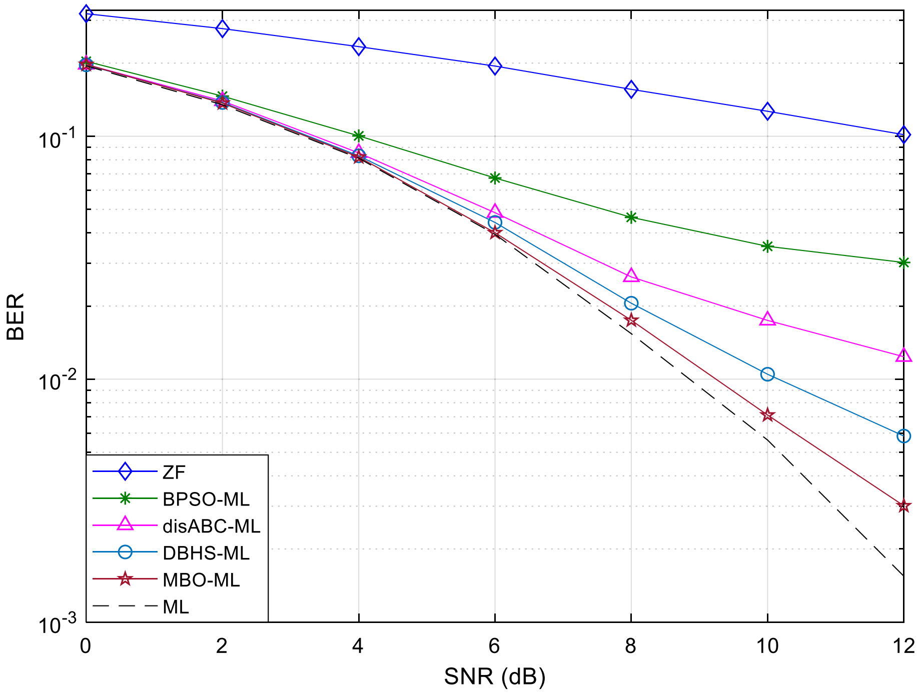

Fig. 7 The BER performances of the existing detection procedures in MIMO-FBMC/OQAM scheme for 6 × 6 antenna configuration

and decisively the MBO-ML strategy converges the optimum solution. After 200 fitness computations, it clearly leaves behind the other three methods. From that point, while the convergence rates of the benchmark methods significantly slow down, our proposed MBO-ML strategy keeps its convergence speed and reaches the near-optimum BER level in only 580 fitness calculations. Due to its impressive convergence ability, much smaller number of searches will be adequate for the suggested MBO-ML scheme to achieve the BER levels reached by the comparative detection strategies by carrying out 900 fitness computations. To give an example, 0.007153, 0.01451 and 0.03083 BER levels attained by the DBHS-ML, disABC-ML and BPSO-ML at the end of 900 fitness calculations can be reached by MBO-ML with considerably low numbers of fitness evaluations like 419, 355 and 290, respectively. For 6×6 antenna configuration, the fitness evaluation quantity is appointed as 580 for the MBO-ML scheme due to the fact that its convergence graph saturates approximately at that point in the horizontal axis. By doing so, the needless complexity increment is avoided.

In Fig. 7, the BER achievements of the symbol detectors are examined for 6×6 antenna structure in MIMO-FBMC/OQAM transmission scheme. When looking back at the previous BER analysis made for 4×4 antenna configuration in Fig. 5, it will be noticed that the enhancement in the size of antenna structure from 4×4 to 6×6 leads to a considerable improvement in the BER performance of each symbol detector excluding the ZF whose performance gets worse unlike the other ones. Due to being in the category of linear symbol detectors, ZF suffers from performance loss with the increase in the number of antennas. On the other hand, despite the enlargement of antenna configuration from 4×4 to 6×6, our proposed MBO-ML strategy consistently maintains

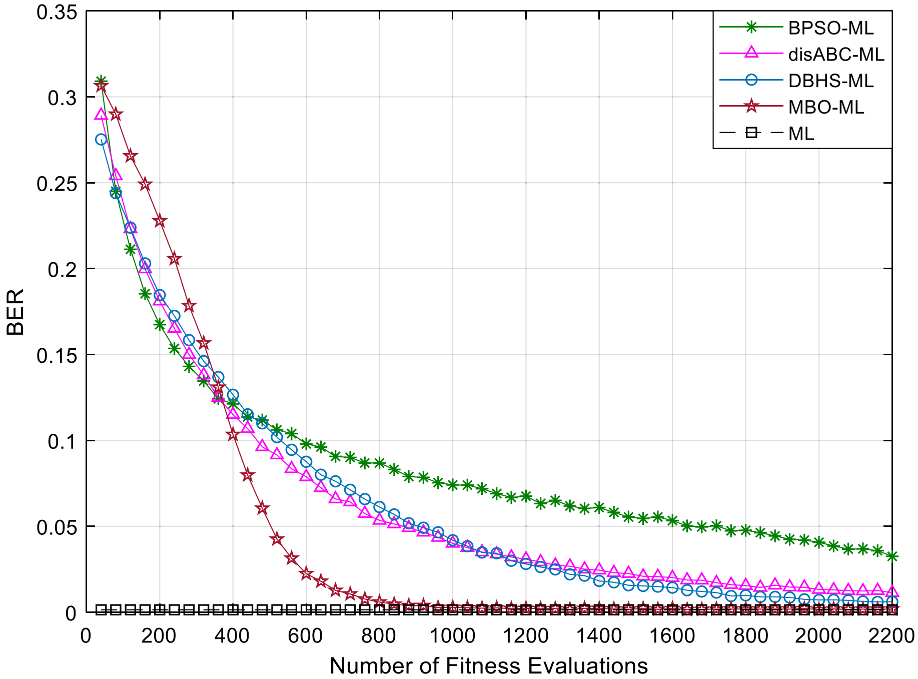

Fig. 8 Converging performances of the meta-heuristic-based ML procedures for 8 × 8 antenna configuration in MIMO-FBMC/OQAM system

its superior performance over the other existing meta-heuristic-based symbol detection methods by obtaining the closest BER results to the optimal ones acquired by the ML detector. For instance, if it is concentrated on the BER values achieved for SNR = 10 dB in Fig. 7, it is observed that the BER of MBO-ML is equal to 7.12×10^{-3}, while those of its three closest opponents are equal to 3.52×10^{-2}, 1.75×10^{-2} and 1.05×10^{-2}, respectively, from highest to lowest.

In Fig. 8, for the purpose of observing what kind of convergence performance the existing strategies under consideration exhibit in a larger search space, the convergence graph of each strategy is acquired for a larger antenna configuration like 8 × 8 in MIMO-FBMC/OQAM system. As in every convergence analysis, SNR value is fixed to 12 dB while acquiring the related convergence curves. As clearly illustrated by Fig. 8, the suggested MBO-ML scheme converges to the near-optimal solution in the most direct way without needing too many fitness evaluations. Its convergence curve completely saturates around 1200 number of fitness evaluations, which is the main reason for setting its search complexity (number of fitness computations) to 1216. Just 1216 fitness calculations have been more than enough for our suggested symbol detection procedure to achieve the lowest BER level, which the other three techniques cannot reach even after 2200 fitness evaluations. Namely, the MBO-ML strategy not only reaches a better solution compared to the other ones, but also carries out this task in the fastest way. If we look at the convergence results from another perspective, it will be realized that the BER levels barely achieved via the three benchmark techniques in 2200 fitness computations can be easily obtained by using our suggested procedure with substantially fewer number of fitness calculations. To give an example, DBHS-ML, disABC-ML and BPSO-ML require 2200 number of fitness computations to attain 0.006172, 0.01143 and 0.03255

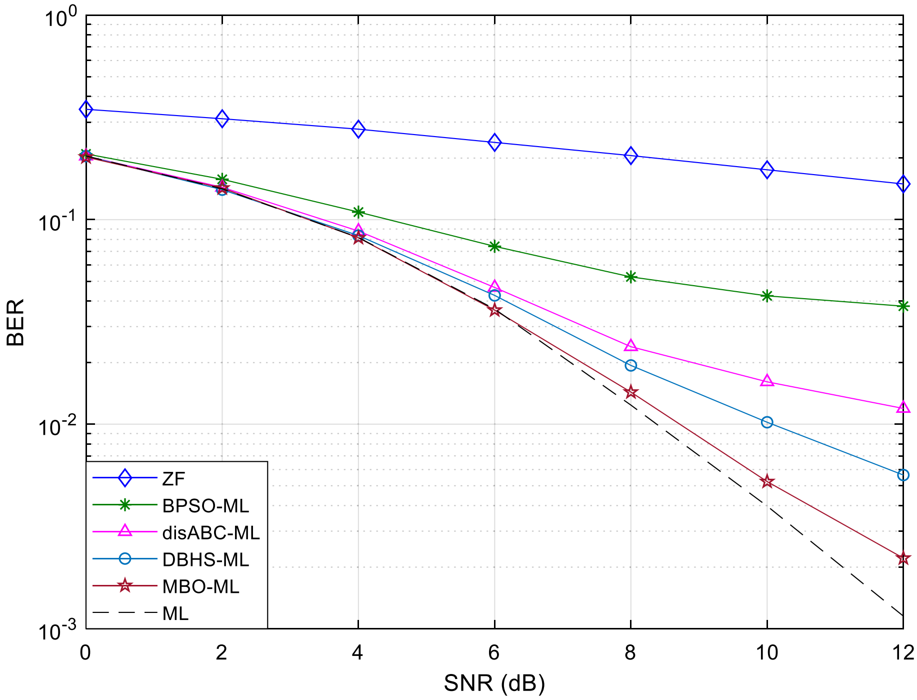

Fig. 9 The BER performances of the existing detection procedures in MIMO-FBMC/OQAM scheme for 8 × 8 antenna configuration

BERs. On the other hand, 795, 705 and 556 fitness computations will be enough for the MBO-ML, respectively, to achieve the same BER levels.

Figure 9 demonstrates the SNR(dB)-BER examination of the considered detectors for 8 × 8 antenna configuration. As mentioned before, adding extra antennas to the MIMO-FBMC/OQAM transceiver results in an exponential expansion of the search space. For this reason, the meta-heuristic-based symbol detectors need much more fitness evaluations after each increment in the number of antennas since it becomes harder for the related symbol detectors to reach an acceptable solution in a much bigger search space without performing a considerable number of fitness evaluations. Excluding the BPSO-ML whose performance gets even worse due to its slow convergence speed that is insufficient for 8 × 8 antenna configuration to reach better BER results in 2200 fitness evaluations, there have been certain improvements in the BER results of the considered meta-heuristic-based ML strategies with the increase of antenna structure from 6 × 6 to 8 × 8 since the higher antenna structures alleviate the fading impacts of the multipath channel on the transmission signals. On the other hand, the suggested MBO-ML has managed to become the best performing symbol detector among the meta-heuristic approaches for 8 × 8 antenna configuration too by reaching the lowest BER level with considerably smaller number of fitness evaluations in comparison to the other ones. It has significantly widened the gap with its nearest rival called DBHS-ML. As an example, while 10.070 dB SNR is required for DBHS-ML to reach 10^{-2} BER level, 8.717 dB SNR value becomes adequate for MBO-ML to achieve the same BER level. According to these results, our proposed symbol detector provides 1.353 dB SNR gain over its closest

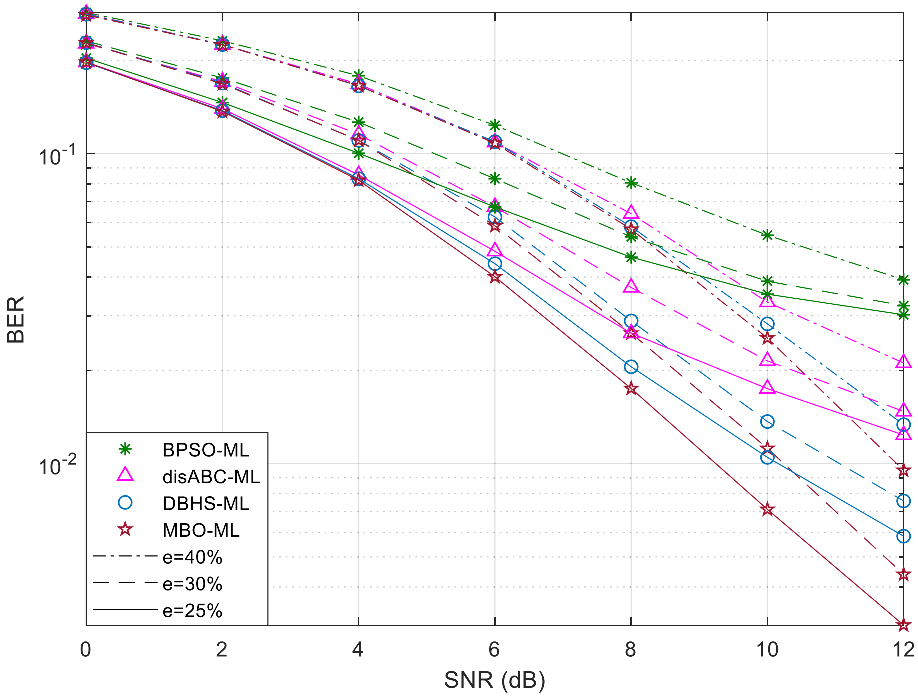

Fig. 10 The BER results obtained via advanced ML strategies based on meta-heuristic algorithms for higher estimation errors in 6 × 6 MIMO-FBMC/OQAM system

opponent in spite of performing 784 less number of iterations in MIMO-FBMC/OQAM system with 8 × 8 antenna structure.

In Fig. 10, the conditions are made more difficult for the considered symbol detectors by increasing the estimation error from $e = 25\%$ to $e = 40\%$ for the purpose of observing whether our proposed MBO-ML strategy will continue to produce consistent BER results under more severe conditions. When detecting a symbol vector at the receiver side, symbol detectors use the channel coefficients that affect the related symbol vector during the transmission. The accuracy of channel coefficients is one of the parameters that affects the performance of the symbol detectors. The more accurately we estimate the channel response on the receiving end, the higher symbol detection performance and accordingly the lower BER results are achieved as confirmed by Fig. 10. For instance, while the BER value achieved in 10 dB SNR by the proposed MBO-ML strategy for $e = 25\%$ is equal to 7.12×10^{-3}, the increase of estimation error from 25% to 30% and 30% to 40% cause the MBO-ML to obtain worse BER results like 1.12×10^{-2} and 2.54×10^{-2}, respectively. The related increments in the estimation error parameter cause certain amounts of performance loss in the considered symbol detection strategies. However, our proposed MBO-ML strategy manages to keep its superiority over the other ones for each e value due to its consistent symbol detection performance. For instance, while the BER values of MBO-ML and its nearest opponent named DBHS-ML are equal to 7.12×10^{-3} and 1.05×10^{-2} for 25% estimation error at 10 dB SNR, the related BER values reach 1.12×10^{-2}, 1.37×10^{-2} and 2.54×10^{-2}, 2.82×10^{-2} after 5% and 10% increments, respectively, in the value of estimation error.

If a general assessment is to be made in light of performance analysis carried out so far, it can be concluded that the proposed MBO-ML strategy has the capability of achieving quite close BER results to the optimum levels for the considered antenna configurations. For this reason, the proposed symbol detection strategy can be affectively utilized in the URLLC applications. Because ultra-reliability needs to be ensured in these applications, this can only be achieved with a high symbol detection performance. The ability to reach near-optimal BER achievement thanks to its powerful symbol detection performance makes the proposed MBO-ML strategy effectively usable in the URLLC applications.

3.1 Computational complexity analysis

In this section, the symbol detectors are evaluated with regard to their computational complexities. To this end, the computational cost created by each strategy in MIMO-FBMC/OQAM scheme is obtained by taking into account the number of complex multiplications [28] carried out during the process of detecting the symbol vectors. Following this, the computational complexity of our suggested MBO-ML strategy is compared with those of both classical ML and meta-heuristic-based modified ML procedures in a numerical way by calculating how much complexity improvements it achieves over the other detection procedures for each of the considered antenna structures. It should be emphasized that the ZF is the simplest symbol detection strategy known in the literature. For this reason, it will be no surprise that the related symbol detector has the lower computational complexity compared to the other ones considered in this paper. However, its poor symbol detection performance, which gets even worse as the number of antennas increase, makes the ZF quite weak against the other symbol detectors.

When detecting the symbol vectors via the ZF method, the pseudo-inverse matrix calculation, which leads to $2 \cdot \left(2 \cdot N_t^3 + N_t^2 \cdot N_r\right)$ multiplications, is needed. Therefore, the complexity expression of the ZF detector can be defined in the following way.

$$C_{ZF} = 2 \cdot \left(2 \cdot N_t^3 + N_t^2 \cdot N_r\right) \tag{29}$$

In the conventional ML strategy, the computational complexity is determined by the square operations and matrix multiplications carried out for the detection of symbol vectors. While $N_r \cdot Z^{N_t}$ multiplications are needed for the square operations, $(N_t \cdot N_r) \cdot Z^{N_t}$ multiplications are required for performing the matrix multiplications. Herewith, the eventual complexity of ML strategy will be equal to the sum of $N_r \cdot Z^{N_t}$ and $(N_t \cdot N_r) \cdot Z^{N_t}$ as follows:

$$C_{ML} = N_r \cdot Z^{N_t} + (N_t \cdot N_r) \cdot Z^{N_t} = N_r \cdot Z^{N_t} \cdot (N_t + 1) \tag{30}$$

If we return to Table 2, we will see that Z^{N_t} corresponds to the search complexity of ML. So, Eq. (30) can be reorganized in the following manner:

$$C_{ML} = C_{search} \cdot N_r \cdot (N_t + 1) \tag{31}$$

where the search complexity is represented by C_{search} for the ML strategy.

In the modern ML variants based on meta-heuristic algorithms, $N_r \cdot N_t$ multiplications are needed for each fitness evaluation. Additionally, the operations carried out for updating the parameters of meta-heuristic algorithms cause an extra β number of multiplications per fitness evaluation. According to this, it can be easily inferred that each fitness evaluation gives rise to $(N_r \cdot N_t + \beta)$ number of multiplications in the meta-heuristic-based ML strategies. So, if the total number of fitness evaluations, which is given as a search complexity for each of the meta-heuristic-based symbol detection strategies in Table 2, is multiplied by $(N_r \cdot N_t + \beta)$, the computational complexity expression that gives the aggregate number of multiplications required by any of the related heuristic approaches throughout the optimization process can be determined as follows:

$$C_{Heuristics} = C_{search} \cdot (N_r \cdot N_t + \beta) \tag{32}$$

From now on, all that needs to be done is to replace C_{search} in Eq. (32) with the search complexity of the corresponding heuristic approach. By doing so, the computational complexities belonging to the meta-heuristic-based ML schemes are acquired as follows:

$$C_{BPSO-ML} = S_{size} \cdot I_{\max} \cdot (N_r \cdot N_t + \beta) \tag{33}$$

$$C_{disABC-ML} = N_B \cdot L_{\max} \cdot (N_r \cdot N_t + \beta) \tag{34}$$

$$C_{DBHS-ML} = S_{\max} \cdot (N_r \cdot N_t + \beta) \tag{35}$$

$$C_{MBO-ML} = [(F-1) \cdot (W-H) + W] \cdot C \cdot T \cdot (N_r \cdot N_t + \beta) \tag{36}$$

Figure 11 shows the change in the computational complexity of our suggested MBO-ML scheme with respect to the antenna increment for varied C_{search} values.

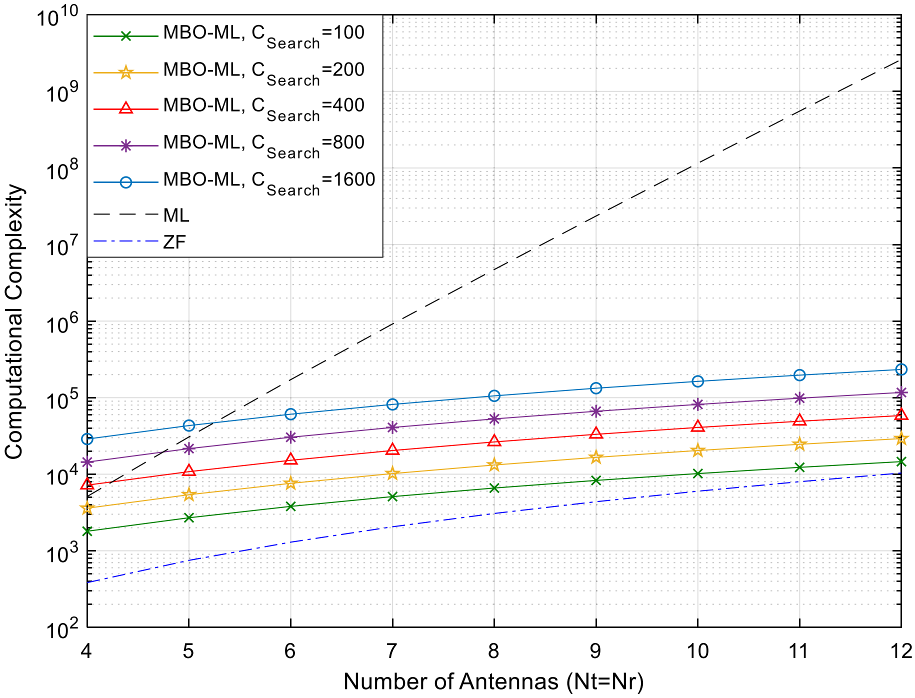

Fig. 11 The effect of increasing the number of antennas on the computational complexity of symbol detectors

In the same figure, the complexity curves of ZF and ML symbol detectors are also obtained for comparison. As it is quite clear in Fig. 11, very huge complexity enhancement is observed in the traditional ML with the increase in the number of antennas. On the other hand, as can be realized from the horizontality degree of its complexity curves acquired for various C_{search} values, each increment in the antenna number results in a very little growth in the MBO-ML complexity in comparison to the traditional ML method. Because of this, the difference between the MBO-ML and conventional ML complexities reaches to incredible levels at higher antenna configurations. Unsurprisingly, the complexity curve of the ZF symbol detector remains at the lower level than those of the other ones throughout the given range. However, it makes no sense for ZF detector to have the lowest computational complexity due to its poor symbol detection performance. To put it more generally, low computational complexity doesn't mean anything if the symbol detection performance is not good enough.

In order to acquire at what rate the complexity of benchmark methods is reduced by our proposed MBO-ML strategy, the following equation can be utilized:

$$Complexity\ Reduction\ (\%) = \left(\frac{C_\alpha - C_{MBO-ML}}{C_\alpha} \right) \times 100 \tag{37}$$

where C_α represents the complexity of the benchmark strategy handled for calculating the reduction rate achieved in its computational complexity.

In Table 4, we compare the computational complexity of our suggested MBO-ML procedure to that of classical ML scheme, numerically for each antenna configuration. The second and third columns of the related table contain numerical complexity values for ML and MBO-ML strategies, respectively, while the last column includes the complexity reductions attained by the suggested procedure in the conventional ML scheme. The numerical analysis carried out in Table 4 evidently demonstrates that the difference between the complexities of ML and MBO-ML strategies reaches to quite high levels in the case that the antenna structure is expanded from 4 × 4 to 8 × 8 due to the fact that each antenna addition brings about much more enhancement in the complexity of ML in comparison with our proposed strategy. Because of

Table 4 The comparison of MBO-ML and the conventional ML complexities for various antenna structures

Antenna configuration	ML complexity	MBO-ML complexity	Complexity reduction
4 × 4	$C_{ML} = N_r \cdot Z^{N_t} \cdot (N_t + 1)$ $= 4 \cdot 4^4 \cdot (4+1)$ $= 5120$	$C_{MBO-ML} = [(F-1) \cdot (W-H) + W]$ $\cdot C \cdot T \cdot (N_r \cdot N_t + \beta)$ $= [(13-1) \cdot (4-1) + 4] \cdot 5 \cdot 1 \cdot (4 \cdot 4 + 2)$ $= 3600$	29.688%
6 × 6	$C_{ML} = N_r \cdot Z^{N_t} \cdot (N_t + 1)$ $= 6 \cdot 4^6 \cdot (6+1)$ $= 172032$	$C_{MBO-ML} = [(F-1) \cdot (W-H) + W]$ $\cdot C \cdot T \cdot (N_r \cdot N_t + \beta)$ $= [(19-1) \cdot (4-1) + 4] \cdot 5 \cdot 2 \cdot (6 \cdot 6 + 2)$ $= 22040$	87.188%
8 × 8	$C_{ML} = N_r \cdot Z^{N_t} \cdot (N_t + 1)$ $= 8 \cdot 4^8 \cdot (8+1)$ $= 4718592$	$C_{MBO-ML} = [(F-1) \cdot (W-H) + W]$ $\cdot C \cdot T \cdot (N_r \cdot N_t + \beta)$ $= [(25-1) \cdot (4-1) + 4] \cdot 8 \cdot 2 \cdot (8 \cdot 8 + 2)$ $= 80256$	98.299%

Table 5 The comparison of MBO-ML to each of the other meta-heuristic-based ML schemes with regard to computational complexity

Antenna configuration	Technique	Search complexity (C_{Search})	Average BER for 12 dB SNR	Computational complexity	Complexity reduction (%)
4 × 4	BPSO-ML	$C_{search} = 260$	0.04564	$C_{BPSO\text{-}ML} = 4680$	58.4615
	MBO-ML	$C_{search} = 108$	0.04564	$C_{MBO\text{-}ML} = 1944$	
6 × 6	BPSO-ML	$C_{search} = 900$	0.03083	$C_{BPSO\text{-}ML} = 34{,}200$	67.7778
	MBO-ML	$C_{search} = 290$	0.03083	$C_{MBO\text{-}ML} = 11{,}020$	
8 × 8	BPSO-ML	$C_{search} = 2200$	0.03255	$C_{BPSO\text{-}ML} = 145{,}200$	74.7273
	MBO-ML	$C_{search} = 556$	0.03255	$C_{MBO\text{-}ML} = 36{,}696$	
4 × 4	disABC-ML	$C_{search} = 260$	0.02939	$C_{disABC\text{-}ML} = 4680$	51.5385
	MBO-ML	$C_{search} = 126$	0.02939	$C_{MBO\text{-}ML} = 2268$	
6 × 6	disABC-ML	$C_{search} = 900$	0.01451	$C_{disABC\text{-}ML} = 34{,}200$	60.5556
	MBO-ML	$C_{search} = 355$	0.01451	$C_{MBO\text{-}ML} = 13{,}490$	
8 × 8	disABC-ML	$C_{search} = 2200$	0.01143	$C_{disABC\text{-}ML} = 145{,}200$	67.9545
	MBO-ML	$C_{search} = 705$	0.01143	$C_{MBO\text{-}ML} = 46{,}530$	
4 × 4	DBHS-ML	$C_{search} = 260$	0.01755	$C_{DBHS\text{-}ML} = 4680$	40.3846
	MBO-ML	$C_{search} = 155$	0.01755	$C_{MBO\text{-}ML} = 2790$	
6 × 6	DBHS-ML	$C_{search} = 900$	0.007153	$C_{DBHS\text{-}ML} = 34{,}200$	53.4444
	MBO-ML	$C_{search} = 419$	0.007153	$C_{MBO\text{-}ML} = 15{,}922$	
8 × 8	DBHS-ML	$C_{search} = 2200$	0.006172	$C_{DBHS\text{-}ML} = 145{,}200$	63.8636
	MBO-ML	$C_{search} = 795$	0.006172	$C_{MBO\text{-}ML} = 52{,}470$	

this, it becomes possible to achieve 29.688%, 87.188% and 98.299% complexity reductions for 4 × 4, 6 × 6 and 8 × 8 antenna structures by using the MBO-ML as a symbol detector instead of conventional ML in the MIMO-FBMC/OQAM system.

Aside from that, we can figure out how much complexity gain is obtained by our suggested MBO-ML scheme in each meta-heuristic-based ML detector. For that purpose, the first thing we need to do is to determine how many fitness evaluations are needed by MBO-ML to attain the BER levels achieved by the other existing heuristic approaches with 260, 900 and 2200 fitness computations for 4 × 4, 6 × 6 and 8 × 8 antenna structures, respectively. By the way, keep in mind that the count of fitness evaluations gives the search complexity value (C_{search}) for the meta-heuristic-based ML detectors. Later on, the computational complexities of the proposed strategy for the determined numbers of fitness evaluations (C_{search} values) and those of the other symbol detectors for $C_{search} = 260$, 900 and 2200 are calculated by using Eq. (32). After all, the resulted computational costs are used in Eq. (37) to calculate the complexity reduction rates achieved in each benchmark strategy for the relevant antenna structures.

The results obtained after fulfilling the above-mentioned operations are given collectively in Table 5. The related table is divided horizontally into three parts. Each part contains three main rows each of which has two sub-rows. In the first part of Table 5, one to one comparisons are made between the MBO-ML and BPSO-ML for the related three antenna structures in the three separate main rows while the comparisons with the disABC-ML and DBHS-ML are carried out in the second and third parts, respectively. For instance, in the first main row of the first part, the MBO-ML scheme is compared with BPSO-ML in terms of computational complexity for 4 × 4 MIMO configuration. In this comparison carried out in the related row, it is determined that 108 fitness computations are enough for MBO-ML to acquire 0.04564 BER value, which can be barely achieved by BPSO-ML with 260 number of fitness evaluations. By using these C_{search} values in Eq. (32), the BPSO-ML and MBO-ML complexities are calculated as $C_{BPSO-ML} = 4680$ and $C_{MBO-ML} = 1944$. Finally, the resulting complexity values are utilized to obtain 58.4615% complexity reduction rate for the related antenna configuration by using Eq. (37). The remaining rows of Table 5 are obtained by repeating these operations carried out for the first main row. As evidently seen in Table 5, our suggested MBO-ML strategy achieves significantly high complexity improvements in each of the other meta-heuristic-based detection procedures. The complexity reduction rates, which are already quite high for 4 × 4 antenna configuration, get even higher with the enlargement of MIMO structure from 4 × 4 to 8 × 8. Therefore, the highest complexity reductions are obtained for 8 × 8 antenna structure. For instance, while 40.3846%, 51.5385% and 58.4615% complexity reduction rates are provided by the proposed strategy in the DBHS-ML, disABC-ML and BPSO-ML, respectively, for 4 × 4 antenna configuration, these complexity improvements rise up to 63.8636%, 67.9545% and 74.7273% in the case that the antenna structure is expanded from 4 × 4 to 8 × 8.

4 Conclusion

In this paper, an efficient MBO variant, in which the neighbor solutions are produced in a systematic way by using the cyclic bit flipping mechanism, was embedded to the conventional ML method as a symbol optimizer to develop a state-of-art symbol detecting procedure called MBO-ML by which the near ML performance can be reached with substantially reduced computational complexity. After that, the symbol detection performance of our new approach was tested in MIMO-FBMC/OQAM scheme by measuring its BER accomplishments for 4 × 4, 6 × 6 and 8 × 8 MIMO structures. In this performance analysis, in addition to the widely known conventional symbol detection methods like ML and ZF, three advanced ML versions called disABC-ML, DBHS-ML and BPSO-ML are also used for comparison. In the simulations carried out for evaluating the considered methods in terms of symbol detecting capability, our suggested MBO-ML has been the method that most closely approaches the conventional ML in terms of BER achievement in the MIMO-FBMC/OQAM system for each antenna configuration. It established a clear superiority over both the classical ZF and the other three modified ML versions by reaching lower BER levels due to its greater symbol detection performance. In addition to its predominant BER achievement, the suggested MBO-ML detector also has a huge advantage with regard to computational complexity. While it achieves 29.688% complexity reduction in the conventional ML for 4 × 4 antenna structure, the increase of MIMO configuration from 4 × 4 to 8 × 8 makes that reduction rate to reach a very high value like 98.299%. Apart from this, our proposed strategy has the capability of achieving minimum 40.3846% complexity reduction over even its nearest opponent called DBHS-ML for the considered MIMO configurations.

Abbreviations

5G	Fifth generation
IoT	Internet of things
MTC	Machine-type communication
LTE	Long-term evolution
OFDM	Orthogonal frequency division multiplexing
FBMC/OQAM	Filter bank multicarrier/offset quadrate amplitude modulation
MIMO	Multiple-input multiple-output
BER	Bit error rate
ML	Maximum likelihood
ZF	Zero forcing
MBO	Migrating birds optimization
DBHS	Discrete binary harmony search
disABC	Discrete artificial bee colony
BPSO	Binary particle swarm optimization
ACO	Ant colony optimization
BSA	Back-tracking search algorithm
NOMA	Non-orthogonal multiple access
BLAST	Bell Laboratories layered space time
SDMA	Space division multiple access
DE	Differential evolution
SNR	Signal-to-noise ratio
$a_{m,n}$	Real-valued symbol
$h_{m,n}$	Channel coefficients
$u_{m,n}$	Intrinsic interference
$n_{m,n}$	Noise component
m	Subcarrier index
n	Time index
$y_{m,n}$	Demodulated signal
N_t	Number of transmit antennas
N_r	Number of receive antennas
i	Index of transmit antenna
j	Index of receive antenna
D	Number of time samples
M	Number of subcarriers
N	Number of symbols
$\mathbf{G} \in \mathbb{C}^{\mathrm{D} \times \mathrm{MN}}$	Transmit matrix
$\mathbf{g}_{m,n} \in \mathbb{C}^{\mathrm{D} \times 1}$	Transmit vectors
vec{}	Transforms the matrix in its parenthesis into a vector
$\mathbf{a} \in \mathbb{C}^{\mathrm{MN} \times 1}$	Real-valued symbol vector
$\mathbf{s} \in \mathbb{C}^{\mathrm{D} \times 1}$	Transmission signal
$\mathbf{H} \in \mathbb{C}^{\mathrm{D} \times \mathrm{D}}$	Channel matrix
$\mathbf{r} \in \mathbb{C}^{\mathrm{D} \times 1}$	Arriving signal
$\mathbf{n}$	White Gaussian noise added to the received signal
P_n	Noise power
$\mathbf{I}_\mathrm{D}$	$D \times D$ Identity matrix
$\mathrm{CN}(0,\ P_n\,\mathbf{I}_\mathrm{D})$	Complex Gaussian distribution with zero mean and $P_n\,\mathbf{I}_D$ variance
$\mathbf{C} \in \mathbb{C}^{\mathrm{MN} \times \frac{\mathrm{MN}}{2}}$	Precoding matrix
$\mathbf{x} \in \mathbb{C}^{\frac{\mathrm{MN}}{2} \times 1}$	QAM symbol matrix
$(\,)^\mathrm{H}$	Hermitian of a matrix
$x_{m,n}$	Transmitted QAM symbol vector
$\hat{y}_{m,n}$	Received symbol vector
Z	Modulation order
$x^*_{m,n}$	Optimum symbol vector
arg min { }	Finds the optimum parameter minimizing the value of an expression existing in its parenthesis
$H_{m,n}$	Channel coefficient matrix affecting the symbol vector $x_{m,n}$
$\hat{H}_{m,n}$	Imperfect channel coefficients
θ	Complex Gaussian variable with unit variance and zero mean
e	The rate of estimation error
k	The length of binary information carried by each complex-valued QAM symbol
F	The aggregate number of birds
f	Bird index
$x_\mathrm{f}^{(i)}$	Complex-valued QAM symbol sequences expressed as bird positions
$b_\mathrm{f}^{(j)}$	Equivalent binary bit strings of the complex-valued QAM symbol sequences expressed as bird positions
W	The aggregate number of neighbor solutions
w	Neighbor solution index
$G_\mathrm{f}^{(j)}$	The vector representing the *f*th bird
$Q_\mathrm{f,w}^{(j)}$	The vector representing the *w*th neighbor of the *f*th bird
$p_\mathrm{f,w}^{(j)}$	The position of *w*th neighbor belonging to the *f*th bird

$\mathrm{fit}()$	Calculates the fitness values of the position vectors
μ_{f}	Flipping index of the *f*th bird
$\mu_{\mathrm{f,w}}$	Flipping index belonging to the *w*th neighbor of the *f*th bird
$\mathrm{flip}\left(G_{\mathrm{f}}^{(j)}\right)_{\mu_{\mathrm{f}}}$	Carries out the flipping operation on the element that exists in the μ_f th dimension of the vector $G_{\mathrm{f}}^{(j)}$
$\mathrm{mod}(\ )$	Performs the modulo operation
$x_{\mathrm{f,w}}^{(i)}$	Equivalent complex-valued QAM symbol sequences of the binary neighbor position vectors $p_{f,w}^{(j)}$
H	The number of neighbor solutions to be shared with a single follower bird
T	Number of tours
C	Maximum number of cycles
S_{size}	Swarm size for BPSO-ML
$I_{\max}$	Maximum number of iterations for BPSO-ML
N_{B}	Number of bees for disABC-ML
$L_{\max}$	Maximum number of loops for disABC-ML
P_{size}	Size of population for DBHS-ML
$S_{\max}$	Maximum number of searches for DBHS-ML
β	Additional multiplications parameter
C_{search}	Search complexity
C_{ML}	Computational complexity of ML
$C_{\mathrm{Heuristics}}$	Computational complexity of heuristic approaches
$C_{\mathrm{BPSO\text{ - }ML}}$	Computational complexity of BPSO-ML
$C_{\mathrm{disABC\text{ - }ML}}$	Computational complexity of disABC-ML
$C_{\mathrm{DBHS\text{ - }ML}}$	Computational complexity of DBHS-ML
$C_{\mathrm{MBO\text{ - }ML}}$	Computational complexity of MBO-ML

Author contributions

Formal analysis, software, methodology, and writing—review and editing were carried out by Şakir Şimşir.

Funding

No funding was received for conducting this study.

Data availability

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Competing interests

The author has no competing interests to declare.

Received: 17 July 2024 Accepted: 1 October 2024
Published online: 07 October 2024

References

1. S.-Y. Lien, S.-L. Shieh, Y. Huang, B. Su, Y.-L. Hsu, H.-Y. Wei, 5G new radio: Waveform, frame structure, multiple access, and initial access. IEEE Commun. Mag. **55**(6), 64–71 (2017)
2. M. Agiwal, A. Roy, N. Saxena, Next generation 5G wireless networks: a comprehensive survey. IEEE Commun. Surv. Tutor. **18**(3), 1617–1655 (2016)
3. M. Van Eeckhaute, A. Bourdoux, P. De Doncker, F. Horlin, Performance of emerging multi-carrier waveforms for 5G asynchronous communications. EURASIP J. Wirel. Commun. Netw. **2017**, 1–15 (2017)
4. Y. Mehmood, C. Görg, M. Muehleisen, A. Timm-Giel, Mobile M2M communication architectures, upcoming challenges, applications, and future directions. EURASIP J. Wirel. Commun. Netw. **2015**, 1–37 (2015)
5. L.J. Cimini, Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing. IEEE Trans. Commun. **33**(7), 665–675 (1985)
6. Z. Zhao, M. Schellmann, X. Gong, Q. Wang, R. Böhnke, Y. Guo, Pulse shaping design for OFDM systems. EURASIP J. Wirel. Commun. Netw. **2017**, 1–25 (2017)
7. P. Siohan, C. Siclet, N. Lacaille, Analysis and design of OFDM/OQAM systems based on filterbank theory. IEEE Trans. Signal Process. **50**(5), 1170–1183 (2002)
8. B. Farhang-Boroujeny, Filter bank multicarrier modulation: a waveform candidate for 5G and beyond. Adv. Electr. Eng. **2014**, 1–25 (2014)
9. B. Farhang-Boroujeny, OFDM versus filter bank multicarrier. IEEE Signal Process. Mag. **28**(3), 92–112 (2011)
10. R. Nissel, S. Schwarz, M. Rupp, Filter bank multicarrier modulation schemes for future mobile communications. IEEE J. Sel. Areas Commun. **34**(10), 1768–1782 (2017)
11. R. Zakaria, D. Le Ruyet, On maximum likelihood MIMO detection in QAM-FBMC systems, In: *21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications* (2010), pp. 183–187.
12. A.I. Perez-Neira, M. Caus, R. Zakaria, D. Le Ruyet, E. Kofidis, M. Haardt, X. Mestre, Y. Cheng, MIMO signal processing in offset-QAM based filter bank multicarrier systems. IEEE Trans. Signal Process. **64**(21), 5733–5762 (2016)
13. R. Zakaria, D. Le Ruyet, A novel filter-bank multicarrier scheme to mitigate the intrinsic interference: Application to MIMO systems. IEEE Trans. Wirel. Commun. **11**(3), 1112–1123 (2012)

14. R. Nissel, J. Blumenstein, M. Rupp, Block frequency spreading: A method for low-complexity MIMO in FBMC-OQAM, In: *IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)* (2017), pp. 1–5.
15. Q.H. Spencer, A.L. Swindlehurst, M. Haardt, Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels. IEEE Trans. Signal Process. **52**(2), 461–471 (2004)
16. M.-X. Chang, W.-Y. Chang, Maximum-likelihood detection for MIMO systems based on differential metrics. IEEE Trans. Signal Process. **65**(14), 3718–3732 (2017)
17. X. Zhu, R.D. Murch, Performance analysis of maximum likelihood detection in a MIMO antenna system. IEEE Trans. Commun. **50**(2), 187–191 (2002)
18. N. Taşpınar, Ş Şimşir, An efficient SLM technique based on migrating birds optimization algorithm with cyclic bit flipping mechanism for PAPR reduction in UFMC waveform. Phys. Commun. **43**, 1–13 (2020)
19. X. Kong, L. Gao, H. Ouyang, S. Li, A simplified binary harmony search algorithm for large scale 0–1 knapsack problems. Expert Syst. Appl. **42**(12), 5337–5355 (2015)
20. X. Cheng, D. Liu, S. Feng, Q. Pan, H. Fang, PTS based on DisABC algorithm for PAPR reduction in OFDM systems. Electron. Lett. **54**(6), 397–398 (2018)
21. S. Mirjalili, A. Lewis, S-shaped versus V-shaped transfer functions for binary particle swarm optimization. Swarm Evolut. Comput. **9**, 1–14 (2013)
22. A. Prasad, A. Benjebbour, O. Bulakci, K.I. Pedersen, N.K. Pratas, M. Mezzavilla, Agile radio resource management techniques for 5G new radio. IEEE Commun. Mag. **55**(6), 62–63 (2017)
23. M. Mandloi, V. Bhatia, A low-complexity hybrid algorithm based on particle swarm and ant colony optimization for large-MIMO detection. Expert Syst. Appl. **50**, 66–74 (2016)
24. M.N. Seyman, Symbol detection based on back tracking search algorithm in MIMO-NOMA systems. Comput. Syst. Sci. Eng. **40**(2), 795–804 (2022)
25. L. Li, W. Meng, S. Ju, A novel artificial bee colony detection algorithm for massive MIMO system. Wirel. Commun. Mob. Comput. **16**(17), 3139–3152 (2016)
26. H.U. Rehman, S.I. Shah, I. Zaka, J. Ahmad, An MBER–BLAST algorithm for OFDM–SDMA communication using particle swarm optimization. Int. J. Commun. Syst. **24**(2), 185–201 (2011)
27. M.N. Seyman, N. Taşpınar, Symbol detection using the differential evolution algorithm in MIMO-OFDM systems. Turk. J. Electr. Eng. Comput. Sci. **21**(2), 373–380 (2013)
28. A.A. Khan, S. Bashir, M. Naeem, S.I. Shah, X. Li, Symbol detection in spatial multiplexing system using particle swarm optimization meta-heuristics. Int. J. Commun. Syst. **21**(12), 1239–1257 (2008)
29. L. Wei, J. He, Z. Guo, Z. Hu, A multi-objective migrating birds optimization algorithm based on game theory for dynamic flexible job shop scheduling problem. Expert Syst. Appl. **227**, 1–15 (2023)
30. T. Duman, E. Duman, Solving a new application of asymmetric TSP by modified migrating birds optimization algorithm. Evolut. Intell. **17**, 1697–1713 (2024)
31. Z. Li, M.N. Janardhanan, Q. Tang, Multi-objective migrating bird optimization algorithm for cost-oriented assembly line balancing problem with collaborative robots. Neural Comput. Appl. **33**, 8575–8596 (2021)
32. J. Cao, Z. Guan, L. Yue, S. Ullah, R.A.K. Sherwani, A bottleneck degree-based migrating birds optimization algorithm for the PCB production scheduling. IEEE Access **8**, 209579–209593 (2020)
33. Z. Zhang, Q. Tang, D. Han, Z. Li, Multi-manned assembly line balancing with sequence-dependent set-up times using an enhanced migrating birds optimization algorithm. Eng. Optim. **55**(7), 1243–1262 (2023)
34. R. Badgujar, P. Deore, MBO-SVM-based exudate classification in fundus retinal images of diabetic patients. Comput. Methods Biomech. Biomed. Eng. Imaging Visual. **7**(2), 195–206 (2019)
35. G. Niranjana, A. Poongodai, K.L.S. Soujanya, Biological inspired self-organized secure autonomous routing protocol and secured data assured routing in WSN: Hybrid EHO and MBO approach. Int. J. Commun. Syst. **35**(4), 1–16 (2022)
36. V. Tongur, E. Ertunc, M. Uyan, Use of the migrating birds optimization (MBO) algorithm in solving land distribution problem. Land Use Policy **94**, 1–9 (2020)
37. C. Wang, E.K.S. Au, R.D. Murch, W.H. Mow, R.S. Cheng, V. Lau, On the performance of the MIMO zero-forcing receiver in the presence of channel estimation error. IEEE Trans. Wirel. Commun. **6**(3), 805–810 (2007)
38. E. Duman, M. Uysal, A.F. Alkaya, Migrating birds optimization: a new metaheuristic approach and its performance on quadratic assignment problem. Inf. Sci. **217**, 65–77 (2012)

Publisher's Note