



MIMO-FBMC/OQAM sisteminde ayırık guguk kuşu arama algoritması kullanılarak sembol vektörlerinin algılanması

Detection of symbol vectors using discrete cuckoo search algorithm in MIMO-FBMC/OQAM system

Sakir Şimşir^{1,*} 

¹ Nevşehir Hacı Bektaş Veli Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü, 50300, Nevşehir Türkiye

Öz

Çoklu-giriş çoklu-çıkış (MIMO) teknolojisinin filtre bankası çoklu taşıyıcı/ofset dördün genlik modülasyonuna (FBMC/OQAM) uygulanması sonucu elde edilen MIMO-FBMC/OQAM sistemi, kablosuz haberleşmede karşılaşılan birçok soruna çare olabilecek önemli özellikleri bünyesinde barındırmaktadır. Öte yandan, MIMO-FBMC/OQAM sisteminde sembol vektörlerini alıcıda en doğru şekilde algılayabilen etkili bir sembol dedektörüne ihtiyaç duyulmaktadır. Geleneksel maksimum olasılık (ML) dedektörü, kusursuz bir sembol algılama performansına sahiptir. Ancak alıcıya ulaşan sinyal üzerinden hangi sembol vektörünün iletilmiş olduğuna karar verirken, oluşabilecek bütün sembol kombinasyonlarını denemesi, ML dedektörünün karmaşıklığını oldukça yüksek seviyelere çıkarmaktadır. İdeal bir sembol dedektöründen, sembol vektörlerini minimum hata ile tespit etmesi ve bu işi sistem karmaşıklığını çok fazla artırmadan gerçekleştirmesi beklenir. Bu çalışmada, söz konusu beklentiye uygun bir sembol dedektörü geliştirmek için klasik ML dedektörü modifiye edilmiştir. Araştırma uzayında yer alan olası bütün sembol kombinasyonlarını tek tek deneyerek doğru sonuca ulaşmak yerine, sembol vektörleri optimize edilerek, optimum çözüme en kısa yoldan ulaşma yoluna gidilmiştir. Bu amaçla, birçok alanda yaygın olarak kullanılan guguk kuşu arama (CS) algoritmasının etkili bir ayırık versiyonu (DCS) kullanılmıştır. Söz konusu modifikasyon sonucu elde edilen DCS-ML stratejisi ile optimum çözüme oldukça yakın sonuçlar elde edilmekle kalmayıp, klasik ML dedektörünün karmaşıklığı büyük oranda düşürülmüştür.

Anahtar kelimeler: MIMO, Sembol algılama, FBMC/OQAM, Maksimum olasılık, Guguk kuşu arama algoritması

1 Giriş

Dikgen frekans bölmeli çoğullama (OFDM) [1, 2], uzun vadeli evrim (LTE)/4G mobil sistemlerde yıllarca temel dalga formu olarak kullanıldıktan sonra, sade yapısı ve çok yollu kanalların negatif etkilerine karşı dayanıklılığı sebebiyle, 5G sistemlere de adapte edilmiştir. Ancak döngüsel önek kullanımı ve bant dışı radyasyona sebebiyet veren dikdörtgen darbe yapısı, OFDM'nin spektral

Abstract

MIMO-FBMC/OQAM system, which is obtained by applying the multiple-input multiple output (MIMO) technology to the filter bank multicarrier/offset quadrature amplitude modulation (FBMC/OQAM) incorporates important features that can solve many problems encountered in wireless communication. On the other hand, an efficient symbol detector that can detect the symbol vectors at the receiver in the most accurate way is required in the MIMO-FBMC/OQAM system. The conventional maximum likelihood detector has a flawless symbol detection performance. However, when deciding which symbol vector has been transmitted based on the signal reaching the receiver, trying all of the possible symbol combinations enhances the complexity of ML to very high levels. It is expected from an ideal symbol detector to detect the symbol vectors with minimum error and do this task without increasing the system complexity too much. In this study, the classical ML detector was modified to develop a symbol detector compatible with the aforementioned expectation. Instead of reaching the correct result by trying all of the possible symbol combinations existing in the search space one by one, it was tried to reach the optimum solution in the shortest way by optimizing the symbol vectors. To this end, an efficient discrete version (DCS) of the cuckoo search (CS) algorithm, which is commonly used in many fields, was employed. With the DCS-ML strategy obtained as a result of the aforementioned modification, not only the results that are very close to the optimal solution were achieved, but also the complexity of the classical ML detector was substantially reduced.

Keywords: MIMO, Symbol detection, FBMC/OQAM, Maximum likelihood, Cuckoo search algorithm

verimliliğini oldukça düşürmektedir. Bunun haricinde, dikliğin korunabilmesi için farklı kullanıcılar arasındaki senkronizasyonun alıcıda iyi bir şekilde ayarlanması gerekmektedir. Zamanla-değişen kanalların sebep olduğu taşıyıcı frekans kaymalarına (CFO) karşı yüksek hassasiyet, OFDM dalga formunun bir diğer önemli eksikliği olarak göze çarpmaktadır [1, 2]. Gelecek nesil mobil iletişim teknolojilerinin veri hızı, gecikme, senkronizasyon ve

* Sorumlu yazar / Corresponding author, e-posta / e-mail: sakirsimsir@nevsehir.edu.tr (Ş. Şimşir)
Geliş / Received: 14.05.2024 Kabul / Accepted: 20.08.2024 Yayınlanma / Published: 15.10.2024
doi: 10.28948/ngumuh.1484055

güvenlik bakımından çok daha iyi olması beklendiği için, yukarıda bahsedilen dezavantajlarından dolayı OFDM'nin yerini daha gelişmiş bir dalga formuna bırakması kaçınılmaz hale gelecektir. Araştırmacılar uzun süredir bu kaçınılmaz değişime hazırlıklı olmak için çalışmalarını sürdürmekte olup, bu süreçte çeşitli dalga formu adayları geliştirmişlerdir [3]. Filtre bankası çoklu taşıyıcı/ofset dördün genlik modülasyonu (FBMC/OQAM) [4, 5], kablosuz haberleşme teknolojisinin hızla artan ihtiyaçlarını karşılamak amacıyla geliştirilmiş olan dalga formu adaylarının en önemli temsilcilerinden bir tanesidir. Bu yeni dalga formunda, OFDM'nin sahip olduğu zayıflıklar büyük oranda giderilmiştir. Örneğin FBMC/OQAM sisteminde kullanılan dikdörtgen olmayan darbe şekli, bant dışı radyasyonun büyük ölçüde azaltılmasına olanak sağlamaktadır. Bunun haricinde, geleneksel OFDM sisteminde kaçınılmaz bir işlem olan ve spektral verimsizliğe yol açan döngüsel önek ekleme işlemine, FBMC/OQAM sisteminde ihtiyaç duyulmamaktadır. Ayrıca, FBMC/OQAM sisteminde gerçekleştirilen filtreleme işlemi ile alt taşıyıcıların izole edilmesi, OFDM sisteminin en temel zayıflıkları arasında yer alan senkronizasyon hatalarına karşı direnç sağlamaktadır [4, 5]. Tüm bu avantajların yanı sıra, çoklu-giriş çoklu-çıkış (MIMO) teknolojisinden yararlanarak FBMC/OQAM sisteminin mevcut potansiyelini daha da yukarıya çekmek mümkündür [6, 7]. Çoklu anten yapısı kullanılarak gerçekleştirilen veri iletişimi, çok yollu kanalın sönmülme etkilerine karşı sistem dayanıklılığını artırmakla kalmayıp, önemli bir kapasite artışı sağlanmasına yol açmaktadır. Kablosuz kanalın sönmülme etkilerinin azaltılması sayesinde sistemin bit hata oranı (BER) azalırken, söz konusu kapasite artışı nedeniyle de daha yüksek veri oranlarında iletişim gerçekleştirilebilmektedir [6, 7].

Öte yandan, MIMO-FBMC/OQAM sisteminin alıcı kısmında iletilen sembollerin minimum hata ile yeniden elde edilebilmesi için etkili bir sembol dedektörüne ihtiyaç duyulmaktadır. Söz konusu sistem için hayati öneme sahip olan sembol algılama işlemi doğru bir şekilde yerine getirilmediği takdirde, yukarıda bahsedilen üstün özelliklerin hiçbir anlamı kalmayacaktır. Sıfır zorlama (ZF) ve maksimum olasılık (ML) algoritmaları, bu amaç için geliştirilmiş olan ve literatürde oldukça yaygın bir şekilde kullanılan en popüler iki sembol algılama stratejisidir [8, 9]. Bir sembol dedektöründen beklenen, sistem karmaşıklığını çok fazla artırmadan tatmin edici bir sembol algılama performansı sergilemesidir. Daha açık bir ifadeyle, ideal bir sembol dedektörü için yüksek performans ya da düşük karmaşıklık tek başına yeterli değildir. Hem performans hem de karmaşıklık kriterlerinin aynı anda karşılanması gerekir. Ne ZF ne de ML algoritması bu iki kriteri aynı anda sağlamaktadır. Söz konusu algoritmalar, bazı avantajlarının yanı sıra çeşitli dezavantajlara da sahiptirler. Örneğin, düşük karmaşıklığa sahip olması ve kolay uygulanabilir yapısı ZF sembol dedektörünün olumlu yönleri olarak öne çıkarken, nispeten zorlu kanal koşullarında göstermiş olduğu zayıf performans önemli bir dezavantaj olarak göze çarpmaktadır. Bunun haricinde, ZF dedektörünün sembol algılama performansı, anten sayısındaki artışla birlikte daha da

kötüleşmektedir [8]. ML ise, optimum sembol vektörlerine ulaşma yeteneğine sahip yegâne sembol dedektörü olarak bilinmektedir. Diğer bir ifade ile ML, en iyi çözüme ulaşmayı garanti eden bir yöntemdir. Bu nedenle, ilgili yöntemin sembol algılama performansı, literatürde geliştirilen sembol dedektörleri için üst sınır olarak kabul edilmektedir. Ancak, ML dedektörünün bu kusursuz sembol algılama performansı, son derece yüksek bir araştırma maliyetini de beraberinde getirmektedir. ML dedektörünün herhangi bir iletişim teknolojisinde pratik olarak kullanılmayacak kadar yüksek bir karmaşıklığa sahip olmasının temel nedeni, sembol algılama esnasında kullanılmış olduğu kapsamlı arama prosedürüdür. Söz konusu arama prosedürüne göre, sönmümlü kanal tarafından belirli ölçüde bozulmaya uğrayarak alıcıya ulaşan sembol vektörü, alıcıya iletilme olasılığı olan sembol kombinasyonlarından her biri ile Öklid mesafesi bakımından karşılaştırılmaktadır. Sembol vektörü uzunluğunun anten sayısına eşit olduğu ve anten sayısı ile modülasyon seviyesi arttıkça olası sembol kombinasyonu sayısının da artacağı unutulmamalıdır. Olası sembol kombinasyonlarından her birinin alınan sembol vektörüne olan Öklid mesafesi hesaplandıktan sonra, minimum Öklid mesafesinin elde edildiği sembol kombinasyonu, maksimum iletilme olasılığı olan optimum sembol vektörü olarak seçilir. Olası sembol kombinasyonlarından her biri için gerçekleştirilen Öklid hesabı, ML yönteminin karmaşıklığının oldukça yüksek seviyelere çıkmasına sebep olmaktadır. Ayrıca, anten sayısı ve modülasyon seviyesindeki artışla birlikte araştırma uzayında üstel bir genişleme meydana geleceği için, ML algoritmasının karmaşıklığı çok daha yüksek seviyelere ulaşacaktır [9].

Diğer taraftan, kapsamlı arama prosedürü yerine, bir optimizasyon algoritması ile sembol vektörleri optimize edilerek, ayrık uzaydaki optimum çözüm arayışı daha sistematik bir şekilde yerine getirilecek olursa, ML dedektörü çok daha düşük işlem yükü ile optimum çözüme yakınsama kabiliyeti kazanacaktır. Burada asıl farkı yaratacak olan şey, sembol vektörlerinin optimizasyonu için geleneksel ML yöntemine entegre edilecek olan optimizasyon algoritmasının yakınsama performansidir. Birçok alanda yaygın olarak kullanılan popülasyon tabanlı meta-sezgisel algoritmalar, başlangıç koşulundan bağımsız olarak optimum çözüme en etkili şekilde ulaşmalarını sağlayan paralel araştırma yeteneklerinden dolayı, söz konusu optimizasyon işlemini yerine getirmek için en doğru tercih olacaktır. Bu çalışmada, sembol vektörlerinin optimizasyonu direkt olarak ayrık uzayda gerçekleştirilmektedir. Bu amaçla, her bir sembol vektörü, QAM sembol dizisinden ikili bit dizisi haline dönüştürülerek ayrık optimizasyon işlemi için uyumlu hale getirilir. Diğer bir ifadeyle, karmaşık bir sayı ile temsil edilen her bir QAM sembolü, kendisinin eşdeğeri olan ikili sayıya dönüştürülür. Bu şekilde QAM sembol vektörlerinin her biri, karmaşık sayı dizisi halinden ikili bit dizisi haline dönüştürülüp ayrık uzayda optimize edilmeye hazır hale getirilir. Elde edilen ikili formdaki sembol vektörlerini, meta-sezgisel algoritmaların ayrık versiyonlarını kullanarak optimize etmek mümkündür. Bu çalışmada, guguk kuşu arama

algoritmasının yakın zamanda geliştirilmiş olan etkili bir ayrık versiyonu (DCS) [10], ikili araştırma uzayında sembol optimizasyonu gerçekleştirmek üzere klasik ML dedektörüne entegre edilmiştir. İlgili entegrasyon işleminin ardından, DCS-ML olarak isimlendirilen gelişmiş bir ML varyantı elde edilmiştir. MIMO-FBMC/OQAM sisteminde, yeni geliştirilen DCS-ML stratejisinin kullanılması durumunda sembol vektörlerinin ne kadar doğru bir şekilde algılanabileceği hakkında fikir sahibi olabilmek amacıyla, farklı anten konfigürasyonları için BER analizi gerçekleştirilmiştir. İlgili performans analizinde, ZF ve ML gibi literatürde oldukça yaygın bir şekilde kullanılan klasik sembol algılama metodlarının yanı sıra, ayrık ikili harmoni arama (DBHS) algoritması [11] vasıtasıyla sembol optimizasyonu gerçekleştirilen DBHS-ML isimli meta-sezgisel tabanlı bir ML varyantı karşılaştırma amaçlı kullanılmıştır.

Literatürde çeşitli iletim teknolojilerinde meta-sezgisel algoritmalarla yararlanarak sembol algılama işleminin gerçekleştirildiği bazı çalışmalara rastlamak mümkündür. [12]'de, geri-izleme arama algoritması (BSA) tabanlı ML dedektörü geliştirilerek MIMO-NOMA (NOMA: dikgen olmayan çoklu erişim) sistemine uygulanmıştır. Daha sonra, önerilen dedektörün sembol algılama performansı BER grafikleri vasıtasıyla analiz edilmiştir. [13]'te, ağırlık vektörlerinin parçacık sürü optimizasyon (PSO) algoritması ile optimize edildiği minimum bit hata oranı (MBER) dedektörü, Bell Laboratuvarları katmanlı uzay zaman (BLAST) ile birleştirilerek MBER-BLAST isimli yeni bir sembol algılama algoritması OFDM-SDMA (SDMA: uzay bölmeli çoklu erişim) sistemi için geliştirilmiştir. [14]'te, PSO algoritmasının hem sürekli hem de ikili versiyonları ile desteklenmiş birkaç adet sembol algılama algoritması MIMO haberleşme sistemleri için önerilmiştir. [15]'te, geniş-MIMO sistemler için düşük-karmaşıklık bir sembol dedektörü geliştirmek amacıyla PSO ve karınca koloni optimizasyon (ACO) algoritmaları melezlenerek yeni bir hibrit algoritma elde edilmiştir. İlgili algoritmada, PSO'dan hız ölçütü, ACO'dan ise mesafe ölçütü konsepti kullanılarak yeni bir olasılık ölçütü dizayn edilmiştir. [16]'da, masif MIMO sistemlerin alıcı kısmında sembol vektörlerini optimize etmek için yapay arı kolonisi (ABC) algoritmasından faydalanılmıştır. Söz konusu çalışmada yazarlar, ABC algoritmasının başlangıç çözüm vektörleri olarak lineer algılama sonuçlarını kullanmışlardır. [17]'de, MIMO-OFDM sisteminde sembol vektörlerinin algılanması için diferansiyel gelişim (DE) algoritmasına dayanan modern bir ML dedektörü geliştirilmiştir. [18]'de, darbe genlik modülasyonu (PAM) sisteminde işlem yükü bakımından verimli bir sembol algılama işlemi gerçekleştirebilmek için, ML dedektöründe kapsamlı arama stratejisi yerine genetik algoritma (GA) tabanlı sembol optimizasyon prosedürü kullanılmıştır.

Makalenin literatüre sağladığı önemli katkılar aşağıda listelenmiştir:

1. Ayrık guguk kuşu arama (DCS) [10] algoritması, literatürde ilk defa sembol algılama problemine adapte edilmiştir.

2. DCS algoritması geleneksel ML yöntemine entegre edilerek, MIMO-FBMC/OQAM sistemi için DCS-ML isimli yeni bir sembol algılama stratejisi geliştirilmiştir.
3. Kapsamlı arama prosedürü yerine DCS tabanlı sembol optimizasyon stratejisi kullanılarak klasik ML yönteminin karmaşıklığı büyük oranda düşürülmüştür.
4. Önerilen DCS-ML stratejisi hem klasik ZF dedektörünü hem de DBHS-ML isimli gelişmiş ML stratejisini BER performansı bakımından geride bırakmıştır.

2 MIMO-FBMC/OQAM sistem açıklaması

FBMC/OQAM sistemi, en sade formunda bile geleneksel OFDM sisteminde yaşanan birçok probleme etkili çözümler sunabilme kapasitesine sahiptir. Bununla birlikte, MIMO teknolojilerinden yararlanarak FBMC/OQAM sisteminin daha yüksek veri hızlarında iletişim kurabilmesi ve çok yollu sönmüleme etkilerine karşı daha dayanıklı hale gelmesi sağlanabilmektedir. Bu bölümde, MIMO-FBMC/OQAM sisteminde çoklu anten yapısı üzerinden veri iletişiminin nasıl gerçekleştirildiği basit ve anlaşılır bir şekilde açıklanmıştır [6].

Gerçek değerli $a_{m,n}$ sembolünün FBMC/OQAM sisteminde tek bir anten üzerinden iletilmesi durumunda, alıcıda elde edilen sinyal aşağıdaki gibi tanımlanmaktadır.

$$y_{m,n} \approx h_{m,n} \cdot (a_{m,n} + j \cdot u_{m,n}) + n_{m,n} \quad (1)$$

Burada $h_{m,n}$ kanal katsayılarını, $u_{m,n}$ içsel girişimi, $n_{m,n}$ ise gürültü bileşenini temsil ederken, m ve n alt simgeleri sırasıyla alt taşıyıcı ve zaman indislerini ifade etmektedir. FBMC/OQAM sisteminde N_t adet verici N_r adet alıcı anten içeren çoklu anten yapısı üzerinden sinyal iletimi gerçekleştirildiği takdirde, Denklem (1)'de tek antenli iletim için verilmiş olan sinyal ifadesi aşağıdaki forma dönüşmektedir:

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

Burada i ve j sırasıyla verici ve alıcı anten indisleridir. Örneğin $h_{m,n}^{(ji)}$ i 'inci verici anten ile j 'inci alıcı anten arasındaki kanal katsayısına karşılık gelirken, $y_{m,n}^{(j)}$ j 'inci alıcı antende elde edilen sinyali ifade etmektedir. Denklem (2)'yi aşağıdaki gibi matris formunda da ifade etmek mümkündür:

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

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

Burada alınan sinyaller ve gürültü bileşenleri $N_r \times 1$ boyutlu matrislerle ifade edilirken, iletilen semboller ve kanal katsayıları sırasıyla $N_t \times 1$ ve $N_r \times N_t$ boyutlu matrisler tarafından temsil edilmektedir.

3 Problem tanımı

3.1 FBMC/OQAM sisteminin matris tabanlı temsili

Problemi daha sade hale getirmek için FBMC/OQAM sisteminin matris gösteriminden faydalanılabilir [7]. Söz konusu matris-tabanlı sistem tanımlanmasında, prototip filtre aşağıdaki gibi bir $\mathbf{G} \in \mathbb{C}^{D \times MN}$ iletim matrisi tarafından temsil edilmektedir:

$$\mathbf{G} = [\mathbf{g}_{1,1} \ \mathbf{g}_{2,1} \ \dots \ \mathbf{g}_{M,1} \ \mathbf{g}_{1,2} \ \dots \ \mathbf{g}_{M,N}] \quad (5)$$

Burada $\mathbf{g}_{m,n} \in \mathbb{C}^{D \times 1}$, her biri D adet zaman örneği içeren iletim vektörlerine karşılık gelirken, M ve N sırasıyla toplam alt taşıyıcı ve toplam sembol sayılarını ifade etmektedir. Bunun haricinde, iletilen gerçek değerli semboller aşağıdaki formda tanımlanabilmektedir:

$$\mathbf{a} = \text{vec} \left\{ \begin{bmatrix} a_{1,1} & \dots & a_{1,N} \\ \vdots & \ddots & \vdots \\ a_{M,1} & \dots & a_{M,N} \end{bmatrix} \right\} \quad (6)$$

$$= [a_{1,1} \ a_{2,1} \ \dots \ a_{M,1} \ a_{1,2} \ \dots \ a_{M,N}]^T$$

Denklem (5)'te verilen $\mathbf{G} \in \mathbb{C}^{D \times MN}$ iletim matrisi ile Denklem (6)'da ifade edilen $\mathbf{a} \in \mathbb{C}^{MN \times 1}$ sembol vektörü çarpılarak $\mathbf{s} \in \mathbb{C}^{D \times 1}$ iletim sinyali aşağıdaki gibi elde edilir:

$$\mathbf{s} = \mathbf{G}\mathbf{a} \quad (7)$$

Çok yollu kanalın $\mathbf{H} \in \mathbb{C}^{D \times D}$ ile sembolize edilen bir konvolüsyon matrisi olarak modellenmesi durumunda, alıcıya ulaşan sinyal $\mathbf{r} \in \mathbb{C}^{D \times 1}$ aşağıdaki gibi ifade edilir:

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{n} = \mathbf{H}\mathbf{G}\mathbf{a} + \mathbf{n} \quad (8)$$

Yukarıdaki ifadede $\mathbf{n} \sim CN(0, P_n \mathbf{I}_D)$, beyaz Gauss gürültüsünü temsil etmektedir. Burada P_n gürültü gücünü belirtirken \mathbf{I}_D ise $D \times D$ birim matrise karşılık gelmektedir. Son olarak alınan semboller, alıcıya gelen sinyalin \mathbf{G}^H ile çarpılması sonucu aşağıdaki gibi formüle edilir.

$$\mathbf{y} = \mathbf{G}^H \mathbf{r} = \mathbf{G}^H \mathbf{H}\mathbf{G}\mathbf{a} + \mathbf{G}^H \mathbf{n} \quad (9)$$

3.2 Blok frekans yayımlı FBMC/OQAM sistemi

Geleneksel OFDM sistemine rahatlıkla uygulanabilen MIMO yöntemleri, aynı zamanda içsel girişime yol açan

diklik kısıtlaması nedeniyle FBMC/OQAM sistemine doğrudan uygulanamamaktadır. Öte yandan Nissel ve arkadaşları, [7] nolu çalışmada blok frekans yayılımı olarak adlandırılan yeni bir yaklaşım geliştirerek FBMC/OQAM sistemi için karmaşık dikliği yeniden sağlamanın yolunu bulmuşlardır. İlgili yaklaşım sayesinde, MIMO yöntemlerinin FBMC/OQAM sistemine direk olarak uygulanmasının önüne geçen diklik kısıtlamasının ortadan kaldırılması mümkün hale gelmiştir. Bu çalışmada, MIMO sinyal algılama yöntemlerini diklik kısıtlaması engeline takılmadan doğrudan bir şekilde MIMO-FBMC/OQAM sistemine uygulayabilmek için, FBMC/OQAM sistemi blok frekans yayılım stratejisi ile birleştirilmiştir [7].

Blok frekans yayılım stratejisine göre, ilk olarak bilgi bitleri modüle edilerek $\mathbf{x} \in \mathbb{C}^{\frac{MN}{2} \times 1}$ şeklinde belirtilen QAM sembol dizisi elde edilir. Daha sonra, elde edilen sembol dizisi, $\mathbf{C} \in \mathbb{C}^{\frac{MN}{2} \times \frac{MN}{2}}$ ön kodlama matrisi vasıtasıyla yayma işlemine tabi tutulur. Böylelikle, $\mathbf{a} \in \mathbb{C}^{MN \times 1}$ şeklinde bir vektör ile temsil edilen gerçek değerli sembol dizisi aşağıdaki gibi elde edilir:

$$\mathbf{a} = \mathbf{C}\mathbf{x} \quad (10)$$

Alıcı kısımda, alınan semboller aşağıdaki gibi \mathbf{C}^H matrisi ile çarpılarak yayma işleminin tam tersi yerine getirilir:

$$\mathbf{y} = \mathbf{C}^H \mathbf{y} \quad (11)$$

Yukarıdaki denklemin genişletilmiş formu aşağıda verilmiştir:

$$\mathbf{y} = \mathbf{C}^H \mathbf{G}^H \mathbf{H}\mathbf{G}\mathbf{C}\mathbf{x} + \mathbf{C}^H \mathbf{G}^H \mathbf{n} \quad (12)$$

3.3 MIMO-FBMC/OQAM sisteminde sembol algılama işleminin formüle edilmesi

Blok frekans yayımlı FBMC/OQAM sisteminde, QAM modüleli veri sembolleri, $\frac{MN}{2}$ adet bileşene sahip bir

$\mathbf{x} \in \mathbb{C}^{\frac{MN}{2} \times 1}$ matrisi ile tanımlanmaktadır. N_t adet verici, N_r adet alıcı anten kullanılarak çoklu anten üzerinden iletim gerçekleştirildiği takdirde, $x_{m,n}$ ile sembolize edilen her bir bileşen, $x_{m,n} = [x_{m,n}^{(1)}, x_{m,n}^{(2)}, \dots, x_{m,n}^{(N_r)}]^T$ şeklinde $N_t \times 1$ 'lik bir sembol vektörüne dönüşmektedir. Burada m ve n , sırasıyla alt taşıyıcı ve sembol indisleridir. Benzer şekilde, $x_{m,n}$ sembol vektörünün alıcı kısımda elde edilmiş versiyonu

$$y_{m,n} = [y_{m,n}^{(1)}, y_{m,n}^{(2)}, \dots, y_{m,n}^{(N_r)}]^T \quad \text{şeklinde ifade}$$

edilmektedir. Söz konusu vektörlerden de anlaşılacağı üzere, iletilen sembol sayısı verici anteni sayısına eşit olurken, alınan sembol sayısı ise alıcı anteni sayısına eşit olmaktadır. ML tabanlı sembol algılama işleminde, alınan sinyalden yararlanarak en yüksek iletilme olasılığına sahip olan sembol

vektörü belirlenmektedir. QAM modülasyon seviyesi Z ile sembolize edilecek olursa, $x_{m,n}$ sembol vektörü içerisindeki her bir QAM sembolü Z farklı değer (Örneğin; 4-QAM için 4 farklı değer) alabilecektir. Buna göre, ML algoritması Z^{N_t} adet alternatif sembol vektörü arasından en yüksek iletilme olasılığına sahip optimum sembol vektörünü bulmak zorunda kalacaktır. Bu amaçla, iletilme olasılığı bulunan her bir alternatif sembol vektörü, ML algoritması tarafından aşağıdaki denklem vasıtasıyla test edilmektedir:

$$x_{m,n}^* = \arg \min_{x_{m,n}} \left\{ \|y_{m,n} - H_{m,n} \cdot x_{m,n}\|^2 \right\} \quad (13)$$

Burada $H_{m,n}$, $x_{m,n}$ sembol vektörünü etkileyen $N_r \times N_t$ boyutlu kanal matrisidir. Yukarıda verilen ifadede, Z^{N_t} adet alternatif sembol vektörü tek tek test edilerek, parantez içerisindeki Öklid mesafesini minimum yapan optimum sembol kombinasyonu bulunmaktadır. Bu nedenle, modülasyon seviyesi Z ve verici anteni sayısı N_t değerlerindeki artış, ML algoritmasının karmaşıklığında eksponansiyel bir büyümeye yol açacaktır. Bu çalışmada, oldukça düşük araştırma karmaşıklığı ile ML dedektörüne yakın bir sembol algılama performansı elde etmek için, öncelikle sembol algılama problemi ikili bir optimizasyon problemi haline dönüştürülmüş olup, önerilen DCS algoritması kullanılarak N_t uzunluklu sembol vektörleri optimize edilmiştir. Böylelikle, çok fazla işlem yüküne neden olmadan iteratif bir şekilde optimuma yakın bir çözüme ulaşmak mümkün hale gelmiştir.

Denklem (13)'te verilen ML tabanlı sembol algılama formülünde, $H_{m,n}$ ile ifade edilen gerçek kanal katsayılarının alıcı tarafından hatasız bir şekilde önceden bilindiği varsayılmaktadır. Ancak pratik sistemlerde gerçek kanal katsayılarının hatasız bir şekilde kestirilmesi çoğu zaman mümkün değildir. Bu çalışmada, bu durum göz önünde bulundurulmuş olup, MIMO-FBMC/OQAM sistemi kestirim hataları da hesaba katılarak modellenmiştir. Buna göre, ML tabanlı sembol algılama işlemi aşağıdaki gibi yeniden tanımlanmıştır:

$$x_{m,n}^* = \arg \min_{x_{m,n}} \left\{ \|y_{m,n} - \hat{H}_{m,n} \cdot x_{m,n}\|^2 \right\} \quad (14)$$

Yukarıdaki ifadede, hatalı bir şekilde kestirilen kanal katsayıları $\hat{H}_{m,n}$ ile temsil edilmektedir. Söz konusu hatalı kanal katsayıları aşağıdaki gibi tanımlanmaktadır [19]:

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

Burada $H_{m,n}$, gerçek kanal katsayılarına karşılık gelirken, e kestirim hatası yüzdesini, θ ise sıfır ortalamalı ve birim varyanslı kompleks Gauss değişkenini ifade etmektedir.

4 Ayırık guguk kuşu arama (DCS) algoritması

Doğadaki guguk kuşu türleri, yavrularını büyütme sorumluluğundan tamamen kurtulmak amacıyla, farklı kuş türlerinin yuvalarına kendi yumurtalarını bırakırlar. Bunu yaparken, yuva sahibi kuşun yabancı yumurtayı fark etme olasılığını düşürmek için, kendi yumurtalarının renk ve dokusunu, yuvada yer alan diğer yumurtaların renk ve dokusuna benzetme yeteneği geliştirmişlerdir. Çünkü, yuva sahibi kuş yabancı yumurtayı fark ettiği takdirde, ya farklı bir bölgede yeni bir yuva yapmak üzere mevcut yuvasını terk etmekte ya da guguk kuşu tarafından bırakılmış olan yabancı yumurtayı yuvadan aşağı atarak kırmaktadır. Yavrularının başka bir kuş türü tarafından büyütülüp yetiştirileceği en uygun yuvayı bulmak amacıyla guguk kuşu türlerinin geliştirmiş olduğu karmaşık davranışlar aşağıdaki gibi idealize edilerek standart guguk kuşu arama (CS) algoritması geliştirilmiştir [20]:

1. Her bir guguk kuşu, rastgele seçmiş olduğu yuvaya her seferinde tek bir yumurta bırakır.
2. Yumurta bırakmaya uygun olan yüksek kaliteli yuvalar bir sonraki jenerasyona aktarılır.
3. Guguk kuşları tarafından yumurta bırakılabilecek yuva sayısı sabittir. Yuva sahibi kuşun yabancı yumurtayı fark etme olasılığı $p_a \in [0,1]$ şeklinde tanımlanır. Bu olasılığın gerçekleşmesi halinde yuva sahibi kuş başka bir pozisyonda yeni bir yuva inşa etmek üzere mevcut yuvasını terk eder ya da yabancı yumurtayı elimine ederek mevcut yuvasında kalmaya devam eder.

Guguk kuşu türlerinin sıra dışı üreme davranışlarından ilham alınarak geliştirilen guguk kuşu arama (CS) algoritması [20], farklı alanlarda vermiş olduğu başarılı sonuçlar nedeniyle sürü zekâsı tabanlı meta-sezgisel algoritmalar arasındaki popüleritesini hızla artırmıştır. Ancak söz konusu algoritmanın standart versiyonu sadece sürekli optimizasyon problemleriyle uyumludur. Bu nedenle, orijinal CS algoritmasını kullanarak ayırık bir optimizasyon problemi olan sembol algılama problemini direk olarak ayırık uzayda etkin bir şekilde çözmek mümkün değildir. Bu durumu göz önünde bulundurarak, ayırık uzayda QAM sembol vektörlerini ikili bir optimizasyon işlemine tabi tutup optimum çözüme hızlı bir şekilde yakınsayabilmek için CS algoritmasının DCS [10] isimli etkili bir ayırık versiyonu kullanılmıştır. DCS algoritmasında mevcut popülasyondaki guguk kuşu (veya yuva) pozisyonları aşağıda verilen vektör ile temsil edilmektedir:

$$c_s^{(d)} = [c_s^{(1)}, c_s^{(2)}, \dots, c_s^{(D)}] \quad , \quad s = 1, 2, \dots, S \quad (16)$$

Burada $c_s^{(d)} \in \{0, 1\}$ olup, yukarıda ifade edilen D -boyutlu pozisyon vektörü D adet 0 ve 1'lerden oluşmaktadır. Bunun haricinde S parametresi, toplam guguk kuşu (veya yuva) sayısını belirtmektedir.

4.1 Başlangıç pozisyonlarının oluşturulması

Başlangıç guguk kuşu popülasyonunu oluşturmak için rastgele S farklı pozisyon vektörü üretilir. Ardından rastgele üretilmiş olan her bir pozisyon vektörünün uygunluk değeri hesaplanır. Uygunluk hesabı yapılırken, DCS algoritmasının uygulanmış olduğu probleme ait amaç fonksiyonu kullanılır.

4.2 Yeni pozisyon üretimi

Mevcut guguk kuşu popülasyonu içerisinde rastgele bir pozisyon vektörü seçilir. Rastgele seçilmiş olan söz konusu pozisyon vektörü aşağıdaki gibi tanımlanır:

$$c_r^{(d)} = [c_r^{(1)}, c_r^{(2)}, \dots, c_r^{(D)}] \quad , \quad d = 1, 2, \dots, D \quad (17)$$

Burada r değişkeni, $[1, S]$ aralığında üretilen rastgele bir değişkendir. Daha sonra, döngüsel bit çevirme prosedürü [21] kullanılarak $c_r^{(d)}$ vektöründen yeni bir pozisyon vektörü üretilir. Bu amaçla ilk olarak, seçilmiş olan pozisyon vektörüne aşağıdaki gibi bit çevirme işlemi uygulanır:

$$c_r^{(d),yeni} = flip(c_r^{(d)})_{\theta_r} \quad (18)$$

Burada θ_r , rastgele seçilen $c_r^{(d)}$ vektörünün bit çevirme indisidir. Yukarıda verilen $flip(\cdot)_{\theta_r}$ operatörü, $c_r^{(d)}$ vektörünün θ_r 'inci elementini 1'den 0'a ya da 0'dan 1'e dönüştürmektedir. DCS algoritmasında, bit çevirme indislerinin başlangıç değerleri $\theta_s = [\theta_1, \theta_2, \dots, \theta_S] = [1, 1, \dots, 1]$ şeklinde tayin edilmektedir. Bu yüzden, rastgele seçilmiş pozisyon vektörünün bit çevirme indisine karşılık gelen θ_r parametresinin ilk değeri de diğer bit çevirme indislerinin gibi 1'e eşit olacaktır. Bit çevirme işleminin hemen ardından θ_r değeri aşağıdaki gibi güncellenir:

$$\theta_r = \theta_r + 1 \quad (19)$$

Böylelikle, Denklem (18)'de elde edilen yeni pozisyon vektörünün ilerleyen iterasyonlarda komşu çözüm üretmek üzere yeniden seçilmesi durumunda, bit çevirme işleminin bir sonraki vektör bileşeninden devam etmesi sağlanmaktadır. Bit çevirme işleminin döngüsel bir şekilde ilerleyebilmesi için, θ_r bit çevirme indisi 1 artırıldıktan hemen sonra aşağıdaki işleme tabi tutulur:

$$\theta_r = \text{mod}(\theta_r - 1, D) + 1 \quad (20)$$

Burada θ_r indisinin değeri, mod alma operatörü kullanılarak kontrol altında tutulmaktadır. θ_r değeri, D değerini her aştığında, Denklem (20)'de verilen kontrol mekanizması sayesinde tekrardan başlangıç değerine yani 1'e eşitlenir. Örneğin, θ_r indis değeri $D + 1$ 'e çıktığı anda, Denklem (20)'de verilen ifade, $\theta_r = \text{mod}(D, D) + 1$ haline

dönüşecektir. Bu durumda, $\text{mod}(D, D)$ 0'a eşit olduğundan dolayı, θ_r 'nin değeri tekrardan 1'e yani başlangıç değerine döndürülecektir. Böylelikle, iterasyonlar ilerledikçe, bit çevirme işlemi döngüsel bir şekilde ilerleyecektir.

Bit çevirme indisi güncellenip kontrol mekanizmasından geçirildikten sonra, yeni üretilen pozisyon vektörünün uygunluk değeri hesaplanır. $c_r^{(d),yeni}$ vektörünün uygunluk değeri $c_r^{(d)}$ vektörünün uygunluk değerinden daha iyi ise, $c_r^{(d),yeni}$ vektörü, $c_r^{(d)}$ vektörünün yerini alır. Ters durumda, herhangi bir değişiklik yapılmaz.

4.3 Yabancı yumurtanın fark edilmesi

Bu aşamada, ilk olarak $r_1 \in [0, 1]$ aralığında olacak şekilde rastgele bir sayı üretilir. Eğer üretilen r_1 sayısının değeri, yabancı yumurtayı ayırt etme olasılığını tayin eden p_a katsayısından küçükse, guguk kuşu popülasyonunda güncel olarak en kötü uygunluk değerine sahip olan pozisyon vektörü yerine rastgele bir çözüm vektörü üretilir. Bu işlemin ardından 4.2 aşamasından devam edilerek bir sonraki iterasyona geçilir.

5 Ayrık guguk kuşu arama (DCS) tabanlı ML stratejisi

Bir önceki bölümden de açıkça anlaşılacağı üzere DCS algoritması, ikili araştırma uzayında çözülebilen ayrık optimizasyon problemleri için geliştirilmiş bir algoritmadır. Ancak, sembol algılama probleminde optimize edilecek olan $x_{m,n} = [x_{m,n}^{(1)}, x_{m,n}^{(2)}, \dots, x_{m,n}^{(N_r)}]^T$ sembol vektörleri, karmaşık değerli QAM sembollerinden oluştuğu için, ilgili vektörler DCS algoritması ile direk olarak ikili optimizasyon işlemine tabi tutulamamaktadır. Öte yandan, karmaşık sayılardan oluşan söz konusu sembol vektörlerini ikili bit dizileri haline dönüştürüp ikili araştırma uzayında optimizasyon işlemine uygun hale getirmek mümkündür. QAM modülasyon tekniğinde, $x_{m,n}$ vektöründe yer alan karmaşık değerli her bir QAM sembolü, $\log_2^Z = k$ basamaklı belirli bir ikili sayıya karşılık gelmektedir. Dolayısıyla, her bir QAM sembolü yerine eşdeğeri olan ikili sayı kullanıldığı takdirde, karmaşık sayılardan oluşan QAM sembol vektörleri ikili bit dizileri haline dönüştürülebilmektedir. Böylelikle, sembol optimizasyonu işlemi DCS algoritmasını kullanarak doğrudan ikili araştırma uzayında gerçekleştirmek mümkün hale gelmektedir. Önerilen DCS-ML stratejisinde, karmaşık değerli QAM sembol vektörleri ve ilgili vektörlerin eşdeğeri olan ikili bit dizileri aşağıdaki gibi guguk kuşu pozisyonları şeklinde ifade edilmektedir:

$$x_s^{(i)} = [x_s^{(1)}, x_s^{(2)}, \dots, x_s^{(N_r)}] \quad , \quad s = 1, 2, \dots, S \quad (21)$$

$$b_s^{(j)} = [b_s^{(1)}, b_s^{(2)}, \dots, b_s^{(k \cdot N_r)}] \quad , \quad s = 1, 2, \dots, S \quad (22)$$

$x_s^{(i)}$ vektöründeki her bir karmaşık sembol k adet bit taşıdığı için, söz konusu vektörün Denklem (22)'de verilen ikili eşdeğerinin içerdiği toplam bit sayısı $k \cdot N_r$

olmaktadır. DCS-ML stratejisine ait işlem adımları aşağıda verilmiştir:

Adım 1: İlk olarak, **Denklem (22)**'deki gibi ikili bit dizileri formunda rastgele S adet QAM sembol vektörü üretilerek başlangıç guguk kuşu popülasyonu oluşturulur.

Adım 2: Rastgele üretilen başlangıç popülasyon üyelerinin her biri uygunluk hesabına tabi tutulur. Bu işlem iki aşamada gerçekleştirilir: İlk aşamada, ilgili popülasyon üyesini temsil eden $b_s^{(j)}$ ikili bit dizisi, kendi eşdeğeri olan karmaşık değerli $x_s^{(i)}$ QAM sembol vektörü haline dönüştürülür. İkinci aşamada ise, elde edilen $x_s^{(i)}$ sembol vektörü aşağıda verilen uygunluk fonksiyonunda kullanılarak ilgili popülasyon üyesinin uygunluk değeri hesaplanır:

$$f_s = \text{fit}(x_s^{(i)}) = \|y_{m,n} - \hat{H}_{m,n} \cdot x_s^{(i)}\|^2 \quad (23)$$

$$s = 1, 2, \dots, S; \quad i = 1, 2, \dots, N_i$$

Burada f_s , guguk kuşu popülasyonundaki s 'inci bireyin uygunluk değerini ifade etmektedir.

Adım 3: Popülasyon içerisinde rastgele bir üye (r 'inci guguk kuşu: $b_r^{(j)}$) seçilir. Söz konusu popülasyon üyesi aşağıdaki gibi tanımlanır:

$$b_r^{(j)} = [b_r^{(1)}, b_r^{(2)}, \dots, b_r^{(k \cdot N_i)}] \quad , \quad j = 1, 2, \dots, k \cdot N_i \quad (24)$$

Adım 4: Rastgele seçilen $b_r^{(j)}$ vektörü üzerinde bit çevirme işlemi gerçekleştirilerek yeni bir çözüm üretilir:

$$b_r^{(j), \text{yeni}} = \text{flip}(b_r^{(j)})_{\theta_r} \quad (25)$$

Sonrasında, bit çevirme işleminin döngüsel bir şekilde devam etmesini sağlamak amacıyla sırasıyla aşağıdaki işlemler yerine getirilir:

$$\theta_r = \theta_r + 1 \quad (26)$$

$$\theta_r = \text{mod}(\theta_r - 1, k \cdot N_i) + 1 \quad (27)$$

Daha sonra, yeni üretilen $b_r^{(j), \text{yeni}}$ çözümü uygunluk kalitesi bakımından değerlendirilir. Bunun için, öncelikle söz konusu $b_r^{(j), \text{yeni}} = [b_r^{(1), \text{yeni}}, b_r^{(2), \text{yeni}}, \dots, b_r^{(k \cdot N_i), \text{yeni}}]$ ikili çözüm vektörü, kendisine karşılık gelen karmaşık değerli $x_r^{(i), \text{yeni}} = [x_r^{(1), \text{yeni}}, x_r^{(2), \text{yeni}}, \dots, x_r^{(N_i), \text{yeni}}]$ QAM sembol vektörü haline dönüştürülür. Ardından ilgili yeni çözümün uygunluk değeri aşağıdaki gibi hesaplanır:

$$f_r^{\text{yeni}} = \text{fit}(x_r^{(i), \text{yeni}}) = \|y_{m,n} - \hat{H}_{m,n} \cdot x_r^{(i), \text{yeni}}\|^2 \quad (28)$$

Burada f_r^{yeni} , yeni üretilen çözümün uygunluk değerini belirtmektedir. $b_r^{(j), \text{yeni}}$ çözümü, $b_r^{(j)}$ çözümüne göre daha iyi bir uygunluk değerine sahipse, eski çözümün yerini yeni çözüm alır. Bit çevirme işlemi sonrası $b_r^{(j)}$ çözümünün uygunluk değerinde herhangi bir iyileşme elde edilememişse, ilgili çözüm olduğu gibi bırakılarak herhangi bir güncelleme yapılmaz.

Adım 5: $[0, 1]$ aralığında rastgele bir r_1 sayısı üretilir. Eğer $r_1 < p_a$ ise, mevcut popülasyondaki en kötü uygunluk değerine sahip çözüm vektörünün yerine rastgele bir çözüm üretilir.

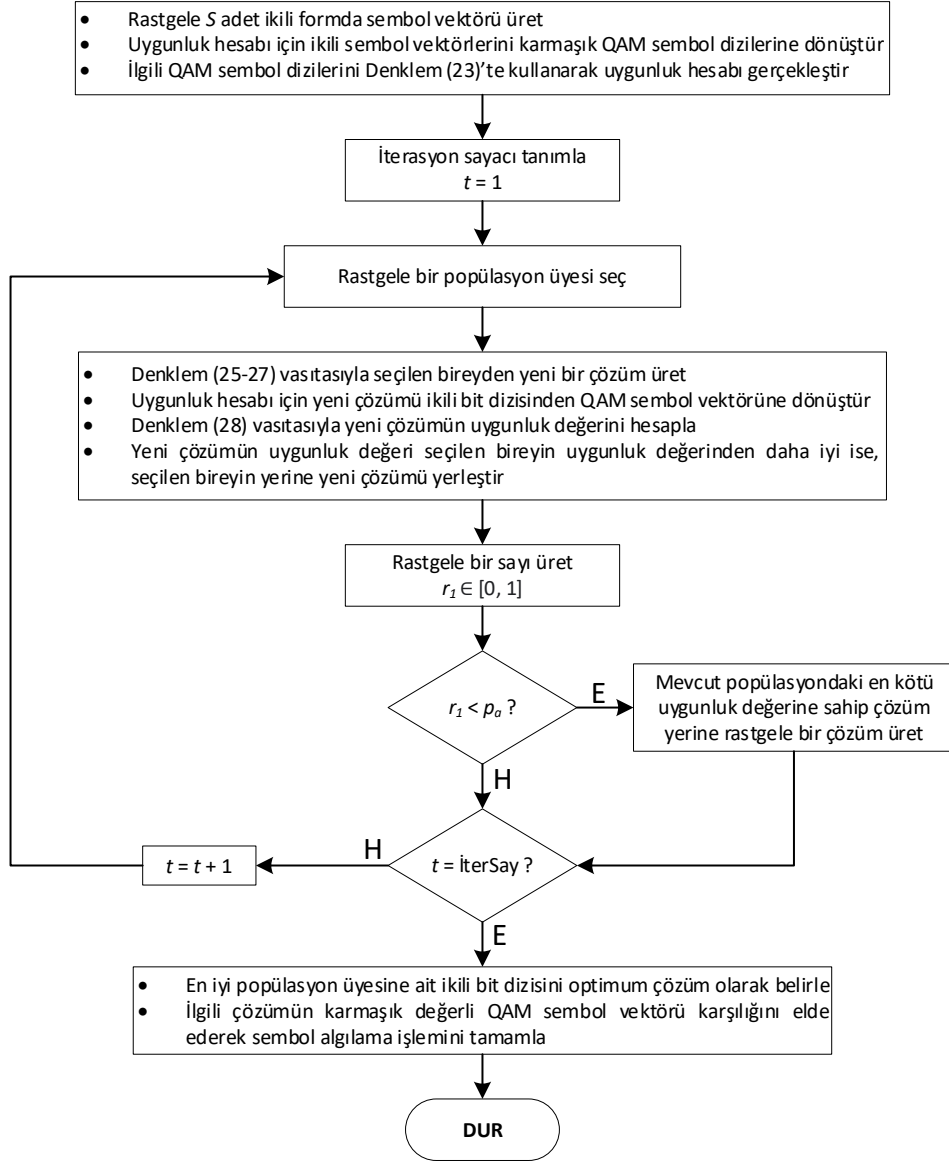
Adım 6: Bir sonraki iterasyona geçmek için Adım 3'ten devam edilir. Adım 3 - Adım 5 arasındaki işlemler, durdurma kriteri olarak belirlenen maksimum iterasyon sayısına (İterSay) ulaşıncaya kadar tekrarlanır. Durdurma kriteri sağlandıktan sonra, en iyi uygunluk değerine sahip olan ikili pozisyon vektörü, optimum çözüm olarak belirlenir. Son olarak, söz konusu ikili bit dizisine karşılık gelen QAM sembol dizisi elde edilerek optimum sembol vektörü elde edilir. Bu son işlemle birlikte sembol algılama işlemi tamamlanır. DCS-ML stratejisine ait akış diyagramı **Şekil 1**'de verilmiştir.

6 Simülasyon sonuçları

Bu bölümde, önerilen DCS-ML stratejisi ve diğer yöntemlerin MIMO-FBMC/OQAM sistemindeki sembol algılama performansları, ilgili metotlarının 4×4 , 6×6 ve 8×8 anten konfigürasyonlarındaki bit hata oranı başarımları karşılaştırılarak test edilmiştir. Bunun haricinde, DCS-ML prosedürünün optimum çözüme ne kadar hızlı bir şekilde yakınsadığını ortaya koyabilmek amacıyla, söz konusu anten konfigürasyonlarının her biri için yakınsama analizi gerçekleştirilmiştir. Simülasyonlarda, $[-2.5, 0, -3, -5, -2, -4]$ dB güç yollarına ve $[0, 300, 1000, 1600, 5000, 6600]$ ns bağıl gecikmelere sahip "Cost 207 Bad Urban" kanal modeli kullanılmıştır. MIMO-FBMC/OQAM sistemi için belirlenen diğer parametre değerleri **Tablo 1**'de verilmiştir.

Tablo 1. MIMO-FBMC/OQAM sistem parametreleri

MIMO konfigürasyonları	$4 \times 4, 6 \times 6, 8 \times 8$
Alt taşıyıcı sayısı	$M = 64$
Taşıyıcı frekansı	2.5 GHz
Alt taşıyıcı aralığı	15 KHz
Prototip filtre modeli	PHYDYAS
QAM modülasyon seviyesi	$Z = 4$
FBMC sembolü sayısı	$N = 30$
Örtüşme faktörü	4
Kanal modeli	Cost 207 Bad Urban
Kestirim hatası yüzdesi	$e = 30\%$



Şekil 1. DCS-ML stratejisinin akış diyagramı

Tablo 2’de klasik ML yöntemi de dahil olmak üzere ML tabanlı sembol dedektörlerinin araştırma karmaşıklıkları sırasıyla 4×4, 6×6 ve 8×8 anten yapıları için analiz edilmiştir. Herhangi bir ML tabanlı sembol algılama stratejisinin araştırma karmaşıklığı tanımlanırken, optimum QAM sembol vektörü arayışı boyunca gerçekleştirilen Öklid hesabı sayısı dikkate alınır. Öklid mesafesinin nasıl hesaplandığı Denklem (13)’te verilmiştir. Klasik ML tekniğinde, alınan sinyalle minimum Öklid mesafesine sahip olan optimum QAM sembol dizisini bulmak için, olası her bir sembol kombinasyonu başına bir Öklid hesabı gerçekleştirilmektedir. Buna göre, ML yönteminde sembol algılama işlemi boyunca gerçekleştirilen toplam Öklid hesabı sayısı, Z^N ifadesi kullanılarak bulunabilen olası sembol kombinasyonlarının toplam sayısına eşit olacaktır. Bu nedenle, klasik ML dedektörünün araştırma karmaşıklığı

(SC), $SC = Z^N$ şeklinde tanımlanmaktadır. Burada Z, modülasyon seviyesini belirtirken N , verici anten sayısını ifade etmektedir. Metasezgisel tabanlı gelişmiş ML stratejilerinde ise, optimizasyon işlemi süresince QAM sembol vektörlerinin alınan sinyale olan Öklid mesafeleri bakımından değerlendirildiği toplam uygunluk hesabı sayısı, direkt olarak araştırma karmaşıklığını vermektedir. Hem DBHS-ML hem de DCS-ML stratejilerinde, popülasyon boyutundan bağımsız olarak her bir araştırma ya da iterasyon için tek bir uygunluk hesabı gerçekleştirilmektedir. Bu nedenle ilgili sembol algılama stratejilerinde, araştırma karmaşıklığına karşılık gelen toplam uygunluk hesabı sayıları sırasıyla maksimum araştırma sayısı (ArSay) ve maksimum iterasyon sayısına (IterSay) eşit olmaktadır.

Anten sayısındaki artışla birlikte araştırma uzayında genişleme meydana geldiğinden dolayı daha yüksek

Tablo 2. ML tabanlı sembol algılama yöntemlerinin araştırma karmaşıklıkları

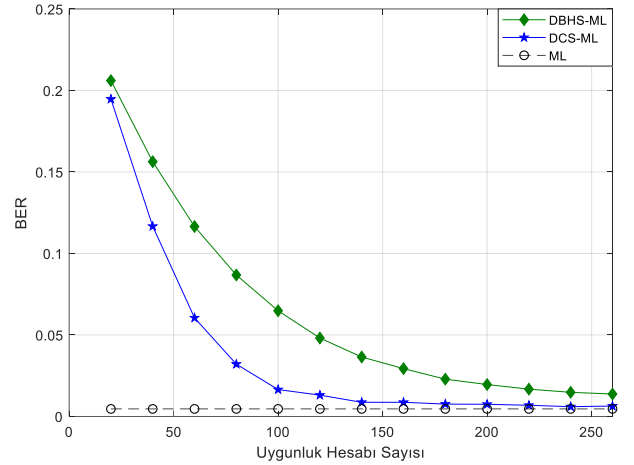
Yöntemler	4×4	6×6	8×8	
DBHS-ML	Harmoni sayısı	$HarSay = 20$	$HarSay = 30$	$HarSay = 40$
	Maksimum araştırma sayısı	$ArSay = 260$	$ArSay = 900$	$ArSay = 2200$
	Araştırma karmaşıklığı	$SC = ArSay = 260$	$SC = ArSay = 900$	$SC = ArSay = 2200$
	SNR = 12 dB için BER	0.01374	0.007995	0.004557
DCS-ML	Guguk kuşu sayısı	$S = 15$	$S = 20$	$S = 25$
	Maksimum iterasyon sayısı	$IterSay = 150$	$IterSay = 500$	$IterSay = 1000$
	Araştırma karmaşıklığı	$SC = IterSay = 150$	$SC = IterSay = 500$	$SC = IterSay = 1000$
	SNR = 12 dB için BER	0.008659	0.003498	0.001634
ML	Araştırma karmaşıklığı	$SC = Z^{N_i} = 4^4 = 256$	$SC = Z^{N_i} = 4^6 = 4096$	$SC = Z^{N_i} = 4^8 = 65536$
	SNR = 12 dB için BER	0.004596	0.002005	0.0006576

araştırma sayılarına ve daha büyük popülasyon boyutlarına ihtiyaç duyulmaktadır. Örneğin 4×4, 6×6 ve 8×8 anten konfigürasyonları için, önerilen DCS-ML stratejisinin popülasyon boyutları 15, 20 ve 25 olarak seçilirken, karşılaştırma amaçlı kullanılan DBHS-ML stratejisinin popülasyon büyüklükleri sırasıyla 20, 30 ve 40 olarak belirlenmiştir. Benzer şekilde, söz konusu anten konfigürasyonları için DCS-ML stratejisinin araştırma karmaşıklıkları sırasıyla 150, 500 ve 1000 olarak belirlenirken, DBHS-ML yönteminin araştırma karmaşıklıkları 260, 900, 2200 değerlerine ayarlanmıştır. DCS-ML yönteminin araştırma karmaşıklıkları, bit hata oranı başarımları ile araştırma karmaşıklığı arasındaki ödünleşim dikkate alınarak belirlenmiştir. Önerilen yöntemin ne kadar etkili bir performansa sahip olduğunu ortaya koymak amacıyla, diğer yöntemin araştırma karmaşıklıkları çok daha yüksek değerlere ayarlanmıştır. DCS-ML stratejisi, daha az araştırma yapma fırsatı verilerek dezavantajlı duruma düşürülmesine rağmen, çok daha iyi sonuçlara ulaşmış rakibini net bir şekilde geride bırakmıştır. Zeki optimizasyon tabanlı ML dedektörlerine ait diğer parametre değerleri Tablo 3'te verilmiştir.

Tablo 3. Metasezgisel tabanlı ML dedektörlerinin simülasyon parametreleri

Yöntemler	Parametreler	Değerler
DBHS-ML	Harmoni hafızası değerlendirme oranı	$HMCR = 0.6$
	Perde ayarlama oranı	$PAR = 0.05$
DCS-ML	Yabancı yumurtayı ayırt etme olasılığı	$p_a = 0.25$

Şekil 2'de, mevcut metasezgisel tabanlı ML stratejilerinin 4×4 anten konfigürasyonu için 12 dB SNR değerindeki yakınsama eğrileri elde edilmiştir. Bunun için ilgili stratejiler her bir uygunluk hesabı sayısı için 10 defa koşuturulmuştur. Böylelikle, her bir uygunluk hesabı için 10 farklı BER değeri elde edilmiştir. Şekil 2'de görülen yakınsama eğrileri, söz konusu BER değerlerinin ortalaması



Şekil 2. Metasezgisel tabanlı sembol dedektörlerinin 4×4 MIMO-FBMC/OQAM sistemindeki yakınsama performansı

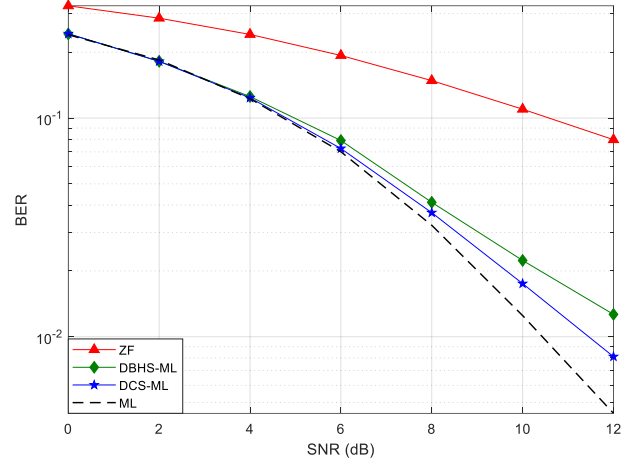
olarak oluşturulmuştur. En altta yer alan doğru ise, ML dedektörü tarafından kapsamlı arama sonucu ulaşılan en düşük BER seviyesidir. Herhangi bir zeki optimizasyon tabanlı sembol dedektöründe amaç, mümkün olduğunca az sayıda uygunluk hesabı gerçekleştirerek en düşük BER seviyesine yakınsamak olmalıdır. Şekil 2'den de açıkça görüleceği üzere, önerilen DCS-ML sembol dedektörü çok hızlı bir şekilde optimum sonuca yakınsamaktadır. Karşılaştırma amaçlı kullanılan diğer yöntem ise, 260 uygunluk hesabı sonrasında bile optimum BER seviyesine yeterince yaklaşamamıştır. Örneğin, DBHS-ML yönteminin 260 adet uygunluk hesabı sonucu ulaşılmış olduğu 0.01374 BER seviyesine, DCS-ML yöntemi kullanılarak sadece 116 adet uygunluk hesabı ile ulaşılabilmektedir. Daha önce de bahsedildiği gibi, önerilen yöntem için uygunluk hesabı sayısı belirlenirken BER performansı ile araştırma karmaşıklığı arasındaki ödünleşim dikkate alınmıştır. Şekil 2'de görüldüğü gibi, önerilen DCS-ML stratejisinin yakınsama davranışı 150 uygunluk hesabı sayısında neredeyse tamamlanmıştır. Bu nedenle, söz konusu ödünleşim dikkate alınarak, 4×4 MIMO yapısı için ilgili stratejinin araştırma karmaşıklığı 150 olarak ayarlanırken,

DBHS-ML yönteminin araştırma karmaşıklığı 260'a sabitlenmiştir.

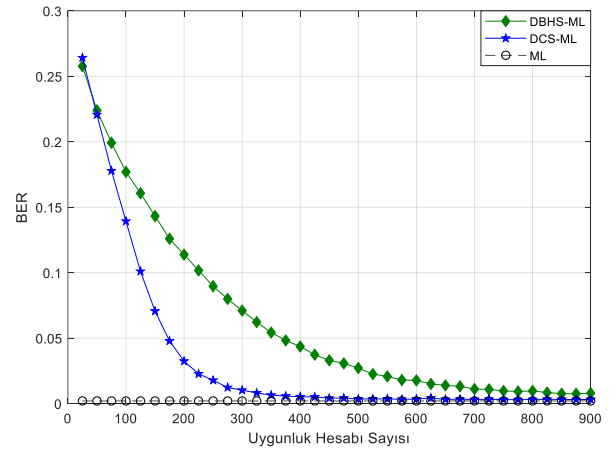
Şekil 3'te, makalede ele alınan sembol algılama stratejileri 4×4 MIMO-FBMC/OQAM sistemindeki bit hata oranı başarımları bakımından birbirleriyle karşılaştırılmıştır. Şekilden de açıkça görüleceği üzere, ZF tekniği BER performansı bakımından diğer yöntemlerin oldukça gerisinde kalırken, sistem karmaşıklığında aşırı bir artışa neden olan ML dedektörü beklenildiği üzere en düşük BER seviyesine ulaşan yöntem olmuştur. Öte yandan, önerilen DCS-ML stratejisi, diğer metasezgisel tabanlı sembol dedektörünü net bir şekilde geride bırakarak ML yöntemine yakın bir performans sergilemiştir. Üstelik bu üstün performansına, DBHS-ML yöntemine göre oldukça düşük sayıda uygunluk hesabı gerçekleştirerek ulaşmıştır.

Şekil 4'te, zeki optimizasyon tabanlı ML stratejilerinin yakınsama eğrileri 6×6 anten konfigürasyonu için elde edilmiştir. Şekil 2 ile Şekil 4 karşılaştırıldığında da görülebileceği gibi, MIMO konfigürasyonunun 4×4'ten 6×6'ya yükseltilmesi, her bir stratejinin optimum çözüme yakınsamak için daha fazla uygunluk hesabına ihtiyaç duymasına neden olmuştur. Şekil 4'te net bir şekilde görüldüğü üzere, önerilen DCS-ML stratejisi, karşılaştırma amaçlı kullanılan DBHS-ML yöntemine oranla çok daha iyi bir yakınsama performansı sergilemiş olup, söz konusu yöntemle göre daha az uygunluk hesabı ile daha düşük bir BER seviyesine ulaşmıştır. Farklı bir açıdan bakacak olursak, DCS-ML stratejisi vasıtasıyla çok daha az sayıda uygunluk hesabı gerçekleştirerek DBHS-ML yöntemi ile aynı BER seviyesine ulaşmak mümkündür. Örneğin DBHS-ML yöntemi, 0.007995 BER seviyesine ulaşmak için 900 adet uygunluk hesabına ihtiyaç duyarken, DCS-ML yönteminin söz konusu BER değerine ulaşması için 327 sayıda uygunluk hesabı yeterli olmaktadır. Önerilen sembol algılama stratejisi, yaklaşık olarak 500. uygunluk hesabında doyuma ulaştığından dolayı, ilgili stratejinin 6×6 anten konfigürasyonundaki uygunluk hesabı sayısı 500 olarak tayin edilmiştir. Böylelikle, gereğinden fazla araştırma karmaşıklığından kaçınılmıştır.

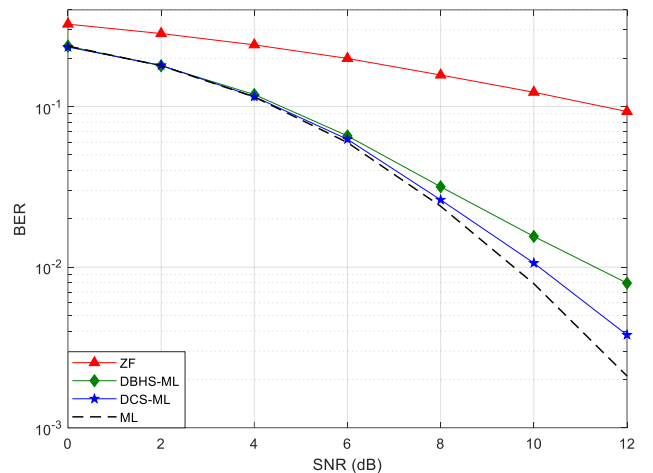
Şekil 5'te, sembol dedektörlerinin MIMO-FBMC/OQAM sistemindeki BER başarımları, 6×6 anten konfigürasyonu için incelenmiştir. Şekil 3 ile Şekil 5 karşılaştırıldığında, anten yapısının 4×4'ten 6×6'ya yükseltilmesi ile birlikte, bir tür lineer sembol dedektörü olması sebebiyle anten sayısındaki artıştan olumsuz olarak etkilenen ZF dedektörü haricinde, her bir sembol dedektörünün BER performansında hatırı sayılır bir iyileşme meydana geldiği görülecektir. Anten konfigürasyonundaki söz konusu artışa rağmen önerilen DCS-ML stratejisi, ML dedektörü tarafından elde edilen optimum BER sonuçlarına en çok yaklaşan yöntem olarak diğer yöntemlere olan üstünlüğünü korumuştur. Örneğin, ZF ve DBHS-ML dedektörlerinin 10 dB SNR değerindeki BER seviyeleri sırasıyla 1.23×10^{-1} ve 1.55×10^{-2} iken, DCS-ML stratejisinin aynı SNR değeri için ulaşmış olduğu BER seviyesi 1.06×10^{-2} 'dir.



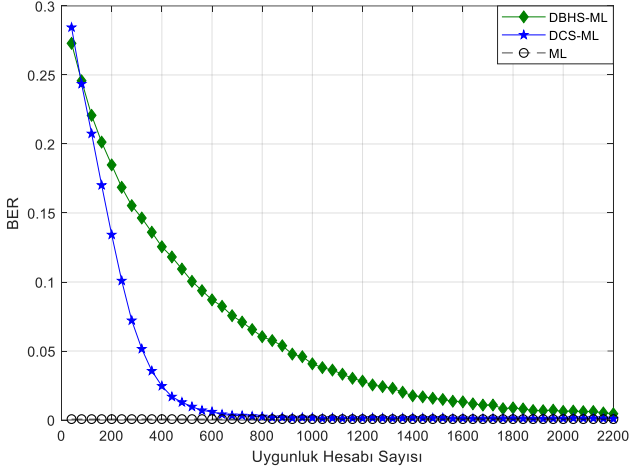
Şekil 3. Sembol dedektörlerinin 4×4 MIMO-FBMC/OQAM sistemi için BER performans analizi



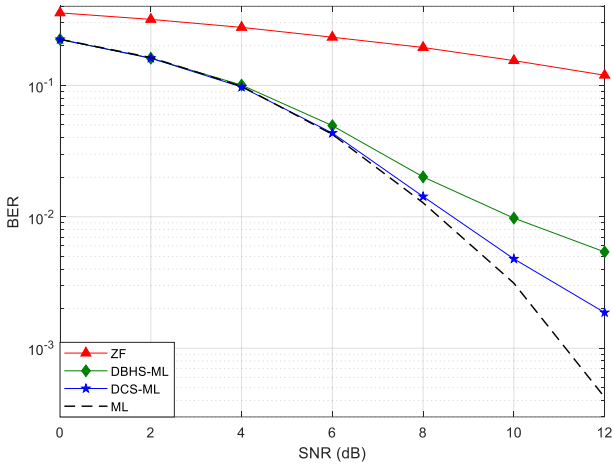
Şekil 4. Metasezgisel tabanlı sembol dedektörlerinin 6×6 MIMO-FBMC/OQAM sistemindeki yakınsama performansı



Şekil 5. Sembol dedektörlerinin 6×6 MIMO-FBMC/OQAM sistemi için BER performans analizi



Şekil 6. Metasezgisel tabanlı sembol dedektörlerinin 8x8 MIMO-FBMC/OQAM sistemindeki yakınsama performansı



Şekil 7. Sembol dedektörlerinin 8x8 MIMO-FBMC/OQAM sistemi için BER performans analizi

Şekil 6'da, mevcut zeki optimizasyon tabanlı sembol algılama stratejilerinin yakınsama performansları 8x8 anten yapısı için karşılaştırılmıştır. Yakınsama eğrilerinden de anlaşılacağı üzere, DCS-ML stratejisi yaklaşık olarak 1000 adet uygunluk hesabı sonucunda optimum çözüme tamamen yakınsamaktadır. Bu nedenle önerilen yöntemin araştırma karmaşıklığı 1000 olarak ayarlanmıştır. Diğer taraftan DBHS-ML dedektörünün, DCS-ML stratejisinin sadece 1000 uygunluk hesabı ile ulaşılmış olduğu BER seviyesine erişebilmesi için 2200 adet uygunluk hesabı bile yeterli gelmemiştir. Bununla birlikte, DBHS-ML yönteminin 2200 uygunluk hesabı sonucunda ulaşılmış olduğu BER sonucuna DCS-ML stratejisi ile çok daha az sayıda uygunluk hesabı ile ulaşılabilmektedir. Örneğin, DBHS-ML dedektörünün 0.004557 BER seviyesine ulaşabilmesi için 2200 adet uygunluk hesabı gerekirken, önerilen sembol algılama stratejisi kullanılarak aynı BER seviyesine sadece 630 uygunluk hesabı ile ulaşmak mümkün olabilmektedir.

Şekil 7'de, her bir sembol algılama stratejisinin SNR(dB)-BER eğrisi 8x8 MIMO konfigürasyonu için elde edilmiştir. Daha önce de bahsedildiği üzere, MIMO-

FBMC/OQAM verici ve alıcısına eklenen her bir anten, araştırma uzayında üstel bir artışa neden olmaktadır. Bu nedenle, anten sayısındaki her bir artış, metasezgisel tabanlı sembol algılama stratejilerinin kabul edilebilir bir çözüme ulaşabilmeleri için daha fazla sayıda uygunluk hesabı gerçekleştirmelerini zorunlu kılmaktadır. Önerilen DCS-ML sembol dedektörü, 4x4 ve 6x6 anten yapılarında olduğu gibi, 8x8 anten yapısında da üstün bir sembol algılama performansı sergilemiştir. En yakın rakibi olan DBHS-ML yöntemine bit hata oranı seviyesi bakımından ciddi bir fark atmıştır. Üstelik bunu rakibine oranla çok daha az sayıda uygunluk hesabı gerçekleştirerek başarmıştır.

7 Sonuç

Bu çalışmada, guguk kuşu arama algoritmasının DCS olarak adlandırılan etkili bir ayrık versiyonu geleneksel ML yöntemine entegre edilerek DCS-ML isimli yeni bir sembol algılama stratejisi geliştirilmiş olup, söz konusu strateji kullanılarak çok daha az araştırma karmaşıklığı ile optimuma yakın bir sembol algılama performansı elde etmek mümkün hale gelmiştir. Geliştirilmiş olan DCS-ML stratejisinin sembol algılama performansı, MIMO-FBMC/OQAM sisteminde test edilmiştir. Bunun için, ilgili stratejinin 4x4, 6x6 ve 8x8 anten konfigürasyonlarındaki bit hata oranı başarımları incelenmiştir. Söz konusu performans analizinde, ZF ve ML gibi yaygın olarak bilinen geleneksel sembol dedektörlerinin yanı sıra, DBHS-ML isimli gelişmiş bir metasezgisel tabanlı sembol algılama yöntemi karşılaştırma amaçlı kullanılmıştır. Makalede ele alınan yöntemlerin sembol algılama performanslarını değerlendirmek amacıyla gerçekleştirilen simülasyon çalışmalarına göre, önerilen DCS-ML stratejisi, her bir anten konfigürasyonu için, geleneksel ML yöntemi ile elde edilebilen optimum BER seviyesine, diğer iki yöntemle oranla daha çok yaklaşmıştır. Göstermiş olduğu yüksek sembol algılama performansı sayesinde hem klasik ZF hem de DBHS-ML yönteminden daha düşük BER seviyelerine ulaşarak söz konusu yöntemler üzerinde net bir üstünlük kurmuştur.

Çıkar çatışması

Yazar çıkar çatışması olmadığını beyan etmektedir.

Benzerlik oranı (iThenticate): %6

Kaynaklar

- [1] L. J. Cimini, Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing. IEEE Transactions on Communications, 33 (7), 665-675, 1985. <https://doi.org/10.1109/TCOM.1985.1096357>.
- [2] M. Yıldırım, Subcarrier-interactive dual-mode OFDM. IEEE Communications Letters, 27 (5), 1472-1476, 2023. <https://doi.org/10.1109/lcomm.2023.3248523>.
- [3] Y. Kabalcı, 5g iletişim sistemleri için aday iletim tekniklerinin bit hata oranı başarımlarının araştırılması. Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi, 9 (2), 821-831, 2020. <https://doi.org/10.28948/ngumuh.654386>.

- [4] B. Farhang-Boroujeny, Filter bank multicarrier modulation: A waveform candidate for 5G and beyond. *Advances in Electrical Engineering*, 2014, 1-25, 2014. <https://doi.org/10.1155/2014/482805>.
- [5] R. Nissel, S. Schwarz and M. Rupp, Filter bank multicarrier modulation schemes for future mobile communications. *IEEE Journal on Selected Areas in Communications*, 35 (8), 1768-1782, 2017. <https://doi.org/10.1109/JSAC.2017.2710022>.
- [6] R. Zakaria and D. Le Ruyet, On maximum likelihood MIMO detection in QAM-FBMC systems. 21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 183-187, Istanbul, Turkey, 2010.
- [7] R. Nissel, J. Blumenstein and M. Rupp, Block frequency spreading: A method for low-complexity MIMO in FBMC-OQAM. *IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1-5, Sapporo, Japan, 2017.
- [8] Q. H. Spencer, A. L. Swindlehurst and M. Haardt, Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels. *IEEE Transactions on Signal Processing*, 52 (2), 461-471, 2004. <https://doi.org/10.1109/TSP.2003.821107>.
- [9] X. Zhu and R. D. Murch, Performance analysis of maximum likelihood detection in a MIMO antenna system. *IEEE Transactions on Communications*, 50 (2), 187-191, 2002. <https://doi.org/10.1109/26.983313>.
- [10] Ş. Şimşir and N. Taşpınar, A novel discrete cuckoo search algorithm-based selective mapping technique to minimize the peak-to-average power ratio of universal filtered multicarrier signal. *International Journal of Communication Systems*, 33 (18), 1-16, 2020. <https://doi.org/10.1002/dac.4640>.
- [11] X. Kong, L. Gao, H. Ouyang and S. Li, A simplified binary harmony search algorithm for large scale 0-1 knapsack problems. *Expert Systems with Applications*, 42 (12), 5337-5355, 2015. <https://doi.org/10.1016/j.eswa.2015.02.015>.
- [12] M. N. Seyman, Symbol detection based on back tracking search algorithm in MIMO-NOMA systems. *Computer Systems Science & Engineering*, 40 (2), 795-804, 2022. <http://dx.doi.org/10.32604/CSSE.2022.019734>.
- [13] H. U. Rehman, S. I. Shah, I. Zaka and J. Ahmad, An MBER-BLAST algorithm for OFDM-SDMA communication using particle swarm optimization. *International Journal of Communication Systems*, 24 (2), 185-201, 2011. <https://doi.org/10.1002/dac.1149>.
- [14] A. A. Khan, S. Bashir, M. Naeem, S. I. Shah and X. Li, Symbol detection in spatial multiplexing system using particle swarm optimization meta-heuristics. *International Journal of Communication Systems*, 21 (12), 1239-1257, 2008. <https://doi.org/10.1002/dac.949>.
- [15] M. Mandloi and V. Bhatia, A low-complexity hybrid algorithm based on particle swarm and ant colony optimization for large-MIMO detection. *Expert Systems with Applications*, 50, 66-74, 2016. <https://doi.org/10.1016/j.eswa.2015.12.008>.
- [16] L. Li, W. Meng and S. Ju, A novel artificial bee colony detection algorithm for massive MIMO system. *Wireless Communications and Mobile Computing*, 16 (17), 3139-3152, 2016. <https://doi.org/10.1002/wcm.2754>.
- [17] M. N. Seyman and N. Taşpınar, Symbol detection using the differential evolution algorithm in MIMO-OFDM systems. *Turkish Journal of Electrical Engineering and Computer Sciences*, 21 (2), 373-380, 2013. <https://doi.org/10.3906/elk-1103-16>.
- [18] S. Chen and Y. Wu, Maximum likelihood joint channel and data estimation using genetic algorithms. *IEEE Transactions on Signal Processing*, 46 (5), 1469-1473, 1998. <https://doi.org/10.1109/78.668813>.
- [19] C. Wang, E. K. S. Au, R. D. Murch, W. H. Mow, R. S. Cheng and V. Lau, On the performance of the MIMO zero-forcing receiver in the presence of channel estimation error. *IEEE Transactions on Wireless Communications*, 6 (3), 805-810, 2007. <https://doi.org/10.1109/TWC.2007.05384>.
- [20] X. S. Yang and S. Deb, Cuckoo search via Lévy flights. 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), pp. 210-214, Coimbatore, India, 2009.
- [21] T. T. Nguyen and L. Lampe, On partial transmit sequences for PAR reduction in OFDM systems. *Transactions on Wireless Communications*, 7 (2), 746-755, 2008. <https://doi.org/10.1109/TWC.2008.060664>.

