

ÇADA Algoritmasının ZDT ve DTLZ Test Problemlerinde Uygulanması

*Makale Bilgisi / Article Info

Alındı/Received: 29.12.2023

Kabul/Accepted: 01.07.2024

Yayımlandı/Published: 01.10.2024

Application of ÇADA in ZDT and DTLZ Test Problems

Zeynep HABER* , Harun UĞUZ 

Konya Teknik Üniversitesi, Mühendislik ve Doğa Bilimler Fakültesi, Bilgisayar Mühendisliği Bölümü, Konya, Türkiye

© Afyon Kocatepe Üniversitesi

Öz

Dağınık arama algoritması, tek amaçlı optimizasyon problemlerinin çözümünde sıkça kullanılan bir yöntemdir. Ancak, çok amaçlı problemlerle başa çıkmak oldukça zorlu bir süreçtir. Bu makale, çok amaçlı optimizasyon problemleriyle başa çıkabilmek için "Dağınık Arama Algoritması" (DA) olarak adlandırılan yöntemin genişletilmesine yönelik bir öneri sunmaktadır. Önerilen yaklaşım, DA algoritmasına çok amaçlı optimizasyon algoritması olan Baskın Olmayan Sıralama Genetik Algoritması II (NSGA-II) yöntemindeki Yoğunluk Mesafesi (CD) ve Hızlı Bastırılmamış Sıralama kavramlarını ekleyerek hibrit çok amaçlı optimizasyon algoritması önermektedir. Bu önerilen algoritma, ZDT ve DTLZ test problemleri kullanılarak değerlendirilmiştir. Yapılan deneysel sonuçlar, önerilen Çok Amaçlı Dağınık Arama (ÇADA) algoritmasının 19 farklı çok amaçlı optimizasyon yöntemi ile karşılaştırıldığında, ZDT problemi için 2.40 IGD ortalama ile birinci sırada, DTLZ probleminde ise 0.0035 IGD ortalama değeri ile altıncı sırada yer aldığını göstermektedir. Bu sonuçlar, önerilen algoritmanın karşılaştırılabilir düzeyde başarılı bir performansa sahip olduğunu ortaya koymaktadır.

Anahtar Kelimeler: Çok Amaçlı Optimizasyon Algoritması; Dağınık Arama Algoritması; NSGA-II Algoritması; ZDT-DTLZ Problemleri

Abstract

The Scatter Search algorithm is a frequently used method in solving single-objective optimization problems. However, dealing with multi-objective problems is a highly challenging process. This article proposes an extension of the method referred to as "Scatter Search Algorithm" (SSA) to tackle multi-objective optimization problems. The suggested approach aims to augment the SSA algorithm by incorporating concepts from the Non-Dominated Sorting Genetic Algorithm II (NSGA-II) method, specifically Density Distance (CD), and Fast Non-Dominated Sorting. This proposed algorithm has been evaluated using ZDT and DTLZ test problems. Experimental results show that the proposed Multi-Objective Scatter Search (ÇADA) algorithm ranks first with an average IGD of 2.40 for the ZDT problem and sixth with an average IGD value of 0.0035 for the DTLZ problem when compared to 19 different multi-objective optimization methods. These results demonstrate that the proposed algorithm exhibits a comparable level of successful performance.

Keywords: Multi-Objective Optimization Algorithm; Scattered Search Algorithm; NSGA-II Algorithm; ZDT-DTLZ Problems

1. Giriş

Optimizasyon, belirli bir problem için en uygun çözümün bulunması sürecidir (Emel ve Taşkın, 2002). Bu problemler farklı türlerde olabilir: sürekli, ayrık, tek amaçlı veya çok amaçlı (Hassan, 2021). Problemin yapısına göre metotlar ve modeller kullanılarak en uygun çözüm bulunur. Gerçek dünyada karşılaşılan optimizasyon problemleri, amaç fonksiyonlarının sayısına bağlı olarak tek veya çok amaçlı olarak sınıflandırılabilir (Kamjoo ve ark., 2016). Tek amaçlı optimizasyon problemleri genellikle tek bir hedefi optimize etmeyi amaçlar, bu yüzden genellikle eniyileme veya karar ağacı tabanlı algoritmalar kullanılırken, çok amaçlı optimizasyon problemleri için evrimsel algoritmalar, çok amaçlı parçacık sürü optimizasyonu veya çözüm bulma algoritmaları gibi özel algoritmalar geliştirilmiştir. Bu algoritmalar, mühendislik (Azizi ve ark., 2023; Chen ve ark., 2023), ekonomi (Bilici ve Özdemir, 2023; Cheraghi ve Jahangir, 2023), tıp (Dörterler ve ark., 2022; Patil ve ark., 2023; Sekkal ve ark., 2024), lojistik

(Özdemir ve Dörterler, 2022; Heidari ve ark., 2023; Peng ve ark., 2023) ve diğer birçok alanda çeşitli problemlerin çözümünde kullanılır. Çok amaçlı problemlerin tek amaçlı optimizasyon yöntemleriyle çözülmesi genellikle zor olabilir, bu nedenle bu tür problemler için özel olarak geliştirilmiş çok amaçlı optimizasyon algoritmaları bulunmaktadır. Çok amaçlı optimizasyon, birden fazla hedefin aynı anda optimize edilmesini ifade eder. Bu tür optimizasyonun genel formülü aşağıdaki eşitliklerde yer almaktadır:

$$\min/\max f_m(x), \quad m = 1, 2, \dots, M \quad (1)$$

$$\text{Kısıtlar: } g_j(x) > 0, j = 1, 2, \dots, J \quad (2)$$

$$h_k(x) = 0, k = 1, 2, \dots, K \quad (3)$$

$$x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, 2, \dots, n \quad (4)$$

Burada $m = 1, 2, \dots, M$ problemdeki amaç kümesidir. $f_m(x)$ amaç fonksiyonunu belirtir. $g_j(x) > 0, j =$

$1, 2, \dots, J$ eşitsizlik kısıtlarını ifade ederken, $h_k(x) = 0, k = 1, 2, \dots, K$ eşitlik kısıtlarını ifade etmektedir.

Son yıllarda, çok amaçlı optimizasyon problemlerini (ÇAOP) çözmek için çeşitli evrimsel algoritmalar geliştirilmiştir (Ma ve ark., 2021). İlk çok amaçlı optimizasyon algoritması Vektör Değerlendirilen Genetik Algoritmasıdır (VEGA)(Schaffer, 1985). Daha sonra, Niched Pareto Genetic Algorithm (NPGA) (rey Horn ve ark., 1994), Multi Objective Genetic Algorithm (MOGA) (Murata ve Ishibuchi, 1995), Nondominated Sorting Genetic Algorithm II (NSGA-II) (Deb ve ark., 2000), Pareto Archived Evolution Strategy (PAES)(Knowles ve Corne, 2000), Strength Pareto Evolutionary Algorithm II (SPEA-II) (Zitzler ve ark., 2001), Pareto Envelope-Based Selection Algorithm II (PESA-II) (Corne ve ark., 2001), Multiple-Single-Objective Pareto Sampling (MSOPS)(Hughes, 2003) ve Orthogonal Multi-Objective Evolutionary Algorithm II (OMOEA-II)(Zeng ve ark., 2005) gibi birçok evrimsel algoritma önerilmiştir. Ancak, bu algoritmaların bazı sınırlamaları vardır. Örneğin, NSGA-II algoritması, özellikle yüksek boyutlu problemler ve çok sayıda amaç fonksiyonu içeren problemler için hesaplama verimliliği açısından sınırlamalara sahiptir(Tian ve ark., 2016). Ayrıca, mevcut algoritmaların çoğu, yerel optima takılma sorunuyla karşı karşıya kalabilir ve bu da çözüm kalitesini olumsuz etkileyebilir(Li ve Li, 2021).

Bu çalışmada, NSGA-II algoritmasının Yoğunluk Mesafesi ve Hızlı Baskınlık Sıralama yaklaşımlarından yararlanarak

tek amaçlı Dağılık Arama algoritmasını çok amaçlı bir algoritmaya dönüştürmek amaçlanmıştır. Bu yaklaşımın, mevcut literatürdeki yöntemlere kıyasla daha iyi performans göstermesi ve yerel optima takılma sorununu azaltması hedeflenmektedir. Önerilen algoritma, ZDT ve DTLZ test problemleri kullanılarak değerlendirilmiştir. Deneysel sonuçlar, 19 farklı çok amaçlı optimizasyon yöntemleri ile karşılaştırılmıştır. Makale şu şekilde düzenlenmiştir. Bölüm 2.1, ZDT ve DTLZ problemleri tanıtılmıştır. Bölüm 2.2'de NSGA-II, Dağılık arama algoritması ve önerilen yöntem açıklanmıştır. Bölüm 3'te algoritmaların ZDT ve DTLZ problemleri üzerindeki performansları karşılaştırılmıştır. Bölüm 4'te sonuçlar sunulmuştur.

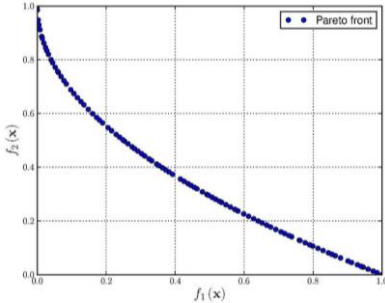
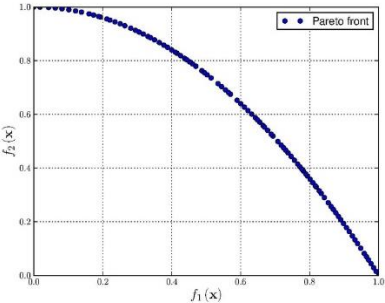
2. Materyal ve Metot

2.1 Materyal

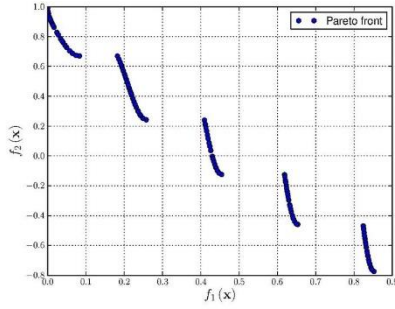
2.1.1. ZDT Test Problemleri

Zitzler, Deb ve Thiele (ZDT) (Zitzler ve ark., 2000) , farklı çok amaçlı evrimsel yöntemlerin performansını karşılaştırmak için ZDT1-ZDT6 fonksiyonlarını önermişlerdir. Bu fonksiyonlar iki hedefle ilgili sorunları ele almaktadır. ZDT problemleri, $[0, 1]$ aralığında değişkenlere sahiptir. Fonksiyonlar, dışbükey (ZDT1 ve ZDT4), içbükey (ZDT2 ve ZDT6) ve bağlantı kesme (ZDT3) olan Pareto cephelerini sunar. ZDT problemlerinin pareto cepheleri, formülleri ve arama uzay bilgileri Çizelge 1'de verilmiştir.

Çizelge 1. ZDT test problemleri

Problem	Pareto Cephe	Formül	Arama Uzayı
ZDT1		$f_1(x) = x_1$ $f_2(x) = g(x)h(f_1(x), g(x))$ $g(x) = 1 + \frac{9}{29} \sum_{i=2}^{30} x_i$ $h(f_1(x), g(x)) = 1 - \sqrt{\frac{f_1(x)}{g(x)}}$	$0 \leq x_i \leq 1$ $1 \leq i \leq 30$
ZDT2		$f_1(x) = x_1$ $f_2(x) = g(x)h(f_1(x), g(x))$ $g(x) = 1 + \frac{9}{29} \sum_{i=2}^{30} x_i$ $h(f_1(x), g(x)) = 1 - \left(\frac{f_1(x)}{g(x)}\right)^2$	$0 \leq x_i \leq 1$ $1 \leq i \leq 30$

ZDT3



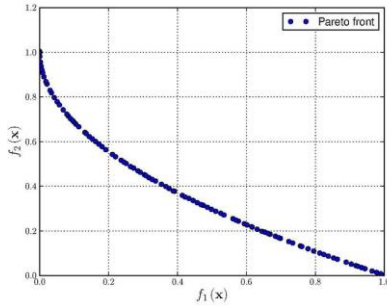
$$f_1(x) = x_1$$

$$f_2(x) = g(x)h(f_1(x), g(x))$$

$$g(x) = 1 + \frac{9}{29} \sum_{i=2}^{30} x_i \quad 0 \leq x_i \leq 1$$

$$h(f_1(x), g(x)) = 1 - \sqrt{\frac{f_1(x)}{g(x)}} - \left(\frac{f_1(x)}{g(x)}\right) \sin(10\pi f_1(x)) \quad 1 \leq i \leq 30$$

ZDT4



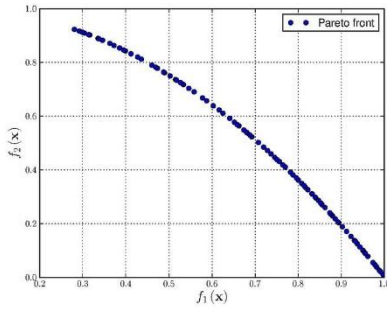
$$f_1(x) = x_1$$

$$f_2(x) = g(x)h(f_1(x), g(x)) \quad 0 \leq x_1 \leq 1$$

$$g(x) = 91 + \sum_{i=2}^{10} (x_i^2 - 10 \cos(4\pi x_i)) \quad -5 \leq x_i \leq 5$$

$$h(f_1(x), g(x)) = 1 - \sqrt{\frac{f_1(x)}{g(x)}} \quad 2 \leq i \leq 10$$

ZDT6



$$f_1(x) = 1 - \exp(-4x_1) \sin^6(6\pi x_1)$$

$$f_2(x) = g(x)h(f_1(x), g(x)) \quad 0 \leq x_i \leq 1$$

$$g(x) = 1 + 9 \left[\frac{\sum_{i=2}^{10} x_i}{9} \right]^{0.25} \quad 1 \leq i \leq 10$$

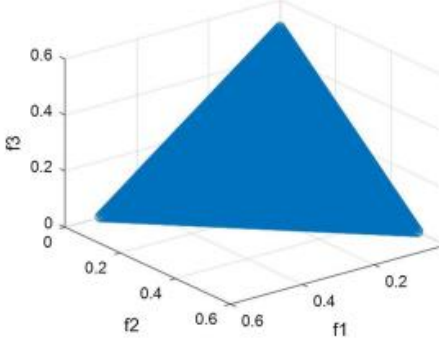
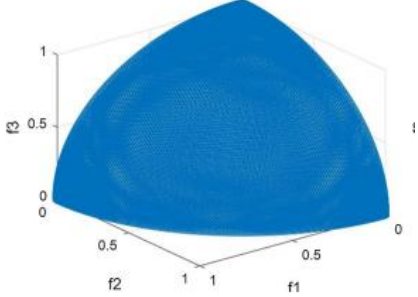
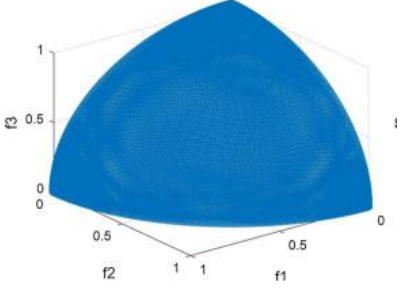
$$h(f_1(x), g(x)) = 1 - \left(\frac{f_1(x)}{g(x)} \right)^2$$

2.1.2. DTLZ Test Problemleri

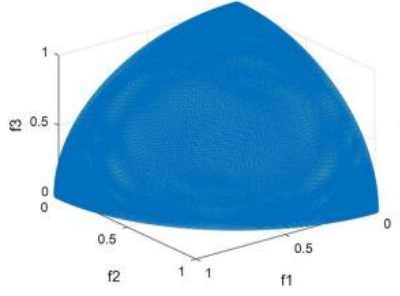
DTLZ, 2005 yılında Deb, Thiele, Laumanns ve Zitzler tarafından önerilen test problemleridir (Deb ve ark., 2005). DTLZ (DTLZ1-DTLZ7) takımında yedi test fonksiyonu bulunmaktadır. DTLZ problem setleri, birkaç özelliğten oluşur ve birkaç ÇAOA yeteneğini değerlendirir. DTLZ1, DTLZ3 ve DTLZ6 tipik olarak çok modludur ve ÇAOA'ların

birden çok yerel Pareto-optimal cephe aracılığıyla yakınsamasını değerlendirir. ÇAOA'lar, yerel Pareto cephelerinin varlığı nedeniyle Pareto optimal cephesine yakınsama sorunları yaşarlar. ÇAOA'ların çeşitli biçimlerdeki sorunları çözme yeteneğini inceleyen DTLZ2 ve DTLZ4-DTLZ7 dahil olmak üzere başka test sorunları da vardır. DTLZ problemlerinin Pareto cepheleri, formülleri ve arama uzay bilgileri Çizelge 2'de verilmiştir.

Çizelge 2. Dtlz test problemleri (Deb ve ark., 2005; Zou ve ark., 2011; Huang ve Chen, 2020)

Problem	Pareto Cephe	Formül	Arama Uzayı
DTLZ1		$f_1(x) = \frac{1}{2} x_1 x_2 \dots x_{M-1} (1 + g(x_M))$ $f_2(x) = \frac{1}{2} x_1 x_2 \dots (1 - x_{M-1}) (1 + g(x_M))$ \vdots $f_M(x) = \frac{1}{2} (1 - x_1) (1 + g(x_M))$	$0 \leq x_i \leq 1$ $1 \leq i \leq n$
DTLZ2		$f_1(x) = (1 + g(x_M)) \cos(x_1 \pi / 2) \cos(x_2 \pi / 2) \dots \cos(x_{M-1} \pi / 2)$ $f_2(x) = (1 + g(x_M)) \cos(x_1 \pi / 2) \cos(x_2 \pi / 2) \dots \sin(x_{M-1} \pi / 2)$ \vdots $f_M(x) = (1 + g(x_M)) \sin(x_1 \pi / 2)$ $g(x_M) = \sum_{x_1 \in x_M} (x_1 - 0.5)^2$	$0 \leq x_i \leq 1$ $1 \leq i \leq n$
DTLZ3		$f_1(x) = (1 + g(x_M)) \cos(x_1 \pi / 2) \cos(x_2 \pi / 2) \dots \cos(x_{M-1} \pi / 2)$ $f_2(x) = (1 + g(x_M)) \cos(x_1 \pi / 2) \cos(x_2 \pi / 2) \dots \sin(x_{M-1} \pi / 2)$ \vdots $f_M(x) = (1 + g(x_M)) \sin(x_1 \pi / 2)$ $g(x_M) = 100 \left[x_M + \sum_{x_1 \in x_M} (x_1 - 0.5)^2 - \cos(20\pi(x_1 - 0.5)) \right]$	$0 \leq x_i \leq 1$ $1 \leq i \leq n$

DTLZ4



$$f_1(x) = (1 + g(x_M)) \cos(x_1^a \pi \sqrt{2}) \cos(x_2^a \pi \sqrt{2}) \dots \cos(x_{M-1}^a \pi \sqrt{2})$$

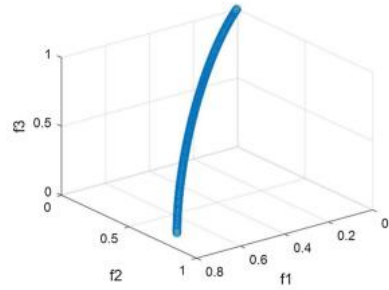
$$f_2(x) = (1 + g(x_M)) \cos(x_1^a \pi \sqrt{2}) \cos(x_2^a \pi \sqrt{2}) \dots \sin(x_{M-1}^a \pi \sqrt{2}) \quad 0 \leq x_i \leq 1$$

$$\vdots \quad 1 \leq i \leq n$$

$$f_M(x) = (1 + g(x_M)) \sin(x_1^a \pi \sqrt{2})$$

$$g(x_M) = \sum_{x_i \in x_M} (x_i - 0.5)^2$$

DTLZ5



$$f_1(x) = (1 + g(x_M)) \cos(\theta_1 \pi \sqrt{2}) \cos(\theta_2 \pi \sqrt{2}) \dots \cos(\theta_{M-1} \pi \sqrt{2})$$

$$f_2(x) = (1 + g(x_M)) \cos(\theta_1 \pi \sqrt{2}) \cos(\theta_2 \pi \sqrt{2}) \dots \sin(\theta_{M-1} \pi \sqrt{2}) \quad 0 \leq x_i \leq 1$$

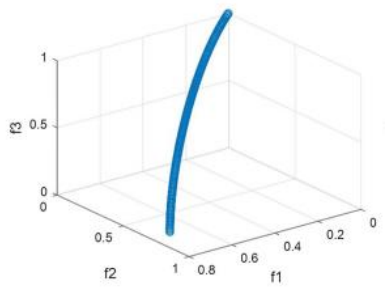
$$\vdots \quad 1 \leq i \leq n$$

$$f_M(x) = (1 + g(x_M)) \sin(\theta_1 \pi \sqrt{2}) \quad 2 \leq l \leq (M - 1)$$

$$g(x_M) = \sum_{x_i \in x_M} (x_i - 0.5)^2$$

$$\theta_i = \frac{x}{4(1 + g(r))} (1 + 2g(r)x_i)$$

DTLZ6



$$f_1(x) = (1 + g(x_M)) \cos(\theta_1 \pi \sqrt{2}) \cos(\theta_2 \pi \sqrt{2}) \dots \cos(\theta_{M-1} \pi \sqrt{2})$$

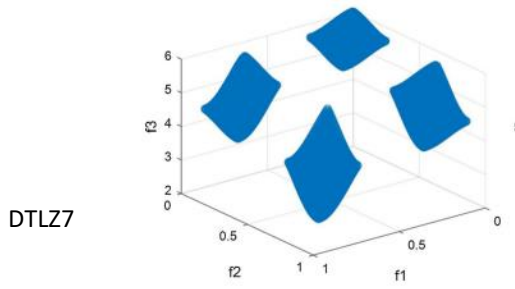
$$f_2(x) = (1 + g(x_M)) \cos(\theta_1 \pi \sqrt{2}) \cos(\theta_2 \pi \sqrt{2}) \dots \sin(\theta_{M-1} \pi \sqrt{2}) \quad 0 \leq x_i \leq 1$$

$$\vdots \quad 1 \leq i \leq n$$

$$f_M(x) = (1 + g(x_M)) \sin(\theta_1 \pi \sqrt{2}) \quad 2 \leq l \leq (M - 1)$$

$$\theta_i = \frac{x}{4(1 + g(r))} (1 + 2g(r)x_i)$$

$$g(x_M) = \sum_{x_i \in x_M} x_i^{0.1}$$



DTLZ7

$$f_1(x) = x_1$$

$$f_3(x) = x_2$$

⋮

$$0 \leq x_i \leq 1$$

$$1 \leq i \leq n$$

$$f_M(x) = (1 + g(x_M))h(f_1, f_2, \dots, f_{M-1}, g),$$

$$g(x_M) = 1 + \frac{9}{|x_M|} \sum_{x_i \in x_M} x_i$$

2.2 Metot

2.2.1. Baskın Olmayan Sıralama Genetik Algoritması II (NSGA-II)

Genetik algoritma temelli çeşitli çok amaçlı optimizasyon algoritmaları geliştirilmiştir. Bu algoritmalar arasında literatürde en çok tercih edilenlerden biri NSGA-II olarak adlandırılan bastırılmamış sıralama genetik algoritmasıdır. NSGA-II algoritması 2002 yılında Srinivas ve Deb tarafından geliştirilmiştir (Deb ve ark., 2002).

NSGA-II algoritması temelde iki bölümden oluşur.

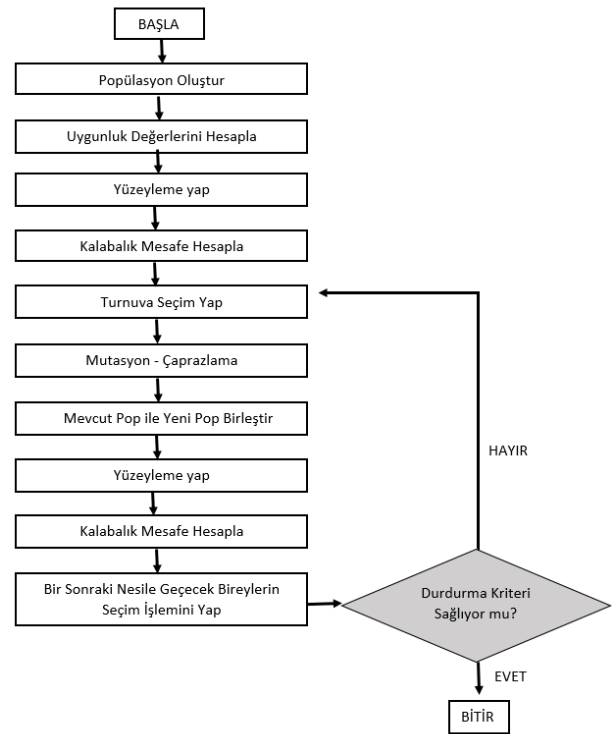
1. *GA Operatörleri*: seçim, çaprazlama, mutasyon
2. *Pareto Fonksiyonları*: Baskın olmayan sıralama fonksiyonu, Kalabalık mesafe fonksiyonu

Rastgele N boyutlu kısıtlara uygun şekilde başlangıç popülasyonu oluşturulur. Ardından çözümlerin tüm amaçları hesaplanır. Elde edilen uygunluk değerleri kullanılarak baskınlık değerleri hesaplanır. Baskınlık değerlerine göre yüzeyler oluşturulup sıralanır ve CD değerleri hesaplanır. Daha sonra çaprazlama ve mutasyon işlemleri için kalabalık mesafe tahmin prosedürü yardımıyla turnuva seçimi kullanılarak parent havuzu oluşturulur. Daha sonra çeşitlilik operatörleri uygulanır. NSGA-II algoritmasının akış diyagramı Şekil 1'de şematik olarak gösterilmiştir.

2.2.2. Dağınık Arama Algoritması

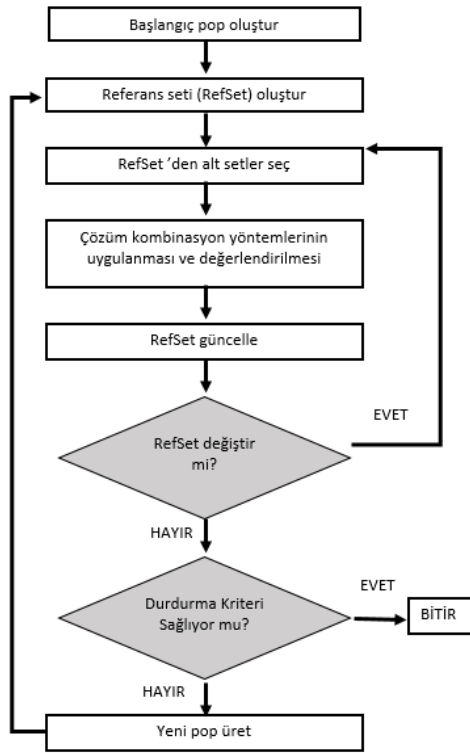
Dağılım arama yöntemi ilk olarak Glover tarafından 1977'de geliştirilmiştir (Glover, 1977). Bu yöntem, yeni çözümler elde etmek için referans seti (RefSet) adı verilen bir dizi çözümün birbirleriyle kombinlenmesini içerir. RefSet, popülasyondaki en iyi bireyleri içerir ve bu bireyler uygunluk ve çeşitlilik bazında seçilir. Örneğin, bir çözümün uygunluk değeri, referans kümesine girmeye aday diğer çözümlerden daha düşük olabilir. Ancak, bu birey çeşitliliği artırıyor, referans kümesine dahil edilir.

Oluşturulan RefSet'ten, birleştirme işlemleri gerçekleştirmek için alt kümeler oluşturulur. Bu alt kümelerdeki bireyler birleştirilerek yeni bireyler elde edilir.



Şekil 1. NSGA-II akış diyagramı

Yeni bireyler ve mevcut RefSet çözümleri bir havuzda toplanır. Daha sonra, bu havuzdan farklı yöntemler kullanılarak bireyler seçilir ve RefSet güncellenir. Eğer RefSet güncellemesi sonrasında herhangi bir değişiklik olmazsa, yeni bir popülasyon oluşturulur ve süreç durdurma kriteri sağlanıncaya kadar devam eder. Dağınık arama yönteminin akış şeması Şekil 2'de şematik olarak gösterilmiştir.



Şekil 2. Dağınık Arama Algoritmasının Akış Diyagramı

2.2.3. Önerilen Çok Amaçlı Dağınık Arama Algoritması

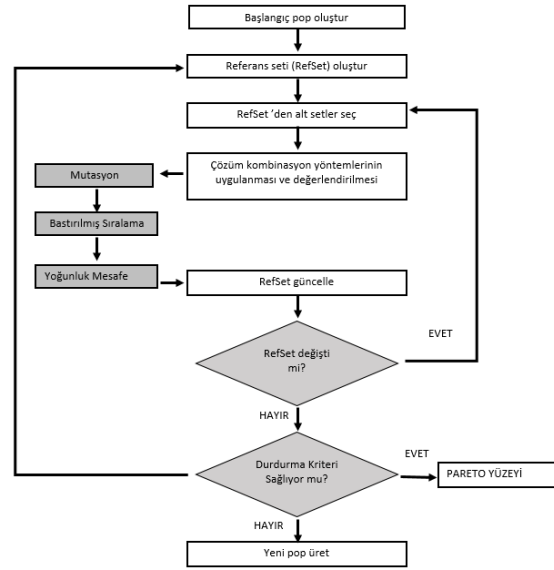
NSGA-II algoritması, hedefler hakkında ön bilgi gerektirmeksizin baskın çözümler kümesini elde edebilmesiyle literatürde en çok tercih edilen yöntemlerden biridir. NSGA-II, hızlı baskın sıralama yaklaşımı ve kalabalık mesafe hesaplamasıyla her bir çözüm için tekrarlamalara gerek kalmadan hesaplama karmaşıklığını en aza indirir. Ancak, NSGA-II'nin özellikle yüksek boyutlu problemler ve çok sayıda amaç fonksiyonu içeren problemler için bazı sınırlamaları bulunmaktadır. Bu sınırlamalar, yerel optimale takılma ve hesaplama verimliliği açısından performans düşüklüğüne yol açabilir.

Bu çalışmada, NSGA-II'nin bu avantajlarından yararlanarak, tek amaçlı Dağınık Arama algoritmasını çok amaçlı hale getirilmiştir. Önerilen algoritmada, NSGA-II'nin Kalabalık Mesafe (CD) ve Baskınlık Sıralaması (Non-Dominated Sort) yöntemlerini kullanılarak RefSet güncellemeleri gerçekleştirilmiştir. Ayrıca, Elitizm yöntemi ile en iyi bireyler hafızada tutulmuştur. Bu sayede, algoritma hem yerel optimale takılma sorununu azaltmakta hem de hesaplama verimliliğini artırmaktadır. Çok amaçlı Dağınık Arama algoritmasının akış diyagramı Şekil 3'te gösterilmiştir.

2.2.3.1. Çaprazlama

Bu çalışmada yeniden üretme işlemi için operatörler aşağıda ayrıntılı olarak açıklanmıştır. Genetik algoritmadaki çaprazlama operatörlerinden uniform ve iki nokta çaprazlama yöntemleri Şekil 4'te, ayırma ve ters

çevirme çaprazlama yöntemleri ise Şekil 5'te gösterilmiştir. Uniform çaprazlama tekniği, rastgele seçilen bir noktada genlerin yer değiştirilmesi işlemidir. İki nokta çaprazlama tekniği ise rastgele belirlenen iki nokta arasındaki genlerin yer değiştirilmesidir.



Şekil 3. Çok Amaçlı Dağınık Arama algoritmasının akış diyagramı

Ebeveyn 1	1	0	1	0	1	0	1	0	0	0
Çocuk	0	0	1	0	1	0	1	1	0	0
Ebeveyn 2	0	1	0	0	1	0	1	1	0	0

(a)

Ebeveyn 1	0	0	0	0	1	0	1	0	1	1
Çocuk	0	0	1	1	1	0	1	0	1	1
Ebeveyn 2	1	0	1	1	1	0	0	0	0	1

(b)

Şekil 4. (a) Uniform ve (b) iki nokta çaprazlama (Haklı ve Ortacay, 2019)

Ebeveyn 1	1	0	1	1	0	1	0	1	1	0
Çocuk	1	0	1	1	0	1	0	1	1	0
Ebeveyn 2	0	1	1	0	1	0	1	1	0	1

(a)

Ebeveyn 1	1	0	1	0	0	1	1	1	0	0
Çocuk	1	1	0	0	0	1	1	1	0	0
Ebeveyn 2	1	0	0	1	1	0	1	0	1	0

(b)

Şekil 5. (a) Ayırma ve (b) Ters çevirme çaprazlama (Haklı ve Ortacay, 2019)

Ayırma çaprazlama yöntemi, belirtilen iki bireyden aynı boyutta iki farklı noktayı seçip bu noktaların yer değiştirilmesi işlemidir. Ters çevirme çaprazlama ise

seçilen iki nokta arasındaki genlerin ters çevrilerek diğer bireye eklenmesidir.

2.2.3.2. Mutasyon

Çok Amaçlı Problemleri (MOP'ler) çözen Çok Amaçlı Evrimsel Algoritmalar (MOEA'lar) için Deb ve Goyal (1996), Polinom Mutasyonu (PLM) adı verilen bir varyasyon mekanizması önermiştir (Deb ve Goyal, 1996). Bu operatör, daha sonra Deb ve Tiwari (2008) tarafından geliştirilmiştir (Deb ve Tiwari, 2008).

Rastgele değer $r < 0,5$ ise, sol tarafa (XLow ve Xi arasındaki bölge) örneklenir; aksi takdirde $r > 0,5$ ise sağ tarafa (Xi ile XUpper arasındaki bölge) örneklenir. PLM karar değişkeninin arama uzayında büyük sıçramalara izin verdiği için, optimizasyon sürecinin yerel optimumdan kaçma şansı daha yüksektir ve sınırdayken bir çözümü değiştirebilir. Polinom mutasyon ile ilgili kaba kod, Şekil 6'da verilmiştir.

```

1:  $i \leftarrow 0$ 
2: repeat
3:    $r \leftarrow U[0, 1]$ 
4:   if ( $r \leq P_m$ ) then
5:      $\delta_1 \leftarrow \frac{x_i - LB_i}{UB_i - LB_i}$ 
6:      $\delta_2 \leftarrow \frac{UB_i - x_i}{UB_i - LB_i}$ 
7:      $r \leftarrow U[0, 1]$ 
8:     if ( $r \leq 0.5$ ) then
9:        $\delta_k \leftarrow [2r + (1 - 2r) \times (1 - \delta_1)^{\eta_{m+1}}]^{\frac{1}{\eta_{m+1}} - 1}$ 
10:    else
11:       $1 - [2(1 - r) + 2(r - 0.5) \times (1 - \delta_2)^{\eta_{m+1}}]^{\frac{1}{\eta_{m+1}}}$ 
12:    end if
13:     $x_i \leftarrow x_i + \delta_k \cdot (UB_i - LB_i)$ 
14:  end if
15: until  $i++ == m$ 

```

Şekil 6. Polinom mutasyon (Abed-Alguni ve Paul, 2020)

3. Bulgular

Önerilen yöntemin başarısını değerlendirmek ve diğer yöntemlerle karşılaştırmak amacıyla elde edilen sonuçlar bu bölümde sunulmaktadır. Yöntemler Intel® Core™ i7, 2.40 GHz CPU, 16GB RAM, 64-bit işletim sistemine sahip bilgisayarda ve Matlab 2021 uygulama geliştirme aracında çalıştırılmıştır.

Yöntemin performans değerlendirilmesi için çok amaçlı optimizasyon benchmark problemleri olan ZDT ve DTLZ test fonksiyon setleri kullanılmıştır. ZDT problemi, belirli özelliklere sahip altı farklı problemden oluşmaktadır. Her fonksiyon, dışbükeylik, içbükeylik, çok modluluk veya süreksizlik gibi belirli bir zorluk derecesine sahiptir. Bu tür özellikler, algoritmaların farklı koşullar altında test edilmesine olanak tanır. DTLZ kıyaslama seti ise esnek ve uygulanması kolay yedi problemden oluşmaktadır. Tüm bu kıyaslama işlevleri, çok amaçlı yaklaşımların

performansını doğrulamak için ilgili literatürde yaygın olarak kullanılmaktadır. Önerilen yaklaşımın kapsamlı bir şekilde test edilmesi için ZDT ve DTLZ problem setleri seçilmiştir. Problemlerin tanımları Çizelge 3'de verilmiştir. Önerilen yöntemin parametreleri Çizelge 4'de gösterilmektedir.

Çizelge 3. Deneysel Test Problemlerinin Tanımlamaları

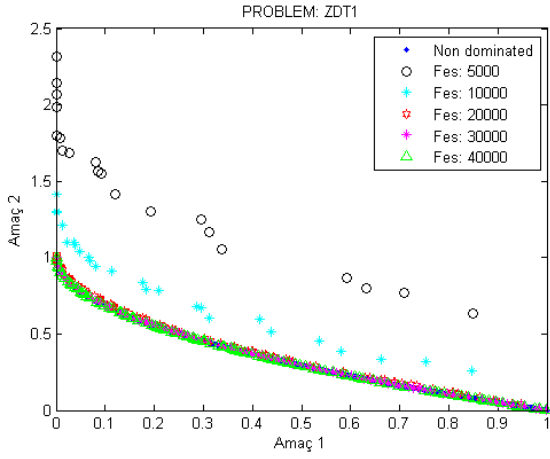
Problem	Amaç Sayısı	Boyut (D)	Popülasyon	Maksimum
			Boyutu (N)	Fes Sayısı (MFC)
ZDT1- ZDT3	2	30	100	4×10^5
ZDT4, ZDT6	2	10	100	4×10^5
DTLZ1	3	7	500	10^5
DTLZ2- DTLZ6	3	12	500	10^5
DTLZ7	3	22	500	10^5

Çizelge 4. Parametreler

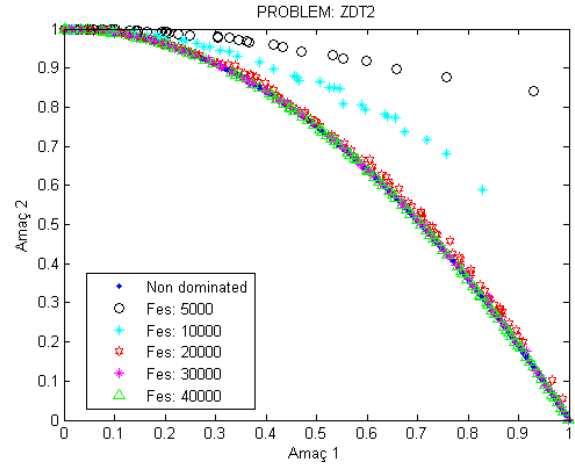
Parametre	Değeri
Çalışma Sayısı (Run)	30
Mutasyon Oranı	0.2
RefSet Boyutu	20
Dağıtım endeksi	20

Parametrelerin değerleri, belirli aralıklarda yapılan denemeler sonucunda belirlenmiştir. Mutasyon oranı 0.07, 0.2, 0.5 ve 0.7 değerleri ile 10 kez çalıştırılmıştır. En uygun değer 0.2 olduğuna karar verilmiştir. RFC değeri ise 10 ve 20 olarak 10 kez çalıştırılmıştır.

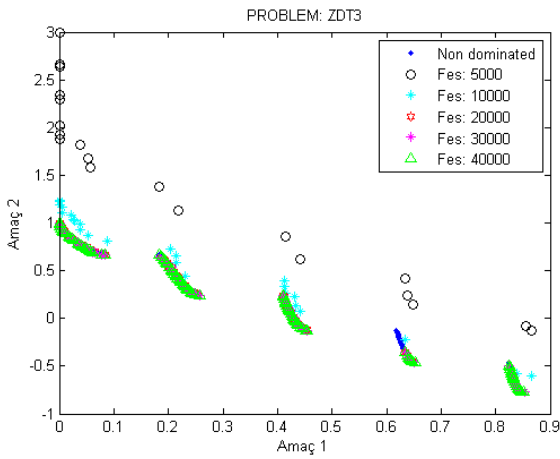
ÇADA algoritmasının performansını değerlendirmek için IGD (Inverted Generational Distance) metriği kullanılmıştır. IGD, algoritmaların ürettiği çözümlerin gerçek Pareto optimal sınır ile ne kadar yakın olduğunu ölçer. Pareto optimal sınır, hiçbir amaç fonksiyonunu diğerini kötüleştirmeden optimize edilemeyen çözümlerin kümesidir. Bu metriğin hesaplanması için, algoritma tarafından üretilen çözümlerin gerçek Pareto optimal sınırına olan mesafeleri ölçülür ve bu mesafelerin ortalaması alınarak bir skor elde edilir. Daha düşük bir IGD değeri, algoritmanın daha iyi performans gösterdiğini gösterir. ÇADA algoritmasının ZDT1, ZDT2, ZDT3 ve ZDT6 problemleri üzerinde yakınsama grafikleri Şekil 7'de gösterilmiştir.



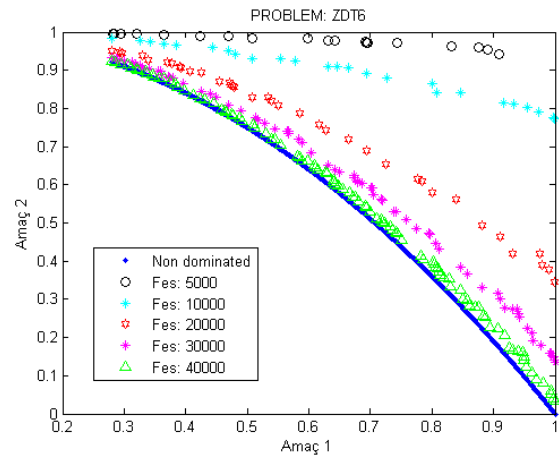
(a)



(b)



(c)



(d)

Şekil 7(a) ZDT1, (b) ZDT2,(c) ZDT3 ve (d) ZDT6 yakınsama grafikleri

Şekil 7 incelendiğinde, ZDT1, ZDT2 ve ZDT3 problemlerinde 20000 Fes'de Pareto yüzeyine ulaşıldığı görülmektedir. Ancak, ZDT6 probleminde 40000 Fes tamamlandığında dahi tam olarak Pareto yüzeyine erişilememiştir. ÇADA yönteminin her bir problem üzerinde 30 kez çalıştırma sonucunda IGD metriğinin en iyi, en kötü, ortalama ve standart sapma değerleri Çizelge 5'te verilmiştir.

Algoritmanın ZDT ve DTLZ problemleri üzerindeki başarısını ölçmek için önerilen ÇADA algoritması, RVEAiNGG (Liu ve ark., 2022), MWDEO(Ewees ve ark., 2021), MOEA/D(Zhang ve Li, 2007) , RVEA(Cheng ve ark., 2016), MOPSO(Coello ve Lechuga, 2002), SMPSO(Nebro ve ark., 2009), NSGA-II (Deb ve ark., 2002), MOALO (Mirjalili ve ark., 2017) , SHAMODEWO (Panagant ve ark., 2019) , MOMVO(Mirjalili ve ark., 2017) , MOGOA(Mirjalili ve ark., 2018) , MOGWO(Mirjalili ve ark., 2016) , KnEA(Zhang ve ark., 2014) , PICEAg (Wang ve ark., 2012) , MOEADSTM (Li ve ark., 2013) , NSGA-III (Deb ve Jain, 2013) ve GREa (Yang ve ark., 2013) isimli çok sayıda çok amaçlı optimizasyon algoritması ile karşılaştırılmıştır.

Çizelge 5. Önerilen yöntemin IGD değerleri

	EN İYİ	EN KÖTÜ	ORT.	STD
ZDT1	0.0001	0.0048	0.0046	0.0001
ZDT2	0.0000	0.0050	0.0041	0.0013
ZDT3	0.0000	0.0345	0.0069	0.0066
ZDT4	0.0000	0.1380	0.0155	0.0343
ZDT6	0.0000	0.0041	0.0033	0.0011
	EN İYİ	EN KÖTÜ	ORT.	STD
DTLZ1	0.0145	0.2964	0.0274	0.0501
DTLZ2	0.0000	0.0314	0.0268	0.0088
DTLZ3	0.0000	1.0056	0.0786	0.1734
DTLZ4	0.0000	0.0317	0.0266	0.0087
DTLZ5	0.0000	0.0017	0.0012	0.0004
DTLZ6	0.0000	0.0085	0.0035	0.0019
DTLZ7	0.0000	0.3413	0.0391	0.0541

Bu algoritmalar, iyi performans ve rekabetçi sonuçlar gösterdikleri için seçilmiştir. Ayrıca, bu tür yöntemler ilgili literatürde karşılaştırmalar için yaygın olarak kullanılmaktadır. Bu algoritmaların sonuçları Ewes ve arkadaşları tarafından yapılan çalışmadan alınmıştır

(Ewees ve ark., 2021). Yöntemlerin 30 kez çalıştırma sonunda elde edilen IGD ölçümlerinin ortalama sonuçları ve karşılık gelen standart sapma (Std) değerleri Çizelge 6'da gösterilmektedir.

Çizelge 6. Önerilen algoritmanın IGD sonuçları ve ZDT problemleri üzerinde karşılaştırılan algoritmalar.

		MWDEO	MWDE	MOEA/D	RVEA	MOPSO	SMPSO	NSGAI	MOALO	SHAMODEWO	ÇADA
	Ortalama	0.0046	38.424	0.0112	0.0148	33.905	0.0049	0.0149	0.826	0.5606	0.0046
ZDT1	Std	0.0003	0.025	0.01	0.0029	11.1	0.0002	0.0065	0.0003	0.4845	0.0001
	Sıra	2	18	5	6	17	4	7	14	12	2
	Ortalama	0.0048	24.672	0.0167	0.0172	46.869	0.0049	0.012	0.6095	0.6095	0.0041
ZDT2	Std	0.0002	0.025	0.0358	0.0044	8.48	0.0003	0.0015	0.0434	0.0503	0.0013
	Sıra	2	15	7	8	16	3	6	11	11	1
	Ortalama	0.0402	35.528	0.0168	0.0355	38.288	0.0053	0.0941	0.7995	0.5472	0.0069
ZDT3	Std	0.0396	0.269	0.0085	0.0141	10.6	0.0002	0.0561	0.0029	0.473	0.0066
	Sıra	8	19	4	6	20	1	10	16	15	2
	Ortalama	0.0049	69.899	0.0113	0.0153	13.533	1.4939	0.0111	0.2735	1.14	0.0155
ZDT4	Std	0.0003	0.0025	0.0054	0.0061	4.24	1.11	0.0016	0.0764	0.6498	0.0343
	Sıra	1	20	4	5	18	15	3	11	14	6
	Ortalama	0.0037	5.1587	0.0054	0.0162	1.4103	0.0039	0.0074	0.1651	0.0294	0.0033
ZDT6	Std	0.0002	0.025	0.0013	0.0044	3.43	0.0002	0.0011	0.1117	0.0065	0.0011
	Sıra	2	15	5	9	13	4	6	12	10	1
	Ortalama	3.00	17.40	5.00	6.80	16.80	5.40	6.40	12.80	12.40	2.40
	Sıra										
		SHAMODE	MOMVO	MOGOA	MOGWO	KnEA	PICEAg	GrEA	NSGAIII	MOEADSTM	RVEAiNG
	Ortalama	0.0036	0.9839	0.9013	0.2818	0.3481	0.1068	0.1702	0.0046	0.0048	0.7883
ZDT1	Std	0.0009	0.016	0.0642	0.4837	0.0781	0.0754	0.0681	0.0006	0.0005	0.0651
	Sıra	1	16	15	10	11	8	9	2	3	13
	Ortalama	0.6095	0.6861	0.7817	0.6095	0.1455	0.281	0.0078	0.0049	0.0057	1.4953
ZDT2	Std	0.0773	0.0091	0.0985	0.0947	0.0136	0.0547	0.0003	0.0017	0.001	0.3503
	Sıra	11	12	13	11	9	10	5	3	4	14
	Ortalama	0.3488	0.912	1.2207	0.275	0.0318	0.1558	0.0395	0.0144	0.0469	0.5362
ZDT3	Std	0.299	0.0852	0.3893	0.4723	0.0131	0.0907	0.0333	0.0131	0.0864	0.114
	Sıra	13	17	18	12	5	11	7	3	9	14
	Ortalama	0.2177	2.2231	2.9814	0.2815	0.3532	0.2377	0.202	0.005	0.0332	13.916
ZDT4	Std	0.2742	0.978	0.9698	0.4839	0.0512	0.0518	0.1033	0.0013	0.0457	6.795
	Sıra	9	16	17	12	13	10	8	2	7	19
	Ortalama	0.0038	0.0147	0.0308	0.0037	0.0097	0.0039	0.0074	0.0039	0.0038	4.1236
ZDT6	Std	0.0005	0.0252	0.0501	0.0012	0.0043	0.0013	0.0035	0.0011	0.0012	0.8426
	Sıra	3	8	11	2	7	4	6	4	3	14
	Ortalama	7.40	13.80	14.80	9.40	9.00	8.60	7.00	2.80	5.20	14.80
	Sıra										

ZDT2 ve ZDT6 problemlerinde Çok Amaçlı Dağılık Arama (ÇADA) algoritması 20 yöntem arasından en iyi sonucu vermiştir. ZDT1 ve ZDT3 problemlerinde ise ÇADA algoritması 2. sırada gelmiştir. ZDT4 probleminde ÇADA yöntemi 6. olarak orta düzeyde başarı elde etmiştir. ÇADA

yöntemi 2.40 ortalama sıralamasıyla en başarılı yöntem olmuştur. MWDEO yöntemi ise 3.00 ortalama ile 2. sırada gelmiştir. Çizelge 7’de DTLZ problemleri üzerinde 30 kez çalıştırma sonucunda elde edilen IGD ölçümlerinin ortalama sonuçları ve karşılık gelen standart sapma (Std) değerleri gösterilmektedir.

Çizelge 7. Önerilen algoritmanın IGD sonuçları ve DTLZ problemleri üzerinde karşılaştırılan algoritmalar

		MWDEO	MWDE	MOEAD	RVEA	MOPSO	SMP SO	NSGAI I	MOALO	SHAMODEWO	ÇADA
DTLZ1	Ortalama	0.0110	43.2200	0.0084	0.0157	5.2281	1.8398	0.0120	0.0944	0.0037	0.0274
	Std	0.0003	17.400	0.0001	0.0085	1.6023	1.6628	0.0003	8.8723	1.0001	0.0501
	Sıra	5	19	3	7	17	15	6	13	2	8
DTLZ2	Ortalama	0.0218	0.4985	0.0219	0.0231	0.0324	0.0304	0.0309	0.2146	0.0501	0.0268
	Std	0.0000	0.0089	0.0000	0.0007	0.0007	0.0005	0.0004	0.0251	0.0016	0.0088
	Sıra	1	19	2	3	10	8	9	17	13	7
DTLZ3	Ortalama	0.0229	248.38	0.0307	0.4486	53.6958	8.7553	0.0344	1.6806	0.0493	0.0786
	Std	0.0022	3.7000	0.0056	0.2974	35.8193	9.7907	0.0024	1.4903	7.3306	0.1734
	Sıra	1	20	2	9	18	16	3	12	6	7
DTLZ4	Ortalama	0.0251	0.8685	0.1257	0.0226	0.1105	0.2563	0.0304	0.8351	0.8997	0.0266
	Std	0.0009	0.1710	0.2076	0.0002	0.0967	0.0902	0.0004	0.0666	0.0013	0.0087
	Sıra	4	16	11	2	10	12	8	14	17	6
DTLZ5	Ortalama	0.0009	0.4268	0.0135	0.0393	0.0025	0.0011	0.0011	0.1107	0.0035	0.0012
	Std	0.0000	0.0575	0.0000	0.0041	0.0002	0.0000	0.0001	0.0091	0.0001	0.0004
	Sıra	1	18	11	14	6	2	2	15	8	3
DTLZ6	Ortalama	0.0010	2.8857	0.0136	0.0446	1.3185	0.0011	0.0011	0.0030	0.0135	0.0035
	Std	0.0000	1.6800	0.0000	0.0081	0.7897	0.0000	0.0000	0.0184	0.0679	0.0019
	Sıra	2	17	12	15	16	3	3	4	11	6
DTLZ7	Ortalama	0.0224	4.5597	0.06	0.0604	0.6932	0.0375	0.0313	0.1415	0.0449	0.0391
	Std	0.0005	0.968	0.0006	0.0038	0.3972	0.0027	0.0012	0.0866	0.0007	0.0541
	Sıra	1	19	11	12	17	6	5	14	8	7
Ortalama Sıra		2.14	18.29	7.43	8.86	13.43	8.86	5.14	12.71	9.29	6.29
		SHAMODE	MOMVO	MOGOA	MOGWO	KnEA	PICEAg	GrEA	NSGAI I I I	MOEADSTM	RVEAI GNG
DTLZ1	Ortalama	0.0028	0.0614	0.1971	0.0742	0.0108	0.0472	0.0318	0.0084	2.5647	10.8794
	Std	0.5908	1.1161	12.8282	4.6956	0.0005	0.0298	0.0221	0.0001	0.7207	1.7325
	Sıra	1	11	14	12	4	10	9	3	16	18
DTLZ2	Ortalama	0.0485	0.0766	0.2978	0.118	0.0265	0.024	0.049	0.0219	0.0281	0.1567
	Std	0.0011	0.0063	0.0392	0.0201	0.0003	0.0001	0.0007	0.0000	0.0007	0.0385
	Sıra	11	14	18	15	5	4	12	2	6	16
DTLZ3	Ortalama	0.1044	0.5908	1.5597	1.7616	2.5219	7.4608	0.0489	0.0400	11.4591	169.3794
	Std	21.6842	18.5734	5.1478	35.5534	1.3436	0.1626	0.0155	0.0094	0.4788	53.0521
	Sıra	8	10	11	13	14	15	5	4	17	19
DTLZ4	Ortalama	0.9095	0.9191	0.8564	0.9171	0.0267	0.0241	0.0479	0.0220	0.0288	0.3855
	Std	0.0011	0.0017	0.0569	0.0015	0.0004	0.0002	0.0011	0.0000	0.0013	0.0967
	Sıra	18	20	15	19	5	3	9	1	7	13
DTLZ5	Ortalama	0.0032	0.0111	0.1596	0.0153	0.0017	0.0011	0.0202	0.0023	0.0043	0.1183
	Std	0.0001	0.0096	0.0091	0.0023	0.0001	0	0.0011	0.0003	0.0002	0.0325
	Sıra	7	10	17	12	4	2	13	5	9	16
DTLZ6	Ortalama	0.0262	0.0076	0.0089	0.0047	0.0009	0.0010	0.0220	0.0037	0.0045	7.0499
	Std	0.1576	0.002	0.2045	0.0002	0.0000	0.0004	0.0004	0.0004	0.0000	1.6570
	Sıra	14	9	10	8	1	2	13	5	7	18
DTLZ7	Ortalama	0.0546	0.1664	0.1155	0.3200	0.0274	0.0268	0.0552	0.0291	1.2677	4.7924
	Std	0.0019	0.0339	0.621	0.4509	0.0008	0.0005	0.0025	0.0005	0.3079	1.2132
	Sıra	9	15	13	16	3	2	10	4	18	20
Ortalama Sıra		9.71	12.71	14.00	13.57	5.14	5.43	10.14	3.43	11.43	17.14

ÇADA yöntemi, IGD sonuçlarına bakıldığında DTLZ1 probleminde 3. başarılı yöntem olmuştur. Diğer problemlerde ise 6-8 arasında başarılar elde etmiştir. MWDEO yöntemi ise 4 problem üzerinde en başarılı sonuca ulaşan algoritma olmuştur. Diğer problemlerde de ilk 5'e girmeyi başarmıştır. Ortalama değerlere bakıldığında, en başarılı yöntem MWDEO olmuştur. ÇADA yöntemi ise 20 yöntem arasından ortalama olarak 6.sırada gelmektedir.

4. Sonuçlar ve Tartışma

Dağınık Arama metodu tek amaçlı optimizasyon problemleri için geliştirilmiş bir algoritmadır. Dolayısıyla, çok amaçlı problemlerin çözümünde etkili sonuçlar elde edememektedir. Bu çalışmada, Dağınık Arama yöntemine NSGA-II metodunun Yoğunluk Mesafesi (CD) ve Hızlı Baskınlık Sıralaması kavramlarını ekleyerek hibrit birçok amaçlı optimizasyon algoritması geliştirilmiştir. Önerilen Çok Amaçlı Dağınık Arama (ÇADA) algoritmasının performansını değerlendirmek için ZDT ve DTLZ test problemleri üzerinde deneysel çalışmalar yapılmıştır. Elde edilen sonuçlar, 19 farklı çok amaçlı optimizasyon yöntemleriyle karşılaştırılmıştır. Karşılaştırma sonuçlarına göre, ÇADA algoritması ZDT problemi üzerinde en başarılı yöntem olurken, DTLZ problemleri üzerinde 6. sırada yer almıştır.

Mevcut optimizasyon algoritmalarının yüksek boyutlu ve çok sayıda amaç fonksiyonu içeren problemler karşısında yetersizlikleri bulunmaktadır. Bu çalışmada geliştirilen hibrit algoritma, bu eksiklikleri gidererek, hesaplama verimliliğini artırmakta ve yerel optima takılma sorunlarını azaltmaktadır. Çalışmamız, NSGA-II algoritmasının avantajlarını Scatter Search ile birleştirerek daha etkin bir çözüm sunmaktadır.

NSGA-II, çok amaçlı optimizasyon problemlerini çözmedeki başarısıyla tanınırken, üç veya daha fazla hedef sunulduğunda performansında düşüş yaşanabilir. Bu durum, ZDT problemlerinde elde edilen başarılı sonuçlara karşılık DTLZ problemlerinde elde edilen altıncılık sıralamasıyla da doğrulanmıştır. Gelecek çalışmalarda, NSGA-II'nin bir uzantısı olan ve üçten fazla hedefi ele almak için geliştirilmiş güçlü bir yöntem olan NSGA-III'ün kullanımı önerilebilir. NSGA-III ve dağınık arama yöntemlerinin hibritlenmesi, incelenen problemler üzerinde NSGA-II ve NSGA-III algoritmalarının karşılaştırılması ilginç bir araştırma konusu olabilir. Ayrıca, diğer gelişmiş çok amaçlı optimizasyon algoritmaları, örneğin HypE ve SPEA2, ile de karşılaştırmalar yapılarak ÇADA algoritmasının performansı daha kapsamlı bir şekilde değerlendirilebilir.

Sonuç olarak, bu çalışma, çok amaçlı optimizasyon problemlerinin çözümüne yönelik daha verimli ve etkili bir yaklaşım sunarak literatüre önemli katkılar sağlamaktadır. ÇADA algoritmasının gösterdiği performans, hem teorik hem de uygulamalı alanlarda potansiyel faydalar sunabilir.

Etik Standartlar Bildirgesi

Bu çalışmanın hazırlanma sürecinde bilimsel ve etik ilkelere uyulduğu ve yararlanılan tüm çalışmaların kaynakçada belirtildiği beyan olunur.

Yazarlık Katkı Beyanı

Yazar 1: Kaynaklar, Araştırma, Deneysel, Yazma – orijinal taslak
Görselleştirme, Yazma – orijinal taslak

Yazar 2: Araştırma, Biçimsel analiz, Doğrulama, Metodoloji,
Görselleştirme, Yazma/inceleme ve düzenleme

Çıkar Çatışması Beyanı

Yazarların bu makalenin içeriğiyle ilgili olarak beyan edecekleri hiçbir çıkar çatışması yoktur.

Verilerin Kullanılabilirliği

Bu çalışma sırasında oluşturulan veya analiz edilen tüm veriler, yayınlanan bu makaleye dahil edilmiştir.

5. Kaynaklar

Abed-Arguni, B. H. ve Paul, D. J., 2020, Hybridizing the cuckoo search algorithm with different mutation operators for numerical optimization problems, *Journal of Intelligent Systems*, **29** (1), 1043-1062. <https://doi.org/10.1515/jisys-2018-0331>

Azizi, S., SoltanAghaei, M., Ghaffarian, H. ve Javadpour, A., 2023, Improving Hierarchical Traffic Engineering and Reducing the Congestion with PSO in SDN and DataCenter Networks, Research Square, Preprint, Version1, <https://doi.org/10.21203/rs.3.rs-1887790/v1>

Bilici, Z. ve Özdemir, D., 2023, Meteorolojik parametreler ile doğal gaz talep tahmini için metasezgisel optimizasyon algoritmalarının karşılaştırmalı analizi, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, **38** (2), 1153-1168. <https://doi.org/10.17341/gazimmfd.1014788>

Chen, H., Feng, Z., Liu, Y., Chen, B., Deng, T., Qin, Y. ve Xu, W., 2023, Multiobjective Optimization of a 3D Laser Scanning Scheme for Engineering Structures Based on RF-NSGA-II, *Journal of Construction Engineering and Management*, **149** (2), 04022169. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0002411](https://doi.org/10.1061/(ASCE)CO.1943-7862.0002411)

Cheng, R., Jin, Y., Olhofer, M. ve Sendhoff, B., 2016, A reference vector guided evolutionary algorithm for many-objective optimization, *IEEE transactions on evolutionary computation*, **20** (5), 773-791. <https://doi.org/10.1109/TEVC.2016.2519378>

- Cheraghi, R. ve Jahangir, M. H., 2023, Multi-objective optimization of a hybrid renewable energy system supplying a residential building using NSGA-II and MOPSO algorithms, *Energy Conversion and Management*, **294**, 117515.
<https://doi.org/10.1016/j.enconman.2023.117515>
- Coello, C. C. ve Lechuga, M. S., 2002, MOPSO: A proposal for multiple objective particle swarm optimization, *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, 1051-1056.
- Corne, D. W., Jerram, N. R., Knowles, J. D. ve Oates, M. J., 2001, PESA-II: Region-based selection in evolutionary multiobjective optimization, *Proceedings of the 3rd annual conference on genetic and evolutionary computation*, 283-290.
- Deb, K. ve Goyal, M., 1996, A combined genetic adaptive search (GeneAS) for engineering design, *Computer Science and informatics*, **26**, 30-45.
- Deb, K., Agrawal, S., Pratap, A. ve Meyarivan, T., 2000, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, *International conference on parallel problem solving from nature*, 849-858.
- Deb, K., Pratap, A., Agarwal, S. ve Meyarivan, T., 2002, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE transactions on evolutionary computation*, **6 (2)**, 182-197.
<https://doi.org/10.1109/4235.996017>
- Deb, K., Thiele, L., Laumanns, M. ve Zitzler, E., 2005, Scalable test problems for evolutionary multiobjective optimization, In: *Evolutionary multiobjective optimization*, Eds: Springer, p. 105-145.
- Deb, K. ve Tiwari, S., 2008, Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization, *European Journal of Operational Research*, **185 (3)**, 1062-1087.
<https://doi.org/10.1016/j.ejor.2006.06.042>
- Deb, K. ve Jain, H., 2013, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints, *IEEE transactions on evolutionary computation*, **18 (4)**, 577-601.
<https://doi.org/10.1109/TEVC.2013.2281535>
- Dörterler, S., Dumlu, H., Özdemir, D. ve Temurtaş, H., 2022, Melezlenmiş K-means ve Diferansiyel Gelişim Algoritmaları ile Kalp Hastalığının Teşhisi, *International Conference on Engineering and Applied Natural Sciences içinde (ss. 1840-1844). Konya*.
- Emel, G. G. ve Taşkın, Ç., 2002, Genetik algoritmalar ve uygulama alanları, *Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, **21 (1)**, 129-152.
- Ewees, A. A., Abd Elaziz, M. ve Oliva, D., 2021, A new multi-objective optimization algorithm combined with opposition-based learning, *Expert Systems with Applications*, **165**, 113844.
<https://doi.org/10.1016/j.eswa.2020.113844>
- Glover, F., 1977, Heuristics for integer programming using surrogate constraints, *Decision sciences*, **8 (1)**, 156-166.
<https://doi.org/10.1111/j.1540-5915.1977.tb01074.x>
- Hakli, H. ve Ortacay, Z., 2019, An improved scatter search algorithm for the uncapacitated facility location problem, *Computers & Industrial Engineering*, **135**, 855-867.
<https://doi.org/10.1016/j.cie.2019.06.060>
- Hassan, B. A., 2021, CSCF: a chaotic sine cosine firefly algorithm for practical application problems, *Neural Computing and Applications*, **33 (12)**, 7011-7030.
<https://doi.org/10.1007/s00521-020-05474-6>
- Heidari, A., Imani, D. M., Khalilzadeh, M. ve Sarbazvatan, M., 2023, Green two-echelon closed and open location-routing problem: application of NSGA-II and MOGWO metaheuristic approaches, *Environment, Development and Sustainability*, **25 (9)**, 9163-9199
<https://doi.org/10.1007/s10668-022-02429-w>
- Huang, Y.-F. ve Chen, S.-H., 2020, Solving multi-objective optimization problems using self-adaptive harmony search algorithms, *Soft Computing*, **24 (6)**, 4081-4107.
<https://doi.org/10.1007/s00500-019-04175-0>
- Hughes, E. J., 2003, Multiple single objective Pareto sampling, *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, 2678-2684.
- Kamjoo, A., Maheri, A., Dizqah, A. M. ve Putrus, G. A., 2016, Multi-objective design under uncertainties of hybrid renewable energy system using NSGA-II and chance constrained programming, *International Journal of Electrical Power & Energy Systems*, **74**, 187-194.
<https://doi.org/10.1016/j.ijepes.2015.07.007>
- Knowles, J. D. ve Corne, D. W., 2000, Approximating the nondominated front using the Pareto archived evolution strategy, *Evolutionary computation*, **8 (2)**, 149-172.
<https://doi.org/10.1162/106365600568167>
- Li, K., Zhang, Q., Kwong, S., Li, M. ve Wang, R., 2013, Stable matching-based selection in evolutionary multiobjective optimization, *IEEE transactions on evolutionary computation*, **18 (6)**, 909-923.
<https://doi.org/10.1109/TEVC.2013.2293776>
- Li, X. ve Li, S., 2021, An adaptive surrogate-assisted particle swarm optimization for expensive problems, *Soft Computing*, **25 (24)**, 15051-15065.
<https://doi.org/10.1007/s00500-021-06348-2>

- Liu, Q., Jin, Y., Heiderich, M., Rodemann, T. ve Yu, G., 2022, An adaptive reference vector-guided evolutionary algorithm using growing neural gas for many-objective optimization of irregular problems, *IEEE Transactions on Cybernetics*, **52**, 5, 2698-2711. <https://doi.org/10.1109/TCYB.2020.3020630>
- Ma, H., Wei, H., Tian, Y., Cheng, R. ve Zhang, X., 2021, A multi-stage evolutionary algorithm for multi-objective optimization with complex constraints, *Information Sciences*, **560**, 68-91. <https://doi.org/10.1016/j.ins.2021.01.029>
- Mirjalili, S., Saremi, S., Mirjalili, S. M. ve Coelho, L. d. S., 2016, Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization, *Expert Systems with Applications*, **47**, 106-119. <https://doi.org/10.1016/j.eswa.2015.10.039>
- Mirjalili, S., Jangir, P. ve Saremi, S., 2017, Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems, *Applied Intelligence*, **46** (1), 79-95. <https://doi.org/10.1007/s10489-016-0825-8>
- Mirjalili, S. Z., Mirjalili, S., Saremi, S., Faris, H. ve Aljarah, I., 2018, Grasshopper optimization algorithm for multi-objective optimization problems, *Applied Intelligence*, **48** (4), 805-820. <https://doi.org/10.1007/s10489-017-1019-8>
- Murata, T. ve Ishibuchi, H., 1995, MOGA: multi-objective genetic algorithms, *IEEE international conference on evolutionary computation*, 289-294.
- Nebro, A. J., Durillo, J. J., Garcia-Nieto, J., Coello, C. C., Luna, F. ve Alba, E., 2009, SMP-PSO: A new PSO-based metaheuristic for multi-objective optimization, *2009 IEEE Symposium on computational intelligence in multi-criteria decision-making (MCDM)*, 66-73.
- Özdemir, D. ve Dörterler, S., 2022, An adaptive search equation-based artificial bee colony algorithm for transportation energy demand forecasting, *Turkish Journal of Electrical Engineering and Computer Sciences*, **30** (4), 1251-1268. <https://doi.org/10.55730/1300-0632.3847>
- Panagant, N., Bureerat, S. ve Tai, K., 2019, A novel self-adaptive hybrid multi-objective meta-heuristic for reliability design of trusses with simultaneous topology, shape and sizing optimisation design variables, *Structural and Multidisciplinary Optimization*, **60** (5), 1937-1955. <https://doi.org/10.1007/s00158-019-02302-x>
- Patil, R. N., Rawandale, S., Rawandale, N., Rawandale, U. ve Patil, S., 2023, An efficient stacking based NSGA-II approach for predicting type 2 diabetes, *International Journal of Electrical and Computer Engineering (IJECE)*, **13** (1), 1015-1023. <https://doi.org/10.11591/ijece.v13i1.pp1015-1023>
- Peng, S., Liu, Q. ve Hu, J., 2023, Green Distribution Route Optimization of Medical Relief Supplies Based on Improved NSGA-II Algorithm under Dual-Uncertainty, *Sustainability*, **15** (15), 11939. <https://doi.org/10.3390/su151511939>
- rey Horn, J., Nafpliotis, N. ve Goldberg, D. E., 1994, A niched Pareto genetic algorithm for multiobjective optimization, *Proceedings of the first IEEE conference on evolutionary computation, IEEE world congress on computational intelligence*, 82-87.
- Schaffer, J. D., 1985, Multiple objective optimization with vector evaluated genetic algorithms, *Proceedings of the first international conference on genetic algorithms and their applications*, 1985.
- Sekkal, M., Benzina, A. ve badir Benkrelifa, L., 2024, Multi-Objective Evolutionary Algorithm based on NSGA-II for Neural Network Optimization Application to the Prediction of Severe Diseases, *Informatica*, **47** (10), 27-40. <https://doi.org/10.31449/inf.v47i10.5126>
- Tian, J., Tan, Y., Sun, C., Zeng, J. ve Jin, Y., 2016, A self-adaptive similarity-based fitness approximation for evolutionary optimization, *2016 IEEE symposium series on computational intelligence (SSCI)*, 1-8.
- Wang, R., Purshouse, R. C. ve Fleming, P. J., 2012, Preference-inspired coevolutionary algorithms for many-objective optimization, *IEEE transactions on evolutionary computation*, **17** (4), 474-494. <https://doi.org/10.1109/TEVC.2012.2204264>
- Yang, S., Li, M., Liu, X. ve Zheng, J., 2013, A grid-based evolutionary algorithm for many-objective optimization, *IEEE transactions on evolutionary computation*, **17** (5), 721-736. <https://doi.org/10.1109/TEVC.2012.2227145>
- Zeng, S., Yao, S., Kang, L. ve Liu, Y., 2005, An efficient multi-objective evolutionary algorithm: OMOEA-II, *International Conference on Evolutionary Multi-Criterion Optimization*, 108-119.
- Zhang, Q. ve Li, H., 2007, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, *IEEE transactions on evolutionary computation*, **11** (6), 712-731. <https://doi.org/10.1109/TEVC.2007.892759>
- Zhang, X., Tian, Y. ve Jin, Y., 2014, A knee point-driven evolutionary algorithm for many-objective optimization, *IEEE transactions on evolutionary computation*, **19** (6), 761-776. <https://doi.org/10.1109/TEVC.2014.2378512>
- Zitzler, E., Deb, K. ve Thiele, L., 2000, Comparison of multiobjective evolutionary algorithms: Empirical results, *Evolutionary computation*, **8** (2), 173-195.
- Zitzler, E., Laumanns, M. ve Thiele, L., 2001, SPEA2: Improving the strength Pareto evolutionary algorithm, *TIK-report*, 103.

<https://doi.org/10.3929/ethz-a-004284029>

Zou, W., Zhu, Y., Chen, H. ve Zhang, B., 2011, Solving multiobjective optimization problems using artificial bee colony algorithm, *Discrete dynamics in nature and society*, 2011, 569784.
<https://doi.org/10.1155/2011/569784>