



T.C.

SELÇUK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

TEK DEĞİŞKENLİ VE ÇOK DEĞİŞKENLİ DAĞILIMLARIN PARAMETRE TAHMİNİ İÇİN METASEZGİSEL YAKLAŞIMLAR

Aynur YONAR

DOKTORA TEZİ

İstatistik Anabilim Dalı

Temmuz-2020
KONYA
Her Hakkı Saklıdır

ÖZET

DOKTORA TEZİ

TEK DEĞİŞKENLİ VE ÇOK DEĞİŞKENLİ DAĞILIMLARIN PARAMETRE TAHMİNİ İÇİN METASEZGİSEL YAKLAŞIMLAR

Aynur YONAR

Selçuk Üniversitesi Fen Bilimleri Enstitüsü
İstatistik Anabilim Dalı

Danışman: Prof. Dr. Nimet YAPICI PEHLİVAN

2020, 161 Sayfa

Jüri
Prof. Dr. Nimet YAPICI PEHLİVAN
Prof. Dr. M. Özgür YENİAY
Prof. Dr. M. Fedai KAYA
Prof. Dr. H. Hasan ÖRKÇÜ
Dr. Öğr. Üyesi Yunus AKDOĞAN

İstatistik teorisinde, dağılımların parametrelerinin tahmin edilmesi istatistiksel sonuç çıkarımı açısından önemli bir yere sahiptir. En çok olabilirlik (ML) ve en küçük kareler (LS) yöntemleri literatürde en yaygın olarak kullanılan parametre tahmin yöntemleridir. ML yöntemi, olabilirlik fonksiyonunun dağılımın parametrelerine göre en büyük yapılması, LS yöntemi ise hata kareler toplamının dağılımın parametrelerine göre en küçük yapılması temeline dayanmaktadır. Kısacası, her iki yöntem de bir optimizasyon problemi içermektedir. 3 parametreli Weibull ve Gamma gibi tek ve çok değişkenli bazı dağılımların parametrelerinin tahminleri açık formda elde edilememekte ve bu dağılımların ML ve LS tahminlerini elde edebilmek için sıralı karesel programlama ve iç nokta algoritması gibi iteratif yöntemler kullanılmaktadır. Bu yöntemler başlangıç değer problemine sahip olduğundan özellikle parametre sayısının fazla olduğu durumlarda başarısız olmaktadırlar. Bu tez çalışmasında, klasik optimizasyon yöntemlerine göre daha avantajlı olan bazı metasezgisel yöntemlerin: Tavlama Benzetimi Algoritması, Genetik Algoritma, Diferansiyel Gelişim Algoritması, Parçacık Sürüsü Optimizasyonu ve Yapay Arı Kolonisi Algoritması, kullanılması önerilmiştir. Bu yöntemlerin performanslarını geliştirmek amacıyla Levy Uçuşları, Uyarlanmış En Çok Olabilirlik, Profil Olabilirlik ve Taguchi Deney Tasarımı yöntemlerine dayalı çeşitli yaklaşımlar önerilmiştir. Monte-Carlo simülasyon çalışmaları ve gerçek veri uygulamaları ile bu yöntemlerin uygulanabilirliği gösterilmiştir. Sonuç olarak, ele alınan dağılımlar için önerilen metasezgisel yöntemlerden elde edilen tahmin edicilerin etkin olduğu gözlenmiştir.

Anahtar Kelimeler: Diferansiyel Gelişim, Genetik Algoritma, Olabilirlik Fonksiyonu, Metasezgisel, Parametre Tahmini, Parçacık Sürüsü Optimizasyonu, Tavlama Benzetimi, Yapay Arı Kolonisi.

ABSTRACT

Ph.D THESIS

METAHEURISTIC APPROACHES FOR ESTIMATING PARAMETERS OF UNIVARIATE AND MULTIVARIATE DISTRIBUTIONS

Aynur YONAR

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF SELÇUK UNIVERSITY
THE DEGREE OF MASTER OF SCIENCE / DOCTOR OF PHILOSOPHY IN MECHANICAL ENGINEERING

Advisor: Prof. Dr. Nimet YAPICI PEHLİVAN

2020, 161 Pages

Jury
Prof. Dr. Nimet YAPICI PEHLİVAN
Prof. Dr. M. Özgür YENİAY
Prof. Dr. M. Fedai KAYA
Prof. Dr. H. Hasan ÖRKÇÜ
Assist Prof. Dr. Üyesi Yunus AKDOĞAN

In statistical theory, estimation of the parameters of distributions has an important place in terms of statistical inference in statistical theory. Maximum Likelihood (ML) and Least Squares (LS) methods are the most commonly used parameter estimation methods in the literature. The ML method is based on the principle that the likelihood function is maximized, while the LS method is based on the principle that the sum of the error squares is minimized according to the parameters of distributions. In short, both methods involve an optimization problem. The ML and the LS estimates of parameters of some distributions such as univariate and multivariate 3 parameter Weibull and Gamma distributions cannot be obtained in explicit form, and some iterative methods such as sequential quadratic programming and interior point algorithm are used to obtain ML or LS estimators of these distributions. However, they fail especially when the number of parameters is high since these methods have initial value problems. In thesis study, it has been suggested to use some metaheuristic methods: Simulating Annealing Algorithm, Genetic Algorithm, Differential Evolution Algorithm, Particle Swarm Optimization, and Artificial Bee Colony Algorithm, which are more advantageous than classical optimization methods. Furthermore, various approaches have been proposed based on Levy flights, Modified Maximum Likelihood (MML) estimators and Taguchi Experimental Design methods to improve the performance of considered methods. The applicability of these methods has been demonstrated with Monte-Carlo simulation studies and real data applications. As a result, it has been observed that the estimators obtained by the proposed metaheuristic methods for the considered distributions are efficient.

Keywords: Artificial Bee Colony, Differential Evolution, Genetic Algorithm, Likelihood Function, Metaheuristic, Parameter Estimation, Particle Swarm Optimization, Simulated Annealing.

ÖNSÖZ

Bu tez çalışmasının her aşamasında öneri ve katkılarıyla beni yönlendiren danışmanım Sayın Prof. Dr. Nimet Yapıcı PEHLİVAN'a, bilgi ve yorumlarıyla tezime yön veren tez izleme komitesi üyeleri Sayın Prof. Dr. H. Hasan ÖRKÇÜ'ye ve Sayın Dr. Öğr. Üyesi Yunus AKDOĞAN'a, tezimi inceleyen ve önemli katkılar sağlayan Sayın Prof. Dr. M. Fedai KAYA'ya ve Sayın Prof. Dr. M. Özgür YENİAY'a, bilgilerini ve yardımlarını benden esirgemeyen Selçuk Üniversitesi Fen Fakültesi İstatistik Bölümü ve Aktüerya Bölümü öğretim üyelerine ve elemanlarına ayrı ayrı teşekkürlerimi ve saygılarımı sunarım. Ayrıca her zaman yanımda olan ve beni destekleyen canım aileme ve değerli eşim Harun YONAR'a en içten duygularımla teşekkür ederim.

Aynur YONAR
KONYA-2020

İÇİNDEKİLER

ÖZET ..... **iv**

ABSTRACT ..... **v**

ÖNSÖZ ..... **vi**

İÇİNDEKİLER ..... **vii**

SİMGELER VE KISALTMALAR ..... **ix**

1. GİRİŞ **1**

2. KAYNAK ARAŞTIRMASI **7**

3. MATERYAL VE YÖNTEM **14**

3.1. Bazı Tek Değişkenli ve Çok Değişkenli Dağılımlar 14
3.1.1. 3-p Weibull dağılımı 14
3.1.2. 3-p Gamma dağılımı 15
3.1.3. Kesikli Weibull dağılımı 17
3.1.4. Çok değişkenli Gamma dağılımı 18
3.1.5. Çok değişkenli kesikli Weibull dağılımı 20
3.2. Parametre Tahmin Yöntemleri 23
3.2.1. En çok olabilirlik yöntemi 23
3.2.2. En küçük kareler yöntemi 24
3.2.3. Uyarlanmış en çok olabilirlik yöntemi 25
3.3. Kısıtlı Doğrusal Olmayan Optimizasyon Algoritmaları 28
3.3.1. Sıralı karesel programlama 28
3.3.2. İç nokta algoritmaları 30
3.4. TOPSIS Yöntemi 32
3.5. Taguchi Yöntemi 34
3.6. Kullanılan İstatistiksel Kriterler 36

4. METASEZGİSEL YÖNTEMLER **39**

4.1. Tavlama Benzetimi Algoritması 42
4.2. Genetik Algoritma 45
4.3. Parçacık Sürüsü Optimizasyonu 48
4.4. Diferansiyel Gelişim Algoritması 52
4.5. Yapay Arı Kolonisi Algoritması 56
4.6. Levy Uçuşlu Yapay Arı Kolonisi Algoritması 60

5. TEK DEĞİŞKENLİ DAĞILIMLAR İÇİN METASEZGİSEL YÖNTEMLER İLE PARAMETRE TAHMİNİ **64**

5.1. 3-p Weibull Dağılımının Parametrelerinin Tahmini İçin Levy Uçuşlu Yapay Arı Kolonisi Algoritması 64
5.1.1. 3-p Weibull dağılımının parametrelerinin tahmini için ML yöntemi 64

5.1.2. Simülasyon çalışması 65
5.1.3. LABC algoritmasının diğer metasezgisel yöntemlerle karşılaştırmalı analizi 70
5.1.4. Gerçek veri uygulaması 74
5.2. 3-p Gamma Dağılımının Parametrelerinin Tahmininde Metasezgisel Yöntemlerin Karşılaştırılması 75
5.2.1. Gamma dağılımının parametrelerinin tahmini için ML yöntemi 76
5.2.2. Simülasyon çalışması 77
5.2.3. TOPSIS yöntemi ile metasezgisel yöntemlerin karşılaştırılması 86
5.2.4. Gerçek veri uygulaması 88
5.3. 3-p Gamma Dağılımının Parametrelerinin Diferansiyel Gelişim Algoritması ile Tahmini İçin Yeni Bir Yaklaşım 89
5.3.1. 3-p Gamma dağılımının parametrelerinin tahmini için LS yöntemi 90
5.3.2. DE algoritması için arama uzayının MML ve PL yöntemleri kullanılarak belirlenmesi 91
5.3.3. Simülasyon çalışması 95
5.3.4. Gerçek veri uygulaması 106
5.4. Kesikli Weibull Dağılımının Parametrelerinin Metasezgisel Yöntemler ile Tahmini 108
5.4.1. Kesikli Weibull dağılımlarının parametrelerinin tahmini için ML yöntemi 108
5.4.2. Simülasyon çalışması 109
5.4.3. Gerçek veri uygulaması 117

6. ÇOK DEĞİŞKENLİ DAĞILIMLAR İÇİN METASEZGİSEL YÖNTEMLER İLE PARAMETRE TAHMİNİ 119

6.1. Çok Değişkenli Gamma Dağılımının Parametrelerinin Parçacık Sürüsü Optimizasyonu ile Tahmini 119
6.1.1. Çok değişkenli Gamma dağılımının parametrelerinin tahmini için ML yöntemi 120
6.1.2. PSO algoritması için Taguchi deney tasarımı 121
6.1.3. Simülasyon çalışması 127
6.2. Çok Değişkenli Kesikli Weibull Dağılımının Parametrelerinin Metasezgisel Yöntemler ile Tahmini 128
6.2.1. Çok değişkenli kesikli Weibull dağılımlarının parametrelerinin tahmini için ML yöntemi 129
6.2.2. Simülasyon çalışması 130
6.2.3. Gerçek veri uygulaması 136

7. SONUÇLAR VE ÖNERİLER 138

7.1 Sonuçlar 138
7.2. Öneriler 141

KAYNAKLAR 142

ÖZGEÇMİŞ 149

SİMGELER VE KISALTMALAR

Simgeler

Θ:	Parametre uzayı
$\mathbb{N}_+$:	Pozitif doğal sayılar kümesi
$\mathbb{R}$:	Reel sayılar kümesi
$\mathbb{R}^+$:	Pozitif reel sayılar kümesi
$\mathbb{R}^d$:	d boyutlu reel sayılar kümesi
$F(.)$:	Dağılım fonksiyonu
$f(.)$:	Olasılık fonksiyonu
$E(.)$:	Beklenen değer
$Var(.)$:	Varyans
$\ln(.)$:	Doğal logaritma fonksiyonu
$\exp(.)$:	Üstel fonksiyonu
φ:	Amaç fonksiyonu değeri
$\hat{\varphi}$:	Amaç fonksiyonu tahmin değeri
$\Gamma(.)$:	Gamma fonksiyonu
$\gamma(.)$:	Tamamlanmamış Gamma fonksiyonu

Kısaltmalar

ABC:	Yapay Arı Kolonisi (Artificial Bee Colony)
AIC:	Akaike Bilgi Kriteri (Akaike Information Criteria)
BDW:	İki Değişkenli Kesikli Weibull
BFGS:	Broyon-Fletcher-Goldfarbo-Shanno
BIC:	Bayes Bilgi Kriteri (Bayesian Information Criteria)
CMA-ES:	Kovaryans Matrisi Adaptasyon Evrim Stratejisi (Covariance Matrix Adaptation Evolution Strategy)
DE:	Diferansiyel Gelişim (Differential Evolution)
Def:	Kusur (Deficiency)
df:	Dağılım Fonksiyonu
DW:	Kesikli Weibull
EDAs:	Dağılım Algoritmalarının Tahmini (Estimation of Distribution Algorithms)
EM:	Beklenti Maksimizasyonu (Expectation Maximization)
GA:	Genetik Algoritma (Genetic Algorithm)
ILS:	İteratif Yerel Arama (Iterative Local Search)
IPA:	İç Nokta Algoritması (Interior Point Algorithm)
KKT:	Karush-Kuhn-Tucker
KO:	Kareler Ortalaması
K-S:	Kolmogorov Smirnov
KT:	Kareler Toplamı
LS:	En Küçük Kareler (Least Square)
MCDM:	Çok Kriterli Karar Verme (Multi Criteria Decision Making)
MGM:	Çok Değişkenli Gamma (Multivariate Gamma)
ML:	En Çok Olabilirlik (Maximum Likelihood)

MML:	Uyarlanmış En Çok Olabilirlik (Modified Maximum Likelihood)
MSE:	Hata Kareler Ortalaması (Mean Square Error)
NFL:	No-Free-Lunch
NM:	Nelder-Mead
of:	Olasılık Fonksiyonu
Ort.:	Ortalama
oyf:	Olasılık Yoğunluk Fonksiyonu
PSO:	Parçacık Sürüsü Optimizasyonu (Particle Swarm Optimization)
PL:	Profil Olabilirlik (Profile Likelihood)
SA:	Tavlama Benzetimi (Simulated Annealing)
sd:	Serbestlik Derecesi
SQP:	Sıralı Karesel Programlama (Sequential Quadratic Programming)
TS:	Tabu Arama (Tabu Search)
TOPSIS:	İdeal Çözüme Yakınlığa Göre Tercih Sıralaması Tekniği (Technique for Order of Preference by Similarity to Ideal Solution)
VNS:	Değişken Komşu Arama (Variable Neighbor Search)
3-p Weibull:	3-parametreli Weibull
3-p Gamma:	3-parametreli Gamma

1. GİRİŞ

Parametre tahmini, istatistiğin en önemli problemlerinden biridir. Üzerinde çalışılan kitleyi daha iyi anlayabilmek için kitlenin bazı bilinmeyen karakteristikleri, yani parametreleri hakkında bilgi sahibi olmak gerekir. Kitle parametreleri, kitleden alınan bir örneklem yardımıyla oluşturulan istatistikler yardımıyla tahmin edilir. Bu tahminleri elde etmek için en çok olabilirlik, momentler, en küçük kareler ve Bayes tahmin yöntemi gibi çeşitli yöntemler mevcuttur (Akdi, 2005). En çok olabilirlik (Maximum Likelihood-ML) yöntemi düzgünlük koşulları altında asimptotik olarak yansız ve minimum varyanslı tahmin ediciler elde etmesi nedeniyle en yaygın olarak kullanılan tahmin yöntemidir. Diğer bir popüler yöntem ise, hesaplama kolaylığı açısından en küçük kareler yöntemi (Least Square-LS)'dir. ML yöntemi ilgilenilen dağılımın olabilirlik fonksiyonunun bilinmeyen parametrelere göre maksimizasyonuna dayalı iken, LS yöntemi hata kareler toplamının bilinmeyen parametrelere göre minimizasyonuna dayanmaktadır. Her iki yöntem de birer optimizasyon problemi içermekte olup bu problemlerin çözümü sonucunda tahminler elde edilebilmektedir. Bu tür problemler, amaç fonksiyonunun sürekli ve ilgili parametrelere göre türevlenebilir olması durumunda genellikle klasik optimizasyon yöntemleri ile çözülebilmektedir. Klasik yöntemler amaç fonksiyonunun bilinmeyen parametrelere göre türevinin alınıp sıfıra eşitlenmesi ile elde edilen denklemlerin çözümüne dayanmaktadır. Bazı durumlarda, bu denklemlerin analitik çözümleri elde edilemeyebilir. Böyle durumlarda, problemin yapısına göre Broyon-Fletcher-Goldfarbo-Shanno (BFGS), Nelder-Mead (NM), kabul-red, sıralı karesel programlama (SQP) ve iç nokta algoritması (IPA) gibi çeşitli iteratif yöntemler kullanılmaktadır. Bu yöntemler, genellikle başlangıç çözümüne bağlı olarak yerel optimal çözümlere takılmaktadırlar (Abbasi ve ark., 2006). Ayrıca, özellikle parametre sayısının ikiden fazla olduğu ve örneklem sayısının yeterince büyük olmadığı (örneğin $n < 250$) durumlarda parametrelerin tahmini için bu iteratif yöntemler güvenilir sonuçlar vermemektedir (Bowman ve Shenton, 2001; Kantar ve Şenoğlu, 2008; Yalçınkaya, 2018). Gerçek hayatta, genellikle örneklem sayısı küçük olduğundan bu yöntemler yardımıyla elde edilen parametre tahminleri güvenilir olmayabilir. Bundan dolayı, klasik optimizasyon yöntemleri yerine, çeşitli stratejiler uygulayarak yerel optimallikten kaçınan metasezgisel yöntemlerin kullanması daha faydalı olabilmektedir.

Metasezgisel yöntemler, herhangi bir amacı gerçekleştirmek veya hedefe varmak için çeşitli alternatif hareketlerden etkili olanlara karar vermek amacı ile çözüm uzayını etkin bir şekilde aramayı sağlayan yüksek seviyeli sezgisel algoritmalardır (Voß ve ark., 2012). Bu tür algoritmalar, yakınsama özelliğine sahip olmasına rağmen kesin çözümü garanti edemezler ve sadece kesin çözüm yakınındaki bir çözümü garanti edebilirler (Karaboğa ve ark., 2014). Klasik yöntemlerin başarısız olduğu yerlerde metasezgisel yöntemlerin kullanımı oldukça yaygındır ve literatürde çok sayıda metasezgisel yöntem mevcuttur. Bu yöntemlerden başlıcaları: Tavlama Benzetimi (Simulated Annealing-SA) (Kirkpatrick ve ark., 1983), Tabu Arama (Tabu Search-TS) (Glover, 1989), Genetik Algoritma (Genetic Algorithm-GA) (Holland, 1975), Karınca Kolonisi Optimizasyonu (Ant Colony Optimization-ACO) (Dorigo ve ark., 1996), Parçacık Sürüsü Optimizasyonu (Particle Swarm Optimization-PSO) (Kennedy ve Eberhart, 1995), Diferansiyel Gelişim (Differential Evolution-DE), (Storn, 1996) ve (Storn ve Price, 1997) ve Yapay Arı Kolonisi (Artificial Bee Colony-ABC) (Karaboga, 2005) algoritmalarıdır.

Literatürde çeşitli dağılımların parametrelerinin tahmini üzerine yapılmış birçok çalışma mevcuttur. 3-parametreli Weibull, 4-parametreli Burr III, 3-parametreli Burr XII, çarpık normal dağılım, genelleştirilmiş Gamma gibi bazı tek değişkenli dağılımların ML parametre tahmini sürecinde SA, DE, PSO, birleştirilmiş VNS ve ILS ve GA gibi metasezgisel yöntemler kullanılmıştır (Shao, 2004; Abbasi ve ark., 2006; Abbasi ve ark., 2011; Zoraghi ve ark., 2012; Örkcü ve ark., 2015a; Örkcü ve ark., 2015b; Özsoy ve ark., 2017; Yalçınkaya ve ark., 2018; Özsoy ve ark., 2020). Bu çalışmalara ilişkin detaylı bilgiler Kaynak Araştırması bölümünde verilmiştir.

Optimizasyon problemlerinin çözümünde kullanılan çok sayıda metasezgisel yöntem mevcuttur ve bunlardan hangisinin en iyi olduğu çoğu araştırmacı için ortak bir sorundur. Gerçek dünya problemlerinin karmaşıklığı ve çeşitliliğinden dolayı bazı problemlerin çözümü kolay iken bazılarının çözümü oldukça zordur. Bundan dolayı, tüm problem çeşitleri için tek bir yöntemin iyi olduğunu söylemek mümkün değildir. Ayrıca, Wolpert ve Macready (1997) tarafından önerilen “no-free-lunch (NFL)” teoremi de tüm problemler için ortak bir yöntemin mevcut olmadığını göstermiştir. Yani, herhangi bir problem için A yöntemi B yönteminden daha iyi iken, başka bir problem için B yöntemi A yönteminden daha iyi olabilmektedir (Yang, 2014). Bu doktora tez çalışmasında, NFL teoremi baz alınarak, 3-p Weibull, 3-p Gamma, kesikli Weibull, çok değişkenli Gamma ve çok değişkenli kesikli Weibull dağılımlarının parametrelerinin

tahmininde literatürde en yaygın kullanılan SA, GA, PSO, DE ve ABC metasezgisel yöntemlerinin kullanılması, ele alınan yöntemlerin karşılaştırılması ve bu yöntemlerin performanslarını arttırmak için çalışmalar yapılması amaçlanmıştır.

3-p Weibull, 3-p Gamma ve kesikli Weibull dağılımları, özellikle güvenirlik ve yaşam zamanı alanlarında en yaygın kullanılan dağılımlardır. Bu dağılımların başarılı uygulamaları, parametrelerinin kabul edilebilir istatistiksel tahminlerine bağlıdır. Bundan dolayı, bu dağılımların parametrelerinin tahmini çok önemlidir. Ayrıca, çok değişkenli Gamma ve çok değişkenli kesikli Weibull dağılımlarında değişken sayısına bağlı olarak parametre sayısı arttığından dolayı bu dağılımlar için parametre tahmin problemi tek değişkenli dağılımlara göre daha da zor olmaktadır.

Metasezgisel yöntemlerin performansı, algoritmaların arama yeteneklerine, arama uzaylarına ve parametrelerine bağlı olarak değişmektedir. Ele alınan metasezgisel yöntemlerin performansını arttırmak amacıyla çeşitli çalışmalar gerçekleştirilmiştir. Örneğin, 3-p Weibull dağılımının parametrelerinin ML tahmini problemi için ABC algoritması Levy uçuşları ile birleştirilerek kullanılmış ve böylece algoritmanın arama kabiliyeti arttırılarak daha iyi sonuçlar elde edilmiştir.

Metasezgisel yöntemlerde arama uzayının belirlenmesine ilişkin belirli bir yöntem mevcut değildir ve genellikle rasgele belirlenmiş arama uzayları kullanılmaktadır (Abbasi ve ark., 2008; Zoraghi ve ark., 2012; Örkcü ve ark., 2015a; Örkcü ve ark., 2015b). Arama uzayının çok geniş belirlenmesi durumunda algoritma kabul edilemeyen sonuçlar verebilirken, çok dar belirlenmesi durumunda arama uzayı gerçek parametre değerlerini içermeyebilir. Bu problemi çözmek için 3-p Gamma dağılımının parametrelerinin ML ve LS ile tahmini için DE algoritmasının arama uzayının oluşturulmasında uyarlanmış en çok olabilirlik (MML) (Tiku, 1967) ve profil olabilirlik (PL) yöntemleri kullanılarak daha dar bir arama uzayı elde edilmiş ve böylece DE algoritmasının performansı arttırılmıştır. Burada, MML tahminlerinin gözlemlerin bir fonksiyonu olarak elde edilebilmesi ve etkin sonuçlar vermesinden dolayı MML yönteminin kullanılması tercih edilmiştir.

Herhangi bir metasezgisel yöntemi uygulamadan önce yöntemin parametrelerinin belirlenmesi gerekmektedir. Örneğin, PSO için popülasyon büyüklüğü, eylemsizlik katsayısı, hızlandırma katsayılar ve iterasyon sayısının önceden belirlenmesi gerekmektedir. Genel olarak, literatürde yer alan çalışmalarda parametre değerleri olarak ya literatürde geniş çapta kabul edilen değerler kullanılmış ya da deneme yanılma yöntemi ile parametreler belirlenmiştir (Schwaab ve ark., 2008; Özsoy

ve ark., 2017; Acıtaş ve ark., 2019). Bu çalışmada, bunlardan farklı olarak çok değişkenli Gamma dağılımının parametrelerinin tahmin problemi için PSO algoritmasının parametrelerinin bir deney tasarımı yöntemi olan Taguchi yöntemi (Taguchi, 1987) ile belirlenmesi önerilmiştir. Taguchi yöntemi, deney tasarımını daha az sayıda deney kullanarak gerçekleştirmesinden dolayı tercih edilmiştir.

Metasezgisel yöntemlerin parametre tahmin sürecindeki etkinlikleri ise Monte-Carlo simülasyonu yardımıyla araştırılmıştır. Simülasyon sonucunda elde edilen ortalama parametre tahmin değerleri, hata kareler ortalaması (MSE) ve kusur (Def) kriteri değerleri kullanılarak çeşitli istatistiksel testler ile algoritmaların performansları karşılaştırılmıştır. Ayrıca, çok kriterli karar verme (MCDM) tekniklerinden biri olan TOPSIS yöntemi kullanılarak ele alınan metasezgisel yöntemlerin çeşitli kriterler altında sıralanabileceği gösterilmiştir.

Bu tez çalışmasının literatüre katkıları,

- 3-p Weibull dağılımının parametrelerinin tahmini için literatürde önerilen SA, DE ve PSO algoritmalarından daha etkin sonuç veren Levy uçuşlu ABC algoritması önerilmesi,
- Literatürde yaygın olarak kullanılan metasezgisel yöntemlerinden SA, GA, PSO, DE ve ABC ile kısıtlı doğrusal olmayan optimizasyon algoritmalarından SQP ve IPA'nın performanslarının parametre tahminleri için karşılaştırılması,
- Metasezgisel yöntemleri sıralamak için MCDM yöntemlerinden biri olan TOPSIS yönteminin ilk defa kullanılması,
- DE algoritmasının performansını arttırmak için algoritmanın arama uzayını oluştururken MML ve PL yöntemleri kullanılarak elde edilen güven aralıklarının kullanılmasının önerilmesi,
- Gerçek veri uygulamalarının sunulması,
- Kesikli dağılımlar için parametre tahmin sürecinde ilk defa metasezgisel yöntemlerin kullanılması
- PSO algoritmasının parametrelerinin belirlenmesi için ilk defa Taguchi deney tasarımı yönteminin kullanılması

olarak verilebilir.

Bu tez çalışması altı bölümden oluşmaktadır.

İkinci bölümde, parametre tahmini ve metasezgisel yöntemler ile ilgili detaylı bir kaynak araştırması verilmiştir.

Üçüncü bölümde, tez kapsamında ele alınan dağılımlar, parametre tahmin yöntemleri, metasezgisel yöntemler, kısıtlı doğrusal olmayan optimizasyon teknikleri, deney tasarımı yöntemlerinden biri olan Taguchi yöntemi ve MCDM yöntemlerinden biri olan TOPSIS yöntemi ayrıntılı olarak verilmiştir. Ayrıca, bu çalışmada kullanılan istatistiksel materyaller açıklanmıştır.

Dördüncü bölümde, tek değişkenli dağılımların parametrelerine ilişkin tahminleri metasezgisel yöntemler yardımıyla elde edilmiştir. İlk olarak, 3-p Weibull dağılımının parametrelerinin ML tahmini için Levy uçuşlu ABC (LABC) algoritması önerilmiştir. Önerilen LABC algoritmasının performansı ABC, PSO, LPSO, DE, SA ve GA'nın performansları ile Monte Carlo simülasyonu çalışmaları üzerinden karşılaştırılmıştır. İkinci olarak, 3-p Gamma dağılımının parametrelerinin ML ve LS tahminleri için metasezgisel yöntemlerden SA, GA, DE, PSO ve ABC algoritmaları ve kısıtlı doğrusal olmayan optimizasyon tekniklerinden SQP ve IPA kullanılmış ve detaylı bir Monte-Carlo simülasyon çalışması sunulmuştur. Simülasyon sonuçları kullanılarak, metasezgisel yöntemlerin çözüm kalitesi, çözüm zamanı, kolaylık ve sağlamlık kriterlerine göre karşılaştırılması TOPSIS yöntemi kullanılarak yapılmıştır ve gerçek bir veri seti uygulaması sunulmuştur. Üçüncü olarak, 3-p Gamma dağılımının parametrelerinin hem ML hem de LS tahminlerini elde etmek amacıyla yeni bir DE algoritması yaklaşımı önerilmiştir. DE algoritmasının arama uzayını oluşturmak için MML tahmin yöntemi ve PL yönteminden faydalanılmıştır. Önerilen yöntemin performansı Monte-Carlo simülasyon çalışmaları ve bir gerçek veri kümesi uygulaması ile araştırılmıştır. Son olarak, kesikli Weibull dağılımının parametrelerinin ML tahminleri için metasezgisel yöntemlerden SA, GA, DE, PSO ve ABC algoritmaları kullanılmış ve detaylı bir Monte-Carlo simülasyon çalışması sunulmuştur. Simülasyon sonuçları kullanılarak, metasezgisel yöntemlerin çözüm kalitesi açısından karşılaştırılması yapılmış ve bir gerçek veri kümesi uygulaması sunulmuştur.

Beşinci bölümde, çok değişkenli dağılımların parametrelerinin tahminleri metasezgisel yöntemler kullanılarak elde edilmiştir. İlk olarak, çok değişkenli Gamma dağılımının parametrelerinin ML tahmini için PSO algoritmasının kullanılması önerilmiştir. PSO'nun performansı, eylemsizlik katsayısı, hızlandırma katsayıları, popülasyon sayısı gibi parametrelerinden çok fazla etkilendiğinden bu parametrelerin Taguchi yöntemi ile belirlenmesi önerilmiştir. Taguchi yöntemi ile parametreleri tasarlanan PSO algoritması ile çok değişkenli Gamma dağılımının parametrelerinin tahminindeki etkinliği Monte-Carlo simülasyon çalışmaları ile test edilmiştir. İkinci

olarak kesikli çok değişkenli Weibull dağılımının parametrelerinin ML tahminlerini elde etmek amacıyla metasezgisel yöntemlerden SA, GA, DE, PSO ve ABC algoritmaları kullanılmış ve detaylı bir Monte-Carlo simülasyon çalışması sunulmuştur. Simülasyon sonuçları yardımıyla metasezgisel yöntemler çözüm kalitesi açısından karşılaştırılmış ve gerçek bir gerçek veri kümesi uygulaması sunulmuştur.

Son bölümde, bu doktora tez çalışmasından elde edilen sonuçlar verilmiş ve öneriler sunulmuştur.

2. KAYNAK ARAŞTIRMASI

Literatürde birçok metasezgisel yöntem mevcuttur. Bu doktora tez çalışmasında, en yaygın olarak kullanılan metasezgisel yöntemlerden Tavlama Benzetimi, Parçacık Sürüsü Optimizasyonu, Genetik Algoritma, Diferansiyel Gelişim ve Yapay Arı Kolonisi ele alınmıştır.

Tavlama Benzetimi (SA), bir maddenin önce ısıtılması daha sonra kristal yapının elde edilmesi için yavaşça soğutulmasına ihtiyaç duyan tavlama sürecinin ilkelerine dayalı bir algoritmadır. SA algoritmasının optimizasyon problemleri için kullanılması, Metropolis ve ark. (1953)'nın istatistiksel mekanizma alanındaki çalışmalarından ilham alınarak ilk olarak Kirkpatrick ve ark. (1983) tarafından önerilmiştir (Abbasi ve ark., 2006; Talbi, 2009; Yang, 2010).

Genetik algoritma (GA), herhangi bir problemin çözümünde biyolojik sistemlerin gelişim sürecini taklit eden stokastik bir arama algoritmasıdır. GA yöntemi, Darwin'in evrimleşme sürecini anlamak için ilk olarak Holland (1975) tarafından önerilmiştir ve daha sonra Goldberg ve Holland (1988) tarafından optimizasyon problemleri ve makine öğrenmesi için kullanılmıştır.

Parçacık sürüsü optimizasyonu (PSO), kuşların ve balıkların yiyecek bulmada sergiledikleri davranışlarından esinlenerek doğrusal olmayan fonksiyonların optimizasyonu için Eberhart ve Kennedy (1995) ve Kennedy ve Eberhart (1995) tarafından önerilen bir algoritmadır.

Diferansiyel gelişim (DE), işleyiş ve operatörleri bakımından GA yöntemine dayalı bir algoritma olup ilk olarak Storn (1996) ve Storn ve Price (1997) tarafından önerilmiştir.

Yapay arı kolonisi (ABC) algoritması arıların yiyecek arama davranışlarının modellenmesi temeline dayalı bir algoritma olup ilk olarak Karaboga (2005) tarafından önerilmiştir. Daha sonra, gerçek-parametreli optimizasyon problemlerini çözmek amacıyla Akay ve Karaboğa (2012) tarafından bu algoritma uyarlanmıştır.

Son yıllarda, GA, SA, PSO, DE ve ABC gibi metasezgisel yöntemlerin parametre tahmininde kullanımı ile ilgili çalışmalar oldukça artış göstermektedir. Bu çalışmalardan bazıları aşağıda verilmiştir.

Shao (2004), 3-parametreli Burr XII dağılımının parametrelerini tahmin etmek için ML yöntemini kullanarak sezgisel bir algoritma önermiştir. Sansürsüz ve sağdan sansürlü gerçek veri kümeleri kullanarak farklı durumları örneklendirmiştir.

Altunkaynak ve Alptekin (2004), doğrusal olmayan regresyonda parametre tahmini için alternatif bir yöntem olarak Genetik Algoritma (GA)'nın kullanılmasını önermişlerdir. Üç farklı veri kümesi üzerinde uygulama yapmışlar ve GA'nın diğer yöntemlere kıyasla daha etkili çözümler elde ettiğini göstermişlerdir.

Abbasi ve ark. (2006), Tavlama Benzetimi (SA) algoritmasını kullanarak 3-parametreli Weibull dağılımının parametre tahmini için yeni bir yaklaşım önermişler ve farklı örnekler üzerinde bu yaklaşımın güçlü ve doğru olduğu sonucuna ulaşmışlardır.

Vera ve Díaz-García (2008), genişletilmiş log-normal dağılımı ve 3 parametreli log-normal dağılımının parametrelerinin bulunmasında ML tahmin problemini çözmek için SA algoritmasını önermişlerdir. Pozitif yönde ve negatif yönde çarpık log-normal dağılımı incelemişlerdir. Önerilen algoritmanın etkinliğini simülasyon çalışması ve gerçek veri üzerinden test etmişlerdir.

Schwaab ve ark. (2008), lineer olmayan modellerde parametrelerin tahmini için parçacık sürüsü optimizasyonu (PSO) algoritmasını önermişlerdir. Kimya mühendisliği alanında üç farklı uygulama ile yöntemin uygulanabilirliğini göstermişlerdir

Zhang ve Lon Ng (2010), çok değişkenli t-copula modellerinde parametre tahmini için metasezgisel yöntemlerden diferansiyel gelişim (DE) algoritması ve eşik değer kabulü kullanmışlardır. Deneysel çalışmalar sonucunda ele alınan metasezgisel yöntemlerin güçlü ve doğru tahminler verdiğini göstermişlerdir.

Akman ve ark. (2010), istatistiksel dağılımların parametrelerinin ML tahmini için GA yöntemini uygulamışlardır. GA için üç farklı popülasyon azaltma yöntemi geliştirmişler ve karma ters normal dağılımı kullanarak uçak verileri üzerine bir uygulama sunmuşlardır.

Abbasi ve ark. (2011), 3-p Weibull dağılımının parametre tahmini için değişken komşu arama (VNS) ve SA yaklaşımlarına dayalı yeni bir karma yöntem önermişlerdir. Önerilen yöntemin doğru tahmin ve hesaplama zamanı açısından performansını çeşitli sayısal örnekler ile değerlendirmiş ve mevcut yöntemler ile karşılaştırmışlardır. Ayrıca, örneklem büyüklüğüne bağlı olarak performans değerlendirmesi yapmışlardır. Karşılaştırma sonuçları önerilen yöntemin daha az hesaplama zamanı ile daha doğru sonuçlara ulaştığını göstermiştir.

Zoraghi ve ark. (2012), 4-p Bur III dağılımının parametre tahmini için VNS ve iteratif yerel arama (ILS) algoritmalarını birleştirerek yeni bir karma algoritma önermişlerdir. Önerdikleri algoritmanın parametrelerini Taguchi deney tasarımı

yöntemini kullanarak belirlemişlerdir. Bazı sayısal örnekler ve simülasyon çalışmalar üzerinden önerilen algoritmanın performansının yeterli olduğunu göstermişlerdir.

Shin ve ark. (2014), karma normal dağılımda parametre tahmini için metasezgisel yöntemlerden biri olan GA yaklaşımını kullanarak yeni bir model geliştirmişlerdir. Bu geliştirilen modelin performansını ölçmek için gerçekleştirilen simülasyon çalışmaları ve pratik uygulamalar sonucunda, önerilen modelin küçük boyutlu örneklemlerde bile beklenti maksimizasyonu (EM) algoritmasına göre daha doğru tahmin değerleri verdiği ve MSE değerlerinin daha düşük olduğu sonucuna ulaşmışlardır.

Wang ve Huang (2014), tam ve çoklu sansürlü verilerle iki Weibull dağılımının karışımından elde edilen dağılımın parametrelerine ilişkin ML tahminlerini elde etmek için PSO algoritmasını önermişlerdir. Sansürlü ve sansürsüz vakalarda farklı parametre değerleri ve örneklem boyutları için ML'nin performansını PSO algoritması, yarı Newton yöntemi ve beklenti maksimizasyonu (EM) algoritması ile değerlendirmek için bir simülasyon çalışması gerçekleştirmişlerdir. PSO algoritmasının çoğu durumda daha iyi performans gösterdiğini ifade etmişlerdir. Ayrıca, önerilen yöntemin uygulanabilirliğini göstermek için iki sayısal örnek sunmuşlardır.

Banerjee ve Abu-Mahfouz (2014), lineer olmayan dinamik sistemlerde parametre tahmini için PSO ve DE algoritmalarının karşılaştırmalı bir analizini sunmuşlardır. Bu algoritmaların çeşitli varyasyonlarını kullanarak dört PSO algoritması ve üç DE algoritmasını, değişen karmaşıklık derecelerine sahip 15 farklı parametre tahmin problemine uygulamışlardır. Elde edilen sonuçları istatistiksel yöntemler ile karşılaştırılarak algoritmalar arasında önemli farklılıklar tespit etmişlerdir.

Örkcü ve ark. (2015a), 3-p Weibull dağılımının parametre tahmini için DE algoritmasını kullanmışlardır. Önerilen algoritmanın daha az hesaplama zamanı ile amaç fonksiyonunun maksimum değerine daha hızlı bir yakınsama sergilediğini, doğru tahmin sonuçları verdiğini ve 3-p Weibull dağılımının parametre tahmini için yeterli olduğunu 3 sayısal örnek üzerinden göstermişlerdir. Ayrıca, DE' den elde edilen sonuçlar ile GA ve SA' dan elde edilen sonuçları karşılaştırmışlardır.

Örkcü ve ark. (2015b), 3-p Weibull dağılımının parametre tahmini için PSO algoritmasını kullanmışlardır. Eylemsizlik katsayısı, hızlandırma katsayısı, parçacık sayısı ve arama uzayı üzerinde çeşitli modifikasyonlar yaparak oluşturulan farklı PSO seçenekleri için detaylı bir araştırma sunmuşlardır. Farklı PSO seçenekleri kullanılarak elde edilen sonuçları karşılaştırmak için 3 sayısal örnek vermişlerdir. PSO yaklaşımının

3-p Weibull dağılımının parametre tahmini için tatmin edici olduğu sonucuna ulaşmışlardır. Ayrıca, PSO yaklaşımından elde edilen sonuçlar ile GA, SA ve DE algoritmalarından elde edilen sonuçları karşılaştırmışlardır.

Carneiro ve ark. (2016), Brezilya Kuzeydoğu Bölgesi'ndeki rüzgâr kaynakları için Weibull parametrelerinin tahmininde PSO yöntemini uygulamışlardır. Ayrıca, Weibull dağılım parametrelerini hesaplanmasında PSO'nun performansını karşılaştırmak için moment, ampirik, eşdeğer enerji, ML ve enerji pateni faktörü gibi beş farklı yöntem uygulamışlardır. Analiz edilen tüm bölgeler için en iyi performansın, %99'dan yüksek bir korelasyon ve sıfıra yakın bir hata ile PSO yönteminden elde edildiği sonucuna ulaşmışlardır.

Jha (2016), sağdan sansürlü 3 parametreli Weibull dağılımında parametrelerin ML tahmini için evrimsel bir hesaplama yöntemi olan DE algoritmasını önermiştir. %20, %30 ve %50 sansürleme yoğunluklarını içeren simülasyon yardımıyla üretilen dört farklı veri seti üzerinde uygulamalar gerçekleştirmiştir. Sağdan sansürlü 3-p Weibull dağılımının parametrelerinin ML tahmininde DE'nin güçlü bir yöntem olduğunu vurgulamıştır.

Aledo ve ark. (2016), genelleştirilmiş Mallows modellerinde parametre tahmini için ILS, VNS, GA, dağılım algoritmalarının tahmini (EDAs) gibi çeşitli metasezgisel algoritmaların karşılaştırmasını yapmışlardır. 22 tane farklı karmaşıklıktaki gerçek veri kümesi kullanarak algoritmaları doğruluk, iterasyon sayısı ve hesaplama zamanı açısından karşılaştırmışlardır.

Özsoy ve ark. (2017), lineer olmayan regresyon modellerinde parametre tahmini için PSO yaklaşımını kullanmışlardır. Farklı zorluk derecelerine sahip en iyi bilinen 28 lineer olmayan regresyon modeli için PSO algoritmasının performansını test etmişlerdir. PSO algoritmasının daha az iterasyonda daha doğru sonuçlar verdiğini ve lineer olmayan regresyon modellerinin tahmini için yeterli olduğunu göstermişlerdir.

Özsoy ve ark. (2017), PSO algoritmasını kullanarak Burr III dağılımı için alternatif bir parametre tahmin yöntemi önermişlerdir. Simülasyon çalışması ile PSO yönteminin, Burr III dağılımın parametre tahmini için oldukça yeterli olduğunu sonucuna ulaşmışlardır.

Okafor ve ark. (2018), iki parametreli Weibull dağılımının parametrelerini tahmin etmek için PSO algoritmasını kullanmışlardır. Genel olarak PSO'nun, ML'ye dayalı analitik yönteme kıyasla daha iyi parametre tahmin değerleri verdiğini göstermişlerdir.

Yalçınkaya ve ark. (2018), çarpık Normal dağılımın en çok olabilirlik parametrelerinin tahmininde GA'yı kullanmışlardır. GA'dan elde edilen tahminleri Newton-Raphson, Nelder-Mead ve İteratif yeniden ağırlıklandırma algoritması gibi iteratif yöntemler kullanarak elde edilen tahminler ile karşılaştırmışlardır. Simülasyon çalışmaları sonucunda, GA kullanarak elde edilen tahminlerin yansızlık, MSE ve Def kriteri açısından diğer yöntemlere göre en iyi olduğu sonucuna ulaşmışlardır.

Acıtaş ve ark. (2019), 3-p Weibull dağılımının parametre tahmini için uyarlanmış en çok olabilirlik (MML) tahminlerine dayalı olarak elde edilen güven aralıklarını PSO algoritmasında arama uzayı olarak alıp yeni bir PSO algoritması yaklaşımı önermişlerdir. Önerilen yaklaşım ile arama uzayını daraltarak PSO algoritmasının performansını arttırmışlardır. Simülasyon çalışmaları ve gerçek veri uygulaması üzerinden önerilen PSO yaklaşımının standart PSO yaklaşımına göre üstünlüğünü göstermişlerdir.

Yang ve ark. (2019), 3-p Weibull dağılımının parametrelerinin ML tahmini için evrim algoritmalarından biri olan Kovaryans Matrisi Adaptasyon Evrim Stratejisi (CMA-ES) algoritmasını kullanmışlardır. Simülasyon çalışmaları ile bu algoritmanın uygulanabilirliğini göstermişlerdir. Ayrıca gerçek veri uygulaması ile CMA-ES algoritmasının GA, SA ve PSO algoritmalarından daha iyi olduğu sonucuna ulaşmışlardır.

Karakoca ve Pekgör (2019), ilerleyen tür tip 2 sansürlü örneklemlerde Weibull dağılımının parametrelerinin ML tahmini için GA'yı kullanmışlardır. Detaylı bir simülasyon çalışması aracılığıyla GA ve Newton yöntemlerinin performanslarını değerlendirmişler ve karşılaştırılan tüm durumlarda GA'nın genellikle ölçek parametresi için daha iyi sonuçlar verdiğini bulmuşlardır.

Özsoy ve ark. (2020), sezgisel optimizasyon yaklaşımlarından GA, DE, PSO ve SA kullanarak genelleştirilmiş Gamma dağılımı için alternatif bir parametre tahmin yöntemi önermişlerdir. Simülasyon çalışmaları ile sezgisel yaklaşımların oldukça doğru tahminler sağladığı ve çoğu durumda DE'nin parametre tahminlerinin yanlılık değerleri açısından diğer yöntemlerden daha iyi performans sergilediği sonucuna ulaşmışlardır. Ayrıca, gerçek bir veri seti uygulaması sunmuşlardır.

Bu tez çalışmasında bazı sürekli ve kesikli tek değişkenli ve çok değişkenli dağılımların parametre tahminlerinin metasezgisel yöntemler ile elde edilmesi ele alınmıştır. Literatürde, sürekli dağılımların parametre tahmininde kullanılan çok sayıda

metasezgisel yöntem mevcut olmasına rağmen kesikli dağılımların parametre tahmininde metasezgisel yöntemleri kullanan bir çalışmaya rastlanılmamıştır.

Tek değişkenli ve çok değişkenli kesikli dağılımların parametre tahmini ile ilgili gerçekleştirilen çalışmalardan bazıları aşağıda verilmiştir.

Khan ve ark. (1989), tip-1 kesikli Weibull dağılımının parametrelerinin tahmini için oranlar yöntemini önermişlerdir. Önerilen yöntemi, çeşitli parametre değerleri ve örneklem büyüklükleri için momentler yöntemi ile karşılaştırılmıştır.

Bracquemond ve Gaudoin (2003), güvenilirlik çalışmalarında kullanılan kesikli olasılık dağılımlarının kapsamlı bir incelemesini sunmuşlardır. Ayrıca, geometrik, tip-1 ve tip-3 kesikli Weibull gibi bazı kesikli dağılımların ML parametre tahminlerini elde etmek için simülasyon çalışmaları sunmuşlardır.

Krishna ve Pundir (2009), kesikli Burr ve Pareto dağılımlarını tanıtmışlar ve bu dağılımların özelliklerini incelemişlerdir. Bu dağılımlarının parametrelerinin ML tahminlerini elde etmek için Newton-Raphson yöntemini kullanarak simülasyon çalışmaları yapmış ve gerçek veri kümesi uygulaması sunmuşlardır.

Nekoukhou ve ark. (2013), tip-2 kesikli genelleştirilmiş üstel dağılımı ve özelliklerini tanıtmışlardır. Simülasyon çalışmaları ve bir gerçek veri kümesi uygulaması ile bu dağılımın parametrelerinin ML tahminlerini Newton Raphson yöntemi kullanarak elde etmişlerdir.

Barbiero (2013), tip-3 kesikli Weibull dağılımının parametrelerinin tahmini için oranlar, ML ve momentler yönteminin karşılaştırmalı bir analizini sunmuştur. Monte-Carlo simülasyon çalışmaları ve gerçek veri uygulamaları ile, oranlar yönteminin sadece küçük örneklemler için belirli parametre değerlerinde iyi tahminler verdiği, ML ve momentler yönteminin ise genellikle daha iyi tahminler verdiği sonucuna ulaşmıştır.

Barbiero (2016), tip-3 kesikli Weibull dağılımının parametrelerinin tahmini için oranlar, ML ve momentler yönteminin karşılaştırmıştır. Monte-Carlo simülasyon çalışmaları ve gerçek veri kümesi uygulamaları sonucunda, küçük örneklemlerde oranlar yönteminin daha kullanışlı olduğu sonucuna ulaşmıştır. Ayrıca, kesikli Weibull dağılımı için R programında bir paket yazılım uygulaması sunulmuştur.

Barbiero (2018), tip-2 kesikli Weibull dağılımının parametrelerinin tahmini için ML, oranlar ve LS yönteminin karşılaştırmalı bir analizini yapmıştır. Monte-Carlo simülasyon çalışmaları ve gerçek veri uygulamaları sonucunda, ML yönteminin tip-2 kesikli Weibull dağılımının parametrelerinin tahmini için en uygun yöntem olduğu sonucuna ulaşılmıştır.

Nekoukhou ve Kundu (2017), iki değişkenli kesikli genelleştirilmiş üstel dağılımı önermişlerdir. Bu dağılımın parametrelerinin ML tahminlerini bulabilmek amacıyla yeni bir Beklenti Maksimizasyonu (EM) algoritması önermişlerdir. Önerilen modelin etkinliğini göstermek için bir gerçek veri kümesi uygulaması yapmışlardır. Hem kesikli genelleştirilmiş üstel dağılımın hem de EM algoritmasının uygulamada çok iyi olduğunu sonucuna ulaşmışlardır.

Kundu ve Nekoukhou (2018), tek ve iki değişkenli geometrik kesikli genelleştirilmiş üstel dağılımları tanıtmışlardır. Bu dağılımların parametrelerinin ML tahmini için EM algoritmasının kullanılmasını önermişlerdir. Simülasyon çalışmaları ve gerçek veri kümesi uygulaması üzerinde önerilen dağılımların ve EM algoritmasının oldukça iyi olduğunu göstermişlerdir.

Eliwa ve El-Morshedy (2018), iki değişkenli kesikli ters Weibull dağılımını önermiş ve bu dağılımın bazı istatistiksel ve matematiksel özelliklerini sunmuşlardır. Dağılımın parametrelerini tahmin etmek için ML yöntemi kullanılmıştır. Simülasyon çalışmaları sonucunda, ML yönteminin bu dağılımın parametrelerinin tahmininde oldukça iyi olduğu sonucuna ulaşmışlardır. Ayrıca, iki gerçek veri kümesi uygulaması sunmuşlar ve bu dağılımın uygulamada kullanışlı olabileceği sonucuna ulaşmışlardır.

Kundu ve Nekoukhou (2019), iki değişkenli kesikli Weibull dağılımı önermişlerdir. Bu dağılımın parametrelerinin ML tahminlerini elde etmek için EM algoritması ve Bayes tahminlerini hesaplamak için bir Gibbs örnekleme yöntemi önermişlerdir. Önerilen dağılımın ve parametre tahmin yöntemlerinin uygulamada nasıl çalıştığını göstermek amacıyla iki gerçek veri kümesi uygulaması gerçekleştirmişlerdir. Hem EM algoritmasının hem Gibbs örnekleme yönteminin oldukça iyi çalıştığı sonucuna ulaşmışlardır.

El-Morshedy ve ark. (2020), iki değişkenli üstelleştirilmiş kesikli Weibull dağılımını önermişler ve bu dağılımın özelliklerini ayrıntılı olarak incelemişlerdir. Önerilen dağılımın parametrelerinin ML tahmini için simülasyon çalışmaları ve gerçek veri kümesi uygulamaları sunmuşlardır.

3. MATERYAL VE YÖNTEM

Bu bölümde, tez kapsamında ele alınan dağılımlar, parametre tahmin yöntemleri, metasezgisel yöntemler, optimizasyon teknikleri, Taguchi yöntemi ve TOPSIS yöntemi detaylı olarak verilmiştir. Son olarak, çalışmada kullanılan istatistiksel kriterler açıklanmıştır.

3.1. Bazı Tek Değişkenli ve Çok Değişkenli Dağılımlar

Bu kesimde, tez kapsamında ele alınan 3-parametreli (3-p) Weibull, 3-parametreli (3-p) Gamma, kesikli Weibull, çok değişkenli Gamma ve kesikli çok değişkenli Weibull dağılımları tanıtılmıştır.

3.1.1. 3-p Weibull dağılımı

İlk olarak Weibull (1939) tarafından önerilen Weibull dağılımı, esnek yapısından ve çok çeşitli deneysel gözlemlerin dağılımına uygunluğundan dolayı güvenirlik ve yaşam zamanı çalışmalarında en yaygın olarak kullanılan sürekli dağılımlardan biridir (Nagatsuka ve ark., 2013). Weibull dağılımı, mühendislik, fizik, kimya, meteoroloji, tıp, eczacılık, kalite kontrol, ekonomi ve iş, biyoloji, jeoloji ve coğrafya gibi birçok alanda uygulanmakta olup modellemede oldukça kullanışlıdır (Almalki ve Nadarajah, 2014).

X rasgele değişkeni, 3-p Weibull dağılımına sahip olmak üzere, olasılık yoğunluk fonksiyonu (oyf) ve birikimli dağılım fonksiyonu (df),

$$f_X(x) = \frac{\beta}{\eta}\left(\frac{x-\gamma}{\eta}\right)^{\beta-1} \mathrm{e}^{-\left(\frac{x-\gamma}{\eta}\right)^{\beta}};\ x \geq \gamma,\ \eta > 0,\ \beta > 0 \tag{3.1}$$

ve

$$F_X(x) = 1 - \mathrm{e}^{-\left(\frac{x-\gamma}{\eta}\right)^{\beta}};\ x \geq \gamma,\ \eta > 0,\ \beta > 0 \tag{3.2}$$

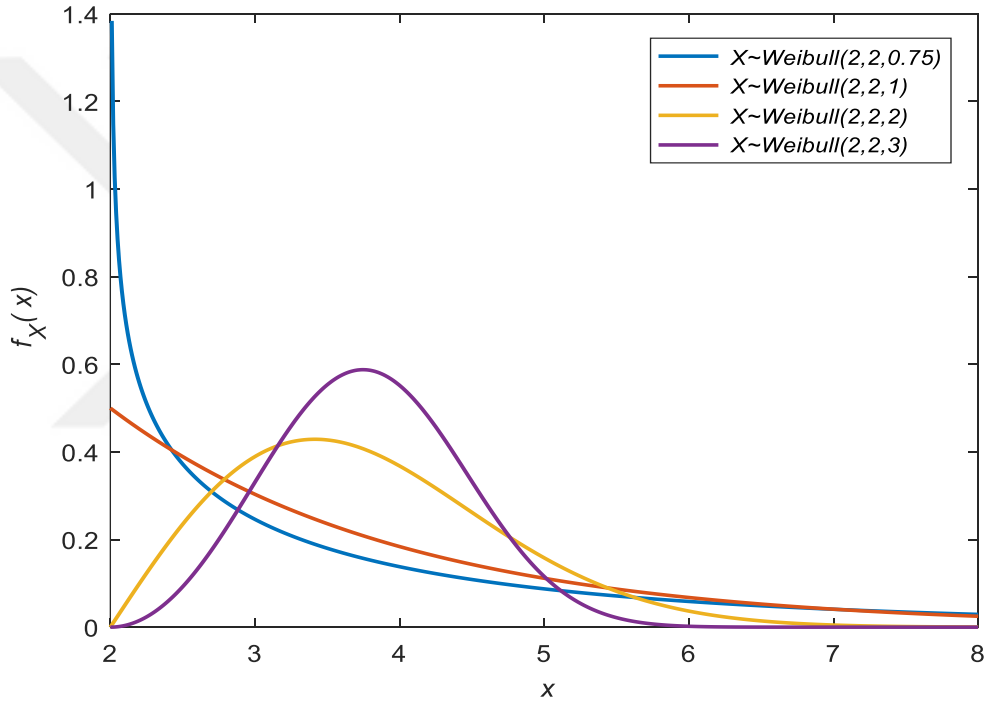
olarak tanımlanır. $X \sim \text{Weibull}(\gamma, \eta, \beta)$ ile gösterilir. Burada:

γ : konum parametresini

η : ölçek parametresini

β : şekil parametresini göstermektedir (Johnson ve ark., 1994).

3-p Weibull dağılımı, şekil parametresi β 'nın değerlerine göre farklı şekillerde olmaktadır. Örneğin, $\beta \leq 1$ olduğunda dağılım J şeklinde, $\beta > 1$ olduğunda ise çan şeklindedir. Ayrıca, $\beta = 1$ alındığında Üstel dağılım ve $\beta = 2$ alındığında Rayleigh dağılımı elde edilmektedir. Çeşitli β değerleri için Weibull dağılımının olasılık yoğunluk fonksiyonu grafikleri, Şekil 3.1'de verilmiştir.



Şekil 3.1. 3-p Weibull dağılımının şekil parametresi β 'nın çeşitli değerleri için oyf gösterimi

3.1.2. 3-p Gamma dağılımı

Gamma dağılımı, özellikle güvenilirlik veya yaşam zamanı için hidroloji, finans gibi çeşitli alanlarda çarpık verilerin modellenmesinde yaygın olarak kullanılan sürekli dağılımlardan biridir (Hirose, 1995; Basak ve Balakrishnan, 2012; Lakshmi ve Vaidyanathan, 2016).

X , 3-p Gamma dağılımına sahip bir rasgele değişken olmak üzere X rd'nin olasılık yoğunluk fonksiyonu ve birikimli dağılım fonksiyonu,

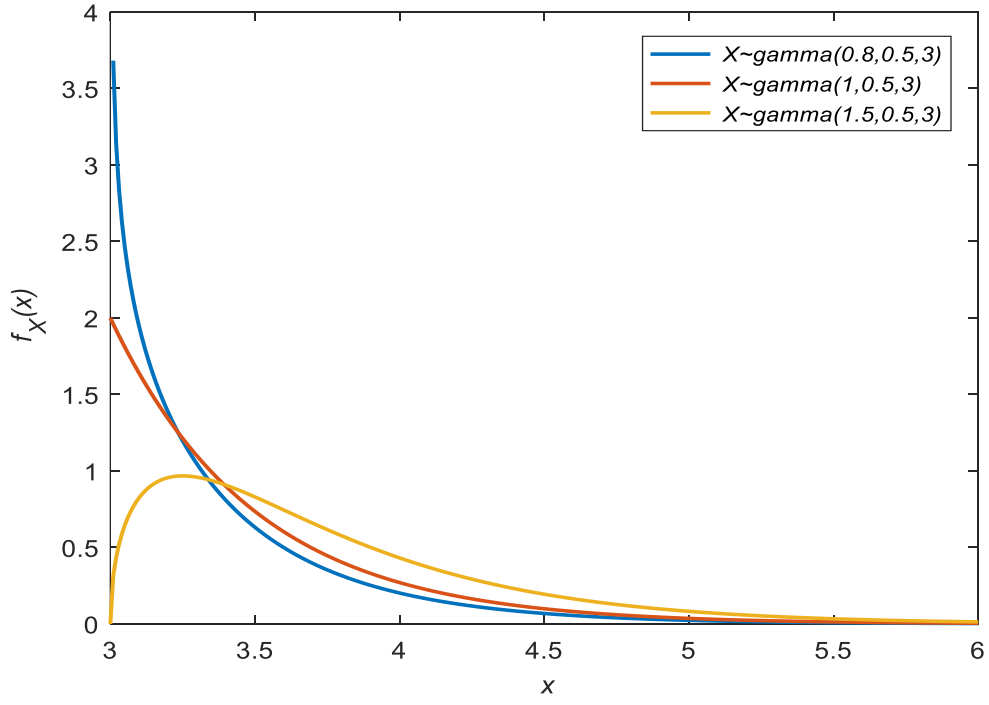
$$f(x;\sigma,\beta,\mu)=\frac{1}{\Gamma(\sigma)\beta^{\sigma}}(x-\mu)^{\sigma-1}\exp\left(-\frac{x-\mu}{\beta}\right),\quad x\geq\mu,\sigma>0,\beta>0. \tag{3.3}$$

ve

$$F(x;\sigma,\beta,\mu)=\frac{1}{\Gamma(\sigma)}\gamma\left(\sigma,\frac{x-\mu}{\beta}\right),\quad x\geq\mu,\sigma>0,\beta>0 \tag{3.4}$$

biçiminde tanımlanır ve $X\sim Gamma(\sigma,\beta,\mu)$ ile gösterilir. Eşitlik (3.3) ve (3.4)'te $\Gamma(\cdot)$ gamma fonksiyonunu, $\gamma(\cdot)$ tamamlanmamış gamma fonksiyonunu göstermektedir ve $\gamma\left(\sigma,\frac{x-\mu}{\beta}\right)=\int_0^{(x-\mu)/\beta}t^{\sigma-1}\exp(-t)dt$ biçiminde ifade edilmektedir. Burada, σ şekil parametresini, β ölçek parametresini ve μ konum parametresini göstermektedir (Johnson ve ark., 1994; Balakrishnan ve Wang, 2000; Lakshmi ve Vaidyanathan, 2016).

3-p Gamma dağılımı, şekil parametresi σ'nın değerlerine göre üç özel durum içermektedir. $\sigma\leq1$ ise dağılım "J" şeklinde, $\sigma>1$ ise dağılım çan-şeklinde olmakta ve $\sigma=1$ ise dağılım üstel dağılım olmaktadır (Basak ve Balakrishnan, 2012). Çeşitli σ değerleri için Gamma dağılımının olasılık yoğunluk fonksiyonu grafikleri, Şekil 3.2'de verilmiştir.



Şekil 3.2. 3-p Gamma dağılımının şekil parametresi σ'nın çeşitli değerleri için oyf gösterimi

3.1.3. Kesikli Weibull dağılımı

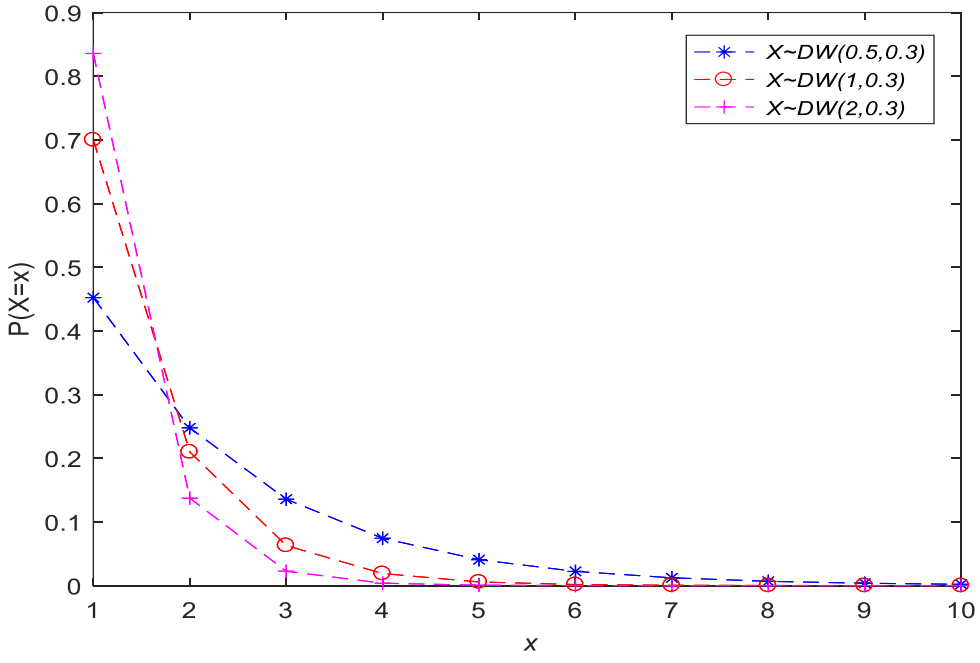
İlk olarak Nakagawa ve Osaki (1975) tarafından önerilen kesikli Weibull dağılımı, kesikli başarısızlık verilerinin modellenmesinde güvenirlik ve yaşam zamanı çalışmalarında kullanılmaktadır (Barbiero, 2016). X , kesikli Weibull dağılımına sahip bir rasgele değişken olmak üzere X rd'nin olasılık ve dağılım fonksiyonları,

$$P(X=x)=q^{x^{\beta}}-q^{(x+1)^{\beta}},\ 0<q<1, \beta>0,\ x=0,1,2,... \tag{3.5}$$

ve

$$F_X(x)=1-q^{x^{\beta}},\ 0<q<1, \beta>0,\ x=0,1,2,... \tag{3.6}$$

olarak tanımlanır ve $X \sim DW(\beta, q)$ ile gösterilir. Burada; q hayatta kalma olasılığı, β ise şekil parametresidir. Kesikli Weibull dağılımında, $\beta=1$ alındığında, $p=1-q$ parametreli Geometrik dağılım elde edilir (Bracquemond ve Gaudoin, 2003; Barbiero, 2016). Çeşitli parametre değerleri için kesikli Weibull dağılımının olasılık fonksiyonu grafikleri, Şekil 3.3'te verilmiştir.



Şekil 3.3. Kesikli Weibull dağılımının farklı parametreleri için of gösterimi

3.1.4. Çok değişkenli Gamma dağılımı

Çok değişkenli Gamma dağılımı, stokastik modelleme, hidroloji ve güvenirlik uygulamalarında yaygın olarak kullanılmaktadır. Literatürde, Cherian (1941), Jensen (1970), Sarmanov (1970), Royen (1991), Mathai ve Moschopoulos (1992), Carpenter ve Diawara (2007), vb. tarafından önerilen çeşitli çok değişkenli Gamma dağılımları bulunmaktadır. Bu çalışmada, Mathai ve Moschopoulos (1992) tarafından önerilen çok değişkenli Gamma dağılımı ele alınmıştır.

$X_i \sim Gamma(\alpha_i, \beta, \mu)_i$, $i = 1, 2, ...k$ birbirinden bağımsız rasgele değişkenler olmak üzere X_i rd'nin olasılık yoğunluk fonksiyonu,

$$f(x_i; \sigma_i, \beta, \mu_i) = \frac{1}{\Gamma(\sigma_i)\beta^{\alpha_i}} (x - \mu_i)^{\alpha_i - 1} \exp\left(-\frac{x - \mu_i}{\beta} \right), x_i \geq \mu_i, \sigma_i > 0, \beta > 0, \mu_i \in \mathbb{R}. \quad (3.7)$$

biçiminde yazılabilir. $Z_1 = X_1,\ Z_2 = X_1 + X_2,\ ...,\ Z_k = X_1 + X_2 + ... + X_k$ olmak üzere, $\boldsymbol{Z} = (Z_1, Z_2, ...Z_k)$ rasgele vektörü k değişkenli Gamma dağılımına sahip olup $\boldsymbol{Z}$'nin olasılık yoğunluk fonksiyonu,

$$f(z_1, z_2, ..., z_k) = \frac{(z_1 - \mu_1)^{\sigma_1 - 1} (z_2 - z_1 - \mu_2)^{\sigma_2 - 1} ... (z_k - z_{k-1} - \mu_k)^{\sigma_k - 1} \exp\left(-\frac{z_k - \sum_{i=1}^{k} \mu_i}{\beta} \right)}{\beta^{\sum_{i=1}^{k} \sigma_i} \prod_{i=1}^{k} \Gamma(\sigma_i)} \quad (3.8)$$

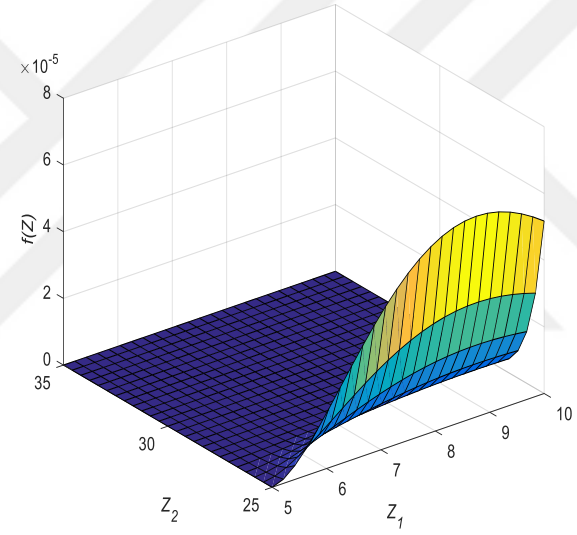
olarak yazılabilir. Burada, σ_i şekil parametresi, β ölçek parametresi ve μ_i konum parametresi olup $\sigma_i > 0, \beta > 0, \mu_i \in \mathbb{R}, z_{i-1} + \mu_i < z_i, i = 1, 2, ..., k, z_k < \infty, \mu_1 < z_1$'dir.

Ayrıca $X_1 = Z_1,\ X_2 = Z_2 - Z_1, ...,\ X_k = Z_k - Z_{k-1}$ olduğundan Eşitlik (3.8)

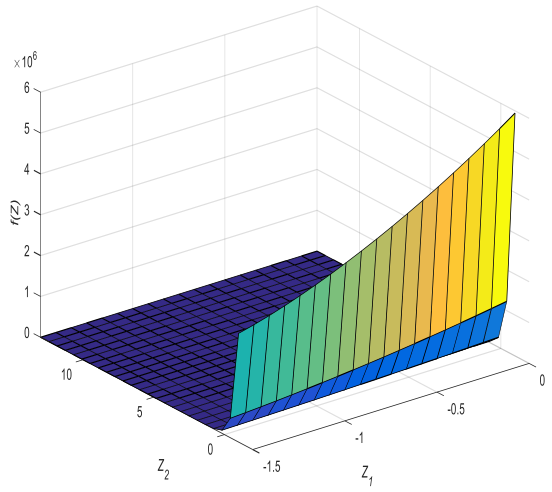
$$f(x_1, x_2, ..., x_k) = \frac{(x_1-\mu_1)^{\sigma_1-1}(x_2-\mu_2)^{\sigma_2-1}...(x_k-\mu_k)^{\sigma_k-1}\exp\left(-\frac{\sum_{i=1}^{k}x_i-\sum_{i=1}^{k}\mu_i}{\beta}\right)}{\beta^{\sum_{i=1}^{k}\sigma_i}\prod_{i=1}^{k}\Gamma(\sigma_i)} \quad (3.9)$$

biçiminde de ifade edilebilmektedir (Balakrishnan ve Lai, 2009; Vaidyanathan ve Lakshmi, 2015).

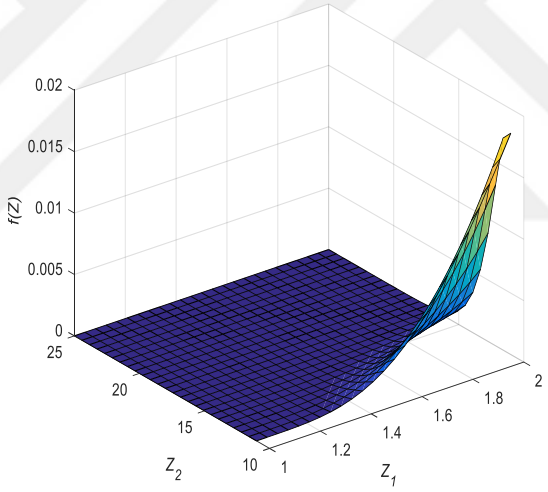
Çeşitli parametre değerleri için iki değişkenli Gamma dağılımına ilişkin olasılık yoğunluk fonksiyonu grafikleri, Şekil 3.4-3.6'da verilmiştir.



Şekil 3.4. $\sigma_1 = 2.8, \sigma_2 = 4, \beta = 0.5, \mu_1 = 5, \mu_2 = 10$ için iki değişkenli Gamma dağılımına ilişkin oyf gösterimi



Şekil 3.5. $\sigma_1 = 0.75, \sigma_2 = 5, \beta = 0.5, \mu_1 = -2, \mu_2 = 5$ için iki değişkenli Gamma dağılımına ilişkin oyf gösterimi



Şekil 3.6. $\sigma_1 = 5, \sigma_2 = 2.6, \beta = 0.75, \mu_1 = 1, \mu_2 = 8$ için iki değişkenli Gamma dağılımına ilişkin oyf gösterimi

3.1.5. Çok değişkenli kesikli Weibull dağılımı

Bu çalışmada, Kundu ve Nekoukhou (2019) tarafından önerilen iki değişkenli kesikli Weibull dağılımı ele alınmıştır. $U_1 \sim \mathrm{DW}(\beta, q_1)$, $U_2 \sim \mathrm{DW}(\beta, q_2)$ ve $U_0 \sim \mathrm{DW}(\beta, q_0)$ birbirinden bağımsız rasgele değişkenler ve $X_1 = \min\{U_1, U_0\}$, $X_2 = \min\{U_2, U_0\}$ olmak üzere (X_1, X_2) rasgele vektörü β, q_0, q_1 ve q_2 parametreli

iki değişkenli kesikli Weibull dağılımına sahiptir ve $(X_1, X_2) \sim BDW(\beta, q_0, q_1, q_2)$ ile gösterilir. (X_1, X_2) rasgele vektörünün ortak olasılık fonksiyonu,

$$f(x_1, x_2) = \begin{cases} f_1(x_1, x_2), & 0 \le x_1 \le x_2 \\ f_2(x_1, x_2), & 0 \le x_2 \le x_1 \\ f_0(x), & 0 \le x_1 = x_2 = x \end{cases} \tag{3.10}$$

biçiminde ifade edilir. Burada,

$$f_1(x_1, x_2) = \left((q_1)^{x_1^\beta} - (q_1)^{(x_1+1)^\beta} \right) \left((q_0 q_2)^{x_2^\beta} - (q_0 q_2)^{(x_2+1)^\beta} \right) \tag{3.11}$$

$$f_2(x_1, x_2) = \left((q_0 q_1)^{x_1^\beta} - (q_0 q_1)^{(x_1+1)^\beta} \right) \left((q_2)^{x_2^\beta} - (q_2)^{(x_2+1)^\beta} \right) \tag{3.12}$$

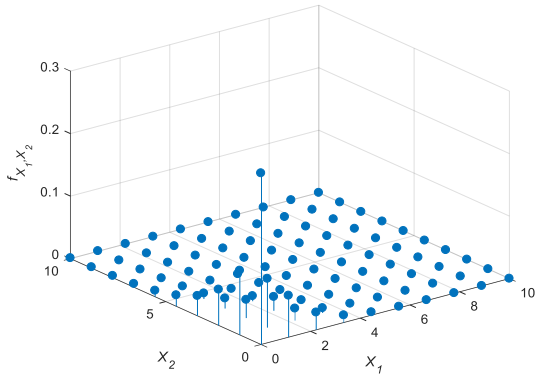
$$f_0(x) = (q_1)^{x^\beta} \times \left((q_0 q_2)^{x^\beta} - (q_0 q_2)^{(x+1)^\beta} \right) - (q_0 q_1)^{(x+1)^\beta} \times \left(q_2^{x^\beta} - q_2^{(x+1)^\beta} \right) \tag{3.13}$$

biçimindedir.

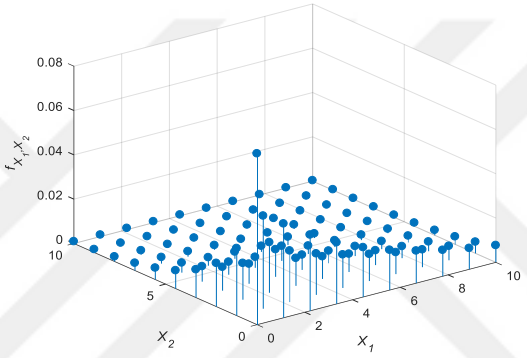
(X_1, X_2) rasgele vektörünün dağılım fonksiyonu,

$$F(x_1, x_2) = \begin{cases} \left(1 - (q_1)^{x_1^\beta}\right)\left(1 - (q_0 q_2)^{x_2^\beta}\right), & 0 \le x_1 \le x_2 \\ \left(1 - (q_0 q_1)^{x_1^\beta}\right)\left(1 - (q_2)^{x_2^\beta}\right), & 0 \le x_2 \le x_1 \\ 1 - (q_0 q_1 q_2)^{x^\beta}, & x_1 = x_2 = x \end{cases} \tag{3.14}$$

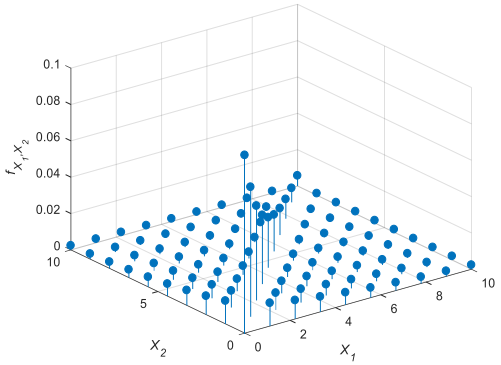
olarak verilir. Çeşitli parametre değerleri için iki değişkenli kesikli Weibull dağılımına ilişkin olasılık fonksiyonu grafikleri, Şekil 3.7-3.9'da verilmiştir.



Şekil 3.7. $\beta = 0.5, p_0 = 0.7, p_1 = 0.3, p_2 = 0.5$ için iki değişkenli kesikli Weibull dağılımına ilişkin of gösterimi



Şekil 3.8. $\beta = 0.7, p_0 = 0.95, p_1 = 0.8, p_2 = 0.6$ için iki değişkenli kesikli Weibull dağılımına ilişkin of gösterimi



Şekil 3.9. $\beta = 0.9, p_0 = 0.9, p_1 = 0.9, p_2 = 0.9$ için iki değişkenli kesikli Weibull dağılımına ilişkin of gösterimi

3.2. Parametre Tahmin Yöntemleri

Bu kesimde, tez çalışması kapsamında ele alınan En Çok Olabilirlik, En Küçük Kareler ve Uyarlanmış En Çok Olabilirlik yöntemleri detaylı biçiminde açıklanmıştır.

3.2.1. En çok olabilirlik yöntemi

En çok olabilirlik (Maximum Likelihood-ML) yöntemi, tahmin edicileri elde etme yöntemleri arasında en popüler olanıdır. Bu yöntem, olabilirlik ilkesine dayanmaktadır ve bir optimizasyon probleminin çözümü ile tahmin edicileri elde etmektedir.

Parametresi tahmin edilmek istenilen kitle $f(\boldsymbol{x};\boldsymbol{\theta}),\ \boldsymbol{\theta}\in\Theta\subseteq\mathbb{R}^d$ dağılımına sahip olsun. Burada $\boldsymbol{\theta}=(\theta_1,\theta_2,...,\theta_d)$ kitle parametre vektörünü, Θ parametre uzayını ve $d\in\mathbb{N}_+$ parametre sayısını göstermektedir. Bu kitleden alınan ve her biri aynı dağılıma sahip $X_1,X_2,...,X_n$ rasgele değişkenler dizisine örneklem adı verilir. $\boldsymbol{X}=(X_1,X_2,...,X_n)$ örnekleminin ortak oyf(of), $f(\boldsymbol{x},\boldsymbol{\theta})=\prod_{i=1}^{n} f(x_i;\boldsymbol{\theta})$ olmak üzere $\boldsymbol{X}=\boldsymbol{x}$ gözlendiğinde $\boldsymbol{\theta}$'nın bir fonksiyonu olan olabilirlik fonksiyonu

$$L(\boldsymbol{\theta}/\boldsymbol{x})=f(\boldsymbol{x};\boldsymbol{\theta}),\ \ \boldsymbol{\theta}\in\Theta\subseteq\mathbb{R}^d \tag{3.15}$$

biçiminde yazılır. Olabilirlik fonksiyonunu Θ parametre uzayı üzerinde maksimum yapan $\hat{\boldsymbol{\theta}}(\boldsymbol{x})$ değerine $\boldsymbol{\theta}$'nın ML tahmini ve $\hat{\boldsymbol{\theta}}(\boldsymbol{X})$ istatistiğine ise $\boldsymbol{\theta}$'nın ML tahmin edicisi adı verilir.

$$\hat{\boldsymbol{\theta}}=\underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmax}}\, L(\boldsymbol{\theta}/\boldsymbol{x}) \tag{3.16}$$

biçiminde ifade edilir. İşlem kolaylığı sağlaması açısından olabilirlik fonksiyonu (L) yerine log-olabilirlik fonksiyonu ($\log L$ veya $\ln L$) kullanılabilmektedir. Logaritmik fonksiyonlar monotonluk özelliğine sahip olduğu için, L fonksiyonunu maksimum yapan $\hat{\theta}$ değeri ile $\ln L$ fonksiyonunu maksimum yapan $\hat{\theta}$ değeri aynıdır. Böylece,

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\boldsymbol{\Theta}}{\operatorname{argmax}} \ln L(\boldsymbol{\theta} / \boldsymbol{x}) \tag{3.17}$$

olarak yazılabilir (Roussas, 1973). Olabilirlik fonksiyonu türevlenebilir olduğunda, ML tahminleri,

$$\frac{\partial L(\boldsymbol{\theta} / \boldsymbol{x})}{\partial \theta_j} = 0, \quad j = 1, 2, ..., m \tag{3.18}$$

veya

$$\frac{\partial \ln L(\boldsymbol{\theta} / \boldsymbol{x})}{\partial \theta_j} = 0, \quad j = 1, 2, ..., m \tag{3.19}$$

denklemlerinin çözümü ile elde edilmektedir. Bu denklemler "olabilirlik denklemleri" olarak da adlandırılmaktadır. Olabilirlik denklemlerinin doğrusal olmayan fonksiyonlar içermesi durumunda, bu denklemlerin analitik çözümünü elde etmek mümkün olmamaktadır (Casella ve Berger, 2002; Akdi, 2005). Böyle tür problemlerin çözümü için çeşitli iteratif ve metasezgisel yöntemlerin kullanılması önerilmektedir.

3.2.2. En küçük kareler yöntemi

En küçük kareler (Least Square-LS) yöntemi gözlemlerin gerçek değerleri ile beklenen değerleri arasındaki farkın en küçük yapılması temeline dayanmaktadır.

$X_{(1)} \le X_{(2)} \le ... \le X_{(n)}$, $F(\boldsymbol{x};\boldsymbol{\theta}), \boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^r$ dağılım fonksiyonuna sahip kitleden alınan örnekleme ait sıra istatistikleri ve F_n^* deneysel (ampirik) dağılım fonksiyonu olmak üzere

$$F_n^*(X_{(i)}) \overset{d}{=} F(X_{(i)}) + \varepsilon_i, \ i = 1, 2, ..., n \tag{3.20}$$

eşitliği yazılabilir. Burada, ε hata terimidir. Deneysel dağılım fonksiyonunun beklenen değeri,

$$E(F_n^*(X_{(i)})) \cong \frac{i}{n+1} \tag{3.21}$$

olarak yazılır. LS tahmin edicisi, hata kareler toplamını minimum yapan değer olup

$$\min_{\boldsymbol{\theta}\in\Theta}\sum_{i=1}^{n}\varepsilon_i^2=\min_{\boldsymbol{\theta}\in\Theta}\sum_{i=1}^{n}\left(\frac{i}{n+1}-F(X_{(i)})\right)^2 \tag{3.22}$$

biçiminde verilen optimizasyon probleminin çözülmesi ile elde edilir (Swain ve ark., 1988).

3.2.3. Uyarlanmış en çok olabilirlik yöntemi

Uyarlanmış en çok olabilirlik (Modified Maximum Likelihood-MML) yöntemi Tiku (1967) tarafından önerilmiştir. MML yöntemi ile elde edilen tahmin ediciler örneklemin birer fonksiyonudur ve açık formda yazılabilmektedir. Bundan dolayı, hesaplama açısından kolaylık sağlamaktadır. Ayrıca, MML tahmin edicileri belirli düzgünlük şartları altında ML tahmin edicilerine yakınsadığından ML yönteminin avantajlarını da korumaktadır. Böylece, ML yöntemi ile tahminlerin analitik çözümlerinin elde edilemediği durumlarda MML yöntemine başvurulabilir.

$y_1, y_2, ...y_n$, $\frac{1}{\sigma} f((y-\mu)/\sigma)$, $\mu\in\mathbb{R}$, $\sigma\in\mathbb{R}^+$ olasılık yoğunluk fonksiyonuna sahip bir dağılımından alınmış rasgele örnekleme ilişkin gözlem değerleri olmak üzere, olabilirlik ve log olabilirlik fonksiyonları sırasıyla

$$L(\mu,\sigma;\mathbf{y})=\prod_{i=1}^{n}\frac{1}{\sigma} f((y_i-\mu)/\sigma) \tag{3.23}$$

ve

$$\ln L(\mu,\sigma;\mathbf{y})=-n\ln\sigma+\sum_{i=1}^{n}\ln f((y_i-\mu)/\sigma) \tag{3.24}$$

biçiminde ifade edilir.

Adım 1. Standartlaştırılmış gözlemler $z_i = \frac{y_i - \mu}{\sigma}$ artan büyüklük sıralamasına göre $z_{(1)} \le z_{(2)} \le ... \le z_{(n)}$ biçiminde sıralanır.

Adım 2. Sıra istatistikleri kullanılarak elde edilen olabilirlik eşitlikleri

$$\frac{\partial \ln L}{\partial \mu} = \sum_{i=1}^{n} g(z_{(i)}) = 0,$$
$$\frac{\partial \ln L}{\partial \sigma} = \sum_{i=1}^{n} z_{(i)} g(z_{(i)}) - \frac{n}{\sigma} = 0. \qquad (3.25)$$

biçiminde yazılabilir. Eşitlik (3.25)'te, $g(z_{(i)}) = \alpha_i + \beta_i z_{(i)}$'dir. $t_{(i)} = E(z_{(i)})$, $z_{(i)}$ sıralı standartlaştırılmış değişkeninin beklenen değeri olmak üzere, $g(z_{(i)})$ doğrusal olmayan fonksiyonunun $t_{(i)}$ civarındaki Taylor serisi açılımının ilk iki terimi kullanılarak lineer bir yaklaşımı,

$$g(z_{(i)}) \cong g(t_{(i)}) + (z_{(i)} - t_{(i)}) \left\{ \frac{\partial}{\partial z} g(z) \right\}_{z=t_{(i)}} = \alpha_i + \beta_i z_{(i)},\ i = 1, 2, ..., n \qquad (3.26)$$

şeklinde yazılabilir. Burada,

$$\beta_i = \frac{\partial}{\partial z} g(z)|_{z=t_{(i)}} \qquad (3.27)$$

$$\alpha_i = g(t_{(i)}) - t_{(i)} \beta_i \qquad (3.28)$$

'dir ve $t_{(i)}$, i. sıra istatistiğinin beklenen değeri olarak tanımlanır. $t_{(i)}$' nin gerçek değerleri mevcut değildir. Fakat, $\int_{-\infty}^{t_{(i)}} f(z)dz = \frac{i}{n+1}$ eşitliğinin çözümünden $t_{(i)}$'nin yaklaşık değerleri $t_{(i)} = E(Z_{(i)}) \cong F^{-1}\left(\frac{i}{n+1}\right)$ olarak elde edilebilir.

Adım 3. Eşitlik (3.26), Eşitlik (3.25)'te yerine konulduğunda uyarlanmış olabilirlik denklemleri,

$$\frac{\partial \ln L}{\partial \mu} = \frac{\partial \ln L^*}{\partial \mu} = \frac{1}{\sigma}\sum_{i=1}^{n} \alpha_i + \beta_i z_{(i)} = 0 \tag{3.29}$$

$$\frac{\partial \ln L}{\partial \sigma} = \frac{\partial \ln L^*}{\partial \sigma} = \frac{1}{\sigma}\sum_{i=1}^{n} z_{(i)}(\alpha_i + \beta_i z_{(i)}) - \frac{n}{\sigma} = 0 \tag{3.30}$$

olarak elde edilir.

Adım 4. Eşitlik (3.30)'da verilen uyarlanmış denklemler, ilgilenilen μ ve σ parametrelerine göre çözülerek MML tahmin edicileri,

$$\hat{\mu}_{MML} = K + D\hat{\sigma}_{MML} \tag{3.31}$$

ve

$$\hat{\sigma}_{MML} = \frac{-B + \sqrt{B^2 + 4nC}}{2n} \tag{3.32}$$

olarak elde edilir. Eşitlik (3.31) ve (3.32)'de;

$$K = \frac{1}{m}\sum_{i=1}^{n} \beta_i y_{[i]}$$

$$m = \sum_{i=1}^{n} \beta_i$$

$$D = \frac{1}{m}\sum_{i=1}^{n} \alpha_i$$

$$B = \sum_{i=1}^{n} \alpha_i \left(y_i - K\right)$$

$$C = \sum_{i=1}^{n} \beta_i \left(y_{(i)} - K \right)^2 = \sum_{i=1}^{n} \beta_i {y_{(i)}}^2 - mK^2$$

olmak üzere, çarpık dağılımlar için $D \neq 0$'dır. Ayrıca, eğer varsa yan (bias)'ı azaltmak için Eşitlik (3.32)'deki $2n$ böleni $2\sqrt{n(n-1)}$ veya bazı diğer uygun sabitlerle yer değiştirebilir (Tiku ve Akkaya, 2004).

3.3. Kısıtlı Doğrusal Olmayan Optimizasyon Algoritmaları

Bu kesimde, tez kapsamında ele alınan kısıtlı doğrusal olmayan optimizasyon yöntemlerinden sıralı karesel programlama ve iç nokta algoritmaları tanıtılmıştır.

3.3.1. Sıralı karesel programlama

Sıralı karesel programlama (Sequential Quadratic Programming-SQP) kısıtlı doğrusal olmayan optimizasyon problemleri için önerilen iteratif bir yöntemdir. SQP yöntemleri bir dizi optimizasyon alt problemlerini çözer. Her bir alt problem kısıtları doğrusallaştırılmış karesel formdaki modeli optimize eder (Yang, 2010; Mathworks, 2020).

SQP yöntemleri doğrusal olmayan programlama problemleri için en gelişmiş yöntemdir. Schittkowski (1986), birçok test problemi için SQP'nin etkinlik, doğruluk ve başarı oranı açısından diğer yöntemlere göre daha üstün performans sergilediğini göstermiştir. Her bir temel iterasyonda, yarı-Newton güncelleme yöntemi kullanılarak Lagrange fonksiyonunun Hessian matrisine bir yaklaşımı yapılır ve karesel programlama alt problemini üretmek amacıyla kullanılır. Karesel programlama probleminin çözümü ise arama yönünü oluşturmak için kullanılır.

SQP yöntemlerinde temel fikir, bir optimizasyon problemi için oluşturulan Lagrange fonksiyonunun karesel yaklaşımına dayalı bir karesel alt problemin formüle edilmesidir.

Bir lineer olmayan optimizasyon problemi,

$$\begin{aligned} &\underset{\mathbf{x}}{M}\text{in } f(\mathbf{x}) \\ &\quad h_i(\mathbf{x}) = 0,\ i = 1,2,...,p \\ &\quad g_j(\mathbf{x}) \le 0,\ j = 1,2,...,q \end{aligned} \tag{3.33}$$

biçiminde gösterilir. Bu problem için oluşturulan Lagrange fonksiyonu Eşitlik (3.35)'te verilmiştir.

$$L(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i h_i(\mathbf{x}) + \sum_{j=1}^{q} \lambda_j g_j(\mathbf{x}) \tag{3.34}$$

Burada, λ'lar Lagrange çarpanlarını göstermektedir. Lineer olmayan kısıtlar Taylor genişletmesi kullanılarak doğrusallaştırılır ve Eşitlik (3.34)'te verilen karesel programlama alt problemi oluşturulur:

$$\begin{aligned} &\underset{d \in \Re^n}{M}\text{in } \frac{1}{2} d^T H_k d + \nabla f(x_k)^T d + f(x_k) \\ &\quad \nabla h_i(x_k)^T d + h_i(x_k) = 0 \quad i = 1,2,...,p \\ &\quad \nabla g_i(x_k)^T d + g_i(x_k) \le 0 \quad j = 1,2,...,q \end{aligned} \tag{3.35}$$

Eşitlik (3.35)'te verilen alt problem, herhangi bir karesel algoritma kullanılarak çözülebilir. Her bir iterasyonda yeni çözüm,

$$x_{k+1} = x_k + \alpha_k d_k \tag{3.36}$$

eşitliği kullanılarak elde edilir. Eşitlik (3.35) ve (3.36)'da, α_k , *k.* iterasyondaki adım uzunluğunu gösterir ve uygun bir yol keşif prosedürü kullanılarak belirlenir. H_k Hessian matrisine pozitif tanımlı bir yaklaşımıdır. H_k herhangi bir yarı Newton yöntemi ile güncellenebilmektedir. Broyon-Fletcher-Goldfarbo-Shanno (BFGS) yöntemi H_k'nın güncellenmesi için en popüler yöntemdir. BFGS yöntemi kullanılarak güncelleme Eşitlik (3.37) yardımıyla gerçekleştirilir.

$$H_{k+1} = H_k + \frac{v_k v_k^T}{v_k^T u_k} - \frac{H_k u_k u_k^T H_k^T}{u_k^T H_k^T u_k} \tag{3.37}$$

Eşitlik (3.36)'da, $u_k = x_{k+1} - x_k$ ve $v_k = \nabla L(x_{k+1}) - \nabla L(x_k)$ olarak verilir.

3.3.2. İç nokta algoritmaları

İç nokta algoritmaları büyük boyutlu doğrusal programlama ve kısıtlı doğrusal olmayan optimizasyon problemlerinin çözümü için önerilmiştir.

Eşitlik (3.33)'te verilen lineer olmayan optimizasyon problemi, artık değişken *s*'ler kullanılarak,

$$\begin{aligned} \underset{\mathbf{x},\mathbf{s}}{M\text{in}}\ f_\mu(\mathbf{x},\mathbf{s}) &= \underset{\mathbf{x},\mathbf{s}}{M\text{in}}\ f(\mathbf{x}) - \mu\sum_{j=1}^{q}\ln(s_j) \\ h_i(\mathbf{x}) &= 0,\ \ i = 1,2,...,p \\ g_j(\mathbf{x}) + s_j &= 0,\ \ j = 1,2,...,q \end{aligned} \tag{3.38}$$

biçiminde yazılır. Burada s_j' ler pozitif değerlidir ve $\mu > 0$ ceza parametresidir. μ'nün değeri 0'a doğru azalarak yaklaştıkça f_μ'nün minimum değeri f'in minimum değerine yaklaşır.

Algoritma, bu yaklaşık problemi çözmek için her bir iterasyonda iki çeşit temel adım kullanır:

***i. Direkt adım*:** Karush-Kuhn-Tucker (KKT) denklemlerini çözmeye çalışır ve Newton adımı olarak da adlandırılır. Direk adımı tanımlamak için aşağıdaki değişkenler kullanılır. f_μ'nin Hessian matrisi H ile gösterilir ve

$$H = \nabla^2 f(x) + \sum_{i=1}^{p}\lambda_i\nabla^2 g_i(x) + \sum_{j=1}^{q}\lambda_j\nabla^2 h_j(x) \tag{3.39}$$

olarak ifade edilir. Eşitlik (3.39)'da:

J_g : g fonksiyonunun jakobiyen vektörü

J_h : h fonksiyonunun jakobiyen vektörü

$S = diag(s)$

λ : g kısıtına ilişkin Lagrange çarpımı vektörü

$\Lambda = diag(\lambda)$

y : h kısıtına ilişkin Lagrange çarpımı vektörü

e : g ile aynı boyutta ve tüm elemanları 1 olan vektör olmak üzere direk adım,

$$\begin{bmatrix} H & 0 & J_h^T & J_g^T \\ 0 & S\Lambda & 0 & -S \\ J_h & 0 & I & 0 \\ J_g & -S & 0 & I \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s \\ -\Delta y \\ -\Delta s \end{bmatrix} = \begin{bmatrix} \Delta f - J_h^T y - J_g^T \lambda \\ s\lambda - \mu e \\ h \\ -\Delta s \end{bmatrix} \tag{3.40}$$

eşitliği ile tanımlanır. Bu eşitlikler doğrusallaştırılmış Lagrange kullanılarak KKT denklemlerinin çözümünden direk gelmektedir (Yang, 2010; Mathworks, 2020).

***ii.Eşlenik gradyan adımı*:** KKT denklemleri yaklaşık olarak çözülerek Lagrange çarpanları elde edilir.

$$\nabla_x L = \nabla^2 f(x) + \sum_{i=1}^{p} \lambda_i \nabla g_i(x) + \sum_{j=1}^{q} y_j \nabla h_j(x) = 0 \tag{3.41}$$

Daha sonra, Eşitlik (3.40)'da verilen lineer kısıtlı optimizasyon problemi çözülerek algoritma bir adım (Δx , Δs) ilerler.

$$\begin{aligned} \underset{\Delta x, \Delta s}{Min} \; & \nabla f^T \Delta x + \frac{1}{2} \Delta x^T \nabla_{xx}^2 L \Delta x + \mu e^T S^{-1} \Delta s + \frac{1}{2} \Delta s^T S^{-1} \Lambda \Delta s \\ & g(x) + J_g \Delta x + \Delta s = 0 \\ & h(x) + J_h \Delta x = 0 \end{aligned} \tag{3.42}$$

Algoritma, ölçeklendirilmiş yarıçapa sahip bir bölge içerisinde doğrusal kısıtların bir normunu minimize etmek için çalışır. Buradan elde edilen sonuçlar yardımıyla Eşitlik (3.33) çözülür (Byrd ve ark., 2000).

3.4. TOPSIS Yöntemi

En yaygın kullanılan Çok Kriterli Karar Verme (MCDM) yöntemlerinden biri olan İdeal çözüme yakınlığa göre tercih sıralaması yöntemi TOPSIS yöntemi, Hwang ve Yoon (1981) tarafından önerilmiştir. Bu yöntem, pozitif ideal çözüme en yakın ve negatif ideal çözüme en uzak alternatifin seçimine dayanmaktadır. *m* alternatif ($A_1, A_2, ..., A_m$) ve *n* kriter ($C_1, C_2, ..., C_n$) içeren bir MCDM probleminin çözümü için TOPSIS yönteminin adımları aşağıdaki gibidir (Opricovic ve Tzeng, 2004; Şahin ve Pehlivan, 2017).

Adım 1. Karar matrisi $D = \left[x_{ij} \right]_{m \times n}$ ve ağırlık vektörü $W = \left[w_j \right]_{1 \times n}$,

$$D = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & ... & C_n \end{array} \\ \begin{bmatrix} x_{11} & x_{12} & ... & x_{1n} \\ x_{21} & x_{22} & ... & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & ... & x_{mn} \end{bmatrix} \end{array} \tag{3.43}$$

$$W = \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & ... & C_n \end{array} \\ \begin{bmatrix} w_1 & w_2 & ... & w_n \end{bmatrix} \end{array} \tag{3.44}$$

biçiminde oluşturulur. Burada, x_{ij} i. alternatifin j. kriter altında değerlendirme skorunu (puanını) ve w_j, j. kriterin ağırlık değerini göstermektedir.

Adım 2. Normalize edilmiş karar matrisi $D' = \left[x'_{ij} \right]_{m \times n}$, Eşitlik (3.45) kullanılarak oluşturulur.

$$x'_{ij} = \frac{x_{ij}}{\sum_{i=1}^{m} \left(x_{ij} \right)^2} \tag{3.45}$$

Adım 3. Ağırlıklandırılmış karar matrisi $D_w = [v_{ij}]_{m \times n}$, Eşitlik (3.46) kullanılarak oluşturulur.

$$v_{ij} = w_j \times x'_{ij} \tag{3.46}$$

Adım 4. Pozitif ideal çözüm $A^+ = \left(v_1^+, v_2^+, ..., v_n^+\right)$ ve negatif ideal çözüm $A^- = \left(v_1^-, v_2^-, ..., v_n^-\right)$ belirlenir. Burada;

$$v_j^+ = \begin{Bmatrix} \max\limits_{1\le i\le m}\{v_{ij}\} \mid c_j \in C_1 \\ \min\limits_{1\le i\le m}\{v_{ij}\} \mid c_j \in C_2 \end{Bmatrix} \tag{3.47}$$

$$v_j^- = \begin{Bmatrix} \min\limits_{1\le i\le m}\{v_{ij}\} \mid c_j \in C_1 \\ \max\limits_{1\le i\le m}\{v_{ij}\} \mid c_j \in C_2 \end{Bmatrix} \tag{3.48}$$

olarak hesaplanır. Burada, C_1 fayda kriterini ve C_2 maliyet kriterini göstermektedir.

Adım 5. Her bir alternatif ile pozitif ideal çözüm arasındaki uzaklık (d_i^+) ve negatif ideal çözüm arasındaki uzaklık (d_i^-) sırasıyla Eşitlik (3.49) ve Eşitlik (3.50)'dan hesaplanır.

$$d_i^+ = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^+)^2} \tag{3.49}$$

$$d_i^- = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^-)^2} \tag{3.50}$$

Adım 6. Kısmi yakınlık (ideal çözüme göreli yakınlık) derecesi CC_i değerleri hesaplanır.

$$CC_i = \frac{d_i^-}{d_i^+ + d_i^-} \tag{3.51}$$

Adım 7. CC_i, $i = 1, 2, ..., m$ $\left(0 \le CC_i \le 1\right)$ değerleri azalan biçimde sıralanır. En büyük CC_i değerine sahip alternatif, en iyi alternatif olarak belirlenir (Opricovic ve Tzeng, 2004; Şahin ve Pehlivan, 2017).

3.5. Taguchi Yöntemi

Kontrol altındaki çeşitli faktörlerin, deney birimlerinin bilinmeyen karakteristik özellikleri üzerindeki etkisini test etmek amacıyla uygulanan işleme "deney" adı verilir. Deney birimlerinin maruz kalacağı kontrol altındaki faktörlerin düzenlenmesi işlemine ise "deney tasarımı" denir. Deneyde birden fazla faktör olması durumunda, faktör düzeylerinin kombinasyonları "deneme" olarak adlandırılır. Faktörler iki ya da daha fazla düzeye sahip olabilir. Deney birimleri hakkında bilgi edinilmek istenen karakteristik özelliklere "yanıt" veya "bağımlı değişken" adı verilir. Bu yanıt değişkenin aldığı değerlere Varyans Analizi (ANOVA) uygulanarak araştırma konusu olan faktörlerin istatistiksel olarak anlamlı olup olmadığı test edilir (Şenoğlu ve Acıtaş, 2011).

İki ya da daha fazla faktörün ana etkilerini ve etkileşim etkilerini aynı anda araştırmak için kullanılan tasarımlara "faktöriyel tasarımlar" adı verilir. Literatürde 2^k, 3^k,… gibi çeşitli faktöriyel tasarımlar bulunmaktadır. 2^k faktöriyel tasarımında, 2 düzey sayısını, *k* ise faktör sayısını gösterir. Örneğin, her biri iki düzeye sahip A, B, C gibi üç faktör olduğunda 2^k faktöriyel tasarım, 2^3 faktöriyel tasarım olarak adlandırılır ve $2^3 = 8$ tane denemeye ihtiyaç duyulur. Faktör sayısı veya düzey sayısı arttıkça deneme sayısı da artacağından hesaplama zorluğu ortaya çıkmaktadır. Bu zorluğun üstesinden gelebilmek için Taguchi (1987) tarafından daha az sayıda deney kullanarak deney tasarımını gerçekleştirmek amacıyla Taguchi yöntemi önerilmiştir (Şenoğlu ve Acıtaş, 2011; Zoraghi ve ark., 2012).

Taguchi yönteminde, çok sayıda karar değişkenini, karar verme sürecinde yeterli bilgiyi sağlayacak daha az sayıda deney ile çalışabilmek için ortogonal diziler kullanılmaktadır. Taguchi yönteminde, $Sinyal(S)/Gürültü(N)$ oranı adında bir performans ölçeği tanımlanmıştır. Bu oran, parametrelerin kontrol edilebilen değerlerinin kontrol edilemeyen değerlerine oranıdır. Taguchi yönteminde, amaç fonksiyonu "en küçük, en iyidir.", "en büyük, en iyidir." ve "nominal değer, beklenendir." olmak üzere 3 gruba ayrılır. Bu üç grubun her biri için (S/N) oranı hesaplanırken

$$S/N_k = -10 \times \log\left(\frac{1}{n}\sum_{i=1}^{n} y_i^2\right) \tag{3.52}$$

$$S/N_b = -10 \times \log\left(\frac{1}{n}\sum_{i=1}^{n}\frac{1}{y_i^2}\right) \tag{3.53}$$

$$S/N_t = -10 \times \log\left(\frac{1}{n}\sum_{i=1}^{n}\frac{y}{s_y^2}\right) \tag{3.54}$$

eşitlikleri kullanılır. Eşitlik (3.52)-(3.54)'te;

n: gözlem sayısını,

y_i: yanıt değişkeni değerlerini,

y: yanıt değişkeni değerlerinin ortalamasını,

s_y^2: y 'nin varyansını göstermektedir (Nalbant ve ark., 2007)

Amaç fonksiyonu türüne bakılmaksızın en büyük (S/N) oranı daha iyi performans özelliğine karşılık gelmektedir. Bundan dolayı bir faktörün en büyük (S/N) değerine sahip olan seviyesi en iyi seviyesi olarak belirlenir. Ayrıca hangi faktörlerin istatistiksel olarak anlamlı olduğunu görmek için ANOVA gerçekleştirilir. (S/N) oranları ve ANOVA sonuçları ile faktörlerin optimal kombinasyonları belirlenir.

Taguchi yönteminin çoklu performans özelliklerine sahip bir işlemi optimize etmek için parametre tasarımındaki kullanımı aşağıdaki adımları içermektedir (Nian ve ark., 1999; Nalbant ve ark., 2007):

Adım 1. Performans özellikleri tanımlanır ve değerlendirilecek faktörler seçilir

Adım 2. Faktörler için seviye sayıları belirlenir

Adım 3. Uygun ortogonal dizi seçilir ve faktörler ortogonal diziye atanır.

Adım 4. Ortogonal dizilimin düzeni temel alınarak deneyler gerçekleştirilir.

Adım 5. (S/N) oranları hesaplanır.

Adım 6. (S/N) oranları ve ANOVA kullanarak deney sonuçları analiz edilir.

Adım 7. Faktörlerin optimal seviyeleri belirlenir.

Uygun ortogonal dizinin seçimi için toplam serbestlik derecesinin hesaplanması gerekir. Örneğin, 4 faktörlü ve her bir faktörün 3 seviyesinin olduğu bir çalışmada her faktör için serbestlik derecesi $3-1=2$ olup toplam serbestlik derecesi $4\times 2=8$'dir. Başka bir ifadeyle seçilen ortogonal dizinin serbestlik derecesi en az 8 olmalıdır. Böyle bir çalışma için, L_9 ortogonal dizisinin seçilmesi önerilir ve bu dizi 26 serbestlik

derecesine sahiptir. L_9 ortogonal dizisi kullanılarak tüm parametre uzayını çalışmak için $9\times3=27$ deneye ihtiyaç duyulur.

Bu tez çalışmasında, algoritmaların parametrelerinin optimal seviyelerinin belirlenmesinde Taguchi yöntemi kullanılmıştır.

3.6. Kullanılan İstatistiksel Kriterler

Bu kesimde, çalışmanın simülasyon ve gerçek veri kümesi uygulamaları kısımlarında kullanılan istatistiksel kriterler açıklanmıştır.

***Ortalama (Ort.)*:** Monte-Carlo simülasyonu ile elde edilen parametre tahmin değerlerinin ortalamasını gösterir ve

$$Ort.(\hat{\theta}) = \bar{\hat{\theta}} = \frac{1}{k}\sum_{i=1}^{k}\hat{\theta}_i \tag{3.55}$$

biçiminde tanımlanır. Eşitlik (3.55)'te $\hat{\theta}_i$, *i.* simülasyon çalışmasında elde edilen parametre tahmin değerini ve k ise toplam simülasyon sayısını göstermektedir.

***Beklenen Değer*:** Tahmin edicinin yoğunlaştığı konum hakkında bilgi verir ve $E(\hat{\theta})$ ile gösterilir. Tez çalışmasında, $E(\hat{\theta})$'nın yerine yansız bir tahmin edicisi olan $\bar{\hat{\theta}}$ kullanılmıştır.

***Yan (Bias)*:** Tahmin edicinin beklenen değeri ile gerçek değeri arasındaki farkı gösterir ve

$$Bias(\hat{\theta}) = E(\hat{\theta}) - \hat{\theta} = \bar{\hat{\theta}} - \hat{\theta} \tag{3.56}$$

biçiminde tanımlanır. $Bias(\hat{\theta}) = 0$ ise, tahmin edici "yansız tahmin edici" olarak adlandırılır.

***Örneklem Varyansı (Var)*:** Tahmin edicinin olasılık dağılımının ortalama etrafındaki yoğunluğu hakkında bilgi verir ve

$$Var(\hat{\theta}) = \frac{1}{k-1}\sum_{i=1}^{k}(\hat{\theta}_i - \bar{\hat{\theta}})^2 \tag{3.57}$$

biçiminde tanımlanır. Varyans değeri, yansız tahmin edicilerin etkinliklerini karşılaştırmak için kullanılabilir. Yansız tahmin ediciler içerisinden küçük varyans değerine sahip tahmin edicinin varyans kriterine göre daha etkin olduğu söylenebilir.

***Hata kareler ortalaması (MSE)*:** Yanlı tahmin edicilerin etkinliklerini karşılaştırmak için kullanılan bir kriterdir ve

$$MSE(\hat{\theta}) = Var(\hat{\theta}) + (Bias(\hat{\theta}))^2 \tag{3.58}$$

biçiminde tanımlanır. Küçük MSE değerine sahip bir tahmin edici MSE ölçütüne göre daha etkindir. Ayrıca, yansız tahmin ediciler için Varyans ve MSE değerleri birbirine eşittir.

***Kusur kriteri (Def)*:** Parametre kümesinin tahmini (örneğin $\hat{\boldsymbol{\theta}} = (\hat{\gamma}, \hat{\eta}, \hat{\beta})$) için kullanılan çeşitli yöntemlerin etkinliğini test etmek için kullanılan önemli bir ölçektir. (Yang, 2010). Birden fazla parametrenin aynı anda tahmin edilecek olması durumunda tahmin edicilerin etkinliklerini karşılaştırmak amacıyla kullanılan bir kriterdir. $\hat{\boldsymbol{\theta}} = (\hat{\gamma}, \hat{\eta}, \hat{\beta})$ olmak üzere Def kriteri,

$$Def(\hat{\gamma}, \hat{\eta}, \hat{\beta}) = MSE(\hat{\gamma}) + MSE(\hat{\eta}) + MSE(\hat{\beta}) \tag{3.59}$$

biçiminde tanımlanır. Küçük MSE değerine sahip tahmin ediciler diğer tahmin edicilerden daha etkin olduğundan, küçük Def değerine sahip tahmin edicilerin de diğer tahmin edicilerden daha etkin olduğu söylenebilir.

***Akaike ve Bayes Bilgi Kriterleri (AIC ve BIC)*:** Bir veri kümesi için önerilen yöntemlerin uygunluğunu karşılaştırmak amacıyla kullanılan AIC (Akaike ve ark., 1973) ve BIC (Stone, 1979) değerleri,

$$AIC = -2\ln L + 2k \tag{3.60}$$

$$BIC = -2\ln L + 2\left(\ln n\right)k \tag{3.61}$$

eşitlikleri ile hesaplanır. Burada, n gözlem sayısını ve k parametre sayısını göstermektedir. Küçük *AIC* ve *BIC* değerlerine sahip yöntemler tercih edilir.

4. METASEZGİSEL YÖNTEMLER

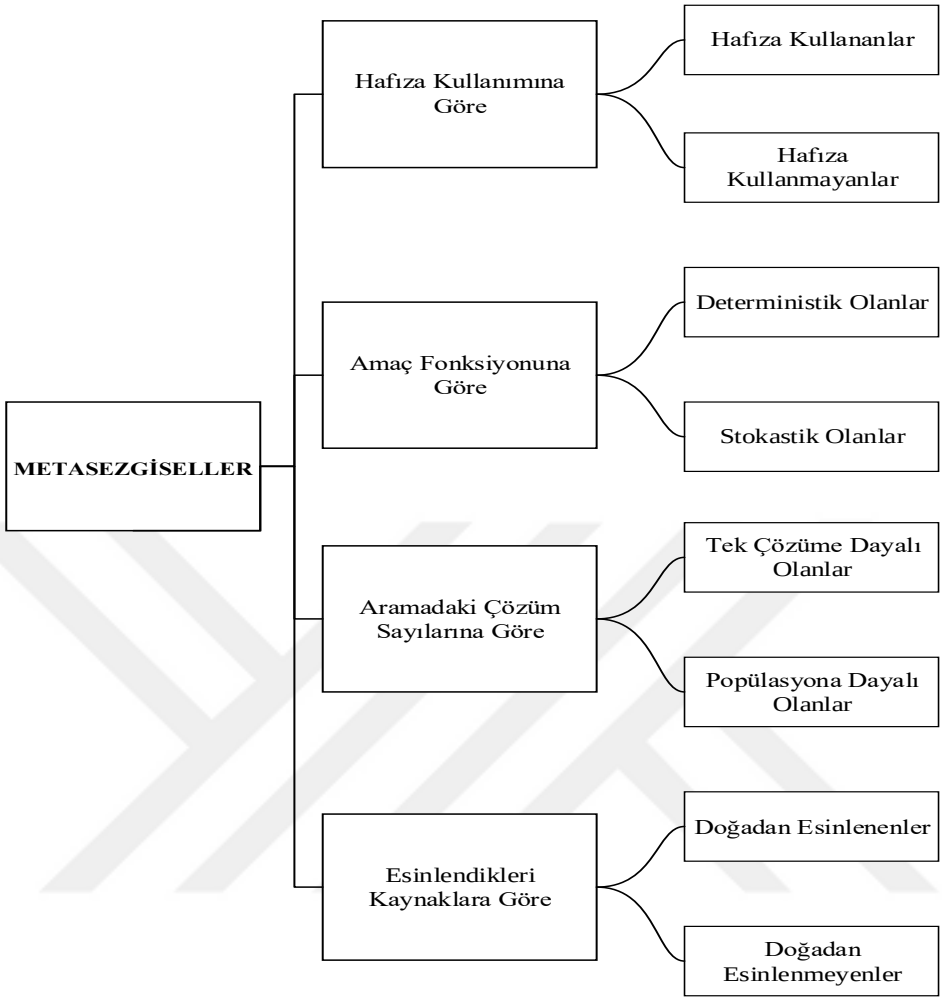
"Sezgisel" kelimesi, bir problemi çözmek için yeni stratejiler veya kurallar keşfetme ya da bulma sanatı anlamına, "Meta" öneki ise "ileri, üst seviye metodoloji" anlamına gelmektedir. Metasezgiseller, optimizasyon problemlerini çözmek için sezgiselleri tasarlamada rehber stratejiler olarak kullanılabilecek üst düzey genel yöntemler olarak tanımlanmaktadır. Literatürde, sezgisel ve metasezgiseller ile ilgili kabul görmüş genel bir tanım bulunmamaktadır. Birçok araştırmacı, sezgisel ve metasezgiseli birbirinin alternatifli olarak kullanmaktadır. Fakat son yıllarda, rasgelelik ve yerel arama içeren tüm stokastik algoritmalar metasezgisel olarak adlandırılmaktır. Rasgelelik, yerel aramadan global aramaya gitmek için iyi bir yol sağlamaktadır. Bundan dolayı, tüm metasezgisel algoritmalar global optimizasyon için uygun olma eğilimindedir (Yang, 2014).

Kesin yöntemlerden farklı olarak, metasezgisel yöntemler optimal bir çözümü garanti etmemektedir. Fakat, makul bir hesaplama süresi sağlayarak kabul edilebilir çözümler üretebilmekte ve böylece büyük boyutlu problemlerin çözülmesine imkân sağlamaktadırlar. Optimizasyon problemlerinin çözümünde sezgisel kavramı, ilk olarak Pólya (1945) tarafından tanıtılmıştır (Talbi, 2009; Yang, 2010).

Metasezgisellerin popülerliği son yirmi yılda oldukça artmıştır. Büyük ve karmaşık problemlerin çözümleri için metasezgisel yöntemlerin bazı uygulamalarda kullanımı, metasezgisel yöntemlerin yeterlilik ve etkinliğini göstermiştir. Metasezgisel yöntemler birçok alanda uygulanmaktadır. Bunlardan bazıları:

- Mühendislik tasarımı, topoloji optimizasyonu ve elektronikteki yapısal optimizasyon, hava bilimi, akışkan dinamikleri, telekominasyon, otomotiv ve robotlar,
- Yapay zeka ile öğrenme ve biyoenformatik, bilgisayarlı biyolojide veri madenciliği ve finans,
- Sistem modelleme, kimya, fizik ve biyolojide kontrol, işaret ve görüntü işlemede simülasyon ve tanımlamada,
- Rotalama problemlerinde planlama, robot tasarımı, problemlerin planlanması ve üretimi, lojistik ve ulaşım, tedarik zinciri yönetimi, çevre vb.

Metasezgiseller için Şekil 4.1'de verilen bazı sınıflandırma kriterleri kullanılmaktadır (Talbi, 2009):



Şekil 4.1. Metasezgisellerin sınıflandırılması

- ***Hafıza Kullanan ve Kullanmayan Metasezgiseller*:** Arama sırasında hiçbir bilgi kullanmayan metasezgisel algoritmalar hafızasızdır, örneğin, Tavlama Benzetimi (SA) yöntemi. Bazı metasezgiseller ise arama sırasında bazı bilgileri içeren bir hafıza kullanırlar. Tabu arama (TS) yöntemindeki kısa ve uzun süreli hafızalar buna örnek olarak verilebilir.
- ***Deterministik ve Stokastik Metasezgiseller*:** Deterministik metasezgiseller, optimizasyon problemini deterministik kararlar alarak çözer. Örneğin, İteratif Yerel Arama (ILS) ve Tabu Arama yöntemleri. Stokastik metasezgiseller ise, arama sırasında bazı rasgele kuralları uygular, örneğin, Tavlama Benzetimi (SA), Yapay Arı Kolonisi (ABC), Diferansiyel Gelişim (DE) ve Genetik Algoritma (GA) yöntemleri. Deterministik algoritmalarda, aynı başlangıç

çözümünü kullanmak aynı sonuçların elde edilmesine yol açabilirken, stokastik algoritmalarda aynı başlangıç çözümünden farklı sonuçlar elde edilmektedir.

- ***Tek Çözüme Dayalı ve Popülasyona Dayalı Metasezgiseller*:** Tek çözüme dayalı algoritmalar, örneğin SA yöntemi arama sırasında tek bir çözümü işler ve değerlendirirken, popülasyona dayalı algoritmalar örneğin, PSO, ABC, DE ve GA yöntemleri çözümlerin tüm topluluğunu kullanır ve geliştirir. Tek çözüme dayalı metasezgiseller işletmeye yöneliktir, yani yerel bölgelerde aramada yoğunlaşma gücüne sahiptir. Popülasyona dayalı metasezgiseller ise, keşfetmeye yöneliktir, yani tüm arama uzayında daha iyi bir çeşitlendirmeye izin verirler.
- ***Doğadan Esinlenen ve Esinlenmeyen Metasezgiseller*:** Bazı metasezgiseller doğal süreçlerden esinlenmektedir. Örneğin, DE ve GA yöntemleri biyolojiden, ACO, ABC ve PSO yöntemleri farklı türlerdeki sürü zekasından, SA yöntemi ise fizikten esinlenmektedir.

Genellikle, metasezgisel yöntemler rasgele bir başlangıç çözümü veya popülasyon üreterek aramaya başlar ve iterasyon bitene kadar bu çözümü/popülasyonu evrimleştirir. D boyutlu bir optimizasyon problemi için $\boldsymbol{x}_i^g = [x_{1,i}^g, x_{2,i}^g, ..., x_{D,i}^g]$ g. iterasyonda popülasyonun i. vektörünü göstermektedir. i. vektörün her bir elemanı için başlangıç popülasyonu, arama uzayı olarak bilinen alt limit ($x_{j,\min}$) ve üst limit ($x_{j,\max}$) değerleri kullanılarak Eşitlik (4.1)'de verildiği gibi rasgele üretilir.

$$x_{j,i}^0 = x_{j,\min} + rand_{ij}[0,1]\left\{x_{j,\max} - x_{j,\min}\right\} \tag{4.1}$$

Burada; $rand_{ij}[0,1]$, $0,1$ aralığında düzgün dağılımdan üretilmiş sayıları temsil eder. Algoritma tek çözüme dayalı ise, $i=1$ alınarak başlangıç çözüm üretilir (Talbi, 2009).

Bu bölümde, ele alınan Tavlama Benzetimi, Genetik Algoritma, Parçacık Sürüsü Optimizasyonu, Diferansiyel Gelişim ve Yapay Arı Kolonisi algoritmaları detaylı olarak açıklanmıştır.

4.1. Tavlama Benzetimi Algoritması

Tavlama Benzetimi (Simulated Annealing-SA) algoritması, bir maddenin önce ısıtılması daha sonra kristal yapının elde edilmesi için yavaşça soğutulmasına ihtiyaç duyan tavlama sürecinin ilkelerine dayanmaktadır. Tavlama süreci, dikkatli bir sıcaklık kontrolü ve soğutma oranını kapsamaktadır. SA'nın optimizasyon problemleri için kullanılması, Metropolis ve ark. (1953)'nın istatistiksel mekanizma alanındaki çalışmalarından ilham alınarak ilk olarak Kirkpatrick ve ark. (1983) tarafından önerilmiştir (Abbasi ve ark., 2006; Talbi, 2009; Yang, 2010). Optimizasyon ile SA algoritması arasındaki ilişkide, maddenin durumları problemin muhtemel çözümlerine, durumların enerjileri ise bu çözümlerin amaç fonksiyonu değerlerine karşılık gelmektedir.

SA'nın en önemli özelliklerinden birisi, kötü çözümlerin kabul edilmesine imkân sağlayan stokastik bir algoritma olmasıdır. Bundaki amaç, yerel optimalden kaçınmak için yakınsamayı geciktirmektir. SA algoritmasında arama, Markov zincirleri kullanılarak modellenir ve bundan dolayı maddenin gelecekteki durumu yalnızca mevcut durumuna bağlıdır.

SA, tek çözüme dayalı bir metasezgisel algoritmadır ve her bir iterasyonda rasgele bir komşu çözüm üretir. Eğer, komşu çözüm amaç fonksiyonu değerini geliştiriyorsa kabul edilir, aksi durumda verilen bir olasılık değerine göre bu çözüm kabul edilir. Bu olasılık değeri, mevcut sıcaklığa ve amaç fonksiyonundaki kötüleşme miktarına bağlıdır ve

$$P(\Delta E, T) = \exp\left(\frac{f(x') - f(x)}{k_B T} \right) \tag{4.2}$$

eşitliğinden hesaplanır. Eşitlik (4.2)'de; x mevcut çözümü, x' komşu çözümü, $f(x)$ amaç fonksiyonu değerini, T mevcut sıcaklığı ve k_B ise Boltzmann sabitini göstermektedir. Boltzmann sabiti genellikle $k_B = 1$ olarak alınır (Abbasi ve ark., 2006; Talbi, 2009; Yang, 2010).

SA algoritmasının performansı yüksek oranda algoritmanın parametrelerine: başlangıç sıcaklığı, soğutma çizelgesi (her sıcaklıkta üretilecek komşuların sayısı ve

soğutma oranı) ve durdurma kriterine bağlıdır (Abbasi ve ark., 2006; Talbi, 2009; Yang, 2010).

***Başlangıç sıcaklığı*:** Başlangıç sıcaklık, başlangıçta kötü çözümleri kabul edecek ve elde edilen son çözümün başlangıç çözümden bağımsız olmasını sağlayacak kadar yüksek olmalıdır. Algoritmanın ilk adımlarında mevcut çözümü iyileştirmeyen çözümlerin en az %80'ini kabul edecek kadar yüksek bir başlangıç sıcaklığı belirlenmelidir (Kirkpatrick ve ark., 1983).

***Soğutma Çizelgesi*:** Her bir sıcaklıkta üretilecek komşuların (aranacak çözümlerin) sayısı ve soğutma oranını içeren soğutma çizelgesinde bu iki faktör arasında bir denge olmalıdır. Bu dengeyi sağlamak için, az sayıda sıcaklıkta çok sayıda çözüm noktası aranmalı veya çok sayıda sıcaklıkta az sayıda çözüm noktası aranmalıdır. Soğutma oranı, algoritmanın sonlarına doğru kötü çözümleri kabul etme olasılığı sıfıra yaklaşacak şekilde belirlenmelidir (Talbi, 2009; Karaboğa, 2014). Sıcaklık düşürme işlemleri farklı yöntemler yardımıyla yapılabilmektedir. En yaygın kullanılan lineer, geometrik ve logaritmik soğutma için kullanılan eşitlikler aşağıda verilmiştir.

Lineer: $T_i = T_0 - i \times \beta$ (4.3)

Geometrik: $T_i = \alpha T_{i-1}$ (4.4)

Logaritmik: $T_i = \frac{T_0}{\ln i}$ (4.5)

Eşitlik (4.3)-(4.5)'de T_i: i. iterasyondaki sıcaklığı, T_0: başlangıç sıcaklığını, β: belirli sabit bir değeri ve α ise $[0,1]$ aralığında bir değeri göstermektedir.

***Durdurma Kriteri*:** SA algoritmasının hem global çözüme yaklaşmasını sağlayacak hem de çözüm zamanını çok uzatmayacak biçimde belirlenmelidir. Toplam iterasyon sayısı, her bir sıcaklıkta yapılacak hareketlerin sayısı veya belirli bir son sıcaklık değeri durdurma kriteri olarak belirlenebilir (Johnson ve ark., 1989; Saab ve Rao, 1991).

D boyutlu bir optimizasyon problemi için, $\boldsymbol{x} = [x_1, x_2, ..., x_D]$ parametre vektörünü ve $f(\boldsymbol{x})$ amaç fonksiyonunu göstermek üzere SA algoritmasının işleyiş adımları aşağıda verilmiştir.

***Tavlama Benzetimi algoritmasının uygulama adımları*:**

Adım 1. SA'nın parametreleri; başlangıç sıcaklığı (T_0), her bir sıcaklıkta üretilecek komşuların sayısı (*nMove*), soğutma oranı (τ) ve alt iterasyon sayısı (s) belirlenir.

Adım 2. Eşitlik (4.1) kullanılarak, önceden belirlenmiş arama uzayında parametreler için rasgele başlangıç çözüm $\boldsymbol{x}^0 = [x_1^0, x_2^0, ..., x_D^0]$ oluşturulur. Başlangıç çözüm için amaç fonksiyonu değeri $f(\boldsymbol{x}^0)$ hesaplanır.

Adım 3. Önceden belirlenmiş bir strateji kullanılarak komşu çözüm $\boldsymbol{x}'$ üretilir ve bu komşu çözümün amaç fonksiyonu değeri $f(\boldsymbol{x}')$ hesaplanır.

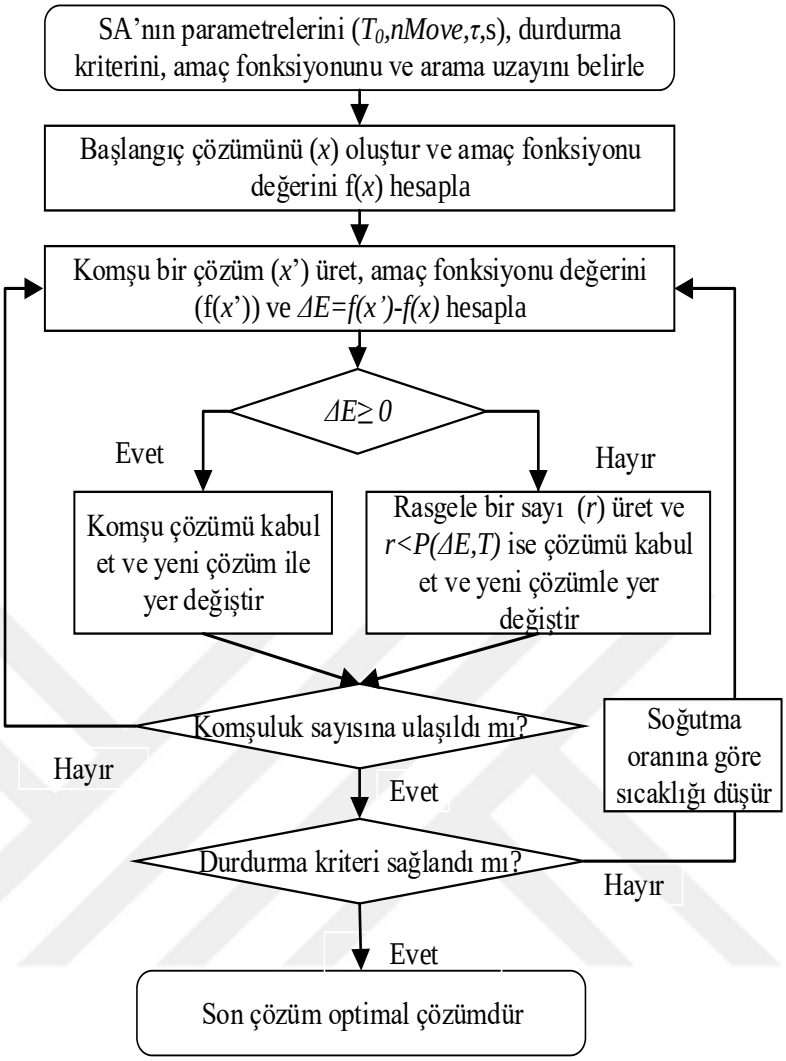
Adım 4. Komşu çözüm $\boldsymbol{x}'$ amaç fonksiyonu değerini geliştiriyor ise, bu çözüm kabul edilir ve mevcut çözüm ile yer değiştirir. Aksi durumda, rasgele bir sayı $r \in (0,1)$ üretilir ve $r < P(\Delta E, T)$ ise komşu çözüm $\boldsymbol{x}'$ kabul edilir.

Adım 5. Her bir sıcaklık için, önceden belirlenen komşuluk sayısına ulaşıncaya dek Adım 3 ve Adım 4 tekrarlanır.

Adım 6. Soğutma oranına göre sıcaklık düşürülür.

Adım 7. Durdurma kriterine ulaşana kadar çözüm evrimleştirilir. Elde edilen son çözüm, problemin çözümü olarak alınır.

Bir maksimizasyon problemi için Tavlama Benzetimi algoritmasının akış diyagramı Şekil 4.2'de verilmiştir.



Şekil 4.2. Benzetimli Tavlama (SA) algoritması akış diyagramı

4.2. Genetik Algoritma

Genetik algoritma (Genetic Algorithm-GA), optimizasyon ve arama problemleri için yaklaşık çözümleri bulmada en yaygın olarak kullanılan arama tekniklerinden biridir. GA, Darwin'in evrimleşme sürecini anlamak için Holland (1975) tarafından önerilmiştir. Daha sonra, Goldberg ve Holland (1988) tarafından optimizasyon problemleri ve makine öğrenmesi için kullanılmıştır. GA, herhangi bir problemin çözümünde biyolojik sistemlerin gelişim sürecini taklit eden bir stokastik arama algoritmasıdır. GA, tıp, mühendislik uygulamaları, gezgin satıcı problemleri, tasarım, planlama, yapay zeka, uzman sitemleri vb. bir çok alanda kullanılmaktadır (Yang, 2010).

En popüler evrimleşme algoritmalarından biri olan GA, popülasyona dayalı bir yöntemdir. GA, genellikle rasgele bir başlangıç popülasyonu üreterek başlar ve iterasyon süreci boyunca popülasyonun yeni çözümlerini oluşturmak için evrimleşmesini gerçekleştirir. Her bir iterasyon bir jenerasyon olarak adlandırılır ve yer değiştirme/seçim ve çaprazlama ve mutasyon işlemlerini içeren evrimleşme (yeniden üretim) operatörlerinden oluşur (Talbi, 2009; Yang, 2014; Asim ve ark., 2018; Yalçınkaya ve ark., 2018).

***Yer Değiştirme/Seçim Operatörü*:** Mevcut popülasyondan bir sonraki popülasyona taşınacak olan çözümlerin seçilmesi işlemidir. Taşınan çözümler, mevcut popülasyonda genetik olarak uygunluğu yüksek olan çözümlerdir. Bu işlem ile özel genetik yapıların bir sonraki popülasyona taşınması sağlanır.

***Evrimleşme (Yeniden Üretim) Operatörü*:** Evrimleşme, çaprazlama ve mutasyon operatörleri kullanılarak gerçekleştirilir. Bu operatörler, genetik bilgileri kullanarak yeni popülasyonun yeni çözümlerini elde etmektedir.

- ***Çaprazlama Operatörü*:** Farklı çözümler arasında bilgi değişimini sağlayarak yeni çözümleri elde eder. Çaprazlama, rasgele seçilen iki çözümün belirli bölümlerini değiştirerek arama uzayında yeni noktaları verecek farklı çözümlerin üretilmesini sağlar. Bu işlem, çaprazlama olasılığı (p_c) kullanılarak gerçekleştirilir. Bu olasılık, çaprazlama operatörünün kullanım sıklığını kontrol eder. Yüksek p_c değeri popülasyonun çok hızlı değişmesine, düşük p_c değeri ise aramanın çok yavaş gerçekleşmesine yol açar.

- ***Mutasyon Operatörü*:** Mevcut çözümlerin bir kısmında rasgele değişimi sağlayarak çözüm uzayında yeni noktaların elde edilmesini sağlar. Bu işlemde, yapılacak çözüm sayısını belirlemek için mutasyon oranı (m) ve mutasyon işlemini gerçekleştirmek için mutasyon olasılığı (p_m) kullanılır. Çözümler p_m olasılığına eşit bir olasılık ile rasgele değişime uğrar. p_m 'nin çok yüksek olması rasgele aramaya yol açar ve bundan dolayı p_m 'nin düşük olması tercih edilir.

D boyutlu bir optimizasyon problemi için $\boldsymbol{x} = [x_1, x_2, ..., x_D]$ parametre vektörünü ve $f(\boldsymbol{x})$ amaç fonksiyonunu göstermek üzere GA'nın işleyiş adımları aşağıda verilmiştir.

***Genetik Algoritma uygulama adımları*:**

Adım 1. GA'nın parametreleri; popülasyon boyutu (NP), mutasyon olasılığı (p_m), çaprazlama olasılığı (p_c) ve mutasyon oranı (m) belirlenir.

Adım 2. Eşitlik (4.1) kullanılarak, önceden belirlenen arama uzayında parametreler için rasgele başlangıç popülasyonu $\boldsymbol{x}_i^0 = [x_{1,i}^0, x_{2,i}^0, ..., x_{D,i}^0], i = 1, 2, ...NP$ oluşturulur. Başlangıç popülasyonu için amaç fonksiyonu değerleri $f(\boldsymbol{x}_i^0)$ hesaplanır.

Adım 3. Hesaplanan amaç fonksiyonu değerlerine göre kötü çözümler popülasyondan atılır ve bunların yerine yeni çözümler üretilir. En iyi amaç fonksiyonu değerine sahip çözümler elit kabul edilir ve bunlar hiçbir değişikliğe uğramadan gelecek popülasyona aktarılır.

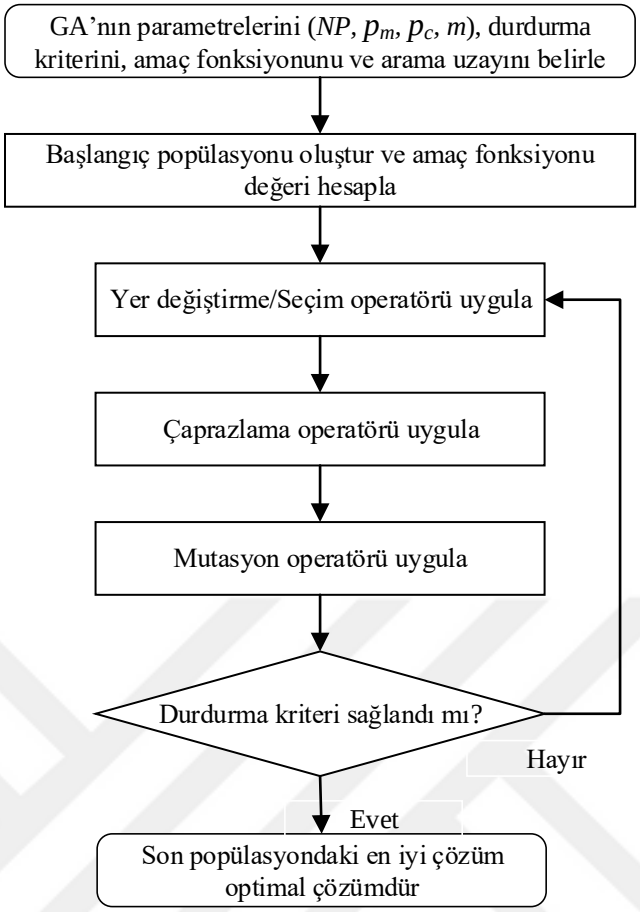
Adım 4. Mevcut popülasyondan çaprazlama olasılığına bağlı olarak bir çift çözüm seçilir. Çaprazlama işlemi uygulanarak iki tane yavru çözüm oluşturulur. Bu çözümlere ilişkin amaç fonksiyonu değerleri hesaplanır.

Adım 5. Bir çözüm seçilir ve verilen mutasyon olasılığı (p_m) ile mutasyon işlemi uygulanarak yeni bir çözüm/birey oluşturulur. Bu çözümün amaç fonksiyonu değeri hesaplanır.

Adım 6. Tüm çözümler birleştirildikten sonra popülasyon boyutu büyüklüğünde en iyi çözümlerden yeni bir popülasyon oluşturulur. Gelecek jenerasyon için yeni popülasyon eski popülasyon ile yer değiştirir.

Adım 7. Durdurma kriterine ulaşıncaya kadar çözüm evrimleştirilir. Son popülasyondaki en iyi çözüm problemin çözüm olarak alınır.

Genetik Algoritmanın akış diyagramı Şekil 4.3'te verilmiştir.



Şekil 4.3. Genetik Algoritma (GA) akış diyagramı

4.3. Parçacık Sürüsü Optimizasyonu

Parçacık sürüsü optimizasyonu (Particle Swarm Optimization-PSO) kuşların ve balıkların yiyecek bulmada sergiledikleri davranışlarından esinlenerek Eberhart ve Kennedy (1995) ve Kennedy ve Eberhart (1995) tarafından önerilmiştir. PSO algoritması, uygulama kolaylığı ve yüksek yakınsama hızından dolayı en yaygın kullanılan metasezgisel yöntemlerden birisidir (Örkcü ve ark., 2015b; Acıtaş ve ark., 2019).

PSO'nun bazı özellikleri optimizasyon problemlerini çözmede etkili olmasını sağlamaktadır. Bu özellikler aşağıda verilmiştir (Rezaee Jordehi ve Jasni, 2013):

- PSO diğer sezgisel yöntemlerle karşılaştırıldığında, kullanıcılar tarafından ayarlanması gereken daha az parametreye sahiptir.
- PSO'nun temel kavramları çok basittir ve kodlaması çok kolaydır.
- PSO, hızlı bir yakınsama sağlamaktadır.

- Diğer birçok sezgisel yönteme göre, daha az hesaplama zamanına ihtiyaç duyar.
- PSO'nun davranışı, boyut artışından yüksek oranda etkilenmez.
- PSO, çok amaçlı, çok modelli, kısıtlı, kesikli ve tam sayılı değişkene sahip problemleri çözmede etkilidir.
- Zamansız yakınsamayı engellemek için PSO'da bazı stratejiler mevcut olduğundan PSO'nun başarı oranı çok yüksektir.

PSO, popülasyona dayalıdır ve kendini kontrol eden bir arama yöntemidir. Popülasyon, parçacıklar kümesinden oluşmaktadır. Her bir parçacık kendi en iyi pozisyonunu (*pbest*) kaydeder ve sürüdeki tüm parçacıklar tarafından kaydedilen en iyi pozisyonu (*gbest*) bilir. Daha sonra, tüm parçacıklar her bir iterasyonda hızını ve pozisyonunu bu bilgiler eşiğinde günceller.

***Hız Güncellemesi*:** Parçacığa uygulanacak değişim miktarı olarak tanımlanan hızın güncellenmesinde

$$v_i^{g+1} = wv_i^g + c_1 r_1 \left(pbest_i^g - x_i^g \right) + c_2 r_2 \left(gbest^g - x_i^g \right) \tag{4.6}$$

eşitliği kullanılır. Eşitlik (4.6)'da,

v_i^g : g . iterasyonda *i* bireyin hızını,

w : eylemsizlik katsayısını,

c_1 ve c_2 : hızlandırma katsayılarını,

r_1 ve r_2 : [0,1] aralığında düzgün dağılımdan rasgele üretilmiş sayıları,

$pbest_i^g$: g . iterasyona kadar *i.* bireyin en iyi pozisyonunu,

$gbest^g$: g . iterasyona kadar popülasyondaki bireylerin en iyi pozisyonunu göstermektedir (Talbi, 2009; Yang, 2014; Örkcü ve ark., 2015b).

Eşitlik (4.6)'da üç terim söz konusudur. Birinci terim, bir parçacığın aynı yönde kalma eğilimini modeller ve "eylemsizlik", "alışkanlık" veya "momentum" olarak adlandırılır. İkinci terim, parçacığın rasgele bir ağırlık ($c_1 R_{rand_1}$) ile kendi en iyi tecrübesine (*pbest*) doğru lineer eğilimidir. Bu terim "hafıza", "nostalji" veya "kendini tanıma" olarak adlandırılır. Üçüncü terim, parçacığın rasgele bir ($c_2 R_{rand_2}$) ile tüm parçacıkların en iyi tecrübesine (*gbest*) doğru lineer eğiliminin bir ölçüsüdür. Bu terim,

"iş birliği", "paylaşılan bilgi" veya "sosyal bilgi" olarak adlandırılır. (Rezaee Jordehi ve Jasni, 2013).

***Pozisyon Güncellemesi*:** Her bir parçacığın karar uzayındaki koordinatlarına göre pozisyonunun güncellenmesi,

$$x_i^{g+1} = x_i^g + v_i^{g+1} \tag{4.7}$$

eşitliği kullanılarak gerçekleştirilir. Burada**,** x_i^g , g . iterasyonda *i* bireyin pozisyonunu gösterir.

PSO algoritmasında kullanılan farklı parametre kümeleri algoritmanın hesaplama davranışını önemli ölçüde etkileyebilir. PSO, bazı parametrelerde etkili bir biçimde çalışırken, bazı parametreler için istenmeyen biçimde çalışabilir. Bundan dolayı, en iyi hesaplama davranışını sergileyen optimal PSO parametrelerini bulmak oldukça önemlidir (Rezaee Jordehi ve Jasni, 2013).

***Parçacık Sayısı*:** Çok küçük parçacık sayısı bir çözümü bulmak için çok uzun zamana neden olur ve hatta çözümü bulamayabilir. Çok fazla parçacık sayısı ise algoritmayı yavaşlatır. Kısacası, parçacık sayısı için bir uzlaşıya varılmalıdır. Deneysel çalışmalarda parçacık sayısının en az 20-30 arasında olması önerilmiş ve bu parçacık sayısı klasik test problemlerini çözmede yeterli olmuştur (Clerc, 2010).

***Hızlandırma Katsayıları (c_1, c_2)*:** Hızlandırma katsayıları parçacıkların yerel en iyi ve global en iyiye doğru hareketinin ağırlığını gösterir. Eğer c_1 ve c_2 sabitlerin değeri çok yüksek ise parçacıklar aniden hareket eder ve bu durum yanlış bir optimal noktaya takılma riskini arttırır. Diğer taraftan, c_1 ve c_2 sabitlerinin değerleri çok küçük ise, parçacıklar yavaşça hareket eder, hesaplama zamanı önemli ölçüde artar ve algoritma yakınsamayabilir.

c_1 arttığı zaman, parçacıkların yerel en iyiye doğru yönelme olasılığı artar ve global en iyiye doğru yönelme olasılığı azalır. Ayrıca, c_2 arttığı zaman parçacıkların global en iyiye doğru yönelme olasılığı artarken yerel en iyiye doğru yönelme olasılığı azalır. Literatürde, birçok problem için genellikle $c_1 = c_2 = 2$ olarak kabul edilir. Fakat

normal olarak bazı deneme yanılmalarla kabul edilen bu değerlerden daha etkili hesaplama yeteneğine sahip olan başka değerler de elde edilebilir (Rezaee Jordehi ve Jasni, 2013).

***Eylemsizlik katsayısı (w)*:** Eylemsizlik katsayısı, parçacığın geçmişteki hızının mevcut hızı üzerindeki etkisini kontrol etmeyi sağlar. PSO' da global ve yerel arama yetenekleri arasındaki denge temel olarak eylemsizlik katsayısı tarafından kontrol edilir. Büyük bir eylemsizlik katsayısı, global aramayı (yeni alanlar aramayı) kolaylaştırırken, küçük eylemsizlik katsayısı yerel aramayı (mevcut bölgede ayrıntılı aramayı) kolaylaştırır. Uygun bir eylemsizlik katsayısının seçimi global ve yerel arama arasında bir denge sağlar ve optimal çözümü bulmak için daha az iterasyona ihtiyaç duyulur. PSO algoritması, en iyi performansını $[0.5,1.2]$ aralığındaki eylemsizlik katsayısı ile sergiler. Dahası, zamanla azalan bir eylemsizlik katsayısı (1.4'ten 0'a doğru) sabit bir eylemsizlik katsayısından daha iyidir. Çünkü, büyük eylemsizlik katsayısı başlangıçta iyi çözümlerin bulunmasına yardımcı olurken, aramanın sonlarına doğru küçük bir eylemsizlik katsayısı aramayı kolaylaştırır (Shi ve Eberhart, 1998).

D boyutlu bir optimizasyon problemi için $\boldsymbol{x}=[x_1, x_2, ..., x_D]$ parametre vektörünü ve $f(\boldsymbol{x})$ amaç fonksiyonunu göstermek üzere PSO algoritmasının işleyiş adımları aşağıda verilmiştir.

***PSO algoritmasının uygulama adımları*:**

Adım 1. Amaç fonksiyonu, maksimum iterasyon sayısı ve PSO'nun parametreleri; eylemsizlik katsayısı (w), hızlandırma katsayıları (c_1 ve c_2) ve NP belirlenir.

Adım 2. Eşitlik (4.1) kullanılarak, önceden belirlenmiş arama uzayında parametreler için rasgele başlangıç popülasyon $\boldsymbol{x}_i^0=[x_{1,i}^0, x_{2,i}^0, ..., x_{D,i}^0], i=1,2,...NP$ oluşturulur ve başlangıç popülasyon için amaç fonksiyonu değerleri $f(\boldsymbol{x}_i^0)$ hesaplanır.

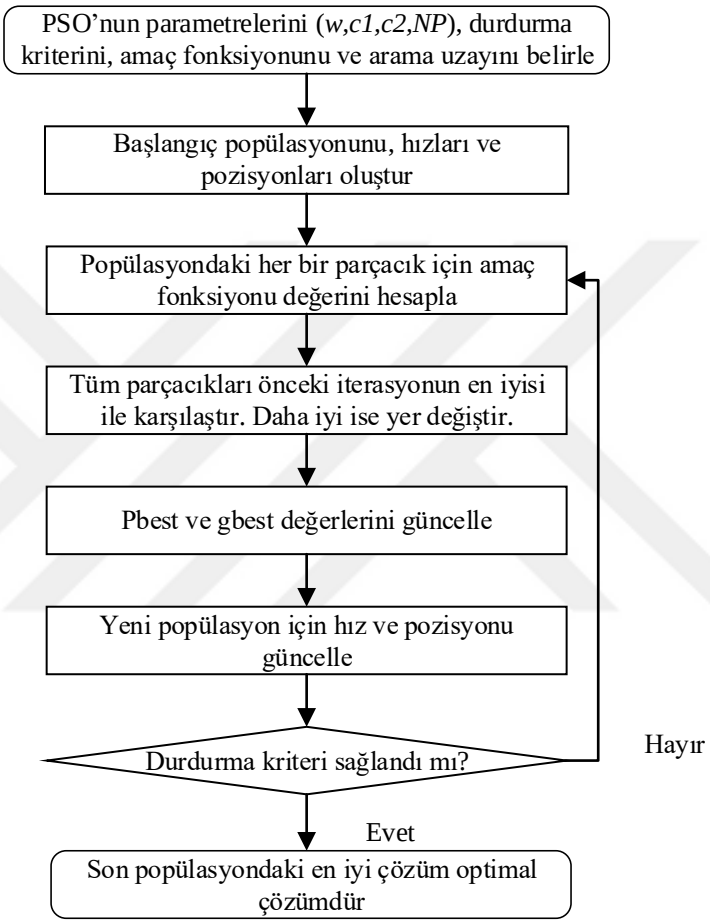
Adım 3. Her bir parçacık için kendi en iyi pozisyonu (*pbest*) ve sürüdeki tüm parçacıkların en iyi pozisyonu (*gbest*) bulunur.

Adım 4. Popülasyondaki her bir çözüm için Eşitlik (4.6) kullanılarak parçacığın hızı ve Eşitlik (4.7) kullanılarak parçacığın pozisyonu güncellenerek yeni popülasyon oluşturulur.

Adım 5. Yeni popülasyon ile mevcut popülasyon karşılaştırılır ve yeni popülasyonun değeri mevcut popülasyondan daha iyi ise yeni popülasyon mevcut popülasyon ile yer değiştirir.

Adım 6. Durdurma kriterine ulaşana kadar çözüm evrimleştirilir. Son popülasyondaki en iyi çözüm, problemin çözümü olarak alınır.

PSO algoritmasının akış diyagramı Şekil 4.4'te verilmiştir.



Şekil 4.4. Parçacık Sürüsü Optimizasyonu (PSO) akış diyagramı

4.4. Diferansiyel Gelişim Algoritması

Diferansiyel gelişim (DE) algoritması, sürekli optimizasyon problemleri için en başarılı yaklaşımlardan biridir ve Storn (1996) ve Storn ve Price (1997) tarafından önerilen popülasyona dayalı bir algoritmadır. DE algoritması, GA' ya benzer olarak çaprazlama, mutasyon ve seçim operatörlerini kullanmaktadır. Fakat, daha iyi çözümlerin elde edilmesinde GA çaprazlama operatörünü kullanırken DE mutasyon operatörünü kullanmaktadır. DE'deki mutasyon operatörü, popülasyondan rasgele

seçilen iki çözümün farkına dayanmaktadır. Algoritma, arama mekanizması olarak mutasyon operatörünü ve arama uzayındaki aday bölgelere doğru yönelmek için seçim operatörünü kullanmaktadır (Storn, 1996; Price ve ark., 2006).

Diğer evrimsel hesaplama algoritmalarının aksine, standart DE algoritması çok basittir ve kodlaması çok daha kolaydır. Ayrıca, standart DE algoritması, mutasyon (ölçekleme) faktörü, çaprazlama faktörü ve popülasyon boyutu olmak üzere yalnızca üç kontrol parametresine sahiptir. Buna rağmen, DE algoritması çeşitli amaç fonksiyonlarını optimize ederken doğruluk, hesaplama hızı ve sağlamlık açısından dikkate değer bir performans sergilemektedir (Das ve ark., 2016; Mashwani ve ark., 2018; Yonar ve Pehlivan, 2020b).

DE algoritmasının farklı çeşitlerini göstermek için $DE/x/y/z$ gösterimi kullanılmaktadır.

x, mutasyonda kullanılacak vektörü belirtir, örneğin, "rand" rasgele seçilen bir popülasyon vektörünü, "best" ise mevcut popülasyondan seçilmiş en iyi vektörü gösterir. y, kullanılan fark vektörlerinin sayısını ve z, çaprazlama aşamasını gösterir. Örneğin, "bin" bağımsız Binom denemelerine bağlı çaprazlamayı, "exp" ise üstel çaprazlamayı simgeler. (Storn ve Price, 1997; Talbi, 2009; Örkcü ve ark., 2015a). Bu tez çalışmasında standart varyasyon $DE/rand/1/bin$ kullanılmıştır.

Diferansiyel gelişim algoritması, mutasyon, çaprazlama ve seçim olmak üzere üç temel adımdan oluşmaktadır. Popülasyon başlatıldıktan sonra, mutasyon, çaprazlama ve seçim süreçleri optimal değere ulaşıncaya kadar veya önceden belirlenen bir durdurma kriteri sağlanana kadar devam eder (Price ve ark., 2006).

***Mutasyon*:** Başlangıç popülasyon oluşturulduktan sonra mevcut jenerasyondaki her bir popülasyon üyesi için mutasyon operatörü ile bir donör vektör oluşturulur. Rasgele seçilen iki amaç vektörü arasındaki ağırlıklandırılmış farka rasgele seçilen üçüncü bir amaç vektörünün eklenmesiyle donör vektörü,

$$\vec{v}_i^{\,g+1} = \vec{x}_{r1}^{\,g} + F\left(\vec{x}_{r2}^{\,g} - \vec{x}_{r3}^{\,g}\right) \tag{4.8}$$

biçiminde oluşturulur. Burada, $r_1, r_2, r_3 \in \{1, 2, ..., NP\}$ birbirinden farklı rasgele seçilen tamsayılardır ve mutasyon (ölçek) faktörü $F \in (0,1+)$ fark vektörlerini ölçeklendiren

pozitif kontrol parametresidir. *F* için herhangi bir üst değer sınırlaması olmamasına rağmen *F*'nin 1'den büyük değerler için etkili sonuçlar nadiren bulunmaktadır.

***Çaprazlama (Rekombinasyon)*:** Mutasyon süreci ile donör vektör oluşturulduktan sonra, popülasyondaki çeşitliliği arttırmak amacıyla çaprazlama operatörü kullanılır. Genel olarak, üstel ve düzgün çaprazlama operatörleri kullanılmaktadır. Çaprazlama sürecinde deneme vektörü olarak adlandırılan bir komşu çözüm $\vec{u}_i^{\,g} = \left[u_{1,i}^g, u_{2,i}^g, ..., u_{D,i}^g\right]$ üretmek için donör vektörün bileşenleri ile amaç vektörün bileşenleri ile karıştırılır. Deneme vektörü, binomial çaprazlama stratejisi kullanılarak

$$\vec{u}_{j,i}^{\,g+1} = \begin{cases} \vec{v}_{j,i}^{\,g}, & eğer\ rand_{i,j}(0,1) \le CR \ \ veya \ \ j = j_{rand} \\ \vec{x}_{j,i}^{\,g}, & dd \end{cases} \tag{4.9}$$

biçiminde oluşturulur. Burada; $CR \in [0,1]$ kullanıcı tarafından önceden belirlenmiş çaprazlama sabitidir. Binomial uygulamada 0 ile 1 arasında üretilen rasgele bir sayının CR'den küçük olması durumunda elemanlar mutasyona uğramış çözümden değil ise ebeveynden alınmaktadır. $j = j_{rand}$ ile yeni üretilen vektörün en az bir parametresinin ebeveynden farklı olması sağlanmaktadır (Karaboğa, 2014).

***Seçim*:** Algoritmanım son adımı seçim olarak adlandırılır, hedef ve deneme vektörlerinden hangisinin gelecek jenerasyona aktarılacağı seçim aşamasında belirlenir. Maksimizasyon problemi için seçim stratejisi,

$$\vec{x}_i^{\,g+1} = \begin{cases} \vec{u}_i^{\,g}, & f(\vec{u}_i^{\,g}) \ge f(\vec{x}_i^{\,g}) \\ \vec{x}_i^{\,g}, & f(\vec{u}_i^{\,g}) < f(\vec{x}_i^{\,g}) \end{cases} \tag{4.10}$$

olarak tanımlanır ve $f(\cdot)$ minimize edilmek istenilen amaç fonksiyonunu göstermektedir. Minimizasyon probleminde, $\vec{u}_i^{\,g}$'nin seçimi için $f(\vec{u}_i^{\,g}) \le f(\vec{x}_i^{\,g})$ şartı uygulanmalıdır. Burada, gelecek jenerasyona aktarılan vektör, daima rakip vektörden daha iyidir ve bu çeşit bir seçim popülasyonun global en iyisinin daima gelecek nesile aktarılmasını garanti etmektedir. DE'de seçim stratejisi olarak elitist strateji uygulanmaktadır.

D boyutlu bir optimizasyon problemi için $\boldsymbol{x} = [x_1, x_2, ..., x_D]$ parametre vektörünü ve $f(\boldsymbol{x})$ amaç fonksiyonunu göstermek üzere DE algoritmasının işleyiş adımları aşağıda verilmiştir.

DE algoritmasının uygulama adımları:

Adım 1. DE parametreleri; NP, ölçekleme faktörü ($F \in (0,1]$), ve çaprazlama faktörü (C_r) belirlenir.

Adım 2. Eşitlik (4.1) kullanılarak, önceden belirlenmiş arama uzayında parametreler için rasgele başlangıç popülasyon $\boldsymbol{x}_i^0 = [x_{1,i}^0, x_{2,i}^0, ..., x_{D,i}^0], i = 1, 2, ...NP$ oluşturulur ve başlangıç popülasyonu için amaç fonksiyonu değerleri $f(\boldsymbol{x}_i^0)$ hesaplanır.

Adım 3. Popülasyondaki her bir çözüm için rasgele üç farklı vektör seçilir ve Eşitlik (4.8) kullanılarak yeni donör vektör üretilir.

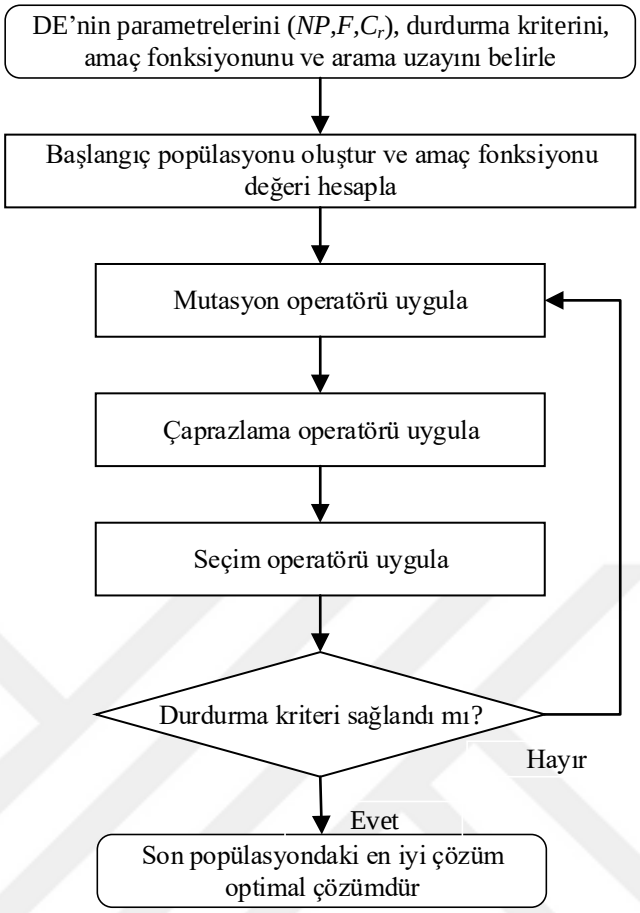
Adım 4. Rasgele bir sayı $j_{rand} \in [0, D]$ seçilir ve popülasyonun çeşitliliğini arttırmak için Eşitlik (4.9) kullanılarak çaprazlama operatörü uygulanır.

Adım 5. Gelecek jenerasyona hangi çözümlerin aktarılacağını belirlemek için Eşitlik (4.10)'da verilen seçim aşaması uygulanır.

Burada, gelecek jenerasyona iletilecek çözüm daima mevcut çözümden daha iyidir ve böyle bir seçim popülasyonun global en iyisinin gelecek jenerasyona aktarılmasını daima garanti eder. Diğer bir ifadeyle, DE'nin seçim aşamasında elitist strateji uygulanmaktadır.

Adım 6. Yeni popülasyon mevcut popülasyon ile yer değiştirir. Durdurma kriterine ulaşana kadar çözüm evrimleştirilir. Son popülasyondaki en iyi çözüm, problemin çözümü olarak alınır.

DE algoritmasının akış diyagramı Şekil 4.5'te verilmiştir.



Şekil 4.5. Diferansiyel Gelişim (DE) algoritması akış diyagramı

4.5. Yapay Arı Kolonisi Algoritması

Yapay arı kolonisi (Artificial Bee Colony-ABC) algoritması arıların yiyecek arama davranışlarını modellenmesiyle ilk olarak Karaboga (2005) tarafından önerilmiştir. Daha sonra, ABC algoritması gerçek-parametreli optimizasyon problemlerini çözmek için Akay ve Karaboğa (2012) tarafından uyarlanmıştır. ABC algoritması, çok basit, güçlü ve popülasyona dayalı stokastik bir metasezgisel algoritmadır (Das ve ark., 2016). Ayrıca, ABC algoritması tasarımından dolayı çok az kontrol parametresine sahiptir. Bu parametreler, yiyecek kaynaklarının sayısı veya koloni boyutu (popülasyon sayısı) ile limit değeridir (Yonar ve Pehlivan, 2020a).

ABC algoritmasında, bir yiyecek kaynağının pozisyonu optimizasyon problemindeki mümkün çözümlere, yiyecek kaynağının miktarı ise ilgili çözümün kalitesi veya uygunluğuna karşılık gelmektedir (Karaboğa ve ark., 2014).

ABC algoritmasında, arılar, “işçi arılar”, “gözcü arılar” ve “izci (kaşif) arılar” olmak üzere üç grupta toplanır. Sürünün öncüsü olan işçi arılar yiyecek kaynaklarını keşfeder, balları toplar ve yiyecek kaynaklarının miktarı ile ilgili bilgiyi kovanda paylaşırlar. Gözcü arılar, bu bilgileri temel alarak daha iyi yiyecek kaynakları keşfetmeye çalışırlar. Yiyecek kaynağı kötü olan bir arı izci arı olarak atanır ve rasgele yeni bir yiyecek kaynağı bulur. Böylece, bu üç arı grubu birbirleriyle etkili bir işbirliği içinde olurlar (Karaboğa ve Öztürk, 2011).

ABC, dört temel aşamadan oluşmaktadır:

- Başlangıç Aşaması
- İşçi Arılar Aşaması
- Gözcü Arılar Aşaması
- İzci Arılar Aşaması

Başlangıç aşaması tamamlandıktan sonra, işçi arılar, gözcü arılar ve izci arılar aşamaları önceden belirlenen bir durdurma kriteri sağlanana kadar devam etmektedir. Maksimizasyon problemi için uyarlanan ABC algoritmasının aşamaları aşağıda özetlenmiştir (Akay ve Karaboğa, 2012).

***Başlangıç Aşaması*:** Arama uzayındaki çözümlere karşılık gelen başlangıç yiyecek kaynaklarının yerleri, parametre sınırları kullanılarak Eşitlik (4.1)’e göre rasgele üretilir.

***İşçi Arılar Aşaması*:** Her bir işçi arı yalnızca bir yiyecek kaynağı ile ilişkilidir. Dolayısıyla, yiyecek kaynaklarının sayısı işçi arıların sayısına eşittir. Her bir işçi arı, mevcut pozisyonunun bir komşusu tarafından sağlanan bilgiyi kullanarak yeni bir yiyecek kaynağı (v_{ij}) üretir. Yeni yiyecek kaynakları,

$$v_{ij} = x_{ij} + \phi_{ij}\left(x_{ij} - x_{kj}\right) \tag{4.11}$$

eşitliği kullanılarak elde edilir. Burada, $k = 1,2,...,NP$ i’den farklı rasgele seçilen bir sayıdır. $\phi_{ij} \in [0,1]$ düzgün dağılım kullanılarak rasgele üretilen bir sayıdır. Eşitlik (4.11)’den görüldüğü gibi eski ve yeni yiyecek kaynakları arasındaki fark azaldıkça, eski çözümdeki değişim miktarının da azalacağı görülmektedir. Yani, optimal çözüme yaklaştıkça değişim miktarı da azalmaktadır. Eğer yeni yiyecek kaynağı (v_{ij}) önceden

belirlenen parametre sınırlarını aşarsa bu değer sınıra eşitlenir. Yani, $v_{ij} > x_j^{\max}$ ise $v_{ij} = x_j^{\max}$ olarak, $v_{ij} < x_j^{\min}$ ise $v_{ij} = x_j^{\min}$ olarak alınır.

Yeni yiyecek kaynağı (v_{ij}) elde edildikten sonra yeni ve eski yiyecek kaynakları arasında açgözlü bir seçim işlemi uygulanır. Başka bir ifade ile eğer $f(\boldsymbol{v}_i) \geq f(\boldsymbol{x}_i)$ ise yeni yiyecek kaynağı eski yiyecek kaynağının yerini alır ve yeni pozisyon olarak hafızaya gönderilir, aksi durumda eski yiyecek kaynağı hafızada kalmaya devam eder. Burada, $\boldsymbol{x}_i = [x_{i1}, x_{i2}, ..., x_{iD}]$ ve $\boldsymbol{v}_i = [v_{i1}, v_{i2}, ..., v_{iD}]$'dir ve $f(\cdot)$ amaç fonksiyonunu göstermektedir. Eğer çözüm iyileşmemişse geliştirememe sayacı 1 arttırılır, iyileşmişse sayaç sıfırlanır.

***Gözcü Arılar Aşaması*:** Gözcü arılar işçi arılar tarafından paylaşılan bilgiyi değerlendirir ve yiyecek kaynağının Eşitlik (4.12)'de verilen uygunluk (fitness) değerine bağlı olarak bir olasılığa dayalı seçim ile yiyecek kaynağı seçerler.

$$fitness_i = \exp\left(f_i / \bar{f} \right) \tag{4.12}$$

Olasılığa dayalı bu seçim "rulet çemberi seçimi" yöntemi kullanılarak yapılabilir. Yiyecek kaynaklarının seçim olasılıkları,

$$p_i = \frac{fitness_i}{\sum_{i=1}^{SN} fitness_i} \tag{4.13}$$

eşitliği ile hesaplanır. Eşitlik (4.13)'ten yiyecek kaynağının uygunluk değeri ne kadar yüksek ise gözcü arıların bu yiyecek kaynağını seçme olasılığının da o kadar yüksek olduğu görülmektedir. Bir yiyecek kaynağı seçildikten sonra, işçi arılar aşamasındakine benzer olarak, Eşitlik (4.1) kullanılarak yeni bir yiyecek kaynağı üretilerek eski ve yeni yiyecek kaynakları arasında açgözlü seçim işlemi uygulanır. Eğer çözüm iyileşmemişse geliştirememe sayacı 1 arttırılır, iyileşmişse sayaç sıfırlanır. Bu süreç, tüm gözcü arılar yiyecek kaynakları alanlarına dağıtılıncaya kadar devam eder.

***İzci Arılar Aşaması*:** Bir arının yiyecek kaynağının nektar miktarı önceden belirlenen bir limit değeri kadar değişmiyorsa, bu arı izci arı olarak atanır ve Eşitlik (4.1)'i kullanarak rasgele yeni bir yiyecek kaynağı bulur. Bu aşamada, geliştirememe sayacı sıfırlanır.

D boyutlu bir optimizasyon problemi için $\boldsymbol{x} = [x_1, x_2, ..., x_D]$ parametre vektörünü ve $f(\boldsymbol{x})$ amaç fonksiyonunu göstermek üzere ABC algoritmasının işleyiş adımları aşağıda verilmiştir.

***ABC algoritmasının uygulama adımları*:**

Adım 1.Amaç fonksiyonu, maksimum iterasyon sayısı ve ABC'nin parametreleri; NP, ve limit değeri (*l*) belirlenir.

Adım 2. Eşitlik (4.1) kullanılarak, önceden belirlenen arama uzayında parametreler için rasgele başlangıç popülasyonu $\boldsymbol{x}_i^0 = [x_{1,i}^0, x_{2,i}^0, ..., x_{D,i}^0]$, $i = 1, 2, ...NP$ oluşturulur ve başlangıç popülasyon için amaç fonksiyonu değerleri $f(\boldsymbol{x}_i^0)$ hesaplanır.

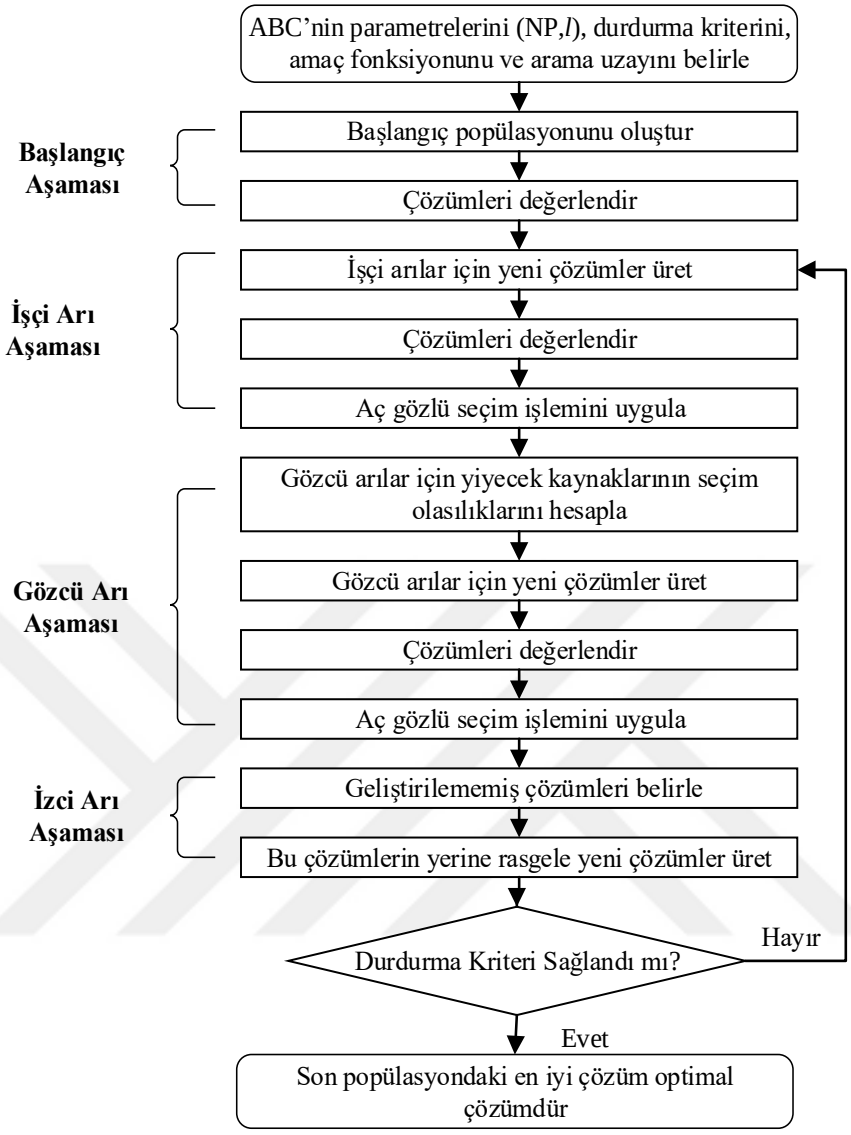
Adım 3. Eşitlik (4.11) kullanılarak işçi arılar için yeni bir yiyecek kaynağı $\boldsymbol{v}_i$ oluşturulur.

Adım 4. Eşitlik (4.12) ve (4.13) kullanılarak hesaplanan p_i olasılığına bağlı olarak seçilen $\boldsymbol{x}_i$ çözümünden gözcü arılar için yeni çözümler ($\boldsymbol{v}_i$) oluşturulur ve bu çözümlerin amaç fonksiyonu değerleri hesaplanır. $\boldsymbol{x}_i$ ve $\boldsymbol{v}_i$ arasında seçim işlemi uygulanır ve iyi olan seçilir. Eğer, $\boldsymbol{x}_i$ çözümü amaç fonksiyonu değerini geliştirmiyorsa geliştirememe sayacı 1 arttırılır. Bu süreç tüm gözcü arılar yiyecek kaynağı alanına dağıtılıncaya kadar devam eder.

Adım 5. İzci arılar için geliştirilememiş çözümler belirlenir. Eğer $c_i > l$ ise, $\boldsymbol{x}_i$ çözümü Eşitlik (4.1) kullanılarak elde edilen yeni çözümle yer değiştirir. Bu aşamada geliştirememe sayacı sıfırlanır.

Adım 6. Durdurma kriterine ulaşana kadar çözüm evrimleştirilir. Son popülasyondaki en iyi çözüm problemin çözümü olarak alınır.

ABC algoritmasının akış diyagramı Şekil 4.6'da verilmiştir.



Şekil 4.6. Yapay Arı Kolonisi (ABC) algoritması akış diyagramı

4.6. Levy Uçuşlu Yapay Arı Kolonisi Algoritması

Levy uçuşları adım uzunluğu Levy dağılımından elde edilen bir rasgele yürüyüştür ve

$$L(s) \sim |s|^{-1-\beta}, \ 0 < \beta \leq 2 \tag{4.14}$$

olarak ifade edilir. Burada, β bir indeks ve s adım uzunluğudur. Matematiksel olarak Levy dağılımı,

$$f(s,\mu,\gamma)=\begin{cases}\sqrt{\dfrac{\gamma}{2\pi}}\exp\left(-\dfrac{\gamma}{2(s-\mu)}\right)\left(\dfrac{1}{(s-\mu)^{3/2}}\right), & 0<\mu\le s<\infty\\ 0, & s\le 0\end{cases} \tag{4.15}$$

biçiminde tanımlanır (Yang ve Deb, 2013; Haklı ve Uğuz, 2014). Burada, μ konum parametresini $\gamma>0$ ölçek parametresini göstermektedir. Levy dağılımı $0<\beta<2$ için sonlu ortalama ve sonlu varyansa sahiptir. Genel olarak, Levy dağılımı Fourier dönüşümü yardımıyla,

$$F(k)=\exp\left[-\alpha|\mathrm{k}|^{\beta}\right],\ 0<\beta\le 2 \tag{4.16}$$

şeklinde tanımlanabilir. Burada, α ölçek parametresidir. Levy dağılımı, $\beta=1$ için Cauchy dağılımı, $\beta=2$ için Normal dağılım olarak bilinmektedir (Haklı and Uğuz, 2014;Yang and Deb, 2013).

"Keşfetme" ve "işletme" kriterleri, bir metasezgisel algoritmadan iyi bir performans elde etmek için dengelenmesi gereken iki önemli zıt kriterdir. Çeşitlendirme kriteri olarak da bilinen keşfetme kriteri, global optimale ulaşmak için arama uzayındaki daha önce keşfedilmemiş bölgeleri ziyaret etmeyi ifade eder. Yoğunlaşma kriteri olarak da bilinen işletme kriteri, daha iyi çözümlere ulaşmak için önceki iyi çözümlerin bilgisini kullanmayı ifade eder.

ABC algoritmasında, arama uzayındaki keşfetme süreci işçi ve gözcü arılar tarafından yapılırken, işletme süreci izci arılar tarafından yürütülür. Karaboğa ve Akay (2009), ABC algoritmasının işletme sürecinin iyi bir performans sergilemediğini ve bundan dolayı ABC algoritmasının lokal minimuma veya lokal maksimuma takıldığını belirlemişlerdir. Arama uzayındaki işletme işlemini yürüten izci arılar mevcut veya global en iyi çözümden daha iyi sonuçlar elde etmek için rasgele yeni çözümler üretir. Fakat, rasgele üretilen bu yeni çözümler genellikle mevcut veya global en iyi çözümden daha kötü sonuçlar verir. Bu durumun üstesinden gelmek için izci arılar aşamasında yeni çözümler üretilirken Levy uçuşlarının kullanılması önerilmiştir (Hakli ve Uguz, 2013).

Levy uçuşları kullanılarak, izci arı aşamasında yeni çözümler

$$x_{ij}^{new}=x_{ij}+\alpha\oplus levy(\beta),\ i=1,2,...,NP,\ j=1,2,...,D \tag{4.17}$$

eşitliği kullanılarak elde edilir. Burada; α adım uzunluğunu, β Levy uçuşunun parametresini ve $\oplus$ çoklu çarpmayı simgelemektedir. β'nın farklı değerleri için dağılım değişmektedir. Küçük β değerleri dağılımın uzun atlayışlar yapmasını sağlarken, büyük β değerleri dağılımın kısa atlayışlar yapmasını sağlamaktadır. (Lee ve Yao, 2001; Haklı ve Uğuz, 2014).

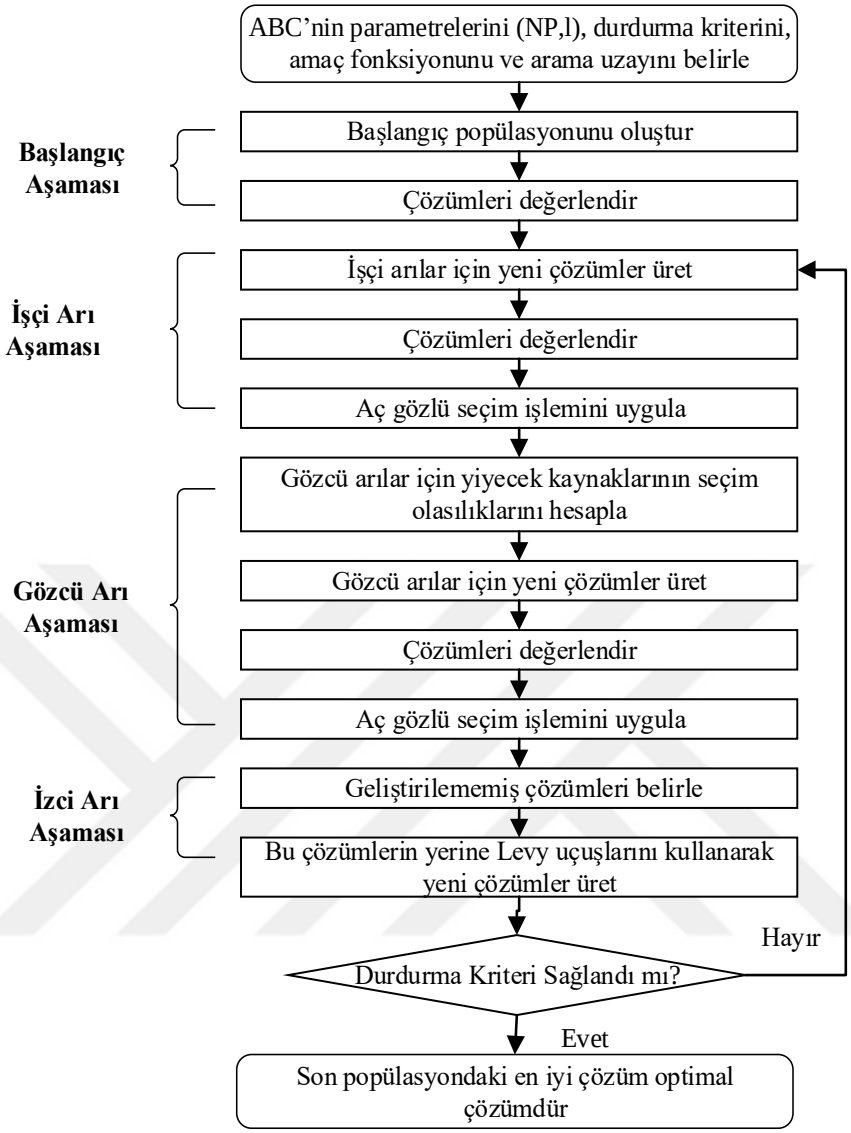
Mantegna (1994), α $(0.3 \le \alpha \le 1.99)$ kontrol parametreli Levy dağılımından rasgele bir sayı üretmek için bir algoritma önermiştir. Bu algoritmaya göre adım uzunluğu $s = \alpha \oplus levy(\beta)$,

$$s = 0.01 \frac{u}{|v|^{\frac{1}{\beta}}} \left(x_i - x_i^{best} \right) \tag{4.18}$$

eşitliği kullanılarak elde edilebilir. Eşitlik (4.18)'de, $u \sim N(0, \sigma_u^2)$ ve $v \sim N(0, \sigma_v^2)$ olmak üzere

$$\sigma_u = \left[\frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma((1+\beta)/2)\beta 2^{(\beta-1)/2}} \right]^{1/\beta} \text{ , } \sigma_v = 1 \text{ ve } \Gamma(t) = \int_0^\infty x^{t-1} e^{-t} dx \text{ 'dir.}$$

LABC algoritmasının akış diyagramı Şekil 4.7'de verilmiştir.



Şekil 4.7. LABC algoritması akış diyagramı

5. TEK DEĞİŞKENLİ DAĞILIMLAR İÇİN METASEZGİSEL YÖNTEMLER İLE PARAMETRE TAHMİNİ

Bu bölüm, sürekli dağılımlardan 3-parametreli Weibull, 3-parametreli Gamma dağılımı ile kesikli dağılımlardan kesikli Weibull dağılımının parametrelerinin metasezgisel yöntemler ile tahminlerini içermektedir.

5.1. 3-p Weibull Dağılımının Parametrelerinin Tahmini İçin Levy Uçuşlu Yapay Arı Kolonisi Algoritması

Bu kesimde, sürekli dağılımlardan biri olan 3-p Weibull dağılımının parametrelerinin ML yöntemi ile tahminine yer verilmiştir. 3-p Weibull dağılımının parametrelerinin ML tahmini için LABC algoritması kullanılarak Monte-Carlo simülasyon çalışması yapılmıştır. Daha sonra, 3-p Weibull dağılımının diğer metasezgisel yöntemlerden ABC (Karaboga, 2005), PSO (Örkcü et al., 2015b), LPSO (Haklı ve Uğuz, 2014), DE (Örkcü et al., 2015a) , SA (Abbasi et al., 2006) ve GA (Goldberg ve Holland, 1988) ile elde edilen parametre tahminleri LABC'den elde edilen değerler ile karşılaştırmalı olarak verilmiştir. Son olarak gerçek bir veri uygulaması sunulmuştur.

5.1.1. 3-p Weibull dağılımının parametrelerinin tahmini için ML yöntemi

$X_1, X_2, ..., X_n$'ler olasılık yoğunluk fonksiyonu Eşitlik (3.1)'de verilen $\text{Weibull}(\gamma, \eta, \beta)$ dağılımından gelen rasgele değişkenler olmak üzere, olabilirlik (L) ve log-olabilirlik ($\ln L$) fonksiyonları sırasıyla,

$$L = \prod_{i=1}^{n} \frac{\beta}{\eta}\left(\frac{x_i-\gamma}{\eta}\right)^{\beta-1} \mathrm{e}^{-\left(\frac{x_i-\gamma}{\eta}\right)^{\beta}} = \left(\frac{\beta}{\eta}\right)^{n} e^{\sum_{i=1}^{n}-\left(\frac{x_i-\gamma}{\eta}\right)^{\beta}} \prod_{i=1}^{n}\left(\frac{x_i-\gamma}{\eta}\right)^{\beta-1} \tag{5.1}$$

ve

$$lnL = nln\left(\frac{\beta}{\eta}\right) - \sum_{i=1}^{n}\left(\left(\frac{x_i-\gamma}{\eta}\right)^{\beta} - (\beta-1)ln\left(\frac{x_i-\gamma}{\eta}\right)\right) \tag{5.2}$$

olarak elde edilir. Eşitlik (5.2)'den olabilirlik denklemleri,

$$\frac{\partial \ln L}{\partial \eta} = -\frac{n}{\eta} - \sum_{i=1}^{n}\left[\frac{\beta}{\eta}\left(\frac{x_i-\gamma}{\eta}\right)^{\beta} + \frac{\beta-1}{\eta}\right] = 0 \tag{5.3}$$

$$\frac{\partial \ln L}{\partial \beta} = \frac{n}{\beta} - \sum_{i=1}^{n}\left[\beta\left(\frac{x_i-\gamma}{\eta}\right)^{\beta-1} + \ln\left(\frac{x_i-\gamma}{\eta}\right)\right] = 0 \tag{5.4}$$

$$\frac{\partial \ln L}{\partial \gamma} = \sum_{i=1}^{n}\left[\beta\left(\frac{x_i-\gamma}{\eta}\right)^{\beta-1} + (\beta-1)\left(\frac{\eta}{x_i-\gamma}\right)\right] = 0 \tag{5.5}$$

biçiminde elde edilir. Olabilirlik denklemleri lineer olmayan fonksiyonlar içerdiğinden bu denklemlerin çözümü analitik olarak elde edilememektedir. Bundan dolayı, bu çalışmada metasezgisel yöntemlerden LABC algoritmasının kullanılması önerilmiştir.

5.1.2. Simülasyon çalışması

Monte-Carlo simülasyon çalışması ile LABC algoritmasının etkinliğini göstermek için Abbasi ve ark. (2006) ve Örkcü ve ark. (2015b)'nın çalışmalarına benzer olarak çeşitli gerçek parametre değerleri ($\theta=(2,2,2)$, $\theta=(4,3,2)$, $\theta=(5,2,3)$), farklı örneklem büyüklükleri (100, 500, 1000 ve 2500), koloni boyutları (50 ve 100) ve arama uzayları ($[0,10]$ ve $[0,100]$) ele alınmıştır. Ayrıca, Levy dağılımındaki limit değeri 5 olarak, β indeksinin değeri 2 olarak alınmıştır. Algoritmanın hızlı yakınsama eğiliminden dolayı, iterasyon sayısı 100 ile sınırlandırılmıştır.

LABC algoritması, her bir durum için 1000 kez çalıştırılmıştır. Parametre tahminleri ($\hat{\gamma},\hat{\eta},\hat{\beta}$) için ortalama simülasyon sonuçları, amaç fonksiyonu değerleri (φ), amaç fonksiyonunun tahmini değerleri ($\hat{\varphi}$) ve Def değerleri tüm durumlar için Çizelge 5.1-5.3'te verilmiştir.

Çizelge 5. 1. **P**arametreleri $(\gamma,\eta,\beta)=(2,2,2)$ olan 3-p Weibull dağılımı için simülasyon sonucunda LABC algoritması ile elde edilen ortalama parametre tahmin sonuçları, $\hat{\phi}$, φ ve Def değerleri

n	Arama Uzayı	[0-10]		[0-100]	
	Koloni Boyutu	50	250	50	250
100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0458,1.9403,1.9677)	(2.0574,1.9223,1,9460)	(2.0167,1.9655,2.0353)	(2.0492,1.9397,1,9679)
	$\hat{\phi}$	-133.703	-128.567	-142.214	-123.567
	φ	-133.691	-128.785	-141.257	-123.555
	Def	0.123159^{ps}	0.125416	0.148753	0.108246^{ps}
500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0189,1.9722,1.9712)	(2.0232,1.9709,1.9730)	(2.0172,1.9780,1.9912)	(2.0149,1.9791,1.9875)
	$\hat{\phi}$	-633.085	-648.309	-641.685	-635.851
	φ	-633.326	-648.632	-641.705	-635.352
	Def	0.017691^{p}	0.017285^{ps}	0.0196^{s}	0.020178
1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0095,1.9929,1.9842)	(2.0060,1.9941,2.0018)	(2.0114,1.9892,1,9909)	(2.0107,1.9940,1.9989)
	$\hat{\phi}$	-1307.84	-1263,63	-1277.61	-1309.69
	φ	-1307.82	-1263,66	-1277.02	-1309.69
	Def	0.009485^{p}	0.008843^{s}	0.00964	0.008475^{ps}
2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0054,1.9944,1.9943)	(2.0053,1.9928,1.9926)	(2.0040,1.9956,1.9949)	(2.0064,1.9900,1.9850)
	$\hat{\phi}$	-3217.05	-3163.87	-3267.08	-3199,38
	φ	-3216.79	-3163.56	-3263.78	-3199,65
	Def	0.003147	0.002817^{ps}	0.003063^{ps}	0.003538

p:Koloni boyutu sabit iken arama uzayına göre en iyi sonuçları gösterir.
s:Arama uzayı sabit iken koloni boyutuna göre en iyi sonuçları gösterir.

Çizelge 5. 2. Parametreleri $(\gamma,\eta,\beta)=(4,3,2)$ olan 3-p Weibull dağılımı için simülasyon sonucunda LABC algoritması ile elde edilen ortalama parametre tahmin sonuçları, $\hat{\phi}$, φ ve Def değerleri

n	Arama Uzayı	[0-10]		[0-100]	
	Koloni Boyutu	50	250	50	250
100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.1004,2.8413,1.9073)	(4.1050,8594,1.9246)	(4.0500,2.9337,1.9828)	(4.0272,2.943,1.9837)
	$\hat{\phi}$	-166.337	-172.259	-175.538	-167.644
	φ	-166.95	-172.625	-175.343	-167.198
	Def	0.205823^{ps}	0.224587^{p}	0.272	0.253827^{s}
500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0261,2.9746,1.9877)	(4.0264,2.9604,1.9715)	(4.0316,2.9713,1.9822)	(4.0200,2.9883,1.9886)
	$\hat{\phi}$	-852.464	-875.416	-839.486	-864.047
	φ	-852.496	-875.068	-839.652	-864.374
	Def	0.023602^{ps}	0.031688	0.031477	0.027989^{ps}
1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0166,2.9732,1.9858)	(4.0167,2.9771,1.9863)	(4.0158,2.9834,1.9864)	(4.0058,2.9983,2.0019)
	$\hat{\phi}$	-1690.86	-1702.73	-1683.31	-1686.41
	φ	-1691.07	-1702.57	-1682.3	-1686.29
	Def	0.014934	0.013949^{s}	0.014442^{p}	0.010786^{ps}
2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0026,2.9950,2.0012)	(4.0050,2.9943,1.9926)	(4.0083,2.9919,1.9959)	(4.0077,2.9900,1.9972)
	$\hat{\phi}$	-4205.6	-4202.95	-4240.18	-4218.29
	φ	-4205.83	-4202.65	-4240.07	-4218.52
	Def	0.004635^{p}	0.003908^{ps}	0.005183	0.00396^{s}

p:Koloni boyutu sabit iken arama uzayına göre en iyi sonuçları gösterir.
s:Arama uzayı sabit iken koloni boyutuna göre en iyi sonuçları gösterir.

Çizelge 5. 3. Parametreleri $(\gamma,\eta,\beta)=(5,2,3)$ olan 3-p Weibull dağılımı için simülasyon sonucunda LABC algoritması ile elde edilen ortalama parametre tahmin sonuçları, $\hat{\varphi}$, φ ve Def değerleri

n	Arama Uzayı	[0-10]		[0-100]	
	Koloni Boyutu	50	250	50	250
100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9324,2.0598,3.1308)	(4.9750,2.0101,3.1444)	(4.9218, 2.0874,3.2245)	(5.0195,1.9678,3.023319)
	$\hat{\varphi}$	-98.5449	-86.8693	-91.807	-108.28
	φ	-98.6378	-87.72	-91.7422	-107.651
	Def	0.639852^{ps}	0.789843	1.105848	0.684895^{ps}
500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9015,2.1015,3.1881)	(4.9986,2.0015,3.0127)	(4.7706,2.2417,3.39013)	(5.0204,1.981,2.9716)
	$\hat{\varphi}$	-494.143	-468.714	-478.059	-494.475
	φ	-494.008	-468.859	-475.82	-494.553
	Def	0.315107^{p}	0.117313^{s}	1.136039	0.103392^{ps}
1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.935,2.0680,3.1083)	(4.9965,2.0006,3.0041)	(4.8731,2.1298,3.2302)	(4.9940,2.0062,2.9999)
	$\hat{\varphi}$	-1021.12	-976.483	-959.739	-994.444
	φ	-1021.01	-976.439	-958.502	-994.711
	Def	0.194966^{p}	0.044625^{ps}	0.619861	0.060351^{s}
2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9363,2.0693,3.1075)	(5.0028,1.9942,2.9921)	(4.9376,2.0704,3.1162)	(4.9957,2.0024,3.0048)
	$\hat{\varphi}$	-2447.69	-2488.27	-2426.86	-2493.4
	φ	-2447.44	-2488.28	-2428.67	-2493.65
	Def	0.131841^{p}	0.022886^{ps}	0.247112	0.037897^{s}

p:Koloni boyutu sabit iken arama uzayına göre en iyi sonuçları gösterir.
s:Arama uzayı sabit iken koloni boyutuna göre en iyi sonuçları gösterir.

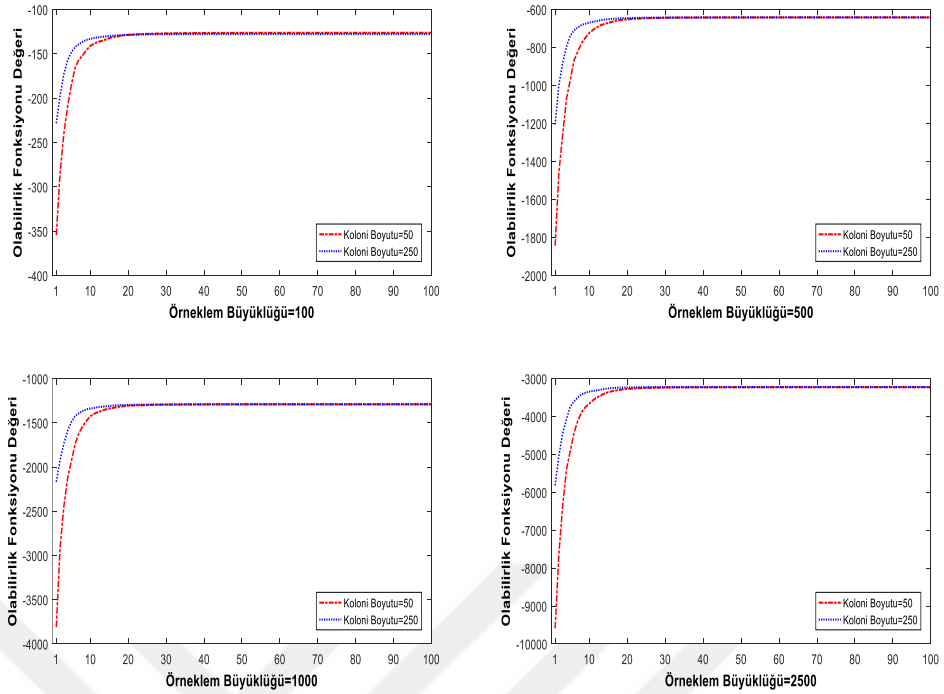
Örneklem büyüklüğü arttıkça, tüm durumlarda Def değeri düşmekte ve $\hat{\varphi}$ değeri φ değerine yakınsamaktadır. Başka bir ifadeyle, örneklem boyutu arttıkça, Def kriterine göre daha iyi parametre tahmini değerleri elde edilmektedir.

$[0,10]$ arama uzayında, $n=100$ için koloni boyutu 50 olan LABC algoritması daha iyi parametre tahmini değerleri verirken, $n=500,1000,2500$ için koloni boyutu 250 olan LABC algoritması Def kriterine göre daha iyi parametre tahmini değerleri vermiştir. $[0,100]$ arama uzayında ise, tüm örneklem büyüklükleri için koloni boyutu 250 olan LABC algoritması Def kriterine göre daha iyi parametre tahmin değerleri vermiştir. Kısacası, LABC algoritmasında koloni boyutunun artması daha iyi parametre tahmin değerlerinin elde edilmesine olanak sağlamıştır. Fakat, koloni boyutu arttıkça hesaplama zamanı da artacağından koloni boyutunun çok yüksek olması istenmeyen bir durumdur. Bundan dolayı bu çalışmada, farklı koloni boyutlarında parametre tahminlerinin Def değerleri arasında istatistiksel olarak önemli bir fark olup olmadığını test etmek için Mann-Whitney *U* testi kullanılmıştır. $[0,10]$ ve $[0,100]$ arama uzayları için Mann-Whitney *U* testi sonucunda p-değerleri 0.590 ve 0.478 olarak elde edilmiştir. Her iki arama uzayı için farklı koloni boyutlarında parametre tahminlerinin Def değerleri arasında istatistiksel olarak anlamlı bir fark olmadığı %95 güven düzeyinde tespit edilmiştir. Böylece, 3-p Weibull dağılımın parametrelerinin tahmini

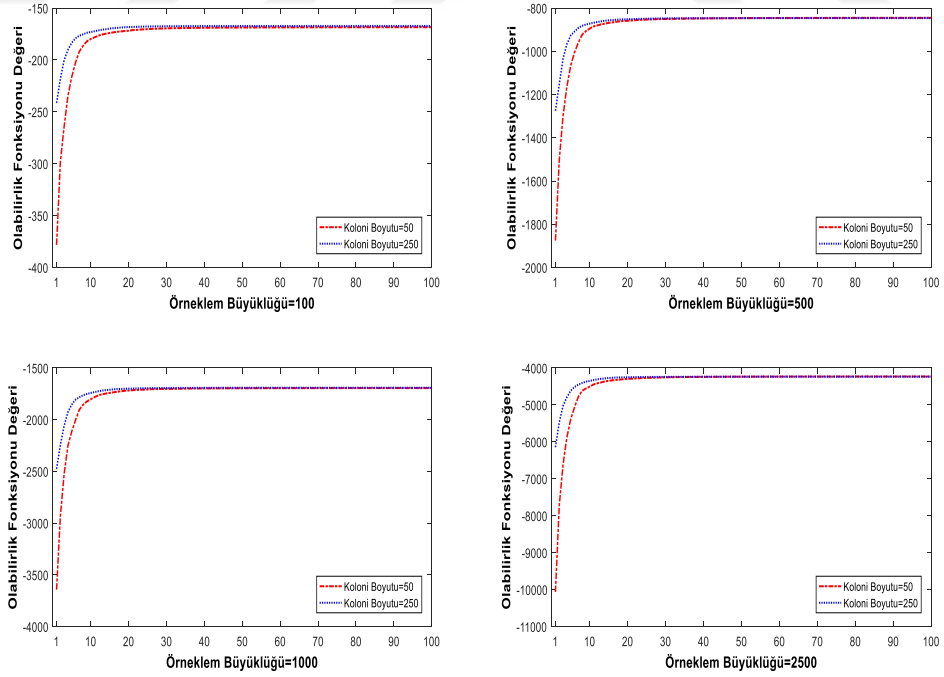
için LABC algoritmasında, koloni boyutu olarak 50 alınmasının hem çözüm kalitesi hem de çözüm zamanı açısından yeterli olduğu söylenebilir.

Arama uzayına göre algoritmanın performansının etkilenmemesi, algoritmanın sağlamlılığı açısından oldukça önemlidir. Başlangıçta tahmin edilecek parametre değerlerinin hangi aralıkta değiştiği bilinmediğinden arama uzayı rasgele belirlenmektedir. Bundan dolayı, algoritmanın arama uzayından etkilenmesi istenmeyen bir durumdur. Bu çalışmada, farklı arama uzaylarında parametre tahminlerinin Def değerleri arasında istatistiksel olarak önemli bir fark olup olmadığını test etmek için Mann-Whitney *U* testi kullanılmıştır. Koloni boyutu 50 ve 100 olan LABC algoritmaları için Man-Whitney *U* testi sonucunda p-değerleri 0.590 ve 1.00 olarak elde edilmiştir. Koloni boyutu 50 ve 100 için farklı arama uzaylarında parametre tahminlerinin Def değerleri arasında istatistiksel olarak önemli bir fark olmadığı %95 güven düzeyinde tespit edilmiştir. Tahminlerin kalitesinin arama uzayına bağlı olmaması, 3-p Weibull dağılımının parametrelerinin ML tahmininde LABC algoritmasının sağlamlığının bir göstergesidir.

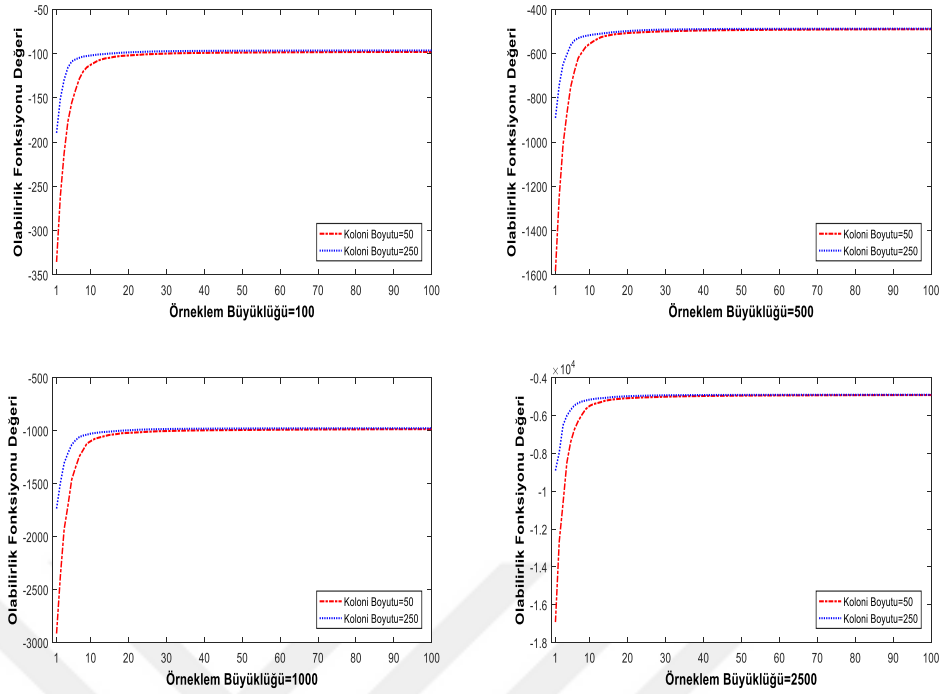
Şekil 5.1-5.3'te parametre değerleri $\theta = (2,2,2)$, $\theta = (4,3,2)$ ve $\theta = (5,2,3)$ için ele alınan tüm koloni boyutları ve örneklem boyutlarına göre $[0,100]$ arama uzayında 1000 iterasyon boyunca LABC algoritması ile elde edilen olabilirlik fonksiyonu değerleri gösterilmiştir.



Şekil 5.1. $(\gamma,\eta,\beta)=(2,2,2)$ parametresine sahip 3-p Weibull dağılımı için koloni boyutu ve örneklem boyutuna göre [0,100] arama uzayı için LABC algoritması ile elde edilen olabilirlik fonksiyonu değerleri



Şekil 5.2. $(\gamma,\eta,\beta)=(4,3,2)$ parametresine sahip 3-p Weibull dağılımı için koloni boyutu ve örneklem boyutuna göre [0,100] arama uzayı için LABC algoritması ile elde edilen olabilirlik fonksiyonu değerleri



Şekil 5.3. $(\gamma, \eta, \beta) = (5, 2, 3)$ parametresine sahip 3-p Weibull dağılımı için koloni boyutu ve örneklem boyutuna göre [0,100] arama uzayı için LABC algoritması ile elde edilen olabilirlik fonksiyonu değerleri

Şekil 5.1-5.3'ten, tüm durumlar için olabilirlik fonksiyonunun tahmin değerleri en fazla 20 iterasyonda gerçek olabilirlik fonksiyonu değerine yakınsadığı görülmektedir. Ayrıca, tüm durumlar için koloni boyutu 250 olan LABC algoritmasının koloni boyutu 50 olan LABC algoritmasına göre olabilirlik fonksiyonunun gerçek değerine daha hızlı yakınsamakta olduğu görülmektedir. Koloni boyutu 250 olan LABC algoritmasında yakınsama en fazla 20 iterasyonda gerçekleşmektedir. Sonuç olarak, verilen durumlar altında 3-p Weibull dağılımının parametrelerinin tahmini için koloni boyutu 50 olan LABC algoritmasında 20 iterasyonun, koloni boyutu 250 olan LABC algoritmasında ise 10 iterasyonun yeterli olduğu söylenebilir.

5.1.3. LABC algoritmasının diğer metasezgisel yöntemlerle karşılaştırmalı analizi

Bu alt kesimde, LABC algoritmasının tahmin sonuçları, simülasyon çalışması ile elde edilen ABC, GA ve LPSO algoritması tahmin sonuçları, Örkcü ve ark. (2015b)'nın çalışmalarında elde edilen PSO algoritması tahmin sonuçları, Örkcü ve ark. (2015a)'nın

çalışmalarında elde edilen SA algoritması tahmin sonuçları ile karşılaştırılmıştır. Bu karşılaştırmayı yapmak için gerçek parametre değerleri ile tahmini parametre değerleri arasındaki farkların toplamı olan E^2 değeri kullanılmıştır. E^2 değeri,

$$E^2 = \sum_{i=1}^{n} \left(\hat{\theta}_i - \theta_i \right)^2 \tag{5.6}$$

eşitliğinden hesaplanmaktadır (Abbasi ve ark., 2011).

Karşılaştırma sonuçları, Çizelge 5.4 ve 5.5'te verilmiştir. $[0,10]$ arama uzayı için tüm algoritmaların tahmin sonuçları mevcut iken, Örkcü ve ark. (2015a) ve Abbasi ve ark. (2006) çalışmalarında $[0,100]$ arama uzayını kullanmadığından $[0,100]$ arama uzayı için LABC, standart ABC, PSO, LPSO ve GA tahmin sonuçları mevcuttur.

Çizelge 5.4 ve 5.5'te her bir durum için en küçük E^2 değerleri, yani en iyi tahmin değerine sahip olanlar koyu renkte yazılmıştır. Sonuçlardan görüldüğü üzere, her iki arama uzayında da LABC algoritması diğer algoritmalara göre genellikle daha iyi tahmin değerleri elde etmiştir. Ayrıca, ABC algoritması özellikle $[0,100]$ arama uzayında çok kötü performans sergilerken LABC algoritması iyi bir performans sergilemiştir. Dolayısıyla, Levy uçuşlarının ABC algoritmasının performansı üzerinde çok büyük etkisi olduğu sonucuna ulaşılmıştır.

Çizelge 5.4. 3-p Weibull dağılımının çeşitli parametreleri için $[0,10]$ arama uzayında metasezgisel yöntemlerden simülasyon sonucunda elde edilen ML tahminleri ve E^2 değerleri

Gerçek Parametre Değerleri	Örneklem boyutu	Yöntem	LABC	ABC	PSO	LPSO	DE	SA	GA
(2,2,2)	100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0458,1.9403,1.9677)	(1.9392,2.1047, 2.1534)	(2.0215,1.69800,1.9934)	(2.0131,1.9772,2.0023)	(2.052,1.684,2.040)	(1.839,1.636,2.252)	(1.9346,2.0739,2.1357)
		E^2	0.0067	0.031026	0.0009	**0.0006**	0.1041	0.221921	0.0281
	500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0189,1.9722,1.9712)	(1.8808,2.1437,2.1523)	(2.0028,1.9994,2.0048)	(1.9858,2.0115,2.0187)	(1.984,2.016,2.237)	(1.833,1.931,2.072)	(1.9298,2.0843,2.1132)
		E^2	0.0019	0.058071	3.12E-05	0.0006	0.0566	0.037979	0.0248
	1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0095,1.9929,1.9842)	(1.8965,2.1423,2.1642)	(2.0032,1.9987,1.9966)	(1.9735,2.0280,2.0294)	(2.031,2.001,2.087)	(1.951,1.905,2.035)	(1.9356,2.0796,2.0990)
		E^2	0.0003	0.057909	**2.35E-05**	0.0023	0.0085	0.012651	0.0203
	2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0054,1.9944,1.9943)	(1.9281,2.0865,2.1079)	(1.9928,2.0104,2.0138)	(1.9697,2.0326,2.0346)	(2.027,1.954,2.010)	(2.030,2.034,1.999)	(1.9263,2.0908,2.1115)
		E^2	**9.15E-05**	0.024296	0.0003	0.0032	0.0029	0.002057	0.0261
(4,3,2)	100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.1004,2.8413,1.9073)	(3.9066,3.1461,2.1492)	(4.0306,2.9746,1.9927)	(4.0787,2.8962,1.951)	(4.108,2.797,2.228)	(4.070,3.006,1.897)	(3.7561,3.2749,2.2555)
		E^2	0.0438	0.052344	**0.0016**	0.0193	0.1048	0.015545	0.2003
	500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0261,2.9746,1.9877)	(3.8604,3.1490,2.1298)	(3.9705,3.0368,2.0373)	(4.0183,2.9754,1.9851)	(4.030,2.856,2.070)	(4.145,2.917,1.826)	(3.8120,3.2232,2.1780)
		E^2	0.0014	0.058557	0.0036	**0.0012**	0.0265	0.05819	0.1169
	1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0166,2.9732,1.9858)	(3.9125,3.0985,2.0850)	(3.9644,3.05,2.0436)	(4.0093,2.9882,1.9962)	(4.003,2.893,2.043)	(3.930,3.216,2.094)	(3.7847,3.2522,2.1992)
		E^2	0.0011	0.024589	0.0056	**0.0002**	0.013307	0.060392	0.1497
	2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0026,2.9950,2.0012)	(3.9356,3.0647,2.0553)	(3.9572,3.0604,2.0505)	(4.0011,2.9988,1.9987)	(4.038,2.895,1.974)	(3.971,3.091,2.041)	(3.7715,3.2689,2.2066)
		E^2	3.32E-05	0.011406	0.0080	**4.40E-06**	0.013145	0.010803	0.1672
(5,2,3)	100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9324,2.0598,3.1308)	(4.7419,2.2723,3.5024)	(4.3135,2.7109,4.2463)	(4.769,2.2335,3.4731)	(5.072,1.873,3.395)	(5.228,1.760,2.609)	(3.6268,3.4167,5.3725)
		E^2	**0.0253**	0.393189	2.5299	0.3314	0.177338	0.262465	9.5218
	500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9015,2.1015,3.1881)	(4.6677,2.3563,3.5924)	(4.262,2.7743,4.2945)	(4.8567,2.1498,3.2632)	(4.939,2.059,3.509)	(5.040,1.997,2.946)	(3.6091,3.4375,5.3200)
		E^2	**0.0554**	0.588394	2.8199	0.1122	0.266283	0.004525	9.3834
	1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.935,2.0680,3.1083)	(4.7066,2.3215,3.5495)	(4.2547,2.7823,4.3117)	(4.8745,2.1322,3.2338)	(5.054,1.937,3.016)	(4.954,1.981,3.125)	(3.5786,3.4670,5.3700
		E^2	**0.0206**	0.49149	2.8880	0.0879	0.007141	0.018102	9.7893
	2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9363,2.0693,3.1075)	(4.7631,2.2518,3.4095)	(4.1788,2.8608,4.4402)	(4.8584,2.1481,3.2555)	(5.016,1.953,3.028)	(5.040,1.915,2.973)	(3.6124,3.4345,5.3058)
		E^2	**0.0204**	0.287214	3.4895	0.1073	0.003249	0.009554	9.2997

Çizelge 5.5. 3-p Weibull dağılımının çeşitli parametreleri için $[0,100]$ arama uzayında metasezgisel yöntemlerden simülasyon sonucunda elde edilen ML tahminleri ve E^2 değerleri

Gerçek Parametre Değerleri	Örneklem boyutu	Yöntem	LABC	ABC	PSO	LPSO	GA
(2,2,2)	100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0167,1.9655,2.0353)	(0.9989,3.2503,3.2154)	(1.8651,2.1651,2.2049)	(1.4677,2.5850,2.5258)	(1.8253,2.4947,2.2380)
		E^2	**0.0027**	4.042541	0.08744	0.9020	0.3318
	500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0172,1.9780,1.9912)	(1.0107,3.1683,3.0984)	(1.8606,2.1701,2.2058)	(1.3398,2.7427,2.6952)	(1.8076,2.4686,2.2334)
		E^2	**0.0008**	3.550284	0.09072	1.4709	0.3111
	1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0114,1.9892,1,9909)	(1.0043,3.2188,3.1857)	(1.8587,2.1723,2.2061)	(1.3847,2.6721,2.6615)	(1.7875,2.5589,2.2523)
		E^2	**0.0003**	3.882922	0.09213	1.2679	0.4212
	2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(2.0040,1.9956,1.9949)	(0.9770,3.2839,3.1719)	(1.8479,2.1858,2.2200)	(1.3651,2.7114,2.6894)	(1.7749,2.6075,2.2651)
		E^2	**6.05E-05**	4.065991	0.106056	1.3846	0.4900
(4,3,2)	100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0500,2.9337,1.9828)	(2.2784,4.9833,3.2767)	(3.6587,3.4054,2.3293)	(2.6570,4.4264,2.9297)	(3.0915,4.1754,2.7916)
		E^2	**0.0071**	8.527339	0.389273	4.7026	2.8335
	500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0316,2.9713,1.9822)	(2.2643,5.0242,3.2508)	(3.631,3.4308,2.3319)	(2.6859,4.3910,2.9305)	(3.0957,4.3168,2.7204)
		E^2	**0.0021**	8.674599	0.431907	4.5276	3.0707
	1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0158,2.9834,1.9864)	(1.8002,5.5245,3.6364)	(3.6866,3.3693,2.2804)	(2.6044,4.4889,2.9947)	(3.0836,4.2679,2.7302)
		E^2	**0.0007**	13.89005	0.313226	5.1541	2.9804
	2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.0083,2.9919,1.9959)	(1.9737,5.3019,3.5188)	(3.6824,3.3695,2.2753)	(2.5434,4.5604,3.0387)	(3.0425,4.3297,2.7629)
		E^2	**0.0001**	11.7115	0.31319	5.6354	3.2669
(5,2,3)	100	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9218,2.0874,3.2245)	(2.3993,4.6990,7.2241)	(3.7771,3.2688,5.1523)	(2.6785,4.4112,6.7826)	(3.1172,4.1934,6.165)
		E^2	**0.0642**	31.89201	7.737733	25.5111	18.3765
	500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.7706,2.2417,3.3901)	(2.3010,4.7551,7.5525)	(3,8551,3.1853,4.9507)	(2.8377,4.2438,6.4375)	(3.0978,4.1858,6.0730)
		E^2	**0.2632**	35.60031	6.520963	21.5264	17.8396
	1000	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.8731,2.1298,3.2302)	(1.8924,5.2379,7.8794)	(3.7893,3.2562,5.0613)	(2.8186,4.2590,6.4624)	(2.9783,4.3453,6.2588)
		E^2	**0.0859**	43.94941	7.292791	21.8499	20.2072
	2500	$(\hat{\gamma},\hat{\eta},\hat{\beta})$	(4.9376,2.0704,3.1162)	(2.0254,5.0883,7.8796)	(3.9578,3.0742,4.7265)	(2.7157,4.3294,6.6179)	(2.9522,4.3196,6.2959)
		E^2	**0.0351**	42.19686	5.220889	23.7328	20.4368

5.1.4. Gerçek veri uygulaması

Bu alt kesimde, önerilen LABC yönteminin performansı gerçek veri uygulaması üzerinden araştırılmıştır. Bu amaçla, 24 mekanik parçanın başarısızlık zamanlarını içeren veri kümesi ele alınmıştır ve Çizelge 5.6'da verilmiştir (Murthy ve ark., 2004; Teimouri ve Gupta, 2013):

Çizelge 5.6. 24 mekanik parçanın başarısızlık zamanı veri kümesi

30.94	18.51	16.62	51.56	22.85	22.38	19.08	49.56	17.12	10.67	25.43	10.24
27.47	14.70	14.10	29.93	27.98	36.02	19.40	14.97	22.57	12.26	18.14	18.84

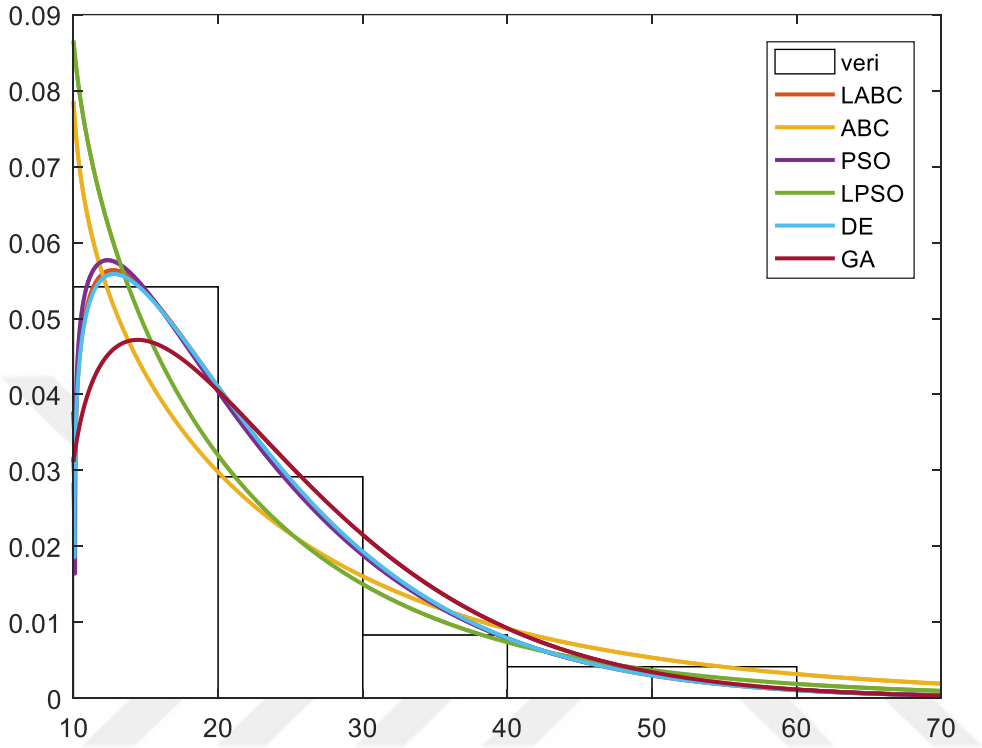
Ele alınan veri kümesinin dağılımının Weibull dağılımından gelip gelmediğini test etmek için Kolmogorov-Smirnov (K-S) testi kullanılmıştır. Ele alınan metasezgisel yöntemlerden elde edilen parametre tahmin değerleri, K-S test istatistiği değerleri ve buna ilişkin p değerleri ile $\ln L$, *AIC* ve *BIC* kriteri değerleri, Çizelge 5.7'de verilmiştir.

Çizelge 5.7. 24 mekanik parçanın başarısızlık zamanı veri kümesi için parametre tahmini değerleri, K-S testine ilişkin değerler, $\ln L$, AIC ve BIC değerleri

Yöntem	$\hat{\gamma}$	$\hat{\eta}$	$\hat{\beta}$	**K-S**	**p-değeri**	$\ln L$	**AIC**	**BIC**
LABC	10.0960	13.5523	1.1715	0.096	0.9798	**-84.8887**	175.7774	188.8457
ABC	9.7053	16.5450	0.9027	0.1607	0.5646	-85.2455	176.4910	189.5593
PSO	10.1104	13.4418	1.1474	0.1028	0.9613	-84.8938	175.7876	188.8559
LPSO	9.5232	12.9797	0.9280	0.1912	0.3440	-87.1013	180.2026	193.2709
DE	10.0661	13.6261	1.1782	**0.094**	0.9831	-84.8908	175.7816	188,8499
SA	9.1453	30.7047	1.1806	0.3881	0.0014	-92.9171	191.8342	204.9025
GA	9.3877	15.5936	1.2996	0.1115	0.9261	-85.2772	176.5544	189,6227

Çizelge 5.7'den, SA haricindeki tüm metasezgisel yöntemlere ilişkin p-değerleri $\alpha = 0.05$'ten büyük olduğundan bu veri kümesinin dağılımı ele alınan yöntemlerden elde edilen parametre değerleri ile 3-p Weibull dağılımına uygundur. SA ile elde edilen parametre değerleri K-S kriteri açısından bu veri kümesini modellemede başarısız olmuştur. LABC algoritması sonuçları en büyük $\ln L$ ve en küçük *AIC* ve *BIC* değerlerine sahip olduğundan, LABC'nin bu gerçek veri kümesi için en iyi parametre tahmini değerlerini verdiği görülmüştür.

Şekil 5.4'te, veriye ilişkin histogram ve ele alınan metasezgisel yöntemlere ilişkin parametre değerleri kullanılarak çizilen olasılık yoğunluk fonksiyonları verilmiştir. Çizelge 5.7'de verilen sonuçların Şekil 5.4 ile desteklendiği görülmüştür.



Şekil 5.4. Mekanik parçaların başarısızlık zamanları veri kümesi için histogram ve uyumlu oyf'ler

5.2. 3-p Gamma Dağılımının Parametrelerinin Tahmininde Metasezgisel Yöntemlerin Karşılaştırılması

Bu kesimde, 3-p Gamma dağılımının parametrelerinin ML yöntemi ile tahminine yer verilmiştir. 3-p Gamma dağılımının ML parametre tahmini için metasezgisel yöntemlerden SA, DE, PSO, ABC ve GA kullanılmıştır. Ayrıca, metasezgisel yöntemleri klasik yöntemler ile karşılaştırmak için doğrusal olmayan optimizasyon yöntemlerinden sıralı karesel programlama ve iç nokta algoritmaları kullanılmıştır. Algoritmalar, Monte-Carlo simülasyon çalışmaları ile algoritmalar çözüm kalitesi ve çözüm zamanı açısından karşılaştırılmıştır. Metasezgisel yöntemler, MCDM yöntemlerinden birisi olan TOPSIS yöntemi kullanılarak çözüm zamanı, çözüm

kalitesi, kolaylık ve sağlamlılık kriterleri açısından sıralanmıştır. Son olarak, gerçek bir veri kümesi için uygulama sunulmuştur.

5.2.1. Gamma dağılımının parametrelerinin tahmini için ML yöntemi

$X_1, X_2, ..., X_n$'ler olasılık yoğunluk fonksiyonu Eşitlik (3.3)'te verilen $\text{Gamma}(\sigma, \beta, \mu)$ dağılımından gelen rasgele değişkenler olmak üzere olabilirlik (L) ve log-olabilirlik ($\ln L$) fonksiyonları sırasıyla

$$L = \left(\frac{1}{\Gamma(\sigma)\beta^{\alpha}} \right)^n \prod_{i=1}^{n} (x_i - \mu)^{\sigma-1} \exp\left(\sum_{i=1}^{n} \left(-\frac{x_i - \mu}{\beta} \right) \right) \tag{5.7}$$

ve

$$\ln L = (\sigma - 1) \sum_{i=1}^{n} \ln(x_i - \mu) - \sum_{i=1}^{n} \left(-\frac{x_i - \mu}{\beta} \right) - n \ln \Gamma(\sigma) - n\sigma \ln \beta \tag{5.8}$$

olarak verilir. Eşitlik (5.8)'den olabilirlik denklemleri,

$$\frac{\partial \ln L}{\partial \sigma} = \sum_{i=1}^{n} \ln(x_i - \mu) - n\Psi(\sigma) - n \ln \beta = 0 \tag{5.9}$$

$$\frac{\partial \ln L}{\partial \beta} = \sum_{i=1}^{n} \left(\frac{x_i - \mu}{\beta^2} \right) - n \frac{\sigma}{\beta} = 0 \tag{5.10}$$

$$\frac{\partial \ln L}{\partial \mu} = -(\sigma - 1) \sum_{i=1}^{n} (x_i - \mu)^{-1} + \frac{n}{\beta} = 0 \tag{5.11}$$

olarak verilir. Eşitlik (5.9)'da, $\Psi(\sigma) = \dfrac{\partial \ln \Gamma(\sigma)}{\partial \sigma}$'dir. Olabilirlik denklemleri lineer olmayan fonksiyonlar içerdiğinden bu denklemlerin çözümü analitik olarak elde edilemez. Bundan dolayı, bu çalışmada metasezgisel yöntemlerin kullanılması önerilmiştir.

5.2.2. Simülasyon çalışması

Bu alt kesimde, 3-p Gamma dağılımının parametrelerinin ML tahmini için Monte-Carlo simülasyon sonuçları sunulmuştur. Önerilen metasezgisel yöntemlerin (SA, GA, PSO, DE, ABC) ve kısıtlı doğrusal olmayan optimizasyon yöntemlerinden SQP ve IPA'nın performansları çözüm kalitesi ve çözüm zamanı açısından karşılaştırılmıştır. Çözüm kalitesini değerlendirmek amacıyla Def kriteri kullanılmıştır.

Yöntemlerin performanslarının değerlendirilmesi için, Gamma dağılımı için gerçek parametre değerleri $\theta=(4,0.5,10)$, $\theta=(2.56,0.625,10)$, $\theta=(1.778,0.75,10)$, $\theta=(1,1,10)$ ve $\theta=(0.64,1.25,10)$ olarak alınmıştır. Ayrıca, parametre tahmin sürecinde örneklem büyüklüğünün etkisini araştırmak amacıyla her bir parametre değeri için örneklem büyüklükleri olarak 50, 100, 250, 500 ve 1000 alınmıştır.

Bu çalışmada, metasezgisel yöntemlerde kullanılan parametre değerleri, çeşitli denemeler ve literatür araştırmaları sonucu belirlenmiş olup Çizelge 5.8'de verilmiştir. Tüm parametreler için arama uzayı $[0,10]$ olarak alınmış ve algoritmalar hızlı bir yakınsama gösterdiğinden maksimum iterasyon sayısı 100 olarak alınmıştır.

Çizelge 5.8. Metasezgisel algoritmaların parametreleri

Algoritmalar	Parametreler
SA	$s=10$, $T_0=0.1$, $\tau=0.99$, nMove$=5$
GA	NP=50, $p_m=0.3$, $p_c=0.8$, $m=0.1$
PSO	NP=50, $w=1$, $c_1=c_2=2$
DE	NP=50, $F\in(0,1]$, $C_r=0.2$
ABC	NP=50, $l=5$

Tahmin stratejisinin güvenirliğini arttırmak için metasezgisel algoritmalar 10.000 kez çalıştırılmış ve elde edilen parametre tahminlerine ilişkin ortalama simülasyon sonuçları MSE ve Def değerleri ile çalıştırma zamanları saniye olarak Çizelge 5.9-5.13'de verilmiştir. En iyi MSE, Def ve çalıştırma zamanı değerleri koyu renk ile yazılmıştır.

Çizelge 5.9. Parametreleri $\theta = (\sigma, \beta, \mu) = (4, 0.5, 10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE, Def ve çalıştırma zamanı değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$			
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**	**Çalıştırma Zamanı**
25	**SA**	3.6050	12.5249	1.1801	1.2051	10.1921	0.3717	14.1016	12.6098
	GA	3.3577	**4.6881**	0.8050	**0.4559**	10.1998	**0.2081**	**5.3521**	1.0072
	PSO	4.0844	9.4128	0.8088	0.5196	10.1274	0.3027	10.2351	**0.7844**
	DE	2.7104	7.1858	1.2074	1.1902	10.3162	0.2940	8.6700	1.0093
	ABC	3.1595	10.6467	1.4434	1.9951	10.1706	0.4288	13.0706	1.8507
	SQP	3.2498	11.2167	1.6304	4.419	10.2388	0.3471	15.983	0.3102
	IPA	3.3906	11.3251	1.7014	6.3605	10.2248	0.3506	18.0362	0.5557
50	**SA**	4.2584	6.9467	0.5863	0.0878	10.0514	0.1998	7.2343	12.5680
	GA	3.7019	**2.8637**	0.5902	**0.0552**	10.1008	**0.1186**	**3.0375**	1.1366
	PSO	4.3519	5.8955	0.5619	0.0700	10.0275	0.1739	6.1394	**0.8166**
	DE	3.4678	3.0755	0.6217	0.0826	10.1674	0.1345	3.2926	1.0498
	ABC	4.8904	8.3538	0.6508	0.2356	9.7964	0.3605	8.9499	2.1556
	SQP	3.1934	7.9665	1.5812	4.5727	10.1731	0.2069	12.7462	0.1877
	IPA	3.3210	8.1547	1.5536	6.0351	10.1581	0.2118	14.4017	0.6730
100	**SA**	4.1258	3.4340	0.5307	0.0197	10.0304	0.0984	3.5521	12.5506
	GA	3.7987	2.1573	0.5616	0.0258	10.0543	0.0811	2.2642	1.1233
	PSO	4.2946	3.0910	0.5138	**0.0181**	9.9965	0.0932	3.2023	**0.8551**
	DE	3.5967	**1.6423**	0.5606	0.0185	10.1129	**0.0677**	**1.7285**	1.1223
	ABC	5.3686	7.7421	0.5182	0.0509	9.6773	0.3427	8.1357	2.2840
	SQP	3.2037	5.406	1.5776	5.1626	10.1233	0.1152	10.6839	0.1791
	IPA	3.5816	4.4791	1.1149	4.0612	10.0914	0.1119	8.6521	0.7324
250	**SA**	4.0361	1.1110	0.5117	**0.0069**	10.0137	0.0348	1.1527	14.8486
	GA	3.8364	1.5502	0.5514	0.0188	10.0256	0.0515	1.6205	1.2283
	PSO	4.1988	1.2402	0.4998	**0.0069**	9.9848	0.0386	1.2858	**0.8384**
	DE	3.7187	**0.7556**	0.5352	0.0071	10.0674	**0.0292**	**0.7919**	1.2216
	ABC	5.4813	7.5010	0.4960	0.0306	9.6431	0.3353	7.8668	2.2186
	SQP	3.4136	2.781	1.3339	4.4518	10.069	0.0439	7.2767	0.1832
	IPA	3.6893	1.7813	0.8761	2.9037	10.0572	0.0434	4.7283	0.6512
500	**SA**	3.9864	0.4538	0.5072	**0.0031**	10.0129	**0.0152**	0.4721	17.1158
	GA	3.8688	1.3913	0.5462	0.0158	10.0107	0.0457	1.4528	1.3915
	PSO	4.1895	0.6707	0.4934	0.0036	9.9769	0.0214	0.6956	**1.0899**
	DE	3.7546	**0.4126**	0.5265	0.0037	10.0536	0.0157	**0.4320**	1.3984
	ABC	5.2944	6.9967	0.5016	0.0268	9.6910	0.2909	7.3143	2.3078
	SQP	3.4244	2.214	1.3457	4.6605	10.0515	0.0226	6.8972	0.2037
	IPA	3.7964	0.7708	0.6543	1.1837	10.0390	0.0203	1.9748	0.8004
1000	**SA**	3.9795	**0.1962**	0.5043	**0.0015**	10.0087	**0.0067**	**0.2043**	21.0536
	GA	3.9076	1.2942	0.5423	0.0145	9.9966	0.0427	1.3514	1.6647
	PSO	4.1775	0.4177	0.4911	0.0021	9.9742	0.0131	0.4329	**1.3248**
	DE	3.7908	0.2397	0.5209	0.0021	10.0424	0.0088	0.2507	1.6241
	ABC	5.0017	5.9974	0.5110	0.0237	9.7600	0.2375	6.2587	2.5252
	SQP	3.5812	1.5204	1.1232	3.5165	10.031	0.0112	5.0482	0.2131
	IPA	3.8684	0.3433	0.5565	0.3924	10.0259	0.0099	0.7456	0.7336

Çizelge 5.10. Parametreleri $\theta = (\sigma, \beta, \mu) = (2.56, 0.625, 10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE, Def ve çalıştırma zamanı değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$			
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**	**Çalıştırma Zamanı**
25	**SA**	2.1720	7.5901	1.4700	1.3486	10.1206	0.1731	9.1118	7.5784
	GA	2.5007	**3.3987**	0.9648	**0.5480**	10.0621	**0.1212**	**4.0679**	0.8187
	PSO	2.7555	6.2907	1.0209	0.6702	10.0485	0.1554	7.1163	**0.8009**
	DE	1.7099	3.8043	1.4799	1.3309	10.1812	0.1180	5.2533	1.0541
	ABC	1.9905	6.7478	1.7315	2.1788	10.1039	0.2097	9.1363	2.2390
	SQP	2.1516	6.3629	1.80928	4.3761	10.1275	0.1555	10.8947	0.1955
	IPA	2.1961	6.4725	1.9717	6.7922	10.1205	0.1544	13.4191	0.6107
50	**SA**	2.5539	2.8254	0.7678	0.1513	10.0529	0.0718	3.0485	8.1300
	GA	2.7438	2.1311	0.6890	**0.0812**	9.9999	0.0778	2.2901	1.0869
	PSO	2.7513	2.7048	0.7152	0.1103	10.0179	0.0725	2.8876	**0.8525**
	DE	2.3160	**1.3711**	0.7810	0.1441	10.0863	**0.0480**	**1.5632**	1.0970
	ABC	3.3310	7.0879	0.8872	0.4186	9.8320	0.2531	7.7596	2.3120
	SQP	1.9959	3.0609	1.734	4.5042	10.1009	0.0667	7.6318	0.1811
	IPA	2.0570	3.0401	1.6571	5.6268	10.0942	0.0692	8.7361	0.5672
100	**SA**	2.5238	0.8173	0.6647	0.0239	10.0365	0.0253	0.8665	13.2867
	GA	2.8045	1.6087	0.6422	0.0276	9.9705	0.0607	1.6970	1.1233
	PSO	2.6809	1.0447	0.6447	0.0251	10.0065	0.0320	1.1018	**0.8795**
	DE	2.4301	**0.5346**	0.6749	**0.0231**	10.0510	**0.0200**	**0.5777**	1.0979
	ABC	3.7878	6.8260	0.6663	0.1176	9.7233	0.2700	7.2136	2.4334
	SQP	1.9563	1.5235	1.6873	4.7996	10.0767	0.0256	6.3487	0.2135
	IPA	2.1801	1.2319	1.2547	3.8675	10.0647	0.0268	5.1263	0.6430
250	**SA**	2.5065	0.1809	0.6423	0.0072	10.0217	0.0061	0.1943	14.7119
	GA	2.8050	1.2905	0.6290	0.0161	9.9552	0.0498	1.3564	1.1600
	PSO	2.6382	0.3937	0.6265	0.0089	9.9981	0.0127	0.4153	**0.1006**
	DE	2.4709	0.1638	0.6483	0.0073	10.0274	0.0059	**0.1770**	1.2283
	ABC	3.6770	5.6510	0.6261	0.0584	9.7406	0.2230	5.9323	2.5921
	SQP	2.1133	0.923	1.5713	4.8359	10.0407	0.0081	5.7672	0.1746
	IPA	2.3271	0.4117	1.0121	2.8257	10.0376	0.0077	3.2451	0.7209
500	**SA**	2.5195	**0.0782**	0.6353	**0.0033**	10.0133	**0.0026**	**0.0842**	17.3935
	GA	2.8319	1.2201	0.6219	0.0129	9.9433	0.0500	1.2829	1.1686
	PSO	2.6416	0.2578	0.6205	0.0051	9.9922	0.0081	0.2710	**0.1170**
	DE	2.4979	0.0809	0.6395	0.0035	10.0167	0.0028	0.0872	1.3578
	ABC	3.2384	3.5731	0.6464	0.0433	9.8307	0.1408	3.7573	2.5476
	SQP	2.1365	0.8299	1.5938	5.1043	10.027	0.0038	5.938	0.1971
	IPA	2.4360	0.1568	0.7569	0.9585	10.0233	0.0033	1.1187	0.7291
1000	**SA**	2.5353	**0.0346**	0.6306	**0.0015**	10.0080	**0.0011**	**0.0372**	19.9183
	GA	2.8257	1.1094	0.6197	0.0107	9.9427	0.0443	1.1644	1.6153
	PSO	2.6561	0.2203	0.6162	0.0033	9.9862	0.0071	0.2307	**0.1299**
	DE	2.5143	0.0385	0.6347	0.0017	10.0114	0.0013	0.0414	1.5652
	ABC	2.9440	2.2817	0.6584	0.0333	9.8955	0.0896	2.4046	2.6287
	SQP	2.2084	0.6633	1.4364	4.3602	10.017	0.0017	5.0253	0.2004
	IPA	2.4890	0.0638	0.6737	0.3229	10.0142	0.0015	0.3882	0.6611

Çizelge 5.11. Parametreleri $\theta = (\sigma, \beta, \mu) = (1.778, 0.75, 10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE, Def ve çalıştırma zamanı değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$			
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**Çalıştırma Zamanı**
25	**SA**	1.2169	3.3012	1.6660	1.2385	10.0963	0.0633	4.6031	5.5191
	GA	1.8060	2.3949	1.1476	**0.6033**	10.0136	0.0752	**3.0734**	1.0010
	PSO	1.8172	3.5000	1.2374	0.7682	10.0313	0.0722	4.3404	**0.8030**
	DE	1.0518	**1.8715**	1.6726	1.2314	10.1170	**0.0437**	3.1465	1.0367
	ABC	1.1218	2.9718	1.9220	2.0505	10.0919	0.0727	5.0951	2.2336
	SQP	1.3303	2.8776	2.0192	4.4487	10.0880	0.0568	7.3832	0.1828
	IPA	1.3399	3.0002	2.2263	6.9822	10.0870	0.0575	10.0399	0.5512
50	**SA**	1.5649	0.8411	1.0054	0.2454	10.0502	0.0179	1.1044	6.1675
	GA	2.0361	1.8314	0.8410	**0.1439**	9.9657	0.0670	2.0422	1.0725
	PSO	1.7797	1.1584	0.9128	0.1801	10.0213	0.0261	1.3646	**0.8680**
	DE	1.5239	**0.6160**	1.0063	0.2390	10.0559	**0.0145**	**0.8694**	1.0629
	ABC	1.8356	3.4181	1.2357	0.6197	9.9571	0.0954	4.1332	2.2585
	SQP	1.2972	0.9626	1.8584	4.1730	10.0665	0.0163	5.1518	0.1836
	IPA	1.3113	0.9876	1.8000	5.2580	10.0663	0.0169	6.2625	0.5918
100	**SA**	1.6850	0.1694	0.8062	**0.0323**	10.0318	0.0048	0.2065	8.3841
	GA	2.0675	1.2917	0.7466	0.0393	9.9549	0.0526	1.3836	1.0836
	PSO	1.7906	0.3808	0.7910	0.0389	10.0129	0.0104	0.4301	**0.9052**
	DE	1.6697	**0.1562**	0.8122	0.0336	10.0333	**0.0046**	**0.1945**	1.1550
	ABC	2.0559	2.7207	0.9588	0.2203	9.9120	0.0883	3.0293	2.3672
	SQP	1.3435	0.5425	1.8413	4.7388	10.0441	0.0053	5.2866	0.1787
	IPA	1.4368	0.4053	1.4035	3.4921	10.0431	0.0053	3.9027	0.6107
250	**SA**	1.7212	**0.0469**	0.7738	**0.0085**	10.0169	**0.0012**	**0.0566**	10.0441
	GA	2.1008	1.2731	0.7249	0.0224	9.9402	0.0526	1.3480	1.2152
	PSO	1.8048	0.1808	0.7587	0.0118	10.0024	0.0052	0.1978	**1.0134**
	DE	1.7148	0.0491	0.7769	0.0091	10.0174	**0.0012**	0.0595	1.2084
	ABC	1.8660	1.0963	0.8713	0.0888	9.9522	0.0392	1.2243	2.5713
	SQP	1.4432	0.3889	1.7891	5.0602	10.0227	0.0015	5.4506	0.1906
	IPA	1.6334	0.1308	1.1149	2.5302	10.0206	0.0014	2.6624	0.5861
500	**SA**	1.7417	**0.0205**	0.7641	**0.0037**	10.0099	**0.0005**	**0.0247**	15.6388
	GA	2.1274	1.2778	0.7167	0.0188	9.9313	0.0545	1.3511	1.3907
	PSO	1.8151	0.1286	0.7493	0.0067	9.9974	0.0038	0.1391	**1.1789**
	DE	1.7352	0.0222	0.7671	0.0041	10.0104	**0.0005**	0.0268	1.4208
	ABC	1.7985	0.6199	0.8480	0.0602	9.9722	0.0229	0.7031	2.9035
	SQP	1.4510	0.3801	1.8667	5.6000	10.0144	0.0006	5.9807	0.1984
	IPA	1.7042	0.0496	0.8938	1.0548	10.0123	0.0005	1.1048	0.8221
1000	**SA**	1.7571	**0.0093**	0.7576	**0.0017**	10.0057	**0.0002**	**0.0112**	19.7226
	GA	2.1161	1.1832	0.7153	0.0165	9.9323	0.0504	1.2501	1.6143
	PSO	1.8251	0.1204	0.7448	0.0046	9.9939	0.0039	0.1289	**1.4486**
	DE	1.7509	0.0103	0.7602	0.0019	10.0063	**0.0002**	0.0124	1.6428
	ABC	1.8010	0.4230	0.8238	0.0458	9.9777	0.0160	0.4848	3.3278
	SQP	1.4926	0.3327	1.7609	5.1727	10.0086	0.0002	5.5057	0.2020
	IPA	1.7403	0.0175	0.7945	0.2932	10.0072	0.0002	0.3109	0.8258

Çizelge 5.12. Parametreleri $\theta = (\sigma, \beta, \mu) = (1,1,10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE, Def ve çalıştırma zamanı değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$			
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**	**Çalıştırma Zamanı**
25	**SA**	0.5988	0.3782	1.7379	0.7463	10.0364	0.0049	1.1295	6.1487
	GA	1.1018	1.4889	1.3692	**0.5094**	9.9655	0.0516	2.0499	0.9317
	PSO	0.8677	0.8354	1.5609	0.7004	10.0108	0.0133	1.5491	**0.7997**
	DE	0.5867	**0.2739**	1.7375	0.7439	10.0376	**0.0039**	**1.0217**	1.0754
	ABC	0.5812	0.3559	2.0319	1.6269	10.0362	0.0053	1.9881	1.0214
	SQP	0.6385	0.4143	2.1946	3.7488	10.0345	0.0049	4.1680	0.1951
	IPA	0.6227	0.4773	2.4453	6.4050	10.0336	0.0055	6.8878	0.5639
50	**SA**	0.7434	0.1127	1.3716	0.2226	10.0186	0.0009	0.3363	6.1915
	GA	1.1618	1.2099	1.1587	**0.1957**	9.9534	0.0478	1.4534	1.0698
	PSO	0.8636	0.2795	1.3009	0.2259	10.0061	0.0048	0.5102	**0.8669**
	DE	0.7423	**0.1001**	1.3709	0.2222	10.0187	**0.0008**	**0.3230**	1.0831
	ABC	0.7305	0.1967	1.5969	0.6567	10.0162	0.0021	0.8555	1.2136
	SQP	0.6943	0.1547	2.1700	4.0165	10.0191	0.0008	4.1720	0.1910
	IPA	0.6504	0.1852	2.1010	4.6665	10.0189	0.0008	4.8524	0.5676
100	**SA**	0.8521	0.0394	1.1850	0.0712	10.0095	**0.0002**	0.1107	6.5034
	GA	1.2070	1.0464	1.0468	0.0913	9.9495	0.0443	1.1820	1.0772
	PSO	0.9160	0.1437	1.1572	0.0794	10.0012	0.0030	0.2262	**0.9207**
	DE	0.8507	**0.0392**	1.1861	**0.0710**	10.0096	**0.0002**	**0.1104**	1.1183
	ABC	0.8262	0.0806	1.3774	0.3327	10.0080	0.0006	0.4139	1.5680
	SQP	0.7417	0.1156	2.1359	4.3693	10.0097	0.0002	4.4852	0.2008
	IPA	0.7424	0.1025	1.7992	3.6429	10.0099	0.0002	3.7456	0.5672
250	**SA**	0.9347	0.0115	1.0746	0.0185	10.0039	**0.0000**	0.0300	8.0523
	GA	1.2807	1.1432	0.9730	0.0546	9.9398	0.0505	1.2484	1.1483
	PSO	0.9824	0.0948	1.0547	0.0256	9.9970	0.0024	0.1228	**0.9780**
	DE	0.9328	**0.0112**	1.0760	**0.0182**	10.0039	**0.0000**	**0.0294**	1.1146
	ABC	0.9162	0.0542	1.2356	0.1936	10.0021	0.0003	0.2481	1.5694
	SQP	0.7748	0.0967	2.2527	5.3845	10.0039	0.0000	5.4812	0.1996
	IPA	0.8643	0.0404	1.4606	2.3702	10.0039	0.0000	2.4107	0.6184
500	**SA**	0.9671	0.0046	1.0363	0.0072	10.0020	**0.0000**	0.0118	11.9227
	GA	1.2971	1.0705	0.9463	0.0456	9.9391	0.0485	1.1646	1.2685
	PSO	1.0071	0.0799	1.0226	0.0140	9.9955	0.0023	0.0962	1.1618
	DE	0.9658	**0.0044**	1.0372	**0.0071**	10.0020	**0.0000**	**0.0116**	**1.1578**
	ABC	0.9534	0.0470	1.1833	0.1521	9.9999	0.0004	0.1995	1.8515
	SQP	0.7993	0.0888	2.2438	5.4225	10.0020	0.0000	5.5112	0.2209
	IPA	0.9308	0.0164	1.2235	1.2437	10.0020	0.0000	1.2601	0.7537
1000	**SA**	0.9821	**0.0019**	1.0184	0.0031	10.0010	**0.0000**	0.0050	16.6259
	GA	1.3004	1.0254	0.9345	0.0417	9.9401	0.0450	1.1121	1.3761
	PSO	1.0189	0.0643	1.0055	0.0093	9.9952	0.0019	0.0755	**1.3333**
	DE	0.9814	**0.0019**	1.0189	**0.0030**	10.0010	**0.0000**	**0.0049**	1.3658
	ABC	0.9666	0.0439	1.1610	0.1370	9.9993	0.0002	0.1811	2.6882
	SQP	0.8259	0.0812	2.1643	5.1512	10.0010	0.0000	5.2323	0.2488
	IPA	0.9666	0.0056	1.0820	0.4130	10.0010	0.0000	0.4186	0.7857

Çizelge 5.13. Parametreleri $\theta = (\sigma, \beta, \mu) = (0.64, 1.25, 10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE, Def ve çalıştırma zamanı değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$			
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**	**Çalıştırma Zamanı**
25	**SA**	0.4725	0.0404	1.7167	0.4728	10.0088	**0.0003**	0.5135	7.6273
	GA	0.8282	1.1742	1.4611	0.4342	9.9544	0.0416	1.6500	0.9449
	PSO	0.6050	0.3324	1.6355	0.5118	9.9959	0.0046	0.8488	**0.6951**
	DE	0.4715	**0.0344**	1.7164	**0.4723**	10.0089	**0.0003**	**0.5070**	0.7928
	ABC	0.4607	0.0519	2.1172	1.4948	10.0088	0.0004	1.5471	1.1183
	SQP	0.4853	0.0775	2.2264	3.4982	10.0088	0.0004	3.5761	0.1898
	IPA	0.4579	0.0748	2.3715	5.6673	10.0087	0.0003	5.7424	0.5303
50	**SA**	0.5496	**0.0130**	1.4743	**0.1545**	10.0031	**0.0000**	**0.1675**	13.2066
	GA	0.8580	0.9997	1.3106	0.2169	9.9500	0.0412	1.2578	1.0308
	PSO	0.6221	0.1488	1.4221	0.1775	9.9952	0.0028	0.3291	0.8009
	DE	0.5496	**0.0130**	1.4742	**0.1545**	10.0031	**0.0000**	**0.1675**	**0.7323**
	ABC	0.5353	0.0277	1.8069	0.7885	10.0031	**0.0000**	0.8163	1.7725
	SQP	0.4977	0.0336	2.1200	4.3063	10.0032	0.0000	4.3399	0.1816
	IPA	0.5336	0.0345	2.2607	3.7385	10.0032	0.0000	3.7730	0.5242
100	**SA**	0.5977	**0.0057**	1.3489	**0.0585**	10.0011	**0.0000**	**0.0641**	14.4397
	GA	0.8855	0.9812	1.2201	0.1272	9.9489	0.0420	1.1504	1.1519
	PSO	0.6450	0.1007	1.3248	0.0760	9.9949	0.0023	0.1790	**0.8404**
	DE	0.5977	**0.0057**	1.3490	**0.0585**	10.0011	**0.0000**	0.0642	0.8528
	ABC	0.5817	0.0213	1.6523	0.5234	10.0011	**0.0000**	0.5447	2.0264
	SQP	0.5499	0.0277	2.2930	4.3694	10.0011	0.0000	4.3971	0.1955
	IPA	0.5400	0.0185	1.9114	3.3346	10.0011	0.0000	3.3531	0.5504
250	**SA**	0.6256	**0.0022**	1.2840	**0.0202**	10.0003	**0.0000**	**0.0224**	9.4667
	GA	0.8908	0.9395	1.1810	0.0886	9.9494	0.0427	1.0708	1.1487
	PSO	0.6645	0.0754	1.2611	0.0351	9.9951	0.0021	0.1126	**0.8456**
	DE	0.6256	**0.0022**	1.2839	**0.0202**	10.0003	**0.0000**	**0.0224**	1.0555
	ABC	0.6077	0.0184	1.5722	0.3947	10.0002	**0.0000**	0.4130	2.2037
	SQP	0.5759	0.0226	2.3641	4.9995	10.0003	0.0000	5.0221	0.1930
	IPA	0.5888	0.0082	1.6474	2.0279	10.0003	0.0000	2.0361	0.5561
500	**SA**	0.6336	**0.0011**	1.2661	**0.0095**	10.0001	**0.0000**	**0.0105**	15.9482
	GA	0.8895	0.8861	1.1705	0.0772	9.9502	0.0404	1.0037	1.3762
	PSO	0.6650	0.0603	1.2465	0.0226	9.9959	0.0017	0.0846	**0.8639**
	DE	0.6337	**0.0011**	1.2660	**0.0095**	10.0001	**0.0000**	0.0106	1.3105
	ABC	0.6132	0.0174	1.5558	0.3750	10.0001	**0.0000**	0.3924	2.7671
	SQP	0.5850	0.0216	2.3702	5.0470	10.0001	0.0000	5.0686	0.2041
	IPA	0.6116	0.0046	1.4998	1.3907	10.0001	0.0000	1.3953	0.6037
1000	**SA**	0.6376	**0.0006**	1.2563	**0.0047**	10.0000	**0.0000**	**0.0053**	17.3379
	GA	0.8969	0.9054	1.1621	0.0739	9.9495	0.0408	1.0201	1.7012
	PSO	0.6650	0.0603	1.2465	0.0226	9.9959	0.0017	0.0846	**1.2426**
	DE	0.6377	**0.0006**	1.2563	**0.0047**	10.0000	**0.0000**	**0.0053**	1.3528
	ABC	0.6189	0.0167	1.5316	0.3436	10.0000	**0.0000**	0.3604	3.3038
	SQP	0.5935	0.0196	2.3471	4.9959	10.0000	0.0000	5.0155	0.2086
	IPA	0.6279	0.0016	1.3401	0.5347	10.0000	0.0000	0.5364	0.6180

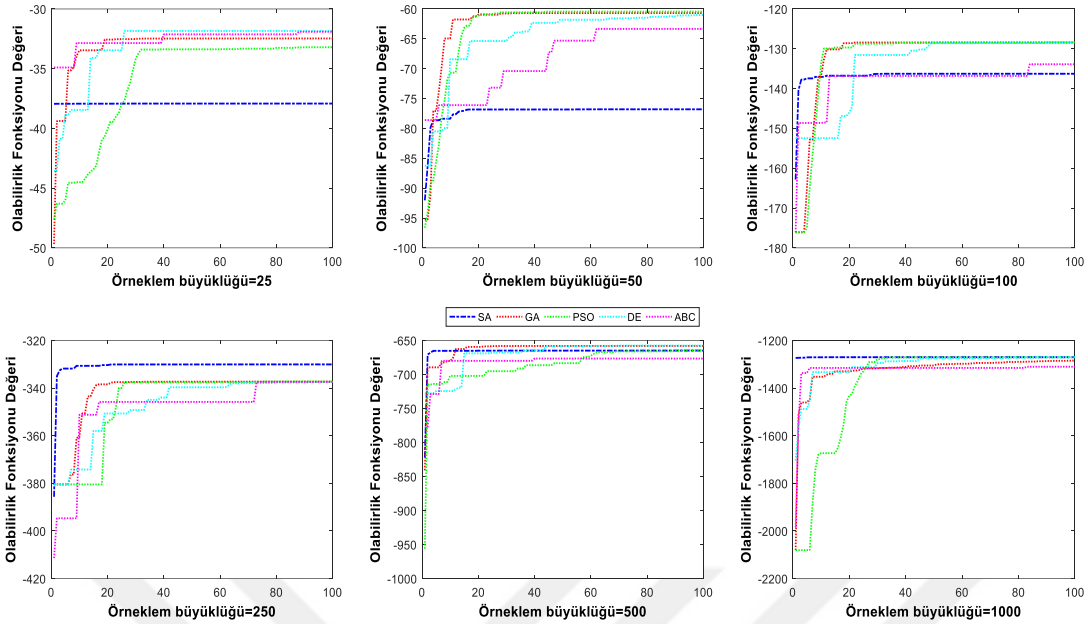
Çizelge 5.9-5.13'ten görüldüğü üzere, örneklem büyüklüğü arttıkça Def değerleri küçülmüştür. Başka bir ifadeyle, örneklem büyüklüğü arttıkça MSE kriterine göre daha etkin parametre tahminleri elde edilmiştir. Fakat, örneklem büyüklüğü artıkça olabilirlik fonksiyonunu optimize etmek daha zor olacağından yöntemlerin çalışma zamanlarının da artacağı açıkça belirtilmelidir.

Çözüm kalitesi açısından, Def kriterine göre en düşük Def değeri ile DE algoritması en yüksek performansı sergilemiştir. Ayrıca, DE dışında küçük örneklemler ($n = 50,100$) için GA ve büyük örneklemler ($n = 500,1000$) için ise SA algoritması da iyi performans göstermiştir. SA algoritmasının tahmin sonuçları özellikle büyük örneklemlerde DE algoritmasından elde edilen sonuçlara çok yakındır. Kısıtlı doğrusal olmayan optimizasyon algoritmaları SQP ve IPA ise çözüm kalitesi açısından çok kötü performans göstermiştir. Bunun nedeni SQP ve IPA algoritmalarının başlangıç değerinden yüksek oranda etkilenmesidir.

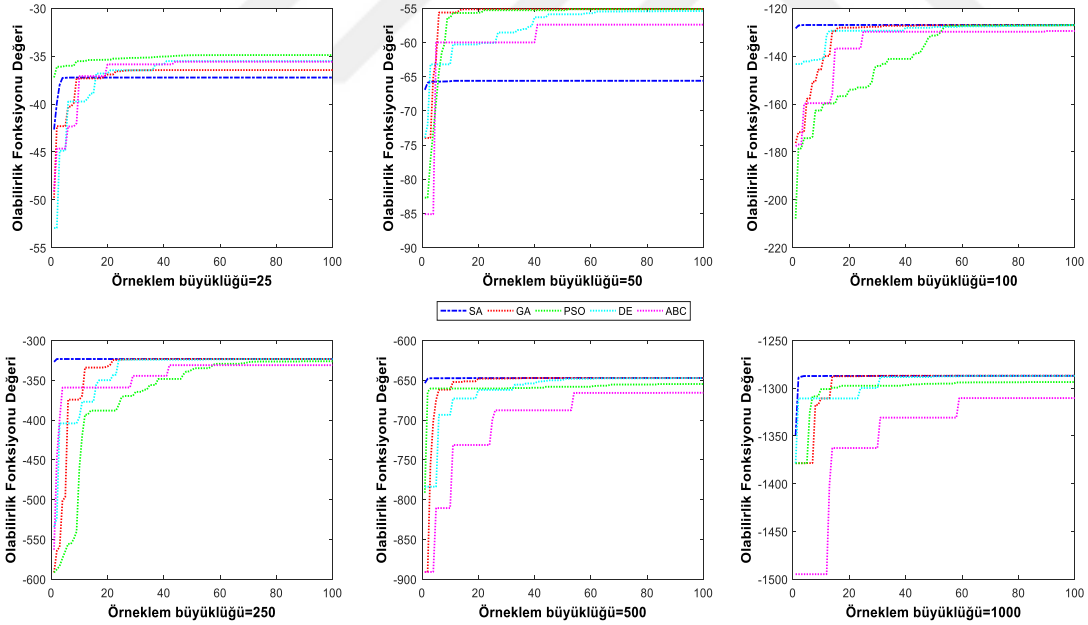
Çözüm zamanı açısından, IPA en iyi sonuçları vermiş ve ardından SQP algoritması gelmiştir. Metasezgisel yöntemler arasından PSO algoritması hemen hemen tüm durumlarda en iyi sonuçları vermiştir. Birkaç durumda ise, DE algoritması iyi performans sergilemiştir. Diğer yöntemler ile karşılaştırıldığında, SA algoritması hesaplama zamanı açısından çok düşük performans göstermiştir.

SQP ve IPA algoritmaları hesaplama zamanı açısından her ne kadar iyi olsalar da çözüm kalitesi açısından çok kötü performans sergilediklerinden metasezgisel yöntemlerin kullanılması tercih edilmiştir.

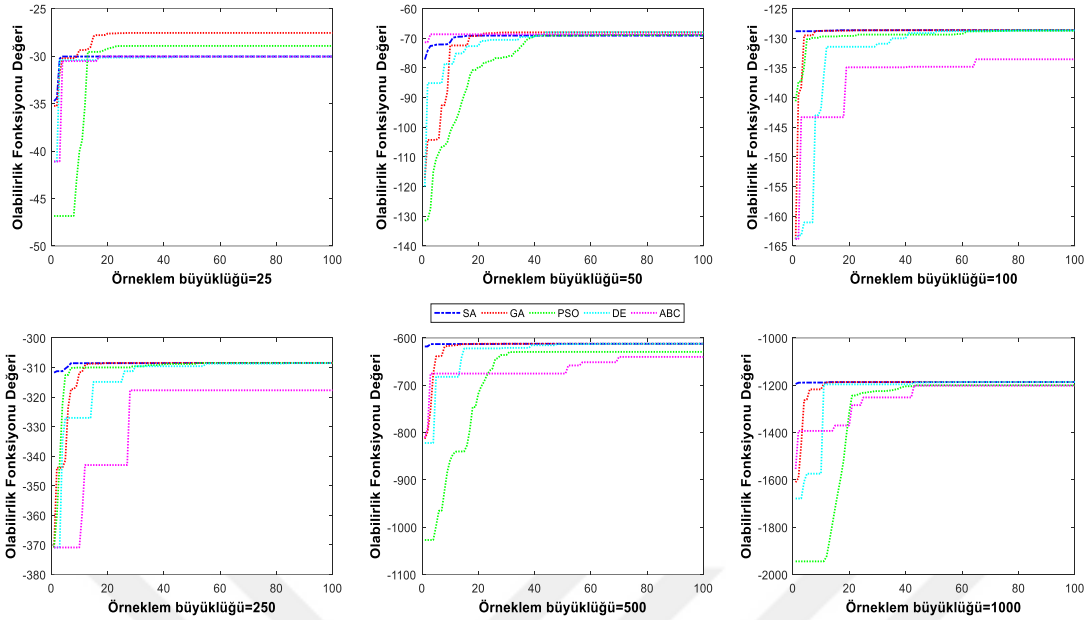
Çözüm kalitesi ve çözüm zamanının yanı sıra, metasezgisel yöntemlerin yakınsama davranışları da rasgelelik içerdiğinden ilgi çekicidir. Şekil 5.5-5.9'da sırasıyla $\theta = (4, 0.5, 10)$, $\theta = (2.56, 0.625, 10)$, $\theta = (1.778, 0.75, 10)$, $\theta = (1, 1, 10)$ ve $\theta = (0.64, 1.25, 10)$ parametreleri için tüm örneklem büyüklüklerinde tüm yöntemlerin maksimizasyon sürecinde log-olabilirlik fonksiyonunun davranışı gösterilmiştir. Şekil 5.5-5.9'da olası mümkün durumlar arasından rasgele seçilen log-olabilirlik fonksiyonu değerleri gösterilmiştir.



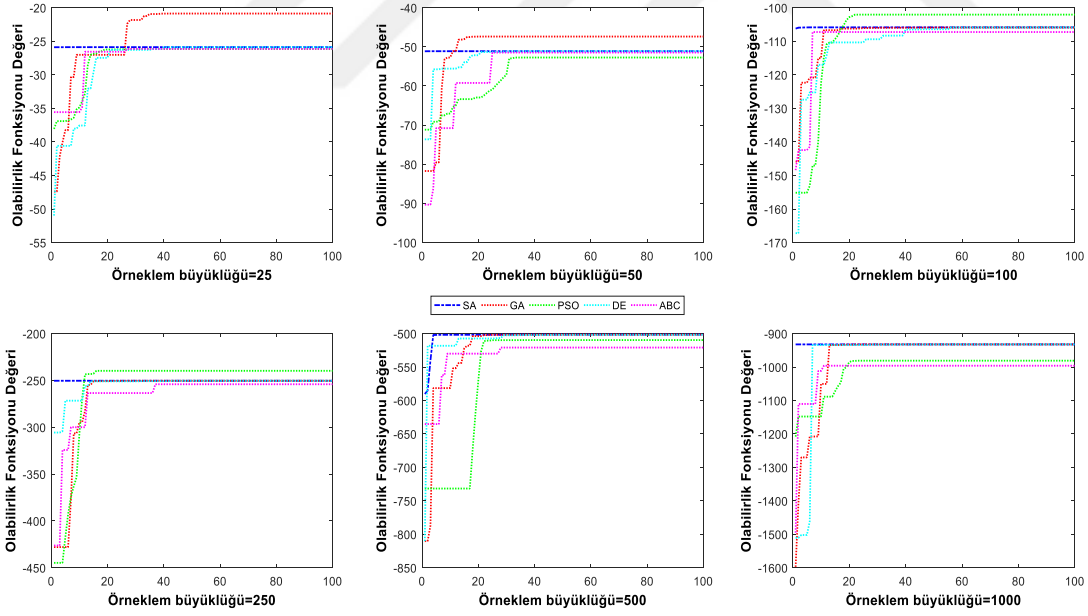
Şekil 5.5. $\theta = (4, 0.5, 10)$ parametresi için log-olabilirlik fonksiyonu değerleri



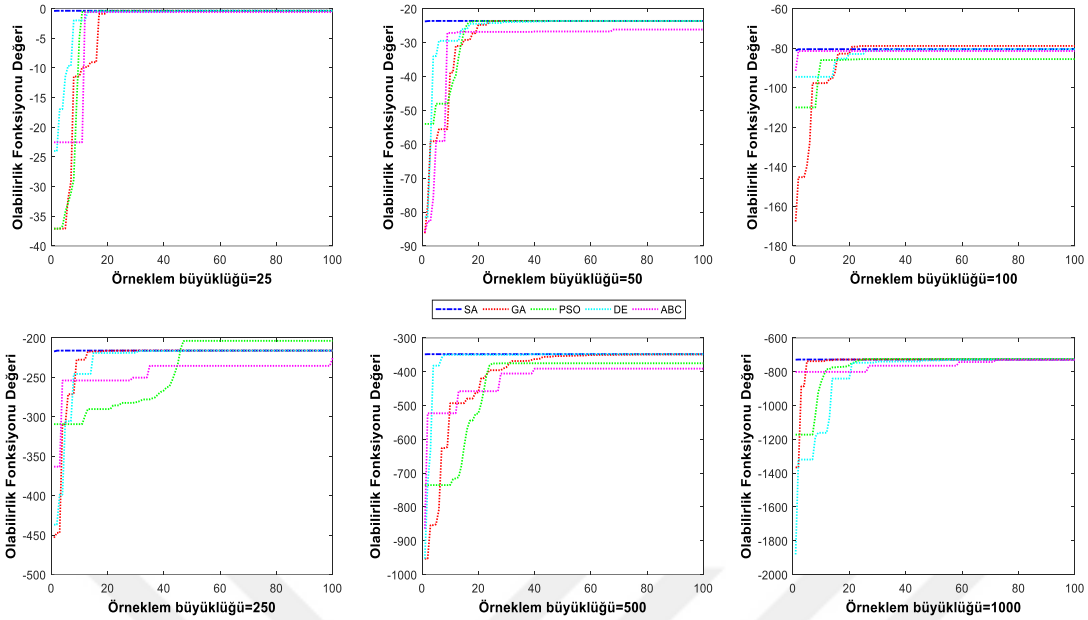
Şekil 5.6. $\theta = (2.56, 0.625, 10)$ parametresi için log-olabilirlik fonksiyonu değerleri



Şekil 5.7. $\theta = (1.778, 0.75, 10)$ parametresi için log-olabilirlik fonksiyonu değerleri



Şekil 5.8. $\theta = (1, 1, 10)$ parametresi için log-olabilirlik fonksiyonu değerleri



Şekil 5.9. $\theta = (0.64, 1.25, 10)$ parametresi için log-olabilirlik fonksiyonu değerleri

Şekil 5.5-5.9'dan görüldüğü üzere, SA algoritması genellikle en hızlı yakınsamayı, ABC algoritması ise en yavaş yakınsamayı sergilemiştir. SA algoritması için en fazla 10 iterasyon, GA için ortalama 20 iterasyon, PSO için ortalama 40 iterasyon, DE için ortalama $20-30$ iterasyon ve ABC için ortalama $50-60$ iterasyon sonunda yakınsama gerçekleşmiştir. Sonuç olarak, SA bu yöntemler arasından en hızlı yakınsama hızına sahip olan algoritmadır. Fakat, bu algoritma küçük örneklemler için yerel optimale takıldığından iyi sonuçlar vermemiştir.

5.2.3. TOPSIS yöntemi ile metasezgisel yöntemlerin karşılaştırılması

3-p Gamma dağılımının parametrelerinin ML tahmininde metasezgisel yöntemleri performanslarına göre sıralamak amacıyla TOPSIS yöntemi kullanılmıştır. Önerilen metasezgisel yöntemler (SA, GA, PSO, DE ve ABC) alternatifler olarak göz önüne alınmıştır. Ayrıca, bu yöntemlerin çözüm kalitesi (C_1), hesaplama zamanı (C_2), basitlik (C_3) ve sağlamlılığı (C_4) kriterler olarak alınmıştır. Çözüm kalitesi, basitlik ve sağlamlılık kriterleri fayda kriteri olarak, hesaplama zamanı ise maliyet kriteri olarak değerlendirilmiştir. Çözüm kalitesi değerleri, ortalama Def değerlerinin tersi alınarak elde edilmiştir. Hesaplama zamanı, yapılan tüm simülasyon çalışmalarının ortalama

çalışma süresi alınarak hesaplanmıştır. Basitlik, yöntemlerin içerdiği parametre sayılarının tersi alınarak değerlendirilmiştir. Örneğin SA algoritması s, T_0, nMove ve τ olmak üzere dört parametre içerdiğinden, basitlik değeri $1/4$ olarak alınmıştır. Sağlamlılık ise farklı örneklemler için elde edilen Def değerlerinin varyansının tersi alınarak belirlenmiştir.

Kriterlerin eşit öneme sahip oldukları düşünülmüş ve ağırlık değerleri 0.25 olarak alınmıştır. Oluşturulan karar matrisi, normalize edilmiş karar matrisi ve ağırlıklı karar matrisi aşağıda sırasıyla verilmiştir.

$$D = \begin{array}{c} \\ SA \\ GA \\ PSO \\ DE \\ ABC \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & C_3 & C_4 \end{array} \\ \begin{bmatrix} 0.6189 & 12.3571 & 0.2500 & 34.5414 \\ 0.5687 & 1.2042 & 0.2500 & 26.5129 \\ 0.6691 & 0.8719 & 0.2500 & 13.6154 \\ 1.0186 & 1.1788 & 0.3333 & 27.4304 \\ 0.2717 & 2.2287 & 0.5000 & 0.1091 \end{bmatrix} \end{array} \tag{5.12}$$

$$D' = \begin{array}{c} \\ SA \\ GA \\ PSO \\ DE \\ ABC \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & C_3 & C_4 \end{array} \\ \begin{bmatrix} 0.4112 & 0.9731 & 0.3375 & 0.6489 \\ 0.3778 & 0.0948 & 0.3375 & 0.4980 \\ 0.4446 & 0.0687 & 0.3375 & 0.2558 \\ 0.6767 & 0.0928 & 0.4500 & 0.5153 \\ 0.1805 & 0.1755 & 0.6751 & 0.0020 \end{bmatrix} \end{array} \tag{5.13}$$

$$D_w = \begin{array}{c} \\ SA \\ GA \\ PSO \\ DE \\ ABC \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & C_3 & C_4 \end{array} \\ \begin{bmatrix} 0.108 & 0.2433 & 0.0844 & 0.1622 \\ 0.0945 & 0.0237 & 0.0844 & 0.1245 \\ 0.1111 & 0.0172 & 0.0844 & 0.0639 \\ 0.1692 & 0.0232 & 0.1125 & 0.1288 \\ 0.0451 & 0.0439 & 0.1688 & 0.0005 \end{bmatrix} \end{array} \tag{5.14}$$

Eşitlik (3.47) ve (3.48) kullanılarak pozitif ideal çözüm $A^+ = (0.1692, 0.1720, 1.6880, 1.6222)$ olarak, negatif ideal çözüm $A^- = (0.0451, 0.2433, 0.0844, 0.0005)$ olarak elde edilmiştir. Ayrıca, hesaplanan d_i^+, d_i^- ve CC_i değerleri Çizelge 5.13'te verilmiştir.

Çizelge 5.14. Alternatifler için d_i^+, d_i^- ve CC_i değerleri

Alternatifler	d_i^+	d_i^-	CC_i
SA	0.0627	0.0295	0.3199
GA	0.0142	0.0660	0.8233
PSO	0.0201	0.0595	0.7471
DE	0.0043	0.0811	0.9495
ABC	0.0423	0.0469	0.5260

Çizelge 5.14'e göre, alternatiflerin sıralaması $CC_4 > CC_2 > CC_3 > CC_5 > CC_1$ olduğundan yöntemlerin sıralaması $DE > GA > PSO > ABC > SA$ biçimindedir. Tüm kriterler göz önüne alındığında tüm alternatifler arasında en iyi olan metasezgisel yöntemin DE algoritması olduğu sonucuna ulaşılmıştır.

5.2.4. Gerçek veri uygulaması

Bu alt kesimde, ilk olarak Dumonceaux ve Antle (1973) tarafından kullanılan gerçek bir veri kümesi ele alınmıştır. "Akış (Flood) veri kümesi" olarak bilinen bu veri kümesi Harrisburg'deki Susqehanna Nehri'ndeki 1969-1980 arasındaki dört dönem boyunca saniyedeki maksimum akış seviyesini içermekte olup veriler Çizelge 5.15'te verilmiştir (Balakrishnan ve Wang, 2000; Lakshmi ve Vaidyanathan, 2016):

Çizelge 5.15. Akış **(**Flood) veri kümesi

0.654	0.613	0.315	0.449	0.297	0.402	0.379	0.423	0.379	0.3235
0.269	0.740	0.418	0.412	0.494	0.416	0.338	0.392	0.484	0.265

Yöntemlerin karşılaştırılmasında, log-olabilirlik fonksiyonu değerine ek olarak, *AIC* ve *BIC* değerleri de kullanılmıştır.

Bu çalışmada kullanılan metasezgisel yöntemler, Lakshmi ve Vaidyanathan (2016) tarafından sunulan en çok olabilirlik (MLE_P) yöntemi ve Clifford ve Jones (1982) tarafından önerilen uyarlanmış en çok olabilirlik (MML) yöntemi için ML tahmin değerleri, $\ln L$ değerleri, *AIC* ve *BIC* değerleri Çizelge 5.16'da verilmiştir.

Çizelge 5.16. Akış (Flood) veri kümesi için parametre tahmini değerleri, $\ln L$, *AIC* ve *BIC* değerleri

Yöntem	$\hat{\sigma}$	$\hat{\beta}$	$\hat{\mu}$	$\ln L$	*AIC*	*BIC*
SA	1.3865	0.1180	0.2595	16.8010	-27.6021	-15.6276
GA	1.3638	0.1195	0.2600	16.8009	-27.6018	-15.6274
PSO	1.7143	0.1020	0.2519	16.7684	-27.5368	-15.5624
DE	0.9681	0.1658	0.2646	**16.8371**	**-27.6742**	**-15.6998**
ABC	1.3972	0.1299	0.2417	15.8753	-25.7506	-13.7762
MLE_P	1.3700	0.1300	0.2700	16.8190	-27.6380	-15.6636
MML	1.1940	0.1343	0.2628	16.7971	-27.5942	-15.6198

Çizelge 5.16'dan, ABC algoritması haricindeki metasezgisel yöntemler birbirine yakın *AIC* ve *BIC* değerlerine sahip olduğu görülmektedir. DE algoritması hem önerilen metasezgisel yöntemler hem de MLE_P ve MML yöntemleri arasında en büyük $\ln L$ ve en küçük *AIC* ve *BIC* değerlerine sahiptir. Böylece, DE yöntemi bu veri kümesi için ML parametre tahmininde diğer yöntemlere göre daha iyi performans sergilemiştir.

5.3. 3-p Gamma Dağılımının Parametrelerinin Diferansiyel Gelişim Algoritması ile Tahmini İçin Yeni Bir Yaklaşım

Bu kesimde, 3-p Gamma dağılımının parametrelerinin ML ve LS yöntemi ile tahminine yer verilmiştir. 3-p Gamma dağılımının parametrelerinin ML ve LS yöntemi ile tahmini için DE algoritması kullanılmıştır. DE algoritmasının performansı, diğer metasezgisel yöntemlerde olduğu gibi arama uzayından oldukça etkilenmektedir. Bundan dolayı, bu çalışmada MML ve PL yöntemleri kullanılarak her bir parametre değeri için güven aralıklarının oluşturulması ve bu güven aralıklarının DE algoritmasının arama uzayı olarak kullanılması önerilmiştir. Burada, MML yönteminin kullanılmasının nedeni, 3-p Gamma dağılımı için MML tahmin edicilerinin açık bir formda elde edilebiliyor olmasıdır. Başka bir ifadeyle, MML tahminlerinin gözlemlerin bir fonksiyonu olarak yazılabilmesidir.

Önerilen yaklaşımın uygulanabilmesi için gerekli bilgiler verildikten sonra, Monte-Carlo simülasyon çalışmaları ile çeşitli parametre değerleri ve arama uzayları kullanılarak yöntemin etkinliği test edilmiştir. Ayrıca, gerçek bir veri kümesi uygulaması sunulmuştur.

3-p Gamma dağılımı için ML yöntemi ile parametre tahmini Bölüm 5.32'de verildiğinden burada yalnızca LS yöntemi açıklanmıştır.

5.3.1. 3-p Gamma dağılımının parametrelerinin tahmini için LS yöntemi

$x_{(1)} \le x_{(2)} \le ... \le x_{(n)}$ $Gamma(\alpha, \beta, \mu)$ dağılımına sahip kitleden alınan sıra istatistiklerinin gözlenen değeri olmak üzere $x_{(i)}$ sıra istatistiğine ilişkin dağılım fonksiyonu

$$F(x_{(i)}) = \frac{1}{\Gamma(\sigma)} \gamma\left(\sigma, \frac{x_{(i)} - \mu}{\beta}\right), \qquad x_{(i)} \ge \mu, \sigma > 0, \beta > 0 \tag{5.15}$$

biçiminde yazılabilir. Eşitlik (5.15)'te verilen $F(x_{(i)})$ yerine ampirik dağılım fonksiyonu yazıldığında,

$$\frac{i}{n+1} = \frac{1}{\Gamma(\sigma)} \gamma\left(\sigma, \frac{x_{(i)} - \mu}{\beta}\right) + \varepsilon_i, i = 1, 2, ..., n \tag{5.16}$$

modeli yazılabilir. ε_i hata terimlerinin kareleri toplamını minimum yapan parametre değerleri Gamma dağılımının parametrelerinin LS tahminleri olur ve

$$\hat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta} \in \Theta} \sum_{i=1}^{n} \varepsilon_i^2 = \min_{\boldsymbol{\theta} \in \Theta} \sum_{i=1}^{n} \left(\frac{i}{n+1} - \frac{1}{\Gamma(\sigma)} \gamma\left(\sigma, \frac{x_{(i)} - \mu}{\beta}\right) \right)^2 \tag{5.17}$$

biçiminde ifade edilir (Lakshmi ve Vaidyanathan, 2016).

5.3.2. DE algoritması için arama uzayının MML ve PL yöntemleri kullanılarak belirlenmesi

DE algoritmasının performansı, parametrelerin arama uzayından büyük ölçüde etkilenmektedir. DE parametre tahmin değerlerini bu arama uzayında aramaktadır. Arama uzayının rastgele belirlenmesi bazı sorunlara neden olabilir:

i. Arama alanı çok geniş belirlenirse, algoritma kabul edilemez sonuçlar üretebilir. Ayrıca, arama uzayı geniş olduğunda alanı iyi tarayabilmek için popülasyon büyüklüğünün arttırılması gerekir. Bu durum, algoritmanın hesaplama zamanında ciddi bir artışa neden olur.

ii. Arama alanı çok dar belirlenirse, aranan gerçek parametre değerleri içermeyebilir ve algoritma global optimal nokta olarak yanlış bir değere yakınsar.

Bu sorunların üstesinden gelebilmek için arama uzayı hem yeterince dar olmalı hem de aranan gerçek değeri içermelidir. Bundan dolayı, bu çalışmada MML tahmin edicilerine ve Profil Olabilirlik (PL) yöntemine dayalı güven aralıklarını kullanarak 3-p Gamma dağılımının parametreleri için arama uzayı oluşturmak üzere yeni bir yaklaşım önerilmiştir.

Gamma dağılımının parametrelerinin MML tahminlerini elde etmek için aşağıdaki adımlar uygulanır (Tiku ve Akkaya, 2004):

Adım 1. Standartlaştırılmış gözlemler $z_i = \dfrac{x_i - \mu}{\beta}$ artan şekilde $z_{(1)} \le z_{(2)} \le \ldots \le z_{(n)}$ olarak sıralanır.

Adım 2. Sıra istatistikleri kullanılarak olabilirlik denklemleri yeniden yazılır.

$$\frac{\partial \ln L}{\partial \sigma} = \sum_{i=1}^{n} \log(z_{(i)}) - \frac{n}{\beta}\psi(\sigma) - \frac{n}{\beta}\log\beta = 0 \qquad (5.18)$$

$$\frac{\partial \ln L}{\partial \beta} = \frac{1}{\beta}\sum_{i=1}^{n} z_{(i)} - \frac{\sigma - 1}{\beta}\sum_{i=1}^{n} z_{(i)} w(z_{(i)}) - \frac{n}{\beta} = 0 \tag{5.19}$$

$$\frac{\partial \ln L}{\partial \mu} = -\frac{(\sigma - 1)}{\beta}\sum_{i=1}^{n} w(z_{(i)}) + \frac{n}{\beta} = 0 \tag{5.20}$$

Eşitlik (5.18)-(5.20)'de, $w(z) = z^{-1}$ ve $z_{(i)} = \left(x_{(i)} - \mu\right)/\beta$ 'dır. Ayrıca, tüm gözlemlerin toplamı, sıralanmış gözlemlerin toplamına eşit olup $\sum_{i=1}^{n} z_{(i)} = \sum_{i=1}^{n} z_i$ 'dir.

Adım 3. $t_{(i)} = E(z_{(i)})$, $z_{(i)}$ sıralı standartlaştırılmış değişkeninin beklenen değeri olmak üzere, $w(z_{(i)})$ lineer olmayan fonksiyonunun $t_{(i)}$ civarındaki Taylor serisi açılımının ilk iki terimi kullanılarak lineer bir yaklaşımı

$$w(z_{(i)}) \cong w(t_{(i)}) + (z_{(i)} - t_{(i)})\left\{\frac{\partial}{\partial z} w(z)\right\}_{z=t_{(i)}} = \lambda_i - \gamma_i z_{(i)},\ i = 1, 2, ..., n \tag{5.21}$$

şeklinde yazılabilir. Burada,

$$\gamma_i = -\frac{\partial}{\partial z} w(z)|_{z=t_{(i)}} = -\frac{\partial}{\partial z} z^{-1}|_{z=t_{(i)}} = \frac{1}{t_{(i)}^2}, \tag{5.22}$$

$$\lambda_i = w(t_{(i)}) + t_{(i)}\gamma_i = \frac{1}{t_{(i)}} + t_{(i)}\frac{1}{t_{(i)}^2} = \frac{2}{t_{(i)}} \tag{5.23}$$

olup $t_{(i)}$, i. sıra istatistiğinin beklenen değeri olarak tanımlanır. $t_{(i)}$' nin gerçek değerleri mevcut değildir. Fakat, $\int_{-\infty}^{t_{(i)}} f(z)dz = \frac{i}{n+1}$ eşitliğinin çözümünden $t_{(i)}$'nin yaklaşık değerleri elde edilebilir. Burada $f(.)$, 3-p Gamma dağılımının olasılık yoğunluk fonksiyonunu göstermektedir.

Adım 4. Eşitlik (5.21)'in Eşitlik (5.20)'de yerine konulmasıyla uyarlanmış olabilirlik eşitlikleri,

$$\frac{\partial \ln L}{\partial \sigma} = \sum_{i=1}^{n} \log(z_{(i)}) - \frac{n}{\beta}\psi(\sigma) - \frac{n}{\beta}\log\beta = 0 \tag{5.24}$$

$$\frac{\partial \ln L}{\partial \beta} = \frac{1}{\beta}\sum_{i=1}^{n} z_{(i)} - \frac{\sigma - 1}{\beta}\sum_{i=1}^{n} z_{(i)}(\lambda_i - \gamma_i z_{(i)}) - \frac{n}{\beta} = 0 \tag{5.25}$$

$$\frac{\partial \ln L}{\partial \mu} = -\frac{(\sigma - 1)}{\beta}\sum_{i=1}^{n}(\lambda_i - \gamma_i z_{(i)}) + \frac{n}{\beta} = 0 \tag{5.26}$$

olarak elde edilir.

Adım 5. Eşitlik (5.24)-(5.26)'da verilen uyarlanmış eşitlikler, ilgilenilen β ve μ parametrelerine göre çözülerek MML tahmin edicileri,

$$\hat{\mu}_{MML} = \hat{\mu}_n - \frac{\Delta}{m}\hat{\beta}_{MML} \tag{5.27}$$

ve

$$\hat{\beta}_{MML} = \frac{-B + \sqrt{B^2 + 4nC}}{2\sqrt{n(n-3)}} \tag{5.28}$$

şeklinde elde edilir. Eşitlik (5.27) ve (5.28)'de;

$\hat{\mu}_n = \frac{1}{m}\sum_{i=1}^{n}\gamma_i x_{[i]}$,

$m = \sum_{i=1}^{n}\gamma_i$,

$\Delta_i = \lambda_i - (\sigma - 1)^{-1}$,

$\Delta = \sum_{i=1}^{n}\Delta_i$

$$B = (\sigma - 1)\sum_{i=1}^{n} \Delta_i \left(x_{[i]} - \hat{\mu}_n \right),$$

$$C = (\sigma - 1)\sum_{i=1}^{n} \gamma_i \left(x_{[i]} - \hat{\mu}_n \right)^2$$

olarak hesaplanır.

Eşitlik (5.27) ve (5.28)'de verilen MML tahmin edicileri gözlemlerin bir fonksiyonu olduğundan açıkça yazılabilmektedir. Bundan dolayı, MML tahminlerini hesaplamak oldukça kolaydır.

MML tahmin edicileri, ML tahmin edicilerine asimptotik olarak eşit olduğundan, asimptotik varyans-kovaryans matrisi $I(\mu, \beta)$, Fisher bilgi matrisinin tersine $I^{-1}(\mu, \beta)$ eşittir. Böylece, $\sigma > 2$ için asimptotik varyanslar

$$Var(\hat{\mu}_{MML}) = \frac{\sigma(\sigma - 2)\beta^2}{2n} \tag{5.29}$$

ve

$$Var(\hat{\beta}_{MML}) = \frac{\beta^2}{2(n-3)} \tag{5.30}$$

olarak elde edilebilir. Konum ve ölçek parametreleri için asimptotik $(1-\alpha)$ anlam seviyeli güven aralıkları ise sırasıyla,

$$\hat{\mu}_{MML} - z_{\alpha/2}\sqrt{Var(\hat{\mu}_{MML})} < \mu < \hat{\mu}_{MML} - z_{\alpha/2}\sqrt{Var(\hat{\mu}_{MML})} \tag{5.31}$$

ve

$$\hat{\beta}_{MML} - z_{\alpha/2}\sqrt{Var(\hat{\beta}_{MML})} < \beta < \hat{\beta}_{MML} - z_{\alpha/2}\sqrt{Var(\hat{\beta}_{MML})} \tag{5.32}$$

şeklindedir.

MML tahmin edicileri, şekil parametresinin bilindiği varsayımı altında elde edilebilmektedir. Bundan dolayı şekil parametresinin tahmini için PL yöntemi kullanılabilir. Bu çalışmada ele alınan PL yöntemi adımları aşağıda verilmiştir.

***Adım 1*:** Verilen bir σ değeri için $\hat{\mu}_{MML}$ ve $\hat{\beta}_{MML}$ hesaplanır.

***Adım 2*:** Log-olabilirlik değeri $\ln L(\sigma, \hat{\mu}_{MML}, \hat{\beta}_{MML} / \underline{x})$ hesaplanır.

***Adım 3*:** Çeşitli σ değerleri için Adım 1 ve Adım 2 tekrar edilir.

***Adım 4*:** Log-olabilirlik fonksiyonunu maksimum yapan σ değeri şekil parametresi için olası tahmin olarak belirlenir.

Şekil parametresi σ için asimptotik $(1-\alpha)$ anlam düzeyindeki güven aralığı

$$\hat{\sigma} - z_{\alpha/2}\sqrt{Var(\hat{\sigma})} < \sigma < \hat{\sigma} + z_{\alpha/2}\sqrt{Var(\hat{\sigma})} \tag{5.33}$$

biçiminde yazılabilir.

5.3.3. Simülasyon çalışması

Bu alt kesimde, önerilen yaklaşımın ML ve LS yöntemlerindeki performansını araştırmak amacıyla Monte-Carlo simülasyon çalışması yapılmıştır.

Gamma dağılımı için gerçek parametre değerleri $\theta = (3,1,-2)$, $\theta = (4,0.5,10)$ ve $\theta = (2.56,0.75,8)$ olarak alınmıştır. Ayrıca, parametre tahmin sürecinde örneklem büyüklüğünün etkisini araştırmak için her bir parametre değeri için örneklem büyüklükleri 10, 30, 50, 100, 250, 500 ve 1000 olarak ele alınmıştır.

Diferansiyel gelişim (DE) algoritması için ölçekleme faktörü $F \in (0,1]$ ve çaprazlama faktörü $C_r = 2$ olarak kullanılmıştır. Popülasyon boyutu 10, 25 ve 50 olarak alınmıştır. Önerilen yaklaşımda ele alınan arama uzayı, MML tahminlerine dayalı güven aralıkları kullanılarak elde edilmiştir. Ayrıca, Çizelge 5.17'de verilen rastgele belirlenen arama uzayları kullanılarak, farklı arama uzayları için tahminlerin etkinliği karşılaştırılmıştır.

Çizelge 5.17. Diferansiyel Gelişim Algoritması için çeşitli arama uzayları

Önerilen Arama Uzayı (SSP_ML ve SSP_LS)	MML tahminlerine dayalı elde edilen %95 güven aralığı
Arama Uzayı 1 (SS1_ML ve SS1_LS)	σ için $[0,20]$, β için $[0,20]$, μ için $[-10,10]$
Arama Uzayı 2 (SS2_ML ve SS2_LS)	σ için $[0,100]$, β için $[0,100]$, μ için $[-50,50]$

Çizelge 5.17'de,

SSP_ML: En çok olabilirlik tahminlerini bulurken uyarlanmış en çok olabilirlik tahminleri ile elde edilen %95 güven aralığını kullanan arama uzayını,

SSP_LS: En küçük kareler tahminlerini bulurken uyarlanmış en çok olabilirlik tahminleri ile elde edilen %95 güven aralığını kullanan arama uzayını,

SS1_ML: En çok olabilirlik tahminlerini bulurken σ için $[0,20]$, β için $[0,20]$, μ için $[-10,10]$ aralığını kullanan arama uzayını,

SS1_LS: En küçük kareler tahminlerini bulurken σ için $[0,20]$, β için $[0,20]$, μ için $[-10,10]$ aralığını kullanan arama uzayını,

SS2_ML: En çok olabilirlik tahminlerini bulurken σ için $[0,100]$, β için $[0,100]$, μ için $[-50,50]$aralığını kullanan arama uzayını,

SS2_LS: En küçük kareler tahminlerini bulurken σ için $[0,100]$, β için $[0,100]$, μ için $[-50,50]$aralığını kullanan arama uzayını göstermektedir.

Her bir simülasyon yapısı 1000 kez tekrarlanmıştır. Böylece, $3\times7\times3\times1000=63000$ bağımsız deney gerçekleştirilmiştir. 3-p Gamma dağılımının parametrelerinin hem ML hem de LS tahminleri DE algoritması kullanılarak hesaplanmıştır. Simülasyonlar sonucunda elde edilen parametre tahmini değerleri, MSE ve Def değerleri Çizelge 5.18-5.26'da verilmiştir.

Çizelge 5.18. $pop = 10$ için parametreleri $\theta = (\sigma, \beta, \mu) = (3,1,-2)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

		$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
n	**Yöntem**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	4.1081	6.3603	0.8273	0.1188	-1.9823	0.7847	**7.2639**
	SS1_ML	2.9587	13.4602	1.9513	2.7452	-1.7448	1.9343	18.1397
	SS2_ML	10.2772	296.1017	2.1231	3.9120	-9.9916	233.4482	533.462
	SSP_LS	4.9978	8.1338	0.8744	0.0855	-3.0801	2.9397	11.1590
	SS1_LS	4.7277	20.6180	1.4697	1.1587	-3.2869	6.5435	28.3202
	SS2_LS	13.1472	528.8568	5.3684	82.8275	-16.4342	451.4205	1063.1049
30	**SSP_ML**	3.1466	0.8031	0.9679	0.0333	-1.9414	0.1927	**1.02916**
	SS1_ML	3.4958	9.9920	1.2642	0.4509	-2.1178	1.3599	11.8028
	SS2_ML	10.6030	300.9108	1.7162	2.3138	-10.1954	245.0608	548.2855
	SSP_LS	3.5873	1.2773	0.9647	0.0307	-2.3604	0.5391	1.8470
	SS1_LS	4.5447	18.5068	1.3150	0.6667	-3.0188	4.9930	24.1665
	SS2_LS	10.5735	310.4276	4.5812	41.1483	-18.2958	539.2399	890.8158
50	**SSP_ML**	2.8464	0.4929	1.0392	0.0279	-1.8731	0.1243	**0.6452**
	SS1_ML	3.6563	9.3872	1.1394	0.2531	-2.2046	1.2856	10.9259
	SS2_ML	10.5346	294.4134	1.7056	3.3944	-10.0279	230.6749	528.4827
	SSP_LS	3.1328	0.5080	1.0199	0.0217	-2.1103	0.2035	0.7331
	SS1_LS	4.9827	22.8831	1.2133	0.4902	-3.1541	5.6937	29.0670
	SS2_LS	12.9022	507.8048	4.4482	52.1437	-15.4235	403.6760	963.6245
100	**SSP_ML**	2.8215	0.2478	1.0430	0.01457	-1.9027	0.0619	**0.3244**
	SS1_ML	3.5953	7.9865	1.1034	0.1841	-2.2270	1.2396	9.4103
	SS2_ML	10.4232	285.6879	1.6719	3.3319	-10.4079	245.7482	534.768
	SSP_LS	2.9650	0.2430	1.0336	0.0106	-2.0246	0.1106	0.3642
	SS1_LS	5.0581	22.0390	1.1738	0.4621	-3.3092	6.0832	28.5843
	SS2_LS	12.4918	468.1017	4.6259	57.5204	-15.2758	407.4964	933.1186
250	**SSP_ML**	2.8218	0.1047	1.0462	0.0067	-1.9367	0.0205	**0.1320**
	SS1_ML	3.6613	6.8480	1.0542	0.1115	-2.2907	1.0608	8.0204
	SS2_ML	10.9778	285.3779	1.5901	5.3516	-10.7245	251.501	542.2306
	SSP_LS	2.8453	0.1041	1.0480	0.0063	-1.9614	0.0376	0.1480
	SS1_LS	4.8449	19.0609	1.1654	0.4484	-3.1308	4.7909	24.3002
	SS2_LS	10.6644	318.6986	4.5701	47.9942	-17.8700	520.3152	887.0080
500	**SSP_ML**	2.8825	0.0503	1.0290	0.0029	-1.9559	0.0102	**0.0634**
	SS1_ML	3.5845	6.3440	1.0549	0.1533	-2.2392	0.8191	7.3166
	SS2_ML	9.8410	481.4533	1.6262	3.0355	-10.4271	246.2865	481.4533
	SSP_LS	2.8758	0.0558	1.0327	0.0030	-1.9561	0.0192	0.0780
	SS1_LS	4.8133	19.5092	1.1741	0.3769	-3.1835	5.1652	25.0513
	SS2_LS	10.8120	332.7709	4.9663	73.5282	-18.4075	529.6455	935.9445
1000	**SSP_ML**	2.9398	0.0242	1.0149	0.0013	-1.9777	0.0044	**0.0300**
	SS1_ML	3.5932	5.8670	1.0566	0.1903	-2.2932	1.0644	7.1218
	SS2_ML	10.8903	301.1794	1.6159	3.3527	-11.0935	272.0797	576.6119
	SSP_LS	2.9421	0.0282	1.0156	0.0012	-1.9801	0.0104	0.0398
	SS1_LS	5.1798	21.4202	1.1375	0.3950	-3.3210	5.6725	27.4877
	SS2_LS	11.4837	402.5591	4.5409	49.0188	-17.7199	497.7178	949.2957

Çizelge 5.19. *pop* = 25 için parametreleri $\theta = (\sigma, \beta, \mu) = (3,1,-2)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	3.8700	7.2861	0.8814	0.1202	-1.8610	0.8559	**8.2622**
	SS1_ML	2.1969	11.6152	2.4064	3.7625	-1.3813	1.2610	16.6388
	SS2_ML	3.4367	73.7615	2.3816	3.5016	-2.2581	15.281	92.5445
	SSP_LS	4.8712	8.9178	0.9023	0.1034	-2.9515	2.7041	11.7253
	SS1_LS	3.8317	15.1783	1.5499	1.6589	-2.3889	3.0184	19.8557
	SS2_LS	8.1162	306.1603	3.7872	37.8444	-6.8014	130.0211	474.0258
30	**SSP_ML**	3.0062	0.8182	0.9973	0.0352	-1.8827	0.1956	**1.0490**
	SS1_ML	2.8752	5.7905	1.3248	0.4839	-1.8127	0.5362	6.8106
	SS2_ML	4.4656	88.1826	1.4980	0.7484	-2.8087	21.236	110.1678
	SSP_LS	3.5372	1.4636	0.9762	0.0375	-2.3149	0.5231	2.0242
	SS1_LS	3.9305	12.5226	1.1817	0.3005	-2.3609	1.9107	14.7339
	SS2_LS	7.9032	240.2697	3.3291	21.8602	-8.8312	195.4372	457.5670
50	**SSP_ML**	2.7374	0.5794	1.0685	0.0348	-1.8285	0.1363	**0.7506**
	SS1_ML	2.8597	2.2978	1.1620	0.1810	-1.8561	0.2551	2.7340
	SS2_ML	4.7705	82.6319	1.3311	0.4161	-2.9714	19.716	102.7642
	SSP_LS	3.0971	0.6096	1.0294	0.0282	-2.0775	0.2051	0.8429
	SS1_LS	3.7560	10.4265	1.1414	0.2191	-2.2635	1.4418	12.0874
	SS2_LS	8.7670	297.4878	2.6827	17.7751	-6.9580	132.6581	447.9209
100	**SSP_ML**	2.7481	0.2872	1.0616	0.0188	-1.8734	0.0650	**0.3710**
	SS1_ML	2.8744	1.2812	1.0826	0.0651	-1.9029	0.1257	1.4721
	SS2_ML	4.2958	66.2131	1.2218	0.2086	-2.7249	15.6844	82.1062
	SSP_LS	2.9168	0.2829	1.0432	0.0149	-1.9899	0.1053	0.4032
	SS1_LS	3.9077	10.3111	1.0583	0.1248	-2.3261	1.3128	11.7487
	SS2_LS	8.8412	317.7574	2.7961	24.0806	-6.5267	107.2595	449.0975
250	**SSP_ML**	2.8023	0.1177	1.0514	0.0079	-1.9292	0.0214	**0.1471**
	SS1_ML	2.8975	0.4320	1.0485	0.0263	-1.9525	0.0446	0.5030
	SS2_ML	4.2272	58.8406	1.1664	0.1259	-2.8007	16.7065	75.6786
	SSP_LS	2.8301	0.1250	1.0500	0.0076	-1.9475	0.0388	0.1714
	SS1_LS	3.8379	9.0928	1.0346	0.0893	-2.3108	1.1798	10.3619
	SS2_LS	8.9133	306.1155	3.1061	28.2233	-8.3519	172.9180	507.2569
500	**SSP_ML**	2.8726	0.0577	1.0315	0.0033	-1.952	0.01093	**0.07206**
	SS1_ML	2.9328	0.3180	1.0319	0.0152	-1.9692	0.03217	0.3654
	SS2_ML	4.3728	65.1725	1.1248	0.1014	-2.7968	15.1136	80.3876
	SSP_LS	2.8743	0.0630	1.0328	0.0035	-1.9529	0.0196	0.0861
	SS1_LS	3.9218	9.9909	1.0106	0.0727	-2.3333	1.1863	11.2498
	SS2_LS	9.6920	335.7865	2.6505	11.6950	-9.0366	197.3591	544.8407
1000	**SSP_ML**	2.9329	0.0263	1.0165	0.0014	-1.9747	0.0045	**0.0323**
	SS1_ML	2.9586	0.2512	1.0225	0.0094	-1.9849	0.0283	0.2890
	SS2_ML	4.1003	45.1303	1.1123	0.08952	-2.7366	13.6658	58.8857
	SSP_LS	2.9375	0.0327	1.0158	0.0015	-1.9753	0.0105	0.0447
	SS1_LS	4.0281	9.4578	0.9909	0.0728	-2.3915	1.1256	10.6562
	SS2_LS	9.6719	337.4216	2.6890	10.6083	-9.4090	200.6948	548.7248

Çizelge 5.20. $pop = 50$ için parametreleri $\theta = (\sigma, \beta, \mu) = (3,1,-2)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	3.8313	7.8802	0.9015	0.1225	-1.8383	0.9051	**8.9079**
	SS1_ML	2.1334	12.4083	2.5724	4.3032	-1.3570	1.3122	18.0238
	SS2_ML	1.8889	25.7838	2.6660	4.3152	-1.3236	1.8228	31.9219
	SSP_LS	4.9744	10.3985	0.9064	0.1129	-2.9611	2.8361	13.3474
	SS1_LS	3.8160	14.0347	1.5306	1.5668	-2.3218	2.5635	18.1650
	SS2_LS	6.2842	270.3826	3.5075	39.5510	-3.4243	42.3600	352.2937
30	**SSP_ML**	2.9827	0.8296	1.0037	0.0365	-1.8740	0.1959	**1.0622**
	SS1_ML	2.9501	6.3661	1.3269	0.5097	-1.8234	0.5480	7.4239
	SS2_ML	2.6886	13.7491	1.4812	0.6489	-1.8020	1.3787	15.7768
	SSP_LS	3.5093	1.5774	0.9852	0.0418	-2.2947	0.5237	2.1429
	SS1_LS	3.7228	11.4252	1.1991	0.3329	-2.2041	1.2908	13.0490
	SS2_LS	6.3778	215.3560	2.4797	8.1826	-4.1700	51.4928	275.0314
50	**SSP_ML**	2.7237	0.6010	1.0734	0.0362	-1.8239	0.1404	**0.7777**
	SS1_ML	2.9369	3.1182	1.1523	0.1777	-1.8685	0.2878	3.5838
	SS2_ML	2.9647	17.7667	1.2830	0.2768	-1.9249	1.3972	19.4408
	SSP_LS	3.0960	0.6808	1.0342	0.0323	-2.0742	0.2098	0.9229
	SS1_LS	3.5481	8.3221	1.1285	0.1844	-2.1361	0.8902	9.3967
	SS2_LS	5.7246	189.0630	2.1140	6.0979	-3.4530	39.9505	235.1114
100	**SSP_ML**	2.7385	0.2973	1.0647	0.0196	-1.8702	0.0665	**0.3835**
	SS1_ML	2.9109	1.1020	1.0716	0.0635	-1.9112	0.1157	1.2813
	SS2_ML	3.0300	14.8081	1.1771	0.1315	-1.9786	1.3586	16.2983
	SSP_LS	2.9116	0.3159	1.0463	0.0170	-1.9853	0.1098	0.4427
	SS1_LS	3.3758	5.7175	1.0838	0.1075	-2.0881	0.6062	6.4312
	SS2_LS	6.6015	253.7575	2.0297	5.3945	-3.6425	35.3657	294.5177
250	**SSP_ML**	2.7993	0.1199	1.0523	0.0081	-1.9283	0.0215	**0.1496**
	SS1_ML	2.9006	0.3569	1.0398	0.0216	-1.9505	0.0328	0.3569
	SS2_ML	2.9673	12.9489	1.1166	0.0587	-1.9987	1.1020	14.1097
	SSP_LS	2.8270	0.1342	1.0514	0.0084	-1.9458	0.0395	0.1821
	SS1_LS	3.3329	3.9782	1.0389	0.0561	-2.0954	0.4253	4.4597
	SS2_LS	7.3622	259.7597	2.0636	7.4142	-4.8842	64.1653	331.3391
500	**SSP_ML**	2.8716	0.0583	1.0319	0.0034	-1.9517	0.0109	**0.0728**
	SS1_ML	2.9383	0.1532	1.0220	0.0106	-1.9664	0.0175	0.1814
	SS2_ML	2.8774	2.7291	1.1009	0.0482	-1.9938	0.7305	3.5079
	SSP_LS	2.8762	0.0690	1.0327	0.0038	-1.9534	0.0206	0.0934
	SS1_LS	3.2780	2.5897	1.0209	0.0385	-2.0835	0.3227	2.9509
	SS2_LS	7.7825	289.9620	2.1047	7.0621	-5.1289	75.5625	372.5865
1000	**SSP_ML**	2.9316	0.02677	1.0169	0.0015	-1.9743	0.0045	**0.0328**
	SS1_ML	2.9559	0.0862	1.0145	0.0054	-1.9788	0.0088	0.1005
	SS2_ML	3.0109	6.9185	1.0835	0.0310	-2.0760	2.8762	9.8258
	SSP_LS	2.9351	0.0359	1.0166	0.0017	-1.9743	0.0107	0.0483
	SS1_LS	3.3312	2.8849	1.0073	0.0322	-2.1094	0.3209	3.2381
	SS2_LS	7.0331	241.5506	2.1044	6.2384	-4.7036	59.3074	307.0964

Çizelge 5.21. $pop = 10$ için parametreleri $\theta = (\sigma, \beta, \mu) = (4, 0.5, 10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	4.4927	6.7053	0.4729	0.0309	10.1617	0.3673	**7.1036**
	SS1_ML	2.9676	17.8085	1.2607	1.2206	10.0444	3.6257	22.6548
	SS2_ML	11.1550	319.2682	1.8011	4.8615	-0.9353	397.4822	721.6119
	SSP_LS	5.3598	6.9735	0.4910	0.0235	9.5633	0.8881	7.8851
	SS1_LS	2.8564	6.7200	1.1727	0.9955	9.6895	3.0230	10.7385
	SS2_LS	12.3061	480.4698	6.6717	121.3014	-8.6757	717.2849	1319.0561
30	**SSP_ML**	3.7075	1.8374	0.5353	0.0138	10.1215	0.1454	**1.9968**
	SS1_ML	3.5933	15.4422	0.8556	0.3576	9.7663	4.6108	20.41078
	SS2_ML	11.253	308.6472	1.6642	6.5920	-0.8054	389.1014	704.3407
	SSP_LS	4.3131	2.0663	0.5222	0.0096	9.8407	0.2614	2.3373
	SS1_LS	3.1978	6.2303	0.9400	0.4778	9.7678	2.0474	8.7555
	SS2_LS	11.1477	358.8691	6.2790	99.5005	-12.2756	899.8214	1358.1909
50	**SSP_ML**	3.5310	1.0042	0.5493	0.0099	10.1216	0.0823	1.0965
	SS1_ML	3.5576	12.0486	0.7767	0.1927	9.8686	2.3052	14.5466
	SS2_ML	11.0921	298.8128	1.5044	5.4673	-0.2465	369.8867	674.1669
	SSP_LS	3.8387	0.8791	0.5435	0.0073	9.9704	0.1160	**1.0024**
	SS1_LS	3.2123	5.7100	0.9441	0.5337	9.7317	1.9430	8.1867
	SS2_LS	12.2297	446.1462	6.0194	101.1597	-8.7481	737.6848	1284.9906
100	**SSP_ML**	3.5612	0.5612	0.5414	0.0050	10.0988	0.0381	0.6044
	SS1_ML	3.9300	13.3435	0.7165	0.1502	9.7671	2.6466	16.1404
	SS2_ML	10.8695	265.7512	1.4271	3.2553	-0.2312	362.0822	631.0888
	SSP_LS	3.6642	0.5171	0.5420	0.0044	10.0441	0.0586	**0.5801**
	SS1_LS	3.3446	5.2311	0.8681	0.3477	9.7217	1.7741	7.3529
	SS2_LS	12.3581	465.5964	6.2125	121.4982	-8.4911	719.5454	1306.6400
250	**SSP_ML**	3.7576	0.2075	0.5208	0.0015	10.0513	0.0138	**0.2230**
	SS1_ML	3.9972	11.4852	0.7153	0.1929	9.5811	4.4080	16.0862
	SS2_ML	10.7623	283.2713	1.3509	2.1354	0.07588	340.723	626.1298
	SSP_LS	3.7763	0.2252	0.5220	0.0015	10.0398	0.0245	0.2512
	SS1_LS	3.6330	5.3033	0.8368	0.3401	9.6270	1.8655	7.5089
	SS2_LS	11.3665	333.8431	5.9330	89.5090	-12.8759	927.8001	1351.1522
500	**SSP_ML**	3.8447	0.1145	0.5142	0.0008	10.0300	0.0067	**0.1222**
	SS1_ML	4.0225	10.4033	0.6983	0.2223	9.6520	3.1474	13.7731
	SS2_ML	11.9374	329.7972	1.3486	2.5385	-1.0968	404.5862	736.9218
	SSP_LS	3.8486	0.1256	0.5155	0.0007	10.0247	0.0131	0.1395
	SS1_LS	3.5365	4.9741	0.8676	0.3499	9.5582	2.2155	7.5395
	SS2_LS	10.9713	309.7106	5.9936	85.7278	-12.9568	962.6655	1358.1039
1000	**SSP_ML**	3.9552	0.0490	0.5037	0.0003	10.0101	0.0031	**0.0525**
	SS1_ML	4.1133	9.7414	0.6776	0.1691	9.6052	3.0781	12.9887
	SS2_ML	9.9975	223.1348	1.3682	2.4225	0.2815	336.5732	562.1306
	SSP_LS	3.9766	0.0602	0.5036	0.0002	10.0003	0.0070	0.0675
	SS1_LS	3.6416	4.8782	0.8607	0.3823	9.5207	2.1090	7.3695
	SS2_LS	11.7555	366.2718	6.4386	109.4302	-13.2351	948.0137	1423.7158

Çizelge 5.22. $pop = 25$ için parametreleri $\theta = (\sigma, \beta, \mu) = (4, 0.5, 10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

		$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
n	**Yöntem**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	4.3236	8.4890	0.5014	0.036	10.2192	0.4208	8.9463
	SS1_ML	1.8934	12.0569	1.3959	1.3241	10.5883	0.6502	14.0313
	SS2_ML	3.2666	75.1719	1.5312	1.5971	9.4402	29.0729	105.8421
	SSP_LS	5.3234	8.0966	0.5010	0.0308	9.6205	0.8489	8.9764
	SS1_LS	2.6049	5.9594	1.0289	0.7892	10.2932	0.5848	**7.3333**
	SS2_LS	8.5574	333.5715	5.5630	119.0918	2.4118	263.4666	716.1299
30	**SSP_ML**	3.4952	2.1289	0.5595	0.0190	10.1714	0.1568	**2.3048**
	SS1_ML	2.9972	6.6971	0.7622	0.1864	10.2651	0.2802	7.1638
	SS2_ML	4.1640	78.4705	1.0280	0.5166	9.1412	28.4627	107.4544
	SSP_LS	4.1747	2.1961	0.5310	0.0134	9.9012	0.2386	2.4480
	SS1_LS	2.7366	4.5384	0.7907	0.1807	10.2745	0.3122	5.0313
	SS2_LS	8.8231	280.2917	4.7460	74.0181	-0.8255	375.4819	729.7918
50	**SSP_ML**	3.3617	1.2959	0.5689	0.0140	10.1615	0.0956	1.4057
	SS1_ML	3.0441	4.7673	0.6879	0.0883	10.2237	0.1930	5.04875
	SS2_ML	4.6207	89.5504	0.9251	0.3488	9.0328	27.4662	117.3655
	SSP_LS	3.7785	1.0971	0.5480	0.0096	10.0017	0.1240	**1.2306**
	SS1_LS	2.9883	3.8932	0.7219	0.1146	10.2106	0.2675	4.2753
	SS2_LS	8.8556	338.8390	4.2220	75.1658	2.4458	260.7591	674.7639
100	**SSP_ML**	3.5018	0.7172	0.5491	0.0064	10.1111	0.0422	0.7172
	SS1_ML	3.2427	2.9336	0.6238	0.0388	10.1610	0.1059	3.0784
	SS2_ML	4.5396	85.4658	0.8147	0.2051	9.1501	24.4157	110.0868
	SSP_LS	3.6452	0.5737	0.5424	0.0052	10.0578	0.0579	**0.6369**
	SS1_LS	3.0900	3.1419	0.6774	0.0741	10.1776	0.1891	3.4051
	SS2_LS	8.9699	329.9158	3.8824	43.5022	2.7742	227.4570	600.8750
250	**SSP_ML**	3.7249	0.2483	0.5243	0.0020	10.0581	0.0152	**0.2656**
	SS1_ML	3.3996	1.6394	0.5831	0.01836	10.1176	0.0573	1.7152
	SS2_ML	4.6783	69.9571	0.7591	0.1640	8.9209	28.6090	98.7302
	SSP_LS	3.7555	0.2492	0.5229	0.0017	10.0495	0.0242	0.2752
	SS1_LS	3.3399	2.5821	0.6300	0.0482	10.1170	0.1309	2.7611
	SS2_LS	9.6891	347.1285	4.5036	60.4417	-0.1056	334.9611	742.5313
500	**SSP_ML**	3.8337	0.1268	0.5152	0.0009	10.0324	0.0072	**0.1350**
	SS1_ML	3.4737	1.3998	0.5692	0.0116	10.0970	0.0463	1.4578
	SS2_ML	5.0636	83.8007	0.7383	0.1671	8.7998	30.8414	114.8093
	SSP_LS	3.8474	0.1332	0.5150	0.0008	10.0279	0.0127	0.1468
	SS1_LS	3.4531	2.4743	0.6183	0.0423	10.0766	0.1374	2.6539
	SS2_LS	10.1780	353.7534	4.1843	57.8931	-1.0371	397.2929	808.9394
1000	**SSP_ML**	3.9431	0.0559	0.5048	0.0003	10.0124	0.0034	**0.0597**
	SS1_ML	3.5021	0.7336	0.5578	0.0078	10.0886	0.0249	0.7664
	SS2_ML	4.8574	69.9133	0.7310	0.1649	8.8523	29.7498	99.8281
	SSP_LS	3.9699	0.0699	0.5039	0.0003	10.0035	0.0070	0.0772
	SS1_LS	3.5570	2.2414	0.6052	0.0399	10.0488	0.1247	2.4061
	SS2_LS	10.2309	327.2825	3.9128	46.5160	-1.7804	410.7385	784.5371

Çizelge 5.23. $pop = 50$ için parametreleri $\theta = (\sigma, \beta, \mu) = (4, 0.5, 10)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	4.2683	9.1652	0.5131	0.0395	10.2371	0.4414	9.6462
	SS1_ML	2.0592	12.3012	1.3890	1.3435	10.5647	0.6424	14.2872
	SS2_ML	1.5514	17.4401	1.5498	1.5544	10.5852	1.5447	20.5392
	SSP_LS	5.3369	9.2060	0.5087	0.0355	9.6356	0.8885	10.1299
	SS1_LS	2.7455	5.3813	0.9579	0.6875	10.2874	0.5161	**6.5850**
	SS2_LS	5.1117	167.9172	4.5674	87.4712	7.5098	91.2558	346.6442
30	**SSP_ML**	3.4435	2.2750	0.5673	0.0210	10.1831	0.1639	**2.4600**
	SS1_ML	3.1289	5.7201	0.7248	0.1661	10.2510	0.2466	6.1329
	SS2_ML	2.8543	17.3835	0.9123	0.3210	10.1711	1.9764	19.6809
	SSP_LS	4.1580	2.5269	0.5374	0.0159	9.9092	0.2488	2.7916
	SS1_LS	2.8942	3.9780	0.7349	0.1350	10.2719	0.2396	4.3526
	SS2_LS	6.8806	251.1246	4.1015	65.9729	6.0875	108.5129	425.6103
50	**SSP_ML**	3.3449	1.3629	0.5731	0.0150	10.1667	0.0990	1.4770
	SS1_ML	3.1075	3.3861	0.6546	0.0651	10.2183	0.1484	3.5997
	SS2_ML	2.8689	13.2331	0.8136	0.1861	10.1837	0.9180	14.3374
	SSP_LS	3.7505	1.2272	0.5530	0.0113	10.0114	0.1276	**1.3662**
	SS1_LS	2.9810	3.4078	0.6900	0.0811	10.2375	0.1966	3.6856
	SS2_LS	5.4397	166.3668	2.9944	26.5011	7.8219	66.6044	259.4723
100	**SSP_ML**	3.4897	0.6931	0.5507	0.0068	10.1134	0.0433	0.7433
	SS1_ML	3.3902	2.0872	0.5885	0.0251	10.1443	0.08131	2.1936
	SS2_ML	3.0705	12.1302	0.7413	0.1235	10.1258	1.0299	13.2837
	SSP_LS	3.6350	0.6271	0.5443	0.0059	10.0620	0.0597	**0.6927**
	SS1_LS	3.1200	2.4466	0.6346	0.0434	10.2113	0.1353	2.6253
	SS2_LS	5.2318	151.6501	2.8139	24.9800	8.2256	47.3678	223.9978
250	**SSP_ML**	3.7187	0.2564	0.5250	0.0020	10.0593	0.0156	**0.2742**
	SS1_ML	3.5323	1.0121	0.5555	0.0101	10.0989	0.0387	1.0610
	SS2_ML	3.2166	6.2031	0.6667	0.06084	10.0771	0.6837	6.9477
	SSP_LS	3.7413	0.2735	0.5244	0.0020	10.0533	0.0248	0.3003
	SS1_LS	3.3449	2.0583	0.5943	0.0242	10.1601	0.0936	2.1761
	SS2_LS	7.2647	242.9537	2.9801	22.7486	5.3469	142.4554	408.1577
500	**SSP_ML**	3.8315	0.1298	0.5155	0.0009	10.0328	0.0073	**0.1381**
	SS1_ML	3.5890	0.6914	0.5472	0.0067	10.0791	0.0259	0.7241
	SS2_ML	3.2630	4.9462	0.6421	0.04873	10.0961	0.3291	5.3241
	SSP_LS	3.8459	0.1501	0.5155	0.0009	10.0280	0.0136	0.1647
	SS1_LS	3.5045	1.7132	0.5734	0.0164	10.1136	0.0783	1.8079
	SS2_LS	8.5492	337.4011	2.9457	27.7306	5.1378	139.5766	504.7083
1000	**SSP_ML**	3.9408	0.0570	0.5050	0.0003	10.0129	0.0034	**0.0609**
	SS1_ML	3.6291	0.4379	0.5391	0.0040	10.0702	0.01640	0.4583
	SS2_ML	3.2683	2.7705	0.6251	0.0391	10.1020	0.101541	2.7705
	SSP_LS	3.9646	0.0768	0.5044	0.0004	10.0048	0.0072	0.0844
	SS1_LS	3.5715	1.3889	0.5644	0.0133	10.0867	0.0628	1.4651
	SS2_LS	7.5491	243.1289	3.0848	36.1980	5.3303	136.3073	415.6343

Çizelge 5.24. $pop = 10$ için parametreleri $\theta = (\sigma, \beta, \mu) = (2.56, 0.75, 8)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	3.7809	7.8255	0.6332	0.0673	7.9465	0.4281	**8.3210**
	SS1_ML	2.6556	15.9085	1.5197	1.3546	7.8879	3.3865	20.6497
	SS2_ML	13.0416	530.8435	6.5157	126.7316	-10.3124	684.8742	1342.449
	SSP_LS	5.0851	11.7824	0.614522	0.0570	7.0941	1.8480	13.6875
	SS1_LS	4.4148	24.8730	1.8853	3.7670	5.2944	27.1118	55.7519
	SS2_LS	12.4602	513.540	6.5240	127.3491	-9.41877	651.4303	1292.32
30	**SSP_ML**	2.8026	0.5396	0.6999	0.0186	8.0349	0.0567	**0.6150**
	SS1_ML	3.1277	12.9576	1.0299	0.3168	7.7246	2.5073	15.7819
	SS2_ML	11.0555	333.3878	1.6511	3.9034	-2.5283	375.5359	712.8272
	SSP_LS	3.1224	0.8023	0.7014	0.0174	7.7921	0.1799	0.9996
	SS1_LS	4.6547	26.5247	1.4689	1.6912	5.2650	27.4967	55.7126
	SS2_LS	10.8520	343.2408	6.0828	110.6547	-13.3322	846.8889	1300.7845
50	**SSP_ML**	2.6326	0.4722	0.7460	0.0141	8.0412	0.0488	**0.5352**
	SS1_ML	3.2429	11.8193	0.9713	0.2713	7.6195	2.9579	15.0486
	SS2_ML	11.0309	332.7216	1.5581	3.3208	-1.90314	336.1126	672.1551
	SSP_LS	3.0319	0.7745	0.7201	0.0107	7.8191	0.1475	0.9327
	SS1_LS	4.7735	27.6503	1.4198	1.4677	5.2535	26.7401	55.8581
	SS2_LS	11.7158	441.8068	5.4833	75.6122	-9.4618	647.0782	1164.4972
100	**SSP_ML**	2.4995	0.0793	0.7672	0.0053	8.0469	0.0125	0.09731
	SS1_ML	3.3625	12.7229	0.9061	0.1880	7.5609	3.3879	16.2989
	SS2_ML	10.7708	316.6808	1.5829	3.7033	-2.6695	379.1355	699.5196
	SSP_LS	2.5638	0.0584	0.7694	0.0056	7.9946	0.0198	**0.0839**
	SS1_LS	4.7279	25.7711	1.3796	1.4570	5.2982	26.7773	54.0055
	SS2_LS	11.7170	441.7485	5.8468	102.3679	-9.4784	646.5951	1190.7115
250	**SSP_ML**	2.4912	0.0575	0.7737	0.0030	8.0318	0.0057	0.0663
	SS1_ML	3.4480	13.0785	0.8621	0.1109	7.5427	2.9737	16.1631
	SS2_ML	11.14087	338.5419	1.5376	4.8033	-2.51524	365.159	708.5043
	SSP_LS	2.5116	0.0522	0.7746	0.0028	8.0164	0.0098	**0.0648**
	SS1_LS	4.8735	27.0486	1.3414	1.3300	5.2525	26.6252	55.0038
	SS2_LS	10.7525	338.3633	5.8998	95.2945	-13.4803	844.3087	1277.9664
500	**SSP_ML**	2.5296	0.0308	0.7658	0.0014	8.0187	0.0028	**0.0351**
	SS1_ML	3.3828	10.3804	0.8375	0.1095	7.5703	2.5217	13.0116
	SS2_ML	11.2535	369.99	1.4355	2.4073	-2.4308	368.8813	741.2870
	SSP_LS	2.5535	0.0298	0.7640	0.0013	8.0068	0.0061	0.0372
	SS1_LS	4.8373	25.7278	1.3230	1.1450	5.2124	26.3145	53.1874
	SS2_LS	10.8576	335.1701	5.9294	96.0040	-13.1429	828.6418	1259.8158
1000	**SSP_ML**	2.5587	0.0160	0.7589	0.0007	8.0118	0.0013	**0.01815**
	SS1_ML	3.3996	010.1453	0.8314	0.1074	7.5603	2.6359	12.8887
	SS2_ML	10.9587	334.4893	1.5165	4.0428	-2.2191	354.4369	692.9637
	SSP_LS	2.5666	0.0174	0.7584	0.0006	8.0078	0.0031	0.0211
	SS1_LS	4.8971	26.7149	1.3392	1.3551	5.1878	27.2078	55.2777
	SS2_LS	10.7706	337.3329	5.8955	97.0441	-13.1495	827.6200	1261.9970

Çizelge 5.25. $pop = 25$ için parametreleri $\theta = (\sigma, \beta, \mu) = (2.56, 0.75, 8)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

		$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
n	**Yöntem**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	3.2213	7.8948	0.7138	0.0615	8.1012	0.4388	**8.3951**
	SS1_ML	1.5156	6.4465	1.7864	1.8210	8.4117	0.4597	8.7274
	SS2_ML	3.0231	58.2342	1.8345	1.9019	7.1528	27.9849	88.1210
	SSP_LS	4.8160	12.7239	0.6559	0.0631	7.2409	1.6786	14.4656
	SS1_LS	2.7116	9.3748	1.3902	1.4248	7.9054	1.2715	12.0711
	SS2_LS	5.0092	200.6225	4.6330	81.3634	6.1252	53.3122	335.2982
30	**SSP_ML**	2.6957	0.5435	0.7177	0.0186	8.0675	0.0581	**0.6203**
	SS1_ML	2.2893	3.2782	1.0215	0.2746	8.1479	0.1658	3.7187
	SS2_ML	4.0812	89.1004	1.2096	0.4759	6.9153	28.4666	118.043
	SSP_LS	3.0385	0.8608	0.7156	0.0200	7.8307	0.1638	1.0446
	SS1_LS	2.9435	10.5749	1.1391	0.5477	7.7234	2.6769	13.7995
	SS2_LS	7.0028	264.3197	3.5858	49.3927	3.8524	120.6242	434.3366
50	**SSP_ML**	2.4833	0.4581	0.7738	0.0156	8.0823	0.0467	**0.5205**
	SS1_ML	2.3215	1.5162	0.9160	0.1202	8.1114	0.0821	1.7186
	SS2_ML	4.2731	85.8568	1.0870	0.2848	6.6770	32.4816	118.6233
	SSP_LS	2.9340	0.7428	0.7332	0.0135	7.8671	0.1209	0.8772
	SS1_LS	3.0642	10.4554	1.0741	0.4427	7.6395	2.7570	13.6551
	SS2_LS	8.5323	329.8226	3.5367	36.6483	1.6463	192.3202	558.7911
100	**SSP_ML**	2.4758	0.0937	0.7720	0.0061	8.0547	0.0131	0.1131
	SS1_ML	2.3935	0.7622	0.8405	0.0435	8.0688	0.0372	0.8430
	SS2_ML	4.0214	74.7848	1.0036	0.1822	6.9634	22.8519	97.8190
	SSP_LS	2.5601	0.0737	0.7685	0.0065	8.0024	0.0193	**0.0995**
	SS1_LS	3.2307	11.1256	0.9765	0.2974	7.6192	2.6028	14.0258
	SS2_LS	8.1413	286.0229	3.6583	44.3889	1.8357	185.2244	515.6362
250	**SSP_ML**	2.4794	0.0677	0.7764	0.0034	8.0353	0.0064	0.0776
	SS1_ML	2.4537	0.3922	0.8036	0.0164	8.0382	0.0214	0.4300
	SS2_ML	4.0328	73.4884	0.9312	0.1048	6.8150	32.9088	106.5021
	SSP_LS	2.5126	0.0596	0.7736	0.0032	8.0196	0.0095	**0.0723**
	SS1_LS	3.3395	10.4881	0.9539	0.3893	7.5744	2.6145	13.4920
	SS2_LS	9.5241	343.5391	4.1394	66.7236	-2.1932	343.7817	754.0444
500	**SSP_ML**	2.5187	0.0351	0.7680	0.0017	8.0217	0.0029	**0.0397**
	SS1_ML	2.4714	0.4105	0.7909	0.0095	8.0317	0.01942	0.4396
	SS2_ML	4.3226	93.7741	0.9101	0.1215	6.9341	22.8579	116.7536
	SSP_LS	2.5480	0.0346	0.7642	0.0016	8.0114	0.0059	0.0421
	SS1_LS	3.3428	10.1838	0.9380	0.2983	7.5713	2.3045	12.7866
	SS2_LS	9.9664	351.4923	3.6560	42.3463	-2.5675	349.9287	743.7673
1000	**SSP_ML**	2.5509	0.0173	0.7601	0.0007	8.0142	0.0013	**0.0195**
	SS1_ML	2.4914	0.1088	0.7795	0.0053	8.0262	0.0055	0.1198
	SS2_ML	4.0531	70.6981	0.8856	0.0891	6.9009	25.7387	96.5267
	SSP_LS	2.5654	0.0198	0.7583	0.0007	8.0092	0.0031	0.0237
	SS1_LS	3.3702	9.4645	0.9322	0.3697	7.5570	2.2224	12.0566
	SS2_LS	9.7969	351.3180	4.0083	57.9936	-2.2125	337.2107	746.5223

Çizelge 5.26. $pop = 50$ için parametreleri $\theta = (\sigma, \beta, \mu) = (2.56, 0.75, 8)$ olan 3-p Gamma dağılımı için simülasyon sonucunda elde edilen ortalama parametre tahmini değerleri, MSE ve Def değerleri

n	Yöntem	$\hat{\sigma}$		$\hat{\beta}$		$\hat{\mu}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	**SSP_ML**	3.1279	9.2329	0.7495	0.0650	8.15072	0.5079	9.8059
	SS1_ML	1.5355	7.1735	1.8501	1.9754	8.4207	0.4459	**9.5948**
	SS2_ML	1.4821	15.9981	1.9248	2.0580	8.3486	1.8519	19.9082
	SSP_LS	4.8225	12.6982	0.6552	0.0620	7.2393	1.6450	14.4052
	SS1_LS	2.6733	8.7975	1.4083	1.4035	7.9239	1.1689	11.3698
	SS2_LS	5.6894	262.1404	4.4139	60.5915	5.8439	62.7841	385.5159
30	**SSP_ML**	2.3751	0.6331	0.7802	0.01831	8.1369	0.0839	**0.7354**
	SS1_ML	2.3001	2.6873	1.0011	0.2861	8.1567	0.1367	3.0922
	SS2_ML	2.1957	9.7988	1.1591	0.3827	8.1245	0.5877	9.7988
	SSP_LS	3.0579	0.9761	0.7168	0.0219	7.8276	0.1690	1.1670
	SS1_LS	2.6126	6.7054	1.0275	0.2962	8.0268	0.5004	7.5020
	SS2_LS	6.8505	263.5265	3.4646	46.1029	4.0231	122.3993	432.0287
50	**SSP_ML**	2.4573	0.4694	0.7805	0.0164	8.0882	0.0474	**0.5333**
	SS1_ML	2.3858	1.8817	0.8917	0.1064	8.1053	0.0860	2.0742
	SS2_ML	2.3464	11.0274	1.0355	0.20514	8.0683	0.9778	12.2104
	SSP_LS	2.9005	0.8080	0.7423	0.0155	7.8799	0.1227	0.9462
	SS1_LS	2.6290	5.6831	0.9679	0.1805	7.9976	0.5036	6.3672
	SS2_LS	5.4041	177.1065	2.5312	14.1883	5.8558	47.2345	238.5293
100	**SSP_ML**	2.4753	0.0947	0.7722	0.0062	8.0548	0.0131	0.1141
	SS1_ML	2.4240	0.4930	0.8194	0.0345	8.0674	0.02801	0.5555
	SS2_ML	2.5523	9.6974	0.9308	0.1081	7.988	0.7037	10.5093
	SSP_LS	2.5587	0.0806	0.7692	0.0068	8.0035	0.0200	**0.1073**
	SS1_LS	2.7013	5.5013	0.9165	0.1350	7.9880	0.3356	5.9720
	SS2_LS	5.4807	186.1740	2.4491	17.8620	6.1878	40.8659	244.9019
250	**SSP_ML**	2.4780	0.0686	0.7767	0.0035	8.0355	0.0064	**0.0786**
	SS1_ML	2.4750	0.1706	0.7859	0.0105	8.0377	0.0098	1.1911
	SS2_ML	2.3214	1.0703	0.8753	0.0456	8.0336	0.1808	1.2987
	SSP_LS	2.5088	0.0670	0.7750	0.0035	8.0206	0.0099	0.0804
	SS1_LS	2.7496	4.3696	0.8596	0.0817	7.9559	0.2698	4.7210
	SS2_LS	6.2232	203.8677	2.8442	26.2443	4.2665	100.9938	331.1058
500	**SSP_ML**	2.5180	0.0354	0.7683	0.0017	8.0218	0.0029	**0.0401**
	SS1_ML	2.5088	0.0966	0.7748	0.0059	8.0259	0.0048	0.1074
	SS2_ML	2.6461	21.0011	0.8529	0.03401	7.8962	3.8415	24.8766
	SSP_LS	2.5482	0.0374	0.7644	0.0017	8.0114	0.0061	0.0451
	SS1_LS	2.8341	4.2734	0.8312	0.0648	7.9349	0.2437	4.5819
	SS2_LS	6.9904	253.8847	2.8179	19.0555	3.9053	121.2519	394.1921
1000	**SSP_ML**	2.5505	0.0175	0.7602	0.0008	8.0142	0.0013	**0.0197**
	SS1_ML	2.5262	0.0571	0.7678	0.0033	8.0213	0.0028	0.0633
	SS2_ML	2.50836	5.0090	0.8324	0.02712	7.9489	2.2257	7.2619
	SSP_LS	2.5657	0.0215	0.7583	0.0008	8.0092	0.0032	0.0255
	SS1_LS	2.9471	4.8092	0.8151	0.0656	7.8984	0.2945	5.1692
	SS2_LS	7.2016	258.1254	2.6375	16.7555	3.8897	115.2599	390.1408

Çizelge 5.18-5.26'dan, SSP ve SS1 tüm durumları için SS2'den daha iyi parametre tahmini değerleri verdiği görülmüştür. SS2 arama uzayı oldukça geniş olduğundan güvenilir olmayan sonuçlar sergilemiştir. Ayrıca, Def kriterine göre SSP çoğu durumda SS1, ML ve LS yöntemlerine göre daha iyi sonuçlar vermiştir. Örneklem sayısı arttıkça, SSP'nin performansında kayda değer bir artış görülmüştür. Sonuç olarak,

3-p Gamma dağılımının parametrelerinin tahmininde SSP'nin LS için $\theta = (4, 0.5, 10)$, $n = 10$, $pop = 25$ ve $pop = 50$ durumları, ML ve LS için ise $\theta = (2.56, 0.75, 8)$, $n = 10$, $pop = 25$ durumları haricinde daha uygun olduğu sonucuna ulaşılmıştır.

Popülasyon sayısı arttıkça, algoritmaların performanslarında değişiklik olup olmadığını incelemek için Kruskal-Wallis testi yapılmıştır. %95 güven seviyesinde SSP arama uzayında hem ML hem de LS yöntemi için de anlamlı farklılık yoktur (p-değerleri sırasıyla 0.865 ve 0.869). SS1 ve SS2 arama uzaylarında anlamlı farklılık vardır (p-değerleri SS1_ML, SS1_LS, SS2_ML ve SS2_LS için 0.000). SS1'de popülasyon büyüklükleri $pop = 10$ ile $pop = 25$ arasında ve $pop = 10$ ile $pop = 50$ arasında, SS2'de ise tüm popülasyon büyüklükleri arasında çözüm kalitesi açısından anlamlı farklılık olduğu tespit edilmiştir.

SSP kullanılarak elde edilen ML ve LS tahminleri arasında anlamlı farklılık olup olmadığını test etmek için Mann-Whitney *U* testi kullanılmıştır. Örneklem büyüklükleri 10, 30, $50, 100, 250, 500$ ve 1000 için p-değerleri sırasıyla 0.004, 0.136, 0.258, 0.863, 0.666, 0.340 ve 0.258 olarak elde edilmiş olup, yalnızca $n = 10$ için ML ve LS tahminleri arasında istatistiksel olarak anlamlı farklılık bulunmuştur.

5.3.4. Gerçek veri uygulaması

Bu alt kesimde, gerçek veri kümesi üzerinde önerilen yöntemin performansı araştırılmıştır. Ele alınan veri kümesi Crowder ve ark. (1991) tarafından literatüre kazandırılmış olup karbon liflerinin tek tek yenilme gerilmelerini içermektedir ve Çizelge 5.27'de verilmiştir.

Çizelge 5.27. Karbon lifi veri kümesi

2.247	2.640	2.908	3.099	3.126	3.245	3.328	3.355
3.383	3.572	3.581	3.681	3.726	3.727	3.72	3.783
3.785	3.786	3.896	3.912	3.964	4.050	4.063	4.082
4.111	4.118	4.141	4.246	4.251	4.26;	4.326;	4.402
4.457	4.466	4.519	4.542	4.555	4.614	4.632	4.634
4.636	4.678	4.698	4.738	4.83	4.924	5.043	5.099
5.134	5.359	5.473	5.571	5.684	5.721	5.998	6.060

Parametre tahmin değerleri

MML tahminlerine dayalı olarak, 3-p Gamma dağılımının parametreleri için $\alpha = 0.05$ anlam seviyesindeki asimptotik güven aralıkları,

$6.5998 < \sigma < 12.7820$

$0.2315 < \beta < 0.3391$

ve

$1.0188 < \mu < 1.9463$

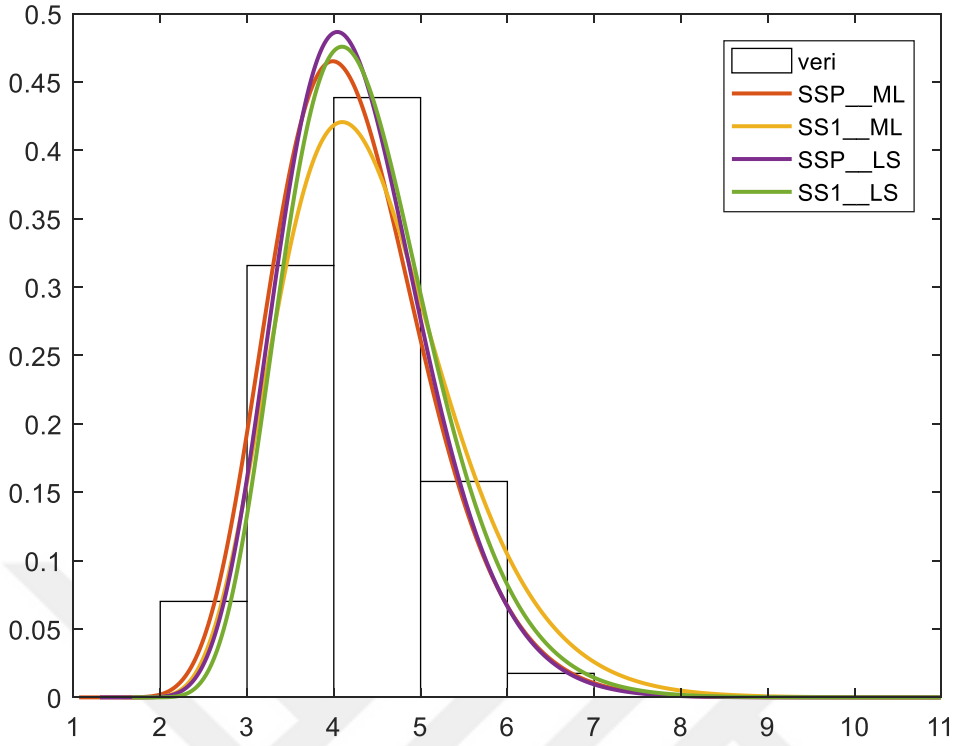
olarak elde edilmiştir. Ele alınan yöntemler için parametre tahmin değerleri, $\ln L$ değerleri, K-S test istatistiği değerleri, *p* değerleri, *AIC* ve *BIC* değerleri Çizelge 5.28'de verilmiştir.

Çizelge 5.28. Karbon lifi veri kümesi için parametre tahmin değerleri, K-S testine ilişkin değerler, $\ln L$, *AIC* ve *BIC* değerleri

	$\hat{\sigma}$	$\hat{\beta}$	$\hat{\mu}$	**K-S**	**p**	$\ln L$	**AIC**	**BIC**
SSP_ML	12.7813	0.2481	1.0648	0.0750	0.9013	-0.0441	6.0882	10.11355
SS1_ML	8.2790	0.3475	1.5679	0.1344	0.2544	-0.1765	6.3530	10.37835
SS2_ML	5.9780	0.5236	1.7295	0.2731	0.0004	-1.3667	8.7334	12.75875
SSP_LS	12.3016	0.2421	1.3047	0.0620	0.9804	**-0.0255**	6.0510	10.07635
SS1_LS	9.4874	0.2850	1.6761	0.0980	0.6370	-0.0900	6.1800	10.20535
SS2_LS	8.1813	3.2249	-1.5607	0.9976	0.0000	-18.7883	43.5766	47.60195

Çizelge 5.28'e göre, p-değerleri $\alpha = 0.05$'dan büyük olduğundan karbon lifleri verisinin dağılımı SS2_ML ve SS2_LS haricindeki ele alınan yöntemler ile elde edilen parametre değerleriyle 3-p gamma dağılımına uygundur. SS2_ML ve SS2_LS ile elde edilen parametre değerleri K-S kriteri açısından Karbon lifleri veri kümesini modellemede başarısız olmuştur. SSP' ler önerilen yöntemler arasında en büyük $\ln L$ ve en küçük *AIC* ve *BIC* değerlerine sahiptir. Böylece, önerilen DE yaklaşımı ele alınan karbon lifleri verisi için hem ML hem de LS parametre tahmininde diğer yöntemlere göre daha iyi performans sergilemiştir.

Şekil 5.10'da veriye ilişkin histogram ve kullanılan yöntemlerden elde edilen parametre değerleri kullanılarak çizilen uygun olasılık yoğunluk fonksiyonları verilmiştir. Çizelge 5.28'de verilen sonuçların Şekil 5.10 ile desteklendiği görülmüştür.



Şekil 5.10. Karbon lifi veri kümesi için histogram ve olasılık yoğunluk fonksiyonu grafiği

5.4. Kesikli Weibull Dağılımının Parametrelerinin Metasezgisel Yöntemler ile Tahmini

Bu kesimde, kesikli Weibull dağılımının parametrelerinin ML yöntemi ile tahminine yer verilmiştir. Bu dağılımın ML parametre tahmini için metasezgisel yöntemlerden SA, DE, PSO, ABC ve GA kullanılmıştır. Algoritmalar Monte-Carlo simülasyon çalışmaları ile algoritmalar çözüm kalitesi açısından karşılaştırılmış ve gerçek bir veri kümesi uygulaması sunulmuştur.

5.4.1. Kesikli Weibull dağılımlarının parametrelerinin tahmini için ML yöntemi

$X_1, X_2, ..., X_n$'ler olasılık yoğunluk fonksiyonu Eşitlik (3.5)'te verilen $\mathrm{DW}(\beta, q)$ dağılımından gelen rasgele değişkenler olmak üzere, olabilirlik (L) ve log-olabilirlik ($\ln L$) fonksiyonları sırasıyla

$$L = \prod_{i=1}^{n} q^{(x_i)^\beta} - q^{(x_i+1)^\beta} \tag{5.34}$$

ve

$$\ln L = \sum_{i=1}^{n} \ln\left(q^{(x_i)^\beta} - q^{(x_i+1)^\beta}\right) \tag{5.35}$$

olarak verilir. Eşitlik (5.35)'ten olabilirlik denklemleri,

$$\frac{\partial \ln L}{\partial q} = \sum_{i=1}^{n} \frac{q^{(x_i)^\beta}\beta(x_i)q^{-1} - q^{(x_i+1)^\beta}\beta\left(x_i+1\right)q^{-1}}{q^{(x_i)^\beta} - q^{(x_i+1)^\beta}} = 0 \tag{5.36}$$

$$\frac{\partial \ln L}{\partial \beta} = \sum_{i=1}^{n} \frac{q^{(x_i)^\beta}\ln(q^{x_i}) - q^{(x_i+1)^\beta}\ln(q^{(x_i+1)})}{q^{(x_i)^\beta} - q^{(x_i+1)^\beta}} = 0 \tag{5.37}$$

biçiminde elde edilir.

Olabilirlik denklemleri lineer olmayan fonksiyonlar içerdiğinden çözümü analitik olarak elde edilemez. Bundan dolayı, bu çalışmada metasezgisel yöntemlerin kullanılması önerilmiştir.

5.4.2. Simülasyon çalışması

Bu alt kesimde, kesikli Weibull dağılımının parametrelerinin ML tahmini için Monte Carlo simülasyon sonuçları sunulmuştur. Önerilen metasezgisel yöntemlerin SA, GA, PSO, DE ve ABC performansları çözüm kalitesi açısından karşılaştırılmıştır. Çözüm kalitesini değerlendirmek amacıyla Def kriteri kullanılmıştır.

Önerilen metasezgisel yöntemlerin performanslarının değerlendirilmesi için, simülasyon çalışmasında kullanılan kesikli Weibull dağılımları için ele alınan gerçek parametre değerleri Çizelge 5.29'da verilmiştir. Ayrıca, parametre tahmin sürecinde örneklem büyüklüğünün etkisini araştırmak için her bir parametre değeri için 20, 50, 100 ve 500 büyüklüğünde örneklemler göz önüne alınmıştır.

Çizelge 5.29. Simülasyon çalışması için ele alınan kesikli Weibull dağılımlarına ilişkin parametreler

β	0.5	0.8	1	1.5	2
q	0.3	0.5	0.7	0.9	0.95

Metasezgisel yöntemlerin parametre değerleri çeşitli denemeler ve literatür araştırmaları sonucu belirlenmiş olup Çizelge 5.30'da verilmiştir. Arama uzayı, ele alınan tüm yöntemlerde, Weibull dağılımının parametresi q için $[0,1]$ ve β için $[0,10]$ olarak dikkate alınmıştır. Metasezgisel algoritmalar hızlı bir yakınsama gösterdiğinden maksimum iterasyon sayısı 100 olarak alınmıştır.

Çizelge 5.30. Ele alınan metasezgisel algoritmalara ilişkin parametreler

Algoritmalar	Parametreler
SA	$s=10$, $T_0=500$, $\tau=0.85$, $\text{nMove}=5$
GA	NP=50 , $p_m=0.1$, $p_c=0.8$, $m=0.1$
PSO	NP=50 , $w=0.7$, $c_1=c_2=2$
DE	NP=50 , $F\in 0.8$, $C_r=0.2$
ABC	NP=50 , $l=5$

Tahmin stratejisinin güvenirliğini arttırmak için her bir algoritma 10000 kez çalıştırılmış ve simülasyon sonuçlarına ilişkin ortalama, MSE ve Def değerleri Çizelge 5.31-5.35'de verilmiştir.

Çizelge 5.31. Parametreleri $\theta = (\beta, q) = (0.5, 0.3)$ olan Kesikli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

n	**Yöntem**	$\hat{\beta}$		$\hat{q}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	0.9134	2.7889	0.3145	0.0139	2.8029
	GA	0.7179	0.5186	0.3361	0.0106	0.5293
	PSO	0.6761	0.6185	0.3076	0.0106	0.6292
	DE	0.6763	0.6184	0.3079	0.0106	0.6291
	ABC	0.6770	0.6476	0.3079	0.0106	0.6582
50	**SA**	0.8294	2.7913	0.3068	0.0083	2.7996
	GA	0.5775	0.0216	0.3277	0.0048	0.0263
	PSO	0.5265	0.0140	0.2993	0.0040	0.0180
	DE	0.5267	0.0139	0.2993	0.0040	0.0179
	ABC	0.5267	0.0139	0.2993	0.0040	0.0179
100	**SA**	0.6735	1.3950	0.3069	0.0050	1.4000
	GA	0.5665	0.0118	0.3309	0.0035	0.0153
	PSO	0.5201	0.0057	0.3033	0.0020	0.0077
	DE	0.5200	0.0057	0.3032	0.0020	0.0077
	ABC	0.5200	0.0057	0.3032	0.0020	0.0077
500	**SA**	0.9112	3.6733	0.3109	0.0055	3.6788
	GA	0.5485	0.0054	0.3286	0.0019	0.0073
	PSO	0.5047	0.0010	0.3011	0.0004	0.0014
	DE	0.5048	0.0010	0.3011	0.0004	0.0014
	ABC	0.5048	0.0010	0.3011	0.0004	0.0014

Çizelge 5.32. Parametreleri $\theta = (\beta, q) = (0.8, 0.5)$ olan Kesikli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

n	Yöntem	$\hat{\beta}$		$\hat{q}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	0.9790	0.8633	0.5061	0.0130	0.8764
	GA	0.9070	0.0641	0.5146	0.0112	0.0754
	PSO	0.8895	0.0636	0.5070	0.0124	0.0761
	DE	0.8901	0.0636	0.5072	0.0124	0.0761
	ABC	0.8902	0.0636	0.5073	0.0124	0.0761
50	**SA**	1.0284	1.7234	0.5019	0.0068	1.7302
	GA	0.8487	0.0199	0.5097	0.0047	0.0246
	PSO	0.8342	0.0196	0.5030	0.0050	0.0247
	DE	0.8340	0.0196	0.5030	0.0050	0.0246
	ABC	0.8341	0.0196	0.5030	0.0050	0.0246
100	**SA**	1.0462	2.0270	0.5013	0.0045	2.0314
	GA	0.8317	0.0093	0.5076	0.0022	0.0115
	PSO	0.8176	0.0087	0.5016	0.0023	0.0110
	DE	0.8177	0.0087	0.5016	0.0023	0.0110
	ABC	0.8177	0.0087	0.5016	0.0023	0.0110
500	**SA**	1.1188	2.7822	0.5009	0.0032	2.7854
	GA	0.8171	0.0022	0.5067	0.0006	0.0028
	PSO	0.8035	0.0015	0.5007	0.0004	0.0019
	DE	0.8034	0.0015	0.5008	0.0004	0.0019
	ABC	0.8035	0.0015	0.5008	0.0004	0.0019

Çizelge 5.33. Parametreleri $\theta = (\beta, q) = (1, 0.7)$ olan Kesikli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

n	**Yöntem**	$\hat{\beta}$		$\hat{q}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	1.1983	0.9595	0.7118	0.0101	0.9695
	GA	1.0940	0.0650	0.7110	0.0086	0.0736
	PSO	1.0942	0.0659	0.7136	0.0091	0.0750
	DE	1.0944	0.0659	0.7136	0.0091	0.0750
	ABC	1.0942	0.0659	0.7135	0.0091	0.0750
50	**SA**	1.3332	2.6612	0.6983	0.0070	2.6683
	GA	1.0321	0.0205	0.7005	0.0035	0.0240
	PSO	1.0306	0.0191	0.7032	0.0034	0.0224
	DE	1.0305	0.0191	0.7032	0.0034	0.0225
	ABC	1.0305	0.0191	0.7032	0.0033	0.0224
100	**SA**	1.3736	3.0412	0.6922	0.0076	3.0488
	GA	1.0225	0.0112	0.6994	0.0020	0.0132
	PSO	1.0230	0.0102	0.7030	0.0018	0.0120
	DE	1.0229	0.0102	0.7030	0.0018	0.0120
	ABC	1.0228	0.0102	0.7030	0.0018	0.0120
500	**SA**	1.3132	2.6501	0.6939	0.0049	2.6550
	GA	1.0045	0.0028	0.6963	0.0006	0.0034
	PSO	1.0053	0.0017	0.7005	0.0004	0.0021
	DE	1.0056	0.0016	0.7007	0.0003	0.0020
	ABC	1.0056	0.0016	0.7008	0.0003	0.0020

Çizelge 5.34. Parametreleri $\theta = (\beta, q) = (1.5, 0.9)$ olan Kesikli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

n	**Yöntem**	$\hat{\beta}$		$\hat{q}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	1.7684	1.6505	0.8879	0.0090	1.6595
	GA	1.5731	0.1069	0.8929	0.0031	0.1100
	PSO	1.6161	0.1172	0.9024	0.0028	0.1200
	DE	1.6161	0.1173	0.9024	0.0028	0.1200
	ABC	1.6108	0.1138	0.9018	0.0028	0.1166
50	**SA**	1.7121	1.6173	0.8865	0.0079	1.6252
	GA	1.4979	0.0464	0.8883	0.0017	0.0481
	PSO	1.5325	0.0339	0.8993	0.0011	0.0349
	DE	1.5327	0.0339	0.8993	0.0011	0.0349
	ABC	1.5293	0.0336	0.8987	0.0011	0.0347
100	**SA**	1.7281	1.7033	0.8904	0.0063	1.7096
	GA	1.4961	0.0276	0.8913	0.0012	0.0287
	PSO	1.5265	0.0162	0.9017	0.0005	0.0167
	DE	1.5264	0.0162	0.9017	0.0005	0.0167
	ABC	1.5235	0.0162	0.9011	0.0005	0.0167
500	**SA**	1.8206	2.5759	0.8866	0.0074	2.5833
	GA	1.4747	0.0164	0.8894	0.0008	0.0172
	PSO	1.5031	0.0037	0.8998	0.0002	0.0039
	DE	1.5036	0.0034	0.9001	0.0001	0.0035
	ABC	1.5012	0.0038	0.8996	0.0001	0.0039

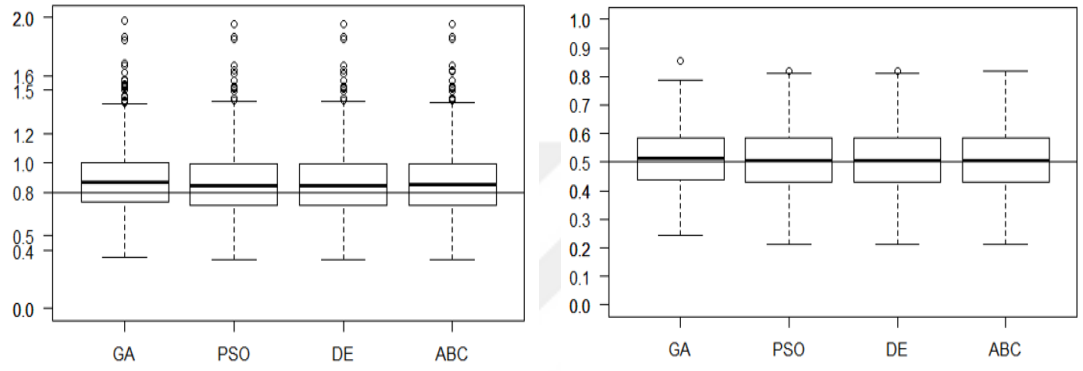
Çizelge 5.35. Parametreleri $\theta = (\beta, q) = (2, 0.95)$ olan Kesikli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

n	**Yöntem**	$\hat{\beta}$		$\hat{q}$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	2.1674	1.0953	0.9354	0.0059	1.1011
	GA	2.0890	0.1657	0.9436	0.0012	0.1669
	PSO	2.1876	0.2155	0.9524	0.0010	0.2165
	DE	2.1873	0.2154	0.9524	0.0010	0.2164
	ABC	2.1666	0.1940	0.9515	0.0010	0.1949
50	**SA**	2.1949	1.7609	0.9306	0.0087	1.7696
	GA	1.9952	0.0717	0.9422	0.0007	0.0724
	PSO	2.0586	0.0626	0.9503	0.0004	0.0630
	DE	2.0589	0.0627	0.9503	0.0004	0.0631
	ABC	2.0476	0.0600	0.9494	0.0004	0.0604
100	**SA**	2.1766	1.5954	0.9343	0.0067	1.6021
	GA	1.9632	0.0559	0.9401	0.0007	0.0565
	PSO	2.0232	0.0316	0.9493	0.0002	0.0319
	DE	2.0232	0.0316	0.9493	0.0002	0.0319
	ABC	2.0165	0.0321	0.9486	0.0002	0.0323
500	**SA**	2.2670	2.2121	0.9314	0.0103	2.2225
	GA	1.9463	0.0383	0.9402	0.0005	0.0388
	PSO	2.0007	0.0070	0.9493	0.0002	0.0071
	DE	2.0019	0.0053	0.9496	0.0000	0.0054
	ABC	1.9959	0.0075	0.9489	0.0001	0.0075

Çizelge 5.31-5.35'ten görüldüğü üzere, örneklem büyüklüğü arttıkça PSO, DE ve ABC algoritmalarında Def değerleri düşmüştür. Başka bir ifadeyle, bu algoritmalar ile örneklem büyüklüğü arttıkça MSE kriterine göre daha etkin parametre tahminleri elde edilmiştir. SA algoritması ele alınan diğer metasezgisel yöntemlerden farklı olarak tek çözüme dayalı bir algoritma olduğundan tutarlı sonuçlar vermemiştir.

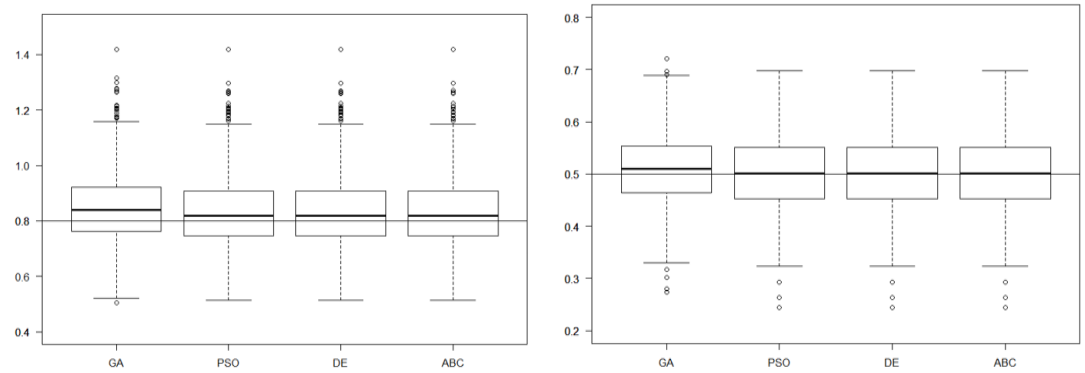
$n = 20$ için Çizelge 5.29'da verilen tüm parametre grupları için en küçük Def değerlerine sahip GA en iyi performansı sergilerken $n = 50$, 100 ve 500 için en küçük Def değerlerine sahip PSO, DE ve ABC algoritmalarını en iyi performansı sergilemiştir. PSO, DE ve ABC algoritmalarının genellikle birbirine çok yakın sonuçlar verdiği görülmüştür. SA algoritması ise, özellikle $\hat{\beta}$ parametresi için yüksek MSE değerine sahip olduğundan diğer metasezgisel algoritmalara göre daha yüksek Def değerine sahiptir. Sonuç olarak, kesikli Weibull dağılımının parametrelerinin tahmininde küçük örneklemler ($n = 20$) için GA'nın kullanılması büyük örneklemler ($n = 50, 100$ ve 500) için PSO, DE veya ABC algoritmalarının kullanılması önerilebilir.

Simülasyon çalışmalarını görselleştirebilmek amacıyla $(\beta, q) = (0.8, 0.5)$ parametresinin $n = 20, 50, 100$ ve 500 örneklem büyüklükleri için metasezgisel algoritmalar kullanılarak elde edilen parametre tahminlerine ilişkin örneklem dağılımlarının kutu grafikleri sırasıyla Şekil 5.11-5.14'te sunulmuştur. SA sonuçları çok fazla uç değerler verdiğinden ve grafiğin yapısını bozduğundan grafiklere dahil edilmemiştir. Kutu grafiklerden, özellikle PSO, DE ve ABC algoritmalarının birbirine çok yakın tahmin değerleri verdiği görülmüştür.



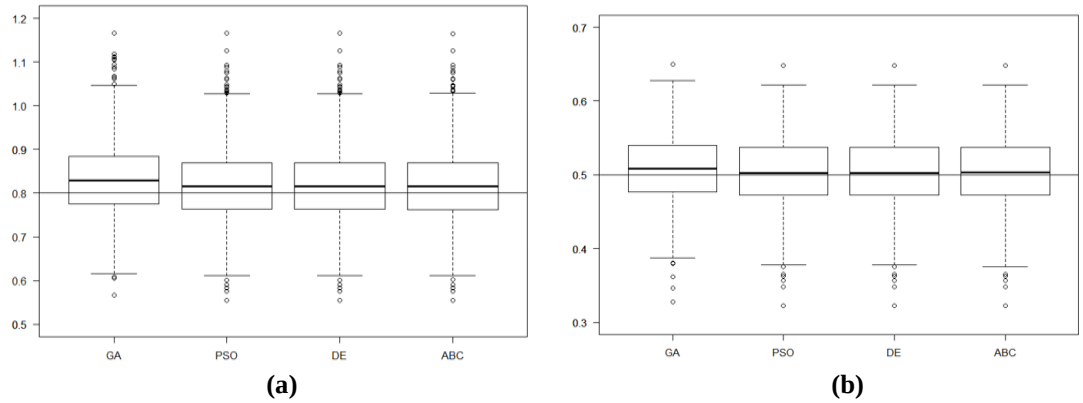
(a) **(b)**

Şekil 5.11. $(\beta, q) = (0.8, 0.5)$ ve $n = 20$ durumunda (a) $\hat{\beta}$ ve (b) $\hat{q}$ için kutu grafiği

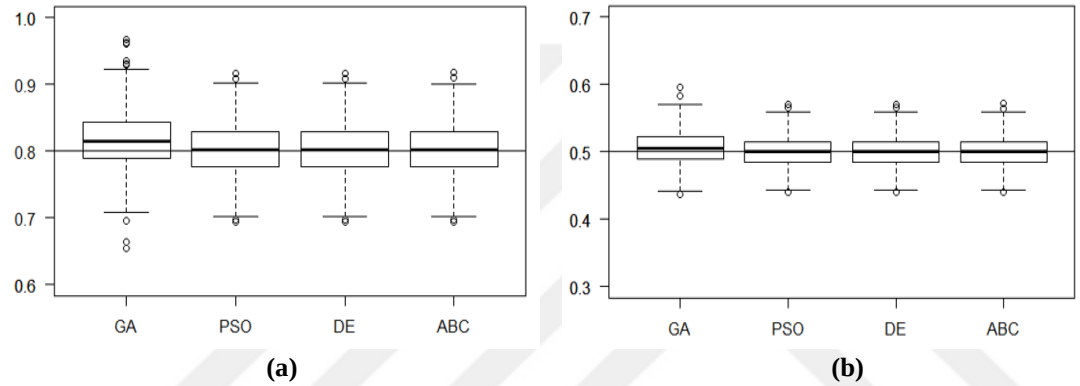


(a) **(b)**

Şekil 5.12. $(\beta, q) = (0.8, 0.5)$ ve $n = 50$ durumunda (a) $\hat{\beta}$ ve (b) $\hat{q}$ için kutu grafiği



Şekil 5.13. $(\beta, q) = (0.8, 0.5)$ ve $n = 100$ durumunda (a) $\hat{\beta}$ ve (b) $\hat{q}$ için kutu grafiği



Şekil 5.14. $(\beta, q) = (0.8, 0.5)$ ve $n = 500$ durumunda (a) $\hat{\beta}$ ve (b) $\hat{q}$ için kutu grafiği

5.4.3. Gerçek veri uygulaması

Bu alt kesimde, havadan iletişimdeki 46 başarısızlık için onarım sürelerini içeren bir veri kümesi ele alınmış ve Çizelge 5.36'da verilmiştir (Murthy ve ark., 2004; Lawless, 2011):

Çizelge 5.36. Havadan iletişimdeki başarısızlık veri kümesi

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2
2	2	3	3	3	3	3	4	4	4	4	5	5	5	6	6	7	8	9	9	11	22	25

Ele alınan veri kümesi için metasezgisel yöntemler kullanılarak elde edilen ML parametre tahmin değerleri Çizelge 5.37'de sunulmuştur. Yöntemlerin karşılaştırılması için $\ln L$, *AIC* ve *BIC* değerleri kullanılmış ve elde edilen parametre tahminlerine ilişkin

bu değerler Çizelge 5.37'de verilmiştir. Ayrıca, Çizelge 5.37'nin son satırında Murthy ve ark. (2004) tarafından elde edilen ML parametre tahmin değerleri verilmiştir.

Çizelge 5.37. Havadan iletişimdeki başarısızlık veri kümesi için parametre tahmin değerleri ve $\ln L$, *AIC* ve *BIC* değerleri

Yöntem	$\hat{\beta}$	$\hat{q}$	$\ln L$	***AIC***	***BIC***
SA	1.1054	0.6171	-113.1912	230.3824	241.697
GA	0.7344	0.6209	-103.7074	211.4148	222.7294
PSO	0.7308	0.6216	-103.6420	211.284	222.5986
DE	0.7288	0.6212	**-103.6298**	**211.2596**	**222.5742**
ABC	0.7289	0.6212	-103.6310	211.2620	222.5766
Murthy ve ark. (2004)	0.7289	0.6213	**-103.6281**	**211.2562**	**222.5708**

Çizelge 5.37'den görüldüğü üzere, SA algoritması haricindeki metasezgisel yöntemler birbirine yakın AIC ve BIC değerlerine sahiptir. Ele alınan metasezgisel yöntemler arasında en büyük $\ln L$ ve en küçük AIC ve BIC değerlerine sahip algoritmanın DE algoritması olduğu görülmüştür. Böylece, DE algoritması bu veri kümesi için ML parametre tahmininde diğer yöntemlere göre daha iyi performans sergilemiştir. Fakat, SA haricindeki yöntemler birbirine çok yakın sonuçlar verdiğinden bu veri kümesi için tüm metasezgisel yöntemler uygulanabilir.

6. ÇOK DEĞİŞKENLİ DAĞILIMLAR İÇİN METASEZGİSEL YÖNTEMLER İLE PARAMETRE TAHMİNİ

Bu bölüm, sürekli dağılımlardan iki değişkenli Gamma dağılımı ve kesikli dağılımlardan iki değişkenli kesikli Weibull dağılımının parametrelerinin metasezgisel yöntemler ile tahminlerini içermektedir.

6.1. Çok Değişkenli Gamma Dağılımının Parametrelerinin Parçacık Sürüsü Optimizasyonu ile Tahmini

Bu kesimde, Mathai ve Moschopoulos (1992) tarafından önerilen çok değişkenli Gamma dağılımının parametrelerinin ML yöntemi ile tahminine ele alınmıştır. Çok değişkenli Gamma dağılımının ML parametre tahmini için Parçacık Sürüsü Optimizasyonu (PSO) algoritması kullanılmıştır. PSO algoritmasını uygulamadan önce algoritmanın eylemsizlik katsayısı, hızlandırma katsayıları, popülasyon sayısı ve iterasyon sayısından oluşan parametrelerinin ayarlanması gerekmektedir. Literatürde, parametrelerin ayarlanması için özel bir yöntem mevcut değildir. Genel olarak, önceki çalışmalarda PSO algoritması uygulanırken literatürde geniş çapta kabul edilen parametre değerleri ya da deneme-yanılma temeline dayalı parametreler kullanılmıştır (Schwaab ve ark., 2008; Özsoy ve ark., 2017; Acıtaş ve ark., 2019). Örkcü ve ark. (2015b)'nın çalışmasında, farklı parametre tasarımları için simülasyon çalışmaları yaparak PSO algoritmasının performansı araştırılmış ve yüksek oranda uygun algoritma parametrelerinin seçimine bağlı olduğu sonucuna ulaştırılmıştır. Bu tez çalışmasında, çok değişkenli Gamma dağılımının parametrelerinin ML tahmininde kullanılmak üzere PSO algoritmasının parametrelerini belirlemek için bir deney tasarımı yöntemi olan Taguchi yönteminin kullanılması önerilmiştir.

PSO algoritmasının parametreleri Taguchi yöntemi ile tasarlandıktan sonra, Monte Carlo simülasyon çalışması ile yöntemin çok değişkenli Gamma dağılımının parametrelerinin ML tahminindeki etkinliği test edilmiştir.

6.1.1. Çok değişkenli Gamma dağılımının parametrelerinin tahmini için ML yöntemi

$\boldsymbol{Z}_1, \boldsymbol{Z}_2, ... \boldsymbol{Z}_k$ olasılık yoğunluk fonksiyonu Eşitlik (3.7)'de verilen k değişkenli Gamma dağılımından bir örneklem olmak üzere dağılıma ilişkin olabilirlik (L) ve log-olabilirlik ($\ln L$) fonksiyonu sırasıyla,

$$L = \prod_{i=1}^{n} \frac{\left(z_{1i} - \mu_1\right)^{\sigma_1 - 1} \left(z_{2i} - z_{1i} - \mu_2\right)^{\sigma_2 - 1} ... \left(z_{ki} - z_{(k-1)i} - \mu_k\right)^{\sigma_k - 1} \exp\left(- \frac{z_{ki} - \sum_{i=1}^{k} \mu_i}{\beta} \right)}{\beta^{\sum_{i=1}^{k} \sigma_i} \prod_{i=1}^{k} \Gamma(\sigma_i)} \tag{6.1}$$

ve

$$\ln L = (\sigma_1 - 1) \sum_{i=1}^{k} \log(z_{1i} - \mu_1) + ... + (\sigma_k - 1) \sum_{i=1}^{k} \ln(z_{ki} - z_{(k-1)i} - \mu_k) - \sum_{i=1}^{n} \frac{z_{ki}}{\beta} + \frac{n \sum_{i=1}^{k} \mu_i}{\beta} - n \ln(\beta) \sum_{i=1}^{k} \sigma_i - n \sum_{i=1}^{k} \ln(\Gamma(\sigma_i) \tag{6.2}$$

olarak elde edilir. Eşitlik (6.2)'den olabilirlik denklemleri,

$$\frac{\partial \ln L}{\partial \sigma_m} = (\sigma_m - 1) \sum_{i=1}^{k} \ln(z_{mi} - z_{(m-1)i} - \mu_m) - n\psi\left(\sigma_1\right) - n \ln \beta = 0 \tag{6.3}$$

$$\frac{\partial \ln L}{\partial \beta} = \frac{\sum_{i=1}^{n} z_{mi} - n \sum_{j=1}^{k} \mu_j}{\beta^2} - n \frac{\sum_{j=1}^{k} \sigma_j}{\beta} = 0 \tag{6.4}$$

$$\frac{\partial \ln L}{\partial \mu_m} = -(\sigma - 1) \sum_{i=1}^{n} (z_{mi} - z_{(m-1)i} - \mu_m)^{-1} + \frac{n}{\beta} = 0 \tag{6.5}$$

olarak elde edilir. Eşitlik (6.3)-(6.5)'te, $z_{0i} = 0$, $m = 1, 2, ..., k$ ve $\psi\left(\sigma_1\right) = \frac{\partial \ln(\Gamma(\sigma_l))}{\partial \sigma_l}$ 'dir.

Olabilirlik denklemleri lineer olmayan fonksiyonlar içerdiğinden çözümü analitik olarak elde edilemez. Bundan dolayı, bu çalışmada metasezgisel yöntemlerden PSO algoritmasının kullanılması önerilmiştir. PSO algoritmasının parametrelerinin belirlenmesinde Taguchi deney tasarımı yöntemi kullanılmıştır.

6.1.2. PSO algoritması için Taguchi deney tasarımı

Bu alt kesimde, çok değişkenli Gamma dağılımının ML parametrelerinin tahmininde kullanılmak üzere PSO algoritmasının tasarlanması ele alınmıştır ve iki amaç göz önünde bulundurulmuştur. Bu amaçlar, algoritmanın yüksek çözüm kalitesine ve düşük hesaplama zamanına sahip olmasıdır. Algoritmanın çözüm kalitesinin değerlendirilmesinde Def kriteri kullanılmış olup, Def değeri ne kadar düşükse çözüm kalitesi o kadar yüksektir. Böylece, ele alınan problem için minimum Def değerine ve minimum hesaplama zamanına sahip bir algoritma tasarımının gerçekleştirilmesi hedeflenmiştir. Bundan dolayı "en küçük, en iyidir" yaklaşımı kullanılmış ve Eşitlik (3.69)'da verilen S/N oranı hesaplanmıştır. Burada, y_i algoritmanın i. çalıştırılmasında yanıt değişkeninin değerini göstermektedir. Bu çalışmada, Def değerleri ve hesaplama zamanları yanıt değişkeni olarak alınmıştır.

PSO algoritmasının performansını etkileyen dört faktör vardır. Bu faktörler; eylemsizlik katsayısı, hızlandırma katsayıları, popülasyon sayısı ve iterasyon sayısıdır. Bu çalışmada, Çizelge 6.1'de verilen her bir faktöre ilişkin üç seviye göz önüne alınarak Taguchi deney tasarımı ile PSO algoritmasının parametreleri için optimal seviyeler belirlenmiştir.

Çizelge 6.1. PSO algoritması için faktörler ve seviyeleri

Seviyeler / Faktörler	Seviye 1	Seviye 2	Seviye 3
Eylemsizlik Katsayısı (A)	S1	S2	S3
Hızlandırma Katsayıları (B)	1	1.45	2
Popülasyon Sayısı (C)	10	25	50
İterasyon Sayısı (D)	100	500	1000

Çizelge 6.1'de, S1 stratejisi sabit eylemsizlik katsayısını ($w = 0.7$), S2 stratejisi rastgele eylemsizlik katsayısını ($w = 0.5 + rand/2$) ve S3 stratejisi doğrusal azalan

eylemsizlik katsayısını ($w = 0.9 - (0.5/Maxit) \times it$) göstermektedir. Burada, $Maxit$ toplam iterasyon sayısını ve it mevcut iterasyon sayısını göstermektedir.

Deney tasarımında, rasgele seçilen $(\alpha_1, \alpha_2, \beta, \mu_1, \mu_2) = (2.8, 4, 0.5, 5, 10)$ değerleri kullanılmıştır. Tam faktör tasarımının tek bir tekrarı $3^4 = 81$ deneye ihtiyaç duyarken $L_9(4^3)$ ortogonal diziye sahip bir Taguchi deneyi 9 deneye ihtiyaç duymaktadır. $L_9(4^3)$ deney düzeneği Çizelge 6.2'de verildiği gibi oluşturulmaktadır.

Çizelge 6.2. Modifiye edilmiş ortogonal dizi

Deney no.	**Faktörler ve Seviyeleri**							
	A	**B**	**C**	**D**	**A**	**B**	**C**	**D**
1	1	1	1	1	S1	1.00	10	100
2	1	2	2	2	S1	1.45	25	500
3	1	3	3	3	S1	2.00	50	1000
4	2	1	2	3	S2	1.00	25	1000
5	2	2	3	1	S2	1.45	50	100
6	2	3	1	2	S2	2.00	10	500
7	3	1	3	2	S3	1.00	50	500
8	3	2	1	3	S3	1.45	10	1000
9	3	3	2	1	S3	2.00	25	100

Her bir deney için algoritmanın 3 kez çalıştırılması sonucunda hesaplanan çözüm kalitesi için Def değerleri ve hesaplama zamanı değerleri sırasıyla Çizelge 6.3 ve Çizelge 6.4'te verilmiştir. Ayrıca, bu değerlere ilişkin ortalama değerler ve S/N oranları da Çizelge 6.3 ve Çizelge 6.4'te gösterilmiştir.

Çizelge 6.3. PSO algoritmasına ilişkin Def değerleri, ortalama değerler ve S/N oranları

Deney no.	**Sonuç 1**	**Sonuç 2**	**Sonuç 3**	$\bar{Y}$	S/N
1	0.6298	0.6319	0.6280	0.6299	4.0141
2	0.5905	0.5789	0.5923	0.5872	4.6225
3	0.5653	0.5662	0.5507	0.5607	5.0239
4	0.5821	0.5675	0.5858	0.5785	4.7528
5	0.5889	0.5692	0.5805	0.5795	4.7366
6	0.6699	0.6491	0.6010	0.6400	3.8667
7	0.5968	0.5487	0.6096	0.5850	4.6469
8	0.7106	0.6791	0.6658	0.6852	3.2802
9	0.6112	0.5795	0.5788	0.5898	4.5821

Çizelge 6.4. PSO algoritmasına ilişkin hesaplama zamanı değerleri, ortalama değerler ve S/N oranları

Deney no.	Sonuç 1	Sonuç 2	Sonuç 3	$\bar{Y}$	S/N
1	0.0529	0.0577	0.0587	0.0564	24.9544
2	0.6827	0.6331	0.6539	0.6566	3.64981
3	2.4488	2.4584	2.5216	2.4763	-7.8768
4	1.2271	1.2002	1.2288	1.2187	-1.7187
5	0.2498	0.2573	0.2558	0.2543	11.8905
6	0.2525	0.2381	0.2502	0.2469	12.1436
7	1.2042	1.2266	1.2743	1.2350	-1.8364
8	0.5154	0.4938	0.4921	0.5005	6.00973
9	0.1281	0.1233	0.1233	0.1249	18.0624

Her bir algoritma parametre seviyesi için faktörlerin Def değerlerinin ve hesaplama zamanlarına ilişkin S/N oranları sırasıyla Çizelge 6.5 ve Çizelge 6.6'da verilmiştir. En büyük S/N değerine sahip olan faktör en etkili faktör olarak belirlenmiştir. Def değerine etkisine göre faktörler; popülasyon sayısı, eylemsizlik katsayısı, hızlandırma katsayıları ve iterasyon sayısı olarak sıralanmıştır. Hesaplama zamanı değerine etkisine göre faktörler; iterasyon sayısı, popülasyon sayısı, eylemsizlik katsayısı ve hızlandırma katsayıları olarak sıralanmıştır.

Çizelge 6.5. Parametre seviyelerinin Def değerlerine ilişkin S/N oranları

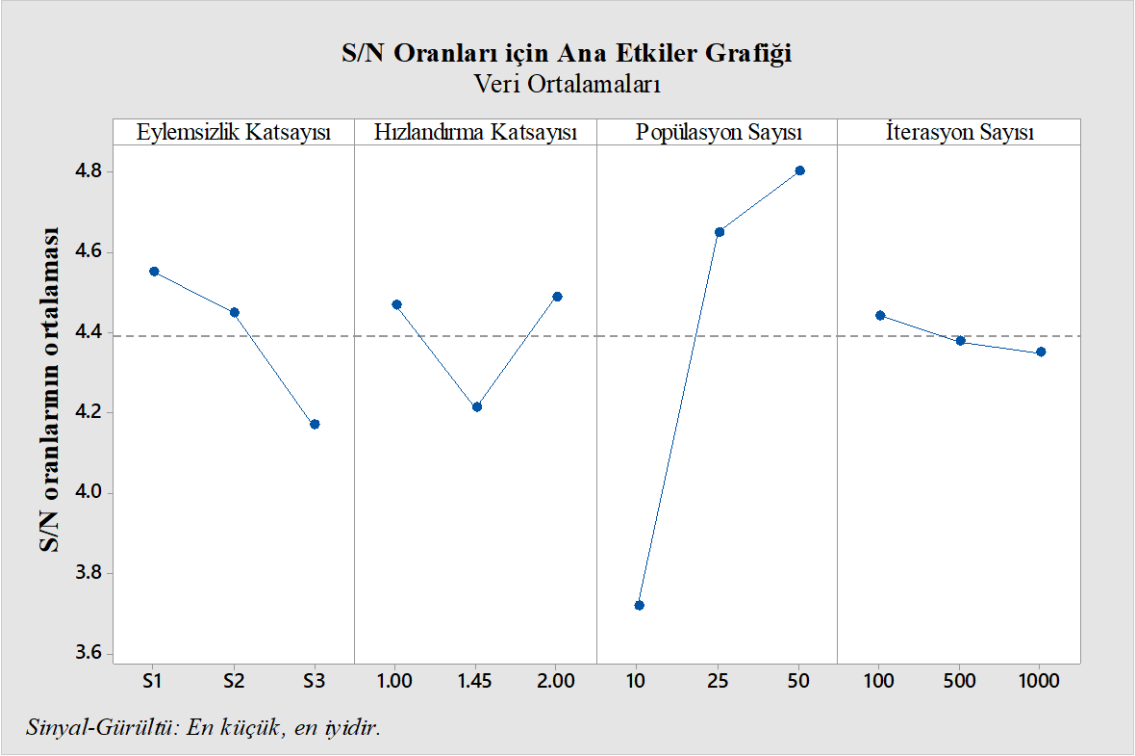
Seviyeler	Eylemsizlik Katsayısı (A)	Hızlandırma Katsayıları (B)	Popülasyon Sayısı (C)	İterasyon Sayısı (D)
1	**4.554**	4.471	3.720	**4.444**
2	4.452	4.213	4.652	4.379
3	4.170	**4.491**	**4.803**	4.352
Max-Min	0.384	0.278	1.082	0.092
Sıralama	2	3	1	4

Çizelge 6.6. Parametre seviyelerinin hesaplama zamanı değerlerine ilişkin S/N oranları

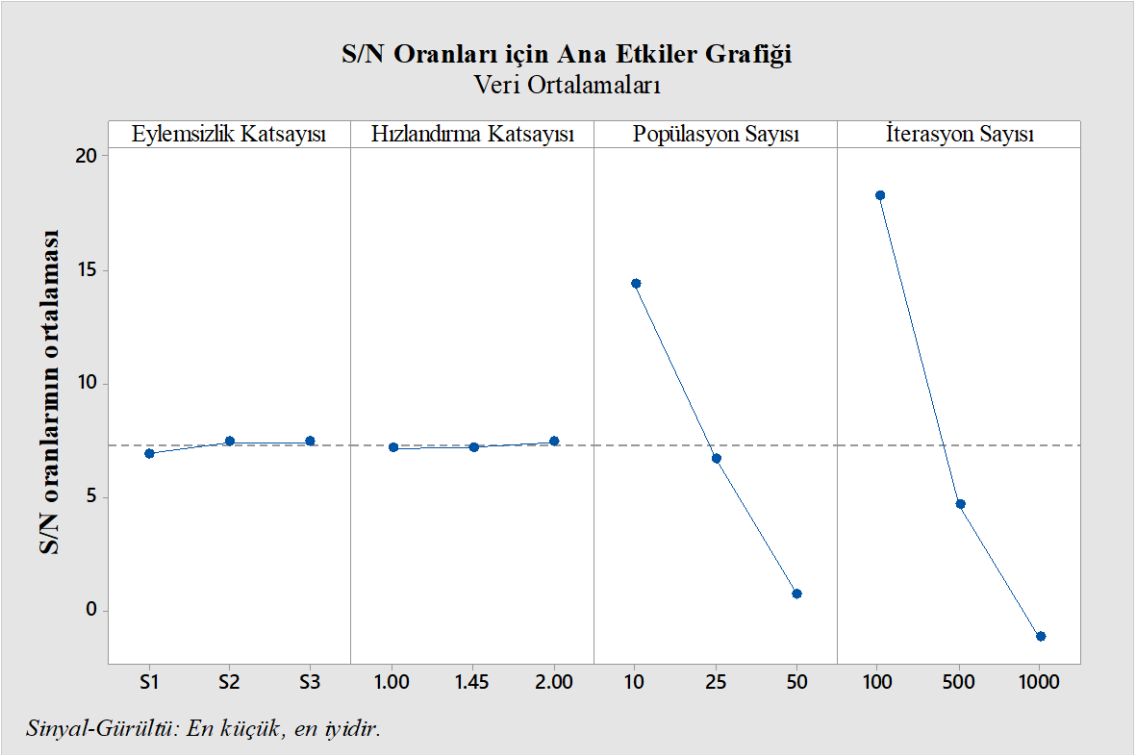
Seviyeler	Eylemsizlik Katsayısı (A)	Hızlandırma Katsayıları (B)	Popülasyon Sayısı (C)	İterasyon Sayısı (D)
1	6.9091	7.1331	**14.3693**	**18.3025**
2	**7.4385**	7.1834	6.6645	4.6523
3	7.4119	**7.4431**	0.7258	-1.1953
Max-Min	0.5293	0.3100	13.6435	19.4978
Sıralama	3	4	2	1

Her bir faktörün en büyük S/N değerine sahip olduğu seviye o faktörün en iyi seviyesini göstermektedir. Def kriteri açısından, eylemsizlik katsayısı için en iyi seviye

1, hızlandırma katsayıları için en iyi seviye 3, popülasyon sayısı için en iyi seviye 3 ve iterasyon sayısı için en iyi seviye 1 olarak tespit edilmiştir. Hesaplama zamanı açısından ise, eylemsizlik katsayısı için en iyi seviye 2, hızlandırma katsayıları için en iyi seviye 3, popülasyon sayısı için en iyi seviye 1 ve iterasyon sayısı için en iyi seviye 1 olarak tespit edilmiştir Ayrıca, Def ve hesaplama zamanı değerleri için S/N oranlarına ilişkin grafikler sırasıyla Şekil 6.1 ve Şekil 6.2'de gösterilmiştir.



Şekil 6.1. Def değerleri için (S/N) oranları grafiği



Şekil 6.2. Hesaplama zamanı değerleri için (S/N) oranları grafiği

Faktörlerin Def değerleri ve hesaplama zamanı üzerindeki etkisinin istatistiksel olarak anlamlı olup olmadığını test etmek için tek yönlü Varyans Analizi (ANOVA) yapılmıştır ve Def değerleri ve hesaplama zamanları için sonuçlar sırasıyla Çizelge 6.7 ve Çizelge 6.8'de verilmiştir.

Çizelge 6.7. Def değerleri için ANOVA tablosu

Faktör	**Serbestlik Derecesi (SD)**	**Kareler Toplamı (KT)**	**Kareler Ortalaması (KO)**	**F-değeri**	**p-değeri**
Eylemsizlik Katsayısı (A)	2	0.0036	0.0018	4.74	0.022*
Hızlandırma Katsayıları (B)	2	0.0024	0.0012	3.11	0.069
Popülasyon Sayısı (C)	2	0.0311	0.0155	40.32	0.000*
İterasyon Sayısı (D)	2	0.0003	0.0001	0.41	0.672
Hata	18	0.0069	0.0003		
Toplam	26	0.0445			

Çizelge 6.8. Hesaplama zamanı değerleri için ANOVA tablosu

Faktör	Serbestlik Derecesi (SD)	Kareler Toplamı (KT)	Kareler Ortalaması (KO)	F-değeri	p-değeri
Eylemsizlik Katsayısı (A)	2	1.3148	0.6573	1484.25	0.000*
Hızlandırma Katsayıları (B)	2	1.1287	0.5643	1274.19	0.000*
Popülasyon Sayısı (C)	2	5.0970	2.5485	5754.10	0.000*
İterasyon Sayısı (D)	2	7.0888	3.5444	8002.66	0.000*
Hata	18	0.0080	0.0004		
Toplam	26	14.6373			

Çizelge 6.7'den görüldüğü üzere, eylemsizlik katsayısı ve popülasyon sayısı için p-değerleri $\alpha = 0.05$'ten küçük olduğundan dolayı bu iki faktörün Def değeri yani çözüm kalitesi üzerindeki etkisi %95 güven seviyesinde istatistiksel olarak anlamlı olduğu görülmektedir. Şekil 6.1'den, Def değerlerine göre algoritmanın eylemsizlik katsayısı için en iyi seviye olan seviye 1 yani S1 stratejisi ve popülasyon sayısı için en iyi seviye olan seviye 3 yani popülasyon sayısı=50 seçildiği görülmektedir.

Çizelge 6.8'den, tüm faktörler için p-değerleri $\alpha = 0.05$'ten küçük olduğundan dolayı tüm faktörlerin hesaplama zamanı üzerindeki etkisi $\%95$ güven seviyesinde istatistiksel olarak anlamlı olduğu görülmektedir. Eylemsizlik katsayısı ve popülasyon sayısı için en iyi seviyeler Def değerlerine bağlı olan çözüm kalitesine göre seçildiğinden hızlandırma katsayıları ve iterasyon sayısı faktörlerinin en iyi seviyeleri hesaplama zamanına göre belirlenebilir. Şekil 6.2'den de görüldüğü üzere, algoritmanın hızlandırma katsayıları için en iyi seviye olan seviye 3 yani hızlandırma katsayıları=2 ve iterasyon sayısı için en iyi seviye olan seviye 1 yani iterasyon sayısı=100 seçilir. Tüm faktörler için seçilen en iyi seviyeler ve seçilme kriterleri Çizelge 6.9'da verilmiştir.

Çizelge 6.9. Faktörlerin seçilen optimal seviyeleri

Kriterler	Faktörler			
	Eylemsizlik Katsayısı (A)	**Hızlandırma Katsayıları (B)**	**Popülasyon Sayısı (C)**	**İterasyon Sayısı (D)**
Çözüm kalitesi (Def değerleri)	Seviye 1	-	Seviye 3	-
Hesaplama zamanı	-	Seviye 3	-	Seviye 1

6.1.3. Simülasyon çalışması

Bu alt kesimde, çok değişkenli Gamma dağılımının parametrelerinin ML yöntemi ile tahmininde kullanılmak üzere tasarlanan PSO algoritmasının performansını araştırmak için Monte-Carlo simülasyon çalışması yapılmıştır.

Çok değişkenli Gamma dağılımı için seçilen gerçek parametre değerleri Çizelge 6.10'da verilmiştir. Ayrıca, parametre tahmin sürecinde örneklem büyüklüğünün etkisini araştırmak için her bir parametre değeri için örneklem büyüklükleri 10, 30, 50, 100, 250, 500 ve 1000 olarak göz önüne alınmıştır. Arama uzayı olarak, parametrelerin Uyarlanmış En Çok Olabilirlik (MML) tahminlerine dayalı güven aralıkları kullanılmıştır.

Çizelge 6.10. Simülasyon çalışması için seçilen parametre değerleri

σ_1	σ_2	β	μ_1	μ_2
2.8	4	0.5	5	10
3	4	1	-2	10
5	2.6	0.75	1	8

PSO algoritması, her bir parametre değeri ve örneklem büyüklüğü için 5000 kez tekrarlanmıştır. Parametreleri Taguchi yöntemi ile tasarlanan PSO algoritması kullanılarak elde edilen iki değişkenli Gamma dağılımına ilişkin parametrelerin ML tahminlerinin ortalaması, MSE ve Def değerleri Çizelge 6.11-6.13'te verilmiştir.

Çizelge 6.11. Parametreleri $(\sigma_1, \sigma_2, \beta, \mu_1, \mu_2) = (2.8,\ 4,\ 0.5,\ 5,\ 10)$ olan iki değişkenli Gamma dağılımı için simülasyon çalışması sonucunda elde edilen ortalama, MSE ve Def değerleri

	$\hat{\sigma}_1$		$\hat{\sigma}_2$		$\hat{\beta}$		$\hat{\mu}_1$		$\hat{\mu}_2$		
n	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	3.3809	2.5371	4.9881	7.6124	0.4252	0.0211	5.0580	0.1367	10.0677	0.4770	10.7842
30	2.8322	0.6879	4.1002	1.4614	0.4891	0.0082	5.0508	0.0443	10.0648	0.1204	2.3222
50	2.6254	0.3040	3.7678	0.7703	0.5127	0.0041	5.0661	0.0239	10.096	0.0763	1.1788
100	2.6262	0.1504	3.7329	0.3468	0.5178	0.0018	5.0454	0.0110	10.0718	0.0322	0.5421
250	2.6850	0.0667	3.8355	0.1651	0.5128	0.0010	5.0253	0.0036	10.0390	0.0130	0.2493
500	2.7037	0.0383	3.8508	0.1058	0.5123	0.0007	5.0176	0.0017	10.0315	0.0073	0.1537
1000	2.7686	0.0165	3.9447	0.0473	0.5041	0.0002	5.0068	0.0007	10.012	0.0031	0.0681

Çizelge 6.12. Parametreleri $(\sigma_1, \sigma_2, \beta, \mu_1, \mu_2) = (3,\ 4,\ 1,\ -2,\ 10)$ olan iki değişkenli Gamma dağılımı için simülasyon çalışması sonucunda elde edilen ortalama, MSE ve Def değerleri

	$\hat{\sigma}_1$		$\hat{\sigma}_2$		$\hat{\beta}$		$\hat{\mu}_1$		$\hat{\mu}_2$		
n	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	3.6854	4.7680	4.7660	4.6047	0.8679	0.0783	-1.8844	0.9185	10.1414	1.3427	11.7122
30	3.2008	0.6729	4.4666	0.6751	0.9283	0.0182	-1.9401	0.2131	9.9102	0.2572	1.8365
50	2.8317	0.3665	3.8235	0.6163	1.0200	0.0186	-1.8593	0.1180	10.1713	0.2489	1.3683
100	2.8087	0.1797	3.7624	0.2344	1.0380	0.0086	-1.8988	0.0530	10.1213	0.0945	0.5702
250	2.8493	0.0870	3.8116	0.1178	1.0340	0.0048	-1.9387	0.0198	10.0761	0.0363	0.2657
500	2.8980	0.0428	3.8546	0.0908	1.0250	0.0025	-1.9639	0.0091	10.0605	0.0247	0.1699
1000	2.9529	0.0212	3.9331	0.0496	1.0105	0.0012	-1.9806	0.0044	10.0284	0.0127	0.0891

Çizelge 6.13. Parametreleri $(\sigma_1, \sigma_2, \beta, \mu_1, \mu_2) = (5,\ 2.6,\ 0.75,\ 1,\ 8)$ olan iki değişkenli Gamma dağılımı için simülasyon çalışması sonucunda elde edilen ortalama, MSE ve Def değerleri

	$\hat{\sigma}_1$		$\hat{\sigma}_2$		$\hat{\beta}$		$\hat{\mu}_1$		$\hat{\mu}_2$		
n	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
10	5.2911	2.3772	2.7719	3.8366	0.7080	0.0381	1.2052	0.8600	8.1357	0.4733	7.5853
30	4.9408	1.7870	2.5738	0.2236	0.7387	0.0069	1.1362	0.5318	8.0728	0.0529	2.6023
50	4.6468	0.9956	2.4430	0.1993	0.7682	0.0050	1.2033	0.3114	8.0814	0.0364	1.5477
100	4.8436	0.7727	2.5195	0.0927	0.7628	0.0058	1.1011	0.1551	8.0480	0.0139	1.0403
250	4.6483	0.3719	2.4434	0.0704	0.7875	0.0039	1.1139	0.0607	8.0400	0.0063	0.5133
500	4.8702	0.2057	2.5351	0.0325	0.7635	0.0016	1.0444	0.0354	8.0209	0.0030	0.2782
1000	4.9156	0.0871	2.5583	0.0154	0.7588	0.0007	1.0265	0.0147	8.0121	0.0012	0.1191

Çizelge 6.11-6.13'te verilen sonuçlar, parametreli Taguchi yöntemi ile ayarlanmış PSO algoritmasının çok değişkenli Gama dağılımın parametrelerini tahmin edebildiğini göstermektedir. Ayrıca, örneklem büyüklüğü arttıkça, gerçek parametre değerlerine daha yakın değerler elde edilmektedir ve MSE değerleri dolayısıyla Def değerleri de her durumda azalmaktadır. Bu sonuç, parametreleri Taguchi yöntemi ile tasarlanan PSO algoritmasının çok değişkenli Gamma dağılımının parametrelerini tahmin etmek için uygun olduğunu göstermektedir.

6.2. Çok Değişkenli Kesikli Weibull Dağılımının Parametrelerinin Metasezgisel Yöntemler ile Tahmini

Bu kesimde, iki değişkenli kesikli Weibull dağılımının parametrelerinin ML yöntemi ile tahminine yer verilmiştir. Bu dağılımın ML parametre tahmini için metasezgisel yöntemlerden SA, DE, PSO, ABC ve GA kullanılmıştır. Algoritmalar,

Monte-Carlo simülasyon çalışmaları ile çözüm kalitesi açısından karşılaştırılmış ve gerçek bir veri kümesi uygulaması sunulmuştur.

6.2.1. Çok değişkenli kesikli Weibull dağılımlarının parametrelerinin tahmini için ML yöntemi

$(x_{11}, x_{21}), ..., (x_{1n}, x_{2n})$'ler olasılık fonksiyonu Eşitlik (3.10)'da verilen $BDW(\beta, q_0, q_1, q_2)$ dağılımından alınan rasgele bir örneklem olmak üzere, olabilirlik (L) ve log-olabilirlik ($\ln L$) fonksiyonları sırasıyla

$$\begin{aligned} L = & \prod_{i \in I_1} \left((q_1)^{x_{1i}^{\beta}} - (q_1)^{(x_{1i}+1)^{\beta}} \right) \left((q_0 q_2)^{x_{2i}^{\beta}} - (q_0 q_2)^{(x_{2i}+1)^{\beta}} \right) \times \\ & \prod_{i \in I_2} \left((q_0 q_1)^{x_{1i}^{\beta}} - (q_0 q_1)^{(x_{1i}+1)^{\beta}} \right) \left((q_2)^{x_{2i}^{\beta}} - (q_2)^{(x_{2i}+1)^{\beta}} \right) \times \\ & \prod_{i \in I_3} (q_1)^{x_i^{\beta}} \times \left((q_0 q_2)^{x_i^{\beta}} - (q_0 q_2)^{(x_i+1)^{\beta}} \right) - (q_0 q_1)^{(x_i+1)^{\beta}} \times \left(q_2^{x_i^{\beta}} - q_2^{(x_i+1)^{\beta}} \right) \end{aligned} \tag{6.6}$$

ve

$$\begin{aligned} \ln L = & \sum_{i \in I_1} \ln \left[\left((q_1)^{x_{1i}^{\beta}} - (q_1)^{(x_{1i}+1)^{\beta}} \right) \left((q_0 q_2)^{x_{2i}^{\beta}} - (q_0 q_2)^{(x_{2i}+1)^{\beta}} \right) \right] + \\ & \sum_{i \in I_2} \ln \left[\left((q_0 q_1)^{x_{1i}^{\beta}} - (q_0 q_1)^{(x_{1i}+1)^{\beta}} \right) \left((q_2)^{x_{2i}^{\beta}} - (q_2)^{(x_{2i}+1)^{\beta}} \right) \right] + \\ & \sum_{i \in I_3} \ln \left[(q_1)^{x_i^{\beta}} \times \left((q_0 q_2)^{x_i^{\beta}} - (q_0 q_2)^{(x_i+1)^{\beta}} \right) - (q_0 q_1)^{(x_i+1)^{\beta}} \times \left(q_2^{x_i^{\beta}} - q_2^{(x_i+1)^{\beta}} \right) \right] \end{aligned} \tag{6.7}$$

biçimindedir. Eşitlik (6.5) ve (6.6)'da, $I_1 = \{i : x_{1i} < x_{2i}\}$, $I_2 = \{i : x_{1i} > x_{2i}\}$ ve $I_0 = \{i : x_{1i} = x_{2i}\}$ indikatör fonksiyonlarıdır. L veya $\ln L$ fonksiyonlarının bilinmeyen parametrelere göre birinci kısmi türevleri alınarak olabilirlik denklemleri elde edilir. Olabilirlik denklemleri lineer olmayan fonksiyonlar içerdiğinden bu denklemlerin çözümü analitik olarak elde edilemez. Bundan dolayı, bu çalışmada metasezgisel yöntemlerin kullanılması önerilmiştir.

6.2.2. Simülasyon çalışması

Bu alt kesimde, iki değişkenli kesikli Weibull dağılımının parametrelerinin ML tahmini için Monte-Carlo simülasyon sonuçları sunulmuştur. Önerilen metasezgisel yöntemlerin (SA, GA, PSO, DE ve ABC) performansları, çözüm kalitesi açısından karşılaştırılmış ve bu amaçla için Def kriteri kullanılmıştır.

Ele alınan metasezgisel yöntemlerin performanslarının değerlendirilmesi için, simülasyon çalışmasında kullanılan kesikli iki değişkenli Weibull dağılımı için gerçek parametre değerleri Çizelge 6.14'te verilmiştir. Ayrıca, parametre tahmin sürecinde örneklem büyüklüğünün etkisini araştırmak amacıyla her bir parametre değeri için 20, 50, 100 ve 500 büyüklüğünde örneklemler göz önüne alınmıştır.

Çizelge 6.14. Simülasyon çalışmasında ele alınan kesikli iki değişkenli Weibull dağılımlarına ilişkin parametreler

β	q_0	q_1	q_2
0.5	0.7	0.3	0.5
0.7	0.95	0.8	0.6
0.9	0.9	0.9	0.9

Metasezgisel yöntemlere ilişkin parametre değerleri çeşitli denemeler ve literatür araştırmaları sonucu belirlenmiş olup Çizelge 5.30'da verilen değerler kullanılmıştır. Ele alınan tüm yöntemlerde, arama uzayı Weibull dağılımının β parametresi için $[0,10]$, q_0, q_1 ve q_2 parametreleri için $[0,1]$ olarak alınmıştır. Metasezgisel algoritmalar hızlı bir yakınsama gösterdiğinden maksimum iterasyon sayısı 100 olarak alınmıştır.

Algoritma, her bir parametre değeri ve örneklem büyüklüğü için 5000 kez tekrarlanmış ve simülasyon sonuçlarına ilişkin ortalama, MSE ve Def değerleri Çizelge 6.15-6.17'de verilmiştir.

Çizelge 6.15. Parametreleri $\theta = (\beta, q_0, q_1, q_2) = (0.5, 0.7, 0.3, 0.5)$ olan Kesikli iki değişkenli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

n	**Yöntem**	$\hat{\beta}$		$\hat{q}_0$		$\hat{q}_1$		$\hat{q}_2$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	0.8189	2.2915	0.6823	0.0368	0.3249	0.0231	0.5409	0.0360	2.3873
	GA	0.6311	0.0798	0.6234	0.0208	0.3883	0.0223	0.5949	0.0266	**0.1496**
	PSO	0.5838	0.1519	0.7363	0.0493	0.3044	0.0203	0.5281	0.0458	0.2672
	DE	0.5873	0.2006	0.7355	0.0487	0.3046	0.0203	0.5279	0.0450	0.3146
	ABC	0.5813	0.0964	0.7304	0.0465	0.3066	0.0201	0.5301	0.0444	0.2074
50	**SA**	0.8850	3.3278	0.7026	0.0255	0.3154	0.0119	0.5112	0.0181	3.3834
	GA	0.5666	0.0129	0.6231	0.0139	0.3929	0.0152	0.5804	0.0138	**0.0558**
	PSO	0.5214	0.0070	0.7371	0.0278	0.3009	0.0085	0.4970	0.0166	0.0599
	DE	0.5215	0.0070	0.7368	0.0277	0.3010	0.0085	0.4971	0.0166	0.0597
	ABC	0.5223	0.0070	0.7312	0.0261	0.3032	0.0083	0.5003	0.0159	0.0573
100	**SA**	0.9036	3.5669	0.7007	0.0175	0.3164	0.0103	0.5032	0.0119	3.6065
	GA	0.5525	0.0077	0.6171	0.0123	0.3919	0.0130	0.5818	0.0116	0.0446
	PSO	0.5132	0.0032	0.7191	0.0156	0.3023	0.0047	0.5011	0.0085	0.0320
	DE	0.5132	0.0032	0.7188	0.0155	0.3025	0.0047	0.5012	0.0085	0.0319
	ABC	0.5135	0.0033	0.7148	0.0147	0.3041	0.0046	0.5037	0.0083	**0.0309**
500	**SA**	1.0036	4.5308	0.6900	0.0121	0.3093	0.0071	0.5013	0.0062	4.5563
	GA	0.5410	0.0038	0.6156	0.0111	0.3881	0.0114	0.5774	0.0095	0.0357
	PSO	0.5018	0.0005	0.7031	0.0030	0.2990	0.0009	0.4993	0.0019	**0.0063**
	DE	0.5018	0.0005	0.7030	0.0030	0.2990	0.0009	0.4994	0.0019	**0.0063**
	ABC	0.5020	0.0005	0.7007	0.0030	0.3002	0.0009	0.5012	0.0019	**0.0063**

Çizelge 6.16. Parametreleri $\theta = (\beta, q_0, q_1, q_2) = (0.7, 0.95, 0.8, 0.6)$ olan Kesikli iki değişkenli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

n	**Yöntem**	$\hat{\beta}$		$\hat{q}_0$		$\hat{q}_1$		$\hat{q}_2$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	1.1650	3.9421	0.9180	0.0155	0.7878	0.0147	0.5904	0.0127	3.9850
	GA	0.6223	0.0189	0.8452	0.0197	0.7382	0.0128	0.5443	0.0105	0.0620
	PSO	0.7478	0.0162	0.9501	0.0028	0.8077	0.0065	0.6035	0.0090	0.0345
	DE	0.7474	0.0161	0.9499	0.0028	0.8076	0.0065	0.6033	0.0090	0.0344
	ABC	0.7448	0.0154	0.9497	0.0028	0.8066	0.0064	0.6019	0.0089	**0.0336**
50	**SA**	1.1751	4.1445	0.9173	0.0193	0.7853	0.0126	0.5870	0.0099	4.1863
	GA	0.6077	0.0182	0.8427	0.0205	0.7394	0.0115	0.5443	0.0092	0.0594
	PSO	0.7145	0.0051	0.9489	0.0015	0.8018	0.0026	0.6004	0.0035	0.0127
	DE	0.7145	0.0051	0.9488	0.0015	0.8019	0.0026	0.6005	0.0034	0.0127
	ABC	0.7122	0.0050	0.9485	0.0015	0.8008	0.0026	0.5992	0.0034	**0.0125**
100	**SA**	1.3027	5.3006	0.9113	0.0249	0.7787	0.0153	0.5918	0.0098	5.3506
	GA	0.5966	0.0191	0.8406	0.0212	0.7341	0.0116	0.5428	0.0085	0.0604
	PSO	0.7093	0.0023	0.9512	0.0008	0.8017	0.0014	0.6018	0.0018	0.0063
	DE	0.7092	0.0023	0.9510	0.0008	0.8018	0.0014	0.6018	0.0018	0.0063
	ABC	0.7071	0.0022	0.9511	0.0008	0.8004	0.0014	0.6003	0.0018	**0.0062**
500	**SA**	1.2329	4.7199	0.9191	0.0201	0.7789	0.0105	0.5911	0.0068	4.7573
	GA	0.5921	0.0195	0.8326	0.0237	0.7354	0.0107	0.5464	0.0079	0.0618
	PSO	0.7011	0.0004	0.9505	0.0002	0.7996	0.0003	0.6000	0.0003	**0.0012**
	DE	0.7010	0.0004	0.9504	0.0002	0.7996	0.0003	0.6000	0.0003	**0.0012**
	ABC	0.6988	0.0004	0.9505	0.0002	0.7980	0.0003	0.5986	0.0004	0.0013

Çizelge 6.17. Parametreleri $\theta = (\beta, q_0, q_1, q_2) = (0.9, 0.9, 0.9, 0.9)$ olan Kesikli iki değişkenli Weibull dağılımı için simülasyon sonucunda elde edilen ortalama, MSE ve Def değerleri

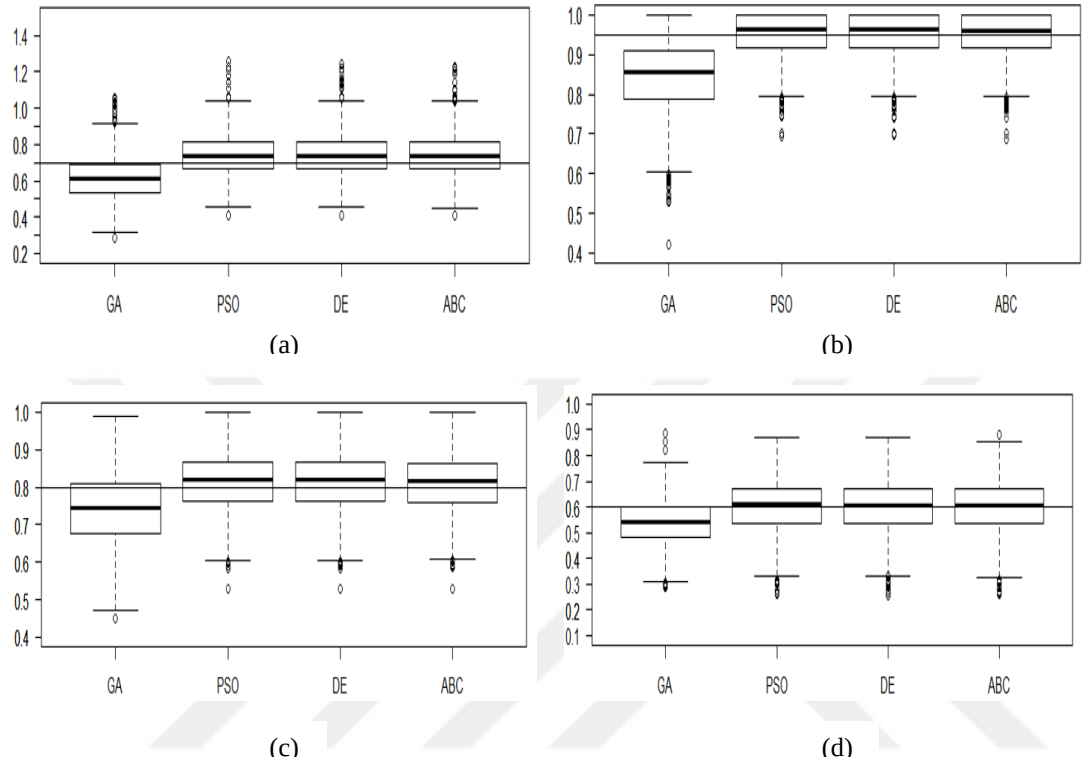
n	**Yöntem**	$\hat{\beta}$		$\hat{q}_0$		$\hat{q}_1$		$\hat{q}_2$		
		Ort.	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Ort.**	**MSE**	**Def**
20	**SA**	1.2523	3.3520	0.8694	0.0132	0.8637	0.0122	0.8646	0.0120	3.3894
	GA	0.6484	0.0774	0.7847	0.0217	0.7447	0.0338	0.7449	0.0334	0.1663
	PSO	0.9561	0.0253	0.9065	0.0021	0.8994	0.0023	0.9005	0.0023	0.0319
	DE	0.9549	0.0248	0.9063	0.0021	0.8991	0.0023	0.9004	0.0022	0.0314
	ABC	0.9353	0.0200	0.9028	0.0020	0.8952	0.0022	0.8964	0.0022	**0.0266**
50	**SA**	1.4098	4.6789	0.8606	0.0194	0.8596	0.0190	0.8603	0.0184	4.7357
	GA	0.6366	0.0798	0.7835	0.0210	0.7423	0.0328	0.7421	0.0330	0.1667
	PSO	0.9183	0.0082	0.9006	0.0008	0.9000	0.0008	0.8997	0.0008	0.0107
	DE	0.9177	0.0081	0.9005	0.0008	0.8999	0.0008	0.8995	0.0008	0.0106
	ABC	0.9046	0.0073	0.8977	0.0009	0.8968	0.0008	0.8966	0.0008	**0.0098**
100	**SA**	1.4376	4.9725	0.8641	0.0159	0.8634	0.0149	0.8628	0.0172	5.0205
	GA	0.6344	0.0806	0.7831	0.0219	0.7384	0.0348	0.7456	0.0314	0.1687
	PSO	0.9099	0.0040	0.9004	0.0004	0.9003	0.0004	0.9007	0.0004	0.0052
	DE	0.9096	0.0040	0.9003	0.0004	0.9002	0.0004	0.9007	0.0004	0.0052
	ABC	0.8968	0.0037	0.8974	0.0005	0.8972	0.0004	0.8976	0.0004	**0.0050**
500	**SA**	1.2964	3.7927	0.8697	0.0100	0.8677	0.0116	0.8656	0.0142	3.8284
	GA	0.6271	0.0837	0.7768	0.0220	0.7428	0.0324	0.7429	0.0325	0.1706
	PSO	0.9006	0.0007	0.8996	0.0001	0.9000	0.0001	0.9003	0.0001	**0.0009**
	DE	0.9000	0.0007	0.8995	0.0001	0.8998	0.0001	0.9001	0.0001	**0.0010**
	ABC	0.8884	0.0011	0.8967	0.0001	0.8969	0.0001	0.8972	0.0001	0.0015

Çizelge 6.15-6.17'den, örneklem büyüklüğü arttıkça PSO, DE ve ABC algoritmalarında Def değerlerinin düştüğü görülmüştür. Başka bir ifadeyle, örneklem büyüklüğü arttıkça MSE kriterine göre daha etkin parametre tahminleri elde edilmiştir. Ele alınan diğer metasezgisel yöntemlerden farklı olarak SA algoritması tek çözüme dayalı bir algoritma olduğundan tutarlı sonuçlar vermemiştir.

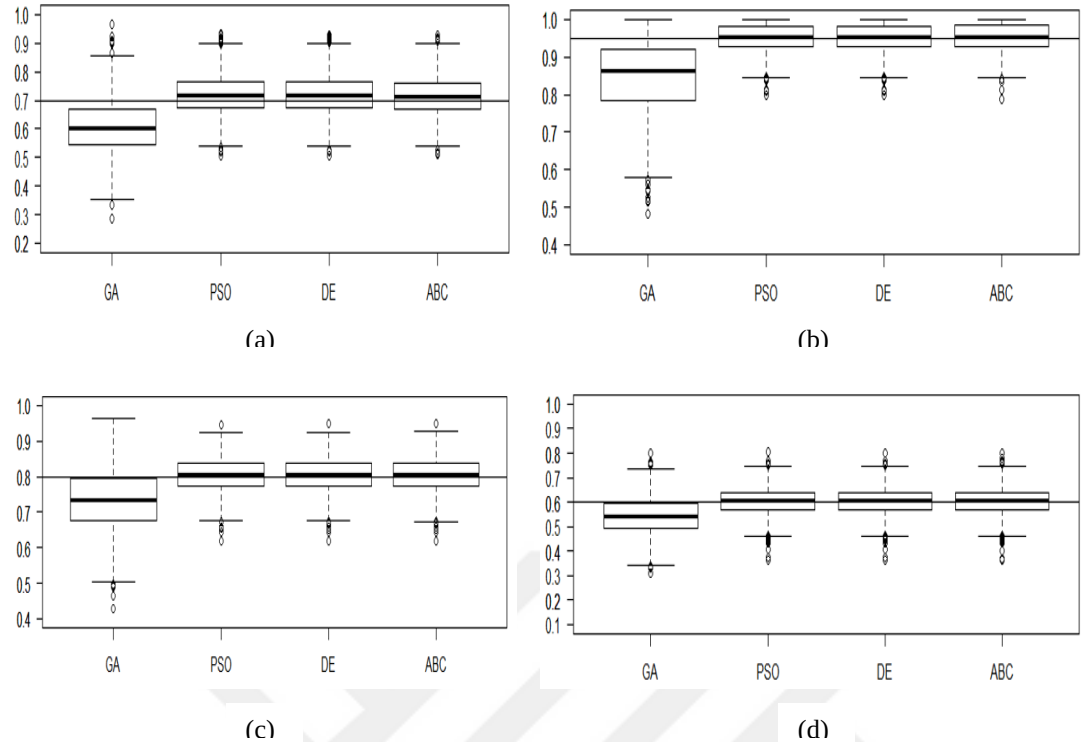
Tüm örneklem büyüklüklerinde özellikle büyük örneklemler ($n = 100$ ve 500) için PSO, DE ve ABC algoritmaları birbirine yakın Def değerleri ile benzer performans sergilemiştir. $n = 50$ ve 100 için çoğu durumda ABC algoritması en küçük Def değerleri ile en iyi performansı sergilemiştir. Sonuç olarak kesikli çok değişkenli Weibull dağılımının parametrelerinin tahmininde PSO, DE veya ABC algoritmalarının kullanılması önerilmektedir.

Simülasyon çalışmalarını görselleştirebilmek amacıyla $(\beta, q_0, q_1, q_2) = (0.7, 0.95, 0.8, 0.6)$ parametreleri ve $n = 20, 50, 100, 500$ örneklem büyüklükleri için ele alınan metasezgisel algoritmalarının parametre tahminlerinin örnek dağılımlarının kutu grafikleri sırasıyla Şekil 6.3-6.6'da sunulmuştur. Bu

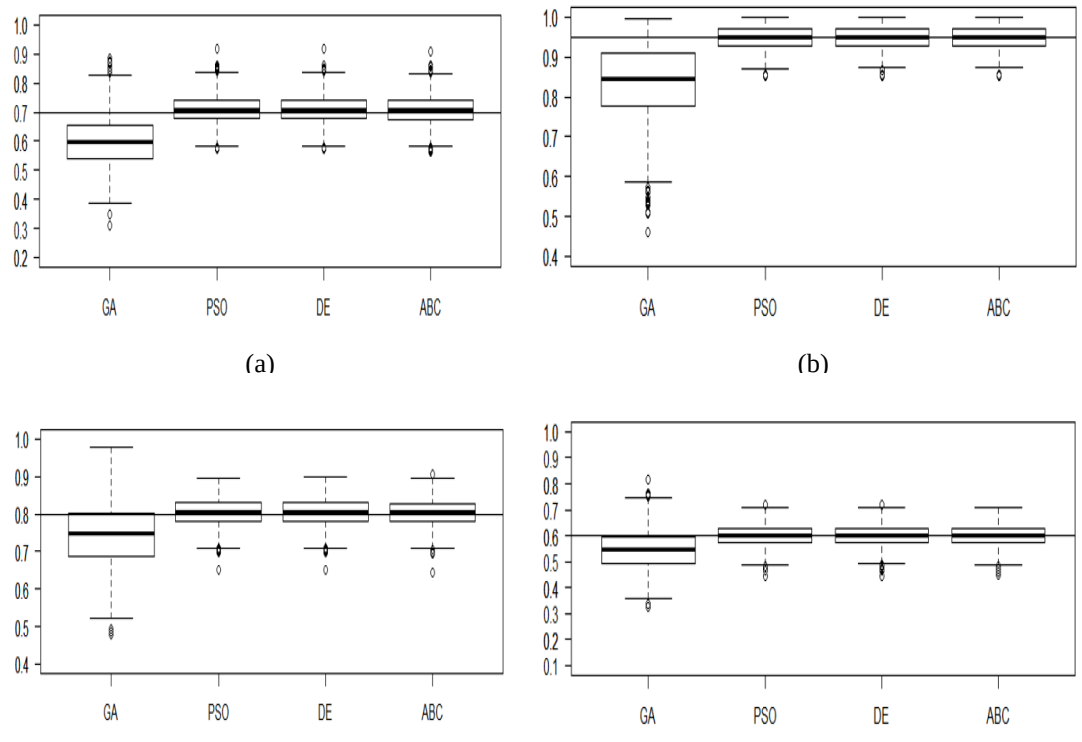
grafiklere, çok fazla uç değerler verdiğinden ve grafiğin yapısını bozduğundan SA sonuçları dahil edilmemiştir. Bu grafiklerden, özellikle PSO, DE ve ABC algoritmalarının birbirine çok yakın tahmin değerleri verdiği görülmektedir.



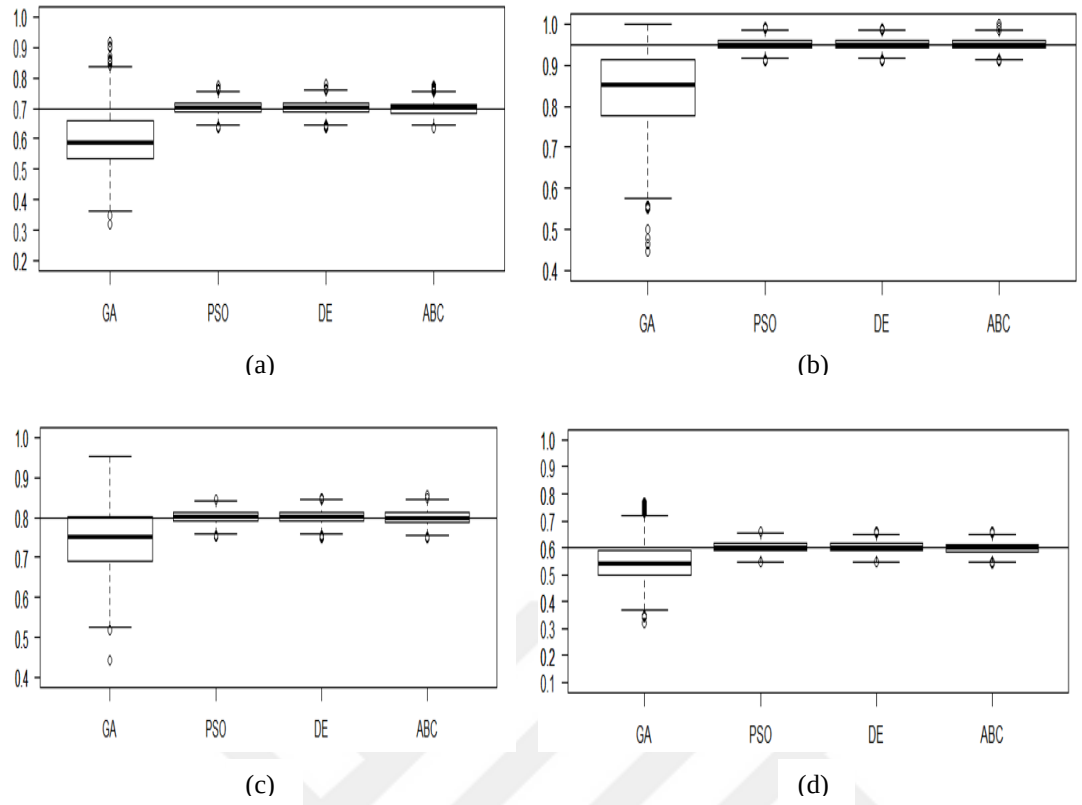
Şekil 6.3. $(\beta, q_0, q_1, q_2) = (0.7, 0.95, 0.8, 0.6)$ ve $n = 20$ durumunda (a) $\hat{\beta}$, (b) $\hat{q}_0$, (c) $\hat{q}_1$ ve (d) $\hat{q}_2$ için kutu grafiği



Şekil 6.4. $(\beta, q_0, q_1, q_2) = (0.7, 0.95, 0.8, 0.6)$ ve $n = 50$ durumunda (a) $\hat{\beta}$, (b) $\hat{q}_0$, (c) $\hat{q}_1$ ve (d) $\hat{q}_2$ için kutu grafiği



Şekil 6.5. $(\beta, q_0, q_1, q_2) = (0.7, 0.95, 0.8, 0.6)$ ve $n = 100$ durumunda (a) $\hat{\beta}$, (b) $\hat{q}_0$, (c) $\hat{q}_1$ ve (d) $\hat{q}_2$ için kutu grafiği



Şekil 6.6. $(\beta, q_0, q_1, q_2) = (0.7, 0.95, 0.8, 0.6)$ ve $n = 500$ durumunda (a) $\hat{\beta}$, (b) $\hat{q}_0$, (c) $\hat{q}_1$ ve (d) $\hat{q}_2$ için kutu grafiği

6.2.3. Gerçek veri uygulaması

Bu alt kesimde, Lee ve Cha (2015) tarafından rapor edilen 1996-2011 döneminde iki İtalyan futbol takımı "ACF Firontina" ve "Juventus" arasında oynanan 26 İtalyan Seri A futbol maçı skorlarını içeren bir veri kümesi ele alınmış ve Çizelge 6.18'de verilmiştir.

Çizelge 6.18. Futbol maçı skorlarına ilişkin veri kümesi

Gözlem no:	**ACF Firontina (X_1)**	**Juventus (X_2)**	**Gözlem no:**	**ACF Firontina (X_1)**	**Juventus (X_2)**
1	1	2	14	1	2
2	0	0	15	1	1
3	1	1	16	1	3
4	2	2	17	3	3
5	1	1	18	0	1
6	0	1	19	1	1
7	1	1	20	1	2
8	3	2	21	1	0
9	1	1	22	3	0
10	2	1	23	1	2
11	1	2	24	1	1
12	3	3	25	0	1
13	0	1	26	0	1

Metasezgisel yöntemler kullanılarak bu veri kümesi için ML parametre tahmin değerleri elde edilmiş ve Çizelge 6.19'da sunulmuştur. Metasezgisel yöntemlerin karşılaştırılmasında $\ln L$, *AIC* ve *BIC* değerleri kullanılmıştır. Elde edilen parametre tahminleri için hesaplanan $\ln L$, *AIC* ve *BIC* değerleri Çizelge 5.19' da verilmiştir.

Çizelge 6.19. Futbol maçı skorları veri kümesi için parametre tahmin değerleri ve $\ln L$, *AIC* ve *BIC* değerleri

Yöntem	$\hat{\beta}$	$\hat{q}_0$	$\hat{q}_1$	$\hat{q}_2$	$\ln L$	***AIC***	***BIC***
SA	1.7982	0.9277	0.7691	0.7459	-67.2350	134.4708	134.4711
GA	1.8702	0.9360	0.7341	0.7881	-66.4195	132.8398	132.8401
PSO	2.1527	0.9307	0.8260	0.8725	**-65.2195**	**130.4398**	**130.4401**
DE	2.1588	0.9326	0.8264	0.8719	-65.2202	130.4412	130.4415
ABC	2.0748	0.9134	0.8177	0.8745	-65.2856	130.5720	130.5723

Çizelge 6.19'dan, ele alınan metasezgisel yöntemler arasında en büyük $\ln L$ ve en küçük *AIC* ve *BIC* değerlerine sahip algoritma PSO algoritmasıdır. PSO, DE ve ABC algoritmaları birbirine çok yakın $\ln L$, *AIC* ve *BIC* değerleri verdiğinden dolayı bu veri kümesi için bu yöntemlerin kullanılması uygundur.

7. SONUÇLAR VE ÖNERİLER

Bu bölümde, çalışmadan elde edilen sonuçlara ve önerilere yer verilmiştir.

7.1 Sonuçlar

Bu doktora tez çalışmasında, parametre tahminleri analitik olarak elde edilemeyen sürekli dağılımlardan 3-p Weibull, 3-p Gamma, çok değişkenli Gamma ve kesikli dağılımlardan kesikli tek ve çok değişkenli Weibull dağılımlarının parametre tahminleri metasezgisel yöntemler kullanılarak elde edilmiştir.

3-p Weibull dağılımının parametrelerinin en çok olabilirlik (ML) tahminini elde etmek için Levy uçuşlu Yapay Arı Kolonisi (LABC) algoritması tanıtılmıştır. LABC algoritmasının performansı, çeşitli parametre değerleri dikkate alınarak simülasyon çalışmaları aracılığıyla Def kriterine göre değerlendirilmiştir. Levy uçuşlarının ABC algoritmasının performansını önemli ölçüde arttırdığı görülmüştür. Ayrıca, yapılan istatistiksel testler ile LABC'nin Weibull dağılımının parametrelerinin tahminlerini elde etmede gösterdiği performansın arama alanına bağlı olmadığı görülmüştür. Bu durum, arama uzayı açısından algoritmanın sağlamlığının bir göstergesidir ve gerçek veri uygulamaları için oldukça önemlidir. LABC'nin performansı diğer metasezgisel yöntemlerden ABC, PSO, LPSO, DE, SA ve GA'nın performansı ile karşılaştırıldığında LABC'nin özellikle geniş arama uzayında daha iyi parametre tahmin sonuçları verdiği belirlenmiştir. Gerçekleştirilen bir gerçek veri uygulaması yardımıyla LABC'nin diğer metasezgisel algoritmalara göre üstünlüğü gösterilmiştir. Bu sonuçlara göre, LABC'nin hemen hemen tüm durumlarda 3-p Weibull dağılımının parametre tahmin değerleri için en iyi ortalama tahminleri sağladığı söylenebilir. Bu nedenle, 3-p Weibull dağılımının parametre tahminleri için LABC algoritmasının, diğer metasezgisel yöntemlere göre tercih edilebilir bir yöntem olduğu görülmüştür.

3-p Gamma dağılımının parametrelerinin ML tahmin edicilerinin bulmak için metasezgisel yöntemlerden SA, GA, PSO, DE ve ABC algoritması ve klasik optimizasyon algoritmalarından SQP ve IPA yöntemleri kullanılmıştır. Bu yöntemleri kullanarak elde edilen tahmin edicilerin etkinliğini karşılaştırmak için Monte-Carlo simülasyon çalışmaları yapılmıştır. Ele alınan yöntemler, hesaplama zamanı ve çözüm kalitesi açısından karşılaştırılmıştır. SQP ve IPA çözüm zamanı açısından çok iyi performans sergilerken çözüm kalitesi açısından çok kötü performans sergilemiştir.

Bundan dolayı, çözüm kalitesi açısından metasezgisel yöntemlerin kullanılması uygun bulunmuştur. Metasezgisel yöntemler arasında DE algoritmasının Def kriterine göre çözüm kalitesi açısından ve PSO algoritmasının hesaplama zamanı açısından diğer yöntemlere göre genellikle daha iyi olduğu gözlenmiştir. Ayrıca, metasezgisel yöntemlerden elde edilen çözüm kalitesi, hesaplama zamanı, basitlik ve sağlamlılık kriterleri açısından performanslarına göre ele alınan yöntemler TOPSIS yöntemi kullanılarak sıralanmıştır. TOPSIS yöntemi sonuçlarına göre, tüm kriterler göz önüne alındığında DE algoritması 3-p Gamma dağılımının parametrelerinin tahmini için en iyi algoritma olarak belirlenmiştir. Önerilen metasezgisel yöntemleri uygulanabilirliğini göstermek amacıyla gerçek bir veri kümesi üzerinde uygulama sunulmuştur. DE algoritmasının diğer algoritmalara göre gerçek veri kümesi uygulamasında da üstün olduğu görülmüştür. Sonuç olarak, DE algoritmasının 3-p Gamma dağılımının parametrelerinin ML tahmininde en etkili sonuçları verdiği söylenebilir.

3-p Gamma dağılımının parametrelerinin ML ve LS tahminlerini bulmak için DE algoritması kullanılmıştır. Tüm metasezgisel yöntemlerde olduğu gibi DE algoritmasında arama uzayının belirlenmesi önemli bir sorundur. DE algoritmasının performansı hem çözüm kalitesi hem de hesaplama zamanı açısından arama uzayından büyük ölçüde etkilenmektedir. Literatürde, genel olarak rastgele arama alanları kullanılmaktadır. Rastgele belirlenen arama alanı gerçek değerleri içermeyebildiğinden global en uygun çözüm bulunamaz. Bu durumun üstesinden gelebilmek için, yeni bir DE algoritması yaklaşımı önerilmiştir. DE algoritması için dar ve gerçek değerlere sahip bir arama alanı oluşturmak için MML ve PL yöntemlerine dayalı güven aralıkları kullanılmıştır. Önerilen yeni yaklaşımın performansını test etmek amacıyla kapsamlı bir Monte-Carlo simülasyon çalışması yapılmıştır. Simülasyon sonuçları, önerilen DE algoritma yaklaşımının parametrelerin hem ML hem de LS tahmininde daha iyi sonuçlar verdiğini göstermektedir. Ayrıca, önerilen DE yaklaşımının uygulanabilirliğini göstermek için gerçek bir veri kümesi üzerinde uygulama yapılmıştır. Önerilen yaklaşımın diğer metasezgisel algoritmalara göre üstünlüğü gerçek veri uygulaması ile de desteklenmiştir.

Kesikli Weibull dağılımının parametrelerinin ML tahmin edicileri elde etmek için metasezgisel yöntemlerden SA, GA, PSO, DE ve ABC algoritmaları ele alınmıştır. Bu yöntemleri kullanılarak elde edilen tahmin edicilerin etkinliğini araştırmak için Monte-Carlo simülasyon çalışmaları yapılmış ve çözüm kalitesi açısından algoritmalar karşılaştırılmıştır. Küçük örneklemlerde GA, büyük örneklemlerde ise PSO, DE ve

ABC algoritmaları iyi sonuçlar vermiştir. Fakat, SA algoritması ile tutarlı sonuçlar elde edilememiştir. Önerilen metasezgisel yöntemlerin uygulanabilirliğini göstermek için gerçek bir veri kümesi üzerinde uygulama sunulmuş ve GA, PSO, DE ve ABC algoritmalarının kesikli Weibull dağılımının parametrelerinin ML tahmininde etkili sonuçlar verdiği gözlenmiştir.

Çok değişkenli Gamma dağılımının parametrelerini tahmini için PSO algoritması kullanılmıştır. Literatürdeki çalışmalardan, PSO algoritmasının performansının, algoritmanın parametrelerinin tasarımına göre önemli ölçüde değişebileceği gözlenmiştir. Ayrıca, algoritmanın performansını parametre tasarımlarının tüm değerlerine karşı test etmek çok zaman alacağından bu çalışmada PSO algoritmasının parametrelerini tasarlamak amacıyla Taguchi deney tasarımı yöntemi önerilmiştir. Taguchi yöntemi, çok sayıda karar değişkeninin az sayıda deneyle test edilmesine imkan sağlamaktadır. Bu sayede, PSO'nun en uygun parametre tasarımı Taguchi yöntemiyle kısa sürede belirlenmiştir. Taguchi yöntemi ile parametreleri tasarlanan PSO algoritmasının çok değişkenli Gamma dağılımının parametrelerinin ML tahminindeki etkinliği Monte-Carlo simülasyon çalışmaları ile test edilmiştir. Elde edilen sonuçlara göre, önerilen yaklaşımın doğru tahminler sağladığı ve çok değişkenli Gamma dağılımının parametre tahmini için oldukça tatmin edici sonuçlar verdiği görülmüştür.

Son olarak, çok değişkenli kesikli Weibull dağılımının parametrelerinin ML tahmin edicilerini bulmak için metasezgisel yöntemlerden SA, GA, PSO, DE ve ABC algoritması ele alınmıştır. Bu yöntemleri kullanılarak elde edilen tahmin edicilerin etkinliğini araştırmak için Monte-Carlo simülasyon çalışmaları yapılmış ve çözüm kalitesi açısından algoritmalar karşılaştırılmıştır. Tüm örneklem büyüklükleri için PSO, DE ve ABC algoritmaları birbirine yakın Def değerleri vererek benzer performans sergilemiştir. Orta büyüklükteki örneklemler için çoğu durumda ABC algoritması en iyi performansı sergilemiştir. Fakat, SA algoritması ile tutarlı sonuçlar elde edilememiştir. Sonuç olarak, çok değişkenli kesikli Weibull dağılımının parametrelerinin tahmininde PSO, DE veya ABC algoritmalarının kullanılması önerilebilir. Önerilen metasezgisel yöntemlerin uygulanabilirliğini göstermek için bir veri kümesi uygulaması da ayrıca sunulmuş ve PSO, DE ve ABC algoritmalarının çok değişkenli kesikli Weibull dağılımının parametrelerinin ML tahmininde etkili sonuçlar verdiği gözlenmiştir.

Sonuç olarak, çalışmada ele alınan metasezgisel yöntemlerin ele alınan tek değişkenli ve çok değişkenli kesikli/sürekli dağılımların parametrelerinin tahmininde oldukça kullanışlı olduğu söylenebilir.

7.2. Öneriler

Bu çalışmada ele alınan ve önerilen tüm metasezgisel yöntemler parametre tahminleri analitik olarak elde edilemeyen dağılımların parametrelerinin tahmini süreçlerinde kullanılabilir ve diğer iteratif yöntemler ile karşılaştırılabilir.

Sürekli dağılımlardan 3-p Weibull, 3-p Gamma, çok değişkenli Gamma ve kesikli dağılımlardan kesikli tek değişkenli ve kesikli çok değişkenli Weibull dağılımlarına ilişkin parametre tahminleri farklı metasezgisel yöntemler yardımıyla elde edilebilir ve bu doktora tez çalışmasının sonuçları ile karşılaştırılabilir.

Levy uçuşları ile diğer metasezgisel algoritmalar birleştirilerek, algoritmaların performanslarını arttırmak amacıyla çeşitli uygulamalar yapılabilir.

Bu çalışmada önerilen MML ve PL yöntemleri yardımıyla elde edilen güven aralıkları kullanılarak oluşturulan arama uzayları, çeşitli metasezgisel yöntemler için uygulanarak global optimal noktaya hızlı bir yakınsama sağlanabilir.

Tüm metasezgisel algoritmalar ile yapılacak çalışmalarda, algoritmaların parametrelerinin tasarlanmasında Taguchi deney tasarımı yöntemi kullanılarak daha etkin sonuçlar elde edilebilir.

KAYNAKLAR

Abbasi, B., Jahromi, A. H. E., Arkat, J. ve Hosseinkouchack, M., 2006, Estimating the parameters of Weibull distribution using simulated annealing algorithm, *Applied Mathematics and Computation*, 183 (1), 85-93.

Abbasi, B., Rabelo, L. ve Hosseinkouchack, M., 2008, Estimating parameters of the three-parameter Weibull distribution using a neural network, *European Journal of Industrial Engineering*, 2 (4), 428-445.

Abbasi, B., Niaki, S. T. A., Khalife, M. A. ve Faize, Y., 2011, A hybrid variable neighborhood search and simulated annealing algorithm to estimate the three parameters of the Weibull distribution, *Expert Systems with Applications*, 38 (1), 700-708.

Acıtaş, Ş., Aladağ, Ç. H. ve Şenoğlu, B., 2019, A new approach for estimating the parameters of Weibull distribution via particle swarm optimization: An application to the strengths of glass fibre data, *Reliability Engineering System Safety*, 183, 116-127.

Akaike, H., Petrov, B. N. ve Csaki, F., 1973, Second international symposium on information theory, Akadémiai Kiadó, Budapest.

Akay, B. ve Karaboğa, D., 2012, A modified artificial bee colony algorithm for real-parameter optimization, *Information Sciences*, 192, 120-142.

Akdi, Y., 2005, Matematiksel istatistiğe giriş, Bıçaklar Kitabevi, p. 227-266.

Akman, F., Akman, O. ve Hallam, J. W., 2010, Maximum Likelihood Estimation Using Accelerated Genetic Algorithms, *Journal of Applied Statistical Science*, 18 (3), 1-7.

Aledo, J. A., Gámez, J. A. ve Molina, D., 2016, Using metaheuristic algorithms for parameter estimation in generalized Mallows models, *Applied Soft Computing*, 38, 308-320.

Almalki, S. J. ve Nadarajah, S., 2014, Modifications of the Weibull distribution: A review, *Reliability Engineering System Safety*, 124, 32-55.

Altunkaynak, B. ve Alptekin, E., 2004, The genetic algorithm method for parameter estimation in nonlinear regression, *Gazi University Journal of Science*, 17 (2), 43-51.

Asim, M., Khan, W. M., Yeniay, Ö., Jan, M. A., Tairan, N., Hussian, H. ve Wang, G.-G., 2018, Hybrid genetic algorithms for global optimization problems, *Hacettepe Journal of Mathematics Statistics Computing*, 47 (3), 539-551.

Balakrishnan, N. ve Wang, J., 2000, Simple efficient estimation for the three-parameter gamma distribution, *Journal of Statistical Planning Inference*, 85 (1-2), 115-126.

Balakrishnan, N. ve Lai, C. D., 2009, Construction of Bivariate Distributions, In: Continuous Bivariate Distributions, Eds, *New York*: Springer, p. 179-228.

Banerjee, A. ve Abu-Mahfouz, I., 2014, A comparative analysis of particle swarm optimization and differential evolution algorithms for parameter estimation in nonlinear dynamic systems, *Chaos, Solitons Fractals*, 58, 65-83.

Barbiero, A., 2013, Parameter estimation for type III discrete Weibull distribution: a comparative study, *Journal of Probability Statistics Computing*, 2013.

Barbiero, A., 2016, A comparison of methods for estimating parameters of the type I discrete Weibull distribution, *Journal of ProbabilityStatistics its Interface*, 9 (2), 203-212.

Barbiero, A., 2018, On methods of estimation for the type II discrete Weibull distribution, *Iranian Journal of Science Technology, Transactions of Electrical Engineering*, 42 (4), 501-514.

Basak, I. ve Balakrishnan, N., 2012, Estimation for the three-parameter gamma distribution based on progressively censored data, *Statistical Methodology*, 9 (3), 305-319.

Bowman, K. ve Shenton, L., 2001, Weibull distributions when the shape parameter is defined, *Computational statistics data analysis*, 36 (3), 299-310.

Bracquemond, C. ve Gaudoin, O., 2003, A survey on discrete lifetime distributions, *International Journal of Reliability, Quality Safety Engineering*, 10 (01), 69-98.

Byrd, R. H., Gilbert, J. C. ve Nocedal, J., 2000, A trust region method based on interior point techniques for nonlinear programming, *Mathematical programming*, 89 (1), 149-185.

Carneiro, T. C., Melo, S. P., Carvalho, P. C. ve Braga, A. P. d. S., 2016, Particle swarm optimization method for estimation of Weibull parameters: a case study for the Brazilian northeast region, *Renewable energy*, 86, 751-759.

Carpenter, M. ve Diawara, N., 2007, A multivariate gamma distribution and its characterizations, *American Journal of Mathematical Management Sciences*, 27 (3-4), 499-507.

Casella, G. ve Berger, R. L., 2002, Statistical inference, *Pacific Grove*, CA: Duxbury, p. 311-373.

Cherian, K., 1941, A bivariate correlated gamma-type distribution function, *Journal of the Indian Mathematical Society*, 5, 133-144.

Clerc, M., 2010, Particle swarm optimization, *USA*, John Wiley & Sons, p. 37-51.

Clifford, C. ve Jones, W., 1982, Modified moment and maximum likelihood estimators for parameters of the three-parameter gamma distribution, *Communications in Statistics-Simulation*, 11 (2), 197-216.

Crowder, M. J., Kimber, A. C., Smith, R. L. ve Sweating, T. J., 1991, The Statistical Analysis of Reliability Data, *Chapman and Hall, London*, p. 81.

Das, S., Mullick, S. S. ve Suganthan, P. N., 2016, Recent advances in differential evolution–an updated survey, *Swarm Evolutionary Computation*, 27, 1-30.

Dorigo, M., Maniezzo, V. ve Colorni, A., 1996, Ant system: optimization by a colony of cooperating agents, *IEEE Transactions on Systems, man,cybernetics, Part B: Cybernetics*, 26 (1), 29-41.

Dumonceaux, R. ve Antle, C. E., 1973, Discrimination between the log-normal and the Weibull distributions, *Technometrics*, 15 (4), 923-926.

Eberhart, R. ve Kennedy, J., 1995, A new optimizer using particle swarm theory, *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 39-43.

El-Morshedy, M., Eliwa, M., El-Gohary, A. ve Khalil, A., 2020, Bivariate exponentiated discrete Weibull distribution: statistical properties, estimation, simulation and applications, *Mathematical Sciences*, 14 (1), 29-42.

Eliwa, M. ve El-Morshedy, M., 2018, Bivariate Discrete Inverse Weibull Distribution.

Glover, F., 1989, Tabu search—part I, *ORSA Journal on computing*, 1 (3), 190-206.

Goldberg, D. E. ve Holland, J. H., 1988, Genetic algorithms and machine learning, *Machine learning*, 3 (2), 95-99.

Hakli, H. ve Uguz, H., 2013, Levy flight distribution for scout bee in artificial bee colony algorithm, *Lecture Notes on Software Engineering*, 1 (3), 254.

Haklı, H. ve Uğuz, H., 2014, A novel particle swarm optimization algorithm with Levy flight, *Applied Soft Computing*, 23, 333-345.

Hirose, H., 1995, Maximum likelihood parameter estimation in the three-parameter gamma distribution, *Computational Statistics & Data Analysis*, 20 (4), 343-354.

Holland, J. H., 1975, Adaptation in natural and artificial systems Ann Arbor, *The University of Michigan Press*, 1.

Hwang, C.-L. ve Yoon, K., 1981, Multiple attribute decision making: methods and applications a state-of-the-art survey, *New York*, Springer Science & Business Media, p. 58-191.

Jensen, D., 1970, The joint distribution of quadratic forms and related distributions, *Australian Journal of Statistics*, 12 (1), 13-22.

Jha, S. K., 2016, Computation of maximum likelihood estimates in threeparameter Weibull for censored data, In: Recent Advances in Mathematics, Statistics and Computer Science, Eds: World Scientific, p. 93-102.

Johnson, D. S., Aragon, C. R., McGeoch, L. A. ve Schevon, C., 1989, Optimization by simulated annealing: An experimental evaluation; part I, graph partitioning, *Operations research*, 37 (6), 865-892.

Johnson, N. L., Kotz, S. ve Balakrishnan, N., 1994, Univariate continuous distributions, *New York:*, John Wiley & Sons, p. 337-695.

Kantar, Y. M. ve Şenoğlu, B., 2008, A comparative study for the location and scale parameters of the Weibull distribution with given shape parameter, *Computers Geosciences*, 34 (12), 1900-1909.

Karaboga, D., 2005, An idea based on honey bee swarm for numerical optimization, *Technical report-tr06, Erciyes University, Engineering Eaculty, Computer Engineering Department*.

Karaboğa, D. ve Akay, B., 2009, A comparative study of artificial bee colony algorithm, *Applied Mathematics and Computation*, 214 (1), 108-132.

Karaboğa, D. ve Öztürk, C., 2011, A novel clustering approach: Artificial Bee Colony (ABC) algorithm, *Applied Soft Computing*, 11 (1), 652-657.

Karaboğa, D., 2014, Yapay Zeka Optimizasyon Algoritmaları, *Ankara*, Nobel Akademik Yayıncılık, p. 1-191.

Karaboğa, D., Gorkemli, B., Ozturk, C. ve Karaboga, N., 2014, A comprehensive survey: artificial bee colony (ABC) algorithm and applications, *Artificial Intelligence Review*, 42 (1), 21-57.

Karakoca, A. ve Pekgör, A., 2019, Maximum Likelihood Estimation of the Parameters of Progressively Type-2 Censored Samples From Weibull Distribution Using Genetic Algorithm, *Akademik Platform Mühendislik ve Fen Bilimleri Dergisi*, 7 (2), 189-199.

Kennedy, J. ve Eberhart, R., 1995, Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks, *Perth, WA, Australia*, 1942–1948.

Khan, M. A., Khalique, A. ve Abouammoh, A., 1989, On estimating parameters in a discrete Weibull distribution, *IEEE transactions on reliability*, 38 (3), 348-350.

Kirkpatrick, S., Gelatt, C. D. ve Vecchi, M. P., 1983, Optimization by simulated annealing, *science*, 220 (4598), 671-680.

Krishna, H. ve Pundir, P. S., 2009, Discrete Burr and discrete Pareto distributions, *Statistical Methodology*, 6 (2), 177-188.

Kundu, D. ve Nekoukhou, V., 2018, Univariate and bivariate geometric discrete generalized exponential distributions, *Journal of Statistical Theory Practice*, 12 (3), 595-614.

Kundu, D. ve Nekoukhou, V., 2019, On bivariate discrete Weibull distribution, *Communications in Statistics-Theory*, 48 (14), 3464-3481.

Lakshmi, V. ve Vaidyanathan, V., 2016, Three-parameter gamma distribution: Estimation using likelihood, spacings and least squares approach, *Journal of Statistics Management Systems*, 19 (1), 37-53.

Lawless, J. F., 2011, Statistical models and methods for lifetime data, *Canada*, John Wiley & Sons, p. 203.

Lee, C.-Y. ve Yao, X., 2001, Evolutionary algorithms with adaptive lévy mutations, *Proceedings of the 2001 Congress on Evolutionary Computation*, 568-575.

Lee, H. ve Cha, J. H., 2015, On two general classes of discrete bivariate distributions, *The American Statistician*, 69 (3), 221-230.

Mantegna, R. N., 1994, Fast, accurate algorithm for numerical simulation of Levy stable stochastic processes, *Physical Review E*, 49 (5), 4677.

Mashwani, W. K., Zaib, A., Yeniay, Ö., Shah, H., Tairan, N. M. ve Sulaiman, M., 2018, Hybrid Constrained Evolutionary Algorithm for Numerical Optimization Problems, *Hacettepe Journal of Mathematics Statistics Computing*, 48 (3), 931-950.

Mathai, A. ve Moschopoulos, P., 1992, A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics*, 44 (1), 97-106.

Mathworks, 2020, https://www.mathworks.com/help/optim/ug/constrained-nonlinear-optimization-algorithms.html#brnox01): [10 Ocak 2020].

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. ve Teller, E., 1953, Equation of state calculations by fast computing machines, *The Journal of Chemical Physics*, 21 (6), 1087-1092.

Murthy, D., Xie, M. ve Jiang, R., 2004, Weibull models, *New York*, John Wiley p. 83.

Nagatsuka, H., Kamakura, T. ve Balakrishnan, N., 2013, A consistent method of estimation for the three-parameter Weibull distribution, *Computational statistics data analysis*, 58, 210-226.

Nakagawa, T. ve Osaki, S., 1975, The discrete Weibull distribution, *IEEE transactions on reliability*, 24 (5), 300-301.

Nalbant, M., Gökkaya, H. ve Sur, G., 2007, Application of Taguchi method in the optimization of cutting parameters for surface roughness in turning, *Materials design*, 28 (4), 1379-1385.

Nekoukhou, V., Alamatsaz, M. ve Bidram, H., 2013, Discrete generalized exponential distribution of a second type, *Statistics Computing*, 47 (4), 876-887.

Nekoukhou, V. ve Kundu, D., 2017, Bivariate discrete generalized exponential distribution, *Statistics Computing*, 51 (5), 1143-1158.

Nian, C., Yang, W. ve Tarng, Y., 1999, Optimization of turning operations with multiple performance characteristics, *Journal of Materials Processing Technology*, 95 (1-3), 90-96.

Okafor, E. G., Ezugwu, O. E., Jemitola, P. O., Sun, Y. ve Lu, Z., 2018, Weibull parameter estimation using particle swarm optimization algorithm, *International Journal of Engineering Technology*, 7 (3.32), 7-10.

Opricovic, S. ve Tzeng, G.-H., 2004, Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS, *European journal of operational research*, 156 (2), 445-455.

Örkcü, H., Aksoy, E. ve Dogan, M. İ., 2015a, Estimating the parameters of 3-p Weibull distribution through differential evolution, *Applied Mathematics and Computation*, 251, 211-224.

Örkcü, H., Özsoy, V. S., Aksoy, E. ve Dogan, M. I., 2015b, Estimating the parameters of 3-p Weibull distribution using particle swarm optimization: A comprehensive experimental comparison, *Applied Mathematics and Computation*, 268, 201-226.

Özsoy, V. S., Örkcü, H. H. ve Bal, H., 2017, Particle Swarm Optimization applied to parameter estimation of thefour-parameter Burr III distribution, *Iranian Journal of Science and Technology, Transactions A: Science*, 42, 1-15.

Özsoy, V. S., Ünsal, M. G. ve Örkcü, H. H., 2020, Use of the heuristic optimization in the parameter estimation of generalized gamma distribution: comparison of GA, DE, PSO and SA methods, *Computational Statistics & Data Analysis*, 1-31.

Pólya, G., 1945, How to solve it, *Oxford*, Princeton University Press, p. 1-242.

Price, K., Storn, R. M. ve Lampinen, J. A., 2006, Differential evolution: a practical approach to global optimization, *Germany*, Springer Science & Business Media, p. 37-135.

Rezaee Jordehi, A. ve Jasni, J., 2013, Parameter selection in particle swarm optimisation: a survey, *Journal of Experimental & Theoretical Artificial Intelligence*, 25 (4), 527-542.

Roussas, G. G., 1973, A first course in mathematical statistics, *Cambridge*, Cambridge University Press, p. 1-42.

Royen, T., 1991, Expansions for the multivariate chi-square distribution, *Journal of Multivariate Analysis*, 38 (2), 213-232.

Saab, Y. G. ve Rao, V. B., 1991, Combinatorial optimization by stochastic evolution, *IEEE Transactions on Computer-Aided Design of Integrated Circuits System*, 10 (4), 525-535.

Sarmanov, I., 1970, On the approximate calculation of the coefficient of correlation between two functions of dependent random variates, *Mathematical notes of the Academy of Sciences of the USSR*, 7 (5), 372-377.

Schittkowski, K., 1986, NLPQL: A FORTRAN subroutine solving constrained nonlinear programming problems, *Annals of operations research*, 5 (2), 485-500.

Schwaab, M., Biscaia Jr, E. C., Monteiro, J. L. ve Pinto, J. C., 2008, Nonlinear parameter estimation through particle swarm optimization, *Chemical Engineering Science*, 63 (6), 1542-1552.

Shao, Q., 2004, Notes on maximum likelihood estimation for the three-parameter Burr XII distribution, *Computational statistics and data analysis*, 45 (3), 675-687.

Shi, Y. ve Eberhart, R. C., 1998, Parameter selection in particle swarm optimization, *International Conference on Evolutionary Programming*, USA, 591-600.

Shin, J.-Y., Heo, J.-H., Jeong, C. ve Lee, T., 2014, Meta-heuristic maximum likelihood parameter estimation of the mixture normal distribution for hydro-meteorological variables, *Stochastic environmental research and risk assessment*, 28 (2), 347-358.

Stone, M., 1979, Comments on model selection criteria of Akaike and Schwarz, *Journal of the Royal Statistical Society. Series B*, 41 (2), 276-278.

Storn, R., 1996, On the usage of differential evolution for function optimization, *Fuzzy Information Processing Society, Biennial Conference of the North American*, 519-523.

Storn, R. ve Price, K., 1997, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, *Journal of global optimization*, 11 (4), 341-359.

Swain, J. J., Venkatraman, S. ve Wilson, J. R., 1988, Least-squares estimation of distribution functions in Johnson's translation system, *Journal of Statistical Computation Simulation*, 29 (4), 271-297.

Şahin, A. ve Pehlivan, N. Y., 2017, Evaluation of life quality by integrated method of AHP and TOPSIS based on interval type-2 fuzzy sets, *Hacettepe Journal of Mathematics and Statistics*, 46 (3), 519-531.

Şenoğlu, B. ve Acıtaş, Ş., 2011, İstatistiksel deney tasarımı: sabit etkili modeller, *Ankara*, Nobel Yayınevi, p. 3-7.

Taguchi, G., 1987, System of experimental design; engineering methods to optimize quality and minimize costs, *USA*, UNIPUB/Kraus International Publications, p. 1-657.

Talbi, E.-G., 2009, Metaheuristics: from design to implementation, *Canada*, John Wiley & Sons, p. 1-308.

Teimouri, M. ve Gupta, A., 2013, On the three-parameter Weibull distribution shape parameter estimation, *Journal of Data Science*, 11 (3), 403-414.

Tiku, M., 1967, Estimating the mean and standard deviation from a censored normal sample, *Biometrika*, 54 (1-2), 155-165.

Tiku, M. L. ve Akkaya, A. D., 2004, Robust estimation and hypothesis testing, *New Delhi*, New Age International Publishers, p. 22-54.

Vaidyanathan, V. ve Lakshmi, R. V., 2015, Parameter Estimation in Multivariate Gamma Distribution, *Statistics, Optimization Information Computing*, 3 (2), 147-159.

Vera, J. F. ve Díaz-García, J. A., 2008, A global simulated annealing heuristic for the three-parameter lognormal maximum likelihood estimation, *Computational Statistics & Data Analysis*, 52 (12), 5055-5065.

Voß, S., Martello, S., Osman, I. H. ve Roucairol, C., 2012, Meta-heuristics: advances and trends in local search paradigms for optimization, *New York*, Springer Science & Business Media, p. 297-392.

Wang, F.-K. ve Huang, P.-R., 2014, Implementing particle swarm optimization algorithm to estimate the mixture of two Weibull parameters with censored data, *Journal of Statistical Computation Simulation*, 84 (9), 1975-1989.

Weibull, W., 1939, A statistical theory of strength of materials, *Stockholm*, Generalstabens litografiska anstalts förlag, p. 1-45.

Wolpert, D. H. ve Macready, W. G., 1997, No free lunch theorems for optimization, *IEEE Transactions on evolutionary computation*, 1 (1), 67-82.

Yalçınkaya, A., 2018, Tam ve sansürlü örneklem durumlarında bazı normal olmayan dağılımların parametrelerinin genetik algoritma yardımıyla tahmini, *Ankara Üniversitesi: Fen Bilimleri Enstitüsü: İstatistik Anabilim Dalı*, Ankara.

Yalçınkaya, A., Şenoğlu, B. ve Yolcu, U., 2018, Maximum likelihood estimation for the parameters of skew normal distribution using genetic algorithm, *Swarm and Evolutionary Computation*, 38, 127-138.

Yang, F., Ren, H. ve Hu, Z., 2019, Maximum Likelihood Estimation for Three-Parameter Weibull Distribution Using Evolutionary Strategy, *Mathematical Problems in Engineering*, 2019.

Yang, X.-S., 2010, Engineering optimization: an introduction with metaheuristic applications, *Cambridge*, John Wiley & Sons, p. 173-233.

Yang, X.-S. ve Deb, S., 2013, Multiobjective cuckoo search for design optimization, *Computers & Operations Research*, 40 (6), 1616-1624.

Yang, X.-S., 2014, Nature-inspired optimization algorithms, Elsevier, p. 1-111.

Yonar, A. ve Pehlivan, N. Y., 2020a, Artificial Bee Colony with Levy Flights for Parameter Estimation of 3-p Weibull Distribution, *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, 44, 851-864.

Yonar, A. ve Pehlivan, N. Y., 2020b, A novel differential evolution algorithm approach for estimating the parameters of Gamma distribution: An application to the failure stresses of single carbon fibres, *Hacettepe Journal of Mathematics and Statistics* (Basımda).

Zhang, J. ve Lon Ng, W., 2010, EML-estimation of multivariate t copulas with heuristic optimization, *Proceedings of International Conference on Computer Science and Information Technology*, 469-473.

Zoraghi, N., Abbasi, B., Niaki, S. T. A. ve Abdi, M., 2012, Estimating the four parameters of the Burr III distribution using a hybrid method of variable neighborhood search and iterated local search algorithms, *Applied Mathematics and Computation*, 218 (19), 9664-9675.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Aynur YONAR
Uyruğu : T.C
Doğum Yeri ve Tarihi : Kaman 08/07/1989
Telefon : 03322233986
Faks : -
e-mail : aynursahin@selcuk.edu.tr

EĞİTİM

Derece		Adı, İlçe, İl	Bitirme Yılı
Lise	:	Pursaklar Lisesi, Pursaklar, Ankara	2006
Üniversite	:	Ankara Üniversitesi, Tandoğan, Ankara	2012
Yüksek Lisans	:	Selçuk Üniversitesi, Selçuklu, Konya	2016
Doktora	:	Selçuk Üniversitesi, Selçuklu, Konya	-

İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
2014-	Selçuk Üniversitesi Fen Fakültesi İstatistik Bölümü	Araştırma Görevlisi

UZMANLIK ALANI

Yöneylem Araştırması, Çok Kriterli Karar Verme, Metasezgisel yöntemler, Optimizasyon Teknikleri, Bulanık Mantık, Uygulamalı İstatistik

YABANCI DİLLER

İngilizce

YAYINLAR

Ulusal ve Uluslararası Hakemli Dergilerde Yayınlanan Makaleler

1. Yonar A., Yapıcı Pehlivan N., 2020, A novel differential evolution algorithm approach for estimating the parameters of Gamma distribution: An application to the failure stresses of single carbon fibres, Hacettepe Journal of Mathematics and Statistics, (Basımda) [Doktora Tezinden üretilmiştir.]

2. Yonar A., Yapıcı Pehlivan N., 2020, Artificial Bee Colony with Levy Flights for Parameter Estimation of 3-p Weibull Distribution. Iranian Journal of Science and -

Technology, Transaction A, Science (ISTT), 44, 851-864. [Doktora Tezinden üretilmiştir.]

3. Şahin A., Yapıcı Pehlivan N., 2018, A Novel Multi-Criteria Decision-Making Method based on The Ranking Values of Interval Type-2 Fuzzy Sets: An Application of a Manager Selection for a Telecommunication Company. Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 22, 290.

4. Yapıcı Pehlivan N., **Şahin A.**, 2018, An Integrated approach for fuzzy logistic regression. İstatistikçiler Dergisi: İstatistik ve Aktüerya, (11), 42-54.

5. Yapıcı Pehlivan N., **Şahin A.**, Zavadskas Edmundas K., Turskis Z. A, 2018, Comparative Study Of Integrated FMCDM Methods For Evaluation Of Organizational Strategy Development. Journal of Business Economics and Management, 19(2), 360-381.

6. Şahin A., Yapıcı Pehlivan N., 2017, Evaluation of life quality by integrated method of AHP and TOPSIS based on interval type-2 fuzzy sets. Hacettepe Journal of Mathematics and Statistics, 46(2), 1-13. [Yüksek Lisans Tezinden üretilmiştir.]

Uluslararası Sempozyumlarda Sunulan Bildiriler

1. Şahin A., Yapıcı Pehlivan N., 2019, Investigating the Effectiveness of Metaheuristic Approaches in Parameter Estimation via TOPSIS Method, 11th International Statistics Congress, Muğla, Türkiye. [Doktora Tezinden üretilmiştir.]

2. Yıldız B., Yapıcı Pehlivan N., **Şahin A.**, 2019, Application of Metaheuristic Methods for Portfolio Optimization Problems. International Conference on Mathematics and Mathematics Education, Konya, Türkiye.

3. Şahin A., Yapıcı Pehlivan N., 2019, Comparison of Differential Evolution And Genetic Algorithms for Estimating the Parameters of 3-p Gamma Distribution. 2nd International Congress on Statistics Mathematics and Analytical Methods, İstanbul, Türkiye. [Doktora Tezinden üretilmiştir.]

4. Yapıcı Pehlivan N., **Şahin A.**, 2018, Particle Swarm Optimization for Parameter Estimation of the Generalized Gamma Distribution. OR 2018: International Conference on Operations Research, 56-57, Brüksel, Belçika.

5. Şahin A., Yapıcı Pehlivan N., 2018, Performance Evaluation of Metaheuristic Methods In Maximum Likelihood Estimation for Distribution Parameters. 4th International Researchers, Statisticians and Young Statisticians Congress, 131-131, İzmir, Türkiye. [Doktora Tezinden üretilmiştir.]

6. Şahin A., Yapıcı Pehlivan N., 2017, Parameter Estimation of Three parameter Gamma Distribution using Particle Swarm Optimization. 10th International Statistics Congress, 186-186, Ankara, Türkiye. [Doktora Tezinden üretilmiştir.]

7. Yapıcı Pehlivan N., **Şahin A.**, 2017, Stochastic Multi Criteria Decision Making Methods for Supplier Selection in Green Supply Chain Management. 10th International Statistics Congress, 184-184, Ankara, Türkiye.

8. Şahin A., Yapıcı Pehlivan N., 2017, Comparative Analysis of Multi-Criteria Decision Making Methods: A Case Study of the Countries' Environmental Performance Index. 3rd International Researchers, Statisticians and Young Statisticians Congress, 287- 287, Konya, Türkiye.

9. Şahin A., Yapıcı Pehlivan N., 2017, Evaluation of Performance of Simulated Annealing (SA) Algorithm for Multidimensional Knapsack Problems. 3rd International Researchers, Statisticians and Young Statisticians Congress, 286-286, Konya, Türkiye.

10. Şahin A., Yapıcı Pehlivan N., 2016, Determination of Well Being Index of Provinces in Turkey Using Multi Criteria Decision Making Methods. 2nd International Conference on Science, Ecology and Technology, Barselona, İspanya.

11. Yapıcı Pehlivan N., **Şahin A.**, 2016, Multi Criteria Decision Making Methods for Ranking the Countries with respect to the Logistic Performance Index. International Conference on Operations Research, Hamburg, Almanya.

12. Yapıcı Pehlivan N., **Şahin A.**, 2016, Fuzzy Logistic Regression. 2nd International Researchers-Statisticians and Young Statisticians Congress, Ankara, Türkiye.

13. Şahin A., Yapıcı Pehlivan N., 2016, Evaluation of Life Quality by Integrated Method of AHP and TOPSIS Based on Interval Type-2 Fuzzy Sets. 2nd International Researchers- Statisticians and Young Statisticians Congress, Ankara, Türkiye. [Yüksek Lisans Tezinden üretilmiştir.]

14. Şahin A., Yapıcı Pehlivan N., 2015, Multi Criteria Decision Making Method Based on The Ranking Values of Interval Type-2 Fuzzy Sets. The International Conference on Science, Ecology and Technology, Viyena, Avusturya.

15. Saraçoglu B., Koç M., Pekgör A., **Şahin A.**, 2014, Statistical Distributions Identification with Neural Networks Approach. 9th International Statistics Day Symposium, Antalya, Türkiye.