

# Optimizasyonun Optimizasyonu Yaklaşımıyla Dağılım Fonksiyonu Tabanlı Kral Kelebeği Optimizasyon Algoritmasının Performansının Artırılması

## Performance Enhancement of Distribution Function Based Monarch Butterfly Optimization Algorithm with Optimization to Optimization Approach

Mehmet Akpamukçu<sup>\*1</sup> , Abdullah Ateş<sup>1</sup> 

<sup>1</sup>Bilgisayar Mühendisliği Bölümü, İnönü Üniversitesi, Malatya, Türkiye  
(akpamukcu@gmail.com, abduallah.ates@inonu.edu.tr)

Received: Sep.3, 2021

Accepted: Sep.26, 2021

Published: Oct.20, 2021

**Abstract**— In this study, the parameters of the distribution functions were adjusted with the optimization to optimization approach to improve the performance of the distribution function-based monarch butterfly optimization algorithm (MBO). For this, the random number generation processes, which greatly affect the flow of stochastic algorithms, were examined and the effect of distribution functions on these processes was determined. Then, the importance of parameter selection in the operation of distribution functions has been determined. It has been seen that the distribution function will be more effective with appropriate parameter selections. At this point, the distribution functions that can be used in the random number generation in the main target algorithm were tried to be determined with appropriate parameters with an upper auxiliary optimization algorithm. In conclusion; with the approach of optimization to optimization, the performance of the target algorithm has been tried to be increased and concrete results are presented in comparison with the tests made on the most used benchmark functions in the literature.

**Keywords:** optimization; distribution functions; stochastic algorithms; optimization to optimization.

### 1. Giriş

Teorik ve gerçek mühendislik problemlerinde en uygun çözümün bulunması için birçok optimizasyon algoritması kullanılır (Ateş et al., 2017; Abdullah Ateş & Yeroğlu, 2018). Bu algoritmalar içerisinde stokastik süreçler içeren nümerik optimizasyon algoritmaları önemli bir yer tutar ve günümüzde de çok geniş bir alanda kullanılır (Ateş, 2021; Elbes et al., 2019). Bu tür algoritmaların birçok alandaki problemde kullanılmasının en önemli sebebi çoğu problemin belirsizlik içeren bir çözüm uzayına sahip olmasıdır (Akdag et al., 2020; A. Ateş & Yeroğlu, 2016). Tam olarak belli olmayan veya geniş bir aralıkta belirlenebilen parametrelere sahip olan bu problemlerde geniş bir çözüm uzayında arama yapılması gerektiğinden stokastik tabanlı optimizasyon algoritmaları ile daha iyi çözümler elde edilebilir. Stokastik algoritmalar ise temelinde rasgele sayı üretme süreçleri içeren yapılara sahiptir. Bu süreçler her ne kadar mühendislikteki kesin deterministik ve analitik yaklaşımlarla ters düşse de, ele alınan problemlerin yapısı gereği kullanılması gereken faktörler olarak göze çarpmaktadır. Bu anlamda literatürde doğadan esinlenen; fizik kurallarından esinlenen; hayvanların besin bulmasından, göç hareketlerine kadar birçok noktada hayvan davranışlarından esinlenen algoritmalar bulunmaktadır (van

Laarhoven & Aarts, 1987; Glover, 1989; Glover, 1990; Guha et al., 2020; Khadanga et al., 2020; Gaidhane & Nigam, 2018; Liang et al., 2020). Bu algoritmalar esinlendiği fizik kuralını veya hayvan davranışını simüle ederek, gerçek hayattaki belirsizlik ortamının içinden bir çözüm üretimini, teorik veya gerçek mühendislik problemine uygulamaktadır. Bu arayışlar genellikle bazı noktalarda stokastik bir karakter içeren algoritmalarla sonuçlanmaktadır. Bu özellik çoğu zaman algoritmalara bir güç kazandırır da algoritma üreticisi açısından çoğu zaman kontrolün kaybedilmesine yol açabilir. Çünkü rasgele sayı üretimi genellikle literatürde belli bir yöntem ile yapılmakta ve bu noktanın önemi üzerinde fazla durulmamaktadır.

Bu çalışmada güncel bir stokastik algoritma olan Kral Kelebeği Optimizasyon algoritması (Monarch Butterfly Optimization (MBO)) performansı artırılacak ana hedef algoritma olarak seçilmiştir (Wang et al., 2019). Bu algoritmanın performansının artırılması için algoritmanın rasgele sayı üretim noktaları hedef olarak alınmıştır. Bu amaç rasgele sayı üretimini sağlayan dağılım fonksiyonu bileşenini, uniform (düzgün) dağılım fonksiyonu dışındaki farklı dağılım fonksiyonlarının en uygun parametrelerinin belirlenmesini sağlayacak üst bir optimizasyon algoritması olan rastsal parametre vektör optimizasyon algoritması (RPVO) ile sağlanmıştır. Dolayısıyla çift katmanlı bir optimizasyon tekniği olan optimizasyonun optimizasyonu yaklaşımı uygulanmıştır (Ates & Akpamukcu, 2021). Bu yaklaşımda KKO ana algoritma, RPVO ise yardımcı algoritma olarak belirlenmiştir. Bu noktada farklı dağılım fonksiyonu tabanlı KKO yöntemi ilgili problemi çözerken, RPVO yöntemi ise ilgili problem için uygun dağılım fonksiyon parametrelerini belirlemektedir. Böylece ana hedef algoritma için rasgele sayı üretiminde en uygun dağılım fonksiyonu ve parametre kombinasyonları belirlenmiş ve algoritmanın performansının artırılması hedeflenmiştir. Önerilen yöntem benchmark fonksiyonları üzerinden test edilmiş ve sonuçlar karşılaştırmalı olarak sunulmuştur.

## 2. Optimizasyonun Optimizasyonu Yaklaşımı

Optimizasyon algoritmaları bir problemin çözüm uzayındaki hedeflenen amaç fonksiyonuna göre en iyi çözüme ulaşmasını sağlayan metotlardır. Algoritmalar oluşturulurken bu algoritmanın bileşenleri algoritmayı optimum bir şekilde sonuca götüreceği şekilde ayarlanmaktadır. Ancak bu bileşenlerin geniş bir aralık içerisinde seçilebilme olasılığı ve bazı belirsizlikler en uygun kombinasyonun seçimini oldukça zorlaştırmaktadır. Bu noktada bir optimizasyon algoritmasının en etkili şekilde çalışabileceği yapının oluşturulması için bileşenlerinin veya parametrelerinin optimize edilmesi gerekir. Buna optimizasyonun optimizasyonu yaklaşımı adı verilmektedir (Ates & Akpamukcu, 2021). Bu yaklaşımda bir ana algoritma ve bir yardımcı algoritma seçilmektedir. Ana algoritmada optimizasyona tabi tutulacak parametreler tespit edilmektedir. Bu parametreler yardımcı algoritma aracılığıyla parametrelerin çözüm aralığı içerisinde ana algoritmanın en iyi performansı üretebileceği şekilde yardımcı algoritma aracılığıyla belirlenir. Böylece ana algoritmanın performansı açısından, orijinal konfigürasyonuna göre daha artırılmış bir biçimde çıktı elde edilmektedir. Çok küçük performans artışlarının gerçek mühendislik veya teorik problemlerde sağlayacağı kazanımlar göz önünde bulundurulduğunda, optimizasyonun optimizasyonu yaklaşımıyla adeta ilgili problem için hedef algoritmanın probleme göre en iyi konfigürasyonun belirlenmesi hedeflenmekte ve nihayetinde oluşan çıktı ile performans artışı sağlanmaktadır. Optimizasyonun optimizasyonu yaklaşımı her ne kadar ana algoritmanın parametrelerini optimize etmeye çalışsa da algoritmanın temel işleyişini değiştirmemekte ve algoritmanın temel felsefesini korumaktadır (Ates & Akpamukcu, 2021). Dolayısıyla bu algoritmada tamamen farklılaşmış ve yeni bir algoritmadan bahsetmek zordur. Hedeflenen amaç, var olan algoritmayı belirlenen problem özelinde daha iyi işleyen etkili bir algoritma haline getirmektir.

### 2.1. Dağılım fonksiyonu Tabanlı KKO

Bu çalışmada ana optimize edilecek algoritma olarak KKO yöntemi belirlenmiştir. Bu algoritma kral keleklerinin göç hareketlerinden esinlenerek problemlerin çözümünü hedefleyen stokastik bir algoritmadır. Bu algoritmada kral keleklerinin iki ana yaşam bölgesi belirlenmiş ve bu iki bölge arasında kral kelek popülasyonu simüle edilmiştir. Bu benzetişim; göç hareketleri sırasında kelek popülasyonunun doğumlar, ölümler gibi olaylar hesaba katılarak temel iki işlem olan göç işlem (migration operator) ve kelek düzenleme işlemi (butterfly adjusting operator) vasıtasıyla yapılmaktadır. Bu algoritmada esasen her bir kelek problem için bir çözüm adayı olmakta ve her

kelebeğe atanmış olan özellikler de çözümün parametrelerini temsil etmektedir. Algoritma boyunca her nesilde popülasyon sabit tutulmakta ve nesil geçişlerinde en iyi kelebekler yani çözüm adayları bir sonraki nesile geçiş yaptırılmakta, nesil döngüleri bir iterasyon biçiminde çalıştırılmakta ve belirlenen nesil sayısına ulaşıldığında ortaya çıkan en iyi kelebek ve parametreleri çözüm olarak sunulmaktadır (Wang et al., 2019).

Bu çalışmada KKO algoritmasının stokastik olmasını sağlayan rasgele sayı üretim noktalarında denenecek en uygun parametreleri almış dağılım fonksiyonlarıyla daha iyi sonuçlar üretecek bir KKO algoritması hedeflenmiş ve dağılım fonksiyonu tabanlı KKO algoritması oluşturulmuştur. Bunun yanında KKO algoritmasının esnek 4 parametresi de optimizasyona tabi tutulmuştur. Dağılım fonksiyonlarının parametrelerinin normal standart parametreleriyle değil de problem ve algoritmaya uygun bir biçimde belirlenmesiyle performans artışı hedeflenmiştir.

## 2.2. Kullanılan Dağılım Fonksiyonları

Bu çalışmada belirlenen hedeflere ulaşmak için Uniform dağılım fonksiyonunun yerine kullanılmak üzere beş dağılım fonksiyonu belirlenmiştir. Bu dağılım fonksiyonları Beta, Normal, Exponential, Poisson ve Rayleigh dağılım fonksiyonlarıdır.

Normal dağılım fonksiyonu; Gauss dağılımı olarak adlandırılan iki parametrelili bir eğri ailesidir. Modelleme için normal dağılımı kullanmanın mantığı, herhangi bir dağılımdan bağımsız olarak sonlu ortalama ve varyans ile örneklerin toplamının, örneklem boyutu sonsuza giderken normal dağılıma yaklaştığını belirten Merkezi Limit Teoremi'dir. Ortalama (Mean) ve Standart Sapma (Standard Deviation) olarak iki parametresi bulunmaktadır (*Probability Density Function*, 2020). Beta dağılımı, sıfır olmadıkları için yalnızca sıfır aralığında (0 1) benzersiz olan bir eğri ailesini tanımlar. Fonksiyonun daha genel bir versiyonu, aralığın uç noktalarına parametreler atar. Birinci Şekil Parametresi (First Shape Parameter) ve İkinci Şekil Parametresi (Second Shape Parameter) adında iki parametresi bulunur (*Probability Density Function*, 2020).

Üstel dağılım, tek parametrelili bir eğri ailesidir. Üstel dağılım modelleri, ne kadar beklediğinizden bağımsız olarak ek bir süre beklerken süreler bekler. Ortalama (Mean) adında tek bir parametresi bulunur (*Probability Density Function*, 2020). Poisson dağılımı, rastgele bir olayın meydana gelme sayısını modelleyen tek parametrelili bir eğri ailesidir. Bu dağılım belirli bir zamana, mesafeye, alana vb. dayalıdır. Rastgele bir olayın meydana gelme sayısını nicelik olarak saymayı içeren uygulamalar için uygundur. Ortalama (Mean) adında tek bir parametresi bulunur (*Probability Density Function*, 2020). Rayleigh dağılımı, Weibull dağılımının özel bir örneğidir. İletişim teorisinde, Nakagami dağılımları, Rician dağılımları ve Rayleigh dağılımları, bir alıcıya çeşitli şekillerde ulaşan dağınık sinyalleri modellemek için kullanılır. Ölçek (Scale) adında tek bir parametresi bulunur (*Probability Density Function*, 2020).

## 2.3. Rastsal Parametre Vektör Optimizasyon Yöntemi

Bu çalışmada ana algoritmayı optimize edecek yardımcı algoritma olarak; Rastsal Parametre Vektör Optimizasyon (Stochastics Multi-parameters Divergence Optimization (SMDO)) algoritması seçilmiştir (Yeroğlu & Ateş, 2014). RPVO algoritması parametreler için önceden belirlenen başlangıç değerlerinden başlayarak çözüm uzayında güncel bulunulan noktadan ileri ve geri deneme adımlarıyla oluşan hata değerine göre gidiş yolunu belirleyen bir algoritmadır. Bulunulan nokta etrafında ileri ve geri yöndeki adım miktarı stokastik bir biçimde belirlenerek yerel nokta tuzağına düşülmekten kaçınılmaktadır. Eğer ilgili adımlar sabit bir şekilde belirlenirse yerel nokta tuzağına düşülmesi çok olası bir durumdur. Bütün parametreler belirlenen adımlar ile oluşan bir matris üzerinden denemekte oluşan hata değerinin amaç fonksiyonunun hata değerine göre algoritma ilerlemektedir (Yeroğlu & Ateş, 2014).

RPVO algoritması her ne kadar stokastik bir süreç içerse de deneme yanılmada izlediği analitik yöntem bu algoritmayı çalışmamızda başka bir algoritmanın parametrelerini optimize edecek yardımcı bir algoritma olarak seçme konusunda güçlü bir aday yapmıştır.

### 3. Benchmark Fonksiyonu Sonuçları

Optimizasyonun optimizasyonu yaklaşımı ile RPVO algoritmasının yardımcı algoritma, KKO'nun da ana algoritma olarak kullanılmasıyla oluşturulan yapı çalışmamızda literatürde sıklıkla kullanılan 10 benchmark fonksiyonu ile denenmiştir. Çıkan sonuçlar Tablo 1'de karşılaştırmalı olarak verilmiştir.

Bu sonuçlara göre; Matyas fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential ve Rayleigh dağılım fonksiyonlarının oluşturduğu 4 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Ackley fonksiyonu ile yapılan denemelerde Normal, Exponential, Poisson ve Rayleigh dağılım fonksiyonlarının oluşturduğu 4 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Rastrigin fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential, Poisson ve Rayleigh dağılım fonksiyonlarının oluşturduğu 5 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Rosenbrock fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential ve Poisson dağılım fonksiyonlarının oluşturduğu 4 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Zakharov fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential ve Poisson dağılım fonksiyonlarının oluşturduğu 4 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Sphere fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential ve Rayleigh dağılım fonksiyonlarının oluşturduğu 4 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Beale fonksiyonu ile yapılan denemelerde sadece Poisson dağılım fonksiyonuyla uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Booth fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential ve Poisson dağılım fonksiyonlarının oluşturduğu 4 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

DixonPrice fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential, Poisson ve Rayleigh dağılım fonksiyonlarının oluşturduğu 5 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Levy fonksiyonu ile yapılan denemelerde Beta, Normal, Exponential ve Poisson dağılım fonksiyonlarının oluşturduğu 4 dağılım fonksiyonu, oluşan uygun parametrelerle Uniform dağılım fonksiyonuna göre daha iyi sonuçlar elde etmiştir.

Sonuçlardan da görüldüğü gibi farklı dağılım fonksiyonu tabanlı KKO yönteminde dağılım fonksiyonu parametrelerinin ve KKO parametrelerinin optimizasyonun optimizasyonu yaklaşımı ile klasik KKO yönteminden daha iyi sonuçların elde edilebileceği gösterilmiştir. Bu şekilde temel KKO yönteminin felsefesi değiştirilmeden algoritmanın performansının artırılacağı gösterilmiştir.

**Tablo 1.** Benchmark Testleri Sonuçları

	REFERANS DEĞERİ	KKO (ORJİNAL)	KKO İLE RPVO ORTALAMA DEĞER (UNIFORM)	KKO İLE RPVO ORTALAMA DEĞER (BETA)	KKO İLE RPVO ORTALAMA DEĞER (NORMAL)	KKO İLE RPVO ORTALAMA DEĞER (EXPONENTIAL)	KKO İLE RPVO ORTALAMA DEĞER (POISSON)	KKO İLE RPVO ORTALAMA DEĞER (RAYLEIGH)
<b>MATYAS</b> MBOParam1 MBOParam2 MBOParam3 MBOParam4 DistParam1 DistParam2	0	0.01380	5.8758e-07	2.9308e-07 1 1 1.2 0.4160	2.24660e-07 2 1.0894 1.2146 0.4160	3.3107e-07 2 0.8915 1.2 0.3939	6.0804e-07 12 0.9335 1.2 0.2854	3.6841e-06 2 0.9862 1.2935 0.4160
<b>ACKLEY</b> MBOParam1 MBOParam2 MBOParam3 MBOParam4 DistParam1 DistParam2	0	11.8038	0.0367	0.0367 10 0.8221 1.0982 0.4007 1.0834 1.1409	0.0201 9 1.0930 1.1493 0.3674 0 0.9184	0.0199 6 1.2207 1.2000 0.5712 1.1612 0	0.0165 5 1.0307 1.1853 0.3383 0.9958 0	0.0265 12 1.0162 1.0996 0.4456 0.9319 0
<b>RASTRIGIN</b> MBOParam1 MBOParam2 MBOParam3 MBOParam4 DistParam1 DistParam2	0	90.8993	0.1005	0.0768 1 0.7544 1.2298 0.4160 0.9891 0.9780	0.0384 10 1.1292 1.2490 0.5558 0 0.9168	0.0284 2 0.9996 1.1841 0.4160 1.1851 0	0.0273 9 1.2097 1.1624 0.3169 0.9196 0	0.0309 3 1.0465 1.3725 0.4489 0.7302 0
<b>ROSENBROCK</b> MBOParam1 MBOParam2 MBOParam3 MBOParam4 DistParam1 DistParam2	0	1.1570e+03	0.2791	0.1810 7 1.0052 1.3010 0.4485 0.9318 1	0.1971 4 0.9527 1.1817 0.3600 0 1	0.0757 3 1.0862 1.1956 0.4791 1.0086 0	0.0979 10 1.1816 1.2017 0.4226 0.9673 0	0.4571 4 0.8989 1.1373 0.4617 0.9601 0
<b>ZAKHAROV</b> MBOParam1 MBOParam2 MBOParam3	0	475.36	0.1740	0.0899 14 1.1680 1.2000	0.1170 5 0.9825 1.2052	0.0618 5 1.0144 1.3246	0.0333 5 1.0813 1.2522	1.0890 2 0.8443 1.2464

MBOPParam4				0.5442	0.4920	0.4364	0.3282	0.4897
DistParam1				0.8289	0.0632	0.9688	0.9126	0.7523
DistParam2				0.9887	1.0071	0	0	0
<b>SPHERE</b>	0	23.5524	5.2710e-04	3.1308e-04	2.4465e-04	2.1068e-04	6.4576e-05	2.1281e-04
MBOPParam1				15	8	6	1	7
MBOPParam2				0.8444	0.8753	1.2391	0.9738	1.1038
MBOPParam3				1.2374	1.2613	1.2339	1.2569	1.3559
MBOPParam4				0.5092	0.6081	0.5376	0.4160	0.4918
DistParam1				0.9823	0.0899	1.0915	1.1381	0.8743
DistParam2				1.2129	0.9251	0	0	0
<b>BEALE</b>	0	0.5360	4.3827e-05	5.3602e-05	4.6656e-05	5.0510e-05	3.2793e-05	8.3889e-05
MBOPParam1				2	6	2	3	2
MBOPParam2				1.0975	1.1384	0.9666	1.0493	1.0222
MBOPParam3				1.2313	1.4305	1.1584	1.3816	1.1769
MBOPParam4				0.3541	0.3635	0.4204	0.3212	0.3759
DistParam1				1	0	1.0594	1	0.9933
DistParam2				1	1.0474	0	0	0
<b>BOOTH</b>	0	2.0084	1.4186e-04	5.2867e-05	1.1099e-04	6.0812e-05	1.3762e-04	5.3627e-04
MBOPParam1				4	4	6	10	2
MBOPParam2				1.1632	0.8853	1	1.1136	1.2502
MBOPParam3				1.1665	1.1306	1.4679	1.2230	1.3926
MBOPParam4				0.4818	0.2947	0.4571	0.4880	0.4131
DistParam1				0.9767	0.0723	1.0582	0.8231	0.9654
DistParam2				1.1415	1.0111	0	0	0
<b>DIXONPRICE</b>	0	3.4355e+05	0.7664	0.5822	0.6829	0.5657	0.4368	0.4259
MBOPParam1				1	2	4	14	8
MBOPParam2				1.1175	1.1006	1.1045	0.9507	0.7965
MBOPParam3				1.2599	1.1647	1.2322	1.1476	1.1886
MBOPParam4				0.5935	0.4708	0.3542	0.3402	0.4160
DistParam1				1.0239	0.0554	1.0333	0.9256	1.1162
DistParam2				1.0945	0.8505	0	0	0
<b>LEVY</b>	0	20.4396	1.3742e-04	8.7495e-05	4.9139e-05	5.7683e-05	9.2682e-06	1.4897e-04
MBOPParam1				12	8	8	2	2
MBOPParam2				1.1303	1	0.8620	1.0141	0.9781
MBOPParam3				1.1302	1.3299	1.2548	1.1696	1.2221
MBOPParam4				0.6843	0.4150	0.5362	0.5676	0.5170
DistParam1				1.0052	0	1	0.9090	0.9717
DistParam2				1.0465	0.9019	0	0	0

## 4. Sonular

Bu alıřmada optimizasyonun optimizasyonu yaklařımıyla yardımcı bir algoritma olan RPVO vasıtasıyla rasgele sayı üretme süreçlerinde farklı dağılım fonksiyonlarının parametreleri optimize edilerek seçilen KKO ana algoritmasının performansının artırılması amaçlanmıştır. Burada özellikle RPVO algoritmasının kontrollü ilerleme ve deneme stratejisi sayesinde her bir dağılım fonksiyonun uygun parametrelerinin bulunmasıyla oluşan performans artışıyla elde edilen sonuçlar Bölüm 3'te gösterilmiştir.

Bu alıřmada özellikle iki algoritmanın ana süreçleri ve temel felsefesi deęiřtirilmeden sadece kritik esnek parametrelerinin optimizasyonu yapılmıř ve var olan algoritmanın optimizasyon gücünün artırılması hedeflenmiştir. Bu esnek parametrelerin dıřında yapılan müdahalelerin var olan başarılı algoritmik yapıyı deęiřtirebileceęinden dolayı, sadece algoritmada deęiřtirilebilecek esneklikteki parametreler optimize edilmiştir. Bu şartlar altında uyarlanan yapıyla yapılan denemeler sonucunda, uygulanan yaklařımın pozitif sonuçları görülmüřtür.

Bu sonuçlar, hem optimizasyonun optimizasyonu yaklařımının geçerlilięini kuvvetlendirmekte hem de stokastik süreçlerde uygun dağılım fonksiyonlarının uygun parametrelerle kullanımının pozitif etkilerini göstermektedir. Dolayısıyla stokastik süreçlerde dağılım fonksiyonlarının uygun parametrelerle ayarlanmasıyla oluşan pozitif etkinin teorik benchmark fonksiyonlar üzerindeki etkisi, gerçek mühendislik problemleri üzerinde de olumlu etkiler göstereceęinin ipuçlarını vermektedir. ok küçük performans artışlarının gerçek mühendislik problemleri üzerindeki etkinlik, verimlilik ve maliyet açısından katkıları düşünöldüğünde, bu alıřmadan ıkan sonuçların literatür açısından ilerideki alıřmalar bakımından cesaret verici olduęu söylenebilir.

## Kaynaka

- Akdag, O., Ates, A., & Yeroglu, C. (2020). Modification of Harris hawks optimization algorithm with random distribution functions for optimum power flow problem. *Neural Computing and Applications*, 5. <https://doi.org/10.1007/s00521-020-05073-5>
- Ates, A. (2021). Enhanced equilibrium optimization method with fractional order chaotic and application engineering. *Neural Computing and Applications*, 1–28. <https://doi.org/10.1007/s00521-021-05756-7>
- Ates, A., & Akpamukcu, M. (2021). Optimization to optimization (OtoO): optimize monarchy butterfly method with stochastics multi-parameter divergence method for benchmark functions and load frequency control. *Engineering with Computers*. <https://doi.org/10.1007/s00366-021-01364-0>
- Ates, A., Alagoz, B. B., Kavuran, G., & Yeroglu, C. (2017). Implementation of fractional order filters discretized by modified Fractional Order Darwinian Particle Swarm Optimization. *Measurement: Journal of the International Measurement Confederation*. <https://doi.org/10.1016/j.measurement.2017.05.017>
- Ateř, A., & Yeroglu, C. (2016). Optimal fractional order PID design via Tabu Search based algorithm. *ISA Transactions*, 60. <https://doi.org/10.1016/j.isatra.2015.11.015>
- Ateř, Abdullah, & Yeroęlu, C. (2018). Modified Artificial Physics Optimization for Multi-parameter Functions. *Iranian Journal of Science and Technology - Transactions of Electrical Engineering*, 42(4), 465–478. <https://doi.org/10.1007/s40998-018-0082-4>
- Elbes, M., Alzubi, S., Kanan, T., Al-Fuqaha, A., & Hawashin, B. (2019). A survey on particle swarm optimization with emphasis on engineering and network applications. In *Evolutionary Intelligence* (Vol. 12, Issue 2, pp. 113–129). Springer Verlag. <https://doi.org/10.1007/s12065-019-00210-z>
- Gaidhane, P. J., & Nigam, M. J. (2018). A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems. *Journal of Computational Science*. <https://doi.org/10.1016/j.jocs.2018.06.008>

- Glover, F. (1989). Tabu Search—Part I. *ORSA Journal on Computing*.  
<https://doi.org/10.1287/ijoc.1.3.190>
- Glover, F. (1990). Tabu Search—Part II. *ORSA Journal on Computing*.  
<https://doi.org/10.1287/ijoc.2.1.4>
- Guha, D., Roy, P. K., & Banerjee, S. (2020). Grasshopper optimization algorithm scaled fractional order PI-D controller applied to reduced order model of load frequency control system. *International Journal of Modelling and Simulation*.  
<https://doi.org/10.1080/02286203.2019.1596727>
- Khadanga, R. K., Kumar, A., & Panda, S. (2020). A novel modified whale optimization algorithm for load frequency controller design of a two-area power system composing of PV grid and thermal generator. *Neural Computing and Applications*. <https://doi.org/10.1007/s00521-019-04321-7>
- Liang, X., Kou, D., & Wen, L. (2020). An Improved Chicken Swarm Optimization Algorithm and its Application in Robot Path Planning. *IEEE Access*.  
<https://doi.org/10.1109/ACCESS.2020.2974498>
- Probability Density Function*. (2020). Retrieved May 9, 2020, from  
<https://www.mathworks.com/help/stats/prob.normaldistribution.pdf.html>
- van Laarhoven, P. J. M., & Aarts, E. H. L. (1987). Simulated Annealing: Theory and Applications. In *Simulated Annealing: Theory and Applications*. <https://doi.org/10.1007/978-94-015-7744-1>
- Wang, G. G., Deb, S., & Cui, Z. (2019). Monarch butterfly optimization. *Neural Computing and Applications*, 31(7), 1995–2014. <https://doi.org/10.1007/s00521-015-1923-y>
- Yeroğlu, C., & Ateş, A. (2014). A stochastic multi-parameters divergence method for online auto-tuning of fractional order PID controllers. *Journal of the Franklin Institute*, 351(5), 2411–2429.  
<https://doi.org/10.1016/j.jfranklin.2013.12.006>