



T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**ELEKTROMANYETİK ALGORİTMANIN KARŞILAŞTIRMALI ANALİZİ VE
GELİŞTİRİLMESİ**

Alkın YURTKURAN

Prof. Dr. Erdal EMEL
(Danışman)

DOKTORA TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

BURSA - 2014

TEZ ONAYI

Danışman : Prof. Dr. Erdal EMEL

- Başkan:** Prof. Dr. Erdal EMEL
Uludağ Üniversitesi. Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı.
- Üye:** Prof. Dr. H. Cenk ÖZMUTLU
Uludağ Üniversitesi. Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı.
- Üye:** Prof. Dr. Necati ARAS
Boğaziçi Üniversitesi. Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı.
- Üye:** Prof. Dr. Nursel ÖZTÜRK
Uludağ Üniversitesi. Mühendislik Fakültesi,
Endüstri Mühendisliği Anabilim Dalı.
- Üye:** Doç. Dr. Deniz AKSEN
Koç Üniversitesi. İktisadi ve İdari Bilimler Fakültesi,
İşletme Anabilim Dalı.

Yukarıdaki sonucu onaylım.

Prof. Dr. Ali Osman DEMİR
Enstitü Müdürü
../12/2014

U.Ü Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

12/12/2014

Alkın YURTKURAN

ÖZET

Doktora Tezi

ELEKTROMANYETİK ALGORİTMANIN KARŞILAŞTIRMALI ANALİZİ VE GELİŞTİRİLMESİ

Alkın YURTKURAN

Uludağ Üniversitesi

Fen Bilimleri Enstitüsü

Endüstri Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Erdal EMEL

Günümüzün rekabetçi koşulları, karşılaşılan problemlerin çözümünde kabul edilebilir süreler içerisinde uygun çözümlere erişmeyi, kaçınılmaz kılmaktadır. Her geçen gün artan oranlarda zorluk derecesine sahip bir çok gerçek hayat problemi bilgisayar ortamında modellenerek, bir optimizasyon problemine dönüştürülmekte ve uygun araçlarla çözülmeye çalışılmaktadır. Ancak, bu tip problemlerin büyük boyutlu ve karmaşık oluşuna karşın kısa sürede optimal çözüm elde etme gerekliliği, kesin çözüm araçlarının yetersiz kalmasına sebep olmaktadır. Bu sebeple, zorluk derecesi yüksek problemlerin çözümünde zeki arama yöntemleri olarak tanımlanabilen meta sezgisel yöntemler, uygun bir çözüm aracı olarak son yıllarda araştırmacıların yoğun ilgisini çekmektedir. Söz konusu tez çalışması kapsamında, son yıllarda oldukça popülerleşen elektromanyetik algoritma (EMA) ele alınmış ve algoritma ile ilgili iyileştirme ve uygulama çalışmaları gerçekleştirilmiştir. Tez çalışmasında, ilk olarak EMA'nın performansının iyileştirilmesi ve geliştirilmesi hedeflenmiş ve bu kapsamda literatürden seçilen ve çokça kullanılan meta sezgisel yöntemlerin temel sürümleri ile EMA eşit koşullarda ve kısıtsız optimizasyon problemleri üzerinde karşılaştırılarak analiz edilmiştir. Yapılan analizler sonucunda, EMA'nın diğer popüler algoritmalarından farkları ortaya konularak, güçlü ve zayıf yönleri tespit edilmiştir. Ardından, hem diğer popüler yöntemlerin üstünlükleri, hem de EMA'nın güçlü ve zayıf yönlerinden esinlenilerek EMA için iyileştirme önerileri sunulmuştur. Geliştirilen önerilerin EMA performansına etkileri, bir deney tasarımı yaklaşımı ile istatistiksel olarak analiz edilmiştir. Analiz sonucunda, EMA'nın performansına anlamlı bir şekilde etki eden öneriler belirlenmiş ve bu önerilerin temel EMA yapısına uygulanması ile iyileştirilmiş EMA yapısı (iEMA) geliştirilmiştir. iEMA'nın performansı daha önce temel sürümleri kullanılan meta sezgisel yöntemlerin iyileştirilmiş sürümleri ile eşit koşullar altında analiz edilmiş ve iEMA'nın üstünlüğü istatistiksel olarak kanıtlanmıştır. Tez çalışmasının izleyen aşamalarında ise iEMA üzerinde küçük uyarlamalar yapılarak geliştirilen sürümler, farklı tipteki optimizasyon problemlerinin çözümüne uygulanmış ve detaylı analizler sonucunda söz konusu iEMA sürümlerinin kalitesi kanıtlanmıştır. Bu doğrultuda, doğrusal ve doğrusal olmayan kısıtlı optimizasyon problemlerini çözebilen iEMA sürümü (ciEMA), 0-1 tipindeki ikili sistem vektörlerle çalışabilen iEMA sürümü (biEMA) ve birleşimsel tipteki problemleri çözebilen iEMA sürümü (combiEMA) geliştirilmiştir. Söz konusu çalışma kapsamında geliştirilen her sürüm, literatürden

alınan yöntemler ile karşılaştırılarak analiz edilmiş ve farklı uygulama çalışmalarında kullanılmıştır. iEMA ve sürümleri (ciEMA, biEMA, combiEMA) ile yapılan uygulamalar ve testlerin sonuçları incelendiğinde; son yıllarda oldukça popülerleşen bir meta sezgisel yöntem olan EMA'nın, bu tez çalışması kapsamında geliştirildiği ve küçük uyarlamalar ile farklı optimizasyon problemlerine rahatlıkla ve başarılı bir şekilde uygulanabildiği gösterilmiştir.

Anahtar Kelimeler: Elektromanyetik algoritma, meta sezgisel algoritma, optimizasyon.

2014, xiv + 249 sayfa

ABSTRACT

PhD Thesis

COMPARATIVE ANALYSIS AND IMPROVEMENT OF ELECTROMAGNETISM-LIKE ALGORITHM

Alkın YURTKURAN

Uludağ University

Graduate School of Natural and Applied Sciences

Department of Industrial Engineering

Supervisor: Prof. Dr. Erdal EMEL

In today's competitive world, solving problems in acceptable computational times is a must to reach feasible solutions. Complex real world problems are modeled in computer environments, transformed into optimization problems and solved via appropriate techniques. However, exact solution methods are not sufficient to solve these large scale complex problems, given the time constraint in finding optimal solutions. In recent years; meta heuristics, described as intelligent search methods, have been studied by many scientists as a proper tool to solve highly complex optimization problems. In this dissertation, Electromagnetism-like Algorithm (EMA), a very popular algorithm in today's world, has been studied, improved and used in several applications. Within this context, the algorithm was improved and elaborated by analyzing and comparing the widely used traditional meta heuristics approaches in literature with EMA under equal test conditions for solving continuous optimization problems. The results revealed the differences of EMA from other popular algorithms. The strengths and weaknesses of EMA compared to the superior properties of the other algorithms are then used to develop improvement proposals for EMA. A design of experiments study is constructed to observe the effects of the proposed improvements on the performance of EMA through statistical analysis. As a result of this analysis, the proposals with statistically significant effects are identified and integrated into the traditional EMA, which turned out to be the improved EMA (iEMA). An additional comparative study for iEMA with the improved versions of the abovementioned meta heuristics is performed under equal test conditions. The results proved the relatively better performance of iEMA. In the following stages of this thesis study, various versions of iEMA with small adaptations are introduced and used to solve different optimization problems. In this scope, the following iEMA versions are introduced: (i) iEMA to solve constrained optimization problems (ciEMA), (ii) iEMA that is capable of working with binary vectors (biEMA), and (iii) iEMA for combinatorial optimization problems (combiEMA). All three versions are compared to the popular methods from literature and used in different applications to show the superior performance of the proposed versions. Experimental results of iEMA and versions (ciEMA, biEMA, combiEMA) revealed that EMA is successfully improved and can be used for solving different optimization problems by employing small adaptations.

Key words: Electromagnetism-like algorithm, meta heuristic algorithm, optimization.

2014, xiv + 249 pages

TEŞEKKÜR

Doktora eğitimim ve tez çalışmam süresince her zaman yanımda olan, büyük bir özveri ile bana her konuda yol gösteren ve motive eden, değerli fikirleriyle tez çalışmasını yönlendiren, desteğini hiçbir zaman esirgemeyen danışman hocam sayın Prof. Dr. Erdal Emel'e en başta sonsuz teşekkür etmek istiyorum. Değerli zamanlarını ayırarak tez izleme komitesinde yer alarak çalışmalarına yaptıkları katkılardan dolayı sayın hocalarım Prof. Dr. H. Cenk Özmutlu ve Prof. Dr. Necati Aras'a teşekkürlerimi sunarım. Ayrıca, tez savunma sınavımda yer alan Prof. Dr. Nursel Öztürk ve Doç. Dr. Deniz Aksen'e önerileri ve düzeltmeleri için şükranlarımı sunarım.

Dostluklarıyla kendimi şanslı hissettiğim tüm dostlarıma ve bana desteklerini esirgemeyen başta İlker Küçükkoğlu ve Özlem Kanga olmak üzere tüm iş arkadaşlarıma teşekkür ederim.

Bugünlere gelmemin tek sebebi, annem Deniz Merih Yurtkuran ve babam Mustafa Abbas Yurtkuran'a, her zaman yanımda olduğunu bildiğim sevgili ablam Selay Yurtkuran'a teşekkürler.

Hayat yolculuğumda bana eşlik eden, her zaman yanımda olan ve tüm akademik kariyerim boyunca sonsuz desteğiyle beni yalnız bırakmayan, biricik eşim, Merve Yurtkuran'a teşekkür ederim. O olmasaydı, her şey eksik olurdu.

Ailemize katılmasıyla hayatımın anlamını değiştiren, meleğim, kızım Ada Yurtkuran'a sonsuz teşekkürler.

Söz konusu tez çalışması, Uludağ Üniversitesi Rektörlüğü Bilimsel Araştırma Projeleri Komisyonu tarafından KUAP(M)-2012/32 numaralı proje ile desteklenmiştir. Uludağ Üniversitesi Bilimsel Araştırmalar Komisyonuna teşekkürlerimi sunuyorum.

Çalışmalarımın tamamını tüm aileme ithaf ediyorum.

Alkın YURTKURAN

12/12/2014

İÇİNDEKİLER

	Sayfa
ÖZET	i
ABSTRACT	iii
TEŞEKKÜR	iv
İÇİNDEKİLER	v
SİMGELER VE KISALTMALAR DİZİNİ	vii
ŞEKİLLER DİZİNİ	xi
ÇİZELGELER DİZİNİ	xii
ALGORİTMALAR DİZİNİ	xiv
1. GİRİŞ	1
1.1. Motivasyon	4
1.2. Tez Çalışmasının Amacı	16
1.3. Tez Çalışmasının Önemi ve Literatüre Katkısı	16
1.4. Tez Çalışmasının Yöntemi	16
2. ELEKTROMANYETİK ALGORİTMA	19
2.1. Kaynak Özetleri	19
2.2. Elektromanyetik Algoritma	26
2.2.1. Yük ve kuvvet hesaplamaları	27
2.2.2. Parçacıkların hareket ettirilmesi	29
2.2.3. Elektromanyetik algoritmanın adımları	30
3. ELEKTROMANYETİK ALGORİTMANIN PERFORMANS ANALİZİ	32
3.1. Karşılaştırma Algoritmaları	32
3.1.1. Yapay arı kolonisi algoritması	32
3.1.2. Parçacık sürüsü optimizasyonu	35
3.1.3. Diferansiyel gelişim algoritması	37
3.2. EMA, ABC, DGA ve PSO'nun kavramsal karşılaştırılması	40
3.3. Test Problemleri	43
3.4. Karşılaştırma Sonuçları	46
3.5. Test Sonuçlarının Analizi	57
4. ELEKTROMANYETİK ALGORİTMA İÇİN İYİLEŞTİRME ÖNERİLERİ	58
4.1. İyileştirme Önerileri	58
4.2. Deneysel Tasarım	64
4.3. Deneysel Tasarım Sonuçları	72
4.4. İyileştirilmiş Elektromanyetik Algoritma	79
5. İYİLEŞTİRİLMİŞ ELEKTROMANYETİK ALGORİTMA İLE KISITSIZ PROBLEMLER ÜZERİNDE TESTLER	82
5.1. Karşılaştırma Algoritmaları	82
5.1.1. İyileştirilmiş yapay arı kolonisi algoritması	82
5.1.2. Global en iyi yapay arı kolonisi algoritması	84
5.1.3. Adaptif parçacık sürüsü optimizasyonu	85
5.1.4. Adaptif diferansiyel gelişim algoritması	88
5.2. Test Ortamı	89
5.3. Karşılaştırma Sonuçları	91
6. İYİLEŞTİRİLMİŞ ELEKTROMANYETİK ALGORİTMANIN KISITLI OPTİMİZASYON PROBLEMLERİNE UYARLANMASI	100
6.1. Kısıtlı Problemler için Uyarlamalar	101

6.2.	Kısıtlı Problemler için İyileştirilmiş Elektromanyetik Algoritma	103
6.3.	Karşılaştırma Algoritmaları	105
6.4.	Kısıtlı Optimizasyon Problemleri	106
6.5.	Testlerin Sonuçları	108
7.	İYİLEŞTİRİLMİŞ ELEKTROMANYETİK ALGORİTMA İLE UYGULAMALAR	113
7.1.	Öznitelik Seçimi Problemi.....	113
7.2.	Kısıtlı mühendislik tasarım problemleri.....	125
7.2.1.	Kaynak yapılmış kiriş problemi.....	126
7.2.2.	Yay tasarım problemi	128
7.2.3.	Hız indirgeyici tasarım problemi	131
7.3.	Tek makinalı çizelgeleme problemi.....	135
8.	SONUÇ VE YORUMLAR	139
	KAYNAKLAR.....	142
	EKLER.....	160
	EK 1 STANDART TEST PROBLEMLERİ.....	161
	EK 2 STANDART TEST PROBLEMLERİ VERİ TABLOLARI	163
	EK 3 KARMA TEST PROBLEMLERİ	165
	EK 4 TEMEL PERFORMANS ANALİZİ SONUÇLARI	174
	EK 5 DENEYSEL TASARIM SONUÇLARI.....	179
	EK 6 DOĞRUSAL KISITLI OPTİMİZASYON PROBLEMLERİ	233
	EK 7 HASTA VERİ SETİ	237
	EK 8 KISITLI MÜHENDİSLİK PROBLEMLERİ SABİT VERİLER	248
	ÖZGEÇMİŞ.....	249

SİMGELER VE KISALTMALAR DİZİNİ

Simgeler	Açıklama
x	Çözüm vektörü
x^*	Optimum çözüm vektörü
$f(x^*)$	Optimum çözüm değeri
D	Problem boyutu
m	Eşitsizlik tipindeki kısıt adedi
l	Eşitlik tipindeki kısıt adedi
g	Eşitsizlik kısıtı
h	Eşitlik kısıtı
\mathbb{F}	Uygun çözüm alanı
\mathbb{S}	Çözüm alanı
\mathbb{R}	Gerçek sayılar kümesi
$MFlops$	Saniyede gerçekleştirilen gezer nokta işlem adedi
L	Alt sınır vektörü
U	Üst sınır vektörü
m	Popülasyon büyüklüğü
$Rand(0,1)$	[0,1] arasında rassal reel sayı
q_i	i . parçacığın elektrik yükü
x_i	i . parçacığın çözümü
x_{best}	Mevcut popülasyondaki en iyi çözüm
F_i	i . parçacık üzerindeki bileşke kuvvet
RNG	Aralık vektörü
λ	[0,1] arasında rassal gerçek sayı
SN	Yiyecek adedi
x_{ij}	i . çözümün j . indis değeri
Φ	[-1,1] arasında rassal gerçek sayı
$prob$	Seçilme olasılığı
fit	Uygunluk değeri
$failure_i$	i . çözüm için başarısız geliştirme denemesi adedi
v_i	i . çözümden oluşturulan yeni komşu çözüm
$x_{i,t}$	i . çözümün t . iterasyondaki değeri
$Pbest_{i,t}$	i . çözümün t . iterasyona kadar eriştiği en iyi çözüm
$Gbest_t$	t . iterasyona kadar bulunmuş en iyi çözüm
c_1	Çözümün geçmişinin etki katsayısı
c_2	Sosyal etki katsayısı
r_1	[0,1] arasında rassal gerçek sayı
r_2	[0,1] arasında rassal gerçek sayı
ω	Atalet momenti
F	Mutasyon faktörü
A	Baz vektör seçim yöntemi
B	Fark vektörü adedi
C	Çaprazlama tipi (DGA algoritması)
CR	Çaprazlama olasılığı

Simgeler	Açıklama
x_n	Kaotik değişken
F_{best}	En iyi parçacığın uyguladığı kuvvet
k	Deneysel tasarım faktör adedi
n	Deneysel tasarım seviye adedi
l	Deney tasarımı çözünürlük değeri
t	t istatistiği değeri
p	p istatistiği değeri
S	Kuvvet hesaplamasında dahil edilen parçacıkların kümesi
V	Kuvvet hesaplamasında dahil edilen parçacıkların kümesi
M	Değiştirilen bileşen adedi
p	Seçim olasılığı
C	Kontrol parametresi (MABC algoritması)
Ψ	[0,C] arasında rassal gerçek sayı
f	Çözümlerin dağınıklık değeri
d_i	i . parçanın ortalama uzaklık değeri
d_g	en iyi parçanın ortalama uzaklık değeri
d_{min}	Popülasyon için minimum ortalama uzaklık değeri
d_{max}	Popülasyon için maksimum ortalama uzaklık değeri
$\omega(f)$	Adaptif atalet moment değeri
ns	Çözümü geliştirme adedi
nf	Çözümü geliştiremememe adedi
$\Phi(x_i)$	i . çözüm için kısıt aşım miktarı
x_{worst}	En kötü çözüm
δ	Uygun alana yakınsama hızı için kontrol parametresi
ε	Hata payı değeri
MR	Modifikasyon oranı
β	Uygun çözüm/Uygun olmayan çözüm alanı oranı
SS	Standart sapma
K	Optimum öznitelik adedi
N	Toplam öznitelik adedi
dönüşüm	Öznitelik seçim sınırı
τ	Kiriş kesme gerilimi
σ	Kiriş bükülme gerilimi
P_c	Burkulma gerilimi
δ	Kiriş sehimi
N	Aktif sarım adedi (Yay problemi)
d	Çap değeri
D	Ana sarım çap değeri
J_j	j . sıradaki iş
E_j	j . iş için erken bitme zamanı
T_j	j . iş için geç kalma zamanı
D_j	j . iş için işlem süresi
C_j	j . işin tamamlanma zamanı
h_j	j . iş için erken bitme penaltı katsayısı
w_j	j . iş için geç kalma penaltı katsayısı

Kısaltmalar	Açıklama
AA	Arı algoritması
ABACO	Gelişmiş ikili arama karınca kolonisi algoritması (Advanced binary ant colony optimization)
ABC	Yapay arı kolonisi algoritması (Artificial bee colony algorithm)
ABOA	Ateş böceği optimizasyon algoritması
ACO _h	Yerel aramalı karınca kolonisi algoritması (Ant colony optimization with local search)
ANOVA	Varyans analizi (Analysis of variance)
APSO	Adaptif parçacık sürüsü optimizasyonu (Adaptive particle swarm optimization)
BA	Yarasa algoritması (Bat algorithm)
bACO	İkili karınca kolonisi algoritması (Binary ACO)
bEMA	Temel ikili sürüm elektromanyetik algoritma
BGA	İkili genetik algoritma (Binary genetic algorithm)
BGSA	İkili yerçekimi arama algoritması (Binary gravitational search algorithm)
biEMA	İkili iyileştirilmiş elektromanyetik algoritma
bin	Binom
BPSO	İkili parçacık sürüsü optimizasyonu (Binary PSO)
CEM	Kısıtlandırılmış elektromanyetik algoritma
CGS	Kombine genetik algoritma (Combined genetic search)
combiEMA	Birleşimsel iyileştirilmiş elektromanyetik algoritma
DEDS	Dinamik stokastik seçimli diferansiyel gelişim algoritması (Differential evolution with dynamic stochastic selection)
DELC	Seviye karşılaştırmalı diferansiyel gelişim algoritması (Differential evaluation algorithm with level comparison)
DEP	Penaltı fonksiyonlu diferansiyel gelişim algoritması (Differential evaluation algorithm with penalty)
DDE	Kısıtlı diferansiyel gelişim algoritması
DGA	Diferansiyel gelişim algoritması
DT	Deneysel tasarım
Eİ	Erişim isabeti
EMA	Elektromanyetik algoritma
ES	Evrimsel strateji
EXP	Ekspansiyon operatörü
FN	Yanlış negatif değer adedi (False negative)
FP	Yanlış pozitif değer adedi (False positive)
GAFAT	Esnek izin teknikli genetik algoritma (Hybrid genetic algorithm with flexible allowance technique)
Gbest	Global en iyi (Global best)
Gbest-ABC	Global en iyi yapay arı kolonisi algoritması
GKA	Guguk kuşu algoritması
HA	Harmonik arama
HEAA	Adaptif melez evrimsel algoritma
HCSGA	Melez guguk kuşu genetik algoritma (Hybrid cuckoo search genetic algorithm)

Kısaltmalar	Açıklama
ciEMA	İyileştirilmiş kısıtlandırılmış elektromanyetik algoritma
iEMA	İyileştirilmiş elektromanyetik algoritma
IPSO	Geliştirilmiş parçacık sürüsü optimizasyonu (Improved PSO)
İYA	İterasyonlu yer değiştirme algoritması
Kİ	Kesin isabet
KKA	Karınca kolonisi algoritması
Lbest	Yerel en iyi (Local best)
LE	Doğrusal eşitlik adedi (Linear equalities)
LCA	Lig şampiyonluk algoritması (League championship algorithm)
LI	Doğrusal eşitsizlik adedi (Linear inequalities)
MABC	Modifiye edilmiş yapay arı kolonisi algoritması (Modified ABC)
MBA	Mayın patlatma algoritması
MCN	Maksimum döngü adedi (Maximum cycle number)
MDE	Modifiye edilmiş diferansiyel gelişim algoritması (Modified differential evaluation algorithm)
MHA	Modifiye edilmiş harmonik arama (Modified HA)
NLIDP	Doğrusal olmayan tamsayı ve kesikli programlama
MR	Modifikasyon oranı (Modification rate)
NE	Doğrusal olmayan eşitlik adedi (Non-linear equalities)
NI	Doğrusal olmayan eşitsizlik adedi (Non-linear inequalities)
NN	En yakın komşu (Nearest neighbor)
ÖAO	Öznitelik azaltım oranı
PESO	Kısıtlandırılmış parçacık sürüsü optimizasyonu
PID	Oransal-integral-türevsel denetleyici
PSO	Parçacık sürüsü optimizasyonu
Rand	Rassal
Regen	Yeniden oluşturma
ConRegen	ciEMA yeniden oluşturma
SaDE	Kendinden adaptif diferansiyel gelişim algoritması (Self adaptive differential evolution algorithm)
SBO	Sosyal davranışlı optimizasyon algoritması
SD	Sınıflandırma doğruluğu
SPSO	Standart parçacık sürüsü optimizasyonu
SS	Standart sapma
TA	Tabu arama algoritması
TB	Tavlama benzetimi
TN	Doğru negatif değerler adedi (True negative)
TP	Doğru pozitif değerler adedi (True pozitive)
UABC	İyileştirilmiş yapay arı algoritması (Unified ABC)
WOS	Web of Science veri tabanı
YAA	Yer çekimi arama algoritması
YBS	Yapay bağışıklık sistemi algoritması

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 1.1. Çalışma kapsamında geliştirilen algoritmalar	18
Şekil 2.1. Yıllara göre EMA ile ilgili yayın adetleri (WOS veri tabanı)	26
Şekil 2.2. EMA için örnek bir parçacık çözüm vektörü	27
Şekil 2.3. EMA'da bileşke kuvvetin yönü.....	28
Şekil 2.4. EMA'nın akış diyagramı.....	31
Şekil 5.1. Bulanık mantık üyelik diyagramları.....	87
Şekil 7.1. Kaynak yapılmış giriş problemi yapısı.....	126
Şekil 7.2. Yay problemi yapısı	129
Şekil 7.3. Hız indirgeyici tasarım problemi yapısı	131

ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 2.1. EMA sürümleri ve yapılan çalışmalar.....	24
Çizelge 2.2. EMA uygulama alanları ve yapılan çalışmalar.....	25
Çizelge 3.1. Algoritmaların kavramsal olarak karşılaştırılması.....	42
Çizelge 3.2. Standart test problemleri.....	45
Çizelge 3.3. Karma test problemleri.....	46
Çizelge 3.4. Standart test problemlerinde istatistiksel anlamlılık.....	51
Çizelge 3.5. Karma test problemlerinde istatistiksel anlamlılık.....	52
Çizelge 3.6. Standart test problemleri için süre ölçümleri (En iyiden sapma oranı).....	54
Çizelge 3.7. Karma test problemleri için süre ölçümleri (En iyiden sapma oranı).....	55
Çizelge 4.1. 2^4 tam faktöriyel tasarım için örnek deneyler.....	66
Çizelge 4.2. 2^{4-1} kısmi faktöriyel tasarım için örnek deneyler (I=ABCD).....	67
Çizelge 4.3. Deneysel tasarım için test problemleri.....	68
Çizelge 4.4 Faktörler ve faktör seviyeleri.....	70
Çizelge 4.5. Uygulanan deneysel tasarımdaki deney noktaları.....	71
Çizelge 4.6. Tüm problem setleri için birinci (L) ve ikinci dereceden (Q) ortalama etki değerleri ve güven aralıkları.....	77
Çizelge 4.7. Problem tipleri ve boyutlarına göre etki değerlerinin değişimi.....	78
Çizelge 4.8. İyileştirilmiş EMA yapısı faktörler ve seviyeleri.....	80
Çizelge 4.9. Önerilen yöntemler ile sağlanan gelişim alanları.....	81
Çizelge 5.1. Çalışmaların atıf adetleri (2014 yılı sonu itibari ile).....	82
Çizelge 5.2. Arama aşamasına göre kontrol parametrelerindeki güncellemeler.....	87
Çizelge 5.3. İyileştirilmiş sürümlerin testleri sonuçları.....	93
Çizelge 5.4. iEMA ve MABC farkı istatistiksel anlamlılık testi sonuçları.....	95
Çizelge 5.5. iEMA ve GBest-ABC farkı istatistiksel anlamlılık testi sonuçları.....	96
Çizelge 5.6. iEMA ve APSO farkı istatistiksel anlamlılık testi sonuçları.....	97
Çizelge 5.7. iEMA ve SaDE farkı istatistiksel anlamlılık testi sonuçları.....	98
Çizelge 5.8. İyileştirilmiş sürümler arasında karşılaştırma sonuçları özeti.....	98
Çizelge 5.9. İyileştirilmiş sürümler arasında bulunan en iyi değer karşılaştırması.....	99
Çizelge 6.1. Karşılaştırma algoritmaları atıf adetleri.....	105
Çizelge 6.2. Kısıtlı test problemleri.....	107
Çizelge 6.3. Kısıtlı test problemleri özellikleri.....	107
Çizelge 6.4. ABC için raporlanan sonuçlar.....	109
Çizelge 6.5. CEM için raporlanan sonuçlar.....	110
Çizelge 6.6. PESO için raporlanan sonuçlar.....	110
Çizelge 6.7. Kısıtlı test problemleri üzerinde ciEMA ile elde edilen sonuçlar.....	111
Çizelge 6.8. En iyi değerlerin karşılaştırması.....	111
Çizelge 6.9. Kısıtlı problemler en iyi değerlerin karşılaştırılması özeti.....	111
Çizelge 6.10. Ortalama değerlerin karşılaştırılması.....	112
Çizelge 6.11. Kısıtlı problemler ortalama değerlerin karşılaştırılması özeti.....	112
Çizelge 7.1. Öznitelik seçim problemi için literatür taraması.....	114
Çizelge 7.2. biEMA için örnek bir dönüşüm ($dönüşüm=0.50$).....	116
Çizelge 7.3. Veri setleri karakteristikleri.....	116
Çizelge 7.4. Veri setleri üzerinde elde edilen sınıflandırma doğrulukları (SD).....	119
Çizelge 7.5. Algoritmalar için KI , EI ve $ÖAO$ değerleri.....	121
Çizelge 7.6. Tasarlanan sistemin öznitelikleri.....	122

Çizelge 7.7. Uygulama çalışması sonuçları ve seçilen öznitelikler.....	124
Çizelge 7.8. Kaynak yapılmış kiriş problemi sonuçları.....	128
Çizelge 7.9. Yay problemi sonuçları	130
Çizelge 7.10. Hız indirgeyici tasarım problemi sonuçları	134
Çizelge 7.11. Rassal anahtar yöntemi.....	137
Çizelge 7.12. Tek makineli çizelgeleme problemi için karşılaştırma sonuçları.....	138

ALGORİTMALAR DİZİNİ

	Sayfa
Algoritma 4.1. Kaotik başlangıç algoritması	60
Algoritma 6.1. Amaç fonksiyon değeri güncellemesi	102
Algoritma 7.1. Öznitelik seçim problemi için dönüşüm.....	115
Algoritma 7.2. Kombinatoryel problemler için AFH prosedürü.....	137

1. GİRİŞ

Günümüzün rekabet ortamı, deęişen her durum karşısında ayakta kalabilmek için her zaman en iyiye erişmeyi zorunlu hale getirmiştir. Gerçek hayatta karşılaşılan problemlerin zorluğu, büyüklüğü ve karmaşıklığı her geçen gün artmakta iken, problemlerin çözümünde bilgisayar ortamında modelleme ve çözüm zorunlu hale gelmektedir. Genellikle kısıtlı kaynakların en uygun kullanımını amaçlayan bu tip problemler, modellenebildiğinde belirli bazı tekniklerin yardımıyla kesin olarak çözülebilmektedir.

Optimizasyon en iyiyi bulmak veya aramak olarak tanımlanabilir (Horst ve Romeijn 2002). Optimizasyon problemlerinin en uygun şekilde çözüme kavuşturulabilmesi için bugüne kadar birçok farklı yöntem geliştirilmiş ve uygulanmıştır. Bu noktada, “en uygun çözüm”, her zaman optimal çözüm anlamına gelmemektedir. Günümüzde, erişilen çözüm değeri kadar çözüme erişilinceye kadar harcanan süre de çok önemli bir ölçüt haline gelmiştir. Gerçek hayatta karşılaşılan çoęu optimizasyon problemi kabul edilebilir süreler içerisinde kesin optimal olarak çözülememektedir. Bu bağlamda, her zaman optimal sonuca erişmeyi hedefleyen kesin optimizasyon yöntemlerinin bu tip problemler üzerinde uygulanabilirliği neredeyse imkansızdır. Bu sebeple, son yıllarda karmaşık optimizasyon problemlerini daha kısa çözüm süreleri içerisinde optimale yakın (yakın optimal) olarak (en uygun şekilde) çözebilen meta sezgisel yöntemler üzerindeki araştırmalar yoğunlaşmış ve literatüre birçok farklı sezgisel ve meta sezgisel algoritma kazandırılmıştır.

Bu bölüm kapsamında, optimizasyon, optimizasyon problemleri ve optimizasyon yöntemlerine ilişkin genel bilgilere motivasyon alt bölümünde yer verilmiştir. Bu alt bölüm kapsamında, optimizasyon problemlerinin yapısı, tipleri incelenmiş ve çözüm yöntemleri özetlenmiştir. Aynı alt bölümde, meta sezgisel yöntemlerin farklı sınıflandırmaları verilmiş ve algoritmalar için kullanılan bazı performans göstergelerinden bahsedilmiştir. Tez çalışmasının amacı, yöntemi ve çalışmanın literatüre katkısı ise devam eden alt bölümlerde açıklanmıştır.

Elektromanyetik Algoritma'nın (EMA) detaylı yapısı, adımları ve sözde kodu Bölüm 2'de verilmiştir. 2. bölümde, önce EMA ile ilgili detaylı bir literatür araştırması verilmiş ve ilgili çalışmalar hem konularına göre, hem de uygulanan problem tiplerine göre sınıflandırılarak literatürde ilk kez EMA için bir gruplama çalışması yapılmıştır. Aynı zamanda çalışmaların yıllara göre dağılımı da bu bölüm kapsamında sunulmuştur.

Üçüncü bölümde, temel EMA ile yapılan karşılaştırma çalışmaları verilmiştir. Bu bölüm kapsamından EMA, parçacık sürüsü optimizasyonu, diferansiyel gelişim algoritması ve yapay arı kolonisi algoritmalarının temel sürümleri ile karşılaştırılmış ve performansı analiz edilmiştir. Yapılan karşılaştırma çalışmalarının tarafsız ve doğru sonuçlar verebilmesi için farklı yapılarda birçok test problemi kullanılmış ve tüm testler eşit şartlar altında gerçekleştirilmiştir. Başka bir deyişle, EMA ile birlikte diğer algoritmaların da temel sürümleri kodlanmış ve aynı donanım üzerinde çalıştırılmıştır. Bu bölümde, elde edilen sonuçlar ile EMA ve diğer algoritmalar arasındaki temel farklar ortaya konmuş ve EMA'nın geliştirilmeye açık yönleri ve geliştirme önerileri belirlenmiştir.

Bölüm 4 kapsamında, 3. bölümde elde edilen sonuçlardan yararlanarak geliştirilen öneriler ve bu önerilerin farklı uyarlamaları açıklanmış ve önerilerin etkilerinin istatistiksel olarak incelendiği deneysel tasarım çalışmasına yer verilmiştir. Deneysel tasarım kapsamında, EMA için öneriler birer faktör ve her öneri için geliştirilen farklı uyarlamalar ise faktör seviyeleri olarak ele alınmıştır. Söz konusu bölüm kapsamında detaylı bir analiz yapılarak, ortaya konan önerilerin EMA'nın performansına etkileri incelenmiştir. Analizler sonucunda, EMA için en verimli yapıya karar verilmiş ve iyileştirilmiş EMA (iEMA) yapısı ortaya çıkarılmıştır.

iEMA ile gerçekleştirilen karşılaştırmalar 5. bölüm kapsamında verilmiştir. Bu bölümün temel amacı, daha önce karşılaştırmalarda kullanılan algoritmaların literatürde en çok atf almış olan iyileştirilmiş sürümleri ile iEMA'yı karşılaştırarak, EMA üzerinde yapılan iyileştirmenin etkisini analiz edebilmektir. Söz konusu bölümde, 3. Bölümüne benzer şekilde tüm testler eşit şartlar altında yapılmış ve analizler için geniş bir test problemi kümesi kullanılmıştır.

Tez çalışmasının 6. bölümünde ise, geliştirilen iEMA, üzerinde temel yapıya bağlı kalınarak yapılan küçük uyarlamalar ile kısıtlı optimizasyon problemlerini çözebilir hale getirilmiştir (ciEMA). Bu bölüm kapsamında, geniş bir test seti kullanılarak ciEMA'nın performansı analiz edilmiştir. Çalışmalarda kullanılan karşılaştırma algoritmaları, literatürde sıklıkla kullanılan ve başarısı kanıtlanmış olanlardan seçilmiştir. Sonuç olarak, iEMA yapısı üzerinde yapılan küçük uyarlamalar sonucunda ortaya konan ciEMA'nın, kısıtlı optimizasyon problemlerinin çözümü için verimli ve etkin bir algoritma olduğu gösterilmiştir.

Bölüm 7'de ise iEMA ile farklı tiplerdeki optimizasyon problemleri üzerinde uygulamalar yapılmıştır. İlk olarak iEMA, gerçek sayı yerine ikili (binary) sayı sisteminde çalışacak şekilde güncellenmiş ve kısaca biEMA olarak adlandırılan bu sürüm gerçek bir öznitelik seçim problemi üzerinde uygulanmıştır. Gerçekleştirilen uygulama kapsamında, Uludağ Üniversitesi Tıp Fakültesi Kalp Damar Cerrahisi Anabilim Dalı ile ortak bir uygulama çalışması yapılmıştır. Bu bağlamda, *karotis arter stenozunun* tedavi yönteminin belirlenmesi için bir karar destek sistemi tasarlanmış ve sistemin özniteliklerinin seçimi için biEMA kullanılmıştır. İkinci olarak, daha önce geliştirilen ciEMA iyi bilinen mühendislik tasarım problemlerinden kaynak yapılmış kırıntı problemi (welded beam problem), yay problemi (tension/compression spring), hız indirgeyici dişli problemi (speed reducer problem) üzerinde çalıştırılmış ve literatürden alınan yöntemler ile detaylı olarak karşılaştırılmıştır. Son olarak, iEMA'ya yeni bir amaç fonksiyonu hesaplama prosedürü katılarak algoritmanın birleşimsel (combinatorial) problemlere de uygulanabilmesi sağlanmış ve combiEMA olarak adlandırılan bu sürüm tek makineli erken bitme-geç kalma çizelgeleme problemi üzerinde sınanmıştır. Elde edilen sonuçlar kullanılarak, combiEMA yönteminin başarılı bir şekilde birleşimsel problemlere uygulanabildiği gösterilmiştir.

Tezin son bölümünde ise yapılan çalışmalar özetlenmiş ve tezin genel sonuçları yorumlanmıştır. Ek olarak bu bölümde, gelecekte yapılması planlanan çalışmalara değinilmiş ve tez çalışmasının sonuçlarının literatüre katkısı özetlenmiştir.

1.1. Motivasyon

Optimizasyon problemleri, bir problemin çözümünde elde edilebilecek en iyi değerlere erişmek için problem değişkenlerinin alabileceği en uygun değerleri araştırırlar. Genel olarak bir optimizasyon problemi aşağıdaki gibi amaç fonksiyonu ve kısıtlar şeklinde modellenebilir (Pintér 1996):

$$z = \min f_1(x), f_2(x), f_3(x), \dots, f_L(x) \quad (1.1)$$

$$g_k(x) \leq 0, \quad k = 1, \dots, m \quad (1.2)$$

$$h_p(x) = 0, \quad p = 1, \dots, l \quad (1.3)$$

Söz konusu modelde, \mathbf{x} çözüm vektörünü gösterir ve $\mathbf{x} = (x_1, x_2, \dots, x_D)^T$ olarak ifade edilir. D problemin sahip olduğu değişken adedini veya başka bir deyişle problem boyutunu göstermektedir. $\mathbf{x} \in \mathbb{S}$ ve \mathbb{S} ise problemde tanımlanan tüm çözüm alanını göstermektedir. Denklem 1.2 ve Denklem 1.3'te kullanılan m ve l ise sırası ile eşitsizlik ve eşitlik kısıtlarının adedini göstermektedir. Söz konusu kısıtlar ile tanımlanan geçerli çözüm alanı \mathbb{F} ($\mathbb{F} \subseteq \mathbb{S}$) iken bu çözüm alanının içerisinde bulunan \mathbf{x} ise herhangi bir uygun çözüm olarak tanımlanır. Herhangi bir \mathbf{x} uygun çözümü, minimizasyon problemlerinde $f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in \mathbb{F}$ veya maksimizasyon problemlerinde ise $f(\mathbf{x}^*) \geq f(\mathbf{x}), \forall \mathbf{x} \in \mathbb{F}$ eşitsizliklerini sağlıyor ise \mathbf{x}^* olarak ifade edilen çözüm optimum çözümü gösterir (Horst 2000).

Optimizasyon problemlerinde yukarıdaki formülasyonda da görüldüğü gibi bazı temel bileşenler vardır. Söz konusu temel bileşenler; değişkenler, amaç fonksiyonu ve kısıtlar olarak gruplandırılabilir. Bu noktada, bileşenlerin açıklanmasından önce, yerel optimum ile global optimum kavramlarından bahsetmek gerekmektedir. Uygun çözüm uzayında herhangi bir çözüm \mathbf{x} ve $f(\mathbf{x}) = f^*$ ise, \mathbf{x} 'in yerel optimum olarak tanımlanabilmesi için; minimizasyon problemlerinde herhangi bir $\delta > 0$ gerçek değeri için $f^* \leq f(\mathbf{x}^l)$, $\mathbf{x} \in F: |\mathbf{x} - \mathbf{x}^l| \leq \delta$ şartını sağlaması gerekmektedir (Horst ve Romeijn 2002). Başka

bir deyişle, x , çevresinde tanımlanan komşularından daha iyi bir çözüm ise yerel optimum olarak tanımlanmaktadır. Bu bağlamda, yerel optimum tamamen tanımlanan komşuluk kriterlerine bağlıdır. Diğer taraftan global optimum, uygun çözüm alanındaki tüm noktalardan daha iyi olan çözüm noktası olarak tanımlanmaktadır. Herhangi bir optimizasyon probleminde, tanımlanan arama uzayı içerisinde bir global optimuma (minimum veya maksimum) sahip olup olmadığı, Weierstrass teoremi ile anlaşılabilir (Akay 2009). Weierstrass teoremine göre, gerçek sayı olarak kodlanmış bir f fonksiyonu, kapalı ve sınırlı olarak tanımlanmış herhangi bir $[a, b]$ aralığında sürekli ise, f fonksiyonunun $[a, b]$ aralığında bir global minimumu ve/veya global maksimumu mevcuttur. Başka bir deyişle, $[a, b]$ kapalı aralığında, $f(c) \geq f(x) \geq f(d), \forall x \in [a, b]$ şartını sağlayan c ve d sayıları bulunur (Akay 2009). Herhangi bir fonksiyonun minimum veya maksimumu bulunurken fonksiyonun gradyan vektörü ve hessian matrisleri önemli rol oynamaktadır. f fonksiyonunun gradyan vektörü ($\nabla f(x)$) fonksiyonun maksimum değişim yönünü verirken (Haupt 1995), simetrik hessian matrisinin ($\nabla^2 f(x)$) öz değerleri optimallik yeterlilik şartlarında önemli rol oynamaktadır (Tahk ve ark. 2009).

Daha önce de belirtildiği gibi, bir optimizasyon problemi temel olarak üç bileşenden oluşmaktadır: (i) amaç fonksiyonu, (ii) değişkenler ve (iii) kısıtlar. Aşağıda söz konusu üç temel bileşen kısaca açıklanmıştır (Blum ve Roli 2003).

- Amaç fonksiyonu: Bir optimizasyon probleminin amacı en uygun amaç fonksiyonu değerini veren çözümü bulmaktır. Hangi çözümlerin kaliteli, hangi çözümlerin kalitesiz olduğu bir uygunluk fonksiyonu tanımlanarak anlaşılabilir. Denklem 1.1'de gösterildiği gibi, bir optimizasyon problemi L değişkenine bağlı olarak tek amaçlı ve çok amaçlı olabilir ($L = 1$ ise tek amaçlı, $L > 2$ ise çok amaçlı). Çok amaçlı problemlerin çözümü için verimli ve kabul görmüş sabit bir yöntem olmasa da, amaç fonksiyonları ağırlıklandırılarak tek bir amaç fonksiyonuna dönüştürülmesi literatürde en çok benimsenen yöntemlerden biridir (Horst ve Romeijn 2002). Amaç fonksiyonu kullanılarak elde edilen çözüm değerleri, çözümün kalitesini gösterir ve aramanın hangi yönde ilerleyeceğine rehberlik eder. Optimizasyon problemleri genelde

minimizasyon problemlerini içerse de, herhangi bir maksimizasyon problemi $f'(x) = -f(x)$ dönüşümü uygulanarak bir minimizasyona problemine çevrilebilir.

- Değişkenler: Bir sistemin modellenmesinde kullanılan parametrelere değişken denir. Değişkenlerin sistem davranışındaki belirleyici rolü, problemin doğru tanımlanarak çözüme ulaştırılması için çok önemlidir. Aynı zamanda problemin en az sayıda değişken ile doğru şekilde tanımlanması problemin çözülebilmesini kolaylaştırır. Optimizasyon problemlerinde kullanılacak değişkenler gerçek sayı, tamsayı veya ikili sayı (0-1) tipinde olabilir. Diğer bir önemli nokta ise, problem için tanımlanan değişkenlerin mümkün olduğunca birbirinden bağımsız olmasıdır (Arora 2004). Birbiriyle korelasyonu yüksek değişkenlerin kullanılması değişken adedini artırarak problemi daha karmaşık hale getirecektir.

- Kısıtlar: Kısıtlar optimizasyon probleminin doğru tanımlanabilmesi için değişkenleri kullanarak veya değişken üzerinde yapılması gereken sınırlamaları içerir. Bu sebeple, her kısıt bir veya birden çok değişkene bağlı olarak tanımlanır. Problemin uygun çözüm alanının doğru olarak tanımlanması, problem kısıtlarının doğru olarak tanımlanmasını gerektirir. Kısıtlar sadece değişkenlerin üst ve alt sınırlarını tarif edebileceği gibi, doğrusal veya doğrusal olmayan karmaşık denklemler ile de ifade edilebilir. Bazı durumlarda ise sadece bütünlük kısıtları ($x_i \in \{0,1\}$ veya $x_i \in \{0,1,2, \dots, n\}$ gibi) olarak tanımlanabilirler.

Her optimizasyon problemi, farklı özelliklere sahiptir ve yukarıda açıklanan üç temel bileşen bakımından farklı gruplara ayrılabilirler. Amaç fonksiyonu, değişken ve kısıt tipleri bakımından optimizasyon problemleri aşağıdaki gibi farklı gruplar altında incelenebilir (Bäck ve Schwefel 1993):

- Kısıtsız problemler: Bu tip problemler herhangi bir kısıta sahip değildir ve herhangi bir $x \in \mathbb{R}^n$ değişkeni çözüm olabilir. Gerçek hayatta çok karşılaşılan bir problem tipi değildir. Bu tip problemler sadece değişkenler için sınırları gösteren basit lineer alt ve/veya üst sınır kısıtlarını içerebilirler.

- Doğrusal kısıtlı problemler: Kısıtların tümü lineer denklemlerden oluşur. Literatürde kısıtlı optimizasyon problemleri olarak geçer. Bazı mühendislik ve tasarım problemleri bu sınıfa girebilir.
- Doğrusal olmayan kısıtlı problemler: En az bir kısıtın doğrusal olmayan bir denklem olduğu optimizasyon problemleridir. Gerçek hayatta birçok farklı alanda bu tip problemler ile karşılaşılabilir.
- Birleşimsel problemler: Sınırlı, fakat çok büyük bir tamsayı değişken kümesine sahiptir. Gezgin satıcı problemi, atama problemleri, çizelgeleme problemleri gibi problemler bu sınıfa girer. Birleşimsel problemlerin uygun değişken kümeleri çoğu zaman çok büyüktür, bu sebeple kesin olarak çözümleri çok zordur.
- Doğrusal programlama: Tüm kısıtlar ve aynı zamanda amaç fonksiyonu doğrusal denklemlerden oluşur. Değişkenler ise süreklidir.
- Doğrusal olmayan programlama: Amaç fonksiyonu ve/veya kısıtlar doğrusal olmayan denklemlerden oluşabilir. Değişkenler ise sürekli tiptedir.
- Tamsayılı programlama: Tüm değişkenlerin tamsayı olması gerekmektedir. Kısıtlar ve amaç fonksiyonu ise doğrusal yapıdadır.
- Karışık tamsayılı programlama: Değişkenler hem tamsayı hem de gerçek sayılardan oluşabilir. Kısıtlar ve amaç fonksiyonu ise doğrusal yapıdadır.
- Stokastik programlama: Amaç fonksiyonu, kısıtlar veya değişkenlerden bir veya birkaçı deterministik değildir. Başka bir deyişle, problem içerisinde olasılıksal yapılar mevcuttur.

Yukarıda açıklanan optimizasyon problemlerinin çözümü için literatürde birçok farklı çözüm yöntemi önerilmiştir. Önerilen optimizasyon yöntemleri temel olarak iki ana grup altında toplanabilir: (i) kesin çözüm bulan klasik yöntemler, (ii) yakın optimal çözüm bulan modern sezgisel yöntemler (Vesterstrom 2002). Günümüzde birçok optimizasyon problemi büyük boyutlu olduğundan, kabul edilebilir süreler içerisinde global optimum noktalara erişmek çok zordur (Johnson ve Garey 1979). Bu sebeple, yakın optimal sonuçları kabul edilebilir süreler içerisinde bulabilen modern yöntemler daha sıklıkla tercih edilmektedir.

Klasik teknikler çözümlerin üretiliş şekline göre analitik ve yapısal metotlar olmak üzere iki alt gruba ayrılmaktadır (Dumitrescu ve Stütze 2003). Analitik yöntemler kategorisinde doğrusal programlama, doğrusal olmayan programlama, türeve dayalı metotlar ve bölgesel araştırma metotları bulunmaktadır. Dinamik programlama, dal-sınır, dallandır-kes, dallandır-ödül ver gibi metotlar yapısal yöntemlere örnek olarak gösterilebilir (Nemhauser ve Wolsey 1988). Modern yöntemlerin ise birçok farklı gruplandırma şekli vardır. Bir sonraki bölümlerde bu gruplar detaylı bir şekilde incelenmiştir.

Klasik yöntemler

Klasik yöntemlerin en temel avantajları kesin sonuç verebilmeleridir. Bu sebeple küçük ve görece çözülebilir problemlerin çözümü için kullanıldıklarında tercih edilebilirler. En temel kesin çözüm yöntemi detaylı arama olarak tanımlanabilir. Detaylı aramada olabilecek uygun çözümlerin tümü denenerek en iyi amaç fonksiyonu değerini veren çözüm aranmaktadır. Fakat uygun çözüm kümesinin çok büyük olduğu problemlerde, detaylı arama ile belirli bir süre içerisinde optimal çözüme erişmek neredeyse imkânsızdır. Doğrusal programlama problemlerinin konveks özelliklerinden yararlanarak optimal çözümün her zaman köşe noktalarda olacağı garanti edilmiştir (Puchinger ve Raidl 2005). Bu sebeple, optimum çözümün bulunması için uygun çözüm uzayının tüm elemanlarının denenmesine gerek yoktur. Simpleks yöntemi, dal-sınır, dallandır-kes, dallandır-ödül ver gibi yöntemler doğrusal programlama problemlerinin dışbükeylik (convexity) özelliğinden faydalanmaktadır (Nemhauser ve Wolsey 1988). Bazı zor optimizasyon problemlerinin, lagrange yöntemi (lagrangean relaxation) gibi yöntemler ile doğrusal programlama modellerine dönüştürülerek çözülmesi literatürde çokça kullanılan yöntemlerden birisidir. Genel optimizasyon problemleri için (kısıtsız, doğrusal kısıtlı optimizasyon problemleri, vb.) tek boyutta minimizasyon yapan teknikler (Golden Section, vb.) (Björck 1996), çok boyutta arama yapan ve türeve dayalı olmayan teknikler, birinci türeve dayalı teknikler (Quasi Newton, vb.) (Boyd ve Vandenberghe 2009) veya ikinci türeve dayalı teknikler (Newton, Levenberg Mardquardt, vb.) (Björck 1996) gibi yöntemler kullanılabilir.

Birleşimsel problemler ise tamsayılı programlama, karışık tamsayılı programlama, stokastik programlama, dinamik programlama gibi çözüm tekniklerine uygun modellenerek kesin olarak çözülebilir. Fakat gerçek hayat problemlerinin birçoğu büyük uygun çözüm alanlarına sahip olduklarından, söz konusu kesin çözüm teknikleri ile kabul edilebilir süreler içinde çözüm elde etmek imkânsız olabilir. Bu sebeple, kesin çözüm yöntemlerinin kullanıldığı problem tipi çok azdır. Ek olarak, kesin çözüm yöntemlerinin çoğu zaman sadece çözülecek probleme uyarlanabilir olması nedeniyle, farklı problemlerde kullanılabilmesi için belirli güncellemelere ihtiyaç duymaktadır. Günümüzde, CPLEX, Gurobi veya CoinMP gibi ileri seviye çözücülerin, üstün donanıma sahip iş istasyonları üzerinde kullanılmasıyla belirli büyüklükteki birleşimsel problemler kesin optimal olarak çözülebilmektedir.

Modern yöntemler

Klasik yöntemlerin yukarıda açıklanan zayıflıklarından dolayı araştırmacılar NP-Zor tipteki problemleri daha hızlı şekilde yakın optimal olarak çözebilecek yöntemler üzerine yoğunlaşmışlardır. Son otuz yılda, daha verimli ve etkili olarak arama yapmak üzere yeni bir sezgisel algoritma sınıfı geliştirilmiştir. Bu sınıf temel sezgiselleri birleştirerek ve/veya onların ötesinde daha akıllı algoritmalar geliştirilerek oluşturulmuştur. Bu tip algoritmalar günümüzde meta sezgisel denilmektedir. Her ne kadar sezgisel algoritmalar da modern yöntemler sınıflandırması altında incelenmekte olsa da, bu tez kapsamında meta sezgisel algoritmalar ağırlık verilmiştir.

Meta sezgisel terimi ilk olarak Glover (1986) tarafından kullanılmış ve iki Yunanca kelimenin birleştirilmesinden meydana getirilmiştir. Sezgisel kelimesi (Heuristics, Hueriskein) Yunanca kökenli olup, ‘bulmak’ anlamındadır. Meta ise bir üst seviyenin daha ilerisi anlamında kullanılmaktadır. Meta sezgiseller modern sezgiseller olarak da anılmaktadırlar (Reeves 1999).

Literatürde meta sezgisel algoritmalar için birçok farklı tanım mevcuttur. Bunlardan bazıları aşağıda verilmiştir.

“Meta sezgiseller, mevcut sezgiselleri zekice birleştirip yöneterek ve/veya akıllı bazı bileşenler kullanarak, aramanın gerçekleştirildiği uzayda keşfetme ve sömürmeyi dengeli bir şekilde gerçekleştirerek verimli ve etkin şekilde yakın optimal sonuçlara erişebilen tasarımlardır.” (Osman ve Laporte 1996).

“Meta sezgiseller, yüksek kalitede çözümler elde edebilmek için farklı sezgisel yöntemleri birleştirerek yöneten, gerekli noktalarda güncelleyen ve bu yöntemlerin çerçevesini oluşturan yinelemeli (iterative) bir süreçtir. Meta sezgisel, her yinelemede bir çözümü veya bir çözüm kümesini güncelleyerek ilerlemektedir.” (Voss ve ark. 1999).

“Meta sezgiseller, problemlere özgü sezgisel yapıları yöneten ve verimlerini arttıran yüksek seviyeli arama stratejileridir. Meta sezgisellerin temel hedefi yinelemeli olarak gerçekleştirilen aramanın yerel optimumlara takılmasını engelleyerek, yakın optimal sonuçlara yönelmesini sağlamaktır. Kullanılan zeki yöntemler kötü çözümlere izin vererek, daha önce arama yapılmamış çözüm alanlarının da aranmasını sağlar ve böylece yerel optimum noktalara takılmayı engellemeye çalışır. Meta sezgisellerde global optimuma erişme prosesinde farklı etkilerden/yönlendirmelerden faydalanılabilir. Bu etkilere örnek olarak hafıza etkisi ve/veya tecrübe etkisi gösterilebilir. Birçok meta sezgisel olasılığa bağlı bir karar verme yöntemiyle aramayı yönetir.” (Stützle 1999).

“Meta sezgiseller, sezgisel yöntemlerin farklı problemlere uygulanmasına imkan sağlayan ve gerçekleştiren tekniklerdir. Başka bir deyişle, meta sezgiseller farklı optimizasyon problemlerine uygulanabilen, zekice tasarlanmış, genel algoritmik bir çerçeve olarak tanımlanabilir.” (Alba 2005).

Meta sezgisellerin temel özellikleri aşağıda özetlenmiştir.

- Meta sezgiseller problemlerden bağımsızlardır. Başka bir deyişle, bir meta sezgisel küçük güncellemeler ile farklı optimizasyon problemlerinde de kullanılabilir.
- Meta sezgiseller çözüm arama sürecine rehberlik eden akıllı stratejilerdir.
- Meta sezgisel kullanımında, verimli ve etkin bir şekilde arama uzayını keşfederek, uygun süreler içerisinde yakın optimal sonuçlara ulaşmak temel hedeftir.
- Meta sezgisel algoritmalar çoğu zaman deterministik değildir.
- Yerel optimumlara yakalanmamak için birçok farklı yöntem ile işbirliği yapabilirler.
- Günümüzde geliştirilen meta sezgiseller, hafıza, tecrübe, global en iyi bilgisi, yerel en iyi bilgisi gibi birçok farklı etkiyi kullanarak aramaya rehberlik ederler.

Özetle, meta sezgiseller farklı yöntemleri akıllı bir şekilde birleştirerek çözüm uzayında arama yapan yüksek seviyeli stratejilerdir. Herhangi bir meta sezgiselin başarısını etkileyen en kritik nokta sömürme ve keşfetme arasında sağladığı dengedir. Keşfetme arama uzayındaki farklı bölgeleri arayabilme yetisini, sömürme ise bulanık kaliteli bir sonuç çevresine yakınsamayı tarif etmektedir. Bu terimler ilk olarak tabu arama algoritmasında ve evrimsel algoritmalar ile ilgili çalışmalarda kullanılmıştır. Başka bir deyişle, söz konusu keşfetme/sömürme dengesi, aranmamış bölgelerdeki kaliteli çözümlere hızlı bir şekilde ulaşma ile daha önce aranmış ve aynı zamanda kaliteli çözüm bulunamamış bölgelerde daha fazla vakit kaybetmeme olarak da açıklanabilir (Eiben ve Schippers 1998).

Meta sezgisellerin birbirinden farklı veya ortak birçok yönü olduğu için farklı başlıklar altında gruplanabilirler. Gruplama yöntemlerinin tek doğrusu yoktur, bu sebeple farklı özelliklere göre yapılan gruplamalarda farklı başlıklar ortaya çıkabilir.

Sınıflandırma için kullanılan özelliğe bağlı olarak birden çok şekilde sınıflandırma yapılabilir. Aşağıda literatürde kullanılan bazı sınıflandırma başlıkları verilmiştir (Birattari ve ark. 2001, Blum ve ark. 2011, Blum ve Roli 2003).

- Doğa tabanlı ve doğa tabanlı olmayan meta sezgiseller:

En genel ve sıklıkla kullanılan sınıflandırma başlıklarından birisidir. Bu gruba tüm evrimsel algoritmalar, sürü tabanlı algoritmalar ve/veya fizik ve biyoloji tabanlı algoritmalar dahil edilebilir. Doğa tabanlı yöntemlere; genetik algoritma, arı algoritmaları, karınca kolonisi algoritması, guguk kuşu algoritması, elektromanyetik algoritma gibi doğadan esinlenen algoritmalar gösterilebilir. Doğa tabanlı olmayan meta sezgisellere ise; tabu arama, yinelemeli (iterasyonlu) yerel arama gibi yöntemler örnek olarak gösterilebilir (El-Abd 2012).

- Popülasyon tabanlı ve tek çözüm tabanlı meta sezgiseller:

Meta sezgiselleri başka bir sınıflandırma yöntemi ise popülasyon tabanlı olup olmadıklarıdır. Her ne kadar son yıllarda geliştirilen birçok meta sezgisel popülasyon tabanlı olsa da, tabu arama, tavlama benzetimi, komşuluk arama gibi meta sezgisel algoritmalar tek çözüm tabanlıdır. Popülasyon tabanlı algoritmaların en temel avantajı, birden çok farklı bireydeki bilgiyi kullanarak aramayı yönlendirmesidir. Genetik algoritma, yapay arı kolonisi, elektromanyetik algoritma, parçacık sürüsü optimizasyonu gibi meta sezgiseller popülasyon tabanlı algoritmalara örnek olarak verilebilir.

- Hafızası olan ve hafızası olmayan meta sezgiseller:

Meta sezgisel algoritmalarda kullanılan bir diğer önemli özellik ise hafıza kullanımınıdır. Meta sezgiseller arama hafızası kullanıp kullanmamalarına göre de sınıflandırılabilirler. Hafıza kullanan yöntemlerde; yakın geçmişteki arama hareketlerini ve sonuçlarını hafızasında tutan kısa dönem hafızası veya tüm arama geçmişini tutan uzun dönem hafızası kullanılabilir. Tabu arama, parçacık sürüsü optimizasyonu gibi yöntemler hafızalı, elektromanyetik algoritma, diferansiyel gelişim algoritması ve genetik algoritma gibi algoritmalar ise hafıza kullanmayan metotlardan bazılarıdır.

Özellikle son dönemde geliştirilen çoğu meta sezgiselin temelde doğadan esinlendikleri söylenebilir. Son dönemdeki meta sezgiseller çoğu zaman fizik, biyoloji veya etoloji gibi dallardan esinlenerek geliştirilmişlerdir (Boussaïd ve ark. 2013). Aynı şekilde birçok meta sezgiselin çözülecek probleme ait gradyan veya hessian matris bilgilerini kullanmak yerine, olasılıksal bileşenler kullanarak akıllı yöntemlerle sonuca gittiği görülmektedir. Meta sezgisellere gösterilen ilgi, özellikle son otuz yıl içerisinde, hızla artmaktadır. Meta sezgisel algoritma geçmişinin temel taşları incelendiğinde şu çalışmalardan söz edilebilir: tavlama benzetimi (TB) (Kirkpatrick ve ark. 1983), tabu arama (TA) (Glover 1986), yapay bağışıklık sistemi (YBS) (Farmer ve ark. 1986), genetik algoritma (GA) (Goldberg ve Holland 1988, Koza 1992), karınca kolonisi algoritması (KKA) (Dorigo 1992), arı algoritmaları (AA) (Walker ve ark. 1993), diferansiyel gelişim algoritması (DGA) (Storn ve Price 1997), parçacık sürüsü optimizasyonu (PSO) (Kennedy ve Eberhart 1995), yapay arı kolonisi (ABC) (Karaboga ve Basturk 2007), yerçekimi arama algoritması (YAA) (Rashedi ve ark. 2009), bakteriyel besin arama optimizasyonu (BBAO) (Passino 2002), elektromanyetik algoritma (EMA) (Birbil ve Fang 2003), guguk kuşu algoritması (GKA) (Yang ve Deb 2009), ateş böceği optimizasyon algoritması (ABOA) (Yang 2009).

Geliştirilen herhangi bir meta sezgiselin gerçekten kullanılabilir olduğu çeşitli testler ile şüpheye yer vermeyecek şekilde kanıtlanmalıdır. Meta sezgisellerin birbirlerine karşı üstünlükleri teorik ve deneysel karşılaştırmalar ile incelenebilir. Literatüre yeni bir yöntem sunulduğunda tarafsız olarak etkin yönlerin ortaya konulması ve diğer yöntemlere karşı üstünlüğünün kanıtlanması gerekir. Meta sezgisel algoritmalar ile bulunan çözümün kalitesi kadar, o çözüme ulaşılması için geçen süre de (çözüm süresi) çok önemlidir. Çözüm süresi ile çözüm kalitesi arasında ödünleşme olduğu söylenebilir (Barr ve ark. 1995). Meta sezgisel algoritmaların verimliliğinin test edilmesi için genel deneysel karşılaştırmalar yapılarak, istatistiksel olarak üstünlük veya zayıflıklar ortaya konulur. Karşılaştırma analizlerinin tarafsız olabilmesi, koşulların benzer şartlar altında gerçekleştirilmesi ve benzer test problemlerinin kullanılması gibi birçok önemli etkene bağlıdır.

Meta sezgisel algoritmalar için gerçekleştirilen karşılaştırma çalışmaları, genel olarak kurulan hipotezlerin doğruluğunu istatistiksel olarak sınamak üzere yapılan testler ile yürütülür. Bu aşamada birçok istatistiksel test yöntemi kullanılabilir. ANOVA, ikili-t testleri, t testleri gibi parametrik analizlerin yanında, parametrik olmayan istatistiksel analizlerden de faydalanabilir (Chiarandini ve ark. 2007).

Geliştirilen bir algoritmanın araştırmacılar arasında kabul görmesi için kesin bir standart olmamasına rağmen, yöntemin aşağıdaki ölçütlerden en az birinde bir gelişim ve/veya üstünlük göstermesi beklenir (Akay 2009).

- Hızlılık: Karşılaştırılan diğer yöntemlere göre daha hızlı şekilde benzer kalitedeki çözümlere erişmelidir.
- Doğruluk: Diğer algoritmalara göre daha kaliteli çözümler bulmalıdır.
- Gürbüzlük: Test problemlerinin özellikleri ve algoritma parametrelerindeki değişimlere mümkün olduğunca duyarsız olmalıdır.
- Basit/kolay kullanılabilirlik: Kolay kodlanabilir ve anlaşılır olması hızla yaygınlaşmasını sağlayacaktır.
- Genellenebilirlik: Birçok farklı uygulama alanında kolayca kullanılabilirdir.

Özet olarak, herhangi bir meta sezgisel algoritmanın performansının analizi için deneysel analizler esnasında bulunan çözümlerin kalitesi, çözüm süresi, çözümlerin dağınıklığı (en iyi çözüm ile diğer çözümler arasındaki fark), yöntemin gürbüzlüğü gibi parametreler detaylı olarak incelenmelidir. Bu bağlamda, incelenecek çıktılar üç ana grupta toplanabilir: (i) bulunan çözümlerin kalitesi, (ii) çözüm süresi, (iii) gürbüzlük (Alba ve ark. 2005).

Çözüm kalitesi

Meta sezgisel algoritmalar ile bulunan amaç fonksiyonu değerlerinin optimuma veya bilinen en iyi değerlere yakınlığı algoritmanın başarısı için en önemli parametredir. Aynı test problemleri üzerinde birden çok tekrar ile koşullar yapılmalı ve bulunan amaç fonksiyonu değerleri belirlenen istatistiksel anlamlılık derecelerinde incelenmelidir.

Çözüm süresi

Bir meta sezgisel için en önemli maliyet kalemi yakın optimal sonuçlara erişirken harcanan süredir. Daha önce de bahsedildiği gibi, meta sezgisellere olan gereksinimin sebebi, zorluk derecesi yüksek optimizasyon problemlerinin kesin çözüm yöntemleri ile kabul edilebilir sürelerde çözülememesidir. Bu sebeple, meta sezgiselin performansı incelenirken çözüm kalitesinin yanında çözüm süresi de detaylı olarak analiz edilmelidir. Çözüm süreleri karşılaştırılırken, meta sezgisel algoritmaların kodlama etkinliği ve çalıştırıldıkları bilgisayarlarda işlemci yeteneklerinin özdeş olması varılan karşılaştırma sonuçlarının objektif ve geçerli olması için en önemli ölçüttür. Farklı bilgisayar ortamlarında yapılan testler ile veya farklı algoritmik teknikler kullanılarak kodlanmış algoritmalar ile elde edilen çözüm sürelerinin karşılaştırılmasının güvenilirliği her zaman sorgulanacaktır. Deneysel karşılaştırma çalışmaları gerçekleştirilirken, toplam çözüm süresi, en iyi çözüme ulaşılma süresi gibi farklı parametreler incelenebilir. Koşum süreleri tam olarak özdeş ortamlarda karşılaştırılmıyorsa, literatürde çokça kullanılan bir diğer parametre ise toplam amaç fonksiyonu hesaplama adedidir. Aynı amaç fonksiyonu hesaplama adedi kadar gerçekleştirilen koşumlar ile farklı meta sezgiseller tarafsız olarak karşılaştırılabilir. Farklı ortamlarda gerçekleştirilen koşumlardan elde edilen çözüm sürelerinin karşılaştırılabilmesi için işlemcilerin birim zamanda yaptığı standart işlem adetleri de kullanılabilir. Dongarra (2005) farklı işlemciler ile aynı doğrusal denklemleri çözerek, her işlemci tarafından bir saniyede yapılabilecek işlem adedini (MFlops/s) hesaplamıştır. Bu liste devamlı olarak güncel tutularak araştırmacıların hizmetine sunulmuştur (<http://www.netlib.org/benchmark/performance.ps>, 2014).

Gürbüzlük

Gürbüzlük, geliştirilen bir yöntemin farklı tipteki problemlerde ve farklı başlangıç koşullarına sahip tekrararlarda benzer etkinliği göstermesi olarak tarif edilebilir. Aynı meta sezgisel ile kontrol parametreleri değiştirilmeden farklı problem setleri üzerinde elde edilen sonuçlar, metodun gürbüzlüğü hakkında fikir verebilir (Alba ve ark. 2005).

1.2. Tez Çalışmasının Amacı

Bu tez çalışması, son yıllarda oldukça popüler bir meta sezgisel yöntem olan EMA'nın detaylı analizler sonucunda daha da geliştirilmesini ve istatistiksel olarak başarısı kanıtlanmış yeni bir sürümünün literatüre kazandırılmasını amaçlamaktadır. Ek olarak, iyileştirilen EMA'nın çeşitli uyarlamalar yapılarak farklı optimizasyon problemlerinde başarılı bir şekilde kullanılabildiğinin kanıtlanması amaçlanmaktadır.

1.3. Tez Çalışmasının Önemi ve Literatüre Katkısı

Söz konusu tez çalışmasının literatüre katkıları aşağıdaki gibi açıklanabilir:

- Kısıtsız optimizasyon problemlerinin çözümünde çok başarılı performans gösteren, literatürde çokça atıf almış yöntemlerle karşılaştırıldığında istatistiksel olarak anlamlı bir üstünlük sağlayan iyileştirilmiş bir EMA yapısı (iEMA),
- Küçük uyarlamalar sonucunda kısıtlı optimizasyon problemleri çözümünde kullanılabilen, literatürde daha önce kullanılan yöntemlere kıyasla daha iyi performans gösteren bir iEMA yapısı (ciEMA),
- İkili sayı tabanlı vektörler ile çalışabilen başarılı bir iEMA yapısının geliştirilmesi (biEMA) ve uygulaması,
- Birleşimsel problemler için iEMA'dan esinlenilerek geliştirilen ve başarısı bir çizelgeleme probleminde kanıtlanan combiEMA yöntemidir.

1.4. Tez Çalışmasının Yöntemi

Tez çalışması kapsamında ele alınan EMA temel olarak elektrik yüklü parçacıkların birbirlerine uyguladıkları itme-çekme kuvvetlerini taklit ederek optimale doğru ilerlemeyi hedefleyen bir meta sezgisel algoritmadır. EMA, son yıllarda giderek popülerleşen ve birçok farklı tipteki optimizasyon probleminin çözümünde kullanılmış olan bir yöntemdir. Tez çalışmasının ilk bölümünde, temel EMA'nın geliştirmeye açık yönlerinin belirlenmesi için deneyler yapılmış ve EMA deneyler kapsamında birçok farklı meta sezgisel algoritma ile karşılaştırmalı olarak analiz edilmiştir. Daha sonra, EMA'nın diğer algoritmalara karşı güçlü ve zayıf yönlerinden esinlenilerek, EMA için oluşturulan geliştirme önerileri, bir deney tasarımı çalışması kapsamında incelenmiş ve

geliştirme önerilerinin istatistiksel olarak etkileri araştırılarak iyileştirilmiş EMA (iEMA) yapısı ortaya çıkarılmıştır.

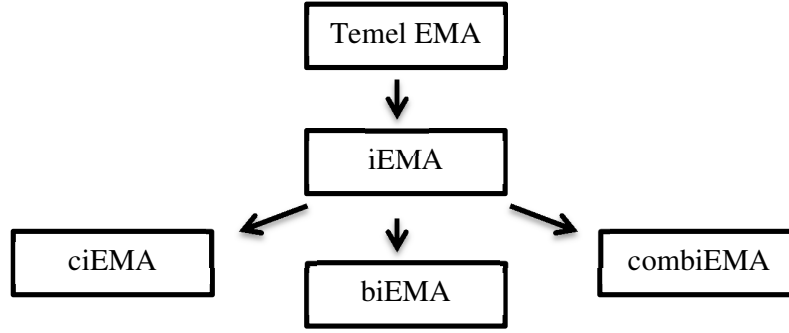
Tez çalışmasının bundan sonraki bölümlerinde ise iEMA üzerinde küçük uyarlamalar gerçekleştirilerek farklı tipteki optimizasyon problemleri üzerinde uygulamalar yapılmış ve detaylı analizler sonucunda söz konusu iEMA sürümlerinin kalitesi kanıtlanmıştır. Bu doğrultuda, ilk olarak doğrusal ve doğrusal olmayan kısıtlı optimizasyon problemlerini çözebilen ciEMA, iEMA kullanılarak geliştirilmiş ve literatürden alınan problemler üzerinden performans testleri gerçekleştirilmiştir. İkinci olarak, iEMA öznitelik seçim problemine uygulanmıştır. Bu kapsamda, 0–1 tipindeki ikili sayı tabanlı vektörlerle çalışabilen biEMA yapısı önerilmiş ve analizler sonucunda biEMA'nın üstünlüğü ortaya konulmuştur. biEMA aynı zamanda bir gerçek hayat uygulaması kapsamında kullanılmıştır. Tezin sonraki bölümünde ise, daha önce kısıtlı problemler için geliştirilen ciEMA kullanılarak farklı mühendislik tasarım problemleri üzerinde koşullar gerçekleştirilmiş ve ciEMA'nın literatürdeki diğer önerilen yöntemlere göre üstünlüğü gösterilmiştir. Son olarak ise, iEMA birleşimsel tipteki problemler ile çalışabilecek şekilde güncellenmiş ve combiEMA geliştirilmiştir. Çalışma kapsamında, combiEMA kullanılarak popüler bir birleşimsel optimizasyon problemi olan tek makineli erken bitme–geç kalma çizelgeleme problemi üzerinde testler gerçekleştirilmiş ve combiEMA'nın en az literatürdeki diğer yöntemler kadar başarılı olduğu sonucuna varılmıştır. iEMA ile ciEMA, biEMA ve combiEMA gibi sürümleri ile yapılan uygulamalar ve sonuçlar incelendiğinde; son yıllarda oldukça popülerleşen bir meta sezgisel yöntem olan EMA'nın tez çalışması kapsamında geliştirildiği ve söz konusu iEMA'nın küçük uyarlamalar ile farklı optimizasyon problemlerine rahatlıkla ve başarılı bir şekilde uygulanabildiği gösterilmiştir.

Tez çalışması yukarıda da açıldığı şekilde beş ana bölümden oluşmaktadır. Bu bölümler;

1. Temel EMA'nın, popüler meta sezgisel yöntemlerin temel sürümleri ile karşılaştırmalı performans analizi, iyileştirme önerilerinin ortaya çıkarılması ve deney tasarımı yöntemi ile iyileştirilmiş sürümünün sunulması (iEMA),
2. iEMA'nın performans analizi,

3. iEMA'dan esinlenilerek, ciEMA'nın geliştirilmesi, performans analizi ve uygulaması,
4. iEMA'dan esinlenilerek, biEMA'nın geliştirilmesi, performans analizi ve uygulaması,
5. iEMA'dan esinlenilerek, combiEMA'nın geliştirilmesi, performans analizi ve geliştirilen sürümün ile yapılan uygulama olarak açıklanabilir.

Tez çalışması kapsamında temel EMA'dan geliştirilen iEMA ve iEMA sürümleri Şekil 1.1'de görsel olarak özetlenmiştir.



Şekil 1.1. Çalışma kapsamında geliştirilen algoritmalar

2. ELEKTROMANYETİK ALGORİTMA

Bu bölüm kapsamında elektromanyetik algoritma (EMA) ile ilgili kaynak özetleri, EMA ile yapılan çalışmaların sınıflandırılması, temel bilgiler, algoritma akışı ve detayları verilmiştir.

2.1. Kaynak Özetleri

EMA ilk olarak Birbil ve Fang (2003) tarafından kısıtsız optimizasyon problemlerinin çözümü için geliştirilmiştir. Geliştirilen yöntem 22 adet kısıtsız, küçük ve orta büyüklükteki test problemine uygulanmış ve diğer temel sezgisel yöntemler ile karşılaştırılmıştır. Söz konusu çalışmada özellikle EMA'nın esnekliği ve uygulama kolaylığı ön plana çıkarılmıştır. Temel EMA mekanizmasının bir yerel arama metodu ile melezlenmiş sürümü, temel yöntem ile karşılaştırılmış ve çözüm kalitesi bakımından üstünlüğü gösterilmiştir. Daha sonra Birbil ve ark. (2004) EMA'nın global optimuma yakınsamasını hızlandırmak için iyileştirilmiş bir EMA yapısı önermişlerdir. Bu yapıda, bir parçacığı etkileyen kuvvetin hesaplanması güncellenmiştir. Algoritmanın kuvvet hesaplama operatöründeki olasılıksal davranışı artırılarak, yerel optimumlara takılma olasılığı düşürülmüştür. Önerilen yeni yapının global optimuma yakınsaması sadece deneysel açıdan değil, söz konusu çalışmada aynı zamanda teorik olarak da ispatlanmıştır. Wu ve ark. (2004) ise, EMA'yı tekstil sektörü için geliştirilmiş bir yapay sinir ağının eğitimi konusunda, en iyi nöron bağlantı ağırlıklarının belirlenmesi için kullanmışlardır. Bu çalışmada EMA mekanizmasının özellikle gerçek sayılar ile çalışmasının avantajından yararlanılmış ve EMA ile eğitilen yapay sinir ağı, hem GA hem de geri yayılım (backpropagation) algoritmaları ile kıyaslanmıştır. Çalışmanın sonuçları incelendiğinde, EMA'nın hem bellek ihtiyacı hem de optimuma yakınsamada diğer algoritmalara karşı üstünlükleri kanıtlanmıştır.

Debels ve Vanhoucke (2006) EMA'yı, dağınık arama sezgiseli ve genetik tabanlı bir yerel arama metodu ile melezleştirerek, kaynak kısıtlı çizelgeleme probleminin çözümünde kullanmışlardır. Bu çalışma kapsamında, uygun çizelgelerin kodlanması için rassal anahtar metodu kullanılmış ve böylece gerçek değerler ile çalışan EMA yapısının birleşimsel problemlerin çözümü için de kullanılabilmesi sağlanmıştır. Ek

olarak, orijinal EMA yöntemine karşılık ikili yük ve kuvvet hesaplama yöntemleri önerilmiştir. Bu çalışma aynı zamanda EMA'nın birleşimsel problemler üzerinde uygulanmasını içeren ilk çalışmadır. Taheri ve ark. (2007) "Arnoldi" algoritması ve EMA'yı melezleştirerek (EMA-RA), geliştirdikleri algoritmayı simetrik olmayan matrisler için öz vektör ve öz değer hesaplamasında kullanmışlardır. Geliştirilen melez algoritma ile temel arnoldi algoritması, simetrik olmayan büyük matrisler üzerinde kıyaslamalı olarak incelenmiş ve geliştirilen melez yapının hem zaman hem de bulunan hata değeri açısından daha üstün olduğu gösterilmiştir.

Maenhout ve Vanhoucke (2007) EMA'yı hemşire çizelgeleme probleminin çözümünde kullanmıştır. Bu çalışmada, uygun olmayan çözümlerin elenmesi veya güncellenmesi için penaltı fonksiyonu ve tamir fonksiyonu geliştirilmiştir. Alikhani ve ark. (2009) EMA'yı "Solis-Wets" arama yöntemi ile melezleştirerek kısıtsız optimizasyon problemlerinin çözümünde kullanmışlardır. Söz konusu çalışmada, Birbil ve ark. (2004) tarafından önerilen iyileştirilmiş EMA yapısı kullanılmış ve bu yapının içerisinde bulunan yerel arama metodunu Solis-Wets (randomized hill climbing) ile değiştirmişlerdir. Temel EMA, geliştirilmiş EMA ve çalışmada önerilen melez EMA yapılarını altı farklı kısıtsız optimizasyon test probleminde incelemiş ve önerdikleri metodun üstünlüğünü göstermişlerdir. Tsou ve Kao (2008) hem maliyet hem de stoksuz kalma miktarını minimize eden, çok amaçlı stok problemlerinin çözümü için EMA tabanlı yeni bir algoritma geliştirmişlerdir. Farklı büyüklüklerde olan amaç fonksiyonları, [0,1] aralığında normalize edilerek kullanılmış ve yük ve kuvvet hesaplamalarında "bulunan en iyiye" olan uzaklık kavramı geliştirilmiştir. Araştırmacılar, geliştirdikleri yeni yük ve kuvvet hesaplamaları ile birden çok amaç fonksiyonu olan problemleri EMA kullanarak çözebildiklerini göstermişlerdir.

Jhang ve Lee (2009) ise elektrik mühendisliği alanında dizi örüntü optimizasyonunda EMA'dan faydalanmışlardır. Rocha ve Fernandes (2009b) yeni bir kuvvet hesaplama denklemi önererek, geliştirdikleri EMA'yı doğrusal kısıtlı optimizasyon problemleri üzerinde test etmişlerdir. Önerilen kuvvet hesaplama yöntemi ile algoritmaya hem hız kazandırılmış, hem de kodlama kolaylığı getirilmiştir. Yazarlar başka bir çalışmalarında ise, EMA'yı yerel arama yöntemi ile melezleyerek farklı mühendislik tasarım

problemlerinin çözümünde kullanmışlardır (Rocha ve Fernandes 2009a). Chang ve ark. (2009) tek makinalı erken bitme–geç kalma çizelgeleme problemi için EMA ve GA'nın melezleştirildiği bir algoritma geliştirmişlerdir. Bu çalışmada EMA mekanizmasından alınan gerçek değerler, rassal anahtar yöntemi kullanılarak uygun bir çizelgeye dönüştürülmüştür. Geliştirilen melez algoritma literatürden alınan standart problemlerde test edilmiş ve GA'ya göre üstünlüğü kanıtlanmıştır. Rocha ve Fernandes (2008) çeşitli uygunluk ve üstünlük kuralları ile EMA'yı doğrusal kısıtlı optimizasyon problemlerinin çözümünde kullanmışlardır. Bu çalışma, EMA'nın doğrusal kısıtlı optimizasyon problemlerinin çözümünde kullanımını içeren ilk çalışmadır. Javadian ve ark. (2008) ise EMA yöntemini gezgin satıcı probleminin çözümünde kullanmışlardır. Yurtkuran ve Emel (2010) kapasiteli araç rotalama probleminin çözümü için melez EMA kullanmışlardır. EMA yapısı yinelemeli yer değiştirme algoritması ile melezlenmiş ve literatürden alınan test problemleri üzerinde diğer temel meta sezgisel algoritmalar (GA, TB, PSO, TA ve KKA) ile kıyaslanmıştır. Sonuçlar incelendiğinde EMA yapısının çözüm kalitesi bakımından en az diğer meta sezgisel algoritmalar kadar iyi olduğu, fakat çözüm süreleri bakımından az da olsa geride kaldığı gözlemlenmiştir. Naji-Azimi ve ark. (2010) küme kaplama probleminin çözümü için EMA'yı kullanmışlardır. Bu çalışmada, temel EMA mekanizmasına ek olarak, aramaya daha uygun çözümler ile başlayabilmek için, başlangıç popülasyonu oluşturma prosedürü eklenmiş ve algoritmanın yerel optimumlara takılmaması için GA benzeri bir mutasyon operatörü kullanılmıştır. Daha önceden tanımlanan eşik değerleri kullanarak gerçek yapıdaki EMA mekanizmasının ikili (0–1) tipindeki değişkenler ile çalışabilmesi sağlanmıştır. Geliştirilen algoritma 80 adet standart test problemi üzerinde test edilmiş ve daha önce geliştirilen Rassal öncelikli arama için meta sezgisel (meta-heuristic for randomised priority search) (META-Raps) algoritması ile kıyaslanmıştır. Sonuçlar incelendiğinde, EMA'nın hem çözüm kalitesi hem de süre bakımından diğer algoritmalara göre daha üstün olduğu gösterilmiştir. Jamili ve ark. (2011) TB ve EMA melez mekanizmasından periyodik iş atölyesi probleminin çözümünde yararlanmışlardır. Bu çalışma kapsamında, gerçek sayılardan oluşan çözüm vektörleri, birleşimsel optimizasyon problemlerinin çözümünde sıklıkla başvurulan rassal anahtar yöntemi kullanılarak uygun birer çizelgeye dönüştürülmüştür. Geliştirilen algoritmada, ilk aşamada TB yapısı kullanarak oluşturulan çözümlerin, daha sonraki aşamada EMA ile iyileştirilerek daha

iyi deęerlere ynelmesi hedeflenmiřtir. Geliřtirilen melez algoritma, temel TA yntemi ve bilinen optimal sonular ile rassal olarak retilen problem tipleri zerinde karřılařtırılmıř ve geliřtirilen melez yapının hem czm kalitesi hem de czm sresi bakımından stnlę gsterilmiřtir. Cuevas ve ark. (2012) Őekil ve resimler zerindeki daireleri tanımlama iřlemi iin EMA'yı kullanmıřlardır. Geliřtirilen algoritma kapsamında, merkez noktası ve yarıapı ile tanımlanan paracıkların, Őekil zerindeki gerek cemberlere uzaklıkları kullanılarak EMA yapısının ierisinde bu uzaklıęın minimize edilmesi ile cemberlerin bulunması saęlanmıřtır. Su ve Lin (2011) veri madencilięi kapsamında znelik seimi probleminin czm iin basit bir komřuluk arama yntemi ile melezleřtirilmiř EMA'yı kullanmıřlardır. alıřma kapsamında, gerek sayılardan oluřan vektrler, tanımlanan limit parametresi ve bileřke kuvvet vektr kullanılarak 0–1 tipindeki deęiřkenlere dnřtrlmř ve hangi deęiřken 1 ise o deęiřkenin gsterdięi znelięin seileceęi varsayılmıřtır. Geliřtirilen algoritma, literatrden bulunan znelik seim algoritmaları ile karřılařtırılmıřtır. Yapılan kıyaslamalar sonucunda, EMA'nın sınıflandırma doęruluęu aısından dięer algoritmalara gre stnlę kanıtlanmıřtır. Kaelo ve Ali (2007) DGA ile EMA'yı melezleyerek, DGA'daki mutasyon operatr yerine, EMA'daki itme–ekme prensibini entegre etmiřlerdir. Yapılan test alıřmalarının sonuları sz konusu yntemin verimlilięini gstermektedir.

EMA son dnemde, izelgeleme problemlerinde cevap sresi deęiřkenlięinin optimizasyonunda (García-Villoria ve Moreno 2010), insansız tařıma aralarının rota tasarımlarında (Guan ve ark. 2011) ve doęrusal olmayan kısıtlardan oluřan optimizasyon problemlerinin czmnde (Ali ve Golalikhani 2010) kullanılmıřtır. Bonyadi ve Li (2012) GA ve EMA'nın melezleřtirilmesiyle geliřtirdikleri algoritmayı ok boyutlu sırt antası probleminin czmnde uygulamıřlardır. Mirabi ve ark. (2010) tedariki seimi probleminin czm iin bir EMA geliřtirmiřlerdir. Godinho ve Branco (2012) belirsizlik altındaki proje izelgeleme problemlerinde EMA'yı kullanmıřlardır. Tm bu alıřmalara ek olarak EMA, oransal-integral-trevsel (PID) denetleyici optimizasyonunda (Lee ve Chang 2010), izelgeleme problemlerinde (Abed ve ark. 2014, Gao ve ark. 2014a, Jiekang ve ark. 2014, Khalili 2014, Khalili ve Tavakkoli-Moghaddam 2012, Naderi ve ark. 2012, Naderi ve ark. 2010, Roshanaei ve

ark. 2010, Yan ve ark. 2014), yapay sinir ağı uygulamalarında (Lee ve ark. 2012, Lee ve ark. 2011), akıllı tahmin sistemlerinde (Wu ve ark. 2014), dinamik optimizasyon problemlerinde (Turky ve Abdullah 2014), üretim hücrelerinin tasarlanması ve tesis yerleşiminde (Guan ve ark. 2012, Javadi ve ark. 2014), kısıtlı optimizasyon problemlerinin çözümünde (Zhang ve ark. 2013), araç rotalama problemlerinde (Mirabi 2014, Yurtkuran ve Emel 2014), maksimum aradalık probleminin çözümünde (Filipović ve ark. 2013), görüntü işleme alanında (Cuevas ve ark. 2013, Jalab ve Abdullah 2013, Oliva ve ark. 2014a, Oliva ve ark. 2014b), üretim sistemleri ve üretim planlamada (Farahnakian ve ark. 2012, Wu ve ark. 2013), doğrusal olmayan sistemlerin kontrolünde (Lee ve Lee 2012), baraj kontrol ünitesi tasarımında (Ouyang ve ark. 2013), elektromanyetik ürünlerin tasarımında (Boucekara 2013), ders programı optimizasyonunda (Abdullah ve ark. 2012) kullanılmıştır.

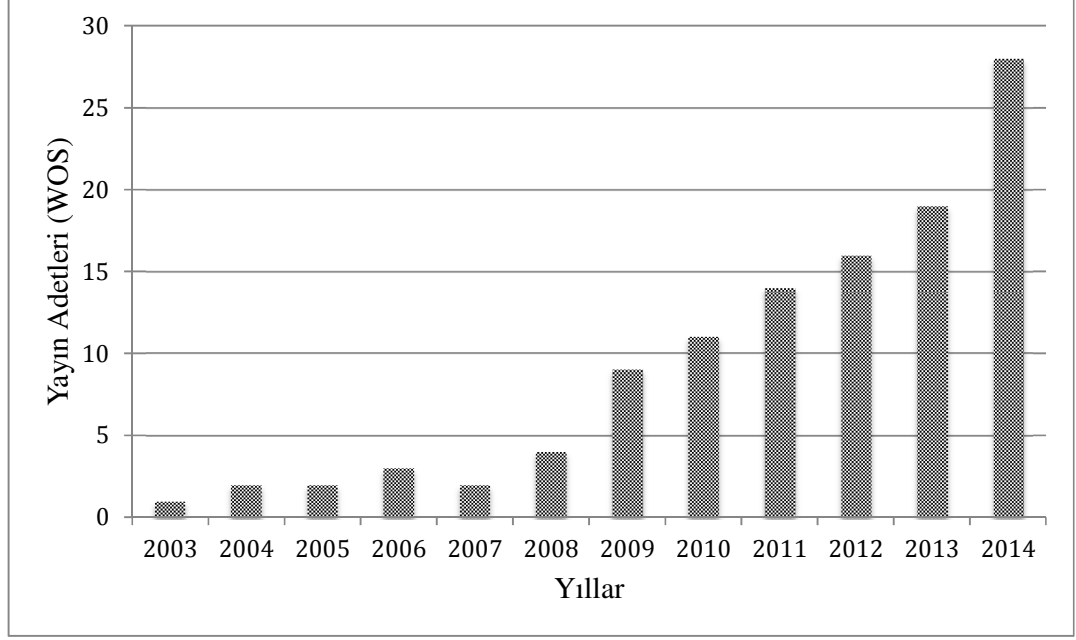
Çizelge 2.1’de, yukarıda kaynak taraması yapılan çalışmalarda kullanılan EMA sürümleri özetlenmiştir. Çizelge 2.1’de de görüldüğü gibi literatürde sürekli, birleşimsel/kesikli, 0–1 tipinde, çok amaçlı, doğrusal/doğrusal olmayan kısıtlı problemler için EMA sürümleri yayınlanmıştır. Yayınlanan çalışmaların araştırma alanlarına göre dağılımı ise Çizelge 2.2’de incelenebilir. Çizelgede de görüleceği gibi EMA bugüne kadar birçok farklı alanda uygulanmıştır. Her iki çizelge incelendiğinde, literatürde kısıtlı problemlerin çözümünde kullanılabilen EMA ve 0–1 tipindeki değişkenlerle çalışılan EMA için daha az çalışma bulunduğundan EMA’nın bu konularda geliştirilmeye açık yönleri olduğu söylenebilir. EMA ile ilgili yapılan çalışma adetlerinin yıllara göre dağılımı Web of Science (WOS) veritabanı incelenerek Şekil 2.1’de verilmiştir. Şekilde de çok açık bir şekilde görüldüğü gibi, EMA’ya olan ilgi her yıl katlanarak artmaktadır. 2014 yılında yapılan 29 adet çalışmanın, önümüzdeki 2–3 yıllık dönem içerisinde yıllık 100 adedin üstüne çıkması beklenmektedir.

Çizelge 2.1. EMA sürümleri ve yapılan çalışmalar

EMA sürümleri	Çalışmalar
Sürekli	Alikhani ve ark. (2009), Birbil ve Fang (2003), Birbil ve ark. (2004), Bouchekara (2013), Cuevas ve ark. (2013), Cuevas ve ark. (2012), Hung (2014), Jalab ve Abdullah (2013), Jhang ve Lee (2009), Kaelo ve Ali (2007), Kratica (2012), Lee ve Chang (2010), Lee ve ark. (2012), Lee ve ark. (2011), Lee ve Jhang (2008), Oliva ve ark. (2014a), Oliva ve ark. (2014b), Rocha ve Fernandes (2009a, 2009b), Taheri ve ark. (2007), Turkey ve Abdullah (2014), Wu ve ark. (2014), Wu ve ark. (2004), Yin ve ark. (2011)
Birleşimsel/Kesikli	Abdullah ve ark. (2012), Abed ve ark. (2014), Bonyadi ve Li (2012), Chang ve ark. (2009), Debels ve Vanhoucke (2006), Farahnakian ve ark. (2012), Filipović ve ark. (2013), Gao ve ark. (2014a), García-Villoria ve Moreno (2010), Godinho ve Branco (2012), Guan ve ark. (2012), Jamili ve ark. (2011), Kartelj ve ark. (2014), Maenhout ve Vanhoucke (2007), Mirabi (2014), Mirabi ve ark. (2010), Naderi ve ark. (2012), Naderi ve ark. (2010), Roshanaei ve ark. (2010), Yan ve ark. (2014), Yurtkuran ve Emel (2010, 2014)
0–1 (İkili değişken)	Javadi ve ark. (2014), Javadian ve ark. (2008), Naji-Azimi ve ark. (2010), Su ve Lin (2011)
Çok amaçlı	Jiekang ve ark. (2014), Khalili (2014), Khalili ve Tavakkoli-Moghaddam (2012), Ouyang ve ark. (2013), Tsou ve ark. (2008), Tsou ve Kao (2008)
Doğrusal kısıtlı	Rocha ve Fernandes (2008), Wu ve ark. (2013), Zhang ve ark. (2013)
Doğrusal olmayan kısıtlı	Ali ve Golalikhani (2010), Lee ve Lee (2012)

Çizelge 2.2. EMA uygulama alanları ve yapılan çalışmalar

Uygulama Alanı	Çalışmalar
Yapay Sinir Ağları	Lee ve ark. (2012), Lee ve Lee (2012), Lee ve ark. (2011), Wu ve ark. (2004)
Endüstri Mühendisliği	Abdullah ve ark. (2012), Abed ve ark. (2014), Bonyadi ve Li (2012), Chang ve ark. (2009), Farahnakian ve ark. (2012), Filipović ve ark. (2013), Gao ve ark. (2014a), García-Villoria ve Moreno (2010), Godinho ve Branco (2012), Guan ve ark. (2012), Jamili ve ark. (2011), Javadi ve ark. (2014), Javadian ve ark. (2008), Khalili (2014), Khalili ve Tavakkoli-Moghaddam (2012), Maenhout ve Vanhoucke (2007), Mirabi (2014), Mirabi ve ark. (2010), Naderi ve ark. (2012), Naderi ve ark. (2010), Naji-Azimi ve ark. (2010), Roshanaei ve ark. (2010), Tsou ve ark. (2008), Tsou ve Kao (2008), Wu ve ark. (2013), Yan ve ark. (2014), Yurtkuran ve Emel (2010, 2014)
Makine Mühendisliği	Guan ve ark. (2011), Rocha ve Fernandes (2009a), Yin ve ark. (2011)
Veri Madenciliği	Su ve Lin (2011)
Elektrik/Elektronik Mühendisliği	Boucekara (2013), Hung (2014), Jhang ve Lee (2009), Jiekang ve ark. (2014), Kratica (2012), Lee ve Chang (2010), Lee ve Jhang (2008)
Görüntü İşleme	Cuevas ve ark. (2012), Jalab ve Abdullah (2013), Oliva ve ark. (2014a), Oliva ve ark. (2014b)
Tıp	Cuevas ve ark. (2013)
İnşaat Mühendisliği	Kartelj ve ark. (2014), Ouyang ve ark. (2013)
Matematik	Taheri ve ark. (2007)
Global Optimizasyon	Ali ve Golalikhani (2010), Alikhani ve ark. (2009), Birbil ve Fang (2003), Birbil ve ark. (2004), Kaelo ve Ali (2007), Rocha ve Fernandes (2008, 2009b), Turkey ve Abdullah (2014), Zhang ve ark. (2013)



Şekil 2.1. Yıllara göre EMA ile ilgili yayın adetleri (WOS veri tabanı)

2.2. Elektromanyetik Algoritma

EMA, kısıtsız optimizasyon problemlerinin çözümü için geliştirilen ve özellikle son yıllarda birleşimsel optimizasyon problemleri üzerindeki uygulamaları hızla artan popülasyon tabanlı yeni sayılabilecek bir meta sezgisel algoritmadır. EMA, elektromanyetik teoriden temel almıştır ve yüklü parçacıklar arasındaki itme-çekme kuvvetlerini taklit eder. EMA sınırlı değişkenlerden oluşan kısıtsız optimizasyon problemlerini hızlı ve etkin bir şekilde çözme yetisine sahiptir. İlk olarak Birbil ve Fang (2003) tarafından önerilmiş olan EMA yapısı, aynı araştırmacılar tarafından algoritmanın yakınsama yetisinin artırılması için revize edilmiştir (Birbil ve ark. 2004).

EMA mekanizmasında, her parçacık bir çözümü gösterir ve parçacığın üzerindeki yük, o parçacığın çözüm kalitesi ile doğru orantılıdır. Başka bir deyişle, parçacığın gösterdiği çözüm ne kadar iyi ise, yükü de o kadar fazla olacaktır. EMA'da elektrik yüklü parçacıklar, fizikte olduğu gibi birbirlerine kuvvet uygularlar ve parçacıklar her seferinde üzerlerindeki bileşke kuvvet boyunca hareket ederler. Temel olarak EMA mekanizması 3 ana bölümden oluşur: (i) Parçacıkların oluşturulması, (ii) Yük ve kuvvet hesaplamaları ve (iii) parçacıkların hareket ettirilmesi.

EMA mekanizması orijinal olarak kısıtsız optimizasyon problemlerinin çözümünde kullanıldığı için, parçacıkların arama uzayındaki pozisyonlarını gösteren pozisyon vektörleri doğrudan amaç fonksiyon değeri hesaplanmasında kullanılmıştır.

Şekil 2.2’de, 5 boyutlu (değişkenli) bir optimizasyon problemi için kullanılacak, parçacık i ’nin örnek bir pozisyon vektörü, $\mathbf{x}_i = [x_i^1, x_i^2, x_i^3, x_i^4, x_i^5]$ bileşenleriyle verilmiştir. Burada i . vektörün k . bileşeni şeklinde verilen x_i^k , gerçekte sayı cinsinden değer alır.

Bileşen indis, k	1	2	3	4	5
k . bileşen değeri	0,23	0,45	1,55	3,42	-4,10
Değişkenin adı	x_i^1	x_i^2	x_i^3	x_i^4	x_i^5

Şekil 2.2. EMA için örnek bir parçacık çözüm vektörü

EMA mekanizmasında x_i^k olarak verilen çözüm parçacıkları başlangıçta Denklem 2.1 kullanılarak rassal bir şekilde oluşturulur.

$$x_i^k = L^k + rand(0,1)(U^k - L^k), \forall i. \quad (2.1)$$

Burada, U^k k . parametrenin üst sınırı, L^k ise k . parametrenin alt sınırını göstermektedir. $rand(0,1)$ ise sıfır ile bir arasında rassal bir sayıyı ifade etmektedir. $i = 1, \dots, m$ ve $k = 1, \dots, D$ ise sırasıyla popülasyon büyüklüğünü (parçacık sayısını) ve problem büyüklüğünü (parametre sayısını) göstermektedir.

2.2.1. Yük ve kuvvet hesaplamaları

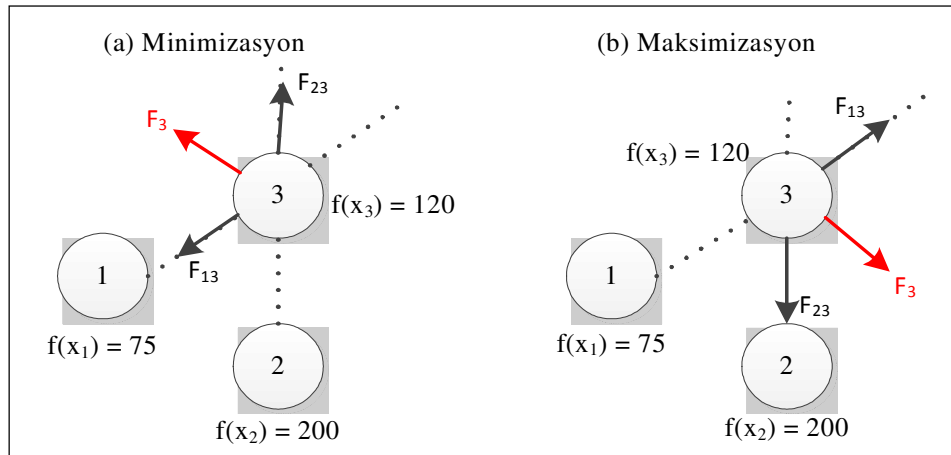
Bir parçacığın taşıdığı elektrik yükü parçacığın gösterdiği çözümün kalitesi kullanılarak hesaplanır. Parçacığın elektrik yükü Denklem 2.2 kullanılarak hesaplanır (Birbil ve Fang 2003).

$$q_i = \exp\left(-D \frac{f(\mathbf{x}_i) - f(\mathbf{x}_{best})}{\sum_{j=1}^m (f(\mathbf{x}_j) - f(\mathbf{x}_{best}))}\right), \forall i. \quad (2.2)$$

Denklem 2.2’de; m popülasyon büyüklüğünü, D problem boyutunu, q_i i . parçacığın yükünü ve $f(\mathbf{x}_i)$, $f(\mathbf{x}_{best})$ ve $f(\mathbf{x}_j)$ ise sırasıyla; i . parçacığın amaç fonksiyonu değerini, popülasyonun en iyi çözüm değerini ve j . parçacığın çözüm değerlerini göstermektedir. Denklem 2.2’de görüldüğü gibi, o anki popülasyonun en iyi parçacığının (\mathbf{x}_{best}) elektrik yükü 1 değerini alacaktır. Ek olarak, tüm popülasyon için q değerleri $[0,1]$ aralığında olacaktır.

Fizikteki teorinin tersine, temel EMA mekanizmasında parçacık yüklerine herhangi bir işaret atanmaz. Parçacıkların birbirlerine itme veya çekme kuvveti uygulayacaklarına, kuvvet hesaplama adımında, parçacık amaç fonksiyon değerleri kullanılarak karar verilir. EMA yaklaşımının en önemli noktası, parçacıkların, kendinden iyi çözümler gösteren parçacıklar üzerinde itme, kendinden kötü çözüm gösteren diğer parçacıklar üzerinde ise çekme kuvveti uygulamasıdır.

Şekil 2.3a’da minimizasyon problemi için, parçacık 3 üstüne uygulanan bileşke kuvvet (\mathbf{F}_3) gösterilmiştir. $f(\mathbf{x}_1) < f(\mathbf{x}_3)$ olduğundan, parçacık 1, parçacık 3’ü çeker. Diğer taraftan, parçacık 2 ile karşılaştırıldığında, parçacık 3 daha iyi bir çözümü gösterdiğinden ($f(\mathbf{x}_3) < f(\mathbf{x}_2)$), parçacık 2 parçacık 3’ü iter. Sonuç olarak, parçacık 3, bileşke kuvvet \mathbf{F}_3 doğrultusunda hareket eder. Maksimizasyon probleminde ise durum tam tersidir (bkz. Şekil 2.3).



Şekil 2.3. EMA'da bileşke kuvvetin yönü

Denklem 2.3'te görüldüğü gibi \mathbf{F}_i , parçacıklar arasındaki mesafe ile ters orantılı ve parçacıkların yükleri çarpımları ile doğru orantılıdır. Parçacık yükleri de, parçacıkların gösterdiği çözümler ile doğru orantılı olduğundan $f(\mathbf{x}_i)$ ile $f(\mathbf{x}_j)$ arasındaki fark ne kadar büyük olursa parçacık i 'ye o kadar büyük bir bileşke kuvvet (\mathbf{F}_i) uygulanır. Denklem 2.3'te m popülasyon büyüklüğünü göstermektedir (Birbil ve Fang 2003).

$$\mathbf{F}_i = \sum_{j=1, j \neq i}^m \begin{cases} (\mathbf{x}_j - \mathbf{x}_i) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2}, & \text{eğer } f(\mathbf{x}_j) < f(\mathbf{x}_i) \\ (\mathbf{x}_i - \mathbf{x}_j) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2}, & \text{eğer } f(\mathbf{x}_j) \geq f(\mathbf{x}_i) \end{cases}, \forall i. \quad (2.3)$$

2.2.2. Parçacıkların hareket ettirilmesi

Temel EMA mekanizmasında, her parçacığın pozisyon vektörünün bileşen değerleri için üst ve alt sınırlar belirlenir. Başka bir deyişle, k . koordinat değerine (karar değişkenine) bir üst sınır (U^k) ve bir alt sınır (L^k) tanımlanır. Bu sınırlama aynı zamanda, çok boyutlu çözüm uzayının genişlemesini de engelleyerek, işlem hacminin büyümesini engeller. Bir parçacığın üzerine etki edecek bileşke kuvvet hesaplandıktan sonra, parçacık bileşke kuvvetin doğrultusunda Denklem 2.4'de gösterildiği gibi hareket ettirilir. Denklem 2.4'te λ daha önce gidilmeyen noktalara da gidilmesini sağlayan rassal bir parametre ($\lambda \in \mathbb{R}: 0 \leq \lambda \leq 1$), RNG ise bir vektördür ve bileşenleri, \mathbf{x}_i pozisyon vektörünün her k . bileşeni için belirlenen üst veya alt sınırlar içerisinde yapılabileceği hareketi belirler. Denklem 2.5 bu sınırlamayı özetlemektedir.

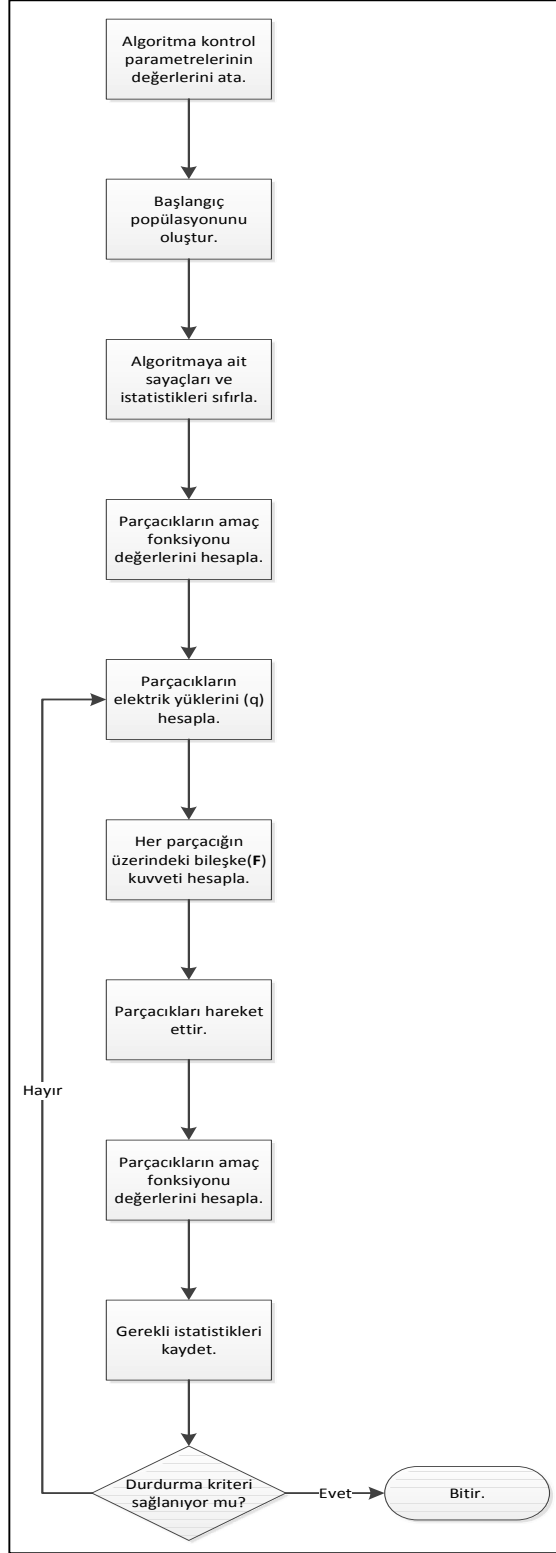
$$\mathbf{x}_i = \mathbf{x}_i + \lambda \frac{\mathbf{F}_i}{\|\mathbf{F}_i\|} (RNG), \forall i. \quad (2.4)$$

$$x_i^k \in \mathbb{R}: \quad U^k \leq x_i^k \leq L^k, \quad k = 1, \dots, D. \quad (2.5)$$

2.2.3. Elektromanyetik algoritmanın adımları

Özetlemek gerekirse, temel EMA yapısı 3 adımdan oluşur: (1) elektrik yükü hesaplaması, (2) bileşke kuvvet hesaplaması ve (3) parçacıkların hareket ettirilmesi (pozisyonlarının güncellenmesi). Bu adımlardan oluşan temel EMA akışı Şekil 2.4’de gösterilmiştir. Önceki bölümlerde detayları açıklanan EMA’nın adımları ise aşağıda verilmiştir.

- Adım 1: Popülasyon büyüklüğü (m) ve durdurma kriteri gibi algoritma parametrelerinin atanması.
- Adım 2: Belirlenen popülasyon büyüklüğü, problem büyüklüğü (D) ve değişkenlerin alt ve üst sınırları (L , U) kullanılarak başlangıç parçacıklarının oluşturulması (bkz. Denklem 2.1)
- Adım 3: Algoritmaya ait tutulacak istatistiklerin (global en iyi çözüm değeri ve parçacık değeri) ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için amaç fonksiyonu $f(\mathbf{x})$ değerlerinin hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For $i = 1$ to m Do**
- Adım 7: Parçacık için elektrik yükünü hesapla (bkz. Denklem 2.2)
- Adım 8: Parçacık üzerindeki bileşke kuvveti güncelle (bkz. Denklem 2.3)
- Adım 9: **End For**
- Adım 10: **For $i = 1$ to m Do**
- Adım 11: Parçacığı bileşke kuvvet ile hareket ettir (bkz. Denklem 2.4).
- Adım 12: Parçacık için amaç fonksiyonu $f(\mathbf{x})$ değerini hesapla.
- Adım 13: Gerekli istatistikleri güncelle.
- Adım 14: **End For**
- Adım 15: **Until** (durdurma kriteri sağlanıyorsa)



Şekil 2.4. EMA'nın akış diyagramı

3. ELEKTROMANYETİK ALGORİTMANIN PERFORMANS ANALİZİ

Bu bölüm kapsamında, temel EMA literatürde sıklıkla kullanılan meta sezgisel algoritmalar ile karşılaştırılmış ve performansı kısıtsız optimizasyon problemleri üzerinde analiz edilmiştir. Bu bağlamda, ABC, DGA ve PSO yapıları karşılaştırma algoritmaları olarak kullanılmış ve özdeş ortamlarda gerçekleştirilen detaylı analizler sonucunda temel EMA'nın güçlü ve zayıf yönleri ortaya çıkarılmıştır.

3.1. Karşılaştırma Algoritmaları

Bu bölümde karşılaştırma algoritmaları olarak kullanılan ABC, PSO ve DGA yöntemlerinin açıklamaları, algoritma adımları ve kısa literatür özetleri verilmiştir.

3.1.1. Yapay arı kolonisi algoritması

Yapay arı kolonisi (ABC) biyolojiden esinlenmiş, sürü tabanlı bir meta sezgisel algoritmadır. ABC arıların akıllı yiyecek arama davranışlarının benzetimini yapar. ABC, ilk olarak Karaboga ve Basturk (2007) tarafından geliştirilmiştir. Literatürde, ABC algoritması ile yapılan birçok başarılı kısıtsız optimizasyon (Akay ve Karaboga 2012, Gao ve Liu 2011, 2012, Kang ve ark. 2011, Wang ve ark. 2014, Zhu ve Kwong 2010) ve birleşimsel optimizasyon (Pan ve ark. 2011, Szeto ve ark. 2011, Tasgetiren ve ark. 2011) uygulamaları vardır. ABC hakkında daha detaylı bilgi, literatür taraması ve uygulama örnekleri için okuyucular, Karaboga ve ark. (2014) ve Liao ve ark. (2013) tarafından yapılan çalışmalardan faydalanabilirler.

ABC arıları üç gruba ayırmıştır: (i) işçi arılar, (ii) kâşif arılar ve (iii) gözcü arılar. İşçi arılar yiyecek kaynaklarını sömürmekten ve yiyecek kaynaklarının kalitesini diğer arı türleri ile paylaşmaktan sorumludurlar. Gözcü arılar ise paylaşılan bilgiler aracılığı ile yiyecek kaynaklarının kalitelerini karşılaştırır ve kaynakların kaliteleri ile orantılı bir olasılıkla kaynakları seçerek kaynağı sömürmeye devam eder. ABC mekanizmasında bir yiyecek kaynağı belirli bir deneme sonra hala geliştirilemiyorsa, o kaynak kapatılır ve o yiyecek kaynağındaki işçi arı kâşif arıya dönüşür. Kâşif arı çözüm alanı içerisinde rasgele yeni yiyecek kaynakları bulur.

ABC’de, başlangıç yiyecek kaynakları, EMA’ya benzer şekilde belirtilen alt ve üst sınırlar içerisinde, Denklem 3.1 kullanılarak rasgele oluşturulur.

$$x_{i,j} = L^j + rand(0,1)(U^j - L^j), \quad \forall i,j. \quad (3.1)$$

Burada, U^j j . karar değişkeninin üst sınırı, L^j ise j . karar değişkeninin alt sınırını göstermektedir. $x_{i,j}$ i . yiyecek kaynağının j . karar değişkenini, $rand(0,1)$ ise 0 ile 1 arasında rassal bir sayıyı ifade etmektedir. $i = 1, \dots, SN$ ve $j = 1, \dots, D$ ise sırasıyla yiyecek kaynağı adedini ve problem büyüklüğünü (karar değişkeni sayısını) göstermektedir.

ABC yapısında işçi arılar Denklem 3.2’yi kullanarak yeni yiyecek kaynakları ararlar.

$$v_{i,j} = x_{i,j} + \Phi_{i,j}(x_{i,j} - x_{k,j}) \quad (3.2)$$

Burada, $v_{i,j}$ oluşturulan yeni yiyecek kaynağını, k ise $[1, SN]$ aralığında seçilen ve i ’ye eşit olmayan rassal yiyecek kaynağını göstermektedir. $\Phi_{i,j}$ ise $[-1, 1]$ arasında üretilen rassal gerçek sayıdır. İşçi arıları v ve i ’yi kıyaslar ve aralarında açgözlü (greedy) bir seçim uygulayarak sömürmeye devam ederler.

Gözcü arılar ise her yiyecek kaynağı için bir seçilme olasılığı ($prob_i$) hesaplar ve rulet çemberi tekniği (Goldberg ve Holland 1988) ile seçtiği yiyecek kaynaklarını, Denklem 3.2’yi kullanarak geliştirir. $prob_i$ değeri, Denklem 3.3’te görüldüğü gibi uygunluk değeri (fit) kullanılarak amaç fonksiyonu değerleri ile orantılı bir şekilde hesaplanır (bkz. Denklem 3.4). Gözcü arılar da, aynı işçi arılar gibi x ve v arasında açgözlü seçim uygular. Buradaki önemli nokta, daha kaliteli yiyecek kaynaklarının daha yüksek seçilme olasılığına sahip olması ve daha çok geliştirilmesidir. ABC yapısında, bir yiyecek kaynağının kaç deneme boyunca geliştirilemediğine karşılık gelen *failure* değişkeni kullanılmaktadır. Algoritmadaki açgözlü seçim noktalarında eğer x geliştirilemediyse, $failure_i$ bir arttırılır. Belirli bir adet (*limit*) denemeden sonra geliştirilemeyen yiyecek kaynağı kapatılır ve o kaynaktaki işçi arı, kâşif arıya dönüştürülür. Kâşif arı Denklem 3.1’i kullanarak rasgele yeni yiyecek kaynağı bulur.

$$prob_i = \frac{fit_i}{\sum_{j=1}^{SN} fit_j} \quad (3.3)$$

$$fit_i = \begin{cases} \frac{1}{1+f(x_i)}, & f(x_i) \geq 0 \\ 1 + |f(x_i)|, & f(x_i) < 0 \end{cases} \quad (3.4)$$

ABC'nin temel adımları aşağıda verilmiştir.

- Adım 1: Popülasyon büyüklüğü (SN), limit değeri ($limit$) ve durdurma kriteri gibi algoritma parametrelerinin atanması.
- Adım 2: Belirlenen popülasyon büyüklüğü (SN), problem büyüklüğü (D) ve değişkenlerin alt ve üst sınırları (L , U) kullanılarak başlangıç parçacıklarının oluşturulması (bkz. Denklem 3.1)
- Adım 3: Algoritmaya ait tutulacak istatistiklerin (global en iyi çözüm değeri ve kaynak değeri) ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için amaç fonksiyonu $f(x)$ değerlerinin hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For $i = 1$ to SN Do**
- Adım 7: İşçi arılar ile yeni v çözümleri üret. (bkz. Denklem 3.2)
- Adım 8: x ile v arasında ağırlıklı seçim yap ve $failure_i$ 'yi güncelle.
- Adım 9: **End For**
- Adım 10: Yiyecek kaynakları için olasılıkları hesapla. (bkz. Denklem 3.3 ve 3.4)
- Adım 11: **For $i = 1$ to SN Do**
- Adım 12: Gözcü arılar tarafında seçilen kaynaklardan yeni çözümler üret. (bkz. Denklem 3.2)
- Adım 13: x ile v arasında ağırlıklı seçim yap ve $failure_i$ 'yi güncelle.
- Adım 14: Gerekli istatistikleri güncelle.
- Adım 15: **End For**
- Adım 16: **For $i = 1$ to SN Do**

Adım 17: Eğer $failure_i > limit$ ise, kâşif arılar ile yeni rassal kaynak üret (bkz. Denklem 3.1).

Adım 18: **End For**

Adım 19: **Until** (durdurma kriteri sağlanıyorsa)

3.1.2. Parçacık sürüsü optimizasyonu

Parçacık sürüsü optimizasyonu (PSO), kuş sürülerinin iki boyutlu hareketlerinden esinlenerek Kennedy ve Eberhart (1995) tarafından geliştirilmiştir. PSO popülasyon tabanlı, sürü zekasına dayalı bir meta sezgisel algoritmadır. PSO’da, her bir aday çözüm (parçacık) bir sonraki pozisyonunu, hız (velocity) vektörü, kendi tecrübesine (kendisinin en iyi pozisyonuna) ve sürü tecrübesine (global en iyi parçacığın pozisyonuna) göre ayarlar (Kennedy ve Eberhart 1995). Literatürde, PSO ve PSO sürümleri birçok farklı optimizasyon probleminde başarılı bir şekilde kullanılmıştır (Chatterjee ve Siarry 2006, Liang ve ark. 2006, Liao ve ark. 2007, Lim ve Mat Isa 2014, Tripathi ve ark. 2007, Zhan ve ark. 2009). Detaylı bir PSO analizi ve literatür taraması için Banks ve ark. (2007, 2008), Poli (2008), Poli ve ark. (2007), Thangaraj ve ark. (2011) çalışmaları incelenebilir.

PSO yapısında popülasyon, limitler dahilinde rasgele olarak oluşturulur ve Denklem 3.5’te özetlenmiştir.

$$x_i^{t=0} = L + rand(0,1)(U - L), \quad \forall i. \quad (3.5)$$

Burada, U üst sınır vektörü, L ise alt sınır vektörü ve t ise o anki jenerasyonu göstermektedir (Başlangıç popülasyonu $t = 0$ olarak gösterilir). $x_i^{t=0}$ i . parçacığın t . jenerasyondaki değerini, $rand(0,1)$ ise sıfır ile bir arasında rassal bir sayıyı ifade etmektedir. $i = 1, \dots, N$ ise popülasyon büyüklüğünü göstermektedir.

PSO’da her parçacığın pozisyonu Denklem 3.6 ve 3.7 kullanılarak güncellenir.

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (gbest^t - x_i^t) \quad (3.6)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3.7)$$

Burada, $i = 1, \dots, N$ popülasyon büyüklüğü, t ise jenerasyonu tutan değişkendir. v_i^t hız vektörü, $pbest_i^t$ i . parçacığın t . jenerasyona kadar elde ettiği en iyi amaç fonksiyonu değerini veren vektör, $gbest^t$ ise t . jenerasyona kadar bulunan global en iyi vektörü gösterir. ω atalet momenti, c_1 ve c_2 sabit algoritma parametreleridir. r_1 ve r_2 ise $[0, 1]$ aralığında rassal gerçek sayılardır. Denklem 3.6'da ilk terim atalet etkisi, ikinci terim kişisel etki ve son terim ise sosyal etki olarak adlandırılır. Sosyal etki için genelde global en iyi kullanılmış olsa da ($gbest$), bazı PSO sürümlerinde yerel en iyi parçacık da kullanılmıştır (Xin ve ark. 2010). Atalet, kişisel ve sosyal etkilerin, yeni çözümler üretmedeki ağırlıkları ω , c_1 ve c_2 parametreleri kullanılarak dengelenir. Oluşturulan hız vektörü v_i^{t+1} , x_i^t 'e eklenerek parçacık için yeni pozisyon elde edilir (bkz. Denklem 3.7).

PSO'nun sözde kodu aşağıda verilmiştir.

- Adım 1: Popülasyon büyüklüğü N , ω , c_1 ve c_2 kontrol parametreleri ve durdurma kriteri gibi algoritma parametrelerinin atanması.
- Adım 2: Belirlenen popülasyon büyüklüğü (N), problem büyüklüğü (D) ve değişkenlerin alt ve üst sınırları (L , U) kullanılarak başlangıç parçacıklarının oluşturulması (bkz. Denklem 3.5)
- Adım 3: Algoritmaya ait tutulacak istatistiklerin (hız vektörleri, global en iyi çözüm, $pbest$ vb.) ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için amaç fonksiyon değerlerinin $f(x)$ hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For** $i = 1$ **to** N **Do**
- Adım 7: Hız vektörünü hesapla (v_i^t) (bkz. Denklem 3.6)
- Adım 8: x_i^{t+1} 'i hesapla (bkz. Denklem 3.7).
- Adım 9: Gerekli ise, $pbest_i$ ve $gbest$ 'i güncelle.
- Adım 10: **End For**
- Adım 11: Gerekli istatistikleri güncelle.
- Adım 12: $t = t + 1$

Adım 13: **Until** (durdurma kriteri sağlanıyorsa)

3.1.3. Diferansiyel gelişim algoritması

Diferansiyel gelişim algoritması (DGA), gerçek sayı değerli fonksiyonların optimizasyonu için geliştirilmiş, popülasyon tabanlı ve gelişime dayalı bir meta sezgisel algoritmadır. DGA, Storn ve Price (1997) tarafından geliştirilmiştir. DGA basit olarak mutasyon, çaprazlama ve seçim operatörlerine bağlı olarak çalışır. Literatürde birçok farklı DGA sürümü geliştirilmiş ve geliştirilen algoritmalar sürekli, doğrusal kısıtlı ve birleşimsel optimizasyon problemlerinin çözümünde kullanılmıştır (Brest ve ark. 2006, Liu ve Lampinen 2005, Pan ve ark. 2008, Pan ve ark. 2009, Qin ve ark. 2009, Qin ve Suganthan 2005). Daha detaylı literatür taramaları için Neri ve Tirronen (2010), Das ve Suganthan (2011) ve Das ve ark. (2009) çalışmaları incelenebilir.

DGA yapısında ilk olarak, diğer meta sezgisel algoritmalarda olduğu gibi başlangıç popülasyonu, Denklem 3.8'te verildiği gibi limitler içerisinde rasgele olarak oluşturulur.

$$x_{i,j,g=0} = L^j + rand(0,1)(U^j - L^j), \quad \forall i, j. \quad (3.8)$$

Burada, U^j j . parametrenin üst sınırını, L^j j . parametrenin alt sınırını ve g ise o anki jenerasyonu göstermektedir (Başlangıç popülasyonu $g = 0$ olarak gösterilir). $x_{i,j,g}$ i . vektörünün j . parametresinin g . jenerasyondaki değerini, $rand(0,1)$ ise 0 ile 1 arasında rassal bir sayıyı ifade etmektedir. $i = 1, \dots, N$ ve $j = 1, \dots, D$ ise sırasıyla popülasyon büyüklüğünü ve problem büyüklüğü olarak parametre sayısını göstermektedir.

DGA'da bir yineleme sonraki jenerasyonu oluşturulmak için mutasyon ve çaprazlama operatörleri kullanılır. Mutasyon vektörü (v), rasgele seçilmiş iki vektörün farkının bir F faktörü ile çarpımı ile hesaplanan vektörün, toplum içerisinde rasgele seçilmiş ancak diğer iki vektörden farklı bir baz vektöre eklenmesi ile elde edilir. Mutasyon işlemi Denklem 3.9'da gösterilmiştir.

$$v_{i,j,g} = x_{r1,j,g} + F(x_{r2,j,g} - x_{r3,j,g}), \quad r1 \neq r2 \neq r3 \neq i. \quad (3.9)$$

Burada F değeri pozitif gerçək bir sayıdır ve $[0, 2]$ aralığındadır. Literatürde DGA için birçok farklı mutasyon stratejisi kullanılmıştır. Genel olarak stratejiler DGA/A/B/C şeklinde gösterilir ve A mutasyona uğrayacak baz vektörün seçim şeklini, B fark vektörlerinin sayısını ve C çaprazlama tipini temsil eder. A'nın temsil ettiği baz vektörü popülasyon içerisinde rasgele seçilebildiği gibi (rand: r_1, r_2, \dots) o ana kadar bulunmuş en iyi çözümü veren vektör (*best*) de baz vektörü yerine kullanılabilir. B değişkenine göre 2 veya daha fazla vektörün farkları mutasyon işleminde kullanılabilir. C'nin temsil ettiği çaprazlama üstel (exp) veya binom (bin) olabilir (Neri ve Tirronen 2010). DGA'nın temel formunda DGA/rand/1/bin kullanılmıştır, ancak literatürde karşılaşılan diğer mutasyon stratejilerinden bazıları aşağıda verilmiştir (Civicioglu ve Besdok 2013).

- DGA/best/1/bin: $v_{i,j,g} = x_{best,j,g} + F(x_{r_2,j,g} - x_{r_3,j,g})$
- DGA/rand-to-best/1/bin: $v_{i,j,g} = x_{r_1,j,g} + F(x_{best,j,g} - x_{r_3,j,g})$
- DGA/best/2/bin: $v_{i,j,g} = x_{best,j,g} + F(x_{r_2,j,g} - x_{r_3,j,g}) + F(x_{r_3,j,g} - x_{r_4,j,g})$
- DGA/rand/2/bin: $v_{i,j,g} = x_{r_1,j,g} + F(x_{r_2,j,g} - x_{r_3,j,g}) + F(x_{r_3,j,g} - x_{r_4,j,g})$

Temel DGA mekanizmasında binom formunda çaprazlama kullanılır. Çaprazlama işleminde, mutasyon vektörü (v) ile önceki nesilden gelen x vektörü Denklem 3.10'a göre çaprazlanır ve $U_{i,g}$ deneme vektörü elde edilir. Deneme vektörünün parametreleri CR olasılığı ile mutasyon vektöründe, $(1-CR)$ olasılığı ile x vektöründe seçilmektedir.

$$U_{i,j,g} = \begin{cases} v_{i,j,g}, & \text{eğer } rand(0,1) \leq CR \text{ veya } j = jrand \\ x_{i,j,g}, & \text{diğer durumlarda.} \end{cases} \quad (3.10)$$

CR , $[0,1]$ aralığında gerçək bir sayıdır. Buradaki $j = jrand$ koşulu, en az bir adet parametrenin mutasyon vektöründen seçilmesini garanti etmek için kullanılmaktadır. Rasgele seçilen $jrand$. parametre CR değerine bakılmaksızın mutasyon vektöründen seçilir.

Son olarak, DGA'da oluşturulan deneme vektörü ile x arasında açgözlü bir seçim uygulanır ve karşılaştırılan vektörlerden daha iyi sonuç vereni bir sonraki jenerasyona aktarılır. Denklem 3.11'de seçim operatörü gösterilmiştir.

$$x_{i,g+1} = \begin{cases} U_{i,g}, & \text{eğer } f(U_{i,g}) < f(x_{i,g}) \\ x_{i,g}, & \text{diğer durumlarda} \end{cases} \quad (3.11)$$

DGA'nın sözde kodu aşağıda verilmiştir.

- Adım 1: N , F ve CR gibi kontrol parametreleri ve durdurma kriteri gibi algoritma parametrelerinin atanması.
- Adım 2: Belirlenen popülasyon büyüklüğü (N), problem büyüklüğü (D) ve değişkenlerin alt ve üst sınırları (L, U) kullanılarak başlangıç bireylerinin oluşturulması (bkz. Denklem 3.8)
- Adım 3: Algoritmaya ait istatistiklerin (global en iyi çözüm değeri, vektör değeri) ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için $f(x)$ amaç fonksiyonu değerlerinin hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For** $i = 1$ to N **Do**
- Adım 7: Mutasyon vektörünü ($v_{i,g}$) oluştur (bkz. Denklem 3.9)
- Adım 8: $v_{i,g}$ ile $x_{i,g}$ çaprazlama yap ve deneme vektörünü ($U_{i,g}$) oluştur (bkz. Denklem 3.10)
- Adım 9: $U_{i,g}$ ile $x_{i,g}$ arasında açgözlü seçim uygula, $x_{i,g+1}$ 'i oluştur (bkz. Denklem 3.11)
- Adım 10: **End For**
- Adım 11: Gerekli istatistikleri güncelle.
- Adım 12: $g = g + 1$
- Adım 13: **Until** (durdurma kriteri sağlanıyorsa)

3.2. EMA, ABC, DGA ve PSO'nun kavramsal karşılaştırılması

Bu bölüm kapsamında, EMA ile diğer meta sezgisel algoritmalar, başlangıç popülasyonu oluşturma yöntemleri, bilgi paylaşımı, keşfetme ve sömürme, operatörler ve algoritma kontrol parametreleri gibi başlıklar altında karşılaştırılmışlardır. Kısaca hatırlatılacak olursa, EMA fizikten esinlenen ve elektrik yüklü parçacıkların birbirlerini itme-çekme prensiplerini esas alan meta sezgisel bir algoritma olup, ABC ve PSO sürü zekası, DGA ise evrimsel tabanlı bir algoritmadır. ABC, PSO ve DGA, EMA gibi popülasyon tabanlı algoritmalarıdır. Tüm algoritmaların başlangıç popülasyonları problem tarafından tanımlanan çözüm alanı içerisinde rassal olarak oluşturulur (bkz. Denklem 2.1, 3.1, 3.5 ve 3.8). EMA'da, diğer algoritmalara nazaran daha az kontrol parametresi kullanılmaktadır. EMA yapısında sadece popülasyon büyüklüğü belirlenirken, ABC'de *limit* ve *SN*, DGA'da *N*, *F* ve *CR* parametreleri ve PSO'da ise *N*, ω , c_1 ve c_2 parametrelerinin belirlenmesi gerekmektedir. Yapılan çalışmalarda, kontrol parametrelerinin algoritma performanslarına büyük oranda etkide bulunduğu gösterilmiştir (Civicioglu ve Besdok 2013). Uygun parametre setlerinin belirlenmesinin zor ve zaman alıcı oluşu, meta sezgisel uygulamaların zayıflıklarından en önemlisi olarak kabul edilmektedir. Bu sebeple, daha az parametre ile çalışan algoritmaların daha kolay uygulanabilir olduğu savından (Yuan ve ark. 2012) yola çıkarak, EMA'nın diğer algoritmalara göre bu açıdan üstün olduğu söylenebilir. EMA'da, yeni komşu çözümler hesaplanan bileşke kuvvet kullanılarak oluşturulmaktadır (bkz. Denklem 2.3). Başka bir deyişle, EMA yeni çözüm üretiminde tüm popülasyonun o anki bilgilerini kullanmaktadır. ABC sadece rassal olarak seçilen başka bir yiyecek kaynağını (bkz. Denklem 3.2), PSO kişisel en iyi ve global en iyi parçacıkları (bkz. Denklem 3.6) ve DGA rassal olarak seçilen üç adet vektörü kullanmaktadır (bkz. Denklem 3.9). Yeni komşuluk üretiminde daha çok ve yoğun bilginin kullanılmasının çeşitli avantajları ve dezavantajları vardır. EMA'nın söz konusu avantaj ve dezavantajları ilerleyen bölümlerde detaylı olarak incelenmiştir. Bir meta sezgisel algoritma, iyi bir performans gösterebilmek için keşfetme/sömürme dengesini çok iyi kurmalıdır. EMA sömürmeyi arttırmak için sadece daha iyi çözümlere daha yüksek elektrik yükü atamaktadır. ABC, gözcü arı fazında daha iyi çözümlere daha yüksek seçilme olasılığı vermektedir ve her yeni komşuluk üretiminde aç gözlü seçim uygulamaktadır (bkz. Denklem 3.3). PSO yeni komşuluk üretiminde hem global en iyiyi hem de kişisel en iyiyi kullanmaktadır.

DGA ise yeni çözüm üretiminde açgözlü seçim uygulamaktadır. EMA, PSO ve ABC yeni çözüm üretiminde rassal çarpanlar kullanarak keşfetmeyi arttırmaktadır. DGA tamamen rassal seçilen üç vektörden oluşturulan aday vektör ile olasılığa bağlı bir çaprazlama gerçekleştirmektedir. Ek olarak, ABC diğer algoritmalarda olmayan bir yeniden oluşturma operatörü kullanmaktadır. Çizelge 3.1’de algoritmaların kavramsal olarak karşılaştırılması özetlenmiştir.

Çizelge 3.1. Algoritmaların kavramsal olarak karşılaştırılması

	EMA	ABC	DGA	PSO
Özellik	Popülasyon tabanlı	Sürü zekası, popülasyon tabanlı	Evrimsel, popülasyon tabanlı	Sürü zekası, popülasyon tabanlı
Esinlenen yöntem	Elektrik yüklü parçacıkların elektrik alan içerisindeki davranışları	Arı kolonilerinin yiyecek arama davranışları	Gelişime dayalı algoritma	Kuş sürülerinin yiyecek arama davranışları
Başlangıç popülasyonu	Belirlenen sınırlar içinde rassal	Belirlenen sınırlar içinde rassal	Belirlenen sınırlar içinde rassal	Belirlenen sınırlar içinde rassal
Kontrol parametreleri	m	$SN, limit$	N, F ve CR	N, ω, c_1 ve c_2
Yeni komşuluk üretimi bilgi kaynağı	Tüm popülasyon	Kendisi ve rassal olarak seçilmiş bir çözüm.	Kendisi ve rassal olarak seçilmiş üç adet vektörün bileşimi.	Kendisi, kişisel en iyi nokta ve global en iyi çözüm
Sömürme mekanizması (İyi çözümlere yönelme)	İyi çözümlere daha yüksek elektrik yükü atanması	Gözcü arıların olasılığa bağlı seçimi, her yeni çözümde uygulanan açgözlü seçim.	Yeni çözümde uygulanan açgözlü seçim.	Global en iyi nokta ve kişisel en iyi nokta bilgilerinin yeni çözümde kullanılması
Keşfetme mekanizması (Yeni alanlara yönelme)	Parçacığın hareket ettirilmesindeki rassal çarpan.	Limit değerine göre yeniden oluşturma, Yeni komşuluk üretiminde rassal çarpan ve rassal çözüm kullanılması.	Rassal çözümlerden aday vektör üretilmesi, CR parametresine bağlı rassal çarpazlama	Yeni çözüm üretiminde kullanılan c_1 ve c_2 rassal çarpanları.

3.3. Test Problemleri

Meta sezgisel algoritmaların performanslarının sürekli fonksiyonlar üzerinde kıyaslaması literatürde yaygın olarak kullanılan bir yöntemdir. Algoritma performanslarını tarafsız ve doğru bir şekilde karşılaştırabilmek için, geniş ve birbirinden farklı özelliklere sahip problemlerden oluşan bir test kümesi tanımlamak gerekmektedir. Bu çalışma kapsamında EMA ve diğer meta sezgisel algoritmaların temel sürümlerinin performansları, (i) standart test problemleri ve (ii) karma test problemleri gibi iki ayrı küme içinde yer alan test problemleri üzerinden analiz edilmiştir.

Çizelge 3.2’de standart test problemlerinin bazı temel özellikleri verilmiştir. Söz konusu standart test problemlerinin formülasyonları Ek 1’de incelenebilir. Çizelge 3.2’de x^* , optimum çözüm vektörünü, $f(x^*)$, optimal amaç fonksiyonu değerini, sınırlar ise değişkenlerin alt ve üst sınırlarını göstermektedir. Karakter sütununda ise problemlerin temel özellikleri verilmiştir (U: tek-modlu, M: çok-modlu, S: ayrıştırılabilir, N: ayrıştırılamayan). Kullanılan standart test problemleri arasında, birden çok yerel optimumu olan (çok-modlu) (Schwefel, Griewank, Weierstrass, Rastrigin, Ackley, Branin, G&P, Michalewick, Shekel, FoxHoles), ayrıştırılamayan (Rosenbrock, Griewank, Weierstrass, Ackley, Matyas, Shekel), global minimumu yerel minimumlara çok yakın konumlanmış (Schaffer), simetrik olmayan (Rastrigin, Branin), rassal gürültü bileşeni olan (Quartic), vb. özelliklere sahip problemler vardır. Standart test problemlerinde ilk sekiz problem dinamik boyutlu, diğerleri ise statik boyutludur. Söz konusu test problemlerinde dokuz adet tek-modlu, 20 adet çok-modlu, 12 adet ayrıştırılabilir, 17 adet ayrıştırılamayan problem kullanılmıştır. Standart test problemleri toplamda 29 problemden oluşmaktadır ve literatürde çokça kullanılan problemler arasından seçilmişlerdir (Gao ve Liu 2011, 2012, Gao ve ark. 2014b, Jia ve ark. 2014, Karaboga ve Basturk 2008, Karaboga ve Gorkemli 2014, Shi ve ark. 2011, Wang ve ark. 2014, Xiang ve ark. 2014a, Zhan ve ark. 2009, Zhu ve Kwong 2010). Bazı fonksiyonların (FoxHoles, Shekel ve Hartmann problemleri) sahip oldukları sabit parametre değerleri ise Ek 2’de verilmiştir.

Çalışma kapsamında kullanılan diğer test problemleri kümesinde ise Suganthan ve ark. (2005) tarafından oluşturulan karma test problemleri yer almaktadır. Test algoritmalarının ve test problemlerinin kendilerine ait bir takım özellikleri karşılaştırmaların objektif bir şekilde yapılmasını bazen engelleyebilir. Meta sezgisel algoritmalar bazı problemler üzerinde çok iyi performans sergilerken, diğerleri üzerinde kötü sonuçlar verebilirler. Örnek olarak, yeni çözümleri daha çok diğer çözüm vektörlerinin kopyalanması ile üreten algoritmalar simetrik problemlerde çok iyi sonuçlar elde edebilirler (Akay 2009, Liang ve ark. 2005). Algoritma ve problemlerden kaynaklanan özel durumlar sebebiyle performans sapmalarının önüne geçebilmek için CEC 2005 konferansında özel bir oturum düzenlenmiş ve söz konusu oturumda özel test problemleri tanımlanmıştır (Suganthan ve ark. 2005). Karma test problemleri, toplamda 25 adet tek-modlu, çok-modlu, kaydırılmış ve döndürülmüş karma fonksiyondan oluşur. Ek 3'te detaylı yapıları verilen fonksiyonların bazı temel bilgileri de Çizelge 3.3'de verilmiştir. Çizelge 3.3'de karma test fonksiyonlarının isimleri, değişkenlerin sınırları ve problemlerin karakterleri görülebilir. Karma test problemleri son yıllarda yapılan birçok çalışmada kullanılmıştır (Akay ve Karaboga 2012, Civicioglu ve Besdok 2013, El-Abd 2012, Gao ve ark. 2014b, Jia ve ark. 2014, Liao ve ark. 2013, Wang ve ark. 2014, Xiang ve ark. 2014b). Karma test problemlerinin detayları, oluşturma yöntemleri ve özellikleri için Suganthan ve ark. (2005) çalışması incelenebilir.

Çizelge 3.2. Standart test problemleri

No	Fonksiyon	x^*	$f(x^*)$	Sınırlar	Karakter
1	Sphere	[0,0, ...,0]	0	$[-100, 100]^D$	US
2	Rosenbrock	[1,1, ...,1]	0	$[-2,048, 2,048]^D$	UN
3	Schwefel	[420.96,420.96, ...,420.96]	0	$[-500, 500]^D$	MS
4	Griewank	[0,0, ...,0]	0	$[-600, 600]^D$	MN
5	Weierstrass	[0,0, ...,0]	0	$[-0,5, 0,5]^D$	MN
6	Quartic	[0,0, ...,0]	0	$[-1,28, 1,28]^D$	US
7	Rastrigin	[0,0, ...,0]	0	$[-5,12, 5,12]^D$	MS
8	Ackley	[0,0, ...,0]	0	$[-32,768, 32,768]^D$	MN
9	Beale	[3, 0,5]	0	$[-4,5, 4,5]^2$	UN
10	Bohachevsky1	[0, 0]	0	$[-100, 100]^2$	MS
11	Bohachevsky1	[0, 0]	0	$[-100, 100]^2$	MN
12	Bohachevsky1	[0, 0]	0	$[-100, 100]^2$	MN
13	Booth	[1, 3]	0	$[-10, 10]^2$	MS
14	Branin	$[-\pi, 12,275], [\pi, 2,275],$ [9,42478, 2,475]	0,397887	$[-5, 10] \times [0, 15]$	MS
15	Colville	[1, 1, 1, 1]	0	$[-10, 10]^4$	UN
16	Easom	$[\pi, \pi]$	-1	$[-100, 100]^2$	UN
17	G&P	[0, 0]	0	$[-2, 2]^2$	MN
18	Hartmann1	[0,114614, 0,55564, 0,852547]	-3,86278	$[0,1]^3$	MN
19	Hartmann2	[0,20169, 0,15001, 0,476874, 0,275332, 0,311652, 0,6573]	-3,32237	$[0, 1]^6$	MN
20	Matyas	[0, 0]	0	$[-10, 10]^2$	UN
21	Michalewick1	[2,2029055, 1,5707963]	-1,8013	$[0, \pi]^2$	MS
22	Michalewick2	[2,2029055, 1,5707963, 1,2819916, 1,9230585, 1,7204698]	-4,6876582	$[0, \pi]^5$	MS
23	Michalewick3	[2,2029055, 1,5707963, 1,2819916, 1,9230585, 1,7204698, 1,5707963, 1,4544140, 1,7560865, 1,5707963]	-9,6601517	$[0, \pi]^{10}$	MS
24	Shekel1	[4, 4, 4, 4]	-10,1532	$[0, 10]^4$	MN
25	Shekel2	[4, 4, 4, 4]	-10,4029	$[0, 10]^4$	MN
26	Shekel3	[4, 4, 4, 4]	-10,5364	$[0, 10]^4$	MN
27	Trid6	[0, 0]	-50	$[D^2, D^2]$	UN
28	Stepint	[0, 0]	0	$[-5,12, 5,12]^4$	US
29	Foxholes	[0, 0]	0	$[-65,536, 65,536]^5$	MS

Çizelge 3.3. Karma test problemleri

No	İsmi	Sınırlar	Karakter
1	Kaydırılmış Sphere fonksiyonu	$[-100, 100]^D$	US
2	Kaydırılmış Schwefel problemi 1.2	$[-100, 100]^D$	UN
3	Kaydırılmış döndürülmüş yüksek dereceli eliptik fonksiyon	$[-100, 100]^D$	UN
4	Ötelenmiş Schwefel's problemi 1.2 gürültülü	$[-100, 100]^D$	UN
5	Genişletilmiş Schwefel's problemi 2.6 global optimum sınırlar üzerinde	$[-100, 100]^D$	UN
6	Kaydırılmış Rosenbrock fonksiyonu	$[-100, 100]^D$	MN
7	Kaydırılmış döndürülmüş Griewank fonksiyonu parametre sınırlamasız	$[-100, 100]^D$	MN
8	Kaydırılmış döndürülmüş Ackley fonksiyonu global minimum sınırlar üzerinde	$[-32, 32]^D$	MN
9	Kaydırılmış Rastrigin fonksiyonu	$[-5, 5]^D$	MS
10	Kaydırılmış döndürülmüş Rastrigin fonksiyonu	$[-5, 5]^D$	MN
11	Kaydırılmış döndürülmüş Weierstrass fonksiyonu	$[-0,5, 0,5]^D$	MN
12	Schwefel problemi 2.13	$[-100, 100]^D$	MN
13	Kaydırılmış genişletilmiş Griewank+Rosenbrock fonksiyonu	$[-3, 1]^D$	MN
14	Kaydırılmış döndürülmüş genişletilmiş Schaffer F6 fonksiyonu	$[-5, 5]^D$	MN
15	Karma birleşim fonksiyonu	$[-5, 5]^D$	MS
16	Döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
17	Gürültülü döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
18	Döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
19	Global optimum dar alanda olan döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
20	Global optimum sınır üzerinde olduğu döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
21	Döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
22	Yüksek dereceli döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
23	Süreksiz döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
24	Döndürülmüş karma birleşim fonksiyonu	$[-5, 5]^D$	MN
25	Sınırlamasız döndürülmüş karma birleşim fonksiyonu	-	MN

3.4. Karşılaştırma Sonuçları

Bu bölüm kapsamında, temel EMA ile diğer seçilen test algoritmaların daha önceki bölümlerde açıklanan temel sürümleri karşılaştırılmıştır. EMA ve test algoritmaların tümü aynı bilgisayar ortamında kodlanmış ve algoritmaların performansları test problemleri üzerinde analiz edilmiştir. Toplam 108 test problemi (33 standart, 75 karma test problemi) üzerinde her algoritma ile 30 kez rassal başlangıçlı koşum gerçekleştirilmiş ve belirlenen durdurma kriteri sonunda elde edilen $f(x)$ fonksiyon değerlerinin ortalaması, standart sapması ve koşum süreleri raporlanmıştır. Çalışma ortamı olarak MATLAB (R2011a) kullanılmış ve Intel Core i5 CPU (1.30 GHz)

işlemcili ve 4 GB belleğe sahip bilgisayarda koşumlar gerçekleştirilmiştir. Bu bölüm kapsamında yapılan testlerde ABC (<http://mf.erciyes.edu.tr/abc/software.htm>, 2014), PSO (SPSO, 2007) (<http://www.particleswarm.info>, 2014) ve DGA (<http://www.icsi.berkeley.edu/~storn/code.html>, 2014) temel kodları esas alınmıştır. Söz konusu sitelerden alınan temel kodlar, üzerinde gerekli uyarlamalar yapıldıktan sonra uygulanmıştır.

Gerçekleştirilen karşılaştırma analizlerinin tarafsız ve doğru olarak yorumlanabilmesi için algoritmalar arasında ortak olan parametre değerleri aynı olacak şekilde atanmıştır. Standart test problemleri ve 10 boyutlu (10D) karma problemler için popülasyon büyüklükleri 50, algoritmaların durdurma kriteri ise 300000 amaç fonksiyonu değerlendirme adedi olarak atanmıştır. 30 ve 50 boyutlu karma test problemlerinde ise popülasyon büyüklükleri 80, durdurma kriteri ise 500000 amaç fonksiyonu değerlendirme adedi olarak alınmıştır. Algoritmalarda kullanılan özel kontrol parametreleri ise literatürdeki uygulamalara benzer şekilde atanmıştır. ABC’de kullanılan limit parametresi $limit = (SN * D) * 0.5$ olacak şekilde tanımlanmıştır (Akay 2009, Karaboga ve Basturk 2007, 2008). DGA yapısında, Bölüm 3.1.3’de belirtildiği gibi DGA/Rand/1 stratejisi kullanılmış, Ronkkonen ve ark. (2005) ve Price ve ark. (2006) tarafından önerilen parametre setleri kullanılmıştır. DGA’da $F = 0.5$ olarak atanmış, çaprazlama operatörünü kontrol eden CR ise Price ve ark. (2006) önerildiği şekilde $CR = 0.9$ olarak atanmıştır. PSO’da ise atalet momenti, kişisel ve sosyal etki katsayıları, w , c_1 ve c_2 sırasıyla 1,193, 1,193 ve 0,721 değerlerine eşitlenmiştir (Chen ve ark. 2013). EMA için tek kontrol parametresi popülasyon büyüklüğü olduğundan başka bir parametre ataması yapılmamıştır.

Çizelge Ek 4.1’de standart test problemlerinde EMA, ABC, PSO ve DGA ile yapılan koşumların sonuçları verilmiştir. Dinamik olarak boyutlandırılabilir problemlerde (Sphere, Rosenbrock, Griewank, Quartic, Ackley ve Rastrigin) 10 ve 30 problem boyutlarında testler yapılmıştır. Çizelgede her hücredeki ilk satır ortalama $f(x)$, ikinci satır ise $\sigma_{f(x)}$ standart sapmasını göstermektedir. Sonuçların daha anlaşılır olması için 10^{-12} (E-12)’den küçük değerler sıfır olarak raporlanmıştır. Çizelge Ek 4.2’de 25 adet karma test problemi için elde edilen sonuçlar verilmiştir. Söz konusu çizelgede, Çizelge

Ek 4.1'in formatına benzer şekilde, 10, 30 ve 50 boyutlu problemler için ilk satırda ortalama $f(x)$, ikinci satırda ise $\sigma_{f(x)}$ standart sapması verilmiştir. Çizelge Ek 4.1 ve Çizelge Ek 4.2'de verilen sonuçların farklarının istatistiksel olarak anlamlı olup olmadığı t testleri ile analiz edilmiştir. %95 anlamlılık düzeyinde yapılan testlerin sonuçları, standart test problemleri için Çizelge 3.4'de, karma test problemleri için ise Çizelge 3.5'de özetlenmiştir.

Çizelge 3.4 ve Çizelge 3.5'de EMA, diğer karşılaştırma algoritmasına göre %95 anlamlılıkta daha iyi performans göstermiş ise "+" ile, eğer diğer algoritma daha iyi performans göstermiş ise "-" ile gösterilmiştir. Son olarak eğer iki algoritma arasında belirlenen güvenilirlikte istatistiksel olarak anlamlı bir fark yok ise "E" ile kodlanmıştır. Çizelgelerin son satırında ise sonuçların toplamları verilmiştir. Sonuçlar incelendiğinde, her ne kadar test edilen algoritmalar birbirlerinden tamamen farklı bilgi kaynakları ile yeni komşu çözümler üretse de Trid6, Stepint, FoxHoles, Beale, Matyas, Bohachevsky1, Bohachevsky2 ve Bohachevsky3 gibi görece düz yüzeye sahip, simetrik veya simetriğe yakın, az parametrelili veya tek-modlu problemlerde tüm algoritmaların istatistiksel olarak benzer performanslar sergiledikleri görülmektedir.

Çizelge 3.4'nin son satırı incelendiğinde, EMA'nın 12 problemde ABC'nin gerisinde kaldığı, bir problemde ise ABC'den daha iyi performans gösterdiği görülmektedir. 20 problemde ise EMA ile ABC arasında istatistiksel olarak anlamlı bir fark gözlemlenmemiştir. EMA özellikle dinamik 30 boyutlu (30D) problemlerde ABC'den daha kötü sonuçlar elde etmiştir ve sadece Quartic 30D probleminde ABC'den daha kaliteli sonuçlar bulmuştur. Geri kalan beş adet 30D problemin dördünde ise ABC'nin gerisinde kalmıştır. EMA ile PSO algoritmasının sonuçları karşılaştırıldığında, 17 problemde benzer performans gösterdikleri, yedi problemde EMA ve dokuz problemde ise PSO daha iyi performans sergilediği görülmektedir. PSO, ABC'ye benzer şekilde 30D problemler üzerindeki testlerde EMA'ya üstünlük sağlamıştır. EMA sadece Quartic 30D probleminde PSO'dan daha kaliteli sonuçlara varmıştır. EMA ile DGA'nin performansları standart test problemleri için incelendiğinde, 20 problem için %95 güvenilirlikte istatistiksel olarak anlamlı bir fark gözlemlenmemiştir. EMA, Rosenbrock 30D, Griewank 10D, Quartic 10D ve 30D ve Shekel3 problemleri için DGA'dan daha

iyi sonuçlara ulaşmıştır. Geri kalan sekiz problemde ise DGA, EMA'ya üstünlük sağlamıştır.

Çizelge Ek 4.2 ve Çizelge 3.5'deki karma test problemleri üzerinde elde edilen sonuçlar incelendiğinde; EMA ile diğer meta sezgisel algoritmalar arasındaki performans farkı daha net bir şekilde görülmektedir. F4 10D, F3 30D, F6 10D, F18 10D ve F22 10D problemleri üzerinde tüm algoritmalar benzer sonuçlar elde etmiştir. EMA toplam 12 adet test probleminde ABC'ye, sekiz test probleminde PSO'ya ve 11 problemde ise DGA'ye karşı üstün performans göstermiştir. Fakat standart test problemleri üzerinde elde edilen sonuçlara benzer şekilde, problem boyutu büyüdükçe EMA performansını koruyamamıştır. ABC, PSO ve DGA test probleminde sırasıyla 46, 41 ve 46 kez EMA'ya karşı üstün performans sergilemiştir. EMA ile ABC arasında 14 problemde, PSO ile EMA arasında 23 problemde ve DGA ile EMA arasında 15 problem üzerinde ise anlamlı bir fark görülmemiştir.

Meta sezgisel algoritmaların başarısı sadece bulunan çözümlerin kalitesi ile ölçülmemektedir. Bulunan çözümün kalitesine ek olarak, çözümlerin bulunması için harcanan süre veya işlem adedi de ölçülmeli ve karşılaştırılmalıdır (Dongarra 2005). Bu sebeple, her iki test problem grubu üzerinde yapılan testlerde süre ölçümleri de yapılmıştır. Tüm algoritmalar için fonksiyon değerlendirme adedi aynı olacak şekilde (standart test problemleri ve 10D boyutlu karma test problemleri için 300000, diğer problemler için ise 500000 adet) belirlendiğinden, süre ölçümü tarafsız sonuçlar sağlayacaktır. Standart test problemleri ve karma test problemleri üzerinde elde edilen karşılaştırmalı süre ölçümleri sırasıyla Çizelge 3.6 ve Çizelge 3.7'da verilmiştir. Tabloların mutlak sürelerden bağımsız okunabilmesi için ölçülen değerler o test için en iyi değerden sapma oranı olarak verilmiştir ($\frac{\text{değer}-\text{en iyi değer}}{\text{en iyi değer}}$). Buna göre, bir satırda en kısa süre veya süreler için 0,00 değeri, diğer süreler için de en kısa süreye göre oranlar verilmiş olmaktadır. Çizelgelerin en son iki satırında ise elde edilen sonuçların ortalamaları ve bu ortalamaların, elde edilen en iyi ortalamaya göre normalize edilmiş değerleri (Normalize) verilmiştir.

Çizelge 3.6 incelendiğinde, EMA'nın standart test problemleri üzerinde ortalama olarak DGA'dan sekiz, PSO'dan beş ve ABC'den ise dört kat daha yavaş olduğu görülmektedir. PSO ve ABC ise süre bakımından benzer performans sergilemişlerdir. Söz konusu çizelge incelendiğinde, DGA'nın diğer algoritmalara göre daha hızlı bir şekilde çözüme giden bir algoritma olduğu görülmektedir. Karma test problemleri üzerinde elde edilen süre ölçümleri analiz edildiğinde (bkz. Çizelge 3.7) EMA'nın DGA, PSO ve ABC'den sırasıyla altı, dört ve iki kat daha yavaş bir yöntem olduğu sonucuna varılmıştır. Standart test problemlerine benzer şekilde, en hızlı performansı sırasıyla DGA, PSO ve ABC göstermiştir.

Çizelge 3.4. Standart test problemlerinde istatistiksel anlamlılık

Fonksiyon	D	EMA - ABC	EMA - PSO	EMA - DGA
Sphere	10	E	E	E
	30	E	E	E
Rosenbrock	10	E	E	E
	30	-	E	+
Griewank	10	-	-	+
	30	-	-	-
Quartic	10	E	E	+
	30	+	+	+
Rastrigin	10	E	E	E
	30	-	E	-
Ackley	10	-	-	E
	30	-	-	-
Beale	2	E	E	E
Bohachevsky1	2	E	E	E
Bohachevsky2	2	E	E	E
Bohachevsky3	2	E	E	E
Booth	2	E	E	E
Branin	2	-	E	E
Colville	4	-	-	-
Easom	2	E	-	E
G&P	2	E	-	-
Hartmann1	3	E	+	E
Hartmann2	6	-	+	-
Matyas	2	E	E	E
Michalewick1	2	E	+	E
Michalewick2	5	-	-	-
Michalewick3	9	-	-	-
Shekel1	4	-	+	-
Shekel2	4	E	+	E
Shekel3	4	E	+	+
Trid6	6	E	E	E
Stepint	2	E	E	E
Foxholes	2	E	E	E
+/-/E		1/12/20	7/9/17	5/8/20

Çizelge 3.5. Karma test problemlerinde istatistiksel anlamlılık

Fonksiyon	D	EMA - ABC	EMA - PSO	EMA - DGA
F1	10	E	E	E
	30	-	-	-
	50	-	-	-
F2	10	+	-	-
	30	+	-	+
	50	+	-	-
F3	10	+	+	-
	30	E	E	E
	50	-	-	-
F4	10	-	-	-
	30	+	E	+
	50	E	-	E
F5	10	+	E	E
	30	+	E	-
	50	+	-	-
F6	10	E	E	E
	30	-	+	+
	50	-	E	-
F7	10	-	-	-
	30	-	-	-
	50	-	-	E
F8	10	-	-	-
	30	-	+	-
	50	-	E	-
F10	10	+	-	+
	30	-	-	-
	50	-	-	+
F11	10	-	-	+
	30	E	E	+
	50	-	-	-
F12	10	-	-	-
	30	-	-	-
	50	-	+	+
F13	10	-	-	-
	30	-	+	+
	50	-	-	-
F14	10	-	-	E
	30	-	-	-
	50	+	+	+
F15	10	-	E	-
	30	-	+	-
	50	-	E	-

Çizelge 3.5. Karma test problemlerinde istatistiksel anlamlılık (devam)

Fonksiyon	D	EMA - ABC	EMA - PSO	EMA - DGA
F16	10	E	-	-
	30	E	-	E
	50	E	E	-
F17	10	+	-	E
	30	-	E	E
	50	E	-	-
F18	10	E	E	-
	30	E	E	E
	50	-	-	-
F19	10	-	E	-
	30	-	-	-
	50	-	-	-
F20	10	E	E	-
	30	-	-	-
	50	-	-	-
F21	10	-	E	E
	30	-	E	-
	50	E	-	E
F22	10	E	E	E
	30	+	-	-
	50	-	-	-
F23	10	-	E	-
	30	-	+	-
	50	-	-	-
F24	10	-	E	-
	30	-	-	+
	50	-	-	-
F25	10	-	E	E
	30	-	-	-
	50	-	-	-
+/-/E		12/46/14	8/41/23	11/46/15

Çizelge 3.6. Standart test problemleri için süre ölçümleri (En iyiden sapma oranı)

Fonksiyon İsmi	D	EMA	ABC	PSO	DGA
Sphere	10	0,00	0,00	0,00	0,00
	30	0,08	0,00	0,00	0,00
Rosenbrock	10	0,00	0,00	0,00	0,00
	30	0,09	0,00	0,05	0,00
Griewank	10	0,04	0,00	0,04	0,00
	30	0,12	0,06	0,00	0,02
Quartic	10	0,04	0,02	0,01	0,00
	30	0,09	0,04	0,05	0,00
Rastrigin	10	0,02	0,00	0,00	0,00
	30	0,09	0,00	0,00	0,03
Ackley	10	0,04	0,00	0,00	0,00
	30	0,09	0,04	0,03	0,00
Beale	2	0,00	0,00	0,00	0,00
Bohachevsky1	2	0,00	0,00	0,00	0,00
Bohachevsky2	2	0,00	0,00	0,00	0,00
Bohachevsky3	2	0,00	0,00	0,00	0,00
Booth	2	0,00	0,00	0,00	0,00
Branin	2	0,00	0,00	0,00	0,00
Colville	4	0,00	0,00	0,00	0,00
Easom	2	0,00	0,00	0,00	0,00
G&P	2	0,00	0,00	0,00	0,00
Hartmann1	3	0,00	0,00	0,00	0,00
Hartmann2	6	0,00	0,00	0,00	0,00
Matyas	2	0,00	0,00	0,00	0,00
Michalewick1	2	0,00	0,00	0,00	0,00
Michalewick2	5	0,03	0,00	0,00	0,00
Michalewick3	9	0,05	0,02	0,02	0,00
Shekel1	4	0,00	0,00	0,00	0,00
Shekel2	4	0,03	0,04	0,00	0,00
Shekel3	4	0,04	0,00	0,00	0,04
Trid6	6	0,04	0,00	0,00	0,03
Stepint	2	0,00	0,00	0,00	0,00
Foxholes	2	0,00	0,00	0,00	0,00
Ortalama		0,0278	0,0066	0,0060	0,0036
Normalize		7,72	1,83	1,67	1,00

Çizelge 3.7. Karma test problemleri için süre ölçümleri (En iyiden sapma oranı)

Fonksiyon	D	EMA	ABC	PSO	DGA
F1	10	0,04	0,03	0,00	0,00
	30	0,07	0,00	0,06	0,00
	50	0,09	0,01	0,03	0,00
F2	10	0,09	0,12	0,00	0,00
	30	0,08	0,03	0,00	0,02
	50	0,19	0,06	0,02	0,00
F3	10	0,03	0,00	0,01	0,00
	30	0,09	0,10	0,00	0,09
	50	0,21	0,04	0,09	0,00
F4	10	0,11	0,13	0,00	0,04
	30	0,06	0,03	0,03	0,00
	50	0,22	0,02	0,00	0,00
F5	10	0,06	0,07	0,12	0,06
	30	0,08	0,07	0,00	0,00
	50	0,24	0,06	0,00	0,03
F6	10	0,05	0,02	0,05	0,00
	30	0,04	0,03	0,04	0,00
	50	0,11	0,01	0,00	0,00
F7	10	0,03	0,05	0,07	0,00
	30	0,08	0,13	0,05	0,00
	50	0,17	0,09	0,00	0,06
F8	10	0,13	0,07	0,08	0,00
	30	0,06	0,03	0,00	0,00
	50	0,16	0,03	0,03	0,01
F10	10	0,11	0,00	0,12	0,00
	30	0,03	0,03	0,00	0,00
	50	0,19	0,02	0,02	0,00
F11	10	0,08	0,00	0,04	0,01
	30	0,05	0,00	0,00	0,01
	50	0,20	0,02	0,00	0,04
F12	10	0,02	0,00	0,00	0,00
	30	0,03	0,06	0,03	0,00
	50	0,22	0,17	0,00	0,09
F13	10	0,05	0,00	0,03	0,09
	30	0,02	0,02	0,00	0,00
	50	0,14	0,09	0,00	0,08
F14	10	0,09	0,00	0,00	0,03
	30	0,08	0,06	0,03	0,00
	50	0,16	0,15	0,09	0,00
F15	10	0,03	0,01	0,00	0,01
	30	0,11	0,12	0,09	0,00
	50	0,15	0,06	0,00	0,01

Çizelge 3.7. Karma test problemleri için süre ölçümleri (En iyiden sapma oranı)
(devam)

Fonksiyon	D	EMA	ABC	PSO	DGA
F16	10	0,12	0,00	0,09	0,03
	30	0,04	0,03	0,00	0,00
	50	0,08	0,07	0,12	0,00
F17	10	0,01	0,00	0,00	0,00
	30	0,02	0,00	0,01	0,00
	50	0,27	0,15	0,00	0,08
F18	10	0,02	0,00	0,00	0,03
	30	0,07	0,02	0,06	0,00
	50	0,23	0,10	0,00	0,00
F19	10	0,07	0,00	0,03	0,00
	30	0,06	0,10	0,00	0,04
	50	0,19	0,09	0,05	0,00
F20	10	0,02	0,00	0,00	0,00
	30	0,04	0,02	0,02	0,00
	50	0,18	0,01	0,01	0,00
F21	10	0,05	0,04	0,00	0,03
	30	0,07	0,05	0,06	0,00
	50	0,18	0,12	0,00	0,00
F22	10	0,06	0,00	0,07	0,00
	30	0,08	0,04	0,00	0,05
	50	0,16	0,09	0,00	0,00
F23	10	0,03	0,00	0,03	0,00
	30	0,02	0,03	0,00	0,00
	50	0,29	0,10	0,07	0,00
F24	10	0,01	0,01	0,01	0,00
	30	0,07	0,05	0,00	0,05
	50	0,22	0,14	0,00	0,09
F25	10	0,04	0,00	0,05	0,00
	30	0,05	0,04	0,00	0,04
	50	0,13	0,11	0,03	0,00
Ortalama		0,0990	0,0479	0,0256	0,0156
Normalize		6,34	3,07	1,64	1,00

3.5. Test Sonuçlarının Analizi

Bölüm 3.4 kapsamında elde edilen sonuçlar aşağıdaki gibi özetlenmiş ve yorumlanmıştır.

- Küçük boyutlu ve görece daha kolay problemler üzerinde algoritmaların ürettiği sonuçlar arasında istatistiksel olarak anlamlı bir fark gözlemlenmemiştir.
- EMA özellikle karmaşık ve büyük boyutlu problemlerde çözüm değeri açısından PSO ve ABC'nin gerisinde kalmıştır. Bu noktada EMA'nın hem yeni çözüm alanlarını keşfetme, hem de bulunan çözümleri sömürme alanlarında geliştirilmesi gerekmektedir.
- EMA'nın temel avantajlarından birisi diğer algoritmalara kıyasla daha az kontrol parametresi içermesidir. Bu EMA'nın kolayca uygulanabilmesini sağlamaktadır.
- EMA ile DGA standart test problemlerinde çözüm kalitesi bazında genel olarak birbirlerine yakın performans göstermişlerdir.
- EMA'nın temel zayıflıklarından birisi çözüm süresidir. Tüm sonuçlar birlikte değerlendirildiğinde, EMA'nın diğer tüm algoritmalarından daha yavaş çalıştığı gözlemlenmiştir. Bunu temel sebebi, elektrik yükü hesaplama adımı ve bileşke kuvvet hesaplama adımlarındaki döngüler olduğu tahmin edilmektedir.
- ABC ve PSO yapısının üstün yönleri (ABC'deki yeniden oluşturma adımı, PSO yapısındaki global en iyi çözüm bilgisini kullanma, vb.) EMA'ya uyarlanabilir olduğu ve EMA'nın bu yönlerde geliştirilebileceği görülmüştür.

4. ELEKTROMANYETİK ALGORİTMA İÇİN İYİLEŞTİRME ÖNERİLERİ

Tez çalışmasının bu bölümünde, 3. bölümde detaylı bir şekilde analiz edilen ve diğer meta sezgisel yöntemler ile karşılaştırılan EMA için iyileştirme önerileri verilmiş ve önerilerin etkileri ve optimum seviyeleri istatistiksel olarak tespit edilmiştir. Bölüm 3.5'teki yorumlar incelendiğinde, EMA'nın sömürme/keşfetme dengesini diğer algoritmalara kadar iyi kuramadığı, özellikle sömürme konusunda yetersiz kaldığı görülmüştür. Ek olarak, çok-modlu problemlerde çok kolay yerel optimumlara takıldığı bu sebeple keşfetme yetisinin zayıf kaldığı görülmektedir. İşlem süresi karşılaştırmaları ve yorumları incelendiğinde ise EMA'nın işlem yükünün fazla olduğu ve işlem süresi açısından da diğer algoritmalara göre geride kaldığı anlaşılmaktadır.

Bu bölümde, EMA'nın hem keşfetme, hem de sömürme yetisini artırıcı ve aynı zamanda aralarındaki dengeyi koruyarak işlem yükünü azaltıcı öneriler sunulmuş ve farklı seviyelerde oluşturulan bu öneriler, birer faktör olarak ele alınıp, oluşturulan deneysel tasarım çerçevesinde etkileri analiz edilmiştir. EMA için geliştirilen iyileştirme önerilerinin çoğunda diğer algoritmaların EMA'ya üstünlük sağladığı alanlardan ve operatörlerden esinlenilmiştir. Bölüm 2.2'de detaylı olarak açıklandığı gibi EMA mekanizmasının üç temel operatörü vardır; (i) yük hesaplaması, (ii) kuvvet hesaplaması ve (iii) parçacıkların hareket ettirilmesi. Çalışma kapsamında bu üç operatör için farklı seviyelerde yeni öneriler getirilmiştir. Aynı zamanda, ABC'nin keşfetme yetisine olumlu yönde etkisi olan, yeniden oluşturma adımının EMA'ya uyarlanması test edilmiştir. Ek olarak, EMA'nın tek kontrol parametresi olan popülasyon büyüklüğünün seviyeleri ile ilgili analizler yapılmıştır. Geliştirilen deney tasarımı kapsamında son olarak, başlangıç popülasyonunun başka bir yöntem ile oluşturulması analiz edilmiştir.

4.1. İyileştirme Önerileri

Yük (q) hesaplama denklemi

Temel EMA mekanizmasında her parçacık bir çözümleri gösterir ve parçacığın üzerindeki yük o parçacığın çözüm kalitesi ile doğru orantılıdır. Başka bir deyişle parçacığın gösterdiği çözüm ne kadar iyi ise yükü de o kadar fazla olacaktır. Bir i parçacığının

taşıdığı q_i elektrik yükünün hesabında, parçacığın temsil ettiği \mathbf{x}_i çözümünün kalitesi esas alınır. Buna göre parçacığın elektrik yükü;

$$q_i = \exp\left(-D \frac{f(\mathbf{x}_i) - f(\mathbf{x}_{best})}{\sum_{j=1}^m (f(\mathbf{x}_j) - f(\mathbf{x}_{best}))}\right). \quad \forall i. \quad (4.1)$$

ifadesi ile hesaplanır. Denklem 4.1’de; m popülasyon büyüklüğünü, D problem boyutunu (değişken sayısı), q_i i . parçacığın yükü ve $f(\mathbf{x}_i)$, $f(\mathbf{x}_{best})$ ve $f(\mathbf{x}_k)$ ise sırasıyla; i . parçacığın çözüm değerini, popülasyonun en iyi çözüm değerini ve k . parçacığın çözüm değerlerini göstermektedir.

Çalışma kapsamında, elektrik yükü hesaplama adımında gereksiz işlem yükünden kurtulmak için yeni bir hesaplama yöntemi önerilmiştir. Önerilen yöntemde, temel mekanizmada olduğu gibi parçacığın gösterdiği çözümün, içinde bulunduğu popülasyondaki diğer parçacıklara göre göreceli kalitesini gösterdiği alternatif bir hesaplama yöntemi geliştirilmiştir. Önerilen parçacık yükü hesabı Denklem 4.2’de verilmiştir.

$$q_i = \exp\left(-D \frac{f(\mathbf{x}_i) - f(\mathbf{x}_{best})}{f(\mathbf{x}_{worst}) - f(\mathbf{x}_{best})}\right). \quad \forall i. \quad (4.2)$$

Denklem 4.2’de, $f(\mathbf{x}_{worst})$, o an, popülasyondaki en kötü çözümü göstermektedir. Denklem 4.2 kullanılarak, orijinal yöntemde göre elektrik yükünün anlamı ve değeri değiştirilmeden, işlem yükünde azalmaya gidilmesi planlanmıştır. Söz konusu iyileştirme önerisinin etkisini belirlemek için, yük hesaplama yöntemi, deneysel tasarımda bir faktör olarak kabul edilmiş ve mevcut yöntem (bkz. Denklem 4.1) ile önerilen yöntem (bkz. Denklem 4.2), iki farklı seviye olarak tanımlanıp, test edilmiştir.

Başlangıç çözümü oluşturma

Başlangıç popülasyonu, meta sezgisel bir algoritmanın başarısını etkileyen önemli faktörlerden birisidir (Toğan ve Daloglu 2008). Çalışma kapsamında, başlangıç çözümü oluşturma adımı bir faktör olarak alınmış ve bu kapsamda kaotik başlangıç algoritması

kullanılmasının etkisi test edilmiştir. Kaotik başlangıç algoritması, başlangıç çözümündeki çeşitliliği arttıran bir algoritmadır (Gao ve Liu 2011). Kaotik başlangıç algoritması ile EMA'nın geniş çözüm alanlarında tarama yapması ve keşfetme yetisinin artırılması sağlanmaya çalışılacaktır. Önerilen kaotik algoritmada kullanılan lojistik denklemi Denklem 4.3'de verilmiştir (Alatas 2010).

$$x_{n+1} = \mu \cdot x_n(1 - x_n), \quad 0 \leq x_0 \leq 1 \quad (4.3)$$

Burada, μ bir kontrol parametresi, x_n kaotik değişken ve $n = 0,1,2, \dots$. Genellikle, μ değerinin 4 alındığı ve x_0 değerinin (0, 0,25, 0,50, 0,75, 1) kümesi dışından seçildiği durumlarda, kaotik dinamiklerin daha etkin görüldüğü raporlanmıştır (Alatas 2010, Coelho ve Mariani 2008).

Kullanılan kaotik başlangıç algoritması Algoritma 4.1'de verilmiştir (Algoritmada; D : problem boyutu, m : popülasyon büyüklüğü, (x_j^{min}, x_j^{max}) : değişken sınırları).

```

//Kaotik değişkenleri rassal olarak üret.
for i=1 to m do
    cxi,1 = rand (0,1)
    while cxi,1 not in (0, 0,25, 0,50, 0,75, 1) do
        cxi,1 = rand(0,1)
    end while
end for
//Yinelemeli kaotik başlangıç algoritması.
for i=1 to m do
    for j=1 to D do
        xi,j = xjmin + cxi,j * (xjmax - xjmin)
        cxi,j+1 = 4 * cxi,j * (1 - cxi,j)
    end for
end for

```

Algoritma 4.1. Kaotik başlangıç algoritması

Başlangıç çözümlerinin oluşturulma yöntemi bir faktör olarak tanımlanmış ve rassal başlangıç popülasyonu oluşturma yöntemi, kaotik algoritma ile birlikte iki farklı seviye olmak üzere analiz edilmek üzere deneysel tasarıma dahil edilmiştir.

Yeniden oluşturma

Bölüm 3'te verilen analizler incelendiğinde, EMA'nın temel problemlerinden bir tanesinin yerel optimum noktalara kolayca takılması olduğu görülmektedir. Bu noktada, ABC yapısında kullanılan ve son yıllarda oldukça popülerleşen yeniden oluşturma adımı kullanılması önerilmiştir (Karaboga ve Basturk 2007). Yeniden oluşturma adımı kapsamında, belirli bir yineleme adedi sonrası geliştirilememiş parçacıklar yok edilerek, problemin alt ve üst sınırları içerisinde kalmak üzere rassal olarak yeniden oluşturulurlar. Yapılan ön deneyler sonrası, önerilen yeniden oluşturma noktası ($m * D$) yineleme sonrası olarak tanımlanmıştır. Burada, m popülasyon büyüklüğü, D ise problem boyutudur. Çalışma kapsamında, yeniden oluşturma bir faktör olarak alınmış ve "Var-Yok" şeklinde iki seviye olarak deneysel tasarıma dahil edilmiştir.

Kuvvet (F) hesabı

Temel EMA mekanizmasında, bir parçacığın üzerindeki kuvvet, popülasyondaki tüm bireyler kullanılarak Denklem 4.4'teki gibi hesaplanmaktadır.

$$\mathbf{F}_i = \sum_{j=1, j \neq i}^m \begin{cases} (\mathbf{x}_j - \mathbf{x}_i) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2} \cdot \text{eğer } f(\mathbf{x}_j) < f(\mathbf{x}_i) \\ (\mathbf{x}_i - \mathbf{x}_j) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2} \cdot \text{eğer } f(\mathbf{x}_j) \geq f(\mathbf{x}_i) \end{cases} \quad \forall i. \quad (4.4)$$

Denklem 4.4'de m popülasyon büyüklüğünü göstermektedir. Denklem 4'de görüldüğü gibi \mathbf{F}_i , parçacıklar arasındaki mesafe ile ters orantılı ve parçacıkların yükleri çarpımları ile doğru orantılıdır. Parçacık yükleri de, parçacıkların gösterdiği çözümler ile doğru orantılı olduğundan $f(x_i)$ ile $f(x_j)$ arasındaki fark ne kadar büyük olursa parçacık i 'ye o kadar büyük bir bileşke kuvvet (\mathbf{F}_i) uygulanır.

Önerilen yeni kuvvet hesabı için diğer temel algoritmalarda bilginin hangi bireyler kullanılarak yeni nesillere aktarıldığı incelenmiştir. Bilginin bireyler arasındaki paylaşımı algoritmanın keşfetme/sömürme dengesini etkileyen en önemli faktördür. ABC'de bir bireyin kendisi ile rassal seçilen başka bir bireydeki bilgi kullanılarak

güncelleme yapılırken (bkz. Denklem 3.2), PSO algoritmasında ise o ana kadar bulunan en iyi çözüm, bireyin geçmişindeki en iyi çözümü ve bireyin o anki pozisyonu kullanılarak güncelleme yapılır (bkz. Denklem 3.6). DGA'de ise rassal olarak seçilen üç bireydeki bilgi kullanılarak güncelleme gerçekleştirilir (bkz. Denklem 3.9). Tüm bu bilgilerin ışığında, EMA yapısının bileşke kuvvet hesaplama adımında kullanılan parçacık sayısının ve niteliğinin değiştirilmesi düşünülmüştür. Bu geliştirme önerisi ile algoritmadaki işlem yükü düşürülürken, çözümün global en iyiye yakınsama hızının artırılması öngörülmektedir. Bu noktada iki farklı strateji üzerinde çalışılmış ve geleneksel kuvvet hesaplama yöntemine karşın iki yöntem önerilmiştir:

- Rassal olarak seçilen k adet parçacık bilgisinin kullanılması: Bu yöntemde, her yinelemede rassal olarak seçilen k adet parçacığın oluşturduğu S kümesi kullanılarak bileşke kuvvet hesaplanır. Yapılan ön değerlendirmeler ile k değerinin popülasyon büyüklüğünün %10'nuna eşit olmasına karar verilmiştir ($k = m * 0.1$). Önerilen yeni kuvvet hesabı Denklem 4.5'de verilmiştir.

$$\mathbf{F}_i = \sum_{j \in S} \begin{cases} (\mathbf{x}_j - \mathbf{x}_i) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2} & \text{eğer } f(\mathbf{x}_j) < f(\mathbf{x}_i) \\ (\mathbf{x}_i - \mathbf{x}_j) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2} & \text{eğer } f(\mathbf{x}_j) \geq f(\mathbf{x}_i) \end{cases} \forall i. \quad (4.5)$$

- Seçilen parçacık sayısının adaptif olarak azaltılması: Önerilen bu yöntemde, her yinelemede en iyi k adet parçacık kullanılarak bileşke kuvvet hesaplanır. Denklemde V kümesi en iyi k adet parçacığın indislerinden oluşur. Burada, k doğrusal olarak koşum sonunda 1'e kadar azaltılacaktır. Algoritma çalıştırılmadan önce durdurma kriteri ve popülasyon büyüklüğü bilindiğinden, son yinelemede bir adet (en iyi) parçacığın bilgisi kullanılacak şekilde popülasyon azaltma denklemi oluşturulacak ve bileşke kuvvet sözü konusu parçacık adedi kullanılarak hesaplanacaktır. Önerilen kuvvet hesaplama yöntemi Denklem 4.6'de verilmiştir.

$$\mathbf{F}_i = \sum_{j \in V} \begin{cases} (\mathbf{x}_j - \mathbf{x}_i) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2} & \text{eğer } f(\mathbf{x}_j) < f(\mathbf{x}_i) \\ (\mathbf{x}_i - \mathbf{x}_j) \frac{q_i q_j}{\|\mathbf{x}_j - \mathbf{x}_i\|^2} & \text{eğer } f(\mathbf{x}_j) \geq f(\mathbf{x}_i) \end{cases} \forall i. \quad (4.6)$$

Yer deęiřtirme

Temel EMA mekanizmasında, bir parçacığın üzerine etki edecek bileřke kuvvet hesaplandıktan sonra, parçacık bileřke kuvvetin doęrultusunda Denklem 4.7’de gösterildięi gibi hareket ettirilir. Denklem 4.7’de λ daha önce gidilmeyen noktalara da gidilmesini saęlayan 0–1 arası rassal bir deęiřkendir ($\lambda \in \mathbb{R}: 0 \leq \lambda \leq 1$). *RNG* ise bir vektördür ve bileřenleri. \mathbf{x}_i pozisyon vektörünün her k . bileřeni için belirlenen üst veya alt sınırlar içerisinde yapılabilecek hareketi sınırlar. Denklem 4.7 ve Denklem 4.8 bu sınırlamayı özetlemektedir.

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \lambda \frac{\mathbf{F}_i}{\|\mathbf{F}_i\|} \text{ (RNG)} \quad (4.7)$$

$$x_i^k \in \mathbb{R}: U^k \leq x_i^k \leq L^k. \quad k = 1. \dots n. \quad (4.8)$$

Yapılan alıřmalarda, PSO yapısına benzer bir řekilde, kuvvet vektörüne bir hız vektörü gibi davranılması önerilmektedir (Kennedy ve Eberhart 1995). Böylece, sadece o anki popülasyon bireyelerinin etkisi deęil, farklı bireyelerin de etkisi pozisyon güncelleme adımına dahil edilebilecektir. Bu kapsamda, geleneksel yöntem yerine iki farklı pozisyon güncelleme adımı önerilmiřtir. İlk yöntem, hem popülasyon ortalamasının bilgisini (\mathbf{x}_{mean}), hem de en iyi parçacığın bilgisini (\mathbf{x}_{best}) kullanarak pozisyon güncellemesi yapmaktadır (bkz. Denklem 4.9). Denklem 4.9’da β deęeri (0, 1) arasında rassal bir deęiřkendir. Önerilen ikinci yöntemde (bkz. Denklem 4.10) ise sadece bileřke kuvvet ile birlikte o ana kadar bulunmuř en iyi bireyin de bilgisi kullanılmaktadır.

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \lambda \frac{\mathbf{F}_i}{\|\mathbf{F}_i\|} (\mathbf{x}_i - \mathbf{x}_{mean}) + \beta \frac{\mathbf{F}_{best}}{\|\mathbf{F}_{best}\|} (\mathbf{x}_i - \mathbf{x}_{best}) \quad (4.9)$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \lambda \frac{\mathbf{F}_i}{\|\mathbf{F}_i\|} (\mathbf{x}_i - \mathbf{x}_{best}) \quad (4.10)$$

Geliřtirilen iyileřtirme önerilerinde (bkz. Denklem 4.9 ve Denklem 4.10), temel EMA yapısında olduęu gibi (bkz. Denklem 4.7) *RNG* vektörü kullanılmamıřtır. Bu sebeple,

parçacıkların değerlerini sınırlara göre kontrol eden bir yapı algoritmaya entegre edilmiştir. Söz konusu yapı kapsamında, her yinelemede, parçacıkların değerleri kontrol edilecek ve sınırları aşan değerler ait oldukları sınıra eşitlenecektir.

Popülasyon büyüklüğü

Diğer meta sezgisel algoritmalarda olduğu gibi EMA yapısında da popülasyon büyüklüğünün hem çözüm kalitesine, hem de çözüm hızına etkisi vardır. Bu sebeple, deneysel tasarım kapsamında popülasyon büyüklüğü için üç farklı seviye belirlenmiştir. Popülasyon büyüklüğü 25, 50 ve 75 seviyelerinde test edilmek üzere deneysel tasarım kapsamında incelenmesine karar verilmiştir.

4.2. Deneysel Tasarım

Deneysel tasarım (DT) yönteminin kökleri 1920'lere kadar dayanmaktadır. DT ilk olarak Box ve Wilson (1951) tarafından tarımsal bir deney probleminin etkin bir şekilde çözümü için geliştirilmiş ve uygulanmıştır. Bu çalışmalar esnasında, DT'nin temeli olan varyans analizi (Analysis of Variance) (ANOVA) yöntemi ortaya konmuştur. Günümüzde, DT yöntemi kalite uygulamalarından, temel bilimlere, tıp çalışmalarından mühendislik uygulamalarına kadar birçok bilim dalında yaygın bir şekilde kullanılmaktadır. DT'nin temel amacı bir sürecin veya sistemin girdilerinin farklı seviyelerinin (değişkenliğinin), çıktılar üzerindeki etkilerinin analiz edilmesidir (Hicks 1964).

DT'nin başlıca üç prensibi vardır: (i) tekrarlama (replication), (ii) rasgeleleştirme (randomization) ve (iii) bloklama (blocking) (Coleman ve Montgomery 1993). Tekrarlama aynı koşumun, aynı deney noktasında belirlenen tekrar adedi kadar yinelenmesidir. Tekrarlamanın önemi, deneysel hatanın ortaya çıkarılmasıdır. Rasgeleleştirme, deneysel materyal ve yapılacak her bir deneyin sırasının rastgele dağılım ile belirlenmesidir. Bloklama ilgilenilen etkenler arasında karşılaştırma yaparak hassasiyeti arttıran bir tasarım tekniğidir. Diğer etkenlerden gelen değişkenliği azaltır ya da yok eder.

DT çalışmasının temel adımları aşağıda verilmiştir (Coleman ve Montgomery 1993).

- Problemin tanımlanması,
- Girdiler (faktörler) ve seviyelerinin tanımlanması,
- Çıktıların tanımlanması,
- Deney tasarımının seçilmesi,
- Deneilerin gerçekleştirilmesi,
- Sonuçların incelenmesi, analiz ve yorumlar.

DT'nin diğer yöntemlere göre en belirgin üstünlüğü faktör seviyelerinin aynı anda değiştirilmesiyle oluşturulan deney noktalarında deneylerin yapılması ve faktörler arasında olası ikili ve/veya daha çok etkileşimlerin de analiz edilebilmesidir (Montgomery 2008). DT kapsamında birçok farklı tasarım yöntemi bulunsa da, söz konusu tez çalışması kapsamında sadece tam ve kısmi faktöriyel tasarımlardan bahsedilmiştir.

Birden çok faktör ve seviyeleri ile gerçekleştirilen tasarım genelde faktöriyel tasarım olarak geçer ve n^k şeklinde ifade edilir. Burada k faktör adedi, n ise her faktörün sahip olduğu seviye adedidir. Örnek olarak 2^5 tasarımda, 2 seviyeli beş adet faktör bulunur. Her faktör aynı seviye adedine sahip olacağı gibi, karışık faktöriyel tasarımlarda her faktör farklı seviyelere sahip olabilir ($2^1 \times 3^2$ tasarımda bir adet 2 seviyeli, iki adet 3 seviyeli faktör vardır). Tam faktöriyel tasarımlarında mevcut olan tüm deney noktalarında analizler yapılmaktadır. Herhangi bir tam faktöriyel deney tasarımda, toplam deney adedi, $N = \prod_{i=1}^k n_i, i = 1, \dots, k$, denklemi ile hesaplanır. Burada, k faktör adedi, n ise i . faktöre ait seviye sayısını göstermektedir (Montgomery 2007). Çizelge 4.1'de tasarım için deney noktaları ve her noktada faktörlerin alması gereken değerler örnek olarak gösterilmiştir. Söz konusu tabloda, faktör seviyeleri "+" ve "-" şeklinde kodlanmıştır.

Çizelge 4.1. 2^4 tam faktöriyel tasarım için örnek deneyler

Deney No	Faktör 1	Faktör 2	Faktör 3	Faktör 4
1	-	-	-	-
2	+	-	-	-
3	-	+	-	-
4	+	+	-	-
5	-	-	+	-
6	+	-	+	-
7	-	+	+	-
8	+	+	+	-
9	-	-	-	+
10	+	-	-	+
11	-	+	-	+
12	+	+	-	+
13	-	-	+	+
14	+	-	+	+
15	-	+	+	+
16	+	+	+	+

Yapılacak tekrarlarla birlikte deney adetlerinin çok fazla oluşu, deneylerin maliyetinin yüksek oluşu veya uzun sürmesi gibi kısıtlardan dolayı, gerçek hayat uygulamalarında genelde tam faktöriyel tasarımlar yerine kısmi faktöriyel tasarımlar tercih edilir (Montgomery 2008). Kısmi faktöriyel tasarımlarda bazı çoklu etkileşim etkilerinden taviz verilerek deney adedinin düşürülmesi hedeflenmektedir. Kısmi faktöriyel tasarımlar genelde n^{k-p} şeklinde gösterilir. Burada, n ve p sırasıyla seviye ve faktör adedini gösterirken, p ise diğer parametreler ile örtüştürülerek elde edilen parametre adedini göstermektedir. Örnek vermek gerekirse; 2^{4-1} tasarımında, 16 yerine toplamda 8 deney yapılmakta ve bir parametrenin etkisi diğer üç parametrenin etkileriyle örtüştürülmüş olmaktadır. 2^{4-1} için örnek deney noktaları Çizelge 4.2’de verilmiştir. Çizelge 4.2’deki tasarımda ana etkiler üçlü etkileşimlerle, ikili etkileşimler ise kendi aralarında örtüştürülerek analizler yapılabilir. Böylece ikili ve üçlü etkileri tek başlarına incelemekten taviz verilerek, deney adedi yarı yarıya azaltılmıştır. Bu tip bir deney, ana etkiler ile ilgili birçok bilgiyi sağlayabilir (Montgomery 2008). Çizelge 4.2’deki tasarımda Faktör 4’ün seviyeleri diğer üç faktörün seviyelerinin çapımı (Faktör 1 x Faktör 2 x Faktör 3) ile elde edilmiştir. Daha teknik olarak açıklanırsa, deneyin kimliği I = ABCD ve deneyin çözünürlüğü (Resolution) IV olarak ifade edilir. Kısmi faktöriyel bir tasarımın kimliği ve çözünürlüğü söz konusunu tasarımın kalitesini gösteren

parametrelerdir. Konu hakkında daha detaylı bilgiye Montgomery (2008) çalışmasından erişilebilir.

Çizelge 4.2. 2^{4-1} kısmi faktöriyel tasarım için örnek deneyler (I=ABCD)

Deney No	Faktör 1	Faktör 2	Faktör 3	Faktör 4
1	-	-	-	-
2	+	-	-	+
3	-	+	-	+
4	+	+	-	-
5	-	-	+	+
6	+	-	+	-
7	-	+	+	-
8	+	+	+	+

Farklı faktör seviyelerinde geliştirilen EMA sürümleri MATLAB(R2011a) ortamında kodlanmıştır ve analiz edilmiştir. EMA için geliştirilen iyileştirme önerilerinin etkilerinin test edilmesi için daha önceki bölümde açıklanan farklı boyutlardaki standart ve karma test problemleri kullanılmıştır. Kullanılan test kümesi, tek-modlu, çok-modlu, küçük (10D), orta (30D) ve büyük (50D) problemlerden oluşmaktadır. Bu bölüm kapsamında, her deney noktasında yapılacak tekrar adetlerini de düşünerek, tüm problem setini (bkz. Çizelge 3.2 ve Çizelge 3.3) yansıtan ve 34 problemten oluşan bir alt küme oluşturulmuş ve testler bu alt küme üzerinde gerçekleştirilmiştir. Deneysel tasarım çalışması kapsamında kullanılan test problemleri, boyutları ve özellikleri Çizelge 4.3'te verilmiştir. Çizelge 4.3'te problemlerin numarası ve ismi, deneysel tasarım kapsamında kullanılan problem kodu, problemin boyutu ve tipi (U: Tek-modlu, M: Çok-modlu, S: Ayrıştırılabilir, N: Ayrıştırılamaz) verilmiştir.

Çizelge 4.3. Deneysel tasarım için test problemleri

No	Problem	Problem kodu	Boyut	Tip
1	Beale	Beale	2	UN
2	Colville	Colville	4	UN
3	Hartmann2	Hartmann2	6	MN
4	Michalewick3	Michalewick3	9	MS
5	Sphere	Sphere.10	10	US
6	Sphere	Sphere.30	30	US
7	Sphere	Sphere.50	50	US
8	Rosenbrock	Rosenbrock.10	10	UN
9	Rosenbrock	Rosenbrock.30	30	UN
10	Rosenbrock	Rosenbrock.50	50	UN
11	Griewank	Griewank.10	10	MN
12	Griewank	Griewank.30	30	MN
13	Griewank	Griewank.50	50	MN
14	Ackley	Ackley.10	10	MN
15	Ackley	Ackley.30	30	MN
16	Ackley	Ackley.50	50	MN
17	Kaydırılmış Schwefel fonksiyonu 1.2	F2.10	10	UN
18	Kaydırılmış Schwefel fonksiyonu 1.2	F2.30	30	UN
19	Kaydırılmış Schwefel fonksiyonu 1.2	F2.50	50	UN
20	Kaydırılmış döndürülmüş Rastrigin fonksiyonu	F10.10	10	MN
21	Kaydırılmış döndürülmüş Rastrigin fonksiyonu	F10.30	30	MN
22	Kaydırılmış döndürülmüş Rastrigin fonksiyonu	F10.50	50	MN
23	Karma birleşim fonksiyonu	F15.10	10	MS
24	Karma birleşim fonksiyonu	F15.30	30	MS
25	Karma birleşim fonksiyonu	F15.50	50	MS
26	Ötelenmiş Schewel fonksiyonu 1.2 Gürültülü	F4.10	10	UN
27	Ötelenmiş Schewel fonksiyonu 1.2 Gürültülü	F4.30	30	UN
28	Ötelenmiş Schewel fonksiyonu 1.2 Gürültülü	F4.50	50	UN
29	Global optimumumu sınır üzerinde olduğu döndürülmüş karma birleşim fonksiyonu	F20.10	10	MN
30	Global optimumumu sınır üzerinde olduğu döndürülmüş karma birleşim fonksiyonu	F20.30	30	MN
31	Global optimumumu sınır üzerinde olduğu döndürülmüş karma birleşim fonksiyonu	F20.50	50	MN
32	Sınırlamasız döndürülmüş karma birleşim fonksiyonu	F25.10	10	MN
33	Sınırlamasız döndürülmüş karma birleşim fonksiyonu	F25.30	30	MN
34	Sınırlamasız döndürülmüş karma birleşim fonksiyonu	F25.50	50	MN

Deneysel tasarım ise Statsoft firması tarafından geliştirilen STATISTICA V12.0 yazılımının, 'Design of experiments' modülü kullanılarak hazırlanmış ve uygulanmıştır (www.statsoft.com, 2014). Geliştirilen deneysel tasarım için kullanılan faktör ve seviyeleri Çizelge 4.4'te açıklanmıştır. Bu faktör ve seviyelerine ait detaylı bilgiler ise Bölüm 4.1'de incelenebilir. Çalışma kapsamında, 2-3 seviyeli karışık faktörlerden oluşan tam tasarım yerine, STATISTICA tarafından da önerilen 1/3 oranına azaltılmış tasarım ($2^3.3^{3-1}$) kullanılmıştır (Montgomery 2008). Başka bir deyişle, toplamda 216 deney noktası olan tam faktöriyel tasarım yerine, 72 adet deney noktasının kullanıldığı kısmi faktöriyel tasarım kullanılmıştır. Kullanılan tasarım noktalarındaki faktör seviyeleri Çizelge 4.5'te verilmiştir. Çizelge 4.5'te, iki seviyeli faktörler için (-1, 1) ve üç seviyeli faktörler için ise (-1, 0, 1) gösterimi kullanılmıştır. İstatistiksel olarak anlamlı sonuçlar elde edebilmek için her deney noktasında 30 rassal koşumlu tekrar gerçekleştirilmiş ve bulunan amaç fonksiyonu değerleri raporlanmıştır. Başka bir deyişle, tez çalışmasında gerçekleştirilen deneysel tasarım kapsamında 34 problem üzerinde, 30 bağımsız tekrar ile 72 deney noktasında toplamda 73 440 bağımsız tekil koşum gerçekleştirilmiştir.

Bölüm 1'de açıklandığı gibi meta sezgisel algoritmaların başarısı sadece bulunan çözümlerin kalitesi ile değil, çözümlerin bulunması için harcanan süre veya işlem adedi de ölçülmeli ve karşılaştırılmalıdır. Bu sebeple, geliştirilen faktör ve seviyelerinin çözüm süresine etkilerinin de incelenebilmesi için çözüm sürelerinde kısıtlamaya gidilmiş ve ön testler ile belirlenen maksimum işlem süreleri tanımlanmıştır. Çalışma kapsamında, küçük test problemleri için 5 saniye, orta test problemleri için 10 saniye ve büyük test problemleri için ise 15 saniye işlem süresi kullanılmıştır. Her problem için gerçekleştirilen yineleme adedine bakılmaksızın, bu süreler sonunda koşum durdurulmuş ve bulunan amaç fonksiyonu değerleri raporlanmıştır.

Çizelge 4.4 Faktörler ve faktör seviyeleri

No	Faktör	Faktör Kodu	Seviye 1	Seviye 2	Seviye 3
1	Yük (q) hesabı	Qcalc	Geleneksel	Denklem 4.2	-
2	Başlangıç çözümü oluşturma	Chaotic	Geleneksel	Kaotik (bkz. Algoritma 4.1)	-
3	Yeniden oluşturma	Regen	Yok	Var	-
4	Kuvvet (F) hesabı	Fcalc	Geleneksel	Adaptif k adet (bkz. Denklem 4.6)	Sabit k adet (bkz. Denklem 4.5)
5	Yer değiştirme	Repos	Geleneksel	Best (bkz. Denklem 4.10)	Mean + Best (bkz. Denklem 4.9)
6	Popülasyon büyüklüğü	Popsize	25	50	75

Çizelge 4.5. Uygulanan deneysel tasarımdaki deney noktaları

Deney No	Qcalc	Chaotic	Regen	Fcalc	Repos	Popsiz
1	-1	-1	-1	-1	-1	-1
2	-1	-1	-1	-1	0	1
3	-1	-1	-1	-1	1	0
4	-1	-1	-1	0	-1	1
5	-1	-1	-1	0	0	0
6	-1	-1	-1	0	1	-1
7	-1	-1	-1	1	-1	0
8	-1	-1	-1	1	0	-1
9	-1	-1	-1	1	1	1
10	-1	-1	1	-1	-1	0
11	-1	-1	1	-1	0	-1
12	-1	-1	1	-1	1	1
13	-1	-1	1	0	-1	-1
14	-1	-1	1	0	0	1
15	-1	-1	1	0	1	0
16	-1	-1	1	1	-1	1
17	-1	-1	1	1	0	0
18	-1	-1	1	1	1	-1
19	-1	1	-1	-1	-1	1
20	-1	1	-1	-1	0	0
21	-1	1	-1	-1	1	-1
22	-1	1	-1	0	-1	0
23	-1	1	-1	0	0	-1
24	-1	1	-1	0	1	1
25	-1	1	-1	1	-1	-1
26	-1	1	-1	1	0	1
27	-1	1	-1	1	1	0
28	-1	1	1	-1	-1	1
29	-1	1	1	-1	0	0
30	-1	1	1	-1	1	-1
31	-1	1	1	0	-1	0
32	-1	1	1	0	0	-1
33	-1	1	1	0	1	1
34	-1	1	1	1	-1	-1
35	-1	1	1	1	0	1
36	-1	1	1	1	1	0
37	1	-1	-1	-1	-1	1
38	1	-1	-1	-1	0	0
39	1	-1	-1	-1	1	-1
40	1	-1	-1	0	-1	0
41	1	-1	-1	0	0	-1
42	1	-1	-1	0	1	1

Çizelge 4.5. Uygulanan deneysel tasarımdaki deney noktaları (devam)

Deney No	Qcalc	Chaotic	Regen	Fcalc	Repos	Popsiz
43	1	-1	-1	1	-1	-1
44	1	-1	-1	1	0	1
45	1	-1	-1	1	1	0
46	1	-1	1	-1	-1	1
47	1	-1	1	-1	0	0
48	1	-1	1	-1	1	-1
49	1	-1	1	0	-1	0
50	1	-1	1	0	0	-1
51	1	-1	1	0	1	1
52	1	-1	1	1	-1	-1
53	1	-1	1	1	0	1
54	1	-1	1	1	1	0
55	1	1	-1	-1	-1	0
56	1	1	-1	-1	0	-1
57	1	1	-1	-1	1	1
58	1	1	-1	0	-1	-1
59	1	1	-1	0	0	1
60	1	1	-1	0	1	0
61	1	1	-1	1	-1	1
62	1	1	-1	1	0	0
63	1	1	-1	1	1	-1
64	1	1	1	-1	-1	-1
65	1	1	1	-1	0	1
66	1	1	1	-1	1	0
67	1	1	1	0	-1	1
68	1	1	1	0	0	0
69	1	1	1	0	1	-1
70	1	1	1	1	-1	0
71	1	1	1	1	0	-1
72	1	1	1	1	1	1

4.3. Deneysel Tasarım Sonuçları

Deneysel tasarım kapsamında, her problem için ANOVA çizelgeleri, tahmini etki değerleri ve tahmini etki grafikleri ve her faktör için ortalama etki grafikleri çizilmiştir (bkz. Şekil Ek 5.1 – 5.108). Tüm bu sonuçlar Ek 5 bölümünde incelenebilir. Faktör seviyelerinin, sadece tek problem değil, birbirinden farklı problemler üzerindeki etkileri hakkında genel değerlendirmeler yapabilmek için tahmini etki değerleri ve bu değerlerin güven aralıkları hesaplanmış ve analiz edilmiştir (bkz. Çizelge 4.6 ve Çizelge 4.7). Buradaki amaç, sadece etkili faktör ve seviyelerinin belirlenmesi ile sınırlı olmayıp,

faktörlerin farklı problemler üzerindeki genel etkilerinin incelenebilmesi ve güven aralıkları yardımı ile istatistiksel olarak anlamlı sonuçların elde edebilmesidir.

Çalışma kapsamında, her deney noktasında, 34 adet minimizasyon problemi üzerinde gerçekleştirilen 30 bağımsız koşum sonucunda bulunun amaç fonksiyonu değerleri deneysel tasarım çalışmasında girdi olarak sunulmuş ve analiz sonucunda elde edilen çıktılar incelenmiştir. Her bir problem için yapılan 72 deney noktalı varyans analizi sonucunda bulunan faktör etkilerinin mutlak değerleri (t istatistiği için $|t|$) analiz edilmiş ve Çizelge 4.6'da verilmiştir. Her faktör için söz konusu t istatistiği değerleri, her problem için oluşturulan tahmini etki değerleri çizelgelerindeki t sütununda veya ortalama etkiler grafiklerindeki bar değerlerinde incelenebilir (bkz. Şekil Ek 5.1 – 5.108). Çizelge 4.6'nın son satırında ise problemlerin tamamına göre değişen faktör etkilerinin mutlak değerlerinin sapmasının %95 güvenirlikle belirlenen güven aralıkları verilmiştir. Problemlerin tamamı için verilen güven aralıklarının, problem tiplerine göre oluşturulan gruplar için ortalama alınarak düzenlenmesi ile elde edilen güven aralığı sonuçları Çizelge 4.7'de verilmiştir. Çizelge 4.6 ve Çizelge 4.7'deki sonuçların yorumlanabilmesi için Şekil Ek 5.1 – 5.108 'de verilen *tahmini etki değerleri çizelgesi*, *ANOVA çizelgesi* ile *ortalama etkiler ve etki grafikleri* birlikte yorumlanmış ve böylece en etkin EMA yapısına karar verilmesine çalışılmıştır. Dikkat edilirse, elde edilen güven aralıklarının anlamlı bir şekilde yorumlanabilmesi için *tahmini etki değerleri çizelgesi* ve *ortalama etkiler grafiklerinde* etki değerleri (t istatistiği değerleri) işaretli olmasına rağmen grafikler Çizelge 4.7'deki gibi işaretlerden arındırılmış, mutlak değerler bazında çizilmiştir. Bu sebeple, istatistiksel olarak anlamlı etkiye sahip bir faktörün etkisinin hangi yönde olduğu t istatistiğinin işareti yerine, ortalama etki grafikleri incelenirken tespit edilmiştir (bkz. Şekil Ek 5.1 – 5.108). Çizelge 4.6 ve Çizelge 4.7'deki herhangi bir faktörün (%95 anlamlılıkta) etkili olabilmesi için *ortalama etkiler* (bkz. Ek-5) şekillerinde kesikli çizgi ile gösterilen eşik değerden ($p=0,05$) daha büyük olması gerekir. Farklı problem tipleri üzerindeki etki değerleri ve değerlerin güven aralıkları da çalışma kapsamında incelenmiş ve yorumlanmıştır. Bu çalışma kapsamında; ANOVA tabloları, tahmini etki değerleri ve güven aralıkları ve ortalama etki grafikleri birlikte değerlendirildiğinde aşağıdaki sonuçlara erişilmiştir:

1. Popülasyon büyüklüğünün beklenildiği gibi problem boyutuna bağlı olarak etkisi değişmektedir. Çizelge 4.7’de popülasyon büyüklüğü (PopSize) etkisinin küçük problemler için 2,682–3,526 ve büyük problemler için 2,618–3,502 güven aralığında olduğu görülmektedir Çizelge 4.6 incelendiğinde, popülasyon büyüklüğü faktörünün tüm problemler üzerindeki ortalama etkisi ise 2,193–2,951 güven aralığındadır. Söz konusu etki değerleri, ortalama etki grafikleri (bkz. Şekil Ek 5.4–5.108) ile birlikte değerlendirildiğinde küçük problemlerde, $PopSize(n) = 25$ (Seviye 1), büyük problemlerde ise $Popsize(n) = 75$ (Seviye 3)’in en etkin sonuçları verdiği görülmektedir. Büyük problemlerde etkinin negatif olmasının sebebinin uygulanan 15 saniye süre kısıtı olduğu değerlendirilmektedir. Küçük problemlerde ise büyük popülasyonlar ile koşum yapılmış olması işlem yükü arttırmış iken, maksimum işlem süresinin 5 saniye ile kısıtlanmış olmasından dolayı, kısıtlı süreler içerisinde iyi sonuçlara erişmek zor hale gelmiştir.
2. Denklem 4.5 ve Denklem 4.6 ile önerilen kuvvet hesaplama yöntemlerinin EMA’nın performansı üzerine olumlu etkisi olduğu söylenebilir. Çizelge 4.7’de kuvvet hesaplama (FCalc) faktörünün tüm problemler üzerindeki etki değerlerinin güven aralıkları incelendiğinde (Doğrusal etki (L): 6,080–11,132; İkinci dereceden etki (Q): 4,164–7,139), söz konusu faktörün etkisinin %95 güvenilirlikte istatistiksel olarak anlamlı olduğu görülmektedir. FCalc faktörü için ortalama etki grafikleri incelendiğinde (bkz. Şekil Ek 5.1 – 5.108) önerilen faktörün Seviye 1’in (bkz. Denklem 4.6) hem geleneksel yönteme (bkz. Denklem 4.4) hem de Seviye 2’ye (bkz. Denklem 4.5) göre daha etkili olduğu görülmektedir. Buna göre Seviye 1 olarak tanımlanan her yinelemede ‘adaptif k adet’ parçacığın kullanılması, Beale, Colville, F4.10 ve F4.50 problemleri dışında diğer tüm test problemlerinde üstün performans elde edilmesini sağlamıştır. Başka bir deyişle, kuvvet hesaplamasında kullanılacak parçacıkların, son yinelemede sadece en iyi parça kalacak şekilde doğrusal olarak azaltılması EMA’nın performansını iyileştirmektedir. Söz konusu yöntem ile her yinelemede bir öncekine göre daha az parçacık kullanıldığından işlem yükü azaltılmakta ve aynı zamanda kuvvet hesaplama işleminde daha kaliteli

parçacıkların kullanılması sağlanarak algoritmanın sömürme yetisi geliştirilmektedir. Çizelge 4.7 incelendiğinde, önerilen yöntemin hem çok modlu (L:7,082–12,633; Q:5,158–8,309) hem de tek modlu (L:5,339–8,232; Q:3,331–4,722) problemlerde ve aynı zamanda da tüm problem boyutlarında (büyük (L:5,978–12,613; Q:3,315–7,456), orta (L:4,530–10,285; Q:3,580–7,433) ve küçük (L:4,949–10,380; Q:3,675–6,764)) istatistiksel olarak etkili olduğu görülmektedir.

3. Sonuçlar incelendiğinde, önerilen yük hesaplama yönteminin (bkz. Denklem 4.2), EMA yapısına istatistiksel olarak anlamlı bir etkisi olmadığı görülmektedir. QCalc etki değerlerinin güven aralıkları incelendiğinde (bkz. Çizelge 4.6), yük hesaplamasının etkisi $p=0,05$ ($t=1,9610$) noktasının altında kalmıştır (0,838–1,756). Bu bağlamda, QCalc faktörünün %95 anlamlılıkta çözüm kalitesine etkisinin olmadığı sonucuna varılabilir. Önerilen yük hesaplama yönteminin özellikle algoritmanın işlem yükünü azaltıcı etkisi olması beklenmişse de, çözüm kalitesine yaptığı negatif etkiden dolayı beklenen iyileştirmenin sağlanamadığı düşünülmektedir. Ek olarak QCalc faktörünün tüm problem tipleri için benzer sonuçlar verdiği görülmektedir (bkz. Çizelge 4.7).
4. Yeniden oluşturma (*Regen*) yönteminin EMA'ya entegre edilmesi, beklenildiği gibi algoritmanın keşfetme yetisini geliştirmiş ve çözüm kalitesine olumlu yönde etkisi olmuştur. Çizelge 4.7 incelendiğinde, yeniden oluşturma özelliği çok modlu problemler (2,949–5,250) ve büyük boyuttaki problemler (2,886–5,664) üzerinde etkisi daha net görülmektedir. *Regen* faktörünün özellikle daha zor tipteki problemlerde daha etkili oluşunun sebebi, geleneksel EMA'ya göre keşfetmeyi geliştirmesi ve bu sebeple yerel optimum noktalara takılma olasılığını azaltması olabilir. Faktörün tüm problem tipleri üzerindeki ortalama etkisinin %95 güvenilirlikte güven aralığı ise 2,743–4,810 olarak görülmektedir (bkz. Çizelge 4.6–4.7). Etki değerlerinin güven aralığı ile ortalama etki grafikleri (bkz. Şekil Ek 5.1–5.108) birlikte incelendiğinde *Regen* faktörü, Seviye 2'de aldığı değer ile (*Regen = Var*) EMA'nın performansını artırıcı

yönde etkide bulunmuştur. Algoritmanın arama uzayının bu şekilde belirli dönemlerde büyütülmesi özellikle zor problemlerin çözümünde daha etkilidir.

5. Başlangıç çözümlerinin kaotik algoritma (bkz. Algoritma 4.1) kullanılarak oluşturulması, çözüm kalitesi üzerinde etkilidir. Çizelge 4.6 ve Çizelge 4.7 incelendiğinde, faktörün ortalama etkisinin güven aralığı (%95 güvenlilikle), tüm problem için 3,902–7,156, çok modlu problemler için 4,565–8,127 tek modlu problemlerde 3,351–5,236, küçük boyuttaki problemlerde 2,879–7,489, orta büyüklükteki problemlerde 3,121–7,351 ve büyük problemlerde ise 3,868–7,123 olarak ölçülmüştür. Ortalama etki grafikleri incelendiğinde ise (bkz. Şekil Ek 5.1–5.108) başlangıç çözümü oluşturma faktörünün Seviye 2’de aldığı değer ile (*Chaotic = Var*), amaç fonksiyonu değerini düşürdüğü ve daha kaliteli sonuçlar elde ettiği görülmektedir.
6. Çizelge 4.6 incelendiğinde, yer değiştirme ifadesinde hem popülasyon ortalamasının (bkz. Denklem 4.9) hem de en iyi parçacığın bilgisinin kullanılmasının (bkz. Denklem 4.10) algoritmanın performansına anlamlı bir etkisi olduğu görülmektedir. Yer değiştirme faktörünün (*Repos*) ortalama etkisinin güven aralıkları incelendiğinde (tüm problemler için L:2,737–4,963; Q:1,875–3,127), faktörün istatistiksel olarak anlamlı bir etkisinin olduğu görülmektedir. Çizelge 4.7 incelendiğinde, faktörün özellikle çok modlu (L:3,275–5,652; Q:2,214–3,271) ve büyük problemlerde (L:2,737–5,623; Q:2,018–3,416) etkili olduğu görülmektedir. Ortalama etki grafikleri ile etki değerleri birlikte değerlendirildiğinde (bkz. Şekil Ek 5.1–5.108) faktörün Seviye 3 değerini aldığı (bkz. Denklem 4.9) diğer seviyelere göre çözüm kalitesi üzerinde daha etkili olduğu görülmektedir. Önerilen Seviye 3 ile tüm test problemlerinde daha üstün performans gözlenmiştir. Seviye 3 ile getirilen öneri temel olarak en iyi parçacığın bilgisinin kullanımına dayandığından, olumlu etki sıklıkla yakınsamanın daha zor gerçekleştiği daha karmaşık problemlerde (çok modlu ve büyük boyutlu) görülmektedir.

Çizelge 4.6. Tüm problem setleri için birinci (L) ve ikinci dereceden (Q) ortalama etki değerleri ve güven aralıkları

Problemler	QCalc(L)	Chaotic(L)	Regen(L)	FCalc(L)	Repos(L)	PopSize(L)	FCalc(Q)	PopSize(Q)	Repos(Q)
Beale	0,5521	4,5736	3,0209	8,1343	5,2126	1,2562	4,5487	1,5846	2,6646
Colville	0,4762	3,9857	4,8155	8,6126	4,7114	1,5280	6,1828	1,9278	3,4880
Hartmann2	1,3011	4,8018	2,2830	5,6152	2,9730	0,4628	5,2608	1,7947	1,8278
Michalewick3	2,6637	14,5139	9,4206	23,4058	12,0394	2,6837	13,9399	4,0218	4,6853
Sphere.10	1,5644	2,3906	3,0845	8,1238	2,5557	1,5935	3,5968	3,5502	2,8414
Sphere.30	1,6694	4,1643	4,3873	4,6532	1,4420	0,9744	4,1046	2,8185	1,9921
Sphere.50	1,4323	5,4406	4,9620	3,4092	5,0063	2,0943	3,3694	4,0322	5,1048
Rosenbrock.10	0,0212	8,0941	4,9593	12,3460	1,1871	0,6692	6,6641	1,9865	2,6059
Rosenbrock.30	0,9237	7,2278	4,7110	10,5847	4,9307	1,0171	5,9155	2,3167	1,7272
Rosenbrock.50	0,3880	6,0948	3,6787	9,9495	4,4862	0,1701	3,9118	1,6999	1,7022
Griewank.10	0,8317	1,6637	1,1852	2,6631	2,0759	1,1274	1,6544	3,7409	0,3832
Griewank.30	0,7350	2,1724	1,1183	2,3934	1,4688	0,5808	1,9398	0,3845	0,7839
Griewank.50	0,7875	9,7687	6,9592	15,5797	8,4512	1,2239	10,1116	4,3236	3,8056
Ackley.10	2,6637	14,5139	9,4206	23,4058	7,0394	2,6837	13,9399	4,0218	4,6853
Ackley.30	0,5955	14,6576	9,1135	21,4001	7,4279	2,1659	13,6991	4,0313	4,9241
Ackley.50	0,2507	10,9564	6,0296	15,6196	8,2128	1,8279	9,4154	3,9641	4,5970
F2.10	1,3504	1,8611	2,2009	4,5658	1,8652	0,5569	3,1246	1,5886	2,1238
F2.30	0,2751	2,2264	2,5798	4,0834	1,2046	1,1857	2,2235	3,5205	1,9601
F2.50	0,7402	2,7995	1,6084	3,3470	1,5830	0,4288	2,9255	3,5570	1,5138
F10.10	1,5213	7,1536	2,4557	9,5716	3,4578	0,8385	7,3607	2,1712	3,0441
F10.30	0,2200	4,6266	2,7757	7,4040	3,0198	0,6040	6,6543	2,4365	1,3227
F10.50	0,0022	3,7667	2,9475	6,7534	2,2567	0,0480	5,1295	2,0191	1,1023
F15.10	0,5521	4,5736	3,0209	8,1343	5,2126	1,2562	4,5487	1,5846	2,6646
F15.30	0,9401	4,1447	3,1639	6,4516	2,5402	1,6691	4,8273	1,4086	0,2383
F15.50	0,9423	3,4700	2,3939	5,0198	1,9090	1,8289	3,5582	2,1438	0,3588
F4.10	0,8582	3,2777	2,1911	6,5499	2,8824	1,8853	2,9858	1,4321	1,8165
F4.30	0,1049	4,3527	1,7952	5,5775	3,1105	1,5172	4,2581	1,8365	1,4163
F4.50	0,1223	3,6170	1,6447	5,0616	2,8257	1,7447	2,5601	2,6386	2,3887

Çizelge 4.6. Tüm problem setleri için birinci (L) ve ikinci dereceden (Q) ortalama etki değerleri ve güven aralıkları (devam)

Problemler	QCalc(L)	Chaotic(L)	Regen(L)	FCalc(L)	Repos(L)	PopSize(L)	FCalc(Q)	PopSize(Q)	Repos(Q)
F20.10	0,6067	5,0320	4,4992	9,4742	3,8695	2,8374	6,3807	1,3162	4,1622
F20.30	0,4762	3,9857	4,8155	8,6126	4,7114	1,5280	6,1828	1,9278	3,4880
F20.50	0,1035	3,9214	3,6835	7,3027	4,1014	1,6918	5,7226	1,2892	3,2668
F25.10	1,3279	3,2766	3,0845	8,1238	2,5557	1,5935	3,5968	1,3741	2,8414
F25.30	1,3011	4,8018	2,2830	5,6152	2,9730	0,4628	5,2608	1,7947	1,8278
F25.50	0,2068	5,1232	1,3429	4,6021	2,9661	0,5457	5,4904	1,9553	1,8406
95% CI	0,538 - 1,156	3,902 - 7,156	2,743 - 4,810	6,080 - 11,132	2,737 - 4,963	2,193 - 2,951	4,164 - 7,139	1,591 - 2,628	1,975 - 3,127

Çizelge 4.7. Problem tipleri ve boyutlarına göre etki değerlerinin değişimi

Problem Tipi	QCalc(L)	Chaotic(L)	Regen(L)	FCalc(L)	Repos(L)	PopSize(L)	FCalc(Q)	PopSize(Q)	Repos(Q)
Tek modlu	0,462 - 1,035	3,351 - 5,236	2,602 - 3,917	5,339 - 8,232	2,302 - 3,841	1,731 - 2,358	3,331 - 4,722	1,367 - 2,417	1,891 - 2,872
Çok modlu	0,585 - 1,218	4,565 - 8,127	2,949 - 5,250	7,082 - 12,633	3,275 - 5,652	1,949 - 3,517	5,158 - 8,309	1,727 - 2,743	2,214 - 3,271
Küçük (10D)	0,701 - 1,558	2,879 - 7,489	2,245 - 4,975	5,978 - 12,613	2,246 - 4,294	2,682 - 3,526	3,315 - 7,456	1,369 - 2,585	1,648 - 3,488
Orta (30 D)	0,431 - 1,017	3,121 - 7,351	2,334 - 5,014	4,530 - 10,825	2,145 - 4,420	0,682 - 1,459	3,580 - 7,433	1,319 - 2,576	1,178 - 2,758
Büyük(50 D)	0,226 - 0,769	3,868 - 7,123	2,886 - 5,664	4,949 - 10,380	2,737 - 5,623	2,618 - 3,502	3,675 - 6,764	1,448 - 3,076	2,018 - 3,416
Genel	0,538 - 1,156	3,902 - 7,156	2,743 - 4,810	6,080 - 11,132	2,737 - 4,963	2,193 - 2,951	4,164 - 7,139	1,591 - 2,628	1,975 - 3,127

4.4. İyileştirilmiş Elektromanyetik Algoritma

Tüm bu sonuçlar ışığında geliştirilen iyileştirilmiş EMA (iEMA) yapısı, faktörler ve optimum seviyeleri ve seviyelerin açıklamaları Çizelge 4.8’de verilmiştir. iEMA’nın algoritma adımları ise aşağıda özetlenmiştir. Adımlarda kullanılan *deneme* değişkeni, ABC yapısındaki *failure* değişkenine (bkz. Bölüm 3.1.1) benzer bir şekilde parçacıkların geliştirilemediği yineleme adetlerini tutan değişkendir. iEMA’nın adımlarında, kaotik algoritma ile başlangıç çözümünün oluşturulması (Satır: 2), kuvvet hesaplama adımında adaptif olarak azaltılan k adet parçacığın kullanılması (Satır: 10–12), geliştirilen yer değiştirme denkleminin kullanılması (Satır: 15–16) ve yeniden oluşturmanın kullanılması (Satır: 18, 21) incelenebilir.

- Adım 1: Popülasyon büyüklüğü (m), durdurma kriteri, *limit* gibi algoritma parametrelerinin atanması.
- Adım 2: Kaotik algoritma kullanılarak başlangıç popülasyonunun oluşturulması (bkz. Algoritma 4.1)
- Adım 3: Algoritmaya ait tutulacak istatistiklerin (global en iyi çözüm değeri ve parçacık değeri, deneme, vb.) ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için amaç fonksiyon değerlerinin ($f(x)$) hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For i = 1 to m Do**
- Adım 7: Elektrik yüklerini hesapla (bkz. Denklem 4.1)
- Adım 8: **End For**
- Adım 9: V kümesini oluştur.
- Adım 10: **For i = 1 to m Do**
- Adım 11: Bileşke kuvveti hesapla (bkz. Denklem 4.6)
- Adım 12: **End For**
- Adım 13: x_{mean} ve x_{best} değerlerini hesapla.
- Adım 14: **For i = 1 to m Do**
- Adım 15: F_{best} hesapla.
- Adım 16: Parçacıkları bileşke kuvvet ile hareket ettir (bkz. Denklem 4.9).
- Adım 17: Yeni amaç fonksiyonlarını ($f(x)$) hesapla.

- Adım 18: Eski ve yeni $f(x)$ değerlerini karşılaştır ve $deneme_i$ değişkenini güncelle.
- Adım 19: Gerekli istatistikleri güncelle.
- Adım 20: **End For**
- Adım 21: Tüm parçacıkları için limit değeri ile $deneme_i$ değerlerini karşılaştır, gerekirse rassal olarak yeniden oluştur (Regen).
- Adım 22: Tüm parçacıkların sınırları kontrol et.
- Adım 23: **Until** (durdurma kriteri sağlanıyorsa)

Çizelge 4.8. İyileştirilmiş EMA yapısı faktörler ve seviyeleri

No	Faktör	Faktör Kodu	Kullanılacak Seviye	Açıklama
1	Yük (q) hesabı	Qcalc	Seviye 1	Yok
2	Başlangıç çözümü oluşturma	Chaotic	Seviye 2	Var
3	Yeniden oluşturma	Regen	Seviye 2	Var
4	Kuvvet (F) hesabı	Fcalc	Seviye 2	Adaptif k adet
5	Yer değiştirme	Repos	Seviye 3	Mean + Best
6	Popülasyon büyüklüğü	PopSize	10D ve 30D için Seviye 1, 50D için Seviye 3	10D ve 30D için 25, 50D için 75

Bölüm 4 kapsamında gerçekleştirilen deneysel tasarım çalışmasının sonuçları aşağıda özetlenmiştir:

- Başlangıç çözümünün kaotik algoritma kullanılarak oluşturulması algoritmanın keşfetme yetisini artırmıştır. Beklenilenin tersine kaotik algoritma kullanmanın çözüm süresine anlamlı bir etki yapmadığı anlaşılmıştır.
- ABC'de kullanılan yapıdan esinlenilerek geliştirilen ve EMA'ya uyarlanan yeniden oluşturma adımı, arama esnasında yeni rassal parçacıkların oluşturulmasına olanak sağladığı için algoritmanın yeni alanları keşfetmesini sağlamış ve yerel optimumlara takılmayı engellemiştir.
- EMA'daki bileşke kuvvet hesaplama adımında, tüm popülasyon yerine adaptif olarak azaltılarak seçilen k adet parçacığın kullanılması, algoritmayı hem hızlandırmış hem de daha iyiye yönelme şansını artırmıştır.

- PSO yapısındakine benzer olarak parçacıkların ortalama etkisini ve global en iyi parçacığın etkisinin parçacıkların hareket ettirilmesin adımımda kullanılmasıyla EMA'nın sömürme yetisinin arttırıldığı görülmektedir.
- Algoritmanın en temel kontrol parametresi olan popülasyon büyüklüğünün problem boyutuna bağlı olarak ayarlanması hem çözüm süresini düşürmüş hem de daha karmaşık problemlerde keşfetmeyi güçlendirmiştir.

Yapılan deneysel tasarım sonucunda, iEMA bünyesinde kullanılmasına karar verilen yöntemlerin algoritmanın keşfetme, sömürme ve çözüm süresi gibi karakteristiklerine etkileri Çizelge 4.9'da verilmiştir.

Çizelge 4.9. Önerilen yöntemler ile sağlanan gelişim alanları

Algoritma Adımı	Önerilen Yöntem	İyileştirme Etkisi
Başlangıç çözümü oluşturma	Kaotik Algoritma ile oluşturma	Keşfetme
Yeniden oluşturma	ABC benzeri yapı kullan	Keşfetme
Kuvvet (F) hesabı	Adaptif k adet parçacık kullan	Sömürme + İşlem yükü
Yer değiştirme	Ortalama etki ve en iyinin etkisini kullan	Sömürme
Popülasyon büyüklüğü	Boyuta göre atama yap	İşlem yükü + keşfetme

5. İYİLEŞTİRİLMİŞ ELEKTROMANYETİK ALGORİTMA İLE KISITSIZ PROBLEMLER ÜZERİNDE TESTLER

5.1. Karşılaştırma Algoritmaları

Tez çalışmasının bu bölümünde, deneysel tasarım sonucu geliştirilen birçok iyileştirme önerisi arasından yapılan tercihler sonrası belirlenen en iyi tasarıma sahip EMA (iEMA) ile daha önce temel sürüm karşılaştırmalarında kullanılan meta sezgisel algoritmaların güncel ve çokça atıf almış iyileştirilmiş sürümleri arasında karşılaştırmalı analizler yapılmıştır. Bu kapsamda, iyileştirilmiş ABC (MABC) (Gao ve Liu 2011), Global eniyi ABC (GBest-ABC) (Zhu ve Kwong 2010), Adaptif PSO (APSO) (Zhan ve ark. 2009) ve Adaptif DGA yöntemleri (SaDE) (Qin ve Suganthan 2005) karşılaştırma algoritmaları olarak kullanılmışlardır. Çizelge 5.1’de söz konusu çalışmaların 2014 yılı sonu itibari ile sahip oldukları atıf adetleri verilmiştir. Analizlerin tarafsız ve tutarlı olabilmesi için temel sürüm testlerinde yapıldığı gibi söz konusu algoritmalar özdeş ortamda kodlanarak eşit şartlarda aynı problem setleri üzerinde çalıştırılmışlardır. Bu kapsamda, testlerde kullanılan iyileştirilmiş meta sezgisel algoritmalar ve içerdikleri temel güncellemeler aşağıda kısaca özetlenmiştir.

Çizelge 5.1. Çalışmaların atıf adetleri (2014 yılı sonu itibari ile)

Referans	Temel Yapı	Geliştirilen Algoritma	Google Scholar Atıf Sayısı	WOS Atıf Sayısı
Gao ve Liu (2011)	ABC	MABC	102	53
Zhu ve Kwong (2010)	ABC	GBest-ABC	251	110
Zhan ve ark. (2009)	PSO	APSO	559	294
(Qin ve Suganthan 2005)	DGA	SaDE	521	117

5.1.1. İyileştirilmiş yapay arı kolonisi algoritması

İyileştirilmiş ABC yapısı (MABC), DGA’dan esinlenilerek geliştirilmiştir (Gao ve Liu 2011). Geleneksel ABC mekanizmasındaki, Denklem 3.2’de verilen komşuluk arama denklemi yerine, MABC’de Denklem 5.1 ve Denklem 5.2’de verilen iki farklı arama denklemi kullanılarak yeni komşuluklar oluşturulmuştur.

$$v_{i,m} = x_{best,m} + \Phi_{i,j}(x_{i,m} - x_{r1,m}) \quad (5.1)$$

$$v_{i,m} = x_{r_1,m} + \Phi_{i,j}(x_{i,m} - x_{r_2,m}) \quad (5.2)$$

Burada, x_{best} iterasyondaki en iyi çözümü, r_1 ve r_2 iki adet rassal olarak seçilen çözümü göstermektedir. Temel ABC yapısındaki gibi tek bir indiste yapılan güncelleme yerine, her yinelemede önceden belirlenen M adet bileşen, m indisi kullanılarak değiştirilmektedir. $\Phi_{i,j}$ ise $[-1, 1]$ aralığında rassal bir sayıdır.

Yeni oluşturulan arama denklemlerine ek olarak, çalışma kapsamında iEMA'da olduğu gibi başlangıç çözümler kaotik yapı kullanılarak oluşturulmaktadır (Gao ve Liu 2011). Her yeni komşuluk oluşturulması adımında hangi denklemin (bkz. Denklem 5.1 ve Denklem 5.2) kullanılacağına, p seçim olasılığına bağlı olarak rasgele karar verilir. Denklem 5.1 daha çok sömürücü bir yapıda, Denklem 5.2 ise, rassal bir çözümün rehberliği nedeniyle, keşfedici yapıdadır. Böylece, geliştirilen algoritmanın keşfetme/sömürme dengesini daha iyi sağlaması hedeflenmiştir. Seçim olasılığı p parametresi ise bu dengeyi kontrol etmektedir. Söz konusu algoritmanın ABC'ye göre çok daha iyi performans gösterdiği kanıtlanmıştır (Gao ve Liu 2011). MABC algoritmasının adımları aşağıda verilmiştir.

- Adım 1: Popülasyon büyüklüğü (SN), limit değeri ($limit$), M , p ve durdurma kriteri gibi algoritma parametrelerinin atanması.
- Adım 2: Kaotik algoritma ile başlangıç popülasyonunun oluşturulması (bkz. Algoritma 4.1).
- Adım 3: Algoritmaya ait tutulacak istatistiklerin ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için amaç fonksiyon değerlerinin ($f(x)$) hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For $i = 1$ to SN Do**
- Adım 7: İşçi arılar tarafında seçilen kaynaklardan yeni çözümler üret. p olasılığına bağlı olarak Denklem 5.1 veya Denklem 5.2'yi seç.
- Adım 8: Toplamda M adet bileşeni seçilen denklem ile güncelleyerek v 'yi oluştur.
- Adım 9: x ile v arasında aç gözlü seçim yap ve $failure_i$ 'yi güncelle.

- Adım 10: **End For**
- Adım 11: Yiyecek kaynakları için olasılıkları hesapla. (bkz. Denklem 3.3 ve 3.4)
- Adım 12: **For $i = 1$ to SN Do**
- Adım 13: Gözcü arılar tarafında seçilen kaynaklardan yeni çözümler üret: p olasılığına bağlı olarak Denklem 5.1 veya Denklem 5.2’yi seç.
- Adım 14: Toplamda M adet bileşeni seçilen denklem ile güncelleyerek v ’yi oluştur.
- Adım 15: x ile v arasında açgözlü seçim yap ve $failure_i$ ’yi güncelle.
- Adım 16: Gerekli istatistikleri güncelle.
- Adım 17: **End For**
- Adım 18: **For $i = 1$ to SN Do**
- Adım 19: Eğer $failure_i > limit$ ise Kaşif arılar ile yeni rassal kaynak üret (bkz. Denklem 3.1).
- Adım 20: **End For**
- Adım 21: **Until** (durdurma kriteri sağlanıyorsa)

5.1.2. Global en iyi yapay arı kolonisi algoritması

Geliştirilen global en iyi ABC (GBest-ABC) kapsamında, geleneksel ABC komşuluk arama denkleminde (bkz. Denklem 3.2) farklı olmak üzere, Denklem 5.3 ile verilen yeni bir komşuluk üretme denklemi kullanılarak, global en iyi nokta bilgisi de işleme katılmıştır (Zhu ve Kwong 2010).

$$v_{i,j} = x_{i,j} + \Phi_{i,j}(x_{i,j} - x_{k,j}) + \Psi_{i,j}(y_j - x_{i,j}) \quad (5.3)$$

Denklem 5.3’de y_j global en iyi noktanın j . bileşeni, $\Psi_{i,j}$ ise $[0,C]$ aralığında düzgün dağılıma sahip rassal bir sayıdır. $\Phi_{i,j}$ $[-1,1]$ aralığında rassal sayı, $j \in \{1,2, \dots, D\}$ ve $k \in \{1,2, \dots, n\}$ geçerli aralıklarından rassal olarak seçilen iki sayıdır. Gbest-ABC kapsamında, Denklem 5.3 kullanılarak, ABC’nin sömürme yetisinin geliştirilmesi hedeflenmiştir. Çalışmanın sonuçları incelendiğinde, geleneksel ABC’ye göre Gbest-ABC’nin üstünlüğü görülmektedir (Zhu ve Kwong 2010).

GBest-ABC'nin adımları aşağıda özetlenmiştir.

- Adım 1: Popülasyon büyüklüğü (SN), limit değeri ($limit$) ve durdurma kriteri gibi algoritma parametrelerinin atanması.
- Adım 2: Belirlenen popülasyon büyüklüğü (SN), problem büyüklüğü (D) ve değişkenlerin alt ve üst sınırları (L,U) kullanılarak başlangıç parçacıklarının oluşturulması (bkz. Denklem 3.1)
- Adım 3: Algoritmaya ait tutulacak istatistiklerin ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için amaç fonksiyon değerlerinin ($f(x)$) hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For $i = 1$ to SN Do**
- Adım 7: İşçi arılar ile yeni çözümler (v) üret. (bkz. Denklem 5.3)
- Adım 8: x ile v arasında aç gözlü seçim yap ve $failure_i$ 'yi güncelle.
- Adım 9: **End For**
- Adım 10: Yiyecek kaynakları için olasılıkları hesapla. (bkz. Denklem 3.3 ve 3.4)
- Adım 11: **For $i = 1$ to SN Do**
- Adım 12: Gözcü arılar tarafında seçilen kaynaklardan yeni çözümler üret. (bkz. Denklem 5.3)
- Adım 13: x ile v arasında aç gözlü seçim yap ve $failure_i$ 'yi güncelle.
- Adım 14: Gerekli istatistikleri güncelle.
- Adım 15: **End For**
- Adım 16: **For $i = 1$ to SN Do**
- Adım 17: Eğer $failure_i > limit$, Kaşif arıları ile yeni rassal kaynak üret (bkz. Denklem 3.1).
- Adım 18: **End For**
- Adım 19: **Until** (durdurma kriteri sağlanıyorsa)

5.1.3. Adaptif parçacık sürüsü optimizasyonu

Söz konusu adaptif PSO (APSO) çalışması kapsamında, temel parçacık sürüsü optimizasyon algoritmasında kullanılan (bkz. Denklem 3.6 ve Denklem 3.7) kontrol parametreleri ω , c_1 ve c_2 , bulanık mantık tekniği ile tespit edilen anlık arama durumuna

göre adaptif olarak güncellenmektedir (Zhan ve ark. 2009). Söz konusu çalışma kapsamında, popülasyondaki parçacıkların dağılımları hesaplanarak, yakınsama veya arama dengesinin sağlanmasına çalışılmaktadır. Bu bağlamda, Denklem 5.4'de gösterilen şekilde her parçacığın diğer parçacıklara ortalama uzaklığı hesaplanmaktadır (Zhan ve ark. 2009).

$$d_i = \frac{1}{N-1} \sum_{j=1, j \neq i}^N \sqrt{\sum_{k=1}^D (x_i^k - x_j^k)^2} \quad (5.4)$$

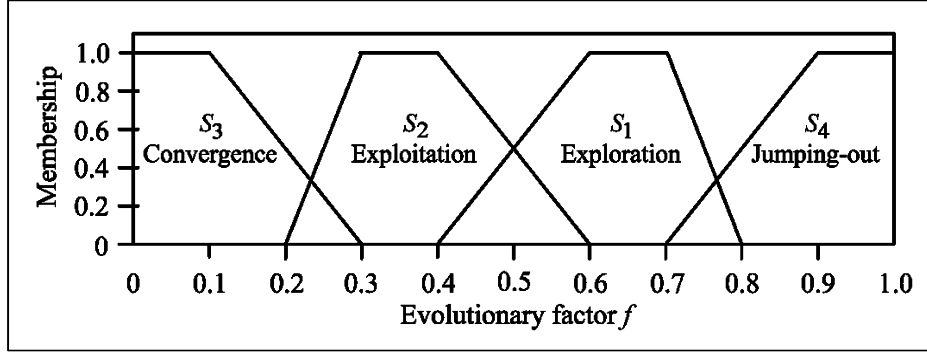
Denklem 5.4'de N popülasyon büyüklüğünü, D ise problem boyutunu göstermektedir. En iyi parça çevresindeki dağılımı gösteren f ise aşağıdaki şekilde hesaplanmaktadır (Zhan ve ark. 2009).

$$f = \frac{d_g - d_{min}}{d_{max} - d_{min}} \in [0,1] \quad (5.5)$$

Burada d_g en iyi parçanın değerini, d_{max} ve d_{min} ise sırasıyla elde edilen maksimum ve minimum ortalama uzaklık değerlerini göstermektedir (bkz. Denklem 5.4). Atalet momenti her yinelemede adaptif olarak Denklem 5.6 kullanılarak hesaplanır (sigmoid mapping) (Zhan ve ark. 2009).

$$\omega(f) = \frac{1}{1 + 1.5e^{-2.6f}} \in [0.4, 0.9] \quad (5.6)$$

Söz konusu çalışmada, hesaplanan f değeri ile hazırlanan bulanık üyelikler kullanılarak aramanın nasıl bir aşamada olduğuna karar verilir (bkz. **Hata! Başvuru kaynağı bulunamadı.**) ve Çizelge 5.2'de verildiği gibi c_1 ve c_2 parametrelerinde güncellemeler yapılır.



Şekil 5.1. Bulanık mantık üyelik diyagramları (Zhan ve ark. 2009)

Çizelge 5.2. Arama aşamasına göre kontrol parametrelerindeki güncellemeler

Durum	Strateji	c_1	c_2
Keşfetme (Exploration)	Strateji 1	Arttır	Azalt
Sömürme (Exploitation)	Strateji 2	Az arttır	Az azalt
Yakınsama (Convergence)	Strateji 3	Az arttır	Az arttır
Atlama (Jumping out)	Strateji 4	Azalt	Arttır

Güncellemeler aşağıdaki şekilde yapılmaktadır:

- Normal değişimler için (arttır veya azalt) (0,05, 1) arasında düzgün dağılımlı rassal sayı kullanılır.
- Az değişimler için (az arttır veya az azalt) 0,05 kullanılır.

Adaptif PSO'da (APSO) kullanılan parametre güncelleme adımları aşağıda özetlenmiştir:

- Adım 1: Parçacıkların birbirlerine uzaklıklarının ortalamalarını hesapla d_i (bkz. Denklem 5.4).
- Adım 2: d_{max} , d_{min} ve d_g değişkenlerini hesapla.
- Adım 3: f parametresini hesapla (bkz. Denklem 5.5).
- Adım 4: Çizelge 5.2 ve **Hata! Başvuru kaynağı bulunamadı.** kullanılarak c_1 ve c_2 parametrelerini hesapla.
- Adım 5: Denklem 5.6 ile ω 'yi güncelle.

Her yinelemede kontrol parametreleri yukarıda açıklandığı şekilde güncellenirken, geleneksel PSO yapısı da Bölüm 3.1.2’de açıklandığı şekilde kullanılmıştır.

5.1.4. Adaptif diferansiyel gelişim algoritması

Söz konusu adaptif DGA yapısında (SaDE), temel DGA algoritmasına sıklıkla uygulanan iki komşuluk üretme stratejisi ‘Rand/1/bin’ (bkz. Denklem 5.7) ve ‘Current to best/2/bin’ (bkz. Denklem 5.8) birlikte kullanılmaktadır. Her komşuluk üretme adımı öncesinde bu iki strateji arasında adaptif olarak hesaplanan bir seçim olasılığı kullanılarak tercih yapılmaktadır (Qin ve Suganthan 2005).

$$\mathbf{v}_{i,G} = \mathbf{x}_{r1,G} + F(\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G}) \quad (5.7)$$

$$\mathbf{v}_{i,G} = \mathbf{x}_{r1,G} + F(\mathbf{x}_{best,G} - \mathbf{x}_{i,G}) + F(\mathbf{x}_{r1,G} - \mathbf{x}_{r2,G}) \quad (5.8)$$

Burada, $\mathbf{x}_{r1,G}$, $\mathbf{x}_{r2,G}$ ve $\mathbf{x}_{r3,G}$, G ’nci nesilden rassal olarak seçilen bireyler, F ise bir parametredir. $\mathbf{x}_{best,G}$ G ’nci yinelemedeki en iyi çözümü gösterir. Çalışma kapsamında Denklem 5.7 ve 5.8 arasında seçim yapılmasını saplayan p seçim olasılığı ise aşağıdaki gibi hesaplanmaktadır.

$$p_1 = \frac{ns1 (ns2+nf2)}{ns2 (ns1+nf1)+ns1 (ns2+nf2)} \quad (5.9)$$

$$p_2 = 1 - p_1 \quad (5.10)$$

Burada, p_1 ve p_2 sırasıyla ‘rand/1/bin’ stratejisinin (strateji 1, Denklem 5.7) ve ‘current to best/2/bin’ stratejisinin (strateji 2, Denklem 5.8) seçim olasılıklarını göstermektedir. $ns(i)$, i . strateji kullanılarak üretilen yeni çözümlerden yeni nesile geçenlerin adedini, $nf(i)$ ise i . strateji kullanarak üretilen yeni çözümlerden yeni nesile geçemeyenlerin adedini göstermektedir. Çalışma kapsamında belirli bir öğrenme dönemi tanımlanmış ve bu dönem sonunda olasılıklar 0,5’e eşitlenerek ($p_1 = p_2 = 0,5$), öğrenme dönemi yeniden başlatılmıştır (Qin ve Suganthan 2005).

SaDE algoritmasının temel adımları aşağıda özetlenmiştir.

- Adım 1: Popülasyon büyüklüğü (N), F ve CR , p_1 ve p_2 kontrol parametreleri ve durdurma kriteri gibi algoritma parametrelerinin atanması.
- Adım 2: Belirlenen popülasyon büyüklüğü (N), problem büyüklüğü (D) ve değişkenlerin alt ve üst sınırları (L, U) kullanılarak başlangıç parçacıklarının oluşturulması (bkz. Denklem 3.8)
- Adım 3: Algoritmaya ait tutulacak istatistiklerin ve sayaçların sıfırlanması.
- Adım 4: Başlangıç popülasyonu için amaç fonksiyon değerlerinin ($f(x)$) hesaplanması.
- Adım 5: **Repeat**
- Adım 6: **For $i = 1$ to N Do**
- Adım 7: p olasılıklarına göre Denklem 5.7 veya Denklem 5.8 arasında seçim yap.
- Adım 8: Seçilen denklem ile mutasyon vektörünü ($v_{i,g}$) oluştur.
- Adım 9: $v_{i,g}$ ile $x_{i,g}$ çaprazlama yap ve deneme vektörünü ($U_{i,g}$) oluştur (bkz. Denklem 3.10)
- Adım 10: $U_{i,g}$ ile $x_{i,g}$ arasında aç gözlü seçim uygula, $x_{i,g+1}$ 'i oluştur (bkz. Denklem 3.11)
- Adım 11: ns ve nf değerlerini güncelle.
- Adım 12: p_1 ve p_2 değerlerini güncelle. (bkz. Denklem 5.9 ve Denklem 5.10)
- Adım 13: **End For**
- Adım 14: Gerekli istatistikleri güncelle.
- Adım 15: $g = g + 1$
- Adım 16: **Until** (durdurma kriteri sağlanıyorsa)

5.2. Test Ortamı

GBest-ABC algoritmasında $C = 1,5$, yeniden oluşturma limiti değeri için ise $SN * D$ (SN : Popülasyon büyüklüğü, D : Problem boyutu) alınmıştır (Zhu ve Kwong 2010). MABC algoritmasında seçim olasılığı $p = 0,25$, M ise problem boyutuna ve tipine

bağlı olarak değişen değerler almıştır (Gao ve Liu 2011). Yeniden oluşturma limit değeri ise diğer çalışmada olduğu gibi $SN * D$ olarak atanmıştır (Gao ve Liu 2011). SaDE mekanizmasında ise F değeri olarak (0,2) aralığında (0.5,0.3) ortalama ve standart sapma değerleri ile normal dağılan rassal sayılardan üretilmiş, çaprazlama oranını gösteren CR (crossover rate) ise 0,5 ortalama ve 0,1 standart sapmaya sahip normal dağılan rassal değer olarak alınmıştır. Ek olarak, popülasyon büyüklüğü (NP) ise küçük problemler için ($D = 10$) 50, büyük problemler ($D = 30, D = 50$) için ise 100 olarak verilmiştir (Qin ve Suganthan 2005). APSO yapısında algoritma parametreleri adaptif olarak kontrol edilmektedir. Buna rağmen, parametreler üzerinde bazı kısıtlamalar yapılmıştır. Atalet momentinin başlangıç değeri 0,9, c_1 ve c_2 katsayılarının ise 2,0 olarak alınmıştır. c_1 ve c_2 değerlerinin koşum boyunca [1,5, 2,5] aralığında kalması sağlanmış ve toplamları 4,0 değerini geçerse Denklem 5.11'deki gibi normalizasyon işlemine tabi tutulmuşlardır. Ek olarak, popülasyon büyüklüğü 20 olarak alınmıştır (Zhan ve ark. 2009).

$$c_i = \frac{c_i}{c_1 + c_2} 4.0 \quad (i = 1, 2) \quad (5.11)$$

Yapılacak karşılaştırmaların eşit koşullar altında gerçekleştirilmesi için iEMA ile birlikte diğer algoritmalar da MATLAB(R2011a) ortamında kodlanmış ve aynı bilgisayar ortamında koşumlar gerçekleştirilmiştir. Karşılaştırma analizleri için daha önceki karşılaştırma çalışmalarında kullanılan (bkz. Çizelge 4.3) farklı boyutlardaki standart ve karma test problemleri kullanılmıştır. Kullanılan test kümesi, tek-modlu, çok-modlu, küçük (10D), orta (30D) ve büyük (50D) problemlerden oluşmaktadır..

Meta sezgisel algoritmaların başarısı, sadece bulunan çözümlerin kalitesi ile değil, çözümlerin bulunması için harcanan süre veya amaç fonksiyonu değerlendirme adedi ile de ölçülmeli ve karşılaştırılmalıdır. Bu sebeple, iEMA ile iyileştirilmiş meta sezgisel algoritmaların karşılaştırılmasında eşit koşum şartları oluşturabilmek için çözüm sürelerinde kısıtlamaya gidilmiş ve ön testler ile belirlenen maksimum işlem süreleri tanımlanmıştır. Karşılaştırma analizi kapsamında, küçük test problemleri için 5 saniye, orta test problemleri için 10 saniye ve büyük test problemleri için ise 15 saniye işlem

süresi kullanılmıştır. Her problem için bu süreler sonunda koşum durdurulmuş ve bulunan en iyi fonksiyon değerleri raporlanmıştır.

5.3. Karşılaştırma Sonuçları

Yapılan çalışma kapsamında, iEMA ile diğer temel meta sezgisel algoritmaların iyileştirilmiş sürümleri belirtilen problem kümesi kullanılarak test edilmiştir. Tüm algoritmalar ile tüm problemler üzerinde her biri rassal bir başlangıca sahip toplam 30 koşum yaptırılmış ve geçerli süre sonunda bulunan fonksiyon değerlerinin ortalamaları ve standart sapmaları raporlanmıştır. Bulunan sonuçlar Çizelge 5.3'te verilmiştir. Çizelge 5.3'te problemin adı, boyutu (D) ve tipi verilmiştir. Çizelgede diğer sütunlar söz konusu algoritmalar ile elde edilen fonksiyon değerlerinin ortalamaları ve standart sapmalarını göstermektedir. Gerçekleştirilen karşılaştırma çalışmasında, Çizelge 5.3'teki sonuçların istatistiksel olarak anlamlılıkları t-testleri ile incelenmiştir (%95 güvenilirlikte). Çizelge 5.4, Çizelge 5.5, Çizelge 5.6 ve Çizelge 5.7'de iEMA ile diğer iyileştirilmiş meta sezgisel algoritmaların sonuçları arasında yapılan t-testleri özetlenmiştir. Tablolarda, “*t*” sütunu iki ortalama arasındaki farkın test edilmesi ile elde edilen *t* istatistik değerini göstermektedir. Başka bir deyişle, herhangi iki algoritma arasındaki farkın anlamlı olabilmesi için $t \leq -2,045$ veya $t \geq 2,045$ olması gerekmektedir. Çizelge 5.4, Çizelge 5.5, Çizelge 5.6 ve Çizelge 5.7'de farkın anlamlı olup olmadığı 2. sütunda ve son sütunda ise fark anlamlı ise kazananın hangi algoritma olduğu verilmiştir. Eğer iki algoritma arasındaki fark anlamlı değil ise, kazana sütunu değeri ‘Yok’ olarak kodlanmıştır. Tüm yapılan testlerin özeti Çizelge 5.8'de verilmiştir. Çizelge 5.8'de ‘+’ sütunu iEMA'nın üstün geldiği problem adedini, ‘-’ sütunu karşılaştırılan algoritmanın üstün geldiği problem adedini göstermektedir. ‘Fark yok’ sütunu ise iki algoritma arasında istatistiksel olarak anlamlı bir farkın olmadığı problem adedini göstermektedir.

Çizelge 5.8 incelendiğinde, tez çalışması kapsamında geliştirilen iEMA yapısının karşılaştırılan tüm algoritmalarından daha iyi performans gösterdiği anlaşılmaktadır. iEMA'ya en yakın sonuçlar veren algoritma MABC olmuştur. Toplam 34 test probleminin 13'ünde iEMA, MABC'ye karşı üstünlük sağlamıştır (Rosenbeock.50, Griwank.50, Ackley.10, Ackley.50, F2.30, F10.10, F10.50, F15.30, F4.10, F20.10,

F20.30, F25.10 ve F25.30). Aynı şekilde iEMA, GbestABC'ye karşı toplam 13 problemde daha iyi performans göstermiş ve 17 problemde ise aralarında istatistiksel olarak anlamlı bir fark oluşmamıştır. Çizelge 5.8'de iEMA'nın hem APSO hem de SaDE'ye karşı üstünlüğü net bir şekilde görülmektedir. APSO 24 problemin sadece birinde (Rosenbrock.10) iEMA'ya üstünlük sağlamıştır. iEMA neredeyse tüm problemlerde SaDE'den daha iyi sonuçlar elde etmiştir. Ek olarak, Çizelge 5.9'da tüm algoritmalar ile 30 rassal koşum sonucunda erişilen en iyi amaç fonksiyonu değerleri karşılaştırılmıştır. iEMA'nın koşumlar sonucunda bulunan en iyi değerler karşılaştırmasında da diğer algoritmalara karşın üstün olduğu görülmektedir.

Ek olarak, sonuçlar detaylı incelendiğinde iEMA'nın tüm problem boyutlarında (10D, 30D, ve 50D) ve tüm problem tiplerinde (tek-çok modlu, ayrıştırılabilir-ayrıştırılmaz) benzer bir şekilde başarılı olduğu görülmektedir. Hesaplama süreleri eşit alınarak yapılan söz konusu analizlerin sonuçları, iEMA'nın, EMA'ya kıyasla dirençli, güçlü ve başarılı bir meta sezgisel haline geldiğinin kanıtıdır. Tüm bu sonuçlar ışığında, yapılan çalışmalar sonucunda EMA'nın başarılı bir şekilde iyileştirildiği ve geliştirildiği söylenebilir.

Çizelge 5.3. İyileştirilmiş sürümlerin testleri sonuçları

No	Problem	D	Tip	iEMA		MABC		GbestABC		APSO		SaDE	
				Ort.	Sapma	Ort.	Sapma	Ort.	Sapma	Ort.	Sapma	Ort.	Sapma
1	Beale	2	UN	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00
2	Colville	4	UN	3,60E-03	1,40E-06	3,45E-03	5,23E-04	7,32E-03	2,35E-04	9,43E-03	4,91E-05	9,03E-03	6,91E-03
3	Hartmann2	6	MN	6,41E-04	2,29E-07	6,21E-04	2,39E-07	3,21E-05	5,19E-05	8,11E-04	9,05E-06	7,32E-04	9,01E-05
4	Michalewick3	9	MS	2,38E-02	5,45E-04	2,38E-02	4,33E-04	3,11E-02	3,21E-04	3,03E-02	4,51E-05	6,29E-02	5,61E-04
5	Sphere.10	10	US	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00
6	Sphere.30	30	US	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00
7	Sphere.50	50	US	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	0,00E00	1,24E-12	4,32E-13	6,36E-09	8,81E-07
8	Rosenbrock.10	10	UN	3,60E-04	5,40E-06	8,03E-05	4,30E-04	3,44E-05	5,40E-05	9,20E-06	5,50E-05	7,70E-03	9,10E-04
9	Rosenbrock.30	30	UN	3,34E-03	8,55E-05	4,75E-03	4,22E-02	6,70E-03	8,06E-02	2,84E-02	3,27E00	2,81E01	6,75E00
10	Rosenbrock.50	50	UN	5,04E-02	2,20E-04	6,20E-01	9,00E-03	9,30E-01	2,00E-03	1,02E00	3,00E-02	4,10E-01	5,13E-03
11	Griewank.10	10	MN	4,53E-08	5,40E-07	9,76E-09	1,53E-07	7,71E-08	4,03E-07	5,87E00	8,30E-01	6,31E-07	6,58E-05
12	Griewank.30	30	MN	4,05E-06	6,61E-08	1,05E-06	4,40E-07	1,24E-05	2,36E-04	1,67E-01	2,41E-02	7,81E-05	4,66E-04
13	Griewank.50	50	MN	3,94E-05	1,53E-07	7,12E-05	7,20E-06	7,76E-04	4,30E-05	4,21E00	2,33E-02	8,05E-04	5,42E-04
14	Ackley.10	10	MN	9,84E-06	1,54E-08	5,94E-05	9,10E-06	3,31E-07	2,90E-06	6,53E-02	4,22E-02	4,52E-01	1,53E-02
15	Ackley.30	30	MN	3,34E-03	8,55E-04	8,75E-03	4,22E-02	6,70E-03	8,06E-02	2,84E-01	3,27E00	2,81E01	6,75E00
16	Ackley.50	50	MN	3,45E-02	3,53E-03	7,53E-02	7,70E-04	1,22E-02	9,53E-03	1,12E01	3,30E-01	4,25E02	1,35E01
17	F2.10	10	UN	1,55E01	6,33E01	1,82E01	6,44E02	2,03E01	1,55E01	5,43E03	9,11E04	9,54E03	1,42E04
18	F2.30	30	UN	3,14E03	1,30E02	1,21E04	9,54E03	3,62E04	8,04E03	6,52E06	7,32E05	3,71E07	6,39E06
19	F2.50	50	UN	2,05E04	9,94E03	2,05E04	9,13E03	8,23E04	5,19E03	6,83E05	8,13E03	5,27E04	9,92E03
20	F10.10	10	MN	2,25E00	3,21E00	9,41E00	7,72E00	9,32E00	5,92E00	4,40E01	5,11E02	7,12E04	3,16E04
21	F10.30	30	MN	7,72E02	3,77E02	9,43E01	5,03E02	4,28E03	5,85E04	9,94E04	9,55E02	5,33E04	1,24E04
22	F10.50	50	MN	1,16E05	5,19E04	4,32E05	5,44E05	7,23E05	9,32E03	5,62E06	1,43E04	7,47E07	8,51E06
23	F15.10	10	MS	3,26E00	2,50E00	3,19E00	9,03E01	2,95E00	5,11E01	6,23E02	5,03E00	2,88E02	5,11E01

Çizelge 5.3. İyileştirilmiş sürümlerin testleri sonuçları (devam)

No	Problem	D	Tip	iEMA		MABC		GbestABC		APSO		SaDE	
				Ort.	Sapma	Ort.	Sapma	Ort.	Sapma	Ort.	Sapma	Ort.	Sapma
24	F15.30	30	MS	5,34E02	4,05E02	5,44E03	8,84E03	1,01E03	7,73E02	8,31E03	4,23E02	4,28E04	9,54E04
25	F15.50	50	MS	2,36E05	8,84E03	1,54E05	3,21E04	8,34E05	2,14E03	4,81E06	1,41E03	6,32E05	3,31E04
26	F4.10	10	UN	2,06E00	6,40E00	9,43E00	4,33E00	1,10E01	6,01E01	3,60E01	6,41E03	5,51E02	8,03E02
27	F4.30	30	UN	4,70E01	9,00E02	2,34E01	9,23E02	5,20E02	6,61E03	3,84E03	3,17E01	5,31E03	1,54E04
28	F4.50	50	UN	7,72E03	6,61E03	9,10E03	7,72E04	5,33E04	5,23E04	2,17E04	4,47E02	6,33E03	2,55E01
29	F20.10	10	MN	2,76E01	6,31E01	8,43E01	5,23E01	1,55E01	9,04E02	5,29E02	1,52E00	8,33E03	9,84E04
30	F20.30	30	MN	5,16E04	9,05E03	1,04E05	1,04E05	8,31E04	1,45E05	9,61E06	4,77E04	5,17E04	2,41E04
31	F20.50	50	MN	1,92E06	4,43E04	7,33E05	5,15E02	6,23E06	8,75E04	4,64E06	7,41E05	5,22E06	1,27E05
32	F25.10	10	MN	1,05E01	5,42E01	4,23E01	6,34E01	4,23E01	3,31E02	9,43E02	4,97E00	6,21E02	6,13E02
33	F25.30	30	MN	9,30E03	5,54E02	2,17E05	9,45E03	4,11E04	7,65E03	5,93E04	5,18E03	9,94E04	5,33E07
34	F25.50	50	MN	3,75E05	8,53E05	6,43E05	5,61E04	5,23E05	1,44E04	2,29E08	5,32E05	5,04E06	6,09E05

Çizelge 5.4. iEMA ve MABC farkı istatistiksel anlamlılık testi sonuçları

Problem	D	Tip	t	Anlamlılık (%95)	Kazanan
Beale	2	UN	0,000	Anlamlı değil	Yok
Colville	4	UN	1,571	Anlamlı değil	Yok
Hartmann2	6	MN	330,949	Anlamlı	MABC
Michalewick3	9	MS	0,000	Anlamlı değil	Yok
Sphere.10	10	US	0,000	Anlamlı değil	Yok
Sphere.30	30	US	0,000	Anlamlı değil	Yok
Sphere.50	50	US	0,000	Anlamlı değil	Yok
Rosenbrock.10	10	UN	3,562	Anlamlı	MABC
Rosenbrock.30	30	UN	-0,183	Anlamlı değil	Yok
Rosenbrock.50	50	UN	-346,544	Anlamlı	iEMA
Griewank.10	10	MN	0,347	Anlamlı değil	Yok
Griewank.30	30	MN	36,930	Anlamlı	MABC
Griewank.50	50	MN	-24,186	Anlamlı	iEMA
Ackley.10	10	MN	-29,830	Anlamlı	iEMA
Ackley.30	30	MN	-0,702	Anlamlı değil	Yok
Ackley.50	50	MN	-61,852	Anlamlı	iEMA
F2.10	10	UN	-0,023	Anlamlı değil	Yok
F2.30	30	UN	-5,144	Anlamlı	iEMA
F2.50	50	UN	0,000	Anlamlı değil	Yok
F10.10	10	MN	-4,691	Anlamlı	iEMA
F10.30	30	MN	5,905	Anlamlı	MABC
F10.50	50	MN	-3,163	Anlamlı	iEMA
F15.10	10	MS	0,004	Anlamlı değil	Yok
F15.30	30	MS	-3,037	Anlamlı	iEMA
F15.50	50	MS	13,489	Anlamlı	MABC
F4.10	10	UN	-5,224	Anlamlı	iEMA
F4.30	30	UN	0,100	Anlamlı değil	Yok
F4.50	50	UN	-0,098	Anlamlı değil	Yok
F20.10	10	MN	-3,789	Anlamlı	iEMA
F20.30	30	MN	-2,749	Anlamlı	iEMA
F20.50	50	MN	146,750	Anlamlı	MABC
F25.10	10	MN	-2,088	Anlamlı	iEMA
F25.30	30	MN	-120,177	Anlamlı	iEMA
F25.50	50	MN	-1,716	Anlamlı değil	Yok

Çizelge 5.5. iEMA ve GBest-ABC farkı istatistiksel anlamlılık testi sonuçları

Problem	D	Tip	t	Anlamlılık (%95)	Kazanan
Beale	2	UN	0,000	Anlamlı değil	Yok
Colville	4	UN	-86,702	Anlamlı	iEMA
Hartmann2	6	MN	64,259	Anlamlı	GBest-ABC
Michalewick3	9	MS	-63,215	Anlamlı	iEMA
Sphere.10	10	US	0,000	Anlamlı değil	Yok
Sphere.30	30	US	0,000	Anlamlı değil	Yok
Sphere.50	50	US	0,000	Anlamlı değil	Yok
Rosenbrock.10	10	UN	32,862	Anlamlı	GBest-ABC
Rosenbrock.30	30	UN	-0,228	Anlamlı değil	Yok
Rosenbrock.50	50	UN	-2394,441	Anlamlı	iEMA
Griewank.10	10	MN	-0,258	Anlamlı değil	Yok
Griewank.30	30	MN	-0,194	Anlamlı değil	Yok
Griewank.50	50	MN	-93,826	Anlamlı	iEMA
Ackley.10	10	MN	17,959	Anlamlı	GBest-ABC
Ackley.30	30	MN	-0,228	Anlamlı değil	Yok
Ackley.50	50	MN	12,019	Anlamlı	GBest-ABC
F2.10	10	UN	-0,403	Anlamlı değil	Yok
F2.30	30	UN	-22,519	Anlamlı	iEMA
F2.50	50	UN	-30,187	Anlamlı	iEMA
F10.10	10	MN	-5,750	Anlamlı	iEMA
F10.30	30	MN	-0,328	Anlamlı değil	Yok
F10.50	50	MN	-63,009	Anlamlı	iEMA
F15.10	10	MS	0,033	Anlamlı değil	Yok
F15.30	30	MS	-3,013	Anlamlı	iEMA
F15.50	50	MS	-360,116	Anlamlı	iEMA
F4.10	10	UN	-0,812	Anlamlı değil	Yok
F4.30	30	UN	-0,388	Anlamlı değil	Yok
F4.50	50	UN	-4,736	Anlamlı	iEMA
F20.10	10	MN	0,073	Anlamlı değil	Yok
F20.30	30	MN	-1,188	Anlamlı değil	Yok
F20.50	50	MN	-240,702	Anlamlı	iEMA
F25.10	10	MN	-0,519	Anlamlı değil	Yok
F25.30	30	MN	-22,709	Anlamlı	iEMA
F25.50	50	MN	-0,950	Anlamlı değil	Yok

Çizelge 5.6. iEMA ve APSO farkı istatistiksel anlamlılık testi sonuçları

Problem	D	Tip	t	Anlamlılık (%95)	Kazanan
Beale	2	UN	0,000	Anlamlı değil	Yok
Colville	4	UN	-650,087	Anlamlı	iEMA
Hartmann2	6	MN	-102,854	Anlamlı	iEMA
Michalewick3	9	MS	-65,102	Anlamlı	iEMA
Sphere.10	10	US	0,000	Anlamlı değil	Yok
Sphere.30	30	US	0,000	Anlamlı değil	Yok
Sphere.50	50	US	-15,722	Anlamlı	iEMA
Rosenbrock.10	10	UN	34,768	Anlamlı	APSO
Rosenbrock.30	30	UN	-0,042	Anlamlı değil	Yok
Rosenbrock.50	50	UN	-177,019	Anlamlı	iEMA
Griewank.10	10	MN	-38,737	Anlamlı	iEMA
Griewank.30	30	MN	-37,953	Anlamlı	iEMA
Griewank.50	50	MN	-989,653	Anlamlı	iEMA
Ackley.10	10	MN	-8,474	Anlamlı	iEMA
Ackley.30	30	MN	-0,470	Anlamlı değil	Yok
Ackley.50	50	MN	-185,808	Anlamlı	iEMA
F2.10	10	UN	-0,326	Anlamlı değil	Yok
F2.30	30	UN	-48,763	Anlamlı	iEMA
F2.50	50	UN	-282,576	Anlamlı	iEMA
F10.10	10	MN	-0,447	Anlamlı değil	Yok
F10.30	30	MN	-526,149	Anlamlı	iEMA
F10.50	50	MN	-559,952	Anlamlı	iEMA
F15.10	10	MS	-604,706	Anlamlı	iEMA
F15.30	30	MS	-72,728	Anlamlı	iEMA
F15.50	50	MS	-2798,654	Anlamlı	iEMA
F4.10	10	UN	-0,029	Anlamlı değil	Yok
F4.30	30	UN	-23,058	Anlamlı	iEMA
F4.50	50	UN	-11,558	Anlamlı	iEMA
F20.10	10	MN	-43,510	Anlamlı	iEMA
F20.30	30	MN	-1078,305	Anlamlı	iEMA
F20.50	50	MN	-20,070	Anlamlı	iEMA
F25.10	10	MN	-93,841	Anlamlı	iEMA
F25.30	30	MN	-52,569	Anlamlı	iEMA
F25.50	50	MN	-1245,268	Anlamlı	iEMA

Çizelge 5.7. iEMA ve SaDE farkı istatistiksel anlamlılık testi sonuçları

Problem	D	Tip	t	Anlamlılık (%95)	Kazanan
Beale	2	UN	0,000	Anlamlı değil	Yok
Colville	4	UN	-4,304	Anlamlı	iEMA
Hartmann2	6	MN	-5,532	Anlamlı	iEMA
Michalewick3	9	MS	-273,812	Anlamlı	iEMA
Sphere.10	10	US	0,000	Anlamlı değil	Yok
Sphere.30	30	US	0,000	Anlamlı değil	Yok
Sphere.50	50	US	-0,040	Anlamlı değil	Yok
Rosenbrock.10	10	UN	-44,178	Anlamlı	iEMA
Rosenbrock.30	30	UN	-22,799	Anlamlı	iEMA
Rosenbrock.50	50	UN	-383,587	Anlamlı	iEMA
Griewank.10	10	MN	-0,049	Anlamlı değil	Yok
Griewank.30	30	MN	-0,870	Anlamlı değil	Yok
Griewank.50	50	MN	-7,737	Anlamlı	iEMA
Ackley.10	10	MN	-161,807	Anlamlı	iEMA
Ackley.30	30	MN	-22,799	Anlamlı	iEMA
Ackley.50	50	MN	-172,066	Anlamlı	iEMA
F2.10	10	UN	-3,674	Anlamlı	iEMA
F2.30	30	UN	-31,798	Anlamlı	iEMA
F2.50	50	UN	-12,559	Anlamlı	iEMA
F10.10	10	MN	-12,341	Anlamlı	iEMA
F10.30	30	MN	-23,192	Anlamlı	iEMA
F10.50	50	MN	-48,003	Anlamlı	iEMA
F15.10	10	MS	-30,484	Anlamlı	iEMA
F15.30	30	MS	-2,427	Anlamlı	iEMA
F15.50	50	MS	-63,309	Anlamlı	iEMA
F4.10	10	UN	-3,744	Anlamlı	iEMA
F4.30	30	UN	-1,869	Anlamlı değil	Yok
F4.50	50	UN	1,152	Anlamlı değil	Yok
F20.10	10	MN	-0,462	Anlamlı değil	Yok
F20.30	30	MN	-0,021	Anlamlı değil	Yok
F20.50	50	MN	-134,381	Anlamlı	iEMA
F25.10	10	MN	-5,434	Anlamlı	iEMA
F25.30	30	MN	-0,009	Anlamlı değil	Yok
F25.50	50	MN	-24,372	Anlamlı	iEMA

Çizelge 5.8. İyileştirilmiş sürümler arasında karşılaştırma sonuçları özeti

Karşılaştırma çifti	+ (iEMA daha iyi)	- (Diğer algoritma daha iyi)	Fark yok
iEMA-MABC	13	6	15
iEMA-GBest-ABC	13	4	17
iEMA-APSO	25	1	8
iEMA-SaDE	23	0	11

Çizelge 5.9. İyileştirilmiş sürümler arasında bulunan en iyi değer karşılaştırması

Karşılaştırma çifti	iEMA daha iyi	Diğer algoritma daha iyi	Eşit
iEMA-MABC	17	13	4
iEMA-GBest-ABC	18	12	4
iEMA-APSO	27	4	3
iEMA-SaDE	31	0	3

6. İYİLEŞTİRİLMİŞ ELEKTROMANYETİK ALGORİTMANIN KISITLI OPTİMİZASYON PROBLEMLERİNE UYARLANMASI

Kısıtlı optimizasyon problemleri günümüzde birçok farklı mühendislik probleminde içerisinde yer almaktadır (Parsopoulos ve Vrahatis 2002). Kısıtlı optimizasyon probleminin (doğrusal veya doğrusal olmayan) matematiksel modeli aşağıda verilmiştir (Michalewicz ve Schoenauer 1996):

$$\min f(\mathbf{x}) \quad (6.1)$$

$$g_k(\mathbf{x}) \leq 0, k = 1, \dots, m. \quad (6.2)$$

$$h_p(\mathbf{x}) = 0, p = 1, \dots, l. \quad (6.3)$$

$$\mathbf{x}_{min} \leq \mathbf{x} \leq \mathbf{x}_{max} \quad (6.4)$$

Burada f fonksiyonu arama uzayı S içerisinde tanımlıdır. S ise n -boyutlu bir uzay olan \mathbb{R}^n içinde tarif edilir ($S \subseteq \mathbb{R}^n$). Söz konusu modelde, tüm değişkenler kendi alt ve üst sınırları içerisinde tanımlıdır (Denklem 6.4). Bu tip problemlerin optimum noktası (\mathbf{x}^*), problem içinde verilen $m + l$ adet kısıtın (Denklem 6.2 ve Denklem 6.3) sağlandığı bir kabul edilebilir (uygun) alan (F) içerisinde yer almaktadır.

Literatürde, meta sezgisel algoritmaların geliştirilmesini veya iyileştirilmesini içeren çalışmalar çoğu zaman kısıtsız optimizasyon problemlerinin çözümünü hedeflemektedir. Bu sebeple, literatürde kısıtlı optimizasyon problemleri için geliştirilen yöntemler çoğu zaman kısıtsız optimizasyon için geliştirilen meta sezgisel algoritmalar üzerinde gerçekleştirilen uyarlamaları içermektedir. Söz konusu temel uyarlamalar dört ana grupta toplanabilir (Karaboga ve Akay 2011, Koziel ve Michalewicz 1999): (i) uygun olmayan çözümleri uygun çözüm haline getiren yöntemlerin kullanılması (Michalewicz ve Janikow 1991, Michalewicz ve Schoenauer 1996), (ii) uygun olmayan çözümleri cezalandırarak, daha az kullanılmalarının sağlanması (Bean ve Hadj-Alouane 1992, Homaifar ve ark. 1994, Joines ve Houck 1994), (iii) sadece uygun çözümler ile çalışan teknikler (Deb 2000, Michalewicz ve Nazhiyath 1995, Schoenauer ve Xanthakis

1993) ve (iv) diğler melez yöntemler (Paredis 1994, Reynolds ve ark. 1995). Tez çalışması kapsamında geliştirilen algoritma “uygun olmayan çözümleri cezalandırarak, daha az kullanılmalarının sağlanması” grubuna ait bir yöntem olarak değerlendirilebilir.

6.1. Kısıtlı Problemler için Uyarlamalar

Tez çalışması kapsamında geliştirilen iEMA'nın kısıtlı optimizasyon problemlerine uyarlanmış sürümü (ciEMA) için iEMA üzerinde üç temel güncelleme yapılmıştır. Bunlar: (i) yük hesaplanma denkleminde uygun olmayan parçacıkların daha az elektrik yüküne sahip olmaları ve (ii) kuvvet hesaplama adımında uygun olmayan parçacıkların daha düşük olasılık ile bileşke kuvvet hesabına katılması sağlanmış; (iii) uygun olmayan çözümler için de bir yeniden oluşturma limiti belirlenmiştir. Söz konusu uyarlamalar aşağıda özetlenmiştir:

Elektrik yükü hesabı:

Bu uyarlama ile uygun olmayan parçacıkların, uygun parçacıklara göre daha kötü amaç fonksiyonu değerlerine sahip olmaları hedeflenmiş ve daha düşük elektrik yükü taşımaları sağlanmıştır. Buradaki temel hedef, uygun olmayan parçacıkların diğler parçacıklara daha az kuvvet uygulamalarını sağlamaya çalışmaktır. Bu kapsamda, önce Denklem 6.5 ile her bir parçacık i için kısıt aşım miktarı $\Phi(x_i)$, hesaplanır.

$$\Phi(x_i) = \sum_{k=1}^m \max(g_k(x), 0) + \sum_{p=1}^l \max(|h_p(x)| - \varepsilon, 0), \quad \forall i. \quad (6.5)$$

Bu ifadede, ε hata payını gösteren bir parametredir. Daha sonra, Algoritma 6.1 kullanılarak parçacıkların amaç fonksiyon değerleri güncellenir. Buradaki amaç, uygun olmayan parçacıkların, uygun çözüm gösteren en kötü parçacıktan (x_{worst}) sahip oldukları kısıt aşım miktarı kadar daha kötü bir çözümü göstermelerini sağlamaktır. Algoritma 6.1'de m ise popülasyon büyüklüğünü göstermektedir.

```

For i= 1 to m
  Eğer  $x_i$  uygun çözüm değil ise
     $f(x_i) = f(x_{worst}) + \Phi(x_i)$ 
  end if
end for

```

Algoritma 6.1. Amaç fonksiyon değeri güncellemesi

Bileşke kuvvet hesabı:

Önerilen söz konusu uyarlama ile daha önce deneysel tasarım yolu güncellenen ve iyileştirilen kuvvet hesaplama adımında değişiklik yapılmıştır. Bu kapsamda, daha önce faktör seviyesi olarak test edilen sabit k adet parçacığın ($k = m * 0.1$) kullanılması yöntemi tercih edilmiştir (bkz. Bölüm 4.1). Bu noktada, yapılan deneysel tasarım sonuçları incelendiğinde önerilen iki yöntemin etkilerinin birbirine çok yakın olduğu görülecektir (bkz. Çizelge 4.7). Bu sebeple yapılan yöntem değişikliğinin çözüm kalitesine negatif yönde bir etkisi olmayacağı tahmin edilmektedir.

Önerilen yöntem kapsamında her yinelemede kuvvet hesaplama adımına dahil edilecek parçacıklar rulet çemberi tekniği ile belirlenmektedir. Her parçacığın rulet çemberindeki seçilme olasılıklarının hesabında, Denklem 6.6 ve 6.7 kullanılmaktadır. Denklem 6.6 incelendiğinde, uygun olan çözümlerin seçilme olasılıklarının $[\delta, 1]$ arasında, uygun olmayanların ise $[0, 1 - \delta]$ arasında olacağı görülmektedir. Burada δ bir parametredir ve çözümün uygun alanlara yakınsama hızını kontrol etmektedir. Yeni kuvvet hesaplama adımı ile uygun olan çözümlerin seçilme olasılıkları daha fazla olacağından, uygun parçacıkların popülasyondaki diğer parçacıkları daha çok etkilemeleri, böylece uygun alanlara yakınsamanın sağlanması hedeflenmiştir.

$$p_i = \begin{cases} \delta + \frac{fitness_i}{\sum_{j=1}^m fitness_j} \times (1 - \delta), & \text{eğer parçacık uygun ise} \\ \left(1 - \frac{\Phi(x_i)}{\sum_{j=1}^m \Phi(x_j)}\right) \times (1 - \delta), & \text{eğer parçacık uygun değil ise} \end{cases} \quad (6.6)$$

$$fitness_i = \begin{cases} 1/(1 + f(x_i)), & \text{eğer } f(x_i) \geq 0 \\ 1 + abs(f(x_i)), & \text{eğer } f(x_i) < 0 \end{cases} \quad (6.7)$$

Yeniden oluřturma adımı (ConRegen):

Tez alıřması kapsamında geliřtirilen iEMA’da kullanılan yeniden oluřturma adımı (*Regen*) kapsamında, belirli bir yineleme sonrasında geliřtirilememiř paracıklar yok edilerek, belirlenen alt ve st sınırlar ierisinde rassal olarak yeniden oluřturulmaktadır. ciEMA kapsamında ise *Regen*’e ek olarak, yeni bir yeniden oluřturma limiti (*ConRegen*) tanımlanmıřtır. Bu kapsamda, belirli bir yineleme sonucunda uygun olan alana gelememiř uygun olmayan zmler yok edilerek (ama fonksiyon deęerlerine bakılmadan, belirlenen alt ve st sınırlar ierisinde rassal olarak yeniden oluřturulmaktadır. Yapılan n deneyler ile tanımlanan *ConRegen*’in $(2 * m * D)$ adet yineleme olmasına karar verilmiřtir. Burada, m poplasyon byklę, D ise problem boyutudur.

Geliřtirilen ciEMA kapsamında, iki zm arasında karřılařtırmalar *Deb* kuralları kullanılarak gerekleřtirilir (*Regen* parametresi iin) (Karaboga ve Akay 2011). *Deb* kuralları ařaęıda verilmiřtir.

- Kabul edilebilir blgedeki her bir zm, kabul edilebilir olmayan blgedeki zme tercih edilir.
- Her iki zm de kabul edilebilir blgede ise daha dřk ama fonksiyonuna sahip zm tercih edilir.
- Her iki zm de kabul edilebilir olmayan blgede ise daha dřk kısıt ařım miktarına sahip zm tercih edilir.

6.2. Kısıtlı Problemler iin İyileřtirilmiř Elektromanyetik Algoritma

Blm 6.1’de aıklanan  adet gncelleme ile iEMA’nın kısıtlı optimizasyon problemlerini zebilecek hale (ciEMA) getirilmesine alıřılmıřtır. ciEMA’nı temel adımları ařaęıda verilmiřtir. Adımlarda kullanılan *deneme1* ve *deneme2* deęiřkenleri sırasıyla *Regen* ve *ConRegen* iin paracıkların geliřtirilememe adetlerini gstermektedir. *limit1* ve *limit2* ise sırasıyla *Regen* ve *ConRegen* iin limit deęerlerini gstermektedir. Adımlar incelendięinde, ciEMA kapsamında gerekleřtirilen elektrik yk hesabı iin uyarılama (Satır: 4–6), bileřke kuvvet hesabı iin uyarılama (Satır: 13)

ve yeniden oluřturma adımı için uyarlama (Satır: 24) ilgili satırlarda görülebılır. ciEMA yapısının temel adımları ařağıda verilmiřtir.

- Adım 1: Popülasyon büyüklüğü (m), durdurma kriteri, $limit1$, $limit2$ gibi algoritma parametrelerinin atanması.
- Adım 2: Kaotik algoritma kullanılarak bařlangıç popülasyonunun oluřturulması. (bkz. Algoritma 4.1)
- Adım 3: Algoritmaya ait tutulacak istatistiklerin (global en iyi çözümler deęeri ve parçacık deęeri, $deneme1$, $deneme2$, vb.) ve sayaçların sıfırlanması.
- Adım 4: Bařlangıç popülasyonu için amaç fonksiyon deęerleri $f(x)$ 'in hesaplanması.
- Adım 5: Tüm parçacıkları için kısıt ařım miktarlarının hesaplanması. (bkz. Denklem 6.5)
- Adım 6: Tüm parçacıklar için amaç fonksiyonu $f(x)$ güncellemesi. (bkz. Algoritma 6.1)
- Adım 7: **Repeat**
- Adım 8: **For $i = 1$ to m Do**
- Adım 9: Elektrik yüklerini hesapla (bkz. Denklem 4.1)
- Adım 10: **End For**
- Adım 11: S kümesini oluřtur.
- Adım 12: **For $i = 1$ to m Do**
- Adım 13: Bileřke kuvveti hesapla. (k adet sabit parçacık ile Denklem 6.6 ve 6.7'yi kullan)
- Adım 14: **End For**
- Adım 15: x_{mean} ve x_{best} deęerlerini hesapla.
- Adım 16: **For $i = 1$ to m Do**
- Adım 17: F_{best} hesapla. (bkz. Denklem 2.3)
- Adım 18: Parçacıkları bileřke kuvvet ile hareket ettir (bkz. Denklem 4.9).
- Adım 19: Yeni amaç fonksiyonu $f(x)$ deęerlerini hesapla.
- Adım 20: $deneme1_i$ ve $deneme2_i$ deęişkenlerini güncelle.
- Adım 21: Gerekli istatistikleri güncelle.
- Adım 22: **End For**

- Adım 23: Tüm parçacıklar için $limit1$ değeri ile $deneme1$ değerlerini karşılaştır, gerekirse rassal olarak yeniden oluştur (*Regen*).
- Adım 24: Tüm parçacıklar için $limit2$ değeri ile $deneme2$ değerlerini karşılaştır, gerekirse rassal olarak yeniden oluştur (*ConRegen*).
- Adım 25: Tüm parçacıkların sınırlarını kontrol et.
- Adım 26: Tüm parçacıkların kısıt aşım miktarlarının hesaplanması. (bkz. Denklem 6.5)
- Adım 27: Tüm parçacıklar için $f(x)$ güncellemesi. (bkz. Algoritma 6.1)
- Adım 28: **Until** (durdurma kriteri sağlanıyorsa)

6.3. Karşılaştırma Algoritmaları

Bölüm 6 kapsamında geliştirilen ciEMA, ABC (Karaboga ve Akay 2011), PESO (Muñoz Zavala ve ark. 2005) ve kısıtlı elektromanyetik algoritma (CEM) (Ali ve Golalikhani 2010) algoritmaları ile karşılaştırılmıştır. Karşılaştırmalarda kullanılan çalışmalar, test algoritmaları ve algoritmaların parametreleri aşağıda özetlenmiştir. Çizelge 6.1’de karşılaştırma algoritmalarının sunulduğu çalışmaların 2014 yılı sonu itibari ile aldıkları atıf adetleri özetlenmiştir.

Çizelge 6.1. Karşılaştırma algoritmaları atıf adetleri

Referans	Temel Yapı	Geliştirilen Algoritma	Google Scholar Atıf Sayısı	WOS Atıf Sayısı
Karaboga ve Akay (2011)	ABC	ABC	165	87
Muñoz Zavala ve ark. (2005)	PSO	PESO	76	10
Alikhani ve ark. (2009)	EMA	CEM	23	7

PESO algoritmasında (Muñoz Zavala ve ark. 2005), *Deb* kurallarına bağlı bir seçim yapısı kullanılmıştır. Popülasyon büyüklüğü 50 ve yineleme sayısı 7000 alınmıştır. Başka bir deyişle toplamda 350000 amaç fonksiyonu hesaplaması yapılmaktadır. PESO’da c_1 ve c_2 katsayıları 1’e eşitlenmiş ve atalet momenti olarak (ω) [0.5,1] aralığında uniform rassal olarak kullanılmıştır. Çalışmada tüm eşitlik kısıtları 0.001 hata (ϵ) ile eşitsizliğe çevrilmiştir ($|h(x) - \epsilon \leq 0|$). Tüm test problemleri üzerinde 30 rassal başlangıçlı koşum yaptırılmış ve sonuçlar raporlanmıştır (Muñoz Zavala ve ark. 2005).

ABC algoritmasında (Karaboga ve Akay 2011), her yeni komşuluk oluşturmada kullanılan modifikasyon oranı (MR) 0.8, koloni büyüklüğü (SN) 40 ve maksimum yineleme sayısı (MCN) 6000 olarak alınmıştır. Sonuç olarak ABC algoritmasında toplam 240000 amaç fonksiyonu hesaplanmıştır. Yeniden oluşturma limiti $0,5 * SN * D$ (SN : Popülasyon büyüklüğü, D : Problem boyutu) olarak alınmıştır. ABC algoritmasında da PESO'da olduğu gibi her problem üzerinde 30 koşum yapılmış ve sonuçlar raporlanmıştır (Karaboga ve Akay 2011).

CEM algoritmasında popülasyon büyüklüğü ve yerel arama parametresi olan (I_p) 10 olarak alınmıştır (Ali ve Golalikhani 2010). Uygun çözüm - uygun olmayan çözüm oranını belirleyen β parametresi her problem için farklı olacak şekilde belirlenmiştir. Diğer algoritmalarda olduğu gibi, eşitlik kısıtları 10^{-3} hata payı (ϵ) ile eşitsizlik haline dönüştürülmüştür. Bu çalışmada da sonuçlar 30 rassal koşum kullanılarak raporlanmış ve her koşumda 350000 amaç fonksiyonu hesaplama adımı gerçekleştirilmiştir (Ali ve Golalikhani 2010).

6.4. Kısıtlı Optimizasyon Problemleri

Kısıtlı optimizasyon problemlerinin çözümünde kullanılmak üzere üzerinde bazı güncellemeler yapılarak uyarlanan ciEMA, literatürde sıklıkla kullanılan 13 adet test problemi üzerinde test edilmiştir (Runarsson ve Yao 2000). Bu test setinde doğrusal, doğrusal olmayan veya kuadratik tipte amaç fonksiyonuna sahip çeşitli test problemleri bulunmaktadır. Çizelge 6.2'de, test setindeki problemlere ait temel özellikler verilmiştir. Bu problemlere ait amaç fonksiyonu tipi, doğrusal eşitliklerin sayısı (LE), doğrusal olmayan eşitliklerin sayısı (NE), doğrusal eşitsizliklerin sayısı (LI), doğrusal olmayan eşitsizliklerin sayısı (NI) ve değişken sayıları Çizelge 6.3'de özetlenmiştir. Çizelge 6.3'de ρ değeri, kabul edilebilir bölgedeki çözüm alanının (F), tüm çözüm alanına (S) oranıdır ($\rho = F/S$). Kullanılan test problemlerinin formülasyonları ise Ek 6'da verilmiştir.

Çizelge 6.2. Kısıtlı test problemleri

Problem	Değişken sınırları	x^*	$f(x^*)$
g01	$0 \leq x_i \leq 1 (i = 1, \dots, 9, 13),$ $0 \leq x_i \leq 100 (i = 10, 11, 12)$	(1,1,1,1,1,1,1,1,1,3,3,3,1)	-15
g02	$n = 20, 0 \leq x_i \leq 1 (i = 1, \dots, n)$	$x_i^* = 1/\sqrt{n} (i = 1, \dots, n)$	0,803619
g03	$n = 10, 0 \leq x_i \leq 1 (i = 1, \dots, n)$	$x_i^* = 1/\sqrt{n} (i = 1, \dots, n)$	1
g04	$78 \leq x_1 \leq 102, 33 \leq x_2 \leq 45,$ $27 \leq x_i \leq 45 (i = 3, 4, 5)$	(78, 33, 29.995256, 45, 36.775812)	-30665,539
g05	$0 \leq x_1 \leq 1200, 0 \leq x_2 \leq 1200,$ $-0,55 \leq x_3 \leq 0,55,$ $-0,55 \leq x_4 \leq 0,55$	(679,9453, 1026,067, 0,1188764, -0,3962336)	5126,4981
g06	$13 \leq x_1 \leq 100, 0 \leq x_2 \leq 100$	(14,095, 0,84296)	-6961,81388
g07	$-10 \leq x_i \leq 10 (i = 1, \dots, 10)$	(2,171996, 2,363683, 8,773926, 5,095984, 0,9906548, 1,430573, 1,321644, 9,828726, 8,280092, 8,375927)	24,3062091
g08	$0 \leq x_i \leq 10 (i = 1, 2)$	(1,2279213, 4,2453733)	0,0095825
g09	$-10 \leq x_i \leq 10 (i = 1, \dots, 7)$	(2,330499, 1,951372, -0,4775414, 4,365726, -0,6244870, 1,038131, 1,594227)	680,6300573
g10	$100 \leq x_1 \leq 1000,$ $1000 \leq x_i \leq 10000 (i = 2, 3),$ $10 \leq x_i \leq 1000 (i = 4, \dots, 8)$	(579,19, 1360,13, 5109,92, 182,0174, 295,5985, 217,9799, 286,40, 395,5979)	7049,25
g11	$-1 \leq x_i \leq 1, (i = 1, 2)$	$(\pm 1/\sqrt{2}, 1/2)$	0,75
g12	$0 \leq x_i \leq 10 (i = 1, 2, 3),$	(5, 5, 5)	1
g13	$-2,3 \leq x_i \leq 2,3 (i = 1, 2),$ $-3,2 \leq x_i \leq 3,2 (i = 3, 4, 5)$	(-1,717143, 1,5957091, -0,36413, -0,763645)	0,0539498

Çizelge 6.3. Kısıtlı test problemleri özellikleri

Problem	D	Problem Tipi	$\rho(\%)$	LI	NI	LE	NE
g01	13	Karesel	0,0003	9	0	0	0
g02	20	Doğrusal olmayan	99,9973	1	1	0	0
g03	10	Doğrusal olmayan	0,0026	0	0	0	1
g04	5	Karesel	27,0079	0	6	0	0
g05	4	Doğrusal olmayan	0,0000	2	0	0	3
g06	2	Doğrusal olmayan	0,0057	0	2	0	0
g07	10	Karesel	0,0000	3	5	0	0
g08	2	Doğrusal olmayan	0,8581	0	2	0	0
g09	7	Doğrusal olmayan	0,5199	0	4	0	0
g10	8	Doğrusal	0,0020	3	3	0	0
g11	2	Karesel	0,0973	0	0	0	1
g12	3	Karesel	4,7697	0	9	0	0
g13	5	Doğrusal olmayan	0,0000	0	0	1	2

6.5. Testlerin Sonuçları

Tez çalışması kapsamında geliştirilen ciEMA'da popülasyon büyüklüğü, yapılan ön deneyler sonucunda kararlaştırılmış ve 50 olarak alınmıştır. Uygun çözümler ile uygun olmayan çözümlerin bileşke kuvvet hesaplama adımındaki seçilme olasılıklarını belirleyen δ parametresi 0,55 olarak alınmıştır. Tüm çözümler için yeniden oluşturma adımı (*Regen*) iEMA'da olduğu gibi $m*D$ ve önerilen uygun olmayan çözümler için yeniden oluşturma adımı (*ConRegen*) $2*m*D$ olarak alınmıştır. Çözümler Deb kuralları kullanılarak karşılaştırılmıştır. Diğer çalışmalarda olduğu gibi, eşitlik kısıtları 10^{-3} hata payı (ϵ) ile eşitsizlik haline dönüştürülmüştür. Sonuçlar, her test probleminde 30 rassal koşum yaptırılarak, raporlanmış ve her koşumda 7000 yineleme yapılmıştır. Başka bir deyişle, her koşumda 350000 amaç fonksiyonu hesaplaması yapılmıştır.

ABC, PESO ve CEM algoritmaları ile yapılan çalışmalarda elde edilen sonuçlar Çizelge 6.4, Çizelge 6.5 ve Çizelge 6.6'da verilmiştir. Söz konusu çizelgelerde, optimal sütunu problemin optimal değerini, En iyi, Ortalama ve En kötü sütunları ise sırası ile 30 koşum sonucunda bulunan en iyi, ortalama ve en kötü sonuçları göstermektedir. Sadece ABC çalışmasında 30 koşumun standart sapma (SS) değerlerini raporlanmıştır.

Çizelge 6.7 ise ciEMA ile elde edilen sonuçları göstermektedir. Sonuçların karşılaştırılmasını ve takibini kolaylaştırmak için Çizelge 6.8'de algoritmaların 30 koşum sonucunda elde ettikleri en iyi değerler verilmiş ve bu karşılaştırmalar sonucunda elde edilen sonuçlar ise Çizelge 6.9'da özetlenmiştir. Çizelge 6.9'da, '+' sütunu ciEMA'nın daha iyi sonuç verdiği problem sayısını, '-' sütunu karşılaştırma algoritmasının daha iyi sonuç verdiği problem sayısını, 'Fark yok' sütunu ise iki algoritma arasında farkın olmadığı probleme sayısını göstermektedir. Çizelge 6.9 incelendiğinde, ciEMA'nın diğer algoritmalara genel olarak üstünlük sağladığı görülmektedir. ciEMA dört problemde (g04, g06, g09, g10 ve g13) ABC'den daha üstün sonuçlar vermiştir. CEM algoritmasından dört (g02, g03, g07 ve g09), PESO'dan ise beş problemde (g02, g03, g04, g11 ve g13) daha başarılı performans sergilemiştir. Diğer taraftan, ciEMA ile elde edilen sonuçlar üç problemde ABC ve CEM'in, dört problemde ise PESO'nun gerisinde kalmıştır (ABC: g02, g07, g11, PESO: g05, g07, g09, g10, CEM: g04, g10, g13). Karşılaştırmaların daha sağlıklı yapılması için

algoritmalar ile bulunan ortalama sonuçlar karşılaştırılmıştır. Ortalama sonuçlar Çizelge 6.10 ve sonuçların özetleri ise Çizelge 6.11’de verilmiştir. Çizelge 6.11’de ortalamalardan elde edilen sonuçlar incelendiğinde, ciEMA’nın ABC ile yakın performans gösterdiği (4’er problemde birbirlerine üstünlük sağlamışlar), PESO ve CEM’den ise daha iyi sonuçlar (5 problemde üstün) elde ettiği görülmektedir. Hem algoritmalar ile bulunan en iyi değerler, hem de ortalama değerler birlikte değerlendirildiğinde, ciEMA’nın diğer algoritmalara göre daha üstün olduğu söylenebilir.

Sonuç olarak, tez çalışmasının temel hedefi olan iyileştirilmiş/geliştirilmiş EMA (iEMA) yapısında üzerinde küçük uyarlamalar yapılarak, kısıtlı optimizasyon problemleri üzerinde de başarılı sonuçlar veren bir yapıya (ciEMA) dönüştürülmüştür. ciEMA’da iEMA için yapılan diğer tüm temel güncellemeler kullanılmıştır. Sonuçlar incelendiğinde ciEMA’nın literatürdeki başarılı algoritmalar ile yarışabilen ve hatta daha iyi sonuçlar veren bir algoritma olduğu görülmektedir.

Çizelge 6.4. ABC için raporlanan sonuçlar

Problem	Optimal	En iyi	Ortalama	En kötü	SS
g01	-15,0000	-15,000	-15,000	-15,000	0,000
g02	0,803619	0,803598	0,792412	0,749797	0,012
g03	1,000000	1,000	1,000	1,000	0,000
g04	-30655,539	-30665,484	-30665,539	-30665,539	0,000
g05	5126,4981	5126,484	5185,714	5438,387	75,358
g06	-6961,81388	-6961,814	-6961,813	-6961,805	0,002
g07	24,306209	24,330	24,473	25,190	0,186
g08	0,095825	0,095825	0,095825	0,095825	0,000
g09	680,630057	680,634	680,640	680,653	0,004
g10	7049,3307	7053,904	7224,407	7604,132	133,870
g11	0,750000	0,750	0,750	0,750	0,000
g12	1,000000	1,000	1,000	1,000	0,000
g13	0,053950	0,760	0,968	1,000	0,055

Çizelge 6.5. CEM için raporlanan sonuçlar

Problem	Optimal	En iyi	Ortalama	En kötü
g01	-15,0000	-15,000	-15,000	-15,000
g02	0,803619	-0,623711	-0,517221	-0,4522348
g03	1,000000	-1,00151	-1,00167	-1,00176
g04	-30655,539	-30665,513	-30660,649	-30654,500
g05	5126,4981	5126,4842	5128,6958	5136,6618
g06	-6961,81388	-6961,813	-6961,813	-6961,813
g07	24,306209	25,11276	27,75496	29,93511
g08	0,095825	-0,095825	-0,095825	-0,095825
g09	680,630057	680,8968	681,3511	681,7680
g10	7049,3307	7049,7581	7154,6709	7292,7241
g11	0,750000	0,7499	0,7499	0,7499
g12	1,000000	-1,0000	-1,0000	-1,0000
g13	0,053950	0,053827	0,056314	0,059852

Çizelge 6.6. PESO için raporlanan sonuçlar

Problem	Optimal	En iyi	Ortalama	En kötü
g01	-15,0000	-15,000	-15,000	-15,000
g02	0,803619	0,792608	0,721749	0,614135
g03	1,000000	1,005010	1,005006	1,004989
g04	-30655,539	-30665,538672	-30665,538672	-30655,538672
g05	5126,4981	5126,484154	5129,178298	5148,859414
g06	-6961,81388	-6961,813876	-6961,813876	-6961,813876
g07	24,306209	24,306921	24,371253	24,593504
g08	0,095825	0,095825	0,095825	0,095825
g09	680,630057	680,630057	680,630057	680,630058
g10	7049,3307	7049,459452	7099,101385	7251,396244
g11	0,750000	0,749000	0,74900	0,749000
g12	1,000000	1,000000	1,000000	1,000000
g13	0,053950	0,081498	0,682109	0,997586

Çizelge 6.7. Kısıtlı test problemleri üzerinde ciEMA ile elde edilen sonuçlar

Problem	Optimal	En iyi	Ortalama	En kötü	SS
g01	-15,0000	-15,0000	-15,000	-15,0000	0,000
g02	0,803619	0,6922	0,6100	0,59002	0,032
g03	1,000000	1,0000	1,0000	1,00000	0,000
g04	-30665,4134	-30665,5534	-30665,6810	-30665,7014	0,091
g05	5126,4981	5126,4842	5201,8811	5441,7845	105,128
g06	-6961,81388	-6961,8138	-6961,8114	-6961,8012	0,002
g07	24,306209	24,5117	25,6274	25,9109	0,639
g08	0,095825	0,0958	0,09582	0,09582	0,000
g09	680,630057	680,6397	680,9940	681,1745	0,117
g10	7049,3307	7049,7987	7201,0024	7599,0932	166,992
g11	0,750000	0,7499	0,74991	0,74991	0,000
g12	1,000000	1,0000	1,0000	1,0000	0,000
g13	0,053950	0,0791	0,7114	0,8974	0,328

Çizelge 6.8. En iyi değerlerin karşılaştırması

Problem	Optimal	ABC	CEM	PESO	ciEMA
g01	-15,0000	-15,000	-15,000	-15,000	-15,0000
g02	0,803619	0,803598	-0,623711	0,792608	0,7922
g03	1,000000	1,000	-1,00151	1,005010	1,0000
g04	-30655,539	-30665,484	-30665,513	-30665,538672	-30665,4134
g05	5126,4981	5126,484	5126,4842	5126,484154	5126,4842
g06	-6961,81388	-6961,814	-6961,813	-6961,813876	-6961,8138
g07	24,306209	24,330	25,11276	24,306921	24,5117
g08	0,095825	0,095825	-0,095825	0,095825	0,0958
g09	680,630057	680,634	680,8968	680,630057	680,6397
g10	7049,3307	7053,904	7049,7581	7049,459452	7049,7987
g11	0,750000	0,750	0,7499	0,749000	0,7499
g12	1,000000	1,000	-1,0000	1,000000	1,0000
g13	0,053950	0,760	0,053827	0,081498	0,0791

Çizelge 6.9. Kısıtlı problemler en iyi değerlerin karşılaştırılması özeti

Karşılaştırma çifti	+ (ciEMA daha iyi)	- (Diğer algoritma daha iyi)	Fark yok
ciEMA -ABC	5	3	5
ciEMA -CEM	4	3	7
ciEMA -PESO	5	4	4

Çizelge 6.10. Ortalama değerlerin karşılaştırılması

Problem	Optimal	ABC	CEM	PESO	ciEMA
g01	-15,0000	-15,000	-15,000	-15,000	-15,000
g02	0,803619	0,792412	-0,517221	0,721749	0,7400
g03	1,000000	1,000	-1,00167	1,005006	1,0000
g04	-30655,539	-30665,539	-30660,649	-30665,538672	-30665,6810
g05	5126,4981	5185,714	5128,6958	5129,178298	5127,8811
g06	-6961,81388	-6961,813	-6961,813	-6961,813876	-6961,8124
g07	24,306209	24,473	27,75496	24,371253	24,6274
g08	0,095825	0,095825	-0,095825	0,095825	0,09582
g09	680,630057	680,640	681,3511	680,630057	680,6390
g10	7049,3307	7224,407	7154,6709	7099,101385	7101,0024
g11	0,750000	0,750	0,7499	0,74900	0,74991
g12	1,000000	1,000	-1,0000	1,000000	1,0000
g13	0,053950	0,968	0,056314	0,682109	0,6114

Çizelge 6.11. Kısıtlı problemler ortalama değerlerin karşılaştırılması özeti

Karşılaştırma çifti	+ (ciEMA daha iyi)	- (Diğer algoritma daha iyi)	Fark yok
ciEMA -ABC	4	4	5
ciEMA -CEM	5	3	5
ciEMA -PESO	5	2	6

7. İYİLEŞTİRİLMİŞ ELEKTROMANYETİK ALGORİTMA İLE UYGULAMALAR

Bu bölüm kapsamında, ikili vektörler ile çalışabilen iEMA sürümü (biEMA) ve birleşimsel problemlerin çözümden kullanılabilen iEMA sürümü (combiEMA) geliştirilmiştir. Geliştirilen biEMA ve combiEMA sürümleri ile birlikte tez çalışmasının 6. bölümünde açıklanan ciEMA sürümü sırasıyla öznitelik seçimi problemi, tek makineli çizelgeleme problemi ve mühendislik tasarım problemlerine uygulanarak analiz edilmiştir.

7.1. Öznitelik Seçimi Problemi

Öznitelikler (feature) bir tahmin modelinin girdileri olarak tanımlanabilir. Öznitelik seçim problemi (feature selection problem), büyük ve karmaşık veri kümelerinin sınıflandırılması öncesi sıklıkla kullanılan bir eleme/indirgeme yaklaşımıdır (Guyon ve Elisseeff 2003). Öznitelik seçim problemi kapsamında, ilgisiz özniteliklerin elenerek, en güçlü özniteliklerin seçilmesi, böylece hem tahmin hızının artırılması, hem de veri setinin küçültülmesi hedeflenmektedir (Guyon ve ark. 2006). Verilen n boyutlu ($X = \{x_1, x_2, \dots, x_n\}$) öznitelik kümesinden olası 2^n adette farklı öznitelik alt kümesi seçilebilir ve her öznitelik kümesi (0,1) dizileri ile tanımlanabilir (0: öznitelik seçilmemiş, 1: öznitelik seçilmiş). Öznitelik seçim probleminin amacı, sınıflandırma başarısını düşürmeden en uygun k ($k < n$) adet öznitelikten oluşan öznitelik alt kümesine karar verilmesidir. Öznitelik seçim problemi iyi bilinen bir birleşimsel optimizasyon problemidir ve en uygun öznitelik alt kümesini bulmak, NP-Zor problem tipidir (Maldonado ve Weber 2009). Üç temel öznitelik seçim yöntemi vardır: paketleme (wrappers) yöntemleri, filtreleme (filters) yöntemleri ve melez yöntemler (hybrid) (Hsu ve ark. 2011, Oh ve ark. 2004). Filtreleme yöntemlerinde “a priori” bir kriter temel alınır. Paketleme yöntemleri ise çeşitli öğrenme metotları kullanırlar (yapay sinir ağları, meta sezgisel algoritmalar, vb.). Paketleme yöntemlerinin tahmin performansları, filtreleme yöntemlerine göre daha iyi olsa da, süre olarak filtreleme yöntemlerinin gerisinde kalırlar. Melez yöntemler ise hem filtreleme, hem de paketleme yöntemlerini karma olarak kullanırlar. Literatürde bugüne kadar GA (Kabir ve ark. 2011, Oh ve ark. 2004), AA (Forsati ve ark. 2011), harmonik arama (HA) (Diao ve

Shen 2010, Forsati ve Mahdavi 2010, Mirkhani ve ark. 2013), KKA (Chen ve ark. 2010, Forsati ve ark. 2014, Jensen ve Shen 2005, Kabir ve ark. 2012, Ke ve ark. 2008), PSO (Chuang ve ark. 2011, Huang ve Dun 2008, Unler ve ark. 2011, Wang ve ark. 2007), YAA (Rashedi ve Nezamabadi-pour 2014) ve EMA (Su ve Lin 2011) gibi meta sezgisel algoritmalar ile ayırıcı yöntemler melezleştirilerek öznitelik seçim probleminde kullanılmıştır. Öznitelik seçim problemi ile ilgili literatürün özeti; kaynak, kullanılan meta sezgisel algoritma ve sınıflandırma yöntemi başlıkları altında, Çizelge 7.1’de verilmiştir.

Çizelge 7.1. Öznitelik seçim problemi için literatür taraması

Referans	Meta sezgisel	Sınıflandırma Yöntemi
Oh ve ark. (2004)	Genetik algoritma	k-en yakın komşu
Forsati ve ark. (2011)	Arı algoritmaları	k-en yakın komşu
Diao ve Shen (2010)	Harmonik arama	k-en yakın komşu
Forsati ve Mahdavi (2010)	Harmonik arama	k-en yakın komşu
Kabir ve ark. (2011)	Genetik algoritma	Yapay sinir ağları
Kabir ve ark. (2012)	Karınca kolonisi algoritması	Yapay sinir ağları
Jensen ve Shen (2005)	Karınca kolonisi algoritması	C4.5 (karar ağacı)
Ke ve ark. (2008)	Karınca kolonisi algoritması	Kural tabanlı
Chen ve ark. (2010)	Karınca kolonisi algoritması	Kural tabanlı
Forsati ve ark. (2014)	Karınca kolonisi algoritması	k-en yakın komşu
Kashef ve Nezamabadi-pour (2014)	Karınca kolonisi algoritması	k-en yakın komşu
Wang ve ark. (2007)	Parçacık sürüsü optimizasyonu	Kural tabanlı
Chuang ve ark. (2011)	Parçacık sürüsü optimizasyonu	k-en yakın komşu
Huang ve Dun (2008)	Parçacık sürüsü optimizasyonu	Destek vektör makinesi
Unler ve ark. (2011)	Parçacık sürüsü optimizasyonu	Destek vektör makinesi
Rashedi ve Nezamabadi-pour (2014)	Yerçekimi arama algoritması	k-en yakın komşu
Su ve Lin (2011)	Elektromanyetik algoritma	k-en yakın komşu

Bu bölüm kapsamında, geliştirilen iEMA, sınıflandırma yöntemi ile birleştirilerek öznitelik seçim probleminde uygulanmıştır. Öznitelik seçim probleminde, çözümler ikili (0–1) sistemdeki vektörler ile gösterilmektedir. Bu sebeple, iEMA’da reel sayılardan oluşan parçacık pozisyon vektörlerinin ikili vektörlere dönüştürülmesi gerekmektedir.

Bu sebeple geliştirilen iEMA'nın, ikili vektörler ile çalışabilecek yeni bir sürümü geliştirilmiştir (biEMA). Bu kapsamda bir dönüşüm parametresi tanımlanması ve bu parametreden büyük olan indislere 1 (öznitelik seçilmiş) değerini, küçük indislere ise 0 değerinin (öznitelik seçilmemiş) atanması planlanmıştır. Geliştirilen dönüşüm algoritması Algoritma 7.1'de verilmiştir. Algoritma 7.1'de, m popülasyon büyüklüğünü, D problem boyutunu (öznitelik sayısı) ve *dönüşüm* ise dönüşüm parametresini göstermektedir. biEMA kapsamında, her amaç fonksiyonu hesaplama adımından önce dönüşüm gerçekleştirilecek ve amaç fonksiyonu 0–1 tipine dönüştürülmüş vektör kullanılarak hesaplanacaktır. Çizelge 7.2'de örnek bir dönüşüm işlemi gösterilmiştir. Toplam dokuz adet öznitelikten oluşan bir problem için dönüşüm parametresi (*dönüşüm*) 0,50 olarak alınmış ve dönüşüm algoritması sonucunda ikinci, beşinci, altıncı ve dokuzuncu öznitelikler seçilmiş, diğer öznitelikler seçilmemiştir. Örnek parçacık için amaç fonksiyonu değeri, söz konusu ikili vektör kullanılarak hesaplanacaktır. Bölüm 4'de detayları verilen iEMA'nın amaç fonksiyonu hesaplama adımı öncesine, Algoritma 7.1 eklenerek oluşturulan optimizasyon yöntemi, biEMA kısaltması ile bu bölüm kapsamında kullanılmıştır.

```

For i= 1 to m
  For j =1 to D
    If  $x_i^j \geq \textit{dönüşüm}$  then
       $x_i^j = 1$ 
    Else  $x_i^j = 0$ 
    end if
  end for
end for

```

Algoritma 7.1. Öznitelik seçim problemi için dönüşüm

Çizelge 7.2. biEMA için örnek bir dönüşüm (*dönüşüm=0.50*)

Öznitelik:	1	2	3	4	5	6	7	8	9
Parçacık:	0,44	0,56	0,21	0,03	0,82	0,58	0,37	0,11	0,81
İkili vektör:	0	1	0	0	1	1	0	0	1

Geliştirilen biEMA'nın performansı, ilk olarak literatürden alınan standart test problemleri üzerinde analiz edilmiştir. Daha sonra ise etkinliği ve verimliliği kanıtlanmış biEMA ile bir gerçek hayat öznitelik seçim problemi uygulaması yapılmıştır.

Deneyler kapsamında ilk olarak, dokuz adet veri kümesi kullanılmıştır. Bu veri kümeleri literatürde sıklıkla kullanılan UCI (University of California, Machine Learning Repository) veri tabanından alınmıştır (<http://archive.ics.uci.edu/ml/datasets.html>, 2014). Kullanılan veri kümeleri Çizelge 7.3'de özetlenmiş olup, 2–26 arasında çıktı kategorileri, 4–57 adet arasında özniteliğe sahip kümelere oluşmaktadır. Bu veri kümeleri, literatürde bir çok çalışmada test problemi olarak kullanılmıştır (Forsati ve ark. 2014, Kashef ve Nezamabadi-pour 2013, 2014, Su ve Lin 2011).

Çizelge 7.3. Veri setleri karakteristikleri

Veri Kümesi	Çıktı kategori adedi	Öznitelik adedi	Örnek adedi
Abalone	11	8	3 842
Glass	6	9	214
Iris	3	4	150
Letter	26	16	20000
Shuttle	7	9	58000
Spambase	2	57	4 601
Tae	3	5	151
Wisconsin	2	9	683
Yeast	9	8	1 484

Bu bölüm kapsamında tez çalışmasının diğer bölümlerinde olduğu gibi çalışma ortamı olarak MATLAB (R2011a) kullanılmış ve Intel Core i5 CPU (1.30 GHz) işlemcili ve 4 GB belleğe sahip bilgisayarda deneyler gerçekleştirilmiştir. biEMA'nın performansını analiz edebilmek için elde edilen sonuçlar, gelişmiş ikili KKA (ABACO) (Kashef ve Nezamabadi-pour 2014), ikili PSO (BPSO) (Chuang ve ark. 2011), ikili genetik

algoritma (BGA) (Rashedi ve Nezamabadi-pour 2014), iyileştirilmiş ikili yerçekimi arama algoritması (IBGSA) (Rashedi ve Nezamabadi-pour 2014), yerel aramalı karınca kolonisi algoritması (ACO_h) (Kashef ve Nezamabadi-pour 2013), yerel aramasız karınca kolonisi algoritması (ACO) (Kashef ve Nezamabadi-pour 2013) ve ikili KKA (bACO) (Kashef ve Nezamabadi-pour 2013) algoritmaları ile karşılaştırılmıştır.

Geliştirilen biEMA ve diğer temel algoritmaların karşılaştırılması için sınıflandırma doğruluğu (*SD*), kesin isabet (*KI*), erişim isabeti (*EI*) ve öznitelik azaltım oranı (*ÖAO*) gibi performans ölçütleri kullanılmıştır. *KI* ölçütü, *i*. kategori için doğru sınıflandırılmış verilerin *i*. kategoriye atanmış tüm verilere oranını; *EI* ise doğru şekilde kategori *i*'ye atanmış verilerin gerçek kategori *i*'deki verilere oranını göstermektedir. *KI* ve *EI* ölçütleri Denklem 7.1 ve Denklem 7.2'de verilmiştir. Çalışmalarda kullanılan bir diğer temel performans ölçütü *SD* ise Denklem 7.3'de verilmiştir. Algoritmaların koşumu sonucunda özniteliklerin hangi oranda azaltıldığını gösteren *ÖAO* ölçütü, Denklem 7.4'de incelenebilir. Denklem 7.4'de *n* toplam öznitelik adedini, *p* ise seçilmiş öznitelik adedini göstermektedir. *ÖAO* değerinin bire yaklaşması, o oranda özniteliğin elenerek seçilmediği anlamına gelmektedir. Denklemlerde, *TP_i* *i*. kategori için doğru sınıflandırılmış örnek adedini, *FP_i* ise *i*. kategori için yanlış sınıflandırılmış örnek adedini göstermektedir. *TN_i* ve *FN_i* ise sırasıyla *i*. kategori dışındaki kategoriler için doğru ve yanlış sınıflandırılmış örnek adedini tutmaktadır.

$$KI_i = \frac{TP_i}{TP_i + FP_i} \quad (7.1)$$

$$EI_i = \frac{TP_i}{TP_i + FN_i} \quad (7.2)$$

$$SD = \frac{\text{Toplam doğru sınıflandırılmış veri adedi}}{\text{Toplam veri adedi}} \quad (7.3)$$

$$ÖAO = \frac{n-p}{n} \quad (7.4)$$

Bu bölüm kapsamında kullanılan karşılaştırma algoritmalarının sonuçları, Kashef ve Nezamabadi-pour (2014)'un çalışmasından alınmış ve biEMA ile elde edilen sonuçlar

söz konusu çalışmaya benzer şekilde raporlanmıştır. Tarafsız bir karşılaştırma yapabilmek için diğer algoritmalar ile benzer şekilde biEMA için popülasyon büyüklüğü (m) ve durdurma kriteri olarak kullanılan maksimum yineleme sayısı 50 olarak atanmıştır. biEMA'nın diğer parametresi olan *limit* ise, Bölüm 4'te tanımlandığı şekilde $limit = m * D$ olarak atanmıştır. Burada D problem boyutunu, başka bir deyişle veri kümesindeki öznitelik adedini göstermektedir. biEMA'da kullanılan dönüşüm parametresi ise 5 olarak atanmış ve tüm değişkenler $[0, 10]$ aralığında sınırlandırılmıştır. Kullanılan karşılaştırma algoritmalarının kontrol parametreleri ve detayları için Kashef ve Nezamabadi-pour (2014) çalışması incelenebilir.

Her veri kümesi üzerinde biEMA ile 30 adet rassal koşum gerçekleştirilmiş ve her koşumda verilerin %60'ı rassal olarak seçilerek, eğitim için kullanılmıştır. Verilerin kalan %40'ının üzerinde ise biEMA'nın performansı test edilmiştir. Çalışma kapsamında, sınıflandırma yöntemi diğer test algoritmalarında olduğu gibi k -en yakın komşu kullanılmış ve $k = 1$ olarak alınmıştır. Başka bir deyişle, test aşamasında her veri seçilmiş olan öznitelikler kullanılarak bulunan en yakın komşusunun ait olduğu kategoriye atanmıştır. K -en yakın komşu yönteminin detayları için Cover ve Hart (1967) çalışması incelenebilir.

Çizelge 7.4'de biEMA ve diğer algoritmalar ile veri kümeleri üzerinde elde edilen SD değerleri özetlenmiştir. Çizelgede her veri kümesi için elde edilmiş en iyi SD değeri koyu olarak gösterilmiştir. Çizelge 7.4'ün son satırında ise tüm veri kümeleri için ortalama SD değerleri incelenebilir. Sonuçlar incelendiğinde, biEMA'nın diğer yöntemlere göre daha üstün performans sergilediği görülmektedir. biEMA ile seçilen öznitelikler kullanılarak dokuz veri kümesinin altısında (Abalone, Glass, Iris, Letter, Shuttle ve Wisconsin) en iyi sınıflandırma performansına erişilmiştir. Karşılaştırma algoritmaları arasında biEMA'ya en yakın performansı ABACO göstermiş ve Spambase, Tae ve Yeast problemlerinde biEMA'ya karşı üstünlük sağlamıştır. biEMA sadece Tae veri setinde ACO yönteminin gerisinde kalmış (ACO: 0,578, ve biEMA: 0,577) ve Spambase veri setinde ise IBGSA metoduyla benzer performans sergilemiş (biEMA ve IBGSA: 0,922), geri kalan tüm problemlerde ise biEMA, ABACO dışındaki tüm yöntemlerden daha iyi sonuçlar elde edilmiştir. Ortalama SD değerleri

incelendiğinde, biEMA'nın 0,761 sınıflandırma doğruluğu ile çalıştığı ve ABACO (0,760) yönteminden az da olsa daha iyi performans gösterdiği anlaşılmaktadır.

Çizelge 7.4. Veri setleri üzerinde elde edilen sınıflandırma doğrulukları (*SD*)

Veri Kümesi	ABACO	BPSO	BGA	IBGSA	ACO _n	ACO	bACO	biEMA
Abalone	0,244	0,241	0,241	0,241	0,241	0,238	0,243	0,248
Glass	0,758	0,734	0,733	0,738	0,728	0,713	0,743	0,758
Iris	0,976	0,965	0,967	0,969	0,971	0,963	0,967	0,979
Letter	0,861	0,825	0,837	0,860	0,856	0,806	0,859	0,864
Shuttle	0,998	0,998	0,997	0,998	0,998	0,998	0,998	0,998
Spambase	0,923	0,900	0,906	0,922	0,913	0,901	0,919	0,922
Tae	0,583	0,567	0,556	0,556	0,559	0,578	0,558	0,577
Wisconsin	0,976	0,968	0,975	0,977	0,974	0,968	0,975	0,978
Yeast	0,524	0,507	0,513	0,515	0,512	0,509	0,515	0,523
Ortalama SD	0,760	0,745	0,747	0,753	0,750	0,742	0,753	0,761

Diğer performans ölçütleri olan *KI*, *EI* ve *ÖAO* sonuçları ise Çizelge 7.5'de özetlenmiştir. Çizelgenin son sütununda ölçüt toplamları verilmiş ve her ölçüt için en iyi değerler koyu olarak işaretlenmiştir. Çizelge 7.5 incelendiğinde ABACO ve biEMA'nın çok yakın *EI* ve *KI* sonuçları elde ettiği ve diğer algoritmaların her iki ölçüt için biEMA ve ABACO'nun gerisinden kaldığı görülmektedir. ABACO, toplam *KI* değeri olarak 6,733 ile ilk sırada yer alırken (biEMA: 6,731), biEMA *EI* ölçütünde (*EI* = 6,728) tüm algoritmalara karşı üstünlük sağlamıştır. Öznitelik seçim problemlerinde kaç adet öznitelik kullanılarak sınıflandırma yapıldığı da önemli bir performans ölçütüdür. Bu sebeple, *ÖAO* değerleri incelendiğinde BGA yönteminin 4,348 toplam *ÖAO* değeri ile diğer algoritmalara göre daha az öznitelik kullanarak sınıflandırma yaptığı fakat toplam *EI* (6,513) ve *KI* (6,612) ve ortalama *SD* (0,747) değerleri incelendiğinde vasat bir sınıflandırma performansı gösterdiği anlaşılmaktadır. biEMA kullanılarak toplam *ÖAO* değeri 3,204'e erişilmiştir. Söz konusu *ÖAO* değeri toplam *EI* (6,728) ve *KI* (6,731) ve ortalama *SD* (0,761) ölçüt değerleri ile birlikte değerlendirildiğinde geliştirilen algoritma ile iyi bir denge yakalandığı görülmektedir. biEMA az da olsa daha iyi sınıflandırma performansı gösterdiği ABACO yöntemine göre toplam *ÖAO* değerinde (3,192) üstünlük sağlamıştır. Başka bir deyişle, biEMA uygulanarak veri setlerinin sahip olduğu özniteliklerin sadece %35,6'sı (3,204/9,000) seçilmiş ve seçilen öznitelik kullanılarak yüksek sınıflandırmada performansına erişilmiştir. Bu sonuç aynı zamanda biEMA kullanılarak yapılacak bir sınıflandırma

işleminin diğer yöntemlere göre daha az işlem yüküne de sahip olduğunu göstermektedir.

Çizelge 7.5. Algoritmalar için *KI*, *EI* ve *ÖAO* değerleri

Algoritma	Ölçüt	Problemler									Toplam
		Abalone	Glass	Iris	Letter	Shuttle	Spambase	Tae	Winconsin	Yeast	
ABACO	<i>KI</i>	0,222	0,743	0,978	0,867	0,912	0,920	0,589	0,973	0,529	6,733
	<i>EI</i>	0,223	0,736	0,975	0,862	0,927	0,920	0,585	0,976	0,521	6,725
	<i>ÖAO</i>	0,337	0,299	0,437	0,343	0,541	0,431	0,357	0,384	0,063	3,192
BPSO	<i>KI</i>	0,221	0,708	0,965	0,863	0,970	0,896	0,578	0,963	0,466	6,630
	<i>EI</i>	0,217	0,673	0,965	0,858	0,910	0,895	0,571	0,966	0,464	6,519
	<i>ÖAO</i>	0,438	0,367	0,450	0,348	0,531	0,477	0,380	0,406	0,206	3,603
BGA	<i>KI</i>	0,225	0,719	0,969	0,853	0,925	0,903	0,567	0,971	0,480	6,612
	<i>EI</i>	0,222	0,682	0,968	0,839	0,894	0,901	0,556	0,975	0,476	6,513
	<i>ÖAO</i>	0,373	0,339	0,388	0,341	0,561	0,454	0,441	0,971	0,480	4,348
IBGSA	<i>KI</i>	0,212	0,672	0,961	0,840	0,911	0,906	0,548	0,966	0,472	6,488
	<i>EI</i>	0,212	0,668	0,962	0,835	0,928	0,905	0,546	0,964	0,474	6,494
	<i>ÖAO</i>	0,542	0,315	0,372	0,420	0,381	0,465	0,283	0,386	0,129	3,293
ACOh	<i>KI</i>	0,221	0,677	0,971	0,861	0,933	0,910	0,568	0,971	0,476	6,588
	<i>EI</i>	0,219	0,682	0,971	0,857	0,921	0,910	0,573	0,973	0,485	6,591
	<i>ÖAO</i>	0,344	0,240	0,313	0,313	0,400	0,461	0,410	0,328	0,113	2,922
ACO	<i>KI</i>	0,222	0,684	0,963	0,812	0,931	0,898	0,589	0,964	0,464	6,527
	<i>EI</i>	0,219	0,654	0,962	0,808	0,914	0,896	0,580	0,966	0,476	6,475
	<i>ÖAO</i>	0,388	0,372	0,313	0,313	0,472	0,461	0,410	0,504	0,389	3,622
bACO	<i>KI</i>	0,223	0,730	0,967	0,865	0,944	0,917	0,568	0,973	0,509	6,696
	<i>EI</i>	0,222	0,695	0,967	0,863	0,930	0,917	0,562	0,974	0,507	6,637
	<i>ÖAO</i>	0,387	0,347	0,459	0,332	0,565	0,465	0,363	0,387	0,089	3,394
biEMA	<i>KI</i>	0,222	0,745	0,979	0,869	0,911	0,917	0,594	0,972	0,522	6,731
	<i>EI</i>	0,225	0,738	0,974	0,868	0,919	0,924	0,578	0,978	0,524	6,728
	<i>ÖAO</i>	0,341	0,301	0,435	0,349	0,544	0,427	0,367	0,381	0,059	3,204

biEMA ile gerçek hayat öznitelik seçim problemi uygulaması:

Tez çalışması kapsamında geliştirilen ve verimliliği kanıtlanmış biEMA yöntemi ile Uludağ Üniversitesi Tıp Fakültesi Kalp Damar Cerrahisi bölümü ile ortak bir uygulama çalışması yapılmıştır. Bu bağlamda, karotis arter stenozunun tedavi yönteminin tahmini için bir karar destek sistemi tasarlanmış ve sistemin özniteliklerinin seçimi için biEMA kullanılmıştır. Gerçekleştirilen çalışma kapsamında, 284 adet hasta dosyası ve konsey kararları incelenmiş ve tasarlanan sistemin öznitelikleri Çizelge 7.6'da görülebileceği şekilde oluşturulmuştur. Detayları Ek 7'de verilmiş olan veriler 139 adet kadın ve 145 adet erkek hastadan dengeli bir şekilde toplanmıştır. 138 adet hasta için ilaç tedavisi/takip (-1) kararı verilmiş, kalan hastalar için ise (146 hasta) cerrahi girişim (+1) kararı alınmıştır.

Çizelge 7.6. Tasarlanan sistemin öznitelikleri

Kod	Özellik	Yorum
Ö1	Yaş	100'e bölünmüş şekilde kullanılacak.
Ö2	Cinsiyet	Kadın = 0, Erkek = 1
Ö3	Lezyon tipi	Tip A = 0, Tip B = 1, Tip C = 3, Tip D = 4
Ö4	Aneztezi duyarlılığı	Düşük = 0, Orta = 1 Yüksek = 2
Ö5	Distal yatak	Yok = 0, Var = 1
Ö6	Emboli oranı	100'e bölünmüş şekilde kullanılacak.
Ö7	LDL kolesterol	Normal = 0, Normalin üstünde = 1, BH = 2, Yüksek = 3, Çok yüksek = 4
Ö8	Sigara kullanımı	Yok = 0, Var = 1
Ö9	Sigara kullanma geçmişi	Yok = 0, Var = 1
Ö10	Hipertansiyon	Yok = 0, Var = 1
Ö11	Kan basıncı	Normal = 0, Pre HTN = 1, Stage I = 2, Stage II = 3
Ö12	Koroner arter hastalığı geçmişi	Yok = 0, Var = 1
Ö13	Celebra vasküler olay geçmişi	Yok = 0, Var = 1
Ö14	Stenoz derecesi	Düşük = 1, Yüksek = 2
Ö15	Diabetes mellitus	Yok = 0, Var = 1
Ö16	Başka periferik hastalık geçmişi	Yok = 0, Var = 1
Ö17	Aile geçmişi	Yok = 0, Var = 1
Ö18	İlaç tedavisi	Yok = 0, Var = 1
S	Tedavi kararı	İlaç tedavisi/takip = -1, Cerrahi girişim = +1

Yapılan çalışmada, biEMA ile daha önce yapılan karşılaştırma analizlerine benzer şekilde hem popülasyon büyüklüğü (m), hem de maksimum yineleme sayısı 50 olarak atanmış, $limit (m * D)$ ve $dönüşüm=5$ olarak alınmıştır. Benzer şekilde veri kümesi üzerinde biEMA ile 30 rassal koşum gerçekleştirilmiş ve her koşumda verilerin %60'ı rassal olarak seçilerek eğitim setini, kalan %40'ı ise test setini oluşturmuştur. Sınıflandırma yöntemi olarak k-en yakın komşu kullanılmış ve $k = 1$ olarak alınmıştır. Benzer şekilde, uygulama çalışmasında çalışma ortamı olarak MATLAB (R2011a) kullanılmış ve Intel Core i5 CPU (1.30 GHz) işlemcili ve 4 GB belleğe sahip bilgisayarda çalışma gerçekleştirilmiştir.

Uygulama çalışması kapsamında, biEMA ile k-en yakın komşu yöntemi (biEMA + 1NN), geleneksel EMA ile k-en yakın komşu (bEMA + 1NN) ve sadece veri üzerinde 1NN ile sınıflandırma yapıldığı yöntem (1NN) karşılaştırılmıştır. Çalışmada geleneksel EMA'nın 0-1 tipindeki vektörler ile çalışabilmesi için algoritmaya sadece dönüşüm algoritması eklenmiş (bEMA) ve o hali ile koşumlar gerçekleştirilmiştir.

Elde edilen sonuçlar Çizelge 7.7'de verilmiştir. Çizelgede ilk üç sütunda, 30 koşum ortalaması olarak SD , EI ve KI değerleri verilmiş ve diğer sütunlarda ise seçilen öznitelikler özetlenmiştir. 30 rassal koşumun yarısı veya daha fazlasında seçilmiş olan bir özellik Çizelge 7.7'de seçilmiş olarak (x) işaretlenmiştir. 1NN yöntemi öncesinde ise herhangi bir öznitelik seçim algoritması kullanılmadığından tüm özniteliklerin seçildiği varsayılmıştır. Sonuçlar incelendiğinde, biEMA+1NN yönteminin, hem bEMA+1NN'e hem de 1NN yöntemine SD , EI ve KI ölçütlerinde üstünlük sağlamıştır. biEMA +1NN $SD=0,904$ değeri elde ederken, bEMA+1NN $SD=0,840$, 1NN ise $SD=0,798$ değeri elde etmiştir. biEMA, KI ve EI ölçütlerinde de benzer şekilde diğer iki yöntemle göre daha iyi performans sergilemiştir. biEMA ile koşumların çoğunda Ö1, Ö3, Ö5, Ö6, Ö7, Ö8, Ö12 ve Ö17 olmak üzere toplam sekiz adet öznitelik seçilmiştir. 1NN'de beklenildiği gibi tüm öznitelikler kullanılırken, bEMA ise onbir adet öznitelik seçmiştir. Çizelge 7.7'deki veriler incelendiğinde biEMA ile daha az öznitelik seçimi gerçekleştirilerek, daha üstün sınıflandırma performansı gösterilmesi sağlanmıştır.

Çizelge 7.7. Uygulama çalışması sonuçları ve seçilen öznelilikler

Yöntem	Ölçüt			Seçilen Özellikler																		
	<i>SD</i>	<i>KI</i>	<i>EI</i>	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	
INN	0,798	0,663	0,711	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
bEMA + INN	0,840	0,701	0,709	x	x	x			x	x	x	x		x			x	x		x		
biEMA + INN	0,904	0,867	0,896	x		x		x	x	x	x				x						x	

7.2. Kısıtlı mühendislik tasarım problemleri

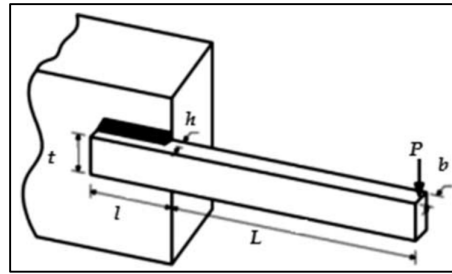
Tez çalışması kapsamında geliştirilen iEMA yapısının kısıtlı optimizasyon problemleri için uyarlanmış sürümü olan ciEMA (bkz. Bölüm 6.2), daha detaylı incelenme için literatürden alınan üç adet mühendislik tasarım problemine uygulanmış ve literatürde sunulan algoritmalar ile karşılaştırmalı olarak analiz edilmiştir. ciEMA (i) kaynak yapılmış kiriş problemine (welded beam problem), (ii) yay problemine (tension/compression spring problem) ve (iii) hız indirgeyici tasarım problemine (speed reducer design problem) uygulanmıştır. Tüm problemler için literatürden alınan iki farklı model kullanılmış ve toplam altı adet kısıtlı mühendislik tasarım problemi üzerinde testler gerçekleştirilmiştir. Tüm deneylerde önceki çalışmalarda karar verilen ciEMA yapısı kullanılmış ve tüm kontrol parametreleri aynı olacak şekilde korunmuştur (bkz. Bölüm 6.2). Diğer çalışmalara benzer şekilde her problem tipinde 30 adet rassal koşum gerçekleştirilmiş ve bulunan çözümlerin ortalaması raporlanmıştır. Aynı şekilde diğer algoritmaların benzer şekilde durdurma kriteri olarak 100000 amaç fonksiyonu hesaplama adedi kullanılmıştır. Karşılaştırma analizleri için kullanılan toplam 23 adet algoritma ise aşağıda verilmiştir:

- Doğrusal olmayan tamsayı ve kesikli programlama (NLIDP) (Sandgren 1990)
- Kombine GA (CGS) (Deb ve Goyal 1996)
- DGA (Lampinen ve Zelinka 1999)
- Sosyal davranışlı optimizasyon algoritması (SBO) (Ray ve Liew 2003)
- Geliştirilmiş PSO (IPSO) (He ve ark. 2004)
- $(\mu + \lambda)ES$: $(\mu + \lambda)$ -Evrimsel strateji (Mezura Montes ve Coello Coello 2005)
- Kısıtlı sayısal optimizasyon için DGA (DDE) (Mezura-Montes ve ark. 2006)
- Dinamik stokastik seçimli DGA (DEDS) (Zhang ve ark. 2008)
- Adaptif melez evrimsel algoritma (HEAA) (Wang ve ark. 2009)
- Seviye karşılaştırmalı DGA (DELC) (Wang ve Li 2010)
- Melez PSO-DE algoritması (PSO-DE) (Liu ve ark. 2010)
- P-Çözücü (Melez PSO) (Kayhan ve ark. 2010)
- ABOA (Gandomi ve ark. 2011)

- Lig Şampiyonluk algoritması (LCA) (Husseinzadeh Kashan 2011)
- Yarasa algoritması (BA) (Gandomi ve ark. 2013b)
- Penaltı fonksiyonlu DGA (DEP) (Melo ve Carosio 2012)
- Esnek izin teknikli GA (GAFAT) (Zhao ve ark. 2012)
- Modifiye edilmiş DGA (MDE) (Mohamed ve Sabry 2012)
- Modifiye edilmiş HA (MHA) (Mun ve Cho 2012)
- İyileştirilmiş ABC (UABC) (Brajevic ve Tuba 2013)
- GKA (Gandomi ve ark. 2013a)
- Mayın patlatma algoritması (MBA) (Sadollah ve ark. 2013)
- Melez guguk kuşu genetik algoritma (HCSGA) (Kanagaraj ve ark. 2013)

7.2.1. Kaynak yapılmış kiriş problemi

Kaynak yapılmış kiriş problemi (Welded beam problem) literatürde sıklıkla kullanılan kısıtlı mühendislik tasarım problemidir. Problem yapısı Şekil 7.1’de gösterilmiştir. Kaynak yapılmış kiriş problemi, kesme gerilimi τ , kirişteki bükülme gerilimi σ , çubuktaki burkulma gerilimi P_c , kiriş sehimini δ ve bunlara ek olarak yön sınırlamaları altında kirişin maliyetinin minimize edildiği mühendislik tasarım problemidir. Söz konusu problemde h , l , t ve b karar değişkenleri sırasıyla x_1 , x_2 , x_3 ve x_4 olarak kodlanarak kullanılmıştır. Çalışma kapsamından kaynak yapılmış kiriş problemi iki durum altında incelenmiş, 1. durum, Denklem 7.5–7.10 (Ray ve Liew 2003), 2. durum ise Denklem 7.11–7.18 (Cagnina ve ark. 2008) arasında açıklanmıştır. Modelde kullanılan sabit değerler ise Ek 8’de verilmiştir.



Şekil 7.1. Kaynak yapılmış kiriş problemi yapısı (Kanagaraj ve ark. 2013)

1. Durum:

$$\text{Min } f(x) = (1 + C_1)x_1^2x_2 + C_2x_3x_4(L + x_2) \quad (7.5)$$

$$g_1: \tau_d - \tau \geq 0 \quad (7.6)$$

$$g_2: \sigma_d - \sigma \geq 0 \quad (7.7)$$

$$g_3: x_4 - x_1 \geq 0 \quad (7.8)$$

$$g_4: P_c - P \geq 0 \quad (7.9)$$

$$g_5: 0.25 - \delta \geq 0 \quad (7.10)$$

$$\text{Burada; } \tau = \sqrt{(\tau')^2 + (\tau'')^2 + \frac{x_2\tau'\tau''}{\sqrt{0,25(x_2^2+(x_1+x_3)^2)}}}; \sigma = \frac{504000}{x_3^2x_4};$$

$$\delta = \frac{2,1952}{x_3^3x_4}; \tau'' = \frac{6000(14+0,5x_2)\sqrt{0,25(x_2^2+(x_1+x_3)^2)}}{2\left\{0,707x_1x_2\left(\frac{x_2^2}{12}\right)+0,25(x_1+x_3)^2\right\}}; P_c = 64746(1 - 0,0282346x_3)x_3x_4^3$$

$$\tau' = \frac{6000}{\sqrt{2}x_1x_2}; 0,125 \leq x_1 \leq 5; 0,1 \leq x_2, x_3 \leq 10; 0,1 \leq x_4 \leq 5$$

2. Durum:

$$\text{Min } f(x) = (1 + C_1)x_1^2x_2 + C_2x_3x_4(L + x_2) \quad (7.11)$$

$$g_1: \tau_d - \tau \geq 0 \quad (7.12)$$

$$g_2: \sigma_d - \sigma \geq 0 \quad (7.13)$$

$$g_3: x_4 - x_1 \geq 0 \quad (7.14)$$

$$g_4: P_c - P \geq 0 \quad (7.15)$$

$$g_5: 0,125 - x_1 \geq 0 \quad (7.16)$$

$$g_6: 0.25 - \delta \geq 0 \quad (7.17)$$

$$g_7: P_c - 6000 \geq 0 \quad (7.18)$$

$$\text{Burada; } \tau = \sqrt{(\tau')^2 + (\tau'')^2 + \frac{x_2\tau'\tau''}{\sqrt{0,25(x_2^2+(x_1+x_3)^2)}}}; \sigma = \frac{504000}{x_3^2x_4};$$

$$\delta = \frac{2,1952}{x_3^3x_4}; \tau'' = \frac{6000(14+0,5x_2)\sqrt{0,25(x_2^2+(x_1+x_3)^2)}}{2\left\{0,707x_1x_2\left(\frac{x_2^2}{12}\right)+0,25(x_1+x_3)^2\right\}}; P_c = 64746(1 - 0,0282346x_3)x_3x_4^3$$

$$\tau' = \frac{6000}{\sqrt{2}x_1x_2}; 0,125 \leq x_1 \leq 5; 0,1 \leq x_2, x_3 \leq 10; 0,1 \leq x_4 \leq 5$$

Problem üzerinde elde edilen sonuçlar ise Çizelge 7.8’de incelenebilir. Çizelgede ciEMA, 1. durum için DLEC, LCA, GAFAT ve HCSGA algoritmaları ile, 2. durum için ise DELC, LCA, GAFAT, HCSGA, UABC, MBA ve MHA yöntemleri ile karşılaştırılmıştır. Çizelge 7.8’de elde edilen ortalama karar değişkeni ve amaç fonksiyonu değerleri raporlanmıştır. 1. durum için ciEMA, DELC, LCA ve GAFAT yöntemlerine karşın üstünlük göstermiş, HCSGA ile ise benzer sonuçlara erişmiştir (HCSGA: 2,38095651, ciEMA: 2,38095649). 2. durum sonuçları incelendiğinde ise, ciEMA’nın amaç fonksiyonu değeri bazında, MHA dışındaki tüm algoritmalarından daha iyi performans gösterdiği görülmektedir (ciEMA: 1,72485194).

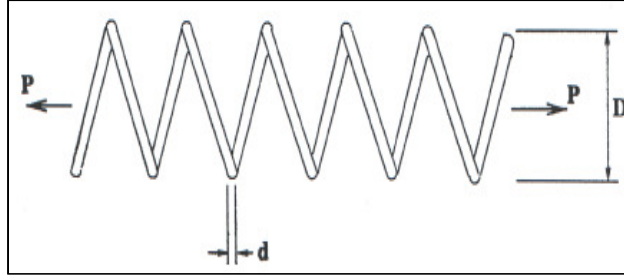
Çizelge 7.8. Kaynak yapılmış kiriş problemi sonuçları

Algoritma	Karar Değişkenleri				Amaç fonk. $f(x)$
	x_1	x_2	x_3	x_4	
1. Durum					
DELC	0,2443689758	6,21751971	8,2914713905	0,2443689758	2,38095658
LCA	0,2443689758	6,21751971	8,2914713904	0,2443689758	2,38095658
GAFAT	0,2443689758	6,21751971	8,29141713905	0,2443689758	2,38095658
HCSGA	0,2443689758	6,21752015	8,29147177	0,244368953	2,38095649
ciEMA	0,2443689758	6,21752015	8,29147177	0,244368961	2,38095651
2. Durum					
DELC	0,20572964	3,47048867	9,03662391	0,20572964	1,724852
LCA	0,2057296397	3,47048866	9,0366239103	0,2057296397	1,72485230
GAFAT	0,205729964	3,47048867	9,03662391	0,20572964	1,724852
HCSGA	0,2057529620	3,47048892	9,03662434	0,20572962	1,72485224
UABC	0,205730	3,470489	9,036624	0,205730	1,724852
MBA	0,205729	3,470493	9,036626	0,205729	1,724853
MHA	0,206704	3,252355	8,9759952	0,208518	1,707009
ciEMA	0,2057324622	3,47048889	9,03662431	0,20572962	1,72485194

7.2.2. Yay tasarım problemi

Yay tasarım problemi (Tension/Compression spring problem) Şekil 7.2’de gösterilmiştir (Akay 2009). Söz konusu problem, minimum burkulma, kesme gerilmeleri, gerilme frekansı ve çap boyutu gibi kısıtlar altında yayın ağırlığının minimizasyonunu hedeflemektedir. Modelde kullanılan x_1 , x_2 ve x_3 karar değişkenleri sırasıyla çap d , ana sarım çapı D , ve aktif sarım sayısı N ’ye karşılık gelmektedir. Yay tasarım problemi de benzer şekilde iki durum altında incelenmiş ve sonuçlar analiz edilmiştir. 1. durumun modeli Denklem 7.19–7.23 ile, 2. durumun modeli ise Denklem

7.24–7.32 ile verilmiştir. Modellerde kullanılan sabit değişkenlerin değerleri ise Ek 8’de incelenebilir.



Şekil 7.2. Yay problemi yapısı

1. durum:

$$\text{Min } f(x) = (x_3 + 2)x_2x_1^2 \quad (7.19)$$

$$g_1: 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0 \quad (7.20)$$

$$g_2: \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0 \quad (7.21)$$

$$g_3: 1 - \frac{140,45x_1}{x_2^2x_3} \leq 0 \quad (7.22)$$

$$g_4: \frac{x_1 + x_2}{1,5} - 1 \leq 0 \quad (7.23)$$

2. durum:

$$\text{Min } f(x) = \frac{\pi^2x_2x_1^2(x_3+2)}{4} \quad (7.24)$$

$$g_1: \frac{8C_fF_{max}x_2}{\pi x_1^3} - S \leq 0 \quad (7.25)$$

$$g_2: l_f - l_{max} \leq 0 \quad (7.26)$$

$$g_3: d_{min} - x_1 \leq 0 \quad (7.27)$$

$$g_4: x_2 - D_{max} \leq 0 \quad (7.28)$$

$$g_5: 3,0 - \frac{x_2}{x_1} \leq 0 \quad (7.29)$$

$$g_6: \sigma_p - \sigma_{pm} \leq 0 \quad (7.30)$$

$$g_7: \sigma_p + \frac{(F_{max} - F_p)}{K} + 1,05(x_3 + 2)x_1 - l_f \leq 0 \quad (7.31)$$

$$g_8: \sigma_p - \frac{F_{max} - F_p}{K} \leq 0 \quad (7.32)$$

$$0,2 \leq x_1 \leq 1; 0,6 \leq x_2 \leq 3; 1 \leq x_3 \leq 70$$

Burada;

$$C_f = \frac{4\left(\frac{x_2}{x_1}\right)^{-1}}{4\left(\frac{x_2}{x_1}\right)^{-4}} + \frac{0,615x_1}{x_2}; K = \frac{Gx_1^4}{8x_3x_2^2}; \sigma_p = \frac{F_p}{K}; l_f = \frac{F_{max}}{K} + 1.05(x_3 + 2)x_1$$

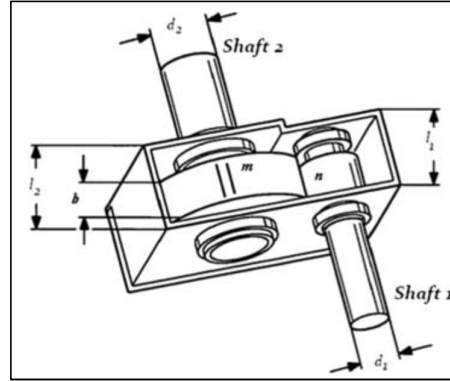
Söz konusu problem üzerinde elde edilen sonuçlar ise Çizelge 7.9'da raporlanmıştır. ciEMA çalışması kapsamında, 1. durum için NLIDP, CGS, DGA, HCSGA ve IPSO yöntemleri; 2. durum için ise DELC, Melez-PSO, LCA, MHA, DEP, GAFAT, YA, HCSGA, UABC ve MBA algoritmaları ile karşılaştırmalı analiz yapılmıştır. Sonuçlar incelendiğinde, ciEMA, kaynak yapılmış kiriş probleminde görüldüğü gibi GSGA yöntemine benzer bir performans sergilenmiş, diğer karşılaştırma algoritmalarından ise daha kaliteli sonuçlar elde etmiştir. 2. Durum için elde edilen sonuçlar incelendiğinde, 30 koşum ortalamasında ciEMA, eriştiği $(x_1, x_2, x_3, x_4) = (0,20573246, 3,47048889, 9,03662431, 0,20572962)$ karar değişkeni vektörü ile diğer algoritmalar arasında en iyi performans gösteren yöntem olmuştur.

Çizelge 7.9. Yay problemi sonuçları

Algoritma	Karar Değişkenleri			Amaç fonk.
	x_1	x_2	x_3	$f(x)$
1. Durum				
NLIDP	0,283	1,180701	10	2,7995
CGS	0,283	1,226	9	2,665
DGA	0,283	1,223041010	9	2,65856
HCSGA	0,283	1,4000032	6	2,2113301
IPSO	0,283	1,223041010	9	2,65856
ciEMA	0,283000	1,399929	9	2,232104
2. Durum				
DELC	0,05168906	0,356717741	11,2889656	0,01266523
Melez-PSO	0,0151686	0,356650	11,292950	0,0126652
LCA	0,05168906	0,356717753	11,2889412	0,01266523
MHA	0,05171296	0,35729285	11,2553284	0,0126652
DEP	0,05168906	0,35671789	11,2889567	0,01266523
GAFAT	0,05168906	0,356717738	11,2889658	0,01266523
YA	0,05169	0,35673	11,2885	0,01267
HCSGA	0,05000000	0,3744328	8,546569	0,00987245
UABC	0,051691	0,356769	11,285988	0,012665
MBA	0,051656	0,355940	11,344665	0,012665
ciEMA	0,50198233	0,36318001	8,22019041	0,00960133

7.2.3. Hız indirgeyici tasarım problemi

Çalışma kapsamında incelenen mühendislik tasarım problemlerinin sonuncusu hız indirgeyici tasarım problemidir (speed reducer problem). Problem, Şekil 7.3'te açıklanmıştır. Söz konusu problem de amaç indirgeyici ağırlığının minimizasyonudur. Diğer problemlerde olduğu gibi hız indirgeyici problem de iki farklı durum için incelenmiş ve bu durumların modelleri aşağıda verilmiştir (1. durum: Denklem 7.32–7.41, 2. durum: Denklem 7.42–7.51). Problemden kullanılan karar değişkenleri ise; yüzey genişliği, dişli sayısı, birinci şaftın rulmanları arasındaki uzunluk, birinci şaftın çapı ve ikinci şaftın çapı olarak açıklanmıştır. Bu problemi diğer tasarım problemlerinden ayıran nokta ise dişli sayısının her zaman tamsayı olma koşuludur. Bu bağlamda ciEMA yöntemi ile dişli sayısına ait karar değişkeni her zaman en yakın tamsayı değerine yuvarlanmış ve amaç fonksiyonu hesaplama adımında bu şekilde kullanılmıştır.



Şekil 7.3. Hız indirgeyici tasarım problemi yapısı (Kanagaraj ve ark. 2013)

1. Durum:

$$\begin{aligned} \text{Min } f(b, m, z, l_1, l_2, d_1, d_2) = & 0,7854bm^2(3,3333z^2 + 14,933z - 43,0934) - \\ & 1,508b \times (d_1^2 + d_2^2) + 7,477(d_1^3 + d_2^3) + 0,7854(l_1d_1^2 + l_2d_2^2) \end{aligned} \quad (7.32)$$

$$g_1: \frac{27}{bm^2z} - 1 \leq 0 \quad (7.33)$$

$$g_2: \frac{397,5}{bm^2z^2} \quad (7.34)$$

$$g_3: \frac{1,93}{mzl_1^3d_1^4} - 1 \leq 0 \quad (7.35)$$

$$g_4: \frac{1,93}{mzl_1^3d_2^4} - 1 \leq 0 \quad (7.36)$$

$$g_5: \frac{\sqrt{\left(\frac{745l_1}{mz}\right)^2 + 1,69 \times 10^6}}{110d_1^3} - 1 \leq 0 \quad (7.37)$$

$$g_6: \frac{\sqrt{\left(\frac{745l_1}{mz}\right)^2 + 157,5 \times 10^6}}{85d_2^3} - 1 \leq 0 \quad (7.38)$$

$$g_7: \frac{mz}{40} - 1 \leq 0 \quad (7.39)$$

$$g_8: \frac{5m}{b-1} - 1 \leq 0 \quad (7.40)$$

$$g_9: \frac{b}{12m} - 1 \leq 0 \quad (7.41)$$

$$2,6 \leq b \leq 3,6; 0,7 \leq m \leq 0,8; 17 \leq z \leq 28; 7,3 \leq l_1 \leq 8,3$$

$$7,3 \leq l_2 \leq 8,3; 2,9 \leq d_1 \leq 3,9; 5,0 \leq d_2 \leq 5,5$$

2. durum:

$$\begin{aligned} \text{Min } f(b, m, z, l_1, l_2, d_1, d_2) = & 0,7854bm^2(3,3333z^2 + 14,933z - 43,0934) - \\ & 1,508b \times (d_1^2 + d_2^2) + 7,477(d_1^3 + d_2^3) + 0,7854(l_1d_1^2 + l_2d_2^2) \end{aligned} \quad (7.42)$$

$$g_1: \frac{27}{bm^2z} - 1 \leq 0 \quad (7.43)$$

$$g_2: \frac{397,5}{bm^2z^2} \quad (7.44)$$

$$g_3: \frac{1,93}{mzl_1^3d_1^4} - 1 \leq 0 \quad (7.45)$$

$$g_4: \frac{1,93}{mzl_1^3d_2^4} - 1 \leq 0 \quad (7.46)$$

$$g_5: \frac{\sqrt{\left(\frac{745l_1}{mz}\right)^2 + 1,69 \times 10^6}}{110d_1^3} - 1 \leq 0 \quad (7.47)$$

$$g_6: \frac{\sqrt{\left(\frac{745l_1}{mz}\right)^2 + 157,5 \times 10^6}}{85d_2^3} - 1 \leq 0 \quad (7.48)$$

$$g_7: \frac{mz}{40} - 1 \leq 0 \quad (7.49)$$

$$g_8: \frac{5m}{b-1} - 1 \leq 0 \quad (7.50)$$

$$g_9: \frac{b}{12m} - 1 \leq 0 \quad (7.51)$$

$$2,6 \leq b \leq 3,6; 0,7 \leq m \leq 0,8; 17 \leq z \leq 28; 7,3 \leq l_1 \leq 8,3$$

$$7,8 \leq l_2 \leq 8,3; 2,9 \leq d_1 \leq 3,9; 5,0 \leq d_2 \leq 5,5$$

Problem üzerinde elde edilen sonuçlar Çizelge 7.10'da verilmiştir. Çalışma kapsamında geliştirilen ciEMA hız indirgeyici tasarım problemi üzerinde, 1. durum için SBA, DEDS, HEAA, DELC, LCA, GAFAT, UABC, MBA ve HCSGA ile karşılaştırılmış, 2. durum için ise $(\mu + \lambda)$ ES, MDE, DDE, PSO-DE, ABOA, LCA, GKA ve HCSGA ile karşılaştırılarak analiz edilmiştir. ciEMA, 1. durum için DEDS, DLEC, LCA, GAFAT, UABC ve HCSGA algoritmaları ile eşit sonuçlar elde ederken, diğer taraftan da, SBO, HEAA ve MBA algoritmalarından daha iyi sonuçlar elde etmiştir.

Çizelge 7.10. Hız indirgeyici tasarım problemi sonuçları

Algoritma	Karar Değişkenleri							Amaç fonk. $f(x)$
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
1. Durum								
SBO	3,5000068	0,7	17	7,32760	7,71532175	3,35026702	5,28665450	2994,744241
DEDS	3,5	0,7	17	7,3	7,71531991	3,35021466	5,28665446	2994,471066
HEAA	3,5	0,7	17	7,30042	7,71537744	3,35023096	5,28666369	2994,499109
DELC	3,50000	0,7	17	7,30000	7,71531991	3,35021466	5,28665446	2994,471066
LCA	3,500000	0,7	17	7,30000	7,71531991	3,35021466	5,28665446	2994,471066
GAFAT	3,5	0,7	17	7,3	7,7153199	3,35021466	5,2866544	2994,471066
UABC	3,5	0,7	17	7,3	7,715320	3,350215	5,286654	2994,471066
MBA	3,5	0,7	17	7,30003	7,715772	3,350218	5,286654	2994,482453
HCSGA	3,5	0,7	17	7,3	7,71531991	3,35021399	5,28665446	2994,471066
ciEMA	3,5	0,7	17	7,3	7,71540144	3,35021409	5,28665446	2994,471066
2. Durum								
$(\mu + \lambda)$ ES	3,506163	0,7	17	7,46018	7,962143	3,3629	5,3090	3025,005
MDE	3,50001	0,7	17	7,30015	7,800027	3,350221	5,286685	2996,356689
DDE	3,500010	0,7	17	3,50001	7,800027	3,350221	5,286685	2996,356689
PSO-DE	3,50000	0,7	17	7,300	7,80000000	3,350214	5,2866832	2996,348165
ABOA	3,5015	0,7	17	7,6050	7,8181	3,3520	5,2875	3000,9810
LCA	3,5	0,7	17	7,3	7,80000000	3,35021466	5,28668322	2996,348164
GKA	3,5015	0,7	17	7,6050	7,8181	3,3520	5,2875	3000,9810
HCSGA	3,5	0,7	17	7,6050	7,8181	3,35	5,26866	2996,348176
ciEMA	3,5	0,7	17	7,6050	7,8181	3,3509	5,2701	2996,350116

7.3. Tek makineli çizelgeleme problemi

Tez çalışması kapsamında geliştirilen ve çeşitli testler ile başarısı kanıtlanan iEMA, son olarak çok kullanılan bir birleşimsel problem olan tek makineli erken bitme–geç kalma çizelgeleme problemine uygulanmıştır. Tek makineli erken bitme–geç kalma çizelgeleme probleminde amaç, toplam erken bitme–geç kalma cezalarının minimizasyonudur. Söz konusu problemde, n adet birbirinden bağımsız J_1, J_2, \dots, J_n işlerinin, herhangi bir anda maksimum bir adet iş işleyebilen bir makinede çizelgenmesi hedeflenmiştir ve işlerin bölünme ve/veya durdurulmasına izin verilmez (Sourd ve Kedad-Sidhoum 2005). Oluşturulan herhangi bir çizelgede, J_j 'ninci işin erken bitme süresi E_j veya geç kalma süresi T_j sırasıyla $E_j = \max\{0, d_j - C_j\}$ ve $T_j = \max\{0, C_j - d_j\}$ olarak ifade edilir. Burada, p_j J_j 'ninci işin işlem süresi, d_j J_j 'ninci işin teslim zamanı ve C_j ise J_j 'ninci işin bitiş süresidir. Toplam erken bitme ve geç kalma cezaları, amaç fonksiyonunda $\sum_{j=1}^n (h_j E_j + w_j T_j)$ şeklinde ifade edilir. Burada h_j ve w_j sırasıyla J_j 'ninci işin erken bitme ve geç kalma ceza katsayılarıdır. Çalışma kapsamında ele alınan tek makineli çizelgeleme probleminin temel hedefi işin tam olarak ihtiyaç duyulduğu zamanda hazırlanmasıdır. Bu bağlamda, söz konusu problem sanayide sıklıkla kullanılan tam zamanında üretim stratejileri ile örtüşen bir yapıdadır. Bu bağlamda, problemdeki erken bitme cezasına stok tutma maliyeti, geç kalma cezasına ise yok satma maliyeti olarak bakılabilir (Chang ve ark. 2009). Tek makineli erken bitme çizelgeleme problemi, literatürde kabul görmüş çizelgeleme gösteriminde $1 \mid (h_j E_j + w_j T_j)$ olarak tanımlanmıştır. Ele alınan problemin, daha basit bir sürümü olan tek makineli ağırlıklandırılmış, geç kalma probleminin $(1 \mid w_j T_j)$ NP–Zor olduğu bilinmektedir (Lenstra ve ark. 1977). Bu sebeple, tek makineli erken bitme geç kalma probleminin de NP–Zor tipinde olduğu söylenebilir (Sourd ve Kedad-Sidhoum 2008).

Tek makineli erken bitme geç kalma çizelgeleme problemi, literatürde bir çok araştırmacı tarafından incelemiştir. Ow ve Morton (1989) problemin çözümü için farklı erken bitme–geç kalma çizelgeleme kuralları ile birlikte “filter beam” arama yöntemini kullanmışlardır. Abdul-Razaq ve Potts (1988) problem için dinamik bir formülasyon önermiş ve birçok test problemi için alt sınırları oluşturmuşlardır. Li (1997) yerel komşuluk arama yöntemi ile birlikte iki tane dal-sınır yöntemi geliştirmiştir. Çalışmada,

geliştirilen yöntemin üç bin adet işten oluşan problemleri verimli ve etkin bir şekilde çözebildiği kanıtlanmıştır. Başka bir dal-sınır çalışması ise Chang (1999) tarafından gerçekleştirilmiştir. Söz konusu çalışmada, tüm ceza katsayısı değerlerinin bir olarak atandığı ağırlıklandırılmamış sürüm üzerinde uygulamalar yapılmıştır. Son yıllarda yapılan çalışmalar sonucu, problemin çözümü için başarılı bir dal-sınır algoritması (Sourd ve Kedad-Sidhoum 2005, 2008), sezgisel algoritmalar (Valente ve Alves 2005), GA (Chang ve ark. 2008), EMA (Chang ve ark. 2009) ve komşuluk arama (Kedad-Sidhoum ve Sourd 2010) gibi yöntemler literatürde görülmektedir.

Tez çalışması kapsamında geliştirilen iEMA bu bölümde birleşimsel bir problem olan çizelgeleme problemine, combiEMA kısaltılmış adı ile uygulanmıştır. Çalışma kapsamında yeni bir amaç fonksiyonu hesaplama (AFH) prosedürü geliştirilmiş ve bu kapsamda, rassal anahtar (RA) yöntemi kullanılarak gerçek sayılardan oluşan pozisyon vektörleri birer iş sırasına dönüştürülmüştür. Son olarak, ortaya çıkan iş sıralarının amaç fonksiyon değerleri hesaplanmıştır.

RA yöntemi, ilk olarak (Bean 1994) tarafından GA'da yapılan çaprazlama işlemleri sonucunda ortaya çıkan sıralama tekrarı sorunun önüne geçebilmek için kullanılmıştır. Birçok farklı araştırmacı RA'yı gerçek değerlerden, birleşimsel problemlerdeki çözüm kümelerine dönüştürmek için kullanmışlardır (Chang ve ark. 2009, Gonçalves ve Resende 2012, Mendes ve ark. 2009, Noronha ve ark. 2011, Snyder ve Daskin 2006, Yurtkuran ve Emel 2010, 2014). RA yönteminde, herhangi n -boyutlu bir düzlem için (n -iş problemi), pozisyon vektörlerinin içerdiği gerçek değerler azalmayan bir şekilde sıralanır. Sıralanan bu değerlerin orijinal indisleri ise bir çözümü, yani iş sırasını gösterir. Çizelge 7.11'de 10 boyutlu bir problem (10 adet iş) için RA yöntemi çözümlenmesi gösterilmiştir. Örnekte verilen parçacık i 'nin pozisyon vektöründeki en küçük değer değeri $\min(x_i^8) = -4,19$ olduğundan, RA uygulandıktan sonra sekiz numaralı indis (iş) ilk sırada olacak ve diğer indisler de aynı şekilde azalmayan değerlerine göre sıralanacaktır. Çizelgede verilen örnek için iş sırası 8-5-6-1-2-3-7-4-10-9 şeklindedir. Çalışma kapsamında kullanılan amaç fonksiyonu hesaplama prosedürü Algoritma 7.2'de özetlenmiştir. Algoritmada x'_i , RA uygulanmış bir parçacık için iş sırasını ifade etmektedir.

Çizelge 7.11. Rassal anahtar yöntemi

Rassal anahtar öncesi:										
İndeks k	1	2	3	4	5	6	7	8	9	10
k, indis değeri: x_i^k	0,23	0,45	1,55	3,42	-3,10	-0,17	2,91	-4,19	3,89	3,71
Rassal anahtar sonrası:										
Sıralanmış indis değeri:	-4,19	-3,10	-0,17	0,23	0,45	1,55	2,91	3,42	3,71	3,89
Sıralanmış indis:	8	5	6	1	2	3	7	4	10	9

```

For i= 1 to m
   $x_i' = RA(x_i)$  //  $x_i$  için RA uygula.
   $f(x_i) = f(x_i')$  //  $x_i'$  için amaç fonksiyonu hesapla.
end for

```

Algoritma 7.2. Kombinatoryel problemler için AFH prosedürü

Bu bölümde, tez çalışması kapsamında geliştirilen iEMA, AFH prosedürü ile birleştirilerek (combiEMA) çok bilinen bir birleşimsel problem olan tek makineli erken bitme-geç kalma problemine uygulanmıştır. Bu kapsamda geliştirilen yöntemin etkinliği ve başarısı ise literatürden alınan test problemleri üzerinde analiz edilmiştir. Kullanılan test problemleri, ilk olarak Sourd ve Kedad-Sidhoum (2005) tarafından literatüre sunulmuştur (<http://www-poleia.lip6.fr/~safia/et>, 2014). Kullanılan test problemleri 20 ile 50 iş arasında değişen iş adetlerinden oluşmakta ve farklı problem karakteristiklerine (geç kalma, erken bitme oranlarına) sahiptirler. Ek olarak, her iş seti içerisinde farklı problem karakteristiklerine sahip 1275 problem bulunmaktadır. Çalışma kapsamında geliştirilen combiEMA, yinelemeli yerel arama (ILS) (Kedad-Sidhoum ve Sourd 2010) ve sezgisel (HEUR) (Sourd ve Kedad-Sidhoum 2005) yöntemleri ile karşılaştırılmış ve performansı analiz edilmiştir.

Genel olarak meta sezgisel algoritmaların kontrol parametrelerinin, algoritma performansı üzerinde ciddi etkileri bulunmaktadır (López-Ibáñez ve ark. 2011). Bu sebeple, yapılan detaylı ön analizler sonucunda, combiEMA'nın kontrol parametrelerinden popülasyon büyüklüğü $m = 50$ olarak alınmışken, durdurma kriteri olarak maksimum yineleme sayısı kullanılmış ve diğer karşılaştırma algoritmalarındaki ile benzer şekilde 5000 olarak kabul edilmiştir. combiEMA'nın diğer bir parametresi olan yeniden oluşturma adımı (*limit*) ise $m * D$ iterasyon olacak şekilde tanımlanmıştır. Çalışmada her test problemi için 30 adet rassal koşum yaptırılmış ve sonuçlar diğer iki

çalışmadaki ile benzer şekilde raporlanmıştır. Elde edilen sonuçlar Çizelge 7.12’de özetlenmiştir. Çizelgede combiEMA ile erişilen sonuçlar, HEUR sürümleri (HEUR1, HEUR10 ve HEURn) ve ILS sürümleri (ILS, ILS2 ve ILS3) ile birlikte verilmiştir. Çizelgede her problem kümesi ($n = 20, n = 30, n = 40$ ve $n = 50$) için sonuçlar optimal değerden yüzde ortalama sapma (Ort. Dev. (%)), optimal değerlerden maksimum sapma (Maks. Dev. (%)) ve optimal sonuca erişilen problem adedinin toplam problem adedine yüzdesi (En iyi (%)) olarak verilmiştir. Sonuçlar incelendiğinde, combiEMA’nın verimli ve etkin bir yöntem olduğu söylenebilir. combiEMA, $n = 50$ problem seti için ILS yöntemlerinin gerisinde kalsa da, HEUR sezgiseline karşı tüm problem setlerinde üstünlük sağlamıştır. Sonuçlar incelenirken, combiEMA’nın diğer algoritmalar gibi özel olarak tek makineli çizelgeleme problemi için geliştirilmediği ve birçok farklı birleşimsel problemde problem bağımsız olarak uygulanabilecek bir yapı olduğu da göz ardı edilmemelidir. Tüm bunlar birlikte değerlendirildiğinde, combiEMA’nın, iEMA’nın birleşimsel problemler için geliştirilmiş başarılı bir uyarlaması olduğu söylenebilir.

Çizelge 7.12. Tek makineli çizelgeleme problemi için karşılaştırma sonuçları

Problem	Ölçüt	HEUR1	HEUR10	HEURn	ILS	ILS2	ILS3	combiEMA
n=20	Ort. Dev. (%)	1,06	0,12	0,08	*	*	*	0,05
	Maks. Dev. (%)	21,9	8,76	8,76	*	*	*	2,61
	En iyi (%)	62,1	89,4	91,9	*	*	*	92,5
n=30	Ort. Dev. (%)	1,42	0,20	0,09	*	*	*	0,08
	Maks. Dev. (%)	20,5	10,4	10,0	*	*	*	7,98
	En iyi (%)	38,4	75,8	86,3	*	*	*	88,5
n=40	Ort. Dev. (%)	1,46	0,26	0,10	*	*	*	0,1
	Maks. Dev. (%)	22,3	6,47	5,53	*	*	*	4,09
	En iyi (%)	23,5	62,6	80,6	*	*	*	87,1
n=50	Ort. Dev. (%)	1,59	0,28	0,11	0,01	0,03	0,01	0,10
	Maks. Dev. (%)	16,2	7,63	4,10	*	*	1,29	3,71
	En iyi (%)	17,4	55,7	76,4	*	*	92,9	81,4

*Söz konusu problemler için sonuçlar raporlanmamıştır.

8. SONUÇ VE YORUMLAR

Optimizasyon problemlerinin en uygun şekilde çözüme kavuşturulabilmesi için bugüne kadar birçok farklı yöntem geliştirilmiş ve kullanılmıştır. Günümüzde, erişilen çözüm değeri kadar çözüme erişilinceye kadar harcanan süre de çok önemli bir hale gelmiştir. Gerçek hayatta karşılaşılan çoğu optimizasyon problemi kabul edilebilir süreler içerisinde optimal olarak çözülememektedir. Bu bağlamda, her zaman optimal sonuca erişmeyi hedefleyen kesin optimizasyon yöntemlerinin bu tip problemler üzerinde uygulanabilirliği neredeyse imkansızdır. Son yıllarda karmaşık optimizasyon problemlerini daha kısa çözüm süreleri içerisinde yakın optimal olarak çözebilen yöntemler üzerindeki çalışmalar yoğunlaşmış ve literatüre bir çok farklı meta sezgisel önerilmiştir.

Bu tez çalışması kapsamında ele alınan elektromanyetik algoritma (EMA) meta sezgisel algoritmalar sınıfı içerisinde bulunan bir optimizasyon yöntemidir. EMA temel olarak elektrik yüklü parçacıkların birbirlerine uyguladıkları itme-çekme kuvvetlerini taklit ederek optimale doğru ilerlemeyi hedefleyen bir meta sezgisel algoritmadır. EMA son yıllarda giderek popülerleşen ve birçok farklı tipteki optimizasyon probleminin çözümünde kullanılan bir yöntemdir.

Bu tez çalışmasının amacı, EMA'nın güçlü ve zayıf yanlarına ilişkin detaylı analizler sonucunda daha da geliştirilmesi ve istatistiksel olarak başarısı kanıtlanmış yeni bir sürümünün literatüre önerilmesidir. Ek olarak, iyileştirilen EMA'nın üzerinde çeşitli uyarlamalar yapılarak, birçok farklı optimizasyon problemine uygulanması ve bu sayede önerilen yöntemin aynı zamanda farklı problem tiplerinin çözümü için kullanılabilen esnek bir arama algoritması olduğunun kanıtlanmasına da çalışılmıştır.

Tezin 1. bölümünde optimizasyon, optimizasyon problemleri ve optimizasyon yöntemleri ile ilgili genel bilgiler verilmiş ve optimizasyon problemlerinin yapısı ve tipleri, çözüm yöntemleri incelenmiştir. Tez çalışmasının amacı, kapsamı ve literatüre katkısı bu bölüm çerçevesinde özetlenmiştir. EMA yönteminin detaylı yapısı, adımları ve sözde kodu Bölüm 2' özetlenmiştir. Bölüm 2'de aynı zamanda EMA ile ilgili detaylı bir literatür taraması yapılmıştır. Bölüm 3'de temel EMA ile yapılan karşılaştırma

çalışmaları verilmiştir. Bu bölüm kapsamında, EMA, parçacık sürüsü optimizasyonu, diferansiyel gelişim ve yapay arı kolonisi algoritmalarının temel sürümleri ile karşılaştırılmış ve performansı analiz edilmiştir. Yapılan karşılaştırma çalışmalarının tarafsız ve doğru sonuçlar verebilmesi için farklı yapılarda birçok test problemi kullanılmış ve tüm testler özdeş ortamlarda gerçekleştirilmiştir. Bu bölümde elde edilen sonuçlar ile EMA ve diğer algoritmalar arasındaki temel farklar ortaya konmuş ve EMA'nın geliştirilmeye açık yönleri ve geliştirme önerileri ortaya çıkarılmıştır. 4. bölüm kapsamında, 3. bölümde elde edilen sonuçlar kullanılarak geliştirme önerileri ve farklı seviyeleri açıklanmış ve deneysel tasarım çalışması gerçekleştirilmiştir. Analizler sonucunda, EMA için en verimli yapıya karar verilmiş ve iyileştirilmiş EMA (iEMA) (bkz. Bölüm 4.4) ortaya çıkarılmıştır. iEMA ile gerçekleştirilen karşılaştırma çalışmaları 5. bölüm kapsamında verilmiştir. Tez çalışmasının 6. bölümünde ise geliştirilen iEMA'nın kısıtlı optimizasyon problemlerini çözebilen sürümü ciEMA (bkz. Bölüm 6.2) sunulmuştur. 7. bölümde ise iEMA ile farklı tiplerdeki optimizasyon problemleri üzerinde uygulamalar gerçekleştirilmiştir. Bu bölüm kapsamında, ilk olarak gerçek sayı yerine ikili (binary) sayı sisteminde çalışabilen bir iEMA (biEMA) geliştirilmiş ve öznitelik seçimi problemi üzerinde uygulanmıştır. İkinci olarak, ciEMA iyi bilinen mühendislik tasarım problemlerinden kaynak yapılmış kiriş problemi (welded beam problem), yay problemi (tension/compression spring) ve hız indirgeyici tasarım problemi (speed reducer problem) üzerinde çalıştırılmış ve literatürden alınan yöntemler ile detaylı olarak karşılaştırılmıştır. Son olarak, iEMA'ya yeni bir amaç fonksiyonu hesaplama prosedürü eklenerek, algoritmanın birleşimsel problemlere de uygulanabilmesi sağlanmış (combiEMA) ve bu kapsamda tek makineli erken bitme-geç kalma çizelgeleme problemi üzerinde testler yapılmıştır.

Hem temel EMA yapısının güçlü ve zayıf yönlerinin belirlenmesi başlığında gerçekleştirilen karşılaştırma çalışmaları, hem de iEMA ile yapılan karşılaştırma çalışmalarında kullanılan tüm algoritmalar eşit şartlar altında çalıştırılmıştır. Bu sebeple, elde edilen sonuçların üzerinde yapılan istatistik testlerin geçerliliği en üst düzeydedir. Başka bir deyişle, tez kapsamında iyileştirilen ve diğer problem tiplerine uyarlanan sürümlerine omurga görevi gören iEMA'nın başarısı, her durum için en kapsamlı testlere dayandırılarak, kanıtlanmıştır.

Bu çalışmanın literatüre birçok farklı katkısı olacağı değerlendirilmektedir. Özellikle iEMA ve iEMA sürümleri ile ilgili çalışmaların her biri, bağımsız bir bilimsel yayın olacak şekilde tasarlanmış olup, literatüre katkıları aşağıdaki gibi açıklanabilir:

- Kısıtsız optimizasyon problemlerinde çok başarılı performans gösteren ve literatürde çokça atf almış yöntemlere karşı istatistiksel olarak anlamlı bir üstünlük sağlayan iyileştirilmiş bir EMA (iEMA) yapısı,
- Küçük uyarlamalar sonucunda kısıtlı optimizasyon problemleri çözümünde kullanılabilen, literatürde daha önce kullanılan yöntemlere göre daha başarılı sonuçlar üreten bir iEMA (ciEMA) yapısı,
- İkili sistem vektörler ile çalışabilen bir iEMA yapısının geliştirilmesi (biEMA) ve başarısı kanıtlanmış biEMA ile gerçekleştirilen bir gerçek hayat problemi uygulaması,
- Birleşimsel problemler için iEMA'dan esinlenilerek geliştirilen ve başarısı bir birleşimsel problem üzerinde kanıtlanan combiEMA yöntemidir.

Geliştirilen algoritmalar ile gelecekte yapılması planlanan çalışmalar kapsamında, ilk olarak iEMA'nın çok amaçlı optimizasyon problemlerine uygulanması planlanmaktadır. İkinci olarak, iEMA kontrol parametrelerinin arama esnasında adaptif olarak güncellenmesi ve diğer temel meta sezgiseller ile melezleştirilen iEMA performansının analiz edilmesi düşünülmektedir. Son olarak, combiEMA'nın daha karmaşık birleşimsel optimizasyon problemleri üzerinde uygulanması planlanmaktadır.

KAYNAKLAR

- Abdul-Razaq, T., Potts, C. 1988.** Dynamic Programming State-Space Relaxation for Single-Machine Scheduling. *Journal of the Operational Research Society*, 141-152.
- Abdullah, S., Turabieh, H., McCollum, B., McMullan, P. 2012.** A hybrid metaheuristic approach to the university course timetabling problem. *Journal of Heuristics*, 18(1): 1-23.
- Abed, I. A., Koh, S., Sahari, K. S. M., Jagadeesh, P., Tiong, S. 2014.** Optimization of the Time of Task Scheduling for Dual Manipulators using a Modified Electromagnetism-Like Algorithm and Genetic Algorithm. *Arabian Journal for Science and Engineering*, 39(8): 6269-6285.
- Akay, B. 2009.** Nümerik Optimizasyon Problemlerinde Yapay Arı Kolonisi Algoritmasının Performans Analizi. *Doktora Tezi*, Bilgisayar Mühendisliği Anabilim Dalı, Erciyes Üniversitesi, Kayseri, Türkiye.
- Akay, B., Karaboga, D. 2012.** A modified artificial bee colony algorithm for real-parameter optimization. *Information Sciences*, 192(1): 120-142.
- Alatas, B. 2010.** Chaotic bee colony algorithms for global numerical optimization. *Expert Systems with Applications*, 37(8): 5682-5687.
- Alba, E. 2005.** Parallel metaheuristics: a new class of algorithms. John Wiley & Sons, New Jersey, USA, 521 pp.
- Alba, E., Luque, G., Alba, E. 2005.** Measuring the performance of parallel metaheuristics. *Parallel metaheuristics: a new class of algorithms*, 4743.
- Ali, M., Golalikhani, M. 2010.** An electromagnetism-like method for nonlinearly constrained global optimization. *Computers & Mathematics with Applications*, 60(8): 2279-2285.
- Alikhani, M. G., Javadian, N., Tavakkoli-Moghaddam, R. 2009.** A novel hybrid approach combining electromagnetism-like method with Solis and Wets local search for continuous optimization problems. *Journal of Global Optimization*, 44(2): 227-234.
- Arora, J. 2004.** Introduction to optimum design. Academic Press, California, USA, 620 pp.
- Bäck, T., Schwefel, H.-P. 1993.** An overview of evolutionary algorithms for parameter optimization. *Evolutionary computation*, 1(1): 1-23.
- Banks, A., Vincent, J., Anyakoha, C. 2007.** A review of particle swarm optimization. Part I: background and development. *Natural Computing*, 6(4): 467-484.
- Banks, A., Vincent, J., Anyakoha, C. 2008.** A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications. *Natural Computing*, 7(1): 109-124.

- Barr, R. S., Golden, B. L., Kelly, J. P., Resende, M. G., Stewart Jr, W. R. 1995.** Designing and reporting on computational experiments with heuristic methods. *Journal of Heuristics*, 1(1): 9-32.
- Bean, J. C. 1994.** Genetic algorithms and random keys for sequencing and optimization. *ORSA journal on computing*, 6(2): 154-160.
- Bean, J. C., Hadj-Alouane, A. 1992.** A dual genetic algorithm for bounded integer programs. *Ann Arbor*, 100148109-42117.
- Birattari, M., Paquete, L., Strutzle, T., Varrentrapp, K. 2001.** Classification of Metaheuristics and Design of Experiments for the Analysis of Components Tech. Rep. AIDA-01-05.
- Birbil, Ş. İ., Fang, S.-C. 2003.** An electromagnetism-like mechanism for global optimization. *Journal of global optimization*, 25(3): 263-282.
- Birbil, Ş. İ., Fang, S.-C., Sheu, R.-L. 2004.** On the convergence of a population-based global optimization algorithm. *Journal of global optimization*, 30(2-3): 301-318.
- Björck, A. 1996.** Numerical methods for least squares problems. Siam, Philadelphia, USA, 401 pp.
- Blum, C., Puchinger, J., Raidl, G. R., Roli, A. 2011.** Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing*, 11(6): 4135-4151.
- Blum, C., Roli, A. 2003.** Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys (CSUR)*, 35(3): 268-308.
- Bonyadi, M. R., Li, X. 2012.** A new discrete electromagnetism-based meta-heuristic for solving the multidimensional knapsack problem using genetic operators. *Operational Research*, 12(2): 229-252.
- Boucekara, H. 2013.** Electromagnetic Device Optimization Based on Electromagnetism-Like Mechanism. *Applied Computational Electromagnetics Society Journal*, 28(3).
- Boussaïd, I., Lepagnot, J., Siarry, P. 2013.** A survey on optimization metaheuristics. *Information Sciences*, 237(1): 82-117.
- Box, G. E., Wilson, K. 1951.** On the experimental attainment of optimum conditions. *Journal of the Royal Statistical Society Series B (Methodological)*, 13(1): 1-45.
- Boyd, S., Vandenberghe, L. 2009.** Convex optimization. Cambridge University Press, Cambridge, UK, 710 pp.
- Brajevic, I., Tuba, M. 2013.** An upgraded artificial bee colony (ABC) algorithm for constrained optimization problems. *Journal of Intelligent Manufacturing*, 24(4): 729-740.

- Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V. 2006.** Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE Transactions on Evolutionary Computation*, 10(6): 646-657.
- Cagnina, L. C., Esquivel, S. C., Coello, C. A. C. 2008.** Solving Engineering Optimization Problems with the Simple Constrained Particle Swarm Optimizer. *Informatica (Slovenia)*, 32(3): 319-326.
- Chang, P.-C., Chen, S.-H., Fan, C.-Y. 2009.** A hybrid electromagnetism-like algorithm for single machine scheduling problem. *Expert Systems with Applications*, 36(2): 1259-1267.
- Chang, P.-C., Chen, S.-S., Fan, C.-Y. 2008.** Mining gene structures to inject artificial chromosomes for genetic algorithm in single machine scheduling problems. *Applied Soft Computing*, 8(1): 767-777.
- Chang, P. C. 1999.** A branch and bound approach for single machine scheduling with earliness and tardiness penalties. *Computers & Mathematics with Applications*, 37(10): 133-144.
- Chatterjee, A., Siarry, P. 2006.** Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization. *Computers & Operations Research*, 33(3): 859-871.
- Chen, S., Montgomery, J., Röhler, A. B. 2013.** Standard Particle Swarm Optimization on the CEC2013 Real-Parameter Optimization Benchmark Functions.
- Chen, Y., Miao, D., Wang, R. 2010.** A rough set approach to feature selection based on ant colony optimization. *Pattern Recognition Letters*, 31(3): 226-233.
- Chiarandini, M., Paquete, L., Preuss, M., Ridge, E. 2007.** Experiments on metaheuristics: Methodological overview and open issues.
- Chuang, L.-Y., Tsai, S.-W., Yang, C.-H. 2011.** Improved binary particle swarm optimization using catfish effect for feature selection. *Expert Systems with Applications*, 38(10): 12699-12707.
- Civicioglu, P., Besdok, E. 2013.** A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. *Artificial Intelligence Review*, 39(4): 315-346.
- Coelho, L. d. S., Mariani, V. C. 2008.** Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization. *Expert Systems with Applications*, 34(3): 1905-1913.
- Coleman, D. E., Montgomery, D. C. 1993.** A systematic approach to planning for a designed industrial experiment. *Technometrics*, 35(1): 1-12.

- Cover, T., Hart, P. 1967.** Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1): 21-27.
- Cuevas, E., Oliva, D., Díaz, M., Zaldivar, D., Pérez-Cisneros, M., Pajares, G. 2013.** White blood cell segmentation by circle detection using electromagnetism-like optimization. *Computational and mathematical methods in medicine*, 2013.
- Cuevas, E., Oliva, D., Zaldivar, D., Pérez-Cisneros, M., Sossa, H. 2012.** Circle detection using electro-magnetism optimization. *Information Sciences*, 182(1): 40-55.
- Das, S., Abraham, A., Chakraborty, U. K., Konar, A. 2009.** Differential evolution using a neighborhood-based mutation operator. *IEEE Transactions on Evolutionary Computation*, 13(3): 526-553.
- Das, S., Suganthan, P. N. 2011.** Differential evolution: A survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 15(1): 4-31.
- Deb, K. 2000.** An efficient constraint handling method for genetic algorithms. *Computer methods in applied mechanics and engineering*, 186(2): 311-338.
- Deb, K., Goyal, M. 1996.** A combined genetic adaptive search (GeneAS) for engineering design. *Computer Science and Informatics*, 2630-45.
- Debels, D., Vanhoucke, M. 2006.** The electromagnetism meta-heuristic applied to the resource-constrained project scheduling problem. *Artificial Evolution*, 259-270.
- Diao, R., Shen, Q. 2010.** Two new approaches to feature selection with harmony search. Fuzzy Systems (FUZZ), 2010 IEEE International Conference on, 18-23 July 2010, Barcelona, Spain.
- Dongarra, J. J. 2005.** Performance of Various Computers Using Standard Linear Equations Software,(Linpack Benchmark Report). CS-89-85, Dept. of Computer Science, University of Tennessee Tennessee, USA.
- Dorigo, M. 1992.** Optimization, Learning and Natural Algorithms. *Ph.D. Thesis*, Dept. of Computer Science, Politecnico di Milano, Milano, Italy.
- Dumitrescu, I., Stützle, T. 2003.** Combinations of local search and exact algorithms, In: Applications of Evolutionary Computing: Springer. pp. 211-223.
- Eiben, A. E., Schippers, C. 1998.** On evolutionary exploration and exploitation. *Fundamenta Informaticae*, 35(1): 35-50.
- El-Abd, M. 2012.** Performance assessment of foraging algorithms vs. evolutionary algorithms. *Information Sciences*, 182(1): 243-263.
- Farahnakian, M., Razfar, M., Beheshti, A. K. 2012.** A hybrid electromagnetism-like algorithm for integration of process planning and capacitated lot-sizing problem. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 226(2): 326-338.

- Farmer, J. D., Packard, N. H., Perelson, A. S. 1986.** The immune system, adaptation, and machine learning. *Physica D: Nonlinear Phenomena*, 22(1): 187-204.
- Filipović, V., Kartelj, A., Matić, D. 2013.** An electromagnetism metaheuristic for solving the Maximum Betweenness Problem. *Applied Soft Computing*, 13(2): 1303-1313.
- Forsati, R., Mahdavi, M. 2010.** Web text mining using harmony search, In: Recent advances in harmony search algorithm: Springer. pp. 51-64.
- Forsati, R., Moayedikia, A., Jensen, R., Shamsfard, M., Meybodi, M. R. 2014.** Enriched ant colony optimization and its application in feature selection. *Neurocomputing*.
- Forsati, R., Moayedikia, A., Safarkhani, B. 2011.** Heuristic Approach to Solve Feature Selection Problem, In: Digital Information and Communication Technology and Its Applications: Springer. pp. 707-717.
- Gandomi, A. H., Yang, X.-S., Alavi, A. H. 2011.** Mixed variable structural optimization using firefly algorithm. *Computers & Structures*, 89(23): 2325-2336.
- Gandomi, A. H., Yang, X.-S., Alavi, A. H. 2013a.** Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Engineering with Computers*, 29(1): 17-35.
- Gandomi, A. H., Yang, X.-S., Alavi, A. H., Talatahari, S. 2013b.** Bat algorithm for constrained optimization tasks. *Neural Computing and Applications*, 22(6): 1239-1255.
- Gao, L., Zhang, C., Li, X., Wang, L. 2014a.** Discrete electromagnetism-like mechanism algorithm for assembly sequences planning. *International Journal of Production Research*, 52(12): 3485-3503.
- Gao, W., Liu, S. 2011.** Improved artificial bee colony algorithm for global optimization. *Information Processing Letters*, 111(17): 871-882.
- Gao, W., Liu, S. 2012.** A modified artificial bee colony algorithm. *Computers & Operations Research*, 39(3): 687-697.
- Gao, W., Liu, S., Huang, L. 2014b.** Enhancing artificial bee colony algorithm using more information-based search equations. *Information Sciences*, 270(1): 112-133.
- García-Villoria, A., Moreno, R. P. 2010.** Solving the response time variability problem by means of the electromagnetism-like mechanism. *International Journal of Production Research*, 48(22): 6701-6714.
- Glover, F. 1986.** Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5): 533-549.
- Godinho, P., Branco, F. G. 2012.** Adaptive policies for multi-mode project scheduling under uncertainty. *European Journal of Operational Research*, 216(3): 553-562.

- Goldberg, D. E., Holland, J. H. 1988.** Genetic algorithms and machine learning. *Machine learning*, 3(2): 95-99.
- Gonçalves, J. F., Resende, M. G. 2012.** A parallel multi-population biased random-key genetic algorithm for a container loading problem. *Computers & Operations Research*, 39(2): 179-190.
- Guan, X., Dai, X., Li, J. 2011.** Revised electromagnetism-like mechanism for flow path design of unidirectional AGV systems. *International Journal of Production Research*, 49(2): 401-429.
- Guan, X., Dai, X., Qiu, B., Li, J. 2012.** A revised electromagnetism-like mechanism for layout design of reconfigurable manufacturing system. *Computers & Industrial Engineering*, 63(1): 98-108.
- Guyon, I., Elisseeff, A. 2003.** An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3: 1157-1182.
- Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L. 2006.** Feature extraction. Springer,
- Haupt, R. 1995.** Comparison between genetic and for solving electromagnetics problems. *IEEE Transactions on Magnetics*, 31(3).
- He, S., Prempan, E., Wu, Q. 2004.** An improved particle swarm optimizer for mechanical design optimization problems. *Engineering Optimization*, 36(5): 585-605.
- Hicks, C. R. 1964.** Fundamental concepts in the design of experiments. Thomson Learning, 367 pp.
- Homaifar, A., Qi, C. X., Lai, S. H. 1994.** Constrained optimization via genetic algorithms. *Simulation*, 62(4): 242-253.
- Horst, R. 2000.** Introduction to global optimization. Springer, Dordrecht, The Netherlands, 678 pp.
- Horst, R., Romeijn, H. E. 2002.** Handbook of global optimization. Springer, Dordrecht, The Netherlands, 551 pp.
- Hsu, H.-H., Hsieh, C.-W., Lu, M.-D. 2011.** Hybrid feature selection by combining filters and wrappers. *Expert Systems with Applications*, 38(7): 8144-8150.
- Huang, C.-L., Dun, J.-F. 2008.** A distributed PSO-SVM hybrid system with feature selection and parameter optimization. *Applied Soft Computing*, 8(4): 1381-1391.
- Hung, H. L. 2014.** Electromagnetism - like method tuned constant modulus algorithm for blind detector in multicarrier CDMA system. *International Journal of Communication Systems*, 27(2): 233-247.

- Husseinzadeh Kashan, A. 2011.** An efficient algorithm for constrained global optimization and application to mechanical engineering design: League championship algorithm (LCA). *Computer-Aided Design*, 43(12): 1769-1792.
- Jalab, H. A., Abdullah, N. A. 2013.** Content-based image retrieval based on electromagnetism-like mechanism. *Mathematical Problems in Engineering*, 2013.
- Jamili, A., Shafia, M. A., Tavakkoli-Moghaddam, R. 2011.** A hybridization of simulated annealing and electromagnetism-like mechanism for a periodic job shop scheduling problem. *Expert Systems with Applications*, 38(5): 5895-5901.
- Javadi, B., Jolai, F., Slomp, J., Rabbani, M., Tavakkoli-Moghaddam, R. 2014.** A hybrid electromagnetism-like algorithm for dynamic inter/intra-cell layout problem. *International Journal of Computer Integrated Manufacturing*, 27(6): 501-518.
- Javadian, N., Alikhani, M. G., Tavakkoli-Moghaddam, R. 2008.** A discrete binary version of the electromagnetism-like heuristic for solving traveling salesman problem, In: *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*: Springer. pp. 123-130.
- Jensen, R., Shen, Q. 2005.** Fuzzy-rough data reduction with ant colony optimization. *Fuzzy Sets and Systems*, 149(1): 5-20.
- Jhang, J.-Y., Lee, K.-C. 2009.** Array pattern optimization using electromagnetism-like algorithm. *AEU-International Journal of Electronics and Communications*, 63(6): 491-496.
- Jia, D., Duan, X., Khan, M. K. 2014.** Modified artificial bee colony optimization with block perturbation strategy. *Engineering Optimization*, (ahead-of-print): 1-14.
- Jiekang, W., Zhuangzhi, G., Fan, W. 2014.** Short-term multi-objective optimization scheduling for cascaded hydroelectric plants with dynamic generation flow limit based on EMA and DEA. *International Journal of Electrical Power & Energy Systems*, 57:189-197.
- Johnson, D. S., Garey, M. R. 1979.** *Computers and Intractability-A Guide to the Theory of NP-Completeness*. Freeman&Co, San Francisco.
- Joines, J. A., Houck, C. R. 1994.** On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's. *Evolutionary Computation*, 1994. IEEE World Congress on Computational Intelligence, 27 June-2 July 1994, Florida, USA.
- Kabir, M. M., Shahjahan, M., Murase, K. 2011.** A new local search based hybrid genetic algorithm for feature selection. *Neurocomputing*, 74(17): 2914-2928.
- Kabir, M. M., Shahjahan, M., Murase, K. 2012.** A new hybrid ant colony optimization algorithm for feature selection. *Expert Systems with Applications*, 39(3): 3747-3763.

- Kaelo, P., Ali, M. 2007.** Differential evolution algorithms using hybrid mutation. *Computational Optimization and Applications*, 37(2): 231-246.
- Kanagaraj, G., Ponnambalam, S., Jawahar, N., Nilakantan, J. M. 2013.** An effective hybrid cuckoo search and genetic algorithm for constrained engineering design optimization. *Engineering Optimization*, (ahead-of-print): 1-21.
- Kang, F., Li, J., Ma, Z. 2011.** Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions. *Information Sciences*, 181(16): 3508-3531.
- Karaboga, D., Akay, B. 2011.** A modified artificial bee colony (ABC) algorithm for constrained optimization problems. *Applied Soft Computing*, 11(3): 3021-3031.
- Karaboga, D., Basturk, B. 2007.** A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of global optimization*, 39(3): 459-471.
- Karaboga, D., Basturk, B. 2008.** On the performance of artificial bee colony (ABC) algorithm. *Applied soft computing*, 8(1): 687-697.
- Karaboga, D., Gorkemli, B. 2014.** A quick artificial bee colony (qABC) algorithm and its performance on optimization problems. *Applied Soft Computing*, 23227-238.
- Karaboga, D., Gorkemli, B., Ozturk, C., Karaboga, N. 2014.** A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, 42(1): 21-57.
- Kartelj, A., Šurlan, N., Cekic, Z. 2014.** Case-based reasoning and electromagnetism-like method in construction management. *Kybernetes*, 43(2): 8-8.
- Kashef, S., Nezamabadi-pour, H. 2013.** A new feature selection algorithm based on binary ant colony optimization. Information and Knowledge Technology (IKT), 2013 5th Conference on Information and Knowledge Technology, 28 May - 30 May 2013, Shiraz, Iran.
- Kashef, S., Nezamabadi-pour, H. 2014.** An advanced ACO algorithm for feature subset selection. *Neurocomputing*.
- Kayhan, A. H., Ceylan, H., Ayvaz, M. T., Gurarslan, G. 2010.** PSOLVER: A new hybrid particle swarm optimization algorithm for solving continuous optimization problems. *Expert Systems with Applications*, 37(10): 6798-6808.
- Ke, L., Feng, Z., Ren, Z. 2008.** An efficient ant colony optimization approach to attribute reduction in rough set theory. *Pattern Recognition Letters*, 29(9): 1351-1357.
- Kedad-Sidhoum, S., Sourd, F. 2010.** Fast neighborhood search for the single machine earliness–tardiness scheduling problem. *Computers & Operations Research*, 37(8): 1464-1471.

- Kennedy, J., Eberhart, R. 1995.** Particle Swarm Optimization. IEEE International Conference on Neural Networks, Perth, WA, Australia.
- Khalili, M. 2014.** A multi-objective electromagnetism algorithm for a bi-objective hybrid no-wait flowshop scheduling problem. *The International Journal of Advanced Manufacturing Technology*, 70(9-12): 1591-1601.
- Khalili, M., Tavakkoli-Moghaddam, R. 2012.** A multi-objective electromagnetism algorithm for a bi-objective flowshop scheduling problem. *Journal of Manufacturing Systems*, 31(2): 232-239.
- Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P. 1983.** Optimization by simulated annealing. *Science*, 220(4598): 671-680.
- Koza, J. R. 1992.** Genetic programming: on the programming of computers by means of natural selection. MIT press, Boston, USA, 490 pp.
- Koziel, S., Michalewicz, Z. 1999.** Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. *Evolutionary computation*, 7(1): 19-44.
- Kratica, J. 2012.** An electromagnetism-like approach for solving the low autocorrelation binary sequence problem. *International Journal of Computers, Communication and Control*, 7687-694.
- Lampinen, J., Zelinka, I. 1999.** Mixed integer-discrete-continuous optimization by differential evolution-part 1: the optimization method. Czech Republic. Brno University of Technology,
- Lee, C.-H., Chang, F.-K. 2010.** Fractional-order PID controller optimization via improved electromagnetism-like algorithm. *Expert Systems with Applications*, 37(12): 8871-8878.
- Lee, C.-H., Chang, F.-K., Kuo, C.-T., Chang, H.-H. 2012.** A hybrid of electromagnetism-like mechanism and back-propagation algorithms for recurrent neural fuzzy systems design. *International Journal of Systems Science*, 43(2): 231-247.
- Lee, C.-H., Lee, Y.-C. 2012.** Nonlinear systems design by a novel fuzzy neural system via hybridization of electromagnetism-like mechanism and particle swarm optimisation algorithms. *Information Sciences*, 186(1): 59-72.
- Lee, C.-H., Li, C.-T., Chang, F.-Y. 2011.** A species-based improved electromagnetism-like mechanism algorithm for TSK-type interval-valued neural fuzzy system optimization. *Fuzzy Sets and Systems*, 171(1): 22-43.
- Lee, K.-C., Jhang, J.-Y. 2008.** Application of electromagnetism-like algorithm to phase-only syntheses of antenna arrays. *Progress In Electromagnetics Research*, 83279-291.
- Lenstra, J. K., Rinnooy Kan, A., Brucker, P. 1977.** Complexity of machine scheduling problems. *Annals of discrete mathematics*, 1343-362.

- Li, G. 1997.** Single machine earliness and tardiness scheduling. *European Journal of Operational Research*, 96(3): 546-558.
- Liang, J., Suganthan, P., Deb, K. 2005.** Novel composition test functions for numerical global optimization. Swarm Intelligence Symposium, 2005 (SIS 2005), 8-10 June 2005, California, USA.
- Liang, J. J., Qin, A. K., Suganthan, P. N., Baskar, S. 2006.** Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Transactions on Evolutionary Computation*, 10(3): 281-295.
- Liao, C.-J., Tseng, C.-T., Luarn, P. 2007.** A discrete version of particle swarm optimization for flowshop scheduling problems. *Computers & Operations Research*, 34(10): 3099-3111.
- Liao, T., Aydın, D., Stützle, T. 2013.** Artificial bee colonies for continuous optimization: Experimental analysis and improvements. *Swarm Intelligence*, 7(4): 327-356.
- Lim, W. H., Mat Isa, N. A. 2014.** Bidirectional teaching and peer-learning particle swarm optimization. *Information Sciences*, 280(1): 111-134.
- Liu, H., Cai, Z., Wang, Y. 2010.** Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. *Applied Soft Computing*, 10(2): 629-640.
- Liu, J., Lampinen, J. 2005.** A fuzzy adaptive differential evolution algorithm. *Soft Computing*, 9(6): 448-462.
- López-Ibáñez, M., Dubois-Lacoste, J., Stützle, T., Birattari, M. 2011.** The irace package, iterated race for automatic algorithm configuration. IRIDIA, Université Libre de Bruxelles, Belgium, Tech. Rep. TR/IRIDIA/2011-004, Université Libre de Bruxelles,
- Maenhout, B., Vanhoucke, M. 2007.** An electromagnetic meta-heuristic for the nurse scheduling problem. *Journal of Heuristics*, 13(4): 359-385.
- Maldonado, S., Weber, R. 2009.** A wrapper method for feature selection using support vector machines. *Information Sciences*, 179(13): 2208-2217.
- Melo, V. V. d., Carosio, G. L. C. 2012.** Evaluating differential evolution with penalty function to solve constrained engineering problems. *Expert Systems with Applications*, 39(9): 7860-7863.
- Mendes, J. J. d. M., Gonçalves, J. F., Resende, M. G. 2009.** A random key based genetic algorithm for the resource constrained project scheduling problem. *Computers & Operations Research*, 36(1): 92-109.

- Mezura Montes, E., Coello Coello, C. A. 2005.** A simple multimembered evolution strategy to solve constrained optimization problems. *IEEE Transactions on Evolutionary Computation*,
- Mezura-Montes, E., Velázquez-Reyes, J., Coello Coello, C. 2006.** Modified differential evolution for constrained optimization. *IEEE Congress on Evolutionary Computation*, 2006. CEC 2006.,
- Michalewicz, Z., Janikow, C. Z. 1991.** Handling Constraints in Genetic Algorithms. 4th International Conference on Genetic Algorithms (ICGA 91), 13-16 June 1991, California, USA.
- Michalewicz, Z., Nazhiyath, G. 1995.** Genocop III: A co-evolutionary algorithm for numerical optimization problems with nonlinear constraints. *IEEE International Conference on Evolutionary Computation*, 29 Nov-01 Dec 1995, Perth, Australia.
- Michalewicz, Z., Schoenauer, M. 1996.** Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary computation*, 4(1): 1-32.
- Mirabi, M. 2014.** A hybrid electromagnetism algorithm for multi-depot periodic vehicle routing problem. *The International Journal of Advanced Manufacturing Technology*, 71(1-4): 509-518.
- Mirabi, M., Ghomi, F., Jolai, F. 2010.** A Hybrid Electromagnetism-Like Algorithm for Supplier Selection in Make-to-Order Planning. *Scientia Iranica Transaction E: Industrial Engineering*, 17(1): 1-11.
- Mirkhani, M., Forsati, R., Shahri, A. M., Moayedikia, A. 2013.** A novel efficient algorithm for mobile robot localization. *Robotics and Autonomous Systems*, 61(9): 920-931.
- Mohamed, A. W., Sabry, H. Z. 2012.** Constrained optimization based on modified differential evolution algorithm. *Information Sciences*, 194: 171-208.
- Montgomery, D. C. 2007.** Introduction to statistical quality control. John Wiley & Sons, California, USA, 773 pp.
- Montgomery, D. C. 2008.** Design and analysis of experiments. John Wiley & Sons, California, USA, 603 pp.
- Mun, S., Cho, Y.-H. 2012.** Modified harmony search optimization for constrained design problems. *Expert Systems with Applications*, 39(1): 419-423.
- Muñoz Zavala, A. E., Aguirre, A. H., Villa Diharce, E. R. 2005.** Constrained optimization via particle evolutionary swarm optimization algorithm (PESO). *Conference on Genetic and evolutionary computation (Gecco 2005)*, 25-29 June 2005, Washington, USA.

- Naderi, B., Najafi, E., Yazdani, M. 2012.** An electromagnetism-like metaheuristic for open-shop problems with no buffer. *Journal of Industrial Engineering International*, 8(1): 1-8.
- Naderi, B., Tavakkoli-Moghaddam, R., Khalili, M. 2010.** Electromagnetism-like mechanism and simulated annealing algorithms for flowshop scheduling problems minimizing the total weighted tardiness and makespan. *Knowledge-Based Systems*, 23(2): 77-85.
- Naji-Azimi, Z., Toth, P., Galli, L. 2010.** An electromagnetism metaheuristic for the unicost set covering problem. *European Journal of Operational Research*, 205(2): 290-300.
- Nemhauser, G. L., Wolsey, L. A. 1988.** Integer and combinatorial optimization. Wiley New York, New York, USA, 309 pp.
- Neri, F., Tirronen, V. 2010.** Recent advances in differential evolution: a survey and experimental analysis. *Artificial Intelligence Review*, 33(1-2): 61-106.
- Noronha, T. F., Resende, M. G., Ribeiro, C. C. 2011.** A biased random-key genetic algorithm for routing and wavelength assignment. *Journal of Global Optimization*, 50(3): 503-518.
- Oh, I.-S., Lee, J.-S., Moon, B.-R. 2004.** Hybrid genetic algorithms for feature selection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(11): 1424-1437.
- Oliva, D., Cuevas, E., Pajares, G., Zaldivar, D. 2014a.** Template matching using an improved electromagnetism-like algorithm. *Applied Intelligence*, 1-17.
- Oliva, D., Cuevas, E., Pajares, G., Zaldivar, D., Osuna, V. 2014b.** A Multilevel Thresholding algorithm using electromagnetism optimization. *Neurocomputing*, 139357-381.
- Osman, I. H., Laporte, G. 1996.** Metaheuristics: A bibliography. *Annals of Operations Research*, 63(5): 511-623.
- Ouyang, S., Zhou, J., Qin, H., Liao, X., Wang, H. 2013.** A novel multi-objective electromagnetism-like Mechanism algorithm with application to reservoir flood control operation.
- Ow, P. S., Morton, T. E. 1989.** The single machine early/tardy problem. *Management Science*, 35(2): 177-191.
- Pan, Q.-K., Fatih Tasgetiren, M., Suganthan, P. N., Chua, T. J. 2011.** A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Information sciences*, 181(12): 2455-2468.

- Pan, Q.-K., Tasgetiren, M. F., Liang, Y.-C. 2008.** A discrete differential evolution algorithm for the permutation flowshop scheduling problem. *Computers & Industrial Engineering*, 55(4): 795-816.
- Pan, Q.-K., Wang, L., Qian, B. 2009.** A novel differential evolution algorithm for bi-criteria no-wait flow shop scheduling problems. *Computers & Operations Research*, 36(8): 2498-2511.
- Paredis, J. 1994.** Co-evolutionary constraint satisfaction, In: *Parallel Problem Solving from Nature—PPSN III*: Springer. pp. 46-55.
- Parsopoulos, K. E., Vrahatis, M. N. 2002.** Particle swarm optimization method for constrained optimization problems. *Intelligent Technologies—Theory and Application: New Trends in Intelligent Technologies*, 76214-220.
- Passino, K. M. 2002.** Biomimicry of bacterial foraging for distributed optimization and control. *Control Systems, IEEE*, 22(3): 52-67.
- Pintér, J. 1996.** Global optimization in action: continuous and Lipschitz optimization: algorithms, implementations and applications. Springer, Dordrecht, The Netherlands, 685 pp.
- Poli, R. 2008.** Analysis of the publications on the applications of particle swarm optimisation. *Journal of Artificial Evolution and Applications*, 2008(1): 3.
- Poli, R., Kennedy, J., Blackwell, T. 2007.** Particle swarm optimization. *Swarm intelligence*, 1(1): 33-57.
- Price, K., Storn, R. M., Lampinen, J. A. 2006.** Differential evolution: a practical approach to global optimization. Springer, Dordrecht, The Netherlands, 481 pp.
- Puchinger, J., Raidl, G. R. 2005.** Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. Springer,
- Qin, A. K., Huang, V. L., Suganthan, P. N. 2009.** Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, 13(2): 398-417.
- Qin, A. K., Suganthan, P. N. 2005.** Self-adaptive differential evolution algorithm for numerical optimization. 2005 IEEE Congress on Evolutionary Computation (CEC 2005), 2-5 September 2005, Edinburgh, Scotland.
- Rashedi, E., Nezamabadi-pour, H. 2014.** Feature subset selection using improved binary gravitational search algorithm. *Journal of Intelligent and Fuzzy Systems*, 26(3): 1211-1221.
- Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S. 2009.** GSA: a gravitational search algorithm. *Information sciences*, 179(13): 2232-2248.

- Ray, T., Liew, K. M. 2003.** Society and civilization: An optimization algorithm based on the simulation of social behavior. *Evolutionary Computation, IEEE Transactions on*, 7(4): 386-396.
- Reeves, C. R. 1999.** Landscapes, operators and heuristic search. *Annals of Operations Research*, 86473-490.
- Reynolds, R. G., Michalewicz, Z., Cavaretta, M. J. 1995.** Using Cultural Algorithms for Constraint Handling in GENOCOP. International Conference on Evolutionary programming, 05-08 June 1995, California, USA.
- Rocha, A. M. A., Fernandes, E. M. 2008.** Feasibility and dominance rules in the electromagnetism-like algorithm for constrained global optimization, In: Computational Science and Its Applications–ICCSA 2008: Springer. pp. 768-783.
- Rocha, A. M. A., Fernandes, E. M. 2009a.** Hybridizing the electromagnetism-like algorithm with descent search for solving engineering design problems. *International Journal of Computer Mathematics*, 86(10-11): 1932-1946.
- Rocha, A. M. A., Fernandes, E. M. 2009b.** Modified movement force vector in an electromagnetism-like mechanism for global optimization. *Optimization Methods & Software*, 24(2): 253-270.
- Ronkkonen, J., Kukkonen, S., Price, K. V. 2005.** Real-parameter optimization with differential evolution. 2005 IEEE Congress on Evolutionary Computation (CEC 2005), 2-5 September 2005, Edinburgh, Scotland.
- Roshanaei, V., Balagh, A. K. G., Esfahani, M. M. S., Vahdani, B. 2010.** A mixed-integer linear programming model along with an electromagnetism-like algorithm for scheduling job shop production system with sequence-dependent set-up times. *The International Journal of Advanced Manufacturing Technology*, 47(5-8): 783-793.
- Runarsson, T. P., Yao, X. 2000.** Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 4(3): 284-294.
- Sadollah, A., Bahreininejad, A., Eskandar, H., Hamdi, M. 2013.** Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems. *Applied Soft Computing*, 13(5): 2592-2612.
- Sandgren, E. 1990.** Nonlinear integer and discrete programming in mechanical design optimization. *Journal of Mechanical Design*, 112(2): 223-229.
- Schoenauer, M., Xanthakis, S. 1993.** Constrained GA Optimization. 5th International Conference on Genetic Algorithms (ICGA 1993), 01-05 July 1993, Illinois, USA.
- Shi, Y., Liu, H., Gao, L., Zhang, G. 2011.** Cellular particle swarm optimization. *Information Sciences*, 181(20): 4460-4493.

- Snyder, L. V., Daskin, M. S. 2006.** A random-key genetic algorithm for the generalized traveling salesman problem. *European Journal of Operational Research*, 174(1): 38-53.
- Sourd, F., Kedad-Sidhoum, S. 2005.** An efficient algorithm for the earliness-tardiness scheduling problem. *Optimisation Online*,(1205).
- Sourd, F., Kedad-Sidhoum, S. 2008.** A faster branch-and-bound algorithm for the earliness-tardiness scheduling problem. *Journal of Scheduling*, 11(1): 49-58.
- Storn, R., Price, K. 1997.** Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4): 341-359.
- Stützle, T. G. 1999.** Local search algorithms for combinatorial problems: analysis, improvements, and new applications. Infix Sankt, Augustin, Germany, 220 pp.
- Su, C.-T., Lin, H.-C. 2011.** Applying electromagnetism-like mechanism for feature selection. *Information Sciences*, 181(5): 972-986.
- Suganthan, P. N., Hansen, N., Liang, J. J., Deb, K., Chen, Y.-P., Auger, A., Tiwari, S. 2005.** Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. KanGAL Report, 2005005,
- Szeto, W., Wu, Y., Ho, S. C. 2011.** An artificial bee colony algorithm for the capacitated vehicle routing problem. *European Journal of Operational Research*, 215(1): 126-135.
- Taheri, S., Ghazvini, H., Saberi-Nadjafi, J., Biazar, J. 2007.** A hybrid of the restarted Arnoldi and electromagnetism meta-heuristic methods for calculating eigenvalues and eigenvectors of a non-symmetric matrix. *Applied Mathematics and Computation*, 191(1): 79-88.
- Tahk, M.-J., Park, M.-S., Woo, H.-W., Kim, H.-J. 2009.** Hessian approximation algorithms for hybrid optimization methods. *Engineering Optimization*, 41(7): 609-633.
- Tasgetiren, M. F., Pan, Q.-K., Suganthan, P. N., Chen, A. H. 2011.** A discrete artificial bee colony algorithm for the total flowtime minimization in permutation flow shops. *Information Sciences*, 181(16): 3459-3475.
- Thangaraj, R., Pant, M., Abraham, A., Bouvry, P. 2011.** Particle swarm optimization: Hybridization perspectives and experimental illustrations. *Applied Mathematics and Computation*, 217(12): 5208-5226.
- Toğan, V., Daloğlu, A. T. 2008.** An improved genetic algorithm with initial population strategy and self-adaptive member grouping. *Computers & Structures*, 86(11): 1204-1218.

- Tripathi, P. K., Bandyopadhyay, S., Pal, S. K. 2007.** Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients. *Information Sciences*, 177(22): 5033-5049.
- Tsou, C.-S., Hsu, C.-H., Yu, F.-J. 2008.** Using multi-objective electromagnetism-like optimization to analyze inventory tradeoffs under probabilistic demand. *Journal of Scientific & Industrial Research*, 67(8): 569-573.
- Tsou, C.-S., Kao, C.-H. 2008.** Multi-objective inventory control using electromagnetism-like meta-heuristic. *International Journal of Production Research*, 46(14): 3859-3874.
- Turky, A. M., Abdullah, S. 2014.** A multi-population electromagnetic algorithm for dynamic optimisation problems. *Applied Soft Computing*.
- Unler, A., Murat, A., Chinnam, R. B. 2011.** mr2PSO: A maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification. *Information Sciences*, 181(20): 4625-4641.
- Valente, J., Alves, R. A. 2005.** Improved heuristics for the early/tardy scheduling problem with no idle time. *Computers & Operations Research*, 32(3): 557-569.
- Vesterstrom, J. 2002.** Particle swarms extensions for improved local, multimodal and dynamic search in numerical optimization. *Master's Thesis*, Dept. of Computer Science, University of Aarhus, Aarhus, Denmark.
- Voss, S., Osman, I. H., Roucairol, C. 1999.** Meta-heuristics: Advances and trends in local search paradigms for optimization. Kluwer Academic Publishers, Dordrecht, The Netherlands, 701 pp.
- Walker, A., Hallam, J., Willshaw, D. 1993.** Bee-havior in a mobile robot: The construction of a self-organized cognitive map and its use in robot navigation within a complex, natural environment. IEEE International Conference on Neural Networks (ICNN 1993), 28 Mar-01 Apr 1993, California, USA.
- Wang, H., Wu, Z., Rahnamayan, S., Sun, H., Liu, Y., Pan, J.-s. 2014.** Multi-strategy ensemble artificial bee colony algorithm. *Information Sciences*, 279(1): 587-603.
- Wang, L., Li, L.-p. 2010.** An effective differential evolution with level comparison for constrained engineering design. *Structural and Multidisciplinary Optimization*, 41(6): 947-963.
- Wang, X., Yang, J., Teng, X., Xia, W., Jensen, R. 2007.** Feature selection based on rough sets and particle swarm optimization. *Pattern Recognition Letters*, 28(4): 459-471.
- Wang, Y., Cai, Z., Zhou, Y., Fan, Z. 2009.** Constrained optimization based on hybrid evolutionary algorithm and adaptive constraint-handling technique. *Structural and Multidisciplinary Optimization*, 37(4): 395-413.

- Wu, P., Hung, Y.-Y., Lin, Z.-p. 2014.** Intelligent forecasting system based on integration of electromagnetism-like mechanism and fuzzy neural network. *Expert Systems with Applications*, 41(6): 2660-2677.
- Wu, P., Yang, W.-H., Wei, N.-C. 2004.** An electromagnetism algorithm of neural network analysis—an application to textile retail operation. *Journal of the Chinese Institute of Industrial Engineers*, 21(1): 59-67.
- Wu, Q., Gao, L., Li, X., Zhang, C., Rong, Y. 2013.** Applying an electromagnetism-like mechanism algorithm on parameter optimisation of a multi-pass milling process. *International Journal of Production Research*, 51(6): 1777-1788.
- Xiang, W., Ma, S., An, M. 2014a.** hABCDE: A hybrid evolutionary algorithm based on artificial bee colony algorithm and differential evolution. *Applied Mathematics and Computation*, 238370-386.
- Xiang, Y., Peng, Y., Zhong, Y., Chen, Z., Lu, X., Zhong, X. 2014b.** A particle swarm inspired multi-elitist artificial bee colony algorithm for real-parameter optimization. *Computational Optimization and Applications*, 57(2): 493-516.
- Xin, B., Chen, J., Peng, Z., Pan, F. 2010.** An adaptive hybrid optimizer based on particle swarm and differential evolution for global optimization. *Science China Information Sciences*, 53(5): 980-989.
- Yan, H.-S., Wan, X.-Q., Xiong, F.-L. 2014.** A hybrid electromagnetism-like algorithm for two-stage assembly flow shop scheduling problem. *International Journal of Production Research*, (ahead-of-print): 1-14.
- Yang, X.-S. 2009.** Firefly algorithms for multimodal optimization, In: Stochastic algorithms: foundations and applications: Springer. pp. 169-178.
- Yang, X.-S., Deb, S. 2009.** Cuckoo search via Lévy flights. World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), 9-11 December 2009, Coimbatore, India.
- Yin, F., Wang, Y., Wei, S. 2011.** A novel hybrid electromagnetism-like algorithm for solving the inverse kinematics of robot. *Industrial Robot: An International Journal*, 38(4): 429-440.
- Yuan, Z., de Oca, M. A. M., Birattari, M., Stützle, T. 2012.** Continuous optimization algorithms for tuning real and integer parameters of swarm intelligence algorithms. *Swarm Intelligence*, 6(1): 49-75.
- Yurtkuran, A., Emel, E. 2010.** A new hybrid electromagnetism-like algorithm for capacitated vehicle routing problems. *Expert Systems with Applications*, 37(4): 3427-3433.
- Yurtkuran, A., Emel, E. 2014.** Efficient Constraint Handling in Electromagnetism-Like Algorithm for Traveling Salesman Problem with Time Windows. *The Scientific World Journal*, 2014.

Zhan, Z.-H., Zhang, J., Li, Y., Chung, H.-H. 2009. Adaptive particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(6): 1362-1381.

Zhang, C., Li, X., Gao, L., Wu, Q. 2013. An improved electromagnetism-like mechanism algorithm for constrained optimization. *Expert Systems with Applications*, 40(14): 5621-5634.

Zhang, M., Luo, W., Wang, X. 2008. Differential evolution with dynamic stochastic selection for constrained optimization. *Information Sciences*, 178(15): 3043-3074.

Zhao, J.-q., Wang, L., Zeng, P., Fan, W.-h. 2012. An effective hybrid genetic algorithm with flexible allowance technique for constrained engineering design optimization. *Expert Systems with Applications*, 39(5): 6041-6051.

Zhu, G., Kwong, S. 2010. Gbest-guided artificial bee colony algorithm for numerical function optimization. *Applied Mathematics and Computation*, 217(7): 3166-3173.

EKLER

- EK 1 Standart test problemleri
- EK 2 Standart test problemleri veri tabloları
- EK 3 Karma test problemleri
- EK 4 Temel performans analizi sonuçları
- EK 5 Deneysel tasarım sonuçları
- EK 6 Doğrusal kısıtlı optimizasyon problemleri
- EK 7 Hasta veri seti
- EK 8 Kısıtlı mühendislik problemleri sabit veriler

EK 1 STANDART TEST PROBLEMLERİ

Sphere: $f(x) = \sum_{i=1}^D (x_i)^2$

Rosenbrock: $f(x) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$

Schwefel: $f(x) = 418,9829 \times D - \sum_{i=1}^D (x_i \sin(\sqrt{|x_i|}))$

Griewank: $f(x) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$

Weierstrass: $f(x) = \sum_{i=1}^{D-1} (\sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (x_i + 0.5)]) - D \sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (0.5)], a = 0,5, b = 3, kmax = 20$

Quartic: $f(x) = \sum_{i=1}^D i x_i^4 + random(0,1)$

Rastrigin: $f(x) = \sum_{i=1}^D [x_i^2 - 10 \cos(2\pi x_i) + 10]$

Ackley: $f(x) = -20 \exp(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}) + 20 + e - \exp(\frac{1}{D} \sum_{i=1}^D \cos 2\pi x_i)$

Beale: $f(x) = (1,5 - x_1 + x_1 x_2)^2 + (2,25 - x_1 + x_1 x_2^2)^2 + (2,625 - x_1 + x_1 x_2^3)^2$

Bohachevsky1: $f(x) = x_1^2 + 2x_2^2 - 0,3 \cos(3\pi x_1) - 0,4(4\pi x_2) + 0,7$

Bohachevsky1: $f(x) = x_1^2 + 2x_2^2 - 0,3 \cos(3\pi x_1) - 0,4(4\pi x_2) + 0,3$

Bohachevsky1: $f(x) = x_1^2 + 2x_2^2 - 0,3 \cos(3\pi x_1) - 0,4(4\pi x_2) + 0,3$

Booth: $f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$

Branin: $f(x) = (x_2 - \frac{5}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6)^2 + 10(1 - \frac{1}{8\pi}) \cos(x_1) + 10$

Colville: $f(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3^2 - x_4)^2 + 10,1((x_2 - 1)^2 + (x_4 - 1)^2) + 19,8(x_2 - 1)(x_4 - 1)$

Easom: $f(x) = -\cos(x_1) \cos(x_2) \exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$

G&P: $f(x) = \sum_{i=1}^n (|x_i + 0,5|)^2$

Hartmann1: $f(x) = -\sum_{i=1}^4 c_i \exp[-\sum_{j=1}^3 a_{ij} (x_i - p_{ij})^2]$

Hartmann2: $f(x) = -\sum_{i=1}^4 c_i \exp[-\sum_{j=1}^6 a_{ij} (x_i - p_{ij})^2]$

Matyas: $f(x) = 0,26(x_1^2 + x_2^2) - 0,48x_1x_2$

Michalewick1: $f(x) = -\sum_{i=1}^2 \sin(x_i) (\sin(ix_i^2/\pi))^{2m}, m = 10$

Michalewick2: $f(x) = -\sum_{i=1}^5 \sin(x_i) (\sin(ix_i^2/\pi))^{2m}, m = 10$

Michalewick3: $f(x) = -\sum_{i=1}^{10} \sin(x_i) (\sin(ix_i^2/\pi))^{2m}, m = 10$

Shekel1: $f(x) = -\sum_{i=1}^5 [(x - a_i)(x - a_i)^T + c_i]^{-1}$

Shekel2: $f(x) = -\sum_{i=1}^7 [(x - a_i)(x - a_i)^T + c_i]^{-1}$

Shekel3: $f(x) = -\sum_{i=1}^{10} [(x - a_i)(x - a_i)^T + c_i]^{-1}$

Trid6: $f(x) = \sum_{i=1}^6 (x_i - 1)^2 - \sum_{i=2}^6 x_i x_{i-1}$

Stepint: $f(x) = 25 + \sum_{i=1}^5 (|x_i|)$

Foxholes: $f(x) = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})} \right]^{-1}$

EK 2 STANDART TEST PROBLEMLERİ VERİ TABLOLARI

Çizelge Ek 2.1 Shekel fonksiyonu için a ve c parametreleri

i	$a_{ij}, j = 1, \dots, 4$	c_i
1	4 4 4 4	0,1
2	1 1 1 1	0,2
3	8 8 8 8	0,2
4	6 6 6 6	0,4
5	3 7 3 7	0,4
6	2 9 2 9	0,6
7	5 5 3 3	0,3
8	8 1 8 1	0,7
9	6 2 6 2	0,5
10	7 3,6 7 3,6	0,5

Çizelge Ek 2.2 Hartmann fonksiyonu için a , c ve p parametreleri (3 parametrelili)

i	$a_{ij}, j = 1, 2, 3$	c_i	$p_{ij}, j = 1, 2, 3$
1	3 10 30	1	0,3689 0,1170 0,2673
2	0,1 10 35	1,2	0,4699 0,4387 0,7470
3	3 10 30	3	0,1091 0,8732 0,5547
4	0,1 10 35	3,2	0,03815 0,5743 0,8828

Çizelge Ek 2.3 Hartmann fonksiyonu için a , c ve p parametreleri (6 parametrelili)

i	$a_{ij}, j = 1, \dots, 6$	c_i	$p_{ij}, j = 1, \dots, 6$
1	10 3 17 3,5 1,7 8	1	0,1310 0,1696 0,5569 0,0124 0,8283 0,5886
2	0,05 10 17 0,1 8 14	12	0,2329 0,4135 0,8307 0,3736 0,1004 0,9991
3	3 3,5 1,7 10 17 8	3	0,2348 0,1415 0,3522 0,2883 0,3047 0,6650
4	17 8 0,05 10 0,1 14	3,2	0,4047 0,8828 0,8732 0,5743 0,1091 0,0381

Çizelge Ek 2.4 FoxHoles fonksiyonu için a parametresi

j	$a_{ij}, i = 1, 2$	
1	-32	-32
2	-16	-32
3	0	-32
4	16	-32
5	32	-32
6	-32	-16
7	-16	-16
8	0	-16
9	16	-16
10	32	-16
11	-32	0
12	-16	0
13	0	0
14	16	0
15	32	0
16	-32	16
17	-16	16
18	0	16
19	16	16
20	32	16
21	-32	32
22	-16	32
23	0	32
24	16	32
25	32	32

EK 3 KARMA TEST PROBLEMLERİ

F1: Kaydırılmış Sphere fonksiyonu

$$F_1(x) = \sum_{i=1}^D z_i^2 + f_{bias}$$

$$z = x - o$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$x \in [-100, 100]$$

$$x^* = o, F_1(x^*) = f_{bias} = -450$$

F2: Kaydırılmış Schwefel fonksiyonu 1.2

$$F_2(x) = \sum_{i=1}^D \sum_{j=1}^i z_j^2 + f_{bias}$$

$$z = x - o$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$x \in [-100, 100]$$

$$x^* = o, F_2(x^*) = f_{bias} = -450$$

F3: Kaydırılmış ve döndürülmüş yüksek dereceli eliptik fonksiyon

$$F_3(x) = \sum_{i=1}^D (10^6) \left(\frac{i-1}{D-1}\right) z_i^2 + f_{bias}$$

$$z = x - o$$

M: ortagonal matris

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$x \in [-100, 100]$$

$$x^* = o, F_3(x^*) = f_{bias} = -450$$

F4: Ötelenmiş Schwefel fonksiyonu 1.2 Gürültülü

$$F_4(x) = (\sum_{i=1}^D \sum_{j=1}^i z_j^2) * (1 + 0.4|N(0,1)|) + f_{bias}$$

$$z = x - o$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$x \in [-100,100]$$

$$x^* = o, F_3(x^*) = f_{bias} = -450$$

F5: Geniřletilmiř Schwefel Problemi 2.6 global optimum sınırlar üzerinde

$$f(x) = \max\{|x_1 + 2x_2 - 7|, |2x_1 + x_2 - 5|\}, i = 1, \dots, n, x^* = [1,3], f(x^*) = 0.$$

$$F_5(x) = \max\{|Aix - Bi|\} + f_{bias}, i = 1, \dots, D$$

$$x = [x_1, x_2, \dots, x_D]$$

$$A_i: D \times D \text{ matris, } a_{ij}: [-500,500] \text{ arasında rassal sayı, } \det(A) \neq 0$$

$$B_i = A_i \times o, o_i \in [-100,100]$$

$$o_i = 100, i = 1, 2, \dots, [D/4]$$

F6: Kaydırılmıř Rosenbrock fonksiyonu

$$F_6(x) = \sum_{i=1}^{D-1} 100(z_i^2 - z_{i-1})^1 + (z_i - 1)^2 + f_{bias}$$

$$z = x - o + 1$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$x \in [-100,100]$$

$$x^* = o, F_6(x^*) = f_{bias} = 390$$

F7: Kaydırılmıř döndürölmüř Griewank fonksiyonu parametre sınırlamasız

$$F_7(x) = \sum_{i=1}^{D-1} 100(z_i^2 - z_{i-1})^1 + (z_i - 1)^2 + f_{bias}$$

$$z = (x - o) * M$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$M: \text{dođrusal } dn \text{ matrisi, condition number} = 3$$

$$M: M'(1 + 0.3|N(0,1)|)$$

$$\text{Bařlangıç popölasyonu} \in [0,600]$$

$$x \in [-100,100]$$

$$x^* = o, F_7(x^*) = f_{bias} = -180$$

F8: Kaydırılmış döndürülmüş Ackley fonksiyonu global minimum sınırlar üzerinde

$$F_8(x) = -20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^D z_i^2}) - \exp(\frac{1}{D}\sum_{i=1}^D \cos(2\pi z_i)) + 20 + e + f_{bias}$$

$$z = (x - o) * M$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M: Lineer transformasyon matrisi, condition number = 100

$$x \in [-32, 32]^D$$

$$x^* = o, F_8(x^*) = f_{bias} = -140$$

F9: Kaydırılmış Rastrigin fonksiyonu

$$F_9(x) = \sum_{i=1}^D z_i^2 - 10 \cos(2\pi z_i) + 10 + f_{bias}$$

$$z = x - o$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$x \in [-5, 5]$$

$$x^* = o, F_9(x^*) = f_{bias} = -330$$

F10: Kaydırılmış döndürülmüş Rastrigin fonksiyonu

$$F_{10}(x) = \sum_{i=1}^D z_i^2 - 10 \cos(2\pi z_i) + 10 + f_{bias}$$

$$z = (x - o) * M$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M: Lineer transformasyon matrisi, condition number = 2

$$x \in [-5, 5]$$

$$x^* = o, F_{10}(x^*) = f_{bias} = -330$$

F11: Kaydırılmış döndürülmüş Weierstrass fonksiyonu

$$F_{11}(x) = \sum_{i=1}^D \sum_{k=0}^{k_{max}} \left(\left[a^k \cos\left(2\pi b^k (z_i + 0.5)\right) \right] \right) - D \sum_{k=0}^{k_{max}} [a^k \cos(\pi b^k)] + f_{bias}$$

$$z = (x - o) * M$$

$$a = 0.5, b = 3, k_{max} = 20$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M: Linear transformasyon matrisi, consition number = 5

$$x \in [-0.5, 0.5]$$

$$x^* = o, F_{11}(x^*) = f_{bias} = 90$$

F12: Schewefel Problem 2.13

$$F_{12}(x) = \sum_{i=1}^D (A_i - B_i(x))^2 + f_{bias}$$

$$A_i = \sum_{j=1}^D a_{ij} \sin \alpha_j + b_{ij} \cos \alpha_j$$

$$B_i = \sum_{j=1}^D a_{ij} \sin x_j + b_{ij} \cos x_j$$

$$\alpha = [\alpha_1, \alpha_2, \dots, \alpha_D], \text{random}[-\pi, \pi]$$

a_{ij}, b_{ij} [-100,100] aralığında tam sayı

$$x = [x_1, x_2, \dots, x_D]$$

$$x \in [-\pi, \pi]$$

$$x^* = o, F_{11}(x^*) = f_{bias} = -460$$

F13: Kaydırılmış genişletilmiş Griewank ve Rosenbrock fonksiyonu

$$F_{13}(x) = F_8(F_2(z_1, z_2)) + F_8(F_2(z_2, z_3)) + \dots + F_8(F_2(z_{D-1}, z_D)) + F_8(F_2(z_D, z_1)) + f_{bias}$$

$$\text{Rosenbrock: } F_2 = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$$

$$\text{Griewank: } F_8 = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$z = (x - o) + 1$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$x \in [-3, 1]$$

$$x^* = o, F_{13}(x^*) = f_{bias} = -130$$

F14: Kaydırılmış döndürülmüş genişletilmiş Schaffer F6 fonksiyonu

$$F(x, y) = 0.5 \frac{(\sin^2(\sqrt{x^2+y^2})-0.5)}{(1+0.001(x^2+y^2))^2}$$

$$F_{14}(x) = EF(z_1, z_2, \dots, z_D) = F(z_1, z_2) + F(z_2, z_3) + \dots + F(z_{D-1}, z_D) + F(z_D, z_1) + f_{bias}$$

$$z = (x - o) * M$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M : Linear transformasyon matrisi, condition number = 3

$$x \in [-100, 100]$$

$$x^* = o, F_{14}(x^*) = f_{bias} = -300$$

F15: Karma birleşim fonksiyonu:

$$f_{1-2}(x): \text{Rastrigin fonksiyonu: } f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$

$$f_{3-4}(x): \text{Weierstrass fonksiyonu: } f(x) = \sum_{i=1}^{D-1} (\sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (x_i + 0.5)]) - D \sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (0.5)], a = 0.5, b = 3, kmax = 20$$

$$f_{5-6}(x): \text{Griewank fonksiyonu: } f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$f_{7-8}(x): \text{Ackley fonksiyonu: } f(x) = -20 \exp(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}) + 20 + e - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos 2\pi x_i\right)$$

$$f_{9-10}(x): \text{Sphere fonksiyonu: } f(x) = \sum_{i=1}^n x_i^2$$

$$z = \left(\frac{x-o}{\lambda_i}\right) * M_i$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M_i : Birim matris

$$x \in [-5, 5]$$

$$x^* = o_1, F_{15}(x^*) = f_{bias} = -120$$

F16: Döndürülmüş karma birleşim fonksiyonu

$$f_{1-2}(x): \text{Rastrigin fonksiyonu: } f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$

$$f_{3-4}(x): \text{Weierstrass fonksiyonu: } f(x) = \sum_{i=1}^{D-1} (\sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (x_i + 0.5)]) - D \sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (0.5)], a = 0.5, b = 3, kmax = 20$$

$$f_{5-6}(x): \text{Griewank fonksiyonu: } f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$f_{7-8}(x): \text{Ackley fonksiyonu: } f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) + 20 + e - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos 2\pi x_i\right)$$

$$f_{9-10}(x): \text{Sphere fonksiyonu: } f(x) = \sum_{i=1}^n x_i^2$$

$$z = \left(\frac{x-o}{\lambda_i}\right) * M_i$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M_i : Lineer transformasyon matrisi, condition number = 2

$$x \in [-5, 5]$$

$$x^* = o_1, F_{16}(x^*) = f_{bias} = 120$$

F17: Gürültülü döndürülmüş karma birleşim fonksiyonu

$$f_{1-2}(x): \text{Rastrigin fonksiyonu: } f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$

$$f_{3-4}(x): \text{Weierstrass fonksiyonu: } f(x) = \sum_{i=1}^{D-1} (\sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (x_i + 0.5)]) - D \sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (0.5)], a = 0.5, b = 3, kmax = 20$$

$$f_{5-6}(x): \text{Griewank fonksiyonu: } f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$f_{7-8}(x): \text{Ackley fonksiyonu: } f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) + 20 + e - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos 2\pi x_i\right)$$

$$f_{9-10}(x): \text{Sphere fonksiyonu: } f(x) = \sum_{i=1}^n x_i^2$$

$$z = \left(\frac{x-o}{\lambda_i}\right) * M_i$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M_i : Lineer transformasyon matrisi, condition number = 2

$$F_{17} = (F_{16} - f_{bias}) * (1 + 0.2|N(0,1)|) + f_{bias}$$

$$x \in [-5,5]$$

$$x^* = o_1, F_{17}(x^*) = f_{bias} = 120$$

F18: Döndürülmüş karma birleşim fonksiyonu

$$f_{1-2}(x): \text{Rastrigin fonksiyonu: } f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$

$$f_{3-4}(x): \text{Weierstrass fonksiyonu: } f(x) = \sum_{i=1}^{D-1} (\sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (x_i + 0.5)]) - D \sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (0.5)], a = 0.5, b = 3, kmax = 20$$

$$f_{5-6}(x): \text{Griewank fonksiyonu: } f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$f_{7-8}(x): \text{Ackley fonksiyonu: } f(x) = -20 \exp(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}) + 20 + e - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos 2\pi x_i\right)$$

$$f_{9-10}(x): \text{Sphere fonksiyonu: } f(x) = \sum_{i=1}^n x_i^2$$

$$z = \left(\frac{x-o}{\lambda_i}\right) * M_i$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

$$M_i: \text{Lineer transformasyon mat, condition num} = [2 \ 3 \ 2 \ 3 \ 2 \ 3 \ 20 \ 30 \ 200 \ 300]$$

$$x \in [-5,5]$$

$$x^* = o_1, F_{18}(x^*) = f_{bias} = 120$$

F19: Global optimumu dar alanda döndürülmüş karma birleşim fonksiyonu

Tüm parametreleri F_{18} ile aynı sadece σ ve λ vektörleri aşağıdaki gibi.

$$\sigma = [0.1, 2, 1.5, 1, 1, 1.5, 1.5, 2, 2]$$

$$\lambda = [0.1 * 5/32; 5/32; 2 * 1; 1; 2 * 5/100; 5/100; 2 * 10; 10; 2 * 5/60; 5/60]$$

F20: Global optimumun sınır üzerinde olduğu döndürülmüş karma birleşim fonksiyonu

$$f_{1-2}(x): \text{Rastrigin fonksiyonu: } f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$

$$f_{3-4}(x): \text{Weierstrass fonksiyonu: } f(x) = \sum_{i=1}^{D-1} (\sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (x_i + 0.5)]) - D \sum_{k=0}^{kmax} [a^k \cos 2\pi b^k (0.5)], a = 0.5, b = 3, kmax = 20$$

$$f_{5-6}(x): \text{Griewank fonksiyonu: } f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$f_{7-8}(x)$: Ackley fonksiyonu: $f(x) = -20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^D x_i^2}) + 20 + e - \exp(\frac{1}{D}\sum_{i=1}^D \cos 2\pi x_i)$

$f_{9-10}(x)$: Sphere fonksiyonu: $f(x) = \sum_{i=1}^n x_i^2$

$$z = \left(\frac{x-o}{\lambda_i}\right) * M_i$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D], o_{10} = [0, 0, \dots, 0], o_{1,2j} = 5, j = 1, 2, \dots, D/2$$

M_i : Lineer transformasyon mat, condition num = [2 3 2 3 2 3 20 30 200 300]

$$x \in [-5, 5]$$

$$x^* = o_1, F_{20}(x^*) = f_{bias} = 10$$

F21: Döndürülmüş karma birleşim fonksiyonu

$f_{1-2}(x)$: Rastrigin fonksiyonu: Döndürülmüş genişletilmiş Schaffer fonk.

$f_{3-4}(x)$: Weierstrass fonksiyonu: Rastrigin fonk.

$f_{5-6}(x)$: Griewank fonksiyonu: F8F2 fonksiyonu

$f_{7-8}(x)$: Ackley fonksiyonu: Weierstrass fonk.

$f_{9-10}(x)$: Griewank Fonk.

$$z = \left(\frac{x-o}{\lambda_i}\right) * M_i$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M_i : ortogonal mat

$$\lambda = [5 * 5/100; 5/100; 5 * 1; 1; 5 * 1; 1; 5 * 10; 10; 5 * 5/200; 5/200]$$

$$\sigma = [1, 1, 1, 1, 1, 2, 2, 2, 2, 2]$$

$$x \in [-5, 5]$$

$$x^* = o_1, F_{21}(x^*) = f_{bias} = 360$$

F22: Yüksek dereceli döndürülmüş karma birleşim fonksiyonu

Tüm parametreler F_{21} ile aynı sadece;

M_i condition numbers = [10 20 50 100 200 1000 2000 3000 4000 5000]

F23: Süreksiz döndürülmüş karma birleşim fonksiyonu

Tüm parametreler F_{21} ile aynı sadece;

$$x_j = \begin{cases} x_j, & |x_j - o_{1j}| < 1/2 \\ \frac{\text{round}(2x_j)}{2}, & |x_j - o_{1j}| \geq 1/2 \end{cases} \text{ for } j = 1, 2, \dots, D$$

$$\text{round}(x) = \begin{cases} a - 1 & \text{if } x \leq 0 \text{ } b \geq 0.5 \\ b & b < 0.5 \\ a + 1 & \text{if } x > 0 \text{ } b \leq 0.5 \end{cases}$$

F24: Döndürülmüş karma birleşim fonksiyonu

$f_1(x) = \text{Weierstrass fonk.}$

$f_2(x) = \text{Döndürülmüş Genişletilmiş Schaffer fonk.}$

$f_3(x) = \text{F8F2 fonk.}$

$f_4(x) = \text{Ackley fonk.}$

$f_5(x) = \text{Rastrigin fonk.}$

$f_6(x) = \text{Griewank fonk.}$

$f_7(x) = \text{Süreksiz genişletilmiş schaffer fonk.}$

$f_8(x) = \text{Süreksiz rastrigin fonk.}$

$f_9(x) = \text{Yüksek dereceli eliptik fonk.}$

$f_{10}(x) = \text{Gürültülü sphere fonk.}$

$$z = \left(\frac{x-o}{\lambda_i} \right) * M_i$$

$$x = [x_1, x_2, \dots, x_D]$$

$$o = [o_1, o_2, \dots, o_D]$$

M_i : ortogonal mat

$$\lambda = [10; 5/20; 1; 5/32; 1; 5/100; 5/50; 1; 5/100; 5/100]$$

$$\sigma = [2, 2, \dots, 2]$$

$$x \in [-5, 5]$$

$$x^* = o_1, F_{24}(x^*) = f_{bias} = 260$$

F25: Sınırlamasız döndürülmüş karma birleşim fonksiyonu

Tüm parametreleri F_{24} ile aynı sadece x üzerinde sınır yok.

EK 4 TEMEL PERFORMANS ANALİZİ SONUÇLARI

Çizelge Ek 4.1. Standart test problemleri sonuçları

Fonksiyon	D	EMA	ABC	PSO	DGA
Sphere	10	0	0	0	0
		0	0	0	0
	30	0	0	0	0
		0	0	0	0
Rosenbrock	10	0	0	0	0
		0	0	0	0
	30	11,557	0,888	15,089	18,204
		9,220	0,077	24,170	5,036
Griewank	10	1,005E-04	0	0	7,002E-03
		6,774E-05	0	0	0,558E-05
	30	0,017	0	0,0174	0,001
		0,005	0	0,020	0,003
Quartic	10	0	0	0	0
		0	0	0	0
	30	2,259E-02	3,001E-02	1,157E-03	1,363E-03
		1,112E-03	4,866E-03	2,760E-04	4,170E-04
Rastrigin	10	0	0	0	0,045E-05
		0	0	0	1,954E-06
	30	51,442	0	43,977	11,717
		16,948	0	11,729	2,538
Ackley	10	0	0	0	0
		0	0	0	0
	30	0,097	0	0,164	0
		0,279	0	0,494	0
Beale	2	0	0	0	0
		0	0	0	0
Bohachevsky1	2	0	0	0	0
		0	0	0	0
Bohachevsky2	2	0	0	0	0
		0	0	0	0
Bohachevsky3	2	0	0	0	0
		0	0	0	0
Booth	2	0	0	0	0
		0	0	0	0
Branin	2	0,404	0,398	0,397	0,398
		0,001	0	0	0
Colville	4	1,445E-01	9,297E-02	0	4,091E-02
		0,026	6,628E-02	0	8,198E-02
Easom	2	-1	-1	-1	-1
		0	0	0	0

Çizelge Ek 4.1. Standart test problemleri sonuçları (devam)

Fonksiyon	D	EMA	ABC	PSO	DGA
G&P	2	3 0	3 0	3 0	3 0
Hartmann1	3	-3,863 0	-3,863 0	-3,633 0,117	-3,863 0
Hartmann2	6	-2,056 0,248	-3,322 0	-1,859 0,439	-3,227 0,048
Matyas	2	0 0	0 0	0 0	0 0
Michalewick1	2	-1,801 0	-1,801 0	-1,573 0,119	-1,801 0
Michalewick2	5	-1,005 0,649	-4,688 0	-2,491 0,257	-4,683 0,013
Michalewick3	9	-8,995 0,989	-9,660 0	-4,007 0,503	-9,591 0,064
Shekel1	4	-8,006 2,259	-10,153 0	-2,087 1,178	-10,153 0
Shekel2	4	0 0	0 0	1,178 0,215	0 0
Shekel3	4	0 0	0 0	-1,989 1,421	0 0
Trid6	6	-210 0	-210 0	-210 0	-210 0
Stepint	2	0 0	0 0	0 0	0 0
Foxholes	2	0,998 0	0,998 0	0,998 0	0,998 0

Çizelge Ek 4.2. Karma test problemleri sonuçları

Fonksiyon	D	EMA	ABC	PSO	DGA
F1	10	0 0	0 0	0 0	0 0
	30	2,01E-10 1,22E-13	1,06E-13 1,92E-14	0 0	5,68E-14 2,57E-29
	50	8,44E-12 6,30E-15	2,04E-13 3,20E-14	0 0	1,10E-13 1,41E-16
F2	10	1,20 3,98E-06	3,73 2,52	0 0	7,53E-09 8,73E-09
	30	88,09 44,05	2,49E03 969,89	0 0	172,10 48,33
	50	5,02E03 2,83E02	1,61E04 4,04E03	0 0	4,21E03 1,57E03

Çizelge Ek 4.2. Karma test problemleri sonuçları (devam)

Fonksiyon	D	EMA	ABC	PSO	DGA
F3	10	3,66E04 1,11E02	6,41E05 2,78E05	5,64E04 3,27E04	3,21E-1 5,35E-1
	30	2,90E07 1,11E08	6,22E06 1,67E06	2,21E05 9,53E04	3,24E06 1,69E06
	50	5,92E07 3,07E05	1,08E07 2,65E06	2,57E05 7,85E04	2,01E07 6,66E06
F4	10	114,56 61,44	29,17 41,83	0 0	2,35E-07 2,16E-07
	30	1,32E01 334,74	1,49E04 5,15E03	8,35E-04 1,29E-03	556,24 838,35
	50	1,97E05 5,03E05	6,42E04 1,24E04	144,41 114,61	2,54E04 1,21E04
F5	10	0 0	7,76E03 3,48E03	0 0	0 0
	30	2,59E03 6,54E02	1,08E04 1,47E03	3,44E03 777,42	269,03 157,47
	50	1,58E04 1,87E03	2,54E04 2,54E03	8,81E03 1,49E03	3,49E03 681,04
F6	10	11,88 6,702	1,46 1,63	39,89 113,23	3,76E-06 6,16E-06
	30	28,05 19,98	7,49 10,99	141,65 226,24	59,01 45,27
	50	194,33 101,40	6,16 7,76	217,74 251,55	97,44 54,90
F7	10	4,32E-01 2,01E-01	2,45E-01 1,27E-01	4,36E-02 1,96E-02	2,01E-01 1,83E-01
	30	3,33E-01 2,08E-02	1,23E-02 5,04E-03	2,04E-02 1,57E-02	1,46E-02 5,37E-02
	50	5,46E-02 5,37E-02	1,68E-04 1,23E-04	9,12E-03 1,44E-02	1,67E-02 1,33E-01
F9	10	18,77 3,09	0 0	5,26 2,74	0 0
	30	9,04 6,88	6,06E-14 1,44E-14	44,37 10,15	3,32E-02 1,82E-01
	50	141,51 99,63	1,14E-13 1,02E-28	111,51 29,61	1,98E-01 5,52E-01
F10	10	6,77 5,05	28,59 8,29	4,59 1,96	10,80 9,27
	30	406,82 55,70	333,43 68,97	46,13 11,91	211,51 40,38
	50	2,52E2 248,01	1,01E2 99,72	121,47 23,66	411,41 21,97
F11	10	7,34 1,11	5,38 6,04E-01	2,97 1,52	9,00 8,1E-01

Çizelge Ek 4.2. Karma test problemleri sonuçları (devam)

Fonksiyon	D	EMA	ABC	PSO	DGA
	30	29,01 5,77	27,94 1,62	30,06 2,85	39,30 1,22
	50	203,71 45,90	51,21 1,98	54,71 2,97	72,87 1,57
F12	10	3,98E04 2,09E03	299,39 177,41	5,97E03 3,00E03	1,08E04 7,69E03
	30	4,99E05 6,21E03	8,55E03 3,70E03	2,96E05 6,28E04	3,46E05 1,71E05
	50	4,41E05 6,28E05	3,52E04 1,05E04	1,60E06 2,94E06	8,53E05 2,59E05
F13	10	2,09 1,33E-01	2,19E-01 8,95E-02	8,6E-01 2,18E-01	1,43 8,65E-01
	30	2,77 0,95	9,44E-01 1,19E-01	4,37 1,27	5,92 3,17
	50	44,52 10,34	1,55 1,99E-01	9,06 3,59	9,70 4,31
F14	10	4,01 5,88E-01	3,34 2,12E-01	2,47 3,93E-01	3,74 6,04E-01
	30	19,09 1,40E-01	12,97 2,16E-01	12,44 3,76E-01	14,26 1,33E-01
	50	20,44 3,96E-01	21,77 2,01E-01	21,65 4,21E-01	21,95 2,33E-01
F15	10	299,61 201,19	9,14E-02 4,07E-01	321,70 140,68	16,82 18,58
	30	384,67 112,80	5,97E-01 1,74	439,56 73,23	63,51 50,84
	50	821,65 77,42	4,42 3,91	923,04 184,66	201,18 55,82
F16	10	139,58 33,51	145,20 14,70	103,51 8,77	109,77 7,11
	30	309,66 201,80	314,67 49,35	203,06 175,03	252,28 45,17
	50	103,44E01 741,02	798,22 100,32E01	902,49 202,45	501,84 109,92
F17	10	139,40 11,52	152,61 16,70	117,60 11,53	143,85 37,65
	30	297,58 106,02	219,71 32,98	237,04 153,46	302,64 54,32
	50	194,21E01 902,54	154,22E01 322,90E01	141,54E01 199,43	994,21 762,24
F18	10	581,89 136,41	554,08 86,66	637,46 265,94	407,51 222,01
	30	1001,93 345,88	899,55 96,78	916,25 5,70	903,74 5,16E-01

Çizelge Ek 4.2. Karma test problemleri sonuçları (devam)

Fonksiyon	D	EMA	ABC	PSO	DGA
	50	881,92E01 903,22	266,50E01 193,22E01	318,43E01 201,88E01	377,61E01 193,01E01
F19	10	701,90 180,41	556,74 104,46	686,87 249,09	350,00 152,56
	30	109,12E01 60,02	915,46 22,24	917,15 8,35	903,90 7,84E-01
		549,22E01 234,10E01	129,47E01 933,27	203,44E01 100,14E01	176,39E01 821,04
F20	10	597,72 199,93	560,35 104,74	618,64 267,62	465,01 261,61
	30	965,40 11,42	920,33 3,91	916,10 5,79	903,75 6,00E-01
	50	992,33E01 114,94E01	301,29E01 527,29	293,33E01 882,70	243,11E01 155,91
F21	10	572,25 201,86	364,00 154,76	655,56 250,30	500,00 1,73E-13
	30	644,77 191,52	488,36 30,20	674,45 294,25	519,39 106,23
	50	126,55E01 863,04	992,04 163,64	875,90 301,04	991,44 309,05
F22	10	791,54 101,55	726,86 169,78	771,44 23,26	772,11 17,47
	30	1002,27 19,01	107,02E01 36,03	889,22 15,47	882,13 11,10
	50	836,29E01 136,27E01	273,55E01 883,25	199,47E01 108,82E01	214,51E01 442,05
F23	10	600,85 87,44	488,67 62,92	798,85 214,82	559,47 1,16E-13
	30	601,49 10,46	531,97 5,80	710,92 275,98	552,92 102,74
	50	151,95E01 195,53	807,56 23,49	992,44 61,98	894,31 58,23
F24	10	244,70 21,59	200,00 2,61E-03	273,33 161,11	200,00 0
	30	551,75 191,73	200,00 0	226,36 144,38	916,66 194,91
	50	841,49E01 418,56	673,99 43,81	991,35 45,59	203,44E01 331,81
F25	10	401,57 91,77	263,16 88,81	426,65 36,68	382,14 3,10
	30	301,46 10,57	203,19 6,93	210,57 2,10E-01	211,24 9,65E-01
	50	203,33E01 55,32	814,82 19,93	955,12 30,04	603,22 11,08

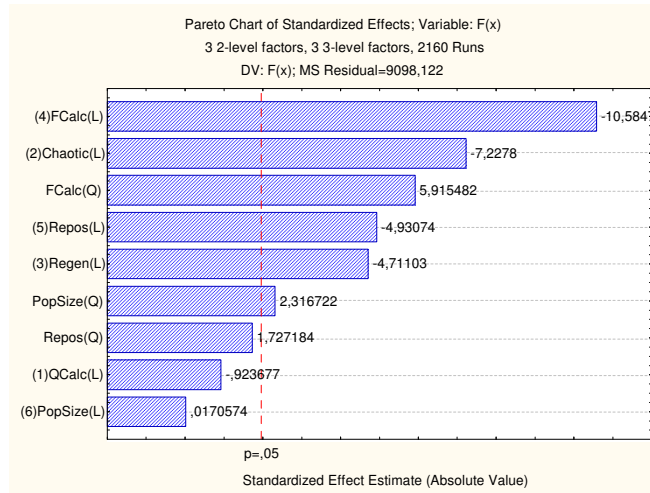
EK 5 DENEYSEL TASARIM SONUÇLARI

Effect Estimates; Var.:F(x); R-sqr=,10602; Adj:;10228 (Rosenbrock.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=9098,122										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	49.6059	2.052339	24.1704	0.000000	45.5811	53.6307	49.6059	2.052339	45.5811	53.6307
(1)QCalc (L)	-3.7914	4.104677	-0.9237	0.355758	-11.8409	4.2582	-1.8957	2.052339	-5.9205	2.1291
(2)Chaotic (L)	-29.6678	4.104677	-7.2278	0.000000	-37.7173	-21.6182	-14.8339	2.052339	-18.8587	-10.8091
(3)Regen (L)	-19.3372	4.104677	-4.7110	0.000003	-27.3868	-11.2877	-9.6686	2.052339	-13.6934	-5.6438
(4)FCalc (L)	-53.2114	5.027182	-10.5847	0.000000	-63.0701	-43.3528	-26.6057	2.513591	-31.5350	-21.6764
FCalc (Q)	25.7540	4.353668	5.9155	0.000000	17.2162	34.2919	12.8770	2.176834	8.6081	17.1459
(5)Repos (L)	-24.7877	5.027182	-4.9307	0.000001	-34.6464	-14.9291	-12.3939	2.513591	-17.3232	-7.4645
Repos (Q)	7.5196	4.353668	1.7272	0.084278	-1.0183	16.0574	3.7598	2.176834	-0.5091	8.0287
(6)PopSize (L)	0.0858	5.027182	0.0171	0.986392	-9.7729	9.9444	0.0429	2.513591	-4.8864	4.9722
PopSize (Q)	10.0862	4.353668	2.3167	0.020612	1.5484	18.6241	5.0431	2.176834	0.7742	9.3120

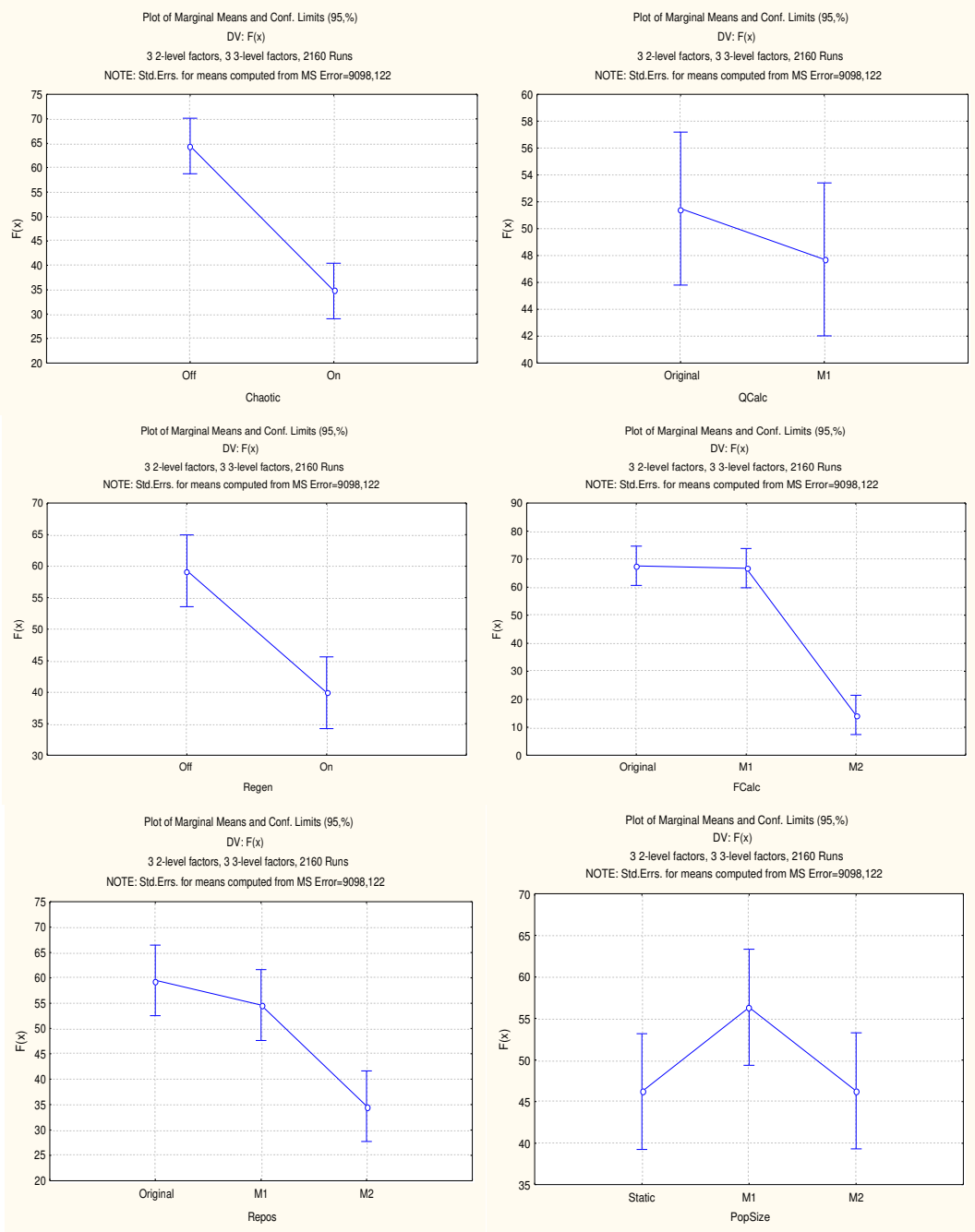
Şekil Ek 5.1 Rosenbrock.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,10602; Adj:;10228 (Rosenbrock.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=9098,122					
Factor	SS	df	MS	F	p
(1)QCalc L	7762	1	7762.3	0.85318	0.355758
(2)Chaotic L	475296	1	475296.2	52.24113	0.000000
(3)Regen L	201922	1	201921.7	22.19377	0.000003
(4)FCalc L+Q	1337695	2	668847.4	73.51488	0.000000
(5)Repos L+Q	248336	2	124168.2	13.64767	0.000001
(6)PopSize L+Q	48834	2	24417.1	2.68375	0.068536
Error	19560963	2150	9098.1		
Total SS	21880808	2159			

Şekil Ek 5.2 Rosenbrock.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.3 Rosenbrock.30 problemi için ortalama etkiler



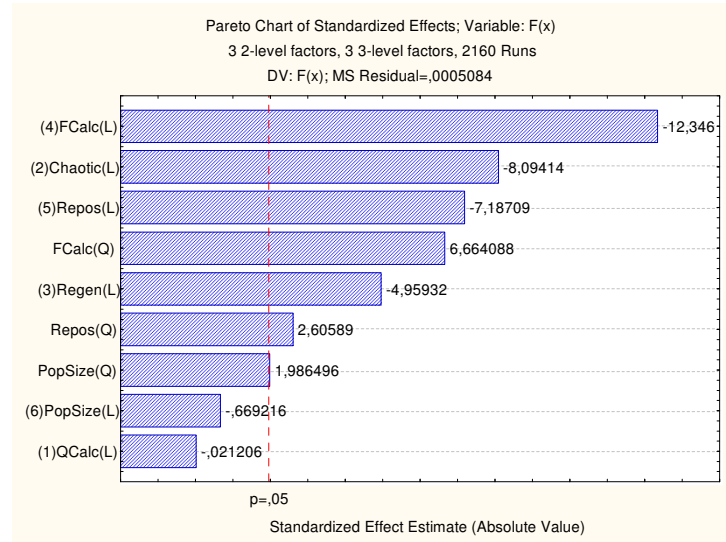
Şekil Ek 5.4 Rosenbrock.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,13993; Adj:,13632 (Rosenbrock.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=,0005084										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	0.012514	0.000485	25.7951	0.000000	0.011563	0.013466	0.012514	0.000485	0.011563	0.013466
(1)QCalc (L)	-0.000021	0.000970	-0.0212	0.983083	-0.001923	0.001882	-0.000010	0.000485	-0.000962	0.000941
(2)Chaotic (L)	-0.007854	0.000970	-8.0941	0.000000	-0.009756	-0.005951	-0.003927	0.000485	-0.004878	-0.002975
(3)Regen (L)	-0.004812	0.000970	-4.9593	0.000001	-0.006715	-0.002909	-0.002406	0.000485	-0.003357	-0.001455
(4)FCalc (L)	-0.014671	0.001188	-12.3460	0.000000	-0.017002	-0.012341	-0.007336	0.000594	-0.008501	-0.006170
FCalc (Q)	0.006858	0.001029	6.6641	0.000000	0.004840	0.008876	0.003429	0.000515	0.002420	0.004438
(5)Repos (L)	-0.008541	0.001188	-7.1871	0.000000	-0.010871	-0.006210	-0.004270	0.000594	-0.005436	-0.003105
Repos (Q)	0.002682	0.001029	2.6059	0.009227	0.000664	0.004700	0.001341	0.000515	0.000332	0.002350
(6)PopSize (L)	-0.000795	0.001188	-0.6692	0.503429	-0.003126	0.001535	-0.000398	0.000594	-0.001563	0.000768
PopSize (Q)	0.002044	0.001029	1.9865	0.047105	0.000026	0.004063	0.001022	0.000515	0.000013	0.002031

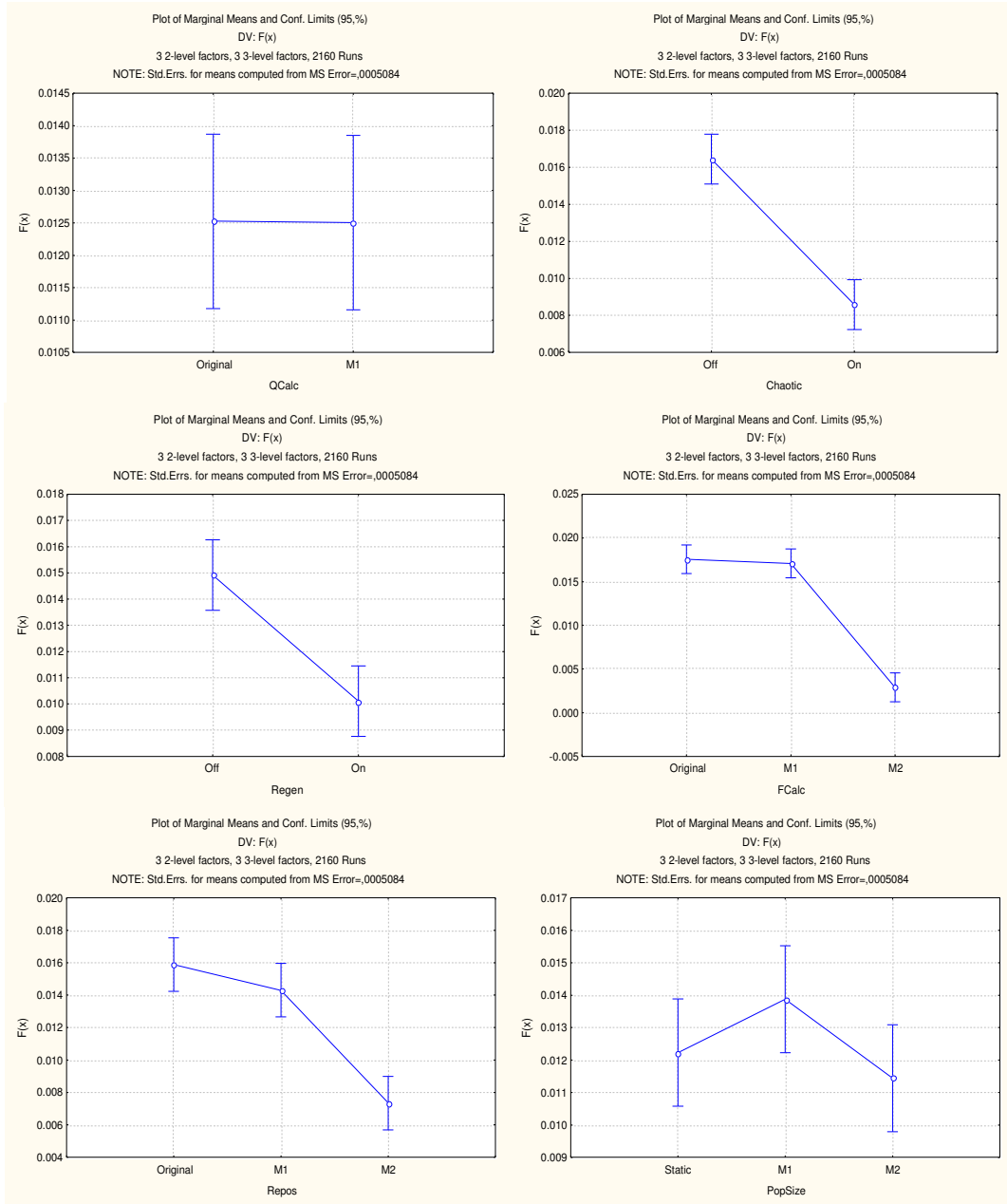
Şekil Ek 5.5 Rosenbrock.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,13993; Adj:,13632 (Rosenbrock.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=,0005084					
Factor	SS	df	MS	F	p
(1)QCalc L	0.000000	1	0.000000	0.00045	0.983083
(2)Chaotic L	0.033306	1	0.033306	65.51514	0.000000
(3)Regen L	0.012503	1	0.012503	24.59484	0.000001
(4)FCalc L+Q	0.100065	2	0.050032	98.41665	0.000000
(5)Repos L+Q	0.029712	2	0.014856	29.22243	0.000000
(6)PopSize L+Q	0.002234	2	0.001117	2.19701	0.111385
Error	1.093003	2150	0.000508		
Total SS	1.270823	2159			

Şekil Ek 5.6 Rosenbrock.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.7 Rosenbrock.10 problemi için ortalama etkiler



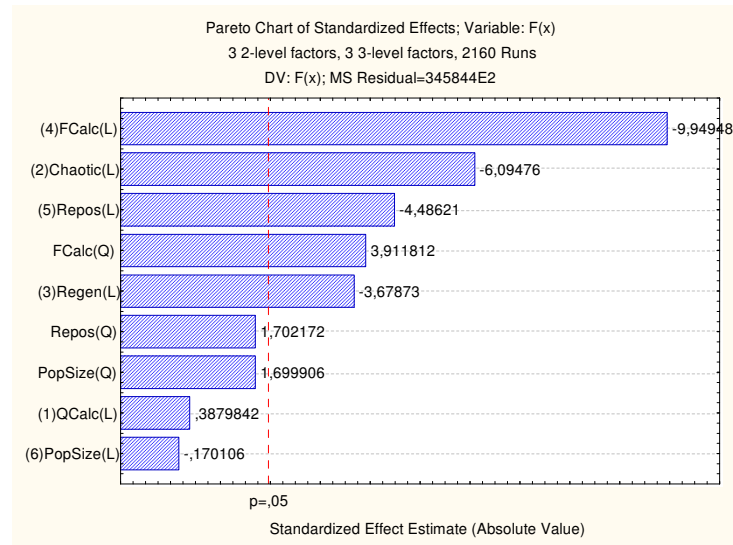
Şekil Ek 5.8 Rosenbrock.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,08162; Adj:,07777 (Rosenbrock.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=345844E2										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	2564.81	126.5358	20.26943	0.000000	2316.66	2812.95	2564.81	126.5358	2316.66	2812.95
(1)QCalc (L)	98.19	253.0717	0.38798	0.698066	-398.10	594.48	49.09	126.5358	-199.05	297.24
(2)Chaotic (L)	-1542.41	253.0717	-6.09476	0.000000	-2038.70	-1046.12	-771.21	126.5358	-1019.35	-523.06
(3)Regen (L)	-930.98	253.0717	-3.67873	0.000240	-1427.27	-434.69	-465.49	126.5358	-713.64	-217.35
(4)FCalc (L)	-3083.82	309.9482	-9.94948	0.000000	-3691.65	-2475.99	-1541.91	154.9741	-1845.83	-1238.00
FCalc (Q)	1050.02	268.4231	3.91181	0.000094	523.62	1576.42	525.01	134.2115	261.81	788.21
(5)Repos (L)	-1390.49	309.9482	-4.48621	0.000008	-1998.32	-782.66	-695.25	154.9741	-999.16	-391.33
Repos (Q)	456.90	268.4231	1.70217	0.088868	-69.49	983.30	228.45	134.2115	-34.75	491.65
(6)PopSize (L)	-52.72	309.9482	-0.17011	0.864943	-660.55	555.11	-26.36	154.9741	-330.28	277.55
PopSize (Q)	456.29	268.4231	1.69991	0.089293	-70.10	982.69	228.15	134.2115	-35.05	491.34

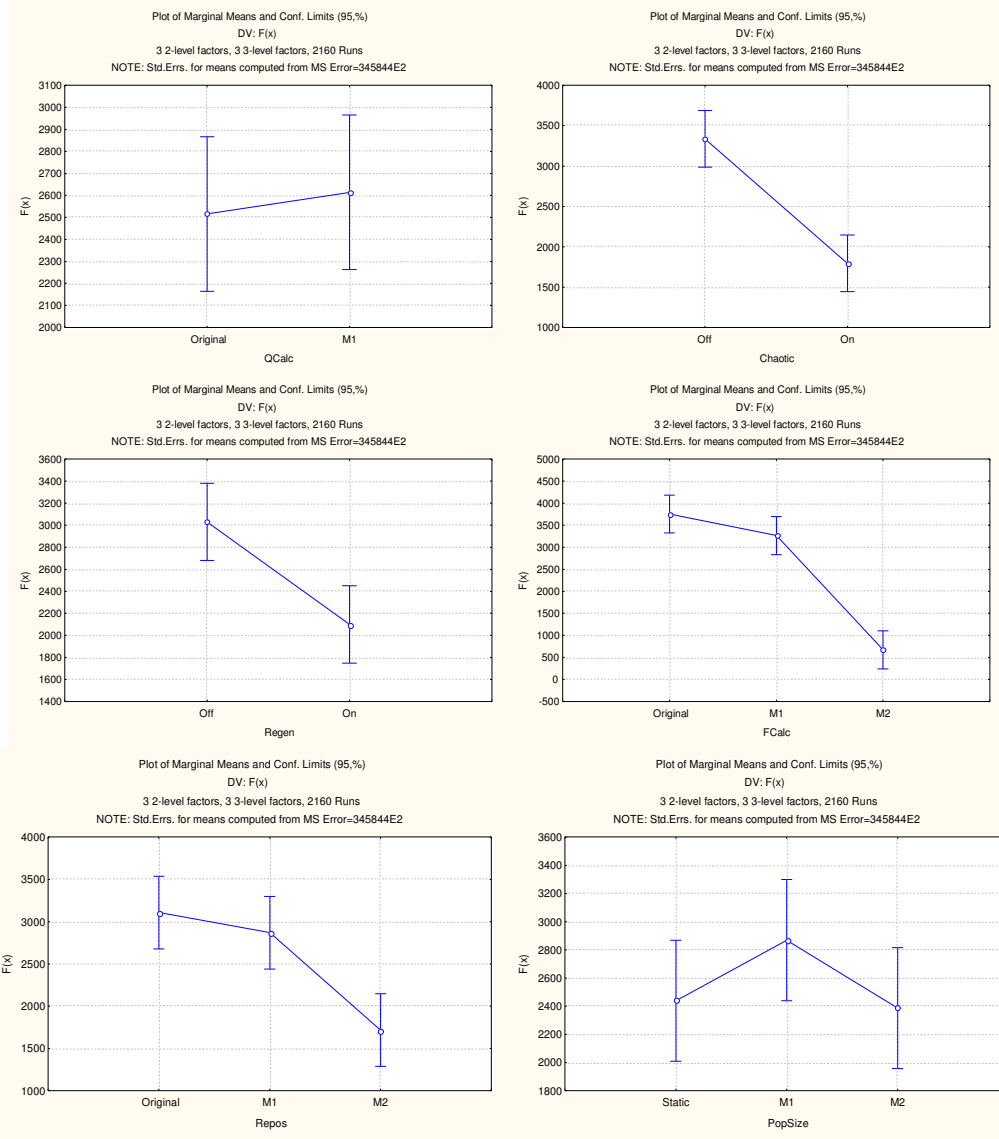
Şekil Ek 5.9 Rosenbrock.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,08162; Adj:,07777 (Rosenbrock.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=345844E2					
Factor	SS	df	MS	F	p
(1)QCalc L	5.206058E+06	1	5.206058E+06	0.15053	0.698066
(2)Chaotic L	1.284677E+09	1	1.284677E+09	37.14609	0.000000
(3)Regen L	4.680326E+08	1	4.680326E+08	13.53304	0.000240
(4)FCalc L+Q	3.952809E+09	2	1.976405E+09	57.14721	0.000000
(5)Repos L+Q	7.962538E+08	2	3.981269E+08	11.51173	0.000011
(6)PopSize L+Q	1.009387E+08	2	5.046936E+07	1.45931	0.232627
Error	7.435657E+10	2150	3.458445E+07		
Total SS	8.096448E+10	2159			

Şekil Ek 5.10 Rosenbrock.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.11 Rosenbrock.50 problemi için ortalama etkiler



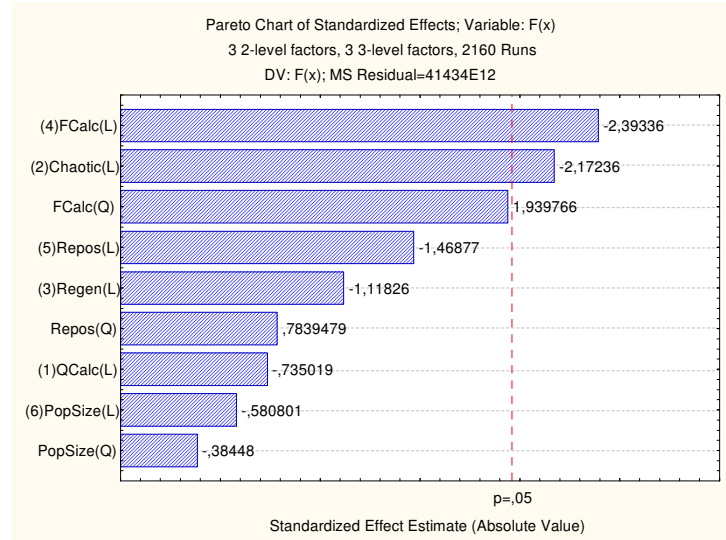
Şekil Ek 5.12 Rosenbrock.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,00888; Adj:.,00473 (Griewank.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=41434E12										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	2359944	437975	5.38830	0.00000	1501044	3218843	2359944	437975	1501044	3218843
(1)QCalc (L)	-643840	875950	-0.7350	0.46240	-2361639	1073959	-321920	437975	-1180819	536979
(2)Chaotic (L)	-1902877	875950	-2.1723	0.02993	-3620676	-185078	-951438	437975	-1810338	-92539
(3)Regen (L)	-979543	875950	-1.1182	0.26358	-2697342	738256	-489771	437975	-1348671	369128
(4)FCalc (L)	-2567636	1072816	-2.3933	0.01678	-4671502	-463770	-1283818	536408	-2335751	-231885
FCalc (Q)	1802209	929086	1.9397	0.05253	-19791	3624210	901104	464543	-9895	1812105
(5)Repos (L)	-1575716	1072816	-1.4687	0.14204	-3679581	528149	-787858	536408	-1839790	264074
Repos (Q)	728355	929086	0.7839	0.43315	-1093646	2550356	364177	464543	-546823	1275178
(6)PopSize (L)	-623092	1072816	-0.5808	0.56143	-2726958	1480773	-311546	536408	-1363479	740386
PopSize (Q)	-357214	929086	-0.3844	0.70066	-2179215	1464786	-178607	464543	-1089607	732393

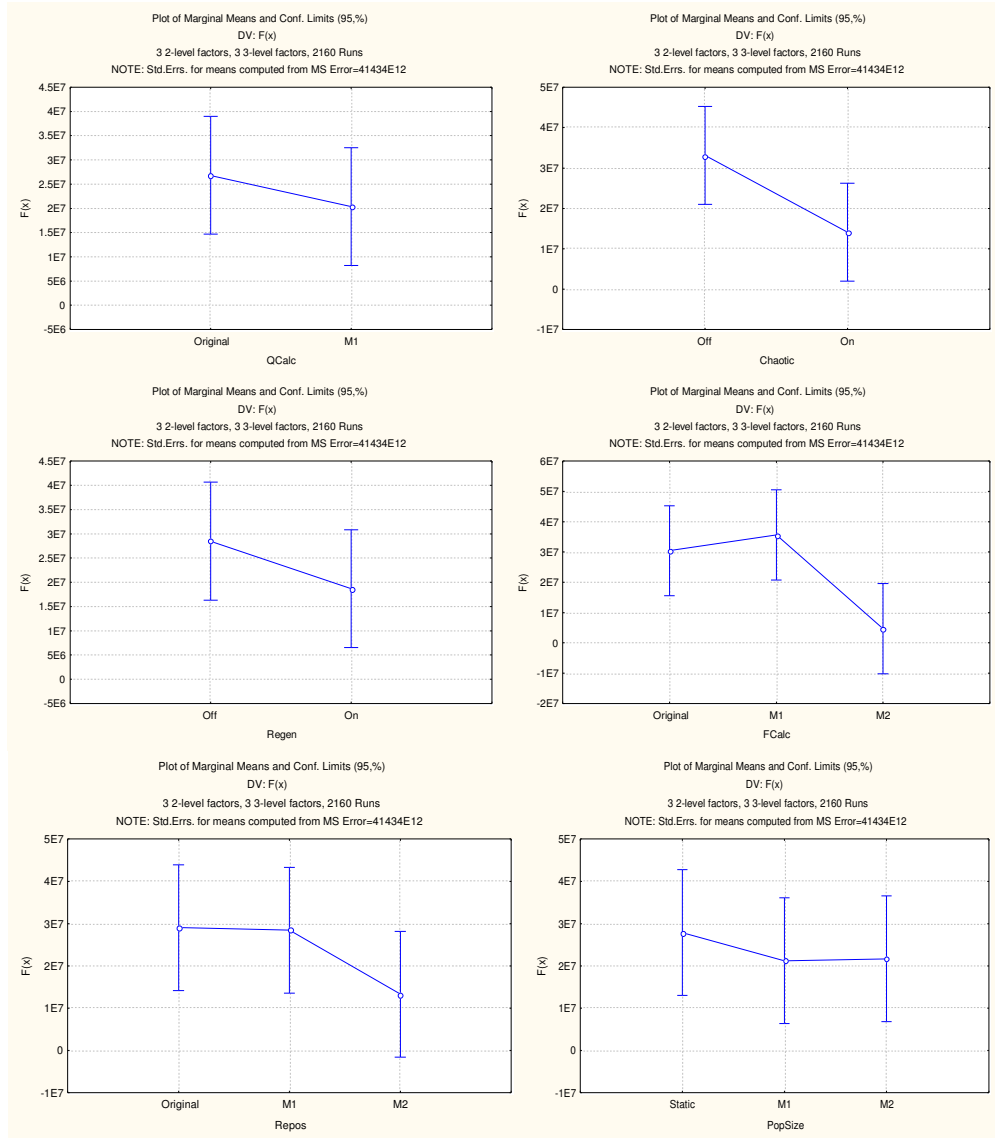
Şekil Ek 5.13 Griewank.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,00888; Adj:.,00473 (Griewank.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=41434E12						
Factor	SS	df	MS	F	p	
(1)QCalc L	2.238463E+16	1	2.238463E+16	0.540252	0.462408	
(2)Chaotic L	1.955308E+17	1	1.955308E+17	4.719132	0.029938	
(3)Regen L	5.181325E+16	1	5.181325E+16	1.250512	0.263580	
(4)FCalc L+Q	3.932413E+17	2	1.966206E+17	4.745434	0.008783	
(5)Repos L+Q	1.148478E+17	2	5.742389E+16	1.385924	0.250316	
(6)PopSize L+Q	2.010170E+16	2	1.005085E+16	0.242577	0.784625	
Error	8.908234E+19	2150	4.143365E+16			
Total SS	8.988026E+19	2159				

Şekil Ek 5.14 Griewank.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.15 Griewank.30 problemi için ortalama etkiler



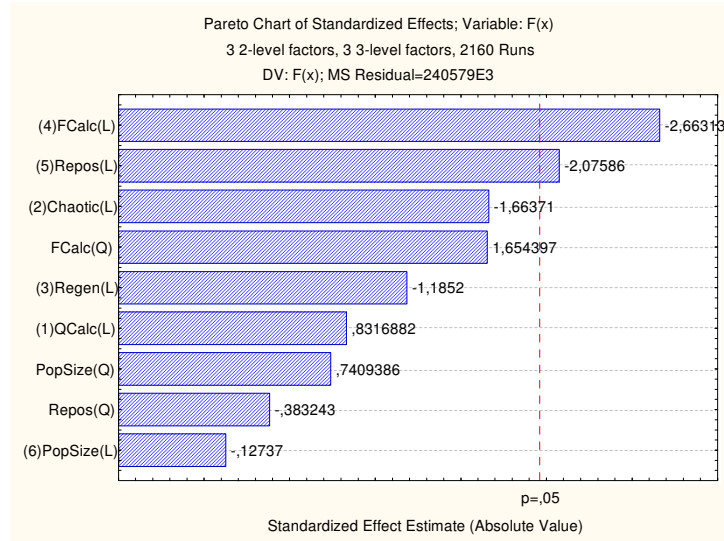
Şekil Ek 5.16 Griewank.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,00909; Adj:,00494 (Griewank.10.sta)											
3 2-level factors, 3 3-level factors, 2160 Runs											
DV: F(x); MS Residual=240579E3											
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt	
Mean/Interc.	1985.59	333.7348	5.94959	0.000000	1331.11	2640.062	1985.59	333.7348	1331.11	2640.062	
(1)QCalc (L)	555.13	667.4697	0.83169	0.405677	-753.83	1864.080	277.56	333.7348	-376.91	932.040	
(2)Chaotic (L)	-1110.48	667.4697	-1.66371	0.096316	-2419.43	198.478	-555.24	333.7348	-1209.71	99.239	
(3)Regen (L)	-791.09	667.4697	-1.18520	0.236069	-2100.04	517.867	-395.54	333.7348	-1050.02	258.934	
(4)FCalc (L)	-2177.05	817.4800	-2.66313	0.007800	-3780.19	-573.921	-1088.53	408.7400	-1890.09	-286.960	
FCalc (Q)	1171.24	707.9585	1.65440	0.098193	-217.11	2559.599	585.62	353.9792	-108.56	1279.800	
(5)Repos (L)	-1696.97	817.4800	-2.07586	0.038026	-3300.11	-93.840	-848.49	408.7400	-1650.05	-46.920	
Repos (Q)	-271.32	707.9585	-0.38324	0.701578	-1659.67	1117.035	-135.66	353.9792	-829.84	558.517	
(6)PopSize (L)	-104.12	817.4800	-0.12737	0.898660	-1707.26	1499.012	-52.06	408.7400	-853.63	749.506	
PopSize (Q)	524.55	707.9585	0.74094	0.458812	-863.80	1912.909	262.28	353.9792	-431.90	956.454	

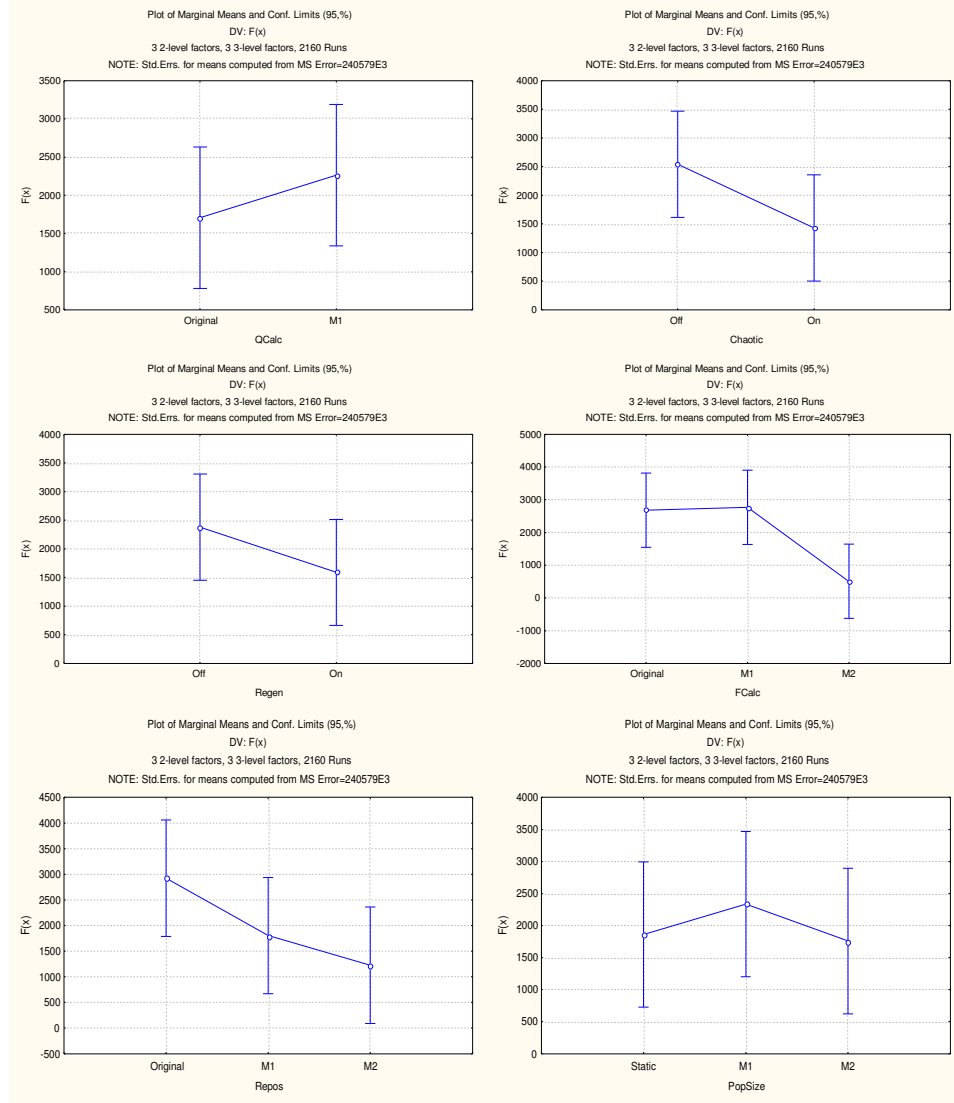
Şekil Ek 5.17 Griewank.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,00909; Adj:,00494 (Griewank.10.sta)						
3 2-level factors, 3 3-level factors, 2160 Runs						
DV: F(x); MS Residual=240579E3						
Factor	SS	df	MS	F	p	
(1)QCalc L	1.664094E+08	1	1.664094E+08	0.691705	0.405677	
(2)Chaotic L	6.659039E+08	1	6.659039E+08	2.767928	0.096316	
(3)Regen L	3.379413E+08	1	3.379413E+08	1.404703	0.236069	
(4)FCalc L+Q	2.364715E+09	2	1.182357E+09	4.914643	0.007421	
(5)Repos L+Q	1.072034E+09	2	5.360172E+08	2.228034	0.107989	
(6)PopSize L+Q	1.359781E+08	2	6.798907E+07	0.282607	0.753844	
Error	5.172438E+11	2150	2.405785E+08			
Total SS	5.219868E+11	2159				

Şekil Ek 5.18 Griewank.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.19 Griewank.10 problemi için ortalama etkiler



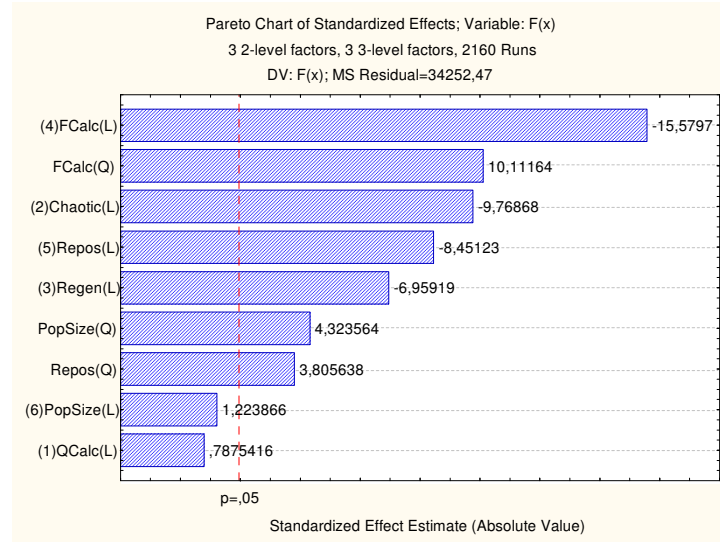
Şekil Ek 5.20 Griewank.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,21691; Adj:,21364 (Griewank.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=34252,47										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	150.607	3.982163	37.8204	0.000000	142.798	158.416	150.6069	3.982163	142.7976	158.4161
(1)QCalc (L)	6.272	7.964326	0.7875	0.431052	-9.346	21.891	3.1361	3.982163	-4.6732	10.9454
(2)Chaotic (L)	-77.801	7.964326	-9.7687	0.000000	-93.420	-62.182	-38.9005	3.982163	-46.7098	-31.0912
(3)Regen (L)	-55.425	7.964326	-6.9592	0.000000	-71.044	-39.807	-27.7126	3.982163	-35.5219	-19.9033
(4)FCalc (L)	-151.969	9.754268	-15.5797	0.000000	-171.098	-132.840	-75.9844	4.877134	-85.5488	-66.4201
FCalc (Q)	85.418	8.447444	10.1116	0.000000	68.852	101.984	42.7088	4.223722	34.4258	50.9918
(5)Repos (L)	-82.436	9.754268	-8.4512	0.000000	-101.564	-63.307	-41.2178	4.877134	-50.7822	-31.6534
Repos (Q)	32.148	8.447444	3.8056	0.000145	15.582	48.714	16.0740	4.223722	7.7909	24.3570
(6)PopSize (L)	11.938	9.754268	1.2239	0.221137	-7.191	31.067	5.9690	4.877134	-3.5954	15.5333
PopSize (Q)	36.523	8.447444	4.3236	0.000016	19.957	53.089	18.2615	4.223722	9.9785	26.5445

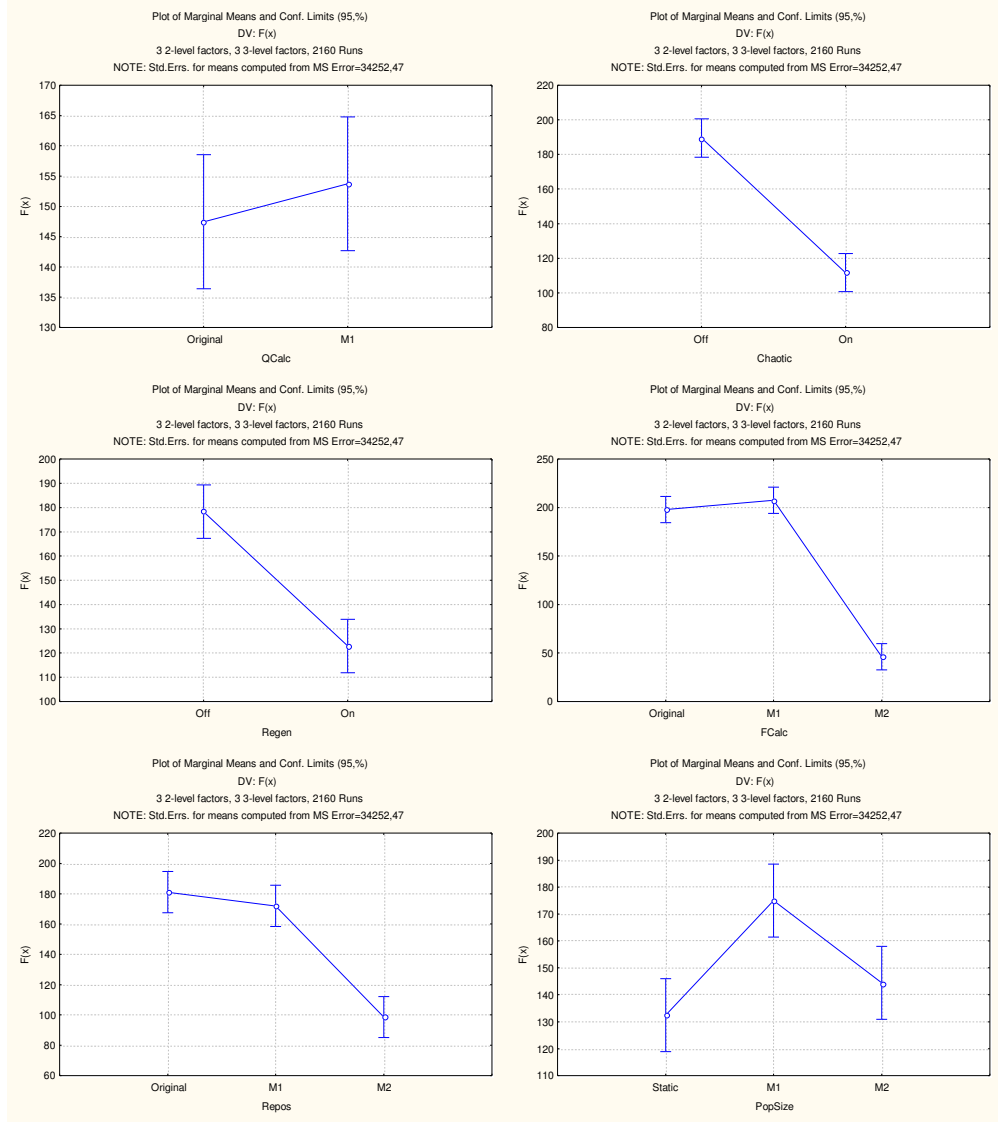
Şekil Ek 5.21 Griewank.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,21691; Adj:,21364 (Griewank.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=34252,47					
Factor	SS	df	MS	F	p
(1)QCalc L	21244	1	21244	0.6202	0.431052
(2)Chaotic L	3268611	1	3268611	95.4270	0.000000
(3)Regen L	1658859	1	1658859	48.4303	0.000000
(4)FCalc L+Q	11816189	2	5908095	172.4867	0.000000
(5)Repos L+Q	2942498	2	1471249	42.9531	0.000000
(6)PopSize L+Q	691593	2	345797	10.0955	0.000043
Error	73642804	2150	34252		
Total SS	94041799	2159			

Şekil Ek 5.22 Griewank.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.23 Griewank.50 problemi için ortalama etkiler



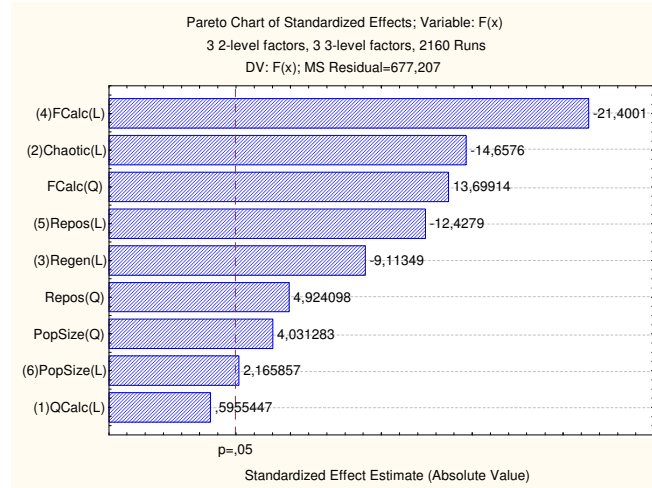
Şekil Ek 5.24 Griewank.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,3472; Adj:,34447 (Ackley.30.sta)										
3 2-level factors, 3 3-level factors, 2160 Runs										
DV: F(x); MS Residual=677,207										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	29.3570	0.559930	52.4298	0.000000	28.2589	30.4551	29.3570	0.559930	28.2589	30.4551
(1)QCalc (L)	0.6669	1.119860	0.5955	0.551542	-1.5292	2.8630	0.3335	0.559930	-0.7646	1.4315
(2)Chaotic (L)	-16.4145	1.119860	-14.6576	0.000000	-18.6106	-14.2184	-8.2072	0.559930	-9.3053	-7.1092
(3)Regen (L)	-10.2058	1.119860	-9.1135	0.000000	-12.4020	-8.0097	-5.1029	0.559930	-6.2010	-4.0049
(4)FCalc (L)	-29.3512	1.371543	-21.4001	0.000000	-32.0409	-26.6615	-14.6756	0.685772	-16.0204	-13.3308
FCalc (Q)	16.2717	1.187791	13.6991	0.000000	13.9424	18.6011	8.1359	0.593896	6.9712	9.3005
(5)Repos (L)	-17.0454	1.371543	-12.4279	0.000000	-19.7351	-14.3557	-8.5227	0.685772	-9.8675	-7.1778
Repos (Q)	5.8488	1.187791	4.9241	0.000001	3.5195	8.1781	2.9244	0.593896	1.7597	4.0891
(6)PopSize (L)	2.9706	1.371543	2.1659	0.030432	0.2809	5.6603	1.4853	0.685772	0.1404	2.8301
PopSize (Q)	4.7883	1.187791	4.0313	0.000057	2.4590	7.1177	2.3942	0.593896	1.2295	3.5588

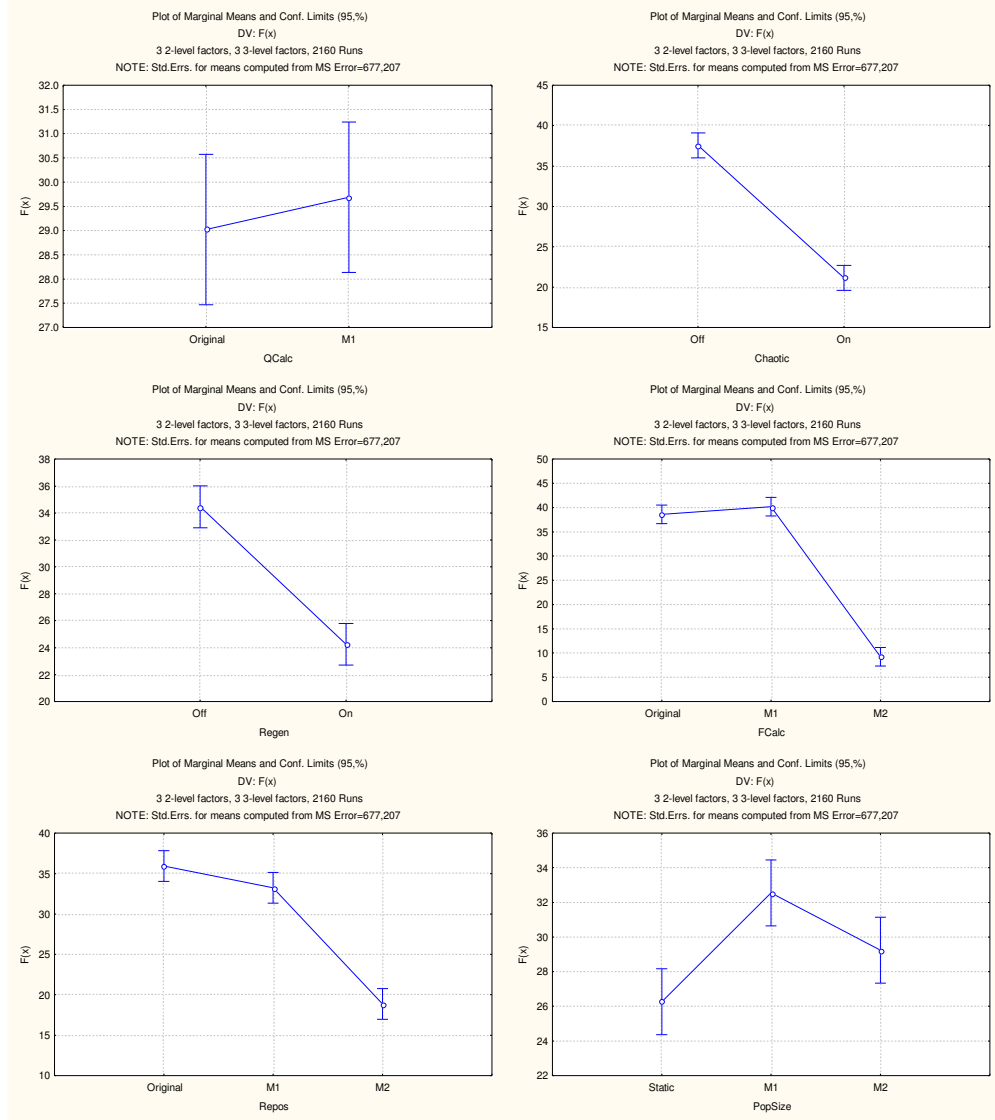
Şekil Ek 5.25 Ackley.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,3472; Adj:,34447 (Ackley.30.sta)					
3 2-level factors, 3 3-level factors, 2160 Runs					
DV: F(x); MS Residual=677,207					
Factor	SS	df	MS	F	p
(1)QCalc L	240	1	240.2	0.3547	0.551542
(2)Chaotic L	145495	1	145495.2	214.8461	0.000000
(3)Regen L	56246	1	56245.9	83.0557	0.000000
(4)FCalc L+Q	437227	2	218613.3	322.8161	0.000000
(5)Repos L+Q	121016	2	60508.0	89.3494	0.000000
(6)PopSize L+Q	14182	2	7091.1	10.4711	0.000030
Error	1455995	2150	677.2		
Total SS	2230401	2159			

Şekil Ek 5.26 Ackley.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.27 Ackley.30 problemi için ortalama etkiler



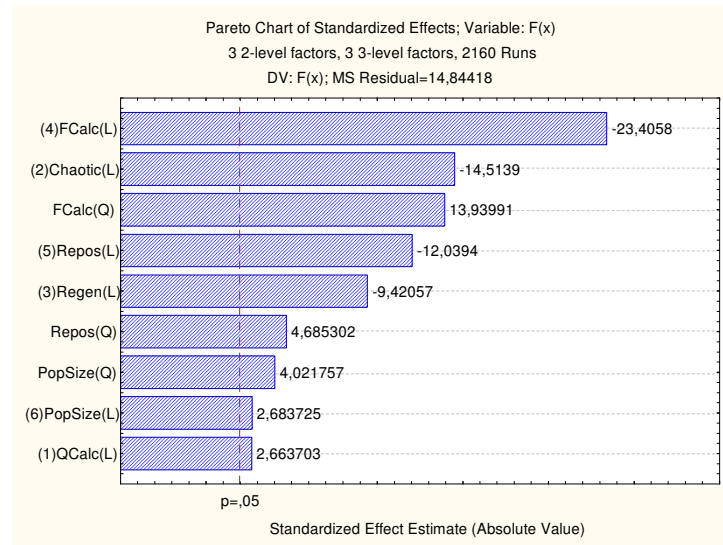
Şekil Ek 5.28 Ackley.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,36558; Adj:,36293 (Ackley.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=14,84418										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	4.65144	0.082899	56.1095	0.000000	4.48887	4.81401	4.65144	0.082899	4.48887	4.81401
(1)QCalc (L)	0.44164	0.165799	2.6637	0.007786	0.11650	0.76678	0.22082	0.082899	0.05825	0.38339
(2)Chaotic (L)	-2.40639	0.165799	-14.5139	0.000000	-2.73153	-2.08125	-1.20320	0.082899	-1.36577	-1.04062
(3)Regen (L)	-1.56192	0.165799	-9.4206	0.000000	-1.88706	-1.23678	-0.78096	0.082899	-0.94353	-0.61839
(4)FCalc (L)	-4.75282	0.203061	-23.4058	0.000000	-5.15103	-4.35460	-2.37641	0.101531	-2.57552	-2.17730
FCalc (Q)	2.45142	0.175856	13.9399	0.000000	2.10655	2.79628	1.22571	0.087928	1.05328	1.39814
(5)Repos (L)	-2.44474	0.203061	-12.0394	0.000000	-2.84296	-2.04653	-1.22237	0.101531	-1.42148	-1.02326
Repos (Q)	0.82394	0.175856	4.6853	0.000003	0.47907	1.16880	0.41197	0.087928	0.23954	0.58440
(6)PopSize (L)	0.54496	0.203061	2.6837	0.007337	0.14674	0.94318	0.27248	0.101531	0.07337	0.47159
PopSize (Q)	0.70725	0.175856	4.0218	0.000060	0.36238	1.05212	0.35363	0.087928	0.18119	0.52606

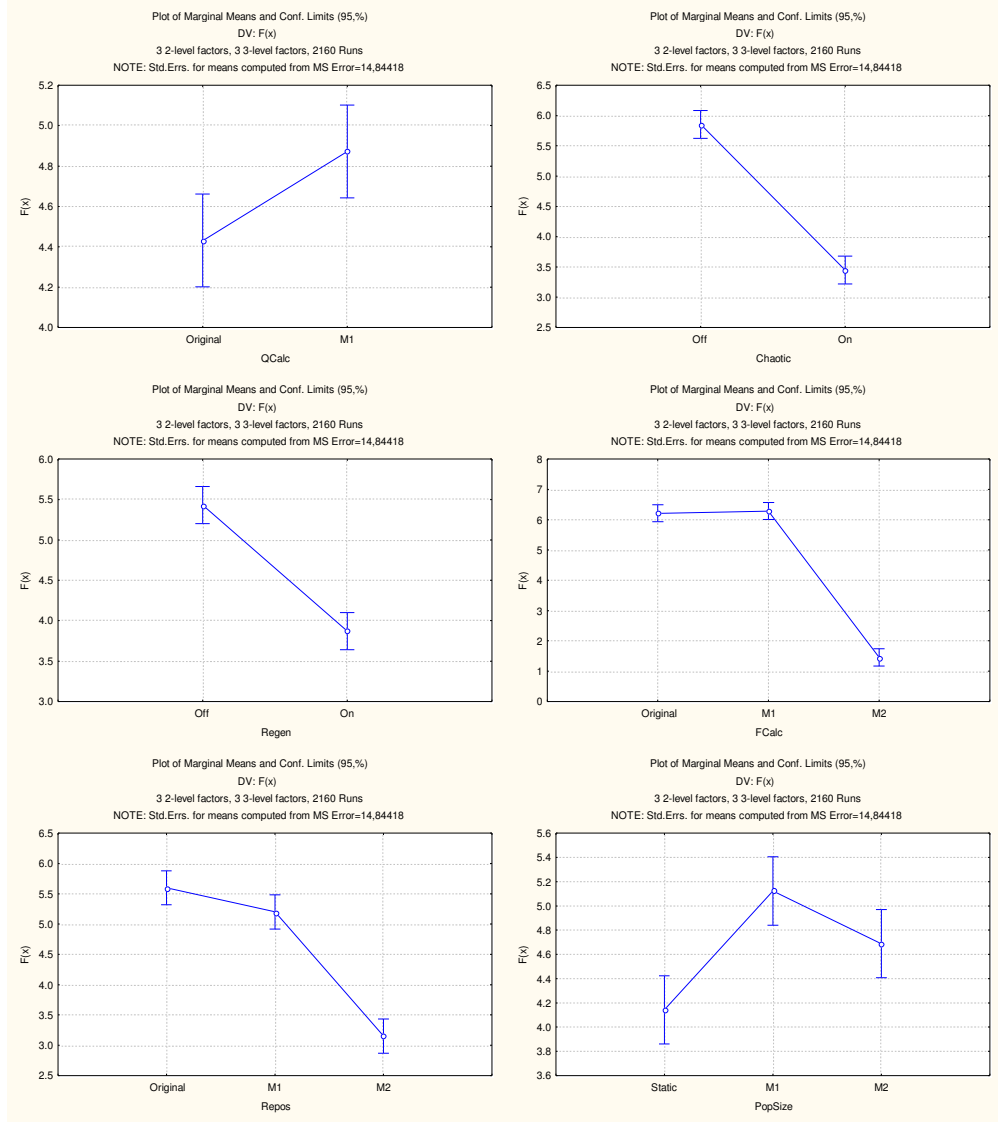
Şekil Ek 5.29 Ackley.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,36558; Adj:,36293 (Ackley.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=14,84418					
Factor	SS	df	MS	F	p
(1)QCalc L	105.32	1	105.324	7.0953	0.007786
(2)Chaotic L	3126.99	1	3126.986	210.6540	0.000000
(3)Regen L	1317.38	1	1317.377	88.7471	0.000000
(4)FCalc L+Q	11016.67	2	5508.333	371.0769	0.000000
(5)Repos L+Q	2477.50	2	1238.749	83.4501	0.000000
(6)PopSize L+Q	347.01	2	173.506	11.6885	0.000009
Error	31914.99	2150	14.844		
Total SS	50305.85	2159			

Şekil Ek 5.30 Ackley.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.31 Ackley.10 problemi için ortalama etkiler



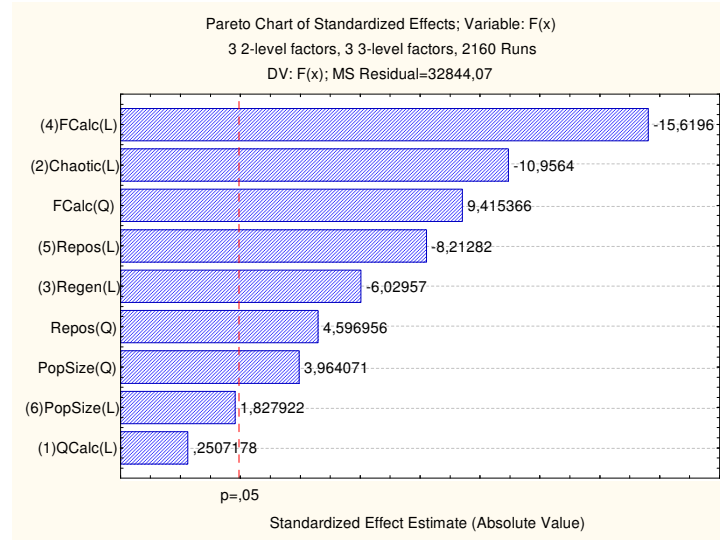
Şekil Ek 5.32 Ackley.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,21725; Adj:,.21397 (Ackley.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=32844,07										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	147.268	3.899434	37.7664	0.000000	139.621	154.915	147.2676	3.899434	139.6206	154.9147
(1)QCalc (L)	1.955	7.798869	0.2507	0.802056	-13.339	17.249	0.9777	3.899434	-6.6694	8.6247
(2)Chaotic (L)	-85.447	7.798869	-10.9564	0.000000	-100.741	-70.153	-42.7236	3.899434	-50.3707	-35.0766
(3)Regen (L)	-47.024	7.798869	-6.0296	0.000000	-62.318	-31.730	-23.5119	3.899434	-31.1590	-15.8649
(4)FCalc (L)	-149.193	9.551625	-15.6196	0.000000	-167.924	-130.461	-74.5963	4.775812	-83.9620	-65.2306
FCalc (Q)	77.883	8.271950	9.4154	0.000000	61.662	94.105	38.9417	4.135975	30.8308	47.0526
(5)Repos (L)	-78.446	9.551625	-8.2128	0.000000	-97.177	-59.714	-39.2229	4.775812	-48.5886	-29.8572
Repos (Q)	38.026	8.271950	4.5970	0.000005	21.804	54.248	19.0129	4.135975	10.9020	27.1238
(6)PopSize (L)	17.460	9.551625	1.8279	0.067700	-1.272	36.191	8.7298	4.775812	-0.6359	18.0955
PopSize (Q)	32.791	8.271950	3.9641	0.000076	16.569	49.012	16.3953	4.135975	8.2844	24.5062

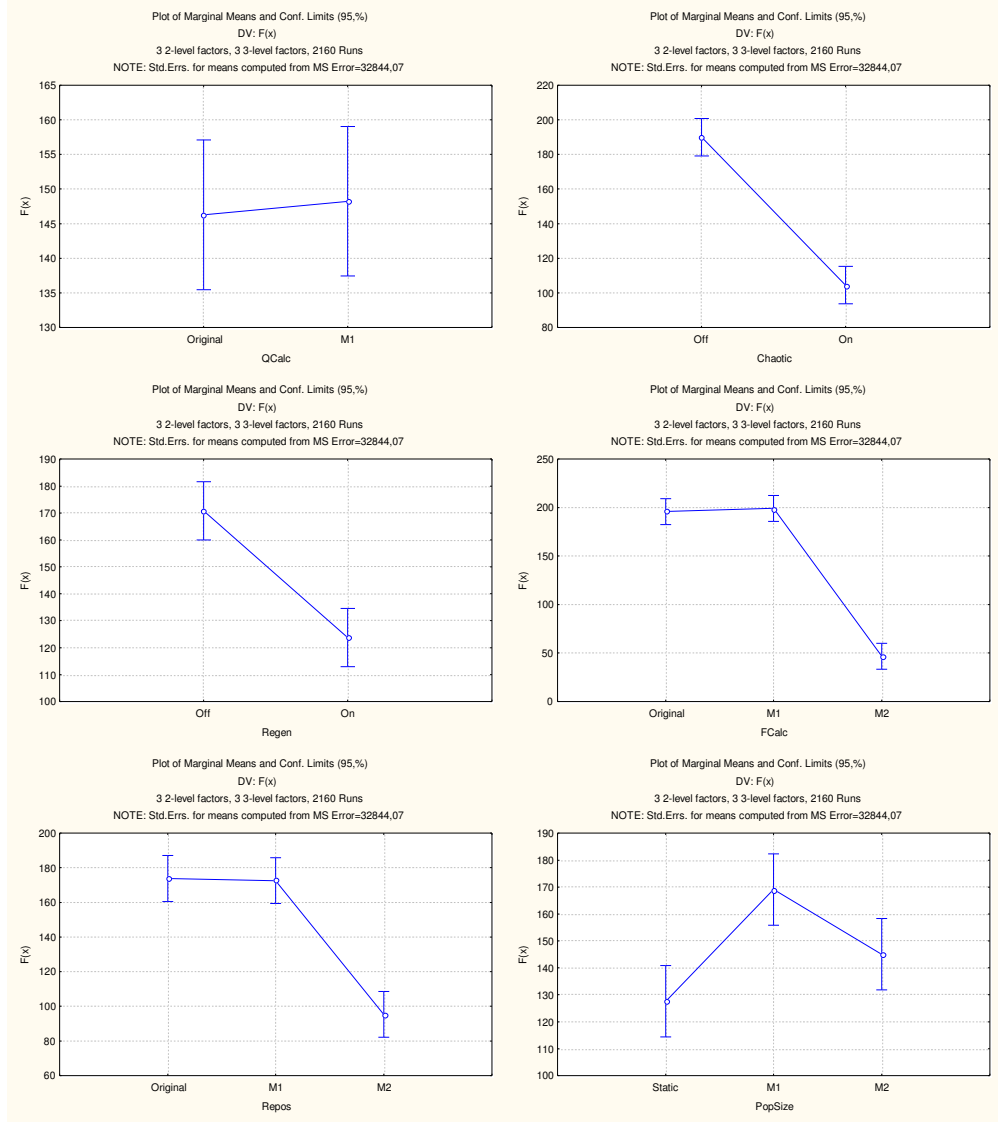
Şekil Ek 5.33 Ackley.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,21725; Adj:,.21397 (Ackley.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=32844,07					
Factor	SS	df	MS	F	p
(1)QCalc L	2065	1	2065	0.0629	0.802056
(2)Chaotic L	3942663	1	3942663	120.0419	0.000000
(3)Regen L	1194069	1	1194069	36.3557	0.000000
(4)FCalc L+Q	10924634	2	5462317	166.3106	0.000000
(5)Repos L+Q	2909405	2	1454703	44.2912	0.000000
(6)PopSize L+Q	625849	2	312924	9.5276	0.000076
Error	70614756	2150	32844		
Total SS	90213440	2159			

Şekil Ek 5.34 Ackley.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.35 Ackley.50 problemi için ortalama etkiler



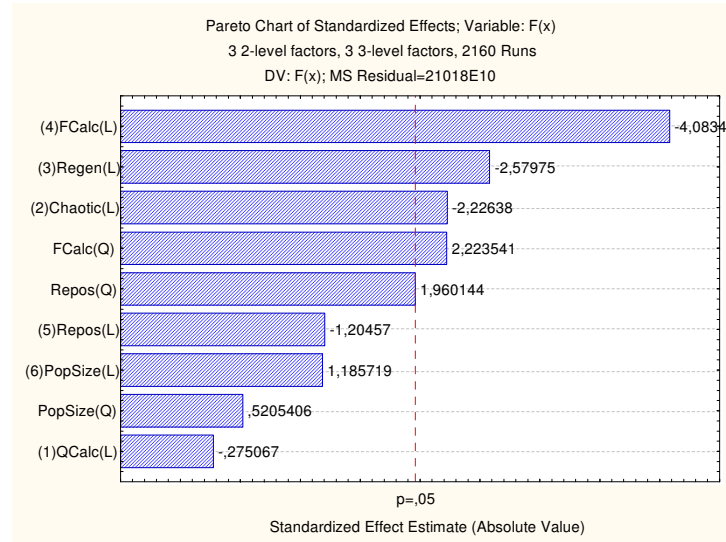
Şekil Ek 5.36 Ackley.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,01839; Adj:,01428 (F2.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=21018E10										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	2427701	311935.8	7.78270	0.000000	1815974	3039429	2427701	311935.8	1815974	3039429
(1)QCalc (L)	-171607	623871.6	-0.27507	0.783291	-1395061	1051848	-85803	311935.8	-697531	525924
(2)Chaotic (L)	-1388973	623871.6	-2.22638	0.026092	-2612428	-165518	-694486	311935.8	-1306214	-82759
(3)Regen (L)	-1609430	623871.6	-2.57975	0.009953	-2832884	-385975	-804715	311935.8	-1416442	-192988
(4)FCalc (L)	-3120061	764083.5	-4.08340	0.000046	-4618480	-1621641	-1560030	382041.8	-2309240	-810820
FCalc (Q)	1471352	661715.7	2.22354	0.026283	173683	2769022	735676	330857.9	86841	1384511
(5)Repos (L)	-920390	764083.5	-1.20457	0.228503	-2418810	578030	-460195	382041.8	-1209405	289015
Repos (Q)	1297058	661715.7	1.96014	0.050108	-611	2594728	648529	330857.9	-306	1297364
(6)PopSize (L)	905988	764083.5	1.18572	0.235864	-592432	2404408	452994	382041.8	-296216	1202204
PopSize (Q)	344450	661715.7	0.52054	0.602740	-953220	1642120	172225	330857.9	-476610	821060

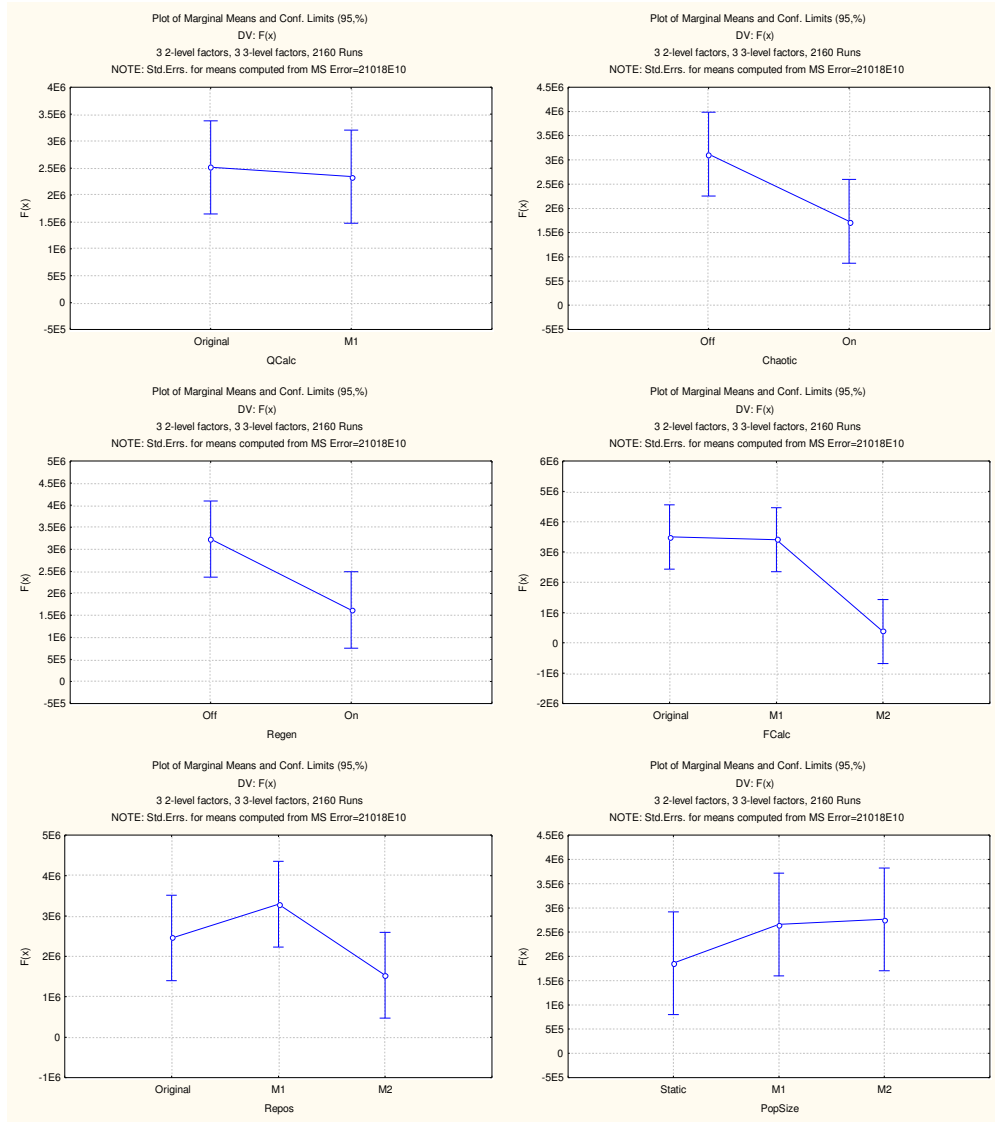
Şekil Ek 5.37 F2.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,01839; Adj:,01428 (F2.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=21018E10					
Factor	SS	df	MS	F	p
(1)QCalc L	1.590235E+13	1	1.590235E+13	0.07566	0.783291
(2)Chaotic L	1.041793E+15	1	1.041793E+15	4.95675	0.026092
(3)Regen L	1.398743E+15	1	1.398743E+15	6.65509	0.009953
(4)FCalc L+Q	4.543661E+15	2	2.271831E+15	10.80916	0.000021
(5)Repos L+Q	1.112495E+15	2	5.562476E+14	2.64657	0.071125
(6)PopSize L+Q	3.524431E+14	2	1.762216E+14	0.83845	0.432523
Error	4.518795E+17	2150	2.101765E+14		
Total SS	4.603445E+17	2159			

Şekil Ek 5.38 F2.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.39 F2.30 problemi için ortalama etkiler



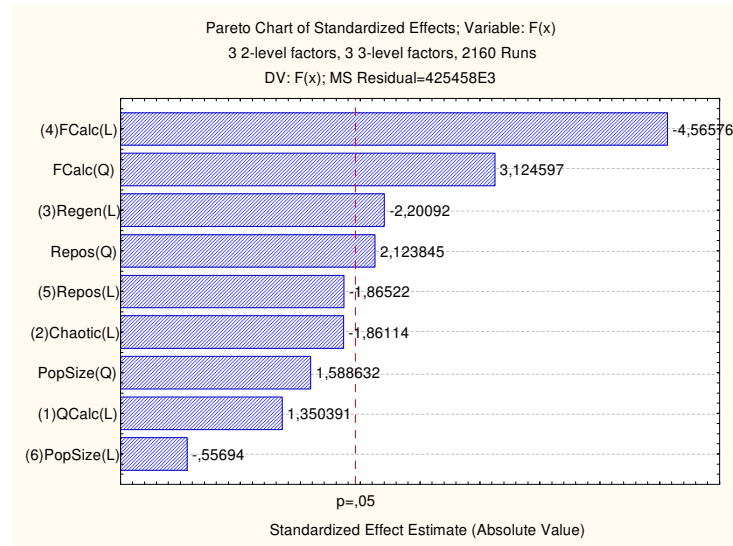
Şekil Ek 5.40 F2.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,02342; Adj:,01933 (F2.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=425458E3										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	4320.22	443.814	9.73430	0.000000	3449.87	5190.57	4320.22	443.8143	3449.87	5190.57
(1)QCalc (L)	1198.65	887.629	1.35039	0.177033	-542.05	2939.35	599.32	443.8143	-271.03	1469.67
(2)Chaotic (L)	-1652.00	887.629	-1.86114	0.062861	-3392.70	88.70	-826.00	443.8143	-1696.35	44.35
(3)Regen (L)	-1953.60	887.629	-2.20092	0.027848	-3694.30	-212.90	-976.80	443.8143	-1847.15	-106.45
(4)FCalc (L)	-4963.52	1087.119	-4.56576	0.000005	-7095.43	-2831.60	-2481.76	543.5593	-3547.72	-1415.80
FCalc (Q)	2941.72	941.472	3.12460	0.001804	1095.43	4788.01	1470.86	470.7362	547.71	2394.01
(5)Repos (L)	-2027.72	1087.119	-1.86522	0.062286	-4159.63	104.20	-1013.86	543.5593	-2079.82	52.10
Repos (Q)	1999.54	941.472	2.12384	0.033797	153.25	3845.83	999.77	470.7362	76.62	1922.92
(6)PopSize (L)	-605.46	1087.119	-0.55694	0.577626	-2737.37	1526.45	-302.73	543.5593	-1368.69	763.23
PopSize (Q)	1495.65	941.472	1.58863	0.112290	-350.64	3341.94	747.83	470.7362	-175.32	1670.97

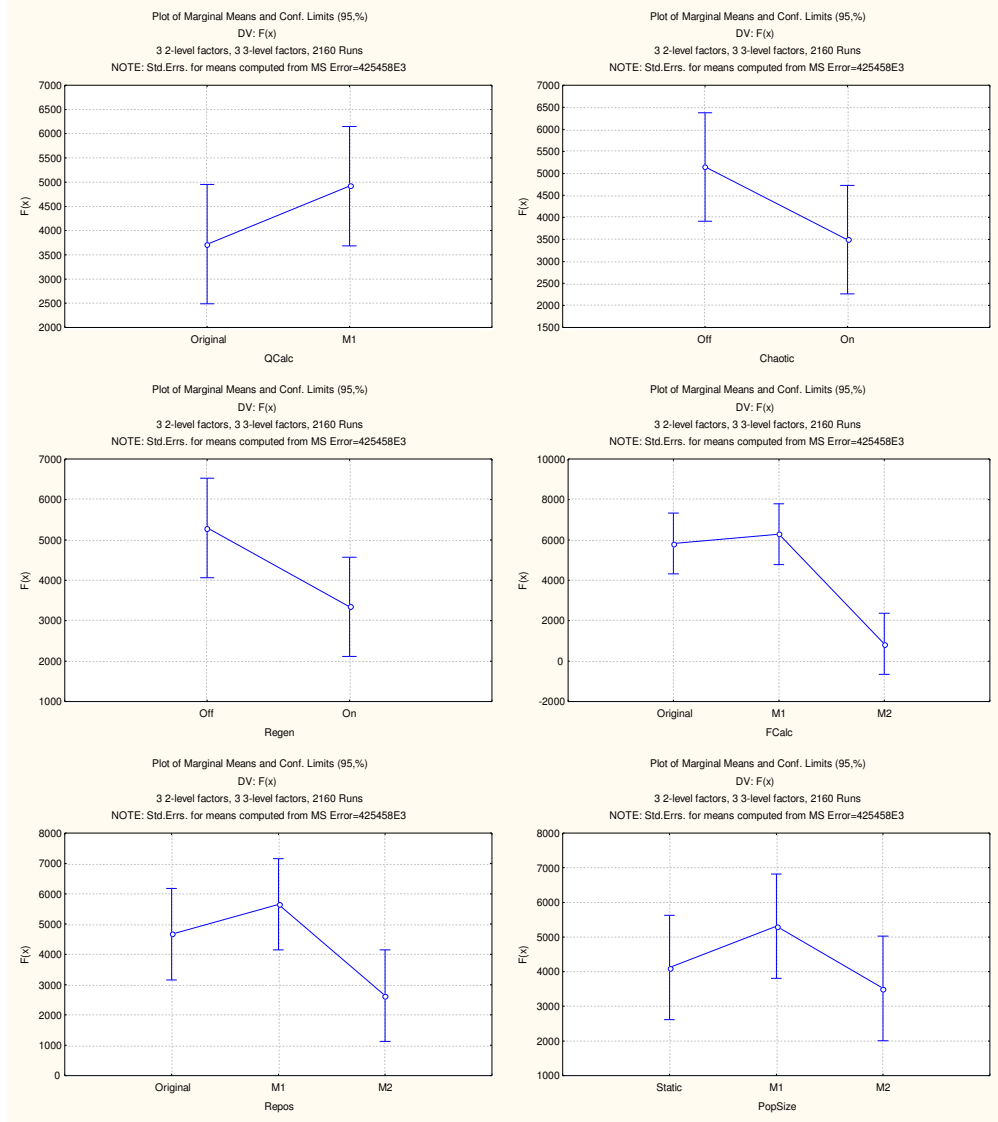
Şekil Ek 5.41 F2.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,02342; Adj:,01933 (F2.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=425458E3					
Factor	SS	df	MS	F	p
(1)QCalc L	7.758463E+08	1	7.758463E+08	1.82356	0.177033
(2)Chaotic L	1.473712E+09	1	1.473712E+09	3.46383	0.062861
(3)Regen L	2.060930E+09	1	2.060930E+09	4.84403	0.027848
(4)FCalc L+Q	1.302293E+10	2	6.511467E+09	15.30462	0.000000
(5)Repos L+Q	3.399308E+09	2	1.699654E+09	3.99488	0.018546
(6)PopSize L+Q	1.205719E+09	2	6.028597E+08	1.41697	0.242674
Error	9.147340E+11	2150	4.254577E+08		
Total SS	9.366724E+11	2159			

Şekil Ek 5.42 F2.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.43 F2.10 problemi için ortalama etkiler



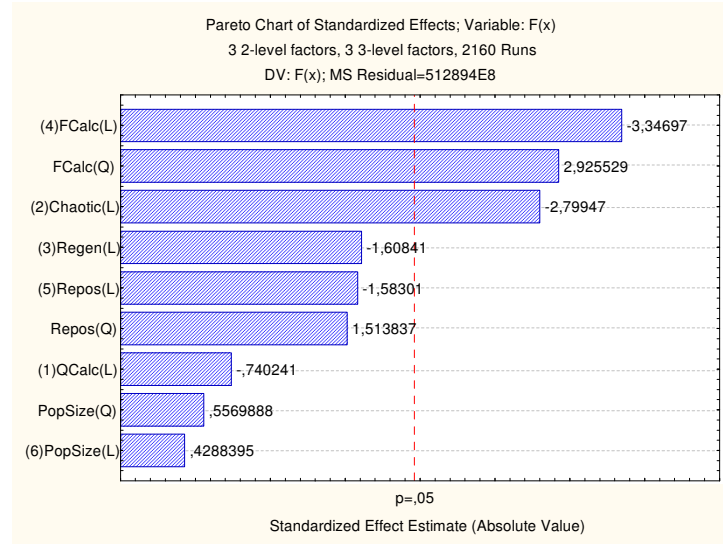
Şekil Ek 5.44 F2.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,01648; Adj:.,01236 (F2.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=512894E8										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	1195376	154094.5	7.75742	0.00000	893186	1497566	1195376	154094.5	893186	1497566
(1)QCalc (L)	-228134	308189.0	-0.74024	0.459235	-832514	376246	-114067	154094.5	-416257	188123
(2)Chaotic (L)	-862767	308189.0	-2.79947	0.005164	-1467147	-258388	-431384	154094.5	-733574	-129194
(3)Regen (L)	-495695	308189.0	-1.60841	0.107892	-1100074	108685	-247847	154094.5	-550037	54343
(4)FCalc (L)	-1263325	377452.9	-3.34697	0.000831	-2003536	-523114	-631663	188726.5	-1001766	-261557
FCalc (Q)	956308	326883.8	2.92553	0.003475	315267	1597350	478154	163441.9	157633	798675
(5)Repos (L)	-597514	377452.9	-1.58301	0.113565	-1337725	142697	-298757	188726.5	-668862	71349
Repos (Q)	494849	326883.8	1.51384	0.130214	-146193	1135890	247424	163441.9	-73096	567945
(6)PopSize (L)	161867	377452.9	0.42884	0.668083	-578344	902078	80933	188726.5	-289172	451039
PopSize (Q)	182071	326883.8	0.55699	0.577593	-458971	823112	91035	163441.9	-229485	411556

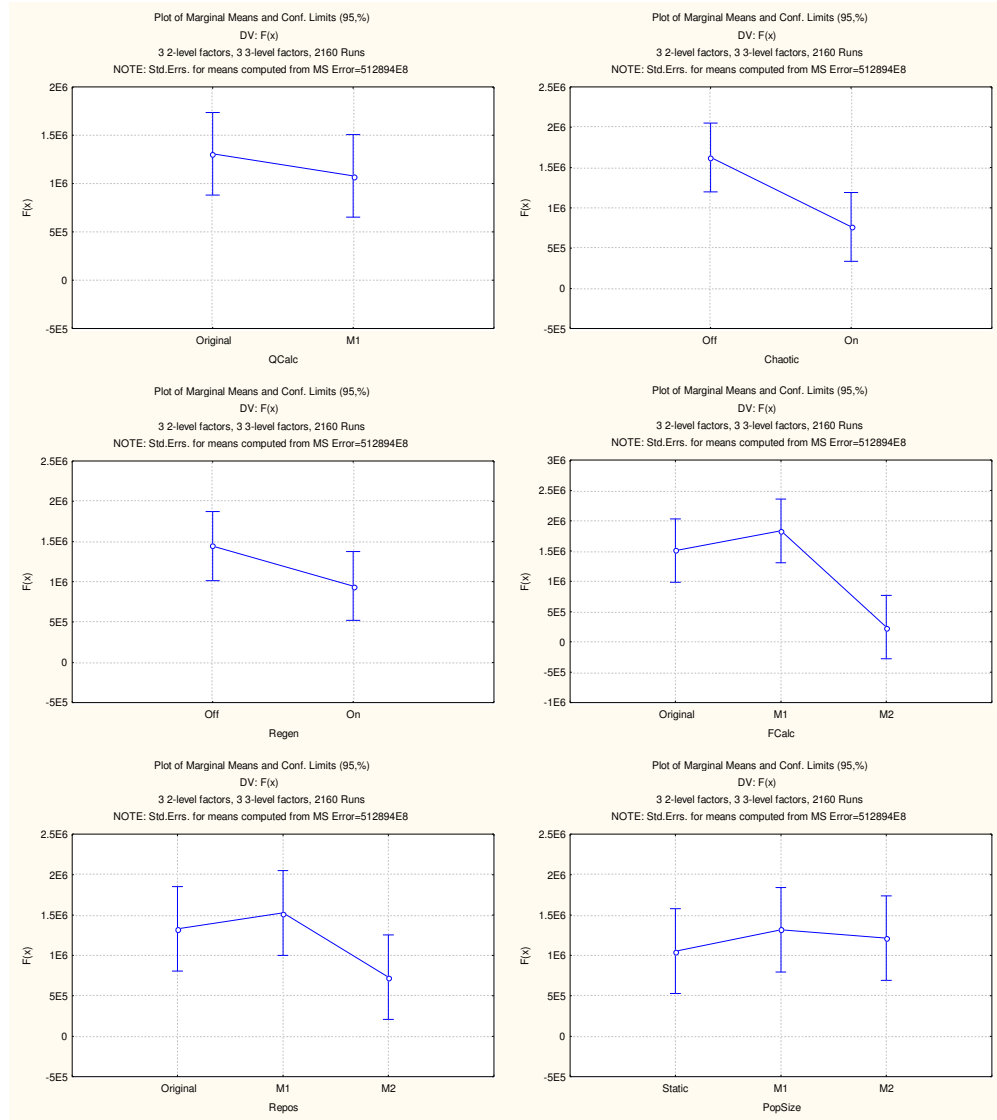
Şekil Ek 5.45 F2.50 problemi için tahmini etki değerleri

ANOVA; Var.:F(x); R-sqr=,01648; Adj:.,01236 (F2.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=512894E8					
Factor	SS	df	MS	F	p
(1)QCalc L	2.810437E+13	1	2.810437E+13	0.547956	0.459235
(2)Chaotic L	4.019586E+14	1	4.019586E+14	7.837059	0.005164
(3)Regen L	1.326851E+14	1	1.326851E+14	2.586985	0.107892
(4)FCalc L+Q	1.013528E+15	2	5.067642E+14	9.880474	0.000054
(5)Repos L+Q	2.460683E+14	2	1.230341E+14	2.398819	0.091068
(6)PopSize L+Q	2.534417E+13	2	1.267208E+13	0.247070	0.781108
Error	1.102723E+17	2150	5.128946E+13		
Total SS	1.121200E+17	2159			

Şekil Ek 5.46 F2.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.47 F2.50 problemi için ortalama etkiler



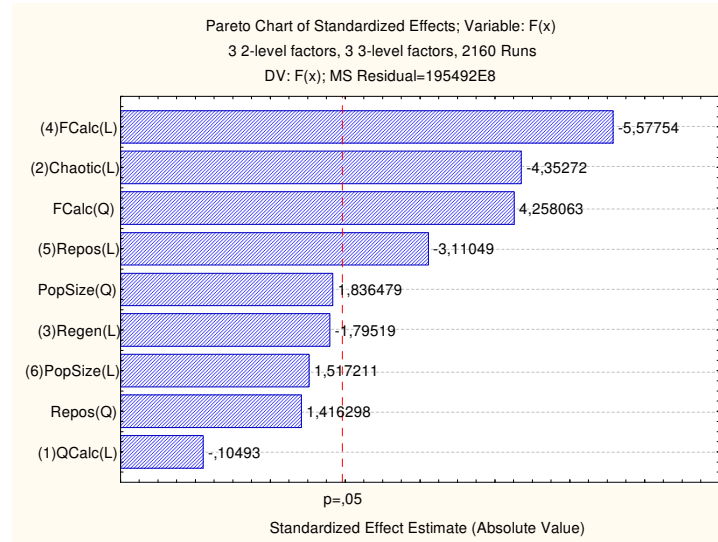
Şekil Ek 5.48 F2.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,03965; Adj:,03563 (F4.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=195492E8										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	1192856	95134.6	12.53862	0.000000	1006291	1379422	1192856	95134.6	1006291	1379422
(1)QCalc (L)	-19965	190269.2	-0.10493	0.916441	-393096	353166	-9983	95134.6	-196548	176583
(2)Chaotic (L)	-828188	190269.2	-4.35272	0.000014	-1201319	-455057	-414094	95134.6	-600659	-227529
(3)Regen (L)	-341570	190269.2	-1.79519	0.072763	-714700	31561	-170785	95134.6	-357350	15781
(4)FCalc (L)	-1299740	233031.2	-5.57754	0.000000	-1756730	-842750	-649870	116515.6	-878365	-421375
FCalc (Q)	859324	201810.9	4.25806	0.000022	463559	1255088	429662	100905.5	231779	627544
(5)Repos (L)	-724841	233031.2	-3.11049	0.001892	-1181831	-267851	-362420	116515.6	-590915	-133925
Repos (Q)	285824	201810.9	1.41630	0.156833	-109941	681589	142912	100905.5	-54970	340795
(6)PopSize (L)	353557	233031.2	1.51721	0.129360	-103433	810547	176779	116515.6	-51716	405274
PopSize (Q)	370622	201810.9	1.83648	0.066425	-25143	766387	185311	100905.5	-12572	383193

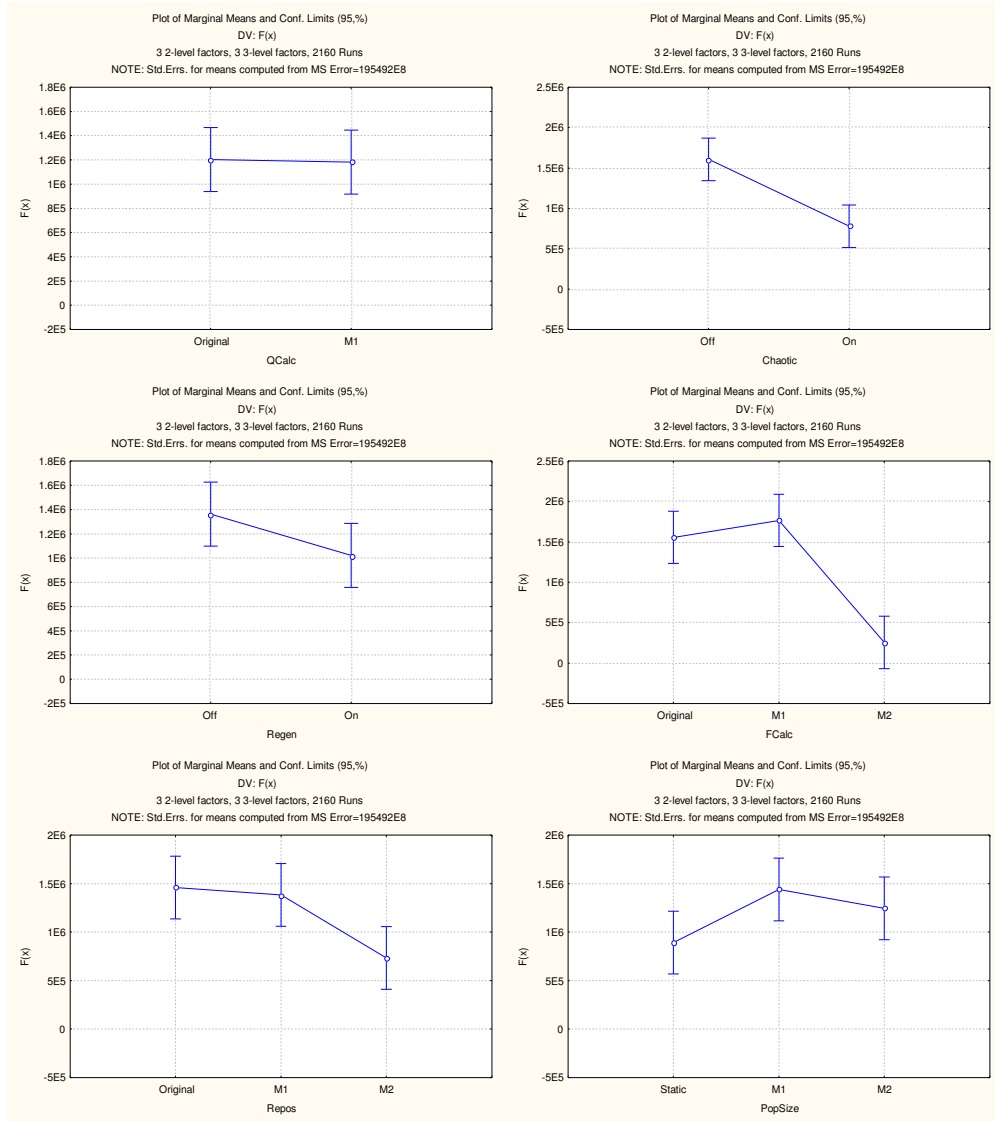
Şekil Ek 5.49 F4.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,03965; Adj:,03563 (F4.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=195492E8						
Factor	SS	df	MS	F	p	
(1)QCalc L	2.152453E+11	1	2.152453E+11	0.01101	0.916441	
(2)Chaotic L	3.703834E+14	1	3.703834E+14	18.94615	0.000014	
(3)Regen L	6.300171E+13	1	6.300171E+13	3.22271	0.072763	
(4)FCalc L+Q	9.626066E+14	2	4.813033E+14	24.62002	0.000000	
(5)Repos L+Q	2.283558E+14	2	1.141779E+14	5.84052	0.002954	
(6)PopSize L+Q	1.109340E+14	2	5.546700E+13	2.83729	0.058803	
Error	4.203093E+16	2150	1.954927E+13			
Total SS	4.376643E+16	2159				

Şekil Ek 5.50 F4.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.51 F4.30 problemi için ortalama etkiler



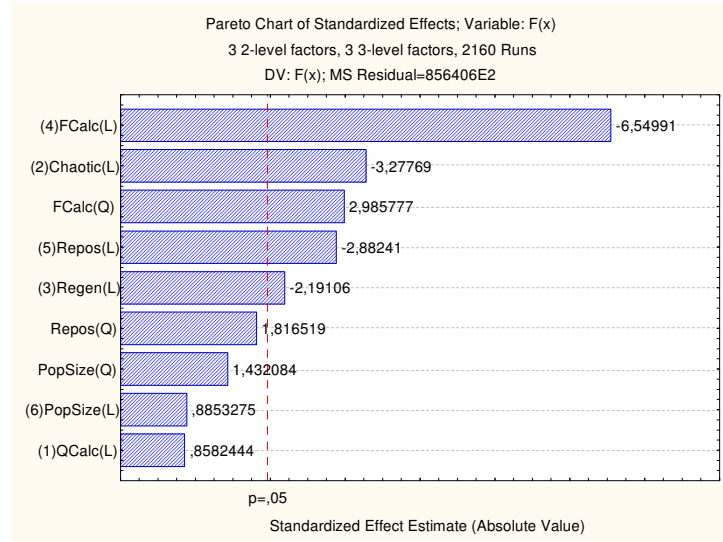
Şekil Ek 5.52 F4.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,03697; Adj:,03294 (F4.10.sta)										
3 2-level factors, 3 3-level factors, 2160 Runs										
DV: F(x); MS Residual=856406E2										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	2536.10	199.1191	12.73659	0.000000	2145.61	2926.58	2536.10	199.1191	2145.61	2926.58
(1)QCalc (L)	341.79	398.2382	0.85824	0.390853	-439.19	1122.76	170.89	199.1191	-219.59	561.38
(2)Chaotic (L)	-1305.30	398.2382	-3.27769	0.001063	-2086.27	-524.33	-652.65	199.1191	-1043.14	-262.16
(3)Regen (L)	-872.56	398.2382	-2.19106	0.028554	-1653.54	-91.59	-436.28	199.1191	-826.77	-45.80
(4)FCalc (L)	-3194.65	487.7402	-6.54991	0.000000	-4151.14	-2238.16	-1597.33	243.8701	-2075.57	-1119.08
FCalc (Q)	1261.18	422.3954	2.98578	0.002861	432.83	2089.52	630.59	211.1977	216.42	1044.76
(5)Repos (L)	-1405.87	487.7402	-2.88241	0.003986	-2362.36	-449.37	-702.93	243.8701	-1181.18	-224.69
Repos (Q)	767.29	422.3954	1.81652	0.069430	-61.06	1595.64	383.64	211.1977	-30.53	797.82
(6)PopSize (L)	431.81	487.7402	0.88533	0.376079	-524.68	1388.30	215.90	243.8701	-262.34	694.15
PopSize (Q)	604.91	422.3954	1.43208	0.152265	-223.44	1433.25	302.45	211.1977	-111.72	716.63

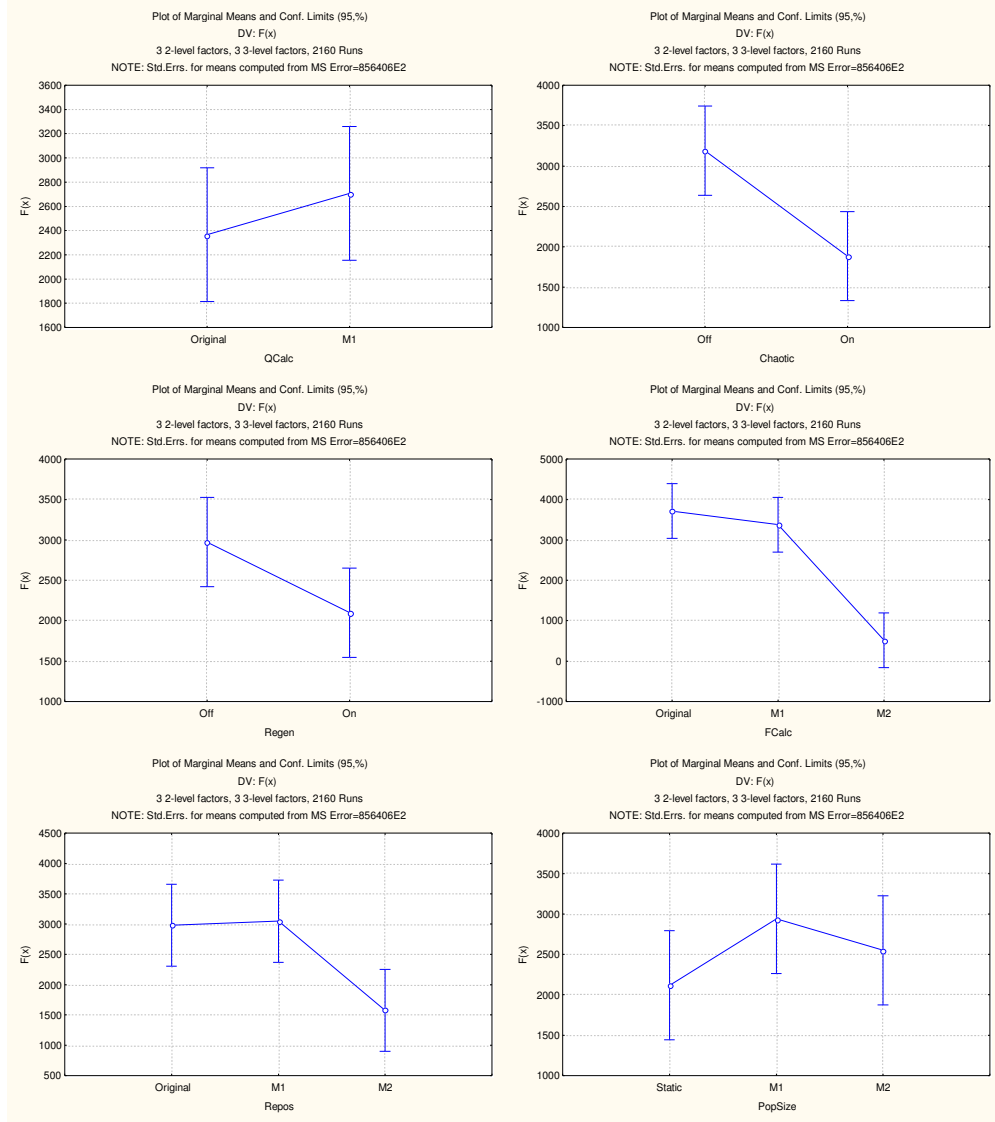
Şekil Ek 5.53 F4.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,03697; Adj:,03294 (F4.10.sta)						
3 2-level factors, 3 3-level factors, 2160 Runs						
DV: F(x); MS Residual=856406E2						
Factor	SS	df	MS	F	p	
(1)QCalc L	6.308145E+07	1	6.308145E+07	0.73658	0.390853	
(2)Chaotic L	9.200585E+08	1	9.200585E+08	10.74325	0.001063	
(3)Regen L	4.111396E+08	1	4.111396E+08	4.80076	0.028554	
(4)FCalc L+Q	4.437565E+09	2	2.218783E+09	25.90807	0.000000	
(5)Repos L+Q	9.941170E+08	2	4.970585E+08	5.80401	0.003063	
(6)PopSize L+Q	2.427627E+08	2	1.213813E+08	1.41733	0.242586	
Error	1.841273E+11	2150	8.564060E+07			
Total SS	1.911960E+11	2159				

Şekil Ek 5.54 F4.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.55 F4.10 problemi için ortalama etkiler



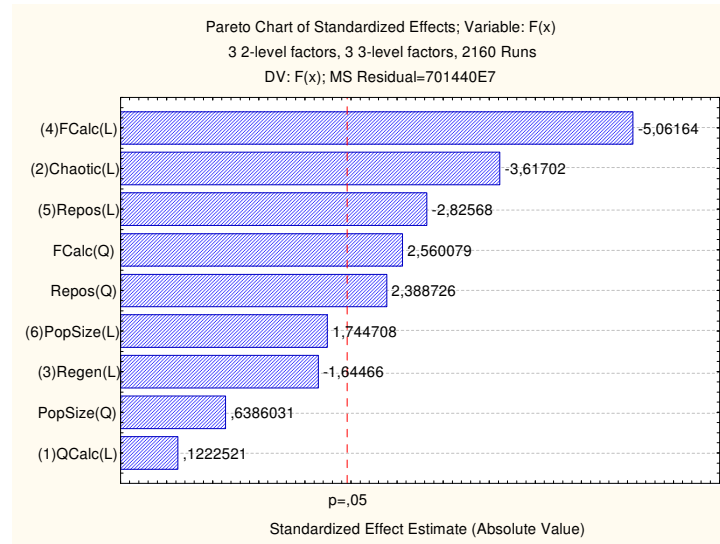
Şekil Ek 5.56 F4.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,0294; Adj:02533 (F4.50.sta)										
3 2-level factors, 3 3-level factors, 2160 Runs										
DV: F(x); MS Residual=701440E7										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	610764	56986.0	10.71779	0.000000	499010	722517	610764	56986.01	499010	722517
(1)QCalc (L)	13933	113972.0	0.12225	0.902711	-209574	237440	6967	56986.01	-104787	118720
(2)Chaotic (L)	-412239	113972.0	-3.61702	0.000305	-635746	-188733	-206120	56986.01	-317873	-94366
(3)Regen (L)	-187445	113972.0	-1.64466	0.100187	-410952	36062	-93722	56986.01	-205476	18031
(4)FCalc (L)	-706538	139586.6	-5.06164	0.000000	-980277	-432799	-353269	69793.32	-490138	-216399
FCalc (Q)	309477	120885.6	2.56008	0.010533	72412	546542	154738	60442.79	36206	273271
(5)Repos (L)	-394427	139586.6	-2.82568	0.004762	-668166	-120689	-197214	69793.32	-334083	-60344
Repos (Q)	288763	120885.6	2.38873	0.016993	51698	525827	144381	60442.79	25849	262914
(6)PopSize (L)	243538	139586.6	1.74471	0.081179	-30201	517277	121769	69793.32	-15101	258638
PopSize (Q)	77198	120885.6	0.63860	0.523149	-159867	314263	38599	60442.79	-79933	157131

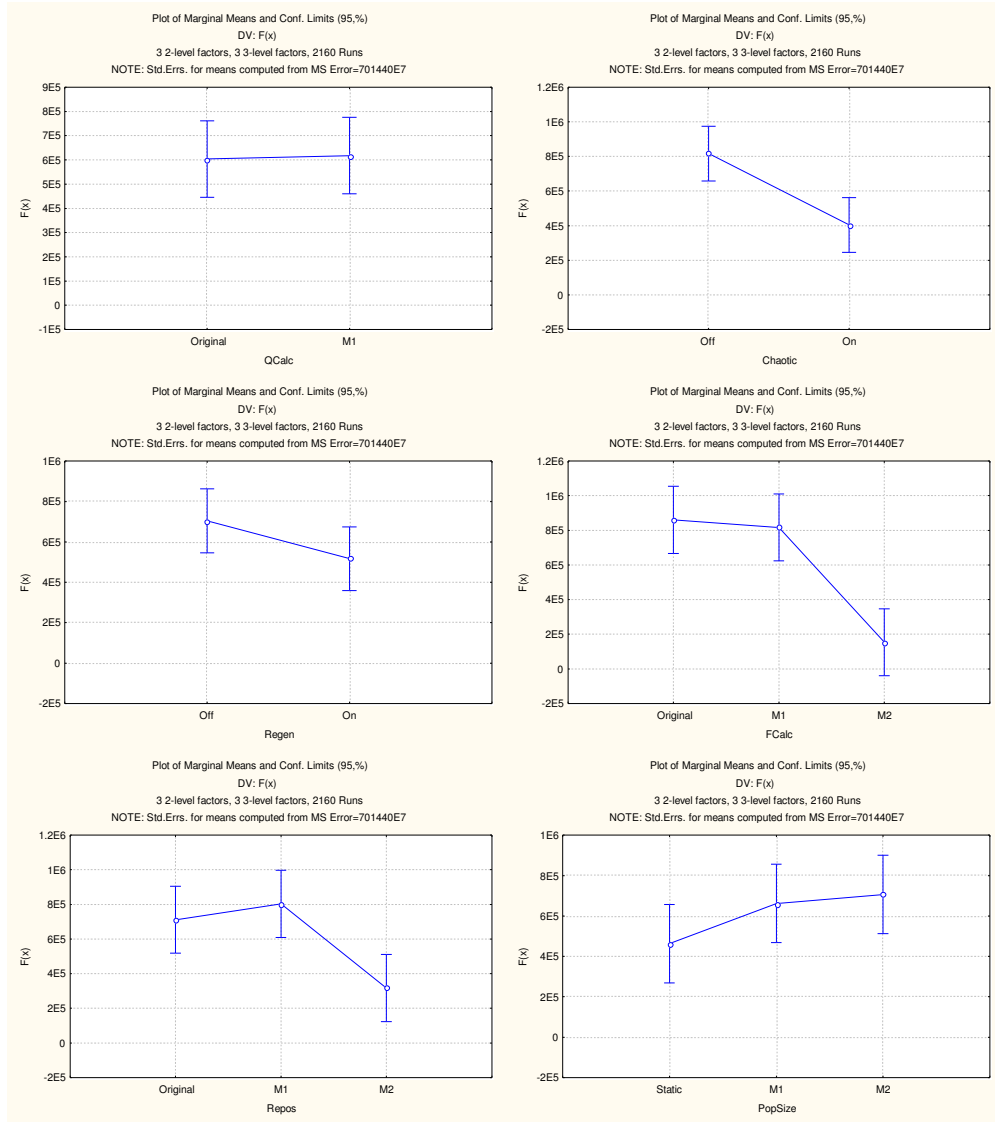
Şekil Ek 5.57 F4.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,0294; Adj:02533 (F4.50.sta)					
3 2-level factors, 3 3-level factors, 2160 Runs					
DV: F(x); MS Residual=701440E7					
Factor	SS	df	MS	F	p
(1)QCalc L	1.048341E+11	1	1.048341E+11	0.01495	0.902711
(2)Chaotic L	9.176836E+13	1	9.176836E+13	13.08286	0.000305
(3)Regen L	1.897319E+13	1	1.897319E+13	2.70489	0.100187
(4)FCalc L+Q	2.256829E+14	2	1.128414E+14	16.08712	0.000000
(5)Repos L+Q	9.603052E+13	2	4.801526E+13	6.84525	0.001088
(6)PopSize L+Q	2.421242E+13	2	1.210621E+13	1.72591	0.178258
Error	1.508095E+16	2150	7.014395E+12		
Total SS	1.553772E+16	2159			

Şekil Ek 5.58 F4.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.59 F4.50 problemi için ortalama etkiler



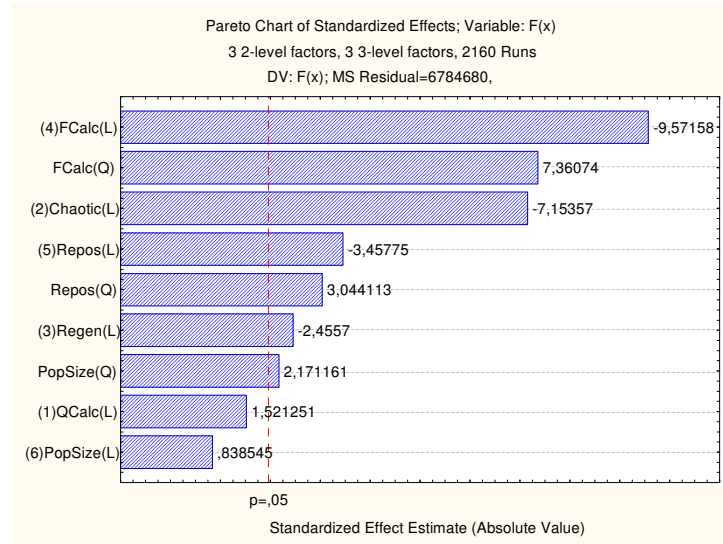
Şekil Ek 5.60 F4.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,09738; Adj:,.0936 (F10.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=6784680,										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	1228.02	56.0451	21.91134	0.000000	1118.12	1337.93	1228.024	56.04512	1118.115	1337.932
(1)QCalc (L)	170.52	112.0902	1.52125	0.128344	-49.30	390.33	85.259	56.04512	-24.650	195.167
(2)Chaotic (L)	-801.85	112.0902	-7.15357	0.000000	-1021.66	-582.03	-400.923	56.04512	-510.831	-291.014
(3)Regen (L)	-275.26	112.0902	-2.45570	0.014140	-495.08	-55.44	-137.630	56.04512	-247.538	-27.721
(4)FCalc (L)	-1314.01	137.2819	-9.57158	0.000000	-1583.22	-1044.79	-657.003	68.64097	-791.612	-522.393
FCalc (Q)	875.12	118.8897	7.36074	0.000000	641.97	1108.27	437.558	59.44483	320.983	554.133
(5)Repos (L)	-474.69	137.2819	-3.45775	0.000555	-743.91	-205.47	-237.344	68.64097	-371.953	-102.734
Repos (Q)	361.91	118.8897	3.04411	0.002362	128.76	595.06	180.957	59.44483	64.381	297.532
(6)PopSize (L)	115.12	137.2819	0.83855	0.401818	-154.10	384.34	57.559	68.64097	-77.051	192.168
PopSize (Q)	258.13	118.8897	2.17116	0.030028	24.98	491.28	129.064	59.44483	12.489	245.640

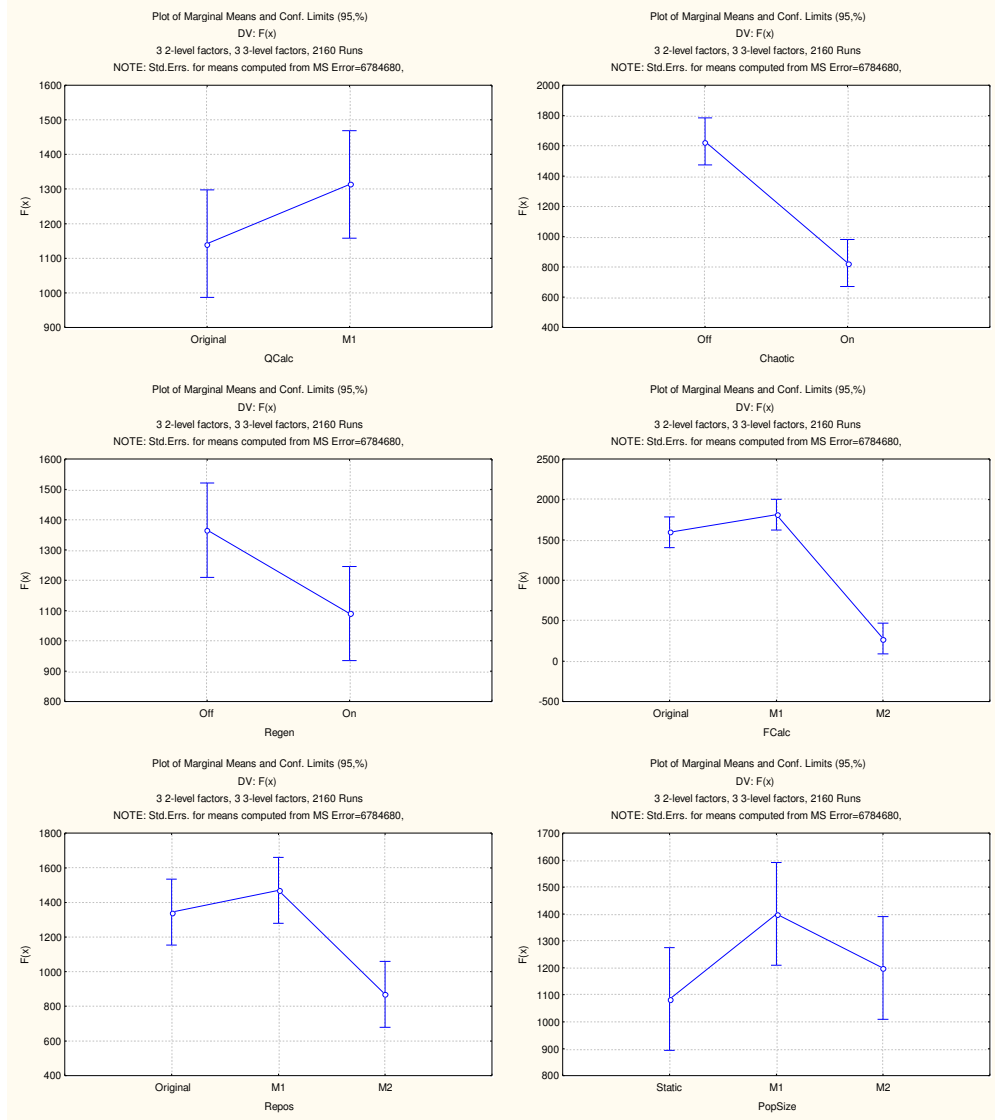
Şekil Ek 5.61 F10.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,09738; Adj:,.0936 (F10.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=6784680,					
Factor	SS	df	MS	F	p
(1)QCalc L	1.570114E+07	1	15701138	2.31420	0.128344
(2)Chaotic L	3.471962E+08	1	347196185	51.17355	0.000000
(3)Regen L	4.091462E+07	1	40914623	6.03044	0.014140
(4)FCalc L+Q	9.891768E+08	2	494588379	72.89782	0.000000
(5)Repos L+Q	1.439891E+08	2	71994569	10.61134	0.000026
(6)PopSize L+Q	3.675328E+07	2	18376640	2.70855	0.066861
Error	1.458706E+10	2150	6784680		
Total SS	1.616079E+10	2159			

Şekil Ek 5.62 F10.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.63 F10.10 problemi için ortalama etkiler



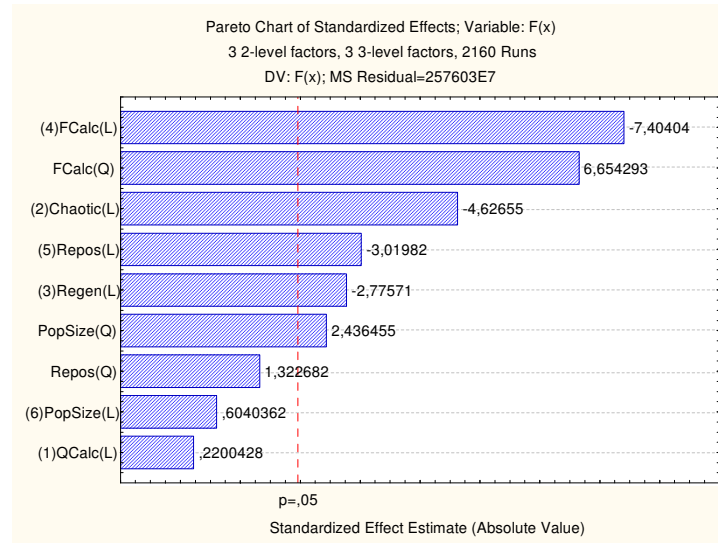
Şekil Ek 5.64 F10.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,06336; Adj:,05943 (F10.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=257603E7										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	600510	34534.15	17.38886	0.000000	532786	668233	600510	34534.15	532786	668233
(1)QCalc (L)	15198	69068.31	0.22004	0.825859	-120250	150646	7599	34534.15	-60125	75323
(2)Chaotic (L)	-319548	69068.31	-4.62655	0.000004	-454995	-184100	-159774	34534.15	-227498	-92050
(3)Regen (L)	-191713	69068.31	-2.77571	0.005556	-327161	-56266	-95857	34534.15	-163580	-28133
(4)FCalc (L)	-626316	84591.05	-7.40404	0.000000	-792204	-460427	-313158	42295.53	-396102	-230213
FCalc (Q)	487480	73258.00	6.65429	0.000000	343816	631144	243740	36629.00	171908	315572
(5)Repos (L)	-255449	84591.05	-3.01982	0.002559	-421338	-89561	-127725	42295.53	-210669	-44780
Repos (Q)	96897	73258.00	1.32268	0.186082	-46767	240561	48449	36629.00	-23383	120281
(6)PopSize (L)	51096	84591.05	0.60404	0.545883	-114793	216985	25548	42295.53	-57396	108492
PopSize (Q)	178490	73258.00	2.43646	0.014913	34826	322154	89245	36629.00	17413	161077

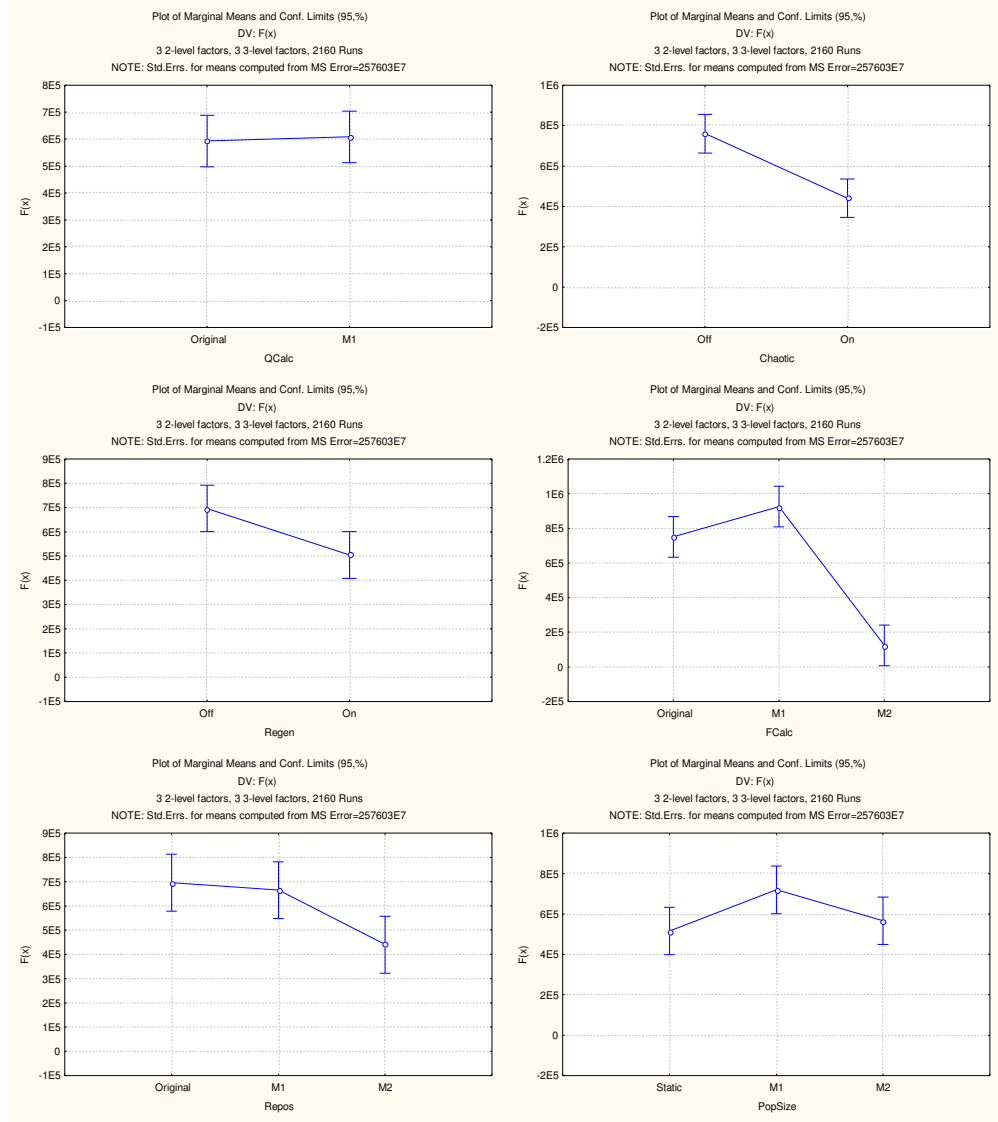
Şekil Ek 5.65 F10.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,06336; Adj:,05943 (F10.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=257603E7					
Factor	SS	df	MS	F	p
(1)QCalc L	1.247285E+11	1	1.247285E+11	0.04842	0.825859
(2)Chaotic L	5.513982E+13	1	5.513982E+13	21.40494	0.000004
(3)Regen L	1.984716E+13	1	1.984716E+13	7.70454	0.005556
(4)FCalc L+Q	2.552834E+14	2	1.276417E+14	49.54971	0.000000
(5)Repos L+Q	2.799833E+13	2	1.399917E+13	5.43439	0.004424
(6)PopSize L+Q	1.623203E+13	2	8.116016E+12	3.15059	0.043025
Error	5.538470E+15	2150	2.576033E+12		
Total SS	5.913096E+15	2159			

Şekil Ek 5.66 F10.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.67 F10.30 problemi için ortalama etkiler



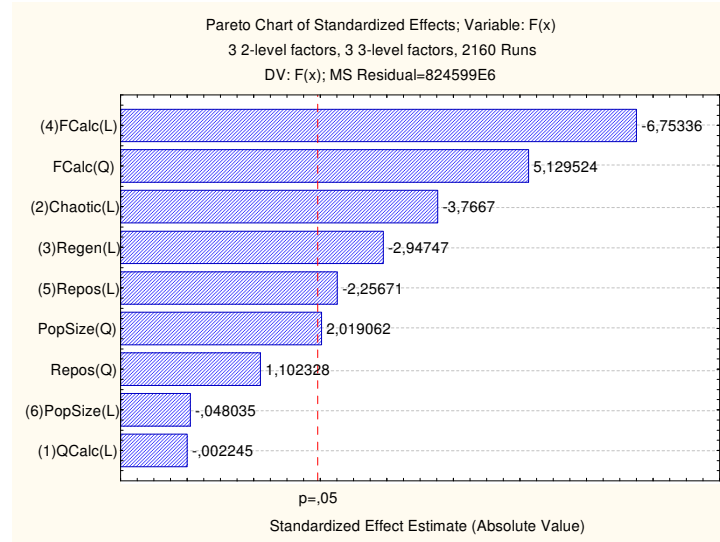
Şekil Ek 5.68 F10.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,04664; Adj:,04265 (F10.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=824599E6										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	291931	19538.65	14.94123	0.000000	253615	330248	291931	19538.65	253615	330248
(1)QCalc (L)	-88	39077.30	-0.00225	0.998209	-76721	76546	-44	19538.65	-38360	38273
(2)Chaotic (L)	-147192	39077.30	-3.76670	0.000170	-223826	-70559	-73596	19538.65	-111913	-35280
(3)Regen (L)	-115179	39077.30	-2.94747	0.003238	-191812	-38546	-57590	19538.65	-95906	-19273
(4)FCalc (L)	-323214	47859.72	-6.75336	0.000000	-417070	-229358	-161607	23929.86	-208535	-114679
FCalc (Q)	212607	41447.73	5.12952	0.000000	131325	293889	106304	20723.87	65663	146944
(5)Repos (L)	-108005	47859.72	-2.25671	0.024126	-201861	-14149	-54003	23929.86	-100931	-7075
Repos (Q)	45689	41447.73	1.10233	0.270443	-35593	126971	22844	20723.87	-17796	63485
(6)PopSize (L)	-2299	47859.72	-0.04803	0.961693	-96155	91557	-1149	23929.86	-48078	45779
PopSize (Q)	83686	41447.73	2.01906	0.043605	2404	164967	41843	20723.87	1202	82484

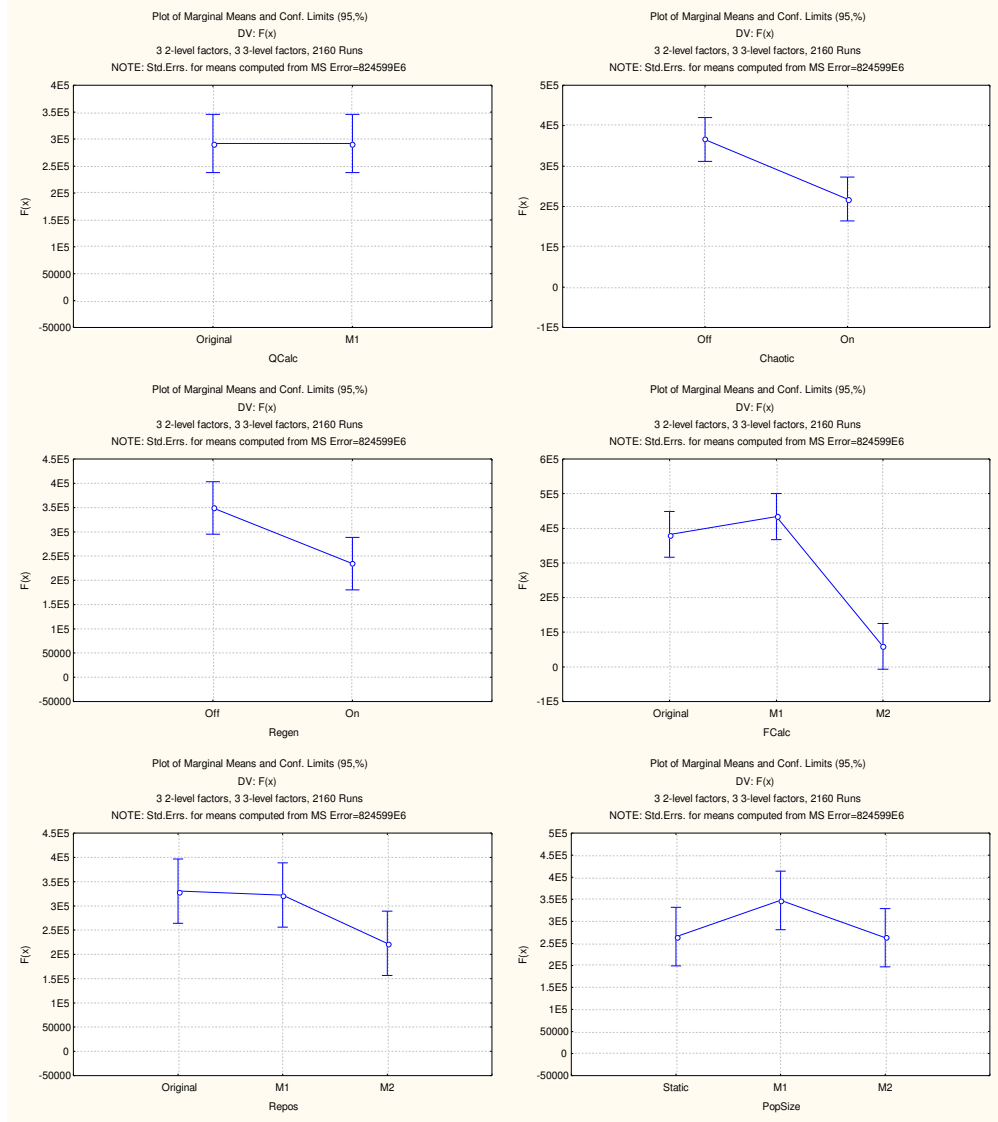
Şekil Ek 5.69 F10.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,04664; Adj:,04265 (F10.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=824599E6						
Factor	SS	df	MS	F	p	
(1)QCalc L	4.156983E+06	1	4.156983E+06	0.00001	0.998209	
(2)Chaotic L	1.169943E+13	1	1.169943E+13	14.18802	0.000170	
(3)Regen L	7.163768E+12	1	7.163768E+12	8.68758	0.003238	
(4)FCalc L+Q	5.930505E+13	2	2.965252E+13	35.95993	0.000000	
(5)Repos L+Q	5.201447E+12	2	2.600723E+12	3.15393	0.042882	
(6)PopSize L+Q	3.363474E+12	2	1.681737E+12	2.03946	0.130351	
Error	1.772888E+15	2150	8.245989E+11			
Total SS	1.859621E+15	2159				

Şekil Ek 5.70 F10.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.71 F10.50 problemi için ortalama etkiler



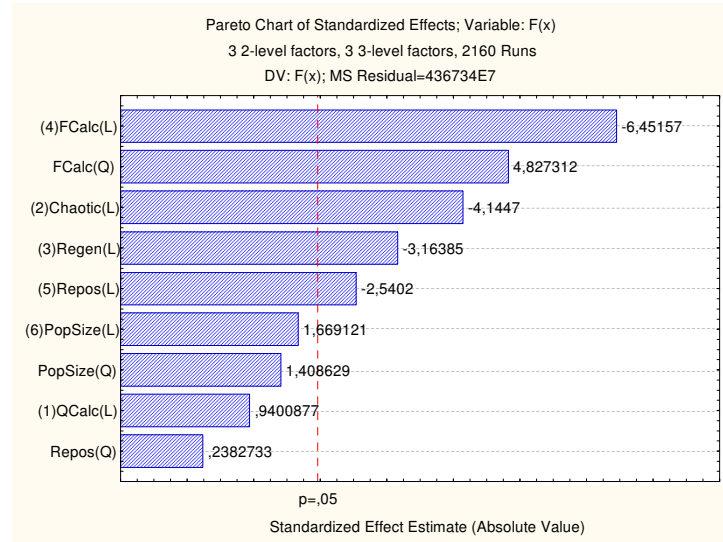
Şekil Ek 5.72 F10.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,04626; Adj:,04227 (F15.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=436734E7										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	641598	44965.7	14.26860	0.000000	553417	729779	641598	44965.74	553417	729779
(1)QCalc (L)	84543	89931.5	0.94009	0.347278	-91818	260905	42272	44965.74	-45909	130453
(2)Chaotic (L)	-372739	89931.5	-4.14470	0.000035	-549101	-196377	-186369	44965.74	-274550	-98189
(3)Regen (L)	-284530	89931.5	-3.16385	0.001579	-460891	-108168	-142265	44965.74	-230446	-54084
(4)FCalc (L)	-710596	110143.1	-6.45157	0.000000	-926594	-494598	-355298	55071.56	-463297	-247299
FCalc (Q)	460462	95386.7	4.82731	0.000001	273402	647521	230231	47693.37	136701	323761
(5)Repos (L)	-279786	110143.1	-2.54020	0.011149	-495784	-63788	-139893	55071.56	-247892	-31894
Repos (Q)	22728	95386.7	0.23827	0.811692	-164332	209788	11364	47693.37	-82166	104894
(6)PopSize (L)	183842	110143.1	1.66912	0.095239	-32156	399840	91921	55071.56	-16078	199920
PopSize (Q)	134365	95386.7	1.40863	0.159090	-52695	321424	67182	47693.37	-26348	160712

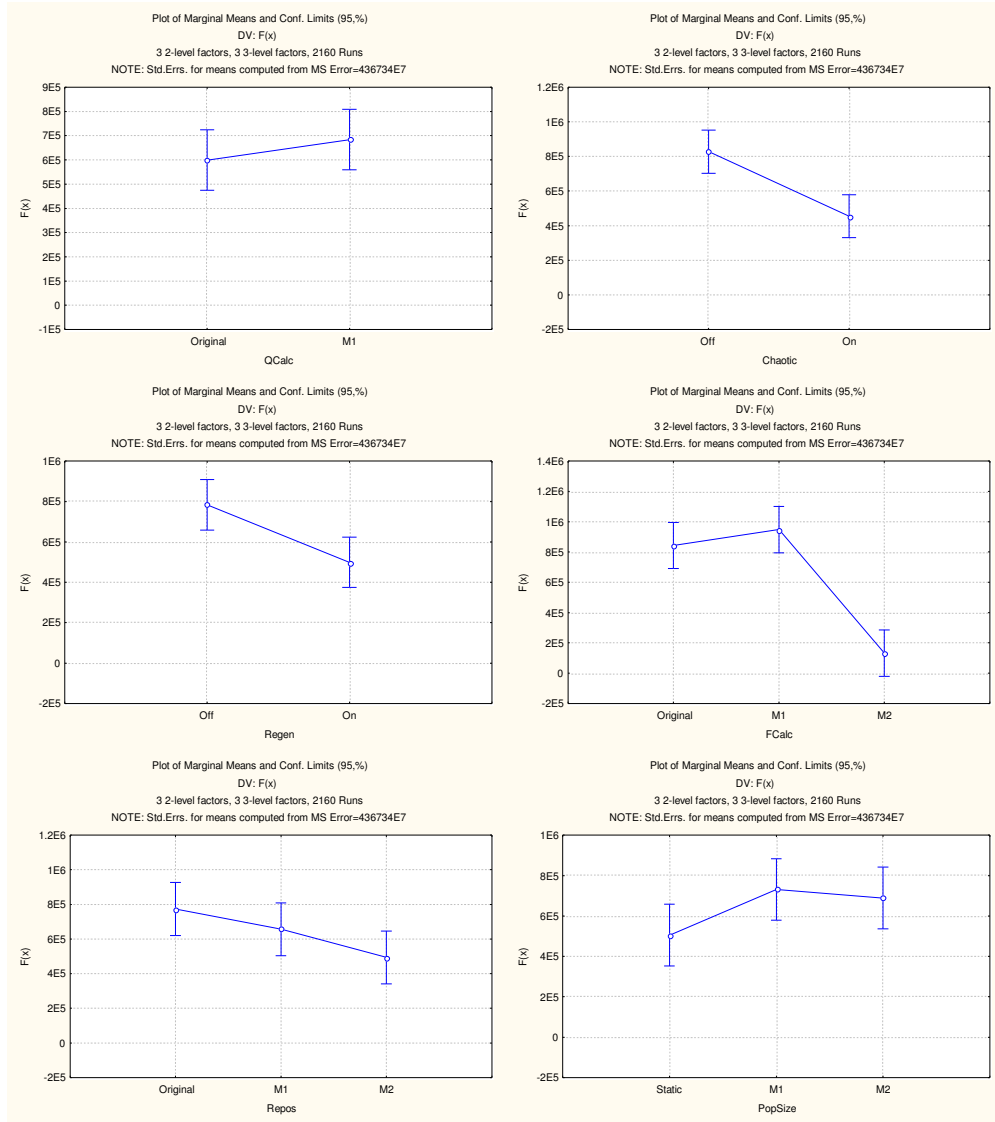
Şekil Ek 5.73 F15.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,04626; Adj:,04227 (F15.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=436734E7					
Factor	SS	df	MS	F	p
(1)QCalc L	3.859704E+12	1	3.859704E+12	0.88376	0.347278
(2)Chaotic L	7.502454E+13	1	7.502454E+13	17.17853	0.000035
(3)Regen L	4.371685E+13	1	4.371685E+13	10.00994	0.001579
(4)FCalc L+Q	2.835526E+14	2	1.417763E+14	32.46283	0.000000
(5)Repos L+Q	2.842884E+13	2	1.421442E+13	3.25471	0.038782
(6)PopSize L+Q	2.083309E+13	2	1.041655E+13	2.38510	0.092323
Error	9.389786E+15	2150	4.367342E+12		
Total SS	9.845201E+15	2159			

Şekil Ek 5.74 F15.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.75 F15.30 problemi için ortalama etkiler



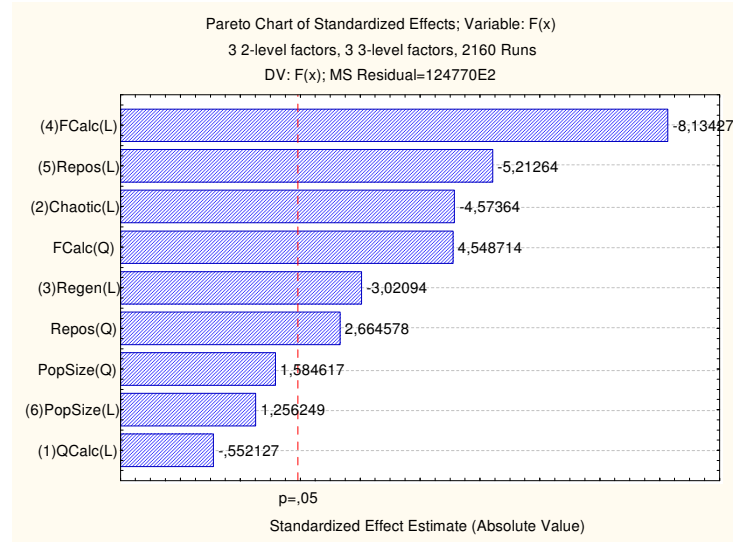
Şekil Ek 5.76 F15.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,06747; Adj:,06357 (F15.10.sta)										
3 2-level factors, 3 3-level factors, 2160 Runs										
DV: F(x); MS Residual=124770E2										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Inter.	1298.10	76.0026	17.07968	0.000000	1149.05	1447.15	1298.101	76.00265	1149.054	1447.147
(1)QCalc (L)	-83.93	152.0053	-0.55213	0.580919	-382.02	214.17	-41.963	76.00265	-191.009	107.083
(2)Chaotic (L)	-695.22	152.0053	-4.57364	0.000005	-993.31	-397.13	-347.609	76.00265	-496.655	-198.563
(3)Regen (L)	-459.20	152.0053	-3.02094	0.002550	-757.29	-161.11	-229.599	76.00265	-378.646	-80.553
(4)FCalc (L)	-1514.34	186.1677	-8.13427	0.000000	-1879.43	-1149.25	-757.169	93.08385	-939.713	-574.625
FCalc (Q)	733.37	161.2260	4.54871	0.000006	417.20	1049.55	366.685	80.61298	208.598	524.773
(5)Repos (L)	-970.43	186.1677	-5.21264	0.000000	-1335.51	-605.34	-485.213	93.08385	-667.757	-302.669
Repos (Q)	429.60	161.2260	2.66458	0.007766	113.42	745.77	214.800	80.61298	56.712	372.887
(6)PopSize (L)	233.87	186.1677	1.25625	0.209162	-131.21	598.96	116.937	93.08385	-65.607	299.480
PopSize (Q)	255.48	161.2260	1.58462	0.113200	-60.69	571.66	127.741	80.61298	-30.347	285.828

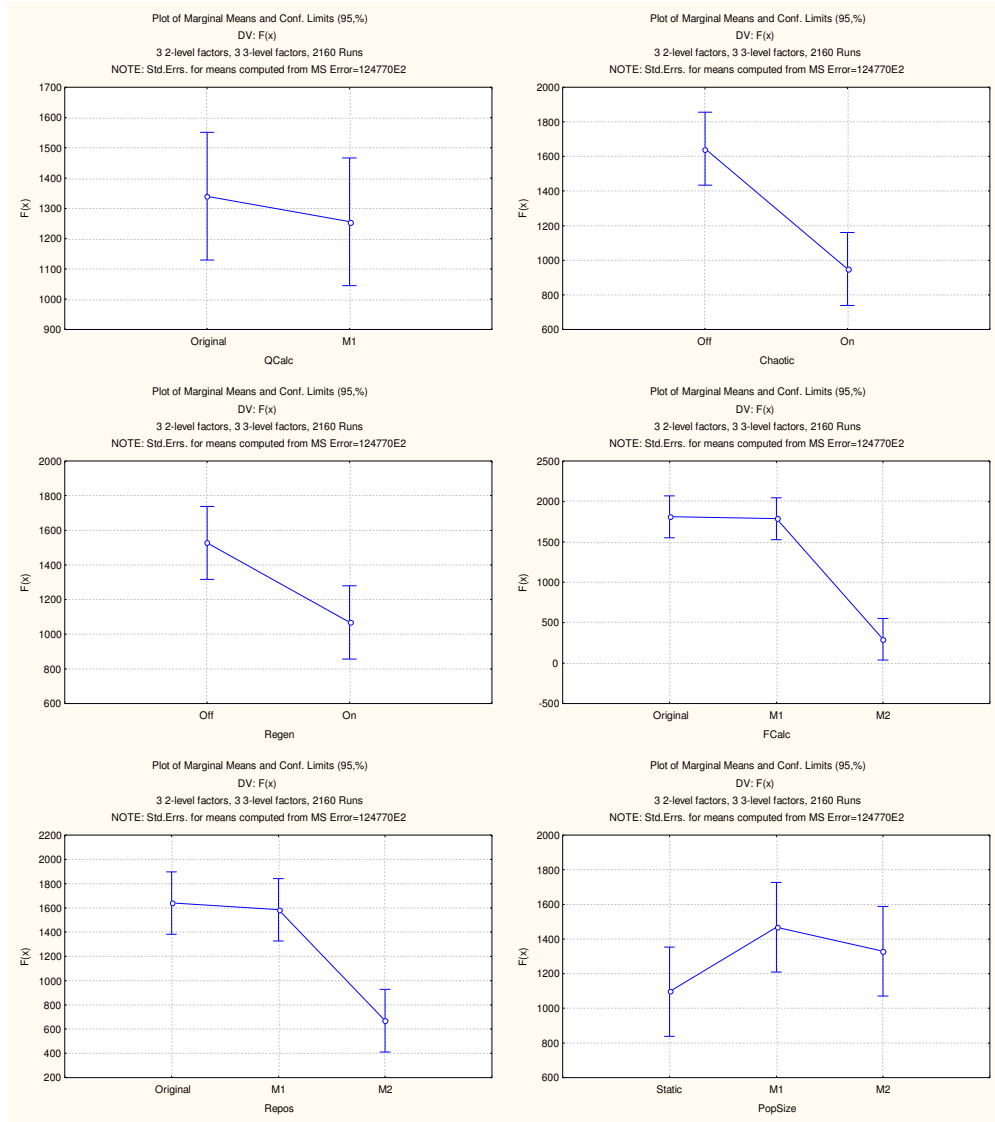
Şekil Ek 5.77 F15.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,06747; Adj:,06357 (F15.10.sta)						
3 2-level factors, 3 3-level factors, 2160 Runs						
DV: F(x); MS Residual=124770E2						
Factor	SS	df	MS	F	p	
(1)QCalc L	3.803545E+06	1	3803545	0.30484	0.580919	
(2)Chaotic L	2.609971E+08	1	260997134	20.91821	0.000005	
(3)Regen L	1.138663E+08	1	113866270	9.12607	0.002550	
(4)FCalc L+Q	1.083718E+09	2	541859125	43.42854	0.000000	
(5)Repos L+Q	4.276079E+08	2	213803934	17.13580	0.000000	
(6)PopSize L+Q	5.102074E+07	2	25510369	2.04459	0.129685	
Error	2.682561E+10	2150	12477030			
Total SS	2.876663E+10	2159				

Şekil Ek 5.78 F15.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.79 F15.10 problemi için ortalama etkiler



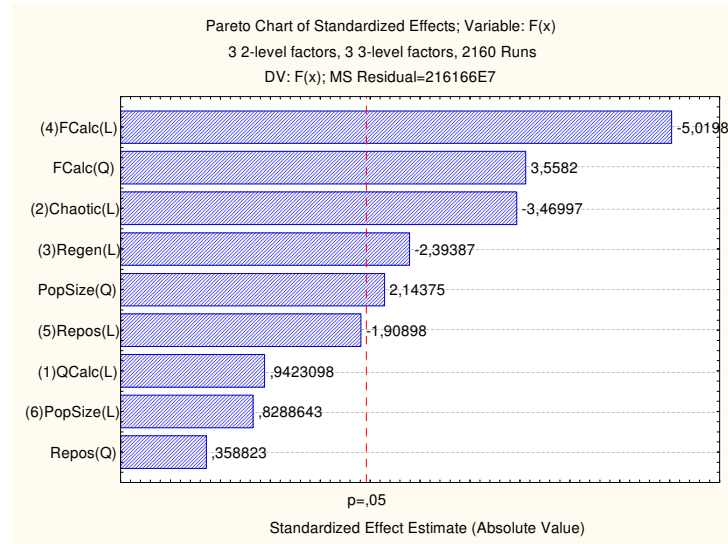
Şekil Ek 5.80 F15.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,0296; Adj:.,02553 (F15.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=216166E7										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	349581	31634.86	11.05049	0.000000	287542	411619	349581	31634.86	287542	411619
(1)QCalc (L)	59620	63269.73	0.94231	0.346140	-64457	183696	29810	31634.86	-32228	91848
(2)Chaotic (L)	-219544	63269.73	-3.46997	0.000531	-343621	-95468	-109772	31634.86	-171810	-47734
(3)Regen (L)	-151459	63269.73	-2.39387	0.016757	-275536	-27383	-75730	31634.86	-137768	-13692
(4)FCalc (L)	-388983	77489.27	-5.01983	0.000001	-540945	-237021	-194492	38744.64	-270472	-118511
FCalc (Q)	238783	67107.68	3.55820	0.000381	107180	370385	119391	33553.84	53590	185193
(5)Repos (L)	-147926	77489.27	-1.90898	0.056397	-299887	4036	-73963	38744.64	-149944	2018
Repos (Q)	24080	67107.68	0.35882	0.719763	-107523	155683	12040	33553.84	-53761	77841
(6)PopSize (L)	64228	77489.27	0.82886	0.407273	-87734	216190	32114	38744.64	-43867	108095
PopSize (Q)	143862	67107.68	2.14375	0.032165	12259	275465	71931	33553.84	6130	137732

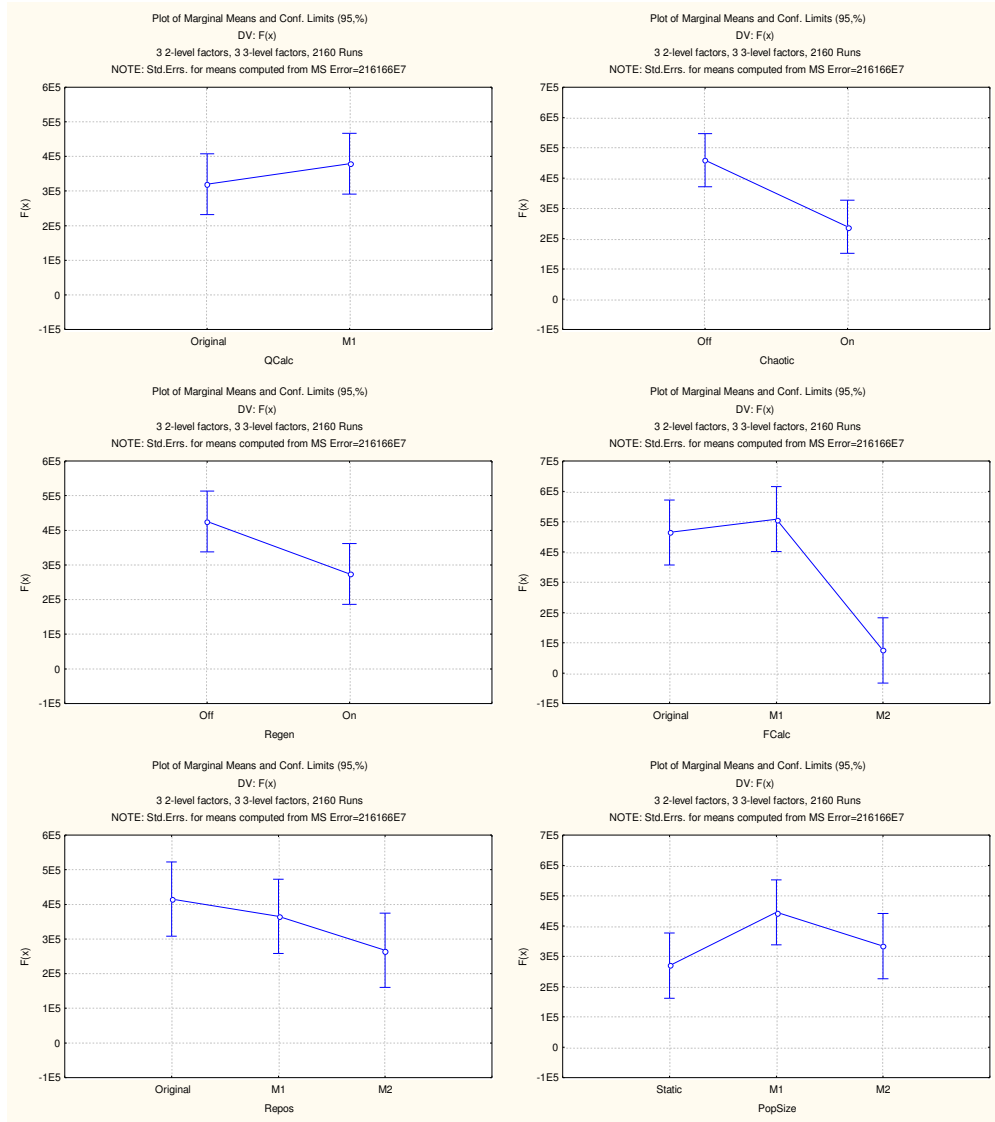
Şekil Ek 5.81 F15.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,0296; Adj:.,02553 (F15.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=216166E7						
Factor	SS	df	MS	F	p	
(1)QCalc L	1.919434E+12	1	1.919434E+12	0.88795	0.346140	
(2)Chaotic L	2.602784E+13	1	2.602784E+13	12.04072	0.000531	
(3)Regen L	1.238757E+13	1	1.238757E+13	5.73060	0.016757	
(4)FCalc L+Q	8.183904E+13	2	4.091952E+13	18.92975	0.000000	
(5)Repos L+Q	8.155848E+12	2	4.077924E+12	1.88649	0.151855	
(6)PopSize L+Q	1.141932E+13	2	5.709658E+12	2.64134	0.071497	
Error	4.647551E+15	2150	2.161651E+12			
Total SS	4.789300E+15	2159				

Şekil Ek 5.82 F15.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.83 F15.50 problemi için ortalama etkiler



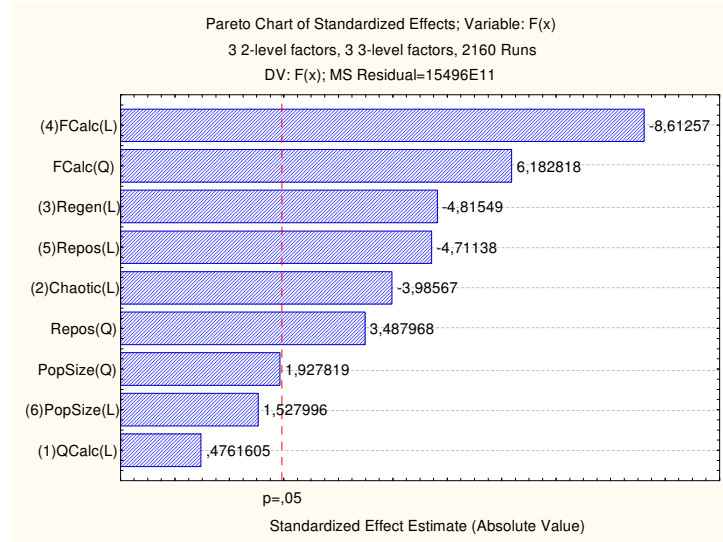
Şekil Ek 5.84 F15.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,08203; Adj:,.07819 (F20.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=15496E11										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Inter.	16115830	847006	19.02683	0.000000	14454794	17776865	16115830	847006	14454794	17776865
(1)QCalc (L)	806621	1694011	0.47616	0.634008	-2515450	4128693	403311	847006	-1257725	2064346
(2)Chaotic (L)	-6751772	1694011	-3.98567	0.000070	-10073843	-3429700	-3375886	847006	-5036922	-1714850
(3)Regen (L)	-8157501	1694011	-4.81549	0.000002	-11479572	-4835429	-4078750	847006	-5739786	-2417714
(4)FCalc (L)	-17868781	2074732	-8.61257	0.000000	-21937471	-13800090	-8934390	1037366	-10968735	-6900045
FCalc (Q)	11109104	1796770	6.18282	0.000000	7585515	14632693	5554552	898385	3792758	7316347
(5)Repos (L)	-9774840	2074732	-4.71138	0.000003	-13843531	-5706150	-4887420	1037366	-6921765	-2853075
Repos (Q)	6267077	1796770	3.48797	0.000496	2743488	9790666	3133539	898385	1371744	4895333
(6)PopSize (L)	3170181	2074732	1.52800	0.126661	-898509	7238871	1585091	1037366	-449255	3619436
PopSize (Q)	3463848	1796770	1.92782	0.054009	-59741	6987437	1731924	898385	-29870	3493719

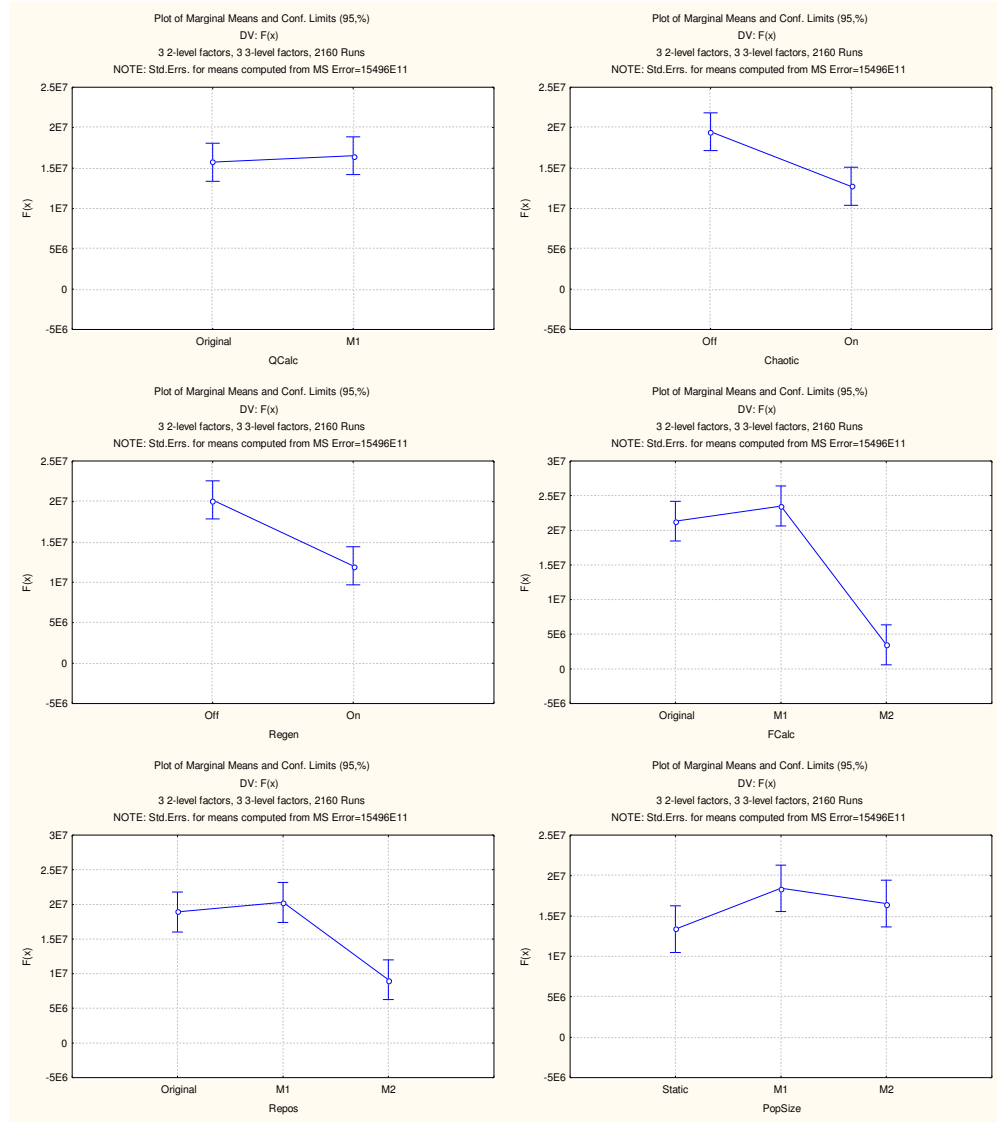
Şekil Ek 5.85 F20.30 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,08203; Adj:,.07819 (F20.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=15496E11						
Factor	SS	df	MS	F	p	
(1)QCalc L	3.513444E+14	1	3.513444E+14	0.22673	0.634008	
(2)Chaotic L	2.461667E+16	1	2.461667E+16	15.88557	0.000070	
(3)Regen L	3.593420E+16	1	3.593420E+16	23.18898	0.000002	
(4)FCalc L+Q	1.741834E+17	2	8.709172E+16	56.20184	0.000000	
(5)Repos L+Q	5.324970E+16	2	2.662485E+16	17.18149	0.000000	
(6)PopSize L+Q	9.377174E+15	2	4.688587E+15	3.02563	0.048734	
Error	3.331692E+18	2150	1.549624E+15			
Total SS	3.629404E+18	2159				

Şekil Ek 5.86 F20.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.87 F20.30 problemi için ortalama etkiler



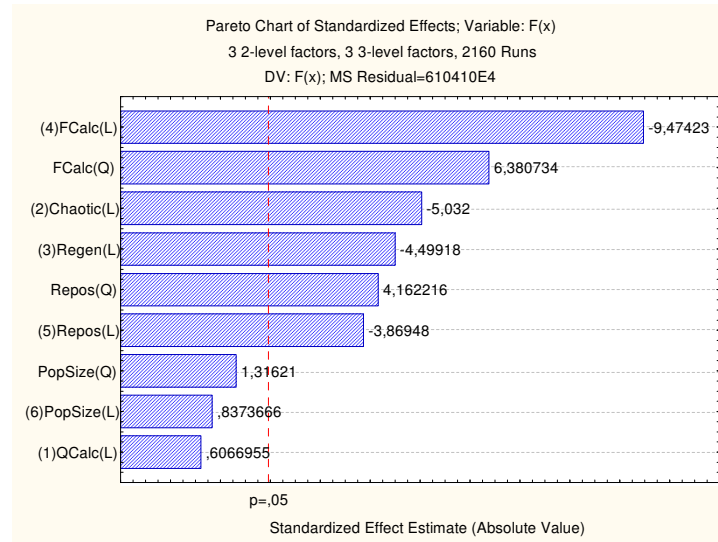
Şekil Ek 5.88 F20.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,08942; Adj:,08561 (F20.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=610410E4										
Factor	Effect	Std.Err.	t(2150)	p	-95.% Cnf.Limt	+95.% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95.% Cnf.Limt	+95.% Cnf.Limt
Mean/Interc.	34467.5	1681.062	20.50342	0.000000	31170.8	37764.2	34467.5	1681.062	31170.8	37764.2
(1)QCalc (L)	2039.8	3362.125	0.60670	0.544117	-4553.6	8633.1	1019.9	1681.062	-2276.8	4316.6
(2)Chaotic (L)	-16918.2	3362.125	-5.03200	0.000001	-23511.6	-10324.8	-8459.1	1681.062	-11755.8	-5162.4
(3)Regen (L)	-15126.8	3362.125	-4.49918	0.000007	-21720.2	-8533.5	-7563.4	1681.062	-10860.1	-4266.7
(4)FCalc (L)	-39012.5	4117.745	-9.47423	0.000000	-47087.6	-30937.3	-19506.2	2058.873	-23543.8	-15468.6
FCalc (Q)	22754.2	3566.072	6.38073	0.000000	15760.8	29747.5	11377.1	1783.036	7880.4	14873.7
(5)Repos (L)	-15933.5	4117.745	-3.86948	0.000112	-24008.7	-7858.4	-7966.8	2058.873	-12004.4	-3929.2
Repos (Q)	14842.8	3566.072	4.16222	0.000033	7849.5	21836.1	7421.4	1783.036	3924.7	10918.0
(6)PopSize (L)	3448.1	4117.745	0.83737	0.402480	-4627.1	11523.2	1724.0	2058.873	-2313.6	5761.6
PopSize (Q)	4693.7	3566.072	1.31621	0.188244	-2299.6	11687.0	2346.9	1783.036	-1149.8	5843.5

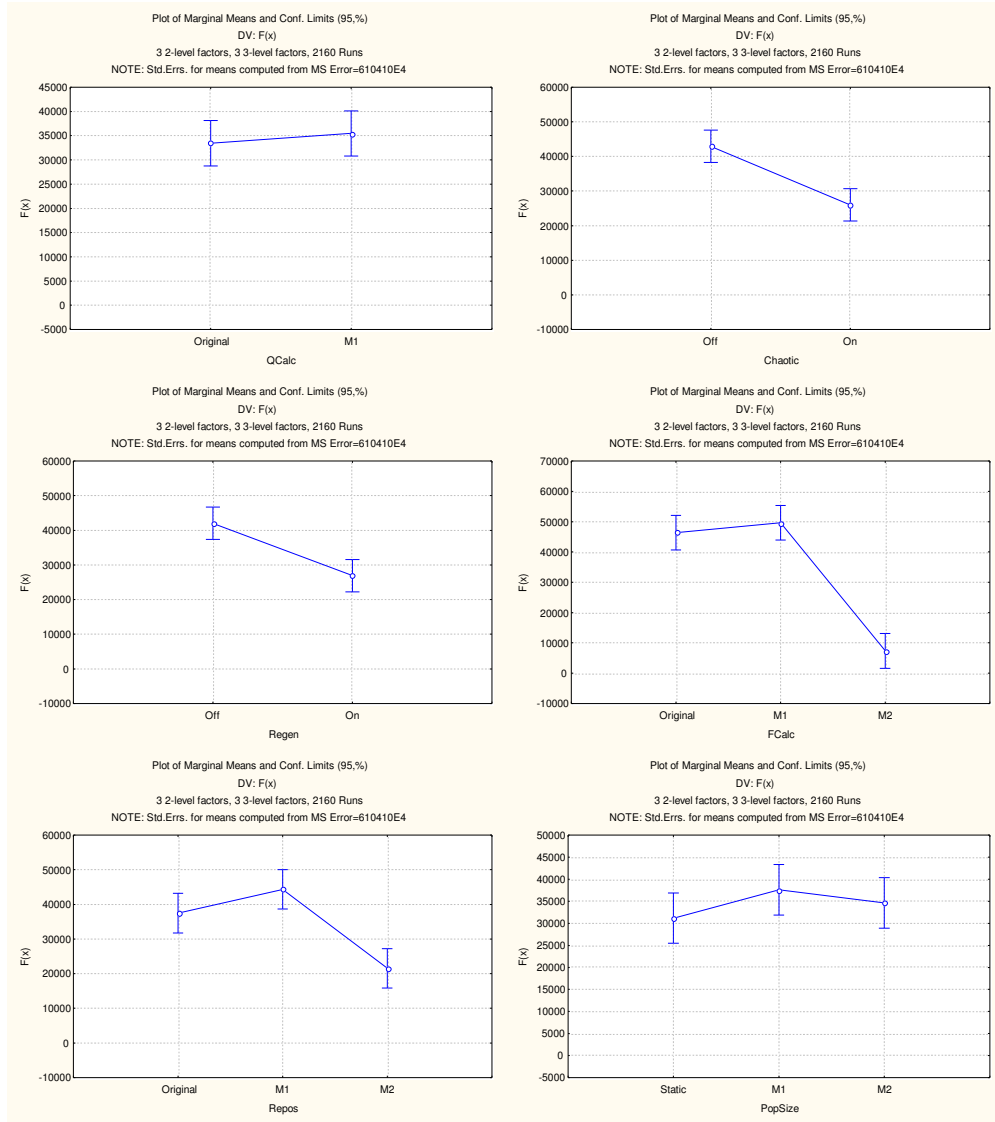
Şekil Ek 5.89 F20.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,08942; Adj:,08561 (F20.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=610410E4					
Factor	SS	df	MS	F	p
(1)QCalc L	2.246793E+09	1	2.246793E+09	0.36808	0.544117
(2)Chaotic L	1.545619E+11	1	1.545619E+11	25.32100	0.000001
(3)Regen L	1.235630E+11	1	1.235630E+11	20.24264	0.000007
(4)FCalc L+Q	7.964308E+11	2	3.982154E+11	65.23739	0.000000
(5)Repos L+Q	1.971437E+11	2	9.857185E+10	16.14847	0.000000
(6)PopSize L+Q	1.485488E+10	2	7.427441E+09	1.21680	0.296381
Error	1.312381E+13	2150	6.104097E+09		
Total SS	1.441261E+13	2159			

Şekil Ek 5.90 F20.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.91 F20.10 problemi için ortalama etkiler



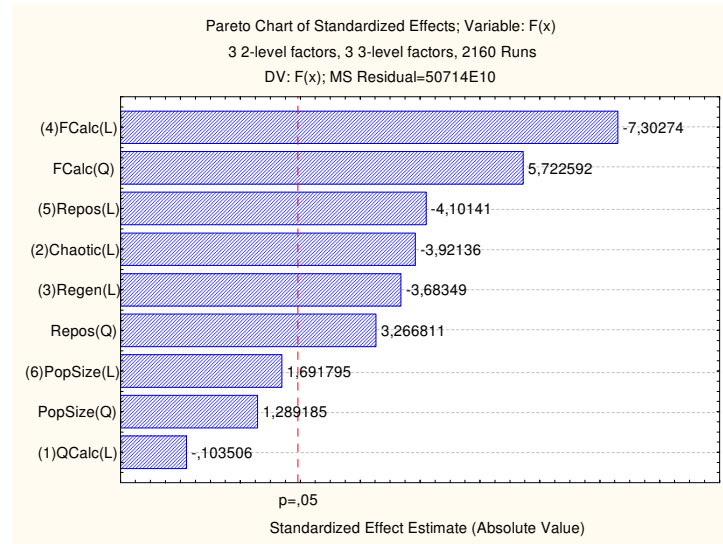
Şekil Ek 5.92 F20.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,06402; Adj:,0601 (F20.50.sta)										
3 2-level factors, 3 3-level factors, 2160 Runs										
DV: F(x); MS Residual=50714E10										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	8192055	484548	16.90659	0.000000	7241823	9142287	8192055	484548.0	7241823	9142287
(1)QCalc (L)	-100307	969096	-0.10351	0.917571	-2000770	1800156	-50153	484548.0	-1000385	900078
(2)Chaotic (L)	-3800174	969096	-3.92136	0.000091	-5700637	-1899711	-1900087	484548.0	-2850315	-949855
(3)Regen (L)	-3569652	969096	-3.68349	0.000236	-5470116	-1669189	-1784826	484548.0	-2735055	-834595
(4)FCalc (L)	-8667584	1186895	-7.30274	0.000000	-10995167	-6340002	-4333792	593447.6	-5497584	-3170001
FCalc (Q)	5882146	1027881	5.72259	0.000000	3866400	7897892	2941073	513940.7	1933200	3948946
(5)Repos (L)	-4867948	1186895	-4.10141	0.000043	-7195531	-2540366	-2433974	593447.6	-3597765	-1270183
Repos (Q)	3357894	1027881	3.26681	0.001105	1342149	5373640	1678947	513940.7	671074	2686820
(6)PopSize (L)	2007984	1186895	1.69180	0.090830	-319599	4335566	1003992	593447.6	-159799	2167783
PopSize (Q)	1325129	1027881	1.28918	0.197473	-690617	3340875	662565	513940.7	-345308	1670437

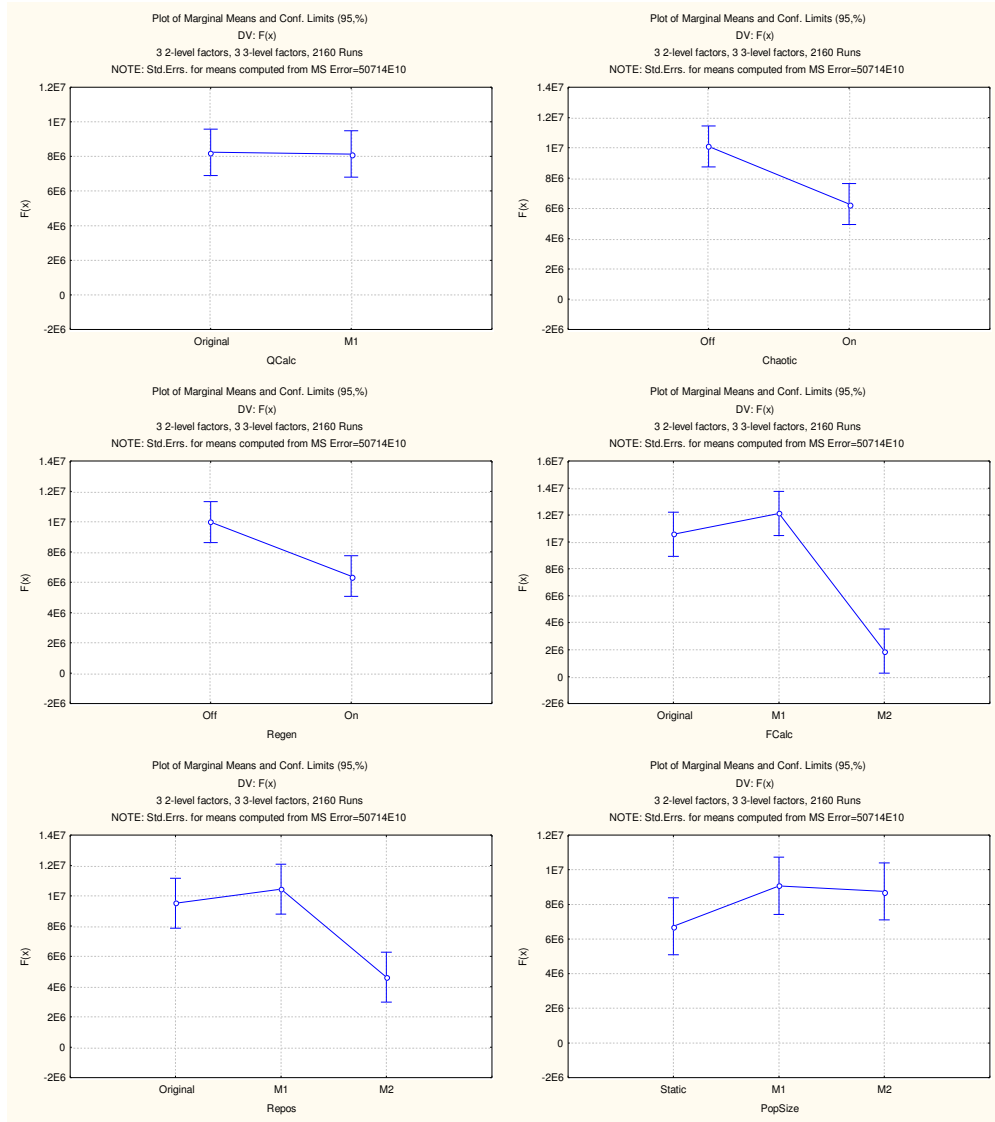
Şekil Ek 5.93 F20.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,06402; Adj:,0601 (F20.50.sta)						
3 2-level factors, 3 3-level factors, 2160 Runs						
DV: F(x); MS Residual=50714E10						
Factor	SS	df	MS	F	p	
(1)QCalc L	5.433193E+12	1	5.433193E+12	0.01071	0.917571	
(2)Chaotic L	7.798314E+15	1	7.798314E+15	15.37706	0.000091	
(3)Regen L	6.880906E+15	1	6.880906E+15	13.56808	0.000236	
(4)FCalc L+Q	4.365355E+16	2	2.182678E+16	43.03901	0.000000	
(5)Repos L+Q	1.394311E+16	2	6.971555E+15	13.74682	0.000001	
(6)PopSize L+Q	2.294384E+15	2	1.147192E+15	2.26208	0.104381	
Error	1.090350E+18	2150	5.071394E+14			
Total SS	1.164925E+18	2159				

Şekil Ek 5.94 F20.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.95 F20.50 problemi için ortalama etkiler



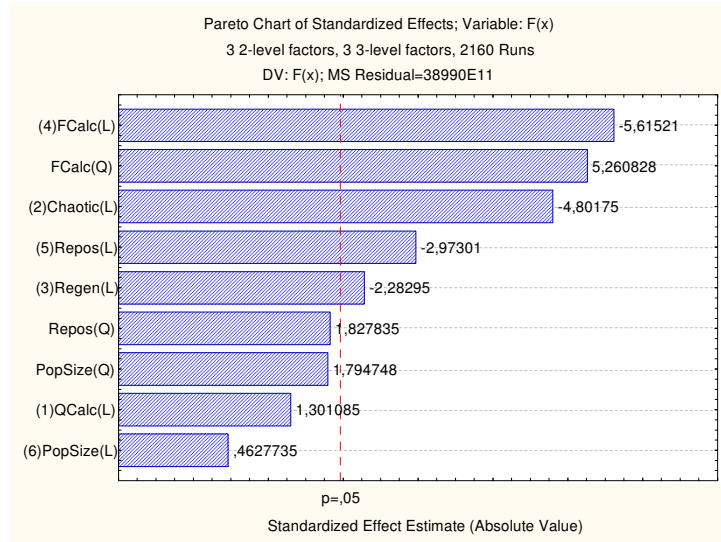
Şekil Ek 5.96 F20.50 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,04647; Adj:,04248 (F25.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=38990E11										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	18944894	1343543	14.10069	0.000000	16310114	21579675	18944894	1343543	16310114	21579675
(1)QCalc (L)	3496130	2687087	1.30109	0.193369	-1773431	8765690	1748065	1343543	-886715	4382845
(2)Chaotic (L)	-12902735	2687087	-4.80175	0.000002	-18172294	-7633172	-6451366	1343543	-9086147	-3816586
(3)Regen (L)	-6134485	2687087	-2.28295	0.022531	-11404046	-864925	-3067243	1343543	-5702023	-432462
(4)FCalc (L)	-18479630	3290996	-5.61521	0.000000	-24933497	-12025762	-9239815	1645498	-12466749	-6012881
FCalc (Q)	14993812	2850086	5.26083	0.000000	9404599	20583025	7496906	1425043	4702299	10291512
(5)Repos (L)	-9784169	3290996	-2.97301	0.002982	-16238037	-3330302	-4892085	1645498	-8119018	-1665151
Repos (Q)	5209488	2850086	1.82784	0.067713	-379725	10798701	2604744	1425043	-189863	5399350
(6)PopSize (L)	1522986	3290996	0.46277	0.643574	-4930882	7976853	761493	1645498	-2465441	3988427
PopSize (Q)	5115185	2850086	1.79475	0.072834	-474028	10704396	2557593	1425043	-237014	5352199

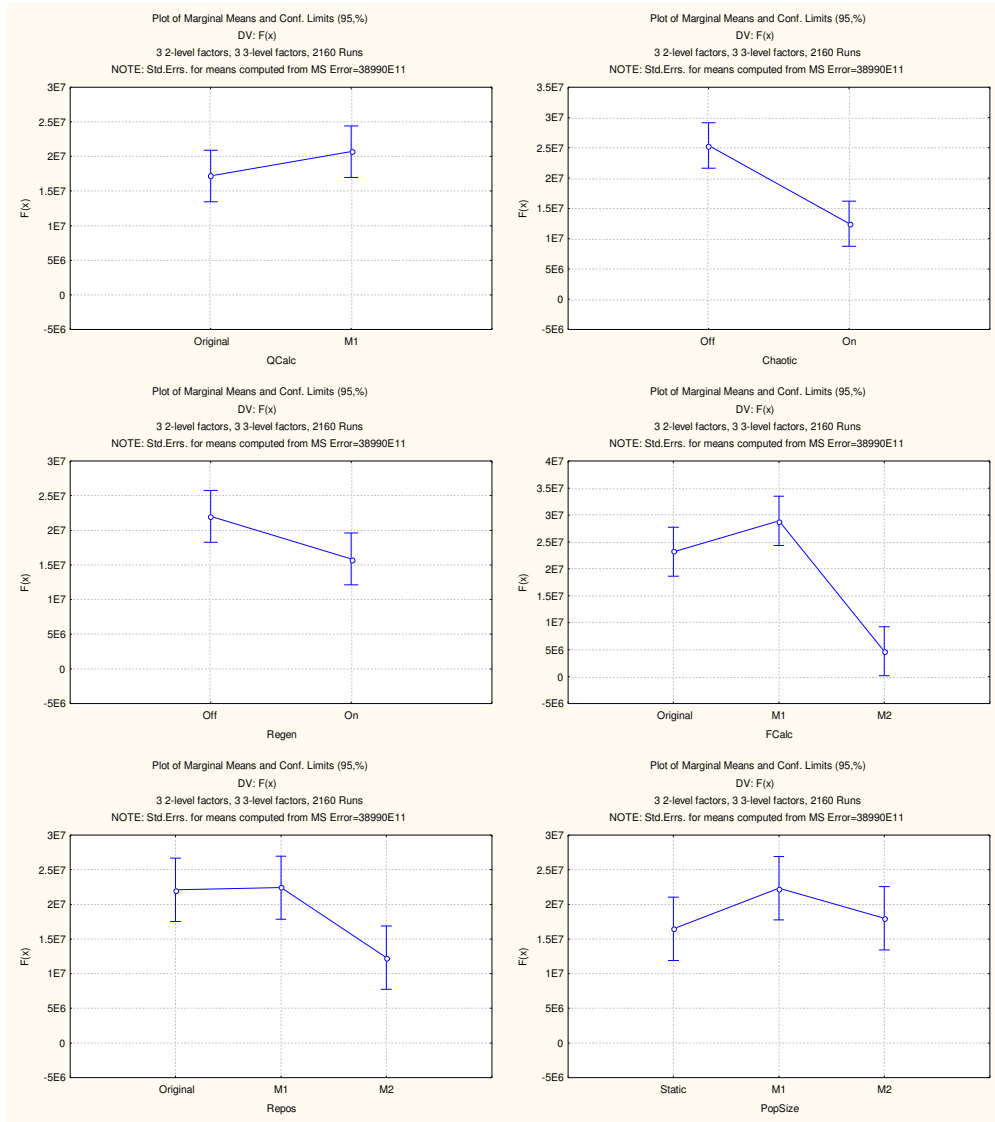
Şekil Ek 5.97 F25.30 problemi için tahmini etki problemleri

ANOVA; Var.:F(x); R-sqr=,04647; Adj:,04248 (F25.30.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=38990E11						
Factor	SS	df	MS	F	p	
(1)QCalc L	6.600378E+15	1	6.600378E+15	1.69282	0.193369	
(2)Chaotic L	8.989948E+16	1	8.989948E+16	23.05685	0.000002	
(3)Regen L	2.032123E+16	1	2.032123E+16	5.21186	0.022531	
(4)FCalc L+Q	2.308497E+17	2	1.154249E+17	29.60344	0.000000	
(5)Repos L+Q	4.748939E+16	2	2.374470E+16	6.08989	0.002305	
(6)PopSize L+Q	1.339427E+16	2	6.697136E+15	1.71764	0.179736	
Error	8.382927E+18	2150	3.899036E+15			
Total SS	8.791481E+18	2159				

Şekil Ek 5.98 F25.30 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.99 F25.30 problemi için ortalama etkiler



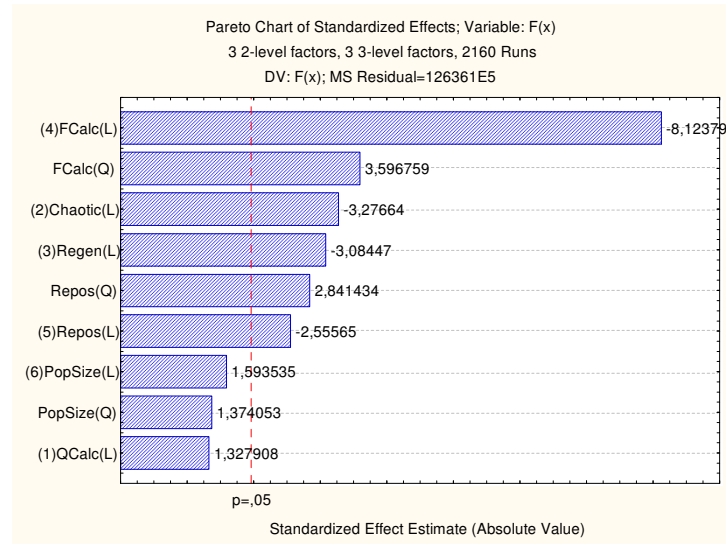
Şekil Ek 5.100 F25.30 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,05285; Adj:,04889 (F25.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=126361E5										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	37046.3	2418.694	15.31666	0.000000	32303.1	41789.6	37046.3	2418.694	32303.1	41789.6
(1)QCalc (L)	6423.6	4837.389	1.32791	0.184349	-3062.8	15910.1	3211.8	2418.694	-1531.4	7955.0
(2)Chaotic (L)	-15850.4	4837.389	-3.27664	0.001067	-25336.8	-6363.9	-7925.2	2418.694	-12668.4	-3182.0
(3)Regen (L)	-14920.8	4837.389	-3.08447	0.002065	-24407.2	-5434.3	-7460.4	2418.694	-12203.6	-2717.2
(4)FCalc (L)	-48130.0	5924.567	-8.12379	0.000000	-59748.4	-36511.5	-24065.0	2962.284	-29874.2	-18255.7
FCalc (Q)	18454.3	5130.826	3.59676	0.000329	8392.4	28516.2	9227.2	2565.413	4196.2	14258.1
(5)Repos (L)	-15141.1	5924.567	-2.55565	0.010667	-26759.6	-3522.6	-7570.6	2962.284	-13379.8	-1761.3
Repos (Q)	14578.9	5130.826	2.84143	0.004533	4517.0	24640.8	7289.5	2565.413	2258.5	12320.4
(6)PopSize (L)	9441.0	5924.567	1.59353	0.111187	-2177.5	21059.5	4720.5	2962.284	-1088.7	10529.7
PopSize (Q)	7050.0	5130.826	1.37405	0.169568	-3011.9	17111.9	3525.0	2565.413	-1505.9	8556.0

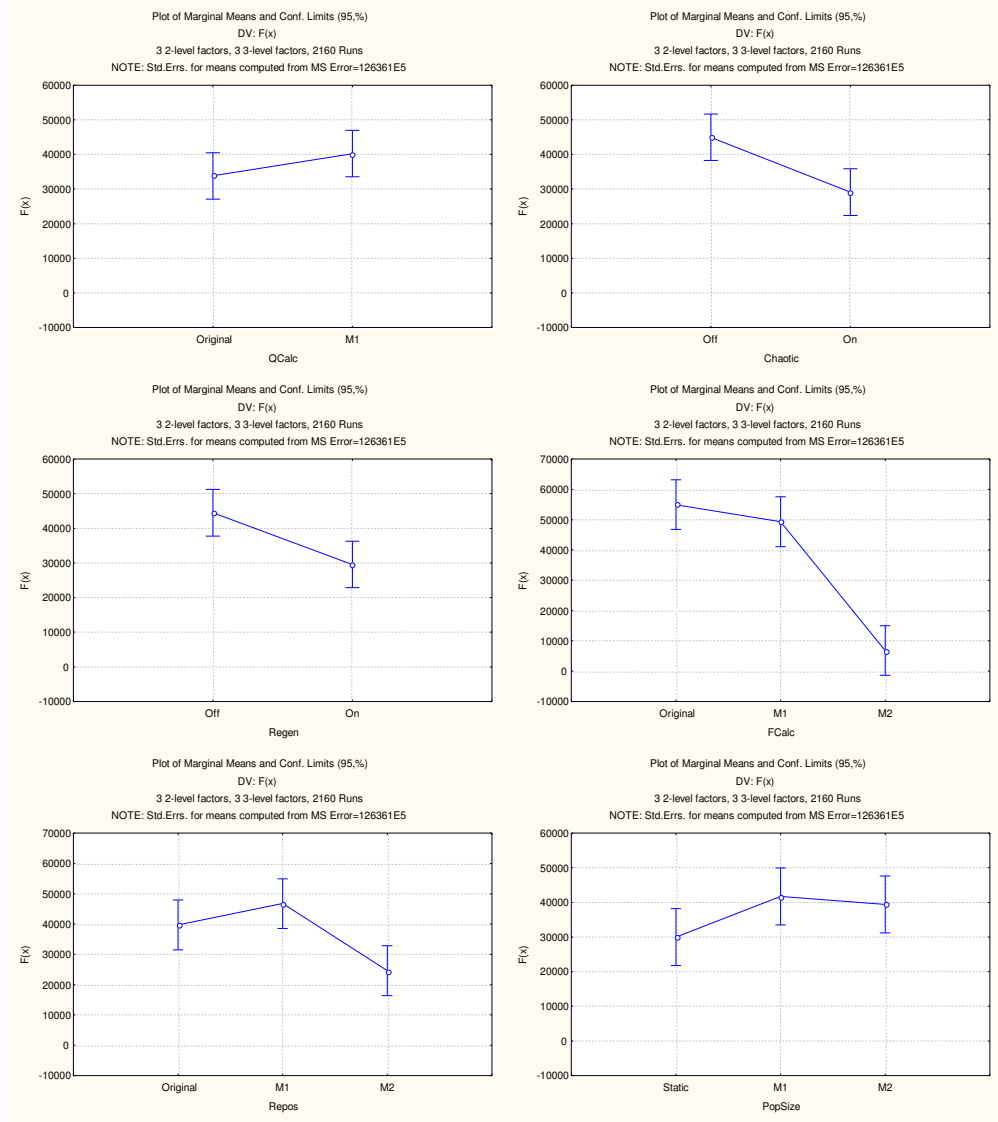
Şekil Ek 5.101 F25.10 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,05285; Adj:,04889 (F25.10.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=126361E5					
Factor	SS	df	MS	F	p
(1)QCalc L	2.228188E+10	1	2.228188E+10	1.76334	0.184349
(2)Chaotic L	1.356670E+11	1	1.356670E+11	10.73639	0.001067
(3)Regen L	1.202203E+11	1	1.202203E+11	9.51397	0.002065
(4)FCalc L+Q	9.974074E+11	2	4.987037E+11	39.46634	0.000000
(5)Repos L+Q	1.845527E+11	2	9.227635E+10	7.30255	0.000691
(6)PopSize L+Q	5.594509E+10	2	2.797254E+10	2.21369	0.109546
Error	2.716778E+13	2150	1.263618E+10		
Total SS	2.868386E+13	2159			

Şekil Ek 5.102 F25.10 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.103 F25.10 problemi için ortalama etkiler



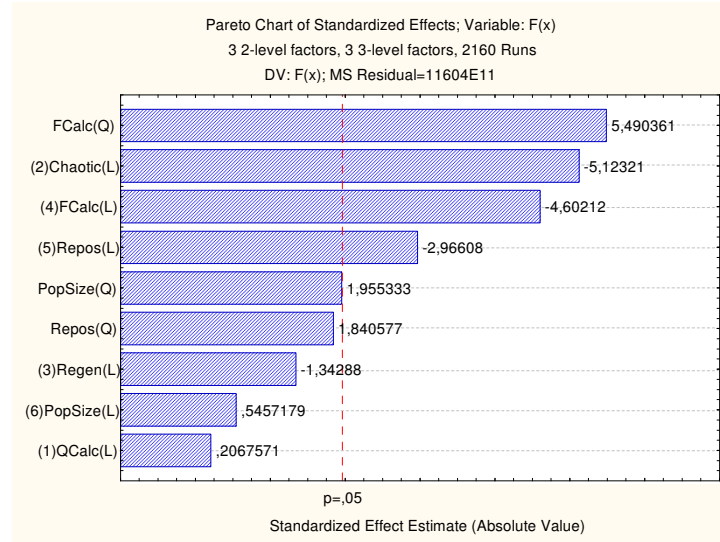
Şekil Ek 5.104 F25.10 problemi için ortalama etki grafikleri

Effect Estimates; Var.:F(x); R-sqr=,04262; Adj:,03862 (F25.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=11604E11										
Factor	Effect	Std.Err.	t(2150)	p	-95,% Cnf.Limt	+95,% Cnf.Limt	Coeff.	Std.Err. Coeff.	-95,% Cnf.Limt	+95,% Cnf.Limt
Mean/Interc.	9534101	732970	13.00749	0.000000	8096697	10971505	9534101	732969.9	8096697	10971505
(1)QCalc (L)	303094	1465940	0.20676	0.836219	-2571714	3177901	151547	732969.9	-1285857	1588951
(2)Chaotic (L)	-7510314	1465940	-5.12321	0.000000	-10385122	-4635506	-3755157	732969.9	-5192561	-2317753
(3)Regen (L)	-1968588	1465940	-1.34288	0.179451	-4843396	906220	-984294	732969.9	-242169	453110
(4)FCalc (L)	-8262652	1795402	-4.60212	0.000004	-11783558	-4741746	-4131326	897701.2	-5891778	-2370873
FCalc (Q)	8536765	1554864	5.49036	0.000000	5487571	11585960	4268383	777432.0	2743786	5792980
(5)Repos (L)	-5325307	1795402	-2.96608	0.003049	-8846214	-1804401	-2662654	897701.2	-4423107	-902201
Repos (Q)	2861847	1554864	1.84058	0.065821	-187347	5911041	1430923	777432.0	-93674	2955521
(6)PopSize (L)	979783	1795402	0.54572	0.585317	-2541123	4500689	489892	897701.2	-1270562	2250345
PopSize (Q)	3040276	1554864	1.95533	0.050673	-8918	6089471	1520138	777432.0	-4459	3044735

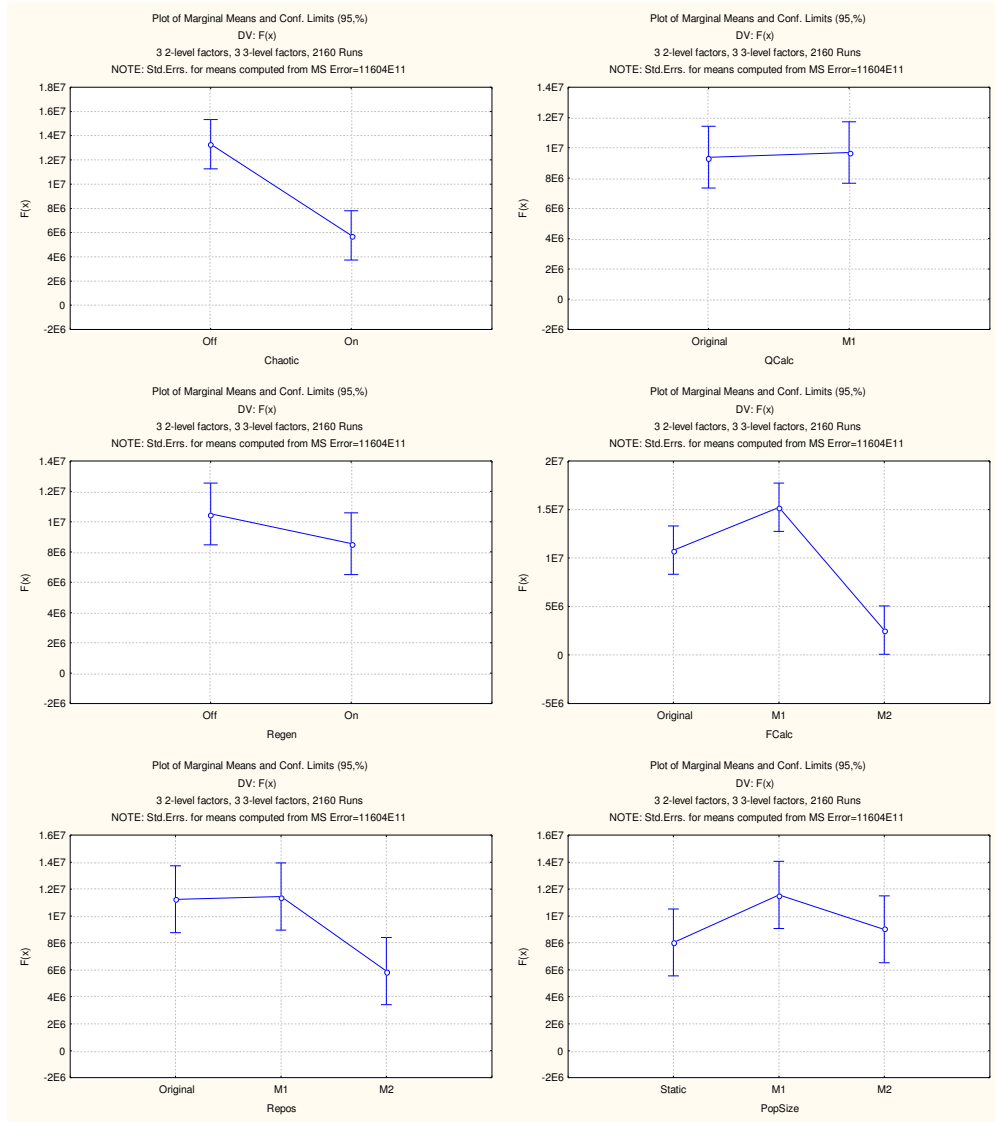
Şekil Ek 5.105 F25.50 problemi tahmini etki değerleri STATISTICA görüntüsü

ANOVA; Var.:F(x); R-sqr=,04262; Adj:,03862 (F25.50.sta) 3 2-level factors, 3 3-level factors, 2160 Runs DV: F(x); MS Residual=11604E11					
Factor	SS	df	MS	F	p
(1)QCalc L	4.960747E+13	1	4.960747E+13	0.04275	0.836219
(2)Chaotic L	3.045860E+16	1	3.045860E+16	26.24726	0.000000
(3)Regen L	2.092683E+15	1	2.092683E+15	1.80334	0.179451
(4)FCalc L+Q	5.955837E+16	2	2.977918E+16	25.66178	0.000000
(5)Repos L+Q	1.414048E+16	2	7.070242E+15	6.09268	0.002299
(6)PopSize L+Q	4.782365E+15	2	2.391183E+15	2.06057	0.127633
Error	2.494965E+18	2150	1.160449E+15		
Total SS	2.606048E+18	2159			

Şekil Ek 5.106 F25.50 problemi için ANOVA tablosu STATISTICA görüntüsü



Şekil Ek 5.107 F25.50 problemi için ortalama etkiler



Şekil Ek 5.108 F25.50 problemi için ortalama etki grafikleri

EK 6 DOĞRUSAL KISITLI OPTİMİZASYON PROBLEMLERİ

G01:

$$\text{Min } f(x) = 5 \sum_{i=1}^4 x_i - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i$$

s.t.

$$g_1(x) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0$$

$$g_2(x) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0$$

$$g_3(x) = 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0$$

$$g_4(x) = -8x_1 + x_{10} \leq 0$$

$$g_5(x) = -8x_2 + x_{11} \leq 0$$

$$g_6(x) = -8x_3 + x_{12} \leq 0$$

$$g_7(x) = -2x_4 - x_5 + x_{10} \leq 0$$

$$g_8(x) = -2x_6 - x_7 + x_{11} \leq 0$$

$$g_9(x) = -2x_8 - x_9 + x_{12} \leq 0$$

G02:

$$\text{Max } f(x) = \left| \frac{\sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i)}{\sqrt{\sum_{i=1}^n i x_i^2}} \right|$$

s.t.

$$g_1(x) = 0,75 - \prod_{i=1}^n x_i \leq 0$$

$$g_2(x) = \sum_{i=1}^n x_i - 7.5n \leq 0$$

G03:

$$\text{Max } f(x) = (\sqrt{n})^2 \prod_{i=1}^n x_i$$

s.t.

$$h(x) = \sum_{i=1}^n x_i^2 - 1 = 0$$

G04:

$$\text{Min } f(x) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141$$

s.t.

$$g_1(x) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \leq 0$$

$$g_2(x) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 - 92 \leq 0$$

$$g_3(x) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 - 0.0021813x_3^2 - 110 \leq 0$$

$$g_4(x) = -80.51249 - 0.0071317x_2x_5 + 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \leq 0$$

$$g_5(x) = 9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 - 25 \leq 0$$

$$g_6(x) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \leq 0$$

G05:

$$\text{Min } f(x) = 3x_1 + 0.0000001x_1^3 + 2x_2 + \left(\frac{0.000002}{3}\right)x_2^3$$

s.t.

$$g_1(x) = -x_4 + x_3 - 0.55 \leq 0$$

$$g_2(x) = -x_3 + x_4 - 0.55 \leq 0$$

$$h_1(x) = 1000 \sin(-x_3 - 0.25) + 1000 \sin(-x_4 - 0.25) + 894.8 - x_1 = 0$$

$$h_2(x) = 1000 \sin(x_3 - 0.25) + 1000 \sin(x_3 - x_4 - 0.25) + 894.8 - x_2 = 0$$

$$h_3(x) = 1000 \sin(x_4 - 0.25) + 1000 \sin(x_4 - x_3 - 0.25) + 1294.8 = 0$$

G06:

$$\text{Min } f(x) = (x_1 - 10)^3 + (x_2 - 20)^3$$

s.t.

$$g_1(x) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0$$

$$g_2(x) = (x_1 - 6)^2 - (x_2 - 5)^2 - 82.81 \leq 0$$

G07:

$$\text{Min } f(x) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45$$

s.t.

$$g_1(x) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0$$

$$g_2(x) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0$$

$$g_3(x) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0$$

$$g_4(x) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0$$

$$g_5(x) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0$$

$$g_6(x) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0$$

$$g_7(x) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \leq 0$$

$$g_8(x) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0$$

G08:

$$\text{Max } f(x) = \frac{\sin^3(2\pi x_1) \sin(2\pi x_2)}{x_1^3(x_1+x_2)}$$

s.t.

$$g_1(x) = x_1^2 - x_2 + 1 \leq 0$$

$$g_2(x) = 1 - x_1 + (x_2 - 4)^2 \leq 0$$

G09:

$$\text{Min } f(x) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 + 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7$$

s.t.

$$g_1(x) = -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0$$

$$g_2(x) = -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0$$

$$g_3(x) = -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0$$

$$g_4(x) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0$$

G10:

$$\text{Min } f(x) = x_1 + x_2 + x_3$$

s.t.

$$g_1(x) = -1 + 0.0025(x_4 + x_6) \leq 0$$

$$g_2(x) = -1 + 0.0025(x_5 + x_7 - x_4) \leq 0$$

$$g_3(x) = -1 + 0.01(x_8 - x_5) \leq 0$$

$$g_4(x) = -x_1x_6 + 883.33252x_4 + 100x_1 - 83.333333 \leq 0$$

$$g_5(x) = -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 \leq 0$$

$$g_6(x) = -x_3x_8 + 1250000 + x_3x_5 - 2500x_5 \leq 0$$

G11:

$$\text{Min } f(x) = x_1^2 + (x_2 - 1)^2$$

s.t.

$$h(x) = x_2 - x_1^2 = 0$$

G12:

$$\text{Max } f(x) = \frac{100 - (x_1 - 5)^2 - (x_2 - 5)^2 - (x_3 - 5)^2}{100}$$

s.t.

$$g(x) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2 - 0.0625 \leq 0$$

G13:

$$\text{Min } f(x) = e^{x_1 x_2 x_3 x_4 x_5}$$

s.t.

$$h_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 = 0$$

$$h_2(x) = x_2 x_3 - 5 x_4 x_5 = 0$$

$$h_3(x) = x_1^3 + x_2^3 + 1 = 0$$

EK 7 HASTA VERİ SETİ

Çizelge Ek 7.1 Hasta veri seti

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
1	0,65	0	0	2	0	0,40	3	1	1	0	2	0	0	2	1	0	1	1	1
2	0,76	0	2	0	0	0,70	3	0	1	1	3	0	0	2	0	0	1	1	1
3	0,64	1	2	1	1	0,40	2	0	0	1	1	0	0	2	1	1	0	0	1
4	0,57	1	3	0	0	0,70	2	0	1	0	2	0	1	2	0	0	0	1	-1
5	0,51	1	0	2	0	0,70	0	1	1	0	0	0	1	2	0	0	1	1	1
6	0,65	0	3	0	1	0,00	4	1	1	0	1	0	0	2	1	1	0	0	1
7	0,72	0	1	0	1	0,70	2	0	1	1	0	1	0	1	0	1	1	0	-1
8	0,69	1	0	0	1	0,30	1	0	1	0	3	1	1	2	1	1	1	0	1
9	0,75	0	1	0	1	0,70	2	0	1	1	0	1	0	1	0	1	1	0	-1
10	0,49	1	3	2	0	0,80	2	0	0	0	1	0	1	2	0	0	1	0	1
11	0,55	0	0	2	0	0,60	4	1	1	1	2	1	0	1	1	1	0	0	-1
12	0,68	0	0	0	1	0,50	4	0	1	0	2	1	1	1	1	1	1	0	-1
13	0,70	0	3	0	0	0,50	4	0	0	1	2	1	0	1	0	1	1	0	-1
14	0,63	1	2	1	1	0,00	0	0	1	0	2	1	1	2	1	0	0	1	-1
15	0,70	0	0	2	1	0,40	4	0	0	0	1	1	0	2	0	1	1	0	1
16	0,75	1	1	1	0	0,60	1	0	1	0	2	1	1	1	1	0	1	0	1
17	0,57	0	0	0	1	0,40	3	1	1	0	3	1	0	1	0	0	1	0	1
18	0,68	1	2	0	0	0,60	0	0	0	0	1	0	1	1	1	0	0	0	-1
19	0,52	0	1	0	1	0,40	1	1	1	1	0	1	0	1	1	1	0	1	1
20	0,73	1	0	2	1	0,40	4	1	1	1	0	0	1	1	1	0	1	0	-1
21	0,72	0	1	0	1	0,70	0	0	1	0	2	0	1	1	0	0	1	0	1
22	0,58	1	1	1	0	0,30	4	1	1	1	0	1	1	2	1	0	0	0	1
23	0,58	0	3	0	1	0,40	1	1	1	0	3	0	1	1	0	0	1	1	-1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
24	0,57	0	0	0	1	0,70	0	1	1	1	2	0	1	1	0	1	0	0	1
25	0,62	1	2	0	0	0,60	0	0	1	1	1	0	0	1	1	1	1	0	1
26	0,68	0	0	1	0	0,40	0	0	1	1	3	1	1	1	0	1	0	0	-1
27	0,51	1	2	0	0	0,60	4	0	1	1	3	1	1	2	1	1	0	0	-1
28	0,65	0	2	2	0	0,60	1	0	0	1	0	0	1	1	0	0	0	1	-1
29	0,56	0	1	0	1	0,70	0	1	1	0	1	1	0	1	1	0	0	1	1
30	0,75	0	3	0	1	0,50	3	0	0	1	2	1	0	2	0	0	0	1	1
31	0,73	1	0	2	0	0,70	3	1	1	1	3	0	1	2	0	1	1	0	-1
32	0,56	0	0	2	0	0,70	0	1	1	0	2	0	1	2	1	0	1	1	1
33	0,58	1	3	2	0	0,10	3	1	1	1	3	0	1	2	1	0	1	0	1
34	0,49	0	1	1	1	0,30	1	1	1	0	1	0	1	1	0	1	0	0	-1
35	0,67	0	2	0	0	0,40	2	1	1	1	1	0	1	1	0	1	0	1	-1
36	0,61	1	3	0	1	0,60	4	1	1	1	0	1	1	1	1	0	0	1	1
37	0,67	1	2	2	0	0,50	0	1	1	1	1	0	0	1	1	0	1	0	1
38	0,59	0	3	1	0	0,60	0	1	1	1	3	1	0	2	0	0	0	0	1
39	0,70	1	1	2	0	0,50	2	1	1	1	0	1	1	2	1	1	0	1	-1
40	0,55	1	2	1	0	0,30	0	0	1	0	2	0	1	1	1	1	0	0	1
41	0,70	0	1	0	1	0,40	1	1	1	0	1	1	1	2	1	0	1	1	1
42	0,79	0	2	0	0	0,80	3	0	1	1	2	0	0	1	0	1	0	0	-1
43	0,65	1	2	2	0	0,30	4	0	0	1	0	0	1	1	0	0	0	0	1
44	0,61	0	3	0	0	0,60	4	1	1	1	0	0	0	2	0	1	1	1	1
45	0,56	0	2	0	1	0,40	2	1	1	1	3	0	1	2	0	0	1	0	-1
46	0,59	0	1	2	1	0,40	1	1	1	0	3	0	0	2	1	1	0	0	1
47	0,58	1	0	1	1	0,50	2	0	1	0	3	0	0	1	1	0	1	0	-1
48	0,61	0	2	1	1	0,30	3	1	1	1	2	0	1	2	0	1	1	1	1
49	0,81	0	3	1	1	0,60	0	1	1	0	0	1	0	2	0	1	1	0	1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
50	0,57	0	2	2	1	0,70	2	0	1	1	0	1	0	2	1	0	0	0	-1
51	0,69	1	3	1	0	0,50	3	0	1	1	3	0	0	1	0	1	0	1	-1
52	0,58	1	1	2	0	0,40	4	0	0	1	0	1	0	1	0	0	0	1	1
53	0,56	0	3	0	0	0,70	2	0	1	1	0	0	0	2	1	0	0	0	1
54	0,60	1	2	1	0	0,50	1	0	1	0	1	1	1	1	1	1	0	1	-1
55	0,84	1	3	1	0	0,40	1	1	1	0	0	0	1	1	0	1	0	1	-1
56	0,69	1	3	0	1	0,80	3	1	1	1	0	0	1	1	1	1	0	1	1
57	0,69	1	2	2	1	0,70	2	0	0	0	3	0	1	1	1	0	0	1	-1
58	0,66	0	2	1	0	0,50	0	0	1	0	0	0	1	2	0	0	0	0	1
59	0,70	1	1	0	0	0,70	2	1	1	0	3	1	0	1	1	0	1	1	1
60	0,73	1	2	1	1	0,80	2	1	1	0	1	1	1	2	0	0	0	0	1
61	0,63	1	3	1	0	0,50	3	1	1	0	1	0	0	2	1	0	0	0	1
62	0,49	0	1	1	0	0,80	4	1	1	1	3	1	1	2	0	0	1	1	1
63	0,59	1	1	0	1	0,30	1	0	0	0	1	1	0	2	0	1	0	0	1
64	0,63	0	0	0	0	0,40	1	1	1	1	0	0	0	2	1	1	0	1	-1
65	0,68	1	2	1	1	0,70	3	0	1	0	3	0	1	2	1	1	1	1	1
66	0,75	0	2	2	1	0,80	4	1	1	0	1	1	1	1	1	0	0	0	-1
67	0,44	1	0	1	0	0,40	2	1	1	1	2	1	0	1	0	1	1	0	1
68	0,63	0	1	2	0	0,40	3	0	1	0	1	0	0	2	1	1	1	0	1
69	0,59	1	1	1	0	0,60	1	1	1	0	2	0	1	1	1	1	0	1	-1
70	0,58	1	3	0	0	0,60	0	0	1	0	0	0	0	1	0	1	1	0	1
71	0,67	1	0	0	1	0,70	0	0	0	0	2	1	0	2	0	0	0	1	1
72	0,70	1	0	1	1	0,80	1	1	1	0	0	1	1	2	1	0	1	1	-1
73	0,62	0	2	0	1	0,40	4	0	0	1	2	1	1	2	1	1	0	0	-1
74	0,68	0	3	1	0	0,60	2	1	1	0	1	0	0	2	0	1	1	1	-1
75	0,61	1	3	2	1	0,60	1	1	1	0	2	1	1	2	1	0	1	1	-1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
76	0,50	1	1	2	0	0,80	1	0	1	1	3	0	0	2	1	1	0	0	1
77	0,72	0	2	0	0	0,50	0	1	1	0	3	0	1	2	0	0	0	1	-1
78	0,65	1	1	0	0	0,60	0	0	0	0	1	1	1	1	1	0	0	0	-1
79	0,73	0	1	1	0	0,30	1	1	1	0	1	0	1	1	0	1	0	0	1
80	0,73	0	3	1	0	0,70	0	0	1	1	0	0	1	1	0	0	0	1	1
81	0,48	0	3	0	1	0,60	4	0	0	0	0	1	0	1	0	0	1	0	1
82	0,57	1	3	0	1	0,80	1	0	0	1	0	1	0	2	1	0	1	1	1
83	0,64	1	1	2	0	0,40	2	1	1	1	0	0	0	1	0	0	1	0	1
84	0,61	1	1	1	1	0,60	2	0	1	0	3	0	0	2	0	1	0	1	-1
85	0,69	1	3	1	0	0,80	3	1	1	1	3	1	0	2	0	1	0	1	-1
86	0,72	0	2	2	0	0,70	4	1	1	0	3	1	0	1	0	0	1	0	1
87	0,74	0	3	2	1	0,60	4	1	1	1	0	1	0	1	1	1	0	1	1
88	0,66	1	3	0	0	0,40	4	1	1	1	3	1	1	2	1	0	0	0	-1
89	0,46	1	1	0	0	0,70	4	1	1	0	3	0	1	2	1	1	0	0	-1
90	0,70	1	1	0	0	0,40	2	0	1	1	3	1	1	1	0	1	1	0	1
91	0,67	1	3	0	1	0,60	3	1	1	0	0	1	1	2	1	1	1	0	1
92	0,58	0	0	0	1	0,70	1	0	1	1	1	1	0	1	0	0	0	1	1
93	0,69	1	2	1	0	0,60	2	0	0	1	2	1	1	1	1	1	1	0	-1
94	0,60	1	3	2	0	0,80	3	1	1	0	2	1	0	2	1	0	0	1	-1
95	0,57	0	0	1	0	0,40	3	0	1	0	0	0	1	2	1	1	1	0	-1
96	0,61	1	2	1	1	0,50	4	1	1	1	3	1	0	1	0	0	0	0	1
97	0,43	0	1	0	1	0,30	1	1	1	0	1	1	0	2	0	1	0	1	1
98	0,66	1	2	2	0	0,70	0	1	1	1	2	1	1	1	1	1	1	1	-1
99	0,74	0	0	0	1	0,70	4	1	1	0	2	1	1	1	0	0	0	0	-1
100	0,61	0	0	2	0	0,80	3	1	1	1	2	1	0	1	0	1	0	1	1
101	0,63	0	3	0	1	0,60	4	0	1	1	2	1	1	1	0	0	0	1	-1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
102	0,57	1	0	2	1	0,40	3	0	0	1	2	0	0	1	1	0	1	1	1
103	0,88	0	3	1	0	0,60	0	1	1	1	3	0	1	1	0	1	1	1	1
104	0,65	1	1	0	1	0,80	0	0	0	0	0	1	0	2	1	1	1	1	-1
105	0,71	0	1	2	0	0,50	4	0	1	0	1	0	0	1	1	1	0	0	1
106	0,71	0	1	0	0	0,50	2	0	0	1	3	0	0	2	1	1	0	1	-1
107	0,75	1	2	2	1	0,70	0	0	0	0	3	1	1	2	0	0	0	0	1
108	0,73	1	0	2	0	0,80	3	0	0	1	0	0	1	1	1	1	1	1	-1
109	0,87	0	1	0	1	0,80	4	0	1	0	2	1	0	2	0	0	0	0	1
110	0,67	1	2	0	0	0,30	1	0	1	1	3	1	1	2	0	1	0	1	-1
111	0,56	0	1	2	0	0,60	4	1	1	0	2	1	1	1	1	1	1	0	-1
112	0,69	0	0	2	1	0,70	0	1	1	0	2	0	1	2	0	0	0	0	-1
113	0,74	0	3	0	1	0,50	2	0	0	0	3	1	0	2	0	0	1	0	1
114	0,66	1	2	1	0	0,60	4	1	1	0	1	1	1	2	1	1	1	1	-1
115	0,70	0	3	0	1	0,50	3	1	1	1	3	1	1	2	0	1	1	1	1
116	0,84	0	3	1	0	0,40	4	0	0	0	3	1	0	2	0	0	0	0	-1
117	0,65	1	2	1	0	0,70	2	1	1	0	2	0	1	2	0	0	0	0	1
118	0,75	1	0	0	0	0,50	4	1	1	1	0	1	1	1	1	1	1	0	-1
119	0,59	1	3	1	1	0,70	0	0	1	0	1	0	0	2	1	0	1	1	1
120	0,70	0	3	1	0	0,60	2	0	0	0	1	1	0	2	1	1	1	1	1
121	0,73	0	2	2	0	0,70	2	0	1	0	1	0	0	2	0	1	1	0	-1
122	0,65	1	3	0	0	0,40	3	1	1	0	0	1	0	2	1	1	1	1	-1
123	0,48	1	1	0	0	0,70	4	1	1	0	1	0	1	2	1	0	0	0	1
124	0,58	0	2	0	1	0,70	3	1	1	0	3	0	0	1	0	0	0	0	1
125	0,55	0	2	1	1	0,70	0	1	1	0	2	1	0	2	0	0	1	0	1
126	0,72	1	1	2	1	0,50	4	0	1	0	0	0	0	2	0	0	1	1	-1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
127	0,63	1	2	0	0	0,40	3	1	1	0	0	0	1	1	0	1	1	1	-1
128	0,60	0	2	2	1	0,60	0	1	1	0	0	1	0	1	0	0	1	1	-1
129	0,60	0	0	0	0	0,40	2	1	1	1	3	1	1	2	1	1	1	0	-1
130	0,47	1	0	0	0	0,40	0	1	1	0	2	0	1	2	1	0	0	0	-1
131	0,63	0	1	2	1	0,70	4	1	1	1	0	0	0	2	1	0	1	0	1
132	0,58	1	2	0	0	0,40	2	0	0	1	2	0	0	1	1	1	0	1	-1
133	0,74	0	2	2	1	0,80	3	0	1	1	0	0	1	1	0	0	1	1	1
134	0,67	1	2	0	1	0,30	3	0	1	1	1	1	0	2	0	0	0	1	-1
135	0,73	0	1	2	1	0,30	4	0	0	1	0	1	1	2	1	0	1	1	1
136	0,63	1	1	0	0	0,50	0	0	1	0	2	1	1	1	1	1	0	1	-1
137	0,53	1	3	0	1	0,60	1	0	1	0	0	0	1	2	0	0	1	0	-1
138	0,61	1	0	2	0	0,40	4	0	0	0	1	0	1	1	0	1	1	0	-1
139	0,70	1	0	2	1	0,60	2	1	1	0	2	1	1	1	0	1	0	1	1
140	0,72	0	3	2	1	0,70	3	1	1	0	0	1	0	2	0	0	1	1	1
141	0,74	1	2	0	1	0,50	0	0	0	0	3	0	0	1	1	1	1	1	-1
142	0,70	0	0	1	1	0,60	4	0	0	0	3	0	0	2	0	0	0	1	-1
143	0,56	1	2	2	0	0,40	2	0	0	0	1	0	1	1	1	0	0	1	1
144	0,55	0	0	0	1	0,70	2	0	1	1	3	0	0	2	0	1	0	1	1
145	0,51	0	2	1	1	0,40	4	1	1	0	2	1	1	2	0	0	1	1	1
146	0,75	0	0	0	1	0,50	4	1	1	0	2	1	0	2	1	1	0	0	1
147	0,75	1	0	1	1	0,70	0	1	1	0	1	1	1	1	1	1	1	1	-1
148	0,72	1	1	2	0	0,60	1	1	1	0	3	1	0	1	1	0	1	0	-1
149	0,56	1	0	0	0	0,70	3	0	0	1	3	1	1	1	0	0	0	0	1
150	0,65	1	2	2	0	0,50	2	1	1	1	3	1	0	1	1	0	1	1	-1
151	0,74	0	2	0	1	0,50	0	0	1	1	3	1	1	2	0	0	1	1	1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
152	0,74	0	1	1	1	0,40	2	0	0	1	3	1	1	1	1	0	1	0	1
153	0,50	1	3	1	0	0,60	3	1	1	1	3	1	0	1	1	1	0	1	-1
154	0,71	1	2	2	1	0,50	3	0	1	0	1	1	1	2	0	1	0	0	1
155	0,65	0	3	1	1	0,70	4	1	1	1	0	0	1	1	0	1	1	0	-1
156	0,73	0	0	2	0	0,60	4	0	0	1	1	1	1	1	0	1	1	0	-1
157	0,73	0	0	1	0	0,70	4	0	1	0	3	1	1	1	1	1	1	0	-1
158	0,71	1	3	2	0	0,50	4	0	1	1	0	0	0	2	1	1	1	1	1
159	0,75	0	0	0	1	0,80	1	1	1	1	3	1	1	2	1	1	0	1	1
160	0,66	1	2	0	0	0,40	4	1	1	0	2	1	1	1	1	0	0	1	1
161	0,73	1	0	1	0	0,60	0	0	0	0	0	0	1	1	1	1	0	1	-1
162	0,67	0	2	2	1	0,50	0	0	1	0	2	0	0	2	0	1	0	1	1
163	0,62	0	2	1	0	0,70	3	1	1	0	1	1	1	2	1	1	0	1	1
164	0,73	1	2	2	1	0,40	0	0	1	0	3	1	0	2	0	1	0	1	1
165	0,66	1	2	0	0	0,70	3	0	1	0	1	1	0	2	1	0	0	1	-1
166	0,47	1	0	0	0	0,40	3	1	1	1	0	0	0	1	0	0	1	0	1
167	0,57	0	0	1	0	0,50	2	0	0	1	0	1	1	1	1	1	0	0	-1
168	0,68	1	2	1	0	0,60	3	1	1	0	0	1	1	2	0	1	1	1	1
169	0,60	1	3	0	1	0,50	0	1	1	1	1	0	1	2	0	1	0	0	1
170	0,72	0	0	2	0	0,80	1	1	1	1	3	0	1	1	1	1	0	0	1
171	0,55	1	0	0	1	0,70	0	1	1	0	1	1	1	1	1	0	0	1	-1
172	0,68	0	2	0	1	0,50	1	0	1	1	3	1	0	2	0	0	1	0	-1
173	0,72	1	3	2	0	0,70	2	0	0	0	3	0	1	1	0	1	0	1	-1
174	0,88	1	3	0	0	0,60	2	1	1	1	1	0	0	2	0	0	1	0	-1
175	0,56	0	0	2	0	0,50	1	0	0	1	1	1	0	1	0	0	0	1	-1
176	0,60	0	2	0	0	0,50	0	1	1	1	3	1	0	1	1	1	0	1	1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
177	0,71	0	3	0	0	0,30	4	1	1	0	2	1	0	1	0	1	1	1	1
178	0,69	0	1	1	1	0,50	2	0	1	0	0	1	1	1	0	0	0	0	1
179	0,70	1	3	1	1	0,50	4	0	1	0	0	1	1	1	1	0	0	1	1
180	0,68	0	2	2	0	0,70	1	1	1	0	1	0	0	2	0	1	1	1	-1
181	0,57	0	3	0	0	0,50	1	0	0	0	2	0	0	2	0	1	0	0	-1
182	0,75	1	0	2	0	0,30	4	0	0	0	0	1	0	2	1	1	0	0	1
183	0,58	1	1	0	0	0,40	0	0	1	0	1	1	1	2	0	0	1	0	1
184	0,73	1	3	2	1	0,30	2	0	0	1	2	1	1	2	0	0	1	0	1
185	0,56	1	1	1	0	0,70	4	1	1	1	3	0	0	2	1	1	1	0	-1
186	0,89	1	0	2	0	0,40	4	0	1	1	3	0	0	2	0	0	0	0	-1
187	0,59	0	1	0	0	0,40	2	1	1	0	2	1	0	2	0	1	1	0	-1
188	0,73	1	2	1	0	0,80	4	1	1	0	3	1	0	1	0	1	0	0	-1
189	0,72	0	0	0	0	0,70	1	0	1	0	0	0	1	2	0	1	1	0	-1
190	0,69	1	3	0	1	0,40	3	0	1	1	3	1	0	1	0	0	1	0	-1
191	0,56	1	1	2	1	0,50	1	1	1	1	2	0	0	1	0	0	1	0	1
192	0,62	0	1	0	0	0,60	0	1	1	0	3	1	0	1	0	0	1	1	-1
193	0,67	0	2	2	1	0,40	1	1	1	0	3	0	1	1	1	1	1	0	-1
194	0,49	1	2	0	1	0,40	4	0	1	1	0	0	0	1	0	1	0	0	1
195	0,74	1	0	0	0	0,60	3	0	1	0	3	0	1	1	1	0	1	0	-1
196	0,63	1	2	1	0	0,60	4	1	1	0	2	1	1	1	1	1	1	0	-1
197	0,71	0	2	2	0	0,50	1	0	1	0	0	1	1	2	0	0	0	0	1
198	0,70	1	3	2	0	0,40	2	0	1	1	2	0	1	2	1	1	1	1	1
199	0,57	1	2	2	1	0,70	1	1	1	1	3	1	1	2	1	0	0	1	-1
200	0,74	0	2	2	0	0,60	1	1	1	0	2	0	0	2	1	0	1	1	-1
201	0,65	0	2	2	0	0,30	1	0	0	0	3	1	0	2	0	0	0	0	-1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
202	0,56	0	0	2	0	0,80	4	0	1	0	2	0	0	2	1	0	1	1	1
203	0,75	0	2	0	0	0,60	2	1	1	1	0	0	0	1	0	1	0	1	1
204	0,72	1	1	1	1	0,50	2	1	1	1	1	1	0	1	0	0	0	1	-1
205	0,58	1	1	1	0	0,50	3	0	0	1	1	1	0	1	0	0	0	0	-1
206	0,61	1	2	0	1	0,60	2	0	0	1	2	1	1	2	1	1	0	0	1
207	0,57	1	2	0	1	0,60	0	0	0	0	2	1	1	1	0	0	0	1	-1
208	0,65	0	0	0	0	0,40	0	0	1	1	2	1	0	2	1	1	1	1	1
209	0,55	1	1	2	1	0,40	1	1	1	1	1	0	0	2	1	0	1	0	1
210	0,62	1	0	1	1	0,50	4	1	1	1	0	1	1	2	1	0	1	0	1
211	0,60	1	2	1	1	0,50	0	0	1	1	0	0	0	1	1	1	0	1	1
212	0,71	1	1	1	0	0,40	1	1	1	0	3	0	1	2	0	1	1	1	1
213	0,67	1	3	1	1	0,30	0	1	1	0	0	0	1	2	0	1	1	1	1
214	0,57	1	3	2	1	0,30	3	1	1	0	3	1	0	2	1	0	1	0	-1
215	0,71	0	3	1	1	0,60	3	0	0	0	2	1	0	2	0	0	0	0	-1
216	0,71	1	1	1	0	0,60	4	0	1	1	0	1	0	2	0	0	0	0	-1
217	0,58	1	1	2	0	0,70	2	0	0	1	3	1	1	1	1	0	1	0	1
218	0,61	0	3	0	1	0,70	1	0	0	1	3	0	0	2	1	0	0	1	-1
219	0,61	0	2	1	0	0,50	2	1	1	1	2	0	0	2	1	0	1	0	1
220	0,71	1	2	1	0	0,70	2	0	0	0	2	1	1	2	0	1	0	1	-1
221	0,58	1	1	0	0	0,70	3	0	0	0	2	0	0	2	0	0	0	0	1
222	0,60	0	1	1	0	0,70	2	0	1	1	2	0	0	1	1	1	0	0	-1
223	0,56	0	2	0	1	0,70	3	0	0	0	3	1	1	2	0	0	0	1	-1
224	0,55	1	0	2	1	0,30	4	0	1	1	2	0	0	2	0	0	0	0	1
225	0,57	1	0	0	1	0,50	3	1	1	0	1	1	0	1	1	1	0	0	1
226	0,57	0	3	2	1	0,70	4	0	1	0	1	0	1	2	1	1	1	0	-1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
227	0,65	0	3	0	0	0,60	3	0	1	1	3	1	1	2	1	1	0	1	1
228	0,72	0	2	2	1	0,80	2	0	1	1	1	0	0	1	0	0	1	1	-1
229	0,74	1	3	1	1	0,70	1	0	0	1	3	1	0	2	1	0	0	1	1
230	0,72	0	0	0	1	0,50	1	0	1	1	0	1	1	2	1	0	0	1	1
231	0,62	1	1	1	1	0,60	0	1	1	0	2	1	0	2	1	0	0	0	1
232	0,69	0	2	0	0	0,50	1	1	1	1	1	0	1	2	1	0	0	0	-1
233	0,72	1	0	1	1	0,50	3	1	1	1	1	1	1	1	1	0	1	0	-1
234	0,75	0	3	2	1	0,70	3	0	0	0	3	1	1	1	1	0	0	1	1
235	0,70	0	0	0	0	0,50	3	1	1	1	0	0	1	2	1	0	1	1	1
236	0,75	0	2	2	0	0,40	3	0	0	0	3	1	0	1	1	1	0	0	-1
237	0,67	1	2	0	0	0,50	0	0	1	0	0	1	0	2	1	0	1	0	1
238	0,61	0	1	2	0	0,60	2	0	1	1	1	0	1	1	0	0	0	0	-1
239	0,73	0	2	2	0	0,40	2	1	1	0	3	1	1	1	1	1	1	1	1
240	0,71	1	0	0	0	0,60	3	1	1	0	0	1	1	1	1	1	1	1	-1
241	0,73	1	0	2	1	0,40	4	1	1	1	1	1	1	2	1	0	1	1	1
242	0,72	1	0	0	0	0,30	1	0	1	1	3	0	1	1	0	1	0	0	-1
243	0,69	0	2	1	0	0,50	2	1	1	0	0	0	0	2	1	0	1	0	-1
244	0,63	1	3	2	1	0,50	2	0	1	0	0	0	1	1	0	1	0	1	-1
245	0,59	1	1	0	1	0,80	2	1	1	1	1	0	0	1	1	1	1	0	1
246	0,63	0	1	0	0	0,40	3	1	1	1	0	0	0	1	0	0	1	1	-1
247	0,57	0	3	2	0	0,40	3	0	1	0	0	0	0	2	1	1	1	1	-1
248	0,61	1	1	2	0	0,80	0	0	1	0	2	1	0	1	0	1	0	1	-1
249	0,59	0	0	2	0	0,30	0	0	1	1	3	0	0	1	1	0	1	1	1
250	0,68	1	3	1	1	0,80	4	0	0	0	3	0	0	2	1	0	0	1	-1
251	0,60	0	1	0	1	0,30	1	0	0	1	0	0	1	1	1	0	1	1	1

Çizelge Ek 7.1 Hasta veri seti (devam)

No	Ö1	Ö2	Ö3	Ö4	Ö5	Ö6	Ö7	Ö8	Ö9	Ö10	Ö11	Ö12	Ö13	Ö14	Ö15	Ö16	Ö17	Ö18	S
252	0,66	0	3	0	0	0,60	2	0	0	1	0	0	1	1	0	1	1	0	1
253	0,48	1	0	1	0	0,50	1	0	1	0	1	1	0	1	0	0	1	0	1
254	0,56	0	2	2	0	0,40	4	0	0	0	0	0	0	2	0	1	1	0	-1
255	0,66	1	3	0	1	0,30	4	0	0	1	3	1	1	2	0	0	1	0	1
256	0,62	0	3	2	0	0,30	0	0	1	0	2	1	0	1	0	1	1	0	-1

EK 8 KISITLI MÜHENDİSLİK PROBLEMLERİ SABİT VERİLER

Çizelge Ek 8.1 Kaynak yapılmış kiriş problemi sabit verileri

Sembol	Değer
C_1	$0.10471 \frac{\$}{in.^3}$
C_2	$0.04811 \frac{\$}{in.^3}$
τ_d	$13\ 600\ psi$
σ_d	$30\ 000\ psi$
δ_d	$0.25\ in.$
E	$30 \times 10^6\ psi$
G	$12 \times 10^6\ psi$
P	$6000\ lb$
L	$14\ in.$

Çizelge Ek 8.2 Yay problemi sabit değerleri

Sembol	Değer
F_{max}	$1000\ lb$
l_{max}	$14\ in.$
d_{min}	$0,2\ in.$
S	$189\ 000\ psi$
D_{max}	$3,0\ in.$
F_p	$300\ lb$
σ_{pm}	$6\ in.$
σ_w	$1,25\ in.$
G	$11,5 \times 10^6\ psi.$

ÖZGEÇMİŞ

Adı Soyadı :Alkın YURTKURAN

Doğum yeri ve Tarihi :Bursa, 21/11/1983

Yabancı Dili :İngilizce

Eğitim Durumu

Lise :Bursa Anadolu Lisesi, 2001.

Lisans :Orta Doğu Teknik Üniversitesi, 2006.

Yüksek Lisans :Uludağ Üniversitesi, 2009.

Çalıştığı Kurum : Uludağ Üniversitesi, Mühendislik Fakültesi, 2006-....

İletişim : alkin@uludag.edu.tr

Yayınları (Sadece uluslararası hakemli dergilerde yayınlanan yayınlar verilmiştir):

1. **Yurtkuran, A., Emel, E. 2010.** A new hybrid electromagnetism-like algorithm for capacitated vehicle routing problems. *Expert Systems with Applications*, 37(4): 3427-3433.
2. **Yurtkuran, A., Emel, E. 2008.** Simulation based decision-making for hospital pharmacy management. Winter Simulation Conference, 2008. WSC 2008.
3. **Yurtkuran, A., Emel, E. 2014.** A Modified Artificial Bee Colony Algorithm for-Center Problems. *The Scientific World Journal*, 2014.
4. **Yıldız, T., Özalp, B. T., Küçüköğlü, İ., Yurtkuran, A., Öztürk, N. 2014.** A Linear Programming Approach for Parallel Cell Scheduling with Sequence-Dependent Setup Times. *American Journal of Operations Research*, 4(03): 155.
5. **Yurtkuran, A., Emel, E. 2014.** Efficient Constraint Handling in Electromagnetism-Like Algorithm for Traveling Salesman Problem with Time Windows. *The Scientific World Journal*, 2014.
6. **Yurtkuran, A., Tok, M., Emel, E. 2013.** A Clinical Decision Support System for Femoral Peripheral Arterial Disease Treatment. *Computational and mathematical methods in medicine*, 2013.