



T.C.
Uludağ Üniversitesi
Fen Bilimleri Enstitüsü

**TAŞIT ELEMANLARININ OPTİMUM
TASARIMI İÇİN BİLGİSAYAR DESTEKLİ
ANALİZ VE SİMÜLASYON TABANLI
BÜTÜNLEŞİK BİR ALGORİTMA
GELİŞTİRİLMESİ**

İDRİS KAREN

Doktora Tezi

**TAŞIT ELEMANLARININ OPTİMUM
TASARIMI İÇİN BİLGİSAYAR DESTEKLİ
ANALİZ VE SİMÜLASYON TABANLI
BÜTÜNLEŞİK BİR ALGORİTMA
GELİŞTİRİLMESİ**

İDRİS KAREN



T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

TAŞIT ELEMANLARININ OPTİMUM TASARIMI İÇİN BİLGİSAYAR DESTEKLİ
ANALİZ VE SİMÜLASYON TABANLI BÜTÜNLEŞİK BİR ALGORİTMA
GELİŞTİRİLMESİ

İdris KAREN

Prof.Dr. Ferruh ÖZTÜRK
(Danışman)

DOKTORA TEZİ
MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI

BURSA – 2011

Her Hakkı Saklıdır

TEZ ONAYI

İDRİS KAREN tarafından hazırlanan “TAŞIT ELEMANLARININ OPTİMUM TASARIMI İÇİN BİLGİSAYAR DESTEKLİ ANALİZ VE SİMÜLASYON TABANLI BÜTÜNLEŞİK BİR ALGORİTMA GELİŞTİRİLMESİ” adlı tez çalışması aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Uludağ Üniversitesi Fen Bilimleri Enstitüsü Makine Mühendisliği Anabilim Dalı’nda **DOKTORA TEZİ** olarak kabul edilmiştir.

Danışman : Prof. Dr. Ferruh ÖZTÜRK

İkinci Danışman : ---
(Varsa Yazılacak)

Başkan:	Prof. Dr. Ferruh ÖZTÜRK Uludağ Ü. Mühendislik Mimarlık Fakültesi, Otomotiv Mühendisliği Anabilim Dalı	İmza
Üye:	Prof. Dr. Recep EREN Uludağ Ü. Mühendislik Mimarlık Fakültesi, Tekstil Mühendisliği Anabilim Dalı	İmza
Üye:	Prof. Dr. Y. Samim ÜNLÜSOY Ortadoğu Teknik Ü. Mühendislik Fakültesi, Makine Mühendisliği Anabilim Dalı	İmza
Üye:	Doç. Dr. Necmettin KAYA Uludağ Ü. Mühendislik Mimarlık Fakültesi, Makine Mühendisliği Anabilim Dalı	İmza
Üye:	Yrd. Doç. Dr. Erol SOLMAZ Uludağ Ü. Mühendislik Mimarlık Fakültesi, Otomotiv Mühendisliği Anabilim Dalı	İmza

Yukarıdaki sonucu onaylarım / ONAY

Prof. Dr. Kadri ARSLAN
Enstitü Müdürü
... / ... / 2011

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.

... / ... /

İmza

İdris KAREN

ÖZET

Doktora Tezi

TAŞIT ELEMANLARININ OPTİMUM TASARIMI İÇİN BİLGİSAYAR DESTEKLİ ANALİZ VE SİMÜLASYON TABANLI BÜTÜNLEŞİK BİR ALGORİTMA GELİŞTİRİLMESİ

İdris KAREN

Uludağ Üniversitesi
Fen Bilimleri Enstitüsü
Makine Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Ferruh ÖZTÜRK

Taşıt tasarımında son yıllarda artan müşteri memnuniyeti ile birlikte üretimin minimum maliyette ve maksimum kalitede gerçekleştirilmesi beklentisinin karşılanabilmesi açısından özellikle yeni tasarımların oluşturulmasında ve ortaya çıkan problemlerin çözümünde optimizasyon yöntemlerinin kullanılma gerekliliği oldukça artmıştır. Tasarımı gerçekleştiren veya tasarım sürecinde oluşan problemleri çözmeye çalışan araştırmacı karşılaştığı optimizasyon problemlerinin çözümünde güvenilir ve etkin bir şekilde kullanabileceği optimizasyon algoritmasına gereksinim duymaktadır. Bu tez çalışmasında kullanışlı, global, gerçek optimum değere yakınsamada güvenilir olan ve aynı zamanda optimizasyon sürecinin başlangıcından bitişine kadar çok az hesaplama zamanı tüketen yeni bir algoritma geliştirmek ve bu algoritmanın taşıt tasarım optimizasyon problemlerinin çözümünde etkin bir şekilde kullanılması amaçlanmıştır.

Bu çalışmada evrimsel algoritmalar arasında yer alan global, güvenilir, kullanışlı ve birçok test probleminde başarılı bir şekilde geçen diferansiyel gelişim algoritması ele alınmış ve popülasyon içindeki en iyi vektörleri fark vektörleri olarak kullanan yeni bir mutasyon stratejisi geliştirilerek yeni geliştirilen algoritmada (DEBVs) kullanılmıştır. Geliştirilen DEBVs algoritması ile literatürde mevcut bulunan ve yeni geliştirilen birçok algoritmanın test edilmesinde sıklıkla kullanılan kısıt içeren test problemleri çözdürülmüş ve diğer yöntemlere göre daha düşük sayıda hesaplama ile daha iyi sonuçlar elde edilmiştir. Böylece geliştirilen yeni algoritmanın doğruluğu mevcut sonuçlardan daha iyi sonuçlara daha hızlı bir şekilde ulaşma imkânı ile kanıtlanmıştır. Ayrıca önerilen yöntem ile literatürde mevcut tek ve çok-amaçlı çeşitli mühendislik problemleri çözdürülmüş ve mevcut sonuçlardan daha iyi sonuçlar elde edilerek geliştirilen yöntemin doğruluğu pekiştirilmiştir.

Taşıt tasarımında üç farklı problem (sürüş konfor optimizasyonu, optimum sac kalıp tasarımı ve optimum motor askı takoz tasarımı) ele alınmış ve geliştirilen DEBVs algoritması ile çözülmüştür. Geliştirilen DEBVs algoritması ile diğer evrimsel algoritmalara ve klasik optimizasyon yöntemlerine göre çok daha güvenilir, global ve etkin sonuçlar elde edilmiştir. Geliştirilen yöntemin taşıt tasarım sürecinde tasarımcıya yardımcı bir araç olarak kullanılması zaman, maliyet, kalite, hız ve kolaylık açılarından büyük avantajlar sağlaması beklenmektedir.

Anahtar Kelimeler: Diferansiyel gelişim algoritması, en iyi vektörleri kullanan diferansiyel gelişim algoritması (DEBVs), hiperelastik malzeme, kısıtlı ve çok-amaçlı optimizasyon, motor askı takozu, sac metal kalıbı, sonlu elemanlar yöntemi, sürüş konforu, topoloji optimizasyonu

2011, xvi + 258 sayfa.

ABSTRACT

Ph.D. Thesis

DEVELOPING AN INTEGRATED COMPUTER AIDED ANALYSES AND SIMULATION BASED ALGORITHM FOR OPTIMUM DESIGN OF VEHICLE COMPONENTS

İdris KAREN

Uludag University
Graduate School of Natural and Applied Sciences
Department of Mechanical Engineering

Supervisor: Prof. Dr. Ferruh ÖZTÜRK

In recent years, expectation of carrying out the production in minimum cost and maximum quality has expanded with the raising customer satisfaction. In order to meet this expectation, the necessity of using optimization methods has increased especially in making brand new designs and solving problems appeared. The researcher who generates the design or tries to solve the problems which appear during the design process needs a reliable and efficient optimization algorithm when solving the optimization problems appeared. In this thesis study, it is aimed to develop a new algorithm that is user-friendly, global, reliable in converging the real optimum value, and also consuming as short time as possible from the beginning to the end of the optimization process.

In this study differential evolution which was among the best evolutionary algorithms with having global, robust and useful properties and which was tested successfully from many test problems was handled and a new mutation strategy which uses the best vectors in the population as differential vectors was developed and used in the new developed algorithm (DEBVs). Constraint test problems which were frequently used as a test tool for many new algorithms were solved with the developed DEBVs algorithm and better results with less function evaluation numbers were handled when comparing the results of other algorithms. In this way the accuracy of the developed algorithm was demonstrated with reaching better solutions faster. Also various single and multi-objective engineering problems available in the literature were solved by proposed algorithm and the accuracy of the algorithm was reinforced with obtaining better results.

In the vehicle design stage three different problem (vehicle ride comfort optimization, optimum sheet metal die design and optimum engine mount design) were handled and solved with developed DEBVs algorithm. More robust, fast and effective results were handled according to other classical optimization methods and evolutionary algorithms. The developed algorithm can be very helpful as an assistant tool for engineers during vehicle design and manufacturing process in terms of time, cost, quality and convenience.

Key words: Constraint and multi-objective optimization, differential evolution, differential evolution using the best vectors (DEBVs), engine mount, finite elements method, hyperelastic material, ride comfort, sheet metal stamping, topology optimization

2011, xvi + 258 pages.

TEŞEKKÜR

Tez çalışmalarım süresince yaptığı tüm katkılardan dolayı tez danışmanım ve değerli hocam Prof. Dr. Ferruh Öztürk'e saygı ve teşekkürlerimi sunarım.

Doktora tez çalışmalarım süresince, yardımlarını esirgemeyen değerli hocam Doç. Dr. Necmettin Kaya'ya ve tez izleme komitesinde bulunup her altı ayda bir değerli görüşleriyle katkıda bulunan sayın Prof. Dr. Recep Eren'e teşekkür ederim.

Bu tez çalışmasına hem Yurtiçi Doktora Bursu – 2211 ile maddi destek sağlayan hem de 104M240 numaralı proje ile finansal destek sağlayan Türkiye Bilimsel ve Teknolojik Araştırma Kurumu'na (TÜBİTAK), M2004-27, M2006-33, M2008-64 kodlu projeler ile geliştirilen algoritmanın test edilmesi ve taşıt tasarım problemlerinin çözümünde gerekli donanımın alımında destek sağlayan Uludağ Üniversitesi Bilimsel Araştırma Projeleri Birimi'ne (BAP) teşekkür ederim.

Sürüş konforu ile ilgili taşıt tasarım probleminin modellenmesinde ve çözülmesinde Yurtdışı Doktora Araştırma Bursu ile maddi destek sağlayan Yüksek Öğretim Kurumu'na (YÖK) ve uygulamadaki yardımlarından dolayı Torino Politeknik Üniversitesi'ndeki (İtalya) değerli hocalarıma, yol testlerinin gerçekleştirilmesinde gerekli ölçüm ekipmanlarını ve gerekli taşıtları sağlayan TOFAŞ Türk Otomobil Fabrikası A.Ş.'ne teşekkür ederim.

Motor titreşim takozunun malzeme ve ürün testlerinin gerçekleştirilmesinde laboratuvarının ve test ekipmanlarının kullanımına izin veren Bayrak Plastik ve TOFAŞ Türk Otomobil Fabrikası A.Ş.'ne teşekkür ederim.

Kalıp tasarımı ve yapısal optimizasyon problemi için değerli bilgilerini esirgemeyen değerli hocam Prof. Dr. Bilgin Kaftanoğlu'na ve kalıp modellerinin temininde, gerilme – yer değiştirme değerlerinin ölçümü için uygun preslerin kullanımında yardımlarını esirgemeyen TOFAŞ Kalıp Birimi'ne teşekkür ederim.

Desteklerini esirgemeyen tüm arkadaşlarıma teşekkür ederim.

Her anımda yanımda olan aileme şükranlarımı sunarım.

Benimle birlikte tüm sıkıntılara katlanarak bana her zaman destek olan değerli eşime teşekkür ederim.

İdris KAREN
21 / 09 / 2011

İÇİNDEKİLER

	Sayfa
ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER	iv
SİMGELER ve KISALTMALAR DİZİNİ	vii
ŞEKİLLER DİZİNİ	xi
ÇİZELGELER DİZİNİ	xv
1. GİRİŞ	1
2. KAYNAK ÖZETLERİ	4
2.1. Diferansiyel Gelişim.....	7
2.2. Emin Tasarım (Robust Design).....	10
2.3. Sürüş Konfor Optimizasyonu.....	10
2.4. Sürüş Konforu ile ilgili Standartlar	11
2.5. Kalıp Tasarım Optimizasyonu	12
2.6. Motor Titreşim Takozu Tasarım Optimizasyonu	15
3. MATERYAL VE YÖNTEM.....	17
3.1. Optimizasyon Algoritmalarının Sınıflandırılması	20
3.2. Tek-Noktalı Türeve Bağlı Optimizasyon Yöntemleri	21
3.3. Tek-Noktalı Türevden Bağımsız Optimizasyon Yöntemleri ve Adım Büyüklüğü Problemi	26
3.3.1. Saydırma veya kaba kuvvet (brute force) ile arama	29
3.3.2. Rastgele yürüyüş (random walk) yöntemi.....	30
3.3.3. Hooke ve Jeeves yöntemi	31
3.4. Lokal ve Global Optimizasyon	32
3.5. Benzetimli Tavlama Yöntemi	34
3.6. Çok-Noktalı Türeve Bağlı Optimizasyon Yöntemleri.....	35
3.7. Çok-Noktalı Türevden Bağımsız Optimizasyon Yöntemleri.....	36
3.7.1. Evrimsel algoritmaların temel özellikleri.....	36
3.7.2. Evrimsel stratejiler	38
3.7.3. Genetik algoritmalar	39
3.7.4. Nelder ve Mead yöntemi	42
3.7.5. Evrimsel programlama	43

3.7.6. Parçacık sürü optimizasyonu	44
3.8. Diferansiyel Gelişim Algoritması.....	45
3.8.1. Diferansiyel gelişim algoritmasının diğer yöntemlere göre zayıf yönleri ve bu zayıf yönleri gidermek için geliştirilen yaklaşımlar.....	61
3.8.2. Dinamik diferansiyel gelişim algoritması	64
3.8.3. Değiştirilmiş diferansiyel gelişim algoritması.....	66
3.8.4. Hibrit diferansiyel gelişim algoritması.....	69
3.8.5. Diferansiyel gelişim algoritmasının performansını etkileyen parametreler	72
3.9. Geliştirilen Optimizasyon Algoritması (DEBVs).....	78
3.9.1 Stratejinin uygulanacağı nesil oranı (N).....	87
3.9.2 Fark vektörü öncelik kriteri (D).....	87
3.9.3 Tekrar sayısı (T)	87
3.9.4 Başlangıç popülasyonu	87
3.9.5 Geliştirilen mutasyon stratejisi	88
3.9.6 Çaprazlama.....	89
3.9.7 Seçim	90
3.10. Kısıt İçeren Optimizasyon	91
3.10.1 Sınır kısıtları.....	92
3.10.2 Eşitlik kısıtları	94
3.10.3 Eşitsizlik kısıtları	95
3.11. Çok Amaçlı Optimizasyon.....	97
3.11.1. Ağırlıklı toplam yöntemi	98
3.11.2. Genelleştirilmiş diferansiyel gelişim.....	99
3.11.3. Pareto diferansiyel gelişim	99
3.12. Deney Tasarımı ve Cevap Yüzeyi Yöntemi.....	101
3.13. Geliştirilen Algoritmanın Test Problemleri ile Değerlendirilmesi	108
3.14. Mühendislik Problemleri ile Geliştirilen Algoritmanın Performansının Değerlendirilmesi.....	126
3.14.1. Basınçlı kap tasarım problemi	127
3.14.2. Kaynaklı kiriş tasarım problemi	129
3.14.3. Çok-amaçlı kaynaklı kiriş tasarım problemi	134
3.14.4. Çok-amaçlı iki-çubuklu kiriş tasarım problemi	137
3.14.5. Çok-amaçlı I-kesitli kiriş tasarım problemi.....	140

4. BULGULAR VE TARTIŞMA.....	144
4.1. DEBVs ile Sürüş Konfor Optimizasyon Probleminin Çözülmesi	144
4.1.1. Sürüş konforunun hesaplanması	146
4.1.2. Fiziksel testler.....	150
4.1.3. Sürüş indeksinin hesaplanması, tam-taşıt modelinin simülasyonu ve parametrelerin tespiti	154
4.1.4. Fiziksel test sonuçları ile simülasyon-tabanlı model sonuçları arasındaki uyum	161
4.1.5. Sürüş konfor optimizasyon probleminin oluşturulması, çözülmesi ve sonuçları	164
4.1.6. Sürüş konfor optimizasyon çalışması ile elde edilen kazanımlar ve yapılabilecek yeni çalışmalar	166
4.2. DEBVs ile Kalıp Tasarım Optimizasyon Probleminin Çözülmesi.....	168
4.2.1. Kalıp tasarım süreci	169
4.2.2. Doğrusal olmayan simülasyonlar.....	171
4.2.3. Fiziksel testler.....	175
4.2.4. Simülasyonların fiziksel testler ile doğrulanması	176
4.2.5. Kalıp tasarım optimizasyon probleminin oluşturulması, çözülmesi ve sonuçları....	178
4.2.6. Kalıp tasarım optimizasyon çalışması ile elde edilen kazanımlar ve yapılabilecek yeni çalışmalar	188
4.3. DEBVs ile Motor Titreşim Takoza Tasarım Optimizasyon Probleminin Çözülmesi	190
4.3.1. Malzeme modelleri	193
4.3.2. Ürün malzeme testleri	197
4.3.3. Ürün model simülasyonu ve tasarım doğrulaması	201
4.3.4. Motor titreşim takoza tasarım optimizasyon probleminin oluşturulması, çözülmesi ve sonuçları.....	202
4.3.6. Motor titreşim takoza tasarım optimizasyon çalışması ile elde edilen kazanımlar ve yapılabilecek yeni çalışmalar	207
4.4. Test Problemleri Seçimi ile ilgili Tartışma	208
4.5. Mühendislik Problemleri ile ilgili Tartışma	215
5. SONUÇ	218
KAYNAKLAR.....	222
EKLER.....	243
ÖZGEÇMİŞ.....	254

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler	Açıklama
a	Amortisör Özellikleri
α_j^i	i . Popülasyonda j . Tasarım Değişkeni için $[0,1]$ Aralığındaki Rastgele Reel Sayı Üretici
β	Cevap Yüzey Fonksiyonundaki Katsayı
γ	Adım Boyutu
$\Delta \mathbf{x}$	Fark Vektörü
$\Delta \mathbf{x}$	Gauss Dağılımı
ε	Eşitlik Kısıtındaki Hata Değeri
η	Boyutsuz Kayıp Faktörü
E	Elastisite Modülü
λ	Evrimsel Stratejilerde Çocuk Popülasyon Vektörü
μ	Ortalama Değer
μ	Evrimsel Stratejilerde Ebeveyn Vektörleri
σ	Standart Sapma
∇	Nabla Operatörü
b	Kiriş Genişliği
b_i^L	x_i Optimizasyon Parametresinin Arama Uzayını İfade Eden Alt Sınır
b_i^U	x_i Optimizasyon Parametresinin Arama Uzayını İfade Eden Üst Sınır
b_j^L	j . Parametre için Belirlenen Alt Sınır
b_j^U	j . Parametre için Belirlenen Üst Sınır
B	x Optimizasyon Parametresi için Alfabe Setindeki Taban
\mathbf{B}^{n+1}	Popülasyon Sayısının İki Katına Sahip Tampon Popülasyonu
$C_i^+(\mathbf{x})$	i . Amaç Fonksiyonunda Büyük Olma Şartını İçeren Eşitsizlik Kısıtı
$C_i^-(\mathbf{x})$	i . Amaç Fonksiyonunda Küçük Olma Şartını İçeren Eşitsizlik Kısıtı
$C_i^=(\mathbf{x})$	i . Amaç Fonksiyonunda Eşitlik Kısıtı
$\mathbf{c}^{n+1,i}$	$(n+1)$. Popülasyonda i . Birey Yerine Çaprazlamadan Sonra Oluşturulan Çocuk Birey
CR	Çaprazlama Oranı
d	Tel Çapı
d	Yeni Vektör ile Eski Vektör Arasındaki Fark Vektörü
D	Fark Vektörü Öncelik Kriteri
D	Ortalama Yay Çapı
D	Koordinat Ekseni Yönleri
D	Bağımsız Değişken Sayısı
D	Sınır Kısıt Sayısı
f	Maksimize Edilecek Özellik olan Konfor
\mathbf{f}	Amaç ve Kısıt Fonksiyon Değerlerinin Birleşimi
f	Titreşim Frekansı

F	Ölçeklendirme Faktörü
$F1$	Nelder-Mead Yönteminde Ağırlık Faktörü
$F2$	Nelder-Mead Yönteminde İkinci Ağırlık Faktörü
F	Frekans
$f(\mathbf{x})$	Amaç Fonksiyonu
$f_i^{min}(\mathbf{x})$	i . Amaç Fonksiyonunda Minimumu Aranana Amaç Fonksiyonu
$f_i^{max}(\mathbf{x})$	i . Amaç Fonksiyonunda Maksimumu Aranana Amaç Fonksiyonu
f_n	Sistemin Doğal Frekansı
F_y	[0,1] Aralığında Olacak Şekilde y 'nci Vektör Farkı için Belirlenen Ölçeklendirme Faktörü
G^*	Elastomerik Kesme Modülü (Kompleks Modül)
G'	Saklanan Modül
G''	Kaybolan Modül
g01	1. Test Fonksiyonu
$\mathbf{g}(\mathbf{x})$	Gradyan Vektörü
g_{ij}	x_i Tasarım Değişkenini Temsil Eden j 'nci Gen
$g(x)$	Eşitsizlik Kısıtı
$\mathbf{G}(\mathbf{x})$	Hessian Kare Matrisi
$G_i(x)$	Eşitsizlik Kısıtında i . Ceza Fonksiyonu
h	Kaynak Yüksekliği
h	Takoz Kol Derinliği
$h_j(x)$	Eşitlik Kısıtı
I	Birim Matris
I	Atalet Momenti
k	Kromozom
K	Probleme Bağlı Olarak Değişebilen Sabit
K'	Dinamik Yay Rijitliği
l	Kaynak Uzunluğu
L	Basıncılı Kabin Silindirik Kısmının Uzunluğu
L_i	x_i Tasarım Değişkeni için Temsil Edilen Genlerin Sayısı
m	Taşıt Kütlesi
m	Eşitsizlik Kısıt Sayısı
m_s	Süspansiyonun Kütlesi
m_t	Tekerleğin Kütlesi
n	Gözlenen Deney Tasarımı Değerleri
N	Stratejinin Uygulanacağı Nesil Oranı
N	Yay Sarmal Sayısı
N_{c^+}	Büyük Olma Şartını İçeren Eşitsizlik Kısıtının Sayısı
N_{c^-}	Küçük Olma Şartını İçeren Eşitsizlik Kısıtının Sayısı
$N_{c=}$	Eşitlik Kısıtının Sayısı
$N_{o^{maks}}$	Optimizasyon Problemi İçerisindeki Maksimumu Aranana Amaç Fonksiyonu Sayısı
$N_{o^{min}}$	Optimizasyon Problemi İçerisindeki Minimumu Aranana Amaç Fonksiyonu Sayısı
NP	Popülasyon Sayısı
N_p	Popülasyon Vektör Sayısı

p	Yol Pürüzlülüğü
p	Eşitlik Kısıt Sayısı
P	Deney Tasarımında Parametre
P	I-kesitli Kirişe Etki Eden Yük
P^0	Başlangıç Popülasyonu
P^n	n . Popülasyon
r	Frekans Oranı
R	İç Yarıçap
R^2	R-Kare Testi
S	Deney Tasarımında Seviye
S/N	Ortogonal Dizi Tabloları ile Oluşturulan Gürültü Oranları
t	Kiriş Yüksekliği
t	Takoz Kol Genişliği
T	Başlangıç Sıcaklığı
T	Tekrar Sayısı
T_{ABS}	Titreşim Frekansındaki Geçirgenlik
T_h	Kafa Kısmının Sac Kalınlığı
T_s	Sac Kalınlığı
u_0	Geçici Vektör
u_1	Yeni Vektör
v	Yeni Vektör Üretme Olasılığı
v^{n+1}	Mutant Vektörü
w	Ağırlıklandırılmış Ceza Katsayısı
W	Sistemin Ağırlığı
w_i^+	i . Amaç Fonksiyonunda Büyük Olma Şartını İçeren Eşitsizlik Kısıtının Ağırlığı
w_i^-	i . Amaç Fonksiyonunda Küçük Olma Şartını İçeren Eşitsizlik Kısıtının Ağırlığı
$w_i^=$	Eşitlik Kısıtının Ağırlığı
w_i^{max}	i . Amaç Fonksiyonunda Maksimumu Aranan Amaç Fonksiyonunun Ağırlığı
w_i^{min}	i . Amaç Fonksiyonunda Minimumu Aranan Amaç Fonksiyonunun Ağırlığı
x	Tasarım Değişkeni
x	Temel Vektörün Seçilme Faktörü
x	D -Boyutlu Parametre Vektörü
x_0	Nelder-Mead Yönteminde En İyi Nokta
x_e	Nelder-Mead Yönteminde Yeni Nokta
x_{eniye}	En İyi Bireyin İfade Ettiği Vektör
x_D	Nelder-Mead Yönteminde En Kötü Nokta
$x_{k,alt}$	Alt Sınır Kısıtı
$x_{k,üst}$	Üst Sınır Kısıtı
x_m	En Kötü Noktanın Karşı Yüzeyinin Kitle Merkezi
$x^{n,b,i}$	n 'nci Popülasyonda i 'nci Bireyin Yerine Oluşturulacak Birey için Seçilen (b) Temel Vektörü
$x^{n,eniye,i}$	n 'nci Popülasyonda i 'nci Bireyin Yerine Oluşturulacak Birey için Seçilen ($eniye$) Temel Vektör

$\mathbf{x}^{n,p_{1y}}$	n 'nci Popülasyonda $[1, N_p]$ Arasındaki Tamsayılardan Rastgele Seçilerek Belirlenen p_{1y} 'nci Birey
$\mathbf{x}^{n+1,v,i}$	Fark Vektörü
x_1, x_2	Çubuk Uzunlukları
x_r	Nelder-Mead Yönteminde Yeni Nokta
\mathbf{x}_{r1}	Rastgele Seçilen Birinci Vektör
\mathbf{x}_{r2}	Rastgele Seçilen İkinci Vektör
$\mathbf{x}_{sınır}$	Sınır Vektör
y	Deney Tasarımında Sonuç Deney
y	Yay Özellikleri
y	Yük ile Mesnet Arasındaki Dik Uzaklık
y	Temel Vektöre Eklenen Fark Vektörü
y_i	Gözlenen Deney Tasarımı Değerleri
\bar{y}	Gözlenen Deney Tasarımı Değerlerinin Ortalaması
z	Mutasyon Vektörü Tarafından Belirlenen Parametre Sayısının Nasıl Belirlendiğini Temsil Eden Faktör

Kısaltmalar

Açıklama

ANOVA	Varyans Analizi
BDT	Bilgisayar Destekli Tasarım
CAD	Bilgisayar Destekli Tasarım (Computer Aided Design)
CAE	Bilgisayar Destekli Mühendislik (Computer Aided Engineering)
CAM	Bilgisayar Destekli Üretim (Computer Aided Manufacturing)
DE	Diferansiyel Gelişim Algoritması (Differential Evolution)
DEBVs	En İyi Vektörler Stratejisini Kullanan Diferansiyel Gelişim Algoritması (Differential Evolution using the Best Vectors)
DOE	Deney Tasarımı (Design of Experiment)
FEA	Sonlu Elemanlar Analizi (Finite Element Analysis)
FEM	Sonlu Elemanlar Yöntemi (Finite Element Method)
FFT	Hızlı Fourier Dönüşümü (Fast Fourier Transform)
GUI	Grafiksel Kullanıcı Arayüzü (Graphical User Interface)
ISO	Uluslararası Standartlar Teşkilatı (International Standards Organization)
RI	Sürüş İndeksi (Ride Index)
RMS	Ortalama Karekök Değeri (Root Mean Square)
RSM	Cevap Yüzeyi Yöntemi (Response Surface Method)
SAE	Otomotiv Mühendisleri Derneği (Society of Automotive Engineers)
VDV	Titreşim Dozu Değeri (Vibration Dose Value)

ŞEKİLLER DİZİNİ

		Sayfa
Şekil 3.1.	Minimuma (a) ve maksimuma (b) sahip iki amaç fonksiyonu	24
Şekil 3.2.	Dik-iniş yönteminde ekstremum noktaya ilerleme aşamaları	25
Şekil 3.3.	Minimuma (a) ve maksimuma (b) sahip iki adım fonksiyonu	27
Şekil 3.4.	Türevden bağımsız yöntemlerin çalışma prensibini gösteren akış şeması	28
Şekil 3.5.	Kaba kuvvet yönteminin çalışma prensibi	30
Şekil 3.6.	Rastgele yürüyüş yönteminin çalışma prensibi	31
Şekil 3.7.	Hooke ve Jeeves yönteminin çalışma prensibi	32
Şekil 3.8.	Birden fazla optimum noktaya sahip amaç fonksiyonu	33
Şekil 3.9.	Benzetimli tavlama yönteminin işlem adımları	34
Şekil 3.10.	Bir kromozomun yapısı	37
Şekil 3.11.	Evrimsel stratejiler yönteminin işlem adımları	39
Şekil 3.12.	Genetik algoritmaların işlem adımları	40
Şekil 3.13.	Nelder-Mead yönteminde 2-boyutlu (D=2) bir cisim için yansıma ve ilerleme operasyonu	43
Şekil 3.14.	Parçacık sürü optimizasyon algoritmasının işlem adımları	44
Şekil 3.15.	Diferansiyel gelişim algoritmasında başlangıç popülasyonu	45
Şekil 3.16.	Fark vektörünün oluşturulması: $x_{r1}-x_{r2}$	46
Şekil 3.17.	Mutasyon işlemi	47
Şekil 3.18.	Seçim işlemi	47
Şekil 3.19.	İkinci geçici vektörün oluşturulması	48
Şekil 3.20.	İlk nesil sonunda oluşan yeni bireyler	49
Şekil 3.21.	Dokuz bireyden oluşan örnek bir popülasyondaki olası tüm fark vektörleri (a) ve bu fark vektörlerinin orijine taşınmış dağılımları (b) (Price ve ark. 2005)	49
Şekil 3.22.	Fark vektörlerine uygulanan ölçeklendirmenin etkisi (a) ve bu fark vektörlerinin mevcut nokta etrafını yeterince tarayabilme özelliği (b) (Price ve ark. 2005)	50
Şekil 3.23.	Nesil 1: Diferansiyel gelişim algoritmasında popülasyon ve fark vektörleri dağılımı	51
Şekil 3.24.	Nesil 6: Popülasyonun iki minimum bölge etrafında yoğunlaşması	51
Şekil 3.25.	Nesil 12: Fark vektörü bulutunun üç temel bölgede yoğunlaşması	52

Şekil 3.26.	Nesil 16: Popülasyonun global minimum noktada yoğunlaşması	52
Şekil 3.27.	Nesil 20: Fark vektörlerinin otomatik olarak küçülmesi ve global minimum noktaya yakınsama işlemi	53
Şekil 3.28.	Nesil 26: Fark vektörlerinin iyice küçülmesi ve global minimum noktaya yakınsama işlemi	53
Şekil 3.29.	Nesil 34: Global minimum noktanın yakalanması	54
Şekil 3.30.	Klasik diferansiyel gelişim algoritmasının akış diyagramı	55
Şekil 3.31.	Rastgele yeniden birey oluşturma işleminin sözde kodu	57
Şekil 3.32.	Kendini toparlama ile birey oluşturma işleminin sözde kodu	57
Şekil 3.33.	Çaprazlama işlemi	58
Şekil 3.34.	Üstel çaprazlama işlemi	58
Şekil 3.35.	Üniform çaprazlama işlemi	59
Şekil 3.36.	Dinamik diferansiyel gelişim algoritmasının akış diyagramı	65
Şekil 3.37.	Geliştirilen algoritmada (DEBVs) fark vektörünün oluşturulması	79
Şekil 3.38.	DEBVs`de mutasyon işlemi	79
Şekil 3.39.	DEBVs`de ikinci geçici vektörün oluşturulması	80
Şekil 3.40.	İlk nesil sonunda oluşan yeni bireyler	81
Şekil 3.41.	Örnek problemin eşyüksekti çizgileri ve 3-boyutlu gösterimi	81
Şekil 3.42.	DE ve DEBVs`de 1. nesilden sonra popülasyonların arama uzayında dağılımı	82
Şekil 3.43.	DE ve DEBVs`de popülasyonların 3. nesilde arama uzayında dağılımı	82
Şekil 3.44.	DE ve DEBVs`de popülasyonların 5. nesilde arama uzayında dağılımı	83
Şekil 3.45.	DE ve DEBVs`de popülasyonların 6. nesilde arama uzayında dağılımı	83
Şekil 3.46.	DE ve DEBVs`de popülasyonların 10. nesilde arama uzayında dağılımı	84
Şekil 3.47.	DE ve DEBVs`de popülasyonların 20. nesil başlangıcında arama uzayında dağılımı	84
Şekil 3.48.	DE ve DEBVs ile elde edilen amaç fonksiyon değerlerinin nesil sayısına bağlı değişimi	85
Şekil 3.49.	Geliştirilen algoritmanın (DEBVs) akış diyagramı	86
Şekil 3.50.	Rastgele yeniden birey oluşturma işleminin sözde kodu	89
Şekil 3.51.	Çaprazlama işlemi	89
Şekil 3.52.	Geliştirilen algoritmada kullanılan üniform çaprazlama işlemi	90

Şekil 3.53.	DEBVs`de sınır kısıtlarını sağlamayan birey için kullanılan kendini toparlama yönteminde yeni bireyin oluşturulması	94
Şekil 3.54.	Deney tasarımında sistemin genel modeli (Montgomery 2001)	102
Şekil 3.55.	Deney tasarımı ve cevap yüzeyi yöntemleri ile bir problemin çözümlenmesi	108
Şekil 3.56.	Basınçlı kabın tasarım parametreleri	127
Şekil 3.57.	Kaynaklı kirişin tasarım parametreleri	130
Şekil 3.58.	İki-çubuklu kirişin tasarım parametreleri	138
Şekil 3.59.	I-kesitli kiriş tasarım probleminin tasarım parametreleri	141
Şekil 4.1.	Oturma pozisyonundaki ivmelenme eksenleri	147
Şekil 4.2.	Konforsuzluk eğrileri	148
Şekil 4.3.	Araç üzerine yerleştirilmesi planlanan ivmeölçerlerin konumları (motor bölümü hariç)	151
Şekil 4.4.	Yol testlerinde kullanılan araç üzerine yerleştirilen ivmeölçerler (akson, direksiyon, koltuk, ayak bölgesi ve döşeme altı)	152
Şekil 4.5.	Test işleminde kullanılan cihaz ve ekipmanları	152
Şekil 4.6.	Araçların koştuğu 6 farklı yol, a) engebeli yol, b) stabilize yol, c) Arnavut kaldırımlı yol, d) otoyol, e) test pisti ve f) yamalı yol	153
Şekil 4.7.	(a) Bir, (b) iki ve (c) üç serbestlik derecesine sahip çeyrek-arac modelleri (Genta ve Morello 2009)	154
Şekil 4.8.	Sürüş indeksini hesaplamak için oluşturulan Simulink modelinin temel üç bölümü (koltuk-oturma, sırt ve ayak bölgesi)	155
Şekil 4.9.	İvme değerlerinin modele alınması ve FFT analizi için yazılan fonksiyon kodları	156
Şekil 4.10.	FFT analizi ile ivme değerlerinin frekans alanına dönüşümü	156
Şekil 4.11.	Geliştirilen simülasyon-tabanlı modelin kullanıcı arayüzü (GUI)	157
Şekil 4.12.	Tam-taşıt modeli için geliştirilen simülasyon-tabanlı model (a) Gövde, ayak ve koltuk bölümleri, (b) ön sağ süspansiyon sistemi (A: ön-sağ, B: ön-sol, C: arka-sağ, D: arka-sol), (c) üç takoz ile birlikte motor bölümü	158
Şekil 4.13.	Son hali ile sürüş konforu için geliştirilen arayüz	161
Şekil 4.14.	Dört farklı taşıtın tüm yollardaki sürüş indeks değerleri	162
Şekil 4.15.	Optimizasyon için hazırlanan arayüz	165
Şekil 4.16.	Kalıp tasarım optimizasyon çalışmasının akış diyagramı	170
Şekil 4.17.	Temas analizleri gerçekleştirilen iki boyutlu (2D) kalıp modelleri	171
Şekil 4.18.	Analizleri gerçekleştirilen üç boyutlu (3D) kalıp modelleri	172

Şekil 4.19.	Optimum eleman boyutundaki kalıp modeli (Toplam eleman sayısı: 3 300 870, toplam düğüm noktası sayısı: 684 917, toplam serbestlik derecesi: 2 054 751)	173
Şekil 4.20.	Eleman yakınsama çalışması	173
Şekil 4.21.	Dişi kalıp yüzeylerine etki eden kuvvetler; a) Pot çemberi, b) Erkek kalıp	174
Şekil 4.22.	İvmeölçerlerin kalıp üzerine montaj işlemi ve konumları	175
Şekil 4.23.	Test ekipmanı	176
Şekil 4.24.	Kalıp üzerindeki maksimum yer değiştirme ve gerilme değerleri	176
Şekil 4.25.	Topoloji optimizasyonunun ilerleme aşamaları	178
Şekil 4.26.	Tasarım uzayı	179
Şekil 4.27.	Topoloji optimizasyonu sonrasında ortaya çıkan malzeme dağılımı	180
Şekil 4.28.	Topoloji sonrası oluşturulan alternatif tasarımlar	181
Şekil 4.29.	Şekil optimizasyonu için seçilen A3 modelinin tasarım değişkenleri	182
Şekil 4.30.	Motor titreşim takozu tasarım optimizasyon çalışmasının iş akış şeması	193
Şekil 4.31.	a) Tek serbestlik dereceli titreşim modeli ve b) eşdeğer elastomerik davranış modeli	195
Şekil 4.32.	Hiperelastik malzemeler için çeşitli testler	198
Şekil 4.33.	Tek eksenli çekme testi	199
Şekil 4.34.	Tek eksenli çekme testinde farklı iki yer değiştirme genliğinde malzeme davranışı	199
Şekil 4.35.	Basma testi	200
Şekil 4.36.	Düzlem çekme testi	200
Şekil 4.37.	Tek eksenli çekme testi ile uydurulan malzeme modeli	201
Şekil 4.38.	Örnek uygulama için oluşturulan takoz modeli	201
Şekil 4.39.	Simülasyon sonucunda oluşan gerilme dağılımı	202
Şekil 4.40.	Ürün testinden ve simülasyondan elde edilen rijitlik eğrileri	202
Şekil 4.41.	Örnek uygulama için oluşturulan takoz modeli	203
Şekil 4.42.	Hedef değer ve deney tasarımı sonucunda elde edilen değerlerin kuvvet – yer değiştirme eğrileri	204

ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 3.1. Optimizasyon algoritmalarının sınıflandırılması	21
Çizelge 3.2. Kural tabanlı optimizasyon algoritmaları ile hibritleştirilmiş diferansiyel gelişim algoritmaları (Qing 2009)	70
Çizelge 3.3. Olasılıksal optimizasyon algoritmaları ile hibritleştirilmiş diferansiyel gelişim algoritmaları (Qing 2009)	71
Çizelge 3.4. Tam faktöriyelde deney tasarımı tablosu	103
Çizelge 3.5. Polinom derecesine ve değişken sayısına bağlı olarak değişen katsayı değerleri	106
Çizelge 3.6. İlk test fonksiyonu için elde edilen sonuçlar	110
Çizelge 3.7. İkinci test fonksiyonu için elde edilen sonuçlar	112
Çizelge 3.8. Üçüncü test fonksiyonu için elde edilen sonuçlar	114
Çizelge 3.9. Dördüncü test fonksiyonu için elde edilen sonuçlar	115
Çizelge 3.10. Beşinci test fonksiyonu için elde edilen sonuçlar	116
Çizelge 3.11. Altıncı test fonksiyonu için elde edilen sonuçlar	117
Çizelge 3.12. Yedinci test fonksiyonu için elde edilen sonuçlar	118
Çizelge 3.13. Sekizinci test fonksiyonu için elde edilen sonuçlar	120
Çizelge 3.14. Dokuzuncu test fonksiyonu için elde edilen sonuçlar	121
Çizelge 3.15. Onuncu test fonksiyonu için elde edilen sonuçlar	122
Çizelge 3.16. On birinci test fonksiyonu için elde edilen sonuçlar	123
Çizelge 3.17. On ikinci test fonksiyonu için elde edilen sonuçlar	124
Çizelge 3.18. On üçüncü test fonksiyonu için elde edilen sonuçlar	125
Çizelge 3.19. Basınçlı kap tasarım probleminde elde edilen sonuçların karşılaştırılması	128
Çizelge 3.20. Kaynaklı giriş tasarım probleminde elde edilen sonuçların karşılaştırılması	132
Çizelge 3.21. Çok-amaçlı kaynaklı giriş tasarım probleminde elde edilen sonuçların karşılaştırılması	136
Çizelge 3.22. Çok-amaçlı iki-çubuklu giriş tasarım probleminde elde edilen sonuçların karşılaştırılması	139
Çizelge 3.23. Çok-amaçlı I-kesitli giriş tasarım probleminde elde edilen sonuçların karşılaştırılması	142
Çizelge 4.1. Frekans ağırlık değerleri	149
Çizelge 4.2. Ölçülen titreşimin lokal ağırlıkları ve faktör çarpanları	150

Çizelge 4.3.	ISO 2631`e göre sürüş konfor değerlendirme	153
Çizelge 4.4.	Binek taşıtı B için fiziksel test ile simülasyon arasındaki uyum	163
Çizelge 4.5.	Fiziksel test sonuçları ile simülasyon-tabanlı model sonuçları arasındaki uyum	163
Çizelge 4.6.	Sürüş konfor optimizasyon problemi için bulunan optimum sonuçlar	166
Çizelge 4.7.	Önerilen alternatif yapılar ve maksimum yer değiştirme değerleri	177
Çizelge 4.8.	Mevcut modelde ve alternatif modellerdeki kütle değişim oranı ve maksimum yer değiştirme oranı	181
Çizelge 4.9.	Hacmin deney tasarımı ile oluşturulmuş ANOVA tablosu	183
Çizelge 4.10.	Gerilmenin deney tasarımı ile oluşturulmuş ANOVA tablosu	183
Çizelge 4.11.	Maksimum yer değiştirmenin deney tasarımı ile oluşturulmuş ANOVA tablosu	184
Çizelge 4.12.	Kalıp tasarım optimizasyon problemi için uydurulan cevap fonksiyonlarının R-kare değerleri	186
Çizelge 4.13.	Optimizasyon öncesi ve sonrasında kalıpta oluşan % değişim	188
Çizelge 4.14.	Kuvvet – yer değiştirme değerleri için oluşturulmuş ANOVA tablosu	204
Çizelge 4.15.	Optimizasyon sonucunda elde edilen optimum t ve h değerleri	206

1. GİRİŞ

Bir ürünün uygun tasarım parametreleri ile tasarlanabilmesi ve maliyet performans değerlerinin istenen kriterleri sağlayabilmesi için optimizasyon tekniklerinin yoğun olarak tasarımda kullanılması gerekmektedir. Özellikle taşıt tasarımında son yıllarda artan müşteri memnuniyeti ile birlikte üretimin minimum maliyette ve maksimum kalitede gerçekleştirilmesi beklentisinin karşılanabilmesi açısından yeni tasarımların oluşturulmasında ve ortaya çıkan problemlerin çözümünde optimizasyon yöntemlerinin kullanılma gerekliliği oldukça artmıştır. Günümüz tasarımlarının içerdiği yapısal parametreler ve istenen kriterlerin oluşturduğu modellerin karmaşıklığı nedeniyle klasik tasarım optimizasyon yöntemleri yetersiz kalmaktadır. Bu nedenle, mevcut olan problemlerin daha kısa zamanda daha iyi sonuç verecek şekilde çözülebilmesi için etkin ileri tasarım optimizasyon yöntemlerinin kullanılması gerekmektedir.

Tasarımı gerçekleştiren veya tasarım sürecinde oluşan problemleri çözmeye çalışan araştırmacı karşılaştığı optimizasyon problemlerinin çözümünde güvenilir ve etkin bir şekilde kullanabileceği optimizasyon algoritmasına gereksinim duymaktadır. Ancak kendi alanında uzman bir araştırmacının çalıştığı konudaki karmaşık problemlerin çözümünde gösterdiği çabayı optimizasyon tekniklerinin uygulanmasında da göstermesi beklenemez. Bu nedenle bu tez çalışmasında kullanışlı, global, temel mühendislik eğitimi alan birisinin uygulayabileceği kadar basit ve gerçek optimum değere yakınsamada güvenilir olan aynı zamanda optimizasyon sürecinin başlangıcından bitişine kadar çok az hesaplama zamanı tüketen yeni bir algoritma geliştirmek amaçlanmıştır.

Bu tez çalışmasında, evrimsel algoritmalar arasında yer alan global, güvenilir, kullanışlı ve birçok test probleminden başarılı bir şekilde geçen diferansiyel gelişim algoritması (DE: Differential Evolution) ele alınmış ve popülasyon içindeki en iyi vektörleri fark vektörleri olarak kullanan yeni bir mutasyon stratejisi geliştirilerek geliştirilen algortmada kullanılmıştır (DEBVs: Differential Evolution using the Best Vectors). Geliştirilen DEBVs algoritması ile literatürde mevcut bulunan ve yeni geliştirilen birçok algoritmanın test edilmesinde kullanılan kısıt içeren test problemleri çözdürülmüş ve diğer yöntemlere göre daha düşük sayıda hesaplama ile daha iyi sonuçlar elde edilmiştir. Böylece geliştirilen yeni algoritmanın doğruluğu mevcut sonuçlardan daha

iyi sonuçlara daha hızlı bir şekilde ulaşma imkânı ile kanıtlanmıştır. Ayrıca önerilen yöntem ile literatürde mevcut tek ve çok-amaçlı çeşitli mühendislik problemleri çözdürülmüş ve mevcut sonuçlardan daha iyi sonuçlar elde edilerek geliştirilen yöntemin doğruluğu pekiştirilmiştir.

Geliştirilen yöntem daha önceden klasik ve diğer evrimsel algoritmalarla çözülmüş üç farklı taşıt tasarım optimizasyon probleminde kullanılmış ve daha kısa zamanda daha iyi sonuçlara ulaşılmıştır. Birinci problemde istenen özelliklerde taşıt sürüş konforu için taşıt geliştirme sürecinde kullanılacak sistematik bir yaklaşım geliştirilmiştir. Günümüz global pazar ihtiyaçlarına cevap verebilecek ve rekabet edebilecek ürünlerin otomotiv endüstrisinde üretilmesi için ürün tasarım maliyetlerinin ve pazara çıkış sürelerinin en aza indirgenmesi gerekmektedir. Bunun için mevcut araç geliştirme sürecinde kullanılacak sistematik bir yaklaşımın geliştirilmesi büyük önem arz etmektedir. Araç sürüş konforu için araç geliştirme sürecinde kullanılacak bir yaklaşım geliştirmek için ilk önce konfor matematiksel modeli oluşturulmuş, taşıt sürüş konfor parametreleri tanımlanmış ve simülasyon modeli kurulmuştur. Sürüş konforu için fiziksel testler ile ölçümler gerçekleştirilmiş ve simülasyon sonuçları ile fiziksel test sonuçları karşılaştırılarak simülasyon doğrulanmıştır. Kurulan model için bir arayüz geliştirilmiş ve maksimum konfor için optimizasyon arayüzü oluşturulmuştur. Geliştirilen DEBVs algoritması ile sürüş konfor optimizasyon problemi çözdürülmüştür. Bu çalışma sonucunda geliştirilen bilgisayar destekli analiz ve simülasyon tabanlı bütünlük bir algoritma ile araç geliştirme sürecinde tasarımcının referans olarak kullanabileceği yardımcı bir tasarım aracı geliştirmek amaçlanmıştır.

İkinci olarak taşıt sac üretiminde kullanılan klasik kalıp tasarım sürecinden farklı olarak kalıp tasarım sürecine istenen rijitlikte kalıp tasarımı için simülasyon tabanlı topoloji yaklaşımı entegre edilmiş problem geliştirilen DEBVs algoritması ile çözdürülmüştür. Mevcut kalıpların yer değiştirmeleri ve gerilme değerleri fiziksel testler ile ölçülmüş, sonlu eleman modelleri oluşturularak simülasyon sonuçları elde edilmiştir. Simülasyon sonuçları fiziksel test sonuçları ile doğrulanmış ve topoloji optimizasyonu ile yeni kalıp tasarımları oluşturulmuştur. Daha hafif ve daha rijit kalıp modelleri hazırlanarak yapısal optimizasyon problemi oluşturulup geliştirilen DEBVs algoritması ile çözdürülmüştür. Bu sayede termin ve maliyet konularında kazanımlar elde etmek ve otomotiv

endüstrisinin kalıp bölümlerinde önerilen yöntemin uygulanmasını sağlamak amaçlanmıştır.

Üçüncü olarak istenen özelliklere sahip motor askı takozlarının tasarımı için kullanılacak simülasyon ve fiziksel testler ile optimizasyona dayalı bu konuda tasarım yeteneği kazanılmasına yardımcı olacak bir yapı oluşturulmuş ve oluşan tasarım optimizasyon problemi geliştirilen DEBVs algoritması ile çözdürülmüştür. Motor titreşim takozlarında kullanılan kauçuk malzemelerin hiperelastik malzeme modelleri, yapılan malzeme testleri aracılığıyla belirlenmiş, simülasyonlar sonucunda elde edilen sonuçlar ile gerçek ürün üzerinde yapılan testler arasındaki korelasyon sağlandıktan sonra optimizasyon ile istenen tasarım hedeflerini sağlayacak takoz modeli geliştirilmiştir. Yeni ürünlerin tasarımında prototip üretim sayısının en aza indirgenmesi ve daha kısa zamanda seri üretime geçiş için kullanılacak simülasyon temelinde testlerle bütünlük tasarım süreci tanımlamaları yapılmıştır. İstenen özelliklerde motor askı takoz tasarımı parametrik olarak gerçekleştirilmiş ve geliştirilen DEBVs algoritması ile çözdürülmüştür.

Taşıt tasarımında üç farklı problem geliştirilen DEBVs algoritması ile çözülmüştür. Geliştirilen DEBVs algoritması kullanılarak diğer evrimsel algoritmalara ve klasik optimizasyon yöntemlerine göre daha kısa zamanda çok daha güvenilir, hızlı, global ve etkin sonuçlar elde edilerek literatüre yeni bir yaklaşım kazandırılmıştır. Geliştirilen yöntemin taşıt tasarım sürecinde tasarımcıya yardımcı bir araç olarak kullanılması zaman, maliyet, kalite, hız ve kolaylık açılarından büyük avantajlar sağlaması beklenmektedir.

2. KAYNAK ÖZETLERİ

Bir sistemin istenen özelliklerini en iyi duruma getirme olarak kısaca tanımlanan optimizasyon işleminin mühendisliğin hemen hemen tümünde karşımıza çıkan problemlerin çözümünde kullanımı gün geçtikçe artmaktadır. Literatürde farklı problem türleri için farklı optimizasyon algoritmaları geliştirilmiştir. Genel anlamda optimizasyon algoritmaları türeğe bağılı olan ve türeğe bağılı olmayan, kural tabanlı ve olasılıksal (deterministik ve stokastik) şekilde farklı gruplara ayrılmaktadır. Son yıllarda türeğe bağılı olan ve deterministik algoritmaların lokal optimum noktalara takılabilmeleri, türevleri alınması zor olan veya türevi alınamayan problemleri çözmede yetersiz kalmaları, yapıları gereği adım büyüklüğü ve başlangıç deęer problemlerine sahip olmaları, vb. nedenlerle tercih edilmemektedirler. Bunun yerine global optimum noktayı yakalayabilme, türev gerektirmeme, arama uzayının her bölgesini tarayabilme, tek nokta ile çalışmak yerine tüm çözümleri bünyesinde barındıran bir popülasyon ile çalışabilme, vb. özelliklere sahip olan evrimsel algoritmalar kullanılmaktadır. Doğada gerçekleşen biyolojik süreçleri taklit eden bu algoritmalar özellikle 1960'lerden sonra kullanılmaya başlanmıştır. Bunlar arasında genetik algoritmalar, evrimsel programlama, genetik programlama, evrimsel stratejiler, memetik algoritmalar ve tabu araması sayılabilir. Ancak son günlerde insan ve hayvan davranışlarını taklit eden karınca ve arı kolonileri sürü optimizasyon algoritmaları (ant colony optimization and bees algorithm), kültürel algoritmalar (cultural algorithms), parçalı sürü optimizasyonu algoritmaları (particle swarm optimization) ve diferansiyel gelişim algoritması (DE) geliştirilmiştir.

Biyolojik ortamda yaşanan evrim sürecini yapay ortamlara aktarabilme fikrini ortaya atan ilk kişi John Holland'dır. Holland'ın öğrencisi olan David E. Goldberg ise, o zamana kadar pratik anlamda uygulama alanı bulamamış olan genetik algoritmaların uygulanabilirliğini öne sürmüştür. (Goldberg 1989).

Genetik algoritmalar alanında çeşitli çalışmalar yapan Deb (1999a), genetik algoritmaların çalışma prensiplerini örnek problemlerle açıklamıştır. Coello (2000), genetik algoritmaların yapısından bahsetmiş ve tek amaçlı, kısıtlar içeren mühendislik problemlerinden basınçlı kap, kaynaklı giriş ve basma / çekme yay problemlerini ele almıştır. Geliştirdiği algoritma yardımıyla bu problemleri çözmüş ve sonuçları

literatürdeki sonuçlarla karşılaştırarak sunmuştur. Literatürde o gün mevcut olan sonuçlara göre daha az hesaplama sayılarında daha iyi sonuçlar elde etmiştir.

Deb (1999b), çok-amaçlı optimizasyon için klasik optimizasyon tekniklerinden daha kullanışlı ve verimli arama algoritmalarına ihtiyaç duyulduğunu bildirmiş ve genetik algoritmaların çok-amaçlı optimizasyon problemlerinin çözümünde verimli bir şekilde kullanılabileceğini kaynaklı kiriş test problemini geliştirdiği algoritmayla çözümlenmiştir. Deb ve ark. (2000), çok-amaçlı optimizasyon için helisel yay, dişli çark mekanizması gibi çeşitli mühendislik problemlerini ele almışlar, bu problemler için çoklu pareto-optimal sonuçlar elde etmişler ve tablolar halinde buldukları sonuçları sunmuşlardır. Yaptıkları çalışma sonucunda kullandıkları yöntemin çok-amaçlı karmaşık mühendislik problemlerini çözümlenmede kolay, hesaplama açısından hızlı ve seçici bir yöntem olduğunu göstermişlerdir. Üler ve ark. (1995), elektromanyetik aletlerin tasarım değişkenlerinin optimizasyonunda genetik algoritmaları kullanmışlardır.

Annicchiarico ve Cerrolaza (1999), iki-boyutlu şekil optimizasyon problemlerinin çözümü için sonlu b-spline eğrilerine dayalı genetik algoritmalarla otomatik olarak elemanlara ayırma işlemini gerçekleştiren bir yaklaşım önermişlerdir.

Obayashi ve ark. (2000), çok-amaçlı genetik algoritma kullanarak süpersonik taşıma için bir kanadın şekil optimizasyonu üzerinde durmuşlardır.

Haslinger ve ark. (2000), genetik ve diğer global optimizasyon yöntemlerinin, sayısal yöntemlerle optimum şekil tasarım problemlerini çözümlenmede, ne kadar yardımcı olabileceklerini araştırmışlardır. Kurpati ve ark. (2002), çok-amaçlı genetik algoritma tekniği üzerinde dört farklı değişiklik yaparak elde ettikleri yeni algoritmaların üstünlüklerini daha önceden incelenmiş iki örnek problem üzerinde göstermişlerdir. Buradaki dikkat çekici nokta, klasik “Önce-Amaç-Sonra-Kısıt” yaklaşımını yerine “Önce-Kısıt-Sonra-Amaç” yaklaşımını kullanmalarındadır.

Xiyu ve ark. (2003), karmaşık şekil fonksiyonlarına, genetik algoritmalar ve sinirsel ağlara dayalı yeni bir yüzey oluşturma yöntemi geliştirmişler ve önerilen yöntemi 3-boyutlu nesnelere ve serbest şekil modellerinin Bilgisayar Destekli Tasarım`la (BDT) modellenmesinde kullanmışlardır.

Lampinen (2003), başlangıç tasarımı ve bilgisayar destekli üretim işletim mekanizmalarında ihtiyaç duyulan bilgisayar destekli üretim profillerinin şekil optimizasyonu için bilgisayar destekli yaklaşıma dayalı bir genetik algoritma tanıtmıştır.

Annicchiarico ve Cerrolaza (2004), sonlu elemanlar yöntemi kullanarak şekil optimizasyon problemlerinin çözümüne odaklanmışlar ve sonlu elemanlar modelleri için genetik algoritmaların ve b-spline yüzey modellemenin kullanımı üzerine yoğunlaşmışlardır.

Kegl (2005), statik olarak yüklü yapıların şekil optimizasyonu için bir parametreleştirme yaklaşımı geliştirmiştir.

Akkaram ve ark. (2006), şekil parametre analizi için lineer gerilme analizi üzerinde analitik türevler teknolojisinin (Analytical Derivatives Technology) geçerliliğini araştırmışlardır. Bunun için 2 ile 6 arasında değişen sayıda şekil tasarım parametresine sahip örnek uygulamalar üzerinde mevcut olan en iyi deney tasarımı teknolojileriyle (DOE) analitik türevler teknolojisini karşılaştırmışlardır.

Meske ve ark. (2006), doğal frekans problemleri için parametrik olmayan yeni bir şekil optimizasyonu yöntemi önermişlerdir. Yöntem, hacim kısıtına sahip frekans maksimizasyonu problemleri için sürekli ortamlardan çıkarılan en iyilik ölçütüne dayanmaktadır. Frekans en iyilik ölçütü ise çoklu frekans kısıtına sahip hacim minimizasyon problemlerine genişletilmiştir. Yöntemin öne çıkan en önemli faydası gerekli sonlu elemanlar analizi sayısının tasarım değişkenleri sayısından bağımsız olmasıdır.

Lingyun ve ark. (2005), kafes-kiriş sistemlerinin şekil ve boyut optimizasyon problemlerini etkin şekilde çözen ve erken yakınsamaları önemli ölçüde azaltan bütünleşik bir algoritma geliştirmişlerdir (NHGA: A Niche Hybrid Genetic Algorithm). Bu yeni melez yöntemde “Niche” tekniği çeşitliliği sağlamada kullanılmıştır.

Jang ve ark. (2006), tasarım uzayı optimizasyonunun performansını arttıran yeni stratejiler ve uygulamalar önermişlerdir. Bu öneriler arasında “fixed grid” genel kavramıyla tasarım uzay değişkenini ele almak ve uygun tasarım uzay optimizasyonu

için yapay bir malzeme önermek sayılabilir. Ayrıca gerçek tasarım uzay hassasiyetini elde edebilmek için ayrışık sınır elemanlardan faydalanan yeni bir sonlu eleman modelleme tekniği önermişlerdir. Önerdikleri stratejileri topoloji optimizasyonuna uygulamışlar ve karmaşık, geniş çaplı 3-boyutlu problemleri basitleştirerek kolay ele alınabilen bir hale dönüştürmüşlerdir.

Vafaesefat (2009), oransal B-spline eğrileri ve genetik algoritmalar kullanarak kompozit malzemeli basınçlı kapların şekil optimizasyonu üzerinde durmuş ve basınçlı kabin ağırlığını geliştirdiği algoritma ile minimize etmeye çalışmıştır.

Li ve ark. (2010), uyarlamalı genetik algoritmalar kullanarak kiriş şekillerinin optimizasyonu üzerinde durmuşlardır. Wei ve ark. (2011), frekans kısıtlı kiriş optimizasyon problemini geliştirdikleri paralel genetik algoritma tabanlı şekil ve boyut optimizasyon algoritması ile çözmüşlerdir.

2.1. Diferansiyel Gelişim

Diferansiyel gelişim algoritmasının geliştiricileri olan Price ve Storn'un (1995) dışında algoritmanın performansını ilk olarak 15 test fonksiyonu üzerinde Brutovsky ve ark. (1995) değerlendirmişlerdir. Storn (1995a) geliştirdikleri algoritmayı şifrelenmiş video çözücü teknolojisi üzerinde, sonsuz darbe cevabı filtresi tasarlamakta (Storn 1995b, 1996a), bir uğultu dindirme ünitesi tasarlamakta (Storn 1996b) ve anahtarlı kapasitör filtresi tasarımında (Storn 1996c) kullanmıştır.

Diferansiyel gelişim algoritması üzerinde öncü çalışmalar Joshi ve Sanderson (1996) tarafından yapılmıştır ve onlar diferansiyel gelişim algoritmasını robotik alanda çok sık karşılaşılan çok-sensörlü bir füzyon problemini çözmek için kullanmışlardır.

Diferansiyel gelişim algoritması aerodinamik, havacılık ve uzay araştırmalarında kanatçık, türbin, motor tasarımları gibi uygulamalarda, uçuş kontrol algoritmalarında ve mekik kalkış platformlarında ortaya çıkan optimizasyon problemlerinin çözümünde, tarım ve hayvancılık alanında verimi arttırmak için tasarlanan yapıların uygulanabilirliğinde ortaya çıkan problemlerin çözümünde, kullanılmıştır. Balsa-Canto ve ark. (1998), biyolojik süreçlerin dinamik optimizasyonunda kural tabanlı

optimizasyon algoritmalarının yanında diferansiyel gelişim algoritmasını da etkin bir şekilde kullanmışlardır.

Diferansiyel gelişim algoritmasının potansiyeli anlaşıldıktan sonra birçok araştırmacı çeşitli uygulamalarda bu algoritmayı kullanmaya başlamışlardır. Örneğin akustik ile ilgili diferansiyel gelişim algoritmasını Ainslie ve ark. (2000), sığ sularda deniz dibindeki akustik parametrelerini tahmin etmede, Ganchev ve ark. (2004), metinden bağımsız hoparlör tasarımında, Karasalo ve Skogqvist (2005), iki statik akustik saçılma ile nesne tanımlamada kullanmışlardır.

Ochi ve Cao (2000), bir ormandaki ağaç gelişimini modellemeyi diferansiyel gelişimin örnek bir uygulaması olarak gerçekleştirmişlerdir. Johnson ve ark. (2002), diferansiyel gelişimi kullanarak toprağın fiziksel parametrelerini çıkarmışlardır. Babu ve Angira (2003a), diferansiyel gelişimi su pompalama sistemlerinin optimizasyonunda kullanmışlardır. Banga ve ark. (2003), modern optimizasyon yöntemlerini kullandığı gıda üretim işlemlerinin geliştirilmesinde diferansiyel gelişimi kullanmıştır. Nikolos ve Brintaki (2005), insansız bir hava aracının güzergâh planlamasında diferansiyel gelişimi kullanmışlardır. Beynon (2005), ekonomi alanında kredi derecelendirme probleminde belirsizlikler içeren masraf sınıflandırma usulünün optimizasyonunda diferansiyel gelişimi kullanmıştır. Derksen ve Hancox (2005), endüstriyel bir pnömatik destekleme sistemi optimizasyonunda diferansiyel gelişimi kullanmışlardır. Zelinka (2005), çalışmada diferansiyel gelişimi evrimsel deterministik bir kaos kontrolü ile karşılaştırarak sonuçları irdelemiştir.

Montes ve ark. (2005), kısıt içeren optimizasyon problemlerini çözmek için ceza fonksiyonu kullanmayan ancak çeşitlilik mekanizması içeren bir diferansiyel gelişim algoritması geliştirmişlerdir. Geliştirdikleri algoritmayı, literatürde kısıt içeren optimizasyon problemlerinin çözümünde kullanılan algoritmaların geçerliliğini ölçmek için başvurulan 13 test problemiyle test etmişlerdir. Algoritma; güvenilirlik, kalite ve amaç fonksiyonunun hesaplanma sayısı bakımlarından diğer algoritmalara göre üstünlükler göstermektedir. Geliştirdikleri algoritmanın en çarpıcı iki özelliğinden birisi, yeni nesillerin bir kısmının uygun bölge dışında kalan çözümlerden elde edilmesi, diğeri ise bir aileden birden fazla yeni neslin üretilmesine izin verilmesidir.

Lorenzoni ve ark. (2006), port operasyonları için çok-modlu kaynak kısıtlı zamanlama probleminde diferansiyel gelişimi kullanmışlardır. Ali ve ark. (2006), atomik ve moleküler küme yapılarının klasik potansiyel kullanılarak gerçekleştirilmiş optimizasyon işleminde diferansiyel gelişimi kullanmışlardır. Yan ve ark (2006), diferansiyel gelişimde kontrol parametrelerinin atamasında uyarlamalı bir sistem geliştirerek bu algoritmayı kritik üstü sularda oksidasyon işlemindeki kinetik parametrelerin çıkarımında kullanmışlardır. Pavlidis ve ark. (2007), ekonomik coğrafyada kısa dönem dengelerin varlığındaki ve hesap yöntemlerindeki optimizasyon problemlerinin çözümünde diferansiyel gelişimi kullanmışlardır. Cannon (2007), iklim değişikliklerinin tahmininde lineer olmayan analog bir öngörü analizinde diferansiyel gelişim algoritmasını kullanmıştır.

Otomotiv alanında, ilk çalışmalar Kyprianou ve ark. (2000) tarafından bir hidrolik motor takozu modelinde parametrelerin belirlenmesinde, Pedchote ve Purdy (2003), bir orta sınıf aile taşıtında tek bir lastiğin parametrelerini ortaya çıkarmada, diferansiyel gelişimi kullanmışlardır. Kim ve ark. (2003), hızlı trenlerde kullanılan süspansiyon sistemlerinin tasarım optimizasyonunda sinirsel ağların öğrenme sürecinde ortaya çıkan çok-amaçlı optimizasyon problemini çözmeye diferansiyel gelişimi kullanmışlardır. Farina ve ark. (2005), içten yanmalı bir dizel motorun parametrik modellerinin evrimsel yöntemlerle çözümünde diferansiyel gelişim algoritmasını kullanmışlardır. Zhang ve ark. (2009), diferansiyel gelişim algoritmasını değiştirilebilir akülere sahip olan bir hibrit elektrikli aracın kontrol stratejisi için ortaya konan global optimizasyon problemini çözmeye kullanmışlardır.

Mingyong ve Erbao (2010), araç doğrultu kontrolü problemini geliştirdikleri yeni mutasyon operatörü ile çalışan diferansiyel gelişim algoritması ile çözmüşlerdir. Ayrıca kullandıkları algoritmada çaprazlama oranını kendi kendine uyarlamalı bir yaklaşımla belirlemişlerdir.

Riley ve ark. (2011), optimizasyon algoritması olarak diferansiyel gelişimin bir varyantını kullanarak çok-amaçlı şekil optimizasyon problemlerinin çözümü üzerine yoğunlaşmışlar ve geliştirdikleri algoritmayı sonlu elaman tabanlı mühendislik paket programları kullanarak üç örnek mühendislik problemini çözmüşlerdir.

2.2. Emin Tasarım (Robust Design)

Gunawan ve Azarm (2005), hassasiyet alanı kavramını kullanıp bir tasarımın çok-amaçlı sağlamlığını ölçen ve bu ölçümü çok-amaçlı optimizasyon problemlerinin emin Pareto çözümlerini elde etmede kullanan bir yöntem geliştirmişlerdir.

Dhavlikar ve ark. (2003), puntasız taşlama işleminde iş parçalarının yuvarlaklık hatalarını minimize etmek için emin (robust) şartları belirlemede kullanılan Taguchi ve “dual response” metodolojilerini birleştiren yeni bir yöntem tanıtmışlardır.

Ghani ve ark. (2004), parmak frezeleme işlemindeki kesme parametrelerinin optimizasyonu için Taguchi optimizasyon metodolojisini kullanmışlardır.

Yang ve ark. (1998), Taguchi yöntemini tornalama işleminde kesme parametrelerinin optimum değerlerini bulmada kullanmışlardır.

Liao ve ark. (2002), Taguchi yöntemindeki çok-cevaplı problemi etkin biçimde optimize etmek için sıralama yaklaşımına dayalı veri zarflama analizi tekniğini kullanmışlardır.

2.3. Sürüş Konfor Optimizasyonu

Sübjektif ve objektif, sayısal ve deneysel olmak üzere sürüş konfor özelliklerinin iyileştirilmesi üzerine yapılmış birçok çalışma mevcuttur.

Nishiyama ve ark. (2000), koltuğun, direksiyonun ve pedalların konumuna bağlı olarak değişen bir fonksiyonun sürüş konforu üzerine olan etkilerini incelemişlerdir.

Demic ve Lukic (2002), sürüş konfor iyileştirmesi için bir kriter geliştirmeye çalışmışlar ve sürücüyeye etki eden titreşimler için kullanılacak bir fonksiyon tanımlamışlardır.

Eriksson ve Arora (2002), beş farklı global optimizasyon yöntemini sürüş konfor optimizasyon probleminin çözümünde kullanarak yöntemlerin performanslarını değerlendirmişlerdir.

Rauh (2003), binek taşıtların konfor ve yol tutuş karakteristikleri için simülasyon modellerini incelemiş ve her bir modelin sınırlamalarını araştırmıştır.

Niekerk ve ark. (2003), koltukların sürüş konfor karakteristiklerini incelemiş ve bu karakteristikleri doğru bir şekilde öngörebilmek için bir yöntem üzerinde çalışmışlardır.

Els ve ark. (2006), bir arazi taşıtı üzerinde sürüş konforu ve yol tutuşu açısından yay-damper karakteristiklerinin optimum değerlerini bulmak için geliştirilen eniyileme yöntemlerini araştırmışlardır.

Lee ve Moon (2006), bir taşıtın dinamik karakteristiklerini belirlemede kullanılabilecek yeni bir matematiksel model önermişlerdir.

Els ve ark. (2007), arazi taşıtlarında sürüş konforu ve yol tutuşunun yay ve damper karakteristiklerinin değişimi ile nasıl değiştiğini araştırmışlar ve çalışmalarında yarı-aktif süspansiyon sistemi kullanmışlardır. Yol tutuşunu düşürmeden sürüş konforunu arttırmak için optimum tasarım uzayını araştırmışlardır.

Pennati ve ark. (2009), arazi taşıtları için sürüş konforunu objektif olarak hesaplayabilecek yeni bir aygıt geliştirmeye çalışmışlardır.

Yang ve ark. (2009), tandem süspansiyon düzenine sahip bir traktörün sürüş konforunu hesaplamada kullanılan sistematik bir yöntem üzerinde çalışmışlardır.

2.4. Sürüş Konforu ile ilgili Standartlar

Araçlar üzerinde yolculuk eden sürücü ve yolcuları ilgilendiren sürüş konforu ölçüm ve hesaplama yöntemleri standartlaştırılmıştır. Bu ölçüm ve hesaplamalar ivme değerlerini kullanmaktadır ve ivmeölçerler ile önceden belirlenen araç bölgelerinden ölçülen ivme değerleri kullanılarak sürüş konfor indeksleri hesaplanmaktadır.

Sürüş konforu ile ilgili yapılan çalışmalarda en çok kullanılan standart Uluslararası Standartlar Teşkilatı'nın (ISO) 1997'de yayınladığı 2631 standardıdır (ISO 2631 1997). Bu standart çalışma kapsamında satın alınmış ve içeriği sürüş konfor hesaplamalarında kullanılmıştır.

Sürüş konforu ile ilgili kullanılan ve özellikle insan üzerine etki eden titreşimler üzerinde duran diğer bir standart İngiliz standardıdır (British Standard 6841 1987).

Otomotiv Mühendisleri Derneği (SAE), ağır ticari araçlardaki titreşimleri ölçmede kullanılan bir standart yayınlamıştır (SAE J1490 2011). SAE ayrıca iş makinesi operatörleri için operatör üzerinde oluşan titreşimleri ölçmede kullanılan bir standardı 1992 yılında yayınlamış ancak bu standardı 2007 yılında yürürlükten kaldırmıştır (SAE J1013 1992).

Yukarıda bahsedilen standartların dışında ayrıca taşıt üretici firmalarının kendi oluşturdukları standartlar ve prosedürler mevcuttur. Bu prosedürler genellikle taşıtların farklı yol şartlarında gerçekleştirilen testleri ile ilgilidir.

Tüm standartlar farklı eksenlerdeki ivme değerlerinden hesaplanan ortalama karekök değerini (Root Mean Square, RMS) kullanmaktadırlar. Ayrıca frekansa bağlı ağırlıklandırma ve eksen katsayısı veya eksen çarpan faktörleri de standartlar içinde tanımlanmıştır. Araçlar üzerindeki farklı bölgelerden elde edilen RMS değerlerinden Sürüş İndeksleri (Ride Index, RI) hesaplanarak konfor değerleri ile karşılaştırılmaktadır.

2.5. Kalıp Tasarım Optimizasyonu

Konu ile ilgili kaynak araştırması yapıldığında kalıp konusu ile ilgili literatürde çok sayıda kaynak yer almasına rağmen çalışma konusu olan sac metal kalıp tasarım optimizasyonu ile ilgili çok az sayıda çalışmaya rastlanmıştır ve bu kapsamda detaylı bir çalışmanın eksikliği gözlenmiştir. Yapılan çalışmalar genellikle kalıpların deformasyonu ve sac metal parçaların geri yaylanma özellikleri ile ilgilidir. Sonlu eleman tabanlı simülasyon yaklaşımları kalıpların ve ilgili parçaların rijit olduğu varsayımı üzerine kurulmuş modeller ile çözülmektedir.

Becchio ve ark. (1998), simetrik olarak derin çekme ile şekillendirilen bir otomobil kapısının kalıbını incelemişler ve sac parçanın orta bölgesinde bir temas problemi gözlemlemişler, bunun nedeni olarak alt kalıbın fazla esnediğini belirlemişlerdir. Yaptıkları çalışmalarda derin çekme simülasyonu ile kalıp yüzeyleri üzerine gelen basınç dağılımını belirlemişlerdir. Erkek kalıp, dişi kalıp ve pot çemberi sonlu elemanlar modelleri üzerinde analizler yapmışlar, alt kalıbın düşük rijitlikte olduğunu analizler sonucu görmüşlerdir. Tasarım çalışmaları sonucunda alt kalıbın orta bölümüne yakın olan boşlukları feder geometrisi ile güçlendirmişler ve maksimum yer

değiştirmeyi %40 oranında azaltmışlardır. İkinci bir çalışma olarak kalıp gövdesinin topoloji optimizasyonu ile yeniden tasarımını yapmışlar ve feder kalınlıklarının aktif yüzeylerin altında artacak şekilde tasarlayarak deformasyonları azaltmışlardır.

Balendra ve ark. (2000), makine ve kalıp elastikiyetinin parça geometrisi üzerindeki etkisini araştırmışlar ve ekstrüzyon kalıpları üzerinde sonlu elemanlar yöntemi (FEM) ile analizleri yapmışlardır. Erkek kalıp üzerine yükleme ve boşaltma simülasyonları ile parça ve kalıp deformasyonlarını hesaplamışlar ve orijinal CAD modeli ile deforme olmuş kalıp modeli formlarını geliştirdikleri yazılım ile karşılaştırmışlardır.

Chodnikiewicz ve Balendrab (2000), kalıp üzerinde oluşan öteleme ve dönme deformasyonlarını yer değiştirme dönüştürücüsü (transducer) ile ölçmüşler ve bunun için 6 dönüştürücü kullanmışlardır (3 yatay, 3 düşey yer değiştirme). Presler için yatay ve düşey kuvvetlerin etkin olduğunu belirlemişlerdir. Hidrolik ve mekanik presler üzerinde yapılan ölçümler ile esneklik matrisleri tanımlamışlar ve yükleme-boşaltma işlemlerinde preslerin performansları için şekilsel matris tanımlamaları yapmışlardır.

Tekkaya (2000), sac metal şekillendirmede kullanılan sayısal hesaplamalar ve simülasyonlar üzerinde durarak mevcut gelişmeleri araştırmıştır.

Rosochowski (2001), soğuk şekillendirme işlemlerinde kalıp esnemesi ve geri yaylanmadan dolayı oluşan deformasyonları incelemiş ve bu deformasyonları gidermeye çalışmıştır. Bu amaçla kalıp yüzeyinin CAD modelini değiştirerek çeşitli denemeler yapmıştır.

Pozo ve ark. (2006), kalıp alıştırmaya ve teslim zamanlarının azaltılması üzerinde durmuşlar ve sonlu elemanlar yöntemi ile pres parçalarının analizini yaparak kalıba olan etkileri incelemişlerdir. Kalıbın özellikle orta bölümünde maksimum deformasyon olduğunu gözlemlemişlerdir.

Conle ve Wang (2007), yaptıkları çalışmada kalıpların kritik noktalarında birim yer değiştirmeleri ölçmüşler ve simülasyon sonuçları ile korelasyon sağlamaya çalışmışlardır. Ölçümleri radyo kontrollü kablosuz veri toplama cihazı ile gerçekleştirmişler ve 22 adet uzama ölçeri pot çemberi üzerinde oluşan yüksek gerilme bölgelerine yerleştirmişlerdir. Yaptıkları çalışmada daha çok ölçüm sırasında ortaya

çıkan problemleri incelemişlerdir. Yaptıkları çalışmanın sonucunda deformasyon ölçümleri için uygun ve yeterli sayıda uzama ölçer kullanılması gerektiğini vurgulamışlar, ancak uzama ölçer sayısının azlığının veri toplama açısından avantaj sağladığını ve bunun konumlama problemleri oluşturabildiğini belirtmişlerdir.

Pozo ve ark. (2007), kalıp deneme sürelerinin azaltılması için metodolojik bir çalışma gerçekleştirmişlerdir. Presi (yastık ve tij milleri) ve kalıbı (erkek kalıp, dişi kalıp, pot çemberi) modelleyerek sonlu elemanlar ile analiz ederek temas yüzeylerinin deformasyon haritasını çıkarmışlardır. Deformasyon değerlerinin kalıp içindeki federlerin konumuna göre değiştiğini gözlemlemişlerdir. Bilgisayar destekli üretim (CAM) ile takım geometrisi üzerinde düzeltme yaparak kalıp yüzeylerini daha hassas işleyerek deformasyonları düşürmeye çalışmışlardır.

Simülasyonları yüksek hassasiyetle doğrulayan çalışmaların yanında özellikle otomotiv sektörü için taşıt geliştirme süreçleri üzerine olan önemli etkilerinden dolayı üretilebilirlik ve maliyet konularına odaklanan çalışmalar da son yıllarda önem kazanmıştır. Araştırmacılar ayrıca gerilme ve yer değiştirme değerlerini arttırmadan kalıpların ağırlıklarını düşüren çalışmalar üzerinde durmaktadırlar. Bu noktada kalıp şekillerinin değiştirilmesi kritik bir rol oynamaktadır. Bu nedenle yapısal optimizasyon tekniklerini kullanma gerekliliği gittikçe artan bir öneme sahip olmaktadır.

Kim ve ark. (2001), kalıpların şekillerini hassasiyet analizleri ile belirlemeye çalışırken sac metal şekillendirme işleminde temelde Lagrange denklemlerine dayanan optimizasyon işlemini kullanmışlardır. İş parçası ve rijit kalıp arasındaki temas tanımlamalarını değiştirilmiş Coulomb sürtünme modeli tabanlı ceza yöntemi ile yapmışlardır. Kalıp şekil tasarım optimizasyon problemini geri yaylanmayı minimize edecek şekilde tasarlanan sac metal parçası kullanarak çözümler ve önerdikleri yöntemin uygulanabilirliğini bu örnek ile göstermişlerdir.

Kok ve Stander (1999), çok-noktalı yaklaşım tabanlı bir optimizasyon yöntemi önermişler ve bunu bir sac metal şekillendirme işlemine uygulamışlardır. Yöntemi ayrıca bir otomotiv parçası üretiminde kalıp şeklini ön şekillendirme ile optimum tasarlayabilmek amacıyla uygulamışlardır. Dört şekil değişkeni tanımlayarak ağırlığı yaklaşık olarak %9,4 oranında hafifletmişlerdir.

Sun ve ark. (2011), sac metal şekillendirme tasarımında yırtılma ve buruşma problemlerini önlemek amacıyla çok-amaçlı parçacık sürü optimizasyon yöntemini kullanarak problemi çözmeye çalışmışlardır. Liew ve ark. (2004), geri yaylanma minimizasyon tasarım problemini çözmek amacıyla bir evrimsel algoritma önermişlerdir. Bu konuda Hou ve ark. (2010), sonlu elemanlar yöntemini ve cevap yüzey yaklaşımını kullanan bir stokastik optimizasyon yaklaşımı önermişlerdir.

2.6. Motor Titreşim Takoğu Tasarım Optimizasyonu

Hiperelastik malzemelerle ilgili literatürde çeşitli çalışmalar olmakla birlikte, motor takoğu tasarım ve optimizasyonu konusunda yapılan çalışma sayısı çok fazla değildir. Son yıllarda simülasyon tabanlı motor askı takozlarının optimum tasarımı alanında yapılan çalışmaların sayısında artış gözlenmektedir. Ishihama (2003) birçok otomotiv firmasının ürettiği taşıtların gürültü ve titreşim değerleri için farklı yöntemler geliştirdiğini belirtmiş ve bunun karşılıklı kullanılabilir olacak sistematik bir yaklaşımın geliştirilmesi önündeki en büyük engel olduğunu vurgulamıştır.

Shoureshi ve Knurek (1996) otomotiv endüstrisinde ve taşıma sistemlerinde oluşan gürültü ve titreşim alanındaki 12 yıllık araştırma ve geliştirme sürecinde buldukları sonuçları yaptıkları çalışmalarda sunmuşlar ve yaptıkları araştırmadan sonra gürültü ve titreşim karakteristiklerinin tanımı üzerinde bir takım sanal ve deneysel testler önermişlerdir.

Kim ve Kim (1997), Mooney-Rivlin malzeme modelini kullanarak istenen rijitlik eğrisine sahip motor takozunun şekil optimizasyonu üzerinde çalışmışlardır.

Sakai ve ark. (2001), bir mini sedan aracın titreşim-gürültü tabanlı tam taşıt modelini geliştirmişler ve motor kaynaklı titreşim ve gürültüyü değerlendirmek için yapılan mühendislik analiz prosedürleri üzerinde durmuşlardır. Bunun için hem sonlu eleman modellerini kullanmışlar hem de testler yapmışlardır.

Yu ve ark. (2001), pasif hidrolik motor takozları ve aktif motor takozları üzerinde durmuşlar ve bu sistemler üzerinde optimizasyon çalışmaları yapmışlardır.

Lee ve Hwang (2003), bir binek tařıtında belirli devirlerde ortaya ıkan gürültüyü minimize etmek için tasarım hassasiyet tabanlı bir optimizasyon algoritması kullanarak motor takozu problemini özmeye alıřmıřlardır.

Rao (2003), binek tařıtlardaki ve hava aralarındaki gürültü ve titreřimi kontrol edebilmek amacıyla viskoelastik malzemeler kullanan bir pasif sönümlleme teknolođisi üzerinde alıřmıřtır.

Song ve ark. (2003), piezoelektrik iřleticiler ve piezoelektrik sensörler kullanan bir aktif titreřim kontrol sistemi geliřtirmıřler ve bir tařıtın üç boyutlu kabin modeli üzerine bu sistemi yerleřtirerek analiz etmiřlerdir.

Kim ve ark. (2004), lineer olmayan sonlu elemanlar analizi için gerekli kauuk malzeme özelliklerinin tespiti için deneysel test yöntemleri üzerinde alıřmıřlardır.

Lee ve Youn (2004), motor takozuna statik ve dinamik davranıřlar için topoloji optimizasyonu uygulamıřlardır. Optimizasyon algoritmaları olarak yoğunluk dađılım yaklařımını ve sıralı dođrusal programlamayı (sequentially linear programming) kullanmıřlardır. Önerdikleri yaklařımı dođrulamak için içinde motor takozu da bulunan bazı tasarım örneklerini özmüřlerdir.

3. MATERYAL VE YÖNTEM

Optimizasyon problemleri mühendisliğin ve diğer uygulamaların hemen hemen tümünde karşımıza çıkan problemlerden birisidir. Örneğin; bir otomobilde fren diski hangi kalınlık ve şekilde olmalı ki en üst seviyede frenleme kapasitesi artsın? Bir motor takozunun özellikleri ne olmalı ki araçtaki sürücü ve yolcular motordan ve yoldan gelen titreşim ve gürültüleri en minimum seviyede hissetsin? Otomobilin dış tasarımını nasıl olmalı ki aracın iç hacmi geniş ve ferah olurken dış hacim rüzgar direnci bakımından en az sürtünme kayıplarına sahip olsun? Helikopter pervaneleri hangi kesitte olsun ki hava direncinden en üst seviyede yararlanabilsin? Testler sonucunda elde ettiğimiz değerlerin yerini alacak eğri ne olsun ki en uyumlu şekilde elde ettiğimiz verilerin yerine kullanılabilirsin? Bir bilgisayar ağında ağ bağlantıları ne şekilde kurulsun ki en az güç kaybıyla maksimum seviyede bilgi taşınabilsin? Hiç kuşku yok ki birçok araştırmacı kendi çalışma konularında karşılaştıkları problemlerin çözümünde güvenilir ve etkin bir şekilde kullanabilecekleri bir optimizasyon algoritmasına ihtiyaç duymaktadırlar (Arora 2004, Price ve ark. 2005).

Kullanışlı, global ve tutarlı bir optimizasyon yönteminden; temel mühendislik eğitimi alan birisinin uygulayabileceği bir düzeyde olması, gerçek optimum değere yakınsamada güvenilir olması, her yapılan çözümde birbirine yakın sonuçları vermesi ve aynı zamanda optimizasyon sürecinin başlangıcından bitişine kadar çok az hesaplama zamanı tüketmesi beklenmektedir.

Dolayısıyla etkin bir optimizasyon algoritmasından temelde şu özellikleri taşıması beklenmektedir:

- Uygulanması basit,
- Kullanımı kolay,
- Güvenilir,
- Hızlı (Price ve ark. 2005).

Diferansiyel gelişim algoritması (DE) bu özellikleri kendinde barındıran en iyi yöntemler arasında ayrıcalıklı bir yere sahiptir. DE ilk defa 1994 yılında Kenneth Price tarafından genetik tavlama (genetic annealing) algoritmasından esinlenilerek geliştirilmiştir (Price ve ark. 2005). Profesör Jouni Lampinen 1998 yılında DE'yi incelemeye başlamış ve makine mühendisliği için DE'nin çok verimli bir araç olduğunu örneklerle göstermiş, ayrıca kısıtlı ve çok-amaçlı optimizasyon problemlerinin çözümünde kullanılabilmesi için DE'nin gelişmesinde katkıda bulunmuştur.

En basit ifadelerle optimizasyon, bir sistemin istenilen özelliklerini maksimize ederken istenmeyen özelliklerini minimize etme girişimidir. Bu özelliklerin ne olduğu ve nasıl verimli bir şekilde geliştirilebildiği problemin yapısına bağlı olarak değişebilir. Örneğin taşıt tasarımı aşamasında süspansiyon sisteminin yumuşaklığını veya sertliğini yol şartlarına göre en maksimum düzeyde uyumlu olacak şekilde yani yoldaki pürüzlükleri en minimum şekilde gövdeye yansıtması için tasarımın oluşturulması girişimi bir optimizasyon problemi olarak ele alınabilir. Burada amaç maksimum sürüş konforudur. Basit bir şekilde matematiksel olarak maksimize edilecek özellik olan konfor (f); yay özellikleri (y), amortisör özellikleri (a) ve yol pürüzlülüğünün (p) bir fonksiyonu olarak ifade edilirse;

$$\text{Amaç Fonksiyonu} = f(y, a, p) \quad (3.1)$$

şeklinde bir denklem kurulabilir. Bu fonksiyonların en ekstremum değerleri optimizasyon hedefini karşıladıkları için Denklem 3.1'deki gibi fonksiyonlar *amaç fonksiyonu* olarak adlandırılırlar. Eğer denklemin minimumu araştırılıyorsa amaç fonksiyonu genellikle *maliyet fonksiyonu* olarak adlandırılır. Tam tersi eğer örnekte olduğu gibi amaç fonksiyonunu maksimum yapan değerler araştırılıyorsa amaç fonksiyonu sıklıkla *uygunluk fonksiyonu* olarak adlandırılmaktadır. Özel bir durum olarak eğer amaç fonksiyonunu sıfır yapan değerler araştırılıyorsa bu sefer amaç fonksiyonu *hata fonksiyonu* olarak adlandırılmaktadır. Amaç fonksiyonunun işareti değiştirilerek maksimumu bulma problemi minimumu bulma problemine, minimumu bulma problemi de maksimumu bulma problemine rahatlıkla dönüştürülebilir.

Sürüş konforu örneğinde 3 tane birbirinden bağımsız değişken mevcuttur. Kompleks problemlerde genellikle komplekslik arttıkça değişken sayısı da artmaktadır. En genel anlamda amaç fonksiyonu,

$$f(\mathbf{x}) = f(x_0, x_1, \dots, x_{D-1}) \quad (3.2)$$

şeklinde optimize edilecek özelliği etkileyen D parametreye sahiptir. Amaç fonksiyonlarını sadece parametre sayısına bağlı olarak sınıflandırmak yetersiz olur. Çeşitli ölçütlere göre amaç fonksiyonları sınıflandırılabilir, bu ölçütlerden bazıları şunlardır (Arora 2004, Price ve ark. 2005, Rao 2009):

- *Parametre niteliği:* Amaç fonksiyonundaki parametrelerin sürekli, ayrık, sembolik veya sadece belirli değerleri alabilmesi ile alakalı olan bir ölçüttür.
- *Parametre bağımsızlığı:* Amaç fonksiyonundaki her bir parametre değişiminin diğer parametreleri etkileyip etkilemediği ile alakalı olan bir ölçüttür. Optimizasyon güvenilirliği için parametrelerin birbirini etkilememesi yani bağımsız olması önem arz etmektedir.
- *Boyut, D :* Amaç fonksiyonundaki parametre sayısı ile ilgili olan ölçüttür.
- *Modal özellik:* Amaç fonksiyonunun sadece bir tane ekstremuma mı yoksa birden fazla sayıda ekstremuma sahip olup olmamasıyla alakalı bir ölçüttür.
- *Zamana bağlılık:* Optimum noktanın zamandan bağımsız yani sabit (statik) veya zamana göre değişen yani dinamik olup olmadığı ile alakalı olan ölçüttür.
- *Gürültü:* Üzerinde araştırma yapılan fonksiyonun ekstremum noktasının kendisini tam olarak belli edip etmemesi ve her bir değerlendirme sürecinde aynı sonucu verip vermemesi ile alakalı bir ölçüttür.
- *Kısıtlar:* Fonksiyonun kısıtlara sahip olup olmaması ile alakalı bir ölçüttür. Kısıtlar; eşitlik kısıtları, eşitsizlik kısıtları ve sınır kısıtları olmak üzere üç çeşitte incelenebilir.
- *Türev alma:* Amaç fonksiyonunun her noktasında türevin tanımlı olup olmaması ile ilgili olan bir ölçüttür.

Sürüş konforu optimizasyonu örneğinde parametrelerin hepsi sürekli ve birbirinden bağımsız parametrelerdir. Fonksiyonda üç tane parametre bulunduğu için amaç fonksiyonunun boyutu 3'tür. Sürüş konforu, yol pürüzlülüğü, yay ve amortisör

özelliklerine göre değişip birden fazla optimum noktaya sahip olabilir. Ayrıca yol şartları zamana göre değiştiğinden aranan optimum nokta da zamana göre değişebilmektedir. Bu örnekte gürültü faktörleri bulunabilir. Eğer araç çok rijit ve sabit olmayan çok uzun bir köprüden geçiyorsa köprünün diğer şeridinden geçen ağır yüklü kamyonun veya trenin oluşturduğu titreşimler sürüş konforunu etkileyebilir. Başka bir durum olarak yüksek hızlarda seyir halinde tek şeritli bir yolda yandan geçen ağır araçların da yakın geçişi esnasında ortaya çıkan anlık basınç düşmesi nedeniyle sürüş konforuna gürültü faktörü oluşturup olumsuz yönde etki edebilir. Parametrelerin alt ve üst limitleri kısıt olabileceği gibi yay ve amortisör özelliklerinde boyut ve malzeme özelliklerinin neden olduğu bazı kısıtlamalar olabilir. Amaç fonksiyonunun türevi, bu basit denklemde yol pürüzlülük değerinin sıfır olmadığı her durumda mevcut iken problemin daha detaylı incelenmek istenmesi halinde yay ve amortisör özelliklerinden veya kısıtlardan kaynaklanabilecek herhangi bir komplekslik veya süreksizlik türevin mevcudiyetini değiştirir. Hemen hemen tüm gerçek mühendislik problemleri türev alınması zor, lineer olmayan, birden fazla optimum noktaya sahip amaç fonksiyonlarına sahip olup birçok kısıt içermektedirler. Hatta sadece bir tane amaç fonksiyonu problemi ifade etmede yetersiz kalmakta birden çok amaç fonksiyonuna sahip çok-amaçlı fonksiyonlar kullanılmaktadır. Ayrıca bu fonksiyonlar hem sürekli hem de süreksiz birçok parametre içermektedirler.

3.1. Optimizasyon Algoritmalarının Sınıflandırılması

Çözümü için uğraşılan bir optimizasyon problemi matematiksel olarak en genel anlamda bir minimizasyon problemine dönüştürüldükten sonra önemli bir adım, bu problemi çözecek uygun bir optimizasyon algoritmasının seçilmesidir. Literatürde farklı problem türleri için farklı algoritmalar geliştirilmiştir. Ancak en genel ifade ile algoritmalar problemin yapısına bağlı olarak, problemin arama algoritmasının tek tek veya aynı anda birden fazla noktada arama yapabilmesine bağlı olarak sınıflandırılabilmesi gibi amaç fonksiyonunun türevinin mevcut olup olmamasına bağlı olarak da sınıflandırılabilir (Çizelge 3.1). Ayrıca optimizasyon algoritmaları kural tabanlı (deterministik), olasılıksal (stokastik) ve hibrit (hem kural tabanlı hem olasılıksal) biçiminde üç farklı türde de incelenebilir.

Çizelge 3.1. Optimizasyon algoritmalarının sınıflandırılması (Price ve ark. 2005)

ALGORİTMALAR	Tek-noktalı	Çok-noktalı
Türeve Bağlı	<ul style="list-style-type: none"> • Dik-İniş Yöntemi • Conjugate Gradyan Yöntemi • Quasi Newton Yöntemi 	<ul style="list-style-type: none"> • Çok-noktadan Başlayan Algoritmalar • Demetleme Algoritmaları
Türevden Bağımsız	<ul style="list-style-type: none"> • Rastlantısal Yürüyüş Yöntemi • Hooke-Jeeves Yöntemi 	<ul style="list-style-type: none"> • Nelder-Mead Yöntemi • Evrimsel Algoritmalar (Genetik Algoritma, vb.) • Diferansiyel Gelişim Algoritması

Çizelge 3.1`de verilen sınıflandırma bir genelleştirme değildir ve tüm optimizasyon algoritmaları için geçerli olmayabilir. Örneğin benzetimli tavlama yöntemi türevden bağımsız olarak hem tek-noktalı hem de çok-noktalı olarak ele alınabilir. Burada diferansiyel gelişim algoritmasının diğer evrimsel algoritmalar gibi türevden bağımsız ve çok-noktalı bir yapıya sahip olduğuna dikkat etmek gerekir. Bu aşamada diferansiyel gelişim algoritmasının farklılığını daha iyi algılayabilmek ve gelişme aşamasında hangi yöntemlerin olumlu özelliklerinden faydalandığını göstermek açısından Çizelge 3.1`de belirtilen yöntemlere kısaca değinmek yerinde olacaktır.

3.2. Tek-Noktalı Türeve Bağlı Optimizasyon Yöntemleri

En temel klasik optimizasyon yöntemlerinden olan türeve bağlı teknikleri araştırmadan önce temel bazı ifadelerin bilinmesi gereklidir. Buna göre; D -boyutlu bir parametre vektörü şu şekilde tanımlanır (Arora 2004):

$$\mathbf{x} = \begin{pmatrix} x_0 \\ x_1 \\ \dots \\ x_{D-1} \end{pmatrix} = (x_0 \ x_1 \ \dots \ x_{D-1})^T \quad (3.3)$$

Burada; italik karakterde ve küçük harfler ile yazılan parametreler *tekil parametreleri* ifade ederken, koyu karakterde ve küçük harf ile yazılan ifade bir **vektörü** temsil etmektedir. Bunların dışında koyu ve büyük harf ile yazılan ifade ise bir **MATRİSİ** temsil etmektedir. Bazı özel operatörler ise formülasyonları basitleştirmek için klasik

optimizasyon yöntemlerinde sıklıkla kullanılmaktadırlar. Bunlardan bir tanesi değişkenlerin birinci dereceden türevlerini almaya yarayan gradyan vektörünü oluşturmada kullanılan *nabla* operatörüdür:

$$\nabla = \begin{pmatrix} \partial/\partial x_0 \\ \partial/\partial x_1 \\ \dots \\ \partial/\partial x_{D-1} \end{pmatrix} \quad (3.4)$$

Gradyan vektörü ise;

$$\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x}) = \begin{pmatrix} \frac{\partial f(\mathbf{x})}{\partial x_0} \\ \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \dots \\ \frac{\partial f(\mathbf{x})}{\partial x_{D-1}} \end{pmatrix} \quad (3.5)$$

şeklinde ifade edilmektedir. Bu ifadeden yararlanarak bir fonksiyonun ikinci-dereceden kısmi türevlerini ifade eden *Hessian* kare matrisi şu şekilde tanımlanır (Arora 2004):

$$\mathbf{G}(\mathbf{x}) = \nabla^2 f(\mathbf{x}) = \begin{pmatrix} \frac{\partial f(\mathbf{x})}{\partial x_0 \partial x_0} & \frac{\partial f(\mathbf{x})}{\partial x_0 \partial x_1} & \dots & \frac{\partial f(\mathbf{x})}{\partial x_0 \partial x_{D-1}} \\ \frac{\partial f(\mathbf{x})}{\partial x_1 \partial x_0} & & & \dots \\ \dots & & & \\ \frac{\partial f(\mathbf{x})}{\partial x_{D-1} \partial x_0} & \dots & \dots & \frac{\partial f(\mathbf{x})}{\partial x_{D-1} \partial x_{D-1}} \end{pmatrix} \quad (3.6)$$

Burada; ∇^2 sembolünün ikinci-dereceden kısmi türevi ifade ettiğine ve *nabla* operatörüne dikkat edilmelidir.

Bu tanımlamalardan sonra bir \mathbf{x}_0 noktası etrafında $f(\mathbf{x})$ amaç fonksiyonu Taylor serisine açılarak aşağıdaki denklem elde edilir:

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \frac{\nabla f(\mathbf{x}_0)}{1!} (\mathbf{x} - \mathbf{x}_0) + \frac{\nabla^2 f(\mathbf{x}_0)}{2!} (\mathbf{x} - \mathbf{x}_0)^2 + \dots \quad (3.7)$$

Temel matematik bilgisinden hatırlanacağı üzere bir noktanın ekstremum olması için türevinin sıfıra eşit olması gerekir:

$$\mathbf{g}(\mathbf{x}_{ekst}) = \nabla f(\mathbf{x}_{ekst}) = f'(\mathbf{x}_{ekst}) = 0 \quad (3.8)$$

Bu nedenle, \mathbf{x}_0 noktası etrafında Taylor serisine açılan $f(\mathbf{x})$ amaç fonksiyonunda optimum nokta, birinci türevin sıfıra eşit olduğu \mathbf{x}_{ekst} noktasıdır. Ancak Denklem 3.7 sadece birinci türevler içermemektedir. Görüleceği üzere ikinci ve daha üst dereceden türevleri de içermektedir. Üçüncü terimden sonraki ifadelerin fonksiyona etkisi çok küçük olduğundan hassasiyet açısından önemsenmeyebilir, dolayısıyla üçüncü terimden sonraki ifadeler değerlendirmeye katılmayabilir. Yine de amaç fonksiyonu olan Taylor serisi açılımı, ikinci derece türev ve \mathbf{x} ile \mathbf{x}_0 'ın farkının karesini ihtiva etmektedir. Burada Hessian matrisi tarafından negatif bir etkinin söz konusu olmaması için Hessian matrisi $\mathbf{G}(\mathbf{x})$ 'in yarı-kesin pozitif matris olması gerekmektedir (Arora 2004). Tüm bu halleri dikkate alarak ekstremum noktayı elde edebilmek için birinci türevin sıfıra eşit olduğu durum Denklem 3.7'ye uygulanacak olursa;

$$f'(\mathbf{x}_{ekst}) = \mathbf{g}(\mathbf{x}_0) + \mathbf{G}(\mathbf{x}_0) \cdot (\mathbf{x}_{ekst} - \mathbf{x}_0) = 0 \quad (3.9)$$

denklemini elde edilmiş olur. Bu denklem \mathbf{x}_{ekst} yalnız başına bırakılacak şekilde tekrar düzenlenecek olursa;

$$\mathbf{x}_{ekst} = -\mathbf{g}(\mathbf{x}_0) \cdot \mathbf{G}^{-1}(\mathbf{x}_0) + \mathbf{x}_0 \quad (3.10)$$

ifadesine ulaşılmış olur. Burada, \mathbf{G}^{-1} ifadesi Hessian matrisinin tersini ifade etmektedir.

Eğer amaç fonksiyonu ikinci dereceden ve türevi mevcut bir denklemse Denklem 3.10'daki ifade yardımı ile bu fonksiyonun gerçek optimum noktası rahatlıkla hesap edilebilir. Bununla birlikte yukarıda çalışma prensibinin üzerinde kısaca durulan klasik optimizasyon yöntemlerinin bu görevlerini başarıyla yerine getirebilmeleri için iki önemli gereksinim vardır:

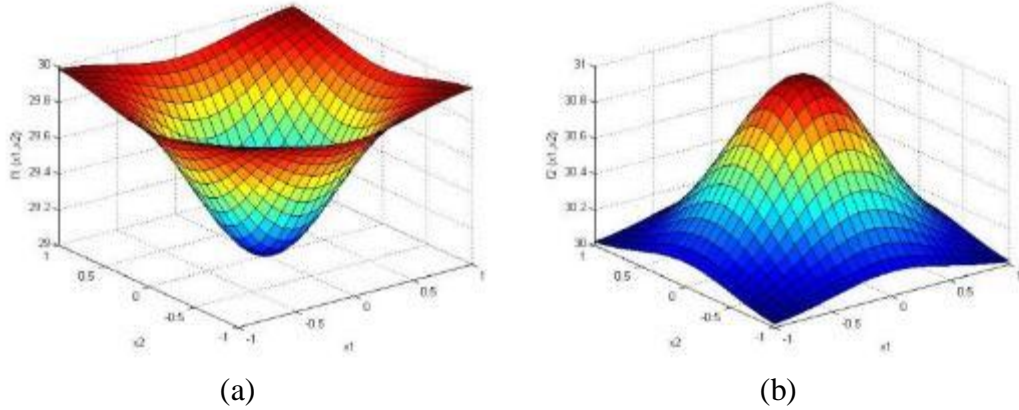
1. Amaç fonksiyonunun iki kez üst üste türevi alınabilir olmalıdır.
2. Amaç fonksiyonu sadece bir tane global optimum noktaya sahip olmalıdır.

İki kez türevi alınabilir ve sadece bir tane optimum noktaya sahip iki tane örnek amaç fonksiyonu şu şekilde olabilir:

$$f_1(x_1, x_2) = 30 - e^{-(2x_1^2 + 2x_2^2)} \quad (3.11)$$

$$f_2(x_1, x_2) = 30 + e^{-(2x_1^2 + 2x_2^2)} \quad (3.12)$$

Denklem 3.11'deki amaç fonksiyonu sadece bir tane minimuma sahip iken Denklem 3.12'de verilen amaç fonksiyonu ise sadece bir tane maksimuma sahiptir. Bu amaç fonksiyonlarının grafikleri Şekil 3.1'de verilmiştir. Bu şekilden de görüleceği üzere belirlenen aralıkta her iki fonksiyon da sürekli ve sadece 1 tane optimum noktaya sahiptirler.



Şekil 3.1 Minimuma (a) ve maksimuma (b) sahip iki amaç fonksiyonu

Dik-iniş yöntemi (steepest descent), türevi alınabilen ve tek minimum noktaya sahip olan bir fonksiyonun minimumunu bulan türeve bağlı en basit yöntemlerden birisidir. Denklem 3.10'a bağlı olarak bu yöntem ters Hessian matrisi $\mathbf{G}^{-1}(\mathbf{x}_0)$ 'ın birim matris ile yer değiştirebileceğini varsaymaktadır:

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{pmatrix} \quad (3.13)$$

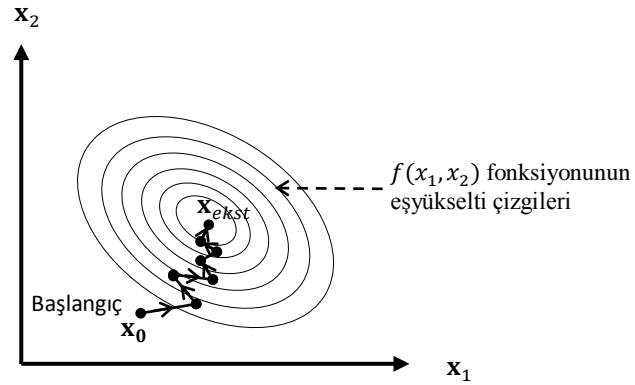
Bu yer deęiřtirme direkt olarak minimuma etki etmese de \mathbf{x}_1 noktasına etki etmektedir:

$$\mathbf{x}_1 = \mathbf{x}_0 - \mathbf{g}(\mathbf{x}_0) \quad (3.14)$$

Burada \mathbf{x}_0 'dan \mathbf{x}_1 'e uygun bir ilerleme veya adım ile hareket edildiğinde gradyanın negatif etkisi ile yeni nokta \mathbf{x}_1 minimuma daha yakın olacaktır. Bu işleme γ adım boyutu ile bir iterasyon süresince devam edilecek olursa dik-iniř yönteminin temel prensibi elde edilmiř olur:

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \gamma \cdot \mathbf{g}(\mathbf{x}_n) \quad (3.15)$$

Burada ilk önce negatif gradyan hesap edilerek yön tayin edilir, sonra belirlenen adım boyutu kadar ilerleme gerçekleştirilerek ekstremum noktaya adım büyüklüğünün deęerine baęlı olarak belirli yakınlıkta ulařılmaya çalıřılır. Őekil 3.2 dik-iniř yönteminde bu prensibi kullanarak ve adım boyutunun her iterasyonda küçültülerek \mathbf{x}_0 bařlangıç noktasından optimum nokta olan \mathbf{x}_{ekst} noktasına ilerlemeyi göstermektedir.



Őekil 3.2 Dik-iniř yönteminde ekstremum noktaya ilerleme ařamaları

Minimum noktaya eriřme zamanı ve bulunan minimum noktanın gerçek minimum deęere olan yakınlığı adım büyüklüğüne baęlıdır. Örneğin eđer adım büyüklüğü çok küçük seçilirse minimum noktaya ulařma zamanı çok artacak ancak bununla birlikte bulunan minimum noktanın gerçek minimum deęerine olan yakınlığı artacaktır yani hassasiyet de artacaktır. Ancak eđer adım büyüklüğü çok büyük seçilirse bu sefer bulunan minimum noktanın gerçek minimum noktadan uzaklığı artacak ve hassasiyet

azalacaktır. Bir diğ er önemli husus da her problem için en uygun adım büyüklüğünün seçilmesi gerekliliğidir. Diferansiyel gelişim algoritmasının bir üstünlüğü de burada bahsedilen adım büyüklüğü probleminin olmamasıdır.

Hessian matrisinin tersini, $\mathbf{G}^{-1}(\mathbf{x}_0)$, birim matris ile değıřtirmenin getirdiğı bir takım olumsuzlukları gidermek için daha detaylı ve düzenli hazırlanmış yöntemler geliştirilmiştir. Gauss-Newton, Davidon-Fletcher-Powell, Broyden-Fletcher-Goldfarb-Shanno bu yöntemler arasında sayılabilir (Arora 2004). Bu yöntemler kabaca iki kısma ayrılmaktadır. *Quasi-Newton* olarak adlandırılan birinci gruptakiler Hessian matrisinin tersinin yerine bir dizi matris hesaplamaları kullanmaktadırlar, ancak burada yoğun matris hesaplamaları yapılmaktadır. Bunun tersine *conjugate gradyan* yöntemleri olarak adlandırılan ikinci gruptakiler ise Hessian matrisinden tamamen vazgeçmekte onun yerine ikinci türev hesaplarından kaçınmak için conjugate yönlerde doğrusal optimizasyonları kullanmaktadırlar. Quasi-Newton ve conjugate gradyan yöntemlerine ek olarak bu iki yöntemin karışımı şeklinde de yöntemler mevcuttur. Buna rağmen tüm bu yöntemler amaç fonksiyonunun bir veya iki defa türev alınabilir olmasını gerektirmektedirler. Bununla birlikte ikinci dereceden amaç fonksiyonlarında gösterdikleri hızlı yakınsama özelliklerini ikinci derece olmayan diğ er fonksiyonlarda kaybetmektedirler. Amaç fonksiyonunun tekillik göstermesi veya çok fazla sayıda ve derecede türevler içermesi durumlarında ise yüksek sayılabilecek sayısal hesaplama hataları oluşabilmektedir. Bu noktada amaç fonksiyonunun türev alınabilir olmasına bir başka deyişle türevlerinin mevcut olmasına gerek duymayan yöntemler bir esneklik getirmektedir.

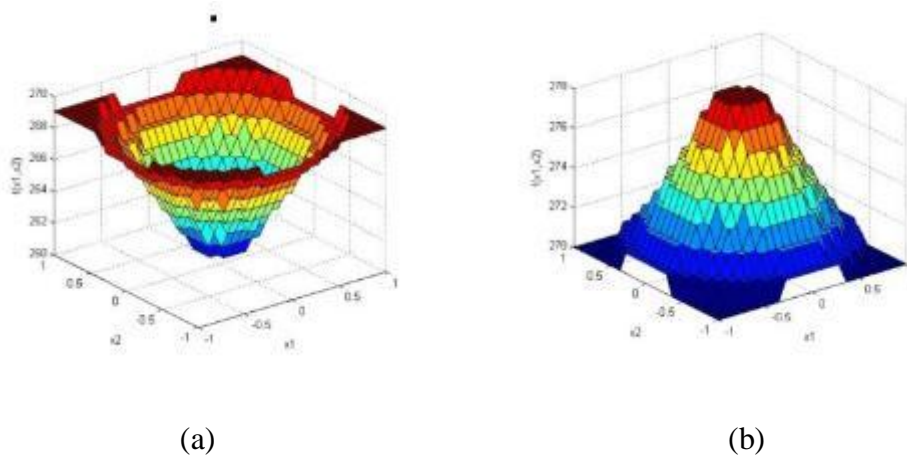
3.3. Tek-Noktalı Türevden Bağımsız Optimizasyon Yöntemleri ve Adım Büyüklüğü Problemi

Bir fonksiyonun türevinin alınamaması birçok nedene bağılı olabilir. Örneğ in bunlardan bir tanesi adım fonksiyonudur. Denklem 3.11 ve 3.12'deki fonksiyonlar adım fonksiyonuna dönüřtürülürse řu formları alacaklardır:

$$f_1(x_1, x_2) = Adım \left(9 \cdot (30 - e^{-(2x_1^2 + 2x_2^2)}) \right) \quad (3.16)$$

$$f_2(x_1, x_2) = \text{Adım} \left(9 \cdot (30 + e^{-(2x_1^2 + 2x_2^2)}) \right) \quad (3.17)$$

Denklem 3.16'daki amaç fonksiyonu yine sadece bir tane minimuma sahip iken Denklem 3.17'de verilen amaç fonksiyonu ise yine sadece bir tane maksimuma sahiptir. Ancak burada her bir adım sınırında türev tanımlı değildir. Bu amaç fonksiyonlarının grafikleri Şekil 3.3'te verilmiştir. Bu şekilden görüleceği üzere belirli adımlarda kesiklikler mevcut olup bu sınırlarda türevler mevcut değildir. Ancak belirlenen aralıkta yine her iki fonksiyon sadece 1 tane optimum noktaya sahiptir.



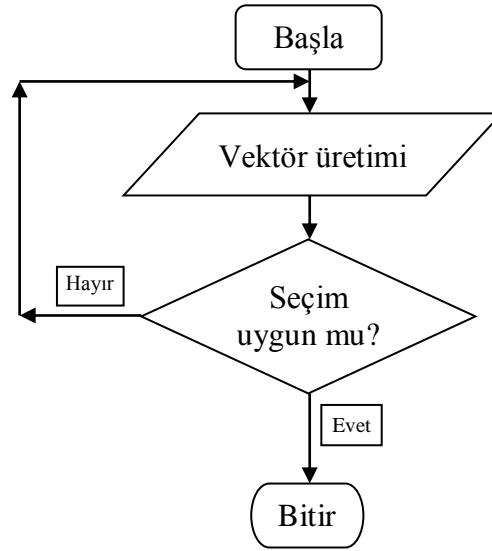
Şekil 3.3 Minimuma (a) ve maksimuma (b) sahip iki adım fonksiyonu

Fonksiyonların türevlerinin mevcut olmaması sadece adım fonksiyonuna bağlı olan bir durum olmayıp farklı durumlar söz konusu olabilmektedir. Bu durumlardan bazıları arasında şunlar sayılabilir (Price ve ark. 2005):

- Bir fonksiyonu kısıtlamak türev alınamayan bölgelerin veya sınırların oluşmasına neden olabilir.
- Eğer amaç fonksiyonu bir bilgisayar programı ise; bu durumda bazı koşullara bağlı olan döngüler, amaç fonksiyonunu türevinin alınamayacağı bir duruma dönüştürebilirler, en azından belirli noktalar veya bölgeler için bu durum söz konusu olabilir.
- Bazı durumlarda amaç fonksiyonu fiziksel bir deneyin sonucu olabilir, bu durumda yeterince açık ve kesin olmayan ifadelerin henüz mevcut olmaması hesap edilen türevlerin kullanışsız ve geçersiz olmasına neden olabilir.

- Bazı sanatsal kullanımlarda amaç fonksiyonu öznel olup analitik bir formülasyona dönüştürülememektedir (Bentley ve Corne 2002).
- Bazı evrimsel algoritmalarda üreme gerçekleştirecek bireylerin seçimi bireylerin diğerleriyle rekabet edebilme ve diğerlerini elimine edebilme özelliklerine göre yapılmaktadır. Bu durumlarda amaç fonksiyonu açık ve belirgin olmayabilmektedir.

Türevden bağımsız yöntemler, hesaplamalardan ziyade buluşsal ve koşullara bağlı olarak değişebilen “üret-ve-dene” prensibine göre çalışan algoritmalarlardır. Şekil 3.4’te verilen akış şeması bu yöntemlerin çalışma prensibini özetlemektedir.



Şekil 3.4 Türevden bağımsız yöntemlerin çalışma prensibini gösteren akış şeması

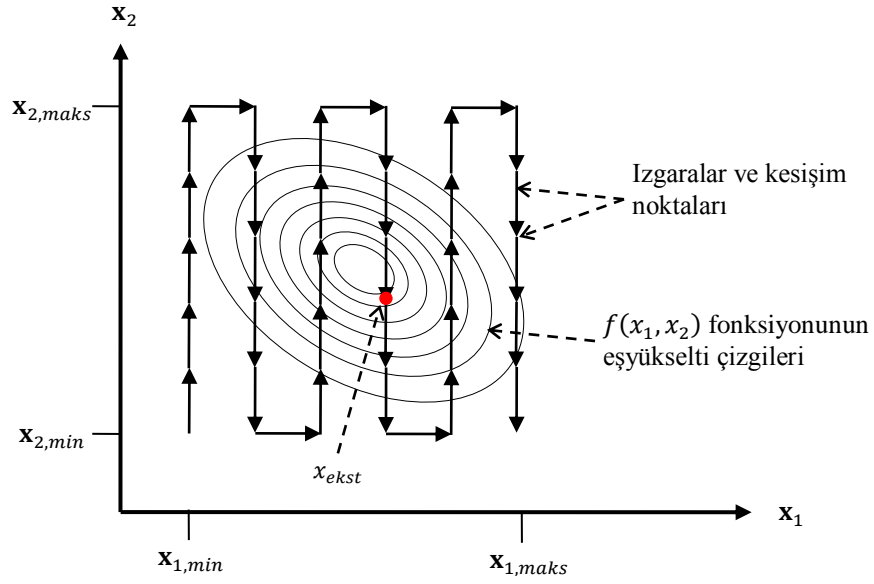
Direkt arama yöntemi olarak da adlandırılabilen türevden bağımsız yöntemler bir başlangıç noktası seçerek işlemlerine başlamaktadırlar. Bu seçim beraberinde *başlangıç noktası seçimi* problemini getirmektedir. Çünkü uygun olmayan bir başlangıç noktası seçimi ya sonuca ulaşma süresini uzatacak ya da hiç çözüm vermeyecek bir süreci başlatacaktır. Başlangıç noktası seçiminden sonra tekrarlanan işlem adımlarının sayısının yani iterasyon sayısının belirlenmesi gerekmektedir. Burada diğer bir uygulama da iterasyon sonlandırma kriterinin belirlenmesidir. Bu işlem de beraberinde *boyut* problemini getirmektedir. Uygun olmayan bir iterasyon sayısı veya sonlandırma kriterinin seçimi gereksiz işlem adımlarına sebep olabilir. Bir sonraki aşama vektör

üretimi ile yeni bir nokta seçme işlemidir. Burada ilerleme yönüne ve adım büyüklüğüne karar verilir. Bu işlem ise beraberinde *adım-büyükülüğü* problemini getirmektedir. Küçük seçilen adım büyüklüğü ile çok küçük ilerlemelerle sonuca ulaşılmaya çalışılacak ve işlem zamanı gereksiz uzatılacaktır. Tersine, büyük seçilen adım büyüklüğü ise optimum noktanın atlanıp yakalanamamasına neden olabilecektir. Bu da *güvenilirlik* problemini beraberinde getirecektir. Son aşama ise seçim aşamasıdır. Burada yeni üretilen nokta ile önceki nokta birbiri ile kıyaslanır ve amaca yönelik olarak yeni nokta ile eski nokta arasında bir seçim yapılır. Eğer yeni nokta amaç fonksiyonuna göre daha uygun ise bu noktaya ulaşmak için kullanılan yön ve adım büyüklüğü kabul edilmiş olur. Aksi durumda yeni bir yön ve adım büyüklüğü belirlenerek yeni bir nokta oluşturulur ve yine bir seçim yapılır. Döngü bu şekilde belirlenen iterasyon sayısına veya iterasyon koşuluna ulaşıncaya kadar devam eder. Türevden bağımsız yöntemlerin en belirgin özelliği sonraki eylemleri doğrudan etkileyen bir seçim mekanizmasına sahip olmalarıdır (Price ve ark. 2005).

3.3.1. Saydırma veya kaba kuvvet (brute force) ile arama

Adından da anlaşılacağı üzere tek-noktalı türevden bağımsız bir diğer deyişle direkt arama yöntemleri tek bir başlangıç noktası ile işlemlere başlamaktadırlar. En basit tek-noktalı direkt arama yöntemlerinden biri kaba kuvvet yaklaşımıdır. Saydırma işlemi olarak da anılan kaba kuvvet yaklaşımında belirlenen sınırlarda bölge ızgaralara bölünür ve her kesişim noktası ziyaret edilir. Ziyaret edilen kesişim noktalarının arasından amaç fonksiyonuna en uygun olanı hafızaya alınarak saklanır ve tüm bölge bu şekilde taranır. İşlem bittiğinde hafızada saklanan kesişim noktası aranılan optimum nokta olur (Şekil 3.5).

Bu yöntemde yine adım büyüklüğü problemi ile karşı karşıya kalınmaktadır. Amaç fonksiyonuna uygun adım büyüklüğü seçilmediği durumlarda ya optimum nokta atlanır ve yakalanamaz (Şekil 3.5) ya da optimum noktayı bulmak için çok fazla zaman harcanır.



Şekil 3.5 Kaba kuvvet yönteminin çalışma prensibi

Probleme uygun adım büyüklüğü seçiminin zor bir işlem olması kaba kuvvet yönteminin çok-parametrelili optimizasyon problemlerinde çok nadir olarak kullanılmasına sebep olmaktadır. Bu nedenle daha iyi örnekleme ve daha akıllı arama yapabilecek yöntemlere gereksinim duyulmuştur (Price ve ark. 2005).

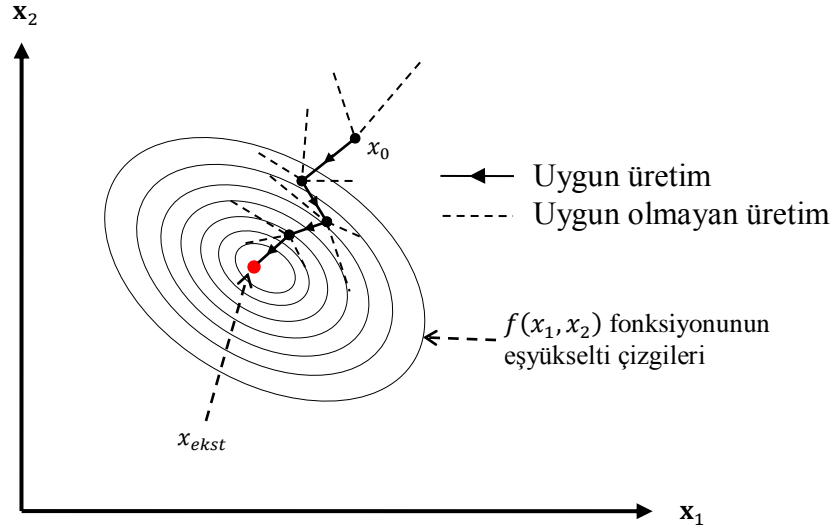
3.3.2. Rastgele yürüyüş (random walk) yöntemi

Bu yöntem kaba kuvvet ile arama yaklaşımının doğasında var olan boyut problemi üzerinde durmakta ve mevcut adımda aranan yeni amaç fonksiyonu değerini rastgele atanmış noktalar içerisinde en uygununu seçerek oluşturmaktadır. Yeni noktalar mevcut x_0 noktasına Δx vektörü eklenerek oluşturulmaktadır. Burada i . adımda Δx_i vektörü Gauss dağılımına göre seçilmektedir. Gauss dağılımı şu ifade ile hesap edilmektedir (Price ve ark. 2005):

$$P(\Delta x_i) = \frac{1}{\sigma_i \cdot \sqrt{2\pi}} e^{-0.5 \frac{(\Delta x_i - \mu_i)^2}{\sigma_i^2}} \quad (3.18)$$

burada σ_i ve μ_i , i . adım için sırasıyla standart sapmayı ve ortalama değeri belirtmektedir. Gauss dağılımı ile üretilen yeni noktanın amaç fonksiyonu değeri mevcut noktanın amaç fonksiyonu değerine göre uygun değilse bu durumda Gauss

dağılımı ile yeni bir nokta üretilmekte ve daha uygun bir amaç fonksiyonu değeri elde edilinceye kadar yeni nokta üretimine devam edilmektedir. Şekil 3.6 rastgele yürüyüş yönteminin işleyişini açık bir şekilde göstermektedir.



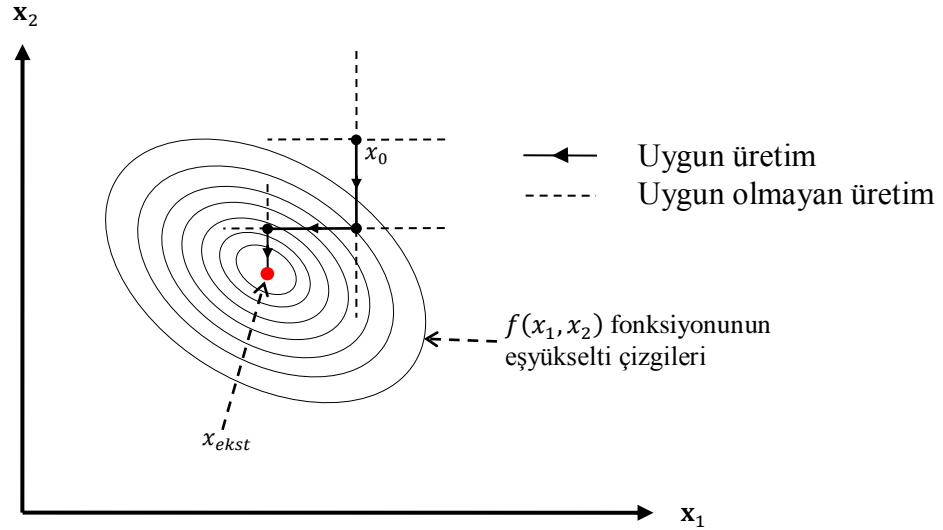
Şekil 3.6 Rastgele yürüyüş yönteminin çalışma prensibi

Rastgele yürüyüş yöntemi de diğer klasik yöntemler gibi adım büyüklüğü problemi ile karşı karşıyadır. Çünkü amaç fonksiyonuna uygun seçilmemiş standart sapma değerleri gereksiz adımların oluşmasına neden olabilmektedir.

3.3.3. Hooke ve Jeeves yöntemi

Hooke ve Jeeves yöntemi tek-noktalı türevden bağımsız yöntemlerden olup özellikle adım büyüklüğü problemini çözmeye yönelik olarak geliştirilen bir yaklaşımdır (Hooke ve Jeeves 1961). Bu yöntem işleme x_0 başlangıç noktası ile başlamakta ve her koordinat eksenini kendi adım büyüklüğü ile taramaktadır. Hem pozitif hem de negatif olmak üzere D ile temsil edilen koordinat eksen yönlerinde geçici noktalar oluşturulmakta ve birbirleri ile kıyas edilerek aralarından en iyisi olan nokta x_1 oluşturulmaktadır. Bu en iyi x_1 noktası eğer başlangıç noktası x_0 'dan daha iyi ise bu sefer en iyi noktanın bulunduğu yönde tekrar bir ilerleme girişiminde bulunulmaktadır. Yeni elde edilen nokta x_1 'den iyi ise aynı adım büyüklüğü ve yön ile bir önceki noktadan daha kötü bir nokta oluşuncaya kadar işleme devam edilmektedir. Sonuçta elde edilen yeni nokta başlangıç noktasının yerini almaktadır. Eğer x_0 başlangıç noktasından sonra tüm

yönlerde oluşturulan yeni noktalar daha iyi amaç fonksiyonu değerlerine sahip olmazlarsa bu sefer adım büyüklüğünün fazla olduğuna karar verilmekte ve adım büyüklüğü küçültülerek yine tüm yönlerde yeni noktalar oluşturulmaktadır. Bu şekilde x_0 başlangıç noktasından daha iyi noktalar üretilene kadar işleme devam edilmektedir. Yöntemin ilerleme aşamaları ve görsel temsili Şekil 3.7`de verilmiştir.



Şekil 3.7 Hooke ve Jeeves yönteminin çalışma prensibi

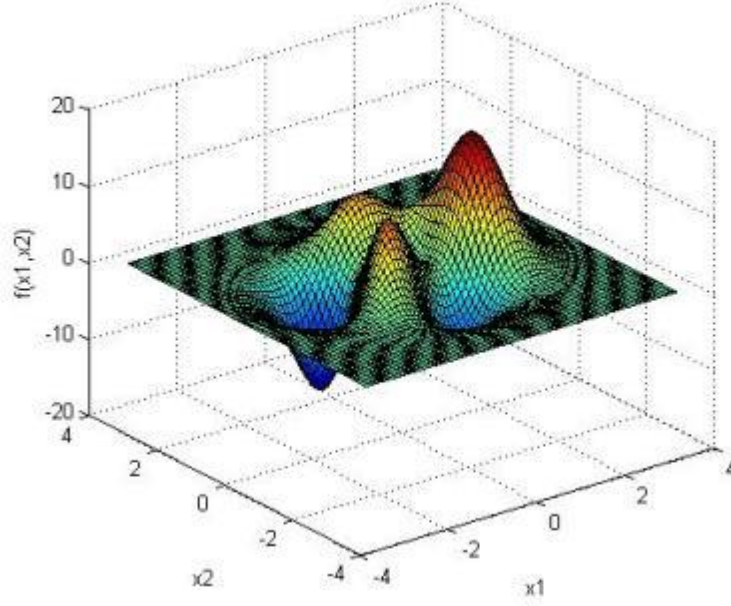
Birçok fonksiyonda Hooke ve Jeeves yöntemi, kaba kuvvet ile arama ve rastgele yürüyüş yöntemlerine göre uyarlamalı adım büyüklüğü özelliği nedeniyle sonuca daha iyi ve hızlı bir şekilde ulaşmaktadır. Ancak burada adım büyüklüğünün devamlı küçülmesi ve hiç artmaması özellikle fonksiyonun vadili bir yapıya sahip olması durumunda vadinin sonuna kadar aynı –muhtemelen küçük- adım büyüklüğü ile devam edilmek zorunda kalınması nedeniyle hız konusunda bir dezavantaj oluşturmaktadır.

3.4. Lokal ve Global Optimizasyon

Yukarıda bahsedilen yöntemlerin hepsi amaç fonksiyonunun sadece bir tane global optimum noktaya sahip olması şartı ile geçerliliklerini sürdürebilirler. Aksi durumda yani amaç fonksiyonunun sadece tek ekstremum nokta yerine birçok ekstremum nokta ihtiva etmesi durumunda yukarıda bahsedilen yöntemler lokal optimumlara takılabilir ve yanlış sonuçlar elde edilebilir. Denklem 3.19`da verilen amaç fonksiyonu birden fazla

ekstremum noktaya sahiptir. Şekil 3.8'den görüleceği üzere bu fonksiyon birçok ekstremum noktaya sahiptir, yani birden fazla tepe ve çukur ihtiva etmektedir.

$$f(x_1, x_2) = 8(1 - x_1)^2 e^{-x_1^2 - (x_2 + 1)^2} - 20 \left(\frac{x_1}{50} - x_1^3 - x_2^5 \right) e^{-(x_1^2 - x_2^2)} - \frac{1}{4} e^{-(x_1 + 1)^2 - x_2^2} \quad (3.19)$$



Şekil 3.8 Birden fazla optimum noktaya sahip amaç fonksiyonu

Birden fazla optimum noktaya sahip olan bu fonksiyonlarda yukarıda bahsedilen tek-noktalı optimizasyon yöntemlerinin lokal optimumlara takılmalarının sebebi olarak uygun başlangıç değerinin seçilememesi gösterilebilir. Dolayısıyla bu yöntemler ile birden fazla optimum noktaya sahip fonksiyonlar çözülmeye çalışıldığında başlangıç değer problemi ile karşı karşıya kalınmaktadır. Doğru sonuçların elde edilebilmesi için global optimum nokta etrafındaki bir başlangıç noktası ile işlemlere başlanması gerekmektedir. Bir lokal optimum noktaya saplanıp kalındığında bu noktadan kurtulabilmek için çeşitli atlama ve bu nokta etrafında belirli uzaklıklarda arama işlemleri gerçekleştirebilen yöntemler geliştirilmiştir. Bu yöntemlerden birisi de benzetimli tavlama yöntemidir.

3.5. Benzetimli Tavlama Yöntemi

Benzetimli tavlama yönteminde arama algoritması, amaç fonksiyonu yüzeyinde minimumun araştırıldığı durumlarda yeni oluşturulan noktaların veya vektörlerin çoğunluğunu minimum oluşturacak şekilde çok az bir kısmını ise minimumun tersine amaç fonksiyonunda artımlar gerçekleştirecek şekilde seçerek işleme başlamaktadır (Kirkpatrick ve ark. 1983). Bu artım vektörlerinin oluşturulma olasılığı, bu vektörlerin amaç fonksiyonundaki değerleri ile daha önce oluşturulan vektör değerleri arasındaki farka bağlı olarak belirlenmektedir. Buna göre farkın arttığı durumlarda olasılık azalmakta farkın azaldığı durumlarda ise olasılık artmaktadır. Ayrıca bu vektörlerin oluşturulma olasılığı fonksiyon hesaplama sayısı ile de ters orantılıdır. Yani ne kadar fazla fonksiyon hesabı yapılmaya başlandıysa o oranda fonksiyon değerini arttırıcı vektör oluşturma olasılığı da azalmaktadır. Böylece arama uzayında lokal minimumlara takılma ihtimali azaltılarak daha etkin aramalar gerçekleştirilebilmektedir. Rastgele yürüyüş yöntemi benzetimli tavlama yönteminde yeni vektörlerin üretiminde geleneksel olarak kullanılmış bir yöntemdir. Ayrıca diğer direkt arama yöntemleri de vektör üretimi aşamasında kullanılabilir. Şekil 3.9 benzetimli tavlama yönteminin temel işlem adımlarını özetlemektedir.

- Bir başlangıç vektörü seç ve algoritmayı başlat.
- Bir başlangıç sıcaklığı seç (T).
- Yakınsama kriterine ulaşıncaya kadar şu işlemleri yap:
 - o Gauss dağılımı ile adım büyüklüğünü ve yönünü seç.
 - o Eğer bu dağılım ile elde edilen yeni vektörde bir gelişme sağlanırsa:
 - Mevcut vektörü yenisi ile değiştir.
 - o Aksi durumda:
 - Yeni vektör ile eski vektör arasında bir fark vektörü oluştur (d).
 - Normal dağılımdan faydalanarak bir rastgele değişken belirle (r).
 - Eğer bu rastgele değişken Metropolis algoritmasından küçük ise ($r < e^{-d/KT}$):
 - Yeni vektörü artım olacak şekilde ata.
 - o Sıcaklığı (T) bir azaltma faktörü ile çarparak güncelle.

Şekil 3.9 Benzetimli tavlama yönteminin işlem adımları

Tavlama terimi malzeme bilimindeki sıcak malzemenin soğuma hızının azaltılarak malzeme içindeki atomların minimum enerjiye sahip olacak şekilde bağlarını

oluşturmasına izin verilmesinden esinlenerek kullanılmıştır. Buradaki enerji ile benzetimli tavlama yöntemindeki fonksiyon değeri denk tutularak tavlama işleminin doğasında olan minimum enerjinin sağlanması için Metropolis algoritmasından faydalanılmıştır (Metropolis ve ark. 1953). Burada minimizasyon problemleri için amaç fonksiyonunun değerini düşüren tüm yeni vektörlerin üretimine izin verilirken amaç fonksiyonunun değerini arttıran yeni vektörlerin üretilmesinde ise Metropolis algoritması kullanılmıştır. Bu algoritmada kullanılan olasılık şu şekilde ifade edilmektedir:

$$v = e^{-d/KT} \quad (3.20)$$

Burada; v yeni vektör üretme olasılığını, d yeni vektör ile eski vektör arasındaki fark vektörünü, K probleme bağlı olarak değişebilen bir sabit ve T ise sıcaklığı temsil eden kontrol değişkenidir. Vektör üretme olasılığı, d ile ters orantılı K ve T ile doğru orantılıdır. Yani fark vektörü azaldığında olasılık artmakta, probleme bağlı olan sabit küçük seçildiğinde veya sıcaklık değişkeni azaldığında olasılık da azalmaktadır.

Benzetimli tavlama yönteminin bir dezavantajı probleme uygun oranda sıcaklık T 'yi düşürebilen bir tavlama planının bulunması için özel bir çaba gerektirmesidir. Eğer T çok hızlı bir şekilde azalırse benzetimli tavlama yöntemi sanki lokal optimizasyon yöntemleri gibi davranıp başlangıç vektörü yakınındaki optimum noktaya takılıp kalacaktır. Ters bir durumda eğer T çok yavaş azalacak şekilde bir oran seçilirse bu sefer gereksiz hesaplamalar yapılacak ve bu hesaplamalar için çok fazla zaman harcanacaktır. Böylece adım büyüklüğü problemi burada da devam etmektedir.

3.6. Çok-Noktalı Türeve Bağlı Optimizasyon Yöntemleri

Adından da anlaşılacağı üzere çok-noktalı optimizasyon yöntemleri arama işlemine sadece bir başlangıç noktası ile değil de birçok başlangıç noktası ile aynı anda başlamaktadır. Ancak burada hangi sayıda ve çeşitte başlangıç noktasının oluşturulacağı problemden probleme değişmekle birlikte tespiti zor bir işlemdir. Ayrıca birden fazla lokal optimum değere sahip amaç fonksiyonlarında başlangıç değerlerinin etkin bir şekilde oluşturulması çok zordur. Örneğin başlangıç noktaları eşit dağıtılamayarak aynı lokal optimum nokta etrafında yığılmalara neden olunabilir. Ayrıca aynı lokal optimum

nokta etrafına konumlanan her bir gereksiz başlangıç noktası zaman kaybına ve fazladan yapılan hesaplamalara neden olmaktadır.

Kümeleme yöntemleri çok-noktalı optimizasyon yöntemindeki gereksiz hesaplamalar üzerinde durarak aynı lokal havuza takılan noktaları tek noktaya indirgeyerek hesaplama sayısını azaltan bir yapıya sahiptir. Ancak kümeleme yöntemleri az sayıda parametre içeren optimizasyon yöntemlerinde etkin bir şekilde kullanılmakta olup fazla sayıda parametre içeren problemlerde bu avantajını kaybetmektedir (Törn 1977).

3.7. Çok-Noktalı Türevden Bağımsız Optimizasyon Yöntemleri

Genellikle bu sınıfa evrimsel algoritmalar olarak adlandırılan ve doğada gerçekleşen biyolojik süreçleri taklit eden algoritmalar girmektedir. Bu algoritmalar özellikle 1960`lardan sonra kullanılmaya başlanmıştır. Bunlar arasında genetik algoritmalar, evrimsel programlama, evrimsel stratejiler, memetik algoritmalar ve tabu araması sayılabilir. Ancak son günlerde insan ve hayvan davranışlarını taklit eden karınca ve arı kolonileri sürü optimizasyon algoritmaları, kültürel algoritmalar ve parçalı sürü optimizasyon algoritmaları geliştirilmiştir.

3.7.1. Evrimsel algoritmaların temel özellikleri

Tüm evrimsel algoritmaların kendine özgü ayırt edici özellikleri olmasıyla birlikte temelde benzer yapıya sahiptirler. Burada evrimsel algoritmaların kullandıkları temel terminoloji verilmeye çalışılmıştır.

- **Gen:** Genler evrimsel algoritmaların temel yapıtaşlarıdır. Genellikle iki tür gen yapısı kullanılmaktadır; genlerin reel bir sayıyla ifade edildiği tür ve bir alfabe setinden değerler alan tür. Bu ikinci grupta genellikle ikili, sekizli, ondalık ve onaltılık setler kullanılmaktadır (Qing 2009).
- **Kromozom:** Genlerin birleşmesinden oluşan kromozomlar da evrimsel algoritmaların temel yapıtaşlarından. Kromozomlar optimizasyon parametreleri olan tasarım değişkenlerinin sembolik gösterimidir. Genler kromozomlar içerisinde zincir halkaları şeklinde birbirlerine bağlanmaktadır (Şekil 3.10).

$$\mathbf{k} = \underbrace{g_{11} \dots g_{1L_1}}_{x_1} \quad \underbrace{g_{21} \dots g_{2L_2}}_{x_2} \quad \dots \quad \underbrace{g_{N1} \dots g_{NL_N}}_{x_N}$$

Şekil 3.10 Bir kromozomun yapısı

Şekil 3.10`da genel yapısı verilen kromozomda g_{ij} , x_i tasarım değişkenini temsil eden j nci geni ifade ederken L_i , x_i tasarım değişkeni için temsil edilen genlerin sayısını göstermektedir. Gerçek optimizasyon parametresi ile ona tekabül eden kromozom içindeki belirli genlerden oluşan alt dizge arasında iki-yönlü bir işlem mevcuttur. Evrimsel hesaplamada bu işlem kodlama/çözümleme olarak ifade edilmektedir. Genellikle bu kodlama iki türde yapılmaktadır. Doğal kodlama olarak adlandırılan birinci yöntemde optimizasyon parametresi genin kendisidir ($\mathbf{x}=\mathbf{g}$). Burada optimizasyon parametresi ile kromozom birbiri yerine kullanılabilir. Sayısallaştırıcı olarak adlandırılan ikinci yöntemde ise optimizasyon parametresi aşağıdaki bağıntı ile sayısallaştırılmaktadır:

$$x_i \equiv b_i^L + (b_i^U - b_i^L) \sum_{j=1}^{L_i} g_{ij} B_i^{-j} \quad (3.21)$$

Burada; B_i , x_i optimizasyon parametresi için alfabe setindeki tabandır. Örneğin 2`li alfabe setinde $B_i = 2$ olurken 10`lu alfabe setinde $B_i = 10$ dur. $D_i = [b_i^L, b_i^U]$, x_i optimizasyon parametresinin arama uzayındaki alt ve üst sınırlarını ifade etmektedir.

- **Uygunluk fonksiyonu:** Kromozomların amaç fonksiyonuna ve kısıtlara göre ne derece uyumlu olduğunu anlamada kullanılan uygunluk fonksiyonları, evrimsel algoritmalarda belirli bir ölçeklendirme ile uygun kromozomların önemini arttırmak ve bu kromozomları diğerlerine göre daha belirgin bir hale getirmek amacıyla kullanılmaktadır. Ancak diferansiyel gelişim algoritması gibi bazı evrimsel algoritmalarda kullanılmayabilmektedir.
- **Birey:** Evrimsel hesaplama alanında bir birey, optimizasyon parametre değerleri \mathbf{x} lerin ifade edildiği genlerden müteşekkil kromozomların (\mathbf{k}),

amaç ve kısıt fonksiyon değerlerinin (f) birleşmesinden oluşmaktadır. Bunların dışında bir bireye uygunluk değeri, nesil, hız, yaş, cinsiyet gibi özellikler de eklenebilmektedir (Qing 2009).

- **Popülasyon:** Bireylerin oluşturduğu topluluğa popülasyon adı verilmektedir. Popülasyonun önemli bir karakteristik özelliği olan yaş nesil ile ifade edilmektedir. Tüm evrimsel algoritmalar popülasyon üzerinden işlemlerini gerçekleştirmektedir ve bir sonraki popülasyonun kalitesini arttırmak amacını taşırlar. Bir popülasyonun en önemli karakteristik özelliği çeşitliliğidir. Ancak son zamanlarda yapılan çalışmalardan anlaşılmaktadır ki daha önemli olan nokta popülasyonun bu çeşitliliği sürdürebilme veya koruyabilme özelliğidir (Qing 2009).

3.7.2. Evrimsel stratejiler

Evrimsel stratejiler yöntemi Rechenberg (1973) ve Schwefel (1994) tarafından geliştirilmiştir. Bu yöntemde *en iyi olan hayatta kalır ve neslini devam ettirir* prensibi kullanılarak sonuçların geliştirilmesi amaçlanır. Bunun için üreme ve mutasyon işlemlerinden faydalanılır.

Çok-noktadan başlayan algoritmalara benzer şekilde evrimsel strateji amaç fonksiyonunun arama uzayında birbirinden farklı birçok örnekleme noktası oluşturur. Ancak her noktayı kendi başına geliştirmeye çalışan diğer çok-noktalı yöntemlerden farklı olarak burada noktalar bir popülasyon oluşturur ve tekrar birleştirme işlemiyle (üreme) birbirlerini etkileyerek optimum sonuca ulaşmaya çalışılır. Optimizasyon işlemine μ ebeveyn vektörlerinden oluşan bir popülasyon oluşturularak başlanır. Evrimsel strateji bir λ çocuk popülasyon vektörünü $\lambda \geq \mu$ olacak şekilde rastgele seçilen ebeveyn vektörlerinden üreme işlemi gerçekleştirilerek oluşturur. Üreme işlemi; bazı parametrelerin bir aileden bazı parametrelerin diğer başka bir aileden alınmak üzere *ayrık* gerçekleştirilebileceği gibi tüm ebeveynlerin parametrelerinin ortalaması alınmak üzere *orta seviyede* de gerçekleştirilebilir (Back 1997). Ebeveynlerin üreme işlemi gerçekleşip çocuk popülasyon vektörü oluşturulduktan sonra bu çocuk vektörler, ortalama değeri sıfır olan Gauss dağılımına göre belirlenen rastgele bir $\Delta \mathbf{x}$ dağılımı (Denklem 3.18) ile *mutasyon* işlemine tabi tutulurlar. Mutasyon işleminden sonra tüm λ

çocuk popülasyonu değerlendirme işlemine tabi tutulur. Burada işlem iki şekilde uygulanabilir; λ çocuk popülasyonu arasından en iyi bireyler veya vektörler seçilebilir (λ) veya ebeveyn popülasyonu μ ile çocuk popülasyonu λ arasından en iyi değerlere sahip olan bireyler seçilebilir ($\mu + \lambda$). Seçilen bireyler yeni popülasyonun ebeveynleri olmaktadır. İşleme bu şekilde devam edilerek global optimum noktaya ulaşılmaya çalışılır. Şekil 3.11 evrimsel stratejiler yönteminin temel işlem adımlarını özetlemektedir.

- μ bireylerinden oluşan bir başlangıç popülasyonu seç ve algoritmayı başlat.
- İterasyon sayısını belirle ve bu sayıya ulaşıncaya kadar şu işlemleri yap:
 - o Çocuk vektörlerini oluştur ($\lambda > \mu$):
 - μ ebeveyn popülasyonu arasından rastgele bir ebeveyn seç (P_1).
 - μ ebeveyn popülasyonu arasından rastgele başka bir ebeveyn seç (P_2).
 - Ebeveynleri üreme işlemine tabi tut ve yeni çocuk bireyi üret (c_1).
 - Yeni çocuk bireyi mutasyona uğrat.
 - Çocuk bireyi yeni popülasyona kaydet (λ).
 - o (λ) veya ($\mu + \lambda$) arasından en iyi bireyleri seç.

Şekil 3.11 Evrimsel stratejiler yönteminin işlem adımları

Evrimsel stratejiler temel yapısı itibariyle adım büyüklüğü probleminin bazı dezavantajlarına takılmaktadır. Bunun için literatürde algoritma üzerinde çeşitli geliştirmeler yapılmış ve yapılmaktadır. Bu değişiklikler özellikle mutasyon operatörü üzerine odaklanmaktadır. Ancak yapılan değişikliklerle birlikte problemin boyutu artmakta ve performans olumsuz etkilenebilmektedir. Özellikle birçok değişken içeren problemlerde algoritmaya eklenen her bir yeni matris hesabı büyük zaman alabilmekte ve yüksek kapasiteli hafızalara sahip bilgisayarlara olan gereksinim artmaktadır.

3.7.3. Genetik algoritmalar

Genetik algoritmalar, doğal evrim sürecini taklit eden, problemde gerçek değerleri değil de gerçek değerleri temsil eden ikili sayı sistemini kullanan ve “*en iyi olan hayatta kalır, diğerleri yaşamlarını devam ettiremez*” prensibine uygun olarak geliştirilmiş bir yöntemdir. Bu doğal seçim mekanizması 1967 yılında J.D. Bagley`in (1967) doktora tezinde gündeme getirilmiş ve 1970`li yıllarda, biyoloji bilimindeki bu prensibi yapay sistemlere uygulamayı ilk olarak John H. Holland düşünüp bilgisayar ortamında

uygulamaya çalışmıştır. 1985 yılında ise bu düşünceyi gerçekleştirmeyi Holland'ın öğrencisi olan Goldberg başarmıştır. Goldberg, gaz boru hatlarının denetimi üzerine yaptığı çalışma ile o güne kadar genetik algoritmaların pratik uygulamalar için pek faydalı olmadığını savunanların aksine genetik algoritmaların pratik uygulamalarda klasik metotlara göre çok daha iyi sonuçlar vermekle kalmayıp çok kısa sürelerde sonuca ulaşabileceğini de kanıtlamıştır. Ayrıca bu çalışma Goldberg'e 1985 yılında "National Science Foundation" (Ulusal Bilim Kurumu) tarafından verilen *Genç Araştırmacı* ödülünü de kazandırmıştır. Genetik algoritmalar genetik bilimindeki genetik yapıları bilgisayar ortamına aktararak bu yapılarla yapay olarak çalışırlar. Burada, 0 ve 1'lerle ifade edilen bireylerden oluşan bir popülasyon topluluğunda en iyi bireyi bulma amaçlanır. Popülasyondaki her birey amaç fonksiyonunu temsil eden bir uygunluk değerine sahiptir. Genetik algoritma, her zaman gerçek fonksiyondan bağımsız olan bu uygunluk değerleriyle çalışır. Algoritma süresince bireyler kendi aralarında ürer, çaprazlanır ve bazen de mutasyona uğrarlar (Goldberg 1989). Şekil 3.12 genetik algoritmaların temel işlem adımlarını özetlemektedir.

- Arama uzayında rastgele bir başlangıç popülasyonu oluştur.
- İterasyon sayısını veya iterasyonun devam edebileceği şartı belirle ve bu sayıya/şarta ulaşıncaya kadar şu işlemleri yap:
 - o Üreme: Popülasyondan rastgele 2 birey seç ve bunları eşleştirerek yeni bireyler oluştur.
 - o Çaprazlama: Yeni üretilen bireylerden rastgele 2 birey seç ve bunların genetik kodlarının bir kısmını yer değiştir.
 - o Mutasyon: Belli bir oranda yeni bireyler arasından mutasyona uğrayacak bireyleri seç ve bu bireylerin genetik kodunda bazı değerleri rastgele değiştir.
- Oluşan yeni bireylerden en iyi olanı seç.

Şekil 3.12 Genetik algoritmaların işlem adımları

Üreme işlemi içindeki eşleşme sonucunda oluşacak yeni bireyler uygunluk değerlerine göre yeni popülasyona kopyalanırlar. Sonuçta uygunluk değeri yüksek olan bireyler daha fazla sayıda kopyalarını yeni nesle aktarmış olurlar. Ayrıca üreme operatörü, uygunluk değerleri düşük olan kötü bireylerin kopyalarını yeni nesle aktarmayarak bu bireylerin ölümüne neden olur ve bunun sonucunda kötü bireyler, 0 ve 1'lerle ifade edilen kromozomlarını yeni nesillere aktaramamış olurlar.

Popülasyondaki bireylerin çeşitliliğini arttırmak ve daha iyi bireyler elde etmek amacıyla popülasyon içinde bulunan iki birey arasında gen değişimi işlemi yapılır. Bu işlem çaprazlama olarak anılmaktadır. Burada çaprazlanacak bireyler ve bu bireylerin genlerinin kesim noktaları rastgele olarak seçilir. Çaprazlama işlemi sonucunda iki yeni birey oluşturulmuş olur. Kesme işleminin yerine ve sayısına göre çeşitli çaprazlama işlemleri mevcuttur. Bunlar arasında tek-noktalı çaprazlama, iki-noktalı çaprazlama, çok-noktalı çaprazlama ve üniform çaprazlama işlemleri sayılabilir.

Mutasyon işlemi popülasyonun çeşitliliğini arttırmak için kullanılmaktadır. Bu operasyonda rastgele bir birey seçilir ve yine bu bireyin kromozomundan rastgele bir gen seçilerek değeri değiştirilir. Gen değeri 1 ise 0, 0 ise 1 yapılır (Karen 2005).

Seçim işlemi oluşturulan yeni bireylerden kötü uygunluk değerlerine sahip olanları öldürmek amacıyla gerçekleştirilir. Seçim işlemi genetik algoritmaların yakınsaması ile yakından ilişkilidir. Birçok seçim mekanizması geliştirilmekle birlikte en önemlileri arasında rulet tekerleği seçimi, sıralama seçimi, turnuva seçim mekanizması sayılabilir. Her mekanizmanın kendine özgü özellikleri olmakla birlikte son zamanlarda turnuva seçim mekanizması hızlı sonuca ulaşma bakımından biraz daha sık kullanılmaktadır.

Genetik algoritmalar genellikle ikili sistemde kodlanmaktadır. Bu kodlama, algoritmayı özellikle gerçek optimizasyon problemlerinde hesapsal açıdan karmaşıktırmakta, ayrıca ek bir kontrol parametresi (kromozom uzunluğu) algoritmaya eklendiğinden algoritmanın kontrolünü zorlaştırmaktadır. Bu nedenle reel-kodlu genetik algoritmalar geliştirilmiştir.

Genetik algoritmalar en optimum sonuca ulaşmada aşama aşama bireyleri iyileştirerek değil de kötü bireyleri öldürerek ve dolayısıyla arama uzayındaki uygun olmayan bölgelerden uzaklaştırarak sonuca ulaşmaktadır. Bu durum bir dezavantaj oluşturabilmektedir; şöyle ki genetik algoritma ile arama gerçekleştirilirken süreç ilerledikçe bireyler birbirlerine benzemekte hatta arama sonuçlandığında hemen hemen tüm bireyler birbirinin aynısı olabilmektedir. Bu durumda eğer başka bir bölgede gerçek optimum nokta mevcutsa ve değeri bulunan optimum noktaya çok yakınsa bu gerçek optimum nokta bulunamayabilir. Çünkü gerçek optimum noktaya yakın oluşabilecek yeni bireyler çok büyük bir olasılıkla öldürülecektir. Ayrıca bu durumda gerçek

optimum noktaya ulaşmak için genetik algoritmanın hassasiyeti arttırılacak olursa bu sefer gerçek optimum noktaya ulaşma olasılığı biraz artarken sonuca ulaşma zamanı oldukça yükselecektir. Bu nedenle başlangıçta genetik algoritmaların yakınsama hızı yüksek iken ilerleyen aşamalarda ters orantılı olarak bu yakınsama hızı oldukça azalmaktadır.

3.7.4. Nelder ve Mead yöntemi

Nelder ve Mead arama algoritması adım büyüklüğü problemi üzerinde durmuş ve adım büyüklüğünün ihtiyaca göre artıp azalmasına izin verecek şekilde çözüme ulaşmayı sağlamıştır (Nelder ve Mead 1965, Schwefel 1994). Aramaya problem uzayında rastgele dağılmış ve $D - 1$ boyutlu çok yüzlü cisim oluşturacak şekilde $x_i = x_0, x_1, \dots, x_D$ noktalarının oluşturulmasıyla başlanır. Örneğin eğer boyut, $D = 2$ olacak şekilde seçilirse aramaya üçgen şeklinde bir cisim ile devam edilir. Arama uzayında rastgele noktalar oluşturulduktan sonra bu noktalar, x_0 en iyi x_D en kötü olacak şekilde amaç fonksiyonuna göre en iyiden en kötüye doğru sıralanır. Bundan sonra yeni bir nokta x_r , oluşturmak için en kötü nokta x_D , çok yüzlü cismin karşı yüzüne doğru yansıtılır ve bir ağırlık faktörü $F1$ ile çarpılır.

$$x_r = x_D + F1 \cdot (x_m - x_D) \quad (3.22)$$

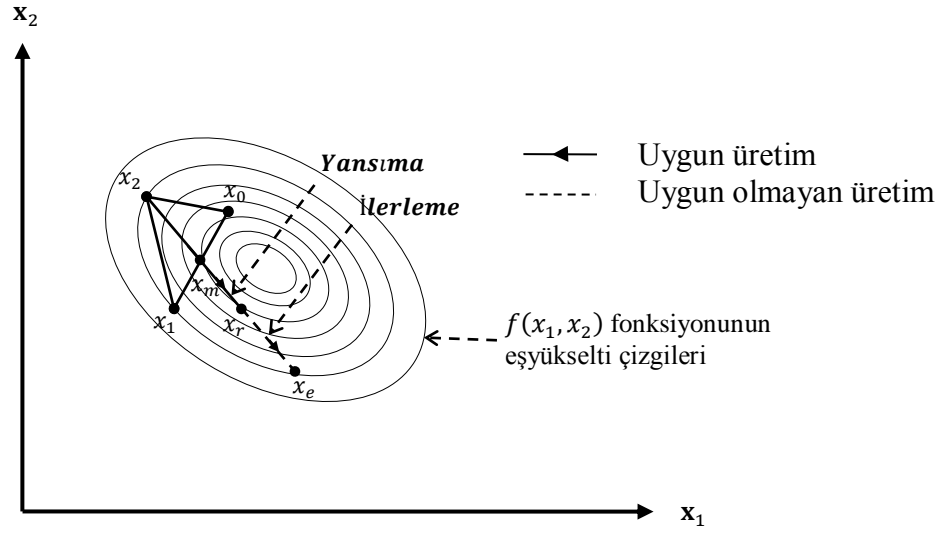
Burada x_m en kötü noktanın karşı yüzeyinin kitle merkezidir:

$$x_m = \frac{1}{D} \left(\sum_{i=0}^{D-1} x_i \right) \quad (3.23)$$

Denklem 3.22`de tanımlanan yansıma operasyonu Şekil 3.13`te gösterilmiştir (Nelder ve Mead 1965, Schwefel 1994).

Yansıma operasyonu ile elde edilen yeni nokta eğer en iyi nokta olan x_0 `dan daha iyi değere sahipse aynı yönde ilerlemeye devam edilir. Ancak bu sefer ikinci bir $F2$ ağırlık faktörü kullanılır ve yeni bir x_e noktası elde edilir:

$$x_e = x_r + F2 \cdot (x_m - x_D) \quad (3.24)$$



Şekil 3.13 Nelder-Mead yönteminde 2-boyutlu ($D = 2$) bir cisim için yansıma ve ilerleme operasyonu

Elde edilen bu yeni x_e noktasının değeri en iyi x_0 noktasından iyi ise en iyi değer olarak yer değiştirilir aksi takdirde bir önce elde edilen x_r noktası ile işlem devam edilir. Oluşturulan yeni set tekrar amaç fonksiyonuna göre en iyiden en kötüye doğru sıralanır ve işlemler tekrar edilerek en optimum sonuca ulaşılmaya çalışılır.

Nelder-Mead yöntemi arama uzayını fark vektörler oluşturarak tarayan en eski yöntemlerden biridir. Adım büyüklüğünün işlem esnasında değiştirilebilmesi ve uygun adım büyüklüğünün probleme göre değişmesi yöntemin adım büyüklüğü problemini ele almasında önemli avantajlar sağlamaktadır. Yöntemde örnekleme sayısı ($D - 1$) nokta ile sınırlandırılmaktadır. Bu kısıtlama karmaşık amaç fonksiyonlarına sahip problemlerde arama uzayının yüzey topografyasının daha ayrıntılı bir şekilde çıkarılamaması gibi bir dezavantaj oluşturmaktadır.

3.7.5. Evrimsel programlama

Evrimsel programlama Fogel (1966), tarafından önerilmiş ve sonlu konum makinelerinin işletilmesi için geliştirilmiştir. Bu algoritma özellikle ayrık sistemlerin çözümünde verimli olarak kullanılmıştır. Ancak son zamanlarda reel-kodlamalı versiyonları kullanılmaktadır. Bu algoritma, evrimsel stratejiler ile ufak bir ayırım dışında çok benzer özelliklere sahiptir. Seçim işleminde rastgele birey seçiminde her

kazanan bireye kazanma sayısı kadar bir özellik atanmakta ve bireyler bu özelliklerine göre seçilmektedir.

3.7.6. Parçacık sürü optimizasyonu

Parçacık sürü optimizasyonu 1995 yılında Kennedy ve Eberhart (1995) tarafından hayvanlar âlemindeki sürü ve koloni hareketlerinden esinlenerek ileri sürülmüş bir algoritmadır. Bir sürü veya koloni, arılar, karıncalar ve kuşlar gibi organizmaların etkileşimleri sonucu toplanmış bir yapıdır. Bir sürüdeki her organizma bir parçacık olarak adlandırılmaktadır ve evrimsel algoritmalarındaki bireyler burada parçacık olarak popülasyon ise sürü olarak ele alınmaktadır. Parçacık sürü optimizasyonunda bir sürü içindeki parçacığın pozisyonu optimizasyon parametrelerinin bir vektörünü temsil etmektedir. Bir parçacık evrimsel algoritmalarındaki bireyin özelliklerini taşımanın yanında iki ek özneliğe de sahiptir. Bunlar karşılaşılan en iyi pozisyon için belirlenen *hız* ve *hafıza* bileşenleridir. Buna ek olarak sürünün de bir hafızası mevcut olup burada her parçacığın en iyi pozisyon bilgileri tutulmaktadır. Benzer olarak her parçacığa ait en iyi hız bileşenleri de sürünün hız belleğinde tutulmaktadır.

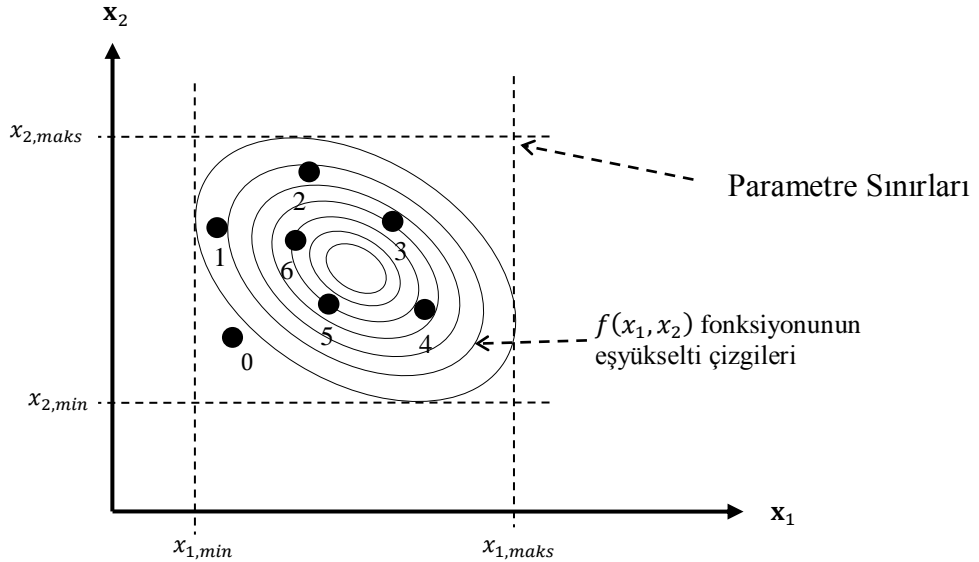
Parçacık sürü optimizasyonunda diğer algoritmalarından farklı olarak sürü içindeki parçacıklar bilgi alışverişinde bulunarak birbirleri ile etkileşimde bulunmaktadırlar. Algoritmanın işlem adımları Şekil 3.14`te verilmiştir.

- Arama uzayında rastgele bir başlangıç sürüsü oluştur.
- İterasyon sayısını veya iterasyonun devam edebileceği şartı belirle ve bu sayıya/şarta ulaşıncaya kadar şu işlemleri yap:
 - o Hız: Parçacıkların ve sürünün hız vektörlerini oluştur.
 - o Pozisyon: Parçacıkların pozisyonlarını hesapla.
 - Belirli kontrol parametrelerine göre hızı güncelle.
 - Belirli kontrol parametrelerine göre pozisyonu güncelle.
 - Belirli kontrol parametrelerine göre hafızayı güncelle.
 - Belirli kontrol parametrelerine göre sürü hafızasını güncelle.
 - o Seçim: En iyi parçacıkları seç.

Şekil 3.14 Parçacık sürü optimizasyon algoritmasının işlem adımları

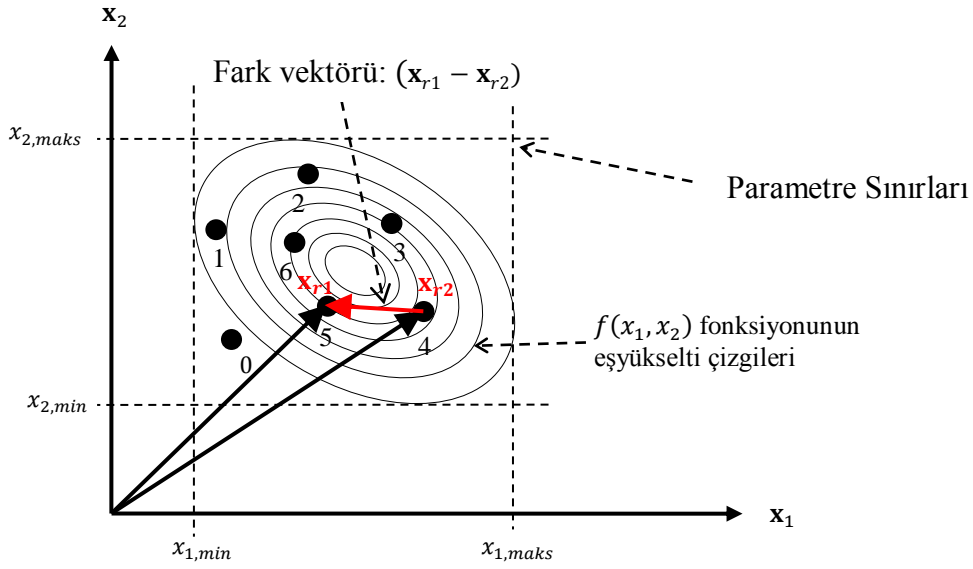
3.8. Diferansiyel Gelişim Algoritması

Storn ve Price (1995) güvenilir, çok yönlü ve kullanımı kolay bir optimizasyon algoritması geliştirmeye çalışmışlar ve 1995'te diferansiyel gelişim adını verdikleri algoritmayı yayınlamışlardır. Price, benzetimli tavlama yönteminin yavaş yakınsama problemi ve kontrol parametrelerinin verimli kullanılamaması gibi dezavantajlara sahip olmasından dolayı bu yöntemi geliştirmeye çalışmıştır. İkili kodlama ve doğru/yanlış operatörlerini kullanan benzetimli tavlama yöntemi üzerinde değişiklik yaparak onun yerine aritmetik operatörler ve gerçek kodlama kullanan ve diferansiyel gelişim algoritmasında kilit rol oynayacak olan yeni bir *diferansiyel mutasyon operatörü* geliştirmiştir. Diğer tüm evrimsel algoritmalar gibi diferansiyel gelişim algoritması da popülasyon tabanlı bir optimizasyon algoritması olup özellikle başlangıç değer probleminin çözümü üzerine geliştirilmiş bir algoritmadır. Parametrelerin önceden belirlenmiş sınırları dâhilinde ortaya çıkan arama uzayı içerisinde rastgele oluşturulmuş başlangıç noktaları N_p adet popülasyon vektörünü simgeler (Şekil 3.15). Diferansiyel gelişim algoritması eşyükselti çizgileri belirli olan bir arama uzayında $N_p = 7$ adet başlangıç noktasını rastgele oluşturarak işleme başlar. Bu noktalar başlangıç bireyleri olarak adlandırılır ve 0'dan $N_p - 1$ 'e kadar indekslenir (Price ve ark. 2005).



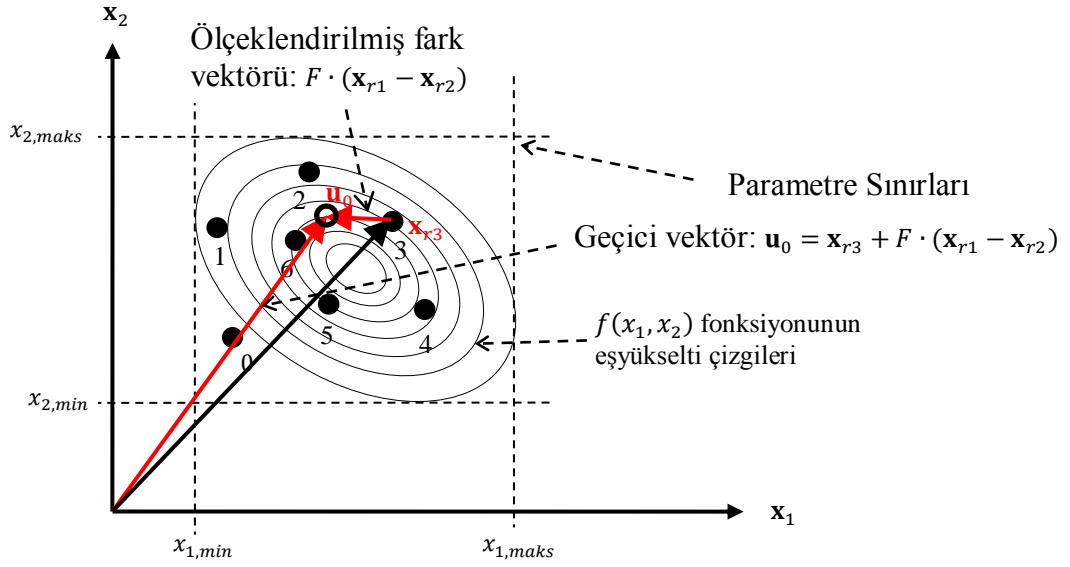
Şekil 3.15 Diferansiyel gelişim algoritmasında başlangıç popülasyonu

Diğer evrimsel algoritmalar gibi diferansiyel gelişim algoritması da yeni noktaları mevcut noktaların pertürbasyonu sonucunda elde eder; ancak burada sapmalar ne Nelder-Mead yöntemindeki gibi yansımalar aracılığıyla olup ne de evrimsel stratejilerdeki gibi önceden tanımlanmış olasılık yoğunluk fonksiyonuna göre örnekler oluşturacak şekildedir. Burada diferansiyel gelişim algoritması yeni vektörleri rastgele seçilen iki vektörün farkları alınıp ölçeklendirilerek oluşturulmuş fark vektörünü kullanarak üretmektedir. Örnek olarak Şekil 3.16'daki arama uzayında oluşturulmuş 7 noktadan rastgele 4 ve 5 noktaları seçilmiş ve bu noktaların temsil ettiği vektörlerden (\mathbf{x}_{r1} ve \mathbf{x}_{r2}) fark vektörü oluşturulmuştur.



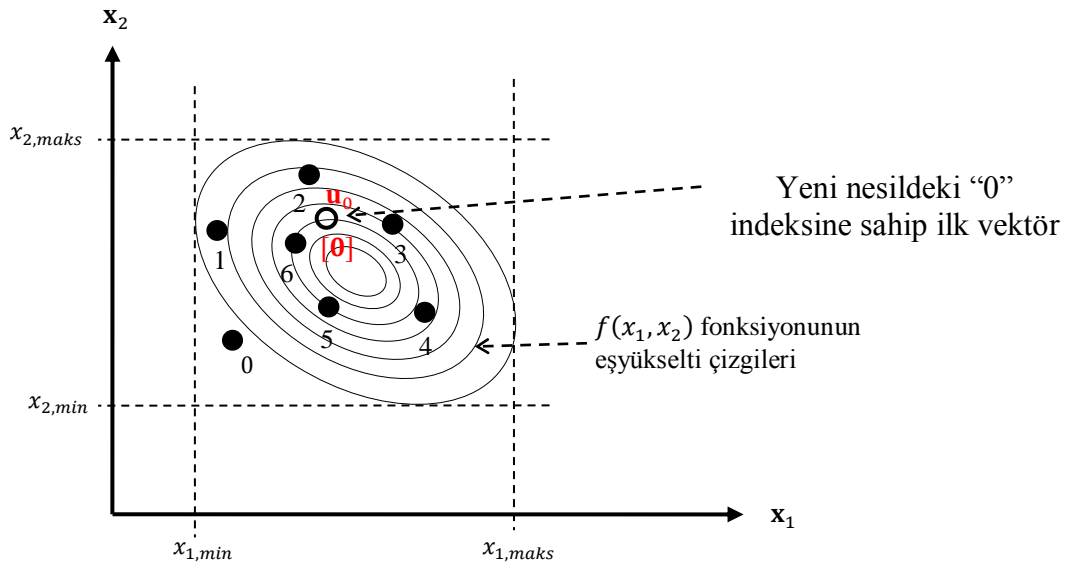
Şekil 3.16 Fark vektörünün oluşturulması: $\mathbf{x}_{r1} - \mathbf{x}_{r2}$

Oluşturulan fark vektörü belirli bir katsayı ile çarpılarak (F) ölçeklendirilir. Ölçeklendirilmiş vektör yine rastgele seçilen bir vektöre (3 no'lu nokta) eklenerek \mathbf{u}_0 geçici vektörü oluşturulur (Şekil 3.17).



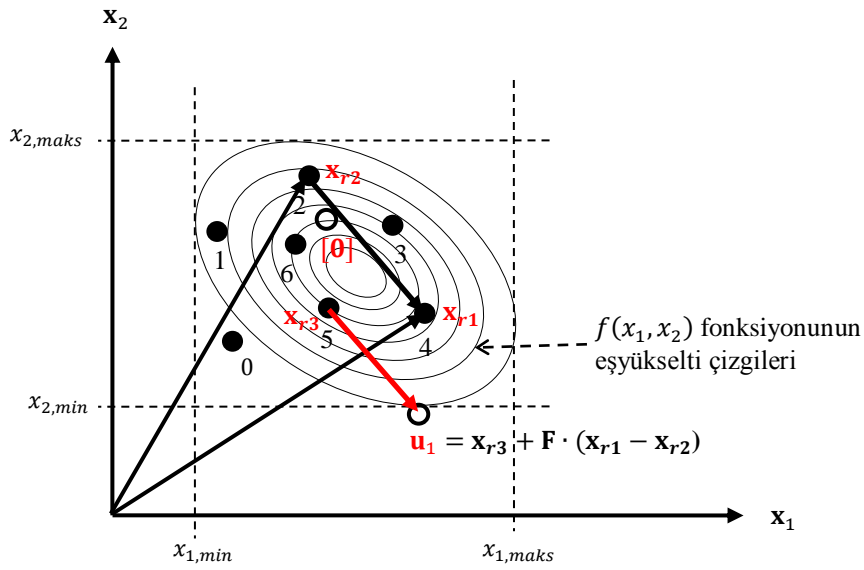
Şekil 3.17 Mutasyon işlemi

Seçim aşamasında oluşturulan geçici vektör ile popülasyonda mevcut olan ve aynı indekse sahip vektör aralarında yarıştırlırlar. Burada henüz sadece bir tek üretim gerçekleştirildiğinden indeks numarası 0'dır. Bu yarış esnasında daha uygun amaç fonksiyonu değerine sahip olan vektör seçimi geçerek bir sonraki nesilde yerini almaktadır. Şekil 3.18'de "0" indeksli eski nokta yerine amaç fonksiyon değeri daha iyi olan yeni [0] indeksli nokta yeni nesildeki vektörü oluşturmak üzere yerini almıştır.



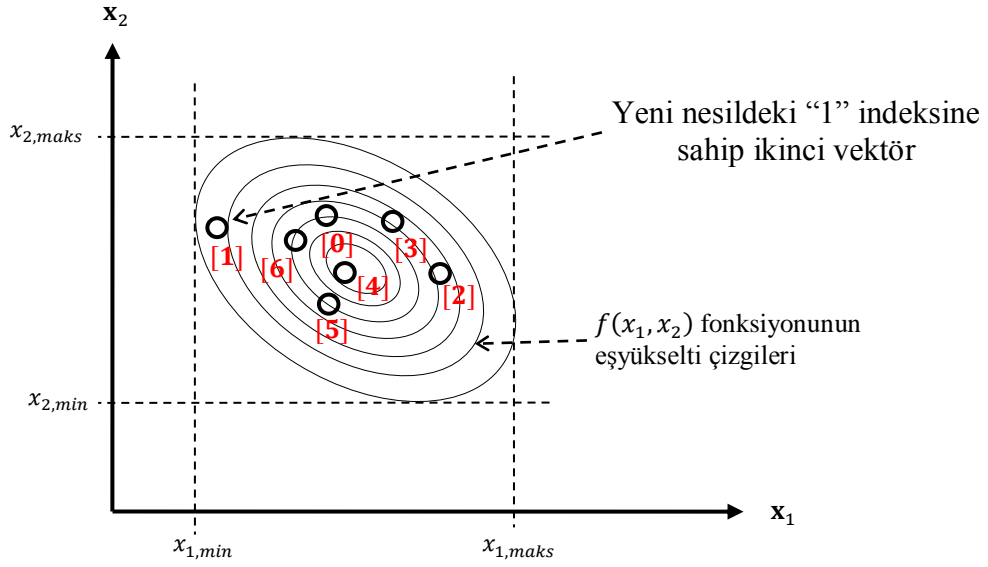
Şekil 3.18 Seçim işlemi

İkinci vektörü oluşturmak için yine rastgele iki nokta seçilir ve bu noktalardan fark vektörü oluşturulur. Örneğin rastgele 4'üncü ve 2'nci noktalar seçilmiş olsun (Şekil 3.19). Oluşturulan geçici vektör ölçeklendirilerek boyutu azaltılarak rastgele seçilen yeni bir vektöre örneğin 5'inci vektöre eklenerek yeni birey olan u_1 elde edilmiş olur (Şekil 3.19). Ancak burada bir önceki işlemin tersine yeni oluşturulan vektörün amaç fonksiyonu değeri aynı indekse sahip olan 1'inci vektörün amaç fonksiyonu değerinden kötüdür. Bu nedenle bir sonraki nesilde yeni oluşturulan vektör yerine eski vektör hayatını sürdürerek indeks değeri "[1]" olarak güncellenir (Şekil 3.20).



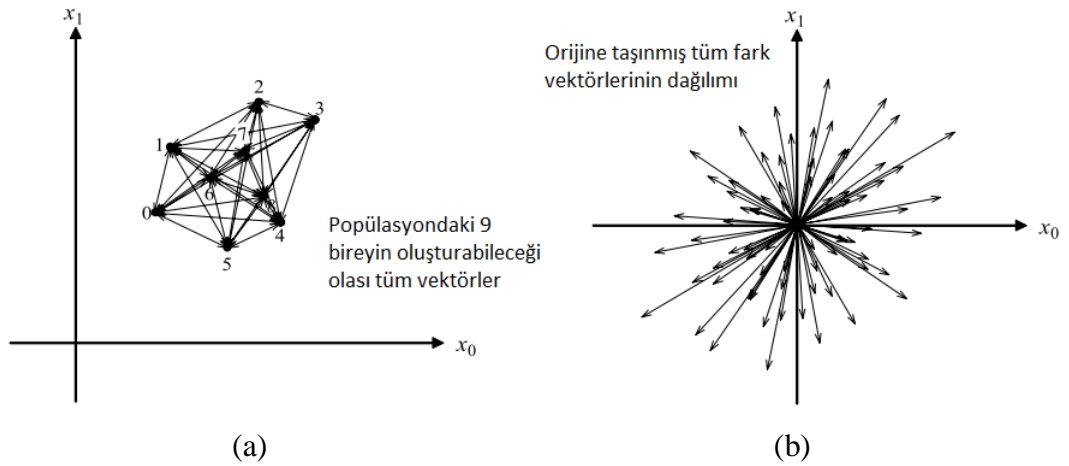
Şekil 3.19 İkinci geçici vektörün oluşturulması

İşleme popülasyon sayısı $N_p = 7$ 'ye ulaşana kadar devam edilir ve yeni geçici vektörler ile indeks numaraları aynı olan eski vektörler karşılaştırılarak amaç fonksiyon değerleri daha iyi olan vektörler yeni nesilde yerini alırlar. Böylece elde edilen yeni vektörler bir sonraki nesil için aile bireylerini oluşturmuş olurlar (Şekil 3.20).



Şekil 3.20 İlk nesil sonunda oluşan yeni bireyler

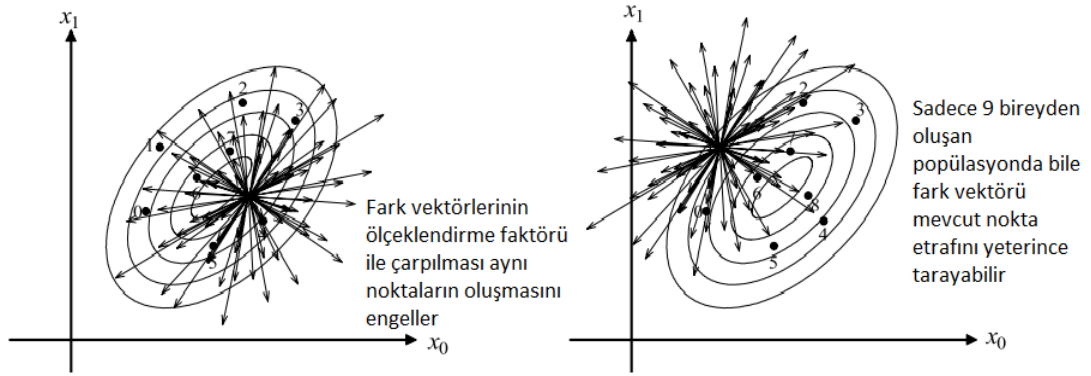
Diferansiyel gelişim algoritmasının en ayırt edici özelliği olan mutasyon işlemindeki fark vektörünün oluşumu ve arama uzayında dağılımı optimizasyon işlemin performansı açısından çok önem arz etmektedir. Bu noktaya açıklık getirmek açısından 9 bireyden oluşan örnek bir popülasyon ele alınabilir. Bu 9 bireyin oluşturabileceği tüm olası vektörler Şekil 3.21a'da verilmiştir. Bu fark vektörlerin tümü orijine taşındığında dağılımlar daha net görülebilmektedir (Şekil 3.21b) (Price ve ark. 2005).



Şekil 3.21 Dokuz bireyden oluşan örnek bir popülasyondaki olası tüm fark vektörleri (a) ve bu fark vektörlerinin orijine taşınmış dağılımları (b) (Price ve ark. 2005)

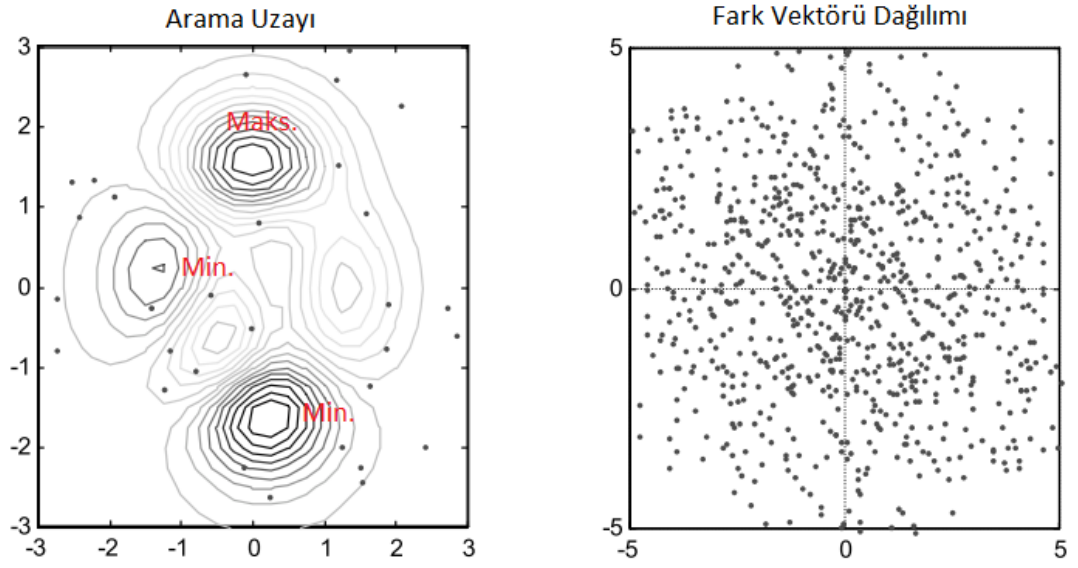
Tüm fark vektörlerinin zıt yönde bir eşleniğinin olduğu dikkate alındığında dağılımın ortalamasının sıfıra eşit olduğu rahatlıkla görülebilir. Fark vektörlerinin ölçeklendirme

faktörü ile çarpımı aynı noktaların oluşma olasılığını hemen hemen yok etmektedir (Şekil 3.22a). Fark vektörü dağılımının sadece 9 bireyden oluşan popülasyonda bile çok çeşitli ve mevcut nokta etrafını yeterince tarıyor olması optimizasyon süresince lokal bölgelere takılma ihtimalini azaltmakta ve yoğunluğun global bölgeye kaymasını sağlamaktadır (Şekil 3.22b) (Price ve ark. 2005).

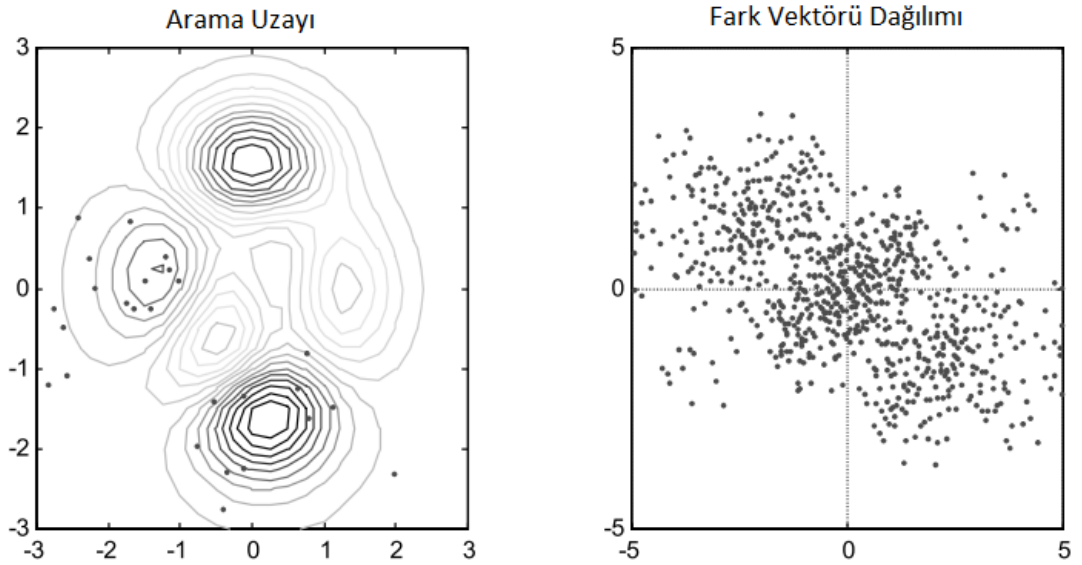


Şekil 3.22 Fark vektörlerine uygulanan ölçeklendirmenin etkisi (a) ve bu fark vektörlerinin mevcut nokta etrafını yeterince tarayabilme özelliği (b) (Price ve ark. 2005)

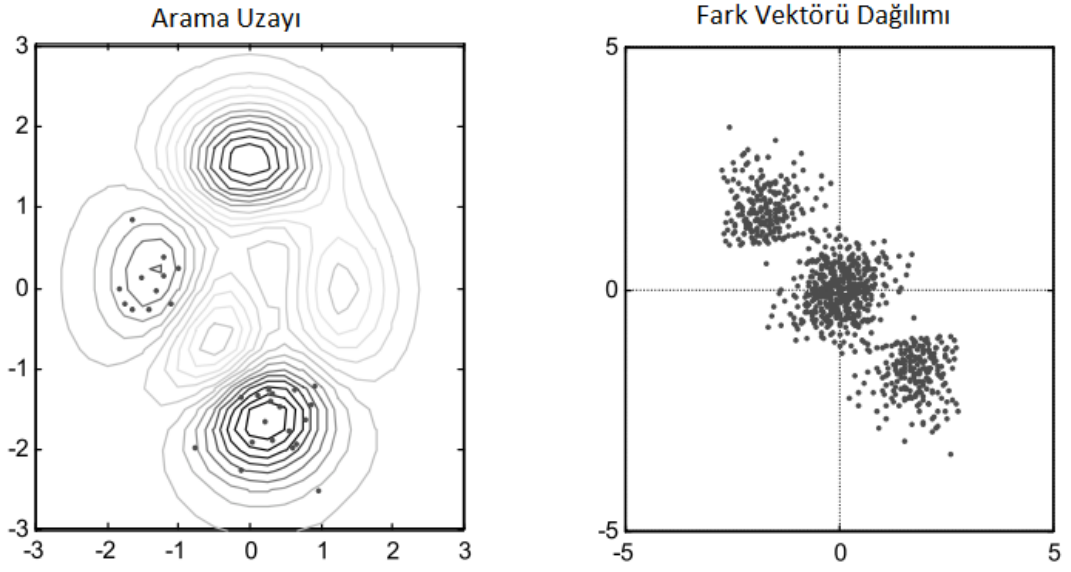
Fark vektörlerinin yukarıda bahsedilen özelliklerinin en önemli avantajlarından birisi hem adım büyüklüğü problemini ortadan kaldırması hem de amaç fonksiyonunun oluşturduğu arama uzayına otomatik olarak uyum sağlayabilmesidir. Şekil 3.23-2.29'dan lokal ve global optimum noktalara sahip bir problemde nesil sayılarına göre diferansiyel gelişim algoritmasının arama uzayında ilerleyişi ve özellikle fark vektörlerinin dağılımının optimizasyon süresince nasıl değişim gösterdiği açıkça izlenebilir (Price ve ark. 2005).



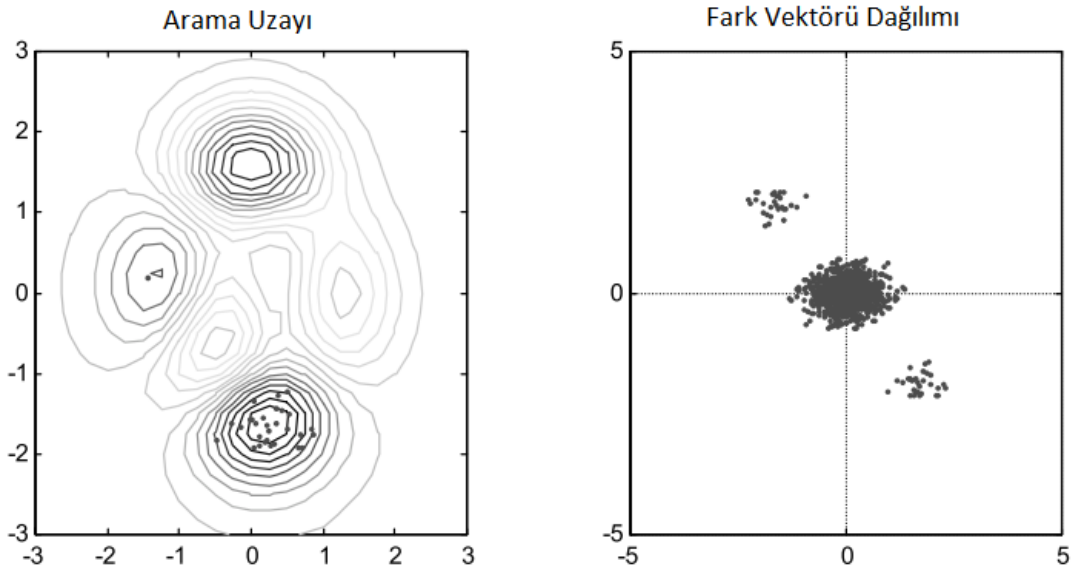
Şekil 3.23 Nesil 1: Diferansiyel gelişim algoritmasında popülasyon ve fark vektörleri dağılımı (Price ve ark. 2005)



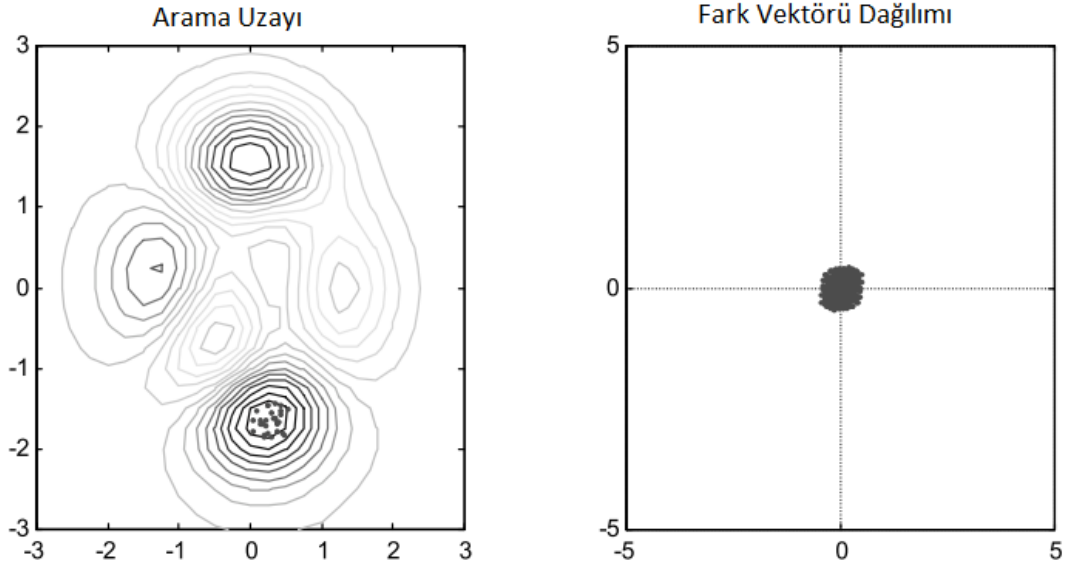
Şekil 3.24 Nesil 6: Popülasyonun iki minimum bölge etrafında yoğunlaşması (Price ve ark. 2005)



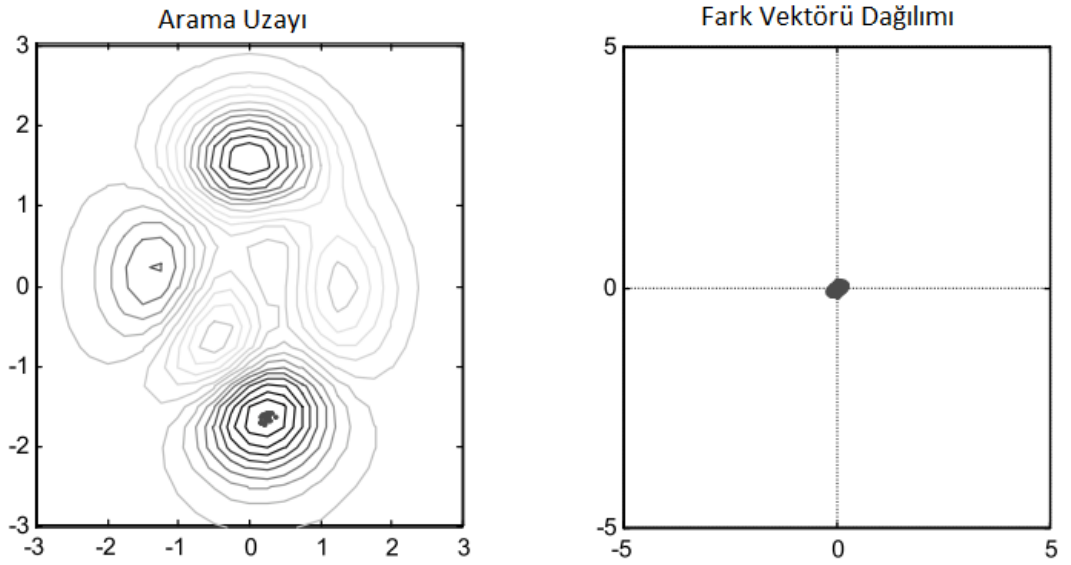
Şekil 3.25 Nesil 12: Fark vektörü bulutunun üç temel bölgede yoğunlaşması (Price ve ark. 2005)



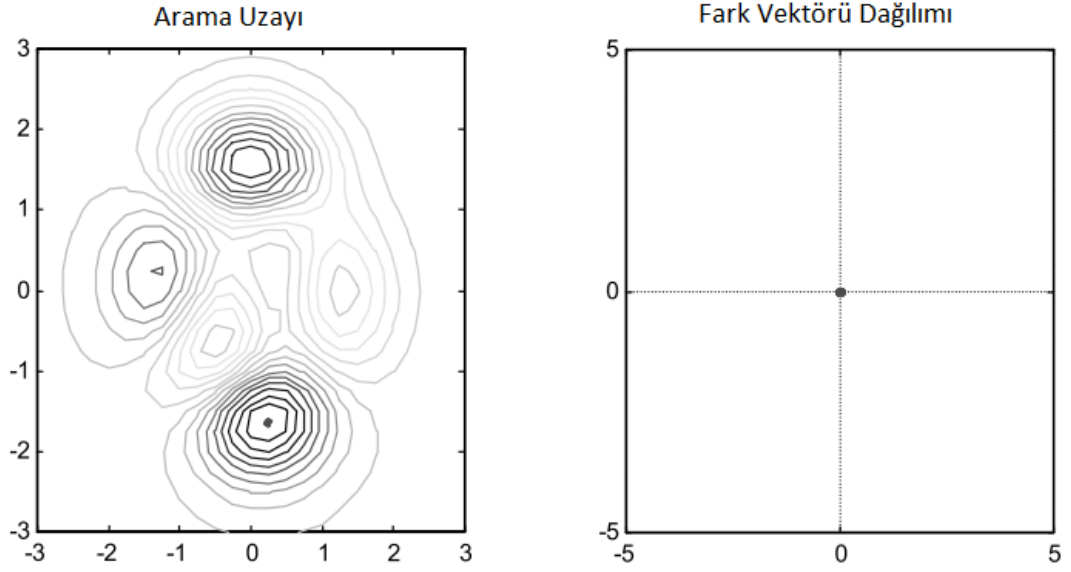
Şekil 3.26 Nesil 16: Popülasyonun global minimum noktada yoğunlaşması (Price ve ark. 2005)



Şekil 3.27 Nesil 20: Fark vektörlerinin otomatik olarak küçülmesi ve global minimum noktaya yakınsama işlemi (Price ve ark. 2005)



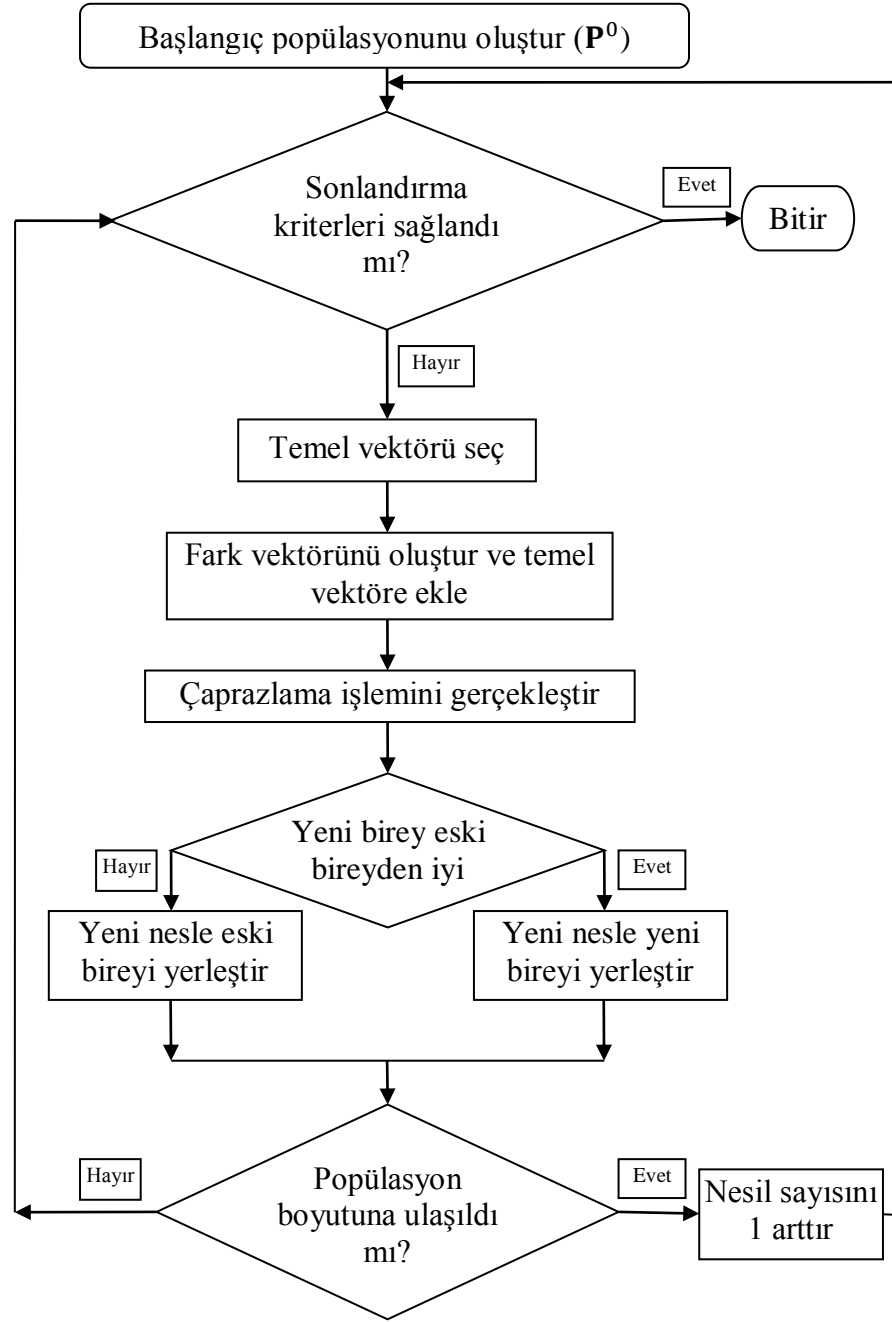
Şekil 3.28 Nesil 26: Fark vektörlerinin iyice küçülmesi ve global minimum noktaya yakınsama işlemi (Price ve ark. 2005)



Şekil 3.29 Nesil 34: Global minimum noktanın yakalanması (Price ve ark. 2005)

Yukarıda kabaca açıklanan klasik diferansiyel gelişim, başlangıçta kısıt içermeyen reel optimizasyon parametreleri (\mathbf{x}) ile tek-amaçlı bir fonksiyonun ($f(\mathbf{x})$) minimizasyonu için önerilmiştir. Doğal gerçek kodlama benimsenerek N_p adet bireyden oluşan bir popülasyonda işlemler gerçekleştirilmiştir. Diferansiyel gelişimde stratejiler, DE/x/y/z şeklinde simgelenmiştir. Burada; x temel vektörün nasıl seçildiğini, y (≥ 1) temel vektöre eklenen fark vektörü sayısını ve z ise mutasyon vektörü tarafından belirlenen parametre sayısının nasıl belirlendiğini temsil etmektedir.

Klasik diferansiyel gelişim iki aşamadan oluşmaktadır: başlangıç ve gelişim. Başlangıç aşamasında tıpkı diğer evrimsel algoritmalarda olduğu gibi bir başlangıç popülasyonu oluşturulur (\mathbf{P}^0). Daha sonra \mathbf{P}^0 popülasyonu gelişerek \mathbf{P}^1 popülasyonunu o da gelişerek \mathbf{P}^2 popülasyonunu oluşturur. Bu şekilde sonlandırma kriteri sağlanana kadar yeni popülasyonlar oluşturulmaya devam edilir. \mathbf{P}^n popülasyonundan \mathbf{P}^{n+1} popülasyonuna gelişim sağlanırken popülasyon içindeki bireylere üç evrimsel operasyon uygulanır. Bu operasyonlar; *diferansiyel mutasyon*, *çaprazlama* ve *seçim* operasyonlarıdır. Klasik diferansiyel gelişim algoritmasının akış diyagramı Şekil 3.30`da verilmiştir.



Şekil 3.30 Klasik diferansiyel gelişim algoritmasının akış diyagramı

Başlangıç aşamasında başlangıç popülasyonu \mathbf{P}^0 , N_p adet bireyden rastgele oluşturulur. Mutasyon operasyonunda bir \mathbf{P}^n popülasyonunda her ebeveyn $\mathbf{p}^{n,i}$ bireyi için bir mutant vektörü ($\mathbf{v}^{n+1,i}$) ve bir fark vektörü ($\mathbf{x}^{n+1,v,i}$) oluşturulur. Burada kalın yazılan \mathbf{x} 'in popülasyon içindeki tüm bireyleri temsil eden bir vektör olduğu unutulmamalıdır ($\mathbf{x} = x_1, x_2, \dots, x_N$). Fark vektörü $\mathbf{x}^{n+1,v,i}$ şu şekilde oluşturulmaktadır (Qing 2009):

$$\mathbf{x}^{n+1,v,i} = \mathbf{x}^{n,b,i} + \sum_{y \geq 1} \mathbf{F}_y(\mathbf{x}^{n,p_{1y}} - \mathbf{x}^{n,p_{2y}}), \quad 1 \leq i \neq p_{1y} \neq p_{2y} \leq N_P \quad (3.25)$$

Burada; $\mathbf{x}^{n,b,i}$ n 'nci popülasyonda i 'nci bireyin yerine oluşturulacak birey için seçilen (b) temel vektörünü, $\mathbf{x}^{n,p_{1y}}$ n 'nci popülasyonda $[1, N_P]$ arasındaki tamsayılardan rastgele seçilerek belirlenen p_{1y} 'nci bireyi, yine $\mathbf{x}^{n,p_{2y}}$ n 'nci popülasyonda $[1, N_P]$ arasındaki tamsayılardan rastgele seçilerek belirlenen p_{2y} 'nci bireyi, $\mathbf{F}_y [0,1]$ aralığında olacak şekilde y 'nci vektör farkı için belirlenen ölçeklendirme faktörünü ifade etmektedir.

$\mathbf{x}^{n,b,i}$ temel vektörü çeşitli şekillerde seçilebilir (Qing 2009);

- Mevcut vektör ile: $\mathbf{x}^{n,b,i} = \mathbf{x}^{n,i,i}$, ($b = i$),
- En iyi vektör ile: $\mathbf{x}^{n,b,i} = \mathbf{x}^{n,eniye,i}$, ($b = eniye$),
- Daha iyi vektör ile: $\mathbf{x}^{n,b,i} = \mathbf{x}^{n,dahaiye,i}$, ($b = dahaiye$) Burada seçim, rastgele seçilen bireyin mevcut bireyden daha iyi olacak şekilde seçime devam edilerek yapılmaktadır.
- Rastgele vektör ile: $\mathbf{x}^{n,b,i} = \mathbf{x}^{n,rastgele,i}$, ($b = rastgele$)

Mutasyon işlemi sonucunda elde edilecek yeni bireyin, belirlenen sınırlar ($[b_j^U, b_j^L]$) dışında oluşabilme ihtimali mevcuttur. Bu durumda genellikle bu bireyin amaç fonksiyonuna hedeften uzak değerler eklenerek cezalandırma işlemi uygulanmaktadır. Ancak bazı problemlerde sınırlar dışında oluşturulan bireyin amaç fonksiyonu hesaplanamayabilir. Bu nedenle mutasyon işlemi sonucunda oluşan uygun olmayan bireyler veya vektörler düzenlenmektedir. Genellikle iki tür düzenleme işlemi gerçekleştirilmektedir. Birinci yaklaşımda rastgele yeniden birey oluşturulmaktadır. Burada oluşan uygun olmayan optimizasyon parametreleri ile oluşan bireyler ($x_j^{n+1,v,i} \notin [b_j^U, b_j^L]$) arama uzayında rastgele üretilen uygun bireyler ile değiştirilir. Rastgele yeniden birey oluşturma işleminin sözde kodu Şekil 3.31'de verilmiştir (Qing 2009).

- İşlemi gerçekleştir: $j = 1, N$
 - o Şartlar sağlanıncaya kadar işlemi gerçekleştir ($x_j^{n+1,v,i} < b_j^L$ veya $x_j^{n+1,v,i} > b_j^U$):
 - $x_j^{n+1,v,i} = b_j^L + \alpha_j^i(b_j^U - b_j^L)$
 - o Dur.
- Dur

Şekil 3.31 Rastgele yeniden birey oluşturma işleminin sözde kodu

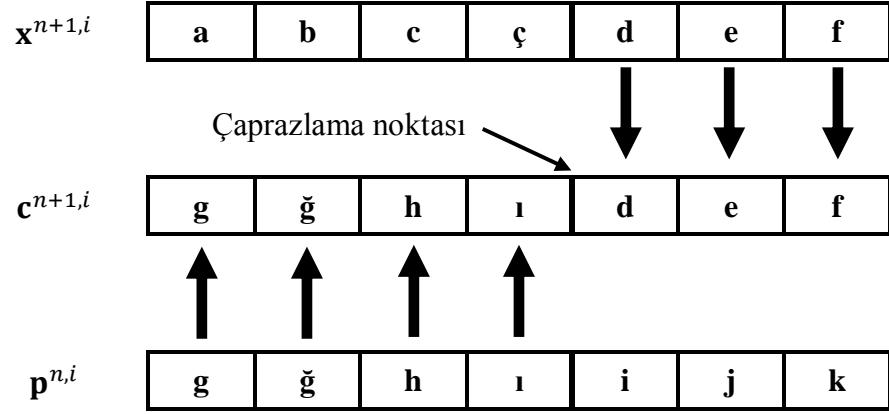
Mutasyon işlemi sonucunda oluşan uygun olmayan bireylerin düzenlenmesi için Qing (2009) kendini toparlama mekanizmasına uygun olarak çalışan ikinci bir yaklaşım önermiştir. Bu yaklaşım ile sınırların dışında oluşabilecek bireyler sınırların içine kaydırılmaktadır. Yaklaşımın sözde kodu Şekil 3.32`de verilmiştir (Qing 2009).

- İşlemi gerçekleştir: $j = 1, N$
 - o Şartlar sağlanıncaya kadar işlemi gerçekleştir ($x_j^{n+1,v,i} < b_j^L$):
 - $x_j^{n+1,v,i} = x_j^{n+1,v,i} + (b_j^U - b_j^L)$
 - o Dur.
 - o Şartlar sağlanıncaya kadar işlemi gerçekleştir ($x_j^{n+1,v,i} > b_j^U$):
 - $x_j^{n+1,v,i} = x_j^{n+1,v,i} - (b_j^U - b_j^L)$
 - o Dur.
- Dur.

Şekil 3.32 Kendini toparlama ile birey oluşturma işleminin sözde kodu

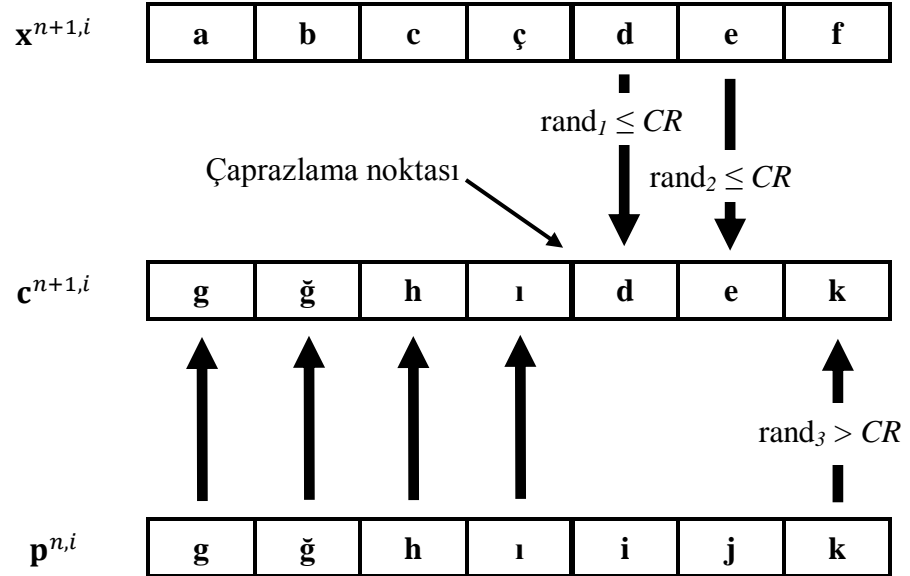
Çaprazlama operasyonunda mutasyon işleminde elde edilen yeni bireyler ($\mathbf{x}^{n+1,i}$) ile mevcut popülasyondaki bireyler ($\mathbf{p}^{n,i}$) rastgele seçilen çaprazlama noktasından kesilerek çocuk bireyler ($\mathbf{c}^{n+1,i}$) oluşturulur. Burada $\mathbf{p}^{n,i}$ anne bireyler olarak, $\mathbf{x}^{n+1,i}$ baba bireyler olarak adlandırılmaktadır. Ancak klasik diferansiyel gelişimde her çift sadece bir tane çocuk birey meydana getirebilmektedir (Şekil 3.33) (Qing 2009).

Şekil 3.33`de şematik gösterimi verilen çaprazlama işlemi tek-noktalı çaprazlama işlemi olarak adlandırılmaktadır. Seçilen çaprazlama noktası sayısına göre iki, üç ve n-noktalı çaprazlama işlemleri tanımlanmıştır. İki-noktalı çaprazlama işleminde bireyler 3 parçaya ayrılırken n-noktalı çaprazlama işleminde bireyler n+1 adet parçaya ayrılmaktadır.



Şekil 3.33 Çaprazlama işlemi

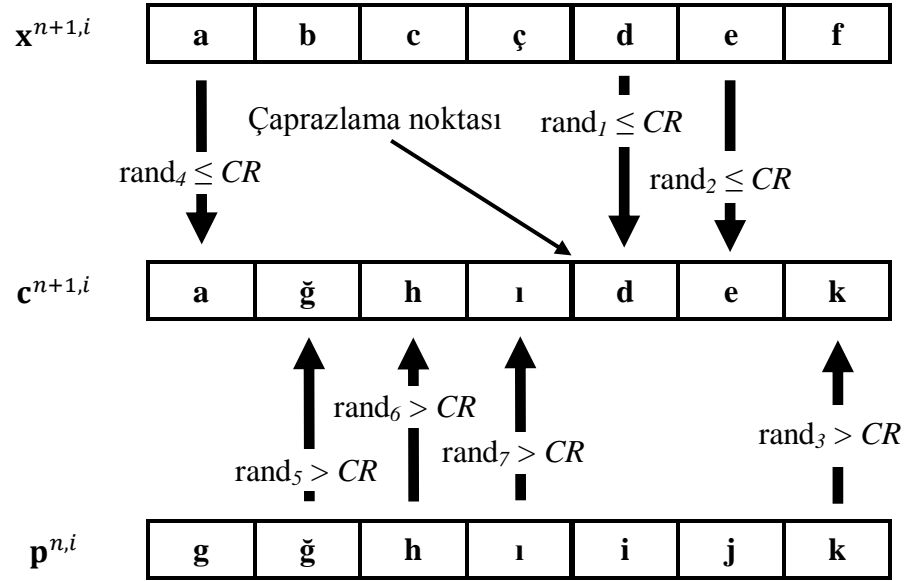
Üstel çaprazlama işleminde rastgele seçilen çaprazlama noktasından sonraki genler, $\text{rand}_j(0,1) \leq CR$ koşulu sağlanana kadar mutasyon işleminde elde edilen yeni bireylerden ($\mathbf{x}^{n+1,i}$) alınır, ilk $\text{rand}_j(0,1) > CR$ koşulu sağlandığında ise genler mevcut popülasyondaki aynı sıradaki bireylerden ($\mathbf{p}^{n,i}$) alınmaya başlanır. Böylece yeni çocuk bireyler ($\mathbf{c}^{n+1,i}$) oluşturulur (Şekil 3.34) (Qing 2009).



Şekil 3.34 Üstel çaprazlama işlemi

Üniform çaprazlama işleminde rastgele seçilen çaprazlama noktasından sonraki genler, $\text{rand}_j(0,1) \leq CR$ koşuluna bakılarak teker teker kıyaslanır. Eğer koşul sağlanırsa yeni gen mutasyon işleminde elde edilen yeni bireyden ($\mathbf{x}^{n+1,i}$) alınır, eğer koşul sağlanmazsa ($\text{rand}_j(0,1) > CR$) yeni gen mevcut popülasyondaki aynı sıradaki bireyden

($\mathbf{p}^{n,i}$) alınır. İşleme yeni çocuk bireyler ($\mathbf{c}^{n+1,i}$) oluşturuluncaya kadar devam edilir (Şekil 3.35) (Qing 2009).



Şekil 3.35 Üiform çaprazlama işlemi

Seçim operasyonunda ebeveyn-çocuk bireyleri arasında bir rekabet söz konusudur. Amaç fonksiyon değerlerine göre yarışmayı kazanan bir sonraki nesilde yerini alacaktır. Bu rekabet matematiksel olarak şu şekilde ifade edilmektedir:

$$\mathbf{p}^{n+1,i} = \begin{cases} \mathbf{c}^{n+1,i}, & \text{eğer uygunluk}(\mathbf{c}^{n+1,i}) > \text{uygunluk}(\mathbf{p}^{n,i}), \\ \mathbf{p}^{n,i}, & \text{aksi halde.} \end{cases} \quad (3.26)$$

Algoritmanın bitişini belirleyen sonlandırma kriteri olarak birçok düzenleme geliştirilmiştir. Bunlardan birkaçı belirli bir amacın karşılanması, amaç fonksiyonu hesaplama sayısının sınırlandırılması ve popülasyon çeşitliliğinin sınırlandırılması kriterleridir.

Belirli bir amacın karşılanması sonlandırma kriterinde amaçlar arasında; belirli sayıda en iyi amaç fonksiyonun farkının önceden belirlenen bir değerden küçük veya eşit olması, yeni üretilen çocuk bireyleri arasındaki amaç fonksiyon değerlerinin farklarının belirli bir değerden küçük veya eşit olması sayılabilir. Bu kriter sadece optimum değerinin bulunduğu optimizasyon problemlerinde uygulanabilmektedir.

Amaç fonksiyonu hesaplama sayısının sınırlandırılması kriteri, pratikte optimum sonuca ulaşmada sonsuz bir zamana sahip olunamadığı için belirli bir limit değere ulaşıldığında sonlandırmanın gerçekleştirilmesine dayanmaktadır. Ayrıca amaç fonksiyonu hesaplamasında zamandan daha çok amaç fonksiyonu hesaplama sayısı dikkate alınmaktadır. Birçok uygulamada hesaplama sayısından daha çok kullanımı daha basit olduğu için nesil sayısına sınırlama getirilmektedir. Dolayısıyla başlangıçta belirlenen nesil sayısına ulaşıldığında algoritma sonlandırılmaktadır.

Popülasyon çeşitliliğinin sınırlandırılması kriteri, popülasyondaki bireylerin ilerleyen aşamalarda zamanından erken birbirine benzemesi veya aynı olması durumuna karşı geliştirilen bir kriterdir. Burada çoğunlukla popülasyondaki bireylerin veya popülasyonun bir sonraki popülasyon ile benzerliğinin önceden belirlenen bir eşik değerden aşağıda kalması durumunda algoritmayı gereksiz hesaplamalardan kurtarmak amaçlanmaktadır.

Klasik diferansiyel gelişimin üç temel kontrol parametresi mevcuttur (Price ve ark. 2005, Qing 2009). Bunlar;

1. Popülasyon sayısı (NP),
2. Ölçeklendirme faktörü (F),
3. Çaprazlama oranı (CR)

dır. Mutasyon operasyonunda eğer birden çok fark vektörü kullanılıyorsa ölçeklendirme faktörü de tek bir tane alınmayıp fark vektörü ile aynı sayıda alınmaktadır (F_1, F_2, \dots, F_y).

Klasik diferansiyel gelişimde reel optimizasyon parametreleri kullanılarak tek bir amaç fonksiyonunun minimumu bulunmaya çalışılır. Ancak gerçekte birçok pratik uygulamada ortaya çıkan optimizasyon probleminde optimizasyon parametreleri ayrık, tamsayı, modal, karakter, vb. gibi çeşitli tiplerde olabilir. Örneğin bir mühendislik probleminde malzeme özellikleri seçiminin önceden mevcut olan bir malzeme kütüphanesinden gerçekleştirilmesi gerekebilir. Aynı şekilde bazı makine elemanlarının seçimi yine o elemanlarla ilgili önceden oluşturulmuş materyal kütüphanesinden gerçekleştirilebilir. Bu kütüphanedeki her bir materyale karşılık gelen bir tamsayı değeri

mevcut olabilir. Bu problemlerdeki optimizasyon parametreleri genellikle tamsayı olarak kullanılmaktadır. Ancak diferansiyel gelişimde fark vektörü reel sayı olacaktır. Bu aşamada tekrar tamsayıya çevirme işlemi gerçekleştirilebilir. Bu durumda diferansiyel gelişimi tamsayı olarak kodlamak düşünülebilir ancak popülasyon çeşitliliğinin azalması bir dezavantaj olarak ortaya çıkmaktadır. Diferansiyel gelişimde parametreleri ele alırken en avantajlı yaklaşım parametrelerin özelliklerine bakmaksızın tüm parametrelerin reel olarak kodlanmasıdır. İstendiği veya gerektiği takdirde reel parametreler yuvarlatılarak tamsayıya kolayca çevrilebilir. Bu sayede erken yakınsamanın ve popülasyon çeşitliliğinin azalmasının önüne geçilmiş olur.

3.8.1. Diferansiyel gelişim algoritmasının diğer yöntemlere göre zayıf yönleri ve bu zayıf yönleri gidermek için geliştirilen yaklaşımlar

Diferansiyel gelişim algoritması diğer evrimsel algoritmalar gibi yeni nesli oluşturmada uygun olmayan çocuk bireyleri öldürmek yerine üretilen çocuk bireyler içerisinde ebeveyn bireylerden daha iyi olanları seçip onları kabul ederek yıkıcı olmak yerine yapıcı bir tutum sergilemektedir. Ayrıca doğada bulunan doğal seleksiyon ve özellikle arılarda, karıncalarda ve kuşlarda görülen işbirliği diferansiyel gelişimde seçim işlemi ve diferansiyel fark vektörü ile gerçekleştirilmektedir. Bunlara ek olarak diğer evrimsel algoritmalara göre diferansiyel gelişim çok daha az sayıda kod satırı içermektedir. Bütün bu avantajlara rağmen diferansiyel gelişimin geliştirilmeye açık yönleri mevcuttur. Gürültülü problemlerde ve epistatik (farklı iki özelliğin başka bir özellikten kısmi olarak etkilenmesi) problemlerde diferansiyel gelişim düşük performans göstermektedir. Ancak tüm evrimsel algoritmaların epistatik problemlerde kötü performans özellikleri gösterdikleri bilinmektedir (Qing 2009).

Diferansiyel gelişim algoritması benzetimli tavlama, genetik algoritma ve parçacık sürü optimizasyonuna göre daha hızlı, programlanması kolay ve global arama yeteneğinin daha iyi olmasına karşılık bir takım zayıf yönleri mevcuttur. Bu zayıf yönlerin bazıları şunlardır:

- Parametrelerinin uygun değerlerde kullanılmaması sonucu düşük doğruluk değerine sahip olmak ve kolayca yakınsayamamak (Xu ve ark. 2007).

- Gürültülü problemlerde çok iyi yakınsama ve kesinlik gösterememek (Vesterstrom ve Thomsen 2004).

Storn ve Price (1995), diferansiyel gelişim algoritmasının performansını uyarlanabilir benzetimli tavlama yöntemi (Ingber 1993) ve tavllanmış Nelder-Mead yöntemi (Press ve ark. 1992) ile 9 test fonksiyonu üzerinden kıyaslamışlardır. Aralarında en basit üç test fonksiyonu haricindeki tüm fonksiyonlarda diferansiyel gelişim algoritması daha az sayıda fonksiyon hesabı ile global optimum noktaya ulaşmıştır. Burada ölçeklendirme faktörü (F) ve ilave kontrol değişkeni (λ) $[0,1]$ aralığında ve ($F, \lambda \geq 0.5$) olduğu durumlarda, ayrıca çaprazlama oranı (CR) yine $[0,1]$ aralığında olduğu durumlarda diferansiyel gelişim algoritması en iyi sonuçlara ulaşmıştır. Diferansiyel gelişim algoritmasının daha etkin kullanımı için popülasyon sayısının (NP) problemdeki bağımsız değişken sayısı (D) ile ilişkili olduğu ve popülasyon sayısının ($3D \leq NP \leq 10D$) aralığında seçilmesi gerektiği sonucuna varmışlardır. Diferansiyel gelişim algoritmasının diğer yöntemlere göre daha hızlı ve kesin olarak optimum noktaya ulaşmasının matematiksel bir ispatı olmamasına karşın üzerinde çalışmaların yapıldığı bilinmektedir. Ancak bu hızlı ve kesin yakınsama durumunun deneysel olarak bazı izahatları yapılabilir. Örneğin geçici vektörün sığ yüzeylerde genişleyerek o bölgeyi hızlı aşması ve dar bölgelerde ise daralarak bu bölgede yoğunlaşması ve bu sayede diğer yöntemlere göre diferansiyel gelişim algoritmasının kendi kendine uyarlama yapabilme özelliğinin ön plana çıktığı rahatlıkla söylenebilir. Diğer yandan kontrol değişkenlerinin problemin yapısına göre hangi aralıkta ve nasıl seçileceği konusu diferansiyel gelişim algoritmasının geliştirilmeye açık bir özelliğidir.

Birçok çalışmada belirli test problemleri için yöntemler kıyaslanmış ve bu yöntemlerin genellikle seçilen test problemleri için iyi sonuçlar verdiği gösterilmiştir. Ancak seçilen bu kısıtlı problemler için denenen algoritmalar daha geniş çaplı ve karşılaşılabilecek hemen hemen tüm kısıtlamaları içeren test problemlerinden oluşan setler ile denendiğinde set içindeki bazı problemlerde uygun sonuca ulaşamadıkları görülmektedir (Yao ve Liu 1999, Vesterstrom ve Thomsen 2004). Bu nedenle en geniş çapta test problemleri içeren setler ile algoritmaların performansının ölçülmesi algoritmalar hakkında daha objektif değerlendirmeler yapılmasına olanak sağlamaktadır.

Diferansiyel gelişim algoritmasının üç kontrol değişkeni olan popülasyon sayısı (NP), çaprazlama oranı (CR) ve ölçeklendirme faktörü (F); tüm test işlemi boyunca $NP = 100$, $CR = 0.9$ ve $F = 0.5$ olacak şekilde ele alınmıştır.

Sadece iki gürültülü test problemi dışında diferansiyel gelişim algoritmasının, parçacık sürü optimizasyonu ve evrimsel algoritmalara göre çok daha hızlı ve etkin olduğu ayrıca çözüm için her denemede aynı sonuçları bulma bakımından çok daha güvenilir olduğu sonucuna varmışlardır. İki gürültülü problemde diferansiyel gelişim algoritması diğer yöntemlerden verimlilik, etkinlik ve güvenilirlik açılarından daha kötü sonuçlar vermiştir. Hem 30 tasarım değişkeninde hem de 100 tasarım değişkeninde diferansiyel gelişim algoritması en kötü sonuçları vermiştir. Buradan anlaşılmaktadır ki belirtilen parametrelerde diferansiyel gelişim algoritmasının gürültülü problemlere karşı bir hassasiyeti vardır. Kontrol değişkenleri olan popülasyon sayısı (NP), çaprazlama oranı (CR) ve ölçeklendirme faktörü (F) ile oynanmamış ve sabit alınmıştır. Burada uygun parametrelerin belirlenmesi diferansiyel gelişim algoritmasının performansını arttırabilir.

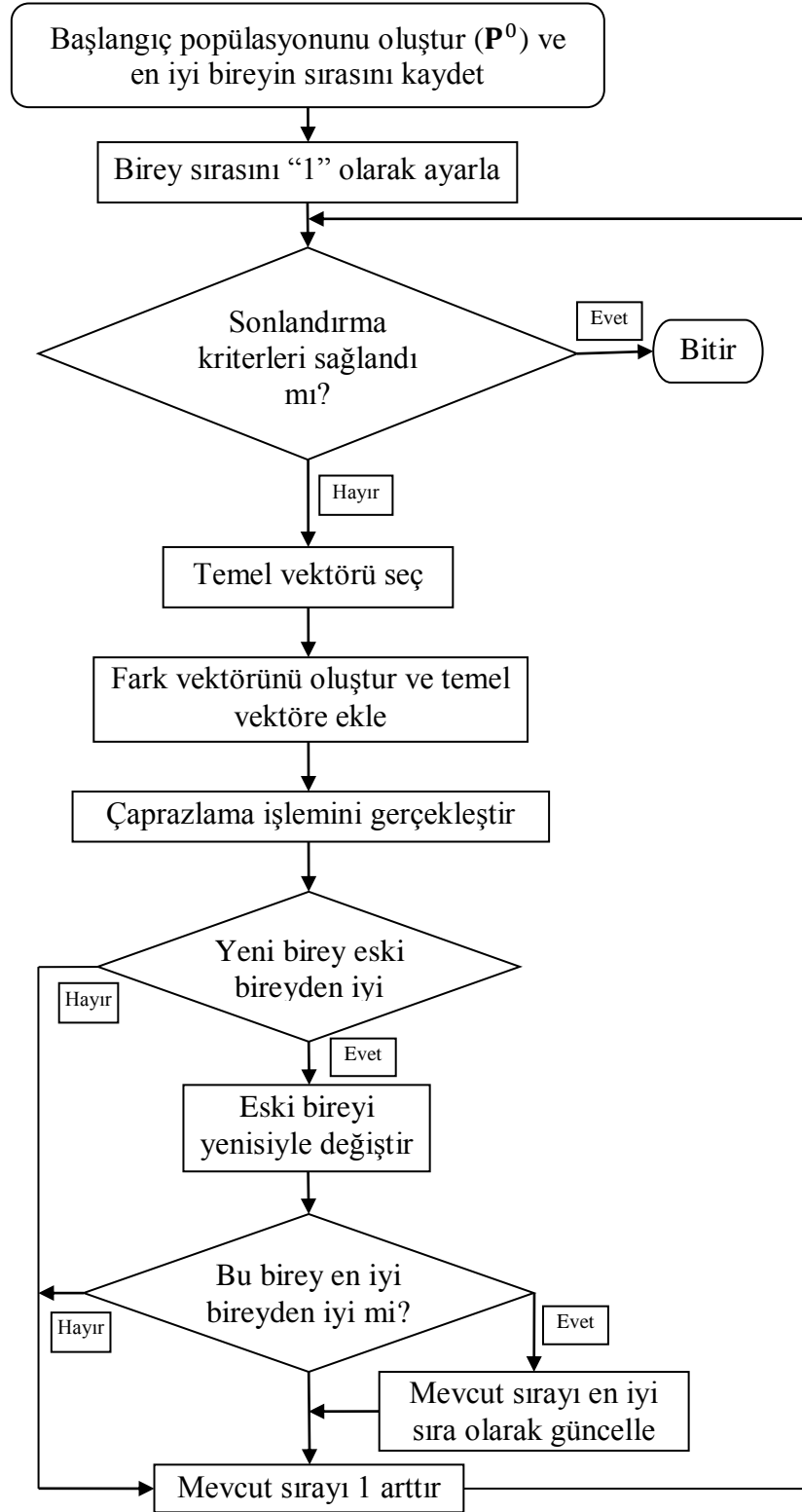
Krink ve ark. (2007), diferansiyel gelişim algoritmasının performansını banka faiz oranı sistemleri problemleri üzerinden genetik algoritma ve parçacık sürü optimizasyonu yöntemleriyle karşılaştırmışlar ve diferansiyel gelişim algoritması ile elde ettikleri sonuçların daha kesin ve tutarlı ayrıca güvenilir yani her denemede birbirine çok yakın hatta hemen hemen aynı sonuçlar elde ettiklerini vurgulamışlardır.

Diferansiyel gelişim algoritmasının ortaya çıkışından itibaren performansın arttırılması için birçok strateji geliştirilmiştir. Her strateji kendi özgün yapısına sahip olmakla birlikte geliştirilen diferansiyel gelişim algoritmaları belirli kriterlere göre sınıflandırılabilir. Qing (2009), diferansiyel gelişim algoritmasını temel gelişim mekanizmasına göre; klasik diferansiyel gelişim, dinamik diferansiyel gelişim, değiştirilmiş diferansiyel gelişim ve hibrit diferansiyel gelişim olmak üzere dört tipte sınıflandırmıştır. Klasik diferansiyel gelişim algoritması daha önceden açıklanmıştı, bu nedenle burada tekrar üzerinde durulmayacaktır.

3.8.2. Dinamik diferansiyel gelişim algoritması

Klasik diferansiyel gelişimde gelişim mekanizmasında popülasyon durumunun güncellenmesi işleminde yavaş cevap verme veya düşük yenilenebilme özelliği ve ekstra hafıza gereksinimine ihtiyaç duyulması gibi iki eksik yanı mevcuttur. Klasik diferansiyel gelişimin yapısı gereği popülasyon güncelleme statik kalmaktadır; şöyle ki n 'nci nesilde mevcut P^n popülasyonu $(n + 1)$ 'inci nesildeki P^{n+1} popülasyonu üretilip mevcut P^n popülasyonunun yerine geçene kadar değişmeden sabit kalmaktadır. Mevcut gelişim döngüsü içinde diğer bir deyişle aynı nesil içinde P^n popülasyonundan P^{n+1} popülasyonuna gelişim sürecinde klasik diferansiyel gelişim herhangi bir ilerleme göstermemektedir. Ayrıca mutasyon operasyonunda seçilecek temel vektör döngü tamamlanana yani bir nesil üretilene kadar her zaman P^n popülasyonu içinden seçilmektedir. Örneğin eğer seçilen temel vektörden daha iyi bireyler oluşursa bu bireyler veya vektörler hemen yeni nesile aktarılmakta böylece bir sonraki mutasyon operasyonunda kullanılamamakta ancak diğer nesilde kullanılabilir. Aynı şekilde mutasyon operasyonunda oluşturulan fark vektörleri de hep mevcut popülasyonda seçilen bireyler ile oluşturulmakta, daha iyi uygunluk değerine sahip bireyler yeni nesle aktarıldığı için bu yeni bireylerden oluşabilecek fark vektörleri ancak yeni nesilde oluşturulabilmektedir. Bu nedenle klasik diferansiyel gelişim popülasyon gelişimini hemen takip edememekte ancak belirli bir gecikme ile takip edebilmektedir. Bu da klasik diferansiyel gelişimde hem P^n popülasyonu hem de P^{n+1} popülasyonu için hafıza tahsis edilmesine ihtiyaç duyulmakta ve gelişim mekanizması ekstra hafıza harcamaktadır.

Yukarıda bahsedilen dezavantajların azaltılması ve algoritmanın geliştirilmesi için Qing (2009), Jacobi yöntemi ile çözülen diferansiyel denklem sisteminin Gauss-Seidel ile çözüldüğünde gereksiz işlemlerden kaçınabilme özelliğinden ve gereksiz hafıza kullanımını giderme özelliğinden esinlenerek dinamik diferansiyel gelişim algoritmasını önermiştir (Şekil 3.36).



Şekil 3.36 Dinamik diferansiyel gelişim algoritmasının akış diyagramı

Dinamik diferansiyel gelişim ile klasik diferansiyel gelişim arasında çok büyük bir fark gözükmemekle birlikte yapılan küçük değişiklik büyük öneme sahiptir. Burada nesil sayısı kaldırılarak sadece bir popülasyon ile çalışılmaktadır. Ancak popülasyon içerisinde oluşan yeni bireylerin uygunluk değerleri eski bireylerden iyi ise anında güncellenerek popülasyon değişikliğe uğratılmaktadır. Bu nedenle gereksiz hafıza kullanımının önüne geçilerek hafıza kullanımı hemen hemen yarı yarıya düşürülmektedir. Ayrıca klasik diferansiyel gelişim ile dinamik diferansiyel gelişim arasındaki en büyük farklardan birisi de en iyi bireyin dinamik bir şekilde anında güncellenebilmesidir. Yeni üretilen birey aynı sırada bulunan ebeveyn birey ile yarışmasının yanında ayrıca popülasyon içerisindeki en iyi birey ile de mücadele ettirilmektedir. Bu sayede eğer yeni bireyin uygunluk değeri mevcut en iyi bireyin uygunluk değerinden de iyi olursa bu durumda yine anında güncelleme yapılmaktadır.

Dinamik diferansiyel gelişim kapsamlı bir şekilde geliştiricisi Qing (2009) tarafından performans değerlendirmesine sokulmuş ve beklenildiği gibi klasik diferansiyel gelişime göre sonuca önemli ölçüde daha hızlı ve güvenilir bir şekilde ulaşmıştır.

3.8.3. Değiştirilmiş diferansiyel gelişim algoritması

Klasik diferansiyel gelişimin bazı problemler için yeterli olmadığını gören araştırmacılar performansı geliştirmek için klasik diferansiyel gelişimde çeşitli değişiklikler yaparak yeni algoritmalar geliştirmişlerdir. Bu değişiklikler popülasyonda, başlangıç işlemlerinde, diferansiyel mutasyonda, çaprazlamada, amaç ve kısıt fonksiyonu değerlendirmelerinde ve seçim işleminde yapılmıştır.

Çok-Popülasyonlu Diferansiyel Gelişim: Popülasyon diferansiyel gelişimin üzerinde işlem gerçekleştirildiği bir nesne olarak ele alınabilir. Klasik diferansiyel gelişim algoritmasında iki popülasyon (P^n, P^{n+1}) kullanılmaktadır, ancak bu popülasyonlar aslında aynı popülasyonun gelişiminde yararlanılan ve farklı aşamalarda hedef popülasyonu oluşturmada kullanılan popülasyonlardır. Bu nedenle klasik diferansiyel gelişim tek-popülasyonlu bir algoritmadır. Çeşitli çok-popülasyonlu diferansiyel gelişim stratejileri önerilmiştir. Bunlardan ilki *yedek popülasyon* kullanmaktır. Üretilen yeni çocuk bireyin ebeveyn birey haricinde popülasyon içindeki tüm bireylerden iyi olduğu durumlarda klasik diferansiyel gelişimde bu çocuk birey yerine ebeveyn birey

kullanılmaktadır. Üretilen uygunluk değeri çok iyi olan çocuk bireylerin heba edilmemesi açısından uygunluk değeri iyi olan çocuk bireylerin yeni bir yedek popülasyonda tutularak popülasyondaki diğer kötü uygunluk değerine sahip bireylerin yerine geçmesi önerilmiştir (Ali ve Törn 2004). Problem boyutunun tam olarak bilinmemesinden kaynaklanan bilgi eksiliği içeren optimizasyon problemlerinde popülasyon içinde belirli *gruplar* oluşturularak bu farklı gruplar ile gelişimi sağlamaya çalışmak da farklı bir yaklaşım olarak kullanılmıştır. Burada her grup, problemin farklı boyutları için arama gerçekleştirmekte ve gruplar kendi aralarında yarışmaktadırlar (Qing 2004). Birden fazla optimum bölgelere sahip problemlerde *altpopülasyon* yaklaşımı geliştirilmiştir. Bu yaklaşıma göre her optimum bölge için popülasyon bölünmekte ve o bölgeler için yerel arama gerçekleştirilmektedir. Arama süresince bazen bu altpopülasyonlar birbirleri ile bilgi alışverişinde bulunup gidişatlarını değiştirebilmektedirler. Bu yaklaşım paralel hesaplamaya çok uygundur. Bu yaklaşımlara ek olarak popülasyonda optimizasyon parametrelerinin tanım aralığına göre ters kullanan ve bu şekilde çeşitliliği arttırmaya çalışan *zıt-tabanlı* yaklaşımlar da geliştirilmiştir.

Rastgele Olmayan Başlangıç İşlemi: Klasik diferansiyel gelişimde başlangıç popülasyonu genellikle rastgele oluşturulmaktadır. Ancak problem hakkında önceden bir bilgi birikimi varsa veya hangi bölgelere daha fazla ağırlık verilebileceği biliniyorsa bu durumda başlangıç popülasyonu taraflı bir şekilde istenen bölgelere daha fazla birey gelecek şekilde oluşturulabilir.

Farklı Biçimlerdeki Diferansiyel Mutasyon: Diferansiyel gelişim algoritmasında mutasyonun kilit rol oynadığı bilinmektedir, bu nedenle en çok mutasyon üzerinde durulmuş ve bunun üzerinde birçok çalışma yapılmıştır. Pertürbasyon mutasyonda zamanından önce yakınsamayı engellemek için fark vektörüne bağımsız bir gürültü terimi eklenmektedir (Chen 2002). Trigonometrik mutasyonda (Fan ve Lampinen 2003) üç fark vektörü en iyi şekilde oluşturularak işlem gerçekleştirilmektedir. Çoklu mutasyonda (Wu ve ark. 2006) birden fazla fark vektörü oluşturulmaktadır. Bu mutasyonların yanında iyi bireylerin mutasyona uğrama olasılığını arttıran üniform olmayan mutasyon (Sarimveis ve Nikolakopoulos 2005) ve bir vektöre uygulanan sadece bir fark vektörü yerine muhtemel fark vektörleri seti içerisinde seçilen en uygun

fark vektörünü kullanan diferansiyelden bağımsız mutasyon işlemi (Ali ve Fatti 2006) geliştirilmiştir.

Özel Çaprazlama İşlemleri: Çaprazlama işlemi genetik algoritmalarda en etkin ve üzerinde en çok çalışılan operasyon olmakla birlikte diferansiyel gelişim algoritmasında mutasyon operasyonu aynı önemi kazanmıştır. Diferansiyel gelişimde çaprazlama işlemi üzerinde genetik algoritmalar kadar durulmamış ve biraz ihmal edilmiştir. Klasik diferansiyel gelişim algoritmasında mutasyon sonucunda oluşturulan vektör ($\mathbf{v}^{n+1,i}$) anne vektör ($\mathbf{p}^{n,i}$) ile eşleşmekte ve çocuk vektör ($\mathbf{c}^{n+1,i}$) oluşturulmaktadır. Bu tertip iki şekilde değiştirilebilmektedir; birinci yaklaşımda eşleşen partner değiştirilmekte, ikinci yaklaşımda ise eşleşme sonucunda oluşan çocuk sayısı değiştirilmektedir. Birinci yaklaşımda değişen eş anne vektör olmakta ve çoğunlukla en iyi vektör (birey) ($\mathbf{p}^{n,eniye}$) anne vektör olarak eşleşmeye maruz bırakılmaktadır (Babu ve Angira 2003). İkinci yaklaşımda ise genellikle çocuk sayısı olarak iki alınmaktadır.

Amaç ve Kısıt Fonksiyonu Değerlendirmede Bazı Yaklaşımların Kullanılması: Birçok pratik uygulamada özellikle mühendislik problemlerinde amaç ve kısıt fonksiyonlarının hesaplanması zaman alıcı ve uğraştırıcı olmaktadır. Bu nedenle bazı araştırmacılar sonucun hassasiyetine etki etmeksizin gelişim işleminin bazı aşamalarında amaç ve kısıt fonksiyonlarının kolay hesaplanması için bazı yaklaşımlar geliştirmişlerdir (Pahner ve Hameyer 2000).

Alternatif Seçim Mekanizmaları: Klasik diferansiyel gelişimde anne-çocuk rekabetinde her çocuk ($\mathbf{c}^{n+1,i}$) kendi annesiyle ($\mathbf{p}^{n,i}$) yarışmaktadır. Eğer çocuk ($\mathbf{c}^{n+1,i}$) annesinden ($\mathbf{p}^{n,i}$) daha iyi uygunluk değerine sahip değilse popülasyon içindeki diğer tüm bireylerden daha iyi uygunluk değerine sahip olsa bile ($\mathbf{p}^{n,i} = \mathbf{p}^{n,eniye}$) elenmektedir. Benzer olarak anne birey ($\mathbf{p}^{n,i}$) popülasyondaki diğer tüm bireylerden daha iyi uygunluk değerine sahip olsa bile ($\mathbf{p}^{n,i} = \mathbf{p}^{n,eniye}$) eğer çocuk bireyin ($\mathbf{c}^{n+1,i}$) uygunluk değerinden daha iyi değilse ($\mathbf{c}^{n+1,i} > \mathbf{p}^{n,i} = \mathbf{p}^{n,eniye}$) elenecektir. Böylece ilerleyen aşamalarda bireyler birbirine çok benzeyecektir ve popülasyon çeşitliliği risk oluşturabilecektir. Buna engel olmak için seçim işleminde birçok farklı yaklaşım önerilmiştir. Çapraz Seçim işleminde üretilen çocuk birey ($\mathbf{c}^{n+1,i}$) yine annesi ($\mathbf{p}^{n,i}$) ile yarıştırmakta ancak yarışı kazanması sonucunda elenen anne birey olmamakta, ya

popülasyon (P^n) içerisinde rastgele seçilen ve uygunluk değeri çocuk bireyinkinden düşük olan birey elenmekte ya da popülasyon içindeki en kötü birey ($p^{n,enkötü}$) elenmektedir. *Grup Seçim* yaklaşımı Noman ve Iba (2006) tarafından uygunluk değeri iyi olan bireylerin eleme işlemine maruz bırakılmaması ve popülasyon içinde neslini devam ettirebilmesi amacıyla uygunluk değeri iyi olan ve yeni üretilen çocuk bireylerden oluşan yeni bir birlik oluşturarak bu birey topluluğunun yeni nesillere aktarılması işlemidir. Ancak bu yaklaşımda ek bir sıralama işlemi yapılması gerektiğinden birey sayısı arttıkça işlem süresi artmakta ve hesapsal açıdan yoğunluk yaşanmaktadır. *Benzerlerin seçimi* işleminde popülasyon çeşitliliğini arttırmak için üretilen çocuk birey annesiyle yarıştırmamakta onun yerine çocuk bireye çok yakın bireyler seçilerek bu bireyler içerisinde çocuk bireye en yakın birey yarışa dahil edilmektedir. Ancak burada benzer bireyleri bulmak için yapılan tüm popülasyon içindeki yakın vektörleri hesaplama işlemi hesapsal açıdan ek bir yük oluşturmaktadır (Thomsen 2004). *Eşik değerlerde seçme* işlemi diferansiyel gelişimin çok iyi sonuç veremediği gürültülü problemler için Das ve ark. (2005d) tarafından önerilen bir yaklaşımdır. Buna göre gürültünün şiddetine göre çocuk birey ($c^{n+1,i}$) ancak belirli eşik değerlerin belirttiği sınırlar içerisinde annesinin ($p^{n,i}$) yerine geçebilmektedir.

3.8.4. Hibrit diferansiyel gelişim algoritması

Diferansiyel gelişim algoritması verimlilik ve hızlı yakınsama açısından kendini evrimsel hesaplama alanında göstermesine rağmen araştırmacılar algoritmanın global arama yeteneğini geliştirmek için diferansiyel gelişimi çeşitli kural tabanlı (deterministik) ve olasılıksal (stokastik) algoritmalar ile güçlendirmeye çalışmışlardır. Genel olarak diferansiyel gelişim gömülü ve ardışık olmak üzere iki şekilde hibritleştirilmiştir. Gömülü yaklaşımda bir optimizasyon algoritması diferansiyel gelişim algoritmasının içine gömülürken ardışık mekanizmada bir optimizasyon algoritması ya diferansiyel gelişim algoritmasından önce ya da sonra gelerek hibrit yapı oluşturulmaktadır. Ardışık yaklaşımda bir optimizasyon algoritmasının sonucunda elde edilen veriler diğer optimizasyon algoritması için başlangıç verileri olmaktadır. Böylece bir algoritma bitmeden diğeri başlamamakta, işlemler sırasıyla sürdürülmektedir.

Kural Tabanlı Optimizasyon Algoritmaları ile Geliştirilmiş Diferansiyel Gelişim:

Genellikle kural tabanlı optimizasyon algoritmaları lokal aramalarda çok başarılıdır. Bu nedenle diferansiyel gelişim algoritmasının lokal arama özelliğini geliştirmede ilk seçenek kural tabanlı bir yöntem ile hibritleştirme yapmaktır. Bu alanda öncü çalışmalar Wang ve ark. (1998), Chiou ve Wang (1998) tarafından yapılmış ve türev-tabanlı hibrit diferansiyel gelişim algoritması önermişlerdir. Önerdikleri hibrit algoritmada mevcut evrimsel operasyonlardan sonra dik-iniş tabanlı bir lokal arama gerçekleştirmişler ve buna ek olarak erken yakınsamadan kaçınmak için en iyi uygunluk değerine sahip birey etrafında bir göç ettirme operasyonu uygulamışlardır. Diferansiyel gelişim algoritması ile kural tabanlı algoritmalar arasında birçok hibritleştirme işlemi gerçekleştirilmiştir. Bu algoritmalar Çizelge 3.2`de özetlenmiştir.

Çizelge 3.2. Kural tabanlı optimizasyon algoritmaları ile hibritleştirilmiş diferansiyel gelişim algoritmaları (Qing 2009)

Kural Tabanlı Optimizasyon Algoritmaları	Geliştiriciler
Simpleks yöntemi	Rogalsky ve Derksen (2000), Bhat ve ark. (2006), Nasimul ve Hitoshi (2008)
Newton-Raphson yöntemi	Crutchley ve Zwolinski (2002)
Dikdörtgen bölme algoritması	He ve Narayana (2002)
Ayarlanabilir kontrol ağırlıklı gradyan yöntemi	Lopez ve ark. (2003)
Davidon-Fletcher-Powell yöntemi	Colaço ve ark. (2003a, 2003b, 2003c, 2004)
Sanki-Newton algoritması	Babu ve Angira (2004)
Birleşik uygun yön bulma yöntemi	Chang ve Wu (2004)
Powell`in yön seti yöntemi	Xu ve Dony(2004)
Sıralı ikinci-dereceden programlama	Coelho ve Mariani (2006a), Menon ve ark. (2006), Menon ve ark. (2007)
Değiştirilmiş Newton-Raphson yöntemi	Li ve ark. (2005)
Levenberg-Marquardt iniş yaklaşımı	Bluszcz ve Adamiec (2006), Chen ve ark. (2002)
Kaba setler teorisi	Hernandez-Diaz ve ark. (2006)
Çok-modlu soldan değiştirme	Lorenzoni ve ark. (2006)

Markov zinciri parçacık filtreleme	Du ve Guan (2006)
Değişen yakın çevrede arama yöntemi	(Qian ve ark. 2006)
Kapalı filtreleme algoritması	Coelho ve Mariani (2006b)
Gauss-Newton algoritması	Xing ve Xue (2007)

Olasılıksal Optimizasyon Algoritmaları ile Geliştirilmiş Diferansiyel Gelişim: Kural tabanlı optimizasyon algoritmalarından farklı olarak olasılıksal optimizasyon algoritmaları diferansiyel gelişim algoritmasının gelişim sürecini değişikliğe uğratmaktadır. Örnek olarak genetik algoritmalarda üç operasyon seçim, çaprazlama ve mutasyon olarak sıralanmaktadır. Ancak diferansiyel gelişimde operasyon sırası mutasyon, çaprazlama ve seçim olarak değişiklik göstermektedir ve genetik algortmada çaprazlama nasıl önemli bir etkiye sahipse diferansiyel gelişimde de mutasyon o derece önemli bir etkiye sahiptir. Birçok araştırmacı diferansiyel gelişim algoritması ile diğer olasılıksal algoritmaları hibritleştirerek yeni algoritmalar geliştirmişlerdir. Literatürde mevcut geliştirilen algoritmalar Çizelge 3.3`te verilmiştir.

Çizelge 3.3. Olasılıksal optimizasyon algoritmaları ile hibritleştirilmiş diferansiyel gelişim algoritmaları (Qing 2009)

Olasılıksal Optimizasyon Algoritmaları	Geliştiriciler
Parçacık sürü optimizasyonu	Hendtlas (2001), Parsopoulos ve Vrahatis (2002), Zhang ve Xie (2003), Kannan ve ark. (2004), Phan ve ark. (2004), Talbi ve Batouche (2004), Das ve ark. (2005b), Das ve ark. (2005c), De ve ark. (2005a), Liu ve ark. (2005), Moore ve Venayagamoorthy (2006), Zheng ve Qian (2006)
Olasılıksal dik iniş	Magoulas ve ark. (2001)
Evrimsel strateji	Yang ve ark. (2001)
Monte Carlo yöntemi	Weber ve Burgi (2002), Weber (2005) Ter Braak (2006)
Eşik kabul etme algoritması	Schmidt ve Thierauf (2002), Schmidt ve Thierauf (2005)
Genetik algoritmalar	Abbass (2002), Abbass ve Sarker (2002), Lü ve ark. (2003), Das ve ark. (2005a),

	De ve ark. (2005b), Kaelo ve Ali (2007)
Çoklu başlangıç yöntemi	Laskari ve ark. (2003)
Karıncı kolonisi optimizasyonu	Chiou ve ark. (2004), Korosec ve ark. (2007), Wang ve ark (2006)
Kültürel algoritma	Becerra ve Coello (2004a), Becerra ve Coello (2004b), Becerra ve Coello (2006a), Becerra ve Coello (2006b), Chong ve Tremayne (2006)
Dağılımın tahmini algoritması	Sun ve ark. (2005)
Ekstrem öğrenme makinesi	Zhu ve ark. (2005)
Benzetimli tavlama	Kong ve ark. (2006), Subramanian ve ark. (2006), Yan ve ark. (2006)
Yapay bağışıklık sistemi	He ve Han (2007)
Yoğunlaşma algoritması	Yalçın ve Gökmen (2006)
Dağınık arama	Davendra ve G. Onwubolu (2007)

3.8.5. Diferansiyel gelişim algoritmasının performansını etkileyen parametreler

Diferansiyel gelişim algoritması diğer yöntemlere göre henüz çok daha yenidir ve bu yöntemin geliştirilmeye açık yönleri mevcuttur. Brest ve ark. (2007), diferansiyel gelişim algoritmasının performansına etki eden parametreleri üçe ayırmıştır. Bunlar;

1. Diferansiyel Ölçeklendirme Faktörü (F),
2. Çaprazlama Oranı (CR),
3. Popülasyon Sayısı (NP)

dır.

Literatürdeki çalışmalarda kontrol parametreleri ayarlamaları iki farklı şekilde yapılmaktadır: parametre ayarlaması ve parametre kontrolü. Çok daha yaygın olarak kullanılan ilk yaklaşımda optimizasyon probleminden önce parametrelerin uygun değerleri bulunmakta ve problem buna göre ayarlanmaktadır; ayrıca problem süresince bulunan parametre değerleri sabit kalıp herhangi bir değişikliğe maruz bırakılmamaktadır. İkinci yaklaşımda ise parametre değerleri optimizasyon süreci esnasında değişmektedir. Klasik diferansiyel gelişim algoritmasında kontrol

parametreleri başlangıçta sabit alınarak işlem boyunca bir değişikliğe uğramamaktadır. Kontrol parametrelerinin probleme bağlı olarak başlangıçta en uygun değerlerinin bulunması ve bu değerler ile işleme devam edilmesi hatta buna ek olarak bu parametrelerin işlem esnasında en uygun değerleri olarak değişebilmesi konusu üzerinde bir bilgi eksiliği ve açık bir alan mevcuttur. Kontrol parametrelerinin probleme bağlı olarak başlangıçta en uygun değerlerinin bulunması sürecinde problemlerle karşılaşılmaktadır. Belirlenen problem hakkında çözüme başlamadan önce hangi parametre değerlerinin kullanılması gerektiği bilinmediğinden genellikle daha önce tecrübe ile bulunmuş ve farklı bir problemde olumlu sonuçlar vermiş parametre değerleri kullanılmaktadır. Bazen de bunun yerine daha önceden bir bilgi birikimi yoksa el ile bazı parametre ayarlamalarının yapılması yani bazı deneme-yanılmalar yaparak problemi o parametreler ile test etmek gerekmektedir.

Babu ve Chaturvedi (2000a), diferansiyel gelişim algoritmasının kontrol parametrelerinin en uygun değerlerini tespit etmek için örnek bir problem üzerinde $NP = \{5, 10, \dots, 100\}$, $F = \{0,4, 0,5, 0,6, 0,7\}$ ve $CR = \{0,3, 0,4, \dots, 0,9\}$ olacak şekilde problemi birçok defa çözmüşlerdir. Çözümler sonucunda, optimum kontrol parametrelerinin $NP = 20$, $F = 0,4$, $CR = 0,6$ olduğunu tespit etmişlerdir. Ancak burada şunu vurgulamak gerekir ki; tespit edilen kontrol parametre değerleri en optimum kontrol parametre değerleri olmaktan çok önceden belirlenen değerler arasındaki en uygun kontrol parametre değerleridir; çünkü burada kontrol parametreleri, önceden belirlenen sınırlı sayıda değerler arasından seçilen değerleri alabilmektedir. Bu değerler daha geniş bir aralıktan seçilebilirdi, ihtimal dâhilindedir ki daha iyi sonuçlar verebilen parametre değerleri bu sınırlar dışında mevcut olabilir. Başka bir çalışmada Babu ve Munawar (2000b), kontrol parametre değerlerini daha geniş bir aralıkta ve çeşitlilikte ($NP = \{10, 15, \dots, 150\}$, $F = \{0, 0,1, \dots, 1\}$ ve $CR = \{0, 0,1, \dots, 1\}$) olarak en uygun kontrol parametre değerlerini bulmaya çalışmışlardır. Buldukları en uygun değerler ($NP = 25$, $F = 0,1$ ve $CR = \{0, 0,1, \dots, 1\}$) ile elde ettikleri sonuçları genetik algoritmalar kullanarak ($NP = \{72, 73, \dots, 224\}$, $CR = 0,8$, *Mutasyon Oranı* = 0,1) elde ettikleri sonuçlar ile karşılaştırmışlar ve daha iyi sonuçlar bulduklarını belirtmişlerdir. Ancak burada vurgulamak gerekir ki her yöntemin en optimum kontrol parametre değeri problemin yapısına göre değişebilir. Bu nedenle daha objektif bir değerlendirme için genetik algoritmalarındaki iki kontrol parametre değerinin (çaprazlama

ve mutasyon oranı) sabit deęerler yerine diferansiyel gelişimdeki gibi en uygun deęerlerin bir küme içinden seçilerek bulunması sonucunda elde edilerek çözüm yapmak ve bu şekilde bulunan sonuçları kıyaslama yapmak için kullanmak daha uygun olabilir. Ayrıca her kontrol parametresi deęeri ile sadece bir çözüm yapmak yerine birden fazla sayıda çözüm gerçekleştirmek güvenilirlik açısından daha uygun olabilir.

En uygun kontrol parametre deęerlerini bulmak için bir başka yaklaşım sadece bir parametre deęerinin sabitlenip dięerlerinin en uygun deęerlerini bulma işlemidir. Sabitlenen kontrol parametresi genellikle popülasyon sayısı olmaktadır. Nearchou (2006), Nearchou ve Omirou (2006) ve Zaharie (2002) sadece bir tane kontrol parametresini sabitleyerek en uygun parametre deęerlerini bulmak için çeşitli çalışmalar yapmışlardır.

En uygun kontrol parametre deęerlerini bulmak için kullanılan en yaygın yaklaşımlardan birisi de iki kontrol parametre deęerini sabitleyerek üçüncü kontrol parametresinin en uygun deęerini bulma yaklaşımıdır.

Diferansiyel gelişimin performansının kontrol parametrelerine çok hassas bir şekilde baęlı olması nedeniyle kontrol parametre deęerlerinin başlangıçta en uygun sabit deęerler alması yerine farklı bir yaklaşım olarak optimizasyon sürecinin başlangıcında farklı, optimizasyon sürecinin sonlarında ise farklı deęerler alması sağlanabilmektedir. Burada çeşitlilik ve yakınsama açısından başlangıçta popülasyon sayısı ve diferansiyel ölçeklendirme parametresi büyük, çaprazlama oranı düşük alınmakta, optimizasyon sürecinin sonlarında ise gereksiz işlemlerden kaçınarak daha verimli bir işlem sürdürebilmek için popülasyon sayısı ve diferansiyel ölçeklendirme parametresi düşük, çaprazlama oranı ise büyük alınmaktadır. Bu işlemi kolaylaştırmak ve etkin şekilde kullanmak için çeşitli yöntemler geliştirilmiştir. Genellikle bu yöntemler *rastgele*, *kural-tabanlı*, *uyarlamalı* ve *kendi kendine uyarlamalı* olmak üzere dört gruba ayrılmaktadır. Eiben ve ark. (1999) ise bu yöntemleri kural-tabanlı, uyarlamalı ve kendi-kendine uyarlamalı olmak üzere üç farklı türde sınıflandırmışlardır.

Rastgele yöntem en basit yaklaşım olmakla birlikte burada popülasyon sayısının rastgele atanmasında programlama güçlüğü çekildiğinden diferansiyel ölçeklendirme faktörü ve çaprazlama oranının rastgele seçilmesi ile işlem gerçekleştirilmektedir. Her

nesilde bu kontrol parametreleri önceden belirlenen aralıklar dâhilinde eşit dağılımlı, Gauss dağılımı veya herhangi bir dağılım ile rastgele seçilmektedirler.

Kural-tabanlı yöntemlerde (deterministik) adından anlaşılacağı üzere parametre değerleri belirleyici bir kurala göre seçilmekte ve değiştirilmektedir. Diferansiyel gelişimin geliştiricileri (Storn 1996b) mutasyondaki ölçeklendirme faktörünün çaprazlama oranına göre optimum sonuca ulaşmada çok daha hassas olduğunu gözlemlemişlerdir. Çaprazlama oranının daha çok ince ayar yapmada kullanılabileceğini belirtmişlerdir. Ayrıca başlangıç olarak popülasyon sayısının 10 ($NP = 10D$), ölçeklendirme faktörünün 0,8 ($F = 0,8$) ve çaprazlama oranının 0,9 ($CR = 0,9$) alınmasının iyi bir seçim olacağını vurgulamışlardır. Burada D tasarım değişkeni sayısıdır. Bununla birlikte Qing (2009), başlangıçta ölçeklendirme faktörünün 0,5 ($F = 0,5$) ve çaprazlama oranının 0,1 ($CR = 0,1$) alınması ile elde edilecek başlangıç değerlerinin daha iyi olabileceğini iddia etmektedir. Algoritma geliştiricileri yine popülasyon sayısının en küçük katsayı değerinin 1,2 ve en büyük katsayı değerinin ise 30 olarak alınabileceğini ancak en makul seçimin 5-10 aralığında ($5D \leq NP \leq 10D$) olabileceğini iddia etmişlerdir. Literatürde araştırmacılar ölçeklendirme faktörünü çeşitli aralıklarda ([0,2], [0,1], [-1,1], [0,4,1], [0,5,1]) almışlar, ancak çaprazlama oranı için genelde [0,1] aralığını tercih etmişlerdir. Bazı araştırmacılar çaprazlama oranı için aralığı daraltarak [0,5,1] veya [0,8,1] aralığını tercih etmişlerdir. Zaharie (2002) popülasyon sayısı, ölçeklendirme faktörü ve çaprazlama oranı arasında aşağıdaki empirik ifadeyi önermiştir:

$$2F^2 - \frac{2}{NP} + \frac{CR}{NP} = 0 \quad (3.27)$$

Michalski (2001), başlangıç popülasyonuna 50 birey ile başlayan ve ilk 9 nesil için bu popülasyon sayısı ile devam eden 10. nesilden itibaren popülasyon sayısını yarıya düşürerek 25 birey ile aramaya devam eden bir yaklaşım önermiştir. Literatürde diferansiyel ölçeklendirme faktörü ve çaprazlama oranı için de bazı kurallar geliştirilmiştir (Chang ve Xu 2000, Mayer ve ark. 2005, Chiou ve ark. 2006, Coelho ve Mariani 2006c).

Uyarlamalı yöntemlerde parametre değişimi geri-beslemeli bir sistem ile belirli bir yöne ve büyüklüğe sahip olan değişim parametresinin kullanılmasıyla yapılmaktadır. Burada kontrol parametrelerinin işlem esnasında problemin yapısına göre dinamik ve otomatik olarak belirlenmesi yani kontrol parametrelerinin uyarlamalı olarak problem esnasında belirlenmesi optimizasyon sürecinde hız, verimlilik, güvenilirlik ve etkinlik yönlerinden büyük önem arz etmektedir. Son yıllarda bu yönde çeşitli çalışmalar yapılmaktadır. Feoktistov ve Janaqi (2004), nesil sayısına bağlı olan enerjik bariyer fonksiyonu yardımıyla popülasyon sayısının azaltılması için bir yaklaşım geliştirmişlerdir. Huang ve ark. (2006), dinamik bir popülasyon yapısına sahip yöntem önermişlerdir. Bu yöntemde popülasyonun durumunu gözlemek ve ilerleyişini takip etmek için Varyans takibi yapılmaktadır. Buna göre en kötü uygunluk değerine sahip birey öldürülerek yerine rastgele üretilen yeni birey geçirmektedir. Liu ve Lampinen (2002) diferansiyel ölçeklendirme faktörü ve çaprazlama oranını dinamik olarak değiştirmek için bulanık mantık kontrolü ile çalışan bir bulanık uyarlamalı diferansiyel gelişim algoritması önermişlerdir. Nobakhti ve Wang (2006), ayrıca diferansiyel ölçeklendirme faktörünün uyarlamalı olarak popülasyondaki en iyi uygunluk değerine sahip birey ile en kötü uygunluk değerine sahip birey arasındaki orana göre değiştirilme prensibine göre çalışan bir yaklaşım geliştirmişlerdir. Bunun yanında popülasyon içindeki en iyi uygunluk değerine sahip birey ile popülasyonun ortalama uygunluk değeri arasındaki orana bağlı ayar yapma prensibine göre uyarlamalı olarak diferansiyel ölçeklendirme faktörüne ve çaprazlama oranına değer atayabilen bir yaklaşım Guo ve ark. (2006) tarafından önerilmiştir.

Brest ve ark. (2007), çalışmalarında uyarlamalı diferansiyel gelişim algoritmalarını ve diğer kendi kendine uyarlamalı diferansiyel gelişim algoritmalarını performans yönünden karşılaştırmışlardır. Kontrol parametrelerinin ayarlamalarında uyarlamanın özellikle dışarıdan kullanıcının bir etkileşimi olmadan kendi kendine uyarlama özelliğinin çok büyük oranda olumlu sonuçlar verdiğini gözlemlemişlerdir.

Kendi kendine uyarlama yaklaşımında genellikle uyarlaması yapılacak kontrol parametreleri kromozom içine kodlanmakta ve bu parametrelere genetik operatörler uygulanmaktadır. Bu parametrelerin probleme uygun değerlere sahip olanları hayatlarını sürdürmekte ve bir sonraki nesiller için aile bireylerini teşkil etmektedirler.

Bu şekilde optimizasyon süresince hep en uygun parametre değerleri ile işlemler sürdürülmektedir. Bir başka deyişle kontrol parametreleri sanki optimizasyonda bir tasarım değişkeni gibi ele alınıp bu şekilde eş zamanlı olarak optimizasyon gerçekleştirilmektedir. Bazen de kontrol parametreleri diferansiyel gelişim algoritması içine veya dışına ek bir döngü eklenerek ele alınabilmektedir (Babu ve Jehan. 2003c). Kendi kendine uyarlama özelliği literatürde genellikle iki kontrol parametresi üzerinde durularak ele alınmıştır. Bu parametreler diferansiyel ölçeklendirme parametresi (F) ve çaprazlama oranıdır (CR). Üçüncü parametre olan popülasyon sayısı (NP) üzerinde fazla durulmamıştır. Bunun nedeni diferansiyel gelişim algoritmasının verimliliği ve güvenilirliği üzerine iki parametrenin (F , CR) daha fazla etkisinin olmasıdır. Ancak yine de üçüncü kontrol parametresinin yani popülasyon sayısının (NP) etkisini ihmal etmek mümkün gözükmemektedir.

Brest ve ark. (2007), diferansiyel ölçeklendirme faktörü (F) ve çaprazlama oranı (CR) değerlerini değiştirerek arama stratejisini ayarlayabilen kendi kendine uyarlamalı diferansiyel gelişim algoritmasını geliştirmişler ve bu algoritmayı, diferansiyel ölçeklendirme faktörü (F) ve çaprazlama oranının (CR) yanında ayrıca popülasyon sayısını (NP) da hesaba katan kendi kendine uyarlamalı diferansiyel gelişim algoritması ile (Teo ve ark. 2006) ve yine aynı iki kontrol parametresini bulanık mantık kontrolüyle ayarlayan kendi kendine uyarlamalı diferansiyel gelişim algoritması ile (Liu ve Lampinen 2005) karşılaştırmışlardır.

Hu ve Yan (2009), diferansiyel gelişim algoritmasının performansını geliştirmek için bağışıklık kavramlarını yukarıda bahsedilen üç parametrenin (NP , F , CR) uygun parametre ayarlamalarını yapmak için kullanan bir yöntem geliştirmişlerdir. Bu yöntemde optimizasyon probleminin yapısına göre algoritma değiştirilebilmekte ve bu değişim belirli kontrol parametrelerine göre gerçekleştirilmektedir. Hu ve Yan (2009), geliştirdikleri yöntemin performansını dokuz test problemini çözerek değerlendirmişler ve kendi kendine uyarlamalı diğer diferansiyel gelişim algoritmaları ile karşılaştırmışlardır.

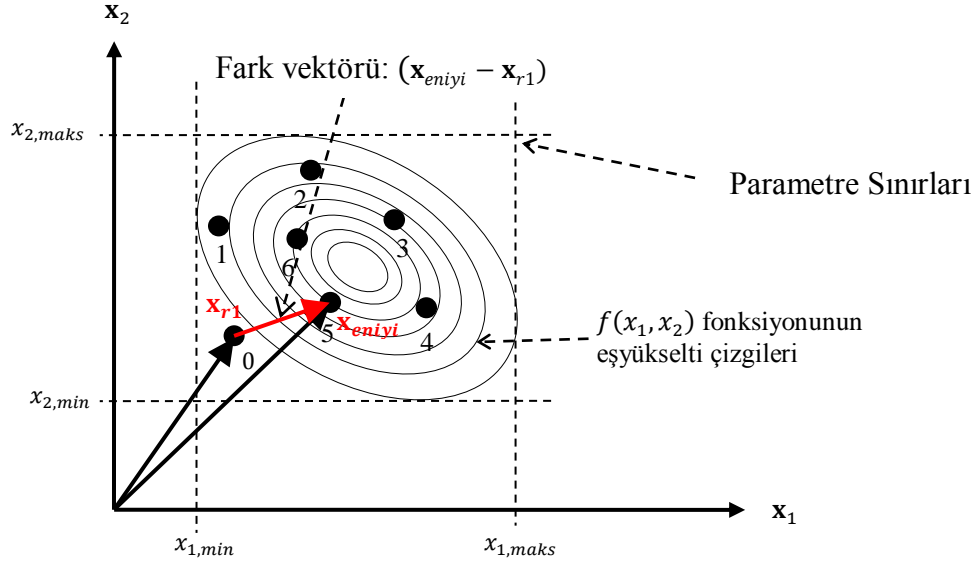
3.9. Geliştirilen Optimizasyon Algoritması (DEBVs)

Diferansiyel gelişim algoritmasının gürültülü problemlerde çok iyi sonuçlar vermediği bilinmektedir. Bu problem başlangıç popülasyonu artırılarak basit bir şekilde çözümlenebilmektedir. Böylece arama uzayı daha iyi taranabilmekte ve global optimum noktanın yakalanma şansı artırılabilir. Ayrıca çeşitli hibrit yöntemler de bu tür problemleri çözmek amacıyla geliştirilmiştir. Diğer yandan bu yöntemlerin hemen hemen hepsi mevcut probleme çözüm olmakla birlikte *hesaplama sayısını artırarak* yeni bir probleme neden olmaktadır. Bunun nedeni ise geliştirilen yeni yöntemlerin sadece global optimum noktaya odaklanmalarından kaynaklanmaktadır.

Etkin bir yöntem hem global optimum noktayı yakalamalı hem de hesaplama sayısını arttırmamalıdır; hatta mümkün ise hesaplama sayısını düşürmelidir. Bunun için bu çalışmada özellikle mutasyon stratejisi üzerinde durularak yeni bir algoritma geliştirilmiştir. Geliştirilen algoritma ile hem global optimum noktanın daha düşük hesaplama sayılarında elde edilmesi hedeflenmiş hem de aynı hesaplama sayılarında daha iyi sonuçlara ulaşılması amaçlanmıştır. Geliştirilen algoritma ile DE'nin güvenilir, çok yönlü ve kullanımı kolay özellikleri iyileştirilerek devam ettirilmiştir. Algoritma tıpkı DE'de olduğu gibi aritmetik operatörler yerine gerçek kodlama kullanmaktadır. Ancak burada kilit rol oynayan ve en iyi bireyler ile bunların farklarını kullanarak oluşturulan en iyi vektörler stratejisini kullanan yeni bir *diferansiyel mutasyon operatörü* geliştirilmiştir (DEBVs). Burada, popülasyon tabanlı optimizasyon özellikleri devam etmekte olup, sırasıyla başlangıç popülasyonunun oluşturulması, mutasyon, çaprazlama ve üreme işlemleri gerçekleştirilmektedir. Tıpkı DE'de olduğu gibi başlangıçta parametrelerin önceden belirlenmiş sınırları dahilinde ortaya çıkan arama uzayı içerisinde N_p adet popülasyon vektörünü simgeleyen rastgele başlangıç noktaları oluşturulmakta ve bu noktalar başlangıç bireyleri olarak adlandırılmaktadırlar.

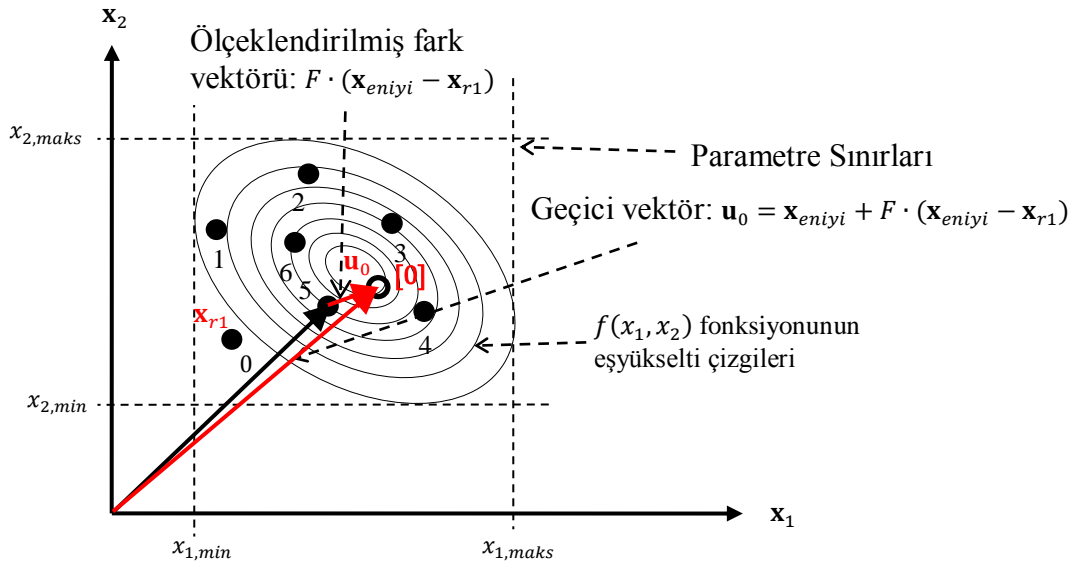
Geliştirilen algoritmada fark vektörü popülasyondaki en iyi birey ile rastgele seçilen birey arasında fark alınarak oluşturulmaktadır. Ancak her nesilde en iyi vektör ile rastgele seçilen vektör yer değiştirmektedir. Örneğin bir nesilde en iyi bireyden olan fark alınırken bir sonraki nesilde rastgele seçilen bireyden olan fark alınmaktadır. Böylece global noktanın bir tarafında olabilecek yığılmanın önüne geçilerek homojen bir dağılım elde edilmektedir. Örnek olarak Şekil 3.37'de verilen arama uzayında

başlangıç popülasyonu rastgele 7 nokta ile oluşturulmuştur. Fark vektörü en iyi birey ile rastgele seçilmiş bir bireyin ifade ettiği vektörden (\mathbf{x}_{eniye} ve \mathbf{x}_{r1}) oluşturulmuştur.



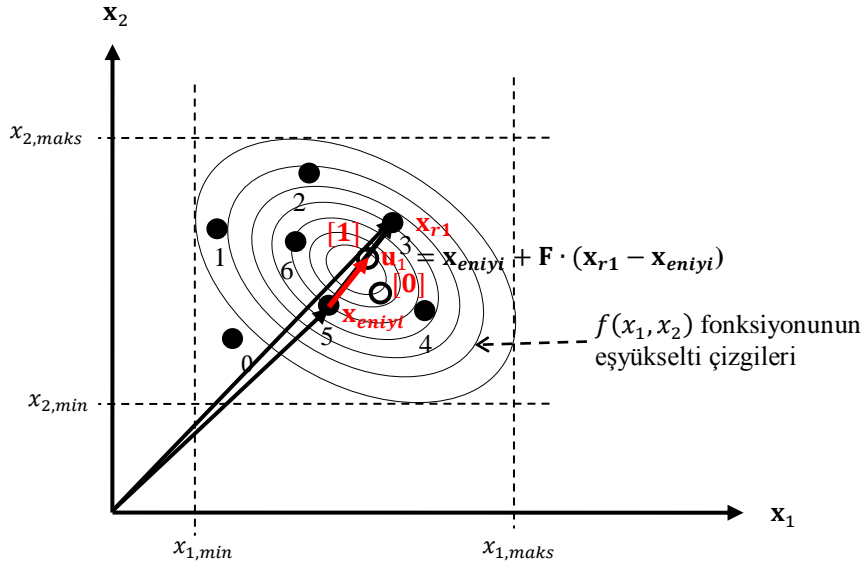
Şekil 3.37 Geliştirilen algoritmada (DEBVs) fark vektörünün oluşturulması

Oluşturulan fark vektörü belirli bir katsayı ile çarpılarak (F) ölçeklendirilmekte ve ölçeklendirilmiş vektör en iyi vektöre (5 no`lu nokta) eklenerek \mathbf{u}_0 çocuk bireyi oluşturulmaktadır (Şekil 3.38).



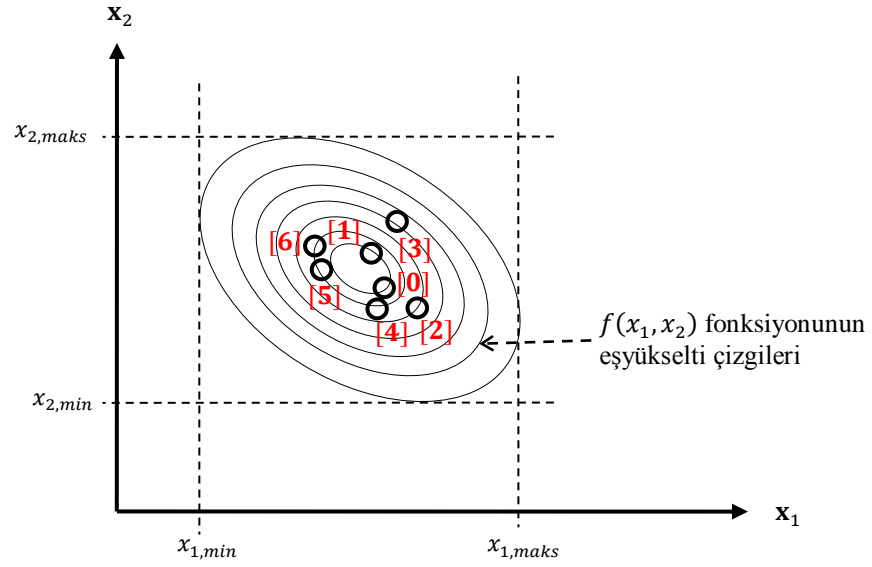
Şekil 3.38 DEBVs`de mutasyon işlemi

Seçim aşamasında, oluşturulan geçici vektör ile popülasyonda mevcut olan aynı indekse sahip vektör aralarında yarıştırlırlar ve daha uygun amaç fonksiyonu değerine sahip olan vektör yarışmayı kazanarak bir sonraki nesilde yerini alır. İkinci geçici vektör oluşturulurken ters olarak rastgele seçilen bir nokta ile en iyi birey arasında fark vektörü oluşturulur ve en iyi bireye eklenerek yeni birey olan \mathbf{u}_1 elde edilmiş olur (Şekil 3.39).



Şekil 3.39 DEBVs`de ikinci geçici vektörün oluşturulması

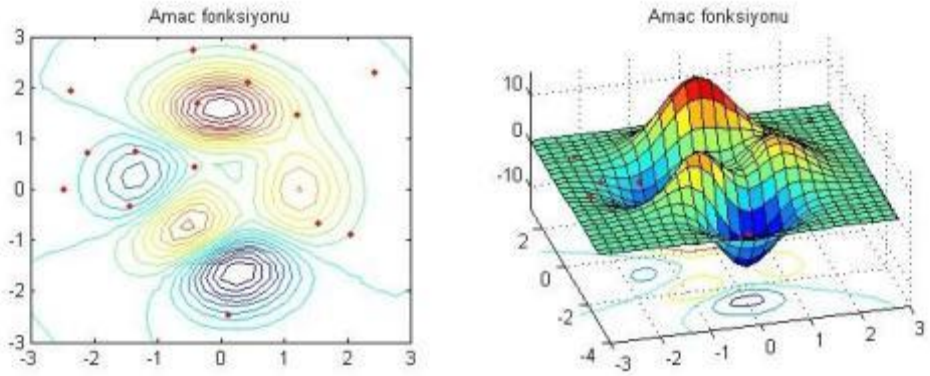
İşleme popülasyon sayısı $N_p = 7$ 'ye ulaşana kadar devam edilir ve DE`de olduğu gibi yeni geçici vektörler ile indeks numaraları aynı olan eski vektörler karşılaştırılarak amaç fonksiyon değerleri daha iyi olan vektörler yeni nesilde yerini alırlar. Böylece elde edilen yeni vektörler bir sonraki nesil için aile bireylerini oluşturmuş olurlar (Şekil 3.40).



Şekil 3.40 İlk nesil sonunda oluşan yeni bireyler

Geliştirilen algoritmanın daha ilk nesilde bile global optimum noktaya yakın bölgelerde yeni popülasyonu oluşturduğu dikkat çekmektedir. Bu nokta performans açısından önem arz etmektedir.

DEBVs'nin hızlı yakınsama özelliğinin daha iyi anlaşılabilmesi açısından arama uzayında hem lokal hem de global optimum noktaya sahip örnek bir optimizasyon problemi (bkz. Denklem 3.19 ve Şekil 3.8) üzerinde hem DE'nin hem de DEBVs'nin ilerleyişi izlenebilir. Ele alınan örnek probleme benzer iki adet minimum noktaya ve üç adet maksimum noktaya sahip bir problem Şekil 3.41'deki gibidir.

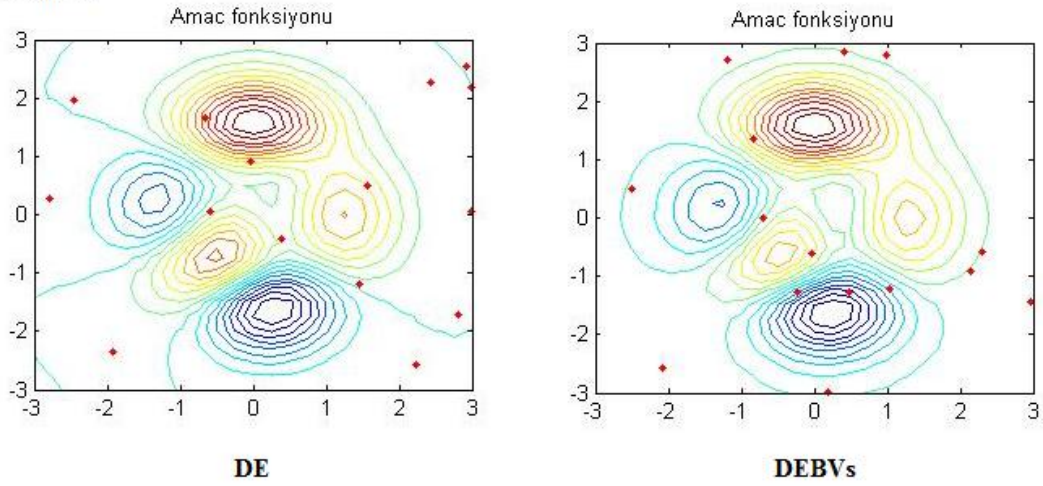


Şekil 3.41 Örnek problemin eşyükselti çizgileri ve 3-boyutlu gösterimi

Lokal ve global optimum noktalara sahip bu problemde nesil sayısı ilerledikçe hem DE'nin arama uzayında ilerleyişi hem de geliştirilen DEBVs algoritmasının ilerleyişi ve

popülasyon dağılımlarının optimizasyon süresince nasıl değişim gösterdiği Şekil 3.41-3.47'den açıkça izlenebilir. Her iki algoritmada popülasyon 15 adet birey içermekte ve her bir birey arama uzayında küçük kırmızı bir nokta ile temsil edilmektedir. Başlangıç popülasyonu rastgele oluşturulup ilk nesil geçtikten sonra DEBVs algoritmasında 15 adet popülasyon bireyinden 3 tanesi global optimum bölgede oluşurken DE'de 15 adet popülasyon bireyinin hiçbirisi henüz global bölgede oluşturulamamıştır (Şekil 3.42).

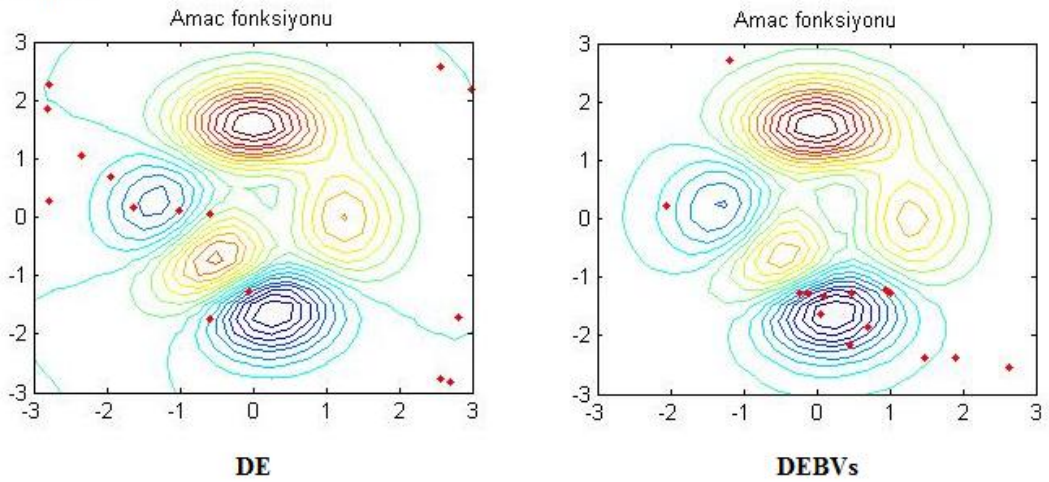
Nesil: 01



Şekil 3.42 DE ve DEBVs'de 1. nesilden sonra popülasyonların arama uzayında dağılımı

DE global optimum bölgede ancak 3. nesil sonrasında 2 birey oluşturabilirken DEBVs aynı bölgede 10 birey oluşturmuştur (Şekil 3.43).

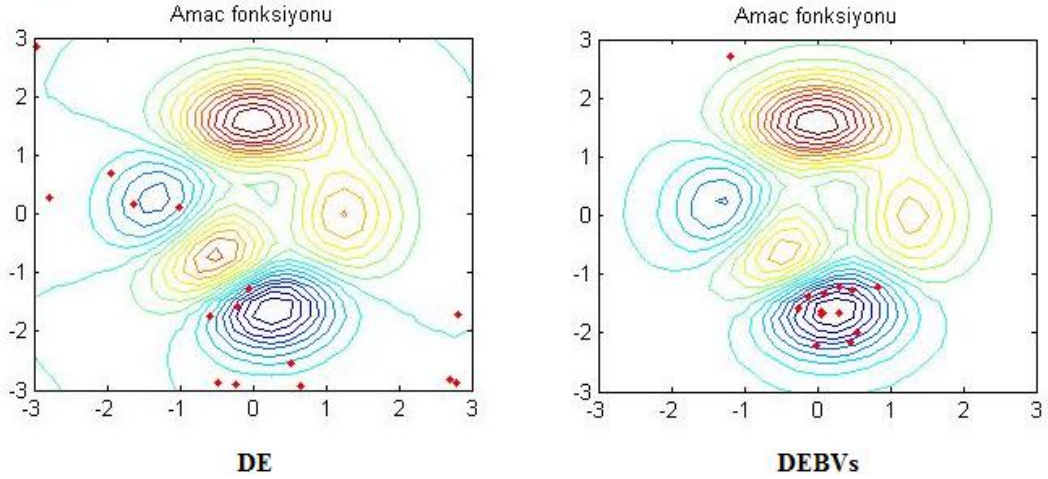
Nesil: 03



Şekil 3.43 DE ve DEBVs'de popülasyonların 3. nesilde arama uzayında dağılımı

Henüz 3. nesil sonrasında bile toplam popülasyonun yarısından fazlasının global optimum nokta etrafında toplanması geliştirilen DEBVs algoritmasının dikkat çekici özelliğini göstermektedir. Ayrıca DEBVs lokal optimum bölgeyi 3. nesilden itibaren terk etmektedir. DE'nin lokal bölgeyi terk etmesi 20. nesilden sonra gerçekleşecektir (Şekil 3.47).

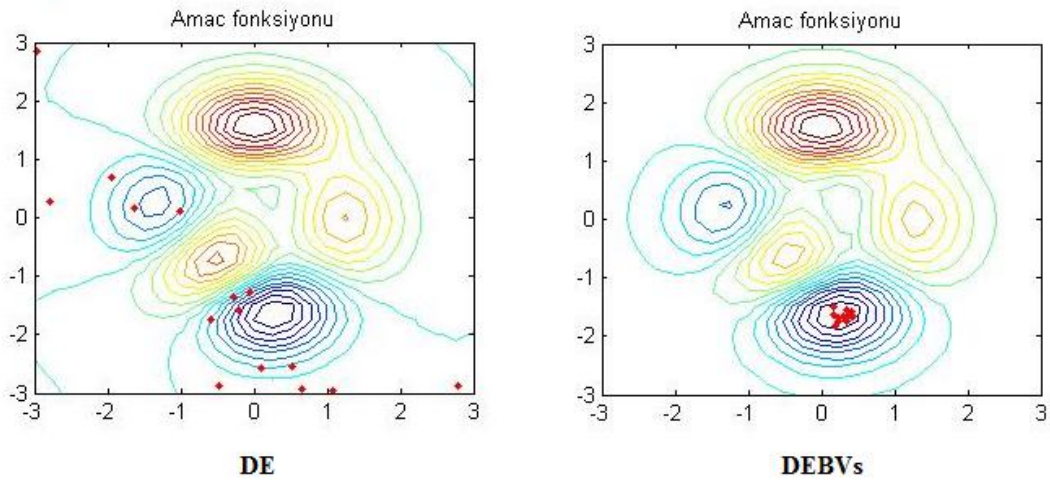
Nesil: 05



Şekil 3.44 DE ve DEBVs`de popülasyonların 5. nesilde arama uzayında dağılımı

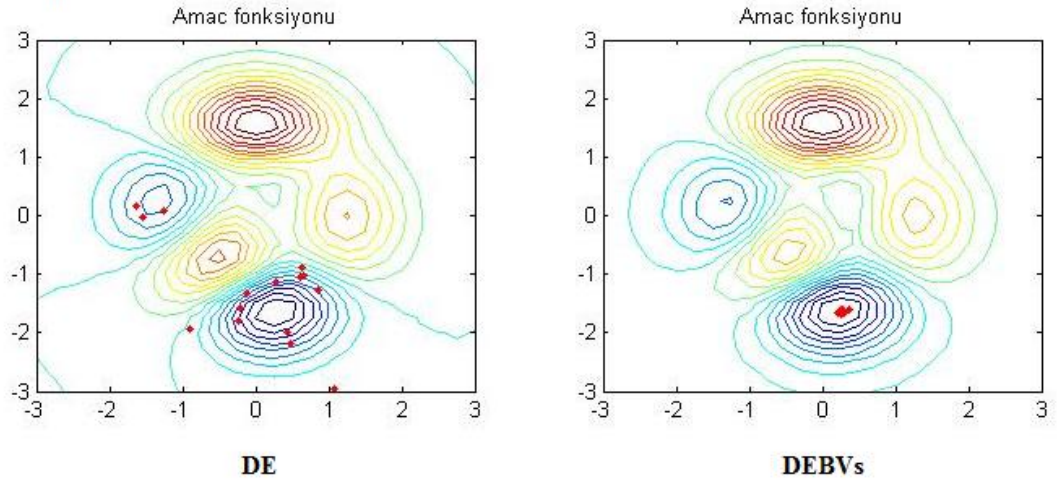
6. nesilden itibaren tüm popülasyon bireylerini global bölgeye aktaran DEBVs artık hızlı bir şekilde global optimum noktaya yoğunlaşarak hedef global minimum noktayı 20 nesilde yakalamaktadır (Şekil 3.44-3.47). DE`de ise aynı düzeyde olmasa da global nokta etrafındaki yoğunlaşma ancak 20. nesilden sonra görülebilecektir, çünkü 20. nesilde bile hala lokal bölgede 2 birey mevcuttur (Şekil 3.47).

Nesil: 06



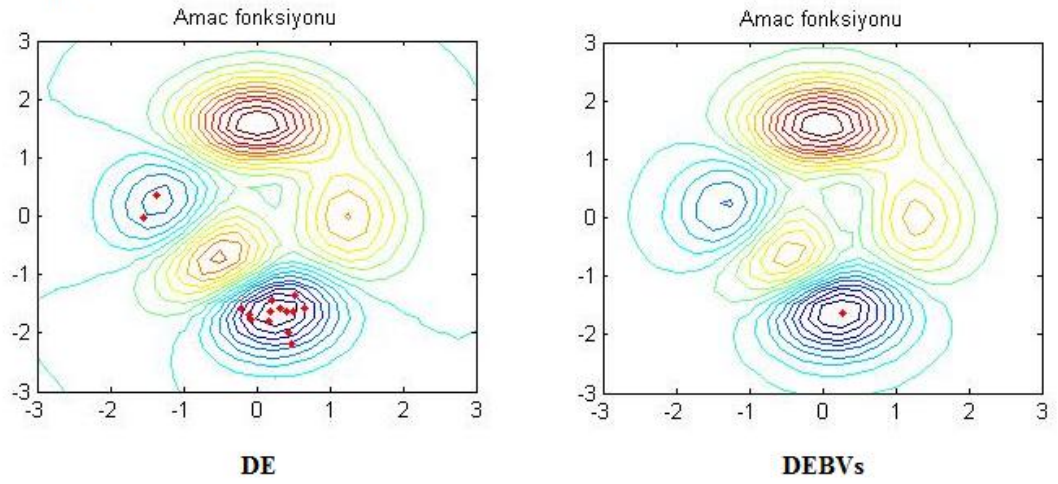
Şekil 3.45 DE ve DEBVs`de popülasyonların 6. nesilde arama uzayında dağılımı

Nesil: 10



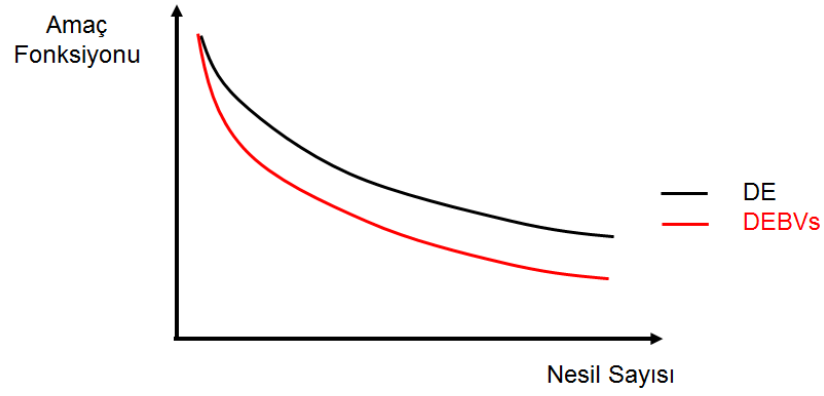
Şekil 3.46 DE ve DEBVs`de popülasyonların 10. nesilde arama uzayında dağılımı

Nesil: 20



Şekil 3.47 DE ve DEBVs`de popülasyonların 20. nesil başlangıcında arama uzayında dağılımı

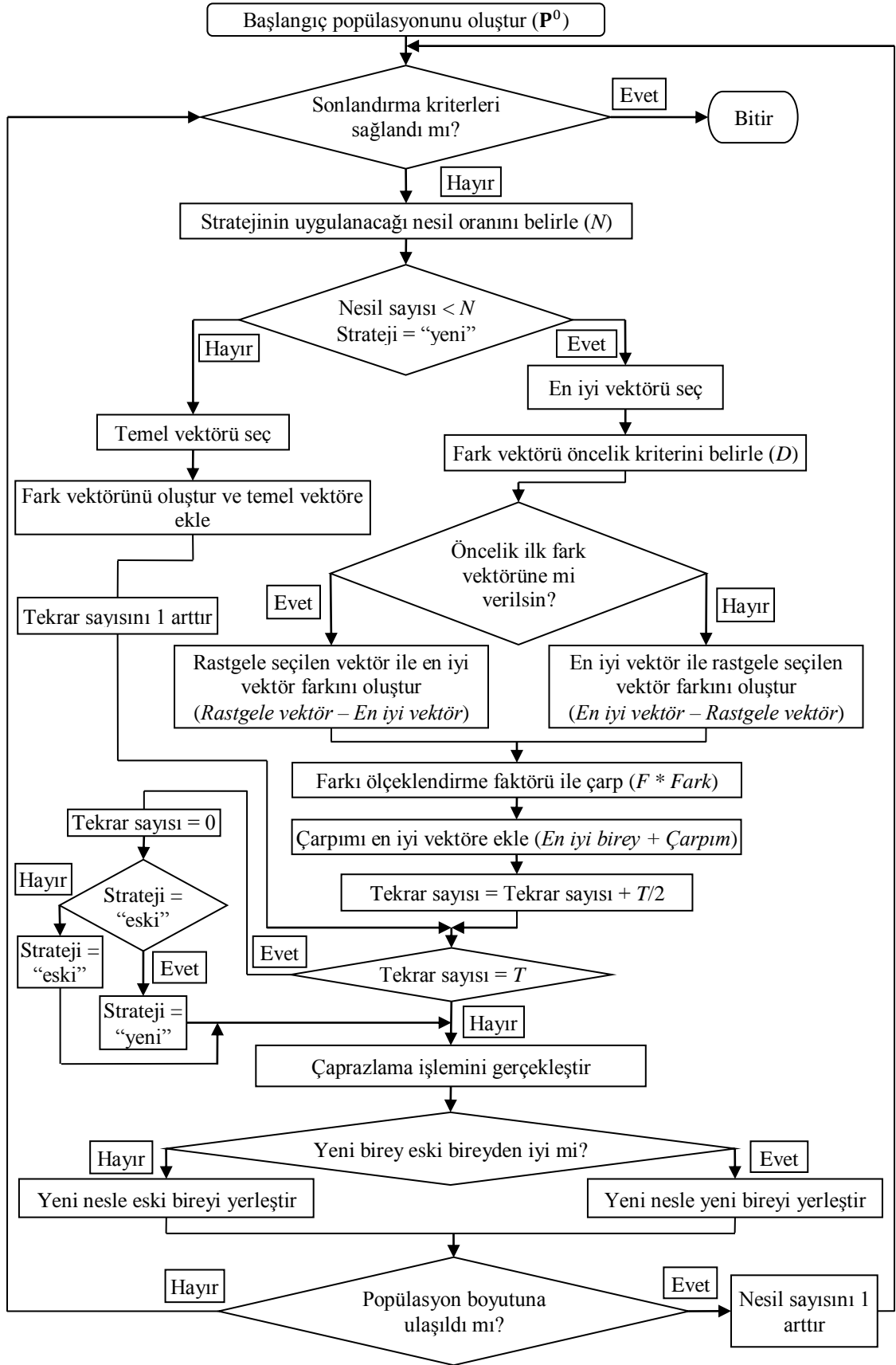
Geliştirilen DEBVs yöntemi ile DE farklı türde çeşitli amaç fonksiyonlarını çözmek için kullanıldığında DEBVs`nin DE`ye göre çok daha hızlı yakınsadığı ve aynı nesil sayılarında çok daha iyi sonuçlar verdiği gözlenmiştir (Şekil 3.48).



Şekil 3.48 DE ve DEBVs ile elde edilen amaç fonksiyon değerlerinin nesil sayısına bağlı değişimi

Yukarıda DEBVs'nin hızlı yakınsama özelliği kısaca açıklanmıştır. Esasında gelişim üç aşamadan oluşmaktadır: başlangıç, gelişim ve hızlı yakınsama. Geliştirilen algoritmanın akış diyagramı Şekil 3.49'da verilmiştir. Başlangıç aşamasında tıpkı DE'de olduğu gibi rastgele bir başlangıç popülasyonu oluşturulmakta (P^0). Daha sonra P^0 popülasyonu gelişerek P^1 popülasyonunu o da gelişerek P^2 popülasyonunu oluşturmaktadır. Bu şekilde tüm bireylerin global optimum bölgede toplanmasına kadar gelişim devam etmektedir. Bu gelişim sağlanırken popülasyon içindeki bireylere üç temel evrimsel operasyon uygulanmaktadır; *diferansiyel mutasyon*, *çaprazlama* ve *seçim* (Şekil 3.49).

Tüm bireyler global optimum bölgede oluştuktan sonra çok hızlı bir şekilde optimum nokta yakalanana kadar yakınsama gerçekleşmektedir. Bu yakınsama sağlanırken popülasyon içindeki bireylere temel üç operasyondan yeni geliştirilen strateji ile çalışan *diferansiyel mutasyonun* büyük etkisi vardır. Bu yeni strateji ile üç yeni katsayı (stratejinin uygulanacağı nesil oranı N , fark vektörü öncelik kriteri D ve tekrar sayısı T) tanımlanmaktadır.



Şekil 3.49 Geliştirilen algoritmanın (DEBVs) akış diyagramı

3.9.1 Stratejinin uygulanacağı nesil oranı (N)

Stratejinin uygulanacağı nesil oranı (N) tüm optimizasyon süresince hangi nesillerde yeni stratejinin uygulanacağını belirlemektedir. Tüm nesillere ($N = 1$) yeni oluşturulan strateji uygulandığında yakınsama çok hızlı gerçekleşmektedir. Ancak bu yakınsama düşük düzeyli optimizasyon problemleri için geçerli olmaktadır. Dolayısıyla düşük zorluk derecesine sahip optimizasyon problemleri için $0,5 < N < 1$ aralığı uygun bir aralık olmaktadır. Yüksek zorluk derecesine sahip (tasarım değişkeni sayısı > 20) optimizasyon problemlerinde en iyi sonuçlar düşük N oranlarında ($0,1 < N < 0,5$) elde edilmiştir.

3.9.2 Fark vektörü öncelik kriteri (D)

Fark vektörü öncelik kriteri olan D , stratejinin homojen olarak her yönde gerçekleştirilmesini sağlamakta en önemli etkidir. Bu katsayı arama ilerledikçe global optimum noktanın bir tarafında olabilecek birikime engel olmaktadır. Test problemlerinde en iyi sonuçlar homojen dağılımın elde edildiği $D = 0,5$ olduğu durumda elde edilmiştir.

3.9.3 Tekrar sayısı (T)

Tekrar sayısı T , yeni stratejinin optimizasyon süresince hangi aralıkla uygulanacağını belirlemektedir. Yeni oluşturulan strateji aralıksız veya sürekli uygulandığında ($T = 1$) düşük karmaşıklık düzeyine sahip optimizasyon problemlerinde çok iyi sonuçlar elde edilmesine karşılık yüksek zorluk derecesine sahip optimizasyon problemlerinde son nesillerde yakınsama hızında yavaşlama eğilimi tespit edilmiştir. Yüksek zorluk derecesine sahip optimizasyon problemlerinde tekrar sayısının $5 < T < 20$ aralığında seçildiği durumlarda ise yakınsama hızında bir yavaşlamanın olmadığı ve en iyi sonuçların bu aralıkta elde edildiği gözlenmiştir.

3.9.4 Başlangıç popülasyonu

Başlangıç aşamasında başlangıç popülasyonu \mathbf{P}^0 , N_p adet bireyden rastgele oluşturulmaktadır (Şekil 3.49). Her bir birey $\mathbf{p}^{0,i}$ ile ifade edilmektedir ($1 \leq i \leq N_p$). Başlangıç popülasyonundaki bireyler şu bağıntı yardımıyla oluşturulmaktadır:

$$x_j^{0,i} = b_j^L + \alpha_j^i (b_j^U - b_j^L), \quad 1 \leq j \leq N \quad (3.28)$$

Burada; $x_j^{0,i}$ ifadesinde 0 başlangıç popülasyonunu, i popülasyonun kaçınıcı sırada olduğunu, j tasarım değişkeni sayısını başka bir ifade ile popülasyon içindeki birey sayısını, α_j^i ifadesi i . popülasyonda j . tasarım değişkeni için $[0,1]$ aralığında rastgele reel sayı üreticini belirtmektedir. b_j^L , j . tasarım değişkeninin alabileceği en küçük değeri, b_j^U , j . tasarım değişkeninin alabileceği en büyük değeri ifade etmektedir.

3.9.5 Geliştirilen mutasyon stratejisi

Geliştirilen mutasyon operasyonunda bir \mathbf{P}^n popülasyonunda her ebeveyn $\mathbf{p}^{n,i}$ bireyi için bir mutant vektörü ($\mathbf{v}^{n+1,i}$) ve bir fark vektörü ($\mathbf{x}^{n+1,v,i}$) oluşturulur. Burada kalın yazılan \mathbf{x} 'in popülasyon içindeki tüm bireyleri temsil eden bir vektör olduğu unutulmamalıdır ($\mathbf{x} = x_1, x_2, \dots, x_N$). Fark vektörü $\mathbf{x}^{n+1,v,i}$ şu şekilde oluşturulmaktadır:

$$\begin{aligned} \mathbf{x}^{n+1,v,i} &= \mathbf{x}^{n,eniyl,i} + \sum_{y \geq 1} \mathbf{F}_y (\mathbf{x}^{n,p_{1y}} - \mathbf{x}^{n,eniyl,i}), \\ \mathbf{x}^{n+1,v,i} &= \mathbf{x}^{n,eniyl,i} + \sum_{y \geq 1} \mathbf{F}_y (\mathbf{x}^{n,eniyl,i} - \mathbf{x}^{n,p_{2y}}), \end{aligned} \quad (3.29)$$

$$1 \leq i \neq p_{1y} \neq p_{2y} \leq N_p$$

Burada; $\mathbf{x}^{n,eniyl,i}$ n 'nci popülasyonda i 'nci bireyin yerine oluşturulacak birey için seçilen ($eniyl,i$) temel vektörü, $\mathbf{x}^{n,p_{1y}}$ n 'nci popülasyonda $[1, N_p]$ arasındaki tamsayılardan rastgele seçilerek belirlenen p_{1y} 'nci bireyi, yine $\mathbf{x}^{n,p_{2y}}$ n 'nci popülasyonda $[1, N_p]$ arasındaki tamsayılardan rastgele seçilerek belirlenen p_{2y} 'nci bireyi, $\mathbf{x}^{n,eniyl,i}$ n 'nci popülasyonda mevcut en iyi bireyi, \mathbf{F}_y $[0,1]$ aralığında olacak şekilde y 'nci vektör farkı için belirlenen ölçeklendirme faktörünü ifade etmektedir.

Mutasyon işlemi sonucunda elde edilecek yeni bireyin, belirlenen sınırlar ($[b_j^U, b_j^L]$) dışında oluşabilme ihtimali mevcut olduğundan bu birey yerine Şekil 3.50'de sözde kodu verilen rastgele yeniden birey oluşturulmaktadır. Burada oluşan uygun olmayan

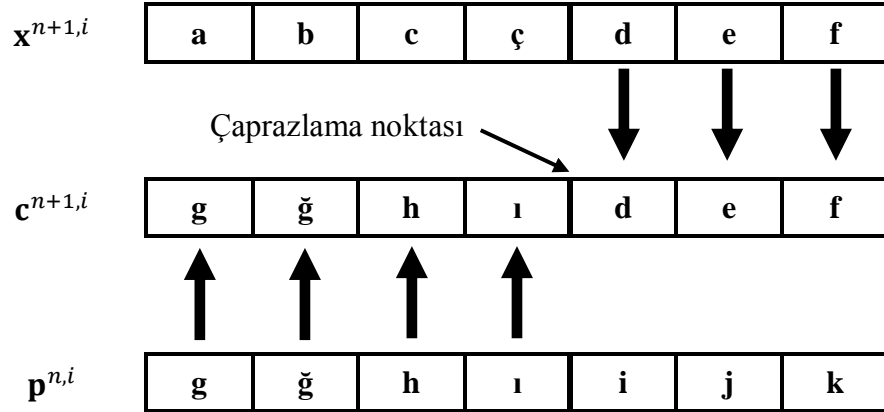
optimizasyon parametreleri ile oluşan bireyler ($x_j^{n+1,v,i} \notin [b_j^U, b_j^L]$) arama uzayında rastgele üretilen uygun bireyler ile değiştirilmektedir.

- İşlemi gerçekleştir: $j = 1, N$
 - o Şartlar sağlanıncaya kadar işlemi gerçekleştir ($x_j^{n+1,v,i} < b_j^L$ veya $x_j^{n+1,v,i} > b_j^U$):
 - $x_j^{n+1,v,i} = b_j^L + \alpha_j^i(b_j^U - b_j^L)$
 - o Dur.
- Dur

Şekil 3.50 Rastgele yeniden birey oluşturma işleminin sözde kodu

3.9.6 Çaprazlama

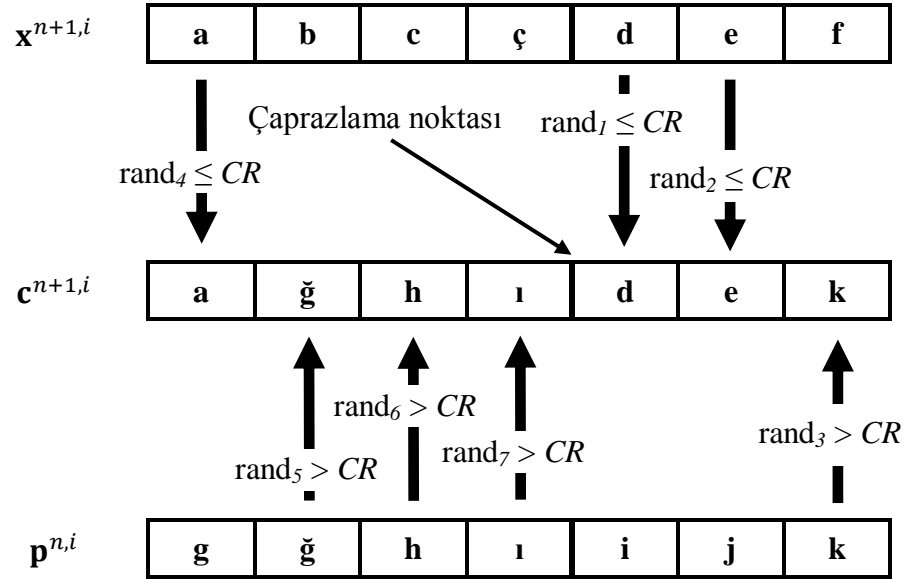
Çaprazlama operasyonunda mutasyon işleminde elde edilen yeni bireyler ($\mathbf{x}^{n+1,i}$) ile mevcut popülasyondaki bireyler ($\mathbf{p}^{n,i}$) rastgele seçilen çaprazlama noktasından kesilerek çocuk bireyler ($\mathbf{c}^{n+1,i}$) oluşturulmaktadır. Burada $\mathbf{p}^{n,i}$ anne bireyler olarak, $\mathbf{x}^{n+1,i}$ baba bireyler olarak adlandırılmaktadır. Her çift sadece bir tane çocuk birey meydana getirebilmektedir (Şekil 3.51).



Şekil 3.51 Çaprazlama işlemi

Şekil 3.51'de şematik gösterimi verilen çaprazlama işlemi tek-noktalı çaprazlama işlemi olarak adlandırılmaktadır. Seçilen çaprazlama noktası sayısına göre iki, üç ve n-noktalı çaprazlama işlemleri tanımlanmıştır. İki-noktalı çaprazlama işleminde bireyler 3 parçaya ayrılırken n-noktalı çaprazlama işleminde bireyler n+1 adet parçaya ayrılmaktadır.

Geliştirilen algoritmada üniform çaprazlama işlemi kullanılmıştır. Burada rastgele seçilen çaprazlama noktasından sonraki genler, $\text{rand}_j(0,1) \leq CR$ koşuluna bakılarak teker teker kıyaslanmakta, eğer koşul sağlanırsa yeni gen mutasyon işleminde elde edilen yeni bireyden ($\mathbf{x}^{n+1,i}$) alınır, koşul sağlanmazsa ($\text{rand}_j(0,1) > CR$) yeni gen mevcut popülasyondaki aynı sıradaki bireyden ($\mathbf{p}^{n,i}$) alınmaktadır. İşleme yeni çocuk bireyler ($\mathbf{c}^{n+1,i}$) oluşturuluncaya kadar devam edilmektedir (Şekil 3.52).



Şekil 3.52 Geliştirilen algoritmada kullanılan üniform çaprazlama işlemi

3.9.7 Seçim

Seçim operasyonunda ebeveyn-çocuk bireyleri arasında bir rekabet söz konusudur. Amaç fonksiyon değerlerine göre yarışmayı kazanan bir sonraki nesilde yerini alacaktır. Bu rekabet matematiksel olarak şu şekilde ifade edilmektedir:

$$\mathbf{p}^{n+1,i} = \begin{cases} \mathbf{c}^{n+1,i}, & \text{eğer uygunluk}(\mathbf{c}^{n+1,i} > \mathbf{p}^{n,i}), \\ \mathbf{p}^{n,i}, & \text{aksi halde.} \end{cases} \quad (3.30)$$

Algoritmanın bitişini belirleyen sonlandırma kriteri olarak birçok düzenleme geliştirilmiştir. Bunlar içerisinde geliştirilen algoritmada hem belirli bir amacın karşılanması hem de amaç fonksiyonu hesaplama sayısının sınırlandırılması kriterleri birlikte kullanılmaktadır. Bunun için bilinen amacın karşılanması yani amaç fonksiyonunun belirli bir değerden küçük veya eşit olması şartı ve amaç fonksiyonu

hesaplama sayısının önceden belirlenen (birçok test probleminde 500 000 hesaplama sayısı) limit değere ulaşma şartı aranmaktadır. Bu iki şarttan hangisi sağlanırsa optimizasyon işlemi sonlandırılmaktadır.

3.10. Kısıt İçeren Optimizasyon

Birçok mühendislik tasarım optimizasyon probleminde amaç fonksiyonunu minimize eden tasarım değişkenlerinin bulunması işlemi gerçekleştirilirken aynı zamanda verilen kısıt şartlarının sağlanması gerekmektedir. Kısıt içeren standart bir optimizasyon modeli aşağıdaki gibidir:

Amaç fonksiyonu;

$$f(x) = f(x_1, x_2, \dots, x_n) \quad (3.31)$$

Eşitlik kısıtları;

$$h_j(x) = h_j(x_1, x_2, \dots, x_n) = 0 \quad j = 1, 2, \dots, p \quad (3.32)$$

Eşitsizlik kısıtları;

$$g_i(x) = g_i(x_1, x_2, \dots, x_n) \leq 0 \quad i = 1, 2, \dots, m \quad (3.33)$$

Sınır kısıtları;

$$x_{k,alt} \leq x_k \leq x_{k,üst} \quad k = 1, 2, \dots, D \quad (3.34)$$

Burada; p eşitlik kısıtlarının toplam sayısını, m eşitsizlik kısıtlarının toplam sayısını ve D sınır kısıtlarının toplam sayısını ifade etmektedir. Sınır kısıtları eşitsizlik kısıtlarından olmasına rağmen burada ayrı olarak ele alınmıştır. Çünkü sınır kısıtları hem hemen hemen tüm optimizasyon problemlerinde ortaya çıkmakta hem de eşitsizlik kısıtlarına nazaran daha kolay ele alınıp çözümlenebilmektedirler.

Kısıtlar arama uzayını bölümlere ayırarak DE'nin ve DEBVs'nin global optimum noktaya ilerlemesini zorlaştırmaktadırlar. Özellikle fazla sayıda kısıt içeren problemlerde arama uzayında adacıklar oluşabilmekte, bu durumda eğer global optimum nokta bu ada üzerinde bulunuyor ise problem daha zor bir hale dönüşmektedir.

Bu durumda kısıtların sağlanmadığı bölgelerde de fark vektörlerinin oluşmasına imkan veren yöntemler kullanmak oldukça kullanışlı olmaktadır. Birçok mühendislik alanındaki tasarım optimizasyon problemi, yukarıda verilen standart modele dönüştürülerek çözülmektedir. Kısıt içeren tasarım optimizasyon problemlerinin kısıt içermeyen forma dönüştürülüp çözülmesi çok sık kullanılan bir yöntemdir.

3.10.1 Sınır kısıtları

Sınır kısıtları gerçek optimizasyon uygulamalarının hemen hemen hepsinde ortaya çıkmaktadır. Gerçek uygulamalarda fiziksel bileşenler kullanıldığından ve onların belirli ölçülerde olmaları gerektiğinden doğal olarak tasarım değişkenleri belirli sınırlar dâhilinde tanımlı olmaktadır. Örneğin fiziksel bileşene sahip hiçbir tasarım değişkeni negatif olamaz. Problemin kendisi kısıtsız bir problem olsa bile tasarım değişkenlerinin alt ve üst limitleri olabilmektedir.

Hem DE`de hem de DEBVs`de tüm popülasyon bireyleri belirlenen sınırlar dahilinde oluşturulmaktadır. Belirtilen sınırların aşılması ancak mutasyon operasyonunda olasıdır. DE`de bu olasılık daha yüksektir, çünkü geliştirilen DEBVs algoritmasında yönelim en iyi bireye doğru olduğundan çoğunlukla mutasyon sonucunda arama uzayının sınırlarının aşılması olmamaktadır. Dolayısıyla sadece mutasyon sonucunda bireylerin sınır kısıtlarını aşıp aşmadığı kontrol edilmesiyle işlem zamanından tasarruf sağlanabilmektedir.

Sınır kısıtlarının aşıldığı durumlarda genellikle iki temel yaklaşım ile çözüme gidilmektedir. Birinci yaklaşımda sınır kısıtlarını sağlamayan parametreler cezalandırılarak sınır kısıtlarını sağlayacak bölgelere doğru yönlendirilmesi sağlanır ve tüm sınır kısıtları sağlanıncaya kadar cezalandırma işlemine devam edilir. İkinci yaklaşımda ise sınır kısıtlarını sağlamayan parametreler sınır kısıtlarını sağlayacak şekilde tekrar üretilir. (Price ve ark. 2005).

Tuğla duvar ceza yönteminde sınır kısıtlarını sağlamayan bireyin amaç fonksiyonuna amaç fonksiyon değeri çok uzakta olan bir değer eklenerek bir sonraki nesilde seçilmesi zorlaştırılmaktadır. Bu yöntemde global optimum noktanın sınırlara yakın bölgelerde olduğu durumlarda arama yavaşlamaktadır; çünkü sınır kısıtlarının sağlanmadığı

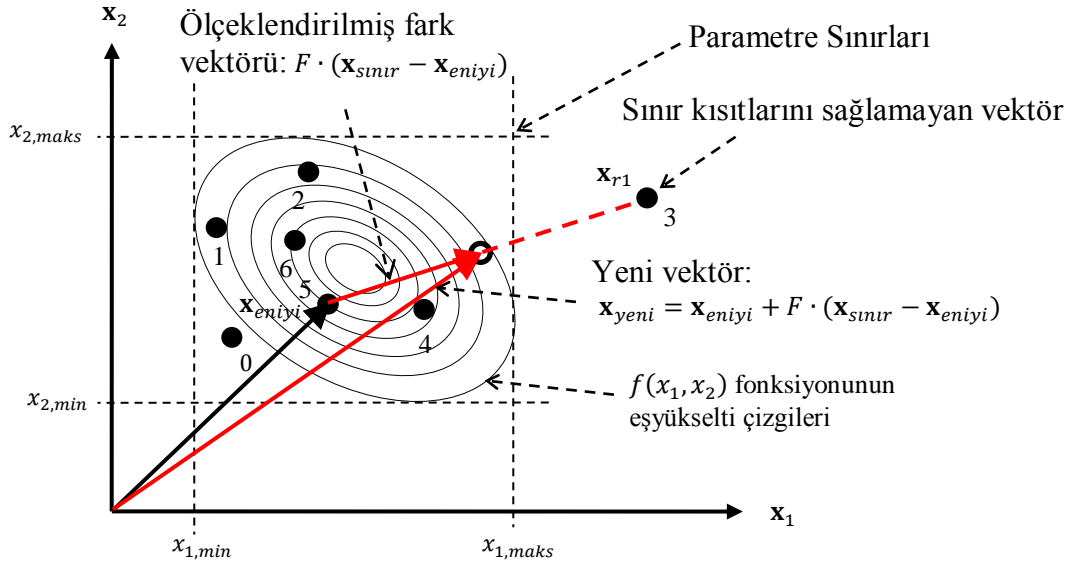
bölgelere yakın bireylerin hayatlarını sürdürmeleri bu ceza yöntemi kullanıldığında zorlaşmaktadır.

Uyarlamalı ceza yönteminde sınır kısıtlarının sağlanmadığı parametre sayısı oranınca bireyler cezalandırılmaktadır. Böylece tuğla duvar ceza yönteminin aksine sınıra yakın bölgelerdeki aramalar kolaylaşmaktadır.

Ceza yöntemlerinde sınır kısıtlarını sağlamayan parametreler yeniden oluşturulmak yerine sonraki nesillerde seçilemeyecek düzeyde kötüleştirilmektedirler. Bunun sonucunda arama yavaşlamakta ve istenilen sonuca ulaşılması için gereken nesil sayısı artmaktadır. Bunu engellemek için sınır kısıtlarını sağlamayan parametreler sınır kısıtlarını sağlayacak şekilde tekrar üretilmektedir.

Rastgele parametre üretme yönteminde adından anlaşılacağı üzere sınır kısıtlarını sağlamayan birey ile sınır kısıtlarını sağlayacak şekilde rastgele üretilmiş yeni birey yer değiştirilir (Lampinen ve Zelinka 1999). Bu yöntemde yine sınırlara yakın bölgelerde mevcut bulunan global optimum noktanın yakalanmasının zorluğu sınır kısıtlarını sağlamayan bireylerin tamamen değiştirilmesi nedeniyle devam etmektedir.

Geliştirilen algortmada da kullanılan ve yukarıda bahsedilen tüm olumsuzlukları içermeyen “kendini toparlama” yönteminde rastgele parametre üretme yöntemine benzer olarak sınır kısıtlarının birini veya tümünü sağlamayan bireyler sınır kısıtlarını sağlayan bireyler ile güncelleştirilmektedir. Ancak burada güncelleştirme veya değiştirme işleminde yeni birey rastgele üretilmeyip sınır kısıtlarını sağlamayan bireyin mutasyon işlemindeki temel vektörüne sınır kısıtını sağlamadığı parametrenin sınır değerden olan farkı eklenerek sınır bölgelere yakın ve sınır kısıtlarını sağlayan yeni bireyler oluşturulur (Şekil 3.53). Kendini toparlama yöntemi kullanıldığında sınıra yakın bölgelerde bireyler rahatlıkla oluşturulabildiğinden bu bölgelerde mevcut bulunan global optimum nokta rahatlıkla yakalanabilmekte ve arama süresince herhangi bir yavaşlama söz konusu olmamaktadır (Price ve ark. 2005).



Şekil 3.53 DEBVs`de sınır kısıtlarını sağlamayan birey için kullanılan kendini toplama yönteminde yeni bireyin oluşturulması

3.10.2 Eşitlik kısıtları

Eşitlik kısıtları aşağıdaki formda yazılabilmektedir:

$$h_j(x) = h_j(x_1, x_2, \dots, x_n) = 0 \quad j = 1, 2, \dots, p \quad (3.35)$$

Burada; h eşitlik kısıt fonksiyonunu, x tasarım değişkenlerini ve p eşitlik kısıtlarının toplam sayısını belirtmektedir. Eğer koşullar izin veriyorsa eşitlik kısıtları amaç fonksiyonunda ele alınabilmektedir. Bu şekilde değişkenlerden bir veya birkaçı elimine edilerek kısıt sayısı azaltılır veya problem tamamen kısıtsız forma dönüştürülebilir. Bunun için eşitlik kısıtındaki bir değişken diğer değişkenler cinsinden ifade edilecek şekilde bir bağıntıya dönüştürülmekte ve amaç fonksiyonunda o değişken yerine elde edilen bağıntı yazılmaktadır.

Ancak bir değişkenin çekilerek o eşitlik kısıtından uygun bir bağıntı elde etmek her zaman mümkün olmamaktadır. Bu nedenle eşitlik kısıtlarını eşitsizlik kısıtlarına dönüştüren yöntemler geliştirilmiştir. Bunun için önceden belirlenen bir hata değeri (ϵ) mertebesinde eşitlik kısıtını ihmal etmeye izin verilir:

$$|h_j(x)| < \varepsilon, \quad j = 1, 2, \dots, p \quad (3.36)$$

Bu şekilde eşitlik kısıtı $h_j(x) < \varepsilon$ ve $h_j(x) > -\varepsilon$ olacak şekilde iki adet eşitsizlik kısıtına dönüştürülmüş olmaktadır. Burada hata değeri ε , ne kadar küçük seçilirse eşitlik kısıtı o derece sağlanmış olmaktadır, ancak bu durumda problemin zorluk derecesi katlanarak artmaktadır.

3.10.3 Eşitsizlik kısıtları

Eşitsizlik kısıtları aşağıdaki form kullanılarak ele alınmaktadır:

$$g_i(x) = g_i(x_1, x_2, \dots, x_n) \leq 0 \quad i = 1, 2, \dots, m \quad (3.37)$$

Burada; g eşitsizlik kısıt fonksiyonunu, x tasarım değişkenlerini ve m eşitsizlik kısıtlarının toplam sayısını belirtmektedir. Eşitsizlik kısıtları çoğunlukla kısıtları sağlamayan bireylerin amaç fonksiyon değerlerine kötü değerler eklenmesi şeklinde uygulanan cezalandırma işlemi ile ele alınmaktadır.

Ceza fonksiyonu yöntemleri, temel optimizasyon problemini alternatif formülasyonlara dönüştürürler öyle ki sayısal çözümlere bir dizi kısıtlanmamış minimizasyon probleminin çözümüyle elde edilen sonuçlar kullanılarak ulaşılır. Kısıt içeren bir problemde kısıtların her terimi ağırlıklandırılmış bir ceza katsayısı (w) ile çarpılır ve elde edilen sonuç amaç fonksiyonuna eklenir. Böylece kısıt içeren fonksiyon kısıt içermeyen bir fonksiyona dönüştürülür:

$$f_{yeni}(x) = f(x) + \sum_{i=1}^m w_i \cdot G_i(x) \quad (3.38)$$

Cezalandırma işleminde kullanılan ağırlık katsayıları (w) tüm ceza terimlerini normalize ederek aynı düzeyde amaç fonksiyonuna etki etmelerine yardımcı olmaktadır.

Yukarıda genel formu verilen ceza fonksiyonu formülasyonları, iç ve dış yöntemler olarak iki kategoriye ayrılmaktadır. İç formülasyonlarda optimum noktaya kısıtların sağlandığı uygun bölgelerden yaklaşılmaya çalışılmakta, dış formülasyonlarda ise optimum noktaya kısıtların sağlanmadığı uygun olmayan yasak bölgelerden

yaklaşılmaya çalışılmaktadır. İç ceza fonksiyonlarında kısıt fonksiyonları bariyer veya logaritmik bariyer fonksiyonları şeklinde tanımlanır (Carrol 1961):

$$G_i(x) = -\frac{1}{g_i(x)} \quad i = 1, 2, \dots, m \quad (3.39)$$

$$G_i(x) = \log(-g_i(x)) \quad i = 1, 2, \dots, m$$

Dış ceza fonksiyonu formülasyonları için kullanılan $G_i(x)$ fonksiyonlarından bazıları aşağıdaki gibidir:

$$G_i(x) = \max[0, g_i(x)] \quad i = 1, 2, \dots, m \quad (3.40)$$

$$G_i(x) = \{\max[0, g_i(x)]\}^2 \quad i = 1, 2, \dots, m$$

İç yöntemlerde, oluşturulan yeni fonksiyonun kısıtlanmamış minimum değeri uygun bölgede yer alırken dış yöntemlerde, kısıtlanmamış minimum değer uygun olmayan bölgede yer almaktadır.

Cezalandırma yöntemleri fazla sayıda kısıt içeren problemlerde iyi sonuç vermeyebilmektedirler. Çünkü ağırlık katsayılarının arama başlamadan önce probleme çok uyumlu bir şekilde seçilmesi gerekmektedir. Özellikle evrimsel algoritmalarda cezalandırma işlemi yakınsamayı yavaşlatmakta veya uygun olmayan çözümlere erken yakınsamaya neden olabilmektedir. Çünkü kısıtlar arama uzayında birbirinden bağımsız adacıklar şeklinde optimum bölgeler oluşturabilmekte ve cezalandırma ile bu bölgelere erişim zorlaşmaktadır. Ayrıca ağırlık katsayılarının uygun seçilemediği durumlarda bir ceza terimi baskın olmakta ve diğer ceza terimlerinin amaç fonksiyonunda öneminin azalmasına neden olunabilmektedir. Ağırlık katsayılarının yeterince büyük seçilemediği durumlarda uygun olan bölgelerdeki amaç fonksiyon değerleri uygun olmayan bölgelerdeki amaç fonksiyon değerlerinden yeterince küçük olamamakta hatta büyük olabilmektedir. Bunun sonucunda popülasyon uygun olmayan bölgelerde yakınsamaya başlamaktadır.

Direkt kısıtları ele alan yöntemlerde, ceza fonksiyonu uygulayan yöntemlerdeki ağırlık katsayıları bulunmadığından ağırlık katsayılarının uygun değerlerini bulmak için yapılan ek hesaplamalara gerek kalmamaktadır. Kısıtları sağlamayan bireyleri

cezalandırma işleminin tersine kısıtları direkt ele alan yöntemlerde ilk nesillerde kısıtlar biraz gevşetilerek ileriki nesillerde kısıtları sağlayan bireylerin oluşmasına imkân verilmektedir. Nesil ilerledikçe kısıtlar sıkılaştırılmakta ve belli nesilden sonra tüm kısıtların sağlanması beklenmektedir.

Lampinen (2002) hızlı yakınsama sağlayan yeni bir direkt kısıtları ele alan yöntem geliştirmiştir. Bu yöntemde sadece amaç fonksiyon değerleri değerlendirmeye dâhil edilmeyip kısıt fonksiyon değerleri de değerlendirmede dikkate alınmaktadır. Yeni nesle aktarılacak birey seçilirken aşağıdaki ölçütler dikkate alınmaktadır:

- Seçilecek ve hedef bireyler tüm kısıtları sağlamaktadırlar ancak seçilecek bireyin amaç fonksiyon değeri hedef bireyin amaç fonksiyon değerinden daha düşüktür veya ona eşittir.
- Seçilecek birey tüm kısıtları sağlamaktadır ancak hedef birey en az bir kısıtı sağlamamaktadır.
- Hem seçilecek birey hem de hedef birey en az bir kısıtı sağlamamaktadır ancak seçilecek birey hedef bireyden daha az sayıda kısıtı sağlamamaktadır.

Geliştirilen DEBVs algoritmasında yukarıda çalışma prensibi açıklanan direkt kısıtları ele alan yöntem kullanılmıştır. Böylece kısıtları sağlamayan bölgelerden kısıtları sağlayan global optimum bölgeye ulaşma imkânı bulunmaktadır. Ayrıca kısıtları sağlama derecelerine göre bireyler sınıflandırılarak global optimum noktaya hızlı bir şekilde yakınsama gerçekleştirilmektedir. Ayrıca ceza fonksiyonlarındaki ağırlık katsayısı gibi herhangi bir parametrenin en uygun değerinin bulunmasına gerek duyulmaması hesaplama sayısını düşürmektedir.

3.11. Çok Amaçlı Optimizasyon

Genel anlamda çok amaçlı problemleri diferansiyel gelişim ile çözmeye yönelik toplam yöntemi, genelleştirilmiş diferansiyel gelişim ve Pareto diferansiyel gelişim olmak üzere üç yaklaşım kullanılmaktadır (Qing 2009).

3.11.1. Ağırlıklı toplam yöntemi

Ağırlıklı toplam yönteminde çok amaçlı optimizasyon problemi matematiksel olarak tek amaçlı optimizasyon problemine tüm amaç ve kısıt fonksiyonları tek bir amaç fonksiyonu içinde tanımlanarak dönüştürülür ve bu optimizasyon problemi tek amaçlı optimizasyon gibi çözülür. Bu dönüşüm aşağıdaki form kullanılarak gerçekleştirilir;

$$f(\mathbf{x}) = \sum_{i=1}^{N_{o\min}} w_i^{\min} f_i^{\min}(\mathbf{x}) - \sum_{i=1}^{N_{o\max}} w_i^{\max} f_i^{\max}(\mathbf{x}) - \sum_{i=1}^{N_{c+}} w_i^+ C_i^+(\mathbf{x}) + \sum_{i=1}^{N_{c-}} w_i^- C_i^-(\mathbf{x}) + \sum_{i=1}^{N_{c=}} w_i^= |C_i^=(\mathbf{x})| \quad (3.41)$$

Burada; $f(\mathbf{x})$ dönüştürülmüş amaç fonksiyonunu, $N_{o\min}$ optimizasyon problemi içerisindeki minimumu aranan amaç fonksiyonu sayısını, w_i^{\min} minimumu aranan amaç fonksiyonunun ağırlığını, $f_i^{\min}(\mathbf{x})$ minimumu aranan amaç fonksiyonunu, $N_{o\max}$ optimizasyon problemi içerisindeki maksimumu aranan amaç fonksiyonu sayısını, w_i^{\max} maksimumu aranan amaç fonksiyonunun ağırlığını, $f_i^{\max}(\mathbf{x})$ maksimumu aranan amaç fonksiyonunu, N_{c+} büyük olma şartını içeren eşitsizlik kısıtının sayısını, w_i^+ büyük olma şartını içeren eşitsizlik kısıtının ağırlığını, $C_i^+(\mathbf{x})$ büyük olma şartını içeren eşitsizlik kısıtını, N_{c-} küçük olma şartını içeren eşitsizlik kısıtının sayısını, w_i^- küçük olma şartını içeren eşitsizlik kısıtının ağırlığını, $C_i^-(\mathbf{x})$ küçük olma şartını içeren eşitsizlik kısıtını, $N_{c=}$ eşitlik kısıtının sayısını, $w_i^=$ eşitlik kısıtının ağırlığını, $C_i^=(\mathbf{x})$ eşitlik kısıtını ifade etmektedir.

Ağırlıklı toplam yöntemi basitliğinden dolayı çok-amaçlı optimizasyon problemlerinde en sık kullanılan yöntem olmakla birlikte bu yöntem ile lokal optimum noktalara takılabilme olasılığının artması ve ağırlık faktörlerinin en uygun değerlerinin en iyi sonucu alabilmek için ne olabileceği kullanıcıların tecrübelerine bağlı olduğu için diferansiyel gelişimin yakınsama özelliğinden tam olarak yararlanılamayabilme olasılığı yüksektir. Ağırlıklı toplam yönteminin performansını arttırmak veya bazı özel problemleri çözebilmek için geliştirilen ve ağırlıklı toplam yönteminin özel kullanımı olan ağırlıklı Chebysheff ve bulanık mantık tabanlı ağırlıklı toplam yöntemleri de mevcuttur (Qing 2009).

3.11.2. Genelleştirilmiş diferansiyel gelişim

Ağırlıklı toplam yönteminin amaç ve kısıt fonksiyonlarının ağırlık faktörlerinin en uygun değerlerini bulmayı kolaylaştırmak için Lampinen (2001), ilk önce kısıt içeren problemleri kısıtsız forma dönüştürmek için genelleştirilmiş bir yöntem önermiş, daha sonra Kukkonen ve Lampinen (2004), birden çok amaç fonksiyonuna ve kısıt fonksiyonuna sahip problemler için genelleştirilmiş diferansiyel gelişim algoritmasını önermişlerdir. Genelleştirilmiş diferansiyel gelişim, seçim mekanizmasında anne ($\mathbf{p}^{n,i}$) ve çocuk ($\mathbf{c}^{n+1,i}$) bireylerin yarışında çocuğun aşağıdaki şartları sağlaması durumunda çocuk bireyin yarışı kazanmasına olanak tanımaktadır;

- Her iki birey de (anne ($\mathbf{p}^{n,i}$) ve çocuk ($\mathbf{c}^{n+1,i}$)) uygun değil ancak çocuk bireyin sağlamadığı kısıtların sayısı anne bireye göre daha küçüktür.
- Anne birey ($\mathbf{p}^{n,i}$) uygun değilken çocuk birey ($\mathbf{c}^{n+1,i}$) tüm kısıtları sağlayarak uygundur.
- Hem anne birey ($\mathbf{p}^{n,i}$) hem de çocuk birey ($\mathbf{c}^{n+1,i}$) uygundur. Ancak çocuk birey amaç fonksiyonu değeri bakımından daha iyidir veya çocuk bireyin yakın civarında bulunan potansiyel anne bireyinkine oranla daha iyidir.

Genelleştirilmiş diferansiyel gelişim yönteminin kısıt fonksiyonlarının sağlanmasına büyük önem verdiği görülmektedir. Ancak üçüncü şartta bulunan yakın civarlardaki potansiyelin tespitinin çok kolay tayin edilebilme olasılığı düşüktür.

3.11.3. Pareto diferansiyel gelişim

Çok amaçlı optimizasyon problemlerinde amaç fonksiyonlarının birbiriyle çeliştiği durumlarda çok sık kullanılan yaklaşımlardan biri de Pareto yaklaşımıdır. Optimizasyon probleminin yapısına göre amaç fonksiyonlarının ve/veya kısıt fonksiyonlarının her birinin önemi veya bir başka deyişle probleme katkısı eşit olabileceği gibi eşit olmayabilmektedir. Probleme olan katkının eşit olmayıp büyük olduğu amaç veya kısıt fonksiyonlarının katkısı eğer diğer fonksiyonlar gibi eşit alınırsa gerçek optimum sonuç yerine koşulları optimum sağlayan başka sonuçlara ulaşılabilir. Ancak bu katkı kullanıcıdan kullanıcıya değişebilmektedir. Örneğin bir amaç fonksiyonunun katkısı bir

kullanıcıya göre fazla olabilirken problemin kullanım şartlarına göre diğer kullanıcıya göre o amaç fonksiyonunun önemi veya katkısı düşük olabilir. Bu şekilde birçok optimum sonuç elde edilebilir ve tüm bu optimum sonuçlar ile bir set oluşturulabilir. Bu setin şekillendirdiği çözümler ilk defa Pareto cephesi adıyla bir mühendis ve ekonomist olan Pareto (1886) tarafından ortaya atılmıştır. Bu yaklaşıma göre hiçbir Pareto çözümü diğer Pareto çözümlerini baskılayamaz çünkü Pareto cephesindeki her çözüm aynı öneme sahiptir.

Diferansiyel gelişim algoritmasında çok amaçlı optimizasyon problemleri için kullanılan Pareto çözümleri Pareto diferansiyel gelişim olarak adlandırılmaktadır. Diferansiyel gelişim, doğası gereği bir Pareto cephesi yerine sadece bir tane optimum çözüme yakınsamaktadır. Bu nedenle diferansiyel gelişimin verdiği her bir sonuç dikkate alındığında birçok optimum çözüm elde edilecektir. Bu çözümler saklanarak yardımcı Pareto popülasyonu oluşturulmaktadır. Ancak diferansiyel gelişim algoritması sonucu mevcut popülasyon içindeki çözümlerden daha iyi çözümler elde edildiği durumlarda popülasyon içindeki mevcut çözüm yenisiyle güncellenmektedir. Ancak teoride bu şekilde sonsuz çözüm elde edilebilir. Pratikte ise hafıza ve zaman sınırlamalarından dolayı belirli sınırlayıcı kriterler yardımcı Pareto popülasyonunu belirli değerlerde tutmaktadır.

Başka bir yaklaşımda, Abbass ve ark. (2001), yardımcı Pareto popülasyonu yerine tüm Pareto çözümlerinin tutulduğu bir Pareto seti kullanıp çözümler elde ederek sadece Pareto setinde bulunan üyelerin mutasyona uğratılmasına olanak vermişlerdir. Ancak yine de bu iki yaklaşım ile Pareto popülasyonu gelişmeye açık değildir. Pareto popülasyonunun da bağımsız olarak gelişimine izin verilmesi ve bu sayede en iyi Pareto cephesinin oluşturulabilmesi için çeşitli yaklaşımlar önerilmiştir.

Ayrık gelişim yaklaşımında hem temel popülasyonun gelişimine izin verilmekte hem de Pareto popülasyonunun gelişimine izin verilmektedir. Her ortak bireyin biri temel popülasyona diğeri de Pareto popülasyonuna olmak üzere iki çocuk birey meydana getirmesi sağlanmaktadır. Böylece en iyi bireylerden elde edilen çözümlerin mevcut Pareto popülasyonundaki daha kötü bireylerle yer değiştirmesine aynı zamanda Pareto popülasyonunun da gelişimine izin verilmektedir.

İşbirliği ile gelişim yaklaşımında yine hem temel popülasyon hem de Pareto popülasyonunun kendi kendilerine gelişimine izin verilmekle birlikte burada temel popülasyon kendi çocuk bireyini Pareto popülasyonu da kendi çocuk bireyini meydana getirmektedir.

Birleşik ortak gelişim yaklaşımında temel popülasyon ile Pareto popülasyonu birleşik olarak ele alınmakta ve bu sayede iki bağımsız gelişim yerine sadece bir gelişim kullanılmaktadır. Burada ortak birey sadece bir çocuk birey meydana getirmektedir.

Baskın olmayan sıralamalı yaklaşım ilk defa Srinivas ve Deb (1995) tarafından çok amaçlı optimizasyon problemlerinin genetik algoritmalar ile çözümü için önerilmiş ve 2002 yılında diferansiyel gelişime uygulanmıştır (Madavan 2002). Bu yaklaşımda popülasyon sayısının iki katına sahip bir tampon popülasyonu (\mathbf{B}^{n+1}) kullanılmaktadır. Tampon popülasyon mevcut popülasyondaki tüm bireyleri (\mathbf{P}^n) ve yeni üretilen çocuk bireyleri (\mathbf{c}^{n+1}) içermektedir. Bir nesil tamamlanınca tampon popülasyon içindeki bireyler baskınlık derecelerine göre sıralanmakta ve yeni nesildeki popülasyon bu şekilde oluşturulmaktadır (Qing 2009).

Geliştirilen DEBVs algoritması için yukarıda açıklanan çok-amaçlı optimizasyon yaklaşımları evrimsel algoritmalarındaki popülasyon özellikleri benzer olduğu için DE`de de olduğu gibi aynı şekilde rahatlıkla uygulanabilmektedir.

3.12. Deney Tasarımı ve Cevap Yüzeyi Yöntemi

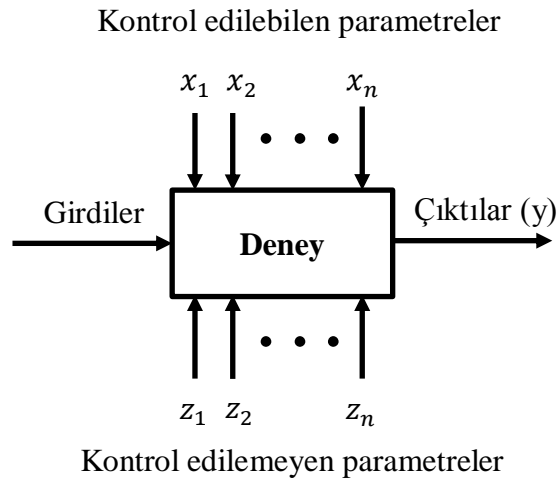
Deney tasarımı istatistikî bir kavram olup deneyi yapanın kontrolünün ne derecede olduğuna göre değişebilen kontrollü bilgi toplama çalışmalarıdır. Deney tasarımı özellikle yapılan deneysel çalışmalarda parametrelerin fazla olması nedeniyle tüm deney kombinasyonlarının yapılmasının imkânsız olduğu durumlarda başvuru bir yöntemdir. Bu yöntem kullanılarak her biri farklı parametrelerden oluşan deney kombinasyonları oluşturulur ve parametre sayısı indirgenip deney sayısı azaltılarak yüksek doğrulukta sonuçlara ulaşılmaktadır.

Deney bir sistemin girdilerinde değişiklik yapılması sonucunda çıktılarda gözlemlenen değişiklikler ve bu değişikliklerin analiz edilmesi işlemi olarak tanımlanmaktadır (Montgomery 2008). Deneylerin düzenlenmesi fikri ilk olarak 1925`te Fisher (1925)

tarafından ortaya atılmış ve deney tasarımı adını verdiği yöntemde deneyler arasındaki ilişkiyi ölçmede kullanılabilecek yeni bir teknik olan Varyans Analizi'ni (ANOVA) geliştirmiştir. Deney tasarımında işlem adımları aşağıdaki sıraya göre yapılmaktadır, ancak bazen bir önceki adıma dönülüp gerekli iyileştirmeler yapılabilmektedir (Dean ve Voss 1999):

1. Deneyin amacının tanımlanması.
2. Uygulama kaynaklarının tümünün belirlenmesi:
 - a. İşlem faktörleri ve düzeyleri,
 - b. Deneysel birimler,
 - c. Engel faktörleri, gürültü faktörleri ve kovaryansları.
3. Uygulamada tüm deney birimlerinin atanması için bir kural oluşturulması.
4. Deneysel prosedür için ölçü birimlerinin belirlenmesi.
5. Pilot bir deneyin gerçekleştirilmesi.
6. Modelin oluşturulması.
7. Analiz taslağının oluşturulması.
8. Gerekli olan gözlem sayısının hesaplanması.
9. Yapılan işlemlerin gözden geçirilmesi ve gerekliyse güncelleştirmelerin yapılması.

Parametreler kontrol edilebilen ve edilemeyen olmak üzere iki şekilde deneye etki etmektedirler (Şekil 3.54).



Şekil 3.54 Deney tasarımında sistemin genel modeli (Montgomery 2001)

Yukarıda verilen şekilde bir deney doğru bir şekilde hazırlırsa sonuçlar o düzeyde güvenilir olacak ve minimum zamanda alınabilecektir. Deney tasarımında her bir parametrenin sonuca olan etkisi diğer parametreler sabit tutularak bulunabilir. Bunun için tam faktöriyel, kesirli faktöriyel ve Taguchi yöntemi gibi farklı yöntemler geliştirilmiştir.

Tam faktöriyel deney tasarımında en az iki veya daha fazla sayıda parametrenin seviyelerinin birbirleri ile çarpımları sonucu oluşan kombinasyonlar kullanılır. Örneğin iki parametreye sahip bir deneyde üç seviye olduğu varsayılırsa, birbirinden farklı 9 (3^2) adet deneyin yapılması tasarlanır (Çizelge 3.4).

Çizelge 3.4. Tam faktöriyelde deney tasarımı tablosu

Deney Numarası	Parametreler ve Seviyeleri		Sonuç Deneyler
	P ₁	P ₂	
1	S ₁	S ₁	y ₁
2	S ₁	S ₂	y ₂
3	S ₁	S ₃	y ₃
4	S ₂	S ₁	y ₄
5	S ₂	S ₂	y ₅
6	S ₂	S ₃	y ₆
7	S ₃	S ₁	y ₇
8	S ₃	S ₂	y ₈
9	S ₃	S ₃	y ₉

Kesirli faktöriyel deney tasarımında zaman ve deney sayısını azaltmak için deney sayısı ile orantılı olarak deneyler azaltılır. Örneğin 5 parametrelili ve 4 seviyeli bir deney tasarımı tam faktöriyel ile yapıldığında $4^5 = 1024$ adet deneyin yapılması gerekirken kesirli faktöriyel deney tasarımında $\frac{1}{2}$ oranı seçildiğinde deney sayısı 512'ye $\frac{1}{4}$ oranı seçildiğinde deney sayısı 128'e düşmektedir. Ancak deney sayısındaki azalma deney sonuçlarının hassasiyetini düşürmektedir.

Taguchi yönteminde (2002) sistem tasarımı, parametre tasarımı ve tolerans tasarımı birlikte deney tasarımını oluşturmaktadır. Bu yöntemde parametrelerin farklı seviyeleri

arasından optimum kombinasyon belirlenmektedir. Bu yöntem ile daha az sayıda deney yapılarak istenilen sonuca ulaşmak mümkün olabilmektedir.

Sistem tasarımı ilk uygulanan adımdır ve bu aşamada kaliteli bir tasarım gerçekleştirmek amacıyla teknolojik tasarım araçları arasından en iyi ve uygun olanları seçilmeye çalışılır.

Parametre tasarımı Taguchi yönteminde en önemli adımdır. Bu adımda üretimde kullanılacak parametrelerin en iyi seviyelerinin tespit edilmesi işlemi gerçekleştirilir. Bunun için kaliteye etkisi çok az olan ve kaliteyi olumsuz etkileyen kontrol edilebilen ve edilemeyen etkenler belirlenir. Kontrol edilemeyen etkenlerin etkisini azaltmak için bu etkenlere neden olan parametreler belirlenir ve bu parametrelerin tasarımda kullanımı minimize edilmeye çalışılır. Bunun için her bir parametre için belirli ortogonal dizi tablolarından gürültü oranları (S/N) hesaplanır. Gürültü oranları o parametre için deney tasarımında deney ve tekrar sayılarının tespitinde kullanılır.

Tolerans tasarımı parametre tasarımı kullanılarak arzu edilen değerler elde edilemediği durumlarda yapılan ilave çalışmaları kapsamaktadır. Bu aşamada deneylerden elde edilen sonuçlardan faydalanılarak hedef değerden olan sapmalar ve kayıplar hesaplanır ve kayıp fonksiyonu hesap edilir. Böylece hedeften sapma miktarlarını veren bir fonksiyon elde edilmiş olur. Bu şekilde etkin parametreler ve etki oranları belirlenerek deney tasarımı bu bilgiler ışığında oluşturulur. Böylece daha verimli deneyler gerçekleştirilmiş olmaktadır (Taguchi 2002).

Mühendislik problemlerinde deney tasarımı ile uygun deneyler tasarlanarak büyük kazançlar elde edilmektedir. Ancak büyük çoğunlukla sayısal çözümler gerektiren deneylerde tüm deney süreci çok fazla zaman gerektirebilmektedir. Örneğin bir sistemin sonlu elemanlar analizi ile çözümünü eleman sayısına bağlı olarak günler hatta haftalar alabilmektedir. Dolayısıyla yukarıda bahsedilen deney tasarımı yöntemlerini kullanmak ve uygun sonuçlara ulaşmak aylar alabilmektedir. Bu nedenle belirlenen deneyleri karakterize eden ve deneylerde kullanılmayan parametre değerlerini de hesap edip uygun sonuçları bulabilecek yeterlikte ve güvenilirlikte olan yöntemler kullanılmaktadır.

Cevap yüzeyi yöntemi, Box ve Wilson (1951) tarafından geliştirilmiş ve girdi değişkenleri ile çıktı değişkenleri arasındaki ilişkilerin modellenmesinde kullanılan istatistiksel ve matematiksel tekniklerin bir kombinasyonudur. Dikkatli tasarım ve deney analizleri ile bir cevap veya çıkış değişkeni ile tahmin edilebilen giriş değişkenleri arasında kurulacak bağıntının dereceleri de belirlenebilmektedir. Cevap yüzeyi yaklaşımında, girdi değişkenleri ile çıktı değişkenleri arasındaki bilinmeyen, muhtemelen oldukça karmaşık yapıdaki gerçek ilişkiye genellikle birinci veya ikinci dereceden bir Polinom ile yaklaşım yapılmaktadır. Bu yöntemde tasarlanmış deneylerle elde edilmiş verilere aşağıda genel yapısı verilen polinomlar uydurulur (Box ve Draper 2007):

$$\begin{aligned}
 f(x_1, x_2) = & \beta_0 + (\beta_1 x_1 + \beta_2 x_2) + (\beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2) \\
 & + (\beta_{111} x_1^3 + \beta_{222} x_2^3 + \beta_{112} x_1^2 x_2 + \beta_{122} x_1 x_2^2) \\
 & + (\beta_{1111} x_1^4 + \beta_{2222} x_2^4 + \beta_{1112} x_1^3 x_2 + \beta_{1222} x_1 x_2^3 \\
 & + \beta_{1122} x_1^2 x_2^2) + \dots
 \end{aligned} \tag{3.42}$$

Burada x_1 ve x_2 bağımsız değişkenleri veya parametreleri, f uydurulan fonksiyonu ve β_0, β_1, \dots ilgili oldukları x terimleri ile tanımlanan ve deneylerdeki veriler ile hesap edilen katsayıları ifade etmektedir.

Cevap yüzeyi fonksiyonunun yapısı Polinom derecesine ve bağımsız değişkenlere bağlı olarak değişim göstermektedir. Polinomdaki katsayıların sayısı tasarım değişkenlerinin sayısına ve polinomun derecesine bağlı olarak değişmektedir (Çizelge 3.5).

İki bağımsız değişkene sahip planar veya birinci-dereceden cevap yüzeyi fonksiyonu şu şekildedir:

$$f(x_1, x_2) = \beta_0 + (\beta_1 x_1 + \beta_2 x_2) \tag{3.43}$$

Bu formülasyonda eğik bir düzlemi temsil eden üç katsayı mevcuttur ve bu düzlemin parametreleri β_0, β_1 ve β_2 katsayıları ile belirlenmektedir.

Çizelge 3.5. Polinom derecesine ve değişken sayısına bağlı olarak değişen katsayı değerleri

Cevap Yüzeyi Fonksiyonundaki Katsayı Adedi			
Polinom Derecesi	Değişken Sayısı		
	2 (x_1, x_2)	3 (x_1, x_2, x_3)	4 (x_1, x_2, x_3, x_4)
1 (Planar)	3	4	5
2 (Kuadratik)	6	10	15
3 (Kübik)	10	20	35
4 (Kuartik)	15	35	70

İki değişkenli kuadratik ikinci-dereceden cevap yüzey fonksiyonunda 6 adet katsayı mevcuttur:

$$f(x_1, x_2) = \beta_0 + (\beta_1 x_1 + \beta_2 x_2) + (\beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2) \quad (3.44)$$

İki değişkenli kübik üçüncü-dereceden cevap yüzey fonksiyonunda 10 adet katsayı mevcuttur:

$$\begin{aligned} f(x_1, x_2) = & \beta_0 + (\beta_1 x_1 + \beta_2 x_2) \\ & + (\beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2) \\ & + (\beta_{111} x_1^3 + \beta_{222} x_2^3 + \beta_{112} x_1^2 x_2 \\ & + \beta_{122} x_1 x_2^2) \end{aligned} \quad (3.45)$$

İki değişkenli kuartik dördüncü-dereceden cevap yüzey fonksiyonunda 15 adet katsayı mevcuttur:

$$\begin{aligned}
f(x_1, x_2) = & \beta_0 + (\beta_1 x_1 + \beta_2 x_2) + (\beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2) \\
& + (\beta_{111} x_1^3 + \beta_{222} x_2^3 + \beta_{112} x_1^2 x_2 + \beta_{122} x_1 x_2^2) \\
& + (\beta_{1111} x_1^4 + \beta_{2222} x_2^4 + \beta_{1112} x_1^3 x_2 + \beta_{1222} x_1 x_2^3 \\
& + \beta_{1122} x_1^2 x_2^2)
\end{aligned} \tag{3.46}$$

Yukarıda açıklanan cevap fonksiyonlarının deney tasarımlarına ne ölçüde uyduğunu ölçmek için R-kare (R^2) testi uygulanmaktadır. R-kare aşağıdaki bağıntı ile hesaplanmaktadır:

$$R^2 = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \bar{y})^2} \tag{3.47}$$

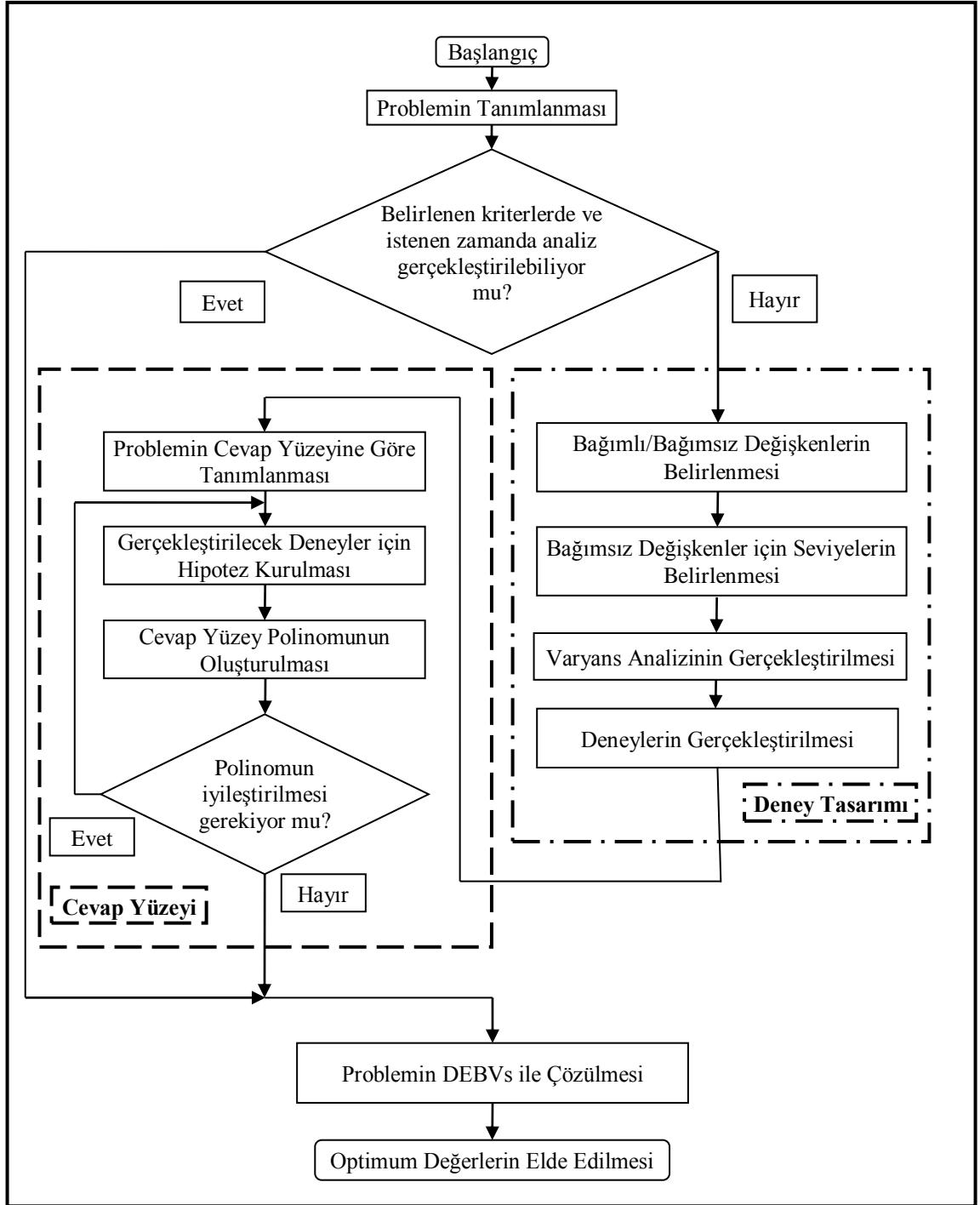
Burada; y_i gözlenen deney tasarımı değerlerini, f_i cevap fonksiyon değerlerini ve \bar{y} gözlenen deney tasarımı değerlerinin ortalamasını ifade etmektedir. Gözlenen deney tasarımı değerlerinin ortalaması (\bar{y}) tüm gözlenen deney tasarımı değerlerinin toplanarak toplam deney sayısına bölünmesiyle elde edilir:

$$\bar{y} = \frac{1}{n} \sum_i^n y_i \tag{3.48}$$

Burada; n gözlenen deney tasarımı değerlerinin toplam sayısını ifade etmektedir.

R-kare “0” ile “1” arasında değerler alabilmektedir; “1” cevap fonksiyonuyla elde edilen değerlerin gözlenen deney tasarımı değerlerine birebir uygunluğunu ifade etmekte iken “0” bu uygunluk değerinin çok kötü olduğunu göstermektedir. Literatürde mühendislik problemleri için uydurulan cevap yüzeyi fonksiyonunda R-kare değerlerinin 0,95`ten büyük olması beklenmektedir. 0,99`dan büyük R-kare değerlerine sahip cevap yüzeyi fonksiyonlarının uygunluğunun mükemmel olduğu söylenebilir.

Geliştirilen DEBVs algoritmasında deney tasarımı ve cevap yüzeyi yöntemleri ile problemler akış diyagramı Şekil 3.55`te verildiği şekilde ele alınmaktadır.



Şekil 3.55 Deney tasarımı ve cevap yüzeyi yöntemleri ile bir problemin çözümlenmesi

3.13. Geliştirilen Algoritmanın Test Problemleri ile Değerlendirilmesi

Geliştirilen bir yöntemin doğruluğunu göstermek için en pratik yol o yöntemi veya algoritmayı kullanarak daha önceden çözülmüş problemleri tekrar çözmektir. Geliştirilen algoritma ile daha iyi sonuçların elde edilmesi veya en azından önceki

sonuçlarla aynı sonucun elde edilmesi beklenir. Bu sayede yeni geliştirilen yöntemin etkinliğinin ispat edilmesi amaçlanır. Bu amaçla literatürde birçok test problemi kullanılmaktadır. Bazıları gerçek hayatta karşılaşılan ve çözümü aranan bir problem iken zorluk derecesi ve geliştiricilerin o problemi çok sık kullanması nedeniyle popülerlik kazanmış, bazıları ise sırf geliştirilen algoritmayı test etmek için ortaya atılmış test fonksiyonlarından oluşmaktadır.

Geliştirilen algoritmanın etkinliğini göstermek için test fonksiyonları seçme işleminde dört önemli noktaya dikkat edilmiştir; test fonksiyonlarını rastgele seçmekten kaçınılmış ve fonksiyonlar hakkında yeterli bilgiye sahip olunmaya çalışılmıştır. Geliştirilen DEBVs algoritmasının etkinliği göstermek için seçilen test fonksiyonları en az 30 defa çözdürülmüş ve ortalaması alınmıştır. Temelsiz veya dayanağı olmayan genelleştirmelerden kaçınmak için geliştirilen DEBVs algoritmasında örnek olay incelemeleri altı kontrol parametresinin tüm durumları için denenmiş ve en iyi sonuçların elde edildiği parametre değerleri test fonksiyonlarında kullanılmıştır. Taraflı veya önyargılı karşılaştırma yapmamak için geliştirilen DEBVs algoritması ile karşılaştırma yapılan DE algoritmasında benzer parametreler (popülasyon sayısı, çaprazlama oranı ve diferansiyel ölçeklendirme faktörü) aynı değerlerde kullanılmış ve bu değerler kullanılarak elde edilen sonuçlar karşılaştırılmıştır. Problem aktarılırken dikkatli olunmaya çalışılmış ve geliştirilen algoritmanın etkinliğini göstermek için ele alınan test fonksiyonlarının sadece birkaçının (muhtemelen en iyi sonuçların elde edildiği fonksiyonlar) üzerinde durmaktan kaçınılmış ve test fonksiyonlarının hepsi ele alınmıştır.

Runnarson ve Yao (2000) geliştirilen algoritmaların etkinliğini ölçmek için literatürde kullanılan ve her biri farklı bir özelliğe sahip 13 test fonksiyonu belirlemişler ve geliştirdikleri algoritmada bu test fonksiyonlarını kullanmışlardır. Daha sonra bu 13 test fonksiyonu objektif kriterlere sahip olmalarından dolayı diğer araştırmacılar tarafından da kullanılmışlardır (Montes ve ark. 2005).

Geliştirilen DEBVs algoritması yukarıda bahsi geçen bu 13 test fonksiyonuna tabi tutulmuştur. Test fonksiyonları hakkında detaylı bilgiler EK1`de verilmiştir. Maksimizasyon problemleri “-“ ile çarpılarak minimizasyon problemine dönüştürülmüştür. Her bir test fonksiyonu 30 defa çözdürülmüş ve maksimum 500 000

hesaplama sayısındaki en iyi değerler ile ortalama değerler elde edilmiştir. Literatürde mevcut olan ve şu ana kadar elde edilen en iyi sonuçlar 500 000 hesaplama sayısı ile elde edilemezse 500 000 hesaplama sayısında elde edilen sonuçlar çizelgede verilmiştir. Her bir test fonksiyonu DE ile ve DEBVs ile çözdürülmüş elde edilen sonuçlar aşağıdaki çizelgelerde verilmiştir. Çözümlerde kullanılan popülasyon sayıları çizelgelerde ayrı ayrı belirtilmiş ve diferansiyel ölçeklendirme faktörü her iki yöntem için 0,85 ve çaprazlama oranı yine her iki yöntem için 0,9 olarak kullanılmıştır. Literatürde mevcut en iyi sonuç 500 000 hesaplama sayısından önce elde edilmiş ise hesaplama 500 000 hesaplama sayısına ulaşıncaya kadar devam edilmiş ve elde edilen sonuç koyu renk ve kırmızı ile vurgulanmıştır. Eğer elde edilen sonuç mümkün olan en iyi teorik sonuç ise hangi hesaplama sayısında bu sonuca ulaşıldığı belirtilmiştir. Çizelgelerde literatürde şu ana kadar elde edilen en iyi sonuçlar “Minimum Değer” sütunu ile verilmiştir (Montes ve ark. 2005). Ayrıca 30 tekrar için elde edilen ortalama fonksiyon değerleri ve standart sapma değerleri ayrı sütunlarda ele alınmıştır. Bilinen en iyi sonuca hangi hesaplama sayısı ile ulaşıldığı çizelgelerin sonunda verilmiş ve benzer şekilde hesaplama sayılarının ortalaması ve standart sapma değerleri farklı sütunlar ile vurgulanmıştır. Geliştirilen DEBVs algoritması ile 2. ve 3. test problemleri haricindeki geri kalan 11 test probleminde literatürde şu ana kadar hesaplanan sonuçlardan daha iyi sonuçlara ulaşılmıştır. 2. test probleminde en iyi sonuca 1 733 000 hesaplama sayısında ulaşılmıştır.

Çizelge 3.6. İlk test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON			Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer	
g01	1. Test Fonksiyonu			13	$0 \leq x_{1,2,\dots,9,13} \leq 1$ $0 \leq x_{10,11,12} \leq 100$	$x^* = [1; 1; 1; 1; 1; 1; 1; 1; 1; 3; 3; 3, 1]$ $f^* = -15$	
	30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	22 500	22 500	0	$x^* = [0,9935 \ 0,9916 \ 0,9939$ $0,9903 \ 0,9656 \ 0,9997 \ 0,9532$ $0,9597 \ 0,9745 \ 2,8254 \ 2,8853$ $2,8706 \ 0,9902]$ $-14,2721$ $g^* = [-0,3191 \ -0,3291 \ -0,2732 \ -$ $5,1229 \ -5,0472 \ -5,0804 \ -0,1209 \ -$ $0,0673 \ -0,0234]$	-13,6475	0,3794	

DEBVS	22 500	22 500	0	$x^* = [1,0000 \ 1,0000 \ 1,0000$ $1,0000 \ 0,9998 \ 1,0000 \ 1,0000$ $0,9999 \ 1,0000 \ 2,9997 \ 2,9992$ $2,9997 \ 0,9999]$ -14,9979 $g^* = [-0,0012 \ -0,0006 \ -0,0011 \ -$ $5,0003 \ -5,0006 \ -5,0002 \ -0,0001 \ -$ $0,0007 \ -0,0001]$	-14,7141	0,6430
30 Tekrar Hedef: Hesap. (500 * 1000 = 500 000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	500 000	500 000	0	$x^* = [0,999999923496497$ $0,999999546543434$ $0,99999998405164$ $0,99999995630841$ $0,999999722105266$ $0,9999999902761$ $0,99999999765212$ $0,99999962954148$ $0,99999999965529$ $2,999994383432239$ $2,999997450775128$ $2,999998902416991$ $0,999999554775528]$ $-14,999987296473536$ $g^* = [-9,225712769733718e-06 \ -$ $6,870347446508163e-06 \ -$ $-4,556910685238336e-06 \ -$ $5,000005004539739$ $-4,999998921572344 \ -$ $5,000001084824325$ $-5,329934708253603e-06 \ -$ $2,548795605239462e-06$ $-1,023456834392533e-06]$	- 14,9999799817 98754	5,8951e-006
DEBVS	500 000	500 000	0	$x^* = [1,000000000000000 \ 1 \ 1 \ 1$ $1,000000000000000 \ 1 \ 1 \ 1$ $1,000000000000000 \ 3$ $2,999999999999999$ $2,999999999999999$ $1,000000000000000]$ -15 $g^* = [0 \ 0 \ -1,776356839400251e-15$ $-4,999999999999999 \ -$ $5,000000000000001 \ -$ $5,000000000000001 \ 0 \ -$ $8,881784197001252e-16 \ -$ $4,440892098500626e-16]$	- 14,9999999999 99991	9,2361e-015
30 Tekrar Hedef: Amaç. (-15 = - 14,9999) 250*2000	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	197 000	213 925	6 596,5	-15	-15	0
DEBVS	76 000	111 742	10 5630	-15	-15	0

Çizelge 3.7. İkinci test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer			
	2. Test Fonksiyonu	20	$0 \leq x_{1,2,\dots,20} \leq 10$	$x^* =$ [3,16246061572185 ; 3,12833142812967; 3,09479212988791; 3,06145059523469; 3,02792915885555; 2,99382606701730; 2,95866871765285; 2,92184227312450; 0,49482511456933; 0,48835711005490; 0,48231642711865; 0,47664475092742; 0,47129550835493; 0,46623099264167; 0,46142004984199; 0,45683664767217; 0,45245876903267; 0,44826762241853; 0,44424700958760; 0,44038285956317] f^* = -0,80361910417			
g02	30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı		Amaç Fonksiyon Değeri			
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	22 500	22 500	0	$x^* = [9,3829 \ 5,1920 \ 2,8291$ $3,4823 \ 2,9328$ $3,0314 \ 2,4521 \ 3,2284 \ 1,9223$ $0,7871$ $0,4709 \ 0,9134 \ 0,2360 \ 0,2937$ $0,1371$ $0,0699 \ 2,7697 \ 0,4081 \ 0,5068$ $0,2934]$ $-0,5364$ $g^* = [-1,7062 \ -108,6606]$	-0,4844	0,0203
DEBVS	22 500	22 500	0	$x^* = [3,2155 \ 3,1213 \ 3,1038$ $3,0621 \ 3,0553$ $3,0000 \ 2,9541 \ 2,9366 \ 0,5043$ $0,4894$ $0,4620 \ 0,4795 \ 0,4740 \ 0,4573$ $0,4805$ $0,4511 \ 0,4466 \ 0,4340 \ 0,4625$ $0,4314]$ $-0,8029$ $g^* = [-0,0035 \ -119,9787]$	-0,7601	0,0491	
30 Tekrar Hedef: Hesap. (200 * 2 500 = 500 000)	Hesaplama Sayısı		Amaç Fonksiyon Değeri				
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma	
DE	500 000	500 000	0	$x^* = [3,3551 \ 3,2168 \ 3,3030$ $2,9170 \ 2,8495 \ 2,4777 \ 0,6964$	-0,5971	0,0194	

				2,7880 0,9224 0,1572 0,7490 0,3595 0,6598 0,4498 0,6267 0,3522 0,6785 1,0132 1,0814 0,3785] -0,6507 g* = [-0,2762 -120,9682]		
DEBVS	500 000	500 000	0	x* = [3,162448778302733 3,128764896853114 3,095028191233483 3,062194025222006 3,028547953687835 2,994078431801336 2,958117578684523 2,921459661288773 0,495082968326328 0,487685536293448 0,482584600024605 0,476647788559696 0,470800199537319 0,466328840469743 0,461564364916814 0,456592093818286 0,452911257902559 0,448040378839722 0,444448129321190 0,440377507218193] -0,803618784948967 g* = [-6,794812467125766e-07 - 120,0662968176983]	- 0,80325127754 7240	0,0020
100 Tekrar Hesap. (200 * 10 000 = 2 000 000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	
DE	---	---	---	---	---	---
DEBVS	1 733 000	1 861 300	6,6493e +004	x* = [3,162460639631373 3,128331430966780 3,094792158528767 3,061450615502404 3,027929180141706 2,993826082754763 2,958668737952891 2,921842273428165 0,494825126532184 0,488357116254093 0,482316415583706 0,476644712223468 0,471295518567547 0,466231015452027 0,461420050685261 0,456836670979450 0,452458752468350 0,448267605731362 0,444246998523650 0,440382859542269] -0,80361910412559 g* = [-7,771561172376096e-16 -1,200674160385498e+02]	- 0,80317866036 3918	0,002168595 172661

Çizelge 3.8. Üçüncü test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer			
g03	3. Test Fonksiyonu	10	$0 \leq x_{1,2,\dots,10} \leq 1$	$x^* =$ [0,316227766016838; 0,316227766016838; 0,316227766016838; 0,316227766016838; 0,316227766016838; 0,316227766016838; 0,316227766016838; 0,316227766016838; 0,316227766016838; 0,316227766016838] f^* = -1,000500100010			
	30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı		Amaç Fonksiyon Değeri			
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	22 500	22 500	0	$x^* = [0,7572 \ 0,3032 \ 0,3415$ $0,3154 \ 0,3027 \ 0,3208 \ 0,3199$ $0,3324 \ 0,2957 \ 0,3141]$ $-0,9830$ $h^* = 0$	-0,9530	0,0141
	DEBVS	22 500	22 500	0	$x^* = [0,3524 \ 0,3162 \ 0,3162$ $0,3162 \ 0,3162 \ 0,3162 \ 0,3162$ $0,3162 \ 0,3162 \ 0,3162]$ $-0,999999999681618$ $h^* = 0$	- 0,9999999930392 27	2,2557e-008
	30 Tekrar Hedef: Hesap. (500 * 1 000 = 500 000)	Hesaplama Sayısı		Amaç Fonksiyon Değeri			
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	500 000	500 000	0	$x^* = [0,9633 \ 0,3162 \ 0,3162$ $0,3162 \ 0,3162 \ 0,3162 \ 0,3162$ $0,3162 \ 0,3162 \ 0,3162]$ $-1,0000000000000002$ $h^* = 0$	- 1,00000000000000 02	4,7908e-016
	DEBVS	500 000	500 000	0	$x^* = [0,068741655882951$ $0,316227765843566$ $0,316227766025377$ $0,316227765150628$ $0,316227766091490$ $0,316227766266603$ $0,316227766414794$ $0,316227765712351$ $0,316227766442713$ $0,316227766681516]$ $-1,0000000000000003$ $h^* = 0$	- 1,00000000000000 03	3,9977e-016
	30 Tekrar Hedef: Amaç. (---)	Hesaplama Sayısı		Amaç Fonksiyon Değeri			
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma	

---	---	---	---	---	---	---
DE	---	---	---	---	---	---
DEBVS	---	---	---	---	---	---

Çizelge 3.9. Dördüncü test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer			
	4. Test Fonksiyonu	5	$78 \leq x_1 \leq 102$ $33 \leq x_2 \leq 45$ $27 \leq x_{3,4,5} \leq 45$	$x^* = [78; 33;$ $29,995256025681$ $5985; 45;$ $36,775812905788$ $2073]$ f^* $= -30\ 665,539$			
g04	30 Tekrar Hedef: Hesap. (50 * 50 = 2 500)	Hesaplama Sayısı		Amaç Fonksiyon Değeri			
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	2 500	2 500	0	$x^* = [78,2518\ 33,0196\ 30,0525$ $44,8209\ 36,7554]$ $-30\ 631$ $g^* = [-0,0047\ -91,9953\ -11,1222$ $-8,8778\ -4,9832\ -0,0168]$	-30 563	51,6720
	DEBVS	2 500	2 500	0	$x^* = [78,0004\ 33,0000\ 30,0046$ $44,9999\ 36,7579]$ $-30\ 664$ $g^* = [-0,0029\ -91,9971\ -11,1624$ $-8,8376\ -4,9992\ -0,0008]$	-30 637	42,1238
	30 Tekrar Hedef: Hesap. (200 * 2 500 = 500 000)	Hesaplama Sayısı		Amaç Fonksiyon Değeri			
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	500 000	500 000	0	$x^* = [78\ 33$ $29,995256025681560$ $44,999999999999930$ $36,775812905788335]$ $-30\ 665,539$ $(-30665,53867178332)$ $g^* = [0\ -92\ -11,159499691073108$ $-8,840500308926892\ -5\ 0]$	-30 665, 53867178331	1,1101e- 011	
DEBVS	500 000	500 000	0	$x^* = [78\ 33$ $29,995256025681556$ $44,999999999999865$ $36,775812905788370]$ $-30\ 665,539$ $(-30665,53867178332)$ $g^* = [0\ -92\ -11,159499691073080$ $-8,840500308926920\ -5\ 0]$	-30 665, 53867178331	1,1101e- 011	
30 Tekrar Hedef:	Hesaplama Sayısı		Amaç Fonksiyon Değeri				
	En İyi	Ortalama	Standart	En İyi	Ortalama		

Amaç. (-30665,539) 200*2 500			Sapma			
DE	59 600	64 647	2,5187e +003	-30 665,539	-30 665,539	0
DEBVS	25 800	32 420	2,7182e +003	-30 665,539	-30 665,539	0

Çizelge 3.10. Beşinci test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON			Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer	
g05	5. Test Fonksiyonu			4	$0 \leq x_{1,2} \leq 1200$ $-0,55 \leq x_{3,4} \leq 0,55$	$x^* =$ [679,945148297028709; 1026,06697600004691; 0,118876369094410433; - 0,396233485215178266] $f^* = 5 126,4967140071$	
	30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	22 500	22 500	0	$x^* = 1,0e+003 * [0,6971$ 1,0078 0,0001 - 0,0004] 5 128,1 $g^* = [-0,0413 -1,0587]$ $h^* = [0,0691 0,0262$ 0,0635] (Kısıtlar sağlanmıyor)	5 277,5 (Kısıtlar sağlanmadığı için)	199,7112
	DEBVS	22 500	22 500	0	$x^* = 1,0e+003 * [0,6885$ 1,0170 0,0001 - 0,0004] 5 126,880187992216 $g^* = [-0,0381 -1,0619]$ $h^* = [-0,0001 -0,0001$ -0,0000]	5 302,2	198,7885
	30 Tekrar Hedef: Hesap. (500 * 1000 = 500 000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	500 000	500 000	0	$x^* = [716,2790$ 987,5237 0,0931 - 0,4085] 5 133,4 $g^* = [-0,0484 -1,0516]$ $h^* = [-0,0000 -0,0001$ -0,0001]	5 214,0 (Tümünde kısıtlar sağlanmamıştır)	53,0591
	DEBVS	500 000	500 000	0	$x^* =$ [6,799451488151658e+	5 249,5	152,8841

				02 1,026066975446403e+0 3 0,118876368724924 -0,396233485390555] 5 126,496714007099 $g^* = [-0,0349 \ -1,0651]$ $h^* = [-0,0000 \ -0,0000$ -0,0000]		
30 Tekrar Hedef: Amaç. (5126,49671 40071) ---*---	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	
DE	---	---	---	---	---	---
DEBVS	---	---	---	---	---	---

Çizelge 3.11. Altıncı test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değeri			
g06	6. Test Fonksiyonu	2	$13 \leq x_1 \leq 100$ $0 \leq x_2 \leq 100$	$x_1^* = 14,09500 \ 000000000064$ $x_2^* = 0,8429607892154795668$ $f^* = -6 \ 961,81387558015$ $g_1 = 0$ $g_2 = 0$			
	30 Tekrar Hedef: Hesap. (50 * 50 = 2 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	2 500	2 500	0	$x_1 = 14,1158$ $x_2 = 0,8845$ $-6,9151e+003$ $g_1 = -0,0352$ $g_2 = -0,0065$	-6,6237e+003	232,2079
	DEBVS	2 500	2 500	0	$x_1 = 14,0952$ $x_2 = 0,8435$ -6,9612e+003 $g_1 = -0,0684e - 3$ $g_2 = -0,4299e - 3$	-6,9157e+003	88,2441
30 Tekrar Hedef: Hesap. (150 * 200 = 30000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri			
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma	
DE	30 000	30 000	0	$x_1 = 14,09500 \ 0008435047$ $x_2 = 0,842960806891268$	- 69618,137477055 63	8,162304257 154681e-05	

					$-6\,961,813855695171$ $g_1 = -6,475602276623249e-09$ $g_2 = -1,039447283801565e-08$		
DEBVS	30 000	30 000	0	$x_1 = 14,094999999999994$ $x_2 = 0,842960789215464$ $-6\,961,81387558016$ $g_1 = 0$ $g_2 = 0$	-	$6961,813875579227$ $2,483973784629165e-09$	
30 Tekrar Hedef: Amaç. (- 6961,813875 58015) 150*500	Hesaplama Sayısı			Amaç Fonksiyon Değeri			
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama		
DE	51 450	54 355	1,4820e+003	$x_1^* = 14,09500\ 00000000000$ $x_2^* = 0,84296078921547956$ $f^* = -6\,961,81387558015$ $g_1 = 0$ $g_2 = 0$	$x_1^* = 14,09500\ 0000$ $x_2^* = 0,84296078921$ $f^* = -6\,961,813875$ $g_1 = 0$ $g_2 = 0$	0	
DEBVS	29 100	34 645	7,8007e+003	$x_1^* = 14,09500\ 00000000000$ $x_2^* = 0,84296078921547956$ $f^* = -6\,961,81387558015$ $g_1 = 0$ $g_2 = 0$	$x_1^* = 14,09500\ 0000$ $x_2^* = 0,84296078921$ $f^* = -6\,961,813875$ $g_1 = 0$ $g_2 = 0$	0	

Çizelge 3.12. Yedinci test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer
g07	7. Test Fonksiyonu	10	$-10 \leq x_{1,2,\dots,10} \leq 10$	$x^* = [2,17199634142692;$ $2,3636830416034;$ $8,77392573913157;$ $5,09598443745173;$ $0,990654756560493;$ $1,43057392853463;$ $1,32164415364306;$ $9,82872576524495;$ $8,2800915887356;$ $8,3759266477347]$ $f^* = 24,30620906818$
30	Hesaplama Sayısı		Amaç Fonksiyon Değeri	

Tekrar Hedef: Hesap. (150 * 150 = 22 500)	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	22 500	22 500	0	$x^* = [2,0995 \ 2,5930 \ 6,9643$ $5,3007 \ -0,2443 \ -0,1673$ $1,5095 \ 9,7177 \ 8,3984$ $9,5397]$ 48,0059 $g^* = [-0,7062 \ -5,9759 \ -0,6972$ $-59,4103 \ -6,8889 \ -8,1936 \ -$ $8,2862 \ -55,6138]$	60,1355	5,3687
DEBVS	22 500	22 500	0	$x^* = [2,1717 \ 2,4120 \ 8,6872$ $5,0910 \ 0,9563 \ 1,3391$ $1,2989 \ 9,7890 \ 8,2460$ $8,3475]$ 24,6099 $g^* = [-0,0491 \ -0,0817 \ -0,0145$ $-3,2312 \ -0,0837 \ -0,0670 \ -$ $6,5674 \ -49,7492]$	24,7814	0,1377
30 Tekrar Hedef: Hesap. (200 * 2 500 = 500 000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	500 000	500 000	0	$x^* = [2,1724 \ 2,3625 \ 8,7733$ $5,0944 \ 0,9898 \ 1,4295$ $1,3226 \ 9,8295 \ 8,2813$ $8,3762]$ 24,3074 $g^* = [-1,6e-05 \ -0,0005 \ -0,0001$ $-0,0038 \ -3,5e-05 \ -0,0023 \ -$ $6,1474 \ -50,0256]$	24,3087	7,1042e- 004
DEBVS	500 000	500 000	0	$x^* = [2,171996373294364$ $2,363682971039643$ $8,773925742619590$ $5,095984510952072$ $0,990654763848056$ $1,430573977482267$ $1,321644210781545$ $9,828725809329887$ $8,280091671849839$ $8,375926657486783]$ 24,306209068179780 $g^* = [0 \ 0 \ 0$ $-1,136868377216160e-13$ $-7,105427357601002e-15$ $-5,329070518200751e-15$ $-6,148503419099406$ $-50,023961760377055]$	24,30620906 8179786	4,6649e- 015
30 Tekrar Hedef: Amaç. (24,3062090 6818) 200*2 500	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	
DE	>500 000	>500 000	---	24,30620906818	24,30620906 818	0

DEBVS	386375	415529	1,7571e+004	24,30620906818	24,30620906818	0
-------	--------	--------	-------------	----------------	----------------	---

Çizelge 3.13. Sekizinci test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON			Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer	
g08	8. Test Fonksiyonu			2	$0 \leq x_1 \leq 10$ $0 \leq x_2 \leq 10$	$x_1^* = 1,22797135260752599$ $x_2^* = 4,24537336612274885$ $f^* = -0,0958250414180359$	
	30 Tekrar Hedef: Hesap. (50 * 50 = 2 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	2 500	2 500	0	$x_1 = 1,227962221865918$ $x_2 = 4,245344974522944$ $-0,095825039419903$ $g_1 = -1,737453756193062$ $g_2 = -0,167768065342254$	- 0,09582477065 3568	2,7522906 75822162e -07
	DEBVS	2 500	2 500	0	$x_1 = 1,227971323752199$ $x_2 = 4,245373407194828$ -0,095825041418028 $g_1 = -1,737459835237099$ $g_2 = -0,167763214793801$	- 0,09582504141 5082	3,9092763 13114911e -12
	30 Tekrar Hedef: Hesap. (150 * 50 = 7 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	7 500	7 500	0	$x_1 = 1,227971341313585$ $x_2 = 4,245373813811280$ $-0,095825041417656$ $g_1 = -1,737460198723795$ $g_2 = -0,167763032809292$	- 0,09582504139 0419	2,3187002 34515172e -11
	DEBVS	7 500	7 500	0	$x_1 = 1,227971352760583$ $x_2 = 4,245373366250582$ -0,095825041418036 $g_1 = -1,737459723049925$ $g_2 = -0,167763263895441$	- 0,09582504141 8036	5,1540810 00242946e -18
	30 Tekrar Hedef: Amaç. (- 0,095825041 4180359) 150*1000	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
En İyi		Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma	
DE	19 200	21 805	1,3256e+003	$x_1^* = 1,22797135260752599$ $x_2^* = 4,24537336612274885$	$x_1^* = 1,227971352$	0	

					$f^* = -0,0958250414180359$	$x_2^* = 4,245373366$ $f^* = -0,09582504$	
DEBVS	10 350	12 775	1,077413089 542373e+03		$x_1^* = 1,22797135260752599$ $x_2^* = 4,24537336612274885$ $f^* = -0,0958250414180359$	$x_1^* = 1,227971352$ $x_2^* = 4,245373366$ $f^* = -0,09582504$	0

Çizelge 3.14. Dokuzuncu test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON			Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer		
g09	9. Test Fonksiyonu			7	$-10 \leq x_{1,2,\dots,7} \leq 10$	$x^* = [2,33049935147405174;$ 1,95137236847114592; - 0,477541399510615805; - 4,36572624923625874; - 0,624486959100388983; - 1,03813099410962173; - 1,5942266780671519] $f^* = 680,630057374402$		
	30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri			
		En İyi	Ortalama	Standart Sapma	En İyi		Ortalama	Standart Sapma
	DE	22 500	22 500	0	$x^* = [1,9408 \ 1,9718 \ -0,1069$ 4,3770 -0,6084 1,4172 1,7452] 685,1793 $g^* = [-0,6300 -257,3990 -$ 149,3828 -4,6138]		693,1379	4,3016
	DEBVS	22 500	22 500	0	$x^* = [2,3367 \ 1,9497 \ -0,4678$ 4,3682 -0,6252 1,0321 1,5976] 680,6318744543892 $g^* = -0,0003 -252,6125 -$ 144,8436 -0,0008]		680,6353	0,0026
30 Tekrar Hedef: Hesap. (200 * 2 500 = 500 000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri				
	En İyi	Ortalama	Standart Sapma	En İyi		Ortalama	Standart Sapma	
DE	500 000	500 000	0	$x^* = [2,3305 \ 1,9514 \ -0,4775$ 4,3657 -0,6245 1,0381 1,5942] 680,6300573744197 $g^* = [-3,271161119755561e-12$ -2,525617301078945e+02 -1,448781765036424e+02 -4,604316927725449e-12]		680,6300573 744694	2,7402e- 011	

	DEBVS	500 000	500 000	0	$x^* = [2,330499401217553$ $1,951372362882665$ $-0,477541432815311$ $4,365726255590722$ $-0,624486989907106$ $1,038130992809247$ $1,594226742666276]$ 680,6300573744014 $g^* = [-1,509903313490213e-14$ $-2,525617156567783e+02$ $-1,448781778653175e+02$ $-2,842170943040401e-14]$	680,6300573744018	3,4689e-013
	30 Tekrar Hedef: Amaç. (680,630057 374402) ---*---	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	
	DE	---	---	---	---	---	---
	DEBVS	---	---	---	---	---	---

Çizelge 3.15. Onuncu test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer			
g10	10. Test Fonksiyonu	8	$100 \leq x_1 \leq 10000$ $1000 \leq x_{2,3} \leq 10000$ $10 \leq x_{4,\dots,8} \leq 1000$	$x^* = [579,306685017979589;$ $1359,97067807935605;$ $5109,97065743133317;$ $182,01769963061534;$ $295,601173702746792;$ $217,982300369384632;$ $286,41652592786852;$ $395,601173702746735]$ $f^* = 7\ 049,24802052867$			
	30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
	DE	22 500	22 500	0	$x^* = 1,0e+003 * [0,1001$ $1,4948\ 6,8053\ 0,0811$ $0,2507$ $0,1126\ 0,2257\ 0,3485]$ $8\ 400,2$ $g^* = 1,0e+004 * [-0,0001\ -0,0000\ -0,0000$ $-1,7038\ -0,4227\ -4,2748]$	9 127,4	377,8319
DEBVS	22 500	22 500	0	$x^* = 1,0e+003 * [0,7043$ $1,2614\ 5,1317\ 0,1908$ $0,2951$ $0,2081\ 0,2949\ 0,3950]$ 7 097,4 $g^* = [-0,0028\ -0,0020\ -$	7 338,7	174,9241	

					0,0009 -430,6685 -852,1061 - 539,6687]		
30 Tekrar Hedef: Hesap. (125 * 4000 = 500 000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri			
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma	
DE	500 000	500 000	0	$x^* = 1,0e+003 * [0,5844$ 1,3500 5,1157 0,1824 0,2954 0,2176 0,2870 0,3954] 7,050094478436117e+03 $g^* = [-0,0001 -$ 1,002509018288578e-05 - 1,645048091081236e-06 -18,0800 -17,1749 -31,8771]	7,0514e+003	0,5657	
DEBVS	500 000	500 000	0	$x^* = [5,793066675516638e+02$ 1,359970681451825e+03 5,109970671525174e+03 1,820176981718450e+02 2,956011731389930e+02 2,179823018281550e+02 2,864165250328521e+02 3,956011731389930e+02] 7,049248020528663e+03 $g^* = [0 0 0 0 0]$	7,049248020528667e+03	4,426694418715532e-12	
30 Tekrar Hedef: Amaç. (7049,24802 052867) ---*---	Hesaplama Sayısı			Amaç Fonksiyon Değeri			
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma	
DE	---	---	---	---	---	---	
DEBVS	---	---	---	---	---	---	

Çizelge 3.16. On birinci test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer		
g11	11. Test Fonksiyonu	2	$-1 \leq x_1 \leq 1$ $-1 \leq x_2 \leq 1$	$x_1^* = \pm 0,7071$ $x_2^* = 0,7071$ $f^* = 0,7499$		
	30 Tekrar Hedef: Hesap. (5 * 5 = 25)	Hesaplama Sayısı			Amaç Fonksiyon Değeri	
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	25	25	0	$x_1 = -0,7063$ $x_2 = -0,7058$ 0,7 50001274612949 $h_1 = 0$	0,753969396691275	0,015315982021811

DEBVS	25	25	0	$x_1 = 0,70777969115592$ $x_2 = 0,70148253630607$ 0,7 50000906477678 $h_1 = 0$	0,752475076996049	0,008345597198943
30 Tekrar Hedef: Hesap. (50 * 50 = 2 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	2 500	2 500	0	$x_1 = 0,70710677539278$ $x_2 = -0,5941626223307$ 0,7 500 0000000000 $h_1 = 0$	0,750000000000659	2,151485320745926e-12
DEBVS	2 500	2 500	0	$x_1 = 0,70710677858926$ $x_2 = 0,91402433348654$ 0,749999 = 0,750000 $h_1 = 0$	0,750000000000000	0
30 Tekrar Hedef: Amaç. (0,75) 150*100	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	6150	9870	1,0595e+003	$x_1^* = \pm 0,7071$ $x_2^* = 0,7071$ $f^* = 0,7499$	$x_1^* = \pm 0,7071$ $x_2^* = 0,7071$ $f^* = 0,7499$	0
DEBVS	4050	5080	589,2134	$x_1^* = \pm 0,7071$ $x_2^* = 0,7071$ $f^* = 0,7499$	$x_1^* = \pm 0,7071$ $x_2^* = 0,7071$ $f^* = 0,7499$	0

Çizelge 3.17. On ikinci test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer		
g12	12. Test Fonksiyonu	6	$0 \leq x_{1,2,3} \leq 10$ $x_{4,5,6} = 1,2, \dots, 9$	$x^* = [5; 5; 5; 5; 5; 5]$ $f^* = -1$		
	30 Tekrar Hedef: Hesap. (50 * 50 = 2 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri	
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama
DE	2 500	2 500	0	$x_{1,2,3,4,5,6} = 4,8072,5,9277,4,8810,5,0000,60000,50000$ -0,9909	-0,9294	0,0495

				$g_1 = -0,0059$		
DEBVS	2 500	2 500	0	$x_{1,2,3,4,5,6}$ = 5,0645,4,9071,5,0899, 5,0000,5,0000,5,0000 -0,9998 $g_1 = -0,0416$	-0,9543	0,0434
30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	22 500	22 500	0	$x_{1,2,3,4,5,6}$ = 4,9550,4,9767,5,0356, 5,0000,5,0000,5,0000 -0,999961704670864 $g_1 = -0,0587$	-0,9946	0,0048
DEBVS	22 500	22 500	0	$x_{1,2,3,4,5,6}$ = 4,9826,4,9926,5,0168, 5,0000,5,0000,5,0000 -1,0000 $g_1 = -0,0619$	-0,9994	0,0019
30 Tekrar Hedef: Amaç. (-1) 150*100	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	38 850	44 305	2,6459e+003	$x_{1,2,3,4,5,6}^* = 5,5,5,5,5,5$ $f^* = -1$	$x_{1,2,3,4,5,6}^* = 5,5,5,5,5,5$ $f^* = -1$	0
DEBVS	23 550	30 380	2,5818e+003	$x_{1,2,3,4,5,6}^* = 5,5,5,5,5,5$ $f^* = -1$	$x_{1,2,3,4,5,6}^* = 5,5,5,5,5,5$ $f^* = -1$	0

Çizelge 3.18. On üçüncü test fonksiyonu için elde edilen sonuçlar

AD	FONKSİYON	Tasarım Değişkeni Sayısı	Tasarım Değişkeni Aralığı	Minimum Değer		
g13	13. Test Fonksiyonu	5	$-2,3 \leq x_{1,2} \leq 2,3$ $-3,2 \leq x_{3,4,5} \leq 3,2$	$x^* = [-1,71714224003;$ 1,59572124049468; 1,8272502406271; -0,763659881912867; -0,76365986736498] $f^* = 0,053941514041898$		
	30 Tekrar Hedef: Hesap. (150 * 150 = 22 500)	Hesaplama Sayısı			Amaç Fonksiyon Değeri	
		En İyi	Ortalama	Standart Sapma	En İyi	Ortalama
DE	22 500	22 500	0	$x^* = [-1,0898$ 0,7028 -2,6997	0,9358	0,2161

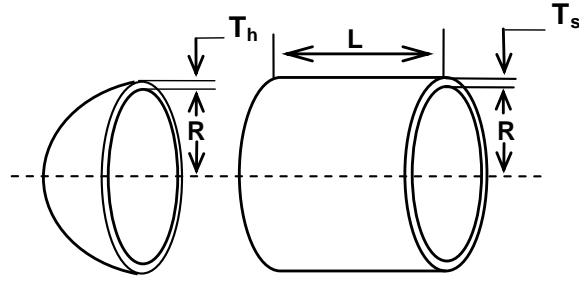
				0,3943 -0,9836] 0,4485 $g^* = [0,0929 \ 0,0414 \ 0,0526]$ (Kısıtlar sağlanmıyor)		
DEBVS	22 500	22 500	0	$x^* = [-1,3461 \ 1,1290 \ -2,3894$ $0,5786 \ -0,9325]$ 0,1410 $h^* = [-0,0268 \ -0,1082 \ -0,1255]$	0,8978	0,3849
30 Tekrar Hedef: Hesap. (100 * 5000 = 500 000)	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	500 000	500 000	0	$x^* = [-1,3162 \ 1,0858 \ 2,4500 \ -$ $0,8079 \ -0,6585]$ 0,1552 $g^* = 1,0e-003 * [-0,0795 \ -0,1216 \ -$ $0,0765]$	0,4118	0,1194
DEBVS	500 000	500 000	0	$x^* = [-1,717141654570403$ $1,595726322464410 \ -$ $1,827252215421233$ $-0,763667264342703$ $0,763667265596057]$ 0,053937847573861 $h^* = [-1,188258475989801e-15 \ -$ $3,000800562896755e-16$ $-3,000800562896755e-16]$	0,3593198 02707527	0,2484
30 Tekrar Hedef: Amaç. (0,05394151 4041898) ---*---	Hesaplama Sayısı			Amaç Fonksiyon Değeri		
	En İyi	Ortalama	Standart Sapma	En İyi	Ortalama	Standart Sapma
DE	---	---	---	---	---	---
DEBVS	---	---	---	---	---	---

3.14. Mühendislik Problemleri ile Geliştirilen Algoritmanın Performansının Değerlendirilmesi

Geliştirilen DEBVs algoritmasının performansının ne derecede etkin olduğunun daha iyi anlaşılması ve gösterilmesi açısından literatürde mevcut bulunan ve daha önce diğer yöntemlerle çözülmüş mühendislik problemleri ele alınmıştır. Bu mühendislik problemlerinin tümü kısıt içermekte, bir kısmı tek amaçlı bir kısmı ise çok-amaçlı olmaktadır. Geliştirilen DEBVs algoritması ile tüm problemler çözülmüş ve elde edilen sonuçlar literatürde elde edilen sonuçlar ile karşılaştırılmıştır.

3.14.1. Basınçlı kap tasarım problemi

Basınçlı kap tasarım problemi birçok tasarım optimizasyon yönteminin etkinliğinin üzerinde denendiği problemlerden birisidir (Şekil 3.56). Dairesel sarılıp kaynatılmış bir sacın her iki ucuna yarı küresel sac monte edilmiş bir basınçlı kaptaki amaç fonksiyonu toplam maliyetin minimizasyonu olarak tanımlanmıştır (Kannan ve Kramer 1994). Burada malzeme maliyeti, sac şekillendirme maliyeti ve kaynak maliyetlerinin toplamı toplam maliyeti oluşturmaktadır. Problem dört tasarım değişkenine sahiptir; sac kalınlığı (T_s), kafa kısmının sac kalınlığı (T_h), iç yarıçap (R) ve kabın silindirik kısmının uzunluğu (L). T_s ve T_h üretilmiş hazır rulo sacların kalınlıkları olduğu için 0,0625 inç değerinin tamsayı katları olarak tanımlanmıştır. R ve L tasarım değişkenlerinde böyle bir sınırlama yoktur ve belirtilen sınırlar dâhilinde sürekli değerler alabilmektedirler.



Şekil 3.56. Basınçlı kabın tasarım parametreleri (Kannan ve Kramer 1994)

Basınçlı kabın matematiksel modelinde, T_s , T_h , R ve L parametreleri sırasıyla x_1 , x_2 , x_3 and x_4 ile ifade edilmektedir. Amaç fonksiyonu; malzeme, imalat ve kaynak işlem fiyatlarını içeren toplam maliyeti minimize eden aşağıda verilen bağıntı ile hesaplanmaktadır.

Amaç Fonksiyonu (Minimizasyon):

$$f(x) = 0,6224x_1x_3x_4 + 1,7781x_2x_3^2 + 3,1661x_1^2x_4 + 19,84x_1^2x_3 \quad (3.49)$$

Kısıtlar:

$$g_1(x) = -x_1 + 0,0193x_3 \leq 0$$

$$g_2(x) = -x_2 + 0,00954x_3 \leq 0$$

$$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3} \pi x_3^3 + 1296000 \leq 0$$

$$g_4(x) = x_4 - 240 \leq 0$$

(3.50)

Sınır Kısıtları:

$$0,0625 \leq x_1 \leq 99 \text{ (0,0625`in tamsayı katları)}$$

$$0,0625 \leq x_2 \leq 99 \text{ (0,0625`in tamsayı katları)}$$

$$10 \leq x_3 \leq 200$$

$$10 \leq x_4 \leq 200$$

(3.51)

Çizelge 3.19. Basınçlı kap tasarım probleminde elde edilen sonuçların karşılaştırılması

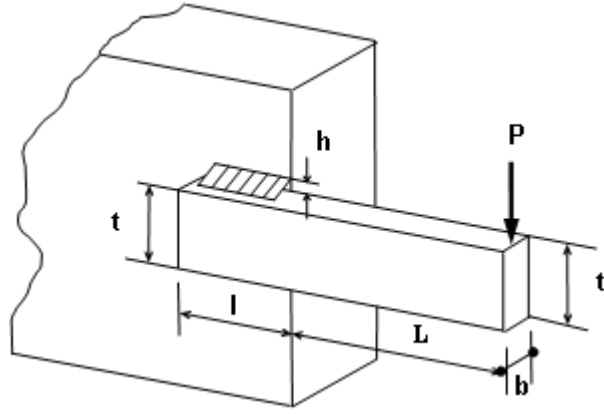
En İyi Sonuçlar	x_1 (T _s)	x_2 (T _h)	x_3 (R)	x_4 (L)	Kısıtlar $g_1(x)$ $g_2(x)$ $g_3(x)$ $g_4(x)$	$f(x)$
						$f_{ort}(x)$
						Standart Sapma
						Fonksiyon Değerlendirme Sayısı
						Tekrar Sayısı
Sandgren (1988)	1,125000	0,625000	47,700000	117,701000	-0,204390 -0,169942 54,226012 -122,299000	8 129,103600 --- --- ---
Kannan ve Kramer (1994)	1,125000	0,625000	58,291000	43,690000	0,000016 -0,068904 -21,220104 -196,310000	7 198,042800 --- --- ---
Deb (1997)	0,937500	0,500000	48,329000	112,679000	-0,004750 -0,038941	6 410,381000 ---

					-3 652,876838 -127,321000	--- --- ---
Coello (2000)	0,812500	0,437500	40,323900	200,000000	-0,034324 -0,052847 -27,105845 -40,000000	6 288,744500 6 293,843232 7,413285 --- 11
Yıldız ve ark. (2004)	0,875000	0,437500	45,025700	143,515000	-0,006003 -0,007954 -312,04189 -96,484743	6 128,055100 --- --- ---
Coello ve Montes (2002)	0,812500	0,437500	42,097398	176,654047	-0,000020 -0,035891 -27,886075 -63,345953	6 059,946341 6 177,253268 130,929702 80 000 30
Karen (2005), Karen ve ark. (2008a)	0,812500	0,437500	42,098436	176,637341	-0,000000 -0,035881 -3,485784 -63,36266	6 059,730000 --- --- ---
DE	0,812500	0,437500	42,098422	176,636968	-0,000000 -0,035881 -0,433426 -63,363032	6 059,719052 6 059,742143 0,015308 5000 30
DEBVs	0,812500	0,437500	42,098446	176,636596	-0,000000 -0,035881 -0,000290 -63,363404	6 059,714337 6 059,714363 0,000035 5000 30

3.14.2. Kaynaklı kiriş tasarım problemi

Kaynaklı kiriş tasarım optimizasyon problemi Ragsdell ve Phillips (1976) tarafından ele alınmış ve Rao (2009) tarafından tekrar düzenlenerek optimizasyon problemi olarak çözülmüştür (Şekil 3.57).

Bu problemde, $h(x_1)$, $l(x_2)$, $t(x_3)$ ve $b(x_4)$ tasarım değişkenleridir ve sırasıyla kaynak yüksekliğini, kaynak uzunluğunu, kiriş yüksekliğini ve kiriş genişliğini ifade etmektedir. Amaç fonksiyonu minimum maliyet olmakla birlikte kesme, eğilme, burkulma gerilmesi ve kirişin yer değiştirmesi gibi çeşitli kısıt fonksiyonları mevcuttur.



Şekil 3.57. Kaynaklı kirişin tasarım parametreleri (Rao 2009)

Kaynaklı kiriş tasarım optimizasyon probleminin matematiksel modeli aşağıdaki gibidir:

Amaç Fonksiyonu (Minimizasyon):

$$f(x) = 1,10471x_1^2 x_2 + 0,04811x_3 x_4 (14,0 + x_2) \quad (3.52)$$

Kısıtlar:

$$g_1(x) = \tau(x) - \tau_{max} \leq 0$$

$$g_2(x) = \sigma(x) - \sigma_{max} \leq 0$$

$$g_3(x) = x_1 - x_4 \leq 0$$

$$g_4(x) = 0,10471x_1^2 + 0,04811x_3 x_4 (14,0 + x_2) - 5,0 \leq 0 \quad (3.53)$$

$$g_5(x) = 0,125 - x_1 \leq 0$$

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

$$g_7(x) = P - P_c(\vec{x}) \leq 0$$

Kısıtlarda mevcut olan gerilme değerlerini Yokota ve ark. (1999) aşağıdaki şekilde tanımlamışlardır:

$$\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau'' \frac{x_2}{2R} + (\tau'')^2}$$

$$\tau' = \frac{P}{\sqrt{2x_1x_2}}$$

$$\tau'' = \frac{MR}{J}$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}$$

$$P = 6000 \text{ lb}$$

$$M = P\left(L + \frac{x_2}{2}\right)$$

$$J = 2\left\{\sqrt{2x_1x_2}\left[\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\} \quad (3.54)$$

$$L = 14 \text{ in}$$

$$\tau_{max} = 13\,600 \text{ psi}$$

$$\sigma(x) = \frac{6PL}{x_4x_3^2}$$

$$\sigma_{max} = 30\,000 \text{ psi}$$

$$\delta(x) = \frac{4PL^3}{Ex_3^3x_4}$$

$$E = 30\,000\,000 \text{ psi}$$

$$\delta_{max} = 0,25 \text{ in}$$

$$P_c(x) = \frac{4,013E \sqrt{\frac{x_3^2 x_4^6}{36}}}{L^2} \left(1 - \frac{x_3}{2L} \sqrt{\frac{E}{4G}} \right)$$

$$G = 12\,000\,000 \text{ psi}$$

Sınır Kısıtları (Rao 2009):

$$0,1 \leq x_1 \leq 2 \text{ (Rao 2009)}$$

$$0,1 \leq x_2 \leq 10 \text{ (Rao 2009)}$$

$$0,1 \leq x_3 \leq 10 \text{ (Rao 2009)}$$

$$0,1 \leq x_4 \leq 2 \text{ (Rao 2009)}$$

(3.55)

Çizelge 3.20. Kaynaklı kiriş tasarım probleminde elde edilen sonuçların karşılaştırılması

En İyi Sonuçlar	x_1 (h)	x_2 (l)	x_3 (t)	x_4 (b)	Kısıtlar $g_1(x)$ $g_2(x)$ $g_3(x)$ $g_4(x)$ $g_5(x)$ $g_6(x)$ $g_7(x)$	$f(x)$
						$f_{ort}(x)$
						Standart Sapma
						Fonksiyon Değerlendirme Sayısı
						Tekrar Sayısı
Deb (1991)	0,248900	6,173000	8,178900	0,253300	-5758,603000 -255,576900 -0,004400 -2,982866 -0,123900 -0,234160 -4 465,270928	2,433116 --- --- 1 350 ---
Ragsdell ve Phillips (1976)	0,245500	6,196000	8,273000	0,245500	-5 743,826517 -4,715097 0,000000 -3,020289 -0,120500 -0,234208 -3 604,275002	2,385937 --- --- --- ---
Siddall (1972)	0,244400	6,218900	8,291500	0,244400	-5 743,502027 -4,015209 0,000000 -3,022561	2,381543 --- --- ---

					-0,119400 -0,234243 -3 490,469418	---
Rao (2009)	0,244400	6,217700	8,291500	0,244400	---	2,381000 --- --- --- ---
Ray ve ark. (2002)	0,167880	4,645894	8,928757	0,219162	-140,417800 -11612,86000 -0,051336 -3,241219 -0,042880 -0,235932 -1 201,706000	1,900000 1,944600 --- 38 744 ---
Coello (1999)	0,208800	3,420500	8,997500	0,210000	-0,337812 -353,902604 -0,00120 -3,411865 -0,08380 -0,235649 -363,232384	1,748309 1,773586 0,011223 900 000 11
DE	0,2015340	3,558397	9,056079	0,206152	-9,092006 -189,999484 -0,004618 -3,418686 -0,076534 -0,235663 -45,591076	1,736723 1,753356 0,010072 5000 30
Coello ve Montes (2002)	0,205986	3,471328	9,020224	0,206480	-0,074090 -0,266220 -0,000490 -3,430040 -0,080980 -0,235510 -58,666400	1,728220 --- --- --- ---
Yıldız ve ark. (2004)	0,205882	3,46774	9,041935	0,20588	-8,069550 -57,468000 0,000000 -3,431140 -0,080880 -0,235570 -15,692800	1,726797 --- --- --- ---
DEBV's	0,205728	3,470883	9,036645	0,205730	-0,817615 -0,225546 -0,000002 -3,432950 -0,080728 -0,235540 -0,059143	1,724897 1,725406 0,001360 5000 30
DEBV's	0,205730	3,470489	9,036624	0,205730	-0,000000 -0,000000 -0,000000 -3,432984 -0,080730 -0,235540 -0,000000	1,724852 1,724852 0,000000 25000 30

3.14.3. Çok-amaçlı kaynaklı kiriş tasarım problemi

Ragsdell ve Phillips (1976) tarafından çözülmüş olan bir kirişe başka bir kirişin dik olarak kaynak ile birleştirilmesi sonucunda ortaya çıkan tek amaçlı kaynaklı kiriş tasarım optimizasyon problemi Deb (1999b) tarafından 6. kısıt fonksiyonunun amaç fonksiyonu olarak ele alınmasıyla çok-amaçlı olarak geliştirilmiştir. Toplam 7 adet olan kısıt fonksiyonu sayısı böylece 4 adete düşürülmüştür.

$h(x_1)$, $l(x_2)$, $t(x_3)$ ve $b(x_4)$ tasarım değişkenleri ve amaç fonksiyonu minimum maliyet olmaktadır. Problemin yapısı gereği maliyetin minimum olması için tüm tasarım değişkenlerinin minimum olması gerekmektedir. Ancak bu defa kirişlerdeki gerilme ve maksimum yer değiştirmeler artmaktadır. Deb (1999b), yer değiştirmenin de minimum olması gerektiğini düşünerek probleme yeni bir amaç fonksiyonu daha eklemiştir. Böylece birbiriyle çelişen iki amaç fonksiyonu elde edilmiş ve kısıtlar 4 adet kısıt fonksiyonuna indirgenmiştir. Maliyetin ve maksimum yer değiştirmenin minimum değerlerini veren tasarım değişkenlerini bulmaya dönüşen problemin matematiksel modeli aşağıdaki gibidir:

Amaç Fonksiyonları (Minimizasyon):

$$\begin{aligned} f_1(x) &= 1,10471x_1^2 x_2 + 0,04811x_3 x_4 (14,0 + x_2) \\ f_2(x) &= \frac{2,1952}{x_3^3 x_4} \end{aligned} \quad (3.56)$$

Kısıtlar:

$$\begin{aligned} g_1(x) &= \tau(x) - \tau_{max} \leq 0 \\ g_2(x) &= \sigma(x) - \sigma_{max} \leq 0 \\ g_3(x) &= x_1 - x_4 \leq 0 \\ g_4(x) &= P - P_c(\bar{x}) \leq 0 \end{aligned} \quad (3.57)$$

Kısıtlarda mevcut olan gerilme değerleri aşağıdaki şekilde tanımlanmıştır (Yokota ve ark. 1999):

$$\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau'' \frac{x_2}{2R} + (\tau'')^2}$$

$$\tau' = \frac{P}{\sqrt{2x_1x_2}}$$

$$\tau'' = \frac{MR}{J}$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}$$

$$P = 6000 \text{ lb}$$

$$M = P\left(L + \frac{x_2}{2}\right)$$

$$J = 2\left\{\sqrt{2x_1x_2}\left[\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\}$$

(3.58)

$$L = 14 \text{ in}$$

$$\tau_{max} = 13\,600 \text{ psi}$$

$$\sigma(x) = \frac{6PL}{x_4x_3^2}$$

$$\sigma_{max} = 30\,000 \text{ psi}$$

$$P_c(x) = \frac{4,013\sqrt{EGx_3^2x_4^6}}{L^2} \left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$$

$$E = 30\,000\,000 \text{ psi}$$

$$G = 12\,000\,000 \text{ psi}$$

Sınır Kısıtları (Deb1999):

$$0,125 \leq x_1 \leq 5$$

$$0,1 \leq x_2 \leq 10$$

$$0,1 \leq x_3 \leq 10$$

$$0,125 \leq x_4 \leq 5$$

(3.59)

Çizelge 3.21. Çok-amaçlı kaynaklı kiriş tasarım probleminde elde edilen sonuçların karşılaştırılması

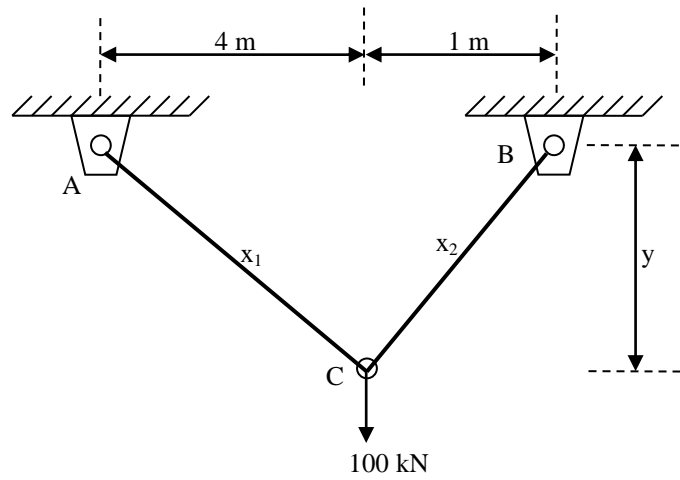
En İyi Sonuçlar	x_1 (h)	x_2 (l)	x_3 (t)	x_4 (b)	Kısıtlar $g_1(x)$ $g_2(x)$ $g_3(x)$ $g_4(x)$	$f_1(x)$
						$f_2(x)$
						Standart Sapma
						Fonksiyon Değerlendirme Sayısı
						Tekrar Sayısı
Deb (2001)	0,222000	7,024000	8,295000	0,244000	---	2,431000 0,015700 --- --- ---
Karen ve ark. (2004b)	0,242245	2,613825	9,865244	0,246825	---	2,115709 0,009263 --- ---
DE	0,230523	2,784602	9,999888	0,240885	-192,728133 -9 076,703027 -0,010362 -4 268,938937	2,108615 0,009113 0,059874 1000 30
DEBV _s	0,241747	2,590743	9,992670	0,242647	-19,504219 -9 198,602557 -0,000900 -4 491,309206	2,102603 0,009067 0,292358 1000 30
DEBV _s	0,242233	2,578461	10,000000	0,242233	-0,000000 -9 193,543626 -0,000000	2,099159 0,009062 0,002850

					-4 442,279619	12500 30
Deb (1999b)	0,422000	2,465000	9,990000	0,439000	---	3,944000 0,005000 ---
Karen ve ark. (2004b)	0,303615	2,120729	9,926882	0,474902	---	3,872224 0,004725 ---
DE	0,417042	1,406607	9,997840	0,481220	-295,513280 -19 522,083144 -0,064177 -75 859,832515	3,836345 0,004565 0,057470 1000 30
DEBV's	0,476867	1,184340	9,997311	0,484162	-30,767721 -19 584,656858 -0,007294 -77 367,896273	3,833464 0,004538 0,351970 1000 30
DEBV's	0,483826	1,162266	10,000000	0,483826	-0,007670 -19 583,038844 -0,000000 -77 208,283082	3,829864 0,004537 0,187719 5000 30
Deb (1999b)	0,422000	2,465000	9,990000	2,558000	---	20,677000 0,001000 ---
DE	1,273866	0,414938	9,997699	2,514171	-454,415376 -27 994,439698 -1,240304 -11 668 052,693	18,175666 0,00087373 0,912778 300 30
DEBV's	1,090471	0,485421	9,999827	2,512669	-277,978969 -27 994,095492 -1,422198 -11 648 655,496	18,147997 0,00087370 2,463337 300 30
DEBV's	1,153156	0,446932	10,000000	2,513308	-0,000017 -0,000280 -1,360152 -11 657 672,868	18,125091 0,00087343 0,638084 3000 30

3.14.4. Çok-amaçlı iki-çubuklu giriş tasarım problemi

İki-çubuklu giriş tasarım optimizasyon problemi Kirsch (1981) tarafından ele alınıp düzenlenmiş bir problem olup Palli ve ark. (1999) tarafından interaktif çok aşamalı e-eşitsizlik kısıt yöntemi kullanılarak çok-amaçlı bir yapıya dönüştürülmüştür. Problem Deb ve ark. (2000) tarafından üç olan amaç fonksiyonu sayısı ikiye düşürülmüş ve yeni

bir kısıt ilave edilerek seçici baskın olmayan sıralamalı genetik algoritma yöntemiyle çözülmüştür. Problemden iki çubuğun bir yükü herhangi bir kopma veya kalıcı şekil değişikliğine uğramadan emniyetle taşıyabilmesi için minimum hacme denk gelen çubuk boyları aranmaktadır. Bu amaç fonksiyonuna ek olarak boyutlar küçüldükçe gerilmeler arttığından ikinci amaç fonksiyonu olarak maksimum gerilmenin minimizasyonu ortaya çıkmaktadır (Deb ve ark. 2000). Problemden her iki çubuğun uzunluğu (x_1 , x_2) ve yük ile mesnet arasındaki dik uzaklık (y) olmak üzere üç tasarım değişkeni mevcuttur (Şekil 3.58).



Şekil 3.58. İki-çubuklu kirişin tasarım parametreleri (Palli ve ark. 1999)

Amaç Fonksiyonları (Minimizasyon):

$$f_1(x) = x_1 \sqrt{16 + y^2} + x_2 \sqrt{1 + y^2} \quad (3.60)$$

$$f_2(x) = \max(\sigma_{AC}, \sigma_{BC})$$

Kısıt:

$$g(x) = \max(\sigma_{AC}, \sigma_{BC}) - 1(10^5) \leq 0 \quad (3.61)$$

Kısıtta mevcut olan gerilme değerleri aşağıdaki şekilde tanımlanmıştır (Deb ve ark. 2000):

$$\sigma_{AC} = \frac{20\sqrt{16 + y^2}}{yx_1}$$

$$\sigma_{BC} = \frac{80\sqrt{1 + y^2}}{yx_2}$$
(3.62)

Sınır Kısıtları (Deb ve ark. 2000):

$$0 \leq x_1 \leq 0,01$$

$$0 \leq x_2 \leq 0,01$$

$$1 \leq y \leq 3$$
(3.63)

Çizelge 3.22. Çok-amaçlı iki-çubuklu kiriş tasarım probleminde elde edilen sonuçların karşılaştırılması

En İyi Sonuçlar	x_1	x_2	y	Kısıt $g(x)$	$f_1(x)$
					$f_2(x)$
					Standart Sapma
					Fonksiyon Değerlendirme Sayısı
					Tekrar Sayısı
Deb ve ark. (2000)	---	---	---	---	0,053040 8 439,000000
DE	0,000748	0,001000	2,999982	-91 567,24981	0,035361 8 432,750183 7,208028 200 30
DEBVs	0,000002	0,001000	3,000000	-91 567,25879	0,031631 8 432,741218 58,531953 200 30

DEBVs	0,000000	0,010000	3,000000	-91 567,25957	0,031623 8 432,740427 0,014146 800 30
Deb ve ark. (2000)	---	---	---	---	0,004070 99 755,000000
Karen ve ark. (2004b)	---	---	---	---	0,004040 99 377,000000
Palli ve ark. (1999)	0,000497	0,000990	1,999000	---	0,004445 89 983,000000
DE	0,000000	0,001633	1,000005	-30 712,36567	0,002309 69 287,634325 51,476002 500 30
DEBVs	0,000000	0,001633	1,000031	-30 717,47712	0,002309 69 282,522879 2 771,405294 500 30
DEBVs	0,000000	0,001633	1,000000	-30 718,17933	0,002309 69 281,820669 1,817586 1000 30

3.14.5. Çok-amaçlı I-kesitli kiriş tasarım problemi

Bu tasarım optimizasyon problemi Osyczka (1985) tarafından ele alınmış ve Coello (1996) ile Coello ve Christiansen (1998) tarafından genetik algoritmalar ile çözülmüştür. Geometrik uzunlukları ifade eden dört tasarım değişkeni mevcuttur (Şekil 3.59).

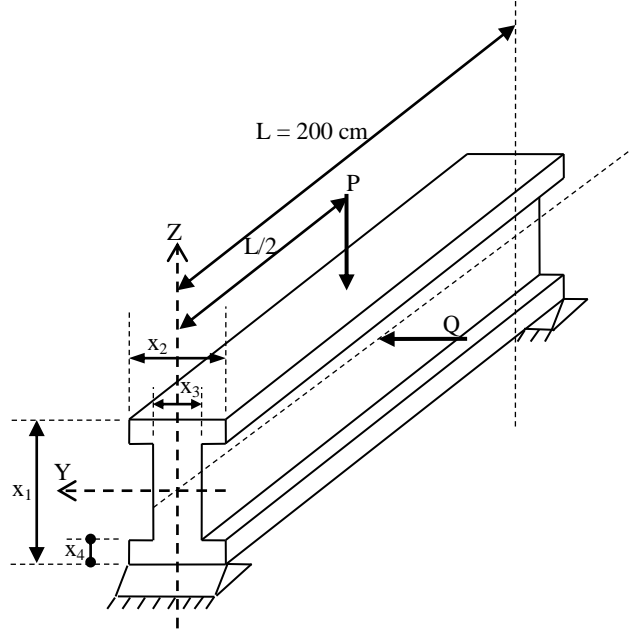
Amaç Fonksiyonları (Minimizasyon):

Kesit Alanı;

$$f_1(x) = 2x_2x_4 + x_3(x_1 - 2x_4) \quad (3.64)$$

Yer Değiştirme;

$$f_2(x) = \frac{Pl^3}{48EI} \quad (3.65)$$



Şekil 3.59. I-kesitli kiriş tasarım probleminin tasarım parametreleri (Coello ve Christiansen 1998)

İkinci amaç fonksiyonunda (yer değiştirme) mevcut olan atalet momenti (I), yük (P) ve elastisite modülü (E) değerleri aşağıdaki şekilde tanımlanmıştır;

$$I = \frac{x_3(x_1 - 2x_4)^3 + 2x_2x_4[4x_4^2 + 3x_1(x_1 - 2x_4)]}{12}$$

$$E = 20\,000 \text{ kN/cm}^2 \quad (3.66)$$

$$P = 600 \text{ kN}$$

Böylece ikinci amaç fonksiyonu aşağıdaki şekilde ifade edilmektedir:

Yer Değiştirme;

$$f_2(x) = \frac{60\,000}{x_3(x_1 - 2x_4)^3 + 2x_2x_4[4x_4^2 + 3x_1(x_1 - 2x_4)]} \quad (3.67)$$

Kısıt:

$$g(x) = 16 - \frac{180\,000x_1}{x_3(x_1 - 2x_4)^3 + 2x_2x_4[4x_4^2 + 3x_1(x_1 - 2x_4)]} - \frac{15\,000x_2}{(x_1 - 2x_4)x_3^3 + 2x_4x_2^3} \geq 0 \quad (3.68)$$

Sınır Kısıtları (Coello ve Christiansen 1998):

$$10 \leq x_1 \leq 80$$

$$10 \leq x_2 \leq 50$$

$$0.9 \leq x_3 \leq 5$$

$$0.9 \leq x_4 \leq 5$$

(3.69)

Çizelge 3.23. Çok-amaçlı I-kesitli kiriş tasarım probleminde elde edilen sonuçların karşılaştırılması

En İyi Sonuçlar	x_1	x_2	x_3	x_4	Kısıt $g(x)$	$f_1(x)$
						$f_2(x)$
						Standart Sapma
						Fonksiyon Değerlendirme Sayısı
						Tekrar Sayısı
Coello ve Christiansen (1998)	80,000000	40,580000	0,920000	3,020000	---	312,770000 0,012700
Karen ve ark. (2004b)	79,988100	41,445370	0,900000	2,952030	---	311,371800 0,012690
DE	79,999998	49,999957	0,900000	2,424873	-11,738476	310,122289 0,012602 0,049494 1200 30
DEBVs	80,000000	50,000000	0,900000	2,424807	-11,738379	310,116083 0,012602 4,286338 1200 30

Coello ve Christiansen (1998)	80,000000	49,590000	1,120000	2,330000	---	315,360000 0,012900
Karen ve ark. (2004b)	79,735000	40,945050	0,900000	2,962040	---	308,991200 0,012890
DE	80,000000	49,999837	0,900000	2,401074	-11,700825	307,784721 0,012708 0,025866 1200 30
DEBVs	80,000000	50,000000	0,900000	2,400933	-11,700617	307,771657 0,012708 5,011037 1200 30

4. BULGULAR VE TARTIŞMA

4.1. DEBVs ile Sürüş Konfor Optimizasyon Probleminin Çözülmesi

Günümüz global pazar ihtiyaçlarına cevap verebilecek ve rekabet edebilecek ürünlerin otomotiv endüstrisinde üretilebilmesi için ürün tasarım maliyetlerinin ve pazara çıkış sürelerinin en aza indirgenmesi gerekmektedir. Bunun için mevcut araç geliştirme sürecinde kullanılabilir sistemik bir yaklaşımın geliştirilmesi büyük önem arz etmektedir.

Sürüş konforu optimizasyon çalışması istenen özelliklerde araç sürüş konforu için araç geliştirme sürecinde kullanılacak sistemik bir yaklaşımın geliştirilerek sürüş konfor problemine çözüm getirme çalışmasıdır. Sürüş konfor hesaplamasında kullanılabilir ve farklı taşıtların farklı yollarda koşturulmasıyla elde edilen bir simülasyon-tabanlı model geliştirilmiştir. Bu model koltuğun oturma bölgesinden, sırt bölgesinden ve ayak bölgesinden elde ettiği ivme değerlerini kullanarak sürüş konfor değerini motor, süspansiyon ve koltuk parametrelerini de dahil ederek hesaplamaktadır. Sürüş konforunu hesaplayabilmek için gerekli verileri sağlamak ve tam-taşıtlı için gerekli parametre değerlerini toplamak amacıyla fiziksel yol testleri gerçekleştirilmiştir. Testlerden elde edilen sonuçlar ile simülasyon çalışması ile elde edilen sonuçlar arasında korelasyon sağlanarak simülasyonlar için gerekli arayüzler oluşturulmuştur. Geliştirilen modeli kullanan sistem taşıtlı tasarım ve/veya mevcut taşıtların sürüş konfor özelliklerinin iyileştirilmesi sürecinde tasarımcıya yardımcı bir simülasyon aracı olarak tasarlanmıştır.

Araç sürüş konforu optimizasyon çalışmasının amacı istenen özelliklerde araç sürüş konforunun sağlanması için araç geliştirme sürecinde kullanılacak sistemik bir yaklaşımın geliştirilmesidir. Araç konfor matematiksel modeli, araç sürüş konfor parametrelerinin tanımları, fiziksel testler ve ölçümlerin belirlenmesi, simülasyon sonuçlarının fiziksel testler ile karşılaştırılması, matematiksel modelin doğruluğunun incelenmesi, araç konfor değerlendirmesi, araç konfor matematiksel modeli optimizasyonu, eşdeğer sistem modeli ve stokastik optimizasyon tekniklerinin kullanılması çalışmaları yapılmıştır. Geliştirilen algoritmanın araç geliştirme sürecinde tasarımcının referans olarak kullanacağı bir yardımcı tasarım aracı olması

beklenmektedir. Geliştirilen simülasyon modelleri fiziksel test sonuçları ile karşılaştırılmış, korelasyon açısından en uygun model belirlenmiş ve optimizasyon çalışmaları doğrulanan model üzerinde gerçekleştirilmiştir.

Sürüş konforu, müşteri memnuniyetini karşılamada bir taşıtta göz ardı edilemeyecek en etkili faktörlerden birisidir. Müşteri memnuniyetini en üst düzeyde tutabilmek için üreticiler sürüş konforunu iyileştirmek için yoğun çalışmalar yapmaktadırlar. Sürüş konforunu mevcut aracın üzerinden alınan veriler ile hesap etme yaklaşımı yerine yeni yöntemler geliştirilmeye çalışılmaktadır. Prototip üretip bu şekilde sürüş konforu değerine ulaşmak hem çok masraflı olmakta hem de gereksiz zaman kayıplarına sebep olmaktadır. Bu nedenle henüz tasarım aşamasının başlarında benzetim çalışmaları ile sürüş konforunu hesaplayabilme ihtiyacı doğmuştur. Son yıllarda bu konu üzerinde yapılan çalışmalar hızla artmaktadır. Birçok sayısal ve deneysel çalışmalar yapılmasına rağmen henüz sürüş konforu ile ilgili yeterli bilgi birikimi arzu edilen düzeyde değildir. Bu nedenle sürüş konforu değerlendirmelerinde yeni yaklaşımlara ihtiyaç vardır. Bu çalışmada taşıt sürüş konfor özelliklerinin değerlendirilmesi ve geliştirilmesi için daha tasarım aşamasında kullanılacak simülasyon-tabanlı bir yaklaşım geliştirilmeye çalışılmıştır.

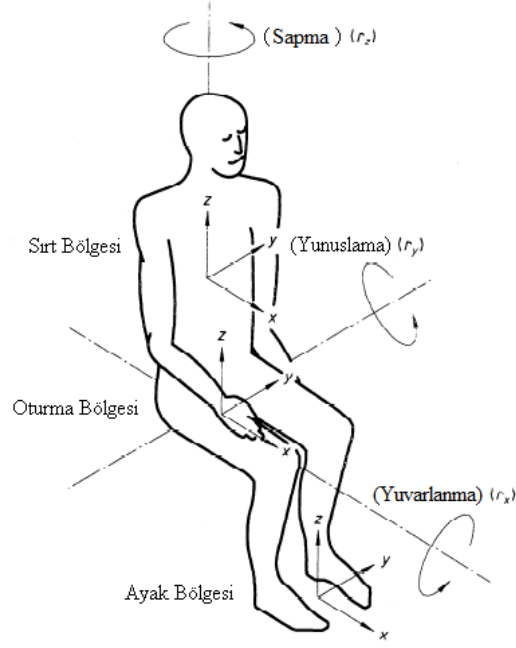
Bir aracın sürüş konforunu uygun bir şekilde değerlendirmek ve sürüş konforuna etki eden faktörleri tanımlamak için kesin bir yöntemden bahsetmek çok zordur, çünkü özellikle değerlendirmelerde subjektif unsurlar ön plana çıkmaktadır. Aynı aracın konforu kişiden kişiye değişmekle birlikte aynı kişinin farklı zamanlarda yaptığı değerlendirmeler arasında dahi tutarsızlıklar görülebilmektedir. Bu nedenle subjektif değerlendirmelerin yanında özellikle objektif değerlendirmeler aracın ilk tasarım aşamasında kullanılabilirliği de göz önüne alındığında büyük öneme sahiptir. Sürücü üzerine gelen titreşimlerin değerlendirilmesi olarak da tanımlanabilen sürüş konforu, lastik özellikleri ve süspansiyon sistemi gibi iki temel unsurdan önemli düzeyde etkilenmektedir. Titreşim kaynakları iki ana kategoride toplanmaktadır; iç titreşimler ve dış titreşimler. Lastiklerden, güç aktarma organlarından ve motordan kaynaklanan titreşimler iç titreşimler olarak sınıflandırılırken yol pürüzlülüğünden kaynaklanan titreşimler ise dış titreşimler olarak sınıflandırılmaktadır. İç titreşimler her bir unsurun (tekerlek, güç aktarma organları ve motor) farklı hızlarda dönmeleriyle farklı

frekanslara sahip olmaları nedeniyle deterministik olarak karakterize edilebilmelerine rağmen dış titreşimler yol pürüzlülüğünün kesin olarak ön görülememesi nedeniyle stokastik yaklaşımlarla karakterize edilmeye çalışılmaktadır. Bu amaçla çeşitli testler gerçekleştirilmekte ve taşıtın belirli bölgelerinden toplanan ivme değerleri ile deneysel olarak hesaplamalar gerçekleştirilmektedir.

4.1.1. Sürüş konforunun hesaplanması

Bir sürücüye sürüş esnasında, koltuğun oturma yüzeyinden, sırt bölgesinden, ayak bölgesinden ve direksiyondan olmak üzere dört farklı bölgeden titreşimler etki etmektedir. Bu titreşimleri karakterize edebilmek veya bu titreşimler için uygun bir kriter bulabilmek için konu üzerinde birçok araştırmacı çalışmaktadır; fakat henüz kabul görmüş tek bir sürüş konfor kriteri mevcut değildir. Sürüş konforu için kullanışlı bir kriter Uluslararası Standartlar Teşkilatı (ISO) tarafından sunulmuştur (ISO 2631 1997). Bu standart, insan üzerine olan etkileri açısından titreşimleri 0,5 Hz ile 80 Hz frekans aralığında konforsuzluk hissi, yorgunluk ve sağlık problemlerine yol açacak düzeyde, 0,1 Hz ile 0,5 Hz frekans aralığındaki titreşimleri ise yol tutmasına yol açacak şekilde sınıflandırmıştır.

Vücut üzerine etki eden titreşimler bir manken üzerine yerleştirilen ivmeölçerler yardımıyla değerlendirmeye tabi tutulabilir. Bunun için manken üzerine toplam 12 adet ivmeölçer yerleştirilebilir. Bu ivmeölçerlerin 3'ü ayak bölgesinde doğrusal ivmeleri ölçecek şekilde, 6'sı oturma bölgesinde hem doğrusal hem de açısal ivmeleri ölçecek şekilde ve 3'ü de sırt bölgesinde doğrusal ivmeleri ölçecek şekilde yerleştirilebilir (Şekil 4.1).



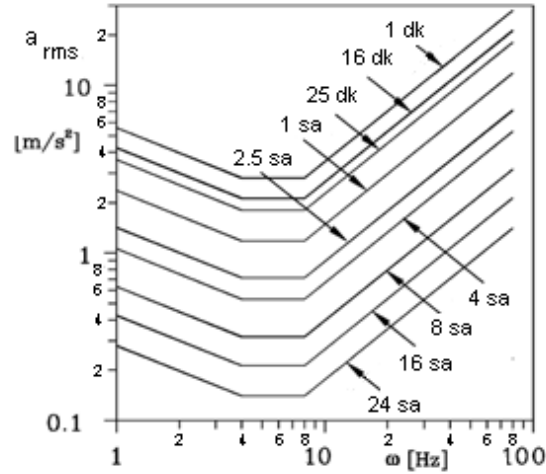
Şekil 4.1. Oturma pozisyonundaki ivmelenme eksenleri (ISO 2631 1997)

Sürücü sürüş esnasında titreşimlere bu üç bölgeden (oturma yüzeyi, sırt ve ayak) maruz kalmaktadır. Aracın sürüş karakteristiklerini ölçmede kullanılan ve insanların ivme değerlerine göre sürüş algılarının nicelendirilmesinde kullanılan yöntemlerden birisi sürüş indeksi oluşturma işlemidir.

Sürüş indeksi hesaplanırken ivme değerlerinin ortalama karekök değerinden (rms) faydalanılır. n adet ivme değerine (a_1, a_2, \dots, a_n) sahip bir veride ortalama karekök değeri şu şekilde hesaplanır;

$$a_{rms} = \sqrt{\frac{1}{n} \sum_{i=1}^n a_i^2} = \sqrt{\frac{a_1^2 + a_2^2 + \dots + a_n^2}{n}} \quad (4.1)$$

Farklı maruz kalma sürelerinde farklı rms değerleri hesap edilebilir. Bu rms değerlerinin konforsuzluk üzerine olan etkilerini incelemek için bir grafik oluşturulabilir. Şekil 4.2`de yatay eksen frekansı, düşey eksen konforsuzluğu ve eğriler ise titreşimlere maruz kalma sürelerini belirtmektedir.



Şekil 4.2. Konforsuzluk eğrileri (ISO 2631 1997)

Farklı maruz kalma süreleri için oluşturulan eğriler birbirlerine benzemekte ve bu eğrilerin minimum olduğu aralık bir başka deyişle insanların olumsuz yönde en fazla etkilendiği ve konforsuz olarak tanımlanan titreşim aralığı 4 – 8 Hz frekans aralığıdır (Şekil 4.2). Bunun en büyük nedeni insan vücudunun bazı bölgelerinin doğal frekansları ile araçtan gelen titreşim frekansının çakışarak rezonans durumunun ortaya çıkmasıdır. Örneğin insanlarda göğüs kafesi sisteminin doğal frekans değerleri 3 – 6 Hz frekans aralığında değişmekte iken kafa ve boyun bölgesinin doğal frekansı 20 – 30 Hz aralığında değişmektedir. Ancak konforsuzluk için esas rahatsız edici durum 4 – 8 Hz arasında değişmektedir. Bunun gibi rezonans frekanslara maruz kalan insanlarda yol tutması gibi çeşitli sağlık sorunları ortaya çıkmakta ve bu frekanslarda sürüş konforu en kötü durum olarak nitelenmektedir. Bu nedenle konforu arttırmak için titreşim frekansları ya 15 – 20 Hz gibi yüksek tutulmaya çalışılmakta ya da 4 Hz'den düşük tutulmaya çalışılmaktadır.

Sürücünün oturma pozisyonunda her bir temas noktası için tüm ivmelenme değerleri belirli zaman periyotlarında toplanır ve bu ivme değerlerinin her bir Kartezyen bileşeni Fourier serisine açılırsa her bir terim için titreşim dozu değeri (VDV) elde edilir (ISO 2631 1997). Bu şekilde her bir terim için dördüncü dereceden titreşim dozu değeri şu şekilde elde edilir:

$$VDV_i = \sqrt[4]{\int_{t=0}^{t=T} a_i^4(t) dt} \quad (4.2)$$

Sürüş indeksi (RI) bu serinin her bir teriminin Çizelge 4.1`den elde edilen frekans ağırlık değerleri ve Çizelge 4.2`den elde edilen eksen faktörleri ile çarpılarak hesap edilmektedir:

$$RI = \sqrt[4]{\sum_{i=1}^3 VDV_i^4} \quad (4.3)$$

Çizelge 4.1. Frekans ağırlık değerleri (ISO 2631 1997)

Ağırlık Faktörü	Frekans f [Hz]	Değer
W_k	$0,5 < f < 2$	0,4
	$2 < f < 5$	$f/5$
	$5 < f < 16$	1
	$16 < f < 80$	$16/f$
W_c	$0,5 < f < 8$	1
	$8 < f < 80$	$8/f$
W_d	$0,5 < f < 2$	1
	$2 < f < 80$	$2/f$
W_e	$0,5 < f < 1$	1
	$1 < f < 20$	$1/f$
W_f	$0,1 < f < 0,125$	$f/0,125$
	$0,125 < f < 0,25$	1
	$0,25 < f < 0,5$	$0,25/f$
W_g	$1 < f < 4$	$f/4$
	$4 < f < 8$	1
	$8 < f < 80$	$8/f$

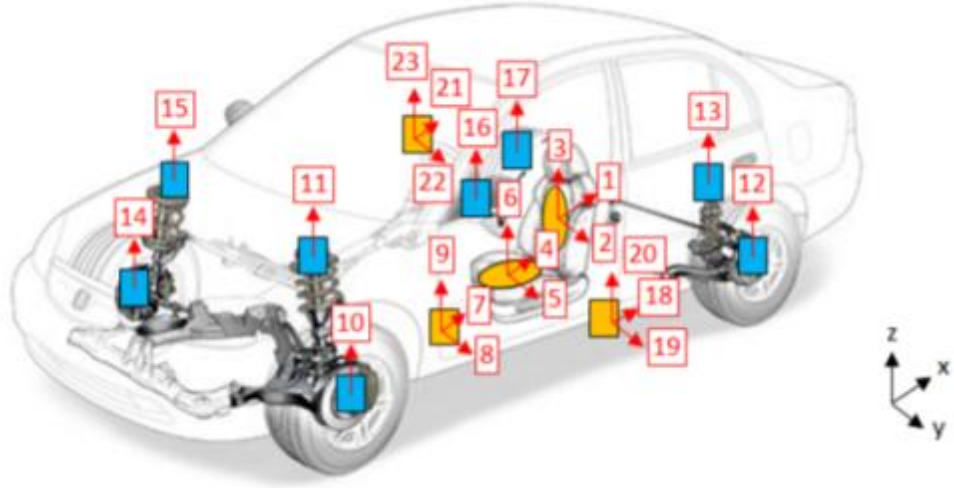
Çizelge 4.2. Ölçülen titreşimin lokal ağırlıkları ve faktör çarpanları (ISO 2631 1997)

Giriş	Eksen	Ağırlık	Faktör
Koltuk	X	W_d	1
	Y	W_d	1
	Z	W_k	1
	R_x	W_e	0,63
	R_y	W_e	0,4
	R_z	W_e	0,2
Sırt	X	W_c	0,8
	Y	W_d	0,5
	Z	W_d	0,4
Ayak	X ve Y	W_k	0,25
	Z	W_k	0,4

Yukarıdaki denklemlerden görüldüğü üzere sürüş indeksini hesaplayabilmek için belirli ivme değerlerinin önceden elde edilmesi gerekmektedir. Bunun için ya fiziksel testler gerçekleştirilmeli ya da benzetim çalışmaları ile bu ivme değerleri hesap edilmelidir.

4.1.2. Fiziksel testler

Bu çalışmada biri sürüş indeksini hesaplayabilmek için gerekli ivme değerlerinin toplanması, diğeri tam-taşıt modeli için gerekli temel parametre değerlerinin toplanması olmak üzere iki farklı fiziksel test gerçekleştirilmiştir. Dört farklı taşıt seçilerek iki farklı tip aracın (binek ve hafif ticari) arasındaki farklılıklar sürüş indeks değerleri açısından karşılaştırılarak belirlenmeye çalışılmıştır. Ayrıca benzer tipteki araçların sürüş indeks değerleri arasındaki değişim karşılaştırılarak model parametreleri arasındaki bağlantılar tespit edilmeye çalışılmıştır. Taşıt seçiminden sonra üzerinde testlerin gerçekleştirileceği uygun yolların seçilmesi de önemli bir işlemdir. Bunun için engelli, stabilize, yamalı, Arnavut kaldırımlı, otoyol ve test pisti olmak üzere taşıt seçimindeki benzer amaçlarla altı farklı yol seçilmiştir. Her bir taşıt üzerine yeterli sayıda ivmeölçer yerleştirilmiş ve veriler toplanmıştır (Şekil 4.3).



Şekil 4.3. Araç üzerine yerleştirilmesi planlanan ivmeölçerlerin konumları (motor bölümü hariç)

Süspansiyon sistemine sadece z-eksenindeki ivmeleri ölçebilen toplam 8 adet ivmeölçer yerleştirilmiştir (Şekil 4.3). Motor bölümüne her biri 3-eksendeki ivmeleri ölçebilecek ve takoz giriş ve çıkışlarına konumlandırılmış şekilde 18 kanala sahip toplam 6 ivmeölçer yerleştirilmiştir. Motor titreşimlerinin sürüş konforu üzerine etkilerinin incelenmesi ve oluşturulacak modele eklenmesi ve böylece kurulacak modelin doğruluğunun artırılması bu çalışmanın amaçlarından birisidir. Koltuk yüzeylerindeki titreşimleri uygun bir şekilde ölçebilmek için içinde her 3 ekseninde oluşan ivmeleri ölçebilen bir ivmeölçer monte edilmiş bir koltuk-ped kullanılmıştır (Şekil 4.4).

Model doğrulaması için yapılan yol testlerinde uygun bir araç seçilerek ivmeölçer ile veri toplanmıştır. Aracın üzerinde gerekli noktalara ivmeölçerler yerleştirilmiştir. Yol testlerinde kullanılan araç üzerine yerleştirilen ivmeölçerlerden bazılarının konumları Şekil 4.4'te verilmiştir.

Test işleminde cihazların yerleştirilmesi, ayarlarının yapılması ve aracın yolda koşturulması hava şartlarının uygun olması halinde maksimum 3 gün sürmektedir. 2 adet Dewe 501 sinyal toplayıcı cihaz ile 40 kanal kullanılarak, 19 ivmeölçer araç üzerine yerleştirilerek test işlemi gerçekleştirilmiştir.



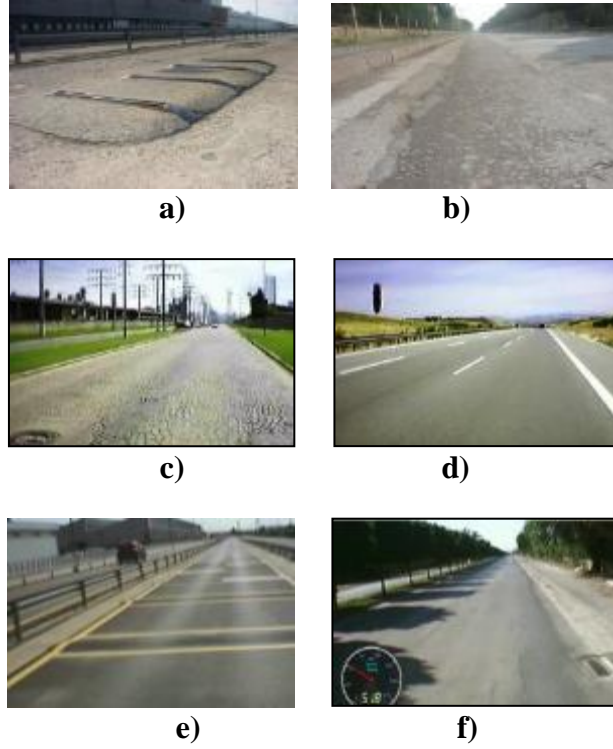
Şekil 4.4. Yol testlerinde kullanılan araç üzerine yerleştirilen ivmeölçerler (akson, direksiyon, koltuk, ayak bölgesi ve döşeme altı)

Dewe 501 cihazı, kendi işletim sistemine sahip olan bir veri toplayıcısıdır. Aynı zamanda kendi monitör ve güç ekipmanları ile 1000 Hz`lik bir örnekleme frekansı ile konfor test işleminde güvenilir bir şekilde kullanılmıştır (Şekil 4.5).



Şekil 4.5. Test işleminde kullanılan cihaz ve ekipmanları

Test işlemlerinin en önemli adımlarından bir tanesi de aracın koşturulacağı farklı yolların belirlenmesidir. Bu yolların seçiminde çeşitli yol karakteristiklerini bünyelerinde barındıracak yolların seçilmesine özen gösterilmiştir. Hâlihazırda Türkiye`de ne yazık ki otomotiv sektörünün profesyonel anlamda kullanabileceği bir test pisti bulunmamaktadır. Bu nedenle aracın sapma, yuvarlanma ve yunuslama hareketlerinin baskın olarak hissedilebileceği yolların seçilmesine dikkat edilmiştir (Şekil 4.6). Seçilen yollardaki test işlemi istatistiksel hassasiyeti yakalayacak şekilde yeterince uzun tutulmuştur.



Şekil 4.6. Araçların koşturulduğu 6 farklı yol, a) engebeli yol, b) stabilize yol, c) Arnavut kaldırımli yol, d) otoyol, e) test pisti ve f) yamalı yol

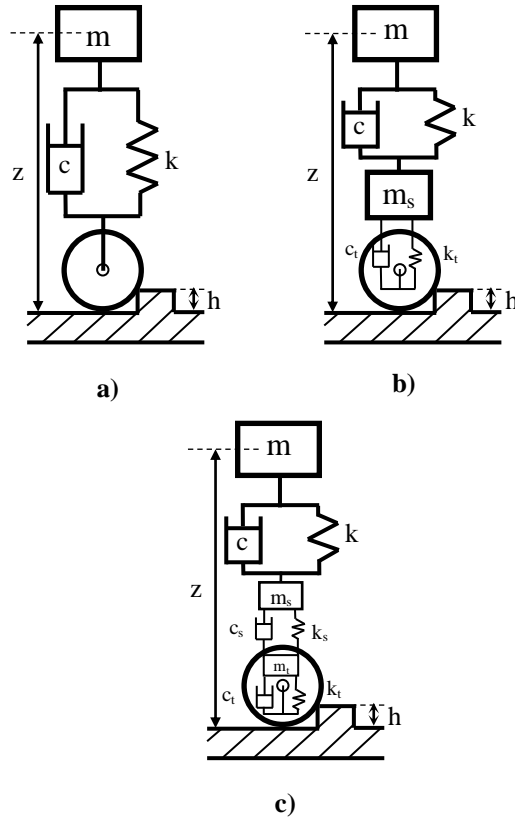
Birinci fiziksel testte birbirinden farklı yollar üzerinde tüm taşıtlar için sürüş konfor değerleri Şekil 4.8`de gösterilen benzetim modeli ile hesap edilmiştir. ISO 2631 (1997) standardına göre sürüş indeks değerlerinin sürüş konforu açısından sınıflandırılması Çizelge 4.3`te verilmiştir.

Çizelge 4.3. ISO 2631`e göre sürüş konfor değerlendirmesi (ISO 2631 1997)

Sürüş İndeks Değeri (m/s^2)	Durum (m/s^2)
$RI < 0.315$	Konforlu
$0.315 < RI < 0.63$	Biraz konforlu
$0.5 < RI < 1$	Çok az konforlu
$0.8 < RI < 1.6$	Konforsuz
$1.25 < RI < 2.5$	Çok konforsuz
$2 < RI$	Aşırı konforsuz

4.1.3. Sürüş indeksinin hesaplanması, tam-taşıt modelinin simülasyonu ve parametrelerin tespiti

Araç geliştirme sürecinde sürüş konfor çalışmaları için kullanılacak hesapsal açıdan en uygun modelin belirlenmesi için çeşitli matematiksel modeller geliştirilmiştir. Bunlardan ilki mevcut aracı 4 parçaya bölen ve sadece bir tekerlek sistemini veya süspansiyon sistemini modellemede kullanan çeyrek araç modelleridir. Çeyrek araç modelleri bir, iki veya üç serbestlik derecesine sahip olmalarına göre 3 şekilde ele alınabilmektedir (Şekil 4.7).

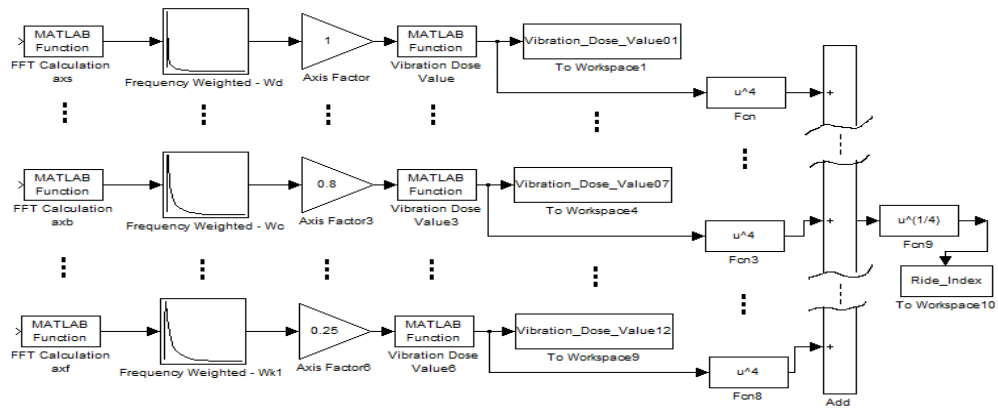


Şekil 4.7. (a) Bir, (b) iki ve (c) üç serbestlik derecesine sahip çeyrek-arac modelleri (Genta ve Morello 2009)

Şekil 4.7a`da gösterilen birinci model sadece bir serbestlik derecesine sahiptir ve tekerlek lastiği rijit olarak ele alınmıştır. Ayrıca burada kütle sadece m kütlesi olup süspansiyon sisteminin kütlesi ihmal edilmektedir. İkinci model iki serbestlik

derecesine sahip olup tekerlek lastiği kütlesiz bir yay-damper sistemi şeklinde ele alınmıştır ve süspansiyon sisteminin kütlesi burada göz ardı edilmeyerek modelde hesaba katılmıştır. Üç serbestlik derecesine sahip olan üçüncü modelde tekerlek lastiği, dinamik karakteristikleri uygun şekilde yansıtan belirli bir kütleye sahip yay-damper sistemi şeklinde modellenmiştir (m_t) ve yine burada süspansiyon sisteminin kütlesi (m_s) ihmal edilmeyerek modele eklenmiştir (Genta ve Morello 2009). Bu çalışmada süspansiyonun kütlesi ihmal edilmeyerek tek serbestlik derecesine sahip model temel alınmıştır. Ancak süspansiyon kütlesi de (m_s) dahil edilerek iki serbestlik derecesine sahip yeni bir model oluşturulup bu çalışmada uygun bir şekilde kullanılmıştır.

Sürüş indeksini hesaplayabilmek için bir Simulink (MATLAB 2009) modeli oluşturulmuştur (Şekil 4.8).



Şekil 4.8. Sürüş indeksini hesaplamak için oluşturulan Simulink modelinin temel üç bölümü (koltuk-oturma, sırt ve ayak bölgesi)

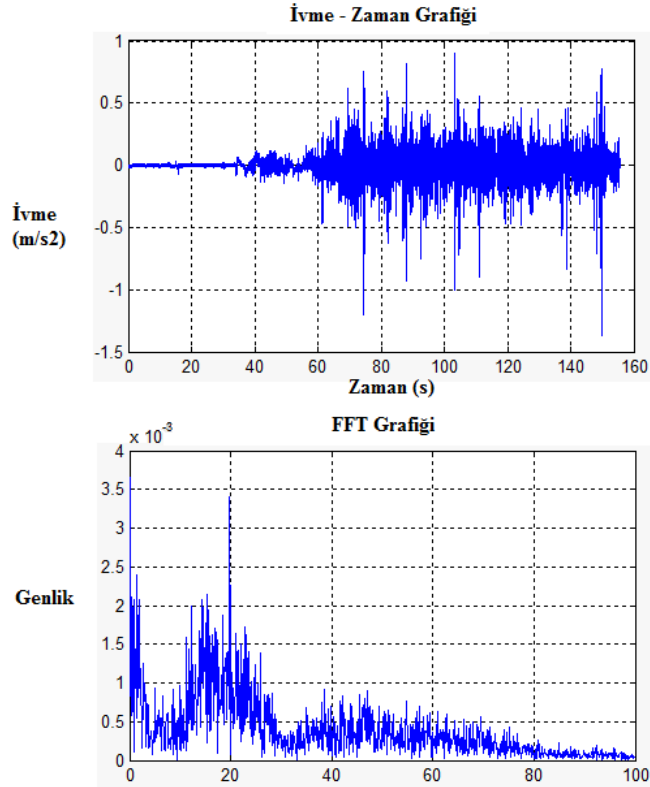
Fiziksel testlerden elde edilen ivme değerlerinin modelde kullanılabilmesi ve 0 - 100 Hz frekans aralığında bu ivme değerlerinin hızlı Fourier dönüşümü (FFT) ile frekans alanına dönüşümü için MATLAB fonksiyon bloğu yardımıyla gerekli kodları içeren bir fonksiyon yazılmıştır (Şekil 4.9). FFT analizinden sonra dönüşen ivme değerlerinin grafiği Şekil 4.10`da gösterilmiştir.


```

function axs=function_axs(~)
veriler=importdata('Hafif_Ticari_B.mat'
);
Fs = 1000;
Ts = 1/Fs;
t =
veriler.Data1_time_DEWE501A_koltuk_X;
n = length(t);
a = veriler.Data1_DEWE501A_koltuk_X;
freqRange = 0:0.1:100;
freqzData =
freqz(a,1,freqRange,Fs)/length(a);
genlik=abs(freqzData);
frekans=freqRange;
fft_matris=[frekans;genlik];
axs=double(genlik');

```

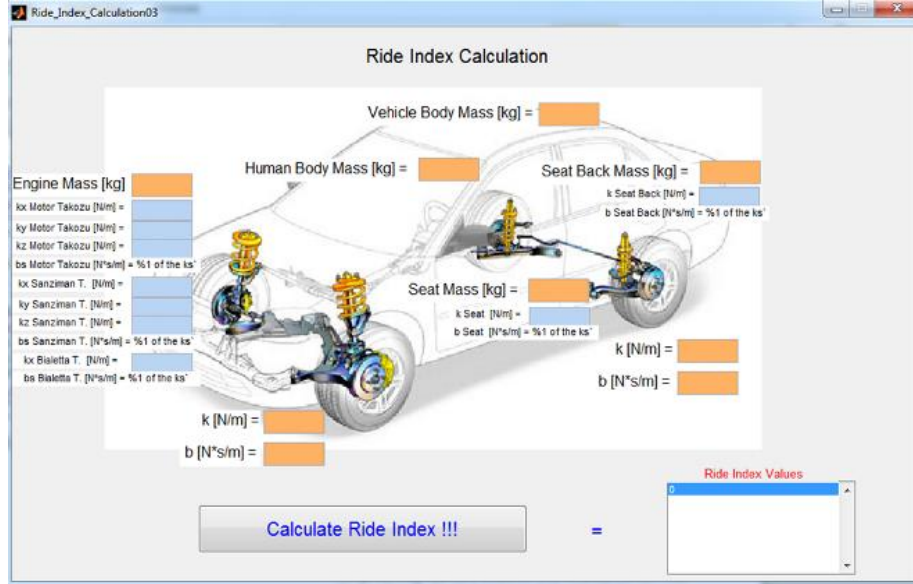
Şekil 4.9. İvme değerlerinin modele alınması ve FFT analizi için yazılan fonksiyon kodları



Şekil 4.10. FFT analizi ile ivme değerlerinin frekans alanına dönüşümü

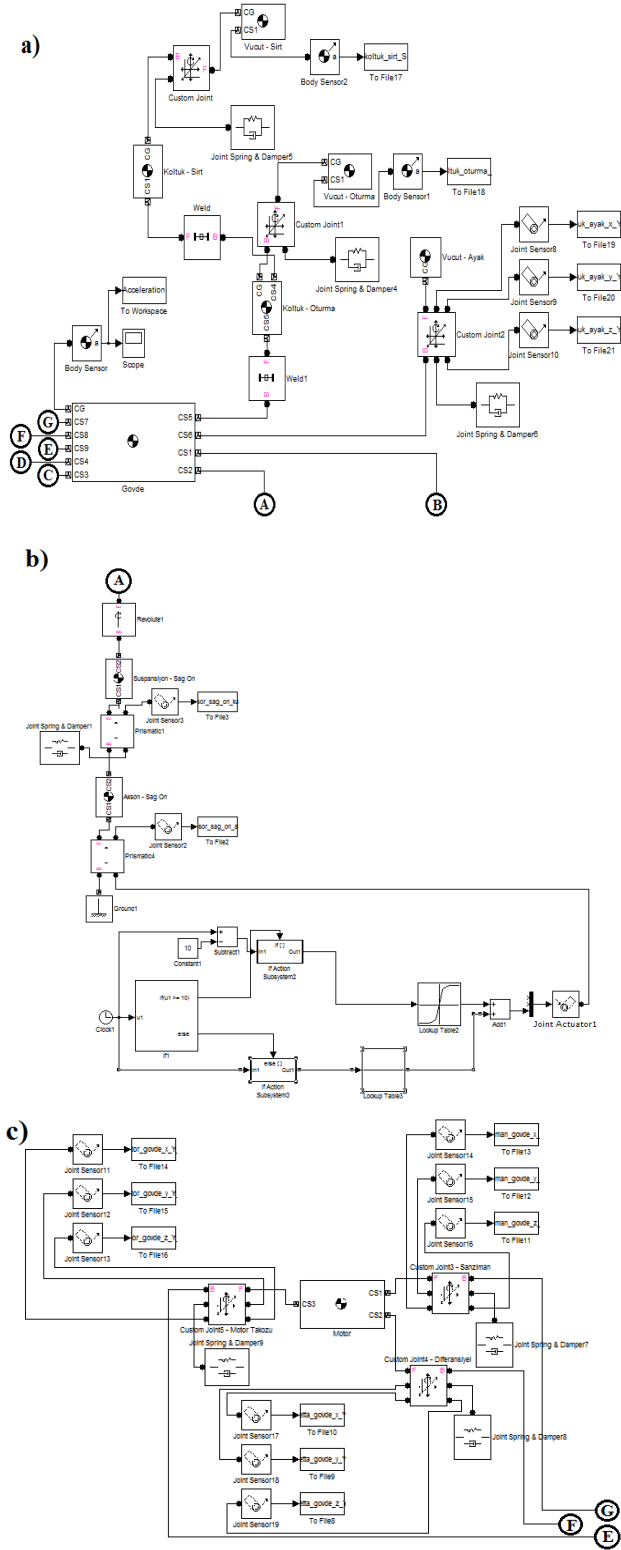
Makinelerin rijit parçalarının hareketlerinin tasarımı ve simülasyonunda faydalanılan ve blok diyagram kullanımını sağlayan, ayrıca tork ile kuvvetlerin standart Newtonsal dinamiklerinin kullanıldığı bir yazılım olan SimMechanics (MATLAB 2009) ile yukarıda bahsi geçen çeyrek taşıt modelinden faydalanılarak oluşturulan ve temelde kütle-yay-damper sistemini kullanan tam taşıt matematiksel modelinin simülasyonu için

bir model geliştirilmiştir. Lastik modeli sistem içinde tanımlanmamıştır. Fiziksel testlerde lastiğin etkisini dikkate almamak için ivmeölçerler aksona monte edilmiştir. Simülasyon-tabanlı bu modelin daha rahat kontrolü ve kullanımı için bir kullanıcı arayüzü (GUI) oluşturulmuştur (Şekil 4.11).



Şekil 4.11. Geliştirilen simülasyon-tabanlı modelin kullanıcı arayüzü (GUI)

Bu arayüzde 18 farklı parametre değeri (taşıtın, sürücünün, motorun, koltuk oturma ve sırt bölümlerinin kütleleri, ön ve arka süspansiyonların, koltuk oturma ve sırt bölgelerinin ve motorda bulunan üç takozun yay ve sönümlleme katsayıları, vb.) modelde sürüş indeksini hesaplayabilmek için girilebilmektedir. Bu arayüz hesaplamaya Şekil 4.12`de verilen simülasyon-tabanlı tam-taşıt modeli ile başlamaktadır.



Şekil 4.12. Tam-taşıt modeli için geliştirilen simülasyon-tabanlı model (a) Gövde, ayak ve koltuk bölümleri, (b) ön sağ süspansiyon sistemi (A: ön-sağ, B: ön-sol, C: arka-sağ, D: arka-sol), (c) uç takoz ile birlikte motor bölümü

Bu modelde tasarımcının süspansiyon, koltuk ve motor etkilerini daha iyi analiz edebilmesi ve tasarım sürecinin başlangıcında sadece süspansiyon sisteminin sürüş konforu üzerine etkilerini incelemek açısından lastik modeli eklenmemiştir. Simülasyon esnasında doğrusal olmayan lastik modelleri kullanılarak alınan sonuçların büyük değişiklikler göstermesi ve hassasiyetinin iyi olmaması da lastik modellerinin simülasyona eklenmeme nedenleri arasındadır. Bunun yerine her yol için belirlenen ve bu yolları karakterize eden aksondan alınan ivme değerleri kullanılmıştır. Bu sayede ilk tasarım aşamasında özellikle sadece süspansiyon parametrelerinin sürüş konforu üzerine olan etkileri incelenmek istendiğinde bu hesaplama rahatlıkla yapılabilecektir (Karen ve ark. 2011).

Modelde yol testleri ile elde edilen ve mevcut yolu karakterize eden ivme değerleri her bir aksone "Lookup Table" aracılığıyla gerekli işlemlerden geçerek "Joint Actuator" ile prizmatik mafsalları uygun bir şekilde harekete geçirmektedirler. Her bir prizmatik mafsala değerleri kullanıcı arayüzünde girilen ve kendi yay-damper özelliğine sahip kütle-yay-sistemleri eklenmiştir ve dönme hareketlerine izin veren bir küresel mafsal ile gövdeye bağlantısı sağlanmıştır. Süspansiyon sistemlerinin kütle değerleri arayüzde girilerek modelde hesaplama katılmaktadır. Böylece dört süspansiyon sisteminden gelen ivme değerleri her bir süspansiyonun kendi karakteristiğine göre değişikliğe uğrayarak ana gövdeye etki etmektedir (Şekil 4.12`de A, B, C, D).

Simülasyondaki ayak alt sistemi gövdeye her üç ekseninde (x, y ve z) doğrusal harekete izin verecek şekilde özel olarak tanımlanmış bir mafsal ile bağlanmaktadır. Koltuk alt sistemi izafi harekete izin vermeyen ve gövdeye rijit bir şekilde bağlanmaya olanak tanıyan kaynak mafsali ile bağlanmıştır. Koltuk bölümü modelde iki parça şeklinde ele alınmıştır. Birinci bölüm koltuk oturma yüzeyini ikinci bölüm ise koltuk sırt kısmını oluşturmaktadır. Her biri özellikleri kullanıcı arayüzünden girilen kendi yay-damper özelliklerine sahiptir ve her iki koltuk bölümü birbirlerine kaynaklı mafsal ile bağlanarak rijit bağlantı sağlanmıştır.

Sürücü ile koltuk arasındaki temas her üç eksenindeki doğrusal harekete izin verecek şekilde özel olarak tanımlanmış bir mafsal ile modellenmiştir. Sürücü kütlesi, biri koltuğun oturma bölgesine diğeri sırt bölgesine ve sonuncusu da ayak bölgesine etki edecek şekilde üçe ayrılarak modellenmiştir (Şekil 4.12a). Sürüş indeksi

hesaplamasında veri olarak kullanılacak ivme deęerleri modelde bu blgelerde oluřturulan mafsalara baęlanan u ayrı sensr ile toplanmaktadır.

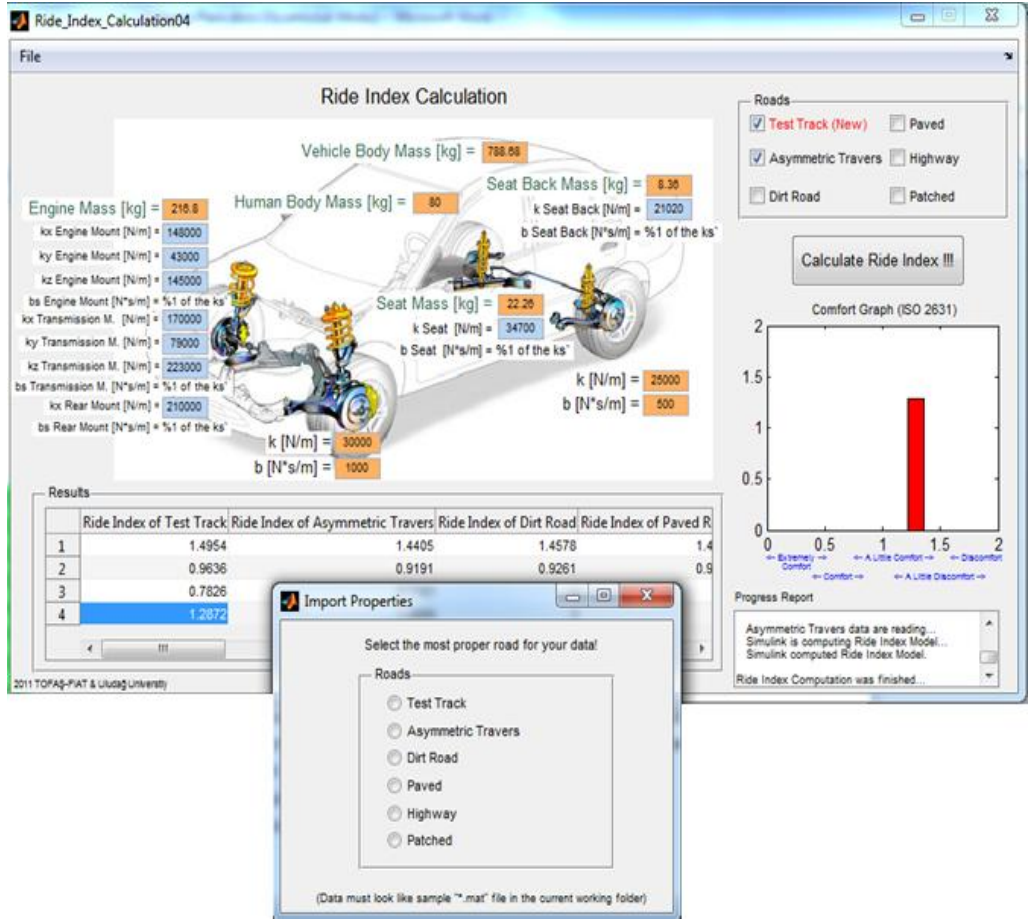
Motor sistemi gvdeye zel olarak tanımlanmıř u adet mafsal ile baęlanmıřtır; birincisi motor takozuna, ikincisi transmisyon takozuna, uncs diferansiyel takozuna ait olmaktadır (řekil 4.12c). Her u mafsalın yay-damper zellikleri ve ktlesi kullanıcı arayznden girilmektedir (řekil 4.11).

Mevcut modelin kararlı duruma gelmesi iin tařıt 10 saniye boyunca bekletilmekte yolu karakterize eden ivme deęerleri 10 saniye sonra etki ettirilmektedir. Bunun nedeni uzayda asılı duran sistem alıřtırıldıęında yerekiminin etkisi ile tařıt modelinin salınımına gemesi ve yaklaşık 7-8 saniyede kararlı duruma gelmesidir. Bunun iin mevcut zaman bir "Clock" bloęu yardımıyla tanımlanmaktadır.

Mevcut kullanıcı arayzne daha kullanıřlı olması aısından ařaęıda belirtilen noktalarda geliřtirmeler yapılmıřtır:

- Dięer test yolları iin de srř indeksi hesaplaması yapılabilmesi iin gerekli kodların ve yapının oluřturulması,
- Hesaplanan srř indeksi deęerinin bir stn grafięi zerinde gsterilerek kullanıcıya grsel aıdan fikir saęlaması,
- Daha sonradan yeni yolların eklenebilmesi iin "import" seeneęinin menye eklenmesi,
- Men seeneklerinin eklenerek optimizasyon modlne geiřin bu yolla gerekleřtirilebilmesi,
- Analiz ve simlasyon sresince ilerleyiřin kk bir bilgi kutucuęunda verilmesi.

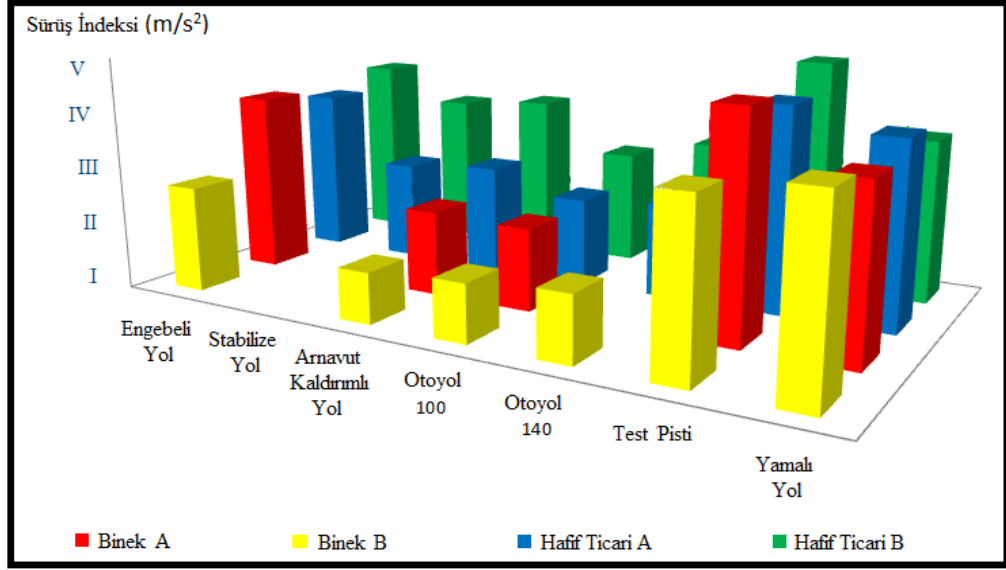
Son gncelleřtirmelerle arayzn mevcut durumu řekil 4.13`te verilmiřtir.



Şekil 4.13. Son hal ile sürüş konforu için geliştirilen arayüz

4.1.4. Fiziksel test sonuçları ile simülasyon-tabanlı model sonuçları arasındaki uyum

Bu çalışmada birisi mevcut aracın istenilen yoldaki sürüş indeksini hesaplayabilmek amacıyla diğeri tam-taşıtlı modeli oluşturabilmek için gerekli verileri elde edebilmek amacıyla iki fiziksel test gerçekleştirilmiştir. Birinci testte 4 farklı taşıtlı için 6 farklı yolda sürüş indeks değerleri hesaplanmıştır. Elde edilen tüm sürüş indeks değerleri görselleştirilmek istendiğinde Şekil 4.14'teki sütun grafiği elde edilmiştir. Burada üç sürüş indeks değeri koltuk pedindeki ivmeölçer hatasından dolayı hesap edilememiştir.



Şekil 4.14. Dört farklı taşıtın tüm yollardaki sürüş indeks değerleri

Tüm taşıtlar için en konforsuz durum beklenildiği üzere tüm olumsuz yol şartlarını bünyesinde barındıran test pistinde oluşmuştur. Yine en konforlu sürüş beklenildiği üzere 100 km/sa hızla gerçekleştirilen otoyolda gerçekleşmiştir. İki farklı tip aracın (binek - hafif ticari) sürüş indeksi karşılaştırmalarında binek araçların sürüş konforu hafif ticari olanlara nazaran daha iyi gerçekleşmiştir. Başka bir deyişle binek taşıtlar hafif ticari olanlara göre düşük sürüş indeks değerlerine sahiptirler. Bu sonuç Arnavut kaldırım ve otoyolda açıkça geçerli olduğu halde test pistinde, yamalı yolda ve engebeli yolda tam anlamıyla geçerli değildir. Sonuç olarak yüksek ivme değerleri oluşturabilecek düzeyde engebeye sahip yollar sürüş konforu açısından olumsuz etkilere neden olmaktadır. Bu durumda taşıtların tasarım aşamasında süspansiyon özelliklerinin optimizasyonu büyük önem arz etmektedir.

Aynı tipteki taşıtların sürüş indeks değerleri kıyaslandığında en iyi sürüş indeks değerleri beklenildiği üzere otoyolda elde edilmiştir. Otoyolda sürüş konforunun hız ile ters orantılı bir şekilde değiştiği gözlemlenmiştir. Başka bir deyişle hız arttıkça sürüş konforu düşmektedir.

İkinci testte geliştirilen model için gerekli temel parametre değerleri toplanmıştır. Bunun için dört araç arasından gerekli özellikleri (gövde, süspansiyon, koltuk kütleleri, atalet momentleri, süspansiyonların, takozların ve koltuğun yay-damper katsayıları vb.)

bilinen bir taşıt seçilmiştir ve 6 farklı yolda koşturularak sürüş indeks değerleri hesaplanmıştır.

Geliştirilen modelin doğruluğunu araştırmak amacıyla benzetim-tabanlı modelden elde edilen sonuçlar ile fiziksel testlerden elde edilen sonuçlar karşılaştırılmak istendiğinde dört araç arasından gerekli özellikleri (gövde, süspansiyon, koltuk kütleleri, atalet momentleri, süspansiyonların, takozların ve koltuğun yay-damper katsayıları vb.) bilinen bir taşıt seçilmiştir. Karşılaştırma sonucunda kabul edilebilir oranda değerler elde edilmiştir. Çizelge 4.4`te test pistinde özellikleri bilinen binek taşıtı B`nin benzetim sonucunda elde edilen sürüş indeksi ile fiziksel testten elde edilen sürüş indeks değeri arasındaki fark % olarak verilmiştir. Aynı şekilde süspansiyon kulelerinden elde edilen ivme değerlerinin ortalama karekök değerleri de aynı tabloda karşılaştırılmıştır. Çizelgeden görüleceği üzere kurulan modelden sapma oranı % 3-5 mertebelerindedir (Karen ve ark. 2011).

Çizelge 4.4. Binek taşıtı B için fiziksel test ile simülasyon arasındaki uyum

	Sürüş indeks Değeri (m/s ²)	Ön-Sağ Süspansiyonun RMS Değeri (m/s ²)
Sapma (%)	5	3

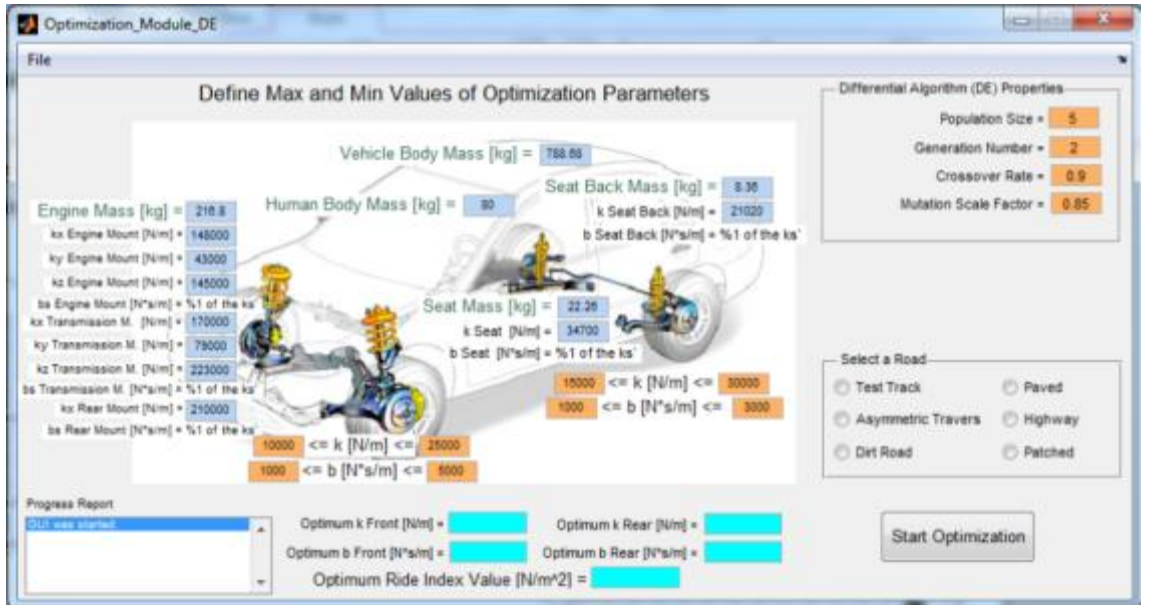
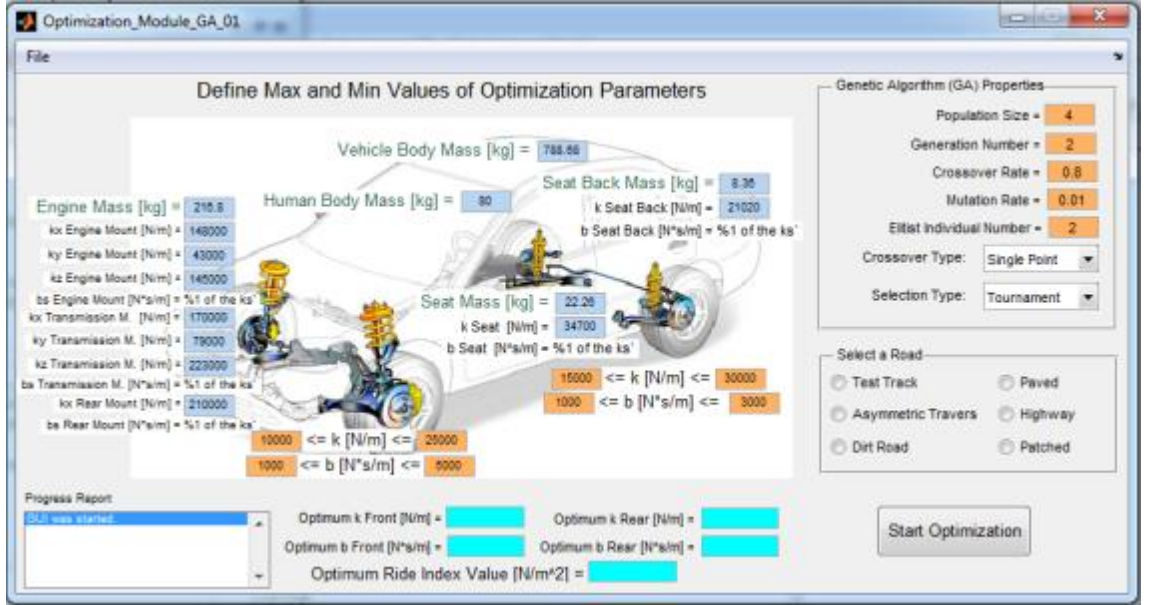
Doğrulama işlemi diğer tüm yollar için de yapıldığında Çizelge 4.5`te görüleceği üzere fiziksel testler ile simülasyon arasındaki uyum % 6-9 arasında olmaktadır.

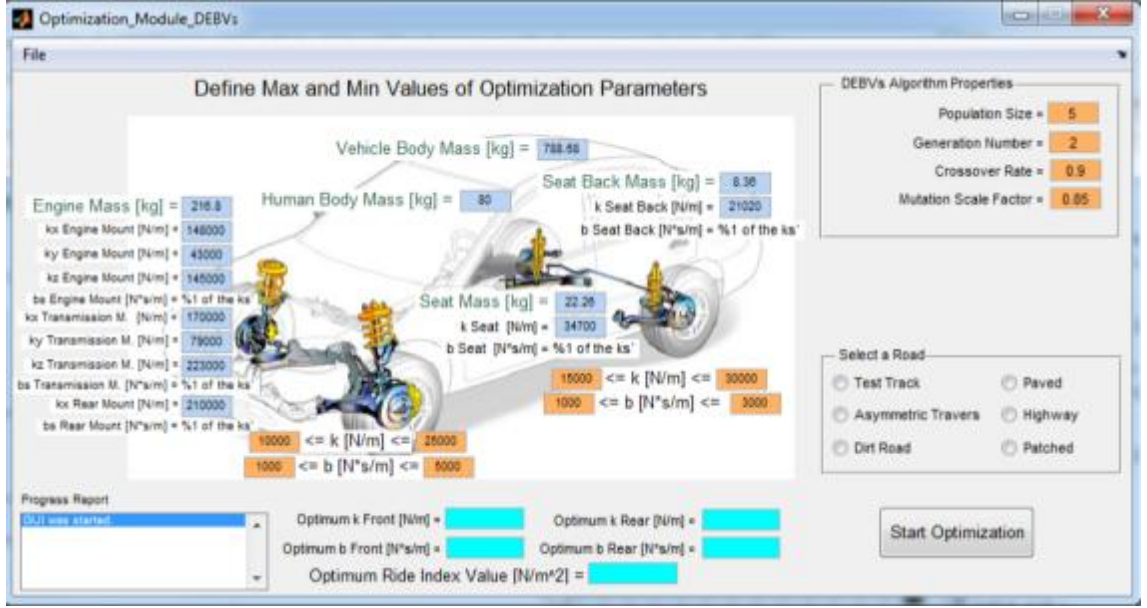
Çizelge 4.5. Fiziksel test sonuçları ile simülasyon-tabanlı model sonuçları arasındaki uyum

	Test Pisti	Engelibeli Yol	Arnavut Kaldırımli Yol	Otoyol	Yamalı Yol
Sapma (%)	9	7,8	8,1	6,9	8,2

4.1.5. Sürüş konfor optimizasyon probleminin oluşturulması, çözülmesi ve sonuçları

Mevcut arayüze genetik algoritma, DE ve geliştirilen DEBVs algoritması ile çalışacak üç optimizasyon modülü eklenmiştir (Şekil 4.15).





c)

Şekil 4.15. Optimizasyon için hazırlanan arayüz: a) GA, b) DE, c) DEBVs

Optimizasyon problemi genetik algoritma, diferansiyel gelişim (DE) ve geliştirilen DEBVs algoritması ile 5 popülasyon sayısı ve 5 nesil sayısı kullanılarak test pisti için çözülmüştür. Sürüş konfor optimizasyon probleminin yapısı gereği her bir çözüm, core i7, 4 çekirdekli 64-bit özelliklerini içeren bir çözücüyle yaklaşık 20-30 saniye sürmektedir (GA \approx 30 sn, DE \approx 25 sn, DEBVs \approx 20 sn). DEBVs algoritması ile hesaplama zamanında yaklaşık olarak %35 oranında tasarruf sağlanmıştır. Daha fazla popülasyon ve nesil sayısı sonucu fazla değiştirmedeğinden gereksiz hesaplamalardan kaçınılmıştır. Toplam optimizasyon süresi aynı özelliklerdeki bir çözücü kullanılarak DEBVs ile yaklaşık 10 dakika sürmektedir. Her üç algoritma için de hesaplama sayısı 25`tir. GA`da çaprazlama oranı 0,8 ve mutasyon oranı 0,01 olarak alınmıştır. Ayrıca 2 birey için elitizm uygulanmış ve tek-noktalı çaprazlama ve turnuva seçim yöntemi kullanılmıştır. DE ve DEBVs`de ölçeklendirme faktörü 0,85 ve çaprazlama oranı 0,9 olarak alınmıştır. En iyi sonuçlar en kısa hesaplama zamanında geliştirilen DEBVs algoritması ile elde edilmiştir. DEBVs algoritması kullanılarak hesap edilen sürüş indeksi **0,77146`**dir. Tüm algoritmalar ile bulunan optimum parametre değerleri ($k_{ön}$, $b_{ön}$, k_{arka} , b_{arka}) ve minimum sürüş indeks değerleri aşağıdaki çizelgede verilmiştir. (Çizelge 4.6).

Çizelge 4.6. Sürüş konfor optimizasyon problemi için bulunan optimum sonuçlar

	Optimum k Ön [N/m]	Optimum b Ön [N*s/m]	Optimum k Arka [N/m]	Optimum b Arka [N*s/m]	Optimum Sürüş İndeksi [m/s ²]	Hesaplama Zamanı (%)
Genetik Algoritma (GA)	11463	4882	17128	2918	0,77894	100
Diferansiyel Gelişim (DE)	12185	5000	16144	2464	0,77649	83
Geliştirilen DEBVs Algoritması	10541	4018	15535	2747	0,77146	65

4.1.6. Sürüş konfor optimizasyon çalışması ile elde edilen kazanımlar ve yapılabilecek yeni çalışmalar

Bu optimizasyon problemi çalışmasında taşıt sürüş konfor özelliklerinin değerlendirilmesi ve geliştirilmesi için taşıt tasarım sürecinde kullanılabilecek bir yaklaşım geliştirmek amacıyla yapılan çalışmalar üzerinde durulmuştur. Sürüş konfor hesaplamasında kullanılabilecek ve dört farklı taşıtın altı farklı yolda koşturulmasıyla elde edilen bir simülasyon-tabanlı model geliştirilmiştir. Bu model koltuğun oturma bölgesinden, sırt bölgesinden ve ayak bölgesinden elde ettiği ivme değerlerini kullanarak sürüş konfor değerini motor, süspansiyon ve koltuk parametrelerini de dâhil ederek hesaplamaktadır. Biri sürüş konforunu hesaplayabilmek için gerekli verileri sağlamak amacıyla diğeri tam-taşıt modeli için gerekli parametre değerlerini toplamak amacıyla iki fiziksel test gerçekleştirilmiştir. Testlerden elde edilen sonuçlar ile benzetim çalışması ile elde edilen sonuçlar arasındaki uyum umut vericidir. Geliştirilen modelin taşıt tasarım sürecinde tasarımcıya yardımcı bir araç olarak kullanılması zaman ve maliyet açısından avantajlar sağlayabilir.

Optimizasyon aşamasında konfor optimizasyon modeli kurulmuş, optimizasyon modeli için bir arayüz geliştirilmiş ve istenen konfor özelliklerinin sağlanması için uygun tasarım optimizasyon yöntemleri seçilerek uygulanmıştır. Geliştirilen DEBVs algoritması ile optimizasyon problemi çözülmüş ve sürüş konfor parametrelerinin

optimum deęerleri elde edilmiřtir. Seilen yntemlerden elde edilen sonularla geliřtirilen DEBVs algoritması ile elde edilen sonular kıyaslanmıřtır. En iyi sonular en kısa hesaplama zamanında geliřtirilen DEBVs algoritması ile elde edilmiřtir.

Günümüzde tüketicilerin araç satın alırken karar vermelerinde önemli rol oynayan ve gittike artan bir eğilim gösteren iki faktör, araç güvenlięi ve konforudur. Bu konfor optimizasyon alıřmasında geliřtirilen yöntemin kullanımı ile araç tasarım sürecinin ilk ařamalarında konfor tasarım özellikleri belirlenmiř olacaęından, tasarım süreci ve tasarım doęrulama için yapılacak mühendislik alıřmaları, testler ve prototipler aısından özellikle zaman ve maliyet konularında kazanımlar elde edilecektir. Araç tasarımından üretime kadar olan sürecin kısaltılması, yeni ürün tasarım maliyetinin azaltılması ve ayrıca mevcut modellerin konfor özelliklerin iyileřtirilmesi için gereken alıřmaların daha kısa sürede ve maliyette yapılabilecek olması bu alıřmanın en temel kazanımlarıdır.

Geliřtirilen yöntem modüler ve yeni uygulamalara açık olarak tasarlandıęı için, araçlarda konfor ile ilgili yeni donanımların sanal ortamda test edilmesi (farklı motor takozları, süspansiyon sistemleri vb.), farklı araç yapısı veya sistemlerin uygulanması durumunda konfor özellikleri üzerinde meydana gelen etkilerin incelenmesi ve yeni donanımların araç konfor özellikleri üzerindeki etkilerinin belirlenmesi kolaylıkla gerekleřtirilebilecektir.

İleride yapılacak alıřmalar ile ilgili otomotiv tasarım ve üretim firmaları için araç tasarım konfor alıřmaları prosedürü hazırlanabilir, yöntem firma veri tabanına girilerek kullanıma aılabilir, bu sayede taşıtların konfor özelliklerinin kolaylıkla deęerlendirilebilmesine imkân saęlanabilir.

Bu alıřma taşıt tasarımında sürüř konfor ve yol tutuř parametrelerini simülasyon-tabanlı bir yöntem geliřtirme yaklařımının ilk basamaęıdır. İkinci basamakta sürüř konfor optimizasyonuna yol tutuř özelliklerini de ieren yeni bir simülasyon-tabanlı yöntem geliřtirilerek eklenmesi planlanmaktadır. Böylece taşıt tasarımında sürüř konfor ve yol tutuř parametrelerini beraber deęerlendiren yeni bir yaklařım geliřtirilmiř olacaktır.

4.2. DEBVs ile Kalıp Tasarım Optimizasyon Probleminin Çözülmesi

Otomotiv üretim sektöründe kalıp tasarım süreci, ürün geliştirme performansına etki eden önemli faktörlerden birisidir. Kalıp tasarımı bu sektördeki en karmaşık ve maliyetli üretim aşamalarından birisidir. Kalıp tasarım aşamalarında yapılan hata veya süre uzaması, yeni bir ürünün maliyetini ve termin sürelerini olumsuz yönde etkilemektedir. Bu nedenle, günümüz pazar gereksinimlerine yanıt verebilecek ürünlerin üretilmesi için tasarım ve imalat işlemlerinde teknolojiye dayalı yeni yaklaşımların uygulanması gerekmektedir. Bu çalışma klasik kalıp tasarım sürecinden farklı olarak kalıp tasarım sürecine tasarım optimizasyon çalışmalarında kullanılan topoloji yaklaşımını entegre ederek, termin ve maliyet konularında kazanımlar sağlamayı hedeflemektedir. Sac metal parçaların kaliteleri çoğunlukla kalıp esneme değerlerine, geri yaylanmaya ve alıştırma işlemlerine bağlıdır. Bu değerlerin henüz kalıp üretilmeden kalıp tasarım aşamasında öngörülmesi, sac parçaların kalitesi açısından büyük önem arz etmektedir. Mevcut kalıp tasarımlarının incelenmesi ve bilgisayar destekli mühendislik analizleri ve testler ile doğrulama çalışmasının yanında sac metal kalıplılığında kullanılan kalıpların kavram aşamasında daha hafif malzemeden daha rijit olarak tasarlanması için kullanılacak bir simülasyon destekli yapısal (topoloji) optimizasyon algoritması geliştirilmiştir. Bu çalışmada klasik kalıp tasarım sürecinden farklı olarak kalıp tasarım sürecine istenen rijitlikte kalıp tasarımı için simülasyon tabanlı topoloji yaklaşımı entegre edilmiştir. Önerilen yaklaşımın doğrulamasını sağlamak için fiziksel testler yapılmış ve test çıktılarının simülasyon sonuçları ile korelasyonu incelenmiştir. Bu çalışmada, dişi kalıbın elastikiyeti ve temas tanımlamalarının yanında ayrıca pres tablasının da etkisi analizlere dâhil edilerek doğrusal olmayan sonlu elemanlar analizleri gerçekleştirilmiştir. Bu durumun, sac parçanın kalıpta nihai şeklini almasında ortaya çıkan gerçek gerilmelerin ve yer değiştirmelerin simülasyonlarla hesaplanmasında daha doğru modelleme yapılabilmesi açısından önemli katkıları olmuştur.

Çevresel zararlar da düşünüldüğünde döküm kalıplarının ağırlıklarının azaltılması oldukça önem kazanmaktadır. Sac kalıplılığında malzeme, döküm ve çelik esaslı hammaddelerden oluşmakta ve bu hammaddeler toplam maliyetin önemli bir kısmını oluşturmaktadır. Özellikle son zamanlarda dünya genelinde çelik fiyatlarındaki artış bu

oranı daha da yukarılara yükseltmiştir. Bu çalışmada sac metal kalıpcılığında kullanılan kalıpların kavram aşamasında daha hafif malzemedeki rijit olarak tasarlanması, tasarlanan kalıpların belirli rijitlikte olması için kullanılan firmalara özgü standartların gözden geçirilmesi ve iyileştirilmesi, kalıp esneme değerleri kontrol altında tutularak kalıp alıştırma işlemine katkı sağlanması ve geliştirilen DEBVs algoritması ile topoloji optimizasyonu ile belirlenmiş tasarımda kullanılan optimum feder ölçülerinin bulunması amaçlanmıştır.

Kalıp tasarım optimizasyon probleminde ilk önce kalıpların katı modelleri elde edilmiş, bu katı modellerden sonlu eleman modelleri oluşturulmuş ve gerçek sınır şartları belirlenmiştir. Bu koşullarda doğrusal olmayan sonlu eleman analizleri gerçekleştirilmiştir. Diğer yandan ise birim yer değiştirme ve gerilme değerleri kalıp üzerinde testler yapılarak elde edilmiş ve sonlu eleman analizleri ile karşılaştırılarak korelasyonu sağlanmıştır. Uygun korelasyon sağlandıktan sonra kalıp feder yapısı üzerinde topoloji optimizasyonu uygulanmış ve yeni modelin sonlu eleman analizleri gerçekleştirilmiştir. Bu şekilde amaçlanan tasarım kriterine ulaşıncaya kadar tekrar yeni modeller oluşturulmuş ve analizleri gerçekleştirilmiştir. Uygun model bulunduğundan sonra bu modelin içerdiği optimum feder boyutları geliştirilen DEBVs algoritması ile çözümlenerek bulunmuştur.

Kalıp tasarım optimizasyon çalışmasında yapılan işlemlerin akış diyagramı Şekil 4.16'da verilmiştir.

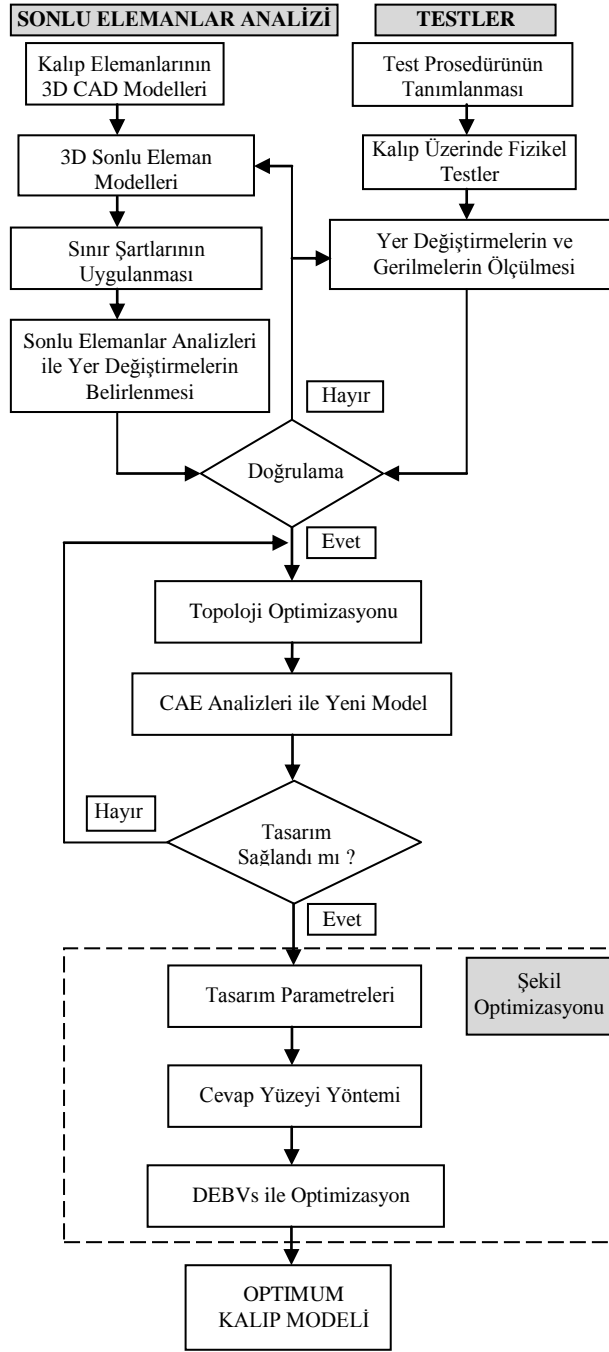
4.2.1. Kalıp tasarım süreci

Mevcut sac kalıp tasarım sürecinde girdilerin iyi analiz edilerek belirlenmesi ve kalıp tasarımcısına iletilmesi, istenen kalitede kalıp tasarımının gerçekleştirilmesi açısından çok önemlidir. 3 boyutlu sac kalıp tasarımındaki temel girdiler şu şekilde sıralanabilir:

- Araç üzerinde kullanılacak 3 boyutlu olarak hazırlanmış CAD yüzey modeli (çamurluk, tavan, motor kaputu vb.).
- 3 boyutlu CAD modele göre hazırlanmış metot çalışmaları. Metot çalışması tasarımcının temel kaynağıdır ve tasarımcıyı yönlendirmektedir. Metot çalışması daha önce elde edilmiş tecrübelerden ve günümüz teknolojisinden faydalanılarak

(gerçek üretim şartları göz önünde bulundurularak yapılan sonlu elemanlar analizleri, FEA) hazırlanmaktadır.

- Tasarlanacak kalıbın üretim yapacağı pres ve pres hattı bilgileri (en, boy, yükseklik, bağlama kanalı ölçüleri, hattın robotlu veya transfer presi olup olmaması vb.).



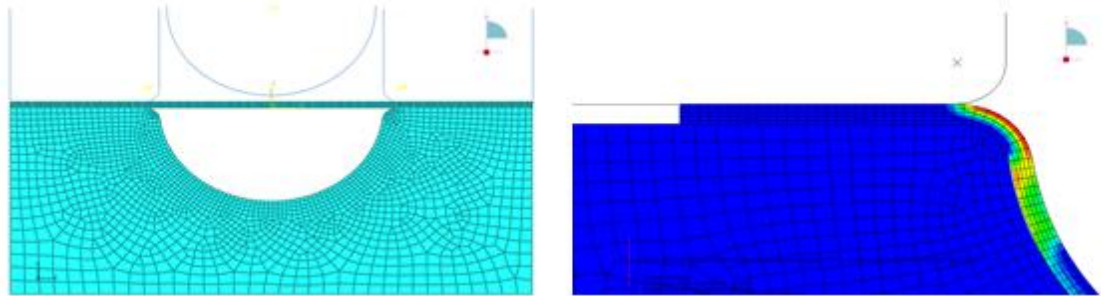
Şekil 4.16. Kalıp tasarım optimizasyon çalışmasının akış diyagramı

Kalıp tasarımcısı yukarıda açıklanan bilgiler ışığında her bir firmanın kendine özgü olarak hazırladığı kalıp tasarım şartnamelerine uyarak kalıp tasarımlarını yapmaktadır. Kalıp tasarımları 3 boyutlu CAD yazılımları kullanılarak tasarlanmaktadır. Kalıp tasarım şartnameleri, tasarımda kullanılacak feder kalınlıkları ve federler arası mesafe gibi temel bilgileri içermektedir. Tasarlanan kalıplar tasarlandığı şekli ile rijit kabul edilmekte ve herhangi bir ek analize tabi tutulmamaktadır. Ancak bu çalışmada klasik kalıp tasarım sürecinden farklı olarak kalıp tasarım sürecine istenen rijitlikte kalıp tasarımı için simülasyon tabanlı topoloji yaklaşımı entegre edilmiş, termin ve maliyet konularında kazanımlar elde etmek ve otomotiv endüstrisinin kalıp bölümlerinde geliştirilen yöntemlerin bir uygulamasını sağlamak için gerekli çalışmalar yapılmıştır.

4.2.2. Doğrusal olmayan simülasyonlar

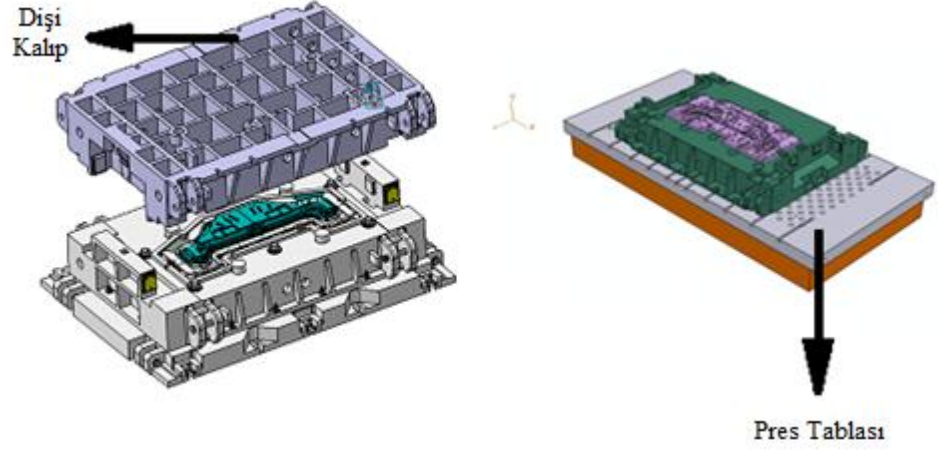
Simülasyon çalışmalarının yapılacağı ve halihazırda kullanılan uygun bir kalıp modeli belirlenmiş ve bu kalıbın katı modeli CAD ortamında simülasyon çalışmalarına hazır hale getirilmiştir. CAD modelinin analiz yazılımına transferi gerçekleştirilmiş ve ağ yapısı oluşturulmuştur. En uygun eleman sayısının belirlenmesi amacıyla eleman yakınsama çalışması yapılmış ve optimum eleman büyüklüğü 20 mm olarak belirlenmiştir.

Başlangıç analiz çalışmalarında iki boyutlu modeller oluşturulup, derin çekme işlemi esnasında sac parçanın kalıp yüzeylerine temas durumları incelenerek kalıplarda oluşan deformasyonlar sonucu temas oluşmayan bölgeler belirlenmiştir (Şekil 4.17).



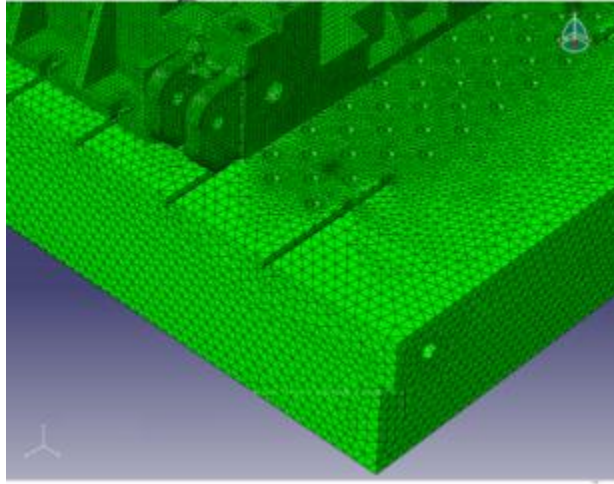
Şekil 4.17. Temas analizleri gerçekleştirilen iki boyutlu (2D) kalıp modelleri

Üç boyutlu analizlere geçmeden önce pres yapısı ve elemanları incelenerek derin çekme işlemi sürecinde presin çalışma şekli ve kalıp yük durumları araştırılmıştır. Pres tablası ve matris konumlamaları göz önünde tutularak temas sınır şartları dikkate alınmıştır (Şekil 4.18).



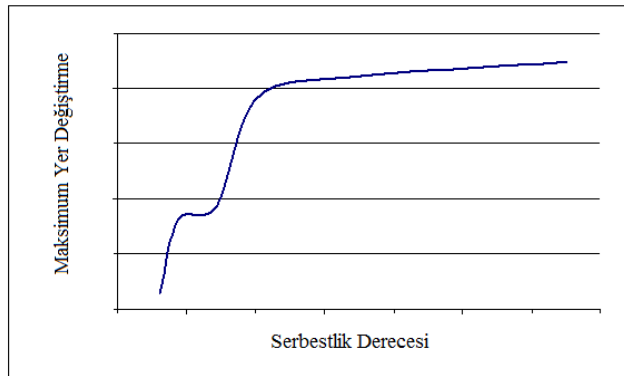
Şekil 4.18. Analizleri gerçekleştirilen üç boyutlu (3D) kalıp modelleri

Sonlu eleman analizinin kullanıldığı simülasyon çalışmalarında, modelin simülasyon için hazırlanması, elemanlara bölünmesi ve sınır şartlarının verilmesi önem arz etmektedir. Sonlu eleman analizinde simülasyon sürecinin yakınsaması, hataların en aza indirgenmesi için ön işleme aşamasında farklı eleman ve bölgesel elemanlara ayırma çalışmalarının çeşitli denemeler yapılarak incelenmesi gerekmektedir. Bu nedenle farklı eleman boyutları için elde edilen sonuçlara bakılarak eleman boyutunun ne oranda etkili olduğu belirlenmiştir (Şekil 4.19). Böylece hatalı veya çok uzun zaman alabilecek simülasyonlardan kaçınılmıştır.



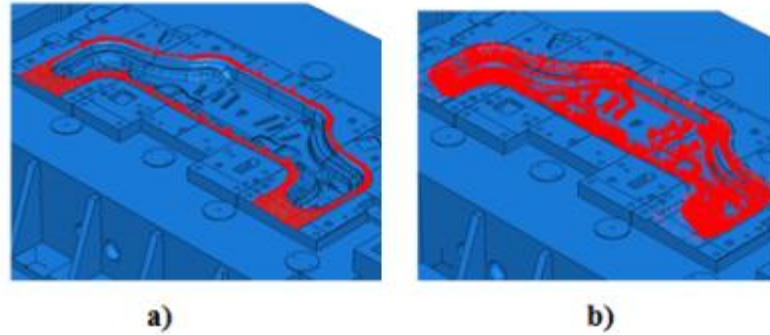
Şekil 4.19. Optimum eleman boyutundaki kalıp modeli (Toplam eleman sayısı: 3 300 870, toplam düğüm noktası sayısı: 684 917, toplam serbestlik derecesi: 2 054 751)

Kalıbın simülasyon çalışmalarına hazırlanması, model tanımlamadan sonra gelen aşamadır. Ön işleme süreci, analiz sonuçları üzerinde önemli etkisi olan eleman oluşturma işlemlerini içermektedir. Eleman serbestlik derecesinin, eleman tipinin ve boyutunun belirlenmesi bu sürecin en önemli iki parametresidir. Modelde mevcut hataların sebep olacağı olumsuzluklar ve eleman bozunumlarının önlenmesi çalışmaları bu aşamada yapılmıştır. Modelde karmaşık yüzeylerin bulunmasından dolayı eleman tipi olarak 4 düğüm noktasına sahip “tetrahedral” (dörtüzlü) eleman seçilmiştir. Optimum eleman sayısının belirlenmesi amacıyla “eleman yakınsaması” çalışması yapılmıştır. Global eleman boyutu olarak 20 mm alınması durumunda optimum eleman boyutuna ulaşılmış, daha küçük boyutlarda sonucun çok fazla değişmediği gözlenmiştir (Şekil 4.20).



Şekil 4.20. Eleman yakınsama çalışması

Simülasyon çalışmaları öncesi önemli bir adım da sınır şartlarının ve yüklerin tanımlanmasıdır. Sonuçlar üzerinde son derece etkisi olan tanımlamaları içeren bu aşama iki alt aşama olarak ele alınmıştır: 1. serbestlik dereceleri kısıtlamaları ve 2. yüklerin sisteme uygulanması. Sistemin gerçek ortam işlevsel tanımlamalarının yapılmaması durumunda simülasyon sonuçlarının hata oranı artmaktadır. Ayrıca sınır şartları ve yük durumlarının sistemin gerçek ortam işlevselliğini tam yansıtamaması durumunun kötü sonuçlar doğuracağından dolayı sistem tanımı üzerinde gerekli çalışmalar yapılarak gerçek ortam işlevselliği belirlenerek sınır şartları kalıp elemanlarına uygulanmıştır. Sınır şartları olarak pres tablasının alt yüzeyleri sabitlenmiş, temas yüzeylerine yayılı yük iki aşamada uygulanmıştır. Birinci aşamada pot çemberi kuvveti (Şekil 4.21a), ikinci aşamada erkek kalıbın uyguladığı kuvvet etki ettirilmiştir (Şekil 4.21b).



Şekil 4.21. Dişi kalıp yüzeylerine etki eden kuvvetler; a) Pot çemberi, b) Erkek kalıp

Çözümleyici olarak ABAQUS yazılımı (Simulia 2008) kullanılmıştır. Çözümleme süresi kullanılan simülasyon tanımlamaları, kullanılan çözümleyici ve donanım özelliklerine göre farklılıklar gösterebilmektedir. Özellikle bu çalışmada uygulanan önışleme süreçte önemli rol oynamaktadır.

3 boyutlu simülasyon çalışmalarında yer değiştirme haritasının çıkarılması hedeflenmiştir. Ayrıca pot çemberinin etkisi de dikkate alınmıştır. Dişi kalıbın ve alt tablanın üç boyutlu modelleri üzerinde özellikle durulmuştur. Sonlu elemanlar analizleri ile kalıp parçaları arasında temas tanımlamaları yapılmış, doğrusal olmayan modeller çözülerek kalıp yer değiştirme değerleri hesaplanarak maksimum yer değiştirme bölgesi bulunmuştur. Çözümleme süresi kullanılan simülasyon tanımlamaları, kullanılan

çözümleyici ve donanım özelliklerine göre farklılıklar gösterebilmektedir. Özellikle bu çalışmada uygulanan ön-işleme ve çözümleyici özellikleri süreçte önemli rol oynamıştır.

4.2.3. Fiziksel testler

Literatürdeki mevcut test çalışmaları incelenmiş ve mevcut yöntemlerden uzama-ölçerler ve/veya ivmeölçerler ile veri toplanmasının uygun olacağına karar verilmiştir. Yer değiştirme değerlerinin ölçümü için kullanılan test yöntemi belirlendikten sonra gerekli donanım tedarik edilmiş ve 16 kanallı bir veri toplama ünitesi ile ölçümler gerçekleştirilmiştir. Test işleminde ilk önce hangi bölgelerden veri toplanacağına karar verilmiştir. Bunun için simülasyonu gerçekleştirilen kalıbın maksimum gerilme ve maksimum yer değiştirme bölgeleri belirlenmiş ve kalıbın ilgili yerlerine ivmeölçerler ve uzama-ölçerler monte edilmiştir (Şekil 4.22).



Şekil 4.22. İvmeölçerlerin kalıp üzerine montaj işlemi ve konumları

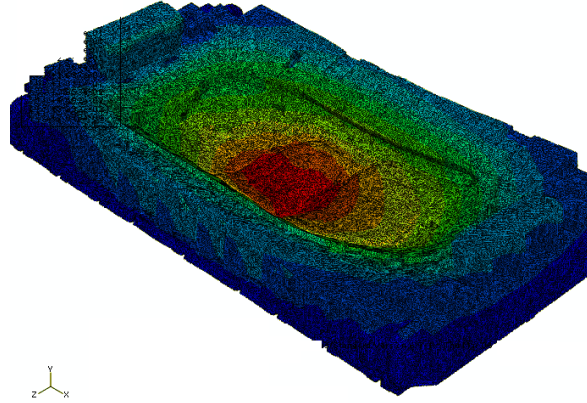
Kablolama donanımının testler sırasında problem oluşturmaması için gerekli önlemler alınmış ve belirlenen noktalardan test ölçümleri yapılmıştır. Belirli bir yakınsama değerine ulaşına kadar test işlemlerine devam edilmiştir ve bu veriler daha sonra yer değiştirme ve gerilme değerlerine dönüştürülmüştür (Şekil 4.23).



Şekil 4.23. Test ekipmanı

4.2.4. Simülasyonların fiziksel testler ile doğrulanması

Sonlu elemanlar analiz yöntemi kullanılarak kalıp yer değiştirme değerleri hesaplanmıştır. Yer değiştirme ve gerilme dağılımlarının simülasyondaki kalıp modeli üzerinde gösterimi Şekil 4.24`te verilmiştir.



Şekil 4.24. Kalıp üzerindeki maksimum yer değiştirme ve gerilme değerleri (Simulia 2008)

Simülasyon çalışmalarının doğrulanması için testler yapılmış, dişi kalıp (matris) üzerinde oluşan yer değiştirme değerleri ile testlerden elde edilen yer değiştirme değerleri karşılaştırılmıştır. Oluşturulan işlem adımları ve uygulanan tekniklerin kalıp tasarım sürecinde kullanımının uygunluğunun doğrulanması için test sonuçları ile simülasyon sonuçları karşılaştırılarak doğruluk oranları hesaplanmıştır. Fiziksel testler yapılmış, test çıktılarının simülasyon sonuçları ile korelasyonu incelenmiş ve yöntemin uygulama güvenilirliği değerlendirilmiştir. Elde edilen korelasyon değerleri yaklaşımın

uygulanabileceğini göstermiştir. Simülasyon sonucunda hesaplanan maksimum yer değiştirme değerine karşılık test sonucunda elde edilen değer arasında %1`lik bir sapma mevcuttur.

Bu aşamada topoloji modeli temel alınarak imalat durumu ve maliyet faktörleri de düşünülerek farklı kalıp yapıları oluşturulmuştur. Alternatif tasarımlar incelendiğinde elde edilen kazanımın kütle değişimi olarak oldukça önemli olduğu görülmüştür. Maksimum yer değiştirme değeri baz alınarak istenen kütle değişim oranını veren model uygulamaya alınabilir. A3 veya A4 modellerinden birisi tercih edilebilir (Çizelge 4.7). Bu aşamada üretim maliyetleri ve özellikle müsaade edilebilir maksimum yer değiştirme değerleri model seçimi açısından en önemli parametrelerdir. Kalıp model seçimi bu parametrelerin önem derecesine göre alternatif yapılar arasından belirlenebilir (Çizelge 4.7).

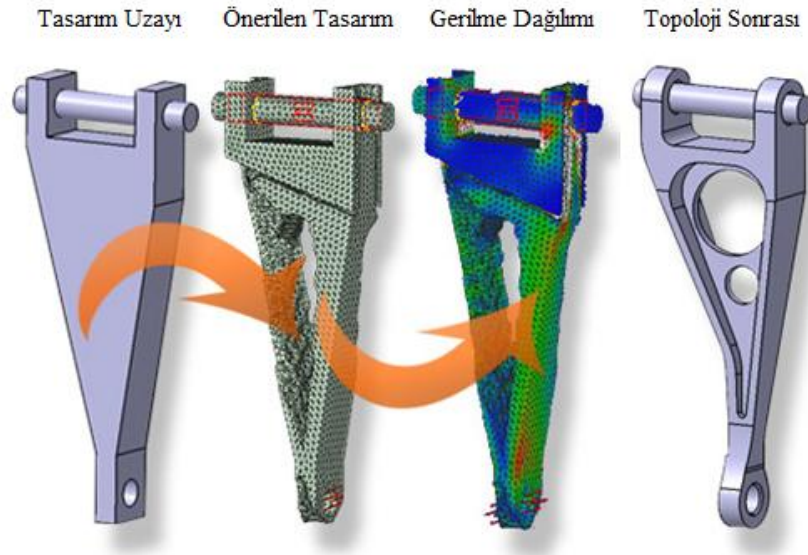
Çizelge 4.7. Önerilen alternatif yapılar ve maksimum yer değiştirme değerleri

Modeller	Sapma (%)
Mevcut model	100
A1	87
A2	84
A3	83
A4	86

Elde edilen korelasyon değerleri yaklaşımın uygulanabileceğini göstermiştir, uygulanan yöntemin doğrulaması açısından oldukça iyi değerlerdir. Dört alternatif yapının denendiği çalışmada üçüncü alternatif (A3) veya dördüncü alternatif (A4) tercih edilebilir.

4.2.5. Kalıp tasarım optimizasyon probleminin oluşturulması, çözülmesi ve sonuçları

Topoloji optimizasyonu istenen özelliklerde ürün tasarımında tasarım sürecinin ilk aşaması olan konsept model belirlemede kullanılan bir yöntemdir. Topoloji tasarımı ürün için tanımlanan tasarım uzayında tasarımcıya ürün modelinin optimum yapısını oluşturmada yardımcı olmaktadır. Malzeme dağılımını temel alan bu yöntemde tasarımcı, tasarım uzayını ve kısıtlarını belirlemektedir. Belirlenen hedef parametreleri (minimum hacim ve maksimum rijitlik) doğrultusunda seçilen araç bileşenini basacak kalıbın başlangıç modelini oluşturmak üzere topoloji optimizasyonu çalışmaları yapılmıştır. Topoloji optimizasyonu ile tasarım işlemlerinin şematik gösterimi örnek bir parça için Şekil 4.25`te verilmiştir.



Şekil 4.25. Topoloji optimizasyonunun ilerleme aşamaları (Topologica 2011)

Topoloji optimizasyonunun temel mantığı; dış sınırları belirlenen başlangıç tasarımında, verilen hacim azalmasını sağlayacak en rijit veya doğal frekansı en yüksek olan yapının oluşturulmasıdır. Şekil 4.25`te verilen örnekte, dış sınırları belirlenen başlangıç modelinde belli bir hacim azalmasını sağlayacak şekilde en rijit ve en az hacme sahip malzeme dağılımı topoloji optimizasyonu ile elde edilmiştir.

Topoloji optimizasyonu için gerekli veriler şunlardır (Karen ve ark. 2009):

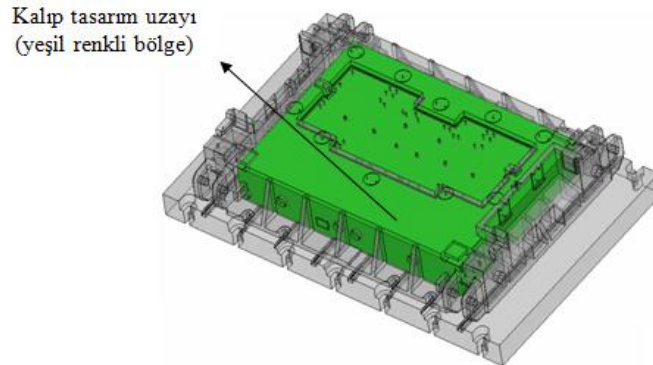
- Tasarım dış sınırları,
- Optimize edilecek ve edilmeyecek bölgeler,
- % olarak boşaltılacak hacim değeri,
- Yer değiştirme kısıtları.

Topoloji optimizasyonu sonucunda elde edilen bilgiler ise şu şekildedir:

- Boşaltılacak ve kalan kısımlar,
- Gerilme, yer değiştirme veya doğal frekans değerleri.

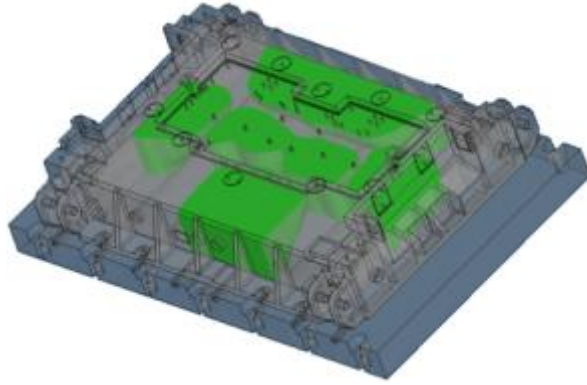
Topoloji tasarımı ile ürünün optimum yapısal modeli belirlenmektedir. Klasik yöntemlerden farkı tasarımcı ürün modelini optimum yapıda tanımlayabilmektedir. Böylece optimum bir konsept ile çalışmaya başlamaktadır. Aksi takdirde klasik yaklaşımda olduğu gibi optimum olmayan bir yapı ile ürün tasarlanabilmekte ve sonradan yapılan optimizasyon çalışmaları yeterli anlamda optimizasyon açısından etkin olmayabilmektedir. Bu çalışmada topoloji tasarım sürecinde izlenen işlem adımları şu şekildedir (Karen ve ark. 2009):

ADIM 1: Tasarım uzayının belirlenmesi: Bu adımda işlevsellik göz önünde tutularak sistemde model için tasarım çalışma uzayı ve kısıtlanan tasarım uzayı tanımları yapılarak kalıbın iç bölgesinde oluşan bir tasarım uzayı belirlenmiştir (Şekil 4.26).



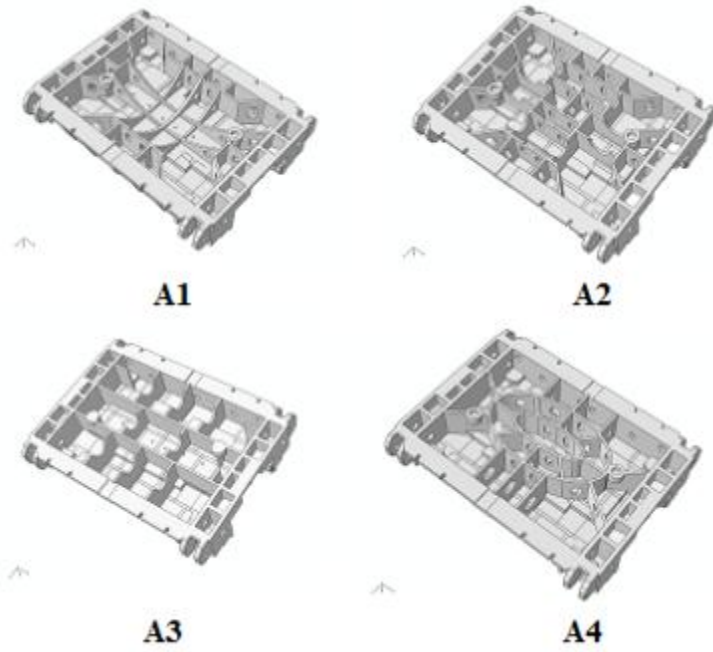
Şekil 4.26. Tasarım uzayı

ADIM 2: Topoloji malzeme dağılım simülasyon çalışması: Bu adımda bir topoloji çözümleyicisi (Altair 2008) kullanılarak bir önceki adımda tanımlanan tasarım uzayı için malzeme dağılımı elde edilmiştir. Tasarım uzayını eleman yoğunluk parametresine göre sonlu elemanlara ayırarak çözümleyen topoloji dağılımında parametrelerin farklı değerlerine göre farklı yapısal çözümler elde edebilmek mümkündür. Topoloji optimizasyon çözümlemesinde amaç, istenen rijitlikte ve hacimde malzeme dağılımını görebilmektir. Matematiksel tasarım optimizasyon modeli için amaç fonksiyonu kriteri olarak komplians ve kısıt olarak hacim azalması seçilmiştir. Optimizasyon sonucunda orta ve yan kısımlarda malzeme yığılması olduğu, köşe ve yan kısımlarda malzeme boşaltılmasının yük ve kalıp prosesi açısından uygun olabileceği görülmektedir (Şekil 4.27).



Şekil 4.27. Topoloji optimizasyonu sonrasında ortaya çıkan malzeme dağılımı (Altair 2008)

ADIM 3: Kalıp yapısının tekrar modellenmesi ve şekil optimizasyonu: Belirlenen topoloji temel alınarak imalat durumu ve maliyet faktörleri de düşünülerek alternatif kalıp modelleri oluşturulmuştur (Şekil 4.28).



Şekil 4.28. Topoloji sonrası oluşturulan alternatif tasarımlar

Alternatif tasarımlar incelendiğinde elde edilen kazanımın kütle değişimi olarak %13-17 arasında olduğu görülmektedir. Maksimum yer değiştirme değeri baz alınarak istenen kütle değişim oranını veren model uygulamaya alınmış ve A3 veya A4 modellerinden birisi tercih edilmeye çalışılmıştır. Bu aşamada üretim maliyetleri ve özellikle müsaade edilebilir maksimum yer değiştirme değerleri model seçimi açısından en önemli parametrelerdir. A3 veya A4 modellerinin kütle değişimlerinin oranı %83 ve %86 olduğu için bu modeller tercih edilebilirler (Çizelge 4.8).

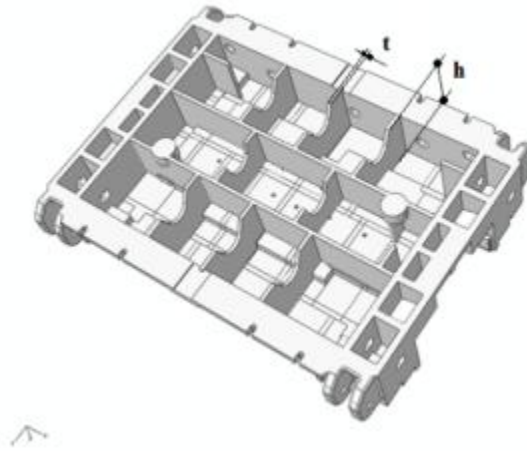
Çizelge 4.8. Mevcut modelde ve alternatif modellerdeki kütle değişim oranı ve maksimum yer değiştirme oranı

Modeller	Kütle Değişim Oranı (%)	Maksimum Yer Değiştirme Oranı (%)
Mevcut Model	100	100
A1	87	134
A2	84	250
A3	83	140
A4	86	125

Üretim maliyetleri ve özellikle izin verilen maksimum yer değiştirme değerleri hangi modelin üretim için uygun olacağını karar vermede en önemli faktörlerdendir. Seçilen modellerin maksimum yer değiştirme değerleri mevcut kalıp modeline kıyasla müsaade edilen oranlarda olduğundan şekil optimizasyonu ile bu oran daha da düşürülebilir.

Bu çalışmada yukarıda bahsedilen nedenlerle diğer dört alternatif arasından üçüncü alternatif (A3) şekil optimizasyonu için tercih edilmiştir (Şekil 4.28). Bu model için kütle yaklaşık olarak %14 oranında düşürülmüştür.

Seçilen kalıp tasarımının şekil optimizasyonu sürecinde A3 modeli şekil parametrelerinin tanımlanması için ele alınmıştır. Bunun için mevcut A3 modelinin optimizasyon formülasyonunda ilk adım tasarımdan istenen özelliği açıkça belirten bir amaç fonksiyonunu tanımlamaktır. Bu çalışmada kullanılan şekil optimizasyonunun amaç fonksiyonu kütlenin minimizasyonudur. Seçilen modelin tasarım değişkenleri; feder kalınlığı (t) ve feder yüksekliği (h)`dir (Şekil 4.29).



Şekil 4.29. Şekil optimizasyonu için seçilen A3 modelinin tasarım değişkenleri

Şekil optimizasyon sürecinde sadece bir analiz yaklaşık olarak bir saat zaman almaktadır, böylece tüm optimizasyon sürecinde tüketilen zaman çok fazla olmaktadır. Bu nedenle optimizasyon süresince tüketilen zamanı azaltmak için deney tasarımı (DOE) uygulanmıştır (Karen ve ark. 2006). Deney tasarımı yöntemi belirli parametreler ile optimizasyon sürecinin hazırlanmasında kullanılmıştır. Ele alınan kalıp modelinde tasarım değişkenleri $10 < t < 100$ mm ve $100 < h < 500$ mm aralığında değişmektedir.

Tasarım uzayını en iyi şekilde temsil edebilmek açısından 50 deneyin yapılmasına karar verilmiştir. Bu şekilde hacim, gerilme ve maksimum yer değiştirme için sırasıyla Çizelge 4.9, Çizelge 4.10 ve Çizelge 4.11`de verilen üç adet Varyans Analiz tablosu (ANOVA) oluşturulmuştur.

Çizelge 4.9. Hacmin deney tasarımı ile oluşturulmuş ANOVA tablosu

Parametreler (mm)	Hacim (mm ³)				
	t=10	t=20	t=30	t=40	t=50
h=100	429 215 360	439 562 880	448 325 376	4 555 85 568	462 585 600
h=200	533 297 792	553 945 792	571 423 744	5 859 44 128	599 944 192
h=300	634 329 472	663 918 976	690 533 632	7 121 94 368	733 077 888
h=400	802 070 400	840 095 872	877 838 592	9 064 84 480	934 100 672
h=500	975 434 752	1 022 036 224	1070 786 880	1 106 441 088	1 140 814 208
	t=60	t=70	t=80	t=90	t=100
h=100	469 550 720	476 405 824	483 150 944	489 786 048	496 311 168
h=200	613 874 432	627 584 640	641 074 880	654 359 232	667 439 168
h=300	753 861 120	774 320 448	794 458 752	814 298 752	833 829 888
h=400	961 592 768	988 665 984	1 015 330 944	1 041 631 424	1 067 526 720
h=500	1 175 039 360	1 208 750 720	1 241 966 464	1 274 751 616	1 307 035 392

Çizelge 4.10. Gerilmenin deney tasarımı ile oluşturulmuş ANOVA tablosu

Parametreler (mm)	Gerilme (N/mm ²)									
	t=10	t=20	t=30	t=40	t=50	t=60	t=70	t=80	t=90	t=100
h=100	100	70	52	49	40	29	25	24	22	20
h=200	92	61	42	40	28	25	23	21	18	16
h=300	80	52	35	32	27	24	21	18	14	12
h=400	70	43	25	24	23	21	18	15	12	10
h=500	62	37	21	20	20	19	17	13	11	8

Çizelge 4.11. Maksimum yer deęiřtirmenin deney tasarımı ile oluşturulmuş ANOVA tablosu

Maximum Yer Deęiřtirme (mm)										
Parametreler (mm)	t=10	t=20	t=30	t=40	t=50	t=60	t=70	t=80	t=90	t=100
h=100	0,62	0,53	0,49	0,47	0,46	0,44	0,43	0,42	0,41	0,40
h=200	0,64	0,52	0,46	0,44	0,42	0,40	0,38	0,37	0,36	0,35
h=300	0,68	0,52	0,45	0,42	0,40	0,37	0,36	0,34	0,33	0,32
h=400	0,72	0,52	0,45	0,41	0,38	0,36	0,34	0,32	0,31	0,30
h=500	0,74	0,51	0,44	0,39	0,36	0,34	0,32	0,31	0,29	0,28

ADIM 4: Cevap yüzeyi yöntemi ile Polinom uydurma: Optimizasyon süresince deneyi yapılmayan parametrelerin ara deęerleri gerekli olabilmekte ve bu ara deęerler ile elde edilecek hacim, gerilme ve maksimum yer deęiřtirme deęerlerine ihtiyaç duyulmaktadır. Bunun için çeřitli derecelerde fonksiyon veya polinom uydurmak gerekmektedir (Kaya ve ark. 2010).

h ve t deęerlerini içeren $f(h,t)$ fonksiyonları tüm dört cevap fonksiyonu yardımıyla hacim, gerilme ve maksimum yer deęiřtirme için yapılan üç deney tasarımı ile en küçük kareler yöntemi ile hesap edilmiştir. Bu fonksiyonlar ařaęıdaki gibidir:

Hacim için birinci-dereceden (planar) cevap fonksiyonu:

$$f_1(h,t) = 148083438,72 + (1713160,43h + 2167638,90t) \quad (4.4)$$

Gerilme için birinci-dereceden (planar) cevap fonksiyonu:

$$f_2(h,t) = 80,46333 + (-0,05110h - 0,602061t) \quad (4.5)$$

Maksimum yer deęiřtirme için birinci-dereceden (planar) cevap fonksiyonu:

$$f_3(h,t) = 0,645410 + (-0,00016h - 0,00312t) \quad (4.6)$$

Hacim için ikinci-dereceden (kuadratik) cevap fonksiyonu:

$$f_4(h,t) = 358212791,89 + (437649,11h + 570052,15t) + (1466,57h^2 - 5091,45t^2 + 7192,15ht) \quad (4.7)$$

Gerilme için ikinci-dereceden (kuadratik) cevap fonksiyonu:

$$f_5(h,t) = 121,29 + (-0,12521h - 2,03287t) + (0,00004h^2 + 0,01063t^2 + 0,00087ht) \quad (4.8)$$

Maksimum yer deęiřtirme için ikinci-dereceden (kuadratik) cevap fonksiyonu:

$$f_6(h,t) = 0,70733 + (-0,00009h - 0,00769t) + (0,0000003h^2 + 0,00005t^2 - 0,000005ht) \quad (4.9)$$

Hacim için üçüncü-dereceden (kübik) cevap fonksiyonu:

$$f_7(h,t) = 348158460,59 + (593017,39h + 691485,25t) + (721,29h^2 - 15207,71t^2 + 9181,55ht) + (0,83065h^3 + 93,78t^3 - 0,04173h^2t - 17,85777ht^2) \quad (4.10)$$

Gerilme için üçüncü-dereceden (kübik) cevap fonksiyonu:

$$f_8(h,t) = 150,56000 + (-0,14201h - 4,37337t) + (-0,000003h^2 + 0,05611t^2 + 0,00229ht) + (0,00000006h^3 - 0,00025t^3 - 0,0000001h^2t - 0,00001ht^2) \quad (4.11)$$

Maksimum yer deęiřtirme için üçüncü-dereceden (kübik) cevap fonksiyonu:

$$f_9(h,t) = 0,74222 + (-0,00007h - 0,01264t) + 0,000002h^2 + 0,00022t^2 - 0,00002ht) + (-0,00000002h^3 - 0,000001t^3 + 0,00000006h^2t + 0,0000001ht^2) \quad (4.12)$$

Hacim için dördüncü-dereceden (kuartik) cevap fonksiyonu:

$$f_{10}(h,t) = 110611572,48 + (5090491,53h + 679040,56t) + (-26802,61h^2 - 24155,66t^2 + 11231,04ht) + (67,49h^3 + 324,58t^3 - 0,10552h^2t - 63,79475ht^2) + (-0,05560h^4 - 1,47364t^4 + 0,00118h^3t + 0,31135ht^3 - 0,00906h^2t^2) \quad (4.13)$$

Gerilme için dördüncü-dereceden (kuartik) cevap fonksiyonu:

$$f_{11}(h,t) = 163,72 + (-0,19361h - 5,92460t) + (0,00064h^2 + 0,12783t^2 - 0,00160ht) - 0,000002h^3 - 0,00136t^3 + 0,000004h^2t + 0,00005ht^2 + (0,000000002h^4 + 0,000005t^4 - 0,000000004h^3t - 0,0000003ht^3 - 0,00000001h^2t^2) \quad (4.14)$$

Maksimum yer değiştirme için dördüncü-dereceden (kuartik) cevap fonksiyonu:

$$f_{12}(h,t) = 0,77062 + (0,00013h - 0,01888t) + (0,000002h^2 + 0,00055t^2 - 0,00004ht) + (-0,000000002h^3 - 0,000007t^3 + 0,000000002h^2t + 0,0000006ht^2) - (0,000000000004h^4 + 0,00000003t^4 + 0,00000000001h^3t - 0,000000003ht^3 - 0,0000000007h^2t^2) \quad (4.15)$$

Tüm 12 cevap fonksiyonunun gerçek değerlere olan uygunluğunu ölçmek için bu fonksiyonlar R-kare testine tabi tutulduğunda Çizelge 4.12'deki R-kare değerleri elde edilmiştir. Buna göre dört yaklaşımın (planar, kuadratik, kübik ve kuartik) arasından dördüncü-dereceden kuartik cevap fonksiyonu deney tasarımındaki verilere en iyi şekilde uyum göstermiş ve 0,99'dan büyük R-kare değerine sahip olmuştur. Bu nedenle optimizasyon işleminde sırasıyla Denklem 4.13, Denklem 4.14 ve Denklem 4.15'te verilen üç kuartik dördüncü-dereceden cevap fonksiyonu; hacmi, gerilmeyi ve maksimum yer değiştirmeyi hesaplamak amacıyla kullanılmıştır.

Çizelge 4.12. Kalıp tasarım optimizasyon problemi için uydurulan cevap fonksiyonlarının R-kare değerleri

R-Kare (R^2) Değerleri				
	Birinci-Dereceden Cevap Fonksiyonu (Planar)	İkinci-Dereceden Cevap Fonksiyonu (Kuadratik)	Üçüncü-Dereceden Cevap Fonksiyonu (Kübik)	Dördüncü-Dereceden Cevap Fonksiyonu (Kuartik)
Hacim	0,976135	0,999057	0,999183	0,999991
Gerilme	0,773411	0,933547	0,980725	0,993063
Maksimum Yer Değiştirme	0,744616	0,922409	0,978365	0,995027

ADIM 5: Diferansiyel gelişim ve geliştirilen DEBVs algoritması ile optimizasyon işlemi:
Optimizasyon işleminde amaç hacmi minimize etmektir, bir başka deyişle Denklem 4.13'te formüle edilen hacmi minimize eden optimum $h(x_1)$ ve $t(x_2)$ değerlerini bulmaktır.

Amaç Fonksiyonu (Minimizasyon):

$$f(x) = 110611572,48 + (5090491,53x_1 + 679040,56x_2) + (-26802,61x_1^2 - 24155,66x_2^2 + 11231,04x_1x_2) + (67,49x_1^3 + 324,58x_2^3 - 0,10552x_1^2x_2 - 63,79475x_1x_2^2) + (-0,05560x_1^4 - 1,47364x_2^4 + 0,00118x_1^3x_2 + 0,31135x_1x_2^3 - 0,00906x_1^2x_2^2) \quad (4.16)$$

Birisi gerilmeyi (Denklem 4.14) diğeri maksimum yer değiştirmeyi (Denklem 4.15) ifade edecek şekilde aşağıdaki eşitsizlikleri sağlaması beklenen iki adet kısıt mevcuttur.

Kısıtlar:

$$g_1(x) = 163,72 + (-0,19361x_1 - 5,92460x_2) + (0,00064x_1^2 + 0,12783x_2^2 - 0,00160x_1x_2) - 0,000002x_1^3 - 0,00136x_2^3 + 0,000004x_1^2x_2 + 0,00005x_1x_2^2) + (0,000000002x_1^4 + 0,000005x_2^4 - 0,000000004x_1^3x_2 - 0,0000003x_1x_2^3 - 0,00000001x_1^2x_2^2) - 50 \leq 0 \quad (4.17)$$

$$g_2(x) = 0,77062 + (0,00013x_1 - 0,01888x_2) + (0,000002x_1^2 + 0,00055x_2^2 - 0,00004x_1x_2) + (-0,000000002x_1^3 - 0,000007x_2^3 + 0,000000002x_1^2x_2 + 0,0000006x_1x_2^2) - (0,0000000000004x_1^4 + 0,00000003x_2^4 + 0,00000000001x_1^3x_2 - 0,000000003x_1x_2^3 - 0,00000000007x_1^2x_2^2) - 0,33 \leq 0$$

Sınır Kısıtları:

$$10 \leq x_1 \leq 100 \quad (4.18)$$

$$100 \leq x_2 \leq 500$$

Optimizasyon problemi diferansiyel gelişim (DE) ve geliştirilen DEBVs algoritması ile 10 popülasyon sayısı ve 25 nesil sayısı kullanılarak çözülmüştür. Her iki algoritma için

de hesaplama sayısı 250`dir. Güvenilirlik açısından her algoritma ile kalıp tasarım optimizasyon problemi 30 defa çözdürülmüştür. Her iki algoritmada ölçeklendirme faktörü 0,85 ve çaprazlama oranı 0,9 olarak alınmıştır. En iyi sonuçlar geliştirilen DEBVs algoritması ile elde edilmiştir. DEBVs algoritması kullanılarak hesap edilen h ve t`nin optimum değerleri $h = 259,6546$ mm, $t = 93,7187$ mm`dir. Bu sonuçlar ile kütle yaklaşık olarak % 24 oranında, gerilme ise %72 oranında düşürülmüştür. Optimizasyon işleminden önce %149 olan maksimum yer değiştirme değeri optimizasyon işleminden sonra %109`a düşürülmüştür. Ayrıca en önemli noktalardan birisi olarak vurgulanmak gerekirse toplam hesaplama zamanı DEBVs ile çözüm yapıldığında yaklaşık olarak %35 oranında düşürülmüştür (Çizelge 4.13).

Çizelge 4.13. Optimizasyon öncesi ve sonrasında kalıpta oluşan % değişim

	Kütle Değişim Oranı (%)	Maksimum Gerilme Değişim Oranı (%)	Maksimum Yer Değişirme Değişim Oranı (%)	Hesaplama Zamanı %
Mevcut Model	100	100	100	---
Topoloji Optimizasyonu	83	96	149	---
Şekil Optimizasyonu (DE kullanarak)	76,43	28	109	100
Şekil Optimizasyonu (DEBVs kullanarak)	76,39	28	109	65

4.2.6. Kalıp tasarım optimizasyon çalışması ile elde edilen kazanımlar ve yapılabilecek yeni çalışmalar

Bu çalışma ile kalıp tasarım sürecinde yeni tekniklerin kullanılarak daha hafif ve istenen rijitlikte kalıpların tasarlanması için kalıp tasarımcılarına yardımcı olacak bir yaklaşım uygulanmıştır. Bu yaklaşımın doğrulamasını sağlamak için fiziksel testler yapılmış, test çıktılarının simülasyon sonuçları ile korelasyonu incelenmiş ve yöntemin uygulama güvenilirliği değerlendirilmiştir. Elde edilen korelasyon değerleri yaklaşımın uygulanabileceğini göstermiştir.

Bu çalışma, tasarımın ilk aşamalarında kalıpların imal edilmeden özelliklerinin belirlenmesine ve istenen özelliklerde tasarlanabilmesine yardımcı olmaktadır. Kalıpların esneme değerleri belirli değerde tutularak alıştırma işlemi azaltılmış ve böylece tasarımdan imalata kadar olan sürecin kısaltılmasına katkıda bulunulmuştur.

Kalıp simülasyon çalışmalarında mevcut çalışmada sadece pres tablası kullanılmış, tüm pres modeli kullanılmamıştır. İleride yapılacak çalışmalarda daha gerçekçi sonuçlar elde etmek için tüm pres yapısı kalıp ile birlikte ele alınarak bütünleşik modeller oluşturulabilir. Ayrıca model analizleri için uygulanan yüklerin pres üzerinden ölçümüne yönelik teknikler kullanılabilir. Böylece daha hızlı ve güvenilir ölçümler gerçekleştirilebilir. Kalıp temas yüzeylerindeki deformasyonların ölçümüne yönelik yeni tekniklerin araştırılarak derin çekme kalıpları için yapılan bu çalışmada kullanılan yöntem diğer kalıp operasyonlarına (kesme, bükme, ütüleme, kamlı bükme-kesme vb.) uyarlanabilir.

Geri yaylanma etkisi incelenerek kalıp deplasman ve geri yaylanma etkisinin birlikte ele alınması ile kalıp tasarım optimizasyonuna yardımcı olacak yeni bir yaklaşım tanımlanabilir. Böylece kalıpların esneme değerlerini belirli değerde tutarak alıştırma işçiliği azaltılabilir.

Sonlu eleman analizlerinde eleman boyutlarının sonuçlar üzerindeki etkisinin belirlenmesi ile kalıp elemanları simülasyon çalışmalarında gereksiz uzun zaman gerektirecek simülasyonlar önlenir. Eleman boyutu, serbestlik derecesi ve tiplerinin analiz sonuçları üzerindeki etkilerinin incelenmesi ile kalıp elemanlarının simülasyon çalışmaları için en uygun eleman özellikleri belirlenerek yakınsamama ve zaman alıcı simülasyonlar önlenir.

Kalıp tasarım işlemleri için cevap yüzeyi yöntemi, diferansiyel gelişim algoritması ve geliştirilen DEBVs algoritması ile topoloji tabanlı yapısal optimizasyon kullanan bir bütünleşik tasarım yaklaşımı kazandırılmıştır. Kalıp tasarım sürecinin erken aşamalarında kalıp özellikleri (gerilme, maksimum yer değiştirme, kütle, vb.) ile ilgili gerekli bilgiler geliştirilen DEBVs algoritması ile çok daha kısa hesaplama sürelerinde kolaylıkla edinilebilecektir.

4.3. DEBVs ile Motor Titreşim Takozu Tasarım Optimizasyon Probleminin Çözülmesi

Motor titreşim takozu optimizasyon probleminde, istenen statik ve dinamik özelliklere sahip motor askı takozlarının tasarımı gerçekleştirilmiştir. Motor titreşim takozlarında kullanılan kauçuk malzemelerin hiperelastik ve viskoelastik malzeme modelleri, yapılan malzeme testleri aracılığıyla belirlenmiş, simülasyonlar sonucunda elde edilen sonuçlar ile gerçek ürün üzerinde yapılan testler arasındaki korelasyon sağlandıktan sonra optimizasyon ile istenen tasarım hedeflerini sağlayacak takoz modeli geliştirilmiştir. Yeni ürünlerin tasarımında prototip üretim sayısının en aza indirgenmesi ve daha kısa zamanda seri üretime geçiş için kullanılacak simülasyon temelinde testlerle bütünleşik tasarım süreci tanımlamaları yapılmıştır. Bu çalışma literatürde yer alan çalışmalara göre tüm tasarım süreci ve optimizasyonu ele alan, fiziksel testler ile simülasyonları süreç doğrulaması için kullanan, statik ve dinamik etkenleri tasarımın ilk aşamalarında modelde uygulamayı amaç edinen öncü çalışmalardan birisidir (Karen ve ark. 2008b).

Modern motor takozu sistemleri, sürücüyü ve yolcuları motordan kaynaklanan ses ve titreşimlerden izole etmede etkin bir şekilde kullanılmaktadır. Ancak hala motor takozu sistemlerinin geliştirilmesine iki nedenden dolayı ihtiyaç vardır:

1. Binek taşıtlar için ses ve titreşim seviyelerinin düşürülmesi gereksinimleri,
2. Modern taşıt tasarımlarının daha hafif gövdeye ve daha güçlü motora sahip olma yönelimi. Bu şekilde azalan ağırlığın ve artan gücün titreşim davranışı üzerine gürültü ve titreşim seviyelerinin yükselmesi yönünde ters bir etkiye neden olmaktadır.

Bu iki durum sıklıkla birbiriyle çelişen durumlar ortaya çıkarmaktadır. Örneğin ses ve titreşim seviyelerinin düşürülmesi daha ağır ve daha güçsüz araçların tasarlanmasını gerektirmektedir. Motor takozu sistemlerinin performansındaki son gelişmeler benzer çelişkili durumları çözümlemede önemli bir rol oynamaktadır.

Bir motor takozu sistemi statik durumlar için motoru taşınabilir, yoldan gelen şok uyarımlar karşısında motorun sıçramasına veya zıplamasına engel olmalı, ayrıca tüm hız oranlarında ortaya çıkan motor titreşimlerini sönmülemelidir. Bu nedenle takozun statik

ve dinamik davranışlarının tümü tasarım sürecinde eş zamanlı olarak dikkate alınmalıdır. Bir otomobil motoru için iki tür dinamik kuvvet söz konusudur:

1. Yakıtın silindir içinde ateşlenmesinden doğan tepki kuvveti,
2. Dönen ve ötelenen elemanların (piston, krank, biyel, vb.) atalet kuvvetleri.

Bu kuvvetler motoru, 6 serbestlik derecesi yönünde titreşime zorlamaktadır (Yu ve ark. 2011). Düşük devirlerde motor titreşimleri nedeniyle araç içindekiler rahatsız edici bir titreşim hissederler. Yüksek hızlarda ise motor titreşimleri, aracın yolcu kabininin akustik doğal frekansı ile çakışması sonucu rahatsız edici bir düzeye erişir. Bu titreşimlerin sönümlenmesi ve düşük geçirgenlik için motor takozlarının düşük rijitlikte olması beklenir. Diğer taraftan motor titreşim takozlarının elastik rijitlikleri düşük olursa ani uyarılarda (ani frenleme, ivmelenme, vb.) problem ortaya çıkabilir. Ayrıca yüksek yer değiştirmeler sonucu motor elemanları hasar görebilir. Bu açıdan takozların motor sarsıntısının ve rezonansının sönümlenmesi ve statik yer değiştirmelerin düşük olması için yüksek rijitlik ve sönümlenme özelliğine sahip olması gerekir. Ayrıca titreşim sönümleyiciler, yer çekiminden ve atalet kuvvetlerinden kaynaklanan yükleri karşılamak için belirli değerlerdeki statik rijitliği sağlamak zorundadırlar. Diğer taraftan bu sönümleyiciler, motordan kaynaklanan titreşimin gövde gibi temel yapılara geçişini azaltmak için belirli bir dinamik rijitliği sağlamak durumundadırlar. Tüm bu nedenlerle, klasik motor askı takozlarının istenen özellikleri karşılayabilecek şekilde tasarlanması gerekmektedir. Uygun tasarlanmış motor takozlarının kullanımı ile daha iyi titreşim izolasyonu, araç hareketlerinde yumuşak geçişler ve gürültünün azaltılması mümkün olabilmektedir. Özetle, motor takozu sistemlerinden;

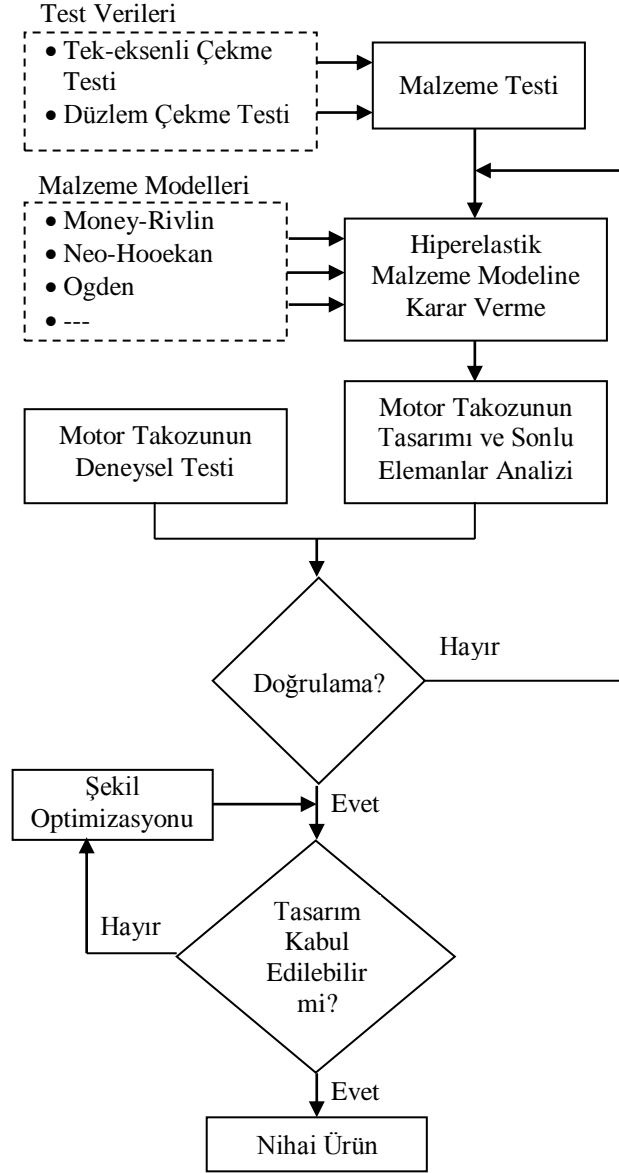
- Motoru statik olarak taşıma,
- Ani uyarılarda motoru dengede tutma,
- Motor titreşimlerini sönümlenme,

gibi fonksiyonları yerine getirmesi beklenmektedir. Ancak elastomerik takozların bu özellikleri sağlaması tüm değişken özelliklerinin ele alındığı tasarım sürecine bağlıdır. Bu konuda yapılan çalışmaların dinamik rijitlik ve sönümlenme oranının hem frekansa hem de titreşim genliğine bağlı olarak değiştirilebilme özelliği üzerinde yoğunlaştığı görülmektedir. Bu amaçla geliştirilen aktif motor titreşim takozları, düşük frekanslarda

yüksek rijitlikte ve yüksek frekanslarda ise düşük rijitlikte davranabildiklerinden dolayı titreşimi daha kolay sönümleyebilmektedirler. Fakat diğer taraftan basitlik, fiyat avantajı ve düşük bakım maliyetleri gibi nedenler elastomerik motor takozlarının kullanımını çekici kılmaktadır. Aktif motor takoz sistemlerinde kullanılan sensörler, kontrol üniteleri ve enerji ihtiyacı nedeniyle ağırlık ve maliyet artmakta ayrıca sistemin güvenilirliği azalmaktadır. Ayrıca taşıt ağırlıklarının azaltılması ve motor gücünün artması eğilimi nedeniyle artan gürültü ve titreşim problemlerine çözüm olarak hiperelastik modellere olan ihtiyaç gün geçtikçe artmaktadır.

Hiperelastik malzemeler, yüksek miktarda şekil değişimi gösterebilen ve tekrar eski şekline dönebilen malzemelerdir (örneğin kauçuk). Bu malzemelerin yük altında uzama davranışı doğrusal olmadığı gibi hemen hemen sıkıştırılmaz yapıdadır (Poisson oranı $\approx 0,5$). Otomotiv, havacılık, biyomedikal, taşıma ve spor malzemelerinde kullanım alanı mevcuttur.

Motor titreşim takozlarında kullanılan kauçuk malzemelerin hiperelastik ve viskoelastik malzeme modelleri, yapılan malzeme testleri aracılığıyla belirlenmiş, simülasyonlar sonucunda elde edilen sonuçlar ile gerçek ürün üzerinde yapılan testler arasındaki korelasyon sağlandıktan sonra optimizasyon ile istenen tasarım hedeflerini sağlayacak takoz modeli geliştirilmiştir. Böylece yeni ürünlerin tasarımında prototip ihtiyacı ortadan kaldırılarak daha kısa zamanda seri üretime geçilmesi hedeflenmektedir. En son aşamada yapılacak prototip testleri sonrası prototipler araç üzerine monte edilerek motor üzerindeki davranışları test edilecektir. Motor titreşim takozu tasarım optimizasyon çalışmasının iş akış şeması Şekil 4.30`da verilmiştir.



Şekil 4.30. Motor titreşim takozu tasarım optimizasyon çalışmasının iş akış şeması

4.3.1. Malzeme modelleri

Hiperelastik malzemelerin davranışı için birçok malzeme modeli geliştirilmiştir. Bunlardan bazıları şunlardır:

- Mooney-Rivlin,
- Polinom Formu,
- Neo-Hookean,
- Ogden Potansiyel,

- Yeoh,
- Rivlin,
- Arruda-Boyce

Hiperelastik malzemelerde yer deęiřtirme iři, gerinme enerjisi olarak depolanır. Malzemenin davranıřı gerinme enerjisi yoęunluk fonksiyonu ile tanımlanır. Gerilme bileřenleri bu fonksiyonun gerinmeye gore turevi alınarak hesaplanır. Malzeme modelleri malzeme testleri ile elde edilen veriler aracılıęıyla tanımlanır. Tek eksenli ekme, basma ve ok eksenli ekme (veya kayma) testleri ile malzeme modelleri tanımlanabilir. Uydurulan eęri ile malzeme modeli tanımındaki katsayılar belirlenir ve lineer olmayan sonlu elemanlar analizleri ile yuklemeler altında malzeme davranıřı hesaplanır. rneęin Mooney-Rivlin doęrusal olmayan hiperelastik malzeme modeli ařaęıdaki forma sahiptir (Simulia 2008):

$$U = C_{10} (I_1 - 3) + C_{01} (I_2 - 3) \quad (4.19)$$

Neo-Hookean doęrusal olmayan hiperelastik malzeme modeli ařaęıdaki forma sahiptir:

$$U = C_{10} (I_1 - 3) \quad (4.20)$$

Polinom formunun doęrusal olmayan hiperelastik malzeme modeli ařaęıdaki forma sahiptir:

$$U = \sum_{i+j=1}^N C_{ij} (I_1 - 3)^i (I_2 - 3)^j + \sum_{i=1}^N \frac{1}{D_i} (J_{el} - 1)^{2i} \quad (4.21)$$

Yeoh doęrusal olmayan hiperelastik malzeme modeli ařaęıdaki forma sahiptir:

$$U = C_{10} (I_1 - 3) + C_{20} (I_1 - 3)^2 + C_{30} (I_1 - 3)^3 \quad (4.22)$$

9 terimli Rivlin doęrusal olmayan hiperelastik malzeme modeli ařaęıdaki forma sahiptir:

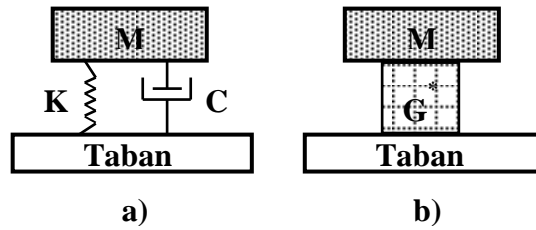
$$U = \sum_{i=0, j=0}^3 (I_1 - 3)^i (I_2 - 3)^j \quad (4.23)$$

Arruda-Boyce doğrusal olmayan hiperelastik malzeme modeli aşağıdaki forma sahiptir:

$$U = nkT \left[\frac{1}{2}(I_1 - 3) + \frac{1}{20N}(I_1^2 - 9) + \frac{11}{1050N^2}(I_1^3 - 27) \right. \\ \left. + \frac{19}{7000N^3}(I_1^4 - 81) + \frac{519}{673750N^4}(I_1^5 - 243) \right] \quad (4.24)$$

Bu modellerde U , gerinme enerjisi yoğunluk fonksiyonunu ve I_i indirgenmiş gerinme sabitlerini ifade etmektedir (Sharma 2003). J_{el} elastik hacim oranını, C_{ij} sıcaklığın bir fonksiyonu olabilen malzeme parametrelerini ifade etmektedir (Simulia 2008).

Bu çalışmada kullanılan viskoelastik (yapışkan) malzeme modeli sönümleyici ve yay içeren tek serbestlik derecesine sahip bir sistem olarak düşünülmüştür (Şekil 4.31). Şekil 4.31a'da gösterilen mekanik sistemdeki yay ve sönümleyicinin fonksiyonunu aynı özellikler gösterebilen basit bir elastomerik parça üstlenebilmektedir.



Şekil 4.31. a) Tek serbestlik dereceli titreşim modeli ve b) eşdeğer elastomerik davranış modeli

Bu şekilde çalışan sistemlerin titreşim sönümlenme problemlerini çözmede temel iki yaklaşım kullanılmaktadır. Birincisi sönümlenme sistemi için titreşimin doğal frekansını tanımlarken ikincisi frekansın bir fonksiyonu olarak sistemin geçirgenliğini tanımlamaktadır (Gent 2000):

$$f_n = \frac{1}{2\pi} \sqrt{\frac{K'g}{W}} \quad (4.25)$$

Yerçekimi ivmesi = 9800 mm/s² alındığında denklem şu şekle dönüşmektedir:

$$f_n = 15,76 \sqrt{\frac{K'}{W}} \quad (4.26)$$

Burada, f_n sistemin doğal frekansını (Hz), K' dinamik yay rijitliğini (N/mm) ve W sistemin ağırlığını (N) ifade etmektedir.

İkinci yaklaşım olarak geçirgenlik şu şekilde tanımlanmaktadır:

$$T_{ABS} = \sqrt{\frac{1 + (\eta r^2)}{(1 - r^2)^2 + (\eta r^2)}} \quad (4.27)$$

$r > \sqrt{2}$ ve $\eta \approx 0$ durumu göz önüne alınırsa geçirgenlik,

$$T_{ABS} = \frac{1}{r^2 - 1} \quad (4.28)$$

formuna dönüşmektedir.

Burada, T_{ABS} titreşim frekansındaki (f) geçirgenliği, r frekans oranını (f/f_n), f titreşim frekansını (Hz) ve η boyutsuz kayıp faktörünü ifade etmektedir. Elastomerlerin viskoz özelliği kinetik enerjiyi ısı enerjisine dönüştürerek düşük frekanslı titreşimleri (ani darbeleri) sönmölemektedir. Diğer taraftan elastik özelliği ise parçayı tekrar orijinal haline getirmeye çalışmaktadır. Bu sayede yüksek frekanslı titreşimleri sönmöleyebilmektedirler.

Elastomerik kesme modülü (kompleks modül) G^* reel ve imajiner kısımlara sahip olan kompleks bir sayıdır. Kompleks modül olarak da adlandırılan elastomerik kesme modülü aşağıdaki gibi ifade edilmektedir:

$$G^* = G' + iG'' \quad (4.29)$$

veya

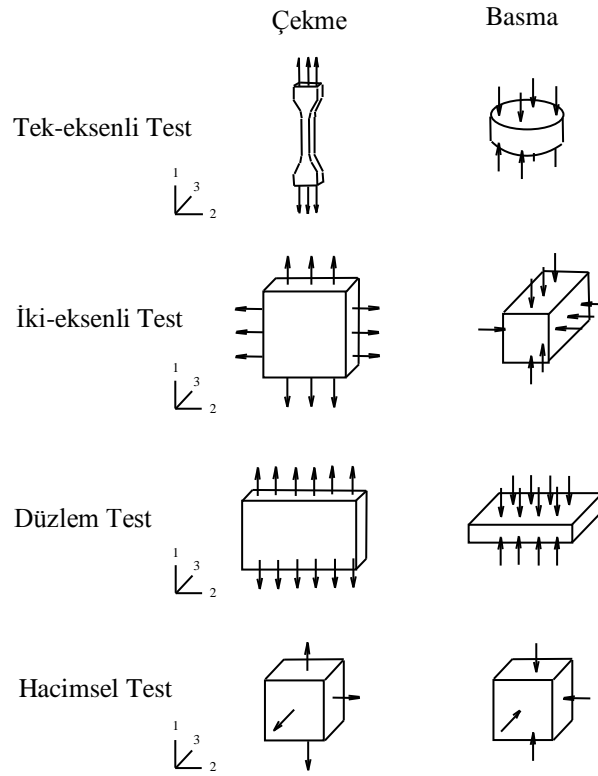
$$G^* = G'(1 + i\eta), \quad \eta = \frac{G''}{G'} \quad (4.30)$$

Burada; G' saklanan modülü (MPa), G'' kaybolan modülü (MPa) ve η boyutsuz kayıp faktörünü ifade etmektedir. Saklanan modül G' , bir malzemenin maruz kaldığı enerjiyi elastik olarak saklayabilmesini temsil ederken kaybolan modül G'' , bu enerjiyi dağıtabilme yeteneğini temsil etmektedir. Kompleks modül G^* ise bir malzemenin deformasyona karşı toplam dayanıklılığını temsil etmektedir.

4.3.2. Ürün malzeme testleri

Motor titreşim takozlarında kullanılan malzemelerin, lineer olmayan sonlu elemanlar analizlerinde kullanılacak malzeme modelleri için gerekli gerilme – birim yer değiştirme verilerinin elde edilmesi amacıyla çekme, basma ve kayma testleri gibi temel testler yapılmaktadır. Hiperelastik malzemelerin test yöntemleri ulusal ve uluslararası standartlarda henüz kesin hatları ile tanımlanmamıştır. Elastomer test numunelerinin çeşitli test yöntemleri ile elde edilen gerilme – birim yer değiştirme verileri eğri uydurma yöntemi ile araştırmacılar tarafından geliştirilen malzeme modellerinin tanımlanmasında kullanılmaktadırlar.

En sık kullanılan testler Şekil 4.32`de gösterilmiştir. Bunların içinde en temel test tek eksenli çekme testidir. Kauçuk ürünün gerçek çalışma şartlarında deformasyon davranışına uygun malzeme testleri yapılmalıdır. Sadece bir test yeterli olabileceği gibi mümkünse birden fazla test ile malzemenin davranışı daha doğru modellenebilir. Basit çekme, kayma ve bası testleri malzemenin farklı davranışlarını modellemek için gereklidir. Kauçuk malzemeler basma ve çekmede farklı davrandıklarından dolayı hem çekme hem de basma testleri yapılmalıdır (Simulia 2008).



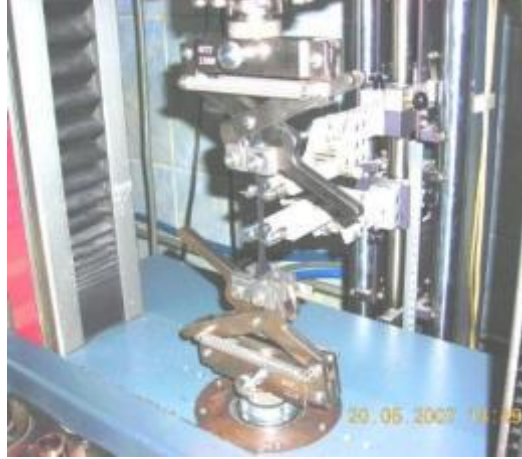
Şekil 4.32. Hiperelastik malzemeler için çeşitli testler (Simulia 2008)

Şekil 4.32`de gösterilen testlerin her birinin birbirinden farklı testler olduğu düşünülmemelidir. Bu testlerin bazıları birbirleri ile eşdeğer özellikler göstermektedir. Bu testler aşağıdaki gibidir (Simulia 2008):

- Tek eksenli çekme = İki eksenli basma
- Tek eksenli basma = İki eksenli çekme
- Düzlem çekme = Düzlem basma

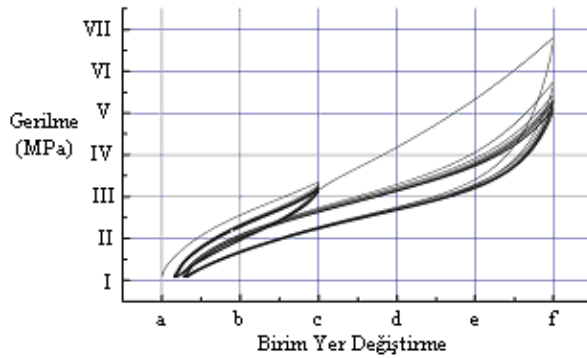
Tek eksenli çekme testi: En yaygın kullanılan test yöntemidir ve bu konuda birçok standart vardır. Ancak analiz için kullanılacak verilerin elde edilmesi için test yöntemi farklılıklar arz etmektedir. Örneğin saf çekme etkisini görmek için numunenin uzatılan boyu, genişlik ve kalınlığa göre oldukça fazla olmalıdır. Numunenin boyunun uzunluk ve genişliğine oranla en az 10 kat fazla olması fayda sağlamaktadır (Simulia 2008). Bu deneydeki amaç numunenin kopma mukavemetini ölçmek olmadığından numune standart boyutlardan farklı olabilmektedir.

Numune parçasının boyu, iki uçtan sıkılan çenelerin arasındaki boy olarak tanımlanmaktadır (Şekil 4.33). Sıkılan kısımdaki uzunluklar dikkate alınmamaktadır. Ölçme işlemi çenelerin uzağında, saf uzama olan bölgeden alınmıştır. Ancak daha hassas sonuçlar elde etmek için temassız video veya lazer ölçme cihazı kullanılmalıdır. Bu çalışmada kullanılan test cihazının ölçüm işlemi temaslı cinstendir (Şekil 4.33).



Şekil 4.33. Tek eksenli çekme testi

Çekme testinde ilk çevrimlerde malzeme özellikleri önemli ölçüde değişmektedir. Bu durum “Mullin Etkisi” olarak tanımlanmıştır. Belli bir çevrim sayısından sonra (3 ile 20 arası) malzeme davranışı kararlı hale gelmektedir. Eğer malzeme farklı daha yüksek gerilme aralığında çevrimsel teste tabi tutulursa yine birkaç çevrimden sonra malzemenin kararlı hale geldiği gözlenmiştir. Bu durum Şekil 4.34’te görülmektedir.



Şekil 4.34. Tek eksenli çekme testinde farklı iki yer değiştirme genliğinde malzeme davranışı

Basit Basma Testi: Bu test ile numunenin saf yer deęiřtirme durumu incelenmektedir. (řekil 4.35). Basma aparatı ile temas eden parça yzeylerinde srtünme söz konusudur. Parçanın basılması sırasında yanal yönlerde yer deęiřtirmeler tamamen serbest deęildir. Çok küçük srtünme katsayısı olsa bile srtünme yzeylerinde kayma gerilmesi oluşabilmektedir. Test esnasında numune ile test plakaları arasındaki srtünme yzeylerinde oluşan kayma gerilmesini düşürmek için ara yzeye kayganlařtırıcı sıvı sürülmüřtür.



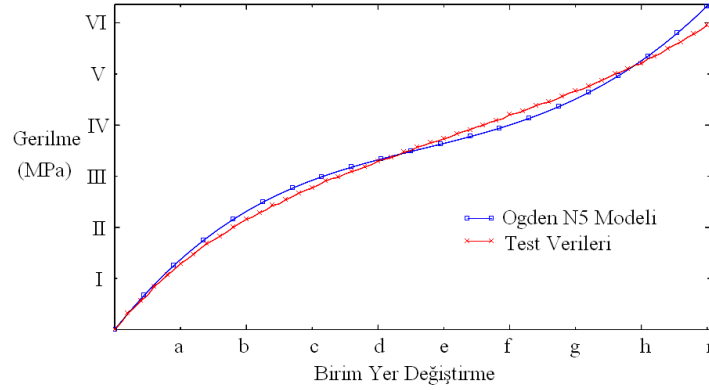
řekil 4.35. Basma testi

Düzlem Çekme Testi: Bu test tek eksenli çekme testine benzemektedir ancak numune uzunluęu geniřlięine oranla oldukça kısadır (řekil 4.36). Numunenin incelmesi kalınlık yönünde olmaktadır. Kauçuk malzeme sıkıřtırılmaz olduęundan dolayı çekme yönüne göre 45° açıda saf kayma meydana gelmektedir. Bunu saęlamak için numune geniřlięinin uzunluęuna göre en az 10 kat daha fazla olmalıdır (Simulia 2008).



řekil 4.36. Düzlem çekme testi

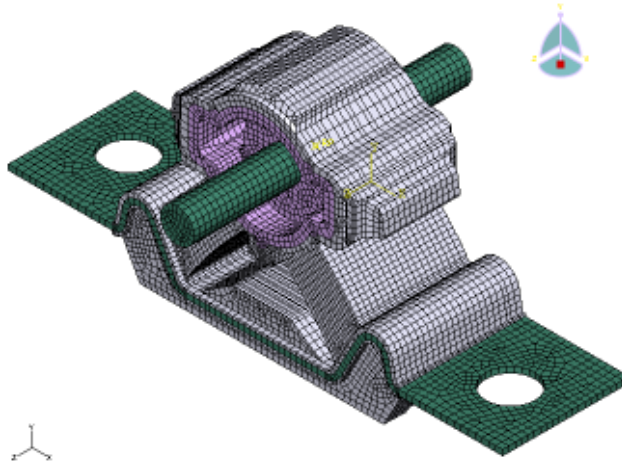
Söz konusu 3 testten (tek eksenli çekme, basit basma ve düzlem çekme testi) elde edilen değerler sonlu elemanlar malzeme modellerinin tanımlanmasında kullanılmıştır. Şekil 4.37'de tek-eksenli çekme testinden elde edilen veriler ile bu verilere uydurulan Ogden N5 malzeme modeli verilmiştir.



Şekil 4.37. Tek eksenli çekme testi ile uydurulan malzeme modeli

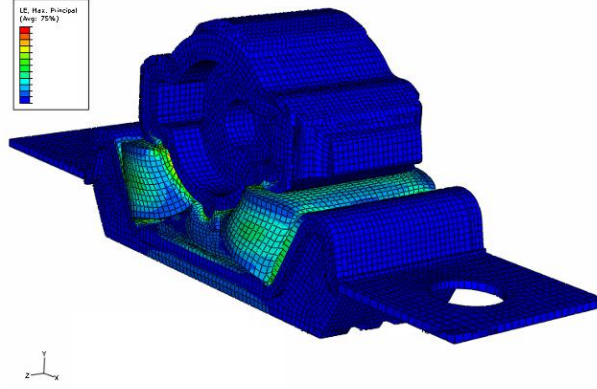
4.3.3. Ürün model simülasyonu ve tasarım doğrulaması

Mevcut ürünün üç boyutlu olarak modellenmesi ABAQUS ortamında gerçekleştirilmiş ve malzeme testlerinden elde edilen veriler ile uygun malzeme modeli belirlenerek örnek uygulama için belirlenen takozun 3 boyutlu modeli oluşturulmuştur (Şekil 4.38).

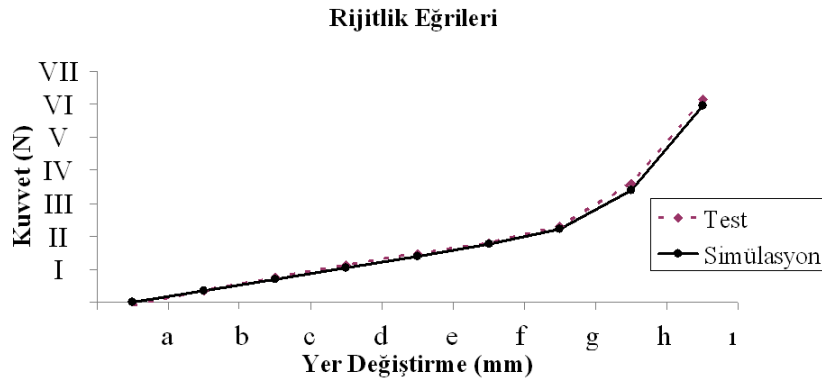


Şekil 4.38. Örnek uygulama için oluşturulan takoz modeli

Sınır şartları ve temas tanımları yapılarak sonlu elemanlar yöntemi ile analizler gerçekleştirilmiş (Şekil 4.39) ve böylece mevcut modelin statik rijitlik eğrisi elde edilmiştir (Şekil 4.40).



Şekil 4.39. Simülasyon sonucunda oluşan gerilme dağılımı



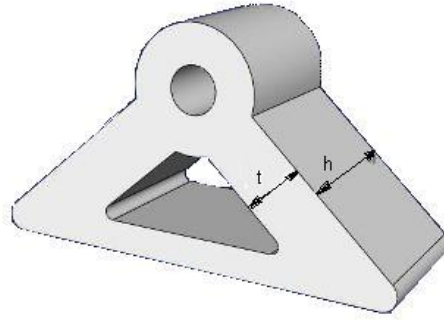
Şekil 4.40. Ürün testinden ve simülasyondan elde edilen rijitlik eğrileri

Grafikten görülebileceği gibi ürün testiyle simülasyondan elde edilen eğriler birbiriyle çakışmaktadır. Bu da arzu edilen korelasyonun elde edildiğini böylece optimizasyon işleminde simülasyondan elde edilen çıktılar rahatlıkla kullanılabilirliğini göstermektedir.

4.3.4. Motor titreşim takozu tasarım optimizasyon probleminin oluşturulması, çözülmesi ve sonuçları

Motor titreşim takozu tasarım optimizasyon probleminde optimizasyon işleminin amacı, motor takoz sisteminin doğal frekansını rezonans durumuna yol açmayacak, titreşim, ses ve şok sönümlenme yeteneğini geliştirecek bir düzeyde tutmaktır. Bunun için

motorda oluşan titreşimleri bağlantı noktalarından şasiye motor karakteristikleri incelenerek önceden karar verilen belli değerlerde sönümleyerek iletmesi beklenmektedir. Dolayısıyla motor titreşim takozunun rijitlik eğrisinin sınırları önceden bellidir. Sınırların içinden geçecek eğri istenen sönümleme değerlerine sahip olacaktır. Böylece farklı şekillere sahip tasarımların rijitlik eğrileri incelenerek istenen rijitlik eğrisini sağlayacak optimum tasarım belirlenebilir. Hedef eğriden olan sapmanın minimum olması optimizasyon işleminde amaç fonksiyonunu oluşturacaktır. Optimizasyon işlemi aşağıdaki model üzerinde gerçekleştirilmiştir (Şekil 4.41).



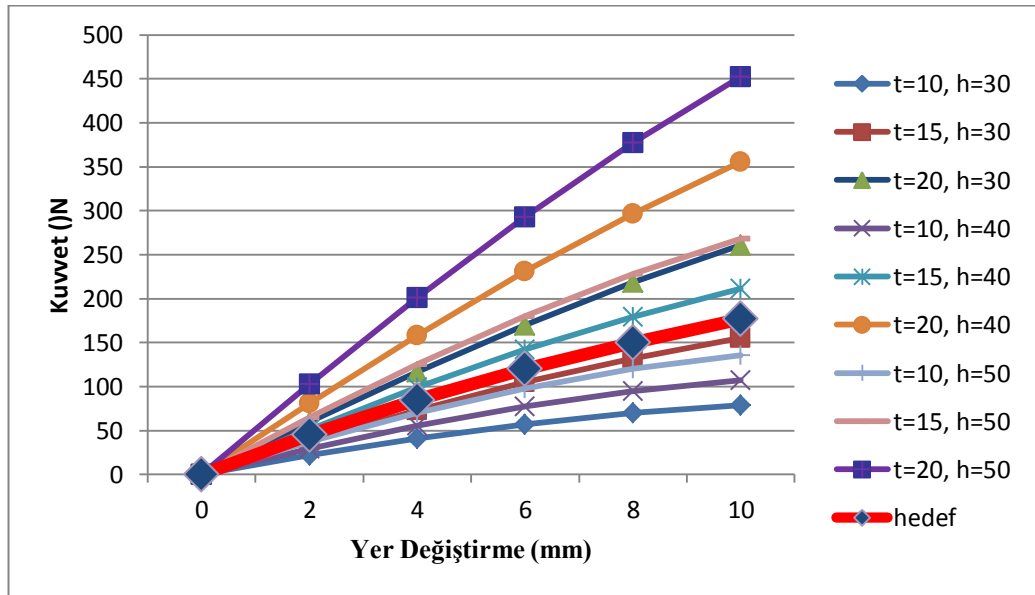
Şekil 4.41. Örnek uygulama için oluşturulan takoz modeli

Tasarım parametreleri olarak takoz kol genişliği (t) ve derinliği (h) belirlenmiştir. Bu parametreler $10 < t < 15$, $30 < h < 50$ aralıklarında değişmektedir. Bu verileri kullanarak gerçekleştirilen optimizasyon işlemi esnasında çok fazla sayıda doğrusal olmayan analiz gerçekleştirilmesi gerektiğinden optimizasyon işlemi süresi çok uzun zamanlar alabilmektedir. Bu nedenle bu verilere deney tasarımı yöntemi uygulanmasına karar verilmiştir. Bu yöntemde belirlenen şekil parametre sayısı ve her bir parametre için belirlenen seviye sayısına göre fazla sayıda çözüm gerektirdiğinden yöntem Şekil 4.41'de verilen model üzerine uygulanmıştır. Bu modelde her bir parametre için 3 seviye kullanılarak 45 çözüm gerçekleştirilmiş ve bu şekilde deney tasarımı tablosu oluşturulmuştur (Çizelge 4.14).

Çizelge 4.14. Kuvvet – yer değiştirme değerleri için oluşturulmuş ANOVA tablosu

Yer Değiştirme (mm)	Kuvvet (N)									Hedef Değer
	t=10 h=30	t=15 h=30	t=20 h=30	t=10 h=40	t=15 h=40	t=20 h=40	t=10 h=50	t=15 h=50	t=20 h=50	
0	0	0	0	0	0	0	0	0	0	0
2	21,8	38,0	59,9	29,5	51,3	81,1	37,2	65,0	102,9	45
4	41,1	73,1	116,8	55,6	98,9	158,4	70,2	125,4	201,0	85
6	57,2	104,8	170,0	77,6	141,9	230,8	98,1	180,0	293,0	120
8	69,8	132,3	218,4	94,9	179,5	296,9	120,1	227,9	377,2	150
10	78,5	155,3	261,2	107,0	211,1	355,7	135,6	268,3	452,3	177

Bu deneylerin her biri bir simülasyon içermektedir. Bu problem için toplam 45 simülasyon gerçekleştirilmiştir. Belirli yer değiştirmelere karşılık lineer olmayan hiperelastik malzeme modelleriyle gerçekleştirilen simülasyonlar sonucunda belirli kuvvet değerleri elde edilmiştir (Şekil 4.42).



Şekil 4.42. Hedef değerlerin ve deney tasarımı sonucunda elde edilen değerlerin kuvvet – yer değiştirme eğrileri

Ara değerlerdeki kuvvetlerin elde edilebilmesi için fonksiyon uydurma ile yer değiştirme x , t ve h değerlerine bağlı bir $F(x, t, h)$ cevap yüzey fonksiyonu elde edilmiştir. Üçüncü-dereceden kübik formdaki bu fonksiyon aşağıdaki forma sahiptir:

$$\begin{aligned}
 F = & -41,7+1,4*h+0,0295*h^2-0,0003126*h^3+5,06*t-0,371*h*t \\
 & +0,000537*h^2*t+0,172*t^2+0,011*h*t^2-0,0137*t^3+15,97*x-0,302*h*x+ \\
 & 0,000636*h^2*x-2,46*t*x+0,067*h*t*x+0,0811*t^2*x+0,481*x^2- \\
 & 0,015*h*x^2-0,0154*t*x^2-0,0178*x^3
 \end{aligned} \tag{4.31}$$

Optimizasyon işleminde 0,9999 R-kare değerine sahip olan bu fonksiyon kullanılmıştır. Optimizasyonun amaç fonksiyonu hedef değer verilerine uygun düşen h ve t değerlerini bulmaktır. Bir başka deyişle hedef değer verilerinden olan sapmanın minimize edilmesi amaç fonksiyonunu oluşturmaktadır. Bunun için; h ve t 'nin x 'in $[0, 2, 4, 6, 8, 10]$ değerlerine karşılık hesaplanan $f_0, f_2, f_4, f_6, f_8, f_{10}$ kuvvet değerleri ile hedef kuvvet değerleri arasındaki farkların mutlak değerlerinin toplamı toplam sapmayı ifade etmektedir. Minimize edilmeye çalışılan bu sapma değeri amaç fonksiyonunu oluşturmaktadır:

Amaç Fonksiyonu (Minimizasyon):

$$\begin{aligned}
 F = & |f_0| + |f_2| + |f_4| + |f_6| + |f_8| + |f_{10}| \quad (h=x1, t=x2) \\
 f_0 = & 0-(-41,7065+1,40545*x1+0,0295746*x1*x1-0,000312677*x1*x1*x1+5,06276*x2- \\
 & 0,371577*x1*x2+0,000537342*x1*x1*x2+0,172579*x2*x2+0,0110843*x1*x2*x2- \\
 & 0,0137163*x2*x2*x2+15,9784*0-0,302889*x1*0+0,000636988*x1*x1*0-2,46595*x2*0 \\
 & +0,0671359*x1*x2*0+0,081152*x2*x2*0+0,48182*0*0-0,0150977*x1*0*0-0,0154724* \\
 & x2*0*0-0,0178772*0*0*0) \\
 f_2 = & 45-(-41,7065+1,40545*x1+0,0295746*x1*x1-0,000312677*x1*x1*x1+5,06276*x2- \\
 & 0,371577*x1*x2+0,000537342*x1*x1*x2+0,172579*x2*x2+0,0110843*x1*x2*x2- \\
 & 0,0137163*x2*x2*x2+15,9784*2-0,302889*x1*2+0,000636988*x1*x1*2-2,46595*x2*2 \\
 & +0,0671359*x1*x2*2+0,081152*x2*x2*2+0,48182*2*2-0,0150977*x1*2*2-0,0154724* \\
 & x2*2*2-0,0178772*2*2*2) \\
 f_4 = & 85-(-41,7065+1,40545*x1+0,0295746*x1*x1-0,000312677*x1*x1*x1+5,06276*x2- \\
 & 0,371577*x1*x2+0,000537342*x1*x1*x2+0,172579*x2*x2+0,0110843*x1*x2*x2- \\
 & 0,0137163*x2*x2*x2+15,9784*4-0,302889*x1*4+0,000636988*x1*x1*4-2,46595*x2* \\
 & 4+0,0671359*x1*x2*4+0,081152*x2*x2*4+0,48182*4*4-0,0150977*x1*4*4-0,0154724* \\
 & x2*4*4-0,0178772*4*4*4) \\
 f_6 = & 120-(-41,7065+1,40545*x1+0,0295746*x1*x1-0,000312677*x1*x1*x1+5,06276*x2- \\
 & 0,371577*x1*x2+0,000537342*x1*x1*x2+0,172579*x2*x2+0,0110843*x1*x2*x2- \\
 & 0,0137163*x2*x2*x2+15,9784*6-0,302889*x1*6+0,000636988*x1*x1*6-2,46595*x2* \\
 & 6+0,0671359*x1*x2*6+0,081152*x2*x2*6+0,48182*6*6-0,0150977*x1*6*6-0,0154724* \\
 & x2*6*6-0,0178772*6*6*6)
 \end{aligned} \tag{4.32}$$

$$f_8=150-(-41,7065+1,40545*x_1+0,0295746*x_1*x_1-0,000312677*x_1*x_1*x_1+5,06276*x_2-0,371577*x_1*x_2+0,000537342*x_1*x_1*x_2+0,172579*x_2*x_2+0,0110843*x_1*x_2*x_2-0,0137163*x_2*x_2*x_2+15,9784*8-0,302889*x_1*8+0,000636988*x_1*x_1*8-2,46595*x_2*8+0,0671359*x_1*x_2*8+0,081152*x_2*x_2*8+0,48182*8*8-0,0150977*x_1*8*8-0,0154724*x_2*8*8-0,0178772*8*8*8)$$

$$f_{10}=177-(-41,7065+1,40545*x_1+0,0295746*x_1*x_1-0,000312677*x_1*x_1*x_1+5,06276*x_2-0,371577*x_1*x_2+0,000537342*x_1*x_1*x_2+0,172579*x_2*x_2+0,0110843*x_1*x_2*x_2-0,0137163*x_2*x_2*x_2+15,9784*10-0,302889*x_1*10+0,000636988*x_1*x_1*10-2,46595*x_2*10+0,0671359*x_1*x_2*10+0,081152*x_2*x_2*10+0,48182*10*10-0,0150977*x_1*10*10-0,0154724*x_2*10*10-0,0178772*10*10*10)$$

Sınır Kısıtları:

$$30 \leq x_1 \leq 50 \quad (4.33)$$

$$10 \leq x_2 \leq 20$$

Optimizasyon problemi kaba kuvvet ile arama algoritması (Brute Force), diferansiyel gelişim (DE) ve geliştirilen DEBVs algoritması ile 10 popülasyon sayısı ve 25 nesil sayısı kullanılarak çözülmüştür. Her iki algoritma için de hesaplama sayısı 250`dir. Güvenilirlik açısından her algoritma ile motor titreşim takozu tasarım optimizasyon problemi 30 defa çözdürülmüştür. Her iki algoritmada ölçeklendirme faktörü 0,85 ve çaprazlama oranı 0,9 olarak alınmıştır. En iyi sonuçlar geliştirilen DEBVs algoritması ile elde edilmiştir. DEBVs algoritması kullanılarak hesap edilen h ve t`nin optimum değerleri h = 38,2891 mm, t = 13,9688 mm`dir. Bu sonuçlar ile sapma 4,1749 olarak hesaplanmıştır. Ayrıca toplam hesaplama zamanının geliştirilen DEBVs algoritması ile yaklaşık olarak %38 oranında düşürülmesi vurgulanması gereken en önemli hususlardan birisidir (Çizelge 4.15).

Çizelge 4.15. Optimizasyon sonucunda elde edilen optimum h ve t değerleri

	h (mm)	t (mm)	Amaç Fonksiyonu, Sapma (N)	Hesaplama Zamanı %
DE	38,1381	14,0020	4,2944	100
Kaba Kuvvet Algoritması	38,2400	13,9800	4,1775	138
DEBVs	38,2891	13,9688	4,1749	62

4.3.6. Motor titreşim takozu tasarım optimizasyon çalışması ile elde edilen kazanımlar ve yapılabilecek yeni çalışmalar

Bu çalışmada, istenen statik ve dinamik özelliklere sahip motor askı takozlarının tasarım süreci işlemlerini tanımlayan bir algoritma geliştirilmiştir. Yeni ürünlerin tasarımında prototip üretim sayısının en aza indirgenmesi ve daha kısa zamanda seri üretime geçilmesi için kullanılacak simülasyon temelinde testlerle bütünlük tasarım süreci tanımlamaları yapılmıştır. İstenilen statik ve dinamik davranışlara sahip motor takozlarının, tanımlanan sürecin ve yöntemlerin uygulanması ile üretimi maliyet giderleri ve termin süreleri açısından ekonomik getirilere ek olarak araçlarda daha güvenli ve konforlu bir sürüş ortamı sağlanmasına yardımcı olacaktır.

Bu çalışma literatürde yer alan çalışmalara göre tüm tasarım süreci ve optimizasyonu ele alan, fiziksel testler ile simülasyonları süreç doğrulaması için kullanan, statik etkenleri tasarımın ilk aşamalarında modelde uygulayan öncü çalışmalardan birisidir. İleride yapılacak çalışmalarda dinamik etkenlerin de tasarım ve optimizasyona dâhil edilmesi planlanmaktadır.

Motor askı takozu tasarımının bütün aşamalarının Türkiye’de yapılması ve geliştirilmesi ile ürün “know-how”ına sahip olunmasına yönelik bir çalışma yürütülerek bu konuda tasarım yeteneğinin kazanılmasına yardımcı teknikleri içeren bir algoritma geliştirilmeye çalışılmıştır.

Yeni ürünlerin tasarımında prototip üretim sayısının en aza indirgenmesi ve daha kısa zamanda seri üretime geçilmesi için kullanılacak simülasyon temelinde testlerle bütünlük tasarım süreci tanımlamaları yapılmıştır. İstenilen özelliklerdeki davranışlara sahip motor takozlarının, tanımlanan sürecin ve yöntemlerin uygulanması ile üretimi maliyet giderleri ve termin süreleri açısından ekonomik getirilere ek olarak araçlarda daha güvenli ve konforlu bir sürüş ortamı sağlanmasına yardımcı olacaktır.

İstenilen statik ve dinamik davranışlara sahip motor takozlarının tasarımında yukarıda elde edilen kazanımlar ışığında tasarımdan üretime kadar olan sürecin kısaltılması ve sorunsuz bir şekilde en kısa zamanda üretime geçilmesi açılarından çalışmanın katkı sağlaması beklenmektedir.

Kimyasal içerik ve imalat proseslerinde yapılacak değişikliklerin ürün performansına olan etkileri incelenmek istendiğinde geliştirilen yaklaşım ile ürün imal edilmeden malzeme testleri yapılarak bilgisayar ortamında gerekli karakteristikler belirlenebilir.

Geliştirilen DEBVs algoritması motor askı takozu tasarım modeline uygulandığında simülasyon zamanı yaklaşık olarak %38 oranında kısaltılmıştır. Sürenin daha fazla kısaltılması için çok çekirdekli ve paralel işlem özelliklerine sahip iş istasyonu donanım ve yazılımları tercih edilebilir.

Bundan sonra yorulma analizi ile ürün ömrünün belirlenmesi çalışması ve sıcaklığın ürün karakteristiğine olan etkisini araştıran yeni bir çalışma yapılabilir.

4.4. Test Problemleri Seçimi ile ilgili Tartışma

Geliştirilen bir yöntemin doğruluğunu göstermek için en pratik yol o yöntemi veya algoritmayı kullanarak daha önceden çözülmüş problemleri tekrar çözmektir. Geliştirilen algoritma ile daha iyi sonuçların elde edilmesi veya en azından önceki sonuçlarla aynı sonucun elde edilmesi beklenir. Bu sayede yeni geliştirilen yöntemin etkinliğinin ispat edilmesi amaçlanır. Bu amaçla literatürde birçok test problemi kullanılmaktadır. Bazıları gerçek hayatta karşılaşılan ve çözümünü aranan bir problem iken zorluk derecesi ve geliştiricilerin o problemi çok sık kullanması nedeniyle popülerlik kazanmış, bazıları ise sırf geliştirilen algoritmayı test etmek için ortaya atılmış test fonksiyonlarından oluşmaktadır.

Qing (2009), test fonksiyonlarının seçiminde dikkat edilmesi gereken bazı noktaların olduğunu vurgulamış ve evrimsel hesaplama alanında test fonksiyonlarını seçerken birçok hatanın yapıldığını bildirmiştir. Bir evrimsel algoritmanın üzerine geliştirilen bir optimizasyon algoritmasının kullanılmasının daha avantajlı olduğunu iddia etmek için ve/veya bir evrimsel algoritmanın temel kontrol parametrelerinin belirli değerlerini kullanarak iyi sonuçlar bulmak için uygun olmayan sayısal ispatlar kullanılabilen ve bu ispatlar istismar edilebilmektedir.

Geliştirilen algoritmanın etkinliğini göstermek için test fonksiyonları seçme işleminde dört önemli noktaya dikkat etmek gerekmektedir (Qing 2009):

1. Test fonksiyonları rastgele seçilecekse fonksiyonlar hakkında yeterli bilgiye sahip olunmalı,
2. Temelsiz veya dayanağı olmayan genelleştirmelerden kaçınılmalı,
3. Tarafalı veya önyargılı karşılaştırma yapılmamalı,
4. Problem aktarılırken dikkatli olunmalı.

Evrimsel algoritmaların stokastik yani olasılıksal doğası gereği algoritmanın tek bir koşturma işlemi sonucunda elde edilen sonuç rastgelelik yüzünden çok küçük bir güvenilirlik değerine sahiptir. Genellikle bir algoritma ile elde edilen sonuçların değerlendirilmesinde algoritmanın kaç defa koşturulduğuna bakılmamaktadır. Bu nedenle sonuç karşılaştırma işleminin tek koşturma yerine birçok koşturma işlemi sonucunda elde edilecek ortalama değerler üzerinden yapılması daha objektif olacaktır. Algoritmaların koşturma sonuçlarının ortalamasının güvenilirliği koşturma sayısı ile doğru orantılı olarak artmaktadır. Bu nedenle geliştirilen DEBVs algoritmasının etkinliğini göstermek için seçilen test fonksiyonları en az 30 defa çözdürülmüş ve ortalaması alınmıştır.

Birçok araştırmacı diferansiyel gelişim algoritmasının hassasiyetinin temel kontrol parametrelerine bağlı olduğunu altını çizmektedir. Bu nedenle örnek olay incelemelerini bu kontrol parametreleri üzerinde gerçekleştirmişlerdir. Ancak bu örnek olay incelemeleri kontrol parametrelerinin çok sınırlı durumları için yapılmıştır. Kontrol parametrelerinin tüm değer aralıkları araştırılmadan birçok araştırmacı kısmi aralıklar kullanarak belirledikleri kontrol parametre değerlerinin en optimum değerler olduğunu iddia etmektedir. Böyle bir iddia hiç şüphesiz ki ikna edicilikten uzak ve şüpheli olacaktır. Başka bir temelsiz genelleştirme ise test problemlerinde yapılabilmektedir. Bazen uygulanan test problemi iyi tasarlanmamış olabilmekte, problem özellikleri özenle dikkate alınamamakta ve sadece sınırlı problem özellikleri için adres gösterilmektedir. Ancak buna rağmen son kabullerde genelleştirmeler yapılabilmektedir. Geliştirilen DEBVs algoritmasında örnek olay incelemeleri altı kontrol parametresinin tüm durumları için denenmiş ve en iyi sonuçların elde edildiği parametre değerleri test fonksiyonlarında kullanılmıştır.

Yeni bir optimizasyon algoritması geliştirildiğinde geliştirici bu algoritmanın yetkinliğini ispatlama ihtiyacı duymaktadır. En yaygın yaklaşım da mevcut diğer

optimizasyon algoritmaları ile karşılaştırma yapmaktır. Karşılaştırma yapılacak en az bir tane rakip seçme işleminde bazı önyargılı seçimler ve taraflı karşılaştırmalar yapılabilmektedir. Bununla ilgili evrimsel hesaplama alanında birçok örnek bulunabilir.

Her evrimsel algoritma kendine özgü kontrol parametrelerine sahiptir. Mesela genetik algoritmalarda popülasyon sayısı, çaprazlama oranı ve mutasyon oranı gibi. Aynı şekilde diferansiyel gelişim algoritmasında da popülasyon sayısı, çaprazlama oranı ve diferansiyel ölçeklendirme faktörü gibi temel üç kontrol parametresi mevcuttur. Bu parametrelerin uygun olmayan değerleri kullanıldığında açıktır ki kötü sonuçlar elde edilecektir. Dolayısıyla karşılaştırma yapılacak algoritmanın performansı da o derece düşecektir. Bu nedenle seçilen rakip evrimsel algoritmanın optimum kontrol parametre değerleri seçilmeyip her zaman sabit ve taraflı değerler kullanıldığı takdirde muhakkak ki yeni geliştirilen optimizasyon algoritmasının performansı daha iyi olacaktır. Dolayısıyla rakip evrimsel algoritmanın en iyi parametre değerleri ile bulunmuş sonuçların kıyaslamada kullanılması daha tarafsız bir değerlendirme yapmaya olanak tanıyacaktır. Bu durumda yeni geliştirilen optimizasyon algoritması eğer diğerlerinden daha iyi performansa sahipse bu değerlendirme daha gerçekçi olacaktır. Ancak maalesef bu işlemi gerçekleştirmek her zaman mümkün olmayabilir çünkü karşılaştırma yapılan algoritmaların en iyi kontrol parametre değerleri bilinmeyebilir. Geliştirilen DEBVs algoritması ile karşılaştırma yapılan DE algoritmasında benzer parametreler (popülasyon sayısı, çaprazlama oranı ve diferansiyel ölçeklendirme faktörü) aynı değerlerde kullanılmış ve bu değerler kullanılarak elde edilen sonuçlar karşılaştırılmıştır.

Diğer yandan algoritma geliştirenler özellikle kontrol parametreleri üzerinde uğraşmaları sonucunda buldukları en uygun kontrol parametrelerini üzerinde denedikleri problemler ile iyi sonuçlar elde ederken sayısı ve çeşitliliği az olan diğer bazı problemler için de uygun sonuçları bulduklarını iddia edebilmektedirler. Aslında geliştirilen algoritmada bulunan en uygun parametre değerleri farklı türde diğer problemlerde çoğunlukla iyi sonuçlar vermemekte ve bu durum kasıtlı veya kasıtsız olarak eksik bırakılarak sanki her problem için aynı iyi sonuçların elde edilebileceği gibi bir kanı oluşturulabilmektedir. Dolayısıyla problem çeşitliliği burada kilit rol oynamaktadır. Örneğin geliştirilen algoritmanın etkinliğini göstermek için ele alınan 13

farklı test fonksiyonunun hepsi ele alınmamakta en iyi bulunan birkaç fonksiyonun sonuçları üzerinde durulabilmektedir. Ancak yukarıda bahsedilen durumdan daha kötü bir durum da ortaya çıkmaktadır ki; belirli problemler için belirlenen ve en iyi değerler olarak bulunan kontrol parametre değerlerine güvenilip aynı değerler daha sonra yapılan çalışmalar için de kullanılmakta ve böylece artık genellikle hep bu bulunan değerler kullanılarak sanki gizli bir kural oluşmuş olmaktadır. Sonuçta belirli tipteki problemler diğer çalışmalara aktararak hata üstüne hata yapılabilmektedir.

Evrimsel hesaplama alanında yapılan hatalar yanlış varsayımlarla sonuçlanabilmektedir. Özellikle genetik algoritmalarda çaprazlamaya aşırı önem verilmesi ve mutasyona çok az önem verilmesi sonucunda beklenen sonuçların elde edilememesi buna örnek gösterilebilir. Benzer bir durum diferansiyel gelişim algoritmasında da mevcuttur. Örneğin DG/eniyi/1/* yaklaşımının lokal optimuma takılmaya DG/rastgele/1/* yaklaşımına göre çok daha fazla meyillidir varsayımı -temelinde en iyi bireylerle çalışan ve bu tez çalışması kapsamında geliştirilen DEBVs algoritması ile DE`den daha iyi sonuçlar elde edilmesi neticesinde- zayıflamaktadır.

Benzer şekilde diferansiyel gelişimde “çaprazlama çok önemli değildir” varsayımı mevcuttur. Ancak çözülen bazı test problemlerinde (özellikle fazla sayıda tasarım değişkenine sahip fonksiyonlarda) görülmüştür ki, çaprazlama oranının etkisi diğer parametrelere kıyasla ön plana çıkabilmektedir.

Evrimsel hesaplama alanında yapılan hatalar önemszenmezse gelecekte hatalar katlanarak artacaktır. Bunun sonucu oluşacak duruma şimdiden çözüm yolları aramak gerekmektedir. Bu nedenle hata yapıldığının farkına varılması veya vardırılması atılacak ilk adım olarak gözükmektedir. Qing (2009), gelecekte yapılması muhtemel hataların önlenmesi için şimdiden objektif bir ölçüm sistemi geliştirmek gerektiğini belirtmekte ve algoritma geliştiricilere iki çözüm yolu önermektedir:

Karşılaştırma yerine uygulama yapmak: Algoritma geliştiriciler geliştirdikleri algoritmanın performansının iyi olduğunu göstermek için ilk önce başka algoritmalar ile kıyaslama yapmadan önce sonucu bilinen ve hesapsal açıdan karmaşık olan uygulama problemlerini kendi geliştirdikleri algoritma ile çözebilirler ve bu hesaplama işlemini

birçok defa tekrar ederek buldukları sonuçların ortalamalarını alarak elde ettikleri sonucu değerlendirerek güvenilirlik açısından daha iyi bir yol izlemiş olacaktırlar.

Kapsamlı parametrik çalışmalar yapmak: Yeni bir optimizasyon algoritması öneren bir araştırmacı algoritmasının rakip evrimsel algoritmalara göre daha avantajlı olduğunu iddia ederek bunu göstermek isteyebilir. Bu durumda algoritmasını birçok defa çözdürmesi yanında kontrol parametreleri ile en azından rakipler kadar kapsamlı bir parametrik çalışma yapması zorunludur ve bu çalışma ile en iyi kontrol parametre değerlerini belirlemesi en iyi performansın parametre değerlerine göre değiştiğini göstermesi bakımından çok önemlidir. Elbette kontrol parametrelerinin sonsuz kombinasyonlarının olabileceği ve bunun hesapsal açıdan çok yorucu olduğu ve pratik olarak tüm kombinasyonları denemenin imkânsız olduğu muhakkaktır. Ancak daha makul bir yaklaşım olarak Qing (2009), sonsuz kontrol parametre setleri yerine belirli parametre setleri alınarak yapılacak çalışmalar sonucunda elde edilecek değerleri kontrol parametrelerinin uygunluk aralıklarının belirlenmesi açısından çok önemli görmüştür.

Geliştirilen DEBVs algoritmasının performansının değerlendirmesinde yukarıda belirtilen iki husus dikkate alınmıştır. Bunun için DEBVs`yi başka algoritmalar ile kıyaslamadan önce sonucu bilinen ve hesapsal açıdan karmaşık olan 13 test problemi (EK1) seçilerek çözülmüş ve güvenilirliği arttırmak için bu hesaplama işlemi en az 30 kez çözümlenerek bulunan sonuçların ortalamaları alınmıştır. Ayrıca tüm altı kontrol parametresi için belirli parametre setleri ile çözümler gerçekleştirilmiş ve elde edilen en iyi parametre değerleri ile problemler çözülmüştür.

Her bir test fonksiyonunda maksimum 500 000 hesaplama sayısındaki en iyi değerler ile ortalama değerler elde edilmiştir. Literatürde mevcut bulunan ve şu ana kadar elde edilen en iyi sonuçlara DEBVs ile 500 000 hesaplama sayısında ulaşamazsa 500 000 hesaplama sayısındaki sonuçlar çizelgelerde (Çizelge 3.6 - 3.18) verilmiştir.

Geliştirilen DEBVs algoritması ile 2. ve 3. test problemleri haricindeki geri kalan tüm 11 test probleminde literatürde şu ana kadar hesaplanan sonuçlardan daha iyi sonuçlara ulaşılmıştır.

Çizelge 3.6`da sonuçları verilen birinci test probleminde önerilen DEBVs yöntemi ile literatürde mevcut olan en iyi sonuca (-15) **76 000 hesaplama sayısı** ile ulaşılmıştır. DE aynı sonuca ancak **197 000** hesaplama sayısında ulaşabilmiştir. Ayrıca problem 30 defa çözülmüş ve ortalama fonksiyon değeri ile standart sapma hesaplanmıştır. 500 000 hesaplama sayısında DEBVs`nin standart sapması 10^{-15} mertebelerinde iken DE`nin standart sapması 10^{-6} mertebelerindedir. Bu sonuç DEBVs`nin güvenilirliğini kanıtlamaktadır.

Çizelge 3.7`de sonuçları verilen ikinci test probleminde önerilen DEBVs yöntemi ile literatürde mevcut olan en iyi sonuca (-0,80361910412559) 1 733 000 hesaplama sayısı ile ulaşılmıştır. DE ile aynı hesaplama sayısında bu sonuca ulaşamamıştır. DEBVs ile 500 000 hesaplama sayısında -0,803618784948967 sonucuna ulaşılmıştır. İkinci test probleminde amaç fonksiyonu ve kısıtlar incelendiğinde ayrık fonksiyonlara sahip olduğu görülmektedir. Böylece DEBVs algoritmasının ayrık fonksiyonlar için geliştirmeye açık bir yönünün olduğu söylenebilir. Çizelge 3.8`de sonuçları verilen üçüncü test probleminde 500 000 hesaplama sayısında hem DE ile hem de DEBVs ile en iyi değerden daha iyi sonuçlara ulaşamamış, çok yakın değerler elde edilmiştir. Bunun nedeni olarak eşitlik kısıtı içeren bu problemde eşitlik kısıtı için özel bir algoritma kullanılmayarak mevcut yaklaşım ile problemin çözülmesidir. Eşitlik kısıtları için yeni etkin bir yöntem ileride yapılacak çalışmalarda geliştirilebilir.

Çizelge 3.9`da sonuçları verilen dördüncü test probleminde önerilen DEBVs yöntemi ile literatürde mevcut olan en iyi sonuca (-30 665,539) 25 800 hesaplama sayısı ile ulaşılmıştır. DE aynı sonuca ancak 59 600 hesaplama sayısında ulaşabilmiştir. Problem 30 defa çözülmüş ve ortalama fonksiyon değeri ile standart sapma hesaplanmıştır. 500 000 hesaplama sayısında hem DEBVs`nin hem de DE`nin standart sapması 10^{-11} mertebelerindedir.

Çizelge 3.10`da sonuçları verilen beşinci test probleminde önerilen DEBVs yöntemi ile 500 000 hesaplama sayısında literatürde mevcut olan en iyi sonuçtan (5 126,4967140071) daha hassas bir değer (5 126,496714007099) elde edilmiştir.

Çizelge 3.11`de sonuçları verilen altıncı test probleminde önerilen DEBVs yöntemi ile 30 000 hesaplama sayısı ile literatürde mevcut olan en iyi sonuçtan (-6 961,81387558015) daha iyi bir sonuç (-6 961, 81387558016) elde edilmiştir.

Çizelge 3.12`de sonuçları verilen yedinci test probleminde önerilen DEBVs yöntemi ile 500 000 hesaplama sayısı ile literatürde mevcut olan en iyi sonuçtan (24,30620906818) daha iyi bir sonuç (24,306209068179780) elde edilmiştir. Ayrıca 30 tekrar çözümde 10^{-15} mertebelerinde olan standart sapma ile hesaplanan ortalama değer literatürde mevcut olan en iyi değerden daha iyi hesaplanmıştır (24,306209068179786). Bu sonuç DEBVs algoritmasının güvenilirliği hakkında çok olumlu bilgiler vermektedir.

Çizelge 3.13`te sonuçları verilen sekizinci test probleminde önerilen DEBVs yöntemi ile 7 500 hesaplama sayısı ile literatürde mevcut olan en iyi sonuç ile (-0,0958250414180359) aynı sonuç (-0,095825041418036) elde edilmiştir. Ayrıca 30 tekrar çözümde standart sapma 10^{-18} mertebelerindedir.

Çizelge 3.14`te sonuçları verilen dokuzuncu test probleminde geliştirilen DEBVs yöntemi ile 500 000 hesaplama sayısı ile literatürde mevcut olan en iyi sonuçtan (680,630057374402) daha hassas sonuç (680,6300573744014) elde edilmiştir. Ayrıca 30 tekrar çözümde 10^{-13} mertebelerinde olan standart sapma ile hesaplanan ortalama değer literatürde mevcut olan en iyi değerden daha iyi olmaktadır (680,6300573744018).

Çizelge 3.15`te sonuçları verilen onuncu test probleminde geliştirilen DEBVs yöntemi ile 500 000 hesaplama sayısı ile literatürde mevcut olan en iyi sonuçtan (7 049,24802052867) daha iyi sonuç (7 049,248020528663) elde edilmiştir.

Çizelge 3.16`da sonuçları verilen on birinci test probleminde geliştirilen DEBVs yöntemi ile 2 500 hesaplama sayısı ile literatürde mevcut olan en iyi sonuca (0,7499) ulaşılmıştır. Çizelge 3.17`de sonuçları verilen on ikinci test probleminde geliştirilen DEBVs yöntemi ile 22 500 hesaplama sayısı ile literatürde mevcut olan en iyi sonuca (-1) ulaşılmıştır.

Çizelge 3.18`de sonuçları verilen on üçüncü son test probleminde geliştirilen DEBVs yöntemi ile 500 000 hesaplama sayısı ile literatürde mevcut olan en iyi sonuçtan (0,053941514041898) daha iyi sonuç (0,053937847573861) elde edilmiştir.

4.5. Mühendislik Problemleri ile ilgili Tartışma

Çizelge 3.19`da sonuçları verilen basınçlı kap tasarım problemi Deb (1997) tarafından genetik uyarlamalı arama algoritması kullanılarak, Kannan ve Kramer (1994) tarafından çoğaltılmış Lagrange çarpanları algoritması kullanılarak ve Sandgren (1988) tarafından dal sınır yöntemi kullanılarak çözülmüştür. Her yöntem ile 30 defa çözdürülen basınçlı kap probleminde bulunan en iyi sonuçlar tabloda verilmiştir. Kannan ve Kramer (1994)`in yöntemi kullanıldığında birinci kısıt sağlanamadan en iyi sonuca ulaşılmıştır. Benzer şekilde Sandgren (1998) tarafından dal sınır yöntemi kullanılarak çözdürülen basınçlı kap probleminde üçüncü kısıt sağlanamamıştır. Coello ve Montes (2002) problemi dominant tabanlı turnuva seçim mekanizması ile 30 defa çözdürmüşler ve en iyi sonuçları tüm kısıtları sağlayacak şekilde 80 000 hesaplama sayısı ile elde etmişlerdir. Problem diferansiyel gelişim algoritması ile çözüldüğünde 5 000 hesaplama sayısı ile diğer yöntemlerden daha iyi sonuç (6 059,719052) elde edilmiştir. Son olarak problem önerilen DEBVs yöntemi ile 30 tekrar ile çözülmüş ve literatürde mevcut olan sonuçlardan daha iyi olan sonuca (6 059,714337) 5 000 hesaplama sayısı ve 0,000035 standart sapma değeri ile tüm kısıtlar sağlanarak ulaşılmıştır.

Çizelge 3.20`de sonuçları verilen kaynaklı giriş tasarım problemi Siddall (1972) tarafından çeşitli optimizasyon yöntemleri (ADRANS: Ceza fonksiyonu ile Gall`ın uyarlamalı rastgele arama algoritması, APPROX: Griffith ve Steward`ın başarılı doğrusal yaklaşım yöntemi, DFP: Ceza fonksiyonlu Davidon-Fletcher-Powell yöntemi, MEMGRD: Ceza fonksiyonlu Miele`nin hafıza gradyan yöntemi, SIMPLEX: Ceza fonksiyonlu Simplex yöntemi ve RANDOM: Richardson`un rastgele yöntemi) kullanılarak, Ragsdell ve Phillips (1976) tarafından geometrik programlama kullanılarak, Deb (1991) tarafından klasik genetik algoritma ve ceza fonksiyonu kullanılarak, Coello (1999) tarafından kendi kendine uyarlamalı ceza fonksiyonu ile kısıtlar ele alınmış genetik algoritma kullanılarak, Coello ve Montes (2002) tarafından dominant tabanlı turnuva seçim mekanizması kullanılarak, Ray ve ark. (2002) tarafından akıllı bilgi paylaşım stratejisini kullanan sürü yöntemi kullanılarak

çözülmüştür. Problem diferansiyel gelişim algoritması ile çözüldüğünde 5 000 hesaplama sayısı ile 1,736723 sonucu elde edilmiştir. Son olarak problem önerilen DEBVs yöntemi ile 30 tekrar ile çözülmüş ve literatürde mevcut olan sonuçlardan daha iyi olan sonuca (1,724897) 5 000 hesaplama sayısı ve 0,001360 standart sapma değeri ile tüm kısıtlar sağlanarak ulaşılmış, 25 000 hesaplama sayısı ile **1,724852** değeri elde edilmiştir.

Çizelge 3.21`de sonuçları verilen çok-amaçlı kaynaklı giriş tasarım problemi Deb (2001) ve Deb (1999b) tarafından çözülmüş ve oluşan Pareto eğrisinin farklı bölgelerinde üç farklı sonuç elde edilmiştir. Problem DE ve DEBVs ile 30 defa çözülmüş ve oluşan standart sapma değerleri hesap edilmiştir. Birinci çözümde Deb (2001)`in elde ettiği sonuçlara (2,431000 - 0,015700) kıyasla DE ile 1000 hesaplama sayısında daha iyi sonuçlar (2,108615 - 0,009113) elde edilirken aynı hesaplama sayısında DEBVs ile en iyi sonuçlar (2,102603 - 0,009067) 0,292358 standart sapma değeriyle elde edilmiştir. 12 500 hesaplama sayısı ile DEBVs kullanılarak şu ana kadarki en iyi sonuçlara (2,099159 - 0,009062) daha düşük standart sapma değeri ile (0,002850) ulaşılmıştır. İkinci çözümde Deb (1999b)`in elde ettiği sonuçlara (3,944000 - 0,005000) kıyasla DE ile 1000 hesaplama sayısında daha iyi sonuçlar (3,836345 - 0,004565) elde edilirken aynı hesaplama sayısında DEBVs ile en iyi sonuçlar (3,833464 - 0,004538) 0,351970 standart sapma değeriyle elde edilmiştir. 5 000 hesaplama sayısı ile DEBVs kullanılarak şu ana kadarki en iyi sonuçlara (3,829864 - 0,004537) tüm kısıtlar sağlanacak şekilde ulaşılmıştır. Üçüncü çözümde ikinci amaç fonksiyonu önemsenerek minimize edilmeye çalışılmış ve Pareto eğrisinin uç noktalarına ulaşılması hedeflenmiştir. Deb (1999b)`in elde ettiği sonuçlara (20,677000 - 0,001000) kıyasla DE ile sadece 300 hesaplama sayısında daha iyi sonuçlar (18,175666 - 0,00087373) elde edilirken aynı hesaplama sayısında DEBVs ile en iyi sonuçlar (18,147997 - 0,00087370) elde edilmiştir. 3 000 hesaplama sayısı ile DEBVs kullanılarak şu ana kadarki en iyi sonuçlara (18,125091 - 0,00087343) tüm kısıtlar sağlanacak şekilde 0,638084 standart sapma değeri ile ulaşılmıştır.

Çizelge 3.22`de sonuçları verilen çok-amaçlı iki-çubuklu giriş tasarım problemi Deb ve ark. (2000) ve Palli ve ark. (2004b) tarafından çözülmüş ve oluşan Pareto eğrisinin iki farklı bölgesinde sonuç elde edilmiştir. Problem DE ve DEBVs ile 30 defa çözülmüş ve

oluşan standart sapma değerleri hesap edilmiştir. Birinci çözümde Deb ve ark. (2000)'nin elde ettiği sonuçlara (0,053040 - 8 439,000000) kıyasla DE ile sadece 200 hesaplama sayısında daha iyi sonuçlar (0,035361 - 8 432,750183) elde edilirken aynı hesaplama sayısında DEBVs ile en iyi sonuçlar (0,031631 - 8 432,741218) kısıt sağlanacak şekilde elde edilmiştir. 800 hesaplama sayısı ile DEBVs kullanılarak şu ana kadarki en iyi sonuçlara (0,031623 - 8 432,740427) daha düşük standart sapma değeri ile (0,014146) ulaşılmıştır. İkinci çözümde Deb ve ark. (2000)'nin elde ettiği sonuçlara (0,004070 - 99 755,000000) ve Palli ve ark. (1999)'nin elde ettiği sonuçlara (0,004445 - 89 983,000000) kıyasla DE ile 500 hesaplama sayısında daha iyi sonuçlar (0,002309 - 69 287,634325) elde edilirken aynı hesaplama sayısında DEBVs ile en iyi sonuçlar (0,002309 - 69 282,522879) elde edilmiştir. 1 000 hesaplama sayısı ile DEBVs kullanılarak şu ana kadarki en iyi sonuçlara (0,002309 - 69 281,820669) tüm kısıtlar sağlanacak şekilde 1,817586 standart sapma değeri ile ulaşılmıştır.

Çizelge 3.23'te sonuçları verilen çok-amaçlı I-kesitli giriş tasarım problemi Coello ve Christiansen (1998) tarafından çözülmüş ve oluşan Pareto eğrisinin iki farklı bölgesinde sonuç elde edilmiştir. Problem DE ve DEBVs ile 30'ar defa çözdürülmüş ve standart sapma değerleri hesap edilmiştir. Birinci çözümde Coello ve Christiansen (1998)'nin elde ettiği sonuçlara (312,770000 - 0,012700) kıyasla DE ile 1 200 hesaplama sayısında daha iyi sonuçlar (310,122289 - 0,012602) elde edilirken aynı hesaplama sayısında DEBVs ile en iyi sonuçlar (310,116083 - 0,012602) kısıt sağlanacak şekilde elde edilmiştir. İkinci çözümde Coello ve Christiansen (1998)'nin elde ettiği sonuçlara (315,360000 - 0,012900) kıyasla DE ile 1 200 hesaplama sayısında daha iyi sonuçlar (307,784721 - 0,012708) elde edilirken aynı hesaplama sayısında DEBVs ile en iyi sonuçlar (307,771657 - 0,012708) kısıt sağlanacak şekilde elde edilmiştir.

Mühendislik optimizasyon problemleri çözülerek DEBVs ile elde edilen sonuçlar bu tez çalışması kapsamında geliştirilen DEBVs optimizasyon algoritmasının etkinlik, kullanılabilirlik ve doğruluk gibi önemli özelliklere sahip olduğunu kanıtlamaktadır.

5. SONUÇ

Bu tez çalışmasında, evrimsel algoritmalar arasında yer alan global, güvenilir, kullanışlı ve birçok test probleminde başarılı bir şekilde geçen diferansiyel gelişim algoritması ele alınmış ve popülasyon içindeki en iyi vektörleri fark vektörleri olarak kullanan yeni bir mutasyon stratejisi geliştirilerek geliştirilen algoritmada kullanılmıştır (DEBVs). Geliştirilen DEBVs algoritması ile literatürde mevcut bulunan ve yeni geliştirilen birçok algoritmanın test edilmesinde kullanılan kısıt içeren test problemleri çözdürülmüş ve diğer yöntemlere göre **daha düşük sayıda hesaplama ile daha iyi sonuçlar** elde edilmiştir. Böylece geliştirilen yeni algoritmanın doğruluğu mevcut sonuçlardan daha iyi sonuçlara **daha hızlı bir şekilde** ulaşma imkânı ile kanıtlanmıştır. Ayrıca önerilen yöntem ile literatürde mevcut tek ve çok-amaçlı çeşitli mühendislik problemleri çözdürülmüş ve mevcut sonuçlardan daha iyi sonuçlar elde edilerek geliştirilen yöntemin doğruluğu pekiştirilmiştir. Bulunan sonuçlar, geliştirilen yöntemin uygulanabilirliği, basitliği ve hesaplama performansı açısından iyi düzeydedir ve klasik metotlarla elde edilmesi çok zor olan neticelere lokal optimumlara yakalanmadan çok daha **kısa zamanda** ulaşılmıştır.

Geliştirilen yöntem daha önce klasik ve diğer evrimsel algoritmalarla çözülmüş üç farklı taşıt tasarım optimizasyon probleminde kullanılmış ve daha kısa zamanda daha iyi sonuçlara ulaşılmıştır. Bilgisayar destekli simülasyon uygulamaları ile yapılan bu örnek çalışmalar, gerçek hayatta karşılaşılan problemlerin daha düşük maliyette ve daha kısa zamanda çözümlenmesine yardımcı olmuştur.

Birinci taşıt tasarım optimizasyon probleminde istenen özelliklerde taşıt sürüş konforu için taşıt araştırma ve geliştirme sürecinde kullanılacak sistematik bir yaklaşım geliştirilmiştir. Sürüş konfor hesaplamasında kullanılacak ve dört farklı taşıtın altı farklı yolda koşturulmasıyla elde edilen bir benzetim-tabanlı model geliştirilmiştir. Bu model koltuğun oturma bölgesinden, sırt bölgesinden ve ayak bölgesinden elde ettiği ivme değerlerini kullanarak sürüş konfor değerini motor, süspansiyon ve koltuk parametrelerini de dâhil ederek hesaplamaktadır. Biri sürüş konforunu hesaplayabilmek için gerekli verileri sağlamak diğeri tam-taşıtlı modeli için gerekli parametre değerlerini toplamak amacıyla iki fiziksel test gerçekleştirilmiştir. Fiziksel testler ile elde edilen sürüş konfor değerleri ile simülasyon sonucunda elde edilen değerler karşılaştırılarak

simülasyon doğrulanmıştır. Testlerden elde edilen sonuçlar ile benzetim çalışması ile elde edilen sonuçlar arasındaki uyum umut vericidir. Kurulan model için bir arayüz geliştirilmiş ve maksimum konfor için optimizasyon arayüzü oluşturulmuştur. Geliştirilen DEBVs algoritması ile sürüş konfor optimizasyon problemi çözdürülmüştür. Çalışma sonucunda geliştirilen algoritma ile araç geliştirme sürecinde tasarımcının referans olarak kullanabileceği bir yardımcı tasarım aracı oluşturulmuştur. Geliştirilen modelin fiziksel prototiplere ihtiyaç duyulmadan taşıt tasarım sürecinde tasarımcıya yardımcı bir araç olarak kullanılması zaman (~%35) ve maliyet açısından avantajlar sağlaması beklenmektedir.

İkinci olarak taşıt sac üretiminde kullanılan klasik kalıp tasarım sürecinden farklı olarak kalıp tasarım sürecine istenen rijitlikte kalıp tasarımı için simülasyon tabanlı topoloji yaklaşımı entegre edilmiş problem geliştirilen DEBVs algoritması ile çözdürülmüştür. Simülasyon sonuçları fiziksel test sonuçları ile doğrulanmış ve topoloji optimizasyonu ile yeni kalıp tasarımları gerçekleştirilmiştir. Daha hafif ve daha rijit kalıp modelleri hazırlanarak yapısal optimizasyon problemi oluşturulup geliştirilen DEBVs algoritması ile çözdürülmüştür. Bu örnek çalışma ile kalıp tasarımcılarının kalıp tasarım sürecinde yeni teknikleri kullanabilme özelliği kazanmasına yardımcı olunmaya çalışılmıştır. Daha hafif ve istenen rijitlikte kalıpların tasarlanması için kullanılacak yeni tekniklerden oluşan, kalıp tasarımcılarına yardımcı olacak bir yaklaşım uygulanmıştır. Elde edilen korelasyon değerleri yaklaşımın uygulanabileceğini göstermiştir ve bu değerler uygulanan yöntemin doğrulaması açısından oldukça iyi değerlerdir. Dört alternatif yapının denendiği çalışmada üçüncü alternatif (A3) veya dördüncü alternatif (A4) tercih edilebilir. Mevcut durumda kalıplarda yaklaşık %24 oranında kütlede azalma ve %72 oranında gerilmelerde azalma gerçekleştirilmiştir. İleriki çalışmalarla ve revizyonlarla kütle azalması yüzdelerinin artırılarak %30'lara ulaşılacağı öngörülebilir. Kalıpların ağırlığının azaltılmasıyla kalıp maliyetlerinde düşüş sağlanacaktır. Kalıpların esneme değerleri belirli değerlerde tutularak alıştırma işçiliğinin azaltılmasına katkı sağlanması beklenmektedir. Ayrıca kullanılan döküm malzeme miktarının azaltılması ile çevreye zararlı gazların salınımı azaltılmış olacaktır. Diğer önemli noktalardan birisi geliştirilen DEBVs algoritması ile toplam hesaplama zamanının %35 oranında azaltılmış olmasıdır. Elde edilen sonuçlardan açıkça görülmektedir ki geliştirilen yaklaşımın kalıpların geliştirme sürecinde tasarımcı

mühendisler tarafından kullanılması daha yenilikçi tasarımların ortaya çıkmasına olanak sağlayacaktır.

Üçüncü olarak istenen özelliklere sahip motor askı takozlarının tasarımı için kullanılacak simülasyon ve fiziksel testler ile optimizasyona dayalı bu konuda tasarım yeteneği kazanılmasına yardımcı olacak bir yapı oluşturularak geliştirilen DEBVs algoritması ile çözdürülmüştür. Çeşitli ürün testleri ile elde edilen sonuçlar ile kurulan doğrusal olmayan sonlu elemanlar modelinin doğruluğu araştırılmıştır. İstenen özelliklerde motor askı takoz tasarımı parametrik olarak gerçekleştirilmiş ve geliştirilen DEBVs algoritması ile çözdürülmüştür. Diğer yöntemlere göre daha iyi sonuçlar elde edilmiş ve **hesaplama zamanında yaklaşık %38 oranında tasarruf sağlanmıştır**. Bu motor takoz optimizasyon çalışması, literatürde yer alan çalışmalara göre tüm tasarım süreci ve optimizasyonu ele alan, fiziksel testler ile simülasyonları süreç doğrulaması için kullanan, statik etkenleri tasarımın ilk aşamalarında modelde uygulayan öncü çalışmalardan birisidir. Yeni ürünlerin tasarımında prototip üretim sayısının en aza indirgenmesi ve daha kısa zamanda seri üretime geçilmesi için kullanılacak simülasyon temelinde testlerle bütünleşik tasarım süreci tanımlamaları yapılarak istenilen statik ve dinamik davranışlara sahip motor takozlarının tanımlanan sürecin ve yöntemlerin uygulanması ile üretimi maliyet giderleri ve termin süreleri açısından ekonomik getirilere ek olarak taşıtlarda daha güvenli ve konforlu bir sürüş ortamı sağlanmasına yardımcı olması beklenmektedir. İleride yapılması planlanan çalışmalarda motor askı takoz tasarım sürecine etki eden yorulmanın, çevresel etkilerin ve malzemenin viskoelastik özelliklerinin dahil edilmesi ve özellikle dinamik özelliklerin sisteme dahil edilerek optimizasyon çalışmasının geliştirilmesi düşünülmektedir.

Taşıt tasarımında üç farklı problem geliştirilen DEBVs algoritması ile çözülmüştür. Diğer evrimsel algoritmalara ve klasik optimizasyon yöntemlerine göre çok daha güvenilir, hızlı, global ve etkin sonuçlar elde edilmiştir. Geliştirilen yöntemin taşıt tasarım sürecinde tasarımcıya yardımcı bir araç olarak kullanılması **zaman, maliyet, kalite, hız ve kolaylık** açılarından büyük avantajlar sağlaması beklenmektedir. Bu tez çalışmasında geliştirilen ve başarıyla taşıt tasarım problemlerine uygulanan simülasyon tabanlı bütünleşik yaklaşım diğer mühendislik problemlerinin çözümünde rahatlıkla kullanılabilir.

Geliştirilen DEBVs optimizasyon algoritması ile ele alınan test problemlerinde ve mühendislik problemlerinde bulunan sonuçların şu ana kadar literatürde geliştirilen yöntemler ile bulunan sonuçlardan daha iyi olması ve örnek taşıt tasarım optimizasyon problemlerinde başarıyla kullanılması bu tez çalışması kapsamında geliştirilen yaklaşımın etkinliğini ve doğruluğunu kanıtlamaktadır.

Bu çalışmada yoğun olarak mutasyon operatörü üzerinde durulmuştur. İleride yapılacak çalışmalarda diğer evrimsel operatörler (özellikle çaprazlama) üzerinde yapılacak geliştirmeler ile DEBVs algoritması daha da iyileştirilebilir.

Ayrıca geliştirilen DEBVs algoritması paralel kodlamaya uygun olarak geliştirilmiştir. İleride yapılacak çalışmalar ile problemler paralel hesaplama kullanılarak çok daha kısa hesaplama süreleri ile çözülebilirler.

KAYNAKLAR

- Abbass, H.A., Sarker, R., Newton, C. 2001.** PDE: A Pareto-Frontier Differential Evolution Approach for Multiobjective Optimization Problems. IEEE Congress Evolutionary Computation, May 27–30, Seoul, South Korea, vol. 2, pp. 971–978.
- Abbass, H.A. 2002.** Self-adaptive Pareto Differential Evolution. The IEEE Congress on Evolutionary Computation (CEC2002).
- Abbass, H.A., Sarker, R. 2002.** The Pareto Differential Evolution Algorithm. *International Journal on Artificial Intelligence Tools*, World Scientific, 11(4): 531-552.
- Ainslie, M.A., Hamson, R.M., Horsley, G.D., James, A.R., Laker, R.A., Lee, M.A., Miles, D.A., Richards, S.D. 2000.** Deductive Multi-Tone Inversion of Seabed Parameters. *Journal of Computational Acoustics*, 8(2): 271-284.
- Akkaram, S., Beley, J.D., Maffeo, B., Wiggs, G. 2006.** Analytical Derivatives Technology for Structural Shape Design. *Structural and Multidisciplinary Optimization*, DOI: 10. 1007/s00158-006-0069-z, 12 Dec 2006. Springer, Berlin.
- Ali, M.M., Törn, A. 2004.** Population Set-Based Global Optimization Algorithms: Some Modifications and Numerical Studies. *Computers and Operations Research*, 31(10): 1703–1725.
- Ali, M.M., Fatti, L.P. 2006.** A Differential Free Point Generation Scheme in the Differential Evolution Algorithm. *Journal of Global Optimization*, 35(4): 551–572.
- Ali, M.M., Smith, R., Hobday, S. 2006.** The Structure of Atomic and Molecular Clusters, Optimised Using Classical Potentials. *Computer Physics Communications*, 175(7): 451–464.
- Altair. 2008.** HyperWorks, OptiStruct Software.
- Annicchiario, W., Cerrolaza, M. 1999.** A Finite Elements, Genetic Algorithms and B-Splines, A Combined Technique for Shape Optimization. *Finite Elements in Analysis and Design*, 33: 125-141.
- Annicchiario, W., Cerrolaza, M. 2004.** A 3D Boundary Element Optimization Approach Based on Genetic Algorithms and Surface Modeling. *Engineering Analysis with Boundary Elements*, 28: 1351-1361.
- Arora, J. S. 2004.** Introduction to Optimum Design. Second Edition. Elsevier Academic Press, San Diego. ISBN: 0-12-064155-0. 728p.
- Babu, B.V., Chaturvedi, G. 2000a.** Evolutionary Computation Strategy for Optimization of an Alkylation Reaction. Int. Symp. 53rd Annual Session IChE, December 18–21, Science City, Calcutta.

Babu, B.V., Munawar, S.A. 2000b. Differential Evolution for the Optimal Design of Heat Exchangers. All India Seminar Chemical Engineering Progress Resource Development: A Vision 2010 Beyond, March 13, Orissa State Centre, Bhubaneswar.

Babu, B.V., Angira, R. 2003a. Optimization of Water Pumping System Using Differential Evolution Strategies. 2nd Int. Conf. Computational Intelligence Robotics Autonomous Systems, December 15–18, Singapore.

Babu, B.V., Angira, R. 2003b. New Strategies of Differential Evolution for Optimization of Extraction Process. International Symposium 56th Annual Session IChE, December 19–22, Bhubaneswar.

Babu, B.V., Jehan, M.M.L. 2003c. Differential Evolution for Multi-Objective Optimisation. 2003 Congress Evolutionary Computation, Canberra, Australia, December 8–12, pp. 2696–2703.

Babu, B.V., Angira, R. 2004. Optimization Using Hybrid Differential Evolution Algorithms. Proceedings of International Symposium & 57th Annual Session of the Indian Institute of Chemical Engineers in association with AIChE (CHEMCON-2004), December 27–30, Mumbai, India.

Back, T., Hammel, U., Schwefel, H.P. 1997. Evolutionary Computation: Comments on the History and Current State. *IEEE Transactions on Evolutionary Computation*, (April 1997), 1(1): 3-17.

Back, T., Schwefel, H.P. 1993. An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, 1(1): 1–23.

Bagley, J.D. 1967. The Behavior of Adaptive Systems Which Employ Genetic and Correlation Algorithms. *Ph.D. Thesis*, University of Michigan.

Balendra, R., Qin, Y., Lu, X. 2000. Analysis, Evaluation and Compensation of Component-Errors in the Nett-Forming of Engineering Components, *Journal of Materials Processing Technology* 106: 204-211.

Balsa-Canto, E., Alonso, A.A., Banga, J.R. 1998. Dynamic Optimization of Bioprocesses: Deterministic and Stochastic Strategies. Automatic Control of Food and Biological Processes, September 21– 23, Gothenburg, Sweden.

Banga, J.R., Balsa-Canto, E., Moles, C.G., ALONSO, A.A. 2003. Improving Food Processing Using Modern Optimization Methods. *Trends in Food Science and Technology*, 14(4): 131–144.

Becchio, E., Chiara, A., Fileccia, R., Mastrococo, M. 1998. Use of FEM in the Drawing Die Structure Design. SAE Technical Papers, UTS SPA, COMAU Sheet Metal Dies.

Becerra, R.L., Coello, C.C. 2004a. A Cultural Algorithm with Differential Evolution to Solve Constrained Optimization Problems. *Advances in Artificial Intelligence – 9th*

Ibero-American Conference on Artificial Intelligence, IBERAMIA 2004, November 22-26, Puebla, Mexico.

Becerra, R.L., Coello, C.C. 2004b. Culturizing Differential Evolution for Constrained Optimization. IEEE Proceedings of the Fifth Mexican International Conference in Computer Science (ENC'04), 20-24 September, Colima, México.

Becerra, R.L., Coello, C.C. 2006a. Cultured Differential Evolution for Constrained Optimization. *Computer Methods in Applied Mechanics and Engineering*, 195(33-36): 4303-4322.

Becerra, R.L., Coello, C.C. 2006b. Solving Hard Multi-Objective Optimization Problems using ϵ -Constraint with Cultured Differential Evolution. The 9th International Conference on Parallel Problem Solving from Nature, Vol. 4193, pp. 543-552.

Bentley, P.J., Corne, D.W. 2002. Creative Evolutionary Systems. Morgan Kaufmann Publishers-Academic Press, San Diego. ISBN: 1-55860-673-4. 577p.

Beynon, M.J. 2005. Optimizing Object Classification under Ambiguity/Ignorance: Application to the Credit Rating Problem. *International Journal of Intelligent Systems in Accounting, Finance and Management*, 13(2): 113-130.

Bhat, T.R., Venkataramani, D., Ravi, V., Murty, C.V.S. 2006. An Improved Differential Evolution Method for Efficient Parameter Estimation in Biofilter Modeling. *Biochemical Engineering Journal*, 28(2): 16-176.

Bluszcz, A., Adamiec, G. 2006. Application of Differential Evolution to Fitting OSL Decay Curves. *Radiation Measurements*, 41: 886-891.

Box, G.E.P., Wilson, K.B. 1951. On the Experimental Attainment of Optimum Conditions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 13(1): 1-45.

Box, G.E.P., Draper, N.R. 2007. Response Surfaces, Mixtures, and Ridge Analyses. 2nd Edition. Hoboken, New Jersey, 857 pp.

Brest, J., Boškovic, B., Greiner, S., Žumer, V., Maucec, M. 2007. Performance Comparison of Self-Adaptive and Adaptive Differential Evolution Algorithms. *Soft Computing*, 11: 617-629.

British Standard 6841. 1987. Measurement and Evaluation of Human Exposure to Whole Body Vibration.

Brutovski, B., Ulicny, J., Miskovsky, P. 1995. Application of Genetic Algorithms Based Techniques in the Theoretical Analysis of Molecular Vibrations. 1st International Conference Genetic Algorithms Occasion, 130th Anniversary Mendel's Laws in Brno, Brno, Czech Republic, September 26-28, 29-33 pp.

Cannon, A.J. 2007. Nonlinear Analog Predictor Analysis: A Coupled Neural Network/Analog Model for Climate Downscaling. *Neural Networks*, 20(4): 444-453.

Carrol, C.W. 1961. The created response surface technique for optimizing nonlinear restrained systems. *Operations Research*, 9: 169–184

Chang, C.S., Xu, D.Y. 2000. Differential Evolution Based Tuning of Fuzzy Automatic Train Operation for Mass Rapid Transit System. *IEEE Proceedings B-Electric Power Applications*, 147(3): 206–212.

Chang, Y.P., Wu, C.J. 2004. Design of Harmonic Filters Using Combined Feasible Direction Method and Differential Evolution. *International Conference on Power System Technology - POWERCON 2004*, Singapore, 21-24 November.

Chen, C.W., Chen, D.Z., Cao, G.Z. 2002. An Improved Differential Evolution Algorithm in Training and Encoding Prior Knowledge into Feedforward Networks with Application in Chemistry. *Chemometrics and Intelligent Laboratory Systems*, 64(1): 27–43.

Chiou, J.P., Wang, F.S. 1998. A Hybrid Method of Differential Evolution with Application to Optimal Control Problems of a Bioprocess System. *IEEE Congress Computational Intelligence*, Anchorage, Alaska, May 4–9, pp. 627–632.

Chiou, J., Chang, C., Su, C. 2004. Ant Direction Hybrid Differential Evolution for Solving Large Capacitor Placement Problems. *IEEE Transactions on Power Systems*, 19(4): 1794–1800.

Chiou, J.P., Chang, C.F., Su, C.T. 2006. Capacitor Placement in Large-Scale Distribution Systems Using Variable Scaling Hybrid Differential Evolution. *International Journal of Electrical Power Energy Systems*, 28(10): 739–745.

Chodnikiewicz, K., Balendrab, R. 2000. The Calibration of Metal-Forming Presses. *Journal of Materials Processing Technology*, 106: 28–33.

Chong, S.Y., Tremayne, M. 2006. Combined Optimization using Cultural and Differential Evolution: Application to Crystal Structure Solution from Powder Diffraction Data. *Chemical Communications*, 2006: 4078–4080.

Coelho, L.S., Mariani, V.C. 2006a. Combining of Chaotic Differential Evolution and Quadratic Programming for Economic Dispatch Optimization With Valve-Point Effect. *IEEE Transactions on Power Systems*, 21(2): 989.

Coelho, L.S., Mariani, V.C. 2006b. Combining of Differential Evolution and Implicit Filtering Algorithm Applied to Electromagnetic Design Optimization. 11th Online Conference on Soft Computing in Industrial Applications.

Coelho, L.S., Mariani, V.C. 2006c. Chaotic Differential Evolution Applied to Electromagnetics Optimisation. 12th Biennial IEEE Conference Electromagnetic Field Computation, April 30–May 3, p. 230, Miami.

Coello, C.A.C. 1996. An Empirical Study of Evolutionary Techniques for Multiobjective Optimization in Engineering Design. *Ph.D. Thesis*, Department of Computer Science, Tulane University, New Orleans, LA, USA.

- Coello, C.A.C., Christiansen, A.D. 1998.** Two New GA-Based Methods for Multiobjective Optimization. *Civil Engineering and Environmental Systems*, 15(3): 207-243.
- Coello, C.A.C. 1999.** Self-adaptive Penalties for GA Based Optimization. Proceedings Congress of Evolutionary Computation, 1: 573–580.
- Coello, C.A.C. 2000.** Use of a Self-Adaptive Penalty Approach for Engineering Optimization Problems. *Computers in Industry*, 41: 113-127.
- Coello, C.A.C., Montes, E. 2002.** Constraint-Handling in Genetic Algorithms Through the Use of Dominance-Based Tournament Selection. *Advanced Engineering Informatics*, 16: 193–203.
- Colaço, M.J., Dulikravich, G.S., Martin, T.J. 2003a.** Reducing Convection Effects in Solidification By Applying Magnetic Fields Having Optimized Intensity Distribution. ASME paper HT2003-47308, ASME Summer Heat Transfer Conference, July 21-23, Las Vegas, Nevada.
- Colaço, M.J., Dulikravich, G.S., Martin, T.J. 2003b.** Optimization of Wall Electrodes for Electro-Hydrodynamic Control of Natural Convection during Solidification. Mechanical Engineering Congress and Exhibition (IMECE'03), November 16–21, as American Society of Mechanical Engineers (ASME) paper IMECE 2003–41703, Washington, D. C., USA.
- Colaço, M.J., Dulikravich, G.S., Orlande, H.R.B., Rodrigues, F.A. 2003c.** A Comparison of Two Solution Techniques for the Inverse Problem of Simultaneously Estimating the Spatial Variations of Diffusion Coefficients and Source Terms. American Society of Mechanical Engineers, *Heat Transfer Division*, (Publication) HTD, 374(1): 221 –230.
- Colaço, M.J., Dulikravich, G.S., Martin, T.J. 2004.** Optimization of Wall Electrodes for Electro-Hydrodynamic Control of Natural Convection During Solidification. *Materials and Manufacturing Processes*, 19(4): 719 –736.
- Conle, E.A., Wang, Z.X. 2007.** Stamping Die Strain Measurements. SAE Technical Papers, Ford Research Center.
- Crutchley, D.A., Zwolinski, M. 2002.** Using Evolutionary and Hybrid Algorithms for DC Operating Point Analysis of Nonlinear Circuits. IEEE World Congress on Computational Intelligence - Congress on Evolutionary Computation. pp. 753-758.
- Davendra, D., Onwubolu, G. 2007.** Enhanced Differential Evolution HYBRID Scatter Search for Discrete Optimisation. Proceeding of the IEEE Congress on Evolutionary Computation, Sept 25-28, Singapore. pp. 1156–1162.
- Das, S., Konar, A., Chakraborty, U.K. 2005a.** Two Improved Differential Evolution Schemes for Faster Global Search, Genetic & Evolutionary Computation Conference (GECCO-2005), pp. 991-998.

- Das, S., Konar, A., Chakraborty, U.K. 2005b.** Improving Particle Swarm Optimization with Differentially Perturbed Velocity. Genetic & Evolutionary Computation Conference (GECCO-2005), Washington, D. C., pp. 177-184.
- Das, S., Konar, A., Chakraborty, U.K. 2005c.** Design of Two-Dimensional Digital IIR Filters with Selforganizing Hierarchical PSO. Proceedings Frontiers of Evolutionary Algorithms.
- Das, S., Konar, A., Chakraborty, U.K. 2005d.** Improved Differential Evolution Algorithms for Handling Noisy Optimization Problems. 2005 IEEE Congress Evolutionary Computation, Edinburgh, September 2– 5, vol. 2, pp. 1691–1698.
- De, D., Ray, S., Konar, A., Chatterjee, A. 2005a.** An Evolutionary SPDE Breeding-Based Hybrid Particle Swarm Optimizer: Application in Coordination of Robot Ants for Camera Coverage Area Optimization. First International Conference on Pattern Recognition and Machine Intelligence (PReMI'05), Pages: 413-416.
- De, D., Ray, S., Konar, A., Chatterjee, A. 2005b.** A Fuzzy Logic Controller Based Dynamic Routing Algorithm with SPDE Based Differential Evolution. Genetic and Evolutionary Computation Conference (GECCO2005).
- Dean, A., Voss, D. 1999.** Design and Analysis of Experiments. Springer-Verlag, New York, 740 pp.
- Deb, K. 1991.** Optimal Design of a Welded Beam via Genetic Algorithms. *AIAA Journal*, 29(11): 2013–2015.
- Deb, K. 1997.** GeneAS: A Robust Optimal Design Technique for Mechanical Component Design: Evolutionary Algorithms in Engineering Applications, Ed.: Dasgupta, D., Michalewicz, Z., Springer, Berlin, pp: 497–514.
- Deb, K., Goyal, M. 1998.** A Flexible Optimization Procedure for Mechanical Component Design Based on Genetic Adaptive Search. *Journal of Mechanical Design*, 120(2): 162–164.
- Deb, K., 1999a.** An Introduction to Genetic Algorithms. *Sadhana Journal*. 24(4-5): 293-315.
- Deb, K., 1999b.** Evolutionary Algorithms for Multi-Criterion Optimization in Engineering Design: Evolutionary Algorithms in Engineering and Computer Science, Ed.: Miettinen K., Neittaanmäki P., Mäkelä M.M., Périaux J., John Wiley & Sons, Ltd., West Sussex, England, pp: 135-161.
- Deb, K., Pratap, A., Moitra, S. 2000.** Mechanical Component Design for Multiple Objectives Using Elitist Non-Dominated Sorting GA. Proceedings of the Parallel Problem Solving from Nature VI Conference, Pages: 859-868.
- Deb, K. 2001.** Nonlinear Goal Programming Using Multi-Objective Genetic Algorithms. *Journal of the Operational Research Society*, 52: 291–302.

- Demic, M., Lukic, J. 2002.** Some Aspects of the Investigation of Random Vibration Influence on Ride Comfort. *Journal of Sound and Vibration*, 253(1): 109–129.
- Derksen, R.W., Hancox, E. 2005.** The Optimization of an Industrial Pneumatic Supply System. 9th Int. Conf. Computer Aided Optimum Design Engineering, May 23–25, Skiathos, Greece, pp. 413–422.
- Dhavlikar, M.N., Kulkarni, M.S., Mariappan, V. 2003.** Combined Taguchi and Dual Response Method for Optimization of a Centerless Grinding Operation. *Journal of Materials Processing Technology*, 132 (2003): 90-94. Elsevier Science.
- Du, M., Guan, L. 2006.** Monocular Human Motion Tracking With the De-Mc Particle Filter. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2006 Proceedings, Volume 5.
- Eiben, A.E., Hinterding, R., Michalewicz, Z. 1999.** Parameter Control in Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2): 124-141.
- Els, P.S., Uys, P.E., Snyman, J.A., Thoresson, M.J. 2006.** Gradient-Based Approximation Methods Applied to the Optimal Design of Vehicle Suspension Systems Using Computational Models with Severe Inherent Noise. *Mathematical and Computer Modelling*, 43: 787-801.
- Els, P.S., Theron, N.J., Uys, P.E., Thoresson, M.J. 2007.** The Ride Comfort vs. Handling Compromise for Off-Road Vehicles, *Journal of Terramechanics*, 44: 303–317.
- Eriksson, P., Arora, J.S. 2002.** A Comparison of Global Optimization Algorithms Applied to a Ride Comfort Optimization Problem, *Structural Multidisciplinary Optimization*, 24: 157–167.
- Fan, H.Y., Lampinen, J. 2003.** A Trigonometric Mutation Operation to Differential Evolution. *Journal of Global Optimization*, 27: 105–129.
- Farina, M., Cesario, N., Ruggiero, D., Amato, P. 2003.** Evolutionary Optimization of Parametric Models: The Test Case of Combustion in a Diesel Engine. Soft Computing: Methodologies and Applications, *Advances in Soft Computing*, 163-176.
- Feoktistov, V., Janaqi, S. 2004.** New Energetic Selection Principle in Differential Evolution. 6th International Conf. Enterprise Information Systems, April 14–17, pp. 151–157, Porto, Portugal.
- Fisher, R.A. 1925.** Statistical Methods for Research Workers. Oliver and Boyd, Edinburg, UK, 237 pp.
- Fogel, L.J., Owens, A.J., Walsh, M.J. 1966.** Artificial Intelligence through Simulated Evolution. John Wiley & Sons, Inc., New York.

- Ganchev, T., Fakotakis, N., Tasoulis, D.K., Vrahatis, M.N. 2004.** Generalized Locally Recurrent Probabilistic Neural Networks for Text-Independent Speaker Verification. *IEEE Int. Conf. Acoustics Speech Signal Processing*, Montreal, Que, Canada, May 17– 21, vol. 1, pp. 141–144.
- Gent, A.N. 2000.** *Engineering with Rubber*, Hanser Publishers, Munich.
- Genta, G., Morello, L. 2009.** *The Automotive Chassis Vol. 2: System Design*. Springer, 832 pp.
- Ghani, J.A., Choudhury, I.A., Hassan, H.H. 2004.** Application of Taguchi Method in the Optimization of End Milling Parameters. *Journal of Materials Processing Technology*, 145(2004): 84-92.
- Goldberg, D.E. 1989.** *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Gunawan, S., Azarm, S. 2005.** Multi-objective Robust Optimization Using a Sensitivity Region Concept. *Structural and Multidisciplinary Optimization*, 29(1): 50-60.
- Guo, Z.Y., Kang, L.Y., Cheng, B., Ye, M., Cao, B.G. 2006.** Chaos Differential Evolution Algorithm With Dynamically Changing Weighting Factor And Crossover Factor. *Journal of Harbin Engineering University*, 27: 523–526.
- Haslinger, J., Jedelsky, D., Kozubek, T., Tvrđík, J. 2000.** Genetic and Random Search Methods in Optimal Shape Design Problems. *Journal of Global Optimization*, 16: 109-131.
- He, R., Narayana, P.A. 2002.** Automatic Delineation of Gd Enhancements on Magnetic Resonance Images in Multiple Sclerosis. *Medical Phys.*, 29(7): 1536-1546.
- He, X., Han, L. 2007.** A Novel Binary Differential Evolution Algorithm Based On Artificial Immune System. *Proceedings of the IEEE congress on Evolutionary Computation*, p. 2267-2272.
- Hendtlass, T. 2001.** A Combined Swarm Differential Evolution Algorithm for Optimization Problems. *Fourteenth International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Lecture Notes in Computer Science*, 2070: 11–18.
- Hernández-Díaz, A.G., Santana-Quintero, L.V., Coello, C.C., Caballero, R., Molina, J. 2006.** A New Proposal for Multi-Objective Optimization using Differential Evolution and Rough Sets Theory. *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*, Pages: 675-682, Seattle, Washington, USA.
- Hooke, R., Jeeves, T.A. 1961.** Direct Search Solution of Numerical and Statistical Problems. *Journal of the Association for Computing Machinery (JACM)*, 8(2): 212-229, ISSN: 0004-5411.

- Hou, B., Wang, W., Li, S., Lin, Z., Cedric, Z.X. 2010.** Stochastic Analysis and Robust Optimization for a Deck Lid Inner Panel Stamping. *Material Design*, 31: 1191-1199.
- Hu, C.P., Yan, X.F. 2009.** An Immune Self-adaptive Differential Evolution Algorithm with Application to Estimate Kinetic Parameters for Homogeneous Mercury Oxidation. *Process Systems Engineering, Chinese Journal of Chemical Engineering*. 17(2): 232-240.
- Huang, F., Wang, L., Liu, B. 2006.** Improved Differential Evolution with Dynamic Population Size. *Intelligent Computing: International Conference on Intelligent Computing, ICIC 2006*, August 16–19, Kunming, China.
- Ingber, L. 1963.** Simulated Annealing: Practice Versus Theory. *Journal of Mathematical and Computer Modeling*, 18: 29-57.
- Ishihama, M. 2003.** Sound Quality R&D in the Japanese Automotive Industry. *Noise Control Engineering Journal*, 51: 191-194.
- ISO Standard 2631 – 1. 1997.** Mechanical Vibration and Shock – Evaluation of Human Exposure to Whole Body Vibration. Part 1: General Requirements.
- Jang, I.G., Kwag, B.M. 2006.** Evolutionary Topology Optimization using Design Space Adjustment based on Fixed Grid. *International Journal for Numerical Methods in Engineering*, 66: 1817-1840, Wiley InterScience.
- Johnson, I.R., Kinghorn, B.P., Murphy, S.R., Lodge, G.M., Meszaros, S.A. 2002.** Estimating Soil Physical Parameters Using Simulation Differential Evolution. *IASTED Int. Conf. Applied Simulation Modelling*, Crete, Greece, June 25–28, pp. 274–279.
- Joshi, R., Sanderson, A.C. 1996.** Multisensor Fusion And Model Selection Using A Minimal Representation Size Framework. *IEEE/SICE/RSJ Int. Conf. Multisensor Fusion Integration Intelligent Systems*, Washington, DC, December 8–11, 1996, pp. 25–32.
- Kaelo, P., Ali, M.M. 2007.** Differential Evolution Algorithms using Hybrid Mutation. *Computational Optimization and Applications*, 37(2): 231-246.
- Kannan, B.K., Kramer, S.N. 1994.** An Augmented Lagrange Multiplier Based Method for Mixed Integer Discrete Continuous Optimization and Its Applications to Mechanical Design. *Journal of Mechanical Design*, 116(2): 405–411.
- Kannan, S., Slochanal, M.R., Subbaraj, P., Padhy, N.P. 2004.** Application of Particle Swarm Optimization Technique and Its Variants to Generation Expansion Planning Problem. *Electric Power Systems Research*, 70(2): 203–210.
- Karasalo, I., Skogqvist, P. 2005.** Object Identification by Bistatic Acoustic Scattering. *IEEE Oceans 2005 Europe International Conference Brest, FRANCE, JUN 20-23.*

Karen, İ., Yıldız, A.R., Kaya, N., Öztürk, F. 2004a. Multi Objective Production Design with Genetic Algorithms. *Advanced Technologies and Applications in Aeronautics, HİTEK 2004*, 9-10 December 2004, İstanbul, Türkiye.

Karen, İ., Yıldız, A.R., Kaya, N., Öztürk, F. 2004b. Evaluation of Multi-Objective Genetic Algorithm Optimization Program using Test Problems. *Intelligent Manufacturing Congress, IMS04*, 06-08 September 2004, Sakarya, Turkey.

Karen, İ. 2005. Bilgisayar Destekli Yapısal ve Şekil Optimizasyonu için Algoritma Geliştirilmesi. Yüksek Lisans Tezi, Uludağ Üniversitesi Fen Bilimleri Enstitüsü, Bursa, s. 24-37.

Karen, İ., Yıldız, A.R., Kaya, N., Öztürk, N., Öztürk, F. 2006. Hybrid Approach for Genetic Algorithm and Taguchi's Method Based Design Optimization in the Automotive Industry. *International Journal of Production Research*, 44(22): 4897-4914.

Karen, İ., Kaya, N., Öztürk, F. 2008a. Genetic and Simulation Based Approach to Multi-objective Optimum Shape Design, 4. Otomotiv Teknolojileri Kongresi, OTEKON'08, 01-04 Haziran 2008, Bursa, Türkiye.

Karen, İ., Kaya, N., Öztürk, F., Korkmaz, İ. 2008b. Motor Titreşim Takozlarının İstenen Özelliklerde Tasarımı ve Doğrulaması, 4. Otomotiv Teknolojileri Kongresi, OTEKON'08, 01-04 Haziran 2008, Bursa, Türkiye.

Karen, İ., Kaya, N., Öztürk, F., Kaftanoğlu, B., Mitmit, C., Polat, E. 2009. Otomotiv Endüstrisinde Sac Metal Kalıp Tasarımı ve Topoloji Tabanlı Optimizasyonu, 11. Otomotiv Sempozyumu, TMMOB Makine Mühendisleri Odası, 08-09 Mayıs 2009, Bursa, Türkiye.

Karen, İ., Kaya, N., Kaya, Öztürk, F., Korkmaz, İ., Yıldızhan, M., Yurttas, A. 2011. A Design Tool to Evaluate the Vehicle Ride Comfort Characteristics: Modeling, Physical Testing and Analysis. *International Journal of Advanced Manufacturing Technology*, (DOI 10.1007/s00170-011-3592-z), 2011.

Kaya, N., Karen, İ., Öztürk, F. 2010. Redesign of a Failed Clutch Fork using Topology and Shape Optimisation by the Response Surface Method. *Materials & Design*, 31(6): 3008-3014.

Kegl, M. 2005. Parameterization Based Shape Optimization: Theory and Practical Implementation Aspects. *International Journal for Computer-Aided Engineering and Software*, 22(5/6): 646-663, Emerald.

Kim, J.J., Kim, H.Y. 1997. Shape Design of an Engine Mount by a Method of Parameter Optimization. *Computers & Structures*, 65(5): 725-731.

Kim, N.H., Choi, K.K., Chen, J.S. 2001. Die Shape Design Optimization of Sheet Metal Stamping Process using Meshfree Method. *International Journal of Numerical Methods in Engineering*, 51(12): 1385-1405.

- Kim, Y.G., Park, C.K., Hwang, H.S., Park, T.W. 2003.** Design Optimization For Suspension System of High Speed Train Using Neural Network. *JSME International Journal Series C-Mechanical Systems Machine Elements and Manufacturing*, 46(2): 727-735.
- Kim, D.K., Kim, W.S., Woo, C.S., Lee, H.J. 2004.** Some Considerations on Mechanical Testing Methods of Rubbery Materials using Nonlinear Finite Element Analysis. *Polymer International*, 53: 850-856.
- Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P. 1983.** Optimization by Simulated Annealing. *Science, New Series*, 220, 4598, 671-680, Jstor.
- Kirsch, U. 1981.** Optimal Structural Design, McGraw-Hill, New York.
- Kok, S., Stander, N. 1999.** Optimization of a Sheet Metal Forming Process using Successive Multipoint Approximations. *Structural Optimization* 1999(18/4): 277-295.
- Kong, X., Xu, W., Liu, J. 2006.** A Permutation-Based Differential Evolution Algorithm Incorporating Simulated Annealing for Multiprocessor Scheduling with Communication Delays. *International Conference on Computational Science*, 1(2006): 514-521.
- Korošec, P., Šilc, J., Oblak, K., Kosel, F. 2007.** Optimizing the Shape of an Impeller using the Differential Ant-Stigmergy Algorithm. *Proceedings of the 7th International Conference on Parallel Processing and Applied Mathematics Table of Contents*, Gdansk, Poland, Pages: 520-529.
- Krink, T., Paterlini, S., Resti, A. 2007.** Using Differential Evolution to Improve the Accuracy of Bank Rating Systems. *Computational Statistics & Data Analysis*, 52: 68-87.
- Kukkonen, S., Lampinen, J. 2004.** An Extension of Generalized Differential Evolution for Multi-Objective Optimization with Constraints, in *Parallel Problem Solving from Nature – PPSN VIII* (eds X. Yao, E. Burke, J. A. Lozano et al.) **Lecture Notes in Computer Science**, 3242, Springer, Berlin, pp. 752–761.
- Kyprianou, A., Giacomini, J., Worden, K., Heidrich, M., Bocking, J. 2000.** Differential Evolution Based Identification of Automotive Hydraulic Engine Mount Model Parameters. *Proceedings of The Institution of Mechanical Engineers Part D- Journal of Automobile Engineering*, 214(D3): 249-264.
- Lampinen, J., Zelinka, I. 1999.** Mechanical engineering design optimization by differential evolution: New ideas in optimisation, Ed.: Corne, D., Dorigo, M., Glover, F., McGraw-Hill Inc., US.
- Lampinen, J. 2001.** Multi-Constrained Nonlinear Optimization by the Differential Evolution Algorithm. *6th Online World Conf. Soft Computing Industrial Applications*, September 10–24.

- Lampinen, J. 2002.** A Constraint Handling Approach for the Differential Evolution Algorithm. Proceedings of the 2002 Congress on Evolutionary Computation (CEC'02), 2: 1468–1473.
- Lampinen, J. 2003.** Cam Shape Optimization by Genetic Algorithm. *Computer Aided Design*, 35: 727-737.
- Laskari, E.C., Parsopoulos, K.E., Vrahatis, M.N. 2003.** Evolutionary Operators in Global Optimization with Dynamic Search Trajectories. *Numerical Algorithms*, 34(2-4): 393-403, Springer.
- Lee, D.H., Hwang, W.S. 2003.** Parametric Optimization of Complex Systems using a Multi-Domain FRF-Based Substructuring Method. *Computers and Structures*, 81: 2249–2257.
- Lee, W.S., Youn, S.K. 2004.** Topology Optimization of Rubber Isolators Considering Static and Dynamic Behaviours. *Structural Multidisciplinary Optimization*, 27: 284-294.
- Lee, C.T., Moon, B.Y. 2006.** Simulation and Experimental Validation of Vehicle Dynamic Characteristics for Displacement-Sensitive Shock Absorber Using Fluid-Flow Modeling. *Mechanical Systems and Signal Processing*, 20: 373–388.
- Li, Y., Rao, L., He, R., Xu, G., Wu, Q., Yan, W., Dong, G., Yang, Q. 2005.** A Novel Combination Method of Electrical Impedance Tomography Inverse Problem for Brain Imaging. *IEEE Transactions on Magnetics*, 41(5): 1848-1851.
- Li, J., Zhu, S. Lu, B. 2010.** Design and Optimization of Phase Plates for Shaping Partially Coherent Beams by Adaptive Genetic Algorithms. *Optics & Laser Technology*, 42: 317–321.
- Liao, H.C., Chen, Y.K. 2002.** Optimizing Multi-Response Problem in the Taguchi Method by DEA Based Ranking Method. *International Journal of Quality & Reliability Management*, 19(7): 825-837, Emerald.
- Liew, K.M., Tan, H., Ray, T., Tan, M.J. 2004.** Optimal Process Design of Sheet Metal Forming for Minimum Springback via an Integrated Neural Network Evolutionary Algorithm. *Structural Multidisciplinary Optimization*, 26: 284-294.
- Lii, G.R., Chiang, C.L., Su, C.T., Hwung, H.R. 2003.** An Induction Motor Position Controller Optimally Designed with Fuzzy Phase-Plane Control and Genetic Algorithms. *Electric Power Systems Research*, 68(2): 103-112.
- Lingyun, W., Mei, Z., Guangming, W., Guang, M. 2005.** Truss Optimization on Shape and Sizing with Frequency Constraints Based on Genetic Algorithm. *Computational Mechanics*, 35: 361-368, Springer-Verlag.
- Liu, J., Lampinen, J. 2002.** Adaptive Parameter Control of Differential Evolution. 8th Int. MENDEL Conf. Soft Computing, Brno, Czech Republic, June 5–7, pp. 19–26.

- Liu, J., Lampinen, J. 2005.** A Fuzzy Adaptive Differential Evolution Algorithm. *Soft Computing*, 9(6): 448-462, Springer.
- Liu, B., Wang, L., Jin, Y.H., Tang, F., Huang, D.X. 2005.** Improved Particle Swarm Optimization Combined with Chaos. *Chaos, Solitons and Fractals* 25(2005): 1261–1271.
- Lopez, C., Willigenburg, L.G., Straten, G. 2003.** Optimal Control of Nitrate in Lettuce by A Hybrid Approach: Differential Evolution And Adjustable Control Weight Gradient Algorithms, *Computers and Electronics in Agriculture* 40: 179–197.
- Lorenzoni, L.L., Ahonen, H., De-Alvarenga, A.G. 2006.** A Multi-Mode Resource-Constrained Scheduling Problem in the Context of Port Operations. *Computers and Industrial Engineering*, 50(1–2): 55–65.
- Madavan, N.K. 2002.** Turbomachinery Airfoil Design Optimization using Differential Evolution. 2nd Int. Conf. Computational Fluid Dynamics, July 15–19 Sydney, Australia.
- Magoulas, G.D., Plagianakos, V.P., Vrahatis, M.N. 2001.** Hybrid Methods Using Evolutionary Algorithms for On-line Training. In Proc. of the IEEE International Joint Conference on Neural Networks (IJCNN'2001), Washington D. C.
- MATLAB. 2009.** Simulink-SimMechanics 2009b. The Mathworks, Inc.
- Mayer, D.G., Kinghorn, B.P., Archer, A.A. 2005.** Differential Evolution – an Easy and Efficient Evolutionary Algorithm for Model Optimisation. *Agricultural Systems*, 83(3): 315–328.
- Menon, P.P., Kim, J., Bates, D.G., Postlethwaite, I. 2006.** Clearance of Nonlinear Flight Control Laws using Hybrid Evolutionary Optimisation. *IEEE Transactions on Evolutionary Computation*, 10(6): 689-699.
- Menon, P.P., Bates, D.G., Postlethwaite, I. 2007.** Deterministic Versus Evolutionary Optimisation Methods For Nonlinear Robustness Analysis of Flight Control Laws. IEEE Congress on Evolutionary Computation, Pages: 1910-1917.
- Meske, R., Lauber, B., Schnack, E. 2006.** A New Optimality Criteria Method for Shape Optimization of Natural Frequency Problems. *Structural and Multidisciplinary Optimization*, 31: 295-310, Springer-Verlag.
- Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. 1953.** Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21: 6, 1087-1092.
- Michalski, K.A. 2001.** Electromagnetic Imaging of Elliptical-Cylindrical Conductors and Tunnels Using a Differential Evolution Algorithm. *Microwave and Optical Technology Letters*, 28(3): 164–169.

- Mingyong, L., Erbao, C. 2010.** An Improved Differential Evolution Algorithm for Vehicle Routing Problem with Simultaneous Pickups and Deliveries and Time Windows. *Engineering Applications of Artificial Intelligence*, 23: 188–195.
- Montes, E.M., Reyes, J.V., Coello, C.A.C. 2005.** Promising Infeasibility and Multiple Offspring Incorporated to Differential Evolution for Constrained Optimization. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2005), Volume 1, Pages: 225-232, New York, June 2005. Washington DC, USA, ACM Press.
- Montgomery, D.C. 2008.** Design and Analysis of Experiments, Seventh Edition. John Wiley & Sons Danvers, MA, USA, 680 pp.
- Moore, P.W., Venayagamoorthy, G.K. 2006.** Evolving Combinational Logic Circuits Using Particle Swarm, Differential Evolution and Hybrid DEPSO. *International Journal of Neural Systems* 16(2): 163-177.
- Nasimul, N., Hitoshi, I. 2008.** Differential Evolution for Economic Load Dispatch Problems. *Elsevier Electric Power Systems Research*, 78(8): 1322-1331.
- Nearchou, A.C. 2006.** Meta-Heuristics from Nature for the Loop Layout Design Problem. *International Journal of Production Economics*, 101(2): 312–328.
- Nearchou, A.C., Omirou, S.L. 2006.** Differential Evolution for Sequencing and Scheduling Optimization. *Journal of Heuristics*, 12(6): 395–411.
- Nelder, J.A., Mead, R. 1965.** A Simplex Method for Function Minimization. *Computer Journal*, 7: 308-313, Oxford Journals.
- Niekerk, J.L. van, Pielemeier, W.J., Greenberg, J.A. 2003.** The Use of Seat Effective Amplitude Transmissibility (SEAT) Values to Predict Dynamic Seat Comfort, *Journal of Sound and Vibration*, 260: 867-888.
- Nikolos, I.K., Brintaki, A.N. 2005.** Coordinated UAV Path Planning Using Differential Evolution. 20th IEEE Int. Symp. Intelligent Control, June 27– 29, pp. 549–556, Limassol, Cyprus.
- Nishiyama, S., Uesugi, N., Takeshima, T., Kano, Y., Togii, H. 2000.** Research on Vibration Characteristics Between Human Body and Seat, Steering Wheel, and Pedals (Effects of Seat Position on Ride Comfort). *Journal of Sound and Vibration*, 236(1): 1–21.
- Nobakhti, A., Wang, H. 2006.** Co-Evolutionary Self-Adaptive Differential Evolution with a Uniform Distribution Update Rule. IEEE Int. Symp. Intelligent Control, October 4–6, pp. 1264–1269, Munich, Germany,
- Noman, N., Iba, H. 2006.** A New Generation Alternation Model for Differential Evolution. *Genetic Evolutionary Computation*, Seattle, WA, July 8– 12, 2: 1265–1272.

- Obayashi, S., Sasaki, D., Takeguchi, Y., Hirose, N. 2000.** Multiobjective Evolutionary Computation for Supersonic Wing-Shape Optimization. *IEEE Transactions on Evolutionary Computation*, 4: 182-187.
- Ochi, N., Cao, Q.V. 2000.** Application of Differential Evolution Strategy in Modeling Forest Tree Growth. Proceedings of the Louisiana Academy Science, January 1.
- Pahner, U., Hameyer, K. 2000.** Adaptive Coupling of Differential Evolution and Multiquadrics Approximation for the Tuning of the Optimization Process. *IEEE Transactions on Magnetics*, 36(4): 1047–1051.
- Palli, N., Azram, S., McCluskey, P., Sundararajan, R. 1999.** An Interactive Multistage ϵ -Inequality Constraint Method for Multiple Objectives Decision Making. *ASME Journal of Mechanical Design*, 120(4): 678–686.
- Pareto, V. 1886.** Cours D'Economie Politique, Vols. 1 and 2. F. Rouge, Lausanne.
- Parsopoulos, K.E., Vrahatis, M.N. 2002.** Recent Approaches to Global Optimization Problems Through Particle Swarm Optimization. *Natural Computing*, 1(2-3): 235-306, Springer.
- Pavlidis, N.G., Vrahatis, M.N., Mossay, P. 2007.** Existence and Computation of Short-Run Equilibria in Economic Geography. *Applied Mathematics Computation*, 184(1): 93–103.
- Pedchote, C., Purdy, D.J. 2003.** Parameter Estimation of a Single Wheel Station Using Hybrid Differential Evolution. Proceedings of the Institution of Mechanical Engineers Part D-Journal of Automobile Engineering, 217(D6): 431-447.
- Pennati, M., Gobbi, M., Mastinu, G. 2009.** A Dummy for the Objective Ride Comfort Evaluation of Ground Vehicles. *Vehicle System Dynamics*, 47(3): 343–362.
- Phan, H.V., Lech, M., Nguyen, T.D. 2004.** Registration of 3D Range Images Using Particle Swarm Optimization. *Advances in Computer Science - ASIAN 2004*, 3321: 3290-3291.
- Pozo, D., Lacalle, L.N.L., López, J.M., Hernández, A. 2006.** Machining of Large Dies Based on the Prediction of the Press/Die Deformation. 2nd Innovative Production Machines and Systems Virtual Conference, IPROMS 06.
- Pozo, D., Lacalle, L.N.L., López, J.M., Hernández, A. 2007.** Prediction of Pres/Die Deformation for an Accurate Manufacturing of Drawing Dies. *International Journal of Advanced Manufacturing Technology*, 37(7-8): 649-656.
- Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P. 1992.** Numerical Recipes in C. Cambridge University Press, UK.
- Price, K.V., Storn, R.M. Lampinen, J.A. 2005.** Differential Evolution – A Practical Approach to Global Optimization. Springer–Verlag, Berlin. 538 pp.

- Rauh, J. 2003.** Virtual Development of Ride and Handling Characteristics for Advanced Passenger Cars, *Vehicle System Dynamics*, 40(1): 135-155.
- Qian, B., Wang, L., Huang, D.X., Wang, X. 2006.** Multi-objective Flow Shop Scheduling Using Differential Evolution. *Intelligent Computing in Signal Processing and Pattern Recognition*, Book Chapter, pp: 1125-1136, Springer.
- Qing, A. 2004.** Electromagnetic Inverse Scattering of Multiple Perfectly Conducting Cylinders by Differential Evolution Strategy with Individuals in Groups (GDES). *IEEE Transactions on Antennas and Propagation*, 52(5): 1223–1229.
- Qing, A. 2009.** *Differential Evolution: Fundamentals and Applications in Electrical Engineering*. John Wiley & Sons, Singapore. 418 p.
- Ragsdell, K.M., Phillips, D.T. 1976.** Optimal Design of a Class of Welded Structures Using Geometric Programming. *ASME Journal of Engineering for Industry*, 98: 1021–1025.
- Rao, M.D. 2003.** Recent Applications of Viscoelastic Damping for Noise Control in Automobiles and Commercial Airplanes. *Journal of Sound and Vibration*, 262: 457–474.
- Rao, S.S. 2009.** *Engineering Optimization: Theory and Practice*, Fourth Edition. John Wiley & Sons, Inc., Hoboken, New Jersey, USA, 813 pp.
- Ray, T., Liew, K.M., Saini, P. 2002.** An Intelligent Information Sharing Strategy within a Swarm for Unconstrained and Constrained Optimization Problems. *Soft Computing*, 6(1): 38-44.
- Rechenberg, I. 1973.** *Evolutionsstrategie*. Frommann-Holzboog, Stuttgart.
- Riley, M.J.W., Peachey, T., Abramson, D., Jenkins, K.W. 2011.** Multi-Objective Engineering Shape Optimization using Differential Evolution Interfaced to the Nimrod/O Tool. *IOP Conf. Series: Materials Science and Engineering*, 10: 012189.
- Rogalsky, T., Derksen, R.W. 2000.** Hybridization of Differential Evolution for Aerodynamic Design. In *Proceedings of the 8th Annual Conference of the Computational Fluid Dynamics Society Of Canada*, Pages: 729-736.
- Rosochowski, A. 2001.** Die Compensation Procedure to Negate Die Deflection and Component Springback. *Journal of Materials Processing Technology* 115: 187–191.
- Runarsson, T.P., Yao, X. 2000.** Stochastic Ranking for Constrained Evolutionary Optimization. *IEEE Transactions on Evolutionary Computation* 4(3): 284–294.
- SAE Standard J1013. 1992.** Measurement of Whole Body Vibration of the Seated Operator of Off-Highway Work Machines.
- SAE Standard J1490. 2011.** Measurement and Presentation of Truck Ride Vibrations.

- Sakai, T., Terada, M., Ono, S., Kamimura, N., Gielen, L., Mas, P. 2001.** Development Procedure for Interior Noise Performance by Virtual Vehicle Refinement Combining Experimental and Numerical Component Models. Society of Automotive Engineers, 2001-01-1538.
- Sandgren, E. 1988.** Nonlinear Integer and Discrete Programming in Mechanical Design. Proceedings of the ASME Design Technology Conference, Kissimee, FL, Pages: 95-105.
- Sarimveis, H., Nikolakopoulos, A. 2005.** A Line Up Evolutionary Algorithm for Solving Nonlinear Constrained Optimization Problems. *Computers and Operations Research*, 32(6): 1499–1514.
- Schmidt, H., Thierauf, G. 2002.** A Combined Heuristic Technique for the Optimal Design of Civil Engineering Structures Implicit Constraints and Nonlinear Deformations. The Sixth International Conference on Computational Structures Technology, Prague, Czech Republic, 4-6 September.
- Schmidt, H., Thierauf, G. 2005.** A Combined Heuristic Optimization Technique. *Advances in Engineering Software*, 36(1): 11-19.
- Schwefel, H. P. 1994.** Evolution and Optimum Seeking. Wiley, New York.
- Shoureshi, R., Knurek, T. 1996.** Automotive Applications of a Hybrid Active Noise and Vibration Control, *IEEE Control Systems*, 16(6): 72-78.
- Sharma, S. 2003.** Critical Comparison of Popular Hyper-Elastic Material Models in Design of Anti-Vibration Mounts for Automotive Industry through FEA. Constitutive Models for Rubber III by Busfield & Muhr (eds), pp. 161-167.
- Siddall, J.N. 1972.** Analytical Decision – Making in Engineering Design. Englewoods Cliffs, N.J., Prentice Hall.
- Simulia. 2008.** ABAQUS 6.8 CAE Software and Documentation.
- Song, C.K., Hwang, J.K., Lee, J.M., Hedrick, J.K. 2003.** Active Vibration Control for Structural-Acoustic Coupling System of 3-D Vehicle Cabin Model. *Journal of Sound and Vibration*, 267: 851-865.
- Srinivas, N., Deb, K. 1995.** Multi-Objective Optimization using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, 2: 221–248.
- Subramanian, K., Slochanal, S.M.R., Subramanian, B., Padhy, N.P. 2006.** Application and Comparison of Metaheuristic Techniques to Generation Expansion Planning Incorporating Both Bilateral and Multilateral Transactions. *International Journal of Emerging Electric Power Systems*, 6(1): 2.
- Sun, J., Zhang, Q., Tsang, E. 2005.** DE/EDA: New Evolutionary Algorithm for Global Optimisation. *Information Sciences* 169(2005): 249-262.

Sun, G., Li, G., Gong, Z., He, G., Li, Q. 2011. Radial Basis Functional Model for Multi-Objective Sheet Metal Forming Optimization. *Engineering Optimization*, 21June, iFirst: 1–16.

Storn, R., Price, K.V. 1995. Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical Report TR- 95-012, ICSI.

Storn, R. 1995a. Modeling and Optimization of PET-Redundancy Assignment for MPEG-Sequences. Technical Report. TR-95-018, International Computer Science Insitute, May.

Storn, R. 1995b. Differential Evolution Design of an IIR-Filter with Requirements for Magnitude and Group Delay. Technical Report TR-95-026, International Computer Science Insitute, June.

Storn, R. 1996a. Differential Evolution Design Of An IIR-Filter. 1996 IEEE International Conference of Evolutionary Computation, Nagoya, May 20– 22, pp. 268–273.

Storn, R. 1996b. On The Usage of Differential Evolution for Function Optimization. Biennial Conference of North American Fuzzy Information Processing Society, Berkeley, CA, 19– 22 June, pp. 519– 523.

Storn, R. 1996c. System Design by Constraint Adaptation and Differential Evolution. Technical Report TR-96-039, International Computer Science Institute, November.

Taguchi, G., Jugulum, R. 2002. The Mahalanobis - Taguchi Strategy, A Pattern Technology System. John Wiley & Sons, New York, 234 pp.

Talbi, H., Batouche, M. 2004. A Swarm Algorithm for Medical Image Registration Based on the Maximization of the Mutual Information, in the Proceeding of the Arab Conference on IT ACIT'2004, pp. 144-152, Constantine, December.

Tekkaya, A.E. 2000. State-of-the-art of Simulation of Sheet Metal Forming. *Journal of Material Processing Technology*, 2000(103): 14-22.

Teo, J. 2006. Exploring Dynamic Self-Adaptive Populations in Differential Evolution. *Soft Computing*, 10(8): 673-686, Springer.

Ter Braak, C.J. 2006. A Markov Chain Monte Carlo Version of the Genetic Algorithm Differential Evolution: Easy Bayesian Computing for Real Parameter Spaces. *Statistics and Computing*, 16(3): 239-249, September.

Thomsen, R. 2004. Multimodal Optimization Using Crowding-Based Differential Evolution. 2004 IEEE Congress Evolutionary Computation, Portland, OR, June 19– 23, vol. 2, pp. 1382–1389.

Topologica, 2011. CATOPO: Topology Optimization and FE Analysis in the CAD. http://www.topologicasolutions.com/software_catopo.html-(Erişim tarihi: 13.10.2011).

- Törn, A.A. 1977.** Cluster Analysis using Seed Points and Density-Determined Hyperspheres as an Aid to Global Optimization. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(8): 610-616, New York.
- Üler, G.F., Osama, A.M., Chang, S.K. 1995.** Design Optimization of Electrical Machines using Genetic Algorithms. *IEEE Transactions on Magnetics*, 31(3): 2008-2011.
- Vafaeseefat, A. 2009.** Dome Shape Optimization of Composite Pressure Vessels Based on Rational B-Spline Curve and Genetic Algorithm. *Appl Compos Mater*, 16: 321–330.
- Vesterstrom, J., Thomsen, R. 2004.** A Comparative Study of Differential Evolution, Particle Swarm Optimization, and Evolutionary Algorithms on Numerical Benchmark Problems. Proceedings of the Sixth Congress on Evolutionary Computation, IEEE Press, USA, 332-339.
- Wang, F.S., Jing, C.H., Tsao, G.T. 1998.** Fuzzy-Decision-Making Problems of Fuel Ethanol Production using a Genetically Engineered Yeast. *Industrial and Engineering Chemistry Research*, 37(8): 3434–3443.
- Wang, S.K., Liu, C.W., Chiou, J.P. 2006.** Ant Direction Hybrid Differential Evolution for Solving Economic Dispatch of Power System. IEEE International Conference on Systems, Man, and Cybernetics October 8-11, Taipei, Taiwan.
- Weber, T. 2005.** Cooperative Evolution — A New Algorithm for the Investigation of Disordered Structures via Monte Carlo Modelling. *Zeitschrift Fur Kristallographie*, 220(12): 1099-1107.
- Weber, T., Bürgi, H.B. 2002.** Determination and Refinement of Disordered Crystal Structures Using Evolutionary Algorithms in Combination with Monte Carlo Methods. *Acta Crystallographica Section A, Foundations of Crystallography*, Volume 58, Part 6 (November).
- Wei, L. Tang, T., Xie, X., Shen, W. 2011.** Truss Optimization on Shape and Sizing with Frequency Constraints Based on Parallel Genetic Algorithm. *Structural Multidisciplinary Optimization*, 43: 665–682.
- Wu, L.H., Wang, Y.N., Yuan, X.F., Shou, S.W. 2006.** Differential Evolution Algorithm with Adaptive Second Mutation. *Control and Decision*, 21(8): 898–902.
- Xing, G., Xue, J. 2007.** A Hybrid Method for Electromagnetic Propagated Resistivity Logging Data Inversion. *IEEE Transactions on Geoscience and Remote Sensing*, March, 45(3): 649.
- Xiyu, L., Mingxi, T., Frazer, J.H. 2003.** Shape Reconstruction by Genetic Algorithms and Artificial Neural Networks. *Engineering Computations*, 2: 129-151.
- Xu, X., Dony, R. 2004.** Differential Evolution with Powell's Direction Set Method in Medical Image Registration. IEEE International Symposium on Biomedical Imaging(ISBI'04), pp. 732–735.

- Xu, D., Li, S., Qian, F. 2007.** An Improved Differential Evolution Algorithm and its Application in Reaction Kinetic Parameters Estimation. Proceedings of the International Conference on Intelligent Systems and Knowledge Engineering (ISKE 2007), Book Series: Advances in Intelligent System Research, Pages: 1456-1456.
- Yalçın, İ.K., Gökmen, M. 2006.** Integrating Differential Evolution and Condensation Algorithms for License Plate Tracking. Proceedings of the 18th International Conference on Pattern Recognition (ICPR), Vol. 01, Pages: 658-661.
- Yan, J.Y., Ling, Q., Sun, D.M. 2006.** A Differential Evolution with Simulated Annealing Updating Method. International Conference on Machine Learning and Cybernetics, Dalian, China, pp. 2103-2106.
- Yang, W.H., Tarng, Y.S. 1998.** Design Optimization of Cutting Parameters for Turning Operations Based on the Taguchi Method. *Journal of Materials Processing Technology*, 84 (1998), 122-129. Elsevier Science.
- Yang, J.M., Horng, J.T., Lin, C.J., Kao, C.Y. 2001.** Optical Coating Designs Using the Family Competition Evolutionary Algorithm. *Evolutionary Computation*, 9(4): 421-443, December.
- Yan, X.F., Yu, J., Qian, F., Ding, J.W. 2006.** Kinetic Parameter Estimation of Oxidation in Supercritical Water Based on Modified Differential Evolution. *Huadong Ligong Daxue Xuebao/J. East China University Science Technology*, 32(1): 94–97 and 124.
- Yang, Y., Ren, W., Chen, L., Jiang, M., Yang, Y. 2009.** Study on Ride Comfort of Tractor with Tandem Suspension Based on Multi-Body System Dynamics. *Applied Mathematical Modelling*, 33: 11–33.
- Yao, X., Liu, Y. 1999.** Evolutionary Programming Made Faster. *IEEE Transactions on Evolutionary Computation*, 3(2): 82–102.
- Yıldız, A.R., Karen, İ., Kaya, N., Öztürk, F., 2004.** Optimal Design of Machine Components Using Evolutionary Genetic Algorithms: Welded Beam and Pressure Vessel. Proceedings of the First Cappadocia International Mechanical Engineering Symposium, Cappadocia Turkey, 708-711.
- Yokota, T., Taguchi, T., Gen, M. 1999.** A Solution Method for Optimal Cost Problem of Welded Beam by Using Genetic Algorithms. *Computers & Industrial Engineering*, 37(1999): 379-382.
- Yu, Y., Naganathan, N.G., Dukkupati, R.V. 2001.** A Literature Review of Automotive Vehicle Engine Mounting Systems. *Mechanism and Machine Theory*, 36: 123-142.
- Zaharie, D. 2002.** Critical Values For The Control Parameters Of Differential Evolution Algorithms. 8th Int. Mendel Conference Soft Computing, Brno, Czech Republic, June 5–7, pp. 62–67.

Zelinka, I. 2005. Investigation on Evolutionary Deterministic Chaos Control – Extended Study. 19th European Conf. Modelling Simulation, Riga, Latvia, June 1–4, pp. 51–58.

Zhang, W.J., Xie, X.F. 2003. DEPSO: Hybrid Particle Swarm with Differential Evolution Operator. IEEE International Conference on Systems, Man and Cybernetics, 4: 3816-3821.

Zhang, L.P., Lin, C., Niu, X. 2009. Optimization of Control Strategy for Plug-in Hybrid Electric Vehicle Based on Differential Evolution Algorithm. IEEE Asia-Pacific Power and Energy Engineering Conference (APPEEC), Vols 1-7: 3085-3089.

Zheng, X.X., Qian, F. 2006. A Modified Particle Swarm Optimization Algorithm. *Computer Engineering*, 32: 25-27.

Zhu, Q.Y., Qin, A.K., Suganthan, P.N., Huang, G.B. 2005. Evolutionary Extreme Learning Machine. *Pattern Recognition* 38(10): 1759-1763.

EKLER

1. Test Fonksiyonu (g01) (Montes ve ark. 2005)

Minimizasyon:

$$f(\mathbf{x}) = 5 \sum_{i=1}^4 x_i - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i$$

Kısıtlar:

$$g_1(\mathbf{x}) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0$$

$$g_2(\mathbf{x}) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0$$

$$g_3(\mathbf{x}) = 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0$$

$$g_4(\mathbf{x}) = -8x_1 + x_{10} \leq 0$$

$$g_5(\mathbf{x}) = -8x_2 + x_{11} \leq 0$$

$$g_6(\mathbf{x}) = -8x_3 + x_{12} \leq 0$$

$$g_7(\mathbf{x}) = -2x_4 - x_5 + x_{10} \leq 0$$

$$g_8(\mathbf{x}) = -2x_6 - x_7 + x_{11} \leq 0$$

$$g_9(\mathbf{x}) = -2x_8 - x_9 + x_{12} \leq 0$$

Sınır Kısıtları:

$$0 \leq x_i \leq 1 \quad (i = 1, \dots, 9)$$

$$0 \leq x_i \leq 100 \quad (i = 10, 11, 12)$$

$$0 \leq x_{13} \leq 1$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = (1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1)$$

$$f(\mathbf{x}^*) = -15$$

$g_1, g_2, g_3, g_7, g_8, g_9$ aktif kısıtlardır.

2. Test Fonksiyonu (g02)

Minimizasyon:

$$f(\mathbf{x}) = - \left| \frac{\sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i)}{\sqrt{\sum_{i=1}^n i x_i^2}} \right|$$

Kısıtlar:

$$g_1(\mathbf{x}) = 0,75 - \prod_{i=1}^n x_i \leq 0$$

$$g_2(\mathbf{x}) = \sum_{i=1}^n x_i - 7,5n \leq 0$$

Sınır Kısıtları:

$$n = 20$$

$$0 \leq x_i \leq 10, \quad (i = 1, \dots, n)$$

Literatürdeki En İyi Sonuçlar:

$$\begin{aligned} \mathbf{x}^* = & [3,16246061572185; 3,12833142812967; 3,09479212988791; \\ & 3,06145059523469; 3,02792915885555; 2,99382606701730; \\ & 2,95866871765285; 2,92184227312450; 0,49482511456933; \\ & 0,48835711005490; 0,48231642711865; 0,47664475092742; \\ & 0,47129550835493; 0,46623099264167; 0,46142004984199; \\ & 0,45683664767217; 0,45245876903267; 0,44826762241853; \\ & 0,44424700958760; 0,44038285956317] \end{aligned}$$

$$f^* = -0.80361910412559$$

g_1 0'a çok yakın değer almaktadır.

3. Test Fonksiyonu (g03)

Minimizasyon:

$$f(\mathbf{x}) = -(\sqrt{n})^n \prod_{i=1}^n x_i$$

Kısıtlar:

$$h(\mathbf{x}) = \sum_{i=1}^n x_i^2 - 1 = 0$$

Sınır Kısıtları:

$$n = 10$$

$$0 \leq x_i \leq 1, (i = 1, \dots, n)$$

Literatürdeki En İyi Sonuçlar:

$$x_i^* = 1/\sqrt{n} \quad (i = 1, \dots, n)$$

$$f(\mathbf{x}^*) = -1,00050010001000$$

4. Test Fonksiyonu (g04)

Minimizasyon:

$$f(\mathbf{x}) = 5,3578547x_3^2 + 0,8356891x_1x_5 + 37,293239x_1 - 40\,792,141$$

Kısıtlar:

$$g_1(\mathbf{x}) = 85,334407 + 0,0056858x_2x_5 + 0,0006262x_1x_4 - 0,0022053x_3x_5 \\ - 92 \leq 0$$

$$g_2(x) = -85,334407 - 0,0056858x_2x_5 - 0,0006262x_1x_4 + 0,0022053x_3x_5 \\ \leq 0$$

$$g_3(x) = 80,51249 + 0,0071317x_2x_5 + 0,0029955x_1x_2 + 0,0021813x_3^2 \\ - 110 \leq 0$$

$$g_4(x) = -80,51249 - 0,0071317x_2x_5 - 0,0029955x_1x_2 - 0,0021813x_3^2 \\ + 90 \leq 0$$

$$g_5(x) = 9,300961 + 0,0047026x_3x_5 + 0,0012547x_1x_3 + 0,0019085x_3x_4 \\ - 25 \leq 0$$

$$g_6(x) = -9,300961 - 0,0047026x_3x_5 - 0,0012547x_1x_3 - 0,0019085x_3x_4 \\ + 20 \leq 0$$

Sınır Kısıtları:

$$78 \leq x_1 \leq 102, 33 \leq x_2 \leq 45, \quad 27 \leq x_i \leq 45 \quad (i = 3, 4, 5)$$

Literatürdeki En İyi Sonuçlar:

$$x^* = (78; 33; 29,9952560256815985; 45; 36,7758129057882073)$$

$$f(x^*) = -30\,665,539$$

g_1 ve g_6 aktif kısıtlardır.

5. Test Fonksiyonu (g05)

Minimizasyon:

$$f(x) = 3x_1 + 0,000001x_1^3 + 2x_2 + (0,000002/3)x_2^3$$

Kısıtlar:

$$g_1(x) = -x_4 + x_3 - 0,55 \leq 0$$

$$g_2(\mathbf{x}) = -x_3 + x_4 - 0,55 \leq 0$$

$$h_3(\mathbf{x}) = 1000 \sin(-x_3 - 0,25) + 1000 \sin(-x_4 - 0,25) + 894,8 - x_1 = 0$$

$$h_4(\mathbf{x}) = 1000 \sin(x_3 - 0,25) + 1000 \sin(x_3 - x_4 - 0,25) + 894,8 - x_2 = 0$$

$$h_5(\mathbf{x}) = 1000 \sin(x_4 - 0,25) + 1000 \sin(x_4 - x_3 - 0,25) + 1\,294,8 = 0$$

Sınır Kısıtları:

$$0 \leq x_1, x_2 \leq 1200$$

$$-0,55 \leq x_3, x_4 \leq 0,55$$

Literatürdeki En İyi Sonuçlar:

\mathbf{x}^*

$$= [679,945148297028709; 1\,026,06697600004691; 0,118876369094410433;$$

$$-0,396233485215178266]$$

$$f(\mathbf{x}^*) = 5\,126,4967140071$$

6. Test Fonksiyonu (g06)

Minimizasyon:

$$f(\mathbf{x}) = (x_1 - 10)^3 + (x_2 - 20)^3$$

Kısıtlar:

$$g_1(\mathbf{x}) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0$$

$$g_2(\mathbf{x}) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82,81 \leq 0$$

Sınır Kısıtları:

$$13 \leq x_1 \leq 100 \text{ ve } 0 \leq x_2 \leq 100$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = [14.09500000000000064; 0,8429607892154795668]$$

$$f(\mathbf{x}^*) = -6\,961,81387558015$$

g_1 ve g_2 aktif kısıtlardır.

7. Test Fonksiyonu (g07)

Minimizasyon:

$$\begin{aligned} f(\mathbf{x}) = & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 \\ & + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 \\ & + (x_{10} - 7)^2 + 45 \end{aligned}$$

Kısıtlar:

$$g_1(\mathbf{x}) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0$$

$$g_2(\mathbf{x}) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0$$

$$g_3(\mathbf{x}) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0$$

$$g_4(\mathbf{x}) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0$$

$$g_5(\mathbf{x}) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0$$

$$g_6(\mathbf{x}) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0$$

$$g_7(\mathbf{x}) = 0,5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \leq 0$$

$$g_8(\mathbf{x}) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0$$

Sınır Kısıtları:

$$-10 \leq x_i \leq 10 \quad (i = 1, \dots, 10)$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = [2,17199634142692; 2,3636830416034; 8,77392573913157; \\ 5,09598443745173; 0,990654756560493; 1,43057392853463; \\ 1,32164415364306; 9,82872576524495; 8,2800915887356; 8,3759266477347]$$

$$f(\mathbf{x}^*) = 24,30620906818$$

g_1, g_2, g_3, g_4, g_5 ve g_6 aktif kısıtlardır.

8. Test Fonksiyonu (g08)

Minimizasyon:

$$f(\mathbf{x}) = -\frac{\sin^3(2\pi x_1) \sin(2\pi x_2)}{x_1^3(x_1 + x_2)}$$

Kısıtlar:

$$g_1(\mathbf{x}) = x_1^2 - x_2 + 1 \leq 0$$

$$g_2(\mathbf{x}) = 1 - x_1 + (x_2 - 4)^2 \leq 0$$

Sınır Kısıtları:

$$0 \leq x_1 \leq 10 \text{ ve } 0 \leq x_2 \leq 10$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = [1,22797135260752599; 4,24537336612274885]$$

$$f(\mathbf{x}^*) = -0,0958250414180359$$

9. Test Fonksiyonu (g09)

Minimizasyon:

$$f(\mathbf{x}) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 \\ - 4x_6x_7 - 10x_6 - 8x_7$$

Kısıtlar:

$$g_1(\mathbf{x}) = -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0$$

$$g_2(\mathbf{x}) = -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0$$

$$g_3(\mathbf{x}) = -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0$$

$$g_4(\mathbf{x}) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0$$

Sınır Kısıtları:

$$-10 \leq x_i \leq 10 \quad (i = 1, \dots, 7)$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = [2,33049935147405174; 1,95137236847114592; - \\ 0,477541399510615805; 4,36572624923625874; -0,624486959100388983; \\ 1,03813099410962173; 1,5942266780671519]$$

$$f(\mathbf{x}^*) = 680,630057374402$$

g_1 ve g_4 aktif kısıtlardır.

10. Test Fonksiyonu (g10)

Minimizasyon:

$$f(\mathbf{x}) = x_1 + x_2 + x_3$$

Kısıtlar:

$$g_1(\mathbf{x}) = -1 + 0,0025(x_4 + x_6) \leq 0$$

$$g_2(\mathbf{x}) = -1 + 0,0025(x_5 + x_7 - x_4) \leq 0$$

$$g_3(\mathbf{x}) = -1 + 0,01(x_8 - x_5) \leq 0$$

$$g_4(\mathbf{x}) = -x_1x_6 + 833,33252x_4 + 100x_1 - 83\,333,333 \leq 0$$

$$g_5(\mathbf{x}) = -x_2x_7 + 1\,250x_5 + x_2x_4 - 1\,250x_4 \leq 0$$

$$g_6(\mathbf{x}) = -x_3x_8 + 1\,250\,000 + x_3x_5 - 2\,500x_5 \leq 0$$

Sınır Kısıtları:

$$100 \leq x_1 \leq 10\,000$$

$$1\,000 \leq x_i \leq 10\,000, \quad (i = 2,3)$$

$$10 \leq x_i \leq 1\,000, \quad (i = 4, \dots, 8)$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = [579,306685017979589; 1\,359,97067807935605; 5\,109,97065743133317; \\ 182,01769963061534; 295,601173702746792; 217,982300369384632; \\ 286,41652592786852; 395,601173702746735]$$

$$f(\mathbf{x}^*) = 7\,049,24802052867$$

g_1, g_2 ve g_3 aktif kısıtlardır.

11. Test Fonksiyonu (g11)

Minimizasyon:

$$f(\mathbf{x}) = x_1^2 + (x_2 - 1)^2$$

Kısıtlar:

$$h(\mathbf{x}) = x_2 - x_1^2 = 0$$

Sınır Kısıtları:

$$-1 \leq x_1 \leq 1$$

$$-1 \leq x_2 \leq 1$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = (\pm \frac{1}{\sqrt{2}}; 1/\sqrt{2})$$

$$f(\mathbf{x}^*) = 0,7499$$

12. Test Fonksiyonu (g12)

Minimizasyon:

$$f(\mathbf{x}) = -\frac{100 - (x_1 - 5)^2 - (x_2 - 5)^2 - (x_3 - 5)^2}{100}$$

Kısıtlar:

$$g_1(\mathbf{x}) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2 - 0,0625 \leq 0$$

Sınır Kısıtları:

$$0 \leq x_i \leq 10 \quad (i = 1, 2, 3)$$

$$p, q, r = 1, 2, \dots, 9$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = (5; 5; 5)$$

$$f(\mathbf{x}^*) = -1$$

13. Test Fonksiyonu (g13)

Minimizasyon:

$$f(\mathbf{x}) = e^{x_1 x_2 x_3 x_4 x_5}$$

Kısıtlar:

$$h_1(\mathbf{x}) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10 = 0$$

$$h_2(\mathbf{x}) = x_2 x_3 - 5x_4 x_5 = 0$$

$$h_3(\mathbf{x}) = x_1^3 + x_2^3 + 1 = 0$$

Sınır Kısıtları:

$$-2,3 \leq x_i \leq 2,3 \quad (i = 1, 2)$$

$$-3,2 \leq x_i \leq 3,2 \quad (i = 3, 4, 5)$$

Literatürdeki En İyi Sonuçlar:

$$\mathbf{x}^* = [-1,71714224003; 1,59572124049468; 1,8272502406271; -0,763659881912867; -0,76365986736498]$$

$$f(\mathbf{x}^*) = 0,053941514041898$$

ÖZGEÇMİŞ

Adı Soyadı : İdris Karen
Doğum Yeri ve Tarihi : Samsun, 20.05.1979
Yabancı Dili : İngilizce (83,75 - ÜDS, Mart-2007)

Eğitim Durumu (Kurum ve Yıl)

Lise : Samsun Atatürk Anadolu Lisesi, 1997
Lisans : Uludağ Üniversitesi, 2002
Yüksek Lisans : Uludağ Üniversitesi, 2005

Çalıştığı Kurum/Kurumlar ve Yıl : Uludağ Üniversitesi, 2002-...

İletişim (e-posta) : ikaren@uludag.edu.tr; idriskaren@gmail.com

Yayımları (SCI ve diğer) :

Karen, İ., Kaya, N., Öztürk, F. 2011. Shape- and Topology-Based Structural Die Design using Differential Evolution and Response Surface Methodology for Sheet Metal Forming. *Materials Testing*, (Yayın aşamasında).

Karen, İ., Kaya, N., Öztürk, F., Korkmaz, İ., Yıldızhan, M., Yurttaş, A. 2011. A Design Tool to Evaluate the Vehicle Ride Comfort Characteristics: Modeling, Physical Testing and Analysis. *International Journal of Advanced Manufacturing Technology*, (DOI 10.1007/s00170-011-3592-z).

Kaya, N., Karen, İ., Öztürk, F. 2010. Redesign of a Failed Clutch Fork using Topology and Shape Optimisation by the Response Surface Method. *Materials & Design*, 31(6): 3008-3014.

Karen, İ., Kaya, N., Öztürk, F., Korkmaz, İ. 2010. Optimisation of Vehicle Engine Mount System using Simulation-Based Design Approach. *International Journal of Vehicle Design*, 52(1/2/3/4): 38-53.

Karen, İ., Yıldız, A.R., Kaya, N., Öztürk, N., Öztürk, F. 2006. Hybrid Approach for Genetic Algorithm and Taguchi's Method Based Design Optimization in Automotive Industry. *International Journal of Production Research*, 44(22): 4897-4914.

Öztürk, F., Öztürk, N., Kaya, N., Karen, İ. 2009. Design and Optimization Approaches to Provide a Frame for Designer in the Vehicle Development Phase with the Focus on Enhanced Safety. 21st International Technical Conference on the Enhanced Safety of Vehicles, ESV2009, 15-18 June 2009, Stuttgart, Germany.

- Karen, İ., Kaya, N., Öztürk, F., Kaftanoğlu, B., Mitmit, C., Polat, E. 2009.** Topology Based Structural Die Design Optimization Technique for Sheet Metal Stamping in the Automotive Industry. 5th International Conference and Exhibition on Design and Production of Machines and Dies/Molds, DieMold2009, 18-21 June 2009, Kuşadası, Aydın, Turkey.
- Karen, İ., Kaya, N., Öztürk, N., Alankuş, O., Öztürk, F. 2008.** Hybrid Artificial Intelligence and Structural Design Optimization Approaches in the Automotive Industry. F2008-04-33, World Automotive Congress, FISITA 2008, 14-19 September, Munich, Germany.
- Karen, İ., Yıldız, A.R., Kaya, N., Öztürk, F. 2004.** Evaluation of Multi-Objective Genetic Algorithm Optimization Program using Test Problems. Intelligent Manufacturing Congress, IMS04, 06-08 September 2004, Sakarya, Turkey.
- Yıldız, A.R., Karen, İ., Kaya, N., Öztürk, F. 2004.** Optimal Design of Machine Components using Evolutionary Genetic Algorithms. The First Cappadocia International Mechanical Engineering Symposium, CMES04, 14-16 July 2004, Cappadocia, Turkey.
- Yıldız, A.R., Karen, İ., Kaya, N., Öztürk, F. 2004.** Optimal Initial Design of Vehicle Components in the Conceptual Design Phase. The First Cappadocia International Mechanical Engineering Symposium, CMES04, 14-16 July 2004, Cappadocia, Turkey.
- Karen, İ., Kaya, N., Öztürk, F., Korkmaz, İ. 2011.** Taşıt Sürüş Konfor Özelliklerinin Fiziksel Testler ile Belirlenmesi ve Analizi. *Mühendis ve Makine*, Otomotiv Özel Sayı(615): 69-74.
- Karen, İ., Kaya, N., Öztürk, F., Korkmaz, İ. 2011.** Taşıt Sürüş Konfor Özelliklerinin Fiziksel Testler ile Belirlenmesi ve Analizi. 12. Otomotiv ve Üretim Teknolojileri Sempozyumu, 13-14 Mayıs 2011, Bursa, Türkiye.
- Karen, İ., Kaya, N., Öztürk, F., Yıldızhan, M., Yurttaş, A., Korkmaz, İ. 2010.** Araç Konfor Özelliklerinin Analizi: Model Tanımı ve Analizler ile Konfor Değerlendirmesi. 5. Otomotiv Teknolojileri Kongresi, OTEKON 2010, 07-08 Haziran 2010, Bursa, Türkiye.
- Özkan, U., Karen, İ., Kaya, N., Öztürk, F. 2010.** Araç İçi Akustik Özelliklerinin Analizi ve Eniyilemesi. 5. Otomotiv Teknolojileri Kongresi, OTEKON 2010, 07-08 Haziran 2010, Bursa, Türkiye.
- Gürarda, A., Burhan, C., Karen, İ., Kaya, N., Öztürk, F., Korkmaz, İ. 2010.** Computer Aided Fatigue Life Time Improvement of Rubber Based Automotive Engine Mounts. 5. Otomotiv Teknolojileri Kongresi, OTEKON 2010, 07-08 Haziran 2010, Bursa, Türkiye.
- Öztürk, T., Karen, İ., Kaya, N., Öztürk, F. 2010.** Taşıt Lastiklerinin Sonlu Elemanlar Yöntemiyle Modellenmesi ve Simülasyonu. 5. Otomotiv Teknolojileri Kongresi, OTEKON 2010, 07-08 Haziran 2010, Bursa, Türkiye.

Karen, İ., Kaya, N., Öztürk, F., Kaftanoğlu, B., Mitmit, C., Polat, E. 2009. Otomotiv Endüstrisinde Sac Metal Kalıp Tasarımı ve Topoloji Tabanlı Optimizasyonu. 11. Otomotiv Sempozyumu, TMMOB Makine Mühendisleri Odası, 08-09 Mayıs 2009, Bursa, Türkiye.

Korkmaz, İ., Karen, İ., Kaya, N., Öztürk, F. 2008. Fren Sistemlerinin NVH Analizleri ve Optimizasyonu. 4. Otomotiv Teknolojileri Kongresi, OTEKON'08, 01-04 Haziran 2008, Bursa, Türkiye.

Karen, İ., Kaya, N., Öztürk, F. 2008. Genetic and Simulation Based Approach to Multi-objective Optimum Shape Design. 4. Otomotiv Teknolojileri Kongresi, OTEKON'08, 01-04 Haziran 2008, Bursa, Türkiye.

Karen, İ., Kaya, N., Öztürk, F., Korkmaz, İ. 2008. Motor Titreşim Takozlarının İstenen Özelliklerde Tasarımı ve Doğrulaması. 4. Otomotiv Teknolojileri Kongresi, OTEKON'08, 01-04 Haziran 2008, Bursa, Türkiye.

Karen, İ., Yıldız, A.R., Kaya, N., Öztürk, F. 2004. Multi Objective Production Design with Genetic Algorithms. Advanced Technologies and Applications in Aeronautics, HİTEK 2004, 9-10 December 2004, İstanbul, Türkiye.

Kaya, N., Karen, İ. 2011. Proje Ref. KMTTG, (Bayrak Ar-Ge. – ULUTEK), Konvansiyonel Motor Titreşim Takoğu Geliştirilmesi (2011 – 2012).

Öztürk, F., Kaya, N., Karen, İ., Yıldızhan, M., Yurttaş, A. 2008. Proje Ref. ASKTO-08, (Platform A.Ş. – ULUTEK), Araç Sürüş Konforu Tasarım Optimizasyonu (Kasım 2008 – Ekim 2011).

Öztürk, F., Kaya, N., Karen, İ. 2008. Proje Ref. M2008-64, (Uludağ Üniversitesi - BAPB), Araç Analiz Simülasyonları için Algoritma Geliştirilmesi, (2008 - 2011).

Öztürk, F., Kaya, N., Karen, İ. 2008. Proje Ref. APÜG-08, (Coşkunöz Ar-Ge San. ve Tic. A.Ş. - ULUTEK), Salıncak Kolu Parçası Ürün Geliştirilmesi (2008 - 2009).

Öztürk, F., Kaya, N., Karen, İ. 2008. Proje Ref. AKTO-08, (Platform A.Ş. – ULUTEK), Kalıp Tasarımı ve Optimizasyonu (Nisan 2008 – Aralık 2008).

Öztürk, F., Kaya, N., Karen, İ., Kaftanoğlu, B. 2007. Proje Ref. KYTO-07, (Platform A.Ş. – ULUTEK), Kalıp Tasarımı ve Yapısal Optimizasyonu (2007 - 2008).

Öztürk, F., Kaya, N., Karen, İ. 2007. Proje Ref. TAEO-07, (Platform A.Ş. – ULUTEK), Motor Titreşim Takoğu Analiz, Test ve Optimizasyonu, (2007 - 2008).

Kaya, N., Öztürk, F., Karen, İ., Öztürk, N. 2006. Proje Ref. M2006-33, (Uludağ Üniversitesi - BAPB), Taşıt Elemanlarının Analizleri ile Optimum Tasarımı, (2006 - 2009).

Öztürk, N., Öztürk, F., Yıldız, A.R., Karen, İ. 2005. Proje Ref. 104M240, (TÜBİTAK – Kariyer Projesi), Yapısal Optimizasyon, Yapay Zekâ ve Evrimsel Algoritma Yaklaşımı ile Tasarım, (2005 - 2009).

Öztürk, F., Kaya, N., Yıldız, A.R., Karen, İ., Öztürk, N. 2004. Proje Ref. M2004-27, (Uludağ Üniversitesi - BAPB), (2004 – 2006).

TÜBİTAK Yurtiçi Doktora Bursu - 2211 (2005 - 2009).

YÖK Yurtdışı Doktora Araştırma Bursu (Torino Politeknik Üniversitesi, İtalya, 2009).