

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DAĞITIM TRANSFORMATÖRLERİNİN
METASEZGİSEL ALGORİTMALARLA
TASARIM OPTİMİZASYONU**

DOKTORA TEZİ

Levent ALHAN

Enstitü Anabilim Dalı : **BİLGİSAYAR VE BİLİŞİM
MÜHENDİSLİĞİ**
Tez Danışmanı : **Prof. Dr. Nejat YUMUŞAK**

Temmuz 2017

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

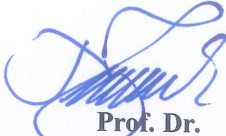
DAĞITIM TRANSFORMATÖRLERİNİN
METASEZGİSEL ALGORİTMALARLA
TASARIM OPTİMİZASYONU

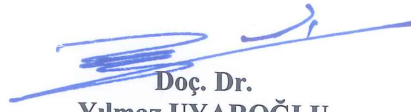
DOKTORA TEZİ


Levent ALHAN

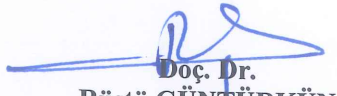
Enstitü Anabilim Dalı : BİLGİSAYAR VE BİLİŞİM
MÜHENDİSLİĞİ

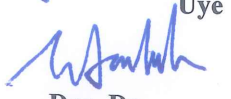
Bu tez 07 / 07 /2017 tarihinde aşağıdaki jüri tarafından oybirliği/oyçokluğu ile kabul edilmiştir.


Prof. Dr.
Nejat YUMUŞAK
Jüri Başkanı


Doç. Dr.
Yılmaz UYAROĞLU
Üye


Doç. Dr.
İhsan PEHLİVAN
Üye


Doç. Dr.
Rüştü GÜNTÜRKÜN
Üye


Doç. Dr.
Ayhan İSTANBULLU
Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Levent ALHAN

25.05.2017

TEŐEKKÜR

4.5 yıl kadar önce 64 yaşında özel öğrenci olarak başladığım doktora öğrenimini devam ettirmem konusunda beni teşvik eden ve bu sürecin her aşamasında bana yardımcı ve destek olan tez danışmanım Prof. Dr. Nejat Yumuşak'a ve tez ilerleme komitesinde yer alan değerli öğretim üyeleri Doç. Dr. Kürşat Ayan ve Doç. Dr. Yılmaz Uyarođlu'na teşekkürlerimi sunarım.

İÇİNDEKİLER

TEŞEKKÜR	i
İÇİNDEKİLER.....	ii
SİMGELER VE KISALTMALAR LİSTESİ	v
ŞEKİLLER LİSTESİ.....	vii
TABLOLAR LİSTESİ	viii
ÖZET.....	x
SUMMARY	xi
BÖLÜM 1.	
GİRİŞ	1
BÖLÜM 2.	
TRANSFORMATÖR TASARIM OPTİMİZASYONU.....	7
2.1. Karma-Tamsayılı Transformatör Tasarım Optimizasyonu (TDO)	7
2.2. Ayrık Transformatör Tasarım Optimizasyonu (DTDO)	9
BÖLÜM 3.	
METASEZGİSEL ALGORİTMALAR	11
3.1. Yapay Arı Kolonisi Algoritması (ABC).....	11
3.1.1. Kısıtsız optimizasyon problemleri için geliştirilen ABC algoritması	11
3.1.2. Kısıtlı optimizasyon problemleri için geliştirilen ABC algoritması	14
3.2. Rekabetçi-Uyarlamalı Diferansiyel Gelişim Algoritması	16
3.3. Geri-İzleme Arama Optimizasyon Algoritması (BSA).....	20
3.3.1. Başlama	21

3.3.2. Seleksiyon-I.....	21
3.3.3. Mutasyon	22
3.3.4. Çaprazlama	22
3.3.5. Seleksiyon-II	22
3.4. Guguk Kuşu Arama Algoritması (CS)	23
3.5. Çiçek Tozlaşma Algoritması (FPA)	25
BÖLÜM 4.	
ÖNBELLEKLEME	29
4.1. Bilgisayarlarda Önbellekleme	32
4.2. Amaç Fonksiyonu Hesaplama Sayısının Azaltılması.....	33
4.3. Mühendislik Tasarım Optimizasyonu Problemleri	35
4.4. Parametre Ayarları, Hesaplama Sonuçları ve Değerlendirme.....	35
4.4.1. Parametre ayarları.....	36
4.4.2. Hesaplama sonuçları ve değerlendirme.....	36
4.4.2.1. Önbelleklemesiz sonuçlar	39
4.4.2.2. Önbelleklemeli sonuçlar, Versiyon I	39
4.4.2.3. Önbelleklemeli sonuçlar, Versiyon II.....	40
BÖLÜM 5.	
DAĞITIM TRANSFORMATÖRLERİ TASARIM OPTİMİZASYONU	41
5.1. Çalışmanın Amacı ve Kapsamı	41
5.2. Problemin Literatürdeki Tanımları.....	42
5.2.1. Amaç fonksiyonları	42
5.2.2. Kısıtlar	43
5.3. Problemin Bu Çalışmada Kullanılan Tanımı	44
5.3.1. Amaç fonksiyonu ve kısıtlar.....	44
5.3.2. Tasarım değişkenleri.....	45
5.4. Tasarım Yöntemi	47
5.4.1. Boşta kayıplar eğrisi.....	47
5.4.2. Kazan konveksiyon ve radyasyon ile ısı transfer katsayısı eğrileri.....	48

5.4.3. Dalgalı panel ısı transfer katsayısı eğrileri	50
5.5. Kısıt Yönetimi	50
5.6. Algoritma Kıyaslama (Benchmarking) Yöntemi	52
5.7. Algoritma Kontrol Parametrelerinin Ayarlanması	54
BÖLÜM 6.	
HESAPLAMA SONUÇLARI VE DEĞERLENDİRME	57
6.1. İncelenen Transformatörlerin Teknik Özellikleri ve Kısıtları.....	57
6.2. Teorik Transformatör Tasarım Optimizasyonu (TDO) Performans Testleri	58
6.2.1. Test yöntemi	58
6.2.2. Test sonuçları ve değerlendirme	59
6.3. Pratik Transformatör Tasarım Optimizasyonu (TDO) Performans Testleri	64
6.3.1. Test yöntemi	64
6.3.2. Test sonuçları ve değerlendirme	65
6.4. Teorik ve Pratik TDO Çözümlerinin Karşılaştırılması	71
6.5. Önbelleklemeli Pratik TDO.....	73
6.6. Grafik Kullanıcı Arayüzü TDO Programı.....	75
BÖLÜM 7.	
SONUÇLAR VE ÖNERİLER	78
KAYNAKLAR.....	81
EKLER	86
ÖZGEÇMİŞ	98

SİMGELER VE KISALTMALAR LİSTESİ

AB	: Avrupa Birliği
ABC	: Yapay Arı Kolonisi (Artificial Bee Colony)
AG	: Alçak Gerilim
b6e6rl	: Rekabetçi-Uyarlamalı Diferansiyel Gelişim (Competitive-Adaptive Differential Evolution)
BSA	: Geri-İzleme Arama Optimizasyon Algoritması (Backtracking Search Optimization Algorithm)
CMA-ES	: Kovaryans Matris Uyum Gelişimi Stratejisi (Covariance Matrix Adaptation Evolution Strategy)
CPU	: Merkezi İşlem Birimi (Central Processing Unit)
CS	: Guguk Kuşu Arama (Cuckoo Search)
DAG	: Yönlendirilmiş Çevrimsiz Çizge (Directed Acyclic Graph)
DE	: Diferansiyel Gelişim (Differential Evolution)
DNS	: Alan Adı Sistemi (Domain Name System)
DTDO	: Ayrık Transformatör Tasarım Optimizasyonu (Discrete Transformer Design Optimization)
EEA	: Avrupa Ekonomik Alanı (European Economic Area)
FA	: Ateşböceği Algoritması (Firefly Algorithm)
FPA	: Çiçek Tozlaşma Algoritması (Flower Pollination Algorithm)
GA	: Genetik Algoritma (Genetic Algorithm)
GHP	: Genetik Programlama (Genetic Programming)
GPU	: Grafik İşlem Birimi (Graphics Processing Unit)
GVA	: Gigavoltamper
kVA	: Kilovoltamper
kWh	: Kilowattsaat
MDM	: Çoklu Tasarım Yöntemi (Multiple Design Method)

MW	: Megawatt
TDO	: Transformatör Tasarım Optimizasyonu (Transformer Design Optimization)
TOC	: Toplam Sahiplik Maliyeti (Total Owning Cost)
TWh	: Terawattsaat
T&D	: İletim ve Dağıtım (Transmission and Distribution)
USD	: ABD Doları
YG	: Yüksek Gerilim

ŞEKİLLER LİSTESİ

Şekil 1.1. Dağıtım transformatörü - dış görünüş.....	6
Şekil 1.2. Dağıtım transformatörü - aktif kısım	6
Şekil 3.1. ABC algoritması sözde kodu	16
Şekil 3.2. DE algoritması sözde kodu	17
Şekil 3.3. CS algoritması sözde kodu	25
Şekil 3.4. FPA algoritması sözde kodu	28
Şekil 4.1. Önbellekleme ile elde edilen hesaplama sayılarındaki azalmalar.....	38
Şekil 5.1. Sarma çakirdek tipi dağıtım transformatörünün aktif kısmı [50]	46
Şekil 5.2. Dağıtım transformatörü kesit resmi ve boyutlar [1]	47
Şekil 5.3. Boşta kayıplar eğrisi [1].....	48
Şekil 5.4. Kazan konveksiyon ve radyasyon ile ısı transfer katsayısı eğrileri [1]	49
Şekil 5.5. Dalgalı panel ısı transfer katsayısı eğrileri [1].....	50
Şekil 6.1. TDO - Her bir transformatör tipi için ve ortalama güvenilirlik puanları ...	64
Şekil 6.2. DTDO-Her bir transformatör tipi için ve ortalama güvenilirlik puanları ..	71
Şekil 6.3. 400 kVA Teorik TDO kullanıcı arayüzü örneği	76
Şekil 6.4. 400 kVA Pratik TDO kullanıcı arayüzü örneği	76
Şekil 6.5. 400 kVA Pratik TDO Excel çıktı örneği (5134 uygulanabilir çözüm).....	77
Şekil A. 1. Basınçlı kap tasarımı problemi [33].....	86
Şekil A. 2. Kaynaklı giriş tasarımı problemi [33].....	87
Şekil A. 3. Sarmal yay tasarımı problemi [33]	89
Şekil A. 4. Güçlendirilmiş betonarme giriş tasarımı problemi [33].....	90

TABLolar LİSTESİ

Tablo 1.1. Dağıtım transformatörleri için Avrupa Standartlarının karşılaştırılması....	4
Tablo 1.2. Standart ve yüksek verimli dağıtım transformatörleri arasındaki boyut ve ağırlık farklılıkları	5
Tablo 3.1. b6e6rl algoritması DE stratejileri ve kontrol parametre değerleri	20
Tablo 4.1. Kaynaklı giriş problemi - farklı durdurma kriterleri için sonuçlar	31
Tablo 4.2. Her problem için FA ve FPA algoritmalarının 100 çalıştırma sonuçları ..	37
Tablo 4.3. Kaynaklı giriş tasarımı problemi için elde edilen en iyi çözümlerin tasarım değişkeni ve kısıtlarının karşılaştırması	38
Tablo 5.1. Ana malzemeler ve birim maliyetleri	45
Tablo 5.2. Teorik TDO tasarım değişkenleri	46
Tablo 5.3. Pratik TDO tasarım değişkenleri	46
Tablo 5.4. Algoritma kıyaslama puanlama sistemi	53
Tablo 5.5. ABC kontrol parametrelerinin TDO problemi üzerindeki etkileri	55
Tablo 5.6. ABC <i>limit</i> = 1280, <i>SPP</i> = 640 ve <i>MR</i> = 0.9 kontrol parametreleri ile 160 kVA optimizasyon sonuçları	56
Tablo 5.7. ABC Delphi kodu ile bu çalışmada kullanılan kontrol parametrelerinin karşılaştırması	56
Tablo 6.1. İncelenen transformatörlerin kısıt ve toleransları	58
Tablo 6.2. TDO - 160 kVA için b6e6rl algoritması optimizasyon sonuçları.....	60
Tablo 6.3. TDO - 400 kVA için CS algoritması optimizasyon sonuçları	60
Tablo 6.4. TDO - 630 kVA için BSA algoritması optimizasyon sonuçları	61
Tablo 6.5. TDO - En iyi 20 güvenilirlik puanı.....	62
Tablo 6.6. TDO - Her algoritma için en iyi güvenilirlik puanı	62
Tablo 6.7. TDO - En iyi 20 toplam puan	63
Tablo 6.8. TDO - Her algoritma için en iyi toplam puan.....	63
Tablo 6.9. DTDO - 160 kVA için b6e6rl algoritması optimizasyon sonuçları.....	66

Tablo 6.10. DTDO - 400 kVA için BSA algoritması optimizasyon sonuçları	66
Tablo 6.11. DTDO - 630 kVA için CS algoritması optimizasyon sonuçları	67
Tablo 6.12. DTDO - En iyi 20 güvenilirlik puanı.....	69
Tablo 6.13. DTDO - Her algoritma için en iyi güvenilirlik puanları.....	69
Tablo 6.14. DTDO - En iyi 20 toplam puan	70
Tablo 6.15. DTDO - Her algoritma için en iyi toplam puanlar.....	70
Tablo 6.16. 630 kVA teorik ve pratik TDO çözümlerinin karşılaştırılması	72
Tablo 6.17. CS için amaç fonksiyonu hesaplama sayılarının karşılaştırılması.....	74
Tablo 6.18. FPA için amaç fonksiyonu hesaplama sayılarının karşılaştırılması	74
Tablo B. 1. 160 kVA Teorik optimizasyon sonuçları - tüm algoritmalar	92
Tablo B. 2. 400 kVA Teorik optimizasyon sonuçları - tüm algoritmalar	93
Tablo B. 3. 630 kVA Teorik optimizasyon sonuçları - tüm algoritmalar	94
Tablo C. 1. 160 kVA Pratik optimizasyon sonuçları - tüm algoritmalar	95
Tablo C. 2. 400 kVA Pratik optimizasyon sonuçları - tüm algoritmalar	96
Tablo C. 3. 630 kVA Pratik optimizasyon sonuçları - tüm algoritmalar	97

ÖZET

Anahtar kelimeler: Dağıtım transformatörü, tasarım optimizasyonu, yüksek verimlilik, metasezgisel algoritmalar, önbellekleme

Dünyadaki pek çok ekonomi, yüksek verimli transformatörlerin kullanımını zorunlu kılan veya teşvik eden enerji verimliliği yönetmelikleri veya teşvik programları kabul etmiştir. Öte yandan, transformatör verimliliğindeki artışlar, transformatör ağırlık ve boyutunda bazen % 50 hatta daha fazla bir artışı gerektirmektedir. Transformatör endüstrisi bu nedenle gerçekten en iyi tasarımları geliştirme uğraşısı ile karşı karşıyadır.

Transformatör tasarım optimizasyonu (TDO) problemi, karmaşık ve süreksiz amaç fonksiyonlu ve kısıtlı karma-tamsayı bir doğrusal olmayan programlama problemidir. TDO'nun amacı, ulusal ve/veya ulusal standartlar ve müşteri şartnameleri uyarınca, mevcut malzemeleri ekonomik olarak kullanarak daha düşük boyut, ağırlık ve maliyet ve daha yüksek işletme performansı elde etmek üzere transformatörün tüm bileşenlerinin niteliklerinin detaylı olarak hesaplanmasıdır.

Bu çalışmada TDO probleminin çözümü için beş modern metasezgisel optimizasyon algoritması uygulamasının ayrıntılı karşılaştırmalı analizi üç test vakası üzerinde gösterilmiş ve iki algoritma önerilmiştir; önerilen bu algoritmaların, rassal özelliklerine rağmen, garanti edilmiş küresel yakınsama özelliklerine sahip oldukları doğrulanmıştır. Algoritmaların karşılaştırılması için pragmatik bir kıyaslama yöntemi geliştirilmiştir.

Literatürde sunulan TDO yöntemleri nadiren üretimde doğrudan uygulanabilir çözümler üretir; tasarım mühendisinin genellikle teorik çözümü pratik olarak uygulanabilir bir hale dönüştürmek için ek çaba harcaması gerekir. Bu problem bu çalışmada ele alınmış ve piyasada mevcut veya üretime uygun boyutlara sahip çözümler üreten bir ayrık transformatör tasarım optimizasyon yöntemi önerilmiştir

Ayrıca, amaç fonksiyonu ve kısıt hesaplamalarını azaltmak için basit bir yöntem önerilmiştir. Yöntem, önbellekleme tekniği kullanılarak arama işlemi sırasında yinelenen tasarım vektörleri için hesaplamaların atlanması esasına dayanmaktadır.

Performans testleri, teorik TDO için Rekabetçi-Uyarlamalı Diferansiyel Gelişim ve Guguk Kuşu Arama, Pratik TDO için de Guguk Kuşu Arama ve Çiçek Tozlaşma algoritmaları kullanıldığında küresel optimum ve ona çok yakın sonuçlar elde edildiğini göstermiştir.

DESIGN OPTIMIZATION OF DISTRIBUTION TRANSFORMERS WITH METAHEURISTIC ALGORITHMS

SUMMARY

Keywords: Distribution transformer, design optimization, high efficiency, metaheuristic algorithms, caching

Many economies in the world have adopted energy-efficiency requirements or incentive programs mandating or promoting the use of energy-efficient transformers. On the other hand, increases in transformer efficiency are subject to increases in transformer weight and size, sometimes as much as 50% or even more. The transformer manufacturing industry is therefore faced with the challenge to develop truly optimum designs.

Transformer design optimization (TDO) is a mixed-integer nonlinear programming problem having complex and discontinuous objective function and constraints, with the objective of detailed calculation of the characteristics of a transformer based on national and/or international standards and transformer user requirements, using available materials and manufacturing processes, to minimize manufacturing cost or total owning cost, while maximizing operating performance.

Detailed comparative analysis of the application of five modern metaheuristic optimization algorithms for the solution of TDO problem are carried out in this study, demonstrated on three test cases; and two algorithms are proposed, for which it has been verified that they possess guaranteed global convergence properties in spite of their inherent stochastic nature. A pragmatic benchmarking scheme is used for comparison of the algorithms.

Transformer design optimization methods presented in the literature rarely yield solutions directly applicable in production; the design engineer usually needs to convert the theoretical solution to a practical one. This problem is addressed in this study, and a discrete transformer design optimization method is proposed which yields solutions with commercially available or productionally feasible dimensions

Furthermore, a simple method is proposed to reduce the number of objective function and constraint calculations. The method is based on skipping calculations for design vectors recurring during the search process, by the use of caching technique

Performance tests showed that global or near-global optimum solutions can be obtained with b6e6rl and CS for TDO, and CS and FPA algorithms for DTDO.

BÖLÜM 1. GİRİŞ

Transformatör, elektromanyetik endüksiyon yoluyla elektrik enerjisini frekansı değiştirmeden bir gerilim seviyesinden diğerine değiştiren statik bir elektrik makinasıdır. Bir transformatörde ferromanyetik bir çekirdeğin etrafına sarılmış iki veya daha fazla sargı bulunur, bu sargılar elektriksel olarak değil, manyetik olarak bağlanmıştır.

Transformatör sargılarından primer sargı olarak adlandırılan elektrik güç kaynağına bağlıdır. Sekonder sargı olarak adlandırılan diğeri de transformatöre bağlı olan yüklere elektrik gücü sağlar.

Bir transformatör, sekonder gerilimi primer gerilimden daha yüksek ise yükseltici, düşük ise indirici transformatör olarak adlandırılır.

İletim hatlarındaki güç kayıpları, hatlardaki akımın karesi ile doğru orantılıdır. Bu nedenle, güç santrallerinde üretilen elektrik enerjisinin gerilimi yükseltici transformatörler ile yükseltilerek uzun mesafelerde daha fazla gücün verimli bir şekilde iletilmesi sağlanır. İndirici transformatörler güç dağıtım şebekelerinde, fabrikalarda, ticari binalarda ve konutlarda gerilimi elektrikli alet ve cihazların çalışabileceği seviyeye düşürmek için kullanılır. Transformatörler, farklı gerilim seviyelerindeki güç sistemlerinin birbirine bağlanmasında da önemli bir rol oynamaktadır. Transformatörler olmadan, bugün birçok alanda kullanılan elektrik enerjisinden yararlanmak mümkün olmayacaktır. Sonuç olarak, transformatörler, elektrik üretim merkezleri ile elektrik enerjisi kullanım noktaları arasındaki hayati bağlantılar olarak güç sistemlerinde önemli yer tutmaktadır [1].

Elektrik enerjisi üretim ve tüketim noktaları arasında ortalama dört kez gerilim dönüşümüne uğramaktadır, bu nedenle İletim ve Dağıtım (T & D) şebekelerinde geniş bir çalışma gerilimi aralığında farklı sınıf ve boyutlarda birçok transformatör kullanılmaktadır. Bu zincirin sonundaki son kullanıcılara mesken tüketici gerilimi seviyelerinde (genellikle 400 V veya daha düşük) güç sağlayan transformatörlere geleneksel olarak dağıtım transformatörü denir.

2014 yılında dünyadaki tahmini dağıtım transformatörleri sayısı 118 milyon adettir. Toplam kurulu güç kapasitesi 13,848 GVA, ortalama ünite gücü 117 kVA'dır. 2030 yılında kurulu kapasitenin 22,400 GVA'ya ulaşacağı tahmin edilmektedir ve bu artışta Asya ve Hint Alt Kıtasının önemli katkıları olacaktır [2].

Özellikle gelişmekte olan ülkelerde kişi başına enerji tüketiminin artması nedeniyle, dağıtım transformatörleri küresel pazarının 2018 yılına kadar 20 milyar USD'yi aşması beklenmektedir [3].

Transformatörün çalışması esnasında ortaya çıkan kayıplar; boşa ve yükte çalışma kayıpları olarak iki grupta incelenmektedir. Demir kayıpları olarak da adlandırılan boşa çalışma kayıpları, transformatörün enerjili olduğu süre boyunca çekirdekte meydana gelen histeresis ve girdap (eddy) akımı kayıplarından oluşmaktadır ve transformatörün yüklenme oranından bağımsız, yani sabittir. Bakır kayıpları olarak da adlandırılan yükte çalışma (yük) kayıpları ise sargılar ve bileşenleri de dâhil olmak üzere transformatörün elektriksel devresinde meydana gelen direnç kayıplarından oluşmaktadır ve yüklenme oranına bağlıdır.

Dağıtım transformatörleri diğer elektrikli cihazlarla karşılaştırıldığında oldukça verimli olup, elektrik enerjisini bir gerilim düzeyinden diğerine dönüştürürken sadece % 2-3'lük kayıplar oluşturmaktadır. Bununla birlikte, tüm elektrik enerjisinin son kullanımdan önce birkaç kez transformatörlerden geçtiği gerçeği, dağıtım transformatörlerindeki kayıpları azaltma fırsatlarının, elektrik şebekelerinin bir bütün olarak verimliliğini artırmak için son derece önemli olduğunu göstermektedir.

2011 yılında iletim ve dağıtım şebekelerindeki kayıp oranı küresel enerji üretiminin yaklaşık % 8.5'i düzeyinde idi ve bunun % 87'si teknik kayıplardan kaynaklanıyordu. Bu kayıpların yaklaşık üçte biri dağıtım transformatörlerinde meydana gelmektedir, bu nedenle yılda tahmini 657 TWh elektrik enerjisi dağıtım transformatörü kayıplarına atfedilebilir [2].

Tahminler, yüksek verimli dağıtım transformatörlerinin benimsenmesiyle birlikte, bugün mevcut olan teknolojinin daha geniş kullanımı ile küresel teknik tasarruf potansiyelinin 2030'da yılda 402 TWh seviyesine ulaşacağını göstermektedir; bu da öngörülen küresel elektrik tüketiminin % 1.3'üne karşı gelmektedir. Bu sayede, yılda 201 milyon ton CO₂ emisyonu ve cari fiyatlarla 28 milyar USD toptan enerji satış maliyeti azalışı elde edilecektir [2].

Önemli ölçüde tasarruf potansiyeli nedeniyle, 2014 yılı itibarıyla kurulu dağıtım transformatörlerinin kapasitesine göre yaklaşık % 54'ünü temsil eden 15 kadar ülke (AB dâhil), enerji tasarruflu tasarımları ön plana çıkaran enerji verimliliği şartnameleri veya teşvik programları kabul etmiştir. Halen kabul edilen enerji verimliliği gerekliliklerinin bir sonucu olarak genel kayıpların 2030 yılında 154 TWh daha düşük olacağı öngörülmektedir. Bununla birlikte, daha fazla ülke benzer önlemleri benimseyip zorunluluk derecesi arttıkça çok daha büyük tasarruflar mümkün olabilecektir [2].

Avrupa Komisyonu, Avrupa Birliği'ndeki enerjinin yaklaşık % 2.5'inin transformatör kayıpları nedeniyle tüketildiğini tahmin etmektedir. Bir hazırlık çalışmasına dayanarak Avrupa Komisyonu tarafından transformatörler için özel Ecodesign standartları tanımlanmış ve 01.07.2015 tarihinde yeni Avrupa Standardı EN 50588-1 ile yürürlüğe girmiştir. Standart iki aşamada uygulanacaktır; Temmuz 2015'ten itibaren geçerli ilk aşamada, izin verilen azami boşta kayıplar, değiştirilen Avrupa Standardı EN 50464-1'in AC' (C₀C_k) kayıp kombinasyonu alternatifine kıyasla % 30 oranında azaltılacaktır. Temmuz 2021'de yürürlüğe girecek ikinci aşamada ise, izin verilen azami boşta kayıplar % 10, yük kayıpları da ~% 30 oranında azaltılacaktır.

Düşük verimli transformatörlerin kullanıma alınması artık mümkün olmayacağı için, yeni standardın Avrupa Ekonomik Alanı'nda (EEA) 2020'den itibaren yılda 16 TWh civarında tasarruf sağlayacağı beklenmektedir; bu da Danimarka'nın yıllık elektrik tüketimi ve 3.7 milyon ton CO₂ emisyonlarının yaklaşık yarısına eşittir.

Önceki ve mevcut Avrupa Standartlarındaki dağıtım transformatörü kayıplarının bir karşılaştırması Tablo 1.1.'de verilmiştir.

Tablo 1.1. Dağıtım transformatörleri için Avrupa Standartlarının karşılaştırılması

Güç	EN 50464-1 BA'			EN 50464-1 AC'			EN 50588-1 01.07.2015			EN 50588-1 01.07.2021		
	P ₀ (W)	P _k (W)	U _k (%)	P ₀ (W)	P _k (W)	U _k (%)	P ₀ (W)	P _k (W)	U _k (%)	P ₀ (W)	P _k (W)	U _k (%)
160 kVA	460	3,100	4	300	2,350	4	210	2,350	4	189	1,750	4
400 kVA	930	6,000	4	610	4,600	4	430	4,600	4	387	3,250	4
630 kVA	1,300	8,400	4	860	6,500	4	600	6,500	4 / 6	540	4,600	4 / 6
1000 kVA	1,700	13,000	6	1,100	10,500	6	770	10,500	6	693	7,600	6
Toleranslar	+15%	+15%	±10%	+15%	+15%	±10%	0%	0%	±10%	0%	0%	±10%
	+10%			+10%			0%			0%		

P₀ = Boşta kayıplar, P_k = Yük kayıpları, U_k = Kısa devre empedansı

Burada önemli olan husus, transformatör verimliliğindeki artışların transformatör ağırlık ve boyutlarında artışlara neden olmasıdır. Bunun nedeni, boşta ve/veya yük kayıplarının azaltılabilmesi için tasarımda daha fazla malzeme kullanılması gerekliliğidir.

Avrupa Standartlarında belirtilen kayıp değerlerine için hazırlanmamış olmasına rağmen, Tablo 1.2.'de verilen standart ve yüksek verimli 100 kVA ve 400 kVA dağıtım transformatörleri arasındaki boyut ve ağırlık farkları, verimliliğin artırılmasının etkisini göstermede yararlı olacaktır [2].

Tablo 1.2. Standart ve yüksek verimli dağıtım transformatörleri arasındaki boyut ve ağırlık farklılıkları

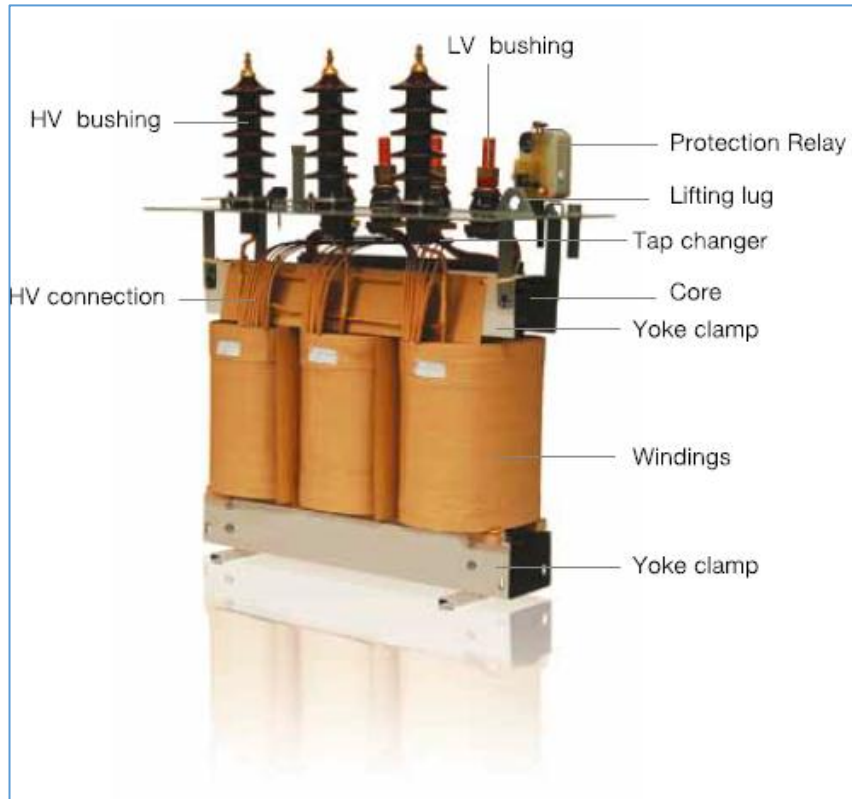
	100 kVA			400 kVA		
	Standart tasarım	Verimli tasarım		Standart tasarım	Verimli tasarım	
		Düşük boşa kayıplar	Düşük yük kayıpları		Düşük boşa kayıplar	Düşük yük kayıpları
Boşa kayıplar (W)	240	180	200	720	530	460
Yük kayıpları (W)	1,680	1.720	1.200	4.100	4.100	3.200
Toplam kayıplar (W)	1.920	1,900	1.400	4.820	4.630	3.660
Ağırlık (kg)	585	585	800	1.355	1,520	2.000
Hacim (m ³)	0.70	0.70	0.91	1.41	1.52	1.60
Topl kayıp değişimi		-1%	-27%		-4%	-24%
Ağırlık değişimi		0%	% 37		% 12	% 48
Hacim değişimi		0%	% 30		% 8	% 14

Kaynak: ABB Küresel web sitesi

Bir dağıtım transformatörünün dış görünüşü ile aktif kısmının resimleri Şekil 1.1. ve 1.2.'de verilmiştir.



Şekil 1.1. Dağıtım transformatörü - dış görünüş



Şekil 1.2. Dağıtım transformatörü - aktif kısım

BÖLÜM 2. TRANSFORMATÖR TASARIM OPTİMİZASYONU

2.1. Karma-Tamsayılı Transformator Tasarım Optimizasyonu (TDO)

Transformator üretim endüstrisi, yüksek verimli dağıtım transformatorlerinin kullanılmasını zorunlu kılan yeni düzenlemelerin yürürlüğe girmesinden sonra, günümüzün oldukça rekabetçi pazar ortamında müşterilerinin maliyetlerdeki kaçınılmaz artışları tam olarak karşılamalarını beklemek çok zor olacağından tartışılmaz en iyi (optimum) tasarımları geliştirme uğraşısı ile karşı karşıyadırlar.

Transformator tasarım optimizasyonunun amacı, ulusal ve/veya ulusal standartlar ve müşteri şartnameleri uyarınca, mevcut malzemeleri ekonomik olarak kullanarak daha düşük boyut, ağırlık ve maliyet ve daha yüksek işletme performansı elde etmek üzere transformatorün tüm bileşenlerinin niteliklerinin detaylı olarak hesaplanmasıdır [4].

TDO probleminin özellikleri şöyle tanımlanmaktadır: TDO problemi, karmaşık kısıtlı karma-tamsayılı bir doğrusal olmayan programlama problemidir. TDO problemi, amaç fonksiyonunun süreksiz olması nedeniyle daha da karmaşıklaşmaktadır [1].

Geleneksel transformator tasarım yöntemlerinde tasarımcılar istenen transformatorü tasarlamak için deneyim ve muhakemelerine güvenmek zorundaydılar. Transformator tasarımında yapılan ilk araştırmalar bu muhakeme gereksiminin çoğunu matematiksel ilişkiler lehine azaltmaya çalıştı.

Geçmişte yapılan araştırmalarda transformatorler için çeşitli tasarım yöntemleri geliştirilmiştir. İteratif tasarım yöntemlerinin gerektirdiği zaman alan hesaplamaları ortadan kaldırma amacıyla bilgisayar destekli tasarım teknikleri için matematiksel modeller türetilmiştir.

Transformatör tasarım optimizasyonu (TDO) karmaşık probleminin çözümü için, yapay sinir ağları, genetik algoritmalar, sürü zekâsı gibi daha çok yapay zekâ tekniklerine dayanan hem deterministik hem de deterministik olmayan metasezgisel yöntemler yaygın olarak kullanılmıştır. Seçilen optimizasyon metodu ne olursa olsun, transformatör performansı ile maliyeti arasında optimum denge kurma çok zor bir işlemdir ve optimum maliyet tasarımının tüm mekanik, ısıl ve elektriksel kısıtları karşılamasını beklemek gerçek dışıdır.

Transformatörler ve transformatör tasarım optimizasyonu alanlarında birçok iyi derleme çalışması bulunmaktadır, bunlardan aşağıda kısaca bahsedilmiştir.

Olivares-Galvan ve ark. tarafından yapılan bir çalışma [5] transformatörler konusunda 2001-2010 yılları arasında 22 dergide yayınlanan 933 makale olduğunu ortaya koymuştur.

2008 yılında yayınlanan makalelerinde Amoiralis ve ark. [6], TDO araştırma alanındaki en önemli gelişmelerin bazılarını özetlemektedir; adı geçen çalışmanın temel amacı, bu alanda yayınlanmış araştırmaların bir sentezini sağlamak ve ilgili konular üzerinde daha fazla araştırma çabalarını teşvik etmektir.

Amoiralis ve ark. [7], 2009'da bir literatür araştırması yapmış ve 420'den fazla yayınlanmış makale, 50 transformatör kitabı ve 65 standardı esas alarak son 35 yılda transformatör tasarım ve optimizasyonu alanındaki araştırma ve gelişmelerin bir özetini ortaya koymuşlardır.

Khatri ve Rahi [8], 2012'de yayınlanan makalelerinde, 126 yayınlanmış makale ve 58 transformatör kitabını esas alarak, son 38 yılda transformatör tasarım ve optimizasyonu alanındaki araştırma ve geliştirmelerin bibliyografik bir inceleme ve özetini sunmuşlardır.

2014 yılında Mehta ve Patel'in yayınladığı makalede [9], transformatör alanında geleneksel optimizasyon yöntemleri ve yapay zeka tabanlı optimizasyon teknikleri

kullanılarak yapılan araştırma ve geliştirme çalışmalarının kısa bir incelemesi verilmekte ve TDO problemi için kullanılabilecek bazı yeni doğa-esinlemeli yapay zeka teknikleri önerilmektedir.

Omorogiuwa ve Obama tarafından yapılan çalışmada [10], son 42 yılda transformatör tasarım ve optimizasyonu alanında yapay zekâ tabanlı teknikler kullanarak yapılan araştırma ve gelişmelerin bibliyografik bir incelemesi ve genel bir derlemesi 80'in üzerinde yayınlanmış makaleye dayanılarak verilmiştir.

Transformatör optimizasyonu uygulaması için kullanılabilecek çeşitli yaklaşımlar henüz yeterince araştırılmadığından konunun aktif bir araştırma alanı olarak kaldığı literatürde açıkça belirtilmektedir [4].

2.2. Ayrık Transformatör Tasarım Optimizasyonu (DTDO)

Birçok gerçek mühendislik tasarım optimizasyonu probleminde tasarım değişkenleri rastgele veya sürekli değerler alamaz. Bunun yerine, tasarım değişkenlerinin birkaçı veya tümünün pratik nedenlerle ayrık veya tamsayı değerlerden seçilmesi gerekir. Örneğin, bir borunun çapı, bir yapısal elemanın kalınlığı veya bir vidanın boyu, piyasada bulunan veya standart ebatlardan seçilmesi gerekebileceğinden ayrık tasarım değişkenleridir. Ayrıca, cıvata sayısı, çark dişli sayısı veya yay sarım sayısı gibi diğer birçok tasarım değişkeni tamsayı olmalıdır.

Bu, TDO problemi için de geçerlidir; örneğin sargı iletkenleri piyasada bulunan boyutlardan seçilmeli, çekirdek malzemesi belirli genişliklerde kesilmelidir.

Literatürde sunulan TDO yöntemleri ender olarak üretimde doğrudan uygulanabilir çözümler üretir; tasarım mühendisinin genellikle teorik çözümü pratik olarak uygulanabilir bir hale dönüştürmek için ek çaba harcaması gerekir.

Transformatör endüstrisinde bu sorunun üstesinden gelmek için büyük oyuncular tarafından da halen kullanılmakta olan bir yöntem, iki adımlı bir optimizasyon

sürecidir. Teorik tasarım optimizasyonu olan bu sürecin ilk aşamasında uygulanabilir tasarım gereksinimi hafifletilir ve bunun sonucunda birkaç iyi teorik tasarım elde edilir. Uygulanabilir tasarım optimizasyonu olan ikinci adımında, birkaç tasarım değişkeninin aralığını seçilen iyi teorik tasarım karşılıklarına daraltmak için temel oluşturmak üzere bir iyi teorik tasarım seçilir ve bu aralıklarla tanımlanan alanda standart malzeme boyutları kullanılarak kapsamlı bir arama yapılır.

Ayrık tasarım değişkenlerinin kullanımının açıkça belirtildiği TDO literatürüne ilişkin genel bir özet aşağıda verilmiştir.

Ayrık tasarım değişkenlerinin kullanıldığı literatürün çoğunda çoklu tasarım yöntemi (MDM) kullanılır. Bu yöntemde çok sayıda alternatif tasarım oluşturmak üzere tasarım değişkenlerine birçok alternatif değerler atanır ve sonuçta tüm problem kısıtlarını karşılayan minimum üretim maliyetli tasarımı seçilir [1]. Bu tür literatüre örnekler [1, 4, 11-13]'de yer almaktadır.

Azizian ve ark. [14], dökme reçine kuru tip transformatörleri hepsi ayrık olmak üzere 18 tasarım değişkeni kullanarak optimize etmek için Genetik Algoritma (GA), Yapay Arı Kolonisi (ABC) ve Parçacık Sürü Optimizasyonu (PSO) algoritmalarını kullanmıştır.

Zhang ve ark. makalelerinde [15] 50 kVA'lık bir transformatörün toplam sahiplik maliyetini (TOC) optimize etmek için kromozomların ikili kodlandığı ve dolayısıyla tasarım değişkenleri ayrık olan geliştirilmiş uyarlamalı genetik algoritma kullanmıştır.

Tamilselvi ve Baskar makalelerinde [16] 400 kVA transformatörün optimizasyonu için dört farklı amaç fonksiyonunu minimize etmek üzere kovaryans matris uyum gelişimi stratejisini (CMA-ES) uygulamışlardır. Kullanılan tüm tasarım değişkenleri tamsayı veya ayrıktır; bununla birlikte, AG ve YG sargılarının akım yoğunlukları için ayrık değişkenler kullanıldığından bu sargılar için seçilen iletkenler büyük olasılıkla standart boyutlu olmayacaktır.

BÖLÜM 3. METASEZGİSEL ALGORİTMALAR

Bu tez kapsamında TDO probleminin çözümü için yaklaşık 20 kadar güncel metasezgisel algoritma incelenmiş ve bunlar arasında en iyi performans göstererek algoritma kıyaslama kapsamına dâhil edilen Yapay Arı Kolonisi (ABC), Geri-İzleme Arama Optimizasyon (BSA), Rekabetçi-Uyarlamalı Diferansiyel Gelişim (b6e6rl), Guguk Kuşu Arama (CS), ve Çiçek Tozlaşma (FPA) algoritmaları aşağıda kısaca açıklanmıştır.

3.1. Yapay Arı Kolonisi Algoritması (ABC)

3.1.1. Kısıtsız optimizasyon problemleri için geliştirilen ABC algoritması

Arıların beslenme davranışlarından esinlenerek geliştirilen ABC algoritmasında [17, 18] yapay arı kolonisi görevli arılar, gözlemci arılar ve kâşif arılar olmak üzere üç gruptan oluşur. Koloninin ilk yarısında görevli arılar, ikinci yarısında da gözlemciler yer alır. Her besin kaynağı için yalnızca bir tane görevli arı vardır. Başka bir deyişle, görevli arıların sayısı, kovan çevresindeki besin kaynakları sayısına eşittir. Arılar tarafından besin kaynağı terkedilen görevli arı kâşif arı olur.

ABC algoritmasında bir gıda kaynağının konumu, optimizasyon probleminin olası bir çözümünü temsil eder ve bir gıda kaynağının nektar miktarı, ilişkili çözümün kalitesine (uygunluğuna) karşılık gelir. Görevli arıların veya gözlemci arıların sayısı popülasyondaki çözüm sayısına eşittir.

İlk aşamada, ABC rasgele dağılımla SN çözümden (gıda kaynağı konumları) oluşan bir başlangıç popülasyonunu üretir. Her x_i çözümü ($i = 1, 2, \dots, SN$) D -boyutlu bir vektördür. Burada D problem boyutu veya optimize edilecek parametre sayısıdır.

Başlatma işleminden sonra konumlar (çözümler) popülasyonuna görevli arılar, gözlemci arılar ve kâşif arıların arama süreçlerinin $C = 1, 2, \dots, MCN$ tekrarlanan döngüleri uygulanır. Görevli bir arı, yerel bilgilere (görsel bilgi) bağlı olarak belleğindeki konum (çözüm) üzerinde değişiklik yapar ve yeni kaynağın nektar miktarını (uygunluk değeri) test eder (yeni çözüm). Yeni kaynağın nektar miktarı bir öncekinden daha fazla ise, arı yeni konumu belleğine alır ve eskisini unuttur. Aksi takdirde bir önceki konumu belleğinde tutar. Tüm görevli arılar arama işlemini tamamladıktan sonra, kovandaki dans alanında gözlemci arılarla gıda kaynaklarının nektar ve konum bilgilerini paylaşırlar. Bir gözlemci arı, tüm görevli arılardan alınan nektar bilgilerini değerlendirir ve nektar miktarı olasılığına göre bir gıda kaynağı seçer. Görevli arılarda olduğu gibi, belleğindeki konum üzerinde bir değişiklik yapar ve aday kaynağının nektar miktarını kontrol eder. Yeni kaynağın nektar miktarı bir öncekinden daha fazla ise arı yeni konumu belleğine alır ve eskisini unuttur.

Her bir görevli arının gıda kaynağının gözcü arılar tarafından seçilme olasılığı aşağıdaki şekilde hesaplanır.

$$q_i = \frac{uygunluk_i}{\sum_{j=1}^{SN} uygunluk_j} \quad (3.1)$$

Burada q_i gözcü arıların i gıda kaynağını seçme olasılığı, $uygunluk_i$ i konumundaki gıda kaynağının nektar miktarı ile orantılı olan çözüm i 'nin uygunluk değeri ve SN görevli arı sayısına eşit olan toplam gıda kaynağı sayısıdır. $uygunluk_i$ aşağıdaki şekilde hesaplanır.

$$uygunluk_i = \begin{cases} 1/(1 + f_i), & f_i \geq 0 \\ 1 + abs(f_i), & f_i < 0 \end{cases} \quad (3.2)$$

Burada f_i , x_i gıda kaynağı (çözüm) için hesaplanan amaç fonksiyonu değeridir.

Bellekteki eskisinden yeni bir gıda konumu oluşturmak için aşağıdaki ifade kullanılır.

$$v_{i,j} = x_{i,j} + \phi_{i,j}(x_{i,j} - x_{k,j}) \quad (3.3)$$

Burada $k \in \{1, 2, \dots, SN\}$ ve $j \in \{1, 2, \dots, D\}$ rastgele seçilen indekslerdir, k 'nin i 'den farklı olması gerekir. $\phi_{i,j}$ $[-1, 1]$ aralığında düzgün dağılımlı rastgele üretilmiş bir sayıdır; $x_{i,j}$ etrafındaki komşu gıda kaynaklarının üretimini kontrol eder ve iki besin konumunun bir arı tarafından görsel olarak karşılaştırılmasını ifade eder.

Denklem 3.3'de görülebileceği gibi, $x_{i,j}$ ve $x_{k,j}$ parametreleri arasındaki fark azaldıkça, $x_{i,j}$ konumundaki pertürbasyon (sapma) da azalır. Böylece, arama süreci arama uzayında optimum çözüme yaklaşıırken, adım uzunluğu da uyarlanabilir bir şekilde azaltılır.

Bu işlemle üretilen bir parametre değeri önceden belirlenmiş sınırını aşarsa, parametre değeri sınır değerine ayarlanır.

Arılar tarafından nektarı tükendiği için terk edilen gıda kaynağı kâşif arılar tarafından yeni bir gıda kaynağı ile değiştirilir. ABC'de bu, terkedilenin yerine rasgele bir konum üretilerek simüle edilir. ABC'de bir konumun daha önceden belirlenmiş bir döngü sayısı ile daha da iyileştirilememesi durumunda, o yiyecek kaynağının terk edildiği varsayılır. Önceden belirlenmiş döngü sayısının değeri, ABC algoritmasının önemli bir kontrol parametresidir ve buna terk etme için *limit* denir. Terk edilmiş olan kaynak x_i ve $j \in \{1, 2, \dots, D\}$ olduğunda, kâşif arı aşağıdaki denklem uyarınca x_i ile değiştirilecek yeni bir gıda kaynağı keşfeder.

$$x_i^j = x_{min}^j + rand(0,1)(x_{max}^j - x_{min}^j) \quad (3.4)$$

Her bir aday gıda konumu $v_{i,j}$ üretildikten ve yapay arı tarafından değerlendirildikten sonra, performansı eskisinininki ile karşılaştırılır. Yeni gıda kaynağının eski kaynağına kıyasla eşit ya da daha iyi bir nektarı varsa, bellekteki eskisi ile değiştirilir. Aksi halde eskisi bellekte saklanır. Başka bir deyişle, eski ve aday arasındaki seçim işlemi olarak açgözlü seçim mekanizması kullanılır.

ABC algoritmasının kısıtsız optimizasyon versiyonunun üç kontrol parametresi vardır: Görevli veya gözcü arıların sayısına eşit olan gıda kaynağı sayısı (SN), *limit* değeri, ve maksimum döngü sayısı (MCN).

3.1.2. Kısıtlı optimizasyon problemleri için geliştirilen ABC algoritması

Karaboğa ve Baştürk [19] tarafından ABC algoritmasının kısıtlı optimizasyon problemlerinin çözümü için geliştirilen versiyonunda bir önceki bölümde açıklanan açgözlü seçim yöntemi yerine Deb'in kısıt ele alma yöntemi kullanılmıştır. Deb'in yöntemi bir seferde iki çözümün karşılaştırıldığı bir turnuva seçimi operatörünü kullanır ve aşağıdaki ölçütler her zaman uygulanır: 1) herhangi bir uygulanabilir çözüm herhangi bir uygulanamaz çözüm yerine tercih edilir, 2) uygulanabilir iki çözüm arasından daha iyi amaç fonksiyonu değerine sahip olan tercih edilir, 3) uygulanamaz iki çözüm arasında daha küçük kısıtlama ihlaline sahip olan tercih edilir.

ABC algoritmasında ilk çözümün uygulanabilir olması gerekli değildir. Açgözlü seçim yerine Deb'in kuralları kullanıldığı için, algoritmanın yapısı çözümleri uygulanabilir bölgeye yöneltir. Algoritmanın kâşif arı süreci, yeni ve muhtemelen uygulanabilir olmayan bireylerin popülasyonda olmasına izin veren bir çeşitlilik mekanizması sağlar.

Bellekteki eskisinden bir aday gıda konumu üretmek için kısıtlı problemler için uyarlanmış ABC algoritması aşağıdaki ifadeyi kullanır.

$$v_j = \begin{cases} x_{i,j} + \phi_{i,j}(x_{i,j} - x_{k,j}), & R_j < MR \text{ ise} \\ x_{i,j}, & \text{aksi takdirde} \end{cases} \quad (3.5)$$

Burada $j \in \{1, 2, \dots, D\}$ 'dir ve $k \in \{1, 2, \dots, SN\}$ rastgele seçilen bir indekstir; k 'nin i 'den farklı olması gerekir. R_j $[0, 1]$ aralığında rastgele üretilmiş gerçek bir sayıdır. Değişim oranı MR , $x_{i,j}$ parametresinin değiştirilip değiştirilmeyeceğini kontrol eden bir algoritma kontrol parametresidir.

Kısıtlı ABC algoritmasında her bir görevli arının gıda kaynağının gözcü arılar tarafından seçilme olasılığı aşağıdaki şekilde hesaplanır.

$$p_i = \begin{cases} 0.5 + 0.5 \times q_i, & \text{çözüm uygulanabilir ise} \\ 1 - 0.5 \times r_i, & \text{çözüm uygulanabilir değilse} \end{cases} \quad (3.6)$$

Burada $ceza_i$, i gıda kaynağında sağlanamayan kısıtlar için hesaplanan ceza değerlerinin toplamı olmak üzere, r_i aşağıdaki şekilde hesaplanır.

$$r_i = \frac{ceza_i}{\sum_{j=1}^{SN} ceza_j} \quad (3.7)$$

ABC algoritmasının kısıtlı optimizasyon problemleri için geliştirilen versiyonunda, yeni gıda kaynaklarını rastgele keşfetmek için önceden belirlenmiş bir döngü periyodunda yapay kâşifler üretilir. Bu periyod algoritmanın kâşif üretim periyodu (*SPP*) olarak adlandırılan başka bir kontrol parametresidir. Her bir *SPP* döngüsünde, terk edilmiş bir gıda kaynağı olup olmadığı kontrol edilir ve varsa kâşif üretim süreci uygulanır.

ABC algoritmasının kısıtlı optimizasyon versiyonunun iki ilave kontrol parametresi vardır: Değişim oranı MR ve kâşif üretim periyodu (*SPP*). Önerilen kontrol parametresi değerleri $limit = 10 \times SN \times D$, $SPP = SN \times D$, ve $MR = 0.9$ 'dur [20].

Bu çalışmada ABC algoritmasının kısıtlı optimizasyon versiyonunun [20]'de verilen Delphi kodu MATLAB'a çevirilerek kullanılmıştır.

ABC algoritmasının sözde kodu Şekil 3.1.'de verilmiştir.

Amaç fonksiyonu min veya max $f(\mathbf{x})$, $\mathbf{x} = (x_1, x_2, \dots, x_D)$ ve kısıtları tanımla

limit, MR, SPP ve **MCN** kontrol parametrelerini tanımla

$\mathbf{x}_{i,j}$, $i = 1, \dots, SN$, $j = 1, \dots, D$ çözümlerinin rastgele başlangıç değerlerini oluştur ve

Denklem 3.2 uyarınca uygunluk değerlerini hesapla

döngü = 1

while (döngü < **MCN**)

Görevli arılar için Denklem 3.5 uyarınca yeni $\mathbf{v}_{i,j}$ çözümlerini oluştur ve uygunluk değerlerini hesapla

Deb yönetimine göre seçim sürecini uygula

Denklem 3.6 uyarınca $\mathbf{x}_{i,j}$ çözümleri için $\mathbf{p}_{i,j}$ olasılık değerlerini hesapla

Gözcü arılar için $\mathbf{x}_{i,j}$ çözümlerinden $\mathbf{p}_{i,j}$ olasılık değerlerine bağlı olarak $\mathbf{v}_{i,j}$ yeni çözümlerini oluştur ve uygunluk değerlerini hesapla

Deb yönetimine göre seçim sürecini uygula

Kaşif arı için varsa terkedilen gıda kaynağını belirle ve bunu Denklem 3.4 uyarınca rastgele oluşturulan bir $\mathbf{x}_{i,j}$ çözümü ile değiştir

En iyi çözümü sakla

döngü = döngü + 1

end while

Şekil 3.1. ABC algoritması sözde kodu

3.2. Rekabetçi-Uyarlamalı Diferansiyel Gelişim Algoritması (b6e6rl)

b6e6rl Josef Tvrdik tarafından geliştirilen popülasyon tabanlı bir diferansiyel gelişim (DE) algoritmasıdır [21].

DE algoritmasında aynı N popülasyon büyüklüğünde P ve Q nesilleri kullanılır. Problem boyutu D ile gösterilir. Popülasyon, iteratif olarak seleksiyon, mutasyon ve çaprazlama operatörlerini kullanarak gelişir. Her iterasyon bir gelişimsel nesile karşı gelir. Birbirini takip eden iki nesil P ve Q olarak gösterilsin. Gelişimsel operatörlerin P eski nesiline uygulanması sonucu Q yeni nesili oluşur. Q yeni nesil işlemleri

tamamlandığında Q eski nesil, P de yeni nesil olur. DE algoritmasının temel sözde kodu Şekil 3.2.'de verilmiştir.

```

Amaç fonksiyonu min veya max  $f(\mathbf{x})$ ,  $\mathbf{x} = (x_1, x_2, \dots, x_D)$  ve kısıtları tanımla
Kontrol parametrelerini tanımla
 $\mathbf{P} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$  başlangıç popülasyonunu rastgele oluştur ve uygunluk
değerlerini hesapla
döngü = 1
while (döngü < MaxCycles)
     $\mathbf{u}$  mutant vektörünü hesapla
     $\mathbf{u}$  ve  $\mathbf{x}_i$  vektörlerini çaprazlayarak  $\mathbf{y}$  vektörünü oluştur
    if  $f(\mathbf{y}) \leq f(\mathbf{x}_i)$  then
         $\mathbf{y}$  vektörünü yeni nesil  $\mathbf{Q}$ 'ya yerleştir
    else
         $\mathbf{x}_i$  vektörünü yeni nesil  $\mathbf{Q}$ 'ya yerleştir
    end if
     $\mathbf{P} := \mathbf{Q}$ 
    döngü = döngü + 1
end while

```

Şekil 3.2. DE algoritması sözde kodu

Yeni bir \mathbf{y} deneme noktası mutasyon ve çaprazlama kullanılarak oluşturulur. Mutasyon ve çaprazlamanın birçok çeşiti vardır. Bunlardan en yaygın kullanılanı olan DE/rand/1 yönteminde \mathbf{u} mutant noktası iki noktanın ağırlıklı farkının toplanması ile bulunur.

$$\mathbf{u} = \mathbf{r}_1 + F(\mathbf{r}_2 - \mathbf{r}_3) \quad (3.8)$$

Burada \mathbf{r}_1 , \mathbf{r}_2 ve \mathbf{r}_3 P 'den rastgele seçilen ve o anki \mathbf{x}_i 'den farklı olan üç ayrı noktadır; F sıfırdan büyük bir girdi parametresidir.

Rassal yerelleştirme olarak adlandırılan ve DE/randrl/1 ile gösterilen yöntemde \mathbf{u} mutant noktası aşağıdaki şekilde bulunur.

$$\mathbf{u} = \mathbf{s}_1 + F(\mathbf{s}_2 - \mathbf{s}_3) \quad (3.9)$$

Burada \mathbf{s}_1 noktası \mathbf{r}_1 , \mathbf{r}_2 ve \mathbf{r}_3 içindeki en iyi nokta olup \mathbf{s}_2 ve \mathbf{s}_3 , \mathbf{r}_1 , \mathbf{r}_2 ve \mathbf{r}_3 'ün diğer iki noktasıdır.

\mathbf{y} deneme noktasının $y_j, j=1, 2, \dots, D$ elemanları o anki \mathbf{x}_i noktası ile \mathbf{u} mutant noktasının çaprazlanması ile oluşturulur. Yaygın olarak kullanılan iki tür çaprazlama vardır. Bunlardan *ikili* (binomial) çaprazlama aşağıdaki şekilde ifade edilir.

$$y_j = \begin{cases} u_j, & \text{if } U_j \leq CR \quad \text{or} \quad j = l \\ x_{i,j}, & \text{if } U_j > CR \quad \text{and} \quad j \neq l \end{cases} \quad (3.10)$$

Burada $l \in \{1, 2, \dots, D\}$ içinden rastgele seçilen bir tamsayı; $U_1, U_2, \dots, U_D \in [0,1]$ aralığında düzgün dağılmış rastgele bağımsız değişkenler, ve $CR \in [0,1]$ çaprazlama ile yer değiştirecek eleman sayısını etkileyen bir girdi parametresidir. Denklem 3.10'da verilen kural, $CR = 0$ olsa dahi \mathbf{x}_i vektörünün en az bir elemanının değiştirilmesini sağlar. Denklem 3.10 uyarınca ikili (binomial) çaprazlama uygulayan DE varyantları literatürde DE/ / /bin şeklinde gösterilir.

Üssel çaprazlama (DE/ / /exp) için çaprazlamanın başlangıç konumu $\{1, 2, \dots, D\}$ içinden rastgele seçilir ve \mathbf{u} mutant vektöründen birbirini takip eden L eleman alınır. $1, 2, \dots, L, L \leq D$ dizisindeki k elemanının değiştirilme olasılığı k arttıkça üssel olarak azalır.

Çaprazlama olasılığı p_m \mathbf{x}_i vektörünün \mathbf{u} mutant vektörü ile değiştirilecek eleman sayısını belirler. p_m ile CR kontrol parametresi arasındaki ilişki aşağıdaki gibidir.

$$CR^D - Dp_m CR + Dp_m - 1 = 0 \quad (3.11)$$

DE algoritmasının etkinliği F ve CR kontrol parametrelerinin seçimine karşı çok duyarlıdır. Bu parametreler bir problemde diğerine farklılıklar gösterdiği için DE algoritmasının birçok özuyarlamalı (self-adaptive) varyantları geliştirilmiştir.

DE algoritmasına rekabet uygulama yoluyla kontrol parametrelerinin seçimi özuyarlanabilir hale getirilebilir. Bu yöntemde, F ve CR kontrol parametrelerinin toplam H kombizonu arasından q_h ; $h = 1, 2, \dots, H$ olasılıkla rastgele seçim yapılır. Bu olasılıklar arama sürecinin önceki adımlarında seçilmiş olan parametrelerin başarı oranına göre değişir. h kombinasyonu, $f(\mathbf{y}) \leq f(\mathbf{x}_i)$ olduğu bir \mathbf{y} deneme noktası oluşturuyorsa başarılıdır. h kombinasyonunun o anki başarı sayısı n_h ise q_h olasılığı aşağıdaki şekilde hesaplanır.

$$q_h = \frac{n_h + n_0}{\sum_{j=1}^H (n_j + n_0)} \quad (3.12)$$

Burada $n_0 > 0$ bir sabittir. Bu şekilde rekabet, çözülen probleme uygun kontrol parametrelerini seçmek için özuyarlamalı bir mekanizma sağlar. Rekabet mekanizmasını kullanan DE varyantları CDE kısaltması ile gösterilir.

CDE'nin b6r6rl varyantında randrl mutasyon ile ikili ve/veya üssel çaprazlama yöntemleri, başka bir deyişle DE/randrl/1/bin ve DE/randrl/1/exp DE stratejileri kullanılır; bu stratejilerden her biri 6 farklı F ve CR kontrol parametreleri kombinasyonunu içerir. b6e6rl algoritmasının rekabet eden DE stratejileri ve parametre değerleri Tablo 3.1.'de gösterilmiştir.

p_1 , p_2 ve p_3 parametre değerleri $(1/D, 1)$ açık aralığında eşmesafeli olarak seçilir. p_2 $(1/D, 1)$ 'in ortasında, p_1 $(1/D, p_2)$ 'nin ortasında, ve p_3 $(p_2, 1)$ 'in ortasındadır. p_i , $i = 1, 2, 3$, değerleri algoritmanın başlangıcında problemin boyutuna göre, bunlara karşı gelen CR_i değerleri de Denklem 3.8'in köklerine göre bulunur.

Tablo 3.1. b6e6rl algoritması DE stratejileri ve kontrol parametre değerleri

h	Mutasyon	Çaprazlama	F	CR	p _m
1			0.5	0	
2			0.5	0.5	
3	randrl/1	ikili	0.5	1	
4			0.8	0	
5			0.8	0.5	
6			0.8	1	
7			0.5	CR ₁	p ₁
8			0.5	CR ₂	p ₂
9	randrl/1	üssel	0.5	CR ₃	p ₃
10			0.8	CR ₁	p ₁
11			0.8	CR ₂	p ₂
12			0.8	CR ₃	p ₃

b6e6rl algoritmasının kullanıcı tarafından belirlenmesi gereken iki rekabet kontrol parametresi vardır, bunlar için de $n_0 = 2$ ve $\delta = 1 / (5H)$ değerlerinin seçilmesi önerilmektedir [21].

Bu çalışmada b6e6rl algoritmasının [22]'de verilen MATLAB kodu kullanılmıştır.

3.3. Geri-İzleme Arama Optimizasyon Algoritması (BSA)

Pınar Çivicioğlu [23] tarafından geliştirilen BSA bijektif formlu, popülasyon tabanlı, gelişimsel ve iteratif bir arama algoritmasıdır. Bijektif arama her bir popülasyon elemanının, kendisinden farklı bir diğer popülasyon elemanına doğru gelişmesi olarak açıklanır. Popülasyon tabanlı arama algoritmalarında, yapay-popülasyon her iterasyonda biraz daha verimli bir alana doğru göç etme eğilimindedir. BSA ileri iterasyonlarda rassal olarak daha önceki iterasyonlarda elde ettiği yapay-popülasyon formunu hatırlayabilir ve mevcut popülasyonu hatırlanan popülasyona doğru geliştirebilir. Mevcut iterasyondaki yapay-popülasyonun, yapay-popülasyon

elemanları arasındaki çeşitliliğin henüz azalmadığı daha erken iterasyonlardaki yapay-popülasyona doğru rassal gelişim eğilimi BSA'nın yerel çözümlerden kaçabilmesini sağlamaktadır.

BSA temelde Başlangıç, Seleksiyon-I, Mutasyon, Çaprazlama Ve Seleksiyon-II olmak üzere beş kısımdan oluşur.

3.3.1. Başlama

BSA'da popülasyon (Pop) aşağıdaki şekilde oluşturulur.

$$Pop_{i,j} \sim U(low_j, up_j) \quad (3.13)$$

Denklem 3.13'de $i=1, 2, 3, \dots, N$ ve $j=1, 2, 3, \dots, D$ olarak tanımlanır ve N ile D sırasıyla popülasyon büyüklüğü ve problem boyutudur. U ise uniform dağılıma uygun rastgele sayı üretim fonksiyonudur.

3.3.2. Seleksiyon-I

BSA'nın Seleksiyon-I kısmında, arama doğrultusunu hesaplamak için kullanılacak eski popülasyon ($oldPop$) belirlenir. Başlama aşamasında eski popülasyon aşağıdaki eşitlikle hesaplanır.

$$if \ a < b, \ oldPop := Pop|a, b \sim U(0,1) \quad (3.14)$$

Denklem 3.14'te $:=$ güncelleme operatörüdür. Denklemden de anlaşılacağı üzere, BSA bir belleğe sahiptir. BSA daha önceki nesilden rastgele seçilmiş bir popülasyonu eski popülasyon ($oldPop$) olarak atar ve bu eski popülasyonu değişene kadar belleğinde tutar. Ardından Denklem 3.15 kullanılarak $oldPop$ içindeki bireylerin sırası rastgele değiştirilir.

$$oldPop := RandShuff(oldPop) \quad (3.15)$$

Burada *RandShuff* rastgele karıştırma fonksiyonudur.

3.3.3. Mutasyon

BSA'da mutasyon işlemi Denklem 3.13 kullanılarak gerçekleştirilir.

$$MutantPop = P + F \cdot (oldPop - Pop) \quad (3.16)$$

Denklem 3.16'da F arama doğrultusu matrisinin genliğini kontrol eder. Buradaki amaç önceki nesildeki deneyimlerin işleme kısmi olarak dahil edilmesidir.

3.3.4. Çaprazlama

BSA'nın çaprazlama aşamasında deneme popülasyonunun (T_{pop}) son hali üretilir. Çaprazlama süreci iki aşamadan oluşur. Bunlardan ilki ikili sayı sisteminde tam değerli ve $N \cdot D$ boyutunda bir matris (*map*) üretmektir. Bu matris ile T_{pop} bireyleri üzerinde değişiklik yapıp yapılmayacağına karar verilir. $map_{i,j}=1$ olması halinde $T_{pop}^{i,j} := P_{i,j}$ olacak şekilde güncellenir.

3.3.5. Seleksiyon-II

Seleksiyon-II aşamasında P_i 'lere göre daha iyi uygunluk değerlerine sahip T_i 'ler; P_i 'leri güncellemek için kullanılırlar. Eğer P 'nin en iyi değeri (P_{best}), küresel minimum değerinden daha iyiye, çözüm değeri (küresel minimum) P_{best} ile güncellenir.

BSA algoritmasının karışım oranı *mixrate* ve ölçekleme faktörü F olmak üzere iki kontrol parametresi vardır; bunların $mixrate = 1$ ve $F = 3 \cdot randn \sim (0,1)$ olarak seçilmesi önerilmektedir, burada *randn* standart normal dağılıma uygun 0 ile 1 arasında rastgele sayı üreten fonksiyondur [23].

Bu çalışmada BSA algoritmasının [24]'te verilen MATLAB kodu kullanılmıştır.

3.4. Guguk Kuşu Arama Algoritması (CS)

Guguk kuşları ilginç agresif üreme stratejilerine sahiptir; örneğin yumurtalarını diğer kuşların yuvalarına koyarlar ve kendi yumurtalarının kuluçkadan çıkma olasılığını artırmak için yuvada bulunan diğer yumurtaları atabilirler. Yuva sahibi kuşun yuvasında kendisine ait olmayan yumurtalar olduğunu fark etmesi durumunda ya yabancı yumurtaları yuvadan atar veya kendi yuvasını terk edip başka bir yere yeni bir yuva kurar.

Guguk Kuşu Arama algoritması (CS) Xin-She Yang ve Suash Deb tarafından bazı guguk kuşu türlerinin kuluçka asalaklığı özelliklerinden esinlenerek geliştirilmiştir [25].

CS algoritması üç basitleştirilmiş temel kurala dayanır:

- a. Her guguk kuşu bir defasında bir yumurta yapar ve bunu rasgele seçilmiş bir yuvaya bırakır.
- b. Yüksek kaliteli yumurta içeren en iyi yuvalar sonraki nesillere taşınır.
- c. Yuva sayısı sabittir ve yuvanın sahibi olan kuş yuvasına guguk kuşu tarafından bırakılan yumurtayı $p_a \in (0, 1)$ olasılıkla fark edebilir. Eğer yumurtayı tanır, yumurtayı yuvadan aşağı atabilir veya yuvayı terk ederek kendisine yeni bir yuva kurabilir.

Yuvadaki her bir yumurta bir çözüme karşı gelir. Algoritma daha da basitleştirilip her yuvada sadece bir yumurta bulunmasına izin verilmektedir. Böylece yumurta, yuva veya guguk kuşu arasında hiçbir ayırım kalmamaktadır, çünkü her yuva bir yumurtayı ve bu da bir guguk kuşunu temsil etmektedir.

CS algoritması küresel rassal yürüyüş ve yerel rassal yürüyüş yaklaşımlarını bir arada dengeleyerek kullanır, bu rassal yürüyüşler de p_a geçiş parametresi tarafından kontrol edilir. Yerel rassal yürüyüş aşağıdaki şekilde ifade edilebilir.

$$x_i^{t+1} = x_i^t + \alpha s \otimes H(p_a - \epsilon) \otimes (x_j^t - x_k^t) \quad (3.17)$$

Burada x_j^t ve x_k^t rassal permütasyonla rastgele seçilen iki farklı çözüm, $H(u)$ Heaviside fonksiyonu, ϵ düzgün dağılımdan gelen rassal bir sayı, s ise adım uzunluğudur. \otimes iki vektörün elemanlarının birbiriyle çarpımıdır.

Küresel rassal yürüyüş ise aşağıda kuralları verilen Lévy uçuşları kullanılarak gerçekleştirilmektedir.

$$x_i^{t+1} = x_i^t + \alpha L(s, \lambda) \quad (3.18)$$

$$L(s, \lambda) = \frac{\lambda \Gamma(\lambda) \sin\left(\frac{\pi\lambda}{2}\right)}{\pi} \frac{1}{s^{1+\lambda}}, \quad (s \gg s_0 > 0) \quad (3.19)$$

Burada $\alpha > 0$ olmak üzere problemin boyutuna bağlı olarak Lévy uçuşunu ölçeklendirmektedir. Denklem 3.18 temelde rassal yürüyüşün stokastik olarak ifadesidir. Genelde rassal yürüyüş bir sonraki konumu şu anki konuma ve geçiş olasılığına bağlı olan bir Markov zinciridir. Ancak, sürecin yerel bir optimumda takılıp kalmaması için yeni çözümlerin o anki en iyi çözümden yeterli seviyede uzaklıkta olması gerekir.

CS algoritmasının küresel yakınsama gereksinimlerini karşılayabileceği ve dolayısıyla garantili küresel yakınsama özelliğine sahip olduğu matematiksel olarak kanıtlanmıştır [25, 26].

CS algoritmasının temelde tek bir kontrol parametresi vardır, p_a geçiş olasılığı, ve bunun $p_a = 0.25$ olarak seçilmesi önerilmektedir [25].

Bu çalışmada CS algoritmasının kısıtlı versiyonunun [27]'de verilen MATLAB kodu kullanılmıştır; bu versiyonda amaç fonksiyonu ve kısıtlar her itersyonda iki kez hesaplanmaktadır.

CS algoritmasının sözde kodu Şekil 3.3.'te verilmiştir.

```

Amaç fonksiyonu min veya max  $f(\mathbf{x})$ ,  $\mathbf{x} = (x_1, x_2, \dots, x_D)$  ve kısıtları tanımla
Geçiş olasılığı  $p_a \in [0, 1]$  ve MaxGen kontrol parametrelerini tanımla
n yuvalı başlangıç popülasyonunu rastgele çözümlerle oluştur ve bireylerin
    uygunluk değerlerini hesapla
döngü = 1
while (döngü < MaxGen)
    Lévy uçuşu ile rastgele bir  $x_i$  guduk kuşu seç
    Bu guduk kuşunun  $F_i$  uygunluk değerini hesapla
    n yuva arasından bir j yuvasını rastgele seç
    if ( $F_i < F_j$ ) then
        j'yi yeni çözümle değiştir
    end if
     $p_a$  oranında kötü yuvayı terk et ve Lévy uçuşları kullanarak yeni yerlerde
        yeni yuvalar kur
    En iyi çözümleri sakla
    Çözümleri sırala ve o anki en iyi çözümü bul
    döngü = döngü + 1
end while

```

Şekil 3.3. CS algoritması sözde kodu

3.5. Çiçek Tozlaşma Algoritması (FPA)

Çiçekli bitkilerin tozlaşma sürecinden esinlenen Çiçek Tozlaşma Algoritması Xin-She Yang [25] tarafından geliştirilmiştir; amacı en güçlü olanın hayatta kalması ve bitki türlerinin en uygun reproduksiyonu olan bir optimizasyon süreci olarak düşünülebilir.

Tozlaşma *abiyotik* veya *biyotik* olabilir; tüm bitkilerin % 90'ı polenlerin böcekler ve diğer hayvanlar gibi tozlaştırıcılar tarafından taşındığı biyotik tozlaşma kullanmaktadır. Diğer % 10'u da rüzgâr ve su gibi polen taşıma mekanizmalarını

kullanan abiyotik tozlaşma yöntemini kullanmaktadır. *Çiçek bağlılığı*, belirli tozlaştırıcıların sadece belirli çiçekli bitki türlerini ziyaret etme eğilimidir.

Tozlaşma, *çapraz tozlaşma* veya *kendi kendine tozlaşma* yoluyla olabilir. Çapraz tozlaşma, farklı bir bitkiden polen taşınmasıdır; buna karşın, kendi kendine tozlaşma, aynı bitki çiçeğinden polen aktarımıdır. Polenlerin böcekler ve diğer hayvanlar tarafından farklı bitkiler arasında taşınması olan biyotik çapraz tozlaşma, uzun bir mesafede meydana gelebilir ve dolayısıyla küresel tozlaşma olarak düşünülebilir.

Tozlaşma işleminin yukarıdaki özellikleri, aşağıdaki dört kuralla idealleştirilmiştir:

- a. Biyotik ve çapraz tozlaşma, polen taşıyan tozlaştırıcıların uzun adımlar atlamak veya uçmak suretiyle Lévy dağılım kurallarına uyduğu küresel tozlaşma olarak kabul edilir.
- b. Abiyotik ve kendi kendine tozlaşma yerel tozlaşma olarak kabul edilir.
- c. Çiçek bağlılığı, ilgili iki çiçeğin benzerliği ile orantılı olan üreme olasılığı olarak düşünülür.
- d. Yerel tozlaşma ve küresel tozlaşma arasındaki etkileşim, yerel tozlaşmaya hafifçe eğilimli bir $p \in [0, 1]$ geçiş olasılığı ile kontrol edilir.

Kolaylık olması açısından, her bitkinin tek bir polen gameti üreten bir çiçeği olduğu varsayılır. Bu basitlik nedeniyle, mevcut optimizasyon probleminde bir x_i çözümü, bir çiçek ve/veya bir polen gamete eşittir.

FPA'da, küresel ve yerel tozlaşma olmak üzere iki temel adım vardır. Küresel tozlaşma adımında birinci ve üçüncü kurallar matematiksel olarak aşağıdaki şekilde ifade edilir.

$$x_i^{t+1} = x_i^t + L(x_i^t - f_{best}) \quad (3.20)$$

Burada x_i^t iterasyon t 'deki polen (veya çiçek) i veya çözüm vektörü x_i , f_{best} hâlihazırdaki en iyi çözüm, L ise bir adım boyutudur ve bir Lévy dağılımından alınmıştır.

İkinci kural üçüncü kuralla birlikte yerel tozlaşma için kullanılır ve matematiksel olarak aşağıdaki şekilde ifade edilir.

$$x_i^{t+1} = x_i^t + \varepsilon(x_j^t - x_k^t) \quad (3.21)$$

Burada x_j^t ve x_k^t aynı bitki türlerinin farklı çiçeklerinden alınan polenlerdir. ε , $[0,1]$ aralığında düzgün bir dağılımdan elde edilir.

Dördüncü kurala göre, iterasyonlarda optimizasyon sürecini kontrol edecek olan tozlaşma türünü seçmek için geçiş olasılığı p kullanılır.

FPA algoritmasının temelde tek bir kontrol parametresi vardır, p geçiş olasılığı, ve bunun $p = 0.8$ olarak seçilmesi önerilmektedir [25].

Bu çalışmada FPA algoritmasının kısıtlı versiyonunun [28]'de verilen MATLAB kodu kullanılmıştır.

FPA algoritmasının sözde kodu Şekil 3.4.'te verilmiştir.

```

Amaç fonksiyonu min veya max  $f(\mathbf{x})$ ,  $\mathbf{x} = (x_1, x_2, \dots, x_D)$  ve kısıtları tanımla
Geçiş olasılığı  $p \in [0, 1]$  ve MaxGen kontrol parametrelerini tanımla
n çiçekli başlangıç popülasyonunu rastgele çözümlerle oluştur
Başlangıç popülasyonu bireylerinin uygunluk değerlerini hesapla ve  $f_{best}$  en iyi
    çözümünü bul
döngü = 1
while (döngü < MaxGen)
    for  $i = 1 : n$  (popülasyondaki tüm çiçekler)
        if  $rand < p$  then
            Lévy dağılımına uyan  $D$ -boyutlu bir  $L$  adım vektörü çiz
            Denklem 3.20 uyarınca küresel tozlaşma uygula
        else
             $[0, 1]$  aralığında düzgün bir dağılımdan bir  $\varepsilon$  elde et
            Tüm çözümler arasından  $j$  ve  $k$  çözümlerini rastgele seç
            Denklem 3.21 uyarınca yerel tozlaşma uygula
        end if
         $x_i$  çözümünün uygunluk değerini hesapla
        Yeni çözümler daha iyi ise popülasyonu güncelle
    end for
    Yeni  $f_{best}$  en iyi çözümünü bul
    döngü = döngü + 1
end while

```

Şekil 3.4. FPA algoritması sözde kodu

BÖLÜM 4. ÖNBELLEKLEME

Metasezgisel algoritmalar, çeşitli karmaşık kısıtlamalara sahip, genellikle doğrusal olmayan ve multimodal olan karmaşık problemler için makul bir sürede yeterince iyi çözüm bulmak için kullanılan rassal yöntemlerdir. Herhangi bir metasezgisel algoritmanın iki ana faktörü, arama alanının küresel ölçekte verimli bir şekilde araştırıldığı çeşitlendirme (araştırma) ve en iyi mevcut çözüm etrafında yoğun bir arama yapılan yoğunlaştırmanın (yararlanma) olmasıdır. Bu iki faktör arasında iyi bir denge, algoritmanın yakınsama oranını artırmak ve küresel optimuma ulaşılmasına yardımcı olmak için şarttır [25].

Metasezgisel yöntemler çok popüler olmasına ve birçok optimizasyon alanına uygulanmasına rağmen, yakınsaklıklarının analizi hala matematiksel olarak kanıtlanmamıştır [29]. Gerçek dünya optimizasyon problemleri genellikle pahalı hesaplamalı amaç fonksiyonlarını ve kısıtları içerir. Bir optimizasyon işlemi birkaç binden yüz binlerce amaç fonksiyonu hesaplamasına kadar sürebilir ve küresel optimuma yakınsama kolay bir iş değildir, özellikle de küresel optimumun bilinmediği problemler için. Sonuç olarak, bir optimizasyon sürecinin ne zaman durdurulacağına karar vermek zordur.

Metasezgisel algoritmaların yakınsamasını saptamak için durdurma kriterleri (durdurma kuralları, sonlandırma kriterleri gibi) kullanılır.

Literatürde, test sonuçlarının sunumu için, yöntemler genellikle çözümleri bilinen problemler üzerindeki performansları temelinde karşılaştırılır. Bununla birlikte, pratik problemlerde, kişi önceden çözümü bilmez ve programa daha iyi bir çözüm aramayı ne zaman durduracağını belirten bir kritere gereksinim duyar. Kriter, küresel minimum bulunmasından sonra çok fazla hesaplama kaynağını boşa harcamayacak kadar sıkı olmalıdır, ancak küresel optimum bulunmadan önce algoritmanın sonlanmadığından

emin olmak için yeterince gevşek olmalıdır. Bu iki hedefin çelişkili olması sorunu zorlaştırmaktadır. Durdurma kriterlerine ilişkin iyi bir derleme çalışması [30]'da verilmiştir.

Literatürde en sık kullanılan durdurma kriterlerinden biri, önceden belirlenmiş iterasyon sayısına veya amaç fonksiyon hesaplama sayısına ulaşıldığında arama sürecini sonlandırmaktır. Durdurma noktası, deneme yanılma ile oluşturulmalıdır, çünkü genellikle bir optimizasyon problemi ile gerekli amaç fonksiyonu hesaplama sayısı arasında bir korelasyon yoktur. Metasezgisellerin rassallık özelliği nedeniyle, yakınsama için gerekli olan amaç fonksiyonu hesaplama sayısının standart sapmasının önemli ölçüde değişebilir [31]. Bu nedenle, küresel optimumun bilinmediği yüksek standart sapmalı bir gerçek dünya probleminin optimizasyon sürecini sonlandırmak için fonksiyon değerlendirmelerinin sayısına karar vermek zordur [32].

Sıkça kullanılan diğer bir sonlandırma yöntemi, iyileştirmeye dayalı durdurma kriterleri kullanmaktır. Optimizasyon işlemi başında amaç fonksiyonu değerlerinde sağlanan büyük gelişmeler süreç ilerledikçe küçülecektir. Bu nedenle, gelişme bir eşik değerinin altında ise, yakınsama sağlandığını varsayabiliriz. Öte yandan, optimum henüz bulunmamasına rağmen iyileşme göstermeyen aşamalar olabileceği için iyileştirmeye dayalı yakınsama ölçütlerinin geçerliliği birkaç ardışık döngü için izlenmelidir.

İyileştirme esaslı durdurma kriterlerinin kullanımına ilişkin bir örnek olarak iyi bilinen kaynaklı kiriş tasarım probleminin Çiçek Tozlaşma Algoritması (FPA) kullanılarak optimizasyonu aşağıda verilmiştir. Durdurma kriteri şöyledir: n sayıda ardışık döngü boyunca en iyi amaç fonksiyonu değeri ortalama bağıl değişimi ε fonksiyon toleransından küçükse ve n anki döngü sayısı $2n'$ den büyükse algoritma sona erdirilir. Farklı n ve ε değerleri için algoritmanın 100 bağımsız çalışmasının sonuçları Tablo 4.1.'de verilmektedir, burada popülasyon boyutu 25 ve maksimum döngü sayısı 2,000'dir. İlk çalıştırma kümesinde iyileştirmeye dayalı durdurma ölçütü kullanılmamış ve amaç fonksiyonu hesaplamaları sayısı 50,000'e ulaştığında algoritma durdurulmuştur. Bütün bu çalıştırmalarda [33] ile karşılaştırıldığında aynı veya daha

iyi küresel optimum değerler elde edilmiş ve sonuçların standart sapması sifira çok yakın olmuştur. Diğer üç çalıştırma kümesinde, durdurma kriterleri aşamalı olarak zorlaştırılmıştır. Durdurma kriterleri daha sıkı hale getirildiğinde sonuçların kalitesi artmış; ancak, en iyi kriterler için elde edilen en kötü sonuç en iyi sonuçtan % 2 daha pahalı olmuştur, ki bu bazen kabul edilemez bir durum olabilir.

Tablo 4.1. Kaynaklı giriş problemi - farklı durdurma kriterleri için sonuçlar

Parametre	İstatistik	Durdurma Kriterleri			
		Yok	$n = 50$ $\varepsilon=1.E-6$	$n = 100$ $\varepsilon=1.E-9$	$n = 200$ $\varepsilon=1.E-12$
Amaç fonksiyonu değeri	En iyi	1.73119	1.73377	1.73119	1.73119
	Ortalama	1.73119	1.94327	1.77156	1.73183
	En kötü	1.73120	2.63205	2.05525	1.76541
	Std sapma	0.00000	0.17845	0.06634	0.00405
Amaç fonksiyonu hesaplama sayısı	En iyi	50,000	2,550	5,050	12,300
	Ortalama	50,000	6,522	22,144	45,998
	En kötü	50,000	23,150	50,000	50,000
	Std sapma	0	3,711	13,581	8,345

n = Ardışık döngü, sayısı, ε = Fonksiyon toleransı

Yukarıdaki örnekte görüldüğü gibi, iyileştirme esaslı durdurma kriterlerinin kullanılmasının her zaman küresel optimum çözümü yakalayamama riskine sahip olduğu açıktır.

Bu bölümde, küresel optimumu yakalayamama riskini arttırmadan amaç fonksiyonu hesaplamalarını azaltmak için bir yöntem önerilmektedir. Önerilen yöntem basit ve etkindir ve herhangi bir metasezgisel optimizasyon algoritması için kolayca uygulanabilir. Bu bölümün geri kalan kısmı aşağıdaki şekilde düzenlenmiştir: Bilgisayarlarda önbellek oluşturma Bölüm 4.1., amaç fonksiyonu hesaplamalarını azaltmak için önerilen yöntem Bölüm 4.2.'de sunulmaktadır. Doğrulama testleri için kullanılan iyi bilinen dört mühendislik tasarım problemi Bölüm 4.3.'teki kısa girişten sonra Ek A'da ayrıntılı olarak anlatılmıştır. Kullanılan test platformu ve hesaplama sonuçları ile elde edilen sonuçların analizi Bölüm 4.4.'te verilmiştir.

4.1. Bilgisayarlarda Önbellekleme

Bilgisayarlarda önbellek verileri depolayan bir donanım veya yazılım bileşeni olup, bu veriler için gelecekteki isteklerin daha hızlı sunulması sağlanır; önbellekte saklanan veriler daha önceki bir hesaplamanın sonucu veya başka bir yerde depolanan verilerin kopyası olabilir.

Bir donanım önbelleği, tekrar kullanılacak olan verilerin geçici olarak depolanması için bir bellek bloğu ayrılması şeklinde uygulanır. Donanım önbellek türleri arasında CPU önbellekleri, GPU önbellekleri, dijital sinyal işlemcileri (DSP) ve çeviri önizleme arabelleği bulunur.

CPU ve GPU önbellekleri genellikle tamamen donanım tarafından yönetilirken, çeşitli yazılımlar diğer önbellekleri yönetir. Örneğin, disk önbelleğine bir örnek olan ana bellekteki sayfa önbellek, işletim sistemi çekirdeği (kernel) tarafından yönetilir.

Yazılım önbelleğinin çok yaygın bir örneği, ağ genelinde iletilmesi gereken bilginin miktarını azaltmak için web sunucuları ve web proxy sunucuları tarafından web sunucularının daha önceki yanıtlarını (web sayfaları ve resimler gibi) saklamak için kullanılan web önbellektir. Bunun nedeni, önbellekte saklanan bilgilerin çoğunlukla yeniden kullanılma gereksinimi olmasıdır. Web sunucularında önbellekleme uygulaması ile ilgili daha fazla bilgi [34-36] 'da verilmiştir.

Bir diğer yazılım önbellek türü olan memoization, öncelikle pahalı fonksiyon çağrılarının sonuçlarını saklayarak ve aynı girdiler tekrar oluştuğunda ön belleğe alınmış sonucu sağlayarak bilgisayar programlarını hızlandırmak için kullanılan bir optimizasyon tekniğidir. Bu çalışmada kullanılan ve bir sonraki bölümde açıklanan amaç fonksiyonu hesaplamalarını azaltmak için kullanılan önbellek tekniği bu kategoriye aittir. Tipik bir memoization uygulaması dinamik programlamadır, bunun için daha fazla bilgi [37, 38]'de verilmektedir.

DNS önbelleği, veritabanı önbellekleme, dağıtılmış önbellek vb. gibi önbellekleme yönteminin başka uygulamaları da vardır.

4.2. Amaç Fonksiyonu Hesaplama Sayısının Azaltılması

Rassal arama, birçok metasezgisel algoritmada ana mekanizma olarak kullanılır; algoritma bir çözüm doğrultusunda ilerledikçe arama yönünde rastgele seçimler yapılır. Sonuç olarak, bir iterasyonda, algoritmanın önceki bir iterasyonda kullanılan aynı tasarım değişken değerleri kümesi ile karşılaşması, dolayısıyla amaç fonksiyonu hesaplamasının yapılmış olması mümkündür.

Önbellekleme tekniği olarak anılan önerilen yöntemde, tasarım değişken değerleri kümesi ve buna karşı gelen amaç fonksiyonu hesaplanan değeri bir tabloda saklanır. Yeni bir tasarım değişkeni değerleri kümesi üretilen her iterasyonda, daha önce aynı kümenin kullanılıp kullanılmadığını belirlemek için, bunlar önce tablo ile karşılaştırılır. Eğer kullanılmışsa, daha önce hesaplanan amaç fonksiyonu değeri tablodan alınır ve böylece amaç fonksiyonu değerinin tekrar hesaplanmasına gerek kalmaz. Aksi takdirde amaç fonksiyonu değeri hesaplanır.

Anahtar/değer çiftlerini saklayan bir veri yapısı olan hash tablosu, yukarıda kısaca açıklanan yöntemi uyarlamak için kullanılır; burada anahtar istenen değeri bulmak için bir hash işlevini kullanarak hash'e dönüştürülür.

Bu çalışmada, hash tablolarının uyarlanması için MATLAB'ın Map Containers özelliği kullanılmıştır. Her iterasyon için anahtar, tasarım değişken değerlerinin ardarda birleştirilerek MATLAB *num2str* fonksiyonunu aracılığıyla bir dizeye dönüştürülmesiyle elde edilmektedir.

Optimizasyonun performansını artırmak için önbellekleme tekniğinin kullanılması, gelişimsel algoritmalar (EA) için, özellikle genetik algoritmalar (GA) ve genetik programlama (GP) için literatürde ele alınmıştır. Önbellekleme tekniği, ilk olarak bireylerin ikili olarak kodlanmış olduğu genetik algoritma için [39] incelenmiştir. Bu

yaklaşımında, hesaplanan amaç fonksiyonu değerleri, bireylerin genetik kodları ile birlikte bir hash veri yapısında (önbellek tablosu) saklanır. Genetik operatörler aynı genetik koda sahip bireyler ürettiğinde, amaç fonksiyonu değeri, amaç fonksiyonunu yeniden hesaplama yerine önbellek tablosundan alınır. Bu uygulamada, önbellek tablosunda saklanan bireylerin sayısı 5,000 ile sınırlandırılmıştır. Önbellek tablosu doluyorsa, yeni girilen alana yer açmak için en son kullanılan blok kaldırılır.

[39]'da önerilen önbellekleme tekniği, tesis yeri seçimi gibi pratik problemler için yapılan birkaç çalışmada uygulanmıştır [40-42]. Gray kodlu bit dizgileri olarak ele alınan karma değişken türleri içeren kısıtlı çok amaçlı bina tasarımı probleminin çözümü için, özdeş çözümleri yeniden hesaplamamak amacıyla daha önce hesaplanan çözümlerin dâhili bir önbellekte saklandığı NSGA-II algoritmasının çeşitli varyantları önerilmiştir [43]. [44]'te bellek alanı gereksinimlerini azaltmak için kromozomları doğrusal tablolar veya bağlantılı listeler yerine iki yönlü dairesel bağlantılı listelerle önbellekleme önerilmiştir.

Yukarıda belirtilen tüm çalışmalarda [39-44] ikili kodlamanın kullanıldığına dikkat edilmelidir.

Genetik programlamada zaman ve bellek gereksinimlerini azaltmak için önbellek tekniğinin kullanılması ilk olarak [45]'te önerilmiştir; burada altağaç hesaplama sonuçları bir yönlendirilmiş çevrimsiz çizgede (DAG) saklanmakta ve dolayısıyla aynı veya sonraki nesillerde tekrar ortaya çıktıkları takdirde yeniden hesaplanmasına gerek kalmamaktadır. Bu yöntem, bir popülasyonu değerlendirmek için iki yeni metod eklenerek [46]'da daha da geliştirilmiş ve böylece zaman gereksinimleri azaltılmıştır. [47]'de tanımlanan maliyet tabanlı altağaç önbellekleme mekanizması ile amaç fonksiyonu hesaplamalarının sayısı % 66 kadar azaltılmış ve bu da geçen süreyi % 52 kadar azaltmıştır.

Önbellek oluşturma özelliği, Watchmaker for GP/EA ve GPTIPS gibi GP yazılım çerçevelerinde de sunulmaktadır.

Yukarıda belirtilen EA/GP'lerin haricinde, sürü zekâsı tabanlı algoritmalar gibi diğer metasezgisel algoritmalar için önbellek tekniğinin uygulanması ile ilgili literatürde herhangi bir referans bulunamamıştır. Dolayısıyla, bu çalışmanın büyük bir olasılıkla bu konuda ilk örnek olduğu söylenebilir.

4.3. Mühendislik Tasarım Optimizasyonu Problemleri

Önerilen yöntemin amaç fonksiyonu hesaplama sayısını azaltma etkisini değerlendirmek için literatürden seçilen dört iyi bilinen karma değişkenli kısıtlı mühendislik tasarım optimizasyonu problemi ile bir test platformu oluşturulmuştur; bu problemler şunlardır: Basınçlı kap, kaynaklı kiriş, sarmal yay ve güçlendirilmiş betonarme kiriş tasarımları. Seçilen problemler [33]'te kullanılanlarla aynıdır ve Ek A'da ayrıntılı olarak açıklanmıştır.

Seçilen problemlerin her biri için tasarım değişkenlerinin alabileceği değerlere dayalı olarak iki versiyon tanımlanmıştır: Versiyon I, [33]'teki ile aynıdır, Versiyon II'de ise Versiyon I'in sürekli tasarım değişkenlerine ondalık basamaklarının sayısını sınırlayan varsayımsal üretim kısıtları empoze edilmiştir. Versiyon II'nin amacı sürekli tasarım değişkenlerine getirilen sınırlamanın hash sürecindeki isabet oranlarına etkisini irdelemektir.

4.4. Parametre Ayarları, Hesaplama Sonuçları ve Değerlendirme

Önerilen yöntem ve kıyaslama problemleri için test platformu MATLAB 2014a'da uygulanmıştır. FPA algoritması orijinal MATLAB kodu yalnızca sürekli tasarım değişkenlerini işlemek üzere tasarlanmıştır; bu kod karma-tamsayı ve ayrık tasarım değişkenlerini kullanmak üzere geliştirilmiş ve hash tablolarını uyarlamak için MATLAB'ın *Map Containers* özelliği eklenmiştir. Optimizasyon çalıştırmaları, 4 GB RAM'e sahip 2.60 GHz Intel Core i5-3320M işlemcili taşınabilir bir bilgisayarda gerçekleştirilmiştir.

4.4.1. Parametre ayarları

[33]'teki tüm kıyaslama problemleri için Ateşböceği Algoritması (FA) popülasyon büyüklüğü 25 olarak kullanılmıştır. Kullanılan maksimum döngü sayısı, basınçlı kap ve güçlendirilmiş betonarme kiriş tasarım problemleri için 1,000, kaynaklı kiriş tasarım problemi için 2,000 ve sarmal yay tasarım problemi için 3,000'dir. Kıyaslama problemleri için FA'nın istatistiksel sonuçları, 100 optimizasyon çalıştırması sonucu elde edilmiştir [33].

Karşılaştırmayı aynı temelde yapmak için, dört mühendislik tasarım probleminin optimizasyonu, popülasyon büyüklüğü 25 olan FPA algoritması kullanılarak ve her problem için [33]'tekilerle aynı maksimum döngü sayıları ile gerçekleştirilmiştir. Her bir kıyaslama problemi için biri önbelleklemesiz ve Versiyon I ile Versiyon II'nin her biri için önbelleklemeli olmak üzere üç seçenekoluşturulmuştur. [33]'e benzer şekilde, istatistiksel olarak anlamlı sonuçların elde edilebilmesi için optimizasyon çalıştırmaları, her kıyaslama problemi ve seçeneği için 100 kez tekrarlanmıştır.

Bu çalışmada FPA algoritmasının *geçiş olasılığı* p kontrol parametresi Bölüm 3.5.'te belirtildiği gibi önerilen değer $p = 0.8$ olarak alındı.

4.4.2. Hesaplama sonuçları ve değerlendirme

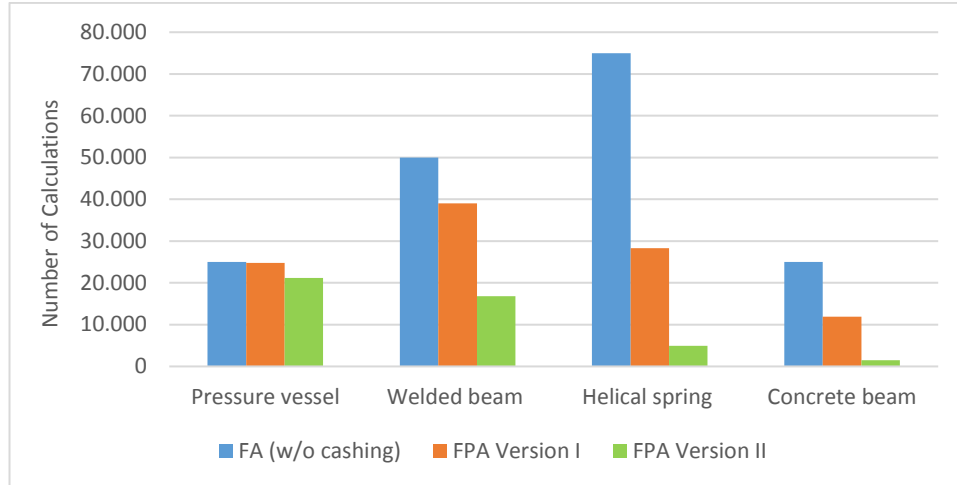
Bu çalışma için yapılan optimizasyon çalıştırmalarının istatistiksel sonuçları [33]'te verilen FA sonuçlarıyla birlikte Tablo 4.2.'de verilmiştir. Ayrıca, her kıyaslama problemi için elde edilen amaç fonksiyonu hesaplama sayısındaki azalmaların bir özeti grafik olarak Şekil 4.1.'de gösterilmiştir.

Kıyaslanan algoritmalar ile elde edilen en iyi sonuçlar için çözüm vektörleri ve bunlara karşılık gelen kısıt değerleri, kaynaklı kiriş tasarımı problemi için Tablo 4.3.'te verilmiştir.

Tablo 4.2. Her problem için FA ve FPA algoritmalarının 100 çalıştırım sonuçları

Problem	Stats	Ateşböceği	Çiçek Tozlaşma Algoritması (FPA)		
		Algoritması (FA)	Önbellek lemesiz	Önbelleklemeli	
				Ver I	Ver II
Başınçlı kap tasarımı	En iyi	5850.38306	5850.38306	5850.45565	5858.97962
	Orta	5937.33790	5852.07031	5857.94801	5863.42902
	En kötü	6258.96825	6018.71333	5976.60798	6072.66541
	Std Sap	164.54747	16.74826	14.28236	22.50275
AFHS		25,000	25,000	24,772	21,168
AFHS % azalma				0.9%	15.3%
Kaynaklı giriş tasarımı	En iyi	1.73121	1.73119	1.73119	1.74387
	Orta	1.87866	1.73119	1.73123	1.74387
	En kötü	2.34558	1.73120	1.73127	1.74387
	Std Sap	0.26780	0.00000	0.00002	0.00000
AFHS		50,000	50,000	39,019	16,799
AFHS % azalma				22.0%	66.4%
Sarmal yay tasarımı	En iyi	2.65858	2.65856	2.65856	2.67369
	Orta	4.38360	2.65856	2.65857	2.67369
	En kötü	7.81629	2.65856	2.65859	2.67369
	Std Sap	4.60763	0.00000	0.00001	0.00000
AFHS		75,000	75,000	28,335	4,927
AFHS % azalma				62.2%	93.4%
Güçlendirilmiş betonarme giriş tasarımı	En iyi	359.20800	359.20800	359.20800	359.20800
	Orta	460.70600	359.20800	359.20813	359.20800
	En kötü	669.15000	359.20800	359.20830	359.20800
	Std Sap	80.73870	0.00000	0.00009	0.00000
AFHS		25,000	25,000	11,911	1,487
AFHS % azalma				52.4%	94.1%

Stats = İstatistikler, AFHS = Amaç fonksiyonu hesaplama sayısı, Std Sap = Standart sapma



Şekil 4.1. Önbellekleme ile elde edilen hesaplama sayılarındaki azalmalar

Tablo 4.3. Kaynaklı kiriş tasarımı problemi için elde edilen en iyi çözümlerin tasarım değişkeni ve kısıtlarının karşılaştırması

Değişken/ Kısıt	Ateşböceği Algoritması (FA)	Çiçek Tozlaşma Algoritması (FPA)		
		Önbellek lemesiz	Önbelleklemeli	
			Ver I	Ver II
h	0.20150	0.20150	0.20150	0.20800
l	3.56200	3.56200	3.56200	3.45800
t	9.04140	9.04141	9.04142	8.95000
b	0.20570	0.20571	0.20571	0.21000
g_1	n/a	-0.00421	-0.00421	-0.00002
g_2	n/a	-0.23556	-0.23556	-0.00235
g_3	n/a	-0.00004	-0.01317	-0.07957
g_4	n/a	-28.25761	-28.60332	-0.38388
g_5	n/a	-3.42433	-3.42431	-0.03417
g_7	n/a	-0.07650	-0.07650	-0.00083
g_8	n/a	-0.00003	-0.16907	-3.41019
f_{best}	1.73121	1.73119	1.73119	1.74387

Aşağıda, en iyi ve ortalama sonuçlar ile standart sapma değerleri, gerekli fonksiyon hesaplama sayılarıyla birlikte, sonuçların değerlendirilmesi için temel oluşturmaktadır. En iyi sonuçlar bir algoritmanın küresel optimumu elde etme yeteneğini belirlerken, ortalama sonuçlar ve standart sapma değerleri güvenilirliğini

ortaya koyar. Versiyon I ve II için fonksiyon hesaplamalarının sayısı önerilen önbellekleme yönteminin etkililiğinin bir ölçüsü olarak düşünülebilir.

4.4.2.1. Önbelleklemesiz sonuçlar

Basınçlı kap ve güçlendirilmiş betonarme kiriş tasarımı problemleri için FPA algoritması ile elde edilen en iyi sonuçlar [33]'te FA algoritması ile elde edilenlerle aynı, kaynaklı kiriş ve sarmal yay tasarım problemleri için en iyi sonuçlar biraz daha iyidir.

Tüm kıyaslama problemleri için, FPA algoritması tarafından elde edilen ortalama sonuçlar ve standart sapma değerleri, FA algoritması için [33]'te verilenlerden önemli ölçüde daha iyidir; gerçekte, FPA algoritması için kaynaklı kiriş, sarmal yay ve güçlendirilmiş betonarme kiriş tasarım problemlerinin ortalama sonuçları en iyi sonuç değerleriyle aynıdır ve dolayısıyla standart sapma değerleri sıfırdır. Bu nedenle, FPA'nın çok güvenilir bir algoritma olduğu söylenebilir ve dolayısıyla bu çalışmada incelenen kıyaslama problemleri için [33]'teki FA ile karşılaştırıldığında açıkça daha iyi bir seçenektir.

4.4.2.2. Önbelleklemeli sonuçlar, Versiyon I

Kaynaklı kiriş, sarmal yay ve güçlendirilmiş betonarme kiriş tasarımı problemleri için FPA algoritması ile elde edilen en iyi sonuçlar [33]'te verilen FA algoritması sonuçları ile aynı veya biraz daha iyi, ortalama sonuçlar ve standart sapma değerleri önemli ölçüde daha iyidir. Basınçlı kap tasarım problemi için FPA algoritması tarafından elde edilen en iyi sonuçlar biraz daha yüksektir, ancak ortalama sonuçlar ve standart sapma değerleri FA algoritması için [33]'te verilenlerden daha düşüktür.

Önerilen yöntem, basınçlı kap tasarım problemi için amaç fonksiyonu hesaplamalarında % 0.9'luk bir marjinal tasarruf sağlamıştır. Kaynaklı kiriş tasarımı problemi için tasarruf % 22.0, sarmal yay ve güçlendirilmiş betonarme kiriş tasarımı problemleri sırasıyla % 62.2 ve % 52.4'tür.

4.4.2.3. Önbelleklemeli sonuçlar, Versiyon II

FPA algoritması ile elde edilen en iyi sonuçlar, Versiyon I'in sürekli tasarım değişkenlerine empoze edilen ek kısıtlar nedeniyle Versiyon I en iyi değerlerinden biraz daha yüksektir. Kaynaklı kiriş, sarmal yay ve güçlendirilmiş betonarme kiriş tasarım problemleri için ortalama ve en kötü sonuçlar en iyi değerler ile aynı, dolayısıyla standart sapma değerleri sıfırdır.

Önerilen yöntem FPA çözümünde basınçlı kap tasarım problemi için amaç fonksiyonu hesaplamalarında % 15.3'lük bir tasarruf sağlamıştır. Kaynaklı kiriş tasarımı problemi için sağlanan tasarruf % 66, sarmal yay ve güçlendirilmiş betonarme kiriş tasarım problemleri için ise % 90'ın üzerindedir.

BÖLÜM 5. DAĞITIM TRANSFORMATÖRLERİ TASARIM OPTİMİZASYONU

Bu bölümde tez çalışmasının amacı, kapsamı, problemin tanımı, kullanılan yöntemler ve performans değerlendirme kriterleri gibi konulara yer verilmiştir.

5.1. Çalışmanın Amacı ve Kapsamı

Bu çalışmanın amacı, doğadan esinlenen birçok metasezgisel optimizasyon algoritmasının küresel transformatör tasarım optimizasyonuna uygulanmasını karşılaştırmak ve transformatör endüstrisinin zorlu gereksinimlerini karşılamak üzere TDO problemine uygun algoritmaları önermektir. Çalışmada sadece TDO problemi için daha önce kullanılmamış algoritmalar ele alınmıştır.

Çalışmada 20 kadar algoritma incelenmiş ve bunlardan en iyi beşinin performans sonuçları bu dokümanda sunulmuştur.

Çalışma kapsamındaki transformatörlerin ortak temel teknik özellikleri aşağıda belirtilmiştir.

- a. Üç fazlı, yağlı tip dağıtım transformatörleri
- b. Sarma çekirdek konstrüksiyon
- c. AG sargı iletkeni bakır folyo, YG sargı iletkeni yuvarlak emaye bakır tel

5.2. Problemin Literatürdeki Tanımları

5.2.1. Amaç fonksiyonları

Literatürde transformatör tasarım optimizasyonu için kullanılmış olan amaç fonksiyonları aşağıda belirtilmiştir [1]:

- a. Aktif kısım ağırlığının minimizasyonu
- b. Aktif kısım maliyetinin minimizasyonu
- c. Ana malzeme maliyetinin minimizasyonu
- d. Üretim maliyetinin minimizasyonu
- e. Verimin maksimizasyonu
- f. Anma gücünün maksimizasyonu
- g. Toplam sahiplik maliyetinin (TOC) minimizasyonu

Toplam sahiplik maliyeti (TOC) yönteminde, transformatörün kullanım ömrü boyunca kayıplar nedeniyle oluşacak işletme maliyetleri günümüz maliyetlerine getirilerek satın alma fiyatına eklenir; basit formülü aşağıda verilmiştir:

$$TOC = SP + A \times P_0 + B \times P_k \quad (5.1)$$

Burada A boşa kayıp maliyeti (USD/W), P_0 boşa kayıplar, B yük kayıpları maliyeti (USD/W), P_k yük kayıpları ve SP transformatörün satış fiyatıdır.

A ve B katsayıları [1] Bölüm 2’de sırasıyla 13.39 USD/W ve 2.09 USD/W olarak verilmektedir. Bu katsayıların hesaplama detayları da [1] Bölüm 8’de verilmektedir.

Düzenleyici kuruluşların yüksek verimli transformatörlerin kullanılmasına ilişkin yönetmelikleri uygulamaya koymasına rağmen, transformatör satın alımlarında TOC'nin kullanılmaya devam edilmesi önerilmektedir [48]. Bunun nedeni,

yönetmeliklerin yalnızca transformatör verimliliği için minimum seviyeyi belirlemesi, buna karşın toplam sahiplik maliyeti yönteminin (TOC), en düşük maliyetle en büyük enerji tasarrufunu elde etmek için gereken yatırımın en aza indirgenmesini sağlamasıdır. Bu da, kayıpları verilen ihtiyaç için ekonomik açıdan en uygun olan transformatörlerin seçimiyle sonuçlanır.

5.2.2. Kısıtlar

Ulusal ve/veya ulusal standartlar ile müşteri şartnameleri uyarınca karşılanması gereken transformatör tasarım kısıtları aşağıda belirtilmiştir [1]. Bu kısıtların bir kısmı isteğe bağlıdır.

- a. Boşta kayıplar kısıtı - hesaplanan boşta kayıplar maksimum boşta kayıplar değerinden az olmalıdır.
- b. Yük kayıpları kısıtı - hesaplanan yük kayıpları maksimum yük kayıpları değerinden az olmalıdır.
- c. Toplam kayıplar kısıtı - hesaplanan toplam kayıplar maksimum toplam kayıplar değerinden az olmalıdır (toplam kayıplar, boşta kayıplar ile yük kayıplarının toplamıdır).
- d. Kısa devre empedansı kısıtı - hesaplanan kısa devre empedansı minimum ve maksimum kısa devre empedansı değerleri arasında olmalıdır.
- e. Sıcaklık yükselişi kısıtı - transformatörün toplam kayıpları nedeniyle oluşan sıcaklık yükselişi maksimum sıcaklık yükselişi değerinden az olmalıdır.
- f. Isı transferi kısıtı - transformatörün toplam kayıpları tarafından üretilen toplam ısı; iletim, konveksiyon ve radyasyonun birleşik etkileri ile atılabilmelidir.
- g. Verim kısıtı - hesaplanan verim minimum verim değerinden yüksek olmalıdır.
- h. Darbe gerilimi kısıtı - sargılardaki darbe geriliminin sargı katları arasındaki yalıtım kâğıdının dayanabileceği maksimum darbe gerilimi değerinden az olması gerekir.

- i. Endüklenmiş gerilim kısıtı - sargılardaki endüklenmiş gerilimin sargı katları arasındaki yalıtım kâğıdının dayanabileceği maksimum endüklenmiş gerilim değerinden az olması gerekir.
- j. Kazan boyutları kısıtı - transformatör kazan boyutlarının maksimum kazan boyutlarından daha az olması gerekir.

5.3. Problemin Bu Çalışmada Kullanılan Tanımı

5.3.1. Amaç fonksiyonu ve kısıtlar

Bu çalışmada kullanılan amaç fonksiyonu transformatör ana malzeme maliyetinin minimize edilmesidir, kısıtlar olarak da Bölüm 5.2.2.'de verilen kısıtlardan ilk altısı kullanılmıştır. Buna göre problemin matematiksel ifadesi aşağıdaki şekilde tanımlanır.

$$\min Z(\vec{x}) = \min \sum_{j=1}^8 c_j f_j(\vec{x}) \quad (5.2)$$

Kısıtlar,

$$P_0 - (1 + \varepsilon_1 / 100) \times P_{0,g} \leq 0$$

$$P_k - (1 + \varepsilon_1 / 100) \times P_{k,g} \leq 0$$

$$(P_0 + P_k) - (1 + \varepsilon_2 / 100) \times (P_{0,g} + P_{k,g}) \leq 0$$

$$U_{k,\min} \leq U_k \leq U_{k,\max}$$

$$\Delta T \leq \Delta T_{\max}$$

$$(P_0 + P_k) \leq TH_{CCR}$$

$$lb_j \leq x_j \leq ub_j, j = 1, 2, \dots, 8$$

$$x_j \geq 0, j = 1, 2, \dots, 8$$

Burada \vec{x} sekiz değişkenden oluşan tasarım vektörü, c_j ve f_j transformatörde kullanılan sekiz ana malzemenin sırasıyla birim maliyeti (USD/kg) ve ağırlığı (kg), P_0 hesaplanan boşta kayıplar, $P_{0,g}$ garanti edilen boşta kayıplar, P_k hesaplanan yük kayıpları, $P_{k,g}$ garanti edilen yük kayıpları, ε_1 boşta kayıplar ve yük kayıpları yüzdesel toleransı, ε_2

toplam kayıplar yüzdesel toleransı, U_k hesaplanan kısa devre empedansı, $U_{k,min}$ ve $U_{k,max}$ garanti edilen minimum ve maksimum kısa devre empedans değerleri, TH_{CCR} iletim, konveksiyon ve radyasyon yoluyla atılan ısı, ΔT hesaplanan sıcaklık yükselişi ve ΔT_{max} garanti edilen maksimum sıcaklık yükselişidir.

Ana malzemeler ve birim maliyetleri Tablo 5.1.'de, tasarım değişkenleri de Bölüm 5.3.2.'de verilmiştir.

Tablo 5.1. Ana malzemeler ve birim maliyetleri

Ana malzeme	Birim maliyeti
	USD/kg
AG sargısı iletkeni	12.01
YG sargısı iletkeni	12.01
Çekirdek malzemesi	6.01
Yalıtım kâğıdı	7.72
Soğutma kanal şeritleri	8.58
Transformatör yağı	1.72
Levha sac	1.03
Dalgalı panel sacı	1.20

5.3.2. Tasarım değişkenleri

Bu çalışma kapsamında karma-tamsayı tasarım değişkenleri kullanan “Teorik TDO” ve ayrık tasarım değişkenleri kullanan “Pratik TDO” olmak üzere iki ayrı yaklaşım oluşturulmuştur. Teorik TDO için seçilen tasarım değişkenleri [49]'da kullanılan değişkenlere benzerdir, ancak bunlara AG ve YG sargıları soğutma kanal sayılarına ilişkin iki değişken eklenmiştir; burada amaç bu değişkenlerin farklı kombinasyonları için ayrı ayrı optimizasyon çalıştırması yapmak yerine bir çalıştırmada sonucu elde edebilmektir.

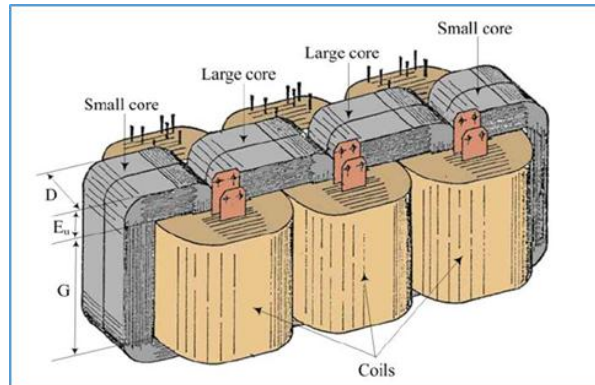
Teorik ve pratik TDO için kullanılan tasarım değişkenleri; birimleri, türleri ve alt ve üst sınırları ile birlikte sırasıyla Tablo 5.2. ve 5.3.'te verilmiştir. Değişkenlerden birkaçı Şekil 5.1.'deki aktif kısım resiminde gösterilmiştir.

Tablo 5.2. Teorik TDO tasarım değişkenleri

No	Tasarım değişkeni	Birim	Tür	Alt ve üst sınırlar	
				Alt	Üst
1	AG sargısı sarım sayısı	-	Tamsayı	10	50
2	Manyetik endüksiyon	Gauss	Reel sayı	14,000	17,500
3	Çekirdek sac genişliği (Şekil 5.1., D)	mm	Reel sayı	100	500
4	Çekirdek pencere yüksekliği (Şekil 5.1., G)	mm	Reel sayı	100	500
5	AG sargısı akım yoğunluğu	A/mm ²	Reel sayı	1.00	5.00
6	YG sargısı akım yoğunluğu	A/mm ²	Reel sayı	1.00	5.00
7	AG sargısı soğutma kanal sayısı	-	Tamsayı	0	12
8	YG sargısı soğutma kanal sayısı	-	Tamsayı	0	12

Tablo 5.3. Pratik TDO tasarım değişkenleri

No	Tasarım değişkeni	Birim	Tür	Alt ve üst sınırlar	
				Alt	Üst
1	AG sargısı sarım sayısı	-	Tamsayı	10	50
2	Manyetik endüksiyon	Gauss	Tamsayı	14,000	17,500
3	Çekirdek sac genişliği (Şekil 5.1., D)	mm	Tamsayı	100	500
4	AG sargısı iletken genişliği	mm	Tamsayı	100	500
5	AG sargısı iletken kalınlığı	mm	Ayrık	0.20	2.24
6	YG sargısı iletken çapı	mm	Ayrık	0.50	5.00
7	AG sargısı soğutma kanal sayısı	-	Tamsayı	0	12
8	YG sargısı soğutma kanal sayısı	-	Tamsayı	0	12

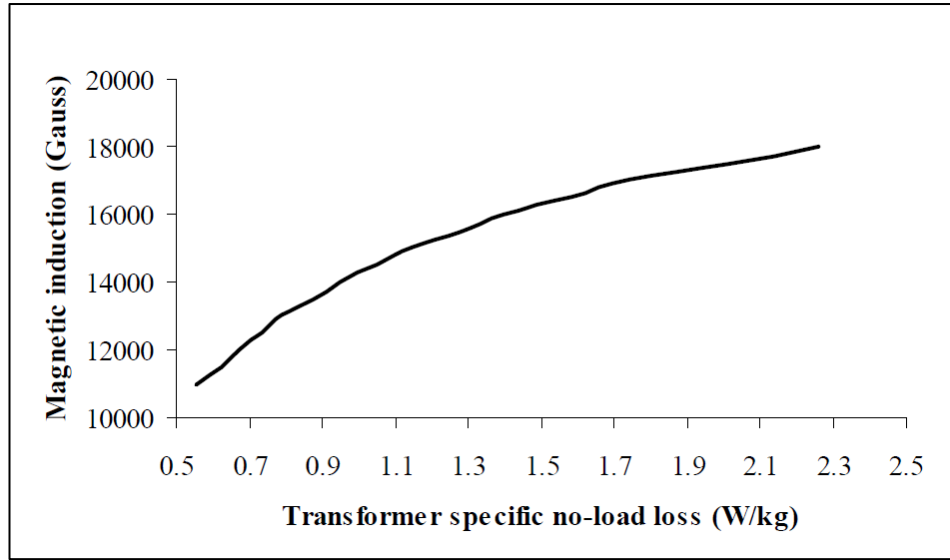


Şekil 5.1. Sarma çekirdek tipi dağıtım transformatörünün aktif kısmı [50]

$$\begin{aligned}
& + 1682.589027202759FD_{max}^4 + 3072.639488094771FD_{max}^3 \\
& + 3146.803358051449FD_{max}^2 - 1712.517232171855FD_{max} \\
& + 387.025142780563)
\end{aligned}$$

Burada $SNLL_{TF}$ boşta kayıplar (W/kg) ve FD_{max} manyetik endüksiyondur (Gauss).

R-kare değeri: 0.9997780405



Şekil 5.3. Boşta kayıplar eğrisi [1]

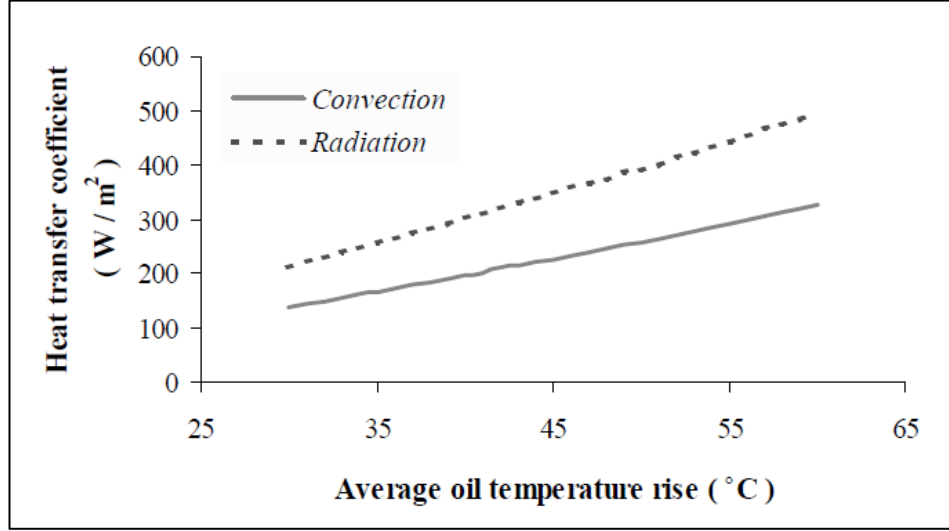
5.4.2. Kazan konveksiyon ve radyasyon ile ısı transfer katsayısı eğrileri

Şekil 5.4.'teki transformatör kazanı konveksiyon ile ısı transferi katsayısı eğrisinin formülü aşağıda verilmiştir.

$$\begin{aligned}
TCC = & -0.0000545744472AOR^4 + 0.0092765754930AOR^3 \\
& - 0.5503041809674AOR^2 + 19.5703665004558AOR \\
& - 163.3915803824412
\end{aligned} \tag{5.4}$$

Burada TCC kazan konveksiyon katsayısı (W/m^2) ve AOR ortalama yağ sıcaklık yükselişidir ($^{\circ}C$).

R-kare değeri: 0.9996598234



Şekil 5.4. Kazan konveksiyon ve radyasyon ile ısı transfer katsayısı eğrileri [1]

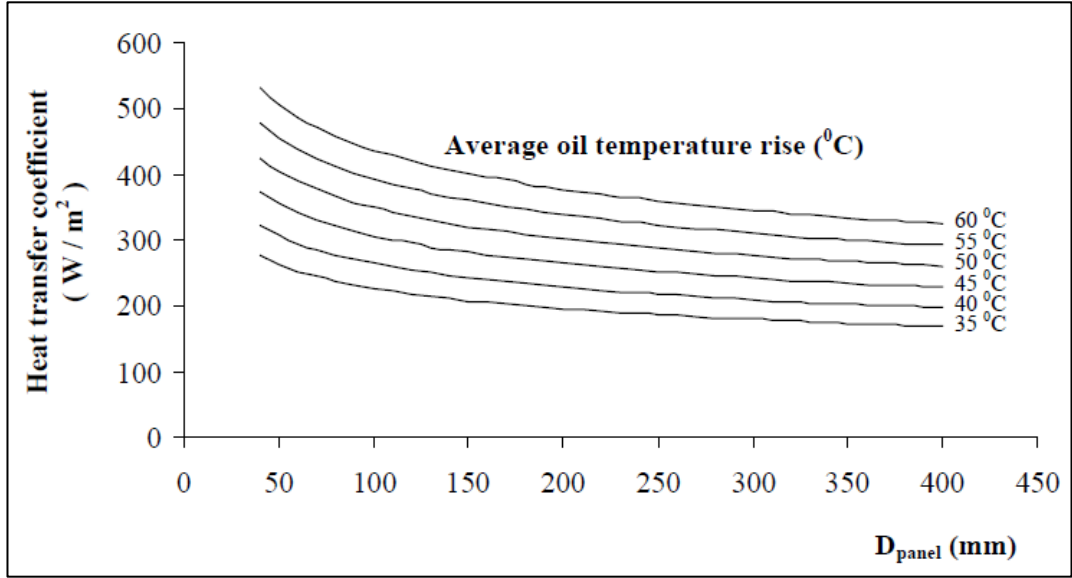
Şekil 5.4.'teki transformatör kazanı radyasyon ile ısı transferi katsayısı eğrisinin formülü aşağıda verilmiştir.

$$\begin{aligned}
 TRC = & 0.88733882883505677AOR^{1.5} \\
 & + 20.258956145521502 \log(AOR) \\
 & + 2.6813522474580846 \sin(AOR) \\
 & - 0.80512690158999911 \cos(AOR)
 \end{aligned} \tag{5.5}$$

Burada TRC kazan radyasyon katsayısı (W/m^2) ve AOR ortalama yağ sıcaklık yükselişidir ($^{\circ}C$).

R-kare değeri: 0.9999857617

5.4.3. Dalgalı panel ısı transfer katsayısı eğrileri



Şekil 5.5. Dalgalı panel ısı transfer katsayısı eğrileri [1]

Şekil 5.5.'teki dalgalı panel ısı transferi katsayısı eğrisinin formülü aşağıda verilmiştir.

$$CPC = e^{(2.1390993595909102 - 0.20995987044882686 \log(D_{Panel}) + 1.2033475993994058 \log(AOR))} - 8.6930169541921476 \quad (5.6)$$

Burada CPC dalgalı panel katsayısı (W/m^2), D_{Panel} dalgalı panel genişliği (mm) ve AOR ortalama yağ sıcaklık yükselişidir ($^{\circ}C$).

R-kare değeri: 0.9997566062

5.5. Kısıt Yönetimi

Bu çalışmada kısıtların yönetimi için statik ceza ve ölüm cezasını içeren ceza fonksiyonu yöntemi kullanılmıştır.

Kuşakçı ve Can'a göre [51] cezaya dayalı kısıt yönetimi yönteminde aşağıdaki iki temel sorun ortaya çıkmaktadır:

- a. Kısıtlara bağılı olarak ceza ağırlıklarının ayarlanması
- b. Popülasyonda bulundurulacak uygulanabilir olmayan birey yüzdesinin ayarlanması

Yukarıda belirtilen iki sorunu önlemek için ceza ağırlıkları çok titizlikle ayarlanmalıdır. Küçük bir ceza düzeyi optimizasyonun uygulanamaz olan bir çözümle sonuçlanmasına yol açabilir (cezalandırılmış bazı bireylerin uygunluk değeri en iyi uygulanabilir bireyinkinden daha iyi olabilir); öte yandan, yüksek ceza seviyeleri, uygulanabilir bölge içindeki aramayı kısıtlar ve algoritmanın optimum çözüme yakınsamasında başarısız olmasına yol açabilir.

Denklem 5.1'deki A ve B katsayıları kullanılarak geliştirilen statik ceza hesaplama yönteminin detayları aşağıda verilmiştir.

- a. Denklem 5.1'de A ve B katsayıları satış fiyatı bazında verilmektedir; bunlar ana malzeme maliyeti seviyesine normalize edilerek A' boşta kayıplar ceza faktörü ve B' yük kayıpları ceza faktörü bulunur ($A' = 5.16$ USD/W, $B' = 0.81$ USD/W).
- b. Toplam kayıplar ve ısı transferi kısıtları için A' ve B' faktörlerinin garanti edilen kayıplara göre ağırlıklı ortalamaları alınarak C' ceza faktörü bulunur ($C' = 1.29$ USD/W).
- c. Garanti edilen boşta kayıplar artı toleransı aşan boşta kayıplar A' ceza faktörü ile cezalandırılır.
- d. Garanti edilen yük kayıpları artı toleransı aşan yük kayıpları B' ceza faktörü ile cezalandırılır.
- e. Garanti edilen toplam kayıplar artı toleransı aşan toplam kayıplar C' ceza faktörü ile cezalandırılır.
- f. Kazan ve dalgalı paneller tarafından atılamayan ısı miktarı C' ceza faktörü ile cezalandırılır.

Yukarıdaki şekilde hesaplanan tüm cezalar toplanarak toplam ceza bulunur.

Kısa devre empedansı ve sıcaklık yükselişi kısıtları için ölüm cezası yöntemi uygulanır.

Bu çalışmada kullanılan altı kısıt için statik ve ölüm cezası yöntemlerinin çeşitli kombinasyonları denenmiş, ancak en iyi sonuçlar detayları yukarıda verildiği şekilde kayıplarla ilgili kısıtlar için statik ceza, kısa devre empedansı ve sıcaklık yükselişi kısıtları için de ölüm cezası uygulandığında elde edilmiştir.

Yukarıda açıklanan yöntemin dışında, kısıt yönetimi konusunda iki yöntem daha test edilmiştir. Bunlardan birincisinde [52] hesaplanan ceza değeri iterasyon numarası ile ilişkilendirilerek arama süreci boyunca dinamik olarak artırılmaktadır; başka bir deyişle uygulanabilir olmayan çözümler üzerindeki baskı süreç boyunca giderek artmaktadır. Test edilen ikinci yöntemde ise [53], popülasyondaki bireylerden birinin bir bileşeni (tasarım değişkeni) tanımlanan sınırların dışına çıkarsa tanımlanan sınır ile o ana kadar bulunan en iyi çözümün ilgili bileşeni arasında rastgele seçilen bir bileşenle değiştirilmektedir. Test edilen her iki yöntem de yukarıda anlatılan yöntemle benzer veya daha kötü sonuçlar verdiği için uygulanmamıştır.

5.6. Algoritma Kıyaslama (Benchmarking) Yöntemi

Deneyimli transformatör tasarım mühendislerinden alınan bilgilere dayanarak, metasezgisel algoritmaların bir tasarım mühendisinin bakış açısından kıyaslaması için aşağıdaki pragmatik yöntem geliştirilmiştir.

a. Güvenilirlik (Robustness)

1. Doğruluk (Accuracy) - Bir algoritma, çok sayıda (en az 10) ardışık test çalıştırması sonucunda elde edilen çözümlerin ortalama doğruluğu küresel optimumla kıyaslandığında % 99.5 ise "orta" olarak derecelendirilir (başka bir deyişle, $\epsilon_{mean} = \% 0.5$)

2. Hassasiyet (Precision) - Bir algoritma, çok sayıda (en az 10) ardışık test çalıştırması sonucunda elde edilen çözümler arasındaki en düşük doğruluk küresel optimumla kıyaslandığında % 99 ise "orta" olarak derecelendirilir (başka bir deyişle, $\epsilon_{max} = \% 1.0$)
- b. Hız (Speed) – Bir algoritma, tek bir optimizasyon çalıştırmasının CPU süresi 15 saniye ise "orta" olarak derecelendirilir (MATLAB'ın cputime fonksiyonuyla ölçülen CPU süresi, yalnızca MATLAB çalışırken genel bir dizüstü bilgisayarın saat süresine eşittir.)

Yukarıdakilere dayanarak, performans testlerini puanlamak için Tablo 5.4.'teki sistem kullanılmıştır.

Tablo 5.4. Algoritma kıyaslama puanlama sistemi

Değerlendirme	Puan	$\epsilon_{mean} \%$	$\epsilon_{max} \%$	CPU
				saniye
Olağanüstü	100	0.0	0.0	0
Mükemmel	90	0.1	0.2	3
Çok iyi	80	0.2	0.4	6
İyi	70	0.3	0.6	9
Orta-iyi	60	0.4	0.8	12
Orta	50	0.5	1.0	15
Orta-kötü	40	0.6	1,2	18
Kötü	30	0.7	1.4	21
Çok kötü	20	0.8	1,6	24
Son derece kötü	10	0.9	1.8	27
Kabul edilemez	0	1.0	2.0	30

5.7. Algoritma Kontrol Parametrelerinin Ayarlanması

Bu tez çalışmasında kullanılan algoritmaların tümü için kontrol parametreleri ayarlama çalışması yapılmıştır. Bu algoritmalar arasında en fazla kontrol parametresi olan ABC algoritması için yapılan çalışma aşağıda verilmiştir.

Bölüm 3.1.'de belirtildiği gibi, ABC algoritmasının *limit*, *SPP* ve *MR* olmak üzere üç kontrol parametresi vardır. [19]'da $limit = \{0.5 \times SN \times D, 2 \times SN \times D, 5 \times SN \times D\}$, $SPP = \{0.5 \times SN \times D, SN \times D, 2 \times SN \times D\}$ ve *MR* da 0.1-0.9 arası 9 değer için sonuçlar üzerindeki etkisinin incelendiği ifade edilmiştir; burada *SN* popülasyon büyüklüğü, *D* ise problemin boyutudur.

[20]'de verilen ABC algoritması Delphi kodunda ise $limit = 10 \times SN \times D$, $SPP = SN \times D$ ve $MR = 0.9$ olarak kullanılmıştır.

ABC algoritmasının TDO problemi için yapılan parametre ayarlama çalışmasında $limit = \{0.5 \times SN \times D, 2 \times SN \times D, 5 \times SN \times D\}$, $SPP = \{0.5 \times SN \times D, SN \times D, 2 \times SN \times D\}$ ve $MR = \{0.6, 0.7, 0.8, 0.9\}$ değerleri alınmıştır. Bu değerler için maksimum döngü sayısı 1000 ve popülasyon büyüklüğü 40 seçilerek 160 kVA transformatör için yapılan optimizasyon çalışmasının sonuçları Tablo 5.5.'te verilmiştir.

En iyi sonuçlardan birini oluşturan $limit = 1280$, $SPP = 640$ ve $MR = 0.9$ kombinasyonu uygulandığında 160 kVA transformatör için maksimum döngü sayısı = {1000, 1500, 2000, 2500} ve popülasyon büyüklüğü = {20, 40, 60, 80, 100} kullanılarak elde edilen optimum çözümler Tablo 5.6.'da verilmiştir.

160 kVA transformatör için yapılan çalışma 400 kVA ve 630 kVA transformatörler için de tekrarlanmış ve her üç tip transformatör için elde edilen sonuçların ortalama hata değerleri ile aynı transformatörler için Delphi kodundaki parametreler kullanılarak elde edilen sonuçların ortalama hata değerlerinin karşılaştırması Tablo 5.7.'de verilmiştir. Delphi kodunda kullanılan kontrol parametreleri ile bu çalışmada

test edilenler arasındaki farklar çok küçük olduğu için bu tez çalışmasında Delphi kodundaki kontrol parametrelerinin kullanılması tercih edilmiştir.

BSA algoritmasının [24]'deki MATLAB kodunda ölçekleme faktörü F kontrol parametresi için beş seçenek bulunmaktadır. Bu seçeneklerin TDO problem üzerindeki etkileri irdelenmiş ve aralarındaki farklar az olmakla birlikte, biraz daha iyi sonuçlar verdiği için bu tez çalışmasında ters gamma dağılımını simüle eden $F = 1 / \text{gamrnd}(1, 0.5)$ seçeneğinin kullanılması tercih edilmiştir.

b6e6r1, CS ve FPA algoritmaları için yapılan parametre ayarlama çalışmalarında bu algoritmaların TDO problem için kontrol parametrelerine duyarlı olmadıkları gözlenmiş ve bu nedenle Bölüm 3.'te verilen önerilen kontrol parametre değerleri bu tez çalışmasında kullanılmıştır.

Tablo 5.5. ABC kontrol parametrelerinin TDO problemi üzerindeki etkileri

<i>limit</i>	<i>spp</i>	<i>MR</i>	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	<i>limit</i>	<i>spp</i>	<i>MR</i>	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
320	320	0.6	0.28	0.45	320	320	0.8	0.23	0.37
320	640	0.6	0.35	0.86	320	640	0.8	0.29	0.60
320	1280	0.6	0.33	0.63	320	1280	0.8	0.23	0.41
1280	320	0.6	0.33	0.63	1280	320	0.8	0.25	0.58
1280	640	0.6	0.43	0.73	1280	640	0.8	0.21	0.37
1280	1280	0.6	0.27	0.50	1280	1280	0.8	0.27	0.42
3200	320	0.6	0.42	0.75	3200	320	0.8	0.25	0.64
3200	640	0.6	0.41	0.71	3200	640	0.8	0.31	0.76
3200	1280	0.6	0.41	0.68	3200	1280	0.8	0.22	0.29
320	320	0.7	0.31	0.70	320	320	0.9	0.18	0.33
320	640	0.7	0.40	0.81	320	640	0.9	0.37	0.86
320	1280	0.7	0.32	0.43	320	1280	0.9	0.25	0.37
1280	320	0.7	0.28	0.39	1280	320	0.9	0.24	0.32
1280	640	0.7	0.39	1.07	1280	640	0.9	0.19	0.34
1280	1280	0.7	0.26	0.48	1280	1280	0.9	0.19	0.33
3200	320	0.7	0.23	0.32	3200	320	0.9	0.29	0.59
3200	640	0.7	0.28	0.56	3200	640	0.9	0.28	0.36
3200	1280	0.7	0.34	0.57	3200	1280	0.9	0.23	0.36

Tablo 5.6. ABC *limit* = 1280, *SPP* = 640 ve *MR* = 0.9 kontrol parametreleri ile 160 kVA optimizasyon sonuçları

Döngü	Pop	Min fval	Ave fval	Max fval	CPU sec	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
1000	20	3,535	3,551	3,596	1.5	0.57	1.85
1000	40	3,532	3,541	3,553	2.8	0.28	0.63
1000	60	3,533	3,540	3,551	4.1	0.25	0.57
1000	80	3,532	3,538	3,545	5.4	0.21	0.40
1000	100	3,534	3,538	3,542	6.8	0.18	0.30
1500	20	3,536	3,550	3,610	2.3	0.53	2.23
1500	40	3,534	3,541	3,557	4.3	0.29	0.73
1500	60	3,534	3,538	3,542	6.4	0.20	0.31
1500	80	3,534	3,537	3,541	8.4	0.17	0.28
1500	100	3,533	3,537	3,541	10.2	0.17	0.29
2000	20	3,534	3,539	3,552	3.0	0.23	0.59
2000	40	3,535	3,539	3,552	5.6	0.23	0.59
2000	60	3,534	3,538	3,542	8.4	0.20	0.32
2000	80	3,535	3,538	3,541	10.8	0.21	0.29
2000	100	3,533	3,538	3,542	13.7	0.18	0.32
2500	20	3,532	3,544	3,573	4.0	0.37	1.20
2500	40	3,533	3,538	3,542	7.4	0.19	0.31
2500	60	3,534	3,538	3,551	10.8	0.20	0.56
2500	80	3,532	3,538	3,542	14.3	0.20	0.32
2500	100	3,534	3,538	3,541	17.2	0.19	0.27
fval = Amaç fonksiyonu değeri					Ortalama	0.25	0.62

Tablo 5.7. ABC Delphi kodu ile bu çalışmada kullanılan kontrol parametrelerinin karşılaştırması

Parametre	160 kVA		400 kVA		630 kVA		Ortalama	
	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
Delphi kodu	0.26	0.62	0.37	1.08	0.40	1.30	0.34	1.00
Bu çalışma	0.25	0.62	0.36	0.99	0.37	0.99	0.33	0.86
Fark	0.01	0.00	0.01	0.09	0.02	0.31	0.01	0.14

BÖLÜM 6. HESAPLAMA SONUÇLARI VE DEĞERLENDİRME

Bu bölümde farklı güçlerdeki üç dağıtım transformatörü için yapılan teorik ve pratik tasarım optimizasyonu çalışmalarının sonuçları ve bu sonuçlara ilişkin değerlendirmeler yer almaktadır.

6.1. İncelenen Transformatörlerin Teknik Özellikleri ve Kısıtları

Tasarım optimizasyonları yapılan transformatörlerin ortak temel teknik özellikleri aşağıda verilmiştir.

- a. Üç fazlı, yağlı tip dağıtım transformatörleri
- b. 160, 400 ve 630 kVA güç değerleri
- c. Primer/sekonder gerilimler 20/0.4 kV
- d. Sarma çekirdek konstrüksiyon
- e. AG sargı iletkeni bakır folyo, YG sargı iletkeni yuvarlak emaye bakır tel
- f. Kayıp ve kısa devre empedans değerleri Tablo 6.1.'e göre [1]
- g. Maksimum sıcaklık yükselişi sınırları:
 1. Ortalama sargı sıcaklığı yükselişi 60 K
 2. Tepe yağ sıcaklığı yükselişi 55 K
- h. IEC 60076-1 standardına uygun diğer özellikler

Transformatörlerin toplam ana malzeme maliyetlerini hesaplamak için Tablo 5.1.'de verilen malzeme birim maliyetleri kullanılmıştır.

Tablo 6.1. İncelenen transformatörlerin kısıt ve toleransları

Güç	Kısıtlar		
	P ₀ (W)	P _k (W)	U _k (%)
160 kVA	425	2,350	4
400 kVA	750	4,600	4
630 kVA	1,100	8,900	6
Toleranslar	+15%	+15%	±10%
	+10%		

P₀ = Boşta kayıplar, P_k = Yük kayıpları, U_k = Kısa devre empedansı

6.2. Teorik Transformatör Tasarım Optimizasyonu (TDO) Performans Testleri

6.2.1. Test yöntemi

Teorik transformatör tasarım optimizasyonu (TDO) performans testleri için MATLAB 2014a kullanılarak hazırlanan yazılım, bir ana program, beş optimizasyon algoritmasının her biri için bir altprogram ve tüm algoritma altprogramları tarafından ortak kullanılan bir tasarım hesaplama altprogramından oluşmuştur. Orijinal kodları yalnızca sürekli tasarım değişkenleri kullanabilen her beş algoritma karma-tamsayı değişkenler kullanmak üzere geliştirilmiştir.

Performans testleri 2.60 GHz Intel Core i5-3320M CPU ve 4 GB RAM'e sahip olan standart bir dizüstü bilgisayar kullanılarak aşağıdaki çerçevede gerçekleştirilmiştir:

- Bölüm 6.1.'de verilen spesifikasyonlara sahip üç tip transformatör için
- Beş algoritma (ABC, b6e6rl, BSA, CS, FPA) kullanılarak
- Tasarım değişkenleri Tablo 5.2.'de verilenler olmak üzere
- Tasarım değişkenlerinin alt ve üst sınır değerleri incelenen üç tip transformatörden daha da geniş bir güç aralığındaki transformatörleri kapsayacak şekilde geniş tutulmuştur (Tablo 5.2.)
- Aşağıda verilen maksimum döngü sayıları ve popülasyon büyüklükleri için
 - Maksimum döngü sayısı 1000, 1500, 2000, 2500

- 2. Popülasyon büyüklüğü 20, 40, 60, 80, 100
- f. Algoritma kontrol parametreleri Bölüm 5.7.'ye göre
- g. Durdurma kriteri kullanılmamıştır
- h. Testler her bir transformatör tipi ve algoritma için 20 kez tekrarlanmıştır

Her algoritmanın en iyi sonucu vereceği maksimum döngü sayısı ve popülasyon büyüklüğü kombinasyonu farklı olabileceğinden performans testleri bu değerlerin toplam 20 farklı kombinasyonu için yapılmıştır.

6.2.2. Test sonuçları ve değerlendirme

160 kVA transformatör için b6e6r1, 400 kVA için CS ve 630 kVA için BSA algoritmaları ile elde edilen optimizasyon sonuçları sırasıyla Tablo 6.2., Tablo 6.3. ve Tablo 6.4.'te; tüm algoritmaları içeren sonuçlar da Ek B'de verilmiştir.

Bölüm 5.6.'da verilen tanımlara dayanarak, doğruluk ve hassasiyet faktörlerinin eşit ağırlıkta alındığı bir güvenilirlik puanı ve güvenilirlik ile hız faktörlerinin ağırlıklarının sırasıyla % 80 ve % 20 olduğu bir toplam puan her test için hesaplanmıştır. En iyi 20 güvenilirlik puanlarının listesi Tablo 6.5.'te, her bir algoritma için en iyi güvenilirlik puanı Tablo 6.6.'da; en iyi 20 toplam puanların listesi Tablo 6.7.'de ve her bir algoritma için en iyi toplam puan Tablo 6.8.'de verilmiştir; bu tablolardaki ε_{mean} üç transformatör tipi için ortalama, ε_{max} ise en kötü değerlerdir. Şekil 6.1. her transformatör tipi için ve genel olarak algoritmaların en iyi güvenilirlik puanlarının karşılaştırmasını göstermektedir.

Tablo 6.2. TDO - 160 kVA için b6e6rl algoritması optimizasyon sonuçları

Döngü	Pop	Min fval	Ave fval	Max fval	CPU sec	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
1000	20	3,531	3,538	3,562	3.3	0.20	0.88
1000	40	3,531	3,533	3,537	6.7	0.06	0.16
1000	60	3,531	3,532	3,536	10.2	0.03	0.15
1000	80	3,531	3,532	3,536	13.5	0.03	0.15
1000	100	3,532	3,532	3,536	16.3	0.03	0.15
1500	20	3,531	3,538	3,554	4.9	0.19	0.65
1500	40	3,531	3,533	3,547	9.9	0.07	0.45
1500	60	3,531	3,532	3,537	14.4	0.04	0.16
1500	80	3,531	3,532	3,536	19.7	0.03	0.15
1500	100	3,531	3,532	3,533	24.5	0.02	0.04
2000	20	3,531	3,539	3,564	6.2	0.22	0.92
2000	40	3,531	3,532	3,533	13.0	0.03	0.04
2000	60	3,532	3,532	3,533	18.8	0.03	0.04
2000	80	3,531	3,532	3,536	25.7	0.03	0.15
2000	100	3,531	3,532	3,536	31.4	0.03	0.15
2500	20	3,531	3,535	3,551	7.7	0.12	0.58
2500	40	3,531	3,532	3,536	15.8	0.03	0.15
2500	60	3,531	3,532	3,536	23.7	0.03	0.15
2500	80	3,531	3,532	3,536	32.9	0.03	0.15
2500	100	3,531	3,532	3,533	42.3	0.02	0.04

Tablo 6.3. TDO - 400 kVA için CS algoritması optimizasyon sonuçları

Döngü	Pop	Min fval	Ave fval	Max fval	CPU sec	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
1000	20	6,363	6,373	6,386	2.0	0.30	0.50
1000	40	6,360	6,373	6,382	3.8	0.29	0.43
1000	60	6,362	6,370	6,378	5.8	0.26	0.38
1000	80	6,363	6,370	6,378	7.6	0.26	0.38
1000	100	6,361	6,368	6,376	9.4	0.22	0.34
1500	20	6,356	6,366	6,380	3.0	0.18	0.41
1500	40	6,359	6,365	6,371	5.7	0.18	0.26
1500	60	6,359	6,364	6,368	8.5	0.16	0.22
1500	80	6,359	6,363	6,367	11.3	0.14	0.20
1500	100	6,357	6,362	6,369	14.1	0.13	0.24
2000	20	6,355	6,362	6,373	4.0	0.12	0.30
2000	40	6,356	6,362	6,369	7.7	0.12	0.24
2000	60	6,357	6,362	6,368	11.4	0.13	0.22
2000	80	6,356	6,361	6,370	15.2	0.11	0.25
2000	100	6,358	6,360	6,364	18.8	0.09	0.15
2500	20	6,354	6,361	6,372	5.0	0.10	0.29
2500	40	6,355	6,359	6,362	9.6	0.07	0.13
2500	60	6,356	6,359	6,364	14.2	0.08	0.16
2500	80	6,356	6,359	6,362	18.8	0.08	0.13
2500	100	6,355	6,358	6,363	23.5	0.06	0.14

Tablo 6.4. TDO - 630 kVA için BSA algoritması optimizasyon sonuçları

Döngü	Pop	Min fval	Ave fval	Max fval	CPU sec	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
1000	20	7,402	7,420	7,453	1.2	0.24	0.69
1000	40	7,403	7,409	7,418	2.2	0.10	0.22
1000	60	7,402	7,408	7,419	3.2	0.08	0.23
1000	80	7,402	7,405	7,413	4.2	0.04	0.14
1000	100	7,402	7,407	7,433	5.2	0.06	0.42
1500	20	7,402	7,416	7,453	1.7	0.19	0.70
1500	40	7,402	7,407	7,415	3.3	0.07	0.18
1500	60	7,402	7,404	7,413	4.8	0.02	0.14
1500	80	7,402	7,403	7,413	6.4	0.02	0.14
1500	100	7,402	7,404	7,413	7.8	0.03	0.14
2000	20	7,402	7,416	7,447	2.2	0.19	0.60
2000	40	7,402	7,406	7,447	4.2	0.05	0.61
2000	60	7,402	7,405	7,413	6.2	0.04	0.15
2000	80	7,402	7,404	7,421	8.3	0.03	0.25
2000	100	7,402	7,403	7,410	10.2	0.02	0.11
2500	20	7,402	7,426	7,695	2.8	0.32	3.96
2500	40	7,402	7,407	7,415	5.4	0.06	0.17
2500	60	7,402	7,403	7,413	7.9	0.02	0.14
2500	80	7,402	7,403	7,413	10.4	0.02	0.14
2500	100	7,402	7,403	7,407	12.9	0.01	0.07

Performans testleri, Rekabetçi-Uyarlamalı Diferansiyel Gelişim (b6e6rl) algoritmasının güvenilirlik açısından diğerlerine göre daha iyi olduğunu açıkça göstermiştir; öte yandan b6e6rl incelenen beş algoritmanın en yavaşıdır. CS'nin güvenilirliği b6e6rl'ye yakındır ve her iterasyon için iki kez amaç fonksiyonu hesaplaması gerekmesine rağmen b6e6rl'dan biraz daha hızlıdır.

BSA, FPA ve ABC algoritmaları b6e6rl ve CS'den daha hızlıdır ve bu üç algoritmanın hızı birbirine yakındır. Güvenilirlik puanı CS'ninkine yakın olan BSA aralarında en güvenilir olanıdır. Bununla birlikte, güvenilirlik puanı BSA ve FPA arasında ve ayrıca FPA ve ABC arasında da hızlı bir şekilde azalmaktadır. Sonuç olarak, teorik TDO uyarlaması için en uygun algoritma b6e6rl'dir, CS ikinci bir seçenek olabilir. Ancak diğer algoritmalar bu amaç için uygun değildir.

Tablo 6.5. TDO - En iyi 20 güvenilirlik puanı

Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
b6e6rl	1000	100	0.01	0.02	15.8	99.1	47.4	88.8
b6e6rl	2000	100	0.01	0.04	31.3	98.6	0.0	0.0
b6e6rl	2500	100	0.01	0.04	39.1	98.6	0.0	0.0
b6e6rl	1000	80	0.01	0.05	12.9	98.4	57.2	90.2
b6e6rl	1500	80	0.01	0.05	18.9	98.4	37.0	86.1
b6e6rl	1500	100	0.01	0.13	23.9	96.2	20.4	81.1
b6e6rl	2000	60	0.01	0.19	18.7	94.8	37.6	83.3
b6e6rl	2500	80	0.01	0.19	31.6	94.7	0.0	0.0
CS	2500	80	0.05	0.13	19.0	94.5	36.5	82.9
CS	2500	100	0.04	0.14	23.8	94.5	20.8	79.7
CS	2500	40	0.05	0.13	9.7	94.4	67.6	89.1
b6e6rl	1500	60	0.01	0.20	14.1	94.4	53.2	86.1
b6e6rl	1000	60	0.02	0.19	9.4	94.1	68.5	89.0
CS	2500	60	0.05	0.16	14.4	93.4	52.1	85.1
CS	2000	100	0.06	0.15	19.0	93.2	36.7	81.9
b6e6rl	2000	80	0.01	0.27	25.1	92.6	16.3	77.3
b6e6rl	2500	60	0.02	0.27	22.9	92.1	23.5	78.4
BSA	2500	100	0.04	0.27	11.8	91.3	60.7	85.2
CS	2000	60	0.07	0.22	11.5	90.9	61.6	85.0
b6e6rl	1000	40	0.05	0.27	6.1	90.8	79.5	88.6

Tablo 6.6. TDO - Her algoritma için en iyi güvenilirlik puanı

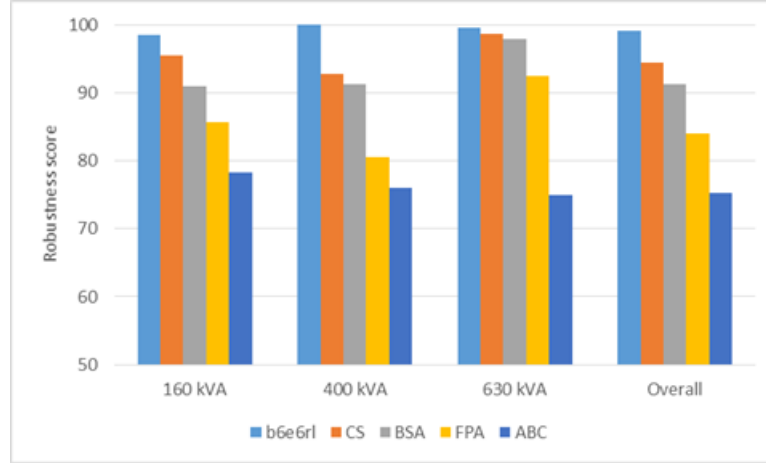
Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
b6e6rl	1000	100	0.01	0.02	15.8	99.1	47.4	88.8
CS	2500	80	0.05	0.13	19.0	94.5	36.5	82.9
BSA	2500	100	0.04	0.27	11.8	91.3	60.7	85.2
FPA	2500	60	0.17	0.31	10.9	84.0	63.6	79.9
ABC	1500	100	0.22	0.55	8.5	75.2	71.7	74.5

Tablo 6.7. TDO - En iyi 20 toplam puan

Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
b6e6rl	1000	80	0.01	0.05	12.9	98.4	57.2	90.2
CS	2500	40	0.05	0.13	9.7	94.4	67.6	89.1
b6e6rl	1000	60	0.02	0.19	9.4	94.1	68.5	89.0
b6e6rl	1000	100	0.01	0.02	15.8	99.1	47.4	88.8
b6e6rl	1000	40	0.05	0.27	6.1	90.8	79.5	88.6
CS	2500	20	0.06	0.29	5.0	89.8	83.3	88.5
BSA	2000	60	0.08	0.27	5.7	89.0	80.9	87.4
CS	2000	40	0.07	0.24	7.8	90.4	74.2	87.1
CS	2000	20	0.08	0.36	4.0	87.0	86.7	86.9
CS	1500	40	0.11	0.26	5.8	88.1	80.6	86.6
b6e6rl	1500	60	0.01	0.20	14.1	94.4	53.2	86.1
b6e6rl	1500	80	0.01	0.05	18.9	98.4	37.0	86.1
BSA	1500	80	0.09	0.33	5.7	87.3	81.0	86.0
BSA	2000	80	0.08	0.29	7.5	88.8	74.9	86.0
BSA	2000	100	0.05	0.30	9.3	90.2	68.8	86.0
CS	1500	60	0.10	0.22	8.6	89.4	71.4	85.8
BSA	1500	100	0.08	0.33	7.1	87.9	76.3	85.6
b6e6rl	1500	40	0.04	0.34	9.4	89.5	68.6	85.3
BSA	2500	80	0.05	0.32	9.5	89.4	68.3	85.2
BSA	1000	80	0.13	0.36	3.8	84.7	87.2	85.2

Tablo 6.8. TDO - Her algoritma için en iyi toplam puan

Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
b6e6rl	1000	80	0.01	0.05	12.9	98.4	57.2	90.2
CS	2500	40	0.05	0.13	9.7	94.4	67.6	89.1
BSA	2000	60	0.08	0.27	5.7	89.0	80.9	87.4
FPA	2500	20	0.16	0.54	3.5	78.5	88.2	80.5
ABC	2000	60	0.24	0.56	6.9	74.1	77.0	74.6



Şekil 6.1. TDO - Her bir transformatör tipi için ve ortalama güvenilirlik puanları

6.3. Pratik Transformatör Tasarım Optimizasyonu (DTDO) Performans Testleri

6.3.1. Test yöntemi

Pratik TDO performans testleri için MATLAB 2014a kullanılarak hazırlanan yazılım, bir ana program, beş optimizasyon algoritmasının her biri için bir altprogram ve tüm algoritma altprogramları tarafından ortak kullanılan bir tasarım hesaplama altprogramından oluşur. Tüm algoritmalara ayırık tasarım değişkenleri kullanma özelliği eklenmiştir.

Performans testleri 2.60 GHz Intel Core i5-3320M CPU ve 4 GB RAM'e sahip olan standart bir dizüstü bilgisayar kullanılarak aşağıdaki çerçevede gerçekleştirilmiştir:

- Bölüm 6.1.'de verilen spesifikasyonlara sahip üç tip transformatör için
- Beş algoritma (ABC, b6e6rl, BSA, CS, FPA) kullanılarak
- Tasarım değişkenleri Tablo 5.3.'te verilenler olmak üzere
- Tasarım değişkenlerinin alt ve üst sınır değerleri incelenen üç tip transformatörden daha da geniş bir güç aralığındaki transformatörleri kapsayacak şekilde geniş tutulmuştur (Tablo 5.3.)

- e. Aşağıda verilen maksimum döngü sayıları ve popülasyon büyüklükleri için
 1. Maksimum döngü sayısı 1000, 1500, 2000, 2500
 2. Popülasyon büyüklüğü 20, 40, 60, 80, 100
- f. Algoritma kontrol parametreleri Bölüm 5.7.'ye göre
- g. Durdurma kriteri kullanılmamıştır
- h. Testler her bir transformatör tipi ve algoritma için 20 kez tekrarlanmıştır

Tasarım değişkenlerinden çekirdek saç genişliği ve AG sargısı iletken genişliği 10 mm varsayılan (default) adımlarla değişmektedir, ancak tasarım mühendisi farklı bir adım boyu seçebilir; adım boyu ondalık sayı dahi olabilir.

AG sargısı iletken kalınlığı tasarım değişkeninin ayırık değerleri 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0 ve 2.24 mm'dir.

YG sargısı iletken çapı tasarım değişkeninin ayırık değerleri 0.50, 0.56, 0.63, 0.71, 0.80, 0.90, 1.00, 1.12, 1.25, 1.40, 1.60, 1.80, 2.00, 2.24, 2.50, 2.80, 3.15, 3.55, 4.00, 4.50 ve 5.00 mm'dir.

Her algoritmanın en iyi sonucu vereceği maksimum döngü sayısı ve popülasyon büyüklüğü kombinasyonu farklı olabileceğinden performans testleri bu değerlerin toplam 20 farklı kombinasyonu için yapılmıştır.

6.3.2. Test sonuçları ve değerlendirme

160 kVA transformatör için b6e6rl, 400 kVA için BSA ve 630 kVA için CS algoritmaları ile elde edilen optimizasyon sonuçları sırasıyla Tablo 6.9., Tablo 6.10. ve Tablo 6.11.'de; tüm algoritmaları içeren sonuçlar da Ek C'de verilmiştir.

Tablo 6.9. DTDO - 160 kVA için b6e6rl algoritması optimizasyon sonuçları

Döngü	Pop	Min fval	Ave fval	Max fval	CPU sec	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
1000	20	3,586	3,569	3,620	3.0	0.48	1.42
1000	40	3,581	3,569	3,582	5.9	0.33	0.35
1000	60	3,578	3,569	3,582	9.0	0.24	0.35
1000	80	3,579	3,569	3,582	12.2	0.28	0.35
1000	100	3,580	3,569	3,582	15.3	0.32	0.35
1500	20	3,592	3,575	3,698	4.8	0.65	3.61
1500	40	3,578	3,569	3,582	9.0	0.26	0.35
1500	60	3,578	3,569	3,582	13.4	0.26	0.35
1500	80	3,576	3,569	3,582	18.5	0.21	0.35
1500	100	3,578	3,569	3,582	22.4	0.24	0.35
2000	20	3,586	3,578	3,631	6.0	0.46	1.74
2000	40	3,581	3,569	3,593	12.8	0.33	0.68
2000	60	3,575	3,569	3,582	18.2	0.18	0.35
2000	80	3,580	3,569	3,582	23.9	0.32	0.35
2000	100	3,578	3,569	3,582	30.6	0.27	0.35
2500	20	3,584	3,569	3,620	7.3	0.42	1.42
2500	40	3,580	3,569	3,591	14.9	0.30	0.62
2500	60	3,576	3,569	3,582	22.4	0.20	0.35
2500	80	3,579	3,569	3,582	30.6	0.27	0.35
2500	100	3,577	3,569	3,582	38.0	0.23	0.35

Tablo 6.10. DTDO - 400 kVA için BSA algoritması optimizasyon sonuçları

Döngü	Pop	Min fval	Ave fval	Max fval	CPU sec	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$
1000	20	6,449	6,428	6,516	0.8	0.33	1.37
1000	40	6,435	6,428	6,467	1.6	0.11	0.61
1000	60	6,433	6,428	6,464	2.3	0.07	0.56
1000	80	6,430	6,428	6,449	3.0	0.03	0.32
1000	100	6,430	6,428	6,432	3.7	0.03	0.06
1500	20	6,460	6,428	6,581	1.2	0.50	2.39
1500	40	6,438	6,428	6,534	2.3	0.15	1.65
1500	60	6,429	6,428	6,432	3.4	0.02	0.06
1500	80	6,431	6,428	6,449	4.6	0.05	0.32
1500	100	6,431	6,428	6,449	5.5	0.05	0.32
2000	20	6,472	6,428	6,716	1.6	0.68	4.49
2000	40	6,435	6,428	6,467	3.1	0.12	0.61
2000	60	6,429	6,428	6,432	4.5	0.02	0.06
2000	80	6,429	6,428	6,432	6.0	0.01	0.06
2000	100	6,429	6,428	6,432	7.4	0.02	0.06
2500	20	6,471	6,428	6,557	2.1	0.66	2.01
2500	40	6,440	6,428	6,497	4.0	0.18	1.07
2500	60	6,430	6,428	6,432	5.8	0.03	0.06
2500	80	6,429	6,428	6,432	7.7	0.02	0.06
2500	100	6,429	6,428	6,438	9.4	0.02	0.16

Tablo 6.11. DTDO - 630 kVA için CS algoritması optimizasyon sonuçları

Döngü	Pop	Min fval	Ave fval	Max fval	CPU sec	$\epsilon_{mean}\%$	$\epsilon_{max}\%$
1000	20	7,448	7,442	7,554	2.7	0.09	1.51
1000	40	7,442	7,442	7,442	5.4	0.00	0.00
1000	60	7,442	7,442	7,442	7.9	0.00	0.00
1000	80	7,442	7,442	7,442	10.1	0.00	0.00
1000	100	7,442	7,442	7,442	12.8	0.00	0.00
1500	20	7,453	7,442	7,554	4.1	0.15	1.51
1500	40	7,442	7,442	7,442	7.7	0.00	0.00
1500	60	7,442	7,442	7,442	11.7	0.00	0.00
1500	80	7,442	7,442	7,442	15.2	0.00	0.00
1500	100	7,442	7,442	7,442	19.6	0.00	0.00
2000	20	7,450	7,442	7,551	5.6	0.11	1.46
2000	40	7,442	7,442	7,442	10.6	0.00	0.00
2000	60	7,442	7,442	7,442	15.3	0.00	0.00
2000	80	7,442	7,442	7,442	20.2	0.00	0.00
2000	100	7,442	7,442	7,442	25.1	0.00	0.00
2500	20	7,453	7,442	7,551	6.6	0.14	1.46
2500	40	7,442	7,442	7,442	12.7	0.00	0.00
2500	60	7,442	7,442	7,442	19.2	0.00	0.00
2500	80	7,442	7,442	7,442	26.1	0.00	0.00
2500	100	7,442	7,442	7,442	31.9	0.00	0.00

Teorik TDO performans test sonuçları için uygulanan puanlama yöntemi pratik TDO performans test sonuçları için de uygulanmıştır. Buna göre, en iyi 20 güvenilirlik puanlarının listesi Tablo 6.12.'de, her bir algoritma için en iyi güvenilirlik puanı Tablo 6.13.'te; en iyi toplam puanların listesi Tablo 6.14.'te ve her bir algoritma için en iyi toplam puan Tablo 6.15.'te verilmiştir; bu tablolardaki ϵ_{mean} üç transformatör tipi için ortalama, ϵ_{max} ise en kötü değerlerdir. Şekil 6.2. her transformatör tipi için ve ortalama olarak algoritmaların en iyi güvenilirlik puanlarının karşılaştırmasını göstermektedir.

Pratik TDO performans testlerinde CS ve FPA algoritmalarının hemen hemen 100 güvenilirlik puanına erişmeleri bu iki algoritmanın DTDO problemi için garantili küresel yakınsama özelliğine sahip olduğunu göstermektedir. Diğer üç algoritmanın (b6e6rl, BSA, ABC) hem CS ve FPA algoritmaları ile, hem de kendi aralarında çok ciddi güvenilirlik puanı farkları vardır; bu algoritmaların kötü performansı özellikle 630 kVA transformatörde belirginleşmektedir, bu güç için ABC algoritması her bir döngü/popülasyonu için sıfır güvenilirlik puanı ile en kötü performans sergilemiştir.

BSA, FPA ve ABC algoritmaları b6e6rl ve CS'den daha hızlıdır ve bu üç algoritmanın hızı birbirine yakındır; b6e6rl ise en ağır çalışan algoritmadır.

Sonuç olarak, pratik TDO uyarlaması için hem CS, hem de FPA uygun algoritmalarıdır; diğer üç algoritma bu amaç için uygun değildir.

Tablo 6.12. DTDO - En iyi 20 güvenilirlik puanı

Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
CS	1000	60	0.00	0.01	7.1	99.6	76.2	94.9
FPA	2000	60	0.00	0.01	9.2	99.6	69.4	93.5
CS	1000	80	0.00	0.01	9.4	99.6	68.8	93.4
CS	1500	60	0.00	0.01	10.6	99.6	64.5	92.6
FPA	2500	60	0.00	0.01	11.6	99.6	61.4	91.9
CS	2500	40	0.00	0.01	11.8	99.6	60.7	91.8
CS	1000	100	0.00	0.01	11.8	99.6	60.7	91.8
CS	1500	80	0.00	0.01	13.9	99.6	53.6	90.4
CS	2000	60	0.00	0.01	14.1	99.6	53.1	90.3
FPA	2500	80	0.00	0.01	15.6	99.6	48.1	89.3
FPA	2000	100	0.00	0.01	15.6	99.6	47.8	89.2
CS	1500	100	0.00	0.01	17.6	99.6	41.4	87.9
CS	2500	60	0.00	0.01	17.6	99.6	41.3	87.9
CS	2000	80	0.00	0.01	18.5	99.6	38.3	87.3
FPA	2500	100	0.00	0.01	19.5	99.6	34.9	86.6
CS	2000	100	0.00	0.01	22.9	99.6	23.6	84.4
CS	2500	80	0.00	0.01	23.6	99.6	21.5	83.9
CS	2500	100	0.00	0.01	29.2	99.6	2.7	80.2
FPA	1500	60	0.00	0.01	6.8	99.6	77.3	95.1
FPA	1500	80	0.00	0.01	9.3	99.5	69.1	93.4

Tablo 6.13. DTDO - Her algoritma için en iyi güvenilirlik puanları

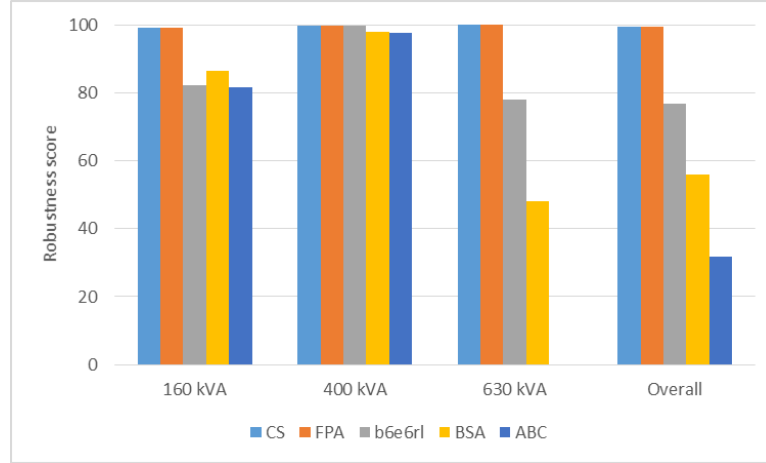
Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
CS	1000	60	0.00	0.01	7.1	99.6	76.2	94.9
FPA	2000	60	0.00	0.01	9.2	99.6	69.4	93.5
b6e6rl	1000	100	0.13	0.77	14.4	74.4	52.0	69.9
BSA	2000	100	0.16	1.49	7.9	54.9	73.8	58.7
ABC	1000	100	0.53	1.68	6.8	31.6	77.3	40.7

Tablo 6.14. DTDO - En iyi 20 toplam puan

Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
FPA	1500	60	0.00	0.01	6.8	99.6	77.3	95.1
CS	1000	60	0.00	0.01	7.1	99.6	76.2	94.9
FPA	2000	60	0.00	0.01	9.2	99.6	69.4	93.5
CS	1000	80	0.00	0.01	9.4	99.6	68.8	93.4
FPA	1500	80	0.00	0.01	9.3	99.5	69.1	93.4
CS	1500	60	0.00	0.01	10.6	99.6	64.5	92.6
FPA	2500	60	0.00	0.01	11.6	99.6	61.4	91.9
CS	2500	40	0.00	0.01	11.8	99.6	60.7	91.8
CS	1000	100	0.00	0.01	11.8	99.6	60.7	91.8
CS	1000	40	0.01	0.26	4.8	93.1	84.0	91.3
FPA	1500	100	0.01	0.08	11.6	97.6	61.4	90.4
CS	1500	80	0.00	0.01	13.9	99.6	53.6	90.4
CS	2000	60	0.00	0.01	14.1	99.6	53.1	90.3
FPA	1500	40	0.02	0.35	4.6	90.5	84.7	89.3
FPA	2500	80	0.00	0.01	15.6	99.6	48.1	89.3
FPA	2000	100	0.00	0.01	15.6	99.6	47.8	89.2
CS	1500	40	0.01	0.31	7.2	91.9	76.1	88.7
FPA	1000	40	0.06	0.35	3.1	88.0	89.8	88.4
FPA	2000	40	0.02	0.35	6.1	90.2	79.5	88.1
CS	1500	100	0.00	0.01	17.6	99.6	41.4	87.9

Tablo 6.15. DTDO - Her algoritma için en iyi toplam puanlar

Algo	Döngü	Pop	$\epsilon_{\text{mean}}\%$	$\epsilon_{\text{max}}\%$	CPUsec	Güven	Hız	Toplam
FPA	1500	60	0.00	0.01	6.8	99.6	77.3	95.1
CS	1000	60	0.00	0.01	7.1	99.6	76.2	94.9
b6e6rl	1000	100	0.13	0.77	14.4	74.4	52.0	69.9
BSA	1500	100	0.16	1.49	5.9	54.9	80.5	60.0
ABC	1000	100	0.53	1.68	6.8	31.6	77.3	40.7



Şekil 6.2. DTDO - Her bir transformatör tipi için ve ortalama güvenilirlik puanları

6.4. Teorik ve Pratik TDO Çözümlerinin Karşılaştırılması

Bu bölümde incelenen üç tip transformatör için hem teorik, hem de pratik TDO çalışması yapılmıştır. Test yöntemi aşağıdaki farklılıklar dışında Bölüm 6.2.1. ve 6.3.1.'de verildiği gibidir.

- Çalışmada sadece CS algoritması kullanılmıştır
- Maksimum döngü sayısı teorik TDO için 2500 ve pratik TDO için 1000; popülasyon büyüklüğü her iki yöntem için de 40 olarak seçilmiştir
- Sargı soğutma kanal sayıları sabit tutulmuştur; başka bir deyişle optimizasyon 6 tasarım değişkeni kullanılarak yapılmıştır
- Testler her bir transformatör tipi ve TDO yöntemi için 100 kez tekrarlanmıştır
- Test çalıştırmaları MATLAB 2016b ortamında yapılmıştır

Elde edilen sonuçlar en iyi çözümlerin tasarım değişkeni değerleri ile birlikte Tablo 6.16.'da verilmiştir.

Tablo 6.16. 630 kVA teorik ve pratik TDO çözümlerinin karşılaştırılması

İstatistikler	Birim	160 kVA		400 kVA		630 kVA	
		TDO	DTDO	TDO	DTDO	TDO	DTDO
En iyi çözüm	USD	3,531	3,569	6,354	6,428	7,402	7,442
Ortalama çözüm	USD	3,532	3,569	6,354	6,428	7,402	7,442
En kötü çözüm	USD	3,534	3,569	6,359	6,428	7,402	7,442
Standart sapma		0.387	0.000	0.842	0.000	0.040	0.000
Ortalama süre	Sn	4.0	2.4	4.3	2.3	4.9	2.4
DTDO/TDO maliyet farkı	%		1.076		1.165		0.540
Tasarım değişkenleri							
AG sargısı sarım sayısı		28	27	18	18	15	14
Manyetik endüksiyon	Gauss	16,780	16,855	16,597	16,610	17,500	17,453
Çekirdek sac genişliği	mm	196.1	220.0	248.9	260.0	268.4	280.0
AG sargısı iletken genişliği	mm	197.9	180.0	239.0	240.0	229.9	210.0
AG sargısı iletken kalınlığı	mm	0.383	0.400	0.772	0.700	0.907	1.000
YG sargısı iletken çapı	mm	0.968	1.000	1.534	1.600	1.649	1.600
Hesaplanan kısıtlar							
Boşta kayıplar	W	489	489	863	862	1,214	1,247
Yük kayıpları	W	2,564	2,530	5,023	5,021	9,786	9,753
Toplam kayıplar	W	3,053	3,019	5,885	5,884	11,000	11,000
Kısa devre empedansı	%	4.068	4.397	4.266	4.318	5.526	5.468

TDO = Teorik TDO, DTDO = Pratik TDO

Performans testleri, CS algoritmasının karma-tamsayı TDO problemi için 0.4'lük bir ortalama standart sapma değeri ile çok güvenilir sonuçlar verdiğini açıkça göstermiştir. Pratik TDO performans testlerinde ise, CS algoritması test edilen her üç tip transformatör için de sıfır standart sapma ile mükemmellik ölçeğine ulaşmaktadır; bu sonuç aynı zamanda CS algoritmasının Bölüm 3.4.'te belirtildiği gibi garanti edilmiş küresel yakınsama özelliğine sahip olduğunu doğrulamaktadır.

En iyi DTDO çözümlerinin en iyi TDO çözümlerine kıyasla % 1'den az olan ortalama maliyet artışı oldukça kabul edilebilir düzeydedir.

160 ve 630 kVA transformatörler için, transformatör tasarımında önemli bir parametre olan AG sargısı sarım sayısı, en iyi TDO çözümleri ile karşılaştırıldığında en iyi

DTDO çözümlerinde azalmıştır. Bunun boşta kayıpların artması ve kısa devre empedansındaki azalma gibi etkileri, AG iletkeninin genişliğini azaltma yoluyla çekirdek pencere yüksekliğini azaltarak dengelenmiştir. Bu da, kullanılan optimizasyon algoritmasının iyi işlediğinin doğrulanması olarak düşünülebilir.

6.5. Önbelleklemeli Pratik TDO

Bölüm 6.3.'te sonuçları verilen pratik TDO performans testlerinde Bölüm 4.'te anlatılan önbellekleme yöntemi kullanılmamıştı. Bu bölümde önbellekleme yöntemi ile amaç fonksiyonu hesaplama sayısında elde edilebilecek tasarruflar incelenmiştir. Test yöntemi aşağıdaki farklılıklar dışında Bölüm 6.4.'te verildiği gibidir.

- a. Çalışmada CS ve FPA algoritmaları kullanılmıştır (bu iki algoritmaya hash tablolarını uyarlamak için MATLAB'ın *Map Containers* özelliği eklenmiştir)
- b. Maksimum döngü sayısı 1000, popülasyon büyüklüğü 40 olarak seçilmiştir
- c. Önceki performans testlerinde kullanılan tasarım değişkeni geniş sınır değerleri seçeneğine ek olarak, deneyimli bir tasarım mühendisinin tasarım değişkenlerini optimum değerlerin % 25 aralığında tahmin edebileceği varsayımından hareketle dar sınır değerleri seçeneği oluşturulmuştur
- d. Testler her bir transformatör tipi, algoritma ve tasarım değişkeni geniş ve dar sınır değerleri seçeneği için 20 kez tekrarlanmıştır
- e. Test çalıştırmaları MATLAB 2016b ortamında yapılmıştır

CS algoritması için elde edilen sonuçlar Tablo 6.17.'de, FPA için elde edilen sonuçlar da Tablo 6.18.'de verilmiştir.

Tablo 6.17. CS için amaç fonksiyonu hesaplama sayılarının karşılaştırılması

	160 kVA	400 kVA	630 kVA	Ortalama	
				Sayı	Süre (*)
Önbelleklemesiz	80,000	80,000	80,000	80,000	3.5
Önbellekleme ile	27,580	25,408	20,911	24,633	9.4
Önbellekleme ve dar sınırlar ile	19,453	18,264	13,280	16,999	10.1

(*) Süre birimi saniyedir

Tablo 6.18. FPA için amaç fonksiyonu hesaplama sayılarının karşılaştırılması

	160 kVA	400 kVA	630 kVA	Ortalama	
				Sayı	Süre (*)
Önbelleklemesiz	40,000	40,000	40,000	40,000	1.6
Önbellekleme ile	24,240	22,759	21,382	22,794	5.2
Önbellekleme ve dar sınırlar ile	16,353	15,540	12,397	14,763	4.2

(*) Süre birimi saniyedir

CS algoritmasının önbelleklemesiz amaç fonksiyonu hesaplama sayısının FPA'ya kıyasla iki katı olması Bölüm 3.4.'te belirtildiği gibi bu algoritmanın her iterasyonda amaç fonksiyonunu iki kez hesaplamasından kaynaklanmaktadır.

Önbellekleme kullanımı CS ve FPA algoritmaları için amaç fonksiyonu hesaplama sayısında sırasıyla % 69 ve % 44, önbellekleme ile birlikte dar tasarım değişkeni sınırları kullanıldığında da % 79 ve % 63 tasarruf sağlamaktadır; önbellekleme kullanıldığında iki algoritmanın hesaplama sayılarının % 10 civarında bir farkla birbirine yakın olduğu söylenebilir.

Önbellekleme önemli sayılabilecek bir işlem yükü gerektirdiğinden TDO problemi için aynı algoritmanın önbelleklemesiz kullanımından daha uzun süre alabilmektedir; öte yandan, yukarıdaki tablolarda görülebileceği gibi, önbellekli FPA önbelleksiz CS yerine rahatlıkla kullanılabilir. Önbellek kullanımının sağlayacağı başka bir yarardan bir sonraki bölümde bahsedilecektir.

6.6. Grafik Kullanıcı Arayüzlü TDO Programı

Bu tez çalışmasında elde edilen sonuçlar esas alınarak temel özellikleri aşağıda verilen grafik kullanıcı arayüzlü bir TDO programı hazırlanmıştır.

- a. Program hem teorik, hem de pratik optimizasyon yapabilmektedir
- b. Teorik optimizasyon için b6e6rl, pratik optimizasyon için önbelleklemeli CS algoritması kullanılmıştır
- c. Maksimum döngü sayısı/popülasyon büyüklüğü değerleri b6e6rl algoritması için 1500/100, CS algoritması için 1000/40 olarak seçilmiştir
- d. Program en iyi 500 uygulanabilir çözümü artan amaç fonksiyonu değeri sırasında listeleyebilmektedir
- e. Bulunan tüm uygulanabilir pratik TDO çözümleri Excel tablosuna aktarılabilmektedir

Kullanıcı arayüzü örnekleri teorik TDO için Şekil 6.3., pratik TDO için de Şekil 6.4.'te verilmiştir.

Pratik TDO için önbellekleme kullanıldığında Excel'e aktarılan uygulanabilir çözümler içinde aynı tasarım vektörü sadece bir kez bulunabilecektir; bu özellik tasarım mühendisine çeşitli filtrelemeler yaparak alternatif çözümleri araştırma açısından kolaylık sağlayacaktır. Artan amaç fonksiyonu değeri sırasında Excel çıktı tablo örneği Şekil 6.5.'te verilmiştir.

TDO - TRANSFORMER DESIGN OPTIMIZATION

Specifications

Design ID: 400 kVA

Power rating (kVA): 400

Primary voltage (V): 20000

Secondary voltage (V): 400

Frequency (Hz): 50

HV Connection: Delta

LV Connection: Wye

HV Conductor material: Copper

LV Conductor material: Copper

HV Conductor type, insulation: Round, enamel

LV Conductor type: Foil

Core material: MS

NLTC Taps: 5 Pos @ -2, -2 x 2.5%

Tank type: With oil conservator

Guaranteed values

Applicable standard: Georgiakis book

No-load losses (W) / +Tol%: 750 / 15

Load losses (W) / +Tol (%): 4600 / 15

Total losses (W) / +Tol (%): 5350 / 10

Impedance (%) / +Tol (%): 4 / 10

Max core sound level (dB): 99

Winding temp rise limit (K): 60

Top oil temp rise limit (K): 55

Optimization

Minimize: Material cost

Optimization method: Theoretical

Design variables

Design variable	Min	Max	Step
LV Winding no of turns	10	50	
Magnetic induction (G)	14000	17500	
Width of core leg (mm)	100	500	10
LV Wdg elect height (mm)	100	500	10
LV Cond thickness (mm)	0.20	2.00	
HV Cond diameter (mm)	0.50	2.50	
LV Curr density (A/mm ²)	1.0	5	
HV Curr density (A/mm ²)	1.0	5	
LV Winding no of ducts	0	12	
HV Winding no of ducts	0	12	
TOC A&B factors (\$/W)			

Reset previous results

Run optimization

Number of design calculations: 150000

Number of feasible solutions: 26207

Calculation time (seconds): 10

Export to Excel

End session

Optimization results (best 500)

	Cost	LV Turns	Induction	Core width	LV height	tLV	dHV	Ducts LV	Ducts HV	JLV	JHV	Po	Pk	Total losses	Uk	Iteration
1	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	141766
2	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	139244
3	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	137444
4	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	136997
5	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	131897
6	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	131447
7	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	127737
8	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	121137
9	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	120588
10	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	120250
11	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	120100
12	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	118600
13	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	117091
14	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	116905
15	6353.78	18	16597	248.8510	238.9619	0.7718	1.5344	2	4	3.1306	3.6055	862.50	5022.50	5885.00	4.2659	115014

Şekil 6.3. 400 kVA Teorik TDO kullanıcı arayüzü örneği

TDO - TRANSFORMER DESIGN OPTIMIZATION

Specifications

Design ID: 400 kVA

Power rating (kVA): 400

Primary voltage (V): 20000

Secondary voltage (V): 400

Frequency (Hz): 50

HV Connection: Delta

LV Connection: Wye

HV Conductor material: Copper

LV Conductor material: Copper

HV Conductor type, insulation: Round, enamel

LV Conductor type: Foil

Core material: MS

NLTC Taps: 5 Pos @ -2, -2 x 2.5%

Tank type: With oil conservator

Guaranteed values

Applicable standard: Georgiakis book

No-load losses (W) / +Tol%: 750 / 15

Load losses (W) / +Tol (%): 4600 / 15

Total losses (W) / +Tol (%): 5350 / 10

Impedance (%) / +Tol (%): 4 / 10

Max core sound level (dB): 99

Winding temp rise limit (K): 60

Top oil temp rise limit (K): 55

Optimization

Minimize: Material cost

Optimization method: Practical

Design variables

Design variable	Min	Max	Step
LV Winding no of turns	10	50	
Magnetic induction (G)	14000	17500	
Width of core leg (mm)	100	500	10
LV Wdg elect height (mm)	100	500	10
LV Cond thickness (mm)	0.20	2.00	
HV Cond diameter (mm)	0.50	2.50	
LV Curr density (A/mm ²)	1.0	5	
HV Curr density (A/mm ²)	1.0	5	
LV Winding no of ducts	0	12	
HV Winding no of ducts	0	12	
TOC A&B factors (\$/W)			

Reset previous results

Run optimization

Number of design calculations: 36169

Number of feasible solutions: 5781

Calculation time (seconds): 8

Export to Excel

End session

Optimization results (best 500)

	Cost	LV Turns	Induction	Core width	LV height	tLV	dHV	Ducts LV	Ducts HV	JLV	JHV	Po	Pk	Total losses	Uk	Iteration
1	6428.33	18	16610	260	240	0.7000	1.6000	2	4	3.4366	3.3157	862.48	5021.10	5883.58	4.3179	27995
2	6428.58	18	16609	260	240	0.7000	1.6000	2	4	3.4366	3.3157	862.36	5021.10	5883.46	4.3179	26349
3	6428.83	18	16608	260	240	0.7000	1.6000	2	4	3.4366	3.3157	862.24	5021.10	5883.34	4.3179	27346
4	6429.08	18	16607	260	240	0.7000	1.6000	2	4	3.4366	3.3157	862.12	5021.10	5883.22	4.3179	25157
5	6429.32	18	16606	260	240	0.7000	1.6000	2	4	3.4366	3.3157	861.99	5021.10	5883.09	4.3179	26192
6	6429.57	18	16605	260	240	0.7000	1.6000	2	4	3.4366	3.3157	861.87	5021.10	5882.97	4.3179	27066
7	6429.82	18	16604	260	240	0.7000	1.6000	2	4	3.4366	3.3157	861.75	5021.10	5882.85	4.3179	23848
8	6430.07	18	16603	260	240	0.7000	1.6000	2	4	3.4366	3.3157	861.63	5021.10	5882.73	4.3179	26895
9	6430.29	18	16610	260	240	0.7000	1.6000	2	3	3.4366	3.3157	862.48	5012.95	5875.43	4.2792	29305
10	6430.32	18	16602	260	240	0.7000	1.6000	2	4	3.4366	3.3157	861.51	5021.10	5882.61	4.3179	29320
11	6430.54	18	16609	260	240	0.7000	1.6000	2	3	3.4366	3.3157	862.36	5012.95	5875.30	4.2792	27465
12	6430.57	18	16601	260	240	0.7000	1.6000	2	4	3.4366	3.3157	861.39	5021.10	5882.49	4.3179	25006
13	6430.79	18	16608	260	240	0.7000	1.6000	2	3	3.4366	3.3157	862.24	5012.95	5875.18	4.2792	28343
14	6430.81	18	16600	260	240	0.7000	1.6000	2	4	3.4366	3.3157	861.27	5021.10	5882.37	4.3179	23785
15	6431.03	18	16607	260	240	0.7000	1.6000	2	3	3.4366	3.3157	862.12	5012.95	5875.06	4.2792	30451

Şekil 6.4. 400 kVA Pratik TDO kullanıcı arayüzü örneği

Mat Cost	LV Turns	Induction	Core width	LV height	tLV	dHV	Ducts LV	Ducts HV	JLV	JHV	Po	Pk	Total loss	Uk	Iteration
6428.33	18	16610	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5884	4.318	28410
6428.58	18	16609	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	23556
6428.83	18	16608	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	23309
6429.08	18	16607	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	23148
6429.32	18	16606	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	26677
6429.57	18	16605	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	24984
6429.82	18	16604	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	20590
6430.07	18	16603	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	23156
6430.29	18	16610	260	240	0.7	1.6	2	3	3.437	3.316	862	5013	5875	4.279	30449
6430.32	18	16602	260	240	0.7	1.6	2	4	3.437	3.316	862	5021	5883	4.318	20655
6430.54	18	16609	260	240	0.7	1.6	2	3	3.437	3.316	862	5013	5875	4.279	24371
6430.57	18	16601	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	20241
6430.79	18	16608	260	240	0.7	1.6	2	3	3.437	3.316	862	5013	5875	4.279	32272
6430.81	18	16600	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	24790
6431.03	18	16607	260	240	0.7	1.6	2	3	3.437	3.316	862	5013	5875	4.279	28129
6431.06	18	16599	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	22801
6431.28	18	16606	260	240	0.7	1.6	2	3	3.437	3.316	862	5013	5875	4.279	28008
6431.31	18	16598	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	28118
6431.56	18	16597	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	22002
6431.78	17	16426	280	230	0.7	1.6	3	4	3.586	3.316	862	5014	5877	4.149	19861
6431.78	18	16604	260	240	0.7	1.6	2	3	3.437	3.316	862	5013	5875	4.279	29118
6431.81	18	16596	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	24229
6432.04	17	16425	280	230	0.7	1.6	3	4	3.586	3.316	862	5014	5876	4.149	20421
6432.06	18	16595	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	20196
6432.27	18	16602	260	240	0.7	1.6	2	3	3.437	3.316	862	5013	5874	4.279	24305
6432.30	17	16424	280	230	0.7	1.6	3	4	3.586	3.316	862	5014	5876	4.149	20670
6432.31	18	16594	260	240	0.7	1.6	2	4	3.437	3.316	861	5021	5882	4.318	19942
6432.56	18	16593	260	240	0.7	1.6	2	4	3.437	3.316	860	5021	5882	4.318	19804
6432.56	17	16423	280	230	0.7	1.6	3	4	3.586	3.316	862	5014	5876	4.149	19434
6432.80	18	16592	260	240	0.7	1.6	2	4	3.437	3.316	860	5021	5881	4.318	19630

Şekil 6.5. 400 kVA Pratik TDO Excel çıktı örneği (5134 uygulanabilir çözüm)

BÖLÜM 7. SONUÇLAR VE ÖNERİLER

Bu tez çalışması kapsamında TDO probleminin çözümü için yaklaşık 20 kadar modern metasezgisel optimizasyon algoritması performans testleri yapılarak incelenmiş ve bunlar arasında en uygun çözümleri üreten beş algoritmanın sonuçları burada sunulmuştur. Bu beş algoritma, Yapay Arı Kolonisi (ABC), Geri-İzleme Arama Optimizasyon (BSA), Rekabetçi-Uyarlamalı Diferansiyel Gelişim (b6e6rl), Guguk Kuşu Arama (CS), ve Çiçek Tozlaşma (FPA) algoritmalarıdır.

Algoritmaların karşılaştırılması için pragmatik bir kıyaslama yöntemi geliştirilmiştir. Bu yöntem oldukça basittir ve farklı ihtiyaçlara kolayca adapte edilebilir.

Üç farklı güçte dağıtım transformatörü kullanılarak yapılan performans testlerinin sonuçları Rekabetçi-Uyarlamalı Diferansiyel Gelişim (b6e6rl) algoritmasının, her çalıştırmada 30 saniyeden daha kısa bir sürede % 99.9'dan daha fazla doğruluk ve hassasiyetle sonuç ürettiğini kanıtlamıştır. Bundan başka, Bölüm 3.4.'te belirtildiği gibi, garantili küresel yakınsama özelliğine sahip olduğu matematiksel olarak gösterilmiş olan CS algoritmasının performansının b6e6rl'ninkine çok yakın olduğu belirlenmiştir. Bu çalışmada yapılan performans testlerinin sonuçları garantili küresel yakınsama özelliğinin geçerliliğini doğrulamaktadır; bu nedenle hem b6e6rl ve hem de CS algoritması TDO probleminin çözümü için deterministik yöntemler yerine geçerli bir alternatif olarak düşünülebilir.

TDO problemi için kullanılan tasarım değişkenlerinin bir kısmı veya tamamı, genellikle mevcut TDO literatüründeki gibi sürekli olduğunda, ortaya çıkan optimum çözüm standart dışı boyutlar nedeniyle teorik bir çözüm olacaktır; ve dolayısıyla tasarım mühendisinin teorik çözümü uygulanabilir bir çözüme dönüştürmek için ek çaba harcaması gerekir. Bu çalışmada anılan soruna çözüm olarak piyasada mevcut

veya üretime uygun boyutlara sahip ve dolayısıyla doğrudan üretime uygulanabilen çözümler üreten bir ayrık transformatör tasarım optimizasyonu yöntemi önerilmiştir.

Bu çalışmada karma-tamsayı tipi tasarım değişkenleri ile TDO problemlerinin çözülmesi teorik TDO olarak, ayrık tasarım değişkenleri ile çözülmesi de pratik TDO (veya DTDO) olarak ifade edilmiştir.

Üç farklı güçte dağıtım transformatörü kullanılarak yapılan performans testlerinin sonuçları hem CS hem de FPA algoritmasının pratik TDO problemi için 15 saniyenin altında gayet makul bir süre alan tek bir çalıştırmada sıfır standart sapmalı çözümler ürettiklerini, böylelikle garantili küresel yakınsama özelliklerinin geçerliliğini kanıtlamıştır.

Sonuçlar ayrıca pratik TDO kullanılarak, teorik TDO ile bulunan en iyi çözümlerden kabul edilebilir bir maliyet sapması ile, uygulanabilir çözümlerin elde edildiğini göstermiştir.

Amaç fonksiyonu hesaplamalarını azaltmak için önbellekleme tekniğinin tercihen dar sınır koşullarıyla birlikte kullanılması önerilir. Önbellekleme işlemi tarafından getirilen ek işlem yükü nedeniyle optimizasyon çalışma süresi kısalmasa dahi, önbellekleme kullanıldığında elde edilebilecek sadece tekil tasarım vektörleri içeren uygulanabilir tüm çözümler listesi tasarım mühendisinin alternatif çözümler araştırması sürecinde yardımcı olacaktır.

Teorik ve pratik optimizasyon yöntemlerinin birlikte kullanılması gerektiğine dikkat edilmelidir; bir tasarım mühendisi, önce teorik optimum çözümün nerede olduğunu belirlemeli ve sonra da pratik optimizasyonu kullanarak teorik optimuma olabildiğince yaklaştırmaya çalışmalıdır. Bunun nedeni pratik optimizasyonun bazen teorik optimumdan çok uzak bir yerel optimumda takılabilme olasılığı olmasıdır, ve teorik optimizasyonu kullanmadan tasarım mühendisi böyle bir durumun farkında olmayabilir. Kısacası, teorik ve pratik optimizasyon yöntemleri bir transformatör tasarım optimizasyonu sürecinde birbirlerini tamamlamalıdır. Bu öneriye uygun

olarak, Bölüm 6.6.'da kısaca bahsedildiği gibi, teorik ve pratik tasarım optimizasyonunu gerçekleştirebilecek şekilde grafiksel kullanıcı arayüzüne sahip bir program geliştirilmiştir.

Aşağıdakiler gelecekteki araştırma konuları olarak düşünülebilir.

- a. 2D/3D sonlu elemanlar modelleme (FEM) gibi ileri teknikler kullanılarak sıcaklık dağılımı, kısa devre empedansı, girdap (eddy) kayıpları vb. gibi hesaplamaların daha hassas bir şekilde yapılabilmesi
- b. Program yürütme hızını artırmak için,
 1. Paralel işleme kullanımı
 2. MATLAB yerine derleyicili bir programlama dili kullanımı
 3. MATLAB'ın *Map Containers* özelliğine alternatif olarak, çok daha hızlı olduğu iddia edilen Java Collections hash tablolarının kullanılması

Bu tezin, dağıtım transformatörü endüstrisinde teklif ve tasarım süreçlerinin kolaylaştırılmasına ve aynı zamanda üretim malzemelerinin standardize edilmesine önemli katkısı olacağı öngörülmektedir.

KAYNAKLAR

- [1] Georgilakis, P. S., Spotlight on Modern Transformer Design. Springer-Verlag London, 45-122, 2009.
- [2] PROPHET II: The potential for global energy savings from high efficiency distribution transformers. Final report, Prepared for the European Copper Institute by Waide Strategic Efficiency Limited and N14 Energy Limited, 2014.
- [3] Distribution Transformers - A Global Strategic Market Report. Global Industry Analysts, Inc., 2014.
- [4] Amoiralis, E. I., Tsili, M. A., Papparigas, D. G., Kladas, A. G., Global transformer design optimization using deterministic and nondeterministic algorithms. IEEE Transactions on Industry Applications, 50(1): 383-394, 2014.
- [5] Olivares-Galvan, J. C., Georgilakis, P. S., Fofana, I., Magdaleno-Adame, S., Campero-Littlewood, E., Esparza-Gonzalez, M. S., A bibliographic analysis of transformer literature 2001-2010. 8th Mediterranean Conference on Power Generation, Transmission, Distribution and Energy Conversion (MEDPOWER 2012), 2012.
- [6] Amoiralis, E. I., Tsili, M. A., Georgilakis, P. S., The state of the art in engineering methods for transformer design and optimization: a survey. Journal of Optoelectronics and Advanced Materials, 10(5): 1149-1158, 2008.
- [7] Amoiralis, E. I., Tsili, M. A., Kladas, A. G., Transformer design and optimization: a literature survey. IEEE Transactions on Power Delivery, 24(4): 1999-2024, 2009.
- [8] Ajay Khatri, A., Rahi, O. P., Optimal design of transformer: a compressive bibliographical survey. International Journal of Scientific Engineering and Technology, 1(2): 159-167, 2012.
- [9] Mehta, H. D., Patel, R. M., A review on transformer design optimization and performance analysis using artificial intelligence techniques. International Journal of Science and Research (IJSR), 3(9): 726-733, 2014.

- [10] Omorogiuwa E., Oboma S. O., A review of intelligent based optimization techniques in power transformer design. *Applied Research Journal*, 1(2): 80-90, 2015.
- [11] Georgilakis, P. S., Tsili, M. A., Souflaris, A. T., A heuristic solution to the transformer manufacturing cost optimization problem. *J Mater Process Tech*, 181: 260-266, 2007.
- [12] Mehta, H. D., Patel, R., Optimal design of transformer using tournament selection based elitist genetic algorithms. *Indian Journal of Science and Technology*, 8(12): 1-9, 2015.
- [13] Zhang, Z., Pan, Z., Pan, X., A regression analysis based heuristic method to the power transformer design optimization problem. *International Conference on Electrical Machines and Systems (ICEMS 2011)*, Beijing, China, 2011.
- [14] Azizian, D., Bigdeli, M., Faiz, J., Design optimization of cast-resin transformer using nature-inspired algorithms. *Arabian Journal for Science and Engineering*, 41(9): 3491-3500, 2016.
- [15] Zhang, S., Hu, O., Wang, X., Wang, D., Research of transformer optimal design modeling and intelligent algorithm. *2011 Chinese Control and Decision Conference (CCDC)*, Mianyang, China, 2011.
- [16] Tamilselvi, S., Baskar, S., Modified parameter optimization of distribution transformer design using covariance matrix adaptation evolution strategy. *Int J Elec Power*, 61: 208–218, 2014.
- [17] Karaboğa, D., An idea based on honey bee swarm for numerical optimization. *Technical Report TR06*, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
- [18] Baştürk, B., Karaboğa, D., An artificial bee colony (ABC) algorithm for numeric function optimization. *IEEE Swarm Intelligence Symposium 2006*, Indianapolis, Indiana, USA, 2006.
- [19] Karaboğa, D., Akay, B., A modified Artificial Bee Colony (ABC) algorithm for constrained optimization problems. *Appl Soft Comput*, 11: 3021-3031, 2011.
- [20] Karaboğa, D., Akay, B., ABC Algorithm Source Code by Delphi for Constrained Optimization. *Artificial Bee Colony (ABC) Algorithm Homepage*. <http://mf.erciyes.edu.tr/abc/software.htm>, Erişim Tarihi: 30.04.2017.

- [21] Tvrđík, J., Self-adaptive variants of differential evolution with exponential crossover. *Annals of the West University of Timisoara: Mathematics and Computer Science*, 47 (2): 151-168, 2009.
- [22] Tvrđík, J., Pavliska, V., Valášek, R., Poláková, R., Habiballa, H., Self-adaptive algorithms for global optimization – MATLAB and C++ library including parallel implementation for many processors.
http://irafm.osu.cz/sago/matlab_routines-DE.html, Erişim Tarihi: 30.04.2017.
- [23] Çivicioğlu, P., Backtracking Search Optimization Algorithm for numerical optimization problems. *Appl Math Comput*, 219: 8121-8144, 2013.
- [24] Çivicioğlu, P., Backtracking Search Optimization Algorithm. MathWorks MATLAB Central.
<https://de.mathworks.com/matlabcentral/fileexchange/44842-backtracking-search-optimization-algorithm>, Erişim Tarihi: 30.04.2017.
- [25] Yang, X.-S., *Nature-Inspired Optimization Algorithms*. Elsevier, Amsterdam, The Netherlands, 2014.
- [26] Wang, F., He, X.-S., Wang, Y., Yang, S. M., Markov model and convergence analysis based on cuckoo search algorithm (in Chinese). *Computer Engineering*, 38(11): 180-185, 2012.
- [27] Yang, X.-S., *Nature-Inspired Metaheuristic Algorithms*. Luniver Press, Frome, UK, 2010.
- [28] Yang, X.-S., Flower Pollination Algorithm. MathWorks MATLAB Central.
<https://de.mathworks.com/matlabcentral/fileexchange/45112-flower-pollination-algorithm>, Erişim Tarihi: 30.04.2017.
- [29] Yang, X.-S., *Metaheuristic optimization: algorithm analysis and open problems*. Proceedings of 10th International Symposium on Experimental Algorithms (SEA 2011), Crete, Greece, 2011.
- [30] Zielinski, K., Peters, D., Laur, R., Stopping criteria for single-objective optimization. Proceedings of the Third International Conference on Computational Intelligence, Robotics and Autonomous Systems, Singapore, 2005.
- [31] Zielinski, K., Laur, R., Constrained single-objective optimization using particle swarm optimization. Proceedings of the IEEE Congress on Evolutionary Computation, Vancouver, BC, Canada, 2006.

- [32] Zielinski, K., Laur, R., Stopping criteria for a constrained single-objective particle swarm optimization algorithm. *Informatica*, 31: 51-59, 2007.
- [33] Gandomi, A. H., Yang, X.-S., Alavi, A. H., Mixed variable structural optimization using Firefly Algorithm. *Comput Struct*, 89: 2325-2336, 2011.
- [34] Shivakumar, S. K., Caching for enterprise web application. İçinde: *Architecting High Performing, Scalable and Available Enterprise Web Applications*. Morgan Kaufmann, Waltham, MA, USA, 2015.
- [35] Kücükıymaz, T., Cambazoglu, B. B., Aykanat, C., Baez-Yates, R., A machine learning approach for result caching in web search engines. *Inform Process Manag*, 53: 834-850, 2017.
- [36] Ahn, W. H., Kim, W. J., Park, D., Content-aware cooperative caching for cluster-based web servers. *J Syst Software*, 69: 75-86, 2004.
- [37] Forisek, M., Towards a better way to teach dynamic programming. *Olympiads in Informatics*, 9: 45-55, 2015.
- [38] Jaffar, J., Santosa, A. E., Voicu, R., Efficient memoization for dynamic programming with ad-hoc constraints. *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, Chicago, IL, USA, 2008.
- [39] Kratica, J., Improving performances of the genetic algorithm by caching. *Comput Artif Intell*, 18: 271-283, 1999.
- [40] Maric, M., An efficient genetic algorithm for solving the multi-level uncapacitated facility location problem. *Comput Inform*, 29: 183-201, 2010.
- [41] Kratica, J., Kostic, T., Tomic, D., Dugosija, D., Filipovic, V., A genetic algorithm for the routing and carrier selection problem. *Comput Sci Inf Syst*, 9: 49-62, 2012.
- [42] Grill, B., Ekelhart, A., Kiesling, E., Stummer, C., Strauss, C., Multi-objective evolutionary optimization of computation-intensive simulations – the case of security control selection. *MIC 2015: The XI Metaheuristics International Conference*, Agadir, Morocco, 2015.
- [43] Brownlee, A. E. I., Wright, J. A., Constrained, mixed-integer and multi-objective optimisation of building designs by NSGA-II with fitness approximation. *Appl Soft Comput*, 33: 114-126, 2015.

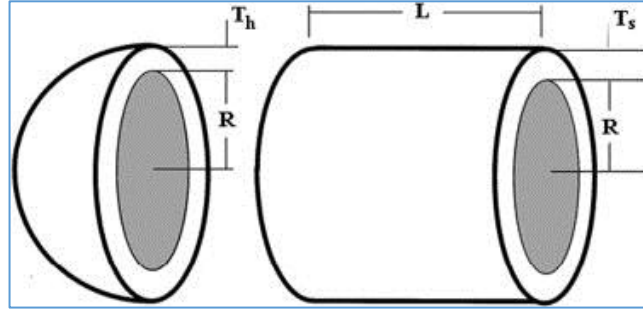
- [44] Tyni, T., Ylinen, J., Bi-directional circular linked lists in fitness caching. Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation (GECCO'02), San Francisco, CA, USA, 2002.
- [45] Handley, S., On the use of a directed acyclic graph to represent a population of computer programs. Proceedings of the IEEE World Congress on Computational Intelligence, Orlando, FL, USA, 1994.
- [46] Ehrenburg, H., Improved directed acyclic graph evaluation and the combine operator in genetic programming. Proceedings of the 1st Genetic Programming Conference, Stanford University, CA, USA, 1996.
- [47] Roberts, M. A., The effectiveness of cost based sub-tree caching mechanisms in typed genetic programming for image segmentation. Proceedings of EvoIASP, Applications of Evolutionary Computation, Essex, UK, 2002.
- [48] Energy efficient transformer solutions, European Minimum Energy Performance Standard (MEPS). ABB Pamphlet No. 1LAB000572, 2014.
- [49] Georgilakis, P. S., Olivares, J. C., Esparza-Gonzalez, M. S., An evolutionary computation solution to transformer design optimization problem. 7th International Conference on Electrical and Electronics Engineering Research (CIIIEE 2010), Aguascalientes, Mexico, 226-231, 2010.
- [50] Amoiralis, E. I., Georgilakis, P. S., Tsili, M. A., Kladas, A. G., Souflaris, A. T., A complete software package for transformer design optimization and economic evaluation analysis. Materials Science Forum, 670: 535-546, 2011.
- [51] Kuşakçı, O. M., Can, M., Constrained optimization with evolutionary algorithms: a comprehensive review. Southeast Europe Journal of Soft Computing, 1(2): 16-24, 2012.
- [52] Francisco, R. B., Costa, M. F. P., Rocha, A. M. A. C., A firefly dynamic penalty approach for solving engineering design problems. AIP Conference Proceedings, 1648(1), 2015.
- [53] Gandomi, A. H., Yang, X.-S., Evolutionary Boundary Constraint Handling Scheme. Neural Comput & Applic, 21: 1449-1462, 2012.

EKLER

EK A: Kıyaslama Problemleri

A.1. Basınçlı Kap Tasarımı

Basınçlı kap tasarım probleminin amacı, Şekil A.1.'de gösterilen bir silindirik kabın toplam üretim maliyetini en aza indirmektir. Dört tasarım değişkeni gövde kalınlığı T_s , kafa kalınlığı T_h , iç yarıçap R ve silindirik bölüm uzunluğu L 'dir.



Şekil A. 1. Basınçlı kap tasarımı problemi [33]

$x = (T_s, T_h, R, L)$ tasarım vektörü kullanılarak problemin amaç fonksiyonu aşağıdaki şekilde ifade edilir.

$$\min f(x) = 0.6224T_sRL + 1.7781T_hR^2 + 3.1661T_s^2L + 19.84T_s^2R$$

Kısıtlar,

$$g_1(x) = -T_s + 0.0193R \leq 0$$

$$g_2(x) = -T_h + 0.00954R \leq 0$$

$$g_3(x) = -\pi R^2L - \frac{4}{3}\pi R^3 + 750 \times 1728 \leq 0$$

$$g_4(x) = L - 240 \leq 0$$

Problemin basit sınırları $10 \leq R \leq 200$ ve $10 \leq L \leq 240$ 'dir.

Versiyon I

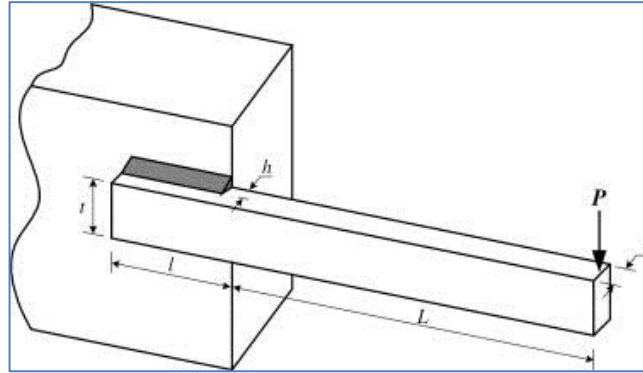
T_s ve T_h kalınlıkları 0.0625 inch'in tamsayı katları olmalıdır. Diğer iki değişken R ve L süreklidir.

Versiyon II

Versiyon I'deki T_s ve T_h değişkenlerinin kısıtlarına ek olarak, R ve L 'nin değerleri 0.1 inch'in tamsayı katları olmalıdır.

A.2. Kaynaklı Kiriş Tasarımı

Kaynaklı kiriş tasarımı probleminin amacı Şekil A.2.'de gösterilen kirişin toplam üretim maliyetini en aza indirmektir. Dört tasarım değişkeni kaynak bölgesinin kalınlığı h ve uzunluğu L ile kirişin genişliği t ve kalınlığı b 'dir.



Şekil A. 2. Kaynaklı kiriş tasarımı problemi [33]

$x = (h, l, t, b)$ tasarım vektörü kullanılarak problemin amaç fonksiyonu aşağıdaki şekilde yazılabilir.

$$\min f(x) = 1.10471h^2l + 0.04811tb(14.0 + l)$$

Kısıtlar,

$$g_1(x) = \tau(x) - 13,600 \leq 0$$

$$g_2(x) = \sigma(x) - 30,000 \leq 0$$

$$g_3(x) = h - b \leq 0$$

$$g_4(x) = 0.10471h^2 + 0.04811tb(14.0 + l) - 5.0 \leq 0$$

$$g_5(x) = 0.125 - h \leq 0$$

$$g_6(x) = \delta(x) - 0.25 \leq 0$$

$$g_7(x) = 6,000 - P(x) \leq 0$$

Burada,

$$\sigma(x) = \frac{504,000}{bt^2}$$

$$Q = 6,000(14.0 + \frac{l}{2})$$

$$D = \frac{1}{2}\sqrt{l^2 + (h + t)^2}$$

$$J = \sqrt{2}hl \left[\frac{l^2}{6} + \frac{(h + t)^2}{2} \right]$$

$$\delta(x) = \frac{65,856}{30,000bt^2}$$

$$\beta = \frac{QD}{J}$$

$$\alpha = \frac{6,000}{\sqrt{2}hl}$$

$$\tau(x) = \sqrt{\alpha^2 + \frac{\alpha\beta l}{D} + \beta^2}$$

$$P(x) = 0.61423 \times 10^6 \frac{tb^3}{6} \left(1 - \frac{t\sqrt{0.625}}{28} \right)$$

Problemin basit sınırları $0.125 \leq h \leq 5$, $0.1 \leq l$, $t \leq 10$ ve $0.1 \leq b \leq 5$ 'dir.

Versiyon I

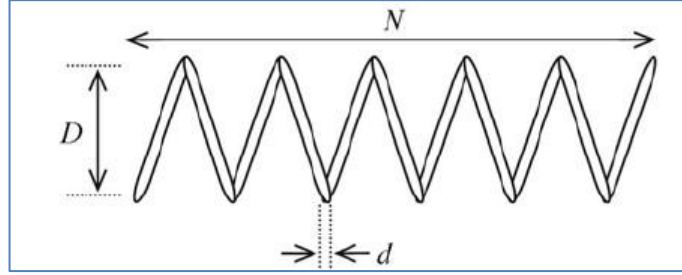
h ve l değişkenlerinin değerleri 0.0065 inch'in tamsayı katları olmalıdır. Diğer iki değişken t ve b süreklidir.

Versiyon II

Versiyon I'deki h ve l değişkenlerinin kısıtlarına ek olarak, t ve b değişkenlerinin değerleri 0.01 inch'in tamsayı katları olmalıdır.

A.3. Sarmal Yay Tasarımı

Sarmal yay tasarımı probleminin amacı Şekil A.3.'de gösterilen yayın üretiminde kullanılan malzemenin hacmini en aza indirmektir. Üç tasarım değişkeni sarmal bobin çapı D , tel çapı d ve aktif bobin sayısı N 'dir.



Şekil A. 3. Sarmal yay tasarımı problemi [33]

$x = (D, d, N)$ tasarım vektörü kullanılarak problemin amaç fonksiyonu aşağıdaki şekilde ifade edilebilir.

$$\min f(x) = \frac{\pi^2 D d^2 (N + 2)}{4}$$

Kısıtlar,

$$g_1(x) = \frac{8C_f P_{max} D}{3.14156 d^3} - S \leq 0$$

$$g_2(x) = \frac{8K P_{max} D N}{C_f d^4} + 1.05(N + 2)d - L_{free} \leq 0$$

$$g_3(x) = d_{min} - d \leq 0$$

$$g_4(x) = (d + D) - D_{max} \leq 0$$

$$g_5(x) = 3 - \frac{D - d}{d} \leq 0$$

$$g_6(x) = \delta - \delta_{pm} \leq 0$$

$$g_7(x) = \frac{8K P_{max} D^3 N}{C_f d^4} + \frac{P_{max} - P_{load}}{K} + 1.05(N + 2)d - L_{free} \leq 0$$

$$g_8(x) = \delta_w - \frac{P_{max} - P_{load}}{K} \leq 0$$

Burada,

$$S_i = \frac{D}{d} \quad C_f = \frac{4S_i - 1}{4S_i - 4} + \frac{0.615}{S_i} \quad K = \frac{Gd^4}{8ND^3}$$

$$D_{max} = 3.0 \text{ (in)} \quad \delta_w = 1.25 \text{ (in)}$$

Maksimum yük $P_{max} = 1000 \text{ lb}$, maksimum kesme gerilmesi $S = 189.000 \text{ psi}$ ve kesme modülü $G = 11.5 \times 10^6 \text{ psi}$ 'dir. Buna ek olarak, en fazla bobin serbest uzunluğu $L_{free} = 14 \text{ inch}$, minimum tel çapı $d_{min} = 0.2 \text{ inch}$, ve ön yük altında izin verilebilir maksimum sapma $\delta_{pm} = 6.0 \text{ inch}$ 'ir.

Versiyon I

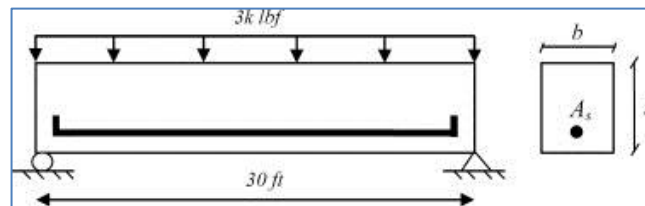
N değişkeni tamsayı ve D süreklidir. D değişkeni yalnızca aşağıdaki ayrık değerleri alabilir: 0.009, 0.0095, 0.0104, 0.0118, 0.0128, 0.0132, 0.014, 0.015, 0.0162, 0.0173, 0.018, 0.020, 0.023, 0.025, 0.028, 0.032, 0.035, 0.041, 0.047, 0.054, 0.063, 0.072, 0.080, 0.092, 0.105, 0.120, 0.135, 0.148, 0.162, 0.177, 0.192, 0.207, 0.225, 0.244, 0.263, 0.283, 0.307, 0.331, 0.362, 0.394, 0.4375, 0.4999.

Versiyon II

Versiyon I'daki N ve D değişkenlerinin kısıtlarına ek olarak, d değişkeni 0.01 inch'in tamsayı katı olmalıdır.

A.4. Güçlendirilmiş Betonarme Kiriş Tasarımı

Şekil A.4.'te gösterilen güçlendirilmiş beton kiriş probleminin amacı, yapının toplam maliyetini en aza indirmektir. Üç tasarım değişkeni güçlendirme alanı A_s ile kirişin genişliği b ve derinliği h 'tir.



Şekil A. 4. Güçlendirilmiş betonarme kiriş tasarımı problemi [33]

$x = (A_s, b, h)$ tasarım vektörü kullanılarak problemin amaç fonksiyonu aşağıdaki gibi yazılabilir:

$$\min f(x) = 29.4A_s + 0.6bh$$

Kısıtlar,

$$g_1(x) = \frac{b}{h} - 4 \leq 0$$

$$g_2(x) = 180 + 7.375 \frac{A_s^2}{h} - \frac{A_s}{b} \leq 0$$

Versiyon I

A_s değişkeni yalnızca aşağıdaki ayrık değerleri alabilir: 6.0, 6.16, 6.32, 6.6, 7.0, 7.11, 7.2, 7.8, 7.9, 8.0, 8.4. b değişkeni $\{28, 29, \dots, 40\}$ kümesinden seçilen bir tamsayı olmalıdır. Diğer değişken h süreklidir.

Versiyon II

Versiyon I'daki A_s ve b değişkenlerinin kısıtlarına ek olarak, h değişkeni 0.1 inch'in tamsayı katı olmalıdır.

EK B: Teorik Optimizasyon Sonuçları – Tüm Algoritmalar

Tablo B. 1. 160 kVA Teorik optimizasyon sonuçları - tüm algoritmalar

Döngü	Pop	ABC			b6e6rl			BSA			CS			FPA		
		Güven	Hız	Top	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam
1000	20	33.6	96.0	46.1	75.1	90.8	78.3	62.2	96.7	69.1	74.5	93.5	78.3	49.4	94.3	58.4
1000	40	70.5	92.4	74.9	93.5	81.9	91.2	66.3	93.8	71.8	80.9	87.5	82.2	38.9	89.3	49.0
1000	60	71.0	88.8	74.6	94.2	72.3	89.8	68.6	91.0	73.1	84.7	81.4	84.0	36.6	83.4	46.0
1000	80	81.3	85.4	82.1	97.8	62.2	90.7	82.3	88.2	83.5	81.8	75.3	80.5	32.5	77.7	41.6
1000	100	79.5	81.8	79.9	98.5	53.9	89.5	80.7	85.4	81.7	83.3	69.4	80.5	33.8	71.7	41.4
1500	20	46.6	94.0	56.1	79.9	86.2	81.2	53.0	95.2	61.4	82.9	90.2	84.4	64.8	91.8	70.2
1500	40	71.4	88.7	74.9	94.0	72.1	89.6	78.8	90.9	81.2	88.6	81.1	87.1	63.7	83.7	67.7
1500	60	80.6	83.3	81.1	97.7	58.7	89.9	85.0	86.5	85.3	89.7	72.2	86.2	64.7	74.4	66.6
1500	80	85.2	78.0	83.8	98.3	45.2	87.7	85.2	82.3	84.6	90.6	63.2	85.1	64.2	67.2	64.8
1500	100	78.2	72.5	77.1	97.9	30.4	84.4	89.5	77.8	87.2	91.4	53.9	83.9	63.9	58.3	62.7
2000	20	47.4	92.0	56.3	76.8	82.0	77.9	72.8	93.7	77.0	86.1	87.1	86.3	80.3	89.7	82.2
2000	40	74.3	84.9	76.5	94.2	63.3	88.0	83.9	87.6	84.7	92.4	75.1	89.0	75.7	79.3	76.4
2000	60	80.3	77.7	79.8	97.6	45.0	87.1	88.4	81.8	87.1	93.6	62.9	87.5	76.3	68.3	74.7
2000	80	82.8	70.6	80.4	98.4	25.3	83.8	88.7	76.3	86.2	94.4	50.8	85.7	76.5	58.9	73.0
2000	100	84.9	63.5	80.6	97.9	8.5	80.0	91.7	70.5	87.4	93.9	38.7	82.9	77.2	46.9	71.1
2500	20	35.6	90.0	46.4	75.7	77.5	76.1	60.1	92.0	66.5	92.1	83.8	90.4	78.5	87.0	80.2
2500	40	84.5	81.1	83.8	93.8	55.4	86.1	65.6	84.9	69.5	95.1	68.5	89.8	84.7	74.1	82.6
2500	60	83.0	72.0	80.8	97.9	33.1	85.0	87.4	77.5	85.5	94.2	53.7	86.1	85.6	59.4	80.4
2500	80	74.8	63.2	72.5	94.9	8.4	77.6	92.0	70.1	87.6	95.4	38.6	84.1	82.9	43.7	75.1
2500	100	84.6	54.4	78.6	97.9	0.0	0.0	90.9	62.8	85.3	95.1	23.3	80.8	83.7	31.4	73.3

Tablo B. 2. 400 kVA Teorik optimizasyon sonuçları - tüm algoritmalar

Döngü	Pop	ABC			b6e6rl			BSA			CS			FPA		
		Güven	Hız	Top	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam
1000	20	0.0	95.9	0.0	42.1	88.5	51.4	48.2	96.7	57.9	72.5	93.3	76.7	11.3	95.4	28.1
1000	40	35.1	92.2	46.5	89.5	76.9	87.0	54.6	93.8	62.5	74.6	87.2	77.1	0.0	90.7	0.0
1000	60	51.5	88.6	59.0	94.0	64.3	88.1	73.5	90.7	77.0	77.6	80.8	78.2	0.0	86.0	0.0
1000	80	71.3	84.9	74.0	100.0	50.8	90.4	84.4	87.6	85.0	77.8	74.8	77.2	0.0	80.9	0.0
1000	100	75.1	81.3	76.4	100.0	39.2	88.0	80.1	85.0	81.1	80.4	68.7	78.0	0.0	75.9	0.0
1500	20	0.0	93.9	0.0	62.6	81.4	66.3	60.5	95.1	67.5	80.7	90.1	82.5	60.8	93.2	67.3
1500	40	50.2	88.4	57.9	88.0	64.0	83.2	77.2	90.6	79.9	84.6	80.9	83.9	53.5	86.1	60.1
1500	60	47.6	82.8	54.7	94.4	45.6	84.6	80.0	86.2	81.3	86.5	71.6	83.5	49.9	78.4	55.6
1500	80	72.9	77.3	73.8	99.2	27.0	84.7	87.2	82.0	86.2	88.1	62.2	82.9	48.7	71.1	53.2
1500	100	76.0	71.7	75.1	96.5	7.5	78.7	87.0	77.1	85.0	87.5	53.0	80.6	52.0	62.4	54.1
2000	20	45.6	91.8	54.8	68.2	76.7	69.9	65.9	93.5	71.4	86.5	86.8	86.5	70.4	90.7	74.5
2000	40	46.9	84.5	54.5	89.1	53.0	81.9	83.2	87.6	84.1	88.1	74.4	85.4	69.6	82.0	72.1
2000	60	76.0	77.2	76.2	95.0	28.7	81.7	87.5	81.7	86.3	88.1	62.1	82.9	66.2	72.8	67.5
2000	80	74.8	69.7	73.8	92.6	6.5	75.4	87.7	75.9	85.4	88.3	49.4	80.5	67.6	61.7	66.4
2000	100	74.7	62.5	72.3	100.0	0.0	0.0	89.5	70.0	85.6	91.5	37.4	80.7	67.3	48.8	63.6
2500	20	0.0	89.8	0.0	60.1	71.1	62.3	51.0	91.9	59.1	87.6	83.5	86.8	80.4	88.2	81.9
2500	40	48.8	80.5	55.2	87.8	40.9	78.4	73.2	84.6	75.5	93.2	68.1	88.2	76.3	76.9	76.4
2500	60	74.2	71.2	73.6	90.9	14.1	75.5	84.2	77.1	82.8	91.9	52.6	84.0	80.5	63.4	77.1
2500	80	59.9	61.9	60.3	94.8	0.0	0.0	88.0	69.5	84.3	92.8	37.2	81.7	76.1	51.6	71.2
2500	100	74.0	53.2	69.9	100.0	0.0	0.0	91.2	62.5	85.5	93.3	21.5	79.0	77.2	39.9	69.8

Tablo B. 3. 630 kVA Teorik optimizasyon sonuçları - tüm algoritmalar

Döngü	Pop	ABC			b6e6rl			BSA			CS			FPA		
		Güven	Hız	Top	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam
1000	20	0.0	95.8	0.0	71.7	89.7	75.3	70.7	96.2	75.8	91.5	92.9	91.8	60.3	95.9	67.4
1000	40	58.9	92.0	65.5	95.4	79.8	92.3	89.6	92.8	90.2	92.4	86.5	91.2	63.0	91.8	68.7
1000	60	66.7	88.1	71.0	95.9	69.0	90.5	90.1	89.4	90.0	94.2	80.1	91.3	65.8	87.4	70.1
1000	80	68.9	84.4	72.0	98.6	58.4	90.6	94.6	85.9	92.8	94.7	73.6	90.4	72.7	82.8	74.7
1000	100	67.7	80.6	70.2	99.5	49.2	89.5	86.4	82.7	85.7	95.4	67.0	89.7	72.6	78.5	73.8
1500	20	56.9	93.7	64.3	93.4	84.8	91.7	73.0	94.3	77.2	92.9	89.6	92.3	77.6	93.8	80.9
1500	40	68.5	87.9	72.4	98.0	69.9	92.4	92.2	89.0	91.5	95.8	79.9	92.6	83.8	87.5	84.5
1500	60	66.9	82.1	69.9	99.5	55.2	90.7	95.3	83.9	93.0	96.2	70.2	91.0	81.7	80.8	81.5
1500	80	70.2	76.5	71.4	98.6	38.7	86.6	95.5	78.8	92.2	96.4	60.3	89.2	81.4	74.2	80.0
1500	100	74.9	70.8	74.1	99.9	23.2	84.6	94.9	74.2	90.7	97.6	50.4	88.2	83.6	67.0	80.3
2000	20	0.0	91.5	0.0	0.0	79.6	0.0	75.4	92.7	78.8	96.1	86.1	94.1	80.8	91.5	83.0
2000	40	39.1	83.8	48.0	95.7	59.3	88.4	82.3	86.0	83.1	97.4	73.0	92.5	87.8	83.2	86.9
2000	60	70.2	76.2	71.4	99.1	39.1	87.1	94.4	79.3	91.4	98.3	59.8	90.6	88.8	74.1	85.9
2000	80	64.3	68.7	65.2	98.5	17.3	82.3	92.0	72.4	88.0	98.3	47.0	88.0	88.9	65.4	84.2
2000	100	69.6	60.9	67.8	99.5	0.0	0.0	96.4	65.9	90.3	98.6	34.2	85.7	90.1	56.3	83.3
2500	20	0.0	89.4	0.0	75.9	74.0	75.5	0.0	90.6	0.0	98.6	82.7	95.4	85.7	89.6	86.5
2500	40	64.3	79.9	67.4	97.8	50.4	88.3	92.6	81.9	90.4	98.1	66.2	91.7	88.6	79.1	86.7
2500	60	65.9	70.4	66.8	98.7	23.3	83.6	95.6	73.5	91.2	98.5	50.0	88.8	92.5	67.9	87.6
2500	80	71.2	60.8	69.1	99.4	0.0	0.0	95.6	65.3	89.6	98.7	33.8	85.7	92.6	56.1	85.3
2500	100	71.1	51.0	67.1	99.9	0.0	0.0	97.8	56.9	89.6	99.2	17.6	82.9	93.5	45.4	83.9

EK C: Pratik Optimizasyon Sonuçları – Tüm Algoritmalar

Tablo C. 1. 160 kVA Pratik optimizasyon sonuçları - tüm algoritmalar

Döngü	Pop	ABC			b6e6rl			BSA			CS			FPA		
		Güven	Hız	Top	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam
1000	20	0.0	95.1	0.0	40.5	89.9	50.4	38.8	97.1	50.5	86.0	92.0	87.2	80.1	94.9	83.0
1000	40	74.7	90.8	77.9	74.8	80.2	75.9	67.3	94.5	72.8	92.4	84.8	90.8	85.0	89.7	86.0
1000	60	77.7	86.5	79.5	79.3	70.1	77.4	84.5	91.9	86.0	99.2	77.3	94.9	86.2	84.7	85.9
1000	80	79.6	81.9	80.1	77.1	59.3	73.5	81.2	89.3	82.9	99.2	68.7	93.1	84.9	79.7	83.9
1000	100	81.3	77.9	80.6	75.4	48.9	70.1	79.7	86.6	81.1	99.2	61.0	91.6	85.5	74.3	83.2
1500	20	33.1	92.6	45.0	0.0	84.1	0.0	0.0	95.5	0.0	87.3	87.4	87.3	83.1	92.4	84.9
1500	40	80.9	86.1	82.0	78.4	70.1	76.7	69.9	91.4	74.2	91.0	76.2	88.1	89.0	84.9	88.2
1500	60	75.2	79.8	76.1	78.4	55.4	73.8	72.0	87.2	75.1	99.2	65.7	92.5	99.2	77.6	94.9
1500	80	81.7	73.2	80.0	80.9	38.2	72.4	78.5	83.3	79.5	99.2	54.7	90.3	99.1	69.3	93.2
1500	100	81.6	67.4	78.8	79.0	25.4	68.3	84.8	79.2	83.7	99.2	44.1	88.2	97.2	61.8	90.1
2000	20	0.0	90.3	0.0	33.3	80.0	42.7	31.8	94.2	44.3	87.3	84.0	86.7	83.9	90.1	85.1
2000	40	78.7	81.5	79.3	66.4	57.3	64.6	0.0	89.2	0.0	89.9	68.6	85.6	88.2	79.6	86.5
2000	60	74.5	72.9	74.1	82.2	39.4	73.7	74.6	84.2	76.5	99.2	54.0	90.2	99.2	69.5	93.3
2000	80	80.5	64.2	77.2	75.4	20.4	64.4	84.2	78.4	83.1	99.2	39.9	87.4	89.9	59.0	83.7
2000	100	79.1	56.2	74.5	78.0	0.0	0.0	86.5	73.0	83.8	99.2	26.2	84.6	99.2	48.4	89.1
2500	20	0.0	87.8	0.0	43.5	75.8	50.0	27.9	92.7	40.8	86.5	80.4	85.3	82.2	87.4	83.2
2500	40	61.3	77.2	64.5	69.1	50.5	65.4	71.9	86.0	74.7	99.2	61.2	91.6	89.9	74.6	86.8
2500	60	80.8	66.8	78.0	81.4	25.2	70.1	74.1	79.8	75.2	99.2	42.6	87.9	99.2	61.8	91.8
2500	80	78.7	54.5	73.8	77.5	0.0	0.0	83.4	73.2	81.4	99.2	23.6	84.1	99.2	48.6	89.1
2500	100	77.5	44.8	71.0	79.7	0.0	0.0	85.8	65.8	81.8	99.2	4.5	80.3	99.2	35.6	86.5

Tablo C. 2. 400 kVA Pratik optimizasyon sonuçları - tüm algoritmalar

Döngü	Pop	ABC			b6e6rl			BSA			CS			FPA		
		Güven	Hız	Top	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam
1000	20	31.1	95.3	44.0	0.0	91.0	0.0	49.3	97.3	58.9	99.6	92.4	98.2	98.7	95.1	98.0
1000	40	68.1	91.1	72.7	84.2	82.0	83.7	79.2	94.7	82.3	99.6	85.3	96.8	97.8	90.2	96.3
1000	60	90.8	86.9	90.0	97.6	72.9	92.7	82.3	92.4	84.3	99.6	77.8	95.3	96.8	85.3	94.5
1000	80	94.2	82.9	92.0	98.1	63.6	91.2	90.5	89.9	90.4	99.6	71.3	94.0	97.2	80.0	93.8
1000	100	95.8	78.9	92.4	98.0	54.4	89.3	97.2	87.5	95.3	99.6	63.8	92.4	96.8	75.2	92.5
1500	20	0.0	92.9	0.0	62.0	86.4	66.9	0.0	95.9	0.0	98.1	88.6	96.2	98.8	92.7	97.6
1500	40	54.1	86.8	60.6	96.4	73.2	91.8	51.1	92.2	59.3	99.6	77.9	95.3	99.6	85.3	96.8
1500	60	87.0	80.8	85.8	97.6	59.4	90.0	97.6	88.7	95.8	99.6	67.1	93.1	99.6	78.0	95.3
1500	80	94.9	74.3	90.8	97.9	46.2	87.5	89.4	84.8	88.5	99.6	56.6	91.0	99.6	70.4	93.8
1500	100	95.7	68.3	90.2	98.0	32.8	85.0	89.3	81.5	87.8	99.6	45.6	88.8	99.6	62.6	92.2
2000	20	39.4	90.8	49.6	0.0	82.2	0.0	0.0	94.7	0.0	99.6	84.8	96.7	98.1	90.2	96.5
2000	40	84.2	82.4	83.9	84.1	64.5	80.2	79.1	89.6	81.2	99.6	70.4	93.8	99.6	80.3	95.8
2000	60	87.1	74.4	84.5	95.7	46.4	85.9	97.7	84.9	95.2	99.6	56.2	90.9	99.6	70.5	93.8
2000	80	93.5	65.7	87.9	97.9	28.8	84.1	97.9	80.1	94.3	99.6	42.4	88.2	99.6	60.3	91.8
2000	100	97.6	56.8	89.5	99.6	10.5	81.8	97.7	75.2	93.2	99.6	28.3	85.3	99.6	49.8	89.7
2500	20	0.0	88.4	0.0	88.8	77.7	86.6	0.0	93.1	0.0	99.6	80.8	95.9	98.1	87.7	96.0
2500	40	82.7	78.2	81.8	95.7	55.7	87.7	64.2	86.8	68.7	99.6	63.1	92.3	99.6	75.4	94.8
2500	60	89.0	67.9	84.8	95.1	33.1	82.7	97.1	80.7	93.8	99.6	45.5	88.8	99.6	63.0	92.3
2500	80	97.1	57.4	89.1	97.7	9.5	80.1	97.5	74.2	92.8	99.6	27.9	85.3	99.6	50.1	89.7
2500	100	97.5	47.1	87.4	98.0	0.0	0.0	95.2	68.7	89.9	99.6	9.9	81.7	99.6	37.8	87.2

Tablo C. 3. 630 kVA Pratik optimizasyon sonuçları - tüm algoritmalar

Döngü	Pop	ABC			b6e6rl			BSA			CS			FPA		
		Güven	Hız	Top	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam	Güven	Hız	Toplam
1000	20	0.0	94.5	0.0	0.0	90.5	0.0	0.0	97.1	0.0	57.9	91.0	64.5	39.5	94.8	50.5
1000	40	0.0	89.4	0.0	21.9	80.6	33.7	18.9	94.6	34.1	100.0	81.9	96.6	90.2	89.5	90.0
1000	60	0.0	84.6	0.0	34.4	71.6	41.8	12.9	91.9	28.7	100.0	73.5	94.9	87.7	84.1	87.0
1000	80	0.0	79.4	0.0	48.2	60.7	50.7	28.6	89.3	40.7	100.0	66.5	93.5	91.7	78.6	89.1
1000	100	0.0	74.9	0.0	77.9	52.7	72.9	38.7	86.9	48.3	100.0	57.3	91.7	90.2	73.3	86.8
1500	20	0.0	91.6	0.0	0.0	86.0	0.0	0.0	95.7	0.0	55.0	86.4	61.3	47.0	92.1	56.0
1500	40	0.0	84.2	0.0	19.1	70.8	29.5	0.0	91.8	0.0	100.0	74.2	95.1	100.0	83.9	97.0
1500	60	0.0	76.4	0.0	31.7	56.3	36.6	11.8	87.8	27.0	100.0	60.8	92.4	100.0	76.2	95.4
1500	80	0.0	69.2	0.0	45.0	39.2	43.8	36.5	83.9	46.0	100.0	49.4	90.1	100.0	67.7	93.8
1500	100	0.0	62.2	0.0	49.7	25.7	44.9	48.0	80.7	54.5	100.0	34.5	87.1	100.0	59.7	92.2
2000	20	0.0	89.0	0.0	10.0	81.2	24.3	0.0	94.3	0.0	57.7	81.3	62.4	51.4	89.4	59.0
2000	40	0.0	78.7	0.0	0.0	62.5	0.0	0.0	88.8	0.0	100.0	64.7	93.2	100.0	78.7	96.0
2000	60	0.0	68.1	0.0	33.3	44.3	35.5	24.1	83.3	36.0	100.0	49.1	90.1	100.0	68.1	93.8
2000	80	0.0	57.8	0.0	69.3	24.6	60.3	42.2	78.4	49.4	100.0	32.6	86.8	100.0	56.8	91.6
2000	100	0.0	48.4	0.0	64.7	7.0	53.1	44.7	73.2	50.4	100.0	16.4	83.5	100.0	45.2	89.3
2500	20	0.0	86.3	0.0	0.0	75.4	0.0	0.0	92.8	0.0	56.3	78.1	60.7	92.1	86.7	91.0
2500	40	0.0	73.2	0.0	12.9	53.8	21.1	0.0	86.2	0.0	100.0	57.8	91.8	100.0	73.3	94.9
2500	60	0.0	60.6	0.0	32.3	31.6	32.1	14.7	79.8	27.7	100.0	35.9	87.4	100.0	59.5	92.1
2500	80	0.0	48.6	0.0	42.7	8.5	35.9	22.4	73.1	32.6	100.0	12.9	82.8	100.0	45.7	89.4
2500	100	0.0	37.4	0.0	46.5	0.0	0.0	37.9	67.2	43.7	100.0	0.0	0.0	100.0	31.4	86.5

ÖZGEÇMİŞ

Levent Alhan, 21.06.1948'de Bafra'da doğdu. İlk, orta ve lise eğitimini Sakarya'da tamamladı. Lisans ve yüksek lisans derecelerini 1971 ve 1976 yıllarında Orta Doğu Teknik Üniversitesi Elektrik-Elektronik Mühendisliği Bölümü'nden, MBA derecesini de 1974 yılında Gazi Üniversitesi'nden aldı. Ziraat Bankası Genel Müdürlüğü'nde 3 yıl, Hacettepe Üniversitesi vakıf şirketlerinden Sisag Ltd.'de 1.5 yıl kadar Sistem Analist-Programcı olarak çalıştı. 1976 yılında o dönem Türkiye'nin önde gelen transformatör üreticilerinden olan ESAŞ Elektrik Sanayi ve Ticaret A.Ş.'de Bilgi İşlem Sorumlusu olarak çalışmaya başladı; burada öncelikli olarak dağıtım ve güç transformatörleri tasarım ve optimizasyon programları ile güç transformatörü yüksek gerilim sargılarında şok gerilimi dağılımı gibi bazı teknik uygulamaları geliştirdi. Yaklaşık 9 yıl ESAŞ'ta üretim planlama, bilgi işlem ve yeni yatırımlardan sorumlu olarak Planlama Müdürlüğü görevini yürüttü. ESAŞ 1988 yılında ABB Grubu'na katıldıktan sonra bu grubun güç transformatörleri tasarım ve ERP uygulamalarının ABB Türkiye'de uyarlanması konularında çalıştı. Bilişim Sistemleri, Toplam Kalite ve Tesis Yönetimi konularından sorumlu ABB Türkiye İcra Kurulu üyesi olarak 12 yıl kadar görev yaptı. 2012'de emekli oldu ve halen serbest yönetim danışmalığı yapmaktadır. Ayrıca Sakarya Üniversitesi Bilgisayar ve Bilişim Bilimleri Fakültesi'nde gönüllü öğretim üyesi olarak ders vermekte ve doktora çalışmasını sürdürmektedir. Araştırma alanları arasında bilgisayar destekli mühendislik tasarımı ve optimizasyonu, metasezgisel algoritmalar, İş Süreçleri Yönetimi konuları yer almaktadır.