

Anten Uygulamalarına Yönelik Diferansiyel Evrim Algoritması İçin Performans Artırımı

Performance Improvement of Differential Evolution Algorithm Used For Antenna Applications

Ertuğrul Aksoy¹, Erkan Afacan²

^{1,2}Gazi Üniversitesi, Mühendislik Mimarlık Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü,
06570, Maltepe, Ankara, Türkiye
ertugrulaksoy@gazi.edu.tr, e.afacan@gazi.edu.tr

Özet

Bu bildiri, optimizasyon parametrelerini türden bağımsızlaştıran bir yöntemin diferansiyel evrim algoritması ile performansı incelenmiştir. İki yöntem birleştirilerek diferansiyel evrim algoritmasında sağlamlık bakımından bir performans artırımı sağlanmıştır.

Abstract

In this paper, an investigation is made between a method that makes optimization parameters independent from type and differential evolution algorithm. With combining these two methods a performance improvement in differential evolution algorithm is obtained in terms of robustness.

1. GİRİŞ

Yakın zamanda Haupt farklı türden parametrelere sahip optimizasyon problemlerinde kullanılabilecek Karışık Tamsayı Genetik Algoritma (KTGA) adında bir yöntem önermiştir [1]. Bu yöntem parametreleri türden bağımsızlaştıran sürekli bir arama uzayında arama işlemini gerçekleştirmektedir. Maliyet fonksiyonunda yapılan dönüşümler sayesinde parametreler türden bağımsız olmakta ve sonuçta problemin tümü tek bir algoritma sayesinde sonuçlandırılmaktadır.

[1] de önerilen algoritma sürekli bir uzayında arama yaptığından aklı son zamanlarda yayınlanmış olan bir evrimsel optimizasyon metodu olan [2] ve çeşitli algoritmalarla karşılaştırması yapılmış ve üstünlüğü gösterilmiş [3] bir algoritma olan Diferansiyel Evrim Algoritması (DEA) gelmektedir.

Bu bildiri KTGA ile sunulan konseptin DEA uygulaması ve bunların birleşimi ile her iki algoritmadan daha sağlam sonuç veren bir sentez önerilmiştir. Sonuçlar sağlamlık ve en iyilerin yakınsama hızları bakımından değerlendirilmiş ve 21 elemanlı doğrusal bir dizi düşünülerek örneklendirilmiştir.

2. KTGA KONSEPTİ

Haupt 2007 yılında farklı türden optimizasyon parametreleri içeren problemlerin çözümünde kullanılabilecek bir yöntem

olarak KTGA konseptini önermiştir [1]. Bu konsept standart genetik algoritmanın modifiye edilmiş bir şeklidir ve tüm aramanın sürekli olabildiği sonradan örneklenebileceğini savunmaktadır.

Bu yöntemde arama uzayı tüm optimizasyon değişkenleri [0,1] kapalı aralığı olmaktadır ve değişkenler sürekli kabul edilmektedir. Sürekli uzayda optimize edilen değişkenler maliyet fonksiyonu her çağırıldığında değişkenin tipine göre, [1]'de yer alan orijinal notasyonla, eşitlik (1)'de verilen dönüşümler ile olmaları gereken türe dönüştürülmektedirler. Bu sayede değişken türleri sürekli olarak optimize edilmelerine rağmen maliyet fonksiyonunda olmaları gereken türde yerlerini almaktadırlar.

$$\begin{aligned}x_n &= (x_{maks} - x_{min})v_{m,n} + x_{min} \\I_n &= \text{rounddown}\{(I_{maks} - I_{min} + 1)v_{m,n}\} + I_{min} \\b_n &= \text{round}\{v_{m,n}\}\end{aligned}\quad (1)$$

Bu ifadede m popülasyon indeksi n ise değişken indeksidir. min ve maks indisleri değişken sınırlarını belirlemektedirler. rounddown sonucu bir sonraki en küçük tamsayıya yuvarlayan, round ise sonucu en yakın tamsayıya yuvarlayan fonksiyonları temsil etmektedirler. Algoritma seçim için iki kromozomlu turnuva metodu ve çaprazlama olarak da düzgün dağılımlı çaprazlama kullanılmaktadır. Mutasyon işlemi için eşitlik (2)'de verilen fonksiyon kullanılmaktadır [1]:

$$\text{kromozom} = \text{rem}\{\alpha_r \beta_{m,n} v_{m,n} + \beta_{m,n} v_{m,n}\} \quad (2)$$

bu ifadede m popülasyon indeksi n değişken indeksidir. α_r mutasyon faktörü, "rem" kalan operatörü, β_r ise rasgele düzeltme faktörü olarak adlandırılan rasgele bir gerçel sayıdır. α_r ve β_r , $0 \leq \alpha_r, \beta_r \leq 1$ aralığındadır.

3. DİFERANSİYEL EVRİM

Diferansiyel evrim sürekli bir uzayda arama işlemi yapan popülasyon tabanlı bir optimizasyon aracıdır [4] ve kısaca şu şekilde özetlenebilir:

- (i) Başlangıç popülasyonunu (aksi bir durum olmadıkça) düzgün dağılımlı olarak oluştur.
- (ii) Popülasyon için maliyet fonksiyonu “ f ” i kullanarak maliyeti hesapla.
- (iii) Parametre vektörlerini kullanarak bir mutant vektör oluştur.
- (iv) Çaprazlama ile bir deneme vektörü oluştur.
- (v) Oluşturulan bu deneme vektörünün maliyetini “ f ” vasıtasıyla hesapla.
- (vi) Deneme vektörü ile popülasyonun j . elemanını karşılaştır ve yeni popülasyon için uygun olanını seç.
- (vii) j popülasyon indeksine ulaşmaya kadar (iii). basamağa dön.
- (viii) Yeni popülasyonu oluştur ve durma koşulu sağlanıncaya kadar (ii). basamağa dön.

Çoğu evrimsel algoritmada olduğu gibi diferansiyel evrimde de yakınsamanın başlıca etkenleri seçim, mutasyon ve çaprazlama işlemleridir. DEA genel olarak seçim için bire-bir metodunu kullanmaktadır ve çaprazlama için düzgün dağılımlı çaprazlama tercih edilmektedir fakat bu durum değişiklik gösterebilir. Mutasyon şemaları DEA için en çok üzerinde durulan konulardan birisidir ve literatürde farklı mutasyon şemaları mevcuttur [4]. Bu bildiride DEA'nın “best/1/bin” [4] ve “best/1/bin_with_uniform_jitter” [4] şemaları ile KTGA'da kullanılan mutasyon operatörünün DEA uygulaması kullanılmıştır. Sonraki bölümde karşılaştırmalı olarak gösterileceği üzere “DE/rand/1/bin” mutasyon şeması ile KTGA mutasyon şemasının birleştirilmesi daha sağlam sonuçlara götürmektedir.

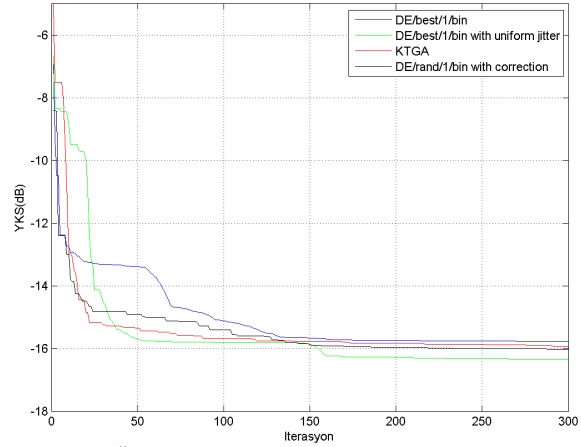
4. KARŞILAŞTIRMA

Karşılaştırma için elemanları arası mesafe yarım dalgaboyu olan 21 elemanlı doğrusal bir dizinin merkezdeki eleman referans olmak üzere ve elemanlar merkeze göre simetrik olduğu kabul edildiğinde sadece faz optimizasyonu ile örüntüsünün şekillendirilmesi senaryosu göz önüne alınmıştır. Bu senaryoda örüntünün kritik parametrelerinden sadece yan kulakçık seviyesi (YKS) göz önüne alınmıştır. İlk örnek için faz ifadesinin $[0, 2\pi]$ aralığında gerçel olduğu kabul edilmiştir. İkinci örnekte ise ilk 6 elemanın fazlarının sürekli olduğu dıştaki 4'er elemanın fazlarının ise $[0, 2\pi]$ aralığında $\pi/4$ aralıklarla değerler aldığı kabul edilmiştir. İlk örnek için toplam optimizasyon değişken sayısı 10 dur. İkinci örnekte ise son 4 eleman 3-bit ile ikili kodlanmış ve toplam değişken sayısı 18 olarak alınmıştır. Karşılaştırma 40 bağımsız deney üzerinden yapılmıştır. Her deney için sonlandırma koşulu 300 iterasyondur ve 300 iterasyon sonucu elde edilen sonuç algoritma çıktısı olarak alınmaktadır. Tüm DEA çeşitleri için mutasyon faktörü 0.6, çaprazlama faktörü 0.95 olarak ayarlanmıştır. KTGA için mutasyon faktörü 0.05, çaprazlama ve mutasyon olasılıkları 1 olarak ayarlanmıştır. DEA ve KTGA için popülasyon büyüklüğü 40 alınmıştır. Tüm algoritmalar için maliyet fonksiyonu eşitlik (3) de verilmiştir.

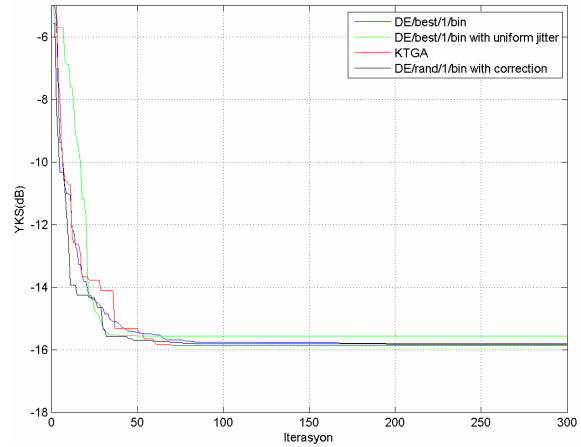
$$F = \max\{(AF(\theta))_n\}, \theta_0 \leq \theta \leq 90^\circ \quad (3)$$

40 bağımsız deney arasından en iyi sonuçların yakınsama grafikleri örnek1 için şekil-1'de, örnek2 içinse şekil-2'de verilmiştir. Bu sonuçlar ile elde edilen dizinin ışıma örüntüleri

sonuçların doğrulanması için sırasıyla şekil-3 ve şekil-4 de verilmiştir. 300 iterasyon sonucu tüm rakiplerin verdiği en iyi sonuçlar -15.5dB civarında çıkmaktadır. Sonuca yakınsama hızları ise en iyi çözümler için birbirlerine çok yakın çıkmaktadır.



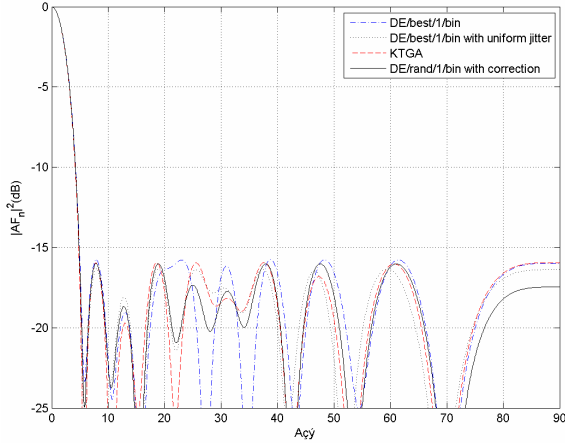
Şekil-1: Örnek-1 için en iyi çözümlerin yakınsama grafiği



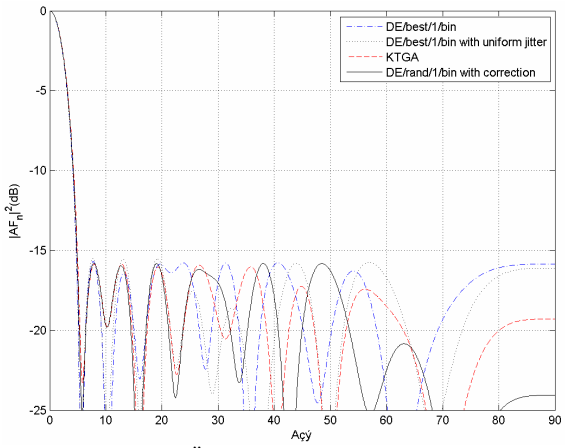
Şekil-2: Örnek-2 için en iyi çözümlerin yakınsama grafiği

En iyi çözümler için sonuçlar çoğu zaman bu tip algoritmalar için yanlıcı sonuçlar verebilmektedirler. Tek bir deney yerine birden fazla deneyin sonuçları ile algoritmalar hakkında daha anlamlı sonuçlar elde etmek mümkündür. Bu bakımdan 40 bağımsız deneyin ortalaması algoritmaların ne sağlamlıkta yakınsadıklarını göstermek açısından daha anlamlıdır. 40 bağımsız deneyin ortalamasının sonuçları örnek1 ve örnek2 için sırasıyla şekil-5 ve şekil-6 da verilmiştir.

Şekil-5 ve Şekil-6 dan da görüleceği üzere tek bir denemede yakın sonuçlar veren algoritmalar ortalama bakımından farklılık arz etmektedir. Şekil-5 ve Şekil-6 da verilen sonuçlar ile standart genetik algoritmalara üstünlük sağlayan standart DEA'nın [3] KTGA'ya karşı aynı üstünlüğünü sürdüremediği görülmektedir. Fakat KTGA da kullanılan mutasyon şeması ile standart DEA'nın sentezi ile her iki algoritmadan daha sağlam sonuçlar verdiği Şekil-5 ve Şekil-6'dan görülmektedir.



Şekil-3: Örnek-1 için ışma örüntüleri



Şekil-4: Örnek-2 için ışma örüntüleri

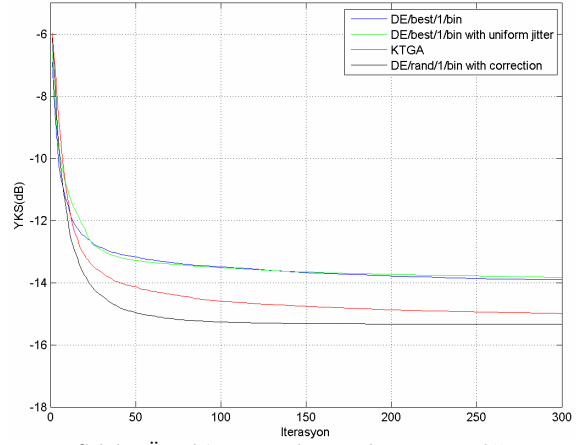
Algoritmaların hangi sağlamlıkta sonuç verdiklerinin gözlemlenmesi adına yapılabilecek bir başka deney ise algoritmaların belirli bir referans değerine altına hangi oranla yakınsadığı olarak düşünülebilir. Referans olarak 21 elemanlı doğrusal dizinin sıfır fazlı durumunun YKS'si olan -13,19dB alınırsa algoritmaların 40 bağımsız deney ile 300 iterasyon sonucu Örnek1 için elde edilen sonuçlar:

DE/best/1/bin=30/40
DE/best/1/bin_with_uniform_jitter=25/40
KTGA=38/40
DE/rand/1/bin_with_correction=40/40

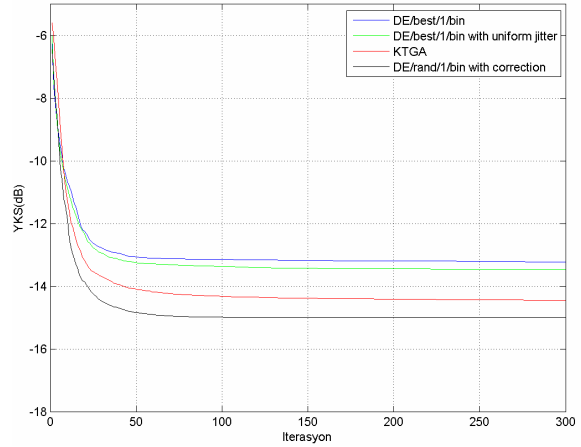
olarak gözlemlenmiştir. Örnek1 için referans biraz daha artırılıp -15dB olarak alınırsa algoritmaların sağlamlık testi biraz daha zorlaştırılmış olacaktır. Referans -15dB olarak alınırsa elde edilen sonuçlar:

DE/best/1/bin=10/40
DE/best/1/bin_with_uniform_jitter=12/40
KTGA=25/40
DE/rand/1/bin_with_correction=34/40

olarak gözlemlenmiştir. Oranlardan da anlaşılacağı üzere sentez sonucu meydana gelen algoritma değerlerine göre örnek1 gözönüne alındığında daha sağlam sonuçlar vermektedir.



Şekil-5: Örnek1 için ortalama yakınsama grafiği



Şekil-6: Örnek2 için ortalama yakınsama grafiği

Örnek2 için aynı prosedür tekrarlanırsa -13,19dB referans değeri için sonuçlar:

DE/best/1/bin=22/40
DE/best/1/bin_with_uniform_jitter=23/40
KTGA=36/40
DE/rand/1/bin_with_correction=39/40

ve -15dB referans değeri için:

DE/best/1/bin=6/40
DE/best/1/bin_with_uniform_jitter=7/40
KTGA=14/40
DE/rand/1/bin_with_correction=27/40

olarak gözlemlenmiştir. Oranlardan da görüleceği üzere Örnek2 için de sentez sonucu meydana gelen algoritma değerlerine göre daha sağlam sonuçlar vermektedir.

5. SONUÇ

Farklı deęişken tiplerini içeren iki örnek üzerinde saęlamlık testleri yapılmış ve sonuç olarak KTGA ve DEA'nın belirli özelliklerinin birlikte kullanılması ile oluşturulan sentezin baęımsız baz algoritmalarına göre daha saęlam sonuçlar verdiği gösterilmiştir.

6. REFERANSLAR

[1] R. L. Haupt , "Antenna Design With a Mixed Integer Genetic Algorithm", IEEE Antennas and Propagat., vol.55, no.3, pp.577-582, 2007.

[2] R. Storn , K. Price, "Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces", J. of Global Optimization, Vol.11, No. 4, pp.341-359, 1997.

[3] S.W. Yang, Y.B. Gan, A.Y. Qing, "Antenna-array pattern nulling using a differential evolution algorithm", Int. J. Of RF and Microwave Computer Aided Engineering, Vol. 14, No.1, pp.57-63, 2004.

[4] K. V. Price, R. M. Storn, J.A. Lampien, "Differential Evolution: A Practical Approach to Global Optimization", Springer, 2005.