CELAL BAYAR ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

IZGARA SİSTEMLERİN ATEŞ BÖCEĞİ ALGORİTMASI OPTİMİZASYON YÖNTEMİ İLE OPTİMUM BOYUTLANDIRILMASI

YÜKSEK LİSANS TEZİ

İnşaat Mühendisi Fatma Esra UZ

Anabilim Dalı : İnşaat Mühendisliği

Programı : Mekanik

MANİSA 2014

CELAL BAYAR ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

IZGARA SİSTEMLERİN ATEŞ BÖCEĞİ ALGORİTMASI OPTİMİZASYON YÖNTEMİ İLE OPTİMUM BOYUTLANDIRILMASI

YÜKSEK LİSANS TEZİ

İnşaat Mühendisi Fatma Esra UZ

Tezin Enstitüye Verildiği Tarih : 19.03.2014

Tezin Savunulduğu Tarih : 31.03.2014

Tez Danışmanı : Yrd. Doç. Dr. Erkan DOĞAN (CBÜ)

Diğer Jüri Üyeleri : Doç. Dr. Mehmet ÇEVİK (CBÜ)

Yrd. Doç. Dr. Abdulkerim ERGÜT (CBÜ)

MANİSA 2014

İÇİNDEKİLER

İÇİNDEKİLER i
ŞEKİL LİSTESİ iv
ÇİZELGE LİSTESİ vi
KISALTMA LİSTESİ vii
TEŞEKKÜR viii
ÖZET ix
ABSTRACT x

1. GİRİŞ 1
1.1 Genel Bilgi 1
1.2 Optimizasyon 2
1.2.1 Optimizasyona Giriş 2
1.2.1.1 Sürekli Optimizasyon 4
1.2.1.2 Kesikli Optimizasyon 4
1.2.2 Yapısal Optimizasyon 5
1.2.2.1 Yapısal Optimizasyon Problemleri 5
1.2.2.1.1 Boyutlandırma Optimizasyonu 7
1.2.2.1.2 Şekil Optimizasyonu 7
1.2.2.1.3 Topoloji Optimizasyonu 8
1.2.2.2 Yapısal Optimizasyon Yöntemleri 8
1.2.2.2.1 Matematiksel Programlama 9
1.2.2.2.2 Optimumluk Kriteri Yöntemi 9
1.2.3 Stokastik Optimizasyon Yöntemleri 9
1.2.3.1 Yapay Arı Koloni Algoritması 10
1.2.3.2 Benzetimli Tavlama 11
1.2.3.3 Genetik Algoritma 11
1.2.3.4 Harmoni Arama 13
1.2.3.5 Tabu Arama 14
1.2.3.6 Av Arama Optimizasyonu 14
1.2.3.7 Karınca Koloni Optimizasyonu 15
1.2.3.8 Parçacık Sürü Optimizasyonu 16
1.2.4 Kısıtlama İşleme Yöntemi 17

1.2.4.1 Ceza Fonksiyonları Yöntemi 17
1.2.4.2 Geri Uçuş Mekanizması 18
1.3 Literatür Araştırması 18
2. ATEŞ BÖCEĞİ SÜRÜ OPTİMİZASYONU 20
2.1 Giriş 20
2.2 Ateş Böceği Algoritması 21
2.3 Ateş Böceği Algoritması Yöntemlerinin Uygulaması 27
2.3.1 Örnek 1 27
2.3.2 Örnek 2 29
2.3.3 Örnek 3 30
2.3.4 Örnek 4 – Kaynaklı Kiriş Tasarım Problemi 31
2.3.5 Örnek 5 – Yay Tasarım Problemi 34
3. IZGARA SİSTEMLERİN LRFD ŞARTNAMESİNE GÖRE TASARIMI 38
3.1 Izgara Sistemler 38
3.2 Matris Deplasman Yöntemi 39
3.2.1 Global Koordinat Sistemi 39
3.2.2 Lokal Koordinat Sistemi 40
3.2.3 Global ve Lokal Koordinatlar Arasındaki İlişki 41
3.2.4 Çubuk Elemanların Uç Kuvvetleri ile Uç Deplasmanlar Arasındaki İlişki 43
3.2.5 Her İki Ucu Rijit Bir Şekilde Bağlanmış Izgara Elemanın Rijitlik Matrisi 45
3.2.6 Çubuk Elemanlarda Global Koordinatlarda Uç Kuvvetleri ve Deplasmanları 46
3.2.7 Dış Yükler ve Eleman Kuvvetleri Arasındaki İlişki 48
3.2.8 İlk Ucu Mafsallı Izgara Elemanın Rijitlik Matrisi 49
3.2.9 İkinci Ucu Mafsallı Izgara Elemanın Rijitlik Matrisi 51
3.3 LRFD Şartnamesine Göre Çelik Yapı Elemanların Tasarımı 53
3.3.1 LRFD Şartnamesine Göre Kirişlerde Eğilme Etkisi 54
3.3.1.1 Kompakt (Compact) Kesit 55
3.3.1.2 Kompact Olmayan (Non-compact) Kesit 55
3.3.1.3 Kısmen Kompact (Compact) Kesit 56
3.3.1.4 Narin (Slender) Kesit 56
3.3.2 LRFD Şartnamesine Göre Kirişlerde Kesme Etkisi 57

4. ATEŞ BÖCEĞİ ALGORİTMASI OPTİMİZASYON YÖNTEMİ İLE IZGARA SİSTEMLERİN OPTİMUM TASARIMI 59
4.1 Izgara Sistemlerin Optimum Tasarımı 59
4.1.1 Izgara Sistemlerin Optimum Boyutlandırılması İçin Matematiksel Model 59
4.2 Ateş Böceği Algoritması Optimizasyon Yöntemi 61
4.3 Ateş Böceği Algoritması Optimizasyon Yöntemi İle Izgara Sistemlerin Analizi 63
4.4 Boyutlandırma Örnekleri 64
4.4.1 40- Elemanlı Basit Izgara Sistem 64
4.4.2 27- Elemanlı Basit Izgara Sistem 67
4.4.3 60- Elemanlı Basit Izgara Sistem 70
4.4.4 25- Elemanlı Ankastre Izgara Sistem 73
4.4.5 61- Elemanlı Ankastre Izgara Sistem 77
5. SONUÇLAR VE ÖNERİLER 82

KAYNAKLAR 84
ÖZGEÇMİŞ 88

ŞEKİL LİSTESİ

Şekil 1.1 İki değişkenli tasarım problemi 4
Şekil 1.2 Boyutlandırma optimizasyonu 7
Şekil 1.3 Şekil optimizasyonu problemi 8
Şekil 1.4 Topoloji optimizasyonu 8
Şekil 1.5 Genetik algoritma işletimi 12
Şekil 1.6 Müzik geliştirme ve mühendislik optimizasyonu arasındaki bağlantı 14
Şekil 1.7 Besin kaynağına giden karıncaların en kısa yolu bulma aşamaları 16
Şekil 2.1 Işık saçan ateş böcekleri 21
Şekil 2.2 Ateş böceği optimizasyon algoritması 24
Şekil 2.3 Ateş böceği optimizasyon algoritması için akış şeması 26
Şekil 2.4 Kaynaklı kiriş tasarımı 32
Şekil 2.5 Kaynaklı kiriş tasarım problemi için ateş böceği algoritması tasarım geçmişi 34
Şekil 2.6 Yay tasarım problemi 35
Şekil 2.7 Yay tasarım problemi için ateş böceği algoritması tasarım geçmişi 36
Şekil 3.1 Izgara sistem 38
Şekil 3.2 Izgara sistem 38
Şekil 3.3 Kartezyen koordinat sistemi 40
Şekil 3.4 Kiriş eleman 40
Şekil 3.5 Farklı kesitler için lokal eksenler 41
Şekil 3.6 Global ve lokal koordinatlar arasındaki ilişki 42
Şekil 3.7 Izgara sistem elemanına etkiyen uç kuvvetler ve momentler 43
Şekil 3.8 İki ucu rijit bağlı eleman 45
Şekil 3.9 Çubuk elemana etkiyen burulma momentleri 45
Şekil 3.10 Kesme kuvveti ve eğilme momenti 45
Şekil 3.11 Izgara elemanın deplasmanları 47
Şekil 3.12 İlk ucu mafsallı eleman 49
Şekil 3.13 İkinci ucu mafsallı eleman 51
Şekil 3.14 Basit kiriş 53
Şekil 3.15 I (W) profil kesit 54

Şekil 3.16 Kesitlerin sınıflandırılması 54
Şekil 3.17 I (W) kesitli profillerin sınır kesme gerilmeleri..... 57
Şekil 4.1 40- elemanlı basit ızgara sistem 65
Şekil 4.2 40- elemanlı basit ızgara sistem için minimum ağırlık zaman grafiği..... 67
Şekil 4.3 27- elemanlı basit ızgara sistem 68
Şekil 4.4 27- elemanlı basit ızgara sistem için minimum ağırlık zaman grafiği..... 70
Şekil 4.5 60- elemanlı basit ızgara sistem 71
Şekil 4.6 60- elemanlı basit ızgara sistem için minimum ağırlık zaman grafiği..... 73
Şekil 4.7 25- elemanlı ankastre ızgara sisteminin grup numaraları ve kiriş açıklıkları(plan) 74
Şekil 4.8 25- elemanlı ankastre ızgara sisteminin yükleme durumu..... 74
Şekil 4.9 25- elemanlı ankastre ızgara sistem için minimum ağırlık zaman grafiği 77
Şekil 4.10 61- elemanlı ankastre ızgara sisteminin grup numaraları ve kiriş açıklıkları(plan) ... 78
Şekil 4.11 61- elemanlı ankastre ızgara sisteminin yükleme durumu..... 78
Şekil 4.12 61- elemanlı ankastre ızgara sistem için minimum ağırlık zaman grafiği 81

ÇİZELGE LİSTESİ

Çizelge 2.1 Örnek 1 için optimum çözümler 28

Çizelge 2.2 Örnek 2 için optimum çözümler 29

Çizelge 2.3 Örnek 3 için optimum çözümler 30

Çizelge 2.4 Kaynaklı kiriş problemi için optimum çözümler 34

Çizelge 2.5 Yay tasarım problemi için optimum çözümler 36

Çizelge 4.1 Optimizasyon tekniklerinin parametreleri 64

Çizelge 4.2 40- elemanlı basit ızgara sistem için PSO, HA, AAO yöntemlerine ait ağırlık sonuçları 66

Çizelge 4.3 40- elemanlı ızgara sisteminin optimum tasarım sonuçları 66

Çizelge 4.4 27- elemanlı basit ızgara sistem için PSO, HA, AAO yöntemlerine ait ağırlık sonuçları 68

Çizelge 4.5 27- elemanlı ızgara sisteminin optimum tasarım sonuçları 69

Çizelge 4.6 60- elemanlı basit ızgara sistem için PSO, HA, AAO yöntemlerine ait ağırlık sonuçları 72

Çizelge 4.7 60- elemanlı ızgara sisteminin optimum tasarım sonuçları 72

Çizelge 4.8 25- elemanlı ankastre ızgara sistem için PSO, HA, AAO yöntemlerine ait ağırlık sonuçları 75

Çizelge 4.9 25- elemanlı ankastre ızgara sisteminin optimum tasarım sonuçları 76

Çizelge 4.10 61- elemanlı ankastre ızgara sistem için PSO, HA, AAO yöntemlerine ait ağırlık sonuçları 79

Çizelge 4.11 61- elemanlı ankastre ızgara sisteminin optimum tasarım sonuçları 80

KISALTMALAR

LRFD-AISC : Yük ve Direnç Faktörü Tasarımı – Amerikan Çelik Konstrüksiyon Enstitüsü – emniyet Gerilmesi Tasarımı

GP : Geometrik Programlama

GA : Genetik Algoritma

PSO : Parçacık Sürü Optimizasyonu

AA : Av Arama

AAO : Ateş Böceği Algoritması Optimizasyonu

EA : Evrimsel Algoritma

HA : Harmoni Arama

TEŞEKKÜR

Tez çalışmam boyunca, her aşamada yardımcı ve yönlendirici olan, öneri ve desteğini esirgemeyen tez danışmanım Yrd.Doç.Dr. Erkan DOĞAN' a, ayrıca hayatım boyunca maddi manevi hiçbir fedakarlıktan kaçınmayan aileme teşekkür ederim.

IZGARA SİSTEMLERİN ATEŞ BÖCEĞİ ALGORİTMASI OPTİMİZASYON YÖNTEMİ İLE OPTİMUM BOYUTLANDIRILMASI

ÖZET

Bu çalışmada, ızgara sistemlerin literatürde yer alan ateş böceği algoritması yöntemi yardımıyla optimum profil boyutlarına ve minimum ağırlığa sahip olacak şekilde tasarımı yapılmıştır. Algoritma adını, hareket mantığının model olarak kabul edildiği ateş böceklerinden alır. Ateş böceklerinin tek cins olmaları ve her bir bireyin kendi aralarındaki çekim gücü etkisi algoritmanın temelini oluşturmaktadır. Tasarım algoritması LRFD-AISC (Yük ve Direnç Faktörü Tasarımı-Amerikan Çelik Konstrüksiyon Enstitüsü) uygulaması sonucu oluşan yer değiştirme ve dayanım sınırlamalarını göz önüne almaktadır. Ateş böceği algoritması yöntemi bu tasarım sınırlayıcılarının sağlanması koşulu ile ızgara sistemin enine ve boyuna kirişleri için LRFD-AISC' de verilen 272 farklı W-profili arasından uygun profili seçer. Oluşturulan her bir tasarımın ağırlığı kendisini meydana getiren bütün profillerin birim ağırlıklarının dikkate alınması suretiyle hesaplanır. Bütün iterasyonlar sonunda minimuma indirgenmiş olan sistem ağırlığı algoritma tarafından kaydedilir ve nihai sonuç olarak kabul edilir. Tasarım örnekleri sunulan algoritmanın etkinliğini göstermeyi amaçlamaktadır.

Anahtar Kelimeler: Ateş böceği algoritması, ızgara sistemler, minimum ağırlık, optimum boyutlandırma, stokastik arama yöntemleri

OPTIMUM DESIGN OF GRILLAGE SYSTEMS WITH THE FIREFLY ALGORITHM OPTIMIZATION METHOD

ABSTRACT

In this study, grillage systems are designed by means of a well-known meta-heuristic optimization method in the literature, called firefly algorithm, such that they have optimum profile sizes and minimum weight. Naming of the algorithm is derived from the fireflies, whose sense of movement is taken as a model in the development of the algorithm. Fireflies' being unisex and their attracting each other constitute the basis of the algorithm. The design algorithm considers the displacement and strength constraints that in generated as a result of the implementation of LRFD-AISC (Load and Resistance Factor Design-American Institute of Steel Construction). Of the 272 different W-profiles provided in LRFD-AISC, the method of firefly algorithm selects the appropriate profile for transverse and longitudinal beams of the grillage system, such that the design limitations described in LRFD-AISC are satisfied and the weight of the system is reduced to minimum. The weight of each candidate design generated during the iteration process is calculated by considering the unit weight of each section, which constitutes the system. Design examples is aimed to demonstrate the efficiency of the algorithm presented.

Keyword: Firefly algorithm, grillage systems, minimum weight, optimum design, stochastic search techniques

1. GİRİŞ

1.1 Genel Bilgi

Bu tez çalışması kapsamında ızgara sistemler için ateş böceği algoritması optimizasyon yöntemini temel alan bir optimum tasarım algoritması sunulmaktadır. Bu optimizasyon yöntemi sezgiye dayalı optimizasyon metotları olarak bilinen, sosyal davranışların taklit edildiği sayısal optimizasyon teknikleri içinde yer alır. Her bir birey optimum tasarım problemine aday bir çözüm ifade eder. Izgara sistemlerin optimum tasarımının gereği olarak; yapı elemanlarının W profiline uygun olması, LRFD-AISC tasarım şartnamelerindeki sınırlayıcıları sağlaması ve sistemin minimum ağırlığa sahip olması gerekmektedir. Bu amaçla deplasmanı da hesaba katan dayanım sınırlayıcıları göz önünde bulundurulmuş ve algoritmanın kirişler için Amerikan Şartnamesindeki 272 kesit içerisinden seçim yapması sağlanmıştır. Her bir grup içindeki elemanlara aynı kesitin tayin edilebilmesi için eleman gruplandırılması yapılmıştır. Şartnamedeki sınırlayıcıları dikkate alarak minimum sistem ağırlığını hesaplayan bu algoritmanın etkinliği örnek problemlerle gösterilmiştir. Çalışmada güncel hayatta çeşitli alanlarda sıklıkla kullanılan bir yapı modeli olan ızgara sistemlerin optimizasyonu araştırılmıştır. Yatay düzlemde birbirine rijit bağlı elemanlardan oluşan ızgara sistemler yapılarda köprü tabliyesi, gemi gövdesi, kat döşemesi gibi geniş alanların kaplanması amacıyla sıklıkla kullanılan yapı elemanlarıdır. Izgara sistemlerin optimum boyutlandırılmasını gerçekleştiren bilgisayar programı, FORTRAN programlama dilinin kullanılmasıyla hazırlanmıştır.

Artan popülasyon sayısı ve gereksinim ile birlikte azalan hammadde miktarı, insanoğlunu malzemeleri en verimli şekilde kullanma yoluna itmiştir. Bunun sonucu olarak yapılan tasarımların en az maliyete sahip olacak şekilde gerçekleştirilmesi hedeflenmiş ve bu amaçla çeşitli yöntemler üretilmiştir. Bu yöntemlerden biri olan optimizasyon, insanlar tarafından en uygun, en rahat, en ekonomik, en kısa, en sağlam gibi sorulara cevap verebilmek adına yapılan çalışmalar sonucunda ortaya çıkmış bir çözümdür. Yapıların ekonomik olarak tasarlanması mühendisliğin en önemli amaçları içerisindedir. Yapıların minimum ağırlıklı olarak tasarımı hem yapılara gelecek yüklerin azaltılması hem de maliyetin en düşük seviyede tutulması açısından önemlidir.

Bu amaçla bir tasarım problemini çözmek için insanoğlu doğayı gözlemlemiş ve bundan esinlenmiştir. Bazen tek başlarına hiçbir iş yapamayan varlıklar, toplu hareket ettiklerinde çok zekice davranışlar sergileyebilmektedir. Bir topluluğa ait bireyler, en iyi bireyin davranışından ya da diğer bireylerin davranışlarından ve kendi deneyimlerinden yararlanarak yorum yapmakta ve bu bilgileri ileride karşılaşacakları problemlerin çözümleri için bir araç olarak kullanmaktadırlar. Örneğin, bir canlı sürüsünü oluşturan bireylerden birisi bir tehlike sezdiğinde bu tehlikeye karşı tepki verir ve bu tepki sürü içinde ilerleyip tüm bireylerin tehlikeye karşı ortak bir davranış

sergilemesini sağlar. Canlıların sürü içerisindeki bu hareketleri gözlemlenerek optimizasyon algoritmaları geliştirilmiştir.

Bu tez kapsamında, ateş böceği algoritması optimizasyon yönteminden faydalanılmıştır. Ateş böceklerinin parlaklığı, sezgisel algoritmanın hedef fonksiyonunu temsil etmektedir. Sürü içerisinde her bir ateş böceği, optimizasyon işlemi için uygun bir aday çözümü ifade etmektedir. Amaç fonksiyonu olarak yapı ağrılığının en düşük olduğu optimizasyon problemleri için, ateş böceği algoritması kullanılarak ızgara sistem tasarımı yapılmıştır.

1.2 Optimizasyon

1.2.1 Optimizasyona Giriş

Optimizasyon, tüm problem sınırlayıcılarının sağlanması koşulu ile amaç fonksiyonunu minimum veya maksimum yapan değerlerin bulunması işlemidir. Optimizasyon aynı zamanda matematiksel problem içinde tanımlanmış olan bütün çözüm kombinasyonları içinden en uygun sonuç verenin bulunması işlemi olarak da tanımlanabilir.

Optimizasyon, en iyileme anlamına da gelmektedir. Bir problem için, verilen şartlar altında tüm çözümler arasından en iyi çözümü elde etme işidir.

Herhangi bir tasarım probleminin bir optimizasyon yöntemiyle çözülebilmesi için tasarım değişkenlerinin belirlenmesi gerekmektedir. Bu değişkenlere bağlı olarak maliyet fonksiyonu ya da kar fonksiyonu belirlenmekte ve bunlardan uygun olanlarını belirleyen kısıt fonksiyonları tanımlanmaktadır. Maliyet fonksiyonunda minimum sayısal değere ulaşmak hedeflenirken, kar fonksiyonunda maksimum sayısal değer amaç fonksiyonu olarak kabul edilir.

Kısıtlayıcı fonksiyonlar problem için uygun sınırlar belirlemekte ve aranan optimum amaç fonksiyonu değeri çizilen bu sınırlar içerisinde yer almaktadır. Sınırlayıcılar bazı durumlarda eşitsizliklerle tanımlanırken bazı durumlarda ise eşitlikler biçiminde ifade edilebilirler. Bir optimizasyon problemlerinde tasarımdan beklentiler ne kadar artarsa problem için çözüm o derece zorlaşmaktadır (Karaboğa, 2011).

Bir optimizasyon problemi matematiksel olarak aşağıdaki gibi ifade edilebilir:

n değişken sayısına sahip amaç fonksiyonu $z = f(x)$

$$f(x) = f(x_1, x_2, x_3, x_4, ...x_n) \tag{1.1}$$

Kısıt fonksiyonları;

$$h_j(x) = h_j(x_1, x_2, x_3, x_4, ...x_p) = 0 \quad , \quad 1 \le j \le p \tag{1.2}$$

$$g_i(x) = g_i(x_1, x_2, x_3, x_4, ...x_m) = 0 \quad , \quad 1 \le i \le m \tag{1.3}$$

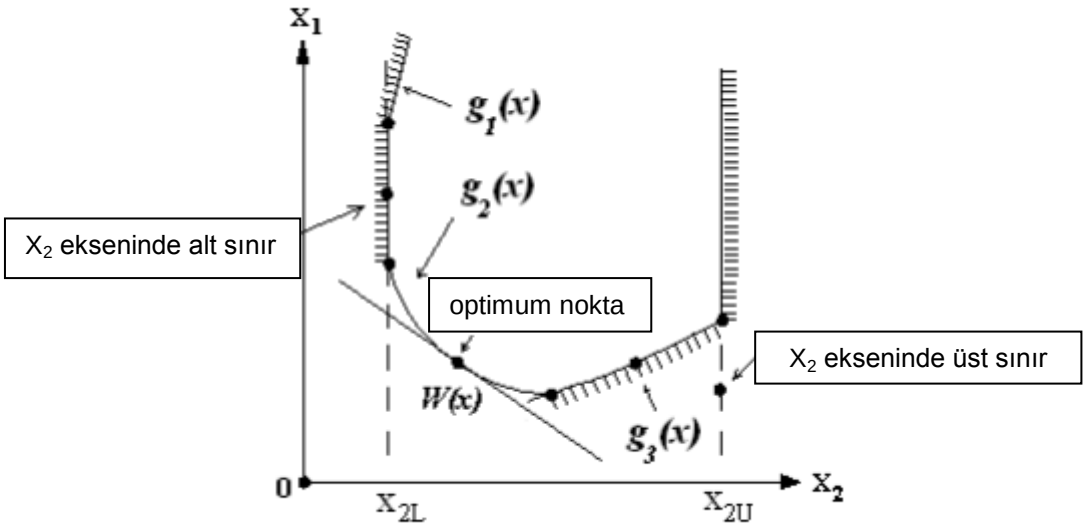
Bazı optimizasyon problemleri problemin tipine göre geometrik sınırlayıcılar da içerebilir. Bu sınırlamalar tasarım değişkenlerine bağımlı olarak tanımlanır. Optimum tasarıma aday bir çözümün uygun bir çözüm olarak kabul edilebilmesi için problemi oluşturan bütün sınırlayıcıların sağlanması gerekir. Aksi takdirde, oluşturulan bu tasarım göz ardı edilerek prosedür dışında tutulur.

Aşağıda formülasyonu verilen optimum tasarım probleminde amaç fonksiyonu olarak minimum ağırlık fonksiyonu hesaba katılmıştır. Grafik çözümü Şekil 1.1' de verilen problem üç adet eşitsizlik ve bir adet kenar sınırlayıcısı içermektedir. İki adet karar değişkeniyle değeri belirlenen amaç fonksiyonunun bir yapısal optimizasyon problemi için yapı ağırlığına tekabül ettiği kabul edilirse, eşitsizlik sınırlayıcıları olarak verilen g_1 , g_2 ve g_3 ün deplasman ve/veya gerilme sınırlayıcıları olduğu düşünülebilir. Bu durumda x_{2L} ve x_{2U} değişkenlerin alt ve üst sınırlarını belirtir. Sınırlayıcılar ile belirlenen alan, problem için uygun çözümlerin yer aldığı çözüm uzayını temsil eder.

$$Min\ W = W(x)$$

Kısıtlar :

$$g_1(x) \le 0$$
$$g_2(x) \le 0$$
$$g_3(x) \le 0$$
$$x_{2L} \le x_2 \le x_{2U}$$



Şekil 1.1 – İki değişkenli tasarım problemi

Optimizasyon problemleri tasarım değişkenlerinin türüne bağlı olarak iki bölüme ayrılmaktadır.

- Sürekli optimizasyon

- Kesikli optimizasyon

1.2.1.1 Sürekli Optimizasyon

Sürekli optimizasyonda, amaç fonksiyonu ve kısıtlar, tasarım parametrelerinin sürekli bir fonksiyonudur. Bir optimizasyon problemindeki değişkenler belirlenen sınırlar dahilinde tüm değerleri alabiliyorlarsa bu tip problemler sürekli optimizasyon problemi olarak adlandırılır. Amaç fonksiyonu, problem verimlilik problemi ise maksimum, maliyet problemi ise minimum çözümler verecek şekilde belirlenir. Sürekli optimizasyon kısıtlı ve kısıtsız optimizasyon olmak üzere ikiye ayrılmaktadır. Kısıtlı sürekli optimizasyonda, kısıtlar eşitlik ya da eşitsizlikler biçiminde yazılabilir (Bayar, 2008). Örnek olarak; etki eden yüklere göre tasarım sınırları içerisinde bir çubuk elemanın sağlaması gereken minimum kesit alanı sürekli bir optimizasyon problemi olarak düşünülebilir. Ama uygulamada kolaylık sağlaması amacıyla bu kesit alanını sağlayabilmek için boyutlar tam sayı değerlerine yuvarlatılır (Şeker,2013).

1.2.1.2 Kesikli Optimizasyon

Kesikli optimizasyonda, amaç fonksiyonu ve kısıtlar, tasarım parametrelerinin sürekli bir fonksiyonu değildir. Karar değişkeni sadece belli değerleri alıyorsa, bu durumda kesikli değişken olarak adlandırılır. Optimizasyon modellerinde bu türden tasarım değişkenleri genellikle (0-1)

olarak ya da negatif olmayan tamsayılar olarak adlandırılır. Bir tasarım parametresinin standart bir veritabanından seçilme zorunluluğu olduğu durumlarda yapısal tasarımlar kesikli karar alanı ile belirlenir. Buna örnek olarak kesit modülünün var olan bir tablodan seçilme zorunluluğu verilebilir. Her ne kadar sürekli optimizasyon mükemmel bir sonuç üretse de, uygulamalarda sonuçların kesikli değerlere dönüştürülerek değiştirilmesi gerekir. Çoğu uygulamada yukarı doğru yuvarlatılmış değerler ele alınmaktadır (Bayar, 2008).

Bahsedilen nedenlerden dolayı, parametrelerin kesikli olarak tanımlandığı problemler literatürde "kesikli optimizasyon problemleri" ve özelde de "tamsayılı optimizasyon problemleri" olarak ele alınır. Kesikli optimizasyon problemleri doğrusal veya doğrusal olmayan türden olabilir (Şeker, 2013).

1.2.2 Yapısal Optimizasyon

Mühendislikte optimum tasarım, fiziksel modellerin matematiksel problemler olarak tanımlanmasında en iyi çözümü saptama işlemi olarak ifade edilir. Optimizasyon alanı içerisinde, optimizasyon için uygulanan ve uygulanabilecek yöntemlerin çeşitli problemler için çözüm algoritmalarının oluşturulması, bu yöntemlerin yapısal olarak incelenmesi ve bilgisayar ortamında bir araya getirilmesi yer almaktadır. Mühendislik alanlarında optimizasyon, yapı davranışı ve eleman kuvvetlerinin göz önünde bulundurulması suretiyle yapısal sistemlerin optimum şekil ve geometrik tasarımı olarak ifade edilebilir. Bir optimum tasarım problemi için üç öğenin tanımlanması gerekmektedir. Bunlar; amaç fonksiyonu, tasarım değişkenleri ve tasarım sınırlayıcılarıdır. Tasarım değişkenleri, optimum çözüme ulaşmak için optimizasyon işlemi boyunca değiştirilen sistem parametreleridir. Sınırlayıcılar ise tasarım ve optimizasyon probleminde istenmeyen çözümlerin oluşmasını engellemek amacıyla problem için belli sınırlar içerisinde uygun çözümlerin bulunmasını sağlarlar (Bayar, 2008).

1.2.2.1 Yapısal Optimizasyon Problemleri

Optimizasyon problemleri genel olarak tasarım değişkenlerinin tipine bağlı olarak sürekli ve kesikli olmak üzere ikiye ayrılmaktadır. Sürekli değişkenler belli bir aralıkta bütün değerleri alabilmesine karşın, kesikli değişkenler belirli bir aralıkta sadece önceden belirlenmiş belirli değerleri alabilir. Yapısal optimizasyon problemlerinde karşılaşılan değişkenler genellikle kesikli değişkenlerdir.

Kesikli yapısal optimizasyon problemleri genel olarak yapının tasarım ve davranış kısıtları ile sınırlandırılmış olan $f(x)$ amaç fonksiyonunu en küçük yapan optimum $''x''$ tasarım vektörünün bulunması işlemi şeklinde ifade edilebilir. Yapısal optimizasyon problemleri aşağıda gösterildiği şekilde ifade edilir.

$$x, x_i \in X, \quad X = \{x_1, x_2, \ldots, x_q\}$$

$$f(x) = \sum_{i=1}^{q} \gamma_i L_i x_i \tag{1.4}$$

$$g_i^{'}(x) = \sigma_i^{'} - \sigma_i^a \leq 0 \qquad i = 1, \ldots, q \tag{1.5}$$

$$f_k^{'}(x) = u_k^{'} - u_k^a \leq 0 \qquad k = 1, \ldots, m \tag{1.6}$$

$$X_{lb} \leq x \leq X_{ub} \tag{1.7}$$

Burada;

$f(x)$: Genellikle yapı ağırlığı olarak belirlenen amaç fonksiyonu

X : Kesikli boyut tablosu

q : Tasarım değişkenleri ya da elemanların toplam sayısı

m : Deplasman kısıtlarının toplam sayısı

γ_i : *i*'nci elemanın özgül ağırlığı

L_i, x_i : *i*'nci elemanın boyu ve kesit alanı

$\sigma_i^{'}, \sigma_i^a$: Sırasıyla *i*'inci yük durumu altındaki gerilme değeri ve *i*'nci elemanın emniyet gerilmesinin mutlak değeri

$u_k^{'}, u_k^a$: Sırasıyla *i*'inci yük durumu altındaki deplasman değeri ve müsaade edilen deplasmanın mutlak değeri

X_{lb} : Tasarım değişkenlerinde en alt sınır vektörü

X_{ub} : Tasarım değişkenlerinde en üst sınır vektörü

Amaç fonksiyonlarının ve kısıtlamaların sayısına ve karar alan boyutuna bağlı olarak optimum tasarım probleminin karmaşıklığı değişebilir (Arora, 2007).

Yapısal optimizasyon problemleri, geometrik özelliğine bağlı olmak üzere;

- Boyutlandırma optimizasyonu

- Şekil optimizasyonu

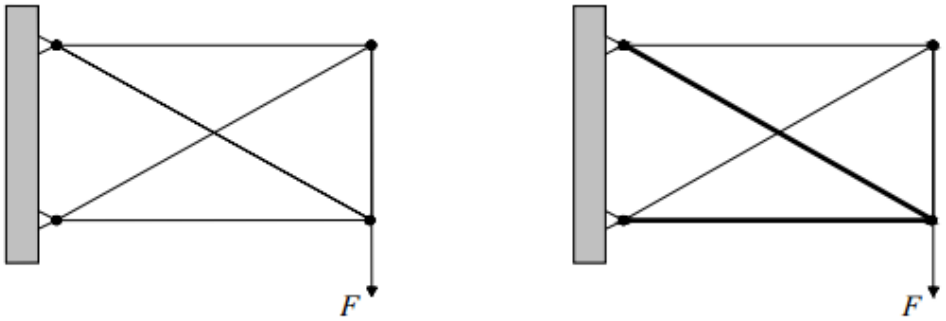
- Topoloji optimizasyonu

şeklinde üç kategoriye ayrılır.

1.2.2.1.1 Boyutlandırma Optimizasyonu

Boyutlandırma optimizasyonu problemlerinde elemanların kesit alanları ve atalet momentleri tasarım değişkenleri olarak ele alınır. Yapılar, ağırlığı minimum ve rijitliği ise maksimum olacak şekilde optimize edilir.

Basit bir kafes sistem için boyut optimizasyon problemi Şekil 1.2 de görülmektedir. Boyut optimizasyonu kavramı, en uygun tasarıma ulaşmak için yapı boyutlarının değiştirilmesini ifade etmektedir.

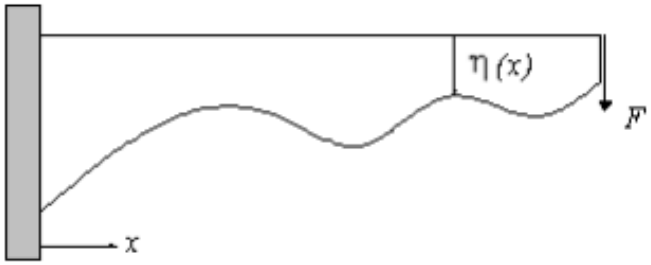


a) İlk tasarım **b)** Optimize edilmiş olan tasarım

Şekil 1.2 – Boyutlandırma optimizasyonu

1.2.2.1.2 Şekil Optimizasyon

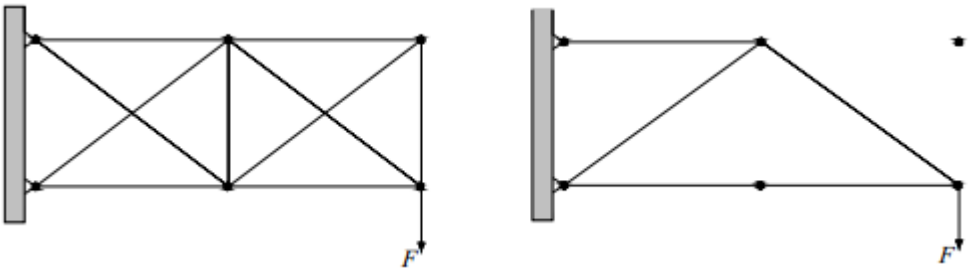
Şekil optimizasyon problemi belirlenen kısıtlamalar ile birlikte maliyeti minimum yapacak uygun şeklin aranmasıdır. Yapıların düğüm noktalarının koordinatları, kolon boyları veya kiriş açıklıkları tasarım parametreleri olarak düşünülebilir. İki boyutlu şekil optimizasyonu problemi Şekil 1.3 te görülmektedir.



Şekil 1.3 –Şekil optimizasyon problemi

1.2.2.1.3 Topoloji Optimizasyonu

Topoloji optimizasyonu problemlerinde, belirli bir tasarım alanında malzemenin en uygun şekilde dağılımı aranmaktadır. Bu optimizasyon uygulamasının temel mantığı, optimizasyonu yapılacak cismin dış boyutlarında herhangi bir değişiklik olmaksızın, parçanın rijitliğini artıracak şekilde belirli bölgelerden malzeme çıkarılması esasına dayanır. Tasarım parametreleri olarak; çerçevedeki açıklık sayısı, düğüm noktası sayısı ve kolon veya kiriş sayıları dikkate alınır.



a) İlk tasarım **b)** Optimize edilmiş olan tasarım

Şekil 1.4 –Topoloji optimizasyonu

1.2.2.2 Yapısal Optimizasyon Yöntemleri

Optimizasyon yöntemleri, amaç fonksiyonunun en iyi alternatif çözümünün seçilmesi için kullanılır. Bu yöntemler sayesinde daha fazla alternatif çözüm elde edilir ve bu şekilde maliyet - verim karşılaştırmasında daha iyi seçimler yapılabilir (Doğan, 2010).

1.2.2.2.1 Matematiksel Programlama

Bir optimizasyon problemi matematiksel olarak, değişkenlere atanan değerlere bağlı kalmak suretiyle belirlenen amaç fonksiyonun belli sınırlayıcılar kullanılarak en düşük veya en büyük değerinin bulunmaya çalışılması olarak ifade edilebilir. Optimizasyon probleminde kullanılan değerler tasarım uzayının koordinatlarıdır. Kısıtlayıcılar problemde değişkenlerin durumuna göre hesaplanan fonksiyonlar olup, çözümün geçerli olabilmesi için bu fonksiyonların aldığı değerlerin de belli sınırlar içerisinde olması gerekir. Matematiksel programlama, lineer programlama ve lineer olmayan programlama olarak ikiye ayrılır. Lineer programlamanın en dikkat çekici özelliği, amaç fonksiyonu ve kısıtlayıcıların, tasarım değişkenlerinin lineer bir kombinasyonu olarak ifade edilmesidir. Lineer programlama yöntemlerinin yapısal optimizasyon problemlerine uygulanabilmesi için amaç fonksiyonu ve kısıtlayıcıların tasarım parametreleriyle arasındaki ilişki lineer hale dönüştürülebilir. Çok değişkenli ve kısıtlayıcı problemlerin çözümüne yardımcı olur. Bu nedenle bazı lineer olmayan programlama problemleri lineer programlama yardımıyla çözülebilir. Lineer olmayan programlamada ise amaç fonksiyonu ve kısıtlayıcılar tasarım değişkenlerinin lineer olmayan bir fonksiyonudur (Mafrouz, 1999).

1.2.2.2.2 Optimumluk Kriteri Yöntemi

Optimumluk kriteri yöntemi problemin fiziki özelliklerini dikkate alarak, yapının davranışlarıyla ilgili bir kriter belirlemekte, kesin veya yaklaşık hesaba dayanan ifadelerle kurulmuş olabilmektedir. Tasarım bu şekilde boyutlandırıldığında, amaç fonksiyonu da aynı zamanda optimum değerini almış olur. Optimumluk kriteri yöntemi, tasarım değişkenlerinin sürekliliğine dayanmaktadır. Bu yöntemde kesikli tasarım değişkenlerinin kullanımının gerektiği durumda, iki adımlı yöntem kullanılmaktadır. İlk olarak, sürekli değişkenler kullanılarak optimizasyon problemi çözülür ve daha sonra sürekli değişkenler kullanılarak elde edilen değerler kesikli değişkenlerden en yakın değere yuvarlatılarak sonuç elde edilir.(Bayar, 2008).

1.2.3 Stokastik Optimizasyon Yöntemleri

Optimizasyon problemleri sabit olmayan birçok parametreye bağlıdır. Mühendislik alanında karşılaşılan moment kapasitesi, sürtünme kuvveti, atalet momenti ve malzeme parametreleri olan elastisite modülü, akma dayanımı, emniyet gerilmesi gibi parametrelerden söz edilebilir. Stokastik belirsizlikler nedeniyle, bu parametreler belirli bir olasılık dağılımına sahip rastgele değişkenler ile modellenmiş olmalıdır. Bu belirsizliklerden dolayı sıradan deterministik optimizasyon yöntemleri yerine stokastik optimizasyon yöntemlerinden yararlanılır (Kurt, 2005).

Aşağıda şimdiye kadar farklı araştırmacıların önerdiği optimizasyon algoritmaları yer almaktadır.

1.2.3.1 Yapay Arı Koloni Algoritması

Doğal yaşamda bal toplayan bir arı kolonisi içinde görev paylaşımı vardır. Bu görev paylaşımı ve kendi kendine organize olabilme yeteneği sürü zekâsının iki önemli özelliğidir. Tereshko tarafından ortaya atılmış reaktif difüzyon denklemine dayalı olarak ortaklaşa zekânın ortaya çıkmasını sağlayan minimal yiyecek arama modelinde üç temel bileşen vardır. Bunlar yiyecek kaynakları, görevli işçi arılar ve görevsiz işçi arılardır. Ayrıca bu model bir yiyecek kaynağına yönelme ve yiyecek kaynağını bırakma olmak üzere iki şekilde çalışmaktadır (Tereshko, 2000).

Kolonide arılar üç gruba ayrılır (Akay, 2009). Bunlar;

- İşçi Arılar: İşçi arılar daha fazla nektarın olduğu besin kaynaklarını araştırmakla görevlidirler. Her bir besin kaynağında bir işçi vardır. Dolayısıyla işçi arı sayısı besin kaynağı sayısına eşittir (Karaboğa, 2004).

- Gözcü Arılar: Gözcü arılar kovanda bekler ve diğer arıların besin kaynakları ile ilgili bilgileri paylaştıktan sonra nektarın fazla olduğu besin kaynağına yönelirler.

- Kâşif Arılar: Yiyecek arama sürecinin başlangıcında kâşif arılar rastgele dağılarak yiyecek aramaya başlarlar. Kaşif arıların sayısının kovandaki diğer arıların sayısına oranı %5-10 arasındadır (Karaboğa, 2011).

Yapay arı koloni algoritmasında yiyecek kaynakları, optimize edilmeye çalışılan problemin olası çözümlerine karşılık gelmektedir. Bir kaynağa ait nektar miktarı, o kaynakla ifade edilen çözümün kalite değerini ifade eder (Karaboğa, 2009). Bir yiyecek kaynağının değeri, yuvaya yakınlığı, çeşidi, nektar yoğunluğu, nektarın çıkarılmasının kolaylığı gibi birçok etkene bağlıdır.

Algoritmanın işleyişi beş temel adımda gerçekleşmektedir. Bu adımlar;

1- Bal kaynağı olan bölgelerin ilk değerlerinin belirlenmesi,
2- İşçi arıların belirlenen bal kaynaklarına yönlendirilmesi,
3- Bal kaynağı olmaya aday bölgeler için olasılık hesaplamalarının yapılması,
4- İşçi arıların aktardıklarına bağlı olarak, gözlemci arıların yeni bal kaynaklarını belirlenmesi,
5- Mevcut bal kaynaklarının kullanım dışı bırakılma kararının verilmesi şeklindedir.

Bu algoritmada yiyecek kaynaklarındaki nektar miktarı, kaynaklarla ilgili çözümlerin uygunluğuna ve bu kaynakların yerleri ise optimizasyon problemine ait olası çözümlere karşılık gelmektedir. Yapay arı koloni algoritması en fazla nektara sahip olan yiyecek kaynağının yerini

bulmaya çalışarak uzaydaki bütün aday çözümler içerisinden problemin minimumunu veya maksimumunu veren çözümü bulmaya çalışmaktadır (Karaboğa, 2011).

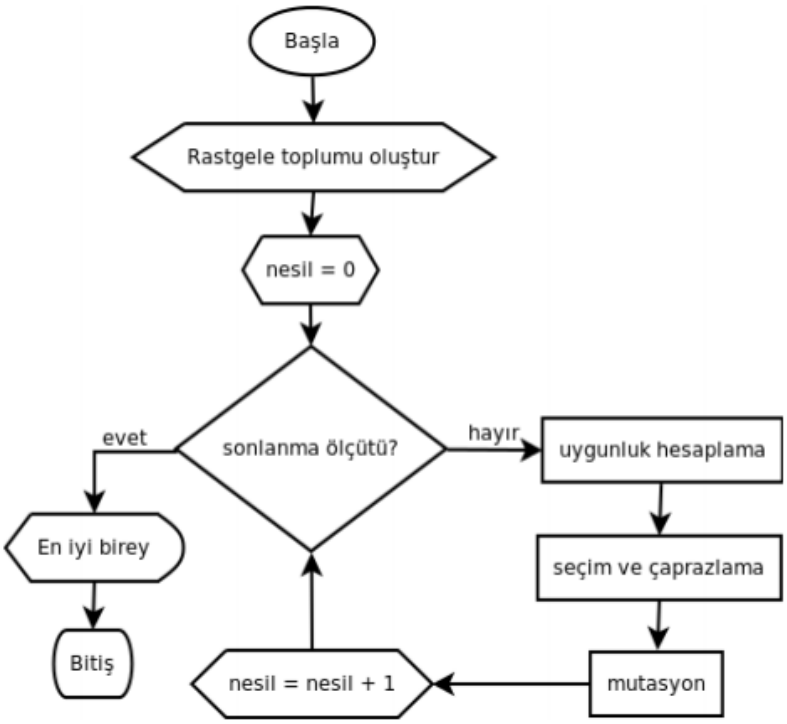
1.2.3.2 Benzetimli Tavlama

Benzetimli tavlama ismi, katıların fiziksel tavlanma süreci ile olan benzerlikten ileri gelmektedir. Bu yöntem, bir katının minimum enerji durumu elde edilene kadar yavaş yavaş soğutulduğu fiziksel tavlama sürecini taklit eden bir stokastik arama yöntemidir. Metallerde ısıl işlem uygulamasından yola çıkılarak geliştirilmiş bir yöntemdir. Yüksek sıcaklıklarda ısıtılarak sıvı hale getirilmiş metallerde atomlar serbestçe hareket edebiliyorken, metalin soğutulmaya başlanması ile birlikte atomların hareket edebilmeleri sınırlandırılmış olur. Uygun kristal yapı, ancak metali eritme sıcaklığı yeteri kadar yüksek ise ve soğutma da yeteri kadar yavaş yapılmış ise elde edilir. Aksi halde metal, istenilen kristal yapıya sahip olmayacaktır. Yani, kusurlu bir kristal yapı elde edilecektir. Soğutulan metalin homojen bir şekilde kristalleşmesi, bu yöntem ile çözülecek problemler için optimum sonucu ifade etmektedir (Kwang ve El-Sharkawi, 2008).

1.2.3.3 Genetik Algoritma

Genetik algoritmalar, yapay zekanın hızlı gelişen bir dalı olan evrimsel hesaplamanın alt alanlarındandır. Darwin'in evrim teorisinden yararlanılarak geliştirilmiştir. Bu algoritma John Holland tarafından ortaya atılmıştır (Holland, 1973).

Biyolojideki kromozom, gen çaprazlama ve mutasyon kavramları genetik algoritmada da yer almaktadır. Genetik algoritma parametreleri, biyolojideki genleri temsil ederken, parametrelerin toplu kümesi de kromozomu temsil etmektedir. Genetik algoritmanın her bir ferdi kromozomlar (bireyler) şeklinde ifade edilen populasyonlardan oluşur. Populasyonun uygunluğu, belirli kurallar içerisinde maksimize veya minimize edilir. Bu algoritmada her bireyin uygunluğu hesaplanarak bir sonraki nesile kalıp kalmayacağı belirlenir. Bir organizmanın yaşamda kalma veya genetik özelliklerini sonraki nesillere aktarıp aktarmamasıyla ilgilidir.



Şekil 1.5 –Genetik algoritma işletimi

Genetik algoritmalar parametre kodlama ilkesine dayanır. Bu algoritma rastgele bir çözüm kümesi seçer ve bu çözüm kümesi başlangıç populasyonu olarak kabul edilir. Her bir birey ikili semboller dizesi şeklinde oluşturulur ve 0 veya 1 değerlerini alacak şekilde kodlanır. Bu kodlama işleminden sonra elde edilen çözümlerin uygunluğu değerlendirilir. Uygunluk değerleri çözümün kalitesini belirlemektedir. Bu değer ne kadar yüksekse yaşama ve çoğalma şansı da aynı oranda fazladır. Evrimsel olarak bir sonraki nesil mevcut populasyon bireylerinden ortaya çıkmaktadır. Her nesilde çaprazlama ve mutasyon kavramları genetik algoritmada da yer almaktadır. Bu genetik operatörler kullanılarak yeni bir populasyon oluşturulur. Birkaç nesil sonunda, populasyon daha iyi uygunluk değerine sahip bireyler içerir. Daha güçlü bireyler iyi genlere sahiptir. Bu genlerin diğer nesillere geçme olasılığı seçim baskısı ile artırılarak, kötü genlere sahip olan güçsüz bireylerin evrim esnasında kaybolması sağlanır. Evrim süreci belirlenen bitiş koşulunu sağlayıncaya kadar sürer ve en son nüfustaki en iyi birey ulaşılan en iyi çözümü temsil eder (Holland, 1973).

Bu algoritmanın önemli bir özelliği, olasılık kurallarına göre çalışması nedeniyle, yalnızca amaç fonksiyonuna gereksinim duymasıdır. Çözüm uzayının tamamı yerine sadece belirli bir kısmı taranır. Bu şekilde etkili bir arama yapılarak çok kısa bir süre içerisinde çözüme ulaşılır (Goldberg, 1989).

Genetik algoritmanın avantajları (Haupt, 2004) aşağıdaki gibi listelenebilir:

- Ayrık veya sürekli değerleri iyileştirir.
- Türetilmiş bilgiye ihtiyaç duymaz.
- Geniş bir yüzeyde, geniş bir örneklemede eş zamanlı arama yapar.
- Çok sayıda değişkenle çalışabilir.
- Çok karmaşık uzaylarda da değişkenleri iyileştirebilir.
- Sadece tek bir çözüm değil, optimum birçok çözüm sağlar.
- Değişkenleri kodlayarak, optimizasyonun kodlanmış değişkenler üzerinde gerçekleştirilmesini destekler.
- Sayısal veri, deneysel veri veya analitik veri fonksiyonlarla çalışabilir.

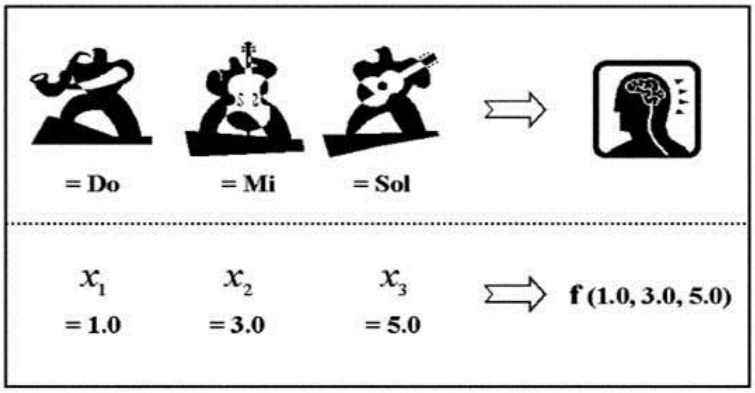
1.2.3.4 Harmoni Arama

Harmoni arama metodu ilk olarak Lee ve Geem (2004) tarafından ortaya atılmıştır. Bu sayısal optimizasyon tekniği, bir müzisyenin daha iyi bir müzikal uyum arayışı içinde uygulamaya çalıştığı müzikal performans sürecine benzetilmektedir. Örneğin jazz doğaçlaması, optimum çözüme ulaşmaya çalışan optimum tasarım sürecine benzer şekilde, müzikal açıdan tatmin edici uyumu bulmayı amaçlar.

Müzik geliştirme sırasında her bir sanatçı kendi enstrümanının akordunu yaparak mümkün olan ses aralıkları dâhilinde icra yapar ve hep birlikte bir harmoni vektörü oluştururlar. Oluşturulan akortlar güzel bir harmoni sağlamışsa her sanatçının hafızasında bu harmoni depolanır. Bu durumun mühendislik problemine uygulanması aşamasında; başlangıçta her bir değişkenin başlangıç değeri mümkün aralıklar dahilinde seçilir ve çözüm vektörü oluşturulur. İyi bir sonuç elde edilmişse değişkenin hafızasında depo edilir. En kötü harmoni hafızadan çıkarılır. Ayrıca her bir adımda bulunan sonuç geliştirilmeye çalışılır (Lee ve Geem, 2004).

Bir müzisyen harmoni oluştururken şu üç kuralı uygular:

1. Hafızasındaki bir akordu çalar.
2. Hafızasındaki akordun bir komşu akordunu kullanır.
3. Mümkün olan ses aralıkları dahilinde rastgele bir seçim yapar.



Şekil 1.6 - Müzik geliştirme ve mühendislik optimizasyonu arasındaki bağlantı (Lee ve Geem, 2005).

1.2.3.5 Tabu Arama

İnsan davranışlarının belli alanları ya da belli şartlarla ilişki kurularak kutsal veya dokunulmaz olarak tanımlanmış oldukça güçlü sosyal yasaklar tabu olarak adlandırılmaktadır. Tabu, bir optimizasyon problemindeki yasakları ve kısıtları temsil etmektedir. Tabu aramanın esasları çok önceleri verilmesine rağmen, bugün kullanılan şekli Glover tarafından ortaya konmuştur (Glover, 1997).

Bu yöntemde; optimum çözüme ulaşmak için çözüm uzayını hareket mekanizmasıyla araştıran ve sahip olduğu yapay hafıza özelliği ile önceki tasarımları üretmeyerek lokal optimumlardan kurtulan gelişmiş bir arama yöntemidir. Yakın çözümlerin bulunduğu kümede arama yapılır. Amaç fonksiyonunu (kar veya maliyet) minimize ya da maksimize edecek uygun çözümlerin bulunmasına yardımcı olur. En iyi komşu tasarımların elde edilmesini sağlayan hareketler yapay bir hafızaya aktarılır. Hafızaya kaydedilen bu hareketler içinde tekrar işlem yapılarak sürecin uzamasının önüne geçmek için bazı hareketler yasaklanır. Hafızada yasaklanmış bu hareketler tabu olarak belirlenir. Yasaklanan her bir hareket, problemde bir tasarım değişkenine karşılık gelir. Tabu aramada her tasarım değişkeni için tanımlanan komşu tasarımlar içerisinde en iyi olan tasarım, yasaklar listesi de kontrol edilerek mevcut tasarımın yerini alır. Bu işlemden sonra mevcut olan tasarım yerine belirlenen yeni en iyi tasarım ele alınarak devam edilir. Aynı işlem diğer tasarım değişkenleri için de tekrarlanarak bir iterasyon adımı tanımlanır. Kabul edilen durdurma kriteri sağlanıncaya kadar her iterasyon adımı için bu işlemler tekrarlanır (Değertekin ve diğerleri, 2006)

1.2.3.6 Av Arama Optimizasyonu

Aslan, kurt ve yunus gibi hayvanların yiyecek bulma amacıyla grup şeklinde gerçekleştirdikleri avlanma davranışlarının taklit edilmeleri sonucunda ortaya çıkmış, olasılığa dayalı bir

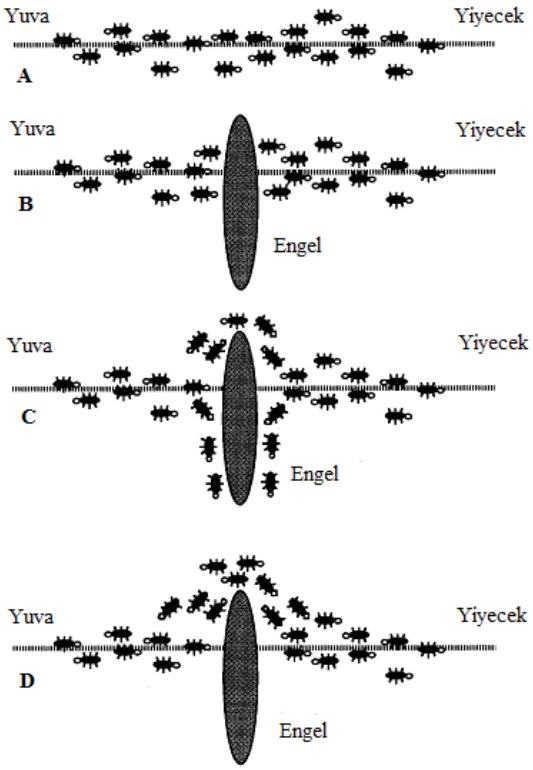
optimizasyon tekniğidir. Avlanma grubu içinde yer alan avcılar, avlarını yakalamak için çeşitli stratejiler yürütürler. Örneklendirmek gerekirse kurtlar ve aslanlar bu şekilde bazen kendilerinden çok hızlı ya da çok daha büyük hayvanları sürü halinde hareket ederek yakalayabilmektedirler. Bir av seçildikten sonra bütün avcı grubu yavaş yavaş bu hayvana doğru yaklaşır. Kokularının av tarafından fark edilmeyeceği şekilde hareket ederler. Her bir bireyin sürü içerisindeki konumu, hızı gibi parametreler avın en kısa sürede yakalanabilmesi için önemlidir (Oftadeh ve diğerleri, 2010).

Geliştirilen algoritmada, avcı hayvanların her biri, problemin bir parçasının çözümüne karşılık gelmektedir. Optimum tasarım işleminde, bir avı yakalamak amacıyla avcı hayvanların biraraya gelmesine benzer şekilde, en iyi çözümün bulunması için araştırma yapılır. Her bir iterasyonda avcıların pozisyonu güncellenerek amaç fonksiyonunun iyileştirilmesi hedeflenir. Bir avcı grubu, grup boyutunu temsil eden ve önceden belirlenmiş miktarda avcıdan oluşur. Bu amaçla öncelikle grup içindeki her bir avcı için bütün ilk pozisyonlar rastgele başlatma prensibiyle oluşturulur. Bu işlem için değeri 0 ile 1 arasında değişen sayılar içinden rastgele sayı seçme uygulamasından faydalanılır.

1.2.3.7 Karınca Koloni Optimizasyonu

Gerçek karınca koloni davranışlarının matematiksel modelleri üzerine dayalı bir algoritmadır. Bu yöntem, karıncaların doğal davranışlarını taklit ederek sürekli ve süreksiz problemleri çözmek için kullanılır. İlk çalışma, Marco Dorigo ve arkadaşları tarafından 1991 yılında ortaya konulmuştur (Dorigo ve diğerleri, 1991). Karınca koloni optimizasyonu; karıncaların besin kaynakları ile yuvaları arasında kullandıkları yolların izlenmesi sonucunda ortaya çıkmış bir optimizasyon tekniğidir.

Karıncalar üzerinde yapılan çalışmalarda, birçok karınca türünün görme yeteneklerinin olmadığı tespit edilmiştir. Yine yapılan çalışmalarda karıncaların birbirleri ile iletişimlerinin vücutlarında üretilen ve feromen (pheromone) olarak isimlendirilen kimyasal salgılar vasıtasıyla yapıldığı anlaşılmıştır. Karınca çevre şartlarına göre besin kaynağı ile evi arasında gidebileceği yolları belirlemektedir. Koloni halinde yaşayan karıncalar yiyecek bulmak için besin kaynağına öncelikle aralarından bir karıncayı, öncü karınca olarak gönderirler. Gönderilen bu karınca çevreyi araştırarak uygun yiyecek kaynağını bulmaya çalışır. Karıncalar ilerlerken, belirli bir miktar feromen depo ederler. Depo ettikleri feromenleri, gıdaya giderken seçtikleri yola bırakarak, kendilerinden sonraki karıncalara yol seçiminde yardımcı olurlar. Öncü karıncaların bulmuş olduğu ilk yol, en kısa yol olmayabilir. Karıncalar birçok yol arasından birini seçmek durumunda kaldıklarında iz miktarının yoğun olduğu yolu tercih ederler. Besin kaynağına giden en kısa yol zamanla çevre şartlarından dolayı en kısa yol olmaktan çıkarsa, koloni üyeleri yeni bir en kısa yol bulmak üzere çalışmalarına başlarlar (Karaboğa, 2011).



Şekil 1.7 – Besin kaynağına giden karıncaların en kısa yolu bulma aşamaları

Dorigo ve arkadaşları, kendi sistemlerini "karınca sistemi", ortaya çıkan algoritmayı ise "karınca algoritması" olarak tanımlamışlardır. Bu sistemde gerçek karıncaların davranışlarından biraz farklı yapay karıncalar tasarlanmıştır. Örneğin, yapay karıncalar belirli bir hafızaya sahiptirler ve tamamen kör değillerdir. Bu modelde, karıncaların yiyecek aramaları problemin çözüm kümesini, yiyeceğe ulaşan yol amaç fonksiyonunu, feromen madde ise hafıza kavramını temsil eder (Dorigo ve diğerleri, 1991).

1.2.3.8 Parçacık Sürü Optimizasyonu

Sezgisel yöntemlerden biri olan parçacık sürü optimizasyon tekniği ilk olarak kuş ve balık sürülerinin hareketlerinden esinlenerek doğrusal olmayan nümerik problemlere en iyi sonuçlar bulmak için 1995–1996 yıllarında sosyolog-psikolog James Kennedy ve elektrik mühendisi Russel Eberhart tarafından ortaya atılmıştır (Kennedy ve Eberhart, 1995).

Parçacık sürü kavramı basitleştirilmiş sosyal sisteme benzetilerek ortaya çıkmıştır. Parçacık sürü optimizasyon tekniği, kuşların gökyüzünde sürü halinde uçarlarken yem bulma hareketlerine benzetilerek ortaya çıkmış bir arama yöntemidir. Kuş toplulukları gerçek yiyecek kaynağını bilmemelerine rağmen, yiyecek kaynağından ne kadar uzakta olduklarını öğrenmeye

çalışırlar. Öğrenmek için izlenen yöntem yiyecek kaynağına en yakın olan kuşu izlemektir. Bu teknikte rastgele çözümlerin bulunduğu bir başlangıç popülasyonu ortaya atılır ve optimum çözüm bulunmaya çalışılır. Kuşlar yiyecek ararken kendine en yakın kuşu takip ederler. Parçacık sürü optimizasyonunda her bir kuş parçacık olarak, kuş topluluğu da sürü olarak temsil edilir. Parçacık hareket ettiğinde, kendi koordinatlarını bir fonksiyona gönderir: Bu şekilde parçacığın uygunluk değeri ölçülmüş olur. Parçacık, optimumu ararken hızını ve uygunluk değerlerini hafızada kayıt altında tutmalıdır. Çözüm uzayında her boyuttaki hızının ve yönünün her defasında nasıl değişeceği, komşularının en iyi koordinatları ve kendi kişisel en iyi koordinatlarının birleşiminden yola çıkılarak belirlenir.

1.2.4 Kısıtları İşleme Yöntemi

Amaç fonksiyonları içinde kısıtlamaları içeren optimizasyon arama teknikleri vardır (Carlos ve Coello, 2002). Bu teknikler içerisinde ceza fonksiyonu kısıtlamaları birleştirmede sıkça kullanılmaktadır. Geri uçuş mekanizması da bu yöntemlere eklenmiştir. Bu teknikler arama uzayında hem uygun hem de uygun olmayan çözümlerin bulunduğu bölgeyi kullanmaktadır.

1.2.4.1 Ceza Fonksiyonları Yöntemi

Ceza fonksiyonları yöntemi, amaç fonksiyonuna belli bir değeri ekleyerek veya çıkararak kısıtlı optimizasyon problemini kısıtsız hale çevirme esasına dayanır. Bu şekilde, amaç fonksiyonuna kısıtlamaların ihlali durumunda yüksek bir maliyet getiren bir terimin eklenmesi ile gerçekleşmektedir (Carlos ve Coello, 2002). Bu yöntemde, uygun çözüm bölgesi genişletilir, fakat orijinal uygun çözüm bölgesi dışında yer alan noktalar için amaç fonksiyonuna yüksek bir maliyet getiren ceza terimi eklenir. Bu sebeple bu şekildeki yöntemlere dış ceza metodu da denir. Dış ceza metodunun genel hali aşağıda verilmiştir.

$$\phi(x) = f(x) \pm \left[\sum_{i=1}^{n} r_i . G_i + \sum_{j=1}^{p} c_j . L_j \right] \tag{1.8}$$

$\phi(x)$ genişletilmiş amaç fonksiyonu, G_i ve L_j kısıt fonksiyonları, r_i ve c_i ise ceza faktörleri olan sabitleri ifade etmektedir. Kısıt fonksiyonları;

$$G_i = \max\left[0, g_i(x)^{\beta}\right] \tag{1.9}$$

$$L_j = \left|h_j(x)\right|^{\gamma} \tag{1.10}$$

denklemleri ile ifade edilmektedir. Burada $g_i(x)$ ve $h_j(x)$ sabitleri gösterirken, γ ve β ise 1 ya da 2 değerlerini almaktadır.

1.2.4.2 Geri Uçuş Mekanizması

Bu mekanizma ilk olarak parçacık sürü optimizasyonu tekniğinin kullanıldığı optimizasyon problemlerindeki kısıtlamaların kontrolü için kullanılmaya başlamıştır. Problemin çözüm bölgesinden dışarı çıkıldığında, bir önceki pozisyona dönülür. Bu nedenle bu yönteme geri uçuş mekanizması adı verilmektedir. Çözüme uygulanabilir bir tasarımla başlanır. Yeni tasarım sınırlayıcılarının dışına çıkılması durumunda önceki tasarıma geri dönülür. Bu tekniğe dayanan algoritmalar, ceza fonksiyonuna dayanan diğer algoritmalara göre daha güvenilirlerdir (He ve diğerleri, 2004).

1.3 Literatür Araştırması

Saka (1998), ızgara sistemlerin genetik algoritma yardımıyla optimum tasarımını yapmıştır. Algoritmanın etkinliğini ve eğilme etkisini incelemek için 40 elemanlı ızgara sisteminin minimum ağırlık optimum tasarımını elde etmiştir.

Saka ve diğerleri (2000), optimum aralığın ızgara sistemlerin minimum ağırlık tasarımında önemli etkiye sahip olduğunu ortaya koymuşlardır. Bu etkiyi genetik algoritma yöntemi kullanılarak incelemişlerdir.

Erdal (2007), harmoni arama yöntemini kullanarak çelik ızgara sistemlerin optimum tasarımını LRFD-AISC yönetmeliğine göre yapmıştır. Yöntemin etkinliğini göstermek için, 3 elemanlı ankastre ızgara sistem, 23 elemanlı ve 40 elemanlı basit ızgara sistem örneği sunulmuştur.

Saka ve Erdal (2008), ızgara sisteminin boyuna ve enine kirişleri arasındaki aralığın minimum ağırlık tasarımında önemli bir etkiye sahip olduğunu belirtmişlerdir. Bu etkiyi harmoni arama yöntemi kullanarak incelemişlerdir.

Saka ve Erdal (2009), harmoni arama olarak bilinen olasılığa dayalı optimizasyon metodu yardımıyla ızgara sistem tasarlamışlardır. Izgara sistemin optimum tasarımı LRFD-AISC şartnamesinde bulunan sınırlayıcıların dikkate alınması suretiyle gerçekleştirmişlerdir.

Lukasik ve Zak (2009), ateş böceği algoritmasından ve bu algoritmanın teknik detaylarından bahsetmişlerdir. Sayısal deneylere ve parametre çalışmalarına yer vermişlerdir. Ayrıca ateş böceği algoritmasının ve parçacık sürü optimizasyonunun performanslarını karşılaştırmışlardır.

Yang (2009), ateş böceği algoritmasının teknik özelliklerinden bahsetmiştir. Parçacık sürü optimizasyonu ve genetik algoritma yöntemleriyle ateş böceği algoritmasını karşılaştırmıştır.

Yang (2010), doğrusal olmayan tasarım problemlerini çözmek için geliştirdiği ateş böceği algoritmasının nasıl kullanılacağını göstermektedir.

Gandomi ve diğerleri (2011), üstsezgisel bir algoritma olan ateş böceği algoritmasını sürekli veya kesikli yapısal optimizasyon problemleri için kullanmışlardır. Literatürde yer alan altı yapısal optimizasyon problemleri üzerinde yapılan çalışmaların sonuçlarıyla ateş böceği algoritmasının geçerliliğini doğrulamışlardır.

Doğan ve diğerleri (2011), LRFD-AISC' de belirlenen tasarım sınırlayıcılarını dikkate alan çelik ızgara sistemlerin minimum ağırlığını elde etmek için parçacık küme yöntemini kullanmışlardır. Parçacık küme algoritmasının etkinliği örnek problemlerle göstermişlerdir.

Farahani ve diğerleri (2011), yönlendirilmiş hareketle geliştirilmiş ateş böceği algoritmasını beş standart fonksiyon üzerinde test etmişlerdir. Deney sonuçları, geliştirilen ateş böceği algoritmasının standart ateş böceğine göre daha iyi performansa ve daha fazla doğruluk payına sahip olduğunu göstermektedir.

Khadwilard ve diğerleri (2011), ateş böceği algoritması için parametre çalışması yapmışlardır. Çeşitli parametreler kullanılarak ateş böceği algoritmasının performansını analiz etmişlerdir.

Doğan ve diğerleri (2013), çelik ızgara sistemlerinin minimum ağırlık tasarımında eleman gruplandırmasındaki önemini vurgulamışlardır. Eleman gruplandırmasında ızgara sisteminin minimum ağırlığa olan etkisini parçacık küme ve armoni arama yöntemleriyle incelemişlerdir. 40 elemanlı basit ızgara sistem 1 grup, 2 grup, 4 grup ve 8 grup olarak ayrı ayrı analiz ederek optimizasyon yöntemlerinin optimum tasarıma olan etkisini belirtmişlerdir.

Erdal ve diğerleri (2013), parçacık sürü tabanlı optimizasyon, geliştirilmiş optimum tasarım algoritması versiyonu için çelik ızgara sistemlerle örneklendirmişlerdir.

Yang ve He (2013), ateş böceği algoritmasının temellerinden söz etmişlerdir. Kesikli arama yöntemleriyle karşılaştırarak ateş böceği algoritmasının optimum kesikli arama yöntemlerinden daha iyi olduğu sonucuna varmışlardır. Ayrıca yüksek boyutlu optimizasyon problemleri için karşılaştırmalar yapmışlardır.

2. ATEŞ BÖCEĞİ SÜRÜ OPTİMİZASYONU

2.1 Giriş

Ateş böceği (Lampyridae)**;** böcekler sınıfının kınkanatlılar takımının bir familyasıdır. Kınkanatlıların çoğunun tersine kanat örtüleri, daha doğrusu kalınlaşmış ön kanatları yumuşaktır. Bedeni uzunca, rengi gri ya da kahverengidir. Bütün ateş böcekleri tek cinstir, bu sebepten dolayı erkek ve dişi ayrımı yoktur. Bu böcek adını, yaydığı parlak sarı-yeşil ışıktan alır. Işıkları yaklaşık 45 metreden görülebilir. Kurtçukları etçildir; öteki böcekleri yiyerek beslendiklerinden yararlı sayılırlar. Erginleri ise çiçekler ve küçük bitkiler üzerinde yaşarlar.

Her türün kendisine özgü sinyal şifresi vardır. Işık üretim organları karın bölümünün son kısmında bulunur. Saydam bir kütüküla tabakası ile örtülüdür. İç kısmı fotojenik hücreler ve otomobil farları gibi ışığı yansıtıcı bir tabakadan oluşur. Işık organında üretilen yağa benzer lüsiferin maddesi lüsiferinaz enziminin yardımı ile kademeli olarak oksijenle yakılır. Bu kimyasal olayda ışık ya da foton salınımı olur. Havadaki oksijenin kontrollü tüketimine bağlı olarak ışık zaman zaman yanıp söner. Bu yanıp sönmeler eşlerin birbiriyle haberleşmesini sağlar. Ateş böceğinin ürettiği ışık, yavaş yavaş meydana gelen oksitlenme sonucu kimyasal enerjinin ışığa dönüşmesidir. Çıkan ışık tamamen soğuktur. Isı kaybı yoktur. Bu reaksiyonlardaki kimyasal enerjiden ışık-foton oluşturulur. Genellikle soğuk kimyasal reaksiyonlardır ve şu şekilde gerçekleşirler:

$$\text{Oksijen } (O_2) + \text{Lüsiferin} \rightarrow \text{Oksilüsiferin} + \text{Karbondioksit } (CO_2) + \text{forton salınımı}$$

$$\text{Fotoprotein} + Ca^{++} \rightarrow \text{Protein-koelentirmid} + \text{Karbondioksit } (CO_2) + \text{forton salınımı}$$

Normal elektrik ampulleri %10 verimle çalışırlar, %90'ı ise ısı olarak açığa çıkar. Buna karşın ateş böcekleri %100'lük bir verimle ışık üretirler. Ateş böceklerinin yaydıkları ışığın en önemli özelliği, ateşle ve sıcaklıkla ilgisinin bulunmamasıdır. Günümüzde aydınlatma teknolojisinin ulaşmaya çalıştığı bir hedef, ateş böceğindeki bu sistemdir. Normal bir ampul kendisine verilen enerjinin yüzde 4'ünü, florasan ampul ise yüzde 10'unu ışığa dönüştürür. Ateş böcekleri ise yüzde 100 verimle ışık üretir. Ateş böceğinin ışığı, hemen hiç ısı çıkmadan oluşur. Sıradan bir ampulde etkinlik derecesi ancak yüzde 3'dür ve yüzde 97'si kaybolur. Bu derece, ateş böceğinde yüzde 90'dır. Bu böceklerin insanlar tarafından parlak bulunmasının nedeni, ışığı insan gözünün çok duyarlı olduğu bir dalga boyunda yaymalarıdır. Bazı türlerin bir insanın kitap okumasına yetecek kadar ışık yayabildiği bilinmektedir.



Şekil 2.1 – Işık saçan ateş böcekleri

Ateş böcekleri genellikle kısa aralıklarla yanıp sönen bir ışık saçar; bu ışığın yanıp sönme ritmi, ateş böceklerinin buluşmasını sağlayan işaret sisteminin bir parçası ve ateş böceklerini öbür ışık saçan böceklerden ayırt eden bir özelliktir. Işık saçmasının hızı, sıklığı ve ateş böceklerinin birbirlerine yanıt vermesinden önce geçen süre özel anlamlar taşır. Haberleşmek ve çiftleşme mesajı verebilmek için bu ışıkları kullanan ateş böceklerinde türe göre ışıldama uzunluğu değişir. Çiftleşme gerçekleştikten sonra ateş böcekleri yumurtalarını bir kayanın ya da bitki örtüsünün altına bırakır. Beş hafta sonra, yumurtalar kırılır ve larvalar çıkar. Ateş böceklerinin başka bir özelliği de hayatlarının her safhasında gelişme halinde olmalarıdır. Bazı türler ise ışıklarını kendilerini savunmak için kullanır. Saçtıkları ışık aynı zamanda tatlarının kötü olduğu mesajını da iletir. Ateş böceklerinin yaşayış şekillerini belirleyen bu mantık temel alınarak geliştirilmiş olan optimizasyon metodu ateş böceği algoritması olarak adlandırılır.

2.2 Ateş Böceği Algoritması

Stokastik sayısal optimizasyon arama yöntemlerinden biri olan ateş böceği yöntemi, K. N. Krishnanand ve D. Ghose tarafından 2005 yılında geliştirilmiştir (Krishnanand ve Ghose, 2005). Çok modelli fonksiyonları optimize etmek için önerilen populasyon tabanlı sezgisel bir algoritmadır.

Ateş böceği sürü optimizasyon yöntemi, ateş böceklerinin sosyal davranışları gözlemlenerek ve taklit edilerek geliştirilmiştir. Bu yöntemi kullanmanın temel amacı tüm yerel maksimumları yakalamayı sağlamaktır. Çok modelli fonksiyon optimizasyon problemlerinde, ateş böceği sürü optimizasyonu ve önceki yaklaşımlar arasındaki en önemli fark, birden çok zirveyi verimli bir

şekilde yerleştiren sürüdeki bireylerin kullandığı dinamik karar alanıdır. Sürüdeki her bir birey komşularını seçmek için karar alanını kullanmaktadır ve komşularından aldığı sinyal gücüyle hareketlerini belirlemektedir. Ateş böceğinin eşlerini veya avlarını çekmek için kullandığı ışığının yanıp sönmesine neden olan ışık üreten hücreler oksijen ve "lusiferaz" adlı bir kimyasalla reaksiyona giren "lusiferin" adlı bir kimyasaldır. Bir ateş böceği ne kadar parlak olursa diğer ateş böcekleri için o kadar çekici olmaktadır.

Ateş böceği algoritması, Dr. Xin-She Yang tarafından geliştirilen ve tropikal iklim bölgelerindeki ateş böceklerinin sosyal davranışlarını baz alan bir metasezgisel optimizasyon algoritmasıdır (Yang, 2009). Bu algoritma diğer sürü zekâsı tabanlı algoritmalarla birçok benzerliği bulunmasına rağmen kavram ve uygulamada daha basittir. Bir ateş böceğinin ışıklarını yakıp söndürmesinin birinci amacı, diğer ateş böceklerini çekmek için bir sinyal sistemi olarak kullanmasıdır. Araştırmacılar, yanıp sönen ışıkların ateş böceğine, arkadaşlarını bulmada, olası avlarını çekmede ve avcılarından kendilerini korumada yardımcı olduğunu düşünmektedirler. Algoritmanın adlandırılması, algoritmanın oluşturulmasında model alınan ateş böceklerinden gelmektedir. Ateş böceklerinin tek cins olması ve birbirlerini çekmeleri algoritmanın temelini oluşturur.

Ateş böceği algoritmasında, verimli optimum çözümler elde etmek için, verilen bir optimizasyon probleminin amaç fonksiyonu, ateş böceği sürüsüne parlak ve daha çekici yerlere gitmede yardım eden yanıp sönen ışık ya da ışık şiddeti ile ilişkili olmaktadır. Bütün ateş böcekleri tek cins olarak kabul edilmektedir. Dolayısıyla erkek ve dişi ayrımı olmadığı gibi, bütün ateş böcekleri diğer ateş böcekleri tarafından cezbedilebilir. Ateş böceklerinin çekiciliği, parlaklıkları ile doğru orantılıdır. Yani bir ateş böceği ne kadar parlak olursa diğer ateş böcekleri için o kadar çekici hale gelmektedir. Ateş böceklerinin aralarındaki mesafe, parlaklığı azalttığı için cazibeyi de azaltmaktadır. Ortamda bir ateş böceğinden daha parlak ateş böceği varsa, bu ateş böceği parlak olana doğru hareket edecektir (Apostolopoulos, Vlachos, 2011; Yang, 2010). Eğer ortamda daha parlak ateş böceği yoksa rastgele yönlerde hareket ederler. Ateş böceği algoritmasında, optimum çözümler elde etmek için verilen bir optimizasyon probleminin amaç fonksiyonu, ateş böceği sürüsüne parlak ve daha çekici yerlere gitmede yardım eden yanıp sönen ışık ya da ışık şiddeti ile ilişkili olmaktadır (Gandomi ve diğerleri, 2011).

Geliştirilmiş olan ateş böceği optimizasyon algoritmasının işleyiş adımlarını aşağıda gösterildiği şekilde özetlemek mümkündür.

- Başlangıçta belli sayıdaki ateş böcekleri uygun arama uzayında yayılmak için rastgele üretilir. Her ateş böceği kendine ait bir pozisyona sahiptir. Ateş böceği algoritmasında,

her bir ateş böceği ilk olarak rastgele konumlara yerleştirilir. Bu başlangıç konumu denklemi 2.1 de verilmiştir.

$$x_i^0 = X_{\min} + r\left(X_{\max} - X_{\min}\right) \tag{2.1}$$

buradaki r değerinin tespiti için 0 ile 1 arasında değişen sayılar içinden rastgele sayı seçme uygulamasından faydalanılır. $X_{\min}$ 1 e eşittir ve $X_{\max}$ değişken seti içindeki toplam değer sayısıdır.

- Tasarım uzayında her x_i^t konum değeri için $f\left(x_i^t\right)$ amaç fonksiyonu değerleri belirlenir.
- i numaralı ateş böceğinin t numaralı iterasyon adımına kadar elde edilen en iyi konumu a_i^t ve ateş böceğinin global optimum konumu a_g^t güncellenir.
- Her bir ateş böceğinin konumu güncellenir. Bir ateş böceğinin o ana kadar elde edilen en iyi konum değeri, o parçacığın en iyi konum değeri olarak tayin edilir ve her ateş böceği için ayrı bir vektör içinde depo edilir. Belirlenen bu konum, optimizasyon sürecinin başından bu yana küme içerisindeki herhangi bir ateş böceği tarafından elde edilen en iyi konum ise bütün ateş böceklerinin en iyi pozisyon değeri olarak adlandırılır ve yine bir vektör içine kaydedilir. Ateş böceklerinin $t+1$'deki konumu, denklem 2.2 (Yang, 2013) yardımıyla hesaplanır.

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j^t - x_i^t\right) + \alpha_t \varepsilon_i^t \tag{2.2}$$

Buradaki x_i^{t+1} ifadesi $i.$nci ateş böceğinin $t+1$ nci adımdaki konumunu belirtmektedir.

Ateş Böceği Algoritması

begin
Amaç fonsiyonu $f(x), \quad x=(x_1,...,x_d)^T$
Ateş böceklerinin başlangıç popülasyonu $x_i (i=1,2,...,n)$
x_i *deki ışık şiddeti* I_i $f(x_i)$ *tarafından belirlenir.*
γ *ışık emme katsayısı*
while *(o andaki iterasyon sayısı = t < maksimum iterasyon)*
for $i=1:n$ *(bütün ateş böcekleri n)*
for $j=1:i$ *(bütün ateş böcekleri n)*
if $(I_j > I_i)$
L'evy flights vasıtasıyla j den i ye doğru hareket eder.
end if
$exp\left[-\gamma r^2\right]$ *aracılığıyla r mesafesiyle çekicilik değişir.*
Yeni sonuçları değerlendirme ve ışık şiddetini yenileme
end for j
end for i
Ateş böceklerinde kademe ve mevcut en iyiyi bulmak
end while
İşlem sonrası sonuçlar ve görselleştirme
end

Şekil 2.2 –Ateş böceği optimizasyon algoritması

Ateş böceği algoritmasında, iki önemli husus vardır. Bunlar; ışık şiddetinin değişimi ve çekiciliğin formülasyonudur. Kolaylık olması açısından ateş böceğinin kendi çekiciliğini kodlanmış amaç fonksiyonu ile ilişkilendirilmiş parlaklıkla belirlediği varsayabilir (Gandomi diğerleri, 2011).

Ateş böceklerinin aralarındaki mesafe, parlaklığı azalttığı için cazibeyi de azaltmaktadır. Bu değer bir çok uygulamada Gaussian formu kullanılarak tahmin edilebilir (Yang 2008, Yang 2009).

$$I = I_0 e^{-\gamma r^2} \tag{2.3}$$

Burada I ışık yoğunluğunu (parlaklık), I_0 başlangıç ışık yoğunluğunu ve γ ışık yoğunluğu katsayısını ifade etmektedir. Işık yoğunluğu r mesafesine bağlı olarak değişir.

Denklem 2.4'te yer alan β çekicilik olarak tanımlanır (Yang 2008, Yang 2009).

$$\beta = \beta_0 e^{-\gamma r^2} \tag{2.4}$$

$r = 0$ daki çekicilik β_0 ile gösterilmektedir.

i ve j noktalarındaki her iki ateş böceğinin arasındaki kartezyen mesafe;

$$r_{ij} = \left\| x_i - x_j \right\| \tag{2.5}$$

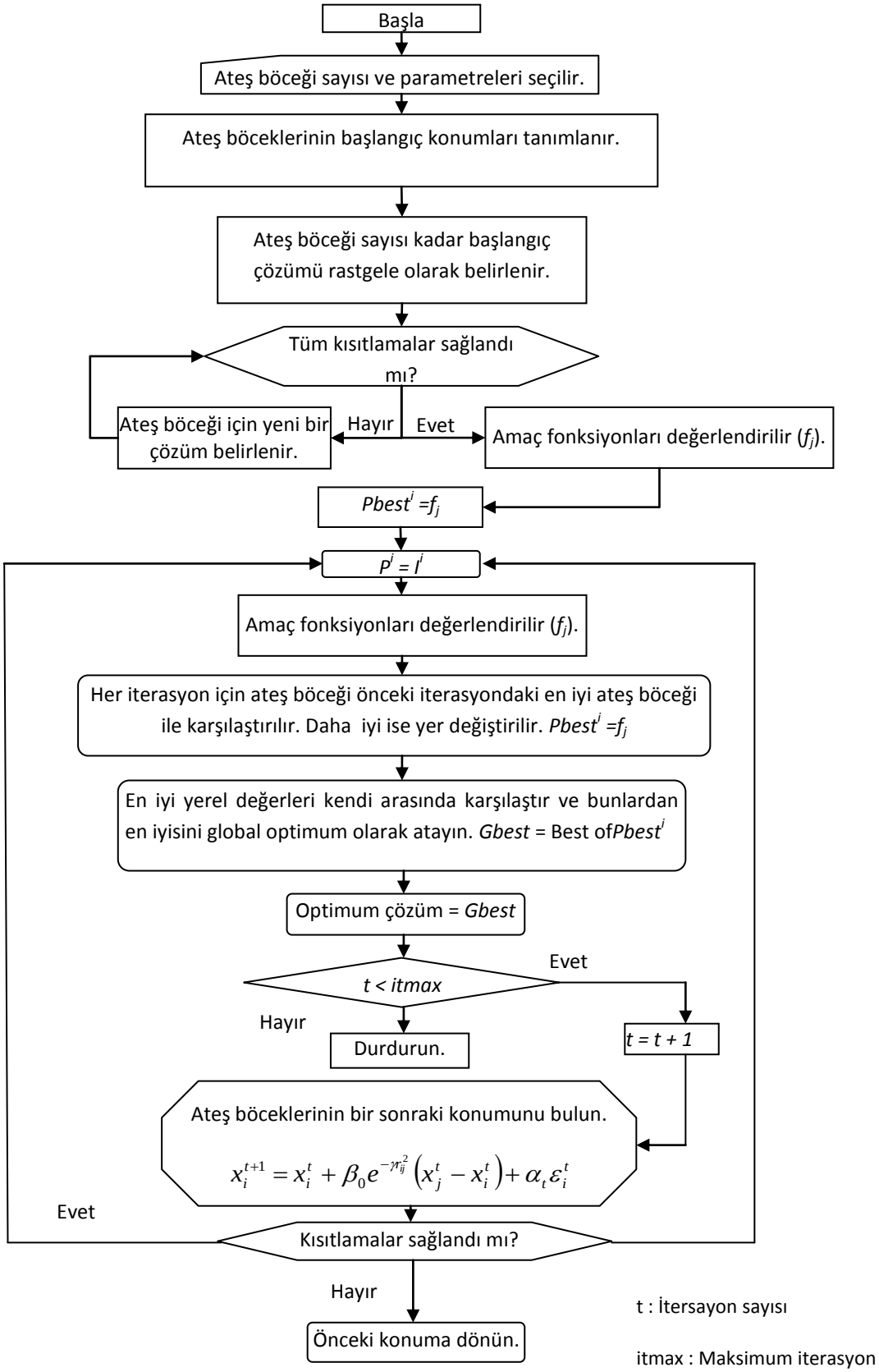
ile hesaplanırken, her bir ateş böceğinin iterasyonlar içerisinde ulaşacağı yeni konumu aşağıdaki denkleme göre belirlenir.

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2} \left(x_j^t - x_i^t \right) + \alpha_t \varepsilon_i^t \tag{2.6}$$

Denklem 2.6'daki ikinci terim çekicilikle ilgilidir. İkinci terim ise rastgele sıralama yapmak için kullanılır.

Burada α , $[0,1]$ (Lukasik ve Zak, 2009) (Babu, 2002) arasında değişen rastgele bir sayıdır. α ise rastgele sıralama parametresi olup, ε 'un ateş böceğinin yeni konumunun belirlenmesindeki ağırlığını ifade eden bir parametredir.

Ateş böceği optimizasyon algoritmasının akış şeması Şekil 2.3'te gösterilmiştir.



Şekil 2.3 –Ateş böceği optimizasyon algoritması için akış şeması

2.3 Ateş Böceği Algoritması Yönteminin Uygulaması

Değişkenlerin gerçek değerler aldığı fonksiyonu optimize etmek üzere çözümü aranan problemler sürekli optimizasyon problemleri olarak tanımlanabilir. Fonksiyonların her biri tasarım uzayında bir noktaya karşılık gelmektedir. Sürekli optimizasyon problemlerinde karar değişkeni, kısıt fonksiyonları ile sınırlanan bir bölgede tüm değerleri alabilmektedir. Aşağıda verilen sayısal optimizasyon problemleri literatürde yeni üretilen optimizasyon yöntemlerinin etkinliğinin test edilmesi amacıyla kullanılmaktadır. Bu problemler benchmark problemleri olarak adlandırılmakta olup sürekli değişkenler içermektedir. Çalışmanın bu bölümünde bu problemlerden beş tanesi seçilmek suretiyle ateşböceği algoritmasının ürettiği optimum çözümün diğer optimizasyon tekniklerinden elde edilen çözümlerle karşılaştırması hedeflenmiştir.

2.3.1 Örnek 1

Amaç fonksiyonu,

$$f(x) = 5\sum_{i=1}^{4} x_i - 5\sum_{i=1}^{4} x_i^2 - \sum_{i=1}^{13} x_i \tag{2.7}$$

Kısıt fonksiyonları,

$$g_1(x) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \le 0 \tag{2.8}$$

$$g_2(x) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \le 0 \tag{2.9}$$

$$g_3(x) = 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \le 0 \tag{2.10}$$

$$g_4(x) = -8x_1 + x_{10} \le 0 \tag{2.11}$$

$$g_5(x) = -8x_2 + x_{11} \le 0 \tag{2.12}$$

$$g_6(x) = -8x_3 + x_{12} \le 0 \tag{2.13}$$

$$g_7(x) = -2x_4 - x_5 + x_{10} \le 0 \tag{2.14}$$

$$g_8(x) = -2x_6 - x_7 + x_{11} \le 0 \tag{2.15}$$

$$g_9(x) = -2x_8 - x_9 + x_{12} \le 0 \tag{2.16}$$

Burada sınırlar $0 \le x_i \le 1 (i = 1,...,9),\ 0 \le x_i \le 100$

$(i = 10,11,12)\ ve\ 0 \le x_{13} \le 1$

Global minimum $x^* = (1,1,1,1,1,1,1,1,1,3,3,3,1)$

6 sınırlayıcı $(g_1, g_2, g_3, g_7, g_8, g_9)$

$f(x^*) = -15$

Çizelge 2.1 – Örnek 1 için optimum çözümler

Tasarım değişkenleri	Yarasa Algoritması	Parçacık Sürü	Gugukkuşu Arama	Ateş Böceği	Av Arama
x_1	0.99266	0,268957	0,99412	0,999749	0,997097
x_2	0.965705	1	0,99898	0,995027	0,9999
x_3	0,998388	0,998273	0,99651	0,996723	0,99993
x_4	0,979785	1	0,99797	0,998828	0,999496
x_5	0,973136	1	0,99576	0,985386	0,999201
x_6	0,976451	0,996442	0,99822	0,995111	0,998902
x_7	0,94594	0,999964	0,99705	0,995311	0,99965
x_8	0,988192	0,996083	0,98819	0,982028	0,999425
x_9	0,935385	1	0,97699	0,99156	0,995757
x_{10}	1749514	2,15165	2,87887	2,969814	2998194
x_{11}	2225278	2,992848	2,79799	2,98175	2997458
x_{12}	2110634	2,992167	2,72006	2,954808	299460
x_{13}	0,992474	0,999999	1	0,995215	0,997209
g_1	-2,108479	-2,317583	-0,3369247	-0,0588777	-0,010346
g_2	-2,302329	-2,321718	-0,4197872	-0,0824335	-0,013139
g_3	-1,880475	-0,0184393	-0,4909381	-0,0799367	-0,008262
g_4	-6,191768	9,54E-07	-5,0740920	-5,02818	-4,978586
g_5	-5,50036	-5,007153	-5,1938610	-4,978459	-5,001773
g_6	-5,298181	-4,994016	-5,2520600	-5,018977	-5,00485
g_7	-1,183193	-0,8483445	-0,1128445	-0,0132269	0
g_8	-0,6702185	-2,38E-07	-0,1955154	-0,0037769	0
g_9	-0,8011355	0	-0,2333125	-0,0008085	0
f(x)	**-11,58456**	**-13,13744**	**-14,28388**	**-14,80281**	**-14,9627**

Bu problem; yarasa algoritması, parçacık sürü, guguk kuşu arama, ateş böceği algoritması ve av arama yöntemleriyle ayrı ayrı çözülmüş ve tekniklerin performans karşılaştırılması yapılmıştır. Bu karşılaştırma sonunda en iyi sonucu -14.9627 ile av arama verirken, bunu

sırasıyla -14.80281 ile ateş böceği, -14.28388 ile gugukkuşu arama, -13.13744 ile parçacık sürü ve son olarak -11.58456 ile yarasa algoritması yöntemleri takip etmektedir.

2.3.2 Örnek 2

Amaç fonksiyonu,

$$f(x) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141 \quad (2.17)$$

Kısıt fonksiyonları,

$$g_1(x) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \le 0 \quad (2.18)$$

$$g_2(x) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \le 0 \quad (2.19)$$

$$g_3(x) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 - 110 \le 0 \quad (2.20)$$

$$g_4(x) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \le 0 \quad (2.21)$$

$$g_5(x) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \le 0 \quad (2.22)$$

$$g_6(x) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \le 0 \quad (2.23)$$

Çizelge 2.2 – Örnek 2 için optimum çözümler

Tasarım değişkenleri	Yarasa Algoritması	Parçacık Sürü	Gugukkuşu Arama	Ateş Böceği	Av Arama
x_1	78,4162	78	78,00524	78,00053	78
x_2	34,1329	33	33,0053	33,00047	33
x_3	30,67748	30,23021	30,00335	29,99722	29,995
x_4	44,8044	44,99995	44,96826	44,99273	44,99998
x_5	35,28625	36,1863	36,7678	36,77384	36,77419
g_1	-91,995700	-91,99998	-91,9984100	-91,99969	-92,0001
g_2	-0,0043039	-1,951E-05	-0,0015970	-0,0003121	9,976E-05
g_3	-9,172581	-8,947772	-8,8428190	-8,840579	-8,840165
g_4	-10,82742	8,09E-07	-11,15718	-11,15942	-11,15983
g_5	-0,0330224	-5,000001	-0,000137	-2,521E-05	9,929E-05
g_6	-4,966978	-4,994016	-4,999863	-4,999975	-5,000099
f(x)	**-30513,07**	**-30628,16**	**-30663,11**	**-30665,01**	**-30665,53**

Bu problemde en iyi sonucu -30665.53 ile av arama verirken, bunu sırasıyla -30665.01 ile ateş böceği, -30663.11 ile gugukkuşu arama, -330628.16 ile parçacık sürü ve son olarak -30513.07 ile yarasa algoritması yöntemleri takip etmektedir.

2.3.3 Örnek 3

Amaç fonksiyonu,

$$f(x) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45 \quad (2.24)$$

Kısıt fonksiyonları,

$$g_1(x) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \le 0 \quad (2.25)$$

$$g_2(x) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \le 0 \quad (2.26)$$

$$g_3(x) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \le 0 \quad (2.27)$$

$$g_4(x) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \le 0 \quad (2.28)$$

$$g_5(x) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \le 0 \quad (2.29)$$

$$g_6(x) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \le 0 \quad (2.30)$$

$$g_7(x) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \le 0 \quad (2.31)$$

$$g_8(x) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \le 0 \quad (2.32)$$

Çizelge 2.3 – Örnek 3 için optimum çözümler

Tasarım değişkenleri	Yarasa Algoritması	Parçacık Sürü	Gugukkuşu Arama	Ateş Böceği	Av Arama
x_1	2,0907	1,915587	2,078724	2,171416	2,05227
x_2	2,8771	3	2,619306	2,363141	2,65083
x_3	8,2555	8,789985	8,735465	8,7633	8,78255
x_4	5	5,022758	5,031492	5,097346	5,00414
x_5	0,9696	0,999882	0,869855	0,986926	0,898279
x_6	1,2443	1,402246	1,11661	1,42075	1,12574
x_7	0,9935	1,044542	1,15185	1,323913	1,09468
x_8	9,4666	9,712804	9,661819	9,827645	9,64676
x_9	8,2826	8,010724	7,813233	8,284069	7,78241

$\mathbf{x_{10}}$	9,2271	8,973871	7,909037	8,388416	7,89777
$\mathbf{g_1}$	-0,030000	-1,19E-07	-0,0054111	-0,021565	9,894E-06
$\mathbf{g_2}$	-0,07	-1,788E-06	-1,3126580	-0,0421929	7,749E-06
$\mathbf{g_3}$	-0,01	4,77E-07	-0,0119624	-0,0015306	9,537E-06
$\mathbf{g_4}$	-18,61	-2,396E-05	-1,20032	-0,343957	-2,66E-01
$\mathbf{g_5}$	-0,04	-3,038869	-0,694864	-0,0728203	7,663E-06
$\mathbf{g_6}$	-0,01	5,45E-07	-0,0641036	-0,011419	2,532E-06
$\mathbf{g_7}$	-8,44	-6,025425	-7,5015210	-6,15719	-7,3E+06
$\mathbf{g_8}$	-52,64	-5,291E+01	-43,3643300	-50,08598	-4,4E+01
f(x)	**27,8985**	**26,531**	**26,34253**	**24,39887**	**25,9168**

Ele alınan bu üçüncü problemde ise en iyi sonucu 24.39887 ile ateş böceği verirken, bunu sırasıyla 25.9168 ile av arama, 26.34253 ile guguk kuşu arama, 26.531 ile parçacık sürü ve son olarak 27.8985 ile yarasa algoritması yöntemleri takip etmektedir.

2.3.4 Örnek 4 – Kaynaklı Kiriş Tasarım Problemi

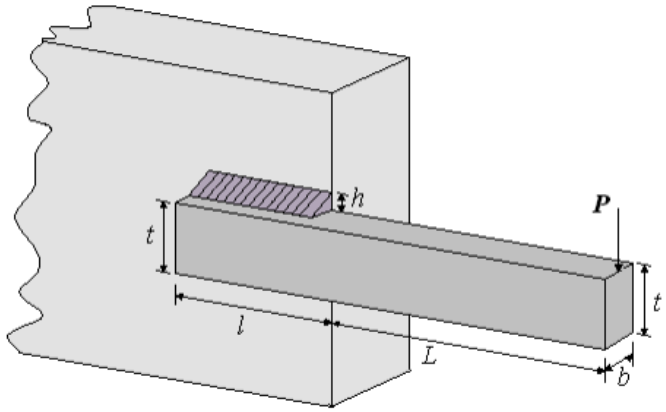
Ateş böceği algoritmasının işleyişini denetlemek için dikdörtgen kesite sahip ankastre bir kirişin tasarım problemi ele alınmıştır. Bu problem belli bir yükü taşıyan bir kirişin minimum maliyete sahip olacak şekilde tasarlanması işlemidir. Bu optimizasyon problemi dört adet tasarım değişkenine sahiptir. Bu değişkenler:

$h = x_1$: kaynak kalınlığı

$l = x_2$: kaynak boyu

$t = x_3$: kiriş derinliği

$b = x_4$: kiriş genişliği



Şekil 2.4 – Kaynaklı kiriş tasarımı

Maliyetin minimum olmasını sağlayacak amaç fonksiyonu;

$$f(x) = 1.10471 x_1^{\,2} x_2 + 0.04811 x_3 x_4 (14.0 + x_2) \tag{2.33}$$

$$g_1(x) = \tau(x) - \tau_{\max} \le 0 \rightarrow \text{ kayma gerilmesi} \tag{2.34}$$

$$g_2(x) = \sigma(x) - \sigma_{\max} \le 0 \quad \rightarrow \quad \text{kiriş eğilme gerilmesi} \tag{2.35}$$

$$g_3(x) = x_1 - x_4 \le 0 \quad \rightarrow \quad \text{geometrik kısıtlama} \tag{2.36}$$

$$g_4(x) = 0.10471 x_1^{\,2} + 0.04811 x_3 x_4 (14.0 + x_2) - 5 \le 0 \quad \rightarrow \text{ geometrik kısıtlama} \tag{2.37}$$

$$g_5(x) = 0.125 - x_1 \le 0 \quad \rightarrow \quad \text{geometrik kısıtlama} \tag{2.38}$$

$$g_6(x) = \delta(x) - \delta_{\max} \le 0 \quad \rightarrow \quad \text{kirişin uç noktasındaki deplasman} \tag{2.39}$$

$$g_7(x) = P - P_c(x) \le 0 \quad \rightarrow \quad \text{kirişin burkulma yükü} \tag{2.40}$$

Burada,

$$\tau(x) = \sqrt{(\tau')^2 + 2\,\tau'\tau''\,\frac{x_2}{2R} + (\tau'')^2} \tag{2.41}$$

$$\tau' = \frac{P}{\sqrt{2}\,x_1 x_2} \tag{2.42}$$

$$\tau'' = \frac{MR}{J} \tag{2.43}$$

$$M = P\left(L + \frac{x_2}{2}\right) \tag{2.44}$$

$$R = \sqrt{\frac{{x_2}^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2} \tag{2.45}$$

$$J = 2\left\{\frac{x_1 x_2}{\sqrt{2}}\left[\frac{{x_2}^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\} \tag{2.46}$$

$$\delta(x) = \frac{4PL^3}{E{x_3}^3 x_4} \tag{2.47}$$

$$\sigma(x) = \frac{6PL}{x_4 {x_3}^2} \tag{2.48}$$

$$P_c(x) = \frac{4.013\sqrt{\frac{(EGx_3^2 x_4^6)}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right) \tag{2.49}$$

$$P = 6000\ lb, \quad L = 14\ in., \quad E = 30\times10^6\ psi, \quad G = 12\times10^6\ psi$$

$$\tau_{\max} = 13{,}600\ psi, \quad \sigma_{\max} = 30{,}000\ psi, \quad \delta_{\max} = 0.25\ in. \tag{2.50}$$

Problemdeki kısıtlamalar;

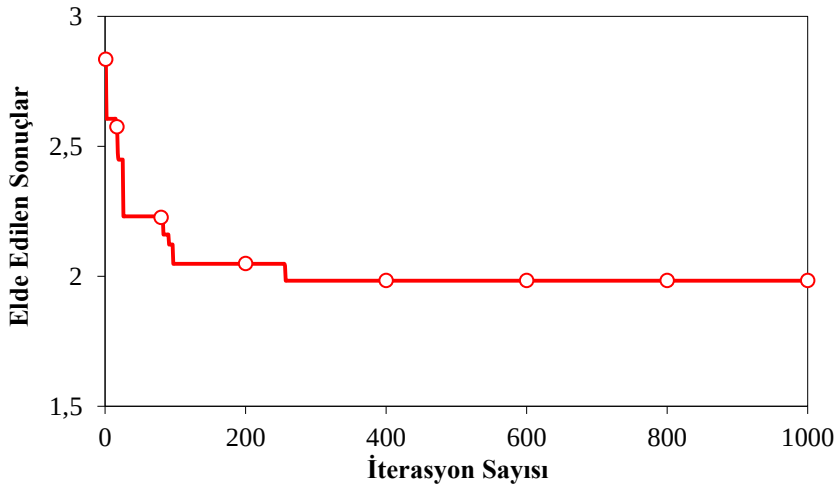
$$\begin{aligned} &0.1 \le x_1 \le 2.0 \\ &0.1 \le x_2 \le 10 \\ &0.1 \le x_3 \le 10 \\ &0.1 \le x_4 \le 2.0 \end{aligned} \tag{2.51}$$

Bu kaynaklı kiriş tasarım problemi daha önce geometrik programlama (GP) (Ragsdell ve Philips, 1976), genetik algoritma (GA) (Deb, 1991), parçacık sürü (PSO) (Doğan, 2010) ve av arama yöntemi (AA) (Şeker, 2013) gibi yöntemlerle çözülmüştür. Bu çalışmada özel olarak, literatürde yer alan ateş böceği algoritması (AAO) kaynaklı kiriş problemi üzerinde uygulanmıştır ve amaç

fonksiyonun minimum değerlerini gösteren sonuçlar, daha önce farklı yöntemlerle bulunmuş olan sonuçlarla birlikte Çizelge 2.4'te gösterilmiştir. Çizelge 2.4'teki sonuçlar irdelendiğinde kaynaklı kiriş optimizasyon problemi için amaç fonksiyonunun en küçük değerinin av arama algoritması ile yapılan çözüm sonucunda ortaya çıktığı görülmektedir.

Çizelge 2.4 – Kaynaklı kiriş problemi için optimum çözümler

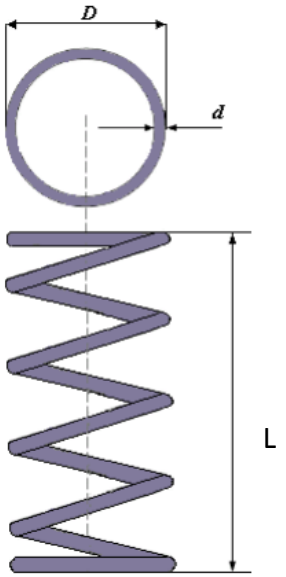
Tasarım değişkenleri	GA (Deb, 1991)	GP (Ragsdell ve Philips, 1976)	AAO	PSO (Doğan, 2010)	AA (Şeker, 2013)
x1	0,2489	0,2455	0,238735	0,2388	0,205731
x2	6,1730	6,1960	2,510584	2,5296	3,47112
x3	8,1789	8,2730	9,201044	9,1796	9,0366
x4	0,2533	0,2455	0,238734	0,2389	0,2057
f(x)	2,4331	2,3859	1,90289	1,90308	1,7249



Şekil 2.5 – Kaynaklı kiriş tasarım problemi için ateş böceği algoritması tasarım geçmişi

2.3.5 Örnek 5 – Yay Tasarım Problemi

Ateş böceği algoritmasının işleyişini denetlemek için ele alınan bir diğer problem yay tasarım problemidir. Bu problem Şekil 2.7'de gösterilmiş olan amaç fonksiyonu yayın gerilme ağırlığının sıkıştırma ağırlığına oranı için en küçük değeri elde etmeyi hedefler. Bu problemde; en az dönme, kayma gerilmesi, dalga frekansı ve dış çap olmak üzere dört adet kısıtlama fonksiyonu bulunmaktadır.



Şekil 2.6 – Yay tasarım problemi

$d = x_1$: yayı oluşturan telin çapı

$D = x_2$: yayın çapı

$N = x_3$: yayda bulunan aktif halka sayısı

Amaç fonksiyonu;

$$f(x) = (x_3 + 2)x_2 x_1^2 \tag{2.52}$$

$$g_1(x) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \leq 0 \tag{2.53}$$

$$g_2(x) = \frac{4x_2^2 - x_1 x_2}{12566(x_2 x_1^3 - x_1^4)} + \frac{1}{5108 x_1^2} - 1 \leq 0 \tag{2.54}$$

$$g_3(x) = 1 - \frac{140.45 x_1}{x_2^2 x_3} \leq 0 \tag{2.55}$$

$$g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \leq 0 \tag{2.56}$$

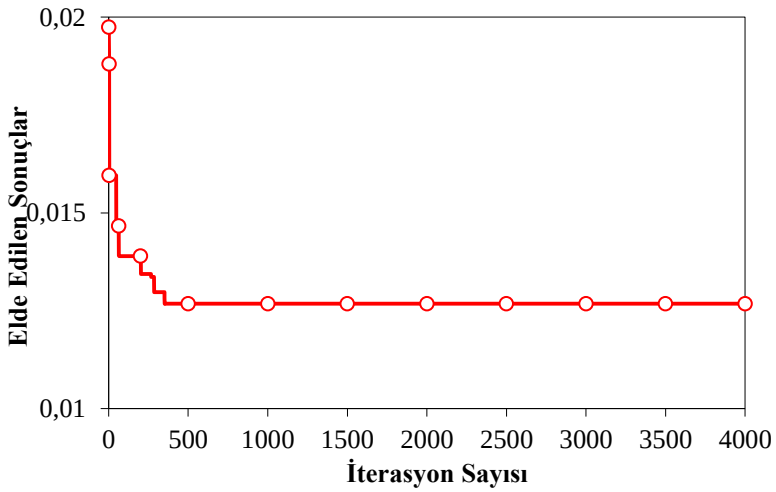
Geometrik kısıtlar,

$$0.05 \le x_1 \le 2$$
$$0.25 \le x_2 \le 1.3 \qquad (2.57)$$
$$2 \le x_3 \le 15$$

Yay tasarım optimizasyon problemi daha önce genetik algoritma (GA) (Carlos ve Coello, 2000), evrimsel algoritma (EA) (Ray ve Liew, 2003), parçacık sürü (PSO) (Doğan, 2010) ve av arama yöntemi (AA) (Şeker, 2013) gibi yöntemlerle çözülmüştür. Bu problemin çeşitli algoritmalar kullanılarak elde edilen optimum çözümleri ile ateş böceği algoritması (AAO) kullanılarak elde edilen optimum çözüm değerleri Çizelge 2.5'te gösterilmiştir. Çizelge 2.5'teki sonuçlar incelendiğinde yay tasarım problemi için amaç fonksiyonunun en küçük değerinin av arama algoritması ile yapılan çözüm sonucunda ortaya çıktığı görülmektedir.

Çizelge 2.5 – Yay tasarım problemi için optimum çözümleri

Tasarım değişkenleri	GA (Carlos ve Coello, 2000)	EA (Ray ve Liew, 2003)	PSO (Doğan, 2010)	AAO	AA (Şeker, 2013)
x1	0,05148	0,05216	0,05194	0,051674	0,051695
x2	0,35166	0,36816	0,36266	0,356352	0,356869
x3	11,6322	10,6484	10,9483	11,31039	11,27982
f(x)	**0,012704**	**0,012669**	**0,012666**	**0,0126652**	**0,012665**



Şekil 2.7 – Yay tasarım problemi için ateş böceği algoritması tasarım geçmişi

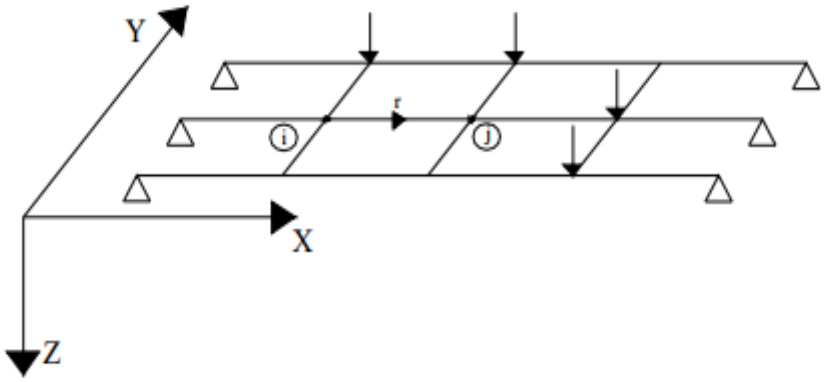
Benchmark problemlerinden elde edilen sonuçlar incelendiğinde ateş böceği algoritmasının genel olarak diğer metodlarla elde edilen çözümlere yakın çözümler bulduğu; 1., 2. ve 3.

örnekte yarasa algoritmasının en iyi çözümü ürettiği, kaynaklı kiriş ve yay probleminde av arama optimizasyonu metoduyla birlikte optimum çözüme ulaştığı görülmüştür. Bu sonuçlar ışığında yöntemin diğer alanlardaki optimizasyon uygulamalarında kullanılabileceği sonucuna ulaşılmıştır.

3. IZGARA SİSTEMLERİN LRFD ŞARTNAMESİNE GÖRE TASARIMI

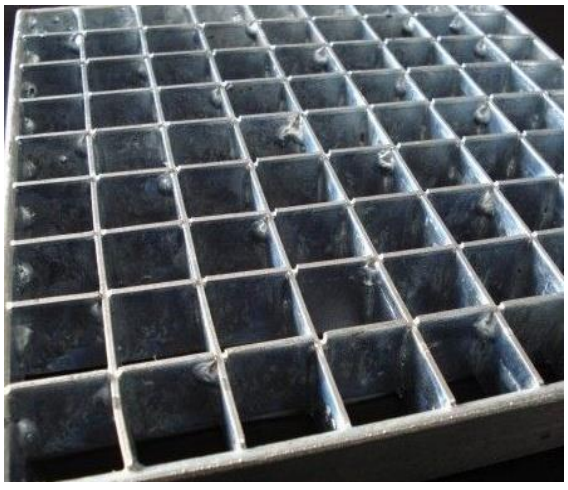
3.1 Izgara Sistemler

Kendi düzlemlerine dik olarak yüklenen rijit bağlantılı çerçeveler ızgara olarak adlandırılır. Izgara sistemler ince cidarlı elemanlar kullanılarak inşa edilirler. Enine veya boyuna olan kirişler mesnetlerle desteklenebilir. Bir kiriş takımı diğer kiriş grubunun üzerine yerleştirilir, kesişim noktalarında da iki grup arasında sadece dikey bir etkileşim vardır.



Şekil 3.1 – Izgara sistem

Yatay düzlemde birbirine rijit bağlı elemanlardan oluşan ızgara sistemler yapılarda köprü tabliyesi, gemi gövdesi, kat döşemesi gibi geniş alanların kaplanması amacıyla sıklıkla kullanılan yapı elemanlarıdır. Bu çalışma ile birlikte farklı tiplerdeki ızgara sistemlerin daha ekonomik bir şekilde boyutlandırılabileceği vurgulanacaktır.



Şekil 3.2 – Izgara sistem

3.2 Matris Deplasman Yöntemi

İnşaat mühendisliğinde, taşıyıcı sistemi çubuk elemanlardan meydana gelen düzlem ya da uzaysal çerçevelerin kesit tesirlerinin ve deplasmanlarının hesaplanmasında kullanılan birçok paket programın temelinde matris deplasman yöntemi bulunmaktadır. Matris deplasman yönteminde, çubuk uç deplasmanlarından, çubuk uç kuvvetlerine geçişi sağlayan rijitlik matrisinin kurulması önemli yer tutmaktadır. Taşıyıcı sistemin tüm çubuklarının düğüm noktalarının konumu, global koordinat takımı ile çubukların bağımsız olarak tek tek konumları, lokal koordinat takımı ile tanımlanabilir.

Rijitlik matrisi yönteminde yapıyı oluşturan elemanların rijitlik değerleri dikkate alınarak, yapıya etkiyen yükler ile yapıda oluşan deplasmanlar arasında oluşturulan denklemler esas alınmıştır. Rijitlik matrisi yardımıyla yapıyı analiz edebilmek için yapının analitik olarak modellenmesi gerekmektedir. Yapıda bulunan her bir düğüm noktası ve elemanlar için bir numara belirlenir.

3.2.1 Global Koordinat Sistemi

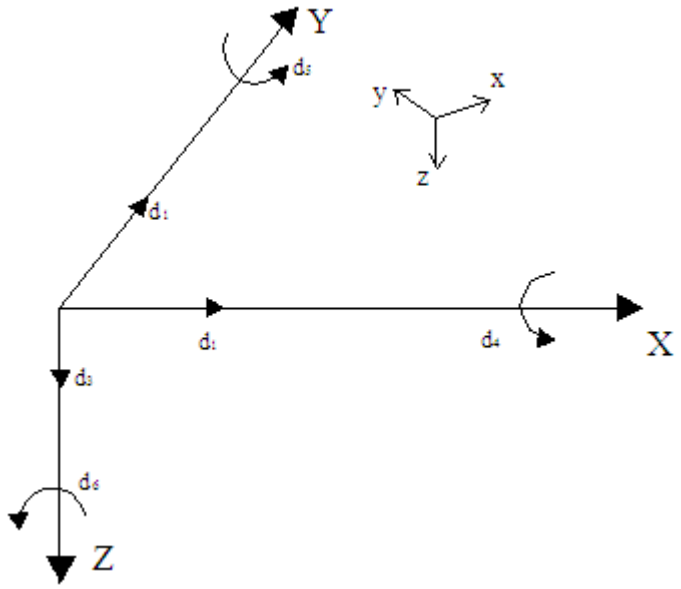
Matris deplasman yönteminde yapının geometrisini ve davranışını tanımlayabilmek üzere kartezyen global koordinat sistemi kullanılır. Şekil 3.3 X, Y ve Z global koordinat sistemini göstermektedir. Bu koordinat sistemi noktaların yerini ve yükleme yönlerini tanımlamak için kullanılabilir. Düğüm noktalarında oluşan yer değiştirmeler d_1, d_2, d_3 , dönmeler ise d_4, d_5, d_6 olarak tanımlanmıştır.

Global koordinatlardaki deplasman vektörü;

$$\{D\}_i = \{\theta_{xi} \quad \theta_{yi} \quad \delta_{zi}\}$$

Global koordinatlardaki dış kuvvet vektörü;

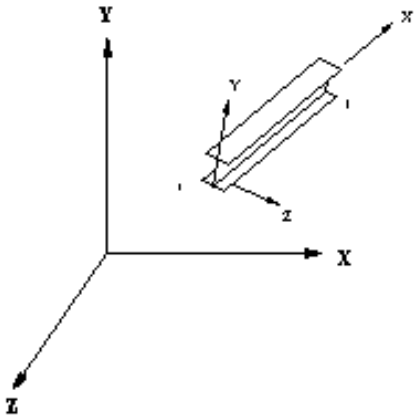
$$\{P\}_i = \{P_{xi} \quad P_{yi} \quad P_{zi}\}$$



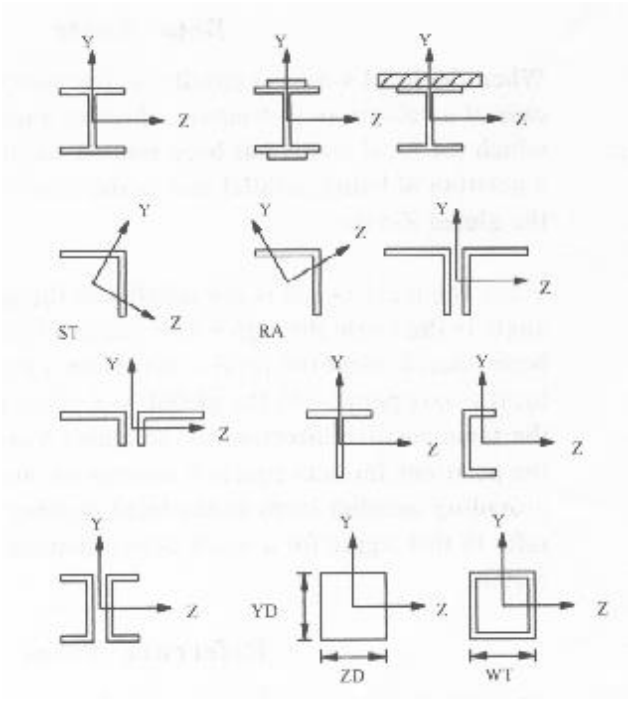
Şekil 3.3 – Kartezyen koordinat sistemi

3.2.2 Lokal Koordinat Sistemi

Lokal koordinat sistemi her bir eleman ile ilgilidir. Lokal koordinat sisteminin her ekseni sağ el kuralına dayanmaktadır. Şekil 3.4' te "i" kiriş elemanın başlangıç noktasını ve "j" ise bitiş noktasını göstermektedir. Şekil 3.4' te görüldüğü üzere, elemanlar için lokal koordinat x ekseninin direkt olarak eleman doğrultusundaki eksen şeklinde belirlenmesi ve y ekseninin de x eksenine göre sağ el kuralıyla tespit edilmesi şeklinde ifade edilir.



Şekil 3.4 – Kiriş eleman



Şekil 3.5 – Farklı kesitler için lokal eksenler

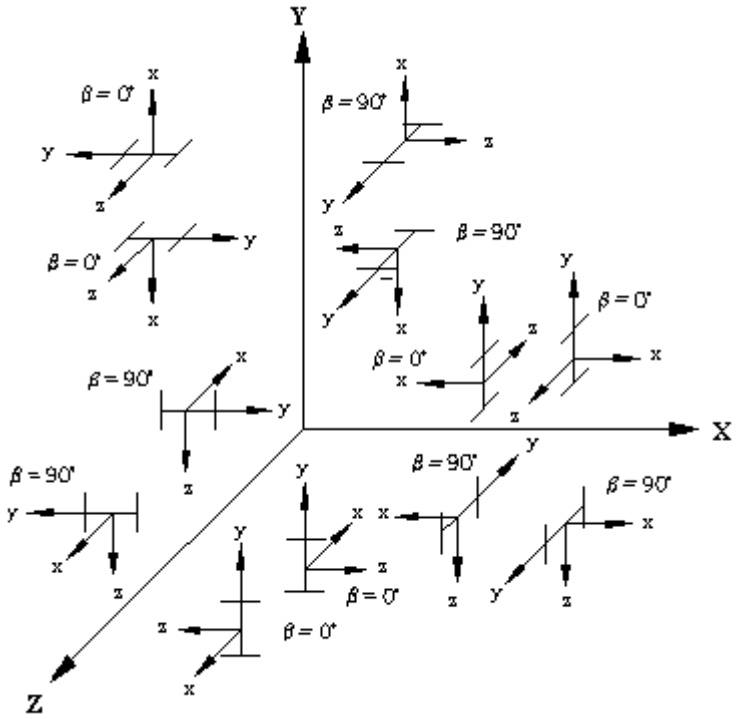
3.2.3 Global ve Lokal Koordinatlar Arasındaki İlişki

Eleman yüklerini temsil eden girdi matrisi lokal ve global koordinat sistemlerinde elde edilebileceğinden, çıktı olarak ortaya çıkan eleman kuvvetleri lokal koordinat düzleminde yazılabilir.

Lokal ve global koordinat sistemleri arasındaki ilişki aşağıda belirtilen şekilde ölçülen beta (β) açısı ile ifade edilir.

Lokal x-ekseni global Y-eksenine paralel olduğunda, beta açısı lokal z-ekseninin global Z ekseninin pozitif yönünde ve ona paralel doğrultudaki bir noktanın lokal x ekseni etrafında döndürülmesiyle elde edilir.

Lokal x-ekseninin global Y-eksenine paralel olmadığı durumlarda ise, beta açısı lokal koordinat sisteminin, global X-Z düzlemine paralel durumdaki lokal z ekseniyle belli bir mesafe yapan bir noktada lokal x ekseni etrafında ve ayrıca global Y ekseniyle aynı mesafedeki bir noktada lokal y ekseni etrafında döndürülmesiyle elde edilir. Şekil 3.6, beta açısının 0 derece ve 90 dereceye eşit olduğu durumları göstermektedir. Lokal eleman eksenlerindeki eleman yüklerinin tespiti aşamasında, lokal eksen sisteminin hızlı tayini için bu şekilden faydalanılabilir.



Şekil 3.6 – Global ve lokal koordinatlar arasındaki ilişki

Düzlemsel taşıyıcı sistemlerde; çubuk elemanların her biri bağlandığı düğüm noktasına bir rijitlik katmaktadır. Yani taşıyıcı sistemin rijitliği, sistemi oluşturan her bir çubuğun rijitliğine de bağlıdır. Çubukların taşıyabileceği kuvvetler ise çubuk elemanların rijitlikleri ile ilgilidir. Bu kuvvetler çubuk eksenine paralel olabileceği gibi çubuk eksenine dik şekilde de olabilir.

Bu özellikleri dikkate alarak, taşıyıcı sistemi oluşturan tüm çubuk elemanların rijitlik matrislerinin birleştirilmesi suretiyle, sistemin rijitlik matrisi elde edilebilir. Elde edilen sistem rijitlik matrisi ile düğüm noktalarındaki deplasmanların ve dış kuvvetlerin ilişkilendirilmesi ve matris şeklindeki denklem takımının çözülmesi suretiyle, bilinmeyen düğüm noktası deplasmanları ve buradan da çubuk uç kuvvetlerini hesaplamak mümkün olabilmektedir (Çağdaş S, 2006).

Yapı elemanlarının düğüm noktası deplasmanlarını u ve düğüm noktası kuvvetlerini F olarak tanımlarsak;

$$F = k.u \tag{3.1}$$

bağıntısı yazılabilir. Burada, k birim deplasman yani rijitlik matrisini ifade eder.

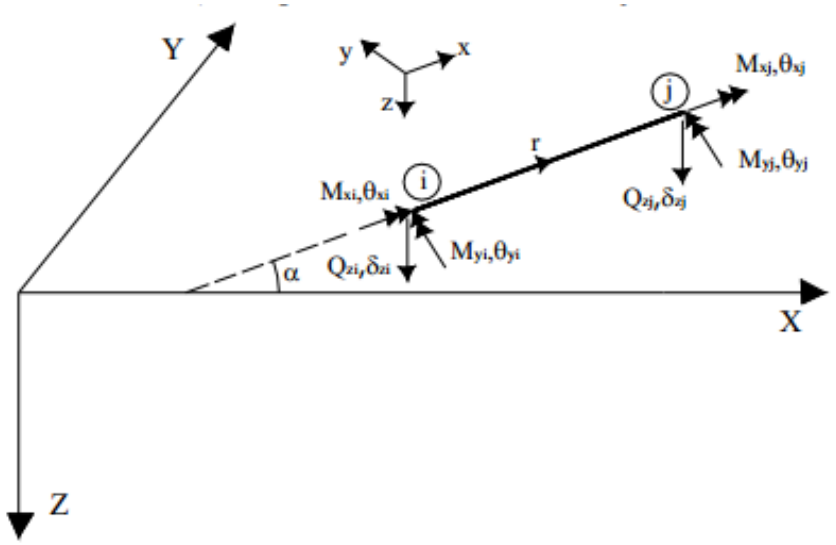
Taşıyıcı sisteme ait denklem takımları; matris formunda genel olarak aşağıdaki gibi ifade edilebilir.

$$\{F\} = [k]\{U\} \tag{3.2}$$

Bu denklemde, elemanlara birim deplasman yaptırmak için gerekli olan kuvvetin bulunması hedeflenmiştir.

3.2.4 Çubuk Elemanların Uç Kuvvetleri ile Uç Deplasmanlar Arasındaki İlişki

Bir çubuk elemanın uç deplasmanları ve uç kuvvetleri şu şekilde gösterilebilir.



Şekil 3.7 – Izgara sistem elemanına etkiyen uç kuvvetler ve momentler

i noktasındaki deplasmanlar,

$$\{D_i\} = \begin{Bmatrix} \theta_{xi} \\ \theta_{yi} \\ \delta_{zi} \end{Bmatrix} \tag{3.3}$$

i noktasındaki kuvvetler,

$$\{P_i\} = \begin{Bmatrix} M_{xi} \\ M_{yi} \\ Q_{zi} \end{Bmatrix} \tag{3.4}$$

Çubuk elemanın uç kuvvetleri ile uç deplasmanları arasında aşağıda belirtildiği gibi bir ilişki söz konusudur.

$$
\begin{aligned}
M_{xi} &= k_{11}u_{xi} + k_{12}u_{yi} + k_{13}u_{zi} + k_{14}u_{xj} + k_{15}u_{yj} + k_{16}u_{zj} \\
M_{yi} &= k_{21}u_{xi} + k_{22}u_{yi} + k_{23}u_{zi} + k_{24}u_{xj} + k_{25}u_{yj} + k_{26}u_{zj} \\
Q_{zi} &= k_{31}u_{xi} + k_{32}u_{yi} + k_{33}u_{zi} + k_{34}u_{xj} + k_{35}u_{yj} + k_{36}u_{zj} \\
M_{xj} &= k_{41}u_{xi} + k_{42}u_{yi} + k_{43}u_{zi} + k_{44}u_{xj} + k_{45}u_{yj} + k_{46}u_{zj} \\
M_{yj} &= k_{51}u_{xi} + k_{52}u_{yi} + k_{53}u_{zi} + k_{54}u_{xj} + k_{55}u_{yj} + k_{56}u_{zj} \\
Q_{zj} &= k_{61}u_{xi} + k_{62}u_{yi} + k_{63}u_{zi} + k_{64}u_{xj} + k_{65}u_{yj} + k_{66}u_{zj}
\end{aligned}
\tag{3.5}
$$

Uç kuvvetler ile uç deplasmanlar arasındaki ilişki matris halinde yazılırsa,

$$
\begin{Bmatrix} M_{xi} \\ M_{yi} \\ Q_{zi} \\ M_{xj} \\ M_{yj} \\ Q_{zj} \end{Bmatrix} =
\begin{bmatrix}
k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\
k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} \\
k_{31} & k_{32} & k_{33} & k_{34} & k_{35} & k_{36} \\
k_{41} & k_{42} & k_{43} & k_{44} & k_{45} & k_{46} \\
k_{51} & k_{52} & k_{53} & k_{54} & k_{55} & k_{56} \\
k_{61} & k_{62} & k_{63} & k_{64} & k_{65} & k_{66}
\end{bmatrix}
\begin{Bmatrix} u_{xi} \\ u_{yi} \\ u_{zi} \\ u_{xj} \\ u_{yj} \\ u_{zj} \end{Bmatrix}
\quad ya\,da \quad \{F\}_r = [k]\{u\}_r
\tag{3.6}
$$

$\{F\}_r$: uç kuvvet vektörü

$\{u\}_r$: uç deplasman vektörü

$[k]$: lokal koordinat düzlemindeki rijitlik matrisini ifade etmektedir.

Çubuğun lokal koordinat düzlemindeki kuvvet vektörü,

$$
\{F\}_r = \{M_{xi} \;\; M_{yi} \;\; Q_{zi} \;\; M_{xj} \;\; M_{yj} \;\; Q_{zj}\}^T
\tag{3.7}
$$

Çubuğun lokal koordinat düzlemindeki deplasman vektörü,

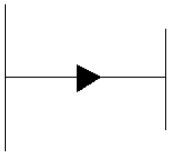
$$
\{u\} = \{u_{xi} \;\; u_{yi} \;\; u_{zi} \;\; u_{xj} \;\; u_{yj} \;\; u_{zj}\}^T
\tag{3.8}
$$

Çubuğun global koordinat düzlemindeki deplasman vektörü,

$$
\{D\}_r = \{\theta_{xi} \;\; \theta_{yi} \;\; \delta_{zi} \;\; \theta_{xj} \;\; \theta_{yj} \;\; \delta_{zj}\}^T
\tag{3.9}
$$

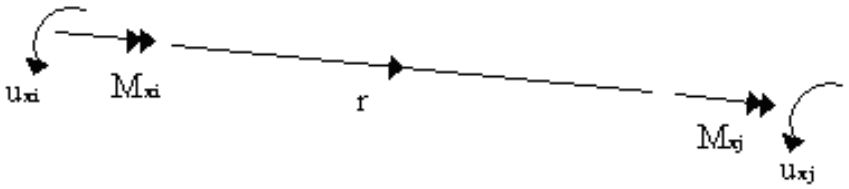
ile gösterilmektedir.

3.2.5 Her İki Ucu Rijit Bir Şekilde Bağlanmış Izgara Elemanın Rijitlik Matrisi



Şekil 3.8 – İki ucu rijit bağlı eleman

Çubuk elemanın uç noktalarına etkiyen burulma momentleri aşağıdaki gibi gösterilmektedir.



Şekil 3.9 – Çubuk elemana etkiyen burulma momentleri

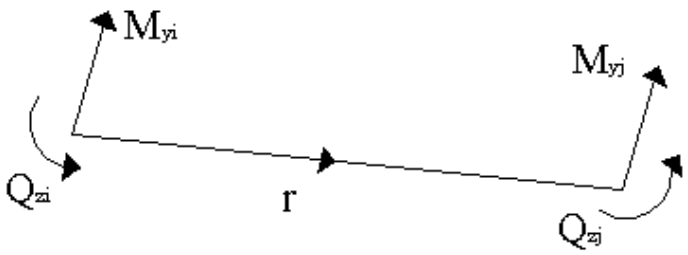
M_{xi} ve M_{xj} çubuğun uç noktalarındaki burulma momentleridir.

Elemanın burulma momenti,

$$M_{xi} = \frac{G\,J}{L}\left(u_{xi} - u_{xj}\right) \tag{3.10}$$

$$M_{xj} = -\frac{G\,J}{L}\left(u_{xi} - u_{xj}\right) \tag{3.11}$$

burada G kayma modülü, J burulma sabitidir.



Şekil 3.10 – Kesme kuvveti ve eğilme momenti

Deplasman dikkate alınarak elde edilen momentler ve kesme kuvvetleri,

$$M_{yi} = \frac{4EI}{L}u_{yi} + \frac{6EI}{L^2}u_{zi} + \frac{2EI}{L}u_{yj} - \frac{6EI}{L^2}u_{zj} \tag{3.12}$$

$$M_{yj} = \frac{2EI}{L}u_{yi} + \frac{6EI}{L^2}u_{zi} + \frac{4EI}{L}u_{yj} - \frac{6EI}{L^2}u_{zj} \tag{3.13}$$

$$Q_{zi} = \frac{6EI}{L^2}u_{yi} + \frac{12EI}{L^3}u_{zi} + \frac{6EI}{L^2}u_{yj} - \frac{12EI}{L^3}u_{zj} \tag{3.14}$$

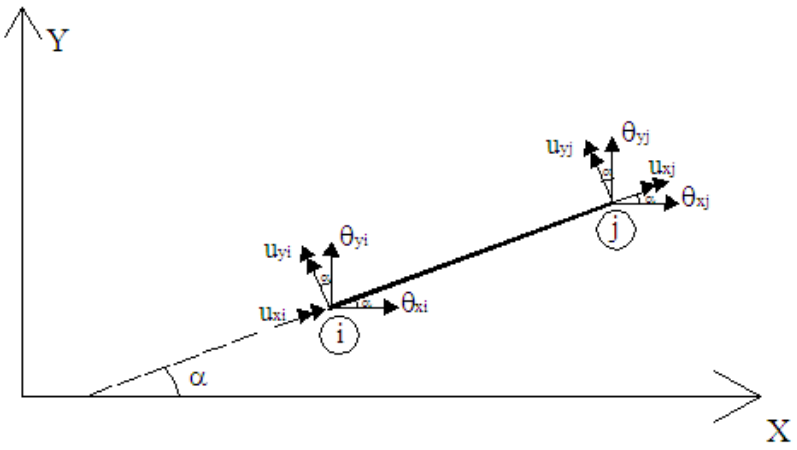
$$Q_{zj} = -\frac{6EI}{L^2}u_{yi} - \frac{12EI}{L^3}u_{zi} - \frac{6EI}{L^2}u_{yj} + \frac{12EI}{L^3}u_{zj} \tag{3.15}$$

Denklem takımı matris şeklinde yazılırsa;

$$[k]_r = \begin{bmatrix} \frac{GJ}{L} & 0 & 0 & -\frac{GJ}{L} & 0 & 0 \\ 0 & \frac{4EI}{L} & \frac{6EI}{L^2} & 0 & \frac{2EI}{L} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{12EI}{L^3} & 0 & \frac{6EI}{L^2} & -\frac{12EI}{L^3} \\ -\frac{GJ}{L} & 0 & 0 & \frac{GJ}{L} & 0 & 0 \\ 0 & \frac{2EI}{L} & \frac{6EI}{L^2} & 0 & \frac{4EI}{L} & -\frac{6EI}{L^2} \\ 0 & -\frac{6EI}{L^2} & -\frac{12EI}{L^3} & 0 & -\frac{6EI}{L^2} & \frac{12EI}{L^3} \end{bmatrix} \tag{3.16}$$

3.2.6 Çubuk Elemanlarda Global Koordinatlarda Uç Kuvvetleri ve Deplasmanları

Bir elemanın global uç kuvvetleri ve uç deplasmanları şekilde görüldüğü gibi global koordinatlar doğrultusunda eleman uçlarında oluşmaktadır.



Şekil 3.11 – Izgara elemanın deplasmanları

Elemanın "i" ucundaki deplasmanlar,

$$\begin{aligned} u_{xi} &= \theta_{xi}\cos\alpha + \theta_{yi}\sin\alpha \\ u_{yi} &= -\theta_{xi}\cos\alpha + \theta_{yi}\sin\alpha \\ u_{zi} &= \delta_{zi} \end{aligned} \tag{3.17}$$

Elemanın "j" ucundaki deplasmanlar,

$$\begin{aligned} u_{xj} &= \theta_{xj}\cos\alpha + \theta_{yj}\sin\alpha \\ u_{yj} &= -\theta_{xj}\sin\alpha + \theta_{yj}\cos\alpha \\ u_{zj} &= \delta_{zj} \end{aligned} \tag{3.18}$$

Denklem takımı matris formunda yazılırsa;

$$\begin{bmatrix} u_{xi} \\ u_{yi} \\ u_{zi} \\ u_{xj} \\ u_{yj} \\ u_{zj} \end{bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & 0 & 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_{xi} \\ \theta_{yi} \\ \delta_{zi} \\ \theta_{xj} \\ \theta_{yj} \\ \delta_{zj} \end{bmatrix} \tag{3.19}$$

$$\{u\}_r = [B]_r \{D\}_r \tag{3.20}$$

3.2.7 Dış Yükler ve Eleman Kuvvetleri Arasındaki İlişki

Enerjinin korunumu ilkesine göre;

$$\frac{1}{2}\{P\}_r^{\ T}\{D\}_r = \frac{1}{2}\{F\}_r^{\ T}\{u\}_r \tag{3.21}$$

$\{F\}_r$: uç kuvvet vektörü

$\{u\}_r$: uç deplasman vektörü

$\{P\}_r$: dış yük vektörü

$\{D\}_r$: yapıdaki deplasman vektörü

$$\frac{1}{2}\{P\}^T\{D\}_r = \frac{1}{2}\{F\}^T[B]_r\{D\}_r \tag{3.22}$$

$$\{P\}^T = \{F\}^T[B]_r \tag{3.23}$$

Her iki tarafın transpozesi alınırsa,

$$\{P\} = [B]_r^T\{F\} \tag{3.24}$$

Denklem (3.6), (3.20) ve (3.24) biraraya getirilirse rijitlik matrisi elde edilir.

$$\{F\}_r = [k]\{u\}_r \tag{3.6}$$

$$\{u\}_r = [B]_r\{D\}_r \tag{3.20}$$

$$\{P\} = [B]_r^T\{F\} \tag{3.24}$$

Denklem (3.20), denklem (3.6)da yerine yazılırsa,

$$\{F\}_r = [k][B]_r\{D\}_r \tag{3.25}$$

Denklem (3.25), denklem (3.24)te yerine yazılırsa,

$$\{P\} = \underbrace{[B]_r^T[k][B]_r}_{[K]}\{D\}_r \tag{3.26}$$

burada $[K]$ global rijitlik matrisini ifade etmektedir.

Global rijitlik matrisi aşağıdaki gibi yazılmaktadır.

Birinci uç İkinci uç

$$[K]=\begin{bmatrix} a & b & c & . & d & e & -c \\ b & f & g & . & e & h & -g \\ c & g & p & . & c & g & -p \\ . & . & . & . & . & . & . \\ d & e & c & . & a & b & -c \\ e & h & g & . & b & f & -g \\ -c & -g & -p & . & -c & -g & p \end{bmatrix} \quad (3.27)$$

$$a=\frac{GJ}{L}\cos^2\alpha+\frac{4EI}{L}\sin^2\alpha \qquad b=\left(\frac{GJ}{L}-\frac{4EI}{L}\right)\cos\alpha\sin\alpha$$

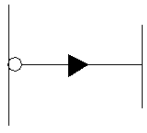
$$c=-\frac{6EI}{L^2}\sin\alpha \qquad d=-\frac{GJ}{L}\cos^2\alpha+\frac{2EI}{L}\sin^2\alpha$$

$$e=\left(-\frac{GJ}{L}-\frac{2EI}{L}\right)\cos\alpha\sin\alpha \qquad f=\frac{GJ}{L}\sin^2\alpha+\frac{4EI}{L}\cos^2\alpha \quad (3.28)$$

$$g=\frac{6EI}{L^2}\cos\alpha \qquad h=-\frac{GJ}{L}\sin^2\alpha+\frac{2EI}{L}\cos^2\alpha$$

$$p=\frac{12EI}{L^3}$$

3.2.8 İlk Ucu Mafsallı Izgara Elemanın Rijitlik Matrisi



Şekil 3.12 – İlk ucu mafsallı eleman

$$M_{xi}=\frac{G\,J}{L}\left(u_{xi}-u_{xj}\right) \quad (3.29)$$

$$M_{xj} = -\frac{G\,J}{L}\left(u_{xi} - u_{xj}\right) \tag{3.30}$$

Mafsal moment taşımayan bir bağ türü olduğundan mafsala göre sisteme etki eden kuvvetlerin momentleri toplamı sıfırdır. $M_{yi} = 0$

$$M_{yi} = 0 \quad \rightarrow \quad u_{yi} = -\frac{3}{2L}u_{zi} - \frac{1}{2}u_{yj} + \frac{3}{2L}u_{zj} \tag{3.31}$$

$$M_{yj} = \frac{3EI}{L^3}u_{zi} + \frac{3EI}{L}u_{yj} - \frac{3EI}{L^2}u_{zj} \tag{3.32}$$

$$Q_{zi} = \frac{3EI}{L^3}u_{zi} + \frac{3EI}{L^2}u_{yj} - \frac{3EI}{L^3}u_{zj} \tag{3.33}$$

$$Q_{zj} = \frac{3EI}{L^2}u_{zi} + \frac{3EI}{L}u_{yj} - \frac{3EI}{L^2}u_{zj} \tag{3.34}$$

Denklem takımı matris formunda yazılırsa;

$$\left[k\right]_r = \begin{bmatrix} \frac{GJ}{L} & 0 & -\frac{GJ}{L} & 0 & 0 \\ 0 & \frac{3EI}{L^3} & 0 & \frac{3EI}{L^2} & -\frac{3EI}{L^3} \\ -\frac{GJ}{L} & 0 & \frac{GJ}{L} & 0 & 0 \\ 0 & \frac{3EI}{L^2} & 0 & \frac{3EI}{L} & -\frac{3EI}{L^2} \\ 0 & -\frac{3EI}{L^3} & 0 & -\frac{3EI}{L^2} & \frac{3EI}{L^3} \end{bmatrix} \tag{3.35}$$

Uygunluk matrisi,

$$\begin{bmatrix} u_{xi} \\ u_{zi} \\ u_{xj} \\ u_{yj} \\ u_{zj} \end{bmatrix} = \begin{bmatrix} \cos\alpha & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_{xi} \\ \delta_{zi} \\ \theta_{xj} \\ \theta_{yj} \\ \delta_{zj} \end{bmatrix} \tag{3.36}$$

$$\left\{u\right\}_r = \left[B\right]_r \left\{D\right\}_r \tag{3.20}$$

Global rijitlik matrisi aşağıdaki gibi yazılmaktadır.

$$[K]=\begin{bmatrix} a & 0 & -a & -b & 0 \\ 0 & c & -d & e & -c \\ -a & -d & f & g & d \\ -b & e & g & h & -e \\ 0 & -c & d & -e & c \end{bmatrix} \tag{3.37}$$

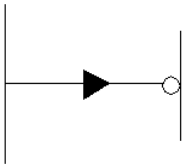
$$a=\frac{GJ}{L}\cos^2\alpha \qquad b=\frac{GJ}{L}\cos\alpha\sin\alpha$$

$$c=\frac{3EI}{L^3} \qquad d=\frac{3EI}{L^2}\sin\alpha \tag{3.38}$$

$$e=\frac{3EI}{L^2}\cos\alpha \qquad f=\frac{GJ}{L}\cos^2\alpha+\frac{3EI}{L}\sin^2\alpha$$

$$g=\left(\frac{GJ}{L}-\frac{3EI}{L}\right)\cos\alpha\sin\alpha \qquad h=\frac{GJ}{L}\sin^2\alpha+\frac{3EI}{L}\cos^2\alpha$$

3.2.9 İkinci Ucu Mafsallı Izgara Elemanın Rijitlik Matrisi



Şekil 3.13 – İkinci ucu mafsallı eleman

$$M_{xi}=\frac{GJ}{L}\left(u_{xi}-u_{xj}\right) \tag{3.39}$$

$$M_{xj}=-\frac{GJ}{L}\left(u_{xi}-u_{xj}\right) \tag{3.40}$$

Bu durumda $M_{yi}=0$ olacaktır. Mafsal moment taşımayan bir bağ türü olduğundan mafsala göre sisteme etki eden kuvvetlerin momentleri toplamı sıfırdır.

$$M_{yj}=0 \quad \rightarrow \quad u_{yj}=-\frac{1}{2}u_{yi}-\frac{3}{2L}u_{zi}+\frac{3}{2L}u_{zj} \tag{3.41}$$

$$M_{yi} = \frac{3EI}{L}u_{yi} + \frac{3EI}{L^2}u_{zi} - \frac{3EI}{L^2}u_{zj} \tag{3.42}$$

$$Q_{zi} = \frac{3EI}{L^2}u_{yi} + \frac{3EI}{L^3}u_{zi} - \frac{3EI}{L^3}u_{zj} \tag{3.43}$$

$$Q_{zj} = -\frac{3EI}{L^2}u_{yi} - \frac{3EI}{L^3}u_{zi} + \frac{3EI}{L^3}u_{zj} \tag{3.44}$$

Denklem takımı matris formunda yazılırsa;

$$[k]_r = \begin{bmatrix} \frac{GJ}{L} & 0 & 0 & -\frac{GJ}{L} & 0 \\ 0 & \frac{3EI}{L} & \frac{3EI}{L^2} & 0 & -\frac{3EI}{L^2} \\ 0 & \frac{3EI}{L^2} & \frac{3EI}{L^3} & 0 & -\frac{3EI}{L^3} \\ -\frac{GJ}{L} & 0 & 0 & \frac{GJ}{L} & 0 \\ 0 & -\frac{3EI}{L^2} & -\frac{3EI}{L^3} & 0 & \frac{3EI}{L^3} \end{bmatrix} \tag{3.45}$$

Uygunluk matrisi,

$$\begin{bmatrix} u_{xi} \\ u_{yi} \\ u_{zi} \\ u_{xj} \\ u_{zj} \end{bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_{xi} \\ \delta_{yi} \\ \theta_{zi} \\ \theta_{xj} \\ \delta_{zj} \end{bmatrix} \tag{3.46}$$

Global rijitlik matrisi aşağıdaki gibi yazılmaktadır.

$$[K] = \begin{bmatrix} a & 0 & -a & -b & 0 \\ 0 & c & -d & e & -c \\ -a & -d & f & g & d \\ -b & e & g & h & -e \\ 0 & -c & d & -e & c \end{bmatrix} \tag{3.47}$$

$$a = \frac{GJ}{L}\cos^2\alpha \qquad b = \frac{GJ}{L}\cos\alpha\sin\alpha$$

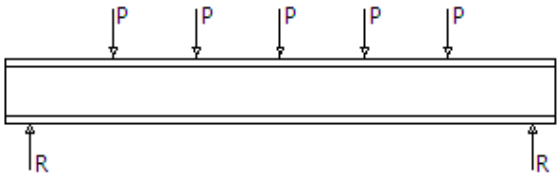
$$c = \frac{3EI}{L^3} \qquad d = \frac{3EI}{L^2}\sin\alpha \qquad (3.48)$$

$$e = \frac{3EI}{L^2}\cos\alpha \qquad f = \frac{GJ}{L}\cos^2\alpha + \frac{3EI}{L}\sin^2\alpha$$

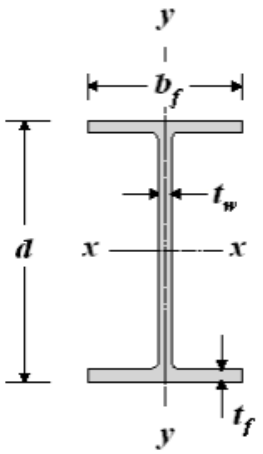
$$g = \left(\frac{GJ}{L} - \frac{3EI}{L}\right)\cos\alpha\sin\alpha \qquad h = \frac{GJ}{L}\sin^2\alpha + \frac{3EI}{L}\cos^2\alpha$$

3.3 LRFD Şartnamesine Göre Çelik Yapı Elemanların Tasarımı

Kirişler genellikle enine düşey yükler etkisi altındadır. Üzerine etkiyen yüklerden birincil derecede eğilme momenti ve kesme kuvveti meydana gelen ve genel olarak bu kesit tesirlerine göre boyutlandırılan taşıyıcı sistemlerdir. Kirişler eğilme momenti etkisinde olan yapı elemanlarıdır. Fakat bazı durumlarda kirişlerde de eksenel yüklerin etkili olduğu durumlarda mevcuttur. Hem eğilme momenti hem de eksenel yükler altında davranış gösteren bu tür elemanlar kiriş-kolon elemanlar olarak düşünülebilir. Rijit düğüm noktalarına sahip düzlemsel çerçevelerde çerçeveyi oluşturan elemanlar kiriş-kolon eleman davranışı gösterir.



Şekil 3.14 – Basit kiriş



Şekil 3.15 – I (W) profil kesit

Bu davranışı gösteren yapı elemanlarının boyutlandırılmasında en yaygın kullanılan profil tipi I (W) profildir.

b_f = profilin başlık genişliği

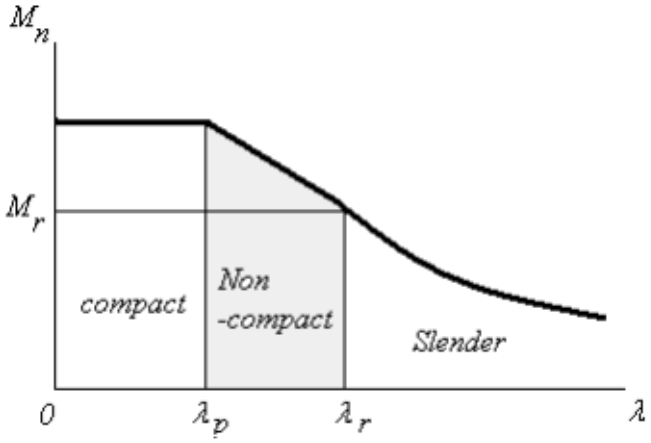
t_f = profilin başlık kalınlığı

t_w = profilin gövde kalınlığı

d = profilin yüksekliği

3.3.1 LRFD Şartnamesine Göre Kirişlerde Eğilme Etkisi

Eksenel basınç ve eğilme için sınır (nominal) dayanımlar, kesitin kompakt, kompakt olmayan, narin veya aşırı narin olarak sınıflandırılmasına bağlıdır.



Şekil 3.16 – Kesitlerin sınıflandırılması

(3.49) numaralı denklem LRFD-F2 şartnamesi uyarınca kirişler için dayanım sınırlayıcısıdır.

$$\phi_b M_n \geq M_u \tag{3.49}$$

Burada; $\phi_b =$ eğilmeye karşı direnç faktörüdür ve değeri 0.9 olarak alınır.

$M_n =$ sınır (nominal) moment dayanımı

$M_u =$ elemana ait servis momenti

Sınır eğilme dayanımı aşağıdaki parametrelere bağlıdır:

Kesitin geometrik şekli, eğilme eksenleri, kesitin kompaktlığı ve yanal-burulmalı burkulma için narinlik parametresidir. Enkesitler ile ilgili bütün profiller için nominal dayanımlar onların eğilme asal eksenleri baz alınarak hesaplanır. Dikdörtgen, I (W), kutu, U, daire, boru, T ve çift köşebentli kesitler için asal eksenler geometrik eksenlerle çakışır. Tek köşebent kesitler için asal eksenler hesaplanır ve bütün hesaplamalar ona göre yapılır.

3.3.1.1 Kompakt (Compact) Kesit

$$\lambda \leq \lambda_p \quad \text{için} \quad M_n \leq M_p \tag{3.50}$$

Burada;

$M_n =$ nominal moment gerilmesi

$M_p =$ plastik moment gerilmesi = Z F_y (3.51)

$Z =$ plastisite modülü

$F_y =$ akma gerilmesi

$\lambda =$ narinlik parametresi

$\lambda_p =$ kompakt elemanlar için sınırlayıcı narinlik parametresi

3.3.1.2 Kompakt Olmayan (Non-compact) Kesit

$$M_n = M_r = S(F_y - F_r) \tag{3.52}$$

$M_r =$ rezidüel moment

$F_r =$ basınç gerilme fazlası

$S =$ elastik kesit modülü

3.3.1.3 Kısmen Kompakt (Compact) Kesit

$$\lambda_p \le \lambda \le \lambda_r \quad \text{için} \quad M_n = M_p - (M_p - M_r)\frac{\lambda - \lambda_r}{\lambda_r - \lambda_p} \tag{3.53}$$

$\lambda_r =$ kompakt olmayan elemanlar için sınırlayıcı narinlik parametresi

3.3.1.4 Narin (Slender) Kesit

$$\lambda \le \lambda_r \quad \text{için} \quad M_n = M_{cr} = S_x F_{cr} \tag{3.54}$$

Burada;

$\lambda =$ narinlik parametresi

$\lambda = b_f / 2t_f$ (başlık için)

$b_f =$ başlık genişliği

$t_f =$ başlık kalınlığı

$\lambda = h/t_w$ (gövde için)

$h = d - 2k$

$d =$ profil yüksekliği

$k =$ başlığın dış yüzünden başlığın gövde ucuna olan mesafe

LRFD-B5.1' e göre λ_p ve λ_r;

$$\left.\begin{aligned} \lambda_p &= 0.38\sqrt{\frac{E}{F_y}} \\ \lambda_r &= 0.83\sqrt{\frac{E}{F_y - F_r}} \end{aligned}\right\} \quad \text{basınç başlığı için} \tag{3.55}$$

$$\left.\begin{aligned}\lambda_p &= 3.76\sqrt{\frac{E}{F_y}}\\ \lambda_r &= 5.70\sqrt{\frac{E}{F_y}}\end{aligned}\right\} \quad \text{gövde için} \qquad (3.56)$$

3.3.2 LRFD Şartnamesine Göre Kirişlerde Kesme Etkisi

Yapı sistemlerinde uzun açıklıklı kirişlerde çökme, orta açıklıklı kirişlerde eğilme etkili olurken kısa açıklıklı kirişlerde ise kesme etkili olmaktadır. Kiriş elemanlar eğilme kapasiteleri temel alınarak seçildikten sonra kesme yönünden de tahkik edilirler.

LRFD-F2'ye göre kesme dayanımı;

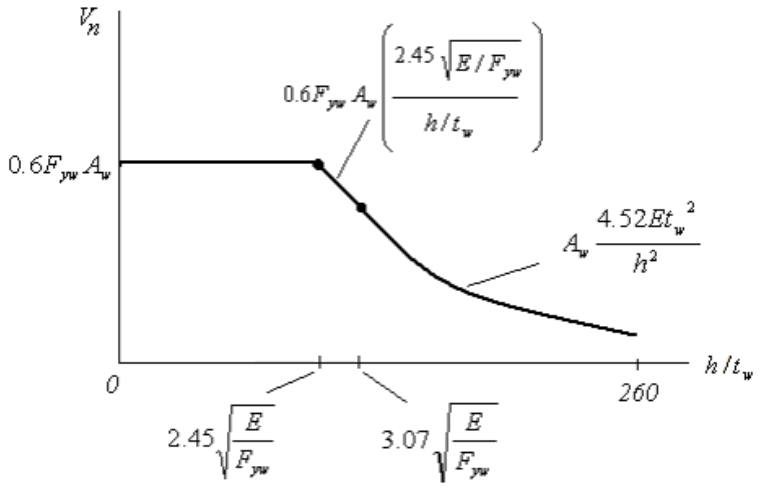
$$\phi_v V_n = V_u \qquad (3.57)$$

Burada;

$\phi_v =$ kesme için direnç faktörü olup 0.9 olarak alınır.

$V_n =$ sınır kesme dayanımı

$V_u =$ gerekli kesme mukavemeti



Şekil 3.17 – I (W) kesitli profillerin sınır kesme gerilmeleri

I (W) kesitli profil kirişlerin LRFD-AISC-F2'ye göre kesme gerilmeleri;

a) $\frac{h}{t_w} \le 2.45\sqrt{\frac{E}{F_{yw}}}$ için $V_n = 0.6F_{yw}A_w$ (3.58)

b) $2.45\sqrt{\frac{E}{F_{yw}}} < \frac{h}{t_w} \le 3.07\sqrt{\frac{E}{F_{yw}}}$ için $V_n = 0.6F_{yw}A_w\left(\frac{2.45\sqrt{\frac{E}{F_{yw}}}}{\frac{h}{t_w}}\right)$ (3.59)

c) $3.07\sqrt{\frac{E}{F_{yw}}} < \frac{h}{t_w} \le 260$ için $V_n = A_w\frac{4.52Et_w^{\,2}}{h^2}$ (3.60)

Burada E elastisite modülünü ve F_{yw} profil gövdesinin akma dayanımını ifade etmektedir. V_n denklem (3.58)-(3.60) bağlı olarak hesaplanır. h/t_w I (W) kesiti dikkate alınarak bulunur.

4. ATEŞ BÖCEĞİ ALGORİTMASI OPTİMİZASYON YÖNTEMİ İLE IZGARA SİSTEMLERİN OPTİMUM TASARIMI

4.1 Izgara Sistemlerin Optimum Tasarımı

Bir yapının tasarımı yapılırken öncelikle yapının taşıyıcı sistemi ve yapıya etkiyen yükler belirlenmelidir. Kritik yükleme durumlarına göre yapı analiz edilerek yapısal elemanlar ve düğüm noktalarında oluşan kuvvetler tespit edilmelidir. Elemanlar ve düğüm noktalarına gelen kuvvetleri karşılayacak şekilde kapasite değerleri dahilinde boyutlandırma yapılmalıdır.

Yapının tasarımında, elemanlarda oluşan gerilmeler ve deplasmanlar kontrol edildiği gibi yapının kullanılabilirliğinin de sağlanması gerekmektedir. Ayrıca yapıların projelendirilmesi aşamasında yapının inşaat maliyetinin de önemi ihmal edilmemelidir. Çelik yapılarda yapı ağırlığının minimum olduğu yapılar maliyetinin de minimum olduğu yapılardır. Bu sebeple çelik yapı tasarımı yapılırken her zaman tüm sınırlayıcıları sağlayacak şekilde en düşük yapı ağırlığı hedeflenerek hazırlanmalıdır. Bu amaçla mevcut çalışmada, geometrisi önceden belirlenmiş ızgara sistem yapıları için tüm sınırlayıcıların sağlandığı ile minimum ağırlıktaki tasarımların üretilmesi hedeflenmiştir.

4.1.1 Izgara Sistemlerin Optimum Boyutlandırılması İçin Matematiksel Model

Izgara sistemlerin boyutlandırılmasındaki ana prensip kirişler için standart çelik kesit tablosundan değerleri W100x19.3 ile W1100x499 arasında değişen 272 adet geniş başlıklı W kesit atamaktır. Bu sistemlerin optimum boyutlandırılmasının gereği olarak; yapı elemanlarının W profiline uygun olması, LRFD-AISC tasarım şartnamelerindeki sınırlayıcıları sağlaması ve sistemin minimum ağırlığa sahip olması gerekmektedir.

Optimizasyon problemi için amaç fonksiyonu ise,

$$W = \sum_{k=1}^{N_d} m_k \sum_{i=1}^{N_k} l_i \tag{4.1}$$

olarak belirlenmiştir.

(4.1) numaralı denklem ızgara sistemin ağırlığını göstermektedir.

$N_d =$ sistemdeki toplam grup sayısı

$m_k =$ *k* numaralı gruba atanan hazır profil kesitinin birim ağırlığı

$l_i =$ *k* numaralı gruptaki elemanların uzunluğu

$N_k =$ *k* grubundaki toplam eleman sayısı

Yapıda deplasmanlar sınırlandırıldığında kalıcı hasarların oluşması engellenmiş olur. Bu amaçla uygulanması gereken düşey deplasman sınırlayıcısı,

$$\delta_j / \delta_{ju} \leq 1 \qquad ve \qquad j = 1,2,....., p \tag{4.2}$$

dikkate alınır.

Burada,

$\delta_j =$ *j* numaralı birleşim noktasında oluşan deplasman

$\delta_{ju} =$ deplasman sınır değeri

$p =$ deplasman açısından sınırlandırılmış eleman sayısı

gösterilmektedir. Kirişlerde düşey deplasman sınır değeri için kiriş açıklığının 1/300'ü, kolonlarda yatay deplasman sınır değeri için ise kat yüksekliğinin 1/300'ü alınmaktadır.

(4.3) numaralı denklem LRFD şartnamesi uyarınca kirişler için dayanım sınırlayıcısıdır.

$$M_{ur} /(\phi_b M_{nr}) \leq 1 \quad ve \quad r = 1,2,....., nm \tag{4.3}$$

$M_{ur} =$ *r* elemanına ait servis momenti

$M_{nr} =$ sınır moment dayanımı

$\text{Ø}_\text{b} =$ eğilmeye karşı direnç faktörüdür ve değeri 0.9 olarak verilir.

$nm =$ eleman sayısı

LRFD' ye göre kesme dayanımı,

$$V_{ur} /(\phi_v V_{nr}) \leq 1 \qquad ve \qquad r = 1,2,....., nm \tag{4.4}$$

ile kontrol edilir.

Burada,

$V_{ur} =$ *r* elemanına ait servis kesme kuvveti

$V_{nr} =$ sınır kesme dayanımı

$Ø_{v} =$ kesme için direnç faktörü olup 0.9 olarak alınır.

$nm =$ eleman sayısı

Bu çalışmada, tüm deplasman sınırlayıcılarını ve gerilme koşullarını sağlayan çözümler arasında optimum olan yani yapı ağırlığının en düşük olduğu çözüm ateş böceği algoritması kullanılarak elde edilmeye çalışılmıştır.

Bu bölümde bir önceki bölümde işleyiş mantığı detaylı olarak ifade edilmiş olan ateş böceği optimizasyon yöntemi kullanılarak ızgara sistemlerin optimum tasarımını yapan bir algoritma geliştirilmiştir.

4.2 Ateş Böceği Algoritması Optimizasyon Yöntemi

Optimum boyutlandırma problemi aşağıdaki gibi ifade edilebilir.

$$\begin{aligned} &Min. \quad f(x_i) \qquad i = 1,....,n: \\ &g_j(x_i) \le 0 \qquad j = 1,....,m \\ &x_i \in X \quad , \quad X = \{x_1, x_2, ..., x_q\} \end{aligned} \tag{4.5}$$

Burada x_i , q sayısı kadar değer içeren X setinden seçilecek olan i numaralı tamsayı değişkenini gösterirken, n tasarım değişkenlerinin sayısını ifade eder. $f(x_i)$ amaç fonksiyonu, $g_j(x_i)$ ise j nolu tasarım sınırlayıcısıdır ve m ise boyutlandırma problemi içindeki bu tasarım sınırlayıcılarının toplam sayısıdır.

Ateş böceği algoritması optimizasyon yöntemi amaç fonksiyonu arama uzayı içinde rastgele seçilmiş olan ateş böceklerinden meydana gelmektedir. İlk olarak tamsayı değerlerini içeren X kümesindeki x_i değerleri artan sayılar şeklinde sıralanır. Bu sayıların satır numaraları tasarım değişkeni olarak atanır. Örnek vermek gerekirse, 272 adet değer içeren tasarım setindeki 1 ile 272 arasındaki satır numaraları ana tasarım değişkenleridir. Tasarım döngüsünün herhangi bir adımında, algoritma tarafından satır numarasının oluşturulmasıyla birlikte bu satır numarasına

karşılık gelen gerçek tasarım değişkeni değeri set içerisinden alınabilir. Uygulanan işlem altı temel adımdan oluşmaktadır.

1. Adım: Bir ateş böceği topluluğu küme boyutunu temsil eden ve önceden belirlenmiş miktarda ateş böceğinden oluşur. Bu amaçla topluluk içindeki ateş böceklerinin başlangıç konumları tanımlanır. Ateş böceklerinin başlangıç konumları x_i^0 ile gösterilmektedir. Her bir ateş böceğinin başlangıç konumu denklem 4.2'de tanımlanmıştır.

$$x_i^0 = X_{\min} + r\left(X_{\max} - X_{\min}\right) \tag{4.6}$$

buradaki r değerinin tespiti için 0 ile 1 arasında değişen sayılar içinden rastgele sayı seçme uygulamasından faydalanılır. $X_{\min}$ 1 e eşittir ve $X_{\max}$ set içindeki toplam değer sayısıdır.

2. Adım: Daha sonra tasarım değişkenlerini temsil eden konum vektörleri x_i^t kullanılarak bütün ateş böcekleri için uyumluluk analizi yapılır ve amaç fonksiyonu değerleri $f\left(x_i^t\right)$ hesaplanır.

3. Adım: Ateş böceklerinin en iyi değerinin ve topluluk içindeki en iyi ateş böceği değerinin güncellenmesi işlemi yapılır. Bir ateş böceğinin o ana kadar ki en iyi konumu a_i^t (minimum amaç fonksiyonu $f\left(x_i^t\right)$ değerine sahip olan en iyi boyutlandırma) ateş böceğinin en iyi değeri olarak kabul edilir ve her bir ateş böceğinin en iyi değeri olarak kaydedilir. Bu işlemin başlangıcından itibaren herhangi bir ateş böceği tarafından elde edilmiş olan en iyi konum ise en iyi global konum a_g^t olarak kaydedilir. Her iterasyon adımında her bir ateş böceği ve topluluğu için en iyi pozisyon değerleri güncellenir.

4. Adım: Her bir ateş böceğinin mevcut konumu, en iyi konumu ve global en iyi konumu dikkate alınarak güncellenir.

5. Adım: Bir önceki adımda güncellenen konum vektörü kullanılarak her bir ateş böceğine ait satır numarası değeri yenilenir.

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j^t - x_i^t\right) + \alpha_t \varepsilon_i^t \tag{4.7}$$

Buradaki x_i^{t+1} ifadesi $i.$ nci ateş böceğinin $(t+1).$ nci adımdaki konumunu belirtmektedir.

6. Adım: Algoritmanın son adımı iterasyon döngüsünün tamamlandırılmasıdır. Optimizasyon döngüsü, işleme başlamadan önce belirlenmiş olan maksimum iterasyon sayısı kadar tekrar

edilerek durdurulur. Bu adımlar sonunda elde edilen amaç fonksiyonu değeri en iyi tasarımı verir.

4.3 Ateş Böceği Algoritması Optimizasyon Yöntemi ile Izgara Sistemlerin Analizi

En düşük yapı ağırlığının hedeflendiği optimizasyon problemi için kullanılacak olan ateş böceği optimizasyon algoritması için FORTRAN programlama dili kullanılması suretiyle bir kod dizini hazırlanmıştır. Ateş böceği algoritması için; optimum çözümün aranmasında aşağıdaki adımlar takip edilmelidir.

- Öncelikle ele alınacak olan yapının geometrisi ve yükleri belirlenmelidir. Yapı elemanları gruplara ayrılmalıdır.
- Algoritmanın başlayabilmesi için, ateş böceklerinin konumu belirlenmelidir. Bu konumlar optimizasyon problemlerinde profil kesitlerini belirleyen sıra numaralarıdır.
- Izgara sistem, matris deplasman yöntemi kullanılarak analiz edilir ve eleman kuvvetleri ile birlikte deplasmanlar hesaplanır.
- Sınırlayıcıların sağlayıp sağlamadığı incelenir. Tasarım sınırlayıcılarının kontrolünü literatürde yer alan en etkin sonuçları verdiği belirlenmiş olan geri dönüş (fly-back) mekanizması üstlenir. Bu mekanizmaya göre ilk olarak bütün ateş böcekleri analiz edilir ve sınırlayıcıları sağlayıp sağlamadığı kontrol edilir. Eğer bu ateş böceklerinden bir veya birkaçı sınırların dışında sonuç veriyorsa bunlar atılır ve yerlerine yenileri üretilir ve yeniden inceleme yapılır.
- Eğer tasarım sınırlayıcılarda sapma çok büyük değilse bu durumda bu ateş böcekleri iterasyon süresince sınırları sağlayabilen yeni ateş böcekleri üretebileceği düşünülerek çözüm içnde tutulur. Bunu yapabilmek için parçacıkların üretilmesi aşamasında hata payı büyük tutulur, ateş böceklerin üretilmesi işlemi tamamlanıp iterasyonlara geçildikten sonra adım adım küçültülerek son değeri olan 0.001 e ulaştırılır ve bu değer ile iterasyona devam edilir.
- Bir sonraki adımda yine tasarım sınırlayıcıları kontrol edilerek uygun çözümler arasından, daha önceki adımda belirlenen optimum çözümden daha iyi (daha hafif yapı) bir sonuç elde edilirse, bu değer yeni optimum çözüm olarak kaydedilir.
- Bu işlemler önceden belirlenmiş olan iterasyon sayısı kadar tekrarlanır ve elde edilmiş olan en hafif yapının elde edildiği çözüm optimum çözümdür.

4.4 Boyutlandırma Örnekleri

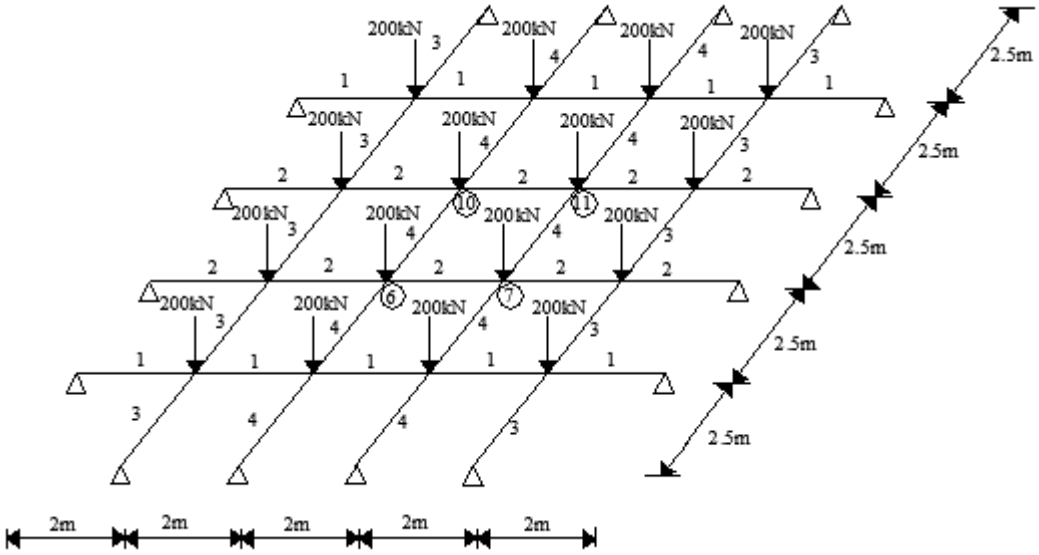
Bu bölümde; beş adet ızgara sistem ateş böceği algoritması kullanılarak elde edilmiş optimum boyutlandırma algoritmasıyla optimize edilmiştir. Izgara sisteminin boyutlandırılması yapılırken tasarım kısıtlayıcıları da hesaba katılmıştır. Profil kesitleri; en küçüğü W100x19.3mm ve en büyüğü W1100x499mm olmak üzere toplam 272 adet geniş başlıklı W-kesitten oluşan ve LRFD-AISC'de verilmiş olan profiller arasından uygun kesitler seçilerek belirlenmiştir. Bütün noktaların deplasmanları maksimum kiriş açıklığının 1/100 üne denk gelen maksimum değer ile sınırlandırılmıştır. Taşıyıcı kirişlerin kapasite değerleri LRFD-AISC şartnamesine göre hesaplanmıştır. Çelik için elastisite modülü olarak 200 kN/mm^2 kabul edilmiştir. Çalışmadaki ızgara sistemlerin tamamı; parçacık sürü optimizasyonu (PSO), harmoni arama (HA) ve ateş böceği algoritması optimizasyon (AAO) yöntemleriyle ayrı ayrı çözülmüş ve tekniklerin performans karşılaştırılması yapılmıştır. Tasarım problemlerinin çözümü için kullanılan her yöntemin optimum tasarım problemlerinde kullanılan parametreleri Çizelge 4.1'de verilmiştir. Beş tasarım örneği için bu parametreler aynı alınmıştır.

Çizelge 4.1 Optimizasyon tekniklerinin parametreleri

Yöntem	Parametre Değerleri
PSO	$\mu = 20,\ c_1 - c_2 = 1.2,\ N_{ite} = 100000,\ w = 0.13,\ V_{\max} = 2$
HA	$\mu = 50,\ hmcr = 0.90,\ par = 0.45, N_{cyc} = 100000$
AAO	$\mu = 40,\ \beta_0 = 0.25,\ \beta = 1,\ \alpha = 0.01,\ \gamma = 1,\ N_{ite} = 100000$

4.4.1 40- Elemanlı Basit Izgara Sistem

İlk tasarım örneği olarak Şekil 4.1'de görülen 40 elemanlı basit ızgara sistem ele alınmıştır. Sistemi oluşturan çelik yapı elemanları dört grup altında toplanmıştır. Dış ve iç enine kirişler birinci ve ikinci grup elemanları olarak düşünülürken, dış ve iç boyuna kirişler üçüncü ve dördüncü grup olarak hesaba katılmışlardır. Kiriş açıklıkları, düşey yükler ve eleman grup numaraları şekil üzerinde gösterilmiştir. Düşey yük olarak 200 kN büyüklüğünde tekil yükler varsayılarak düğüm noktalarına etkitilmiştir. 6, 7, 10 ve 11 nolu birleşim noktalarındaki düşey deplasmanlar 25 mm ile sınırlandırılmıştır.



Şekil 4.1 – 40- elemanlı basit ızgara sistem

40-elemanlı basit ızgara sistem, bahsedilmiş olan üç optimizasyon tekniği ile ayrı ayrı on kere analiz edilmiş ve elde edilen değerler Çizelge 4.2'de gösterilmiştir. Her bir yöntemin ürettiği optimum tasarımlar Çizelge 4.3'de ve bu tasarımlara ait minimum ağırlık değişim grafikleri ise Şekil 4.2'de gösterilmiştir.

En uygun sonuçlar parçacık sürü optimizasyon yönteminde 7842 iterasyon sonunda elde edilirken, harmoni arama yöntemi 4055 iterasyondan sonra optimum sonuca erişmiş ve ateş böceği optimizasyon yöntemi bu sonuca ancak 18037 iterasyon sonunda ulaşabilmiştir.

Çizelge 4.3'de görüldüğü üzere yapılan 10 farklı analiz sonucunda en hafif tasarımı 7138 kg ile PSO ve AAO yöntemi vermiştir ve söz konusu tasarım problemi için optimum çözüm olarak kabul edilmiştir. En kötü tasarım ise 7198.2 kg ile HA yöntemine aittir. Bu üç yönteme ait tasarımlar için her gruba ait kesit ataması Çizelge 4.2'de gösterilmiştir.

Çizelge 4.3'deki sonuçlar irdelendiğinde ateş böceği algoritması yöntemi ile 40- elemanlı basit ızgara sistemde 1 no'lu grup için W150X13.5, 2 no'lu grup için W150X13.5, 3 no'lu grup için W760X147 ve son olarak 4 no'lu grup için W840X176 optimum tasarıma ait kesitler olarak belirlenmiştir. Bu kesitlerin kullanımıyla oluşan sistemin ağırlığı ise 7138 kg dır. Maksimum deplasman 6, 7, 10 ve 11 numaralı birleşim noktalarında ve 24.6 mm olarak ölçülmüştür. Moment sınırlayıcısı için maksimum dayanım oranı 0.99 olup bu değer 3 numaralı elemanda görülmüştür. Bu sonuçlar neticesinde dayanım sınırlayıcılarının bu boyutlandırma problemi için daha baskın olduğu belirlenmiştir.

Çizelge 4.2 40- elemanlı basit ızgara sistem için PSO, HA ve AAO yöntemlerine ait ağırlık sonuçları

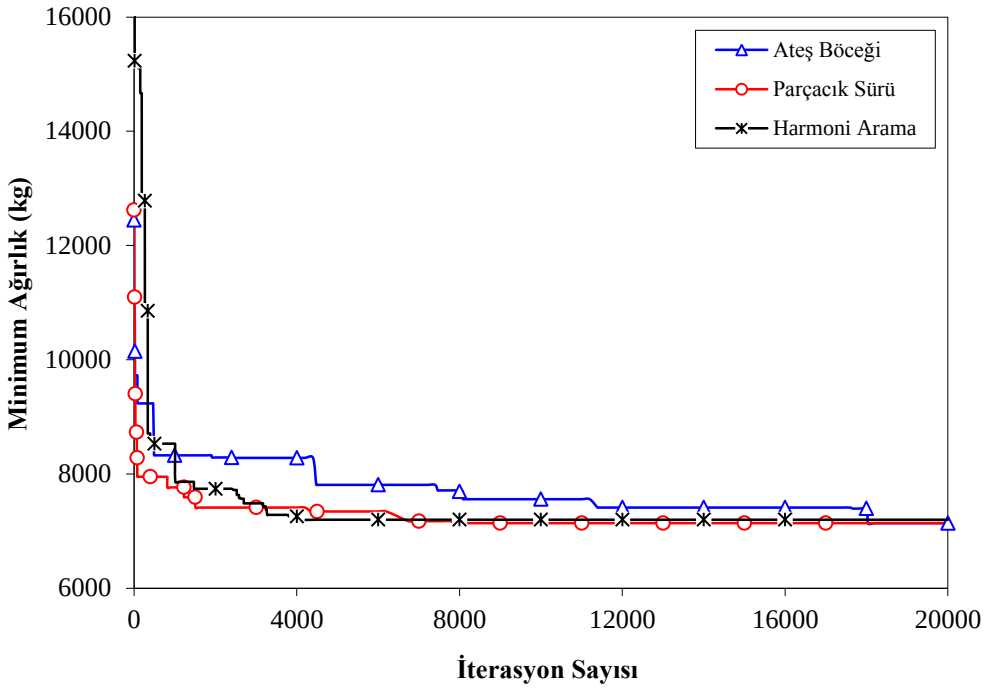
Çözüm Numarası	PSO için ağırlık (kg)	HA için ağırlık (kg)	AAO için ağırlık (kg)
1	7213	7351,2	7213
2	7213	7198,2	7213
3	7138	7351,2	7213
4	7138	7351,2	7213
5	7213	7336,4	7138
6	7213	7198,2	7292,5
7	7213	7198,2	7213
8	7213	7198,2	7213
9	7138	7198,2	7292,5
10	7138	7351,2	7213

Çizelge 4.3 40-elemanlı ızgara sisteminin optimum tasarım sonuçları

Metod	Optimum W-Kesit Grup No.		$\delta_{\max}$ (mm)	Maks. dayanım oranı	Minimum ağırlık (kg)
Parçacık sürü optimizasyonu (PSO)	1 2 3 4	W150X13.5 W150X13.5 W760X147 W840X176	24.6	0.99	7138
Harmoni arama (HA)	1 2 3 4	W410X46.1 W460X52 W200X15 W1000X222	23.6	0.99	7198.2
Ateş böceği algoritması (AAO)	1 2 3 4	W150X13.5 W150X13.5 W760X147 W840X176	24.6	0.99	7138

Aynı soru parçacık sürü optimizasyon yöntemi ile de çözülmüş ve aynı sonuçlar elde edilmiştir. Fakat çözüm için harmoni arama yöntemi kullanıldığında %0.84 daha ağır bir optimum tasarım elde edildiği görülmüştür. Minimum ızgara sistem ağırlığı için; parçacık sürü optimizasyon

yöntemi, harmoni arama ve ateş böceği optimizasyonu olmak üzere üç farklı arama yöntemi kullanılarak elde edilen tasarımların yakınsama grafikleri Şekil 4.2'de verilmiştir.

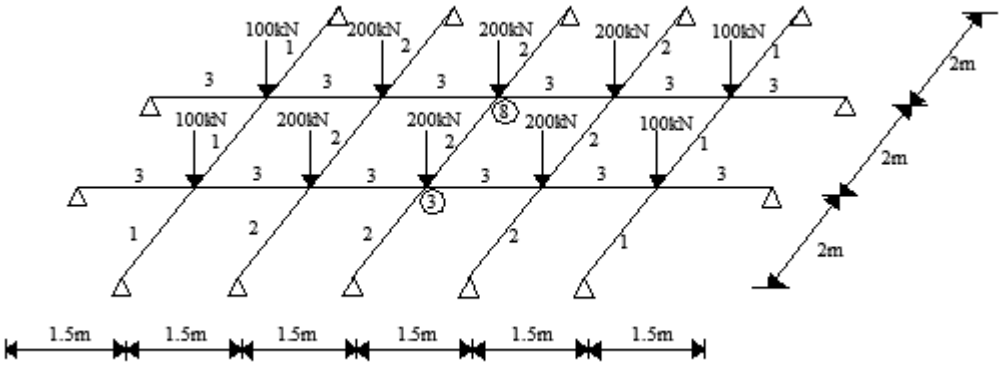


Şekil 4.2 40- elemanlı basit ızgara sistem için minimum ağırlık zaman grafiği

40- elemanlı ızgara sistem; harmoni arama yöntemiyle 46 dakikada, parçacık sürü optimizasyon yöntemiyle 1 saat 18 dakikada ve ateş böceği optimizasyon yöntemiyle de 2 saat 26 dakikada 100000 iterasyona ulaşmıştır.

4.4.2 27- Elemanlı Basit Izgara Sistem

İkinci örnek olarak Şekil 4.3'te görülen 27 elemanlı basit ızgara sistem ele alınmıştır. Sistemi oluşturan çelik yapı elemanları üç grup altında toplanmıştır. Dıştaki boyuna kirişler üçüncü grup elemanları olarak düşünülürken, dış ve iç enine kirişler birinci ve ikinci grup olarak hesaba katılmışlardır. Kiriş açıklıkları, düşey yükler ve eleman grup numaraları şekil üzerinde gösterilmiştir. Düşey yük olarak 100 kN ve 200 kN büyüklüğünde tekil yükler varsayılmış ve bu yükler her bir düğüm noktasına etkitilmiştir. 3 ve 8 nolu birleşim noktalarındaki düşey deplasmanlar 20mm ile sınırlandırılmıştır.



Şekil 4.3 27- elemanlı basit ızgara sistem

Optimizasyon problemi HA, AAO ve PSO ile çözülmüş ve üç yönteme ait 10 farklı analiz sonucu elde edilen ağırlık değerleri Çizelge 4.4'te gösterilmiştir. En uygun sonuçlar ateş böceği optimizasyon yönteminde 312 iterasyon sonunda elde edilirken, parçacık sürü optimizasyon yöntemi 750 iterasyondan sonra optimum tasarımı bulmuş, harmoni arama yöntemi bu sonuca 368 iterasyon sonunda ulaşmıştır.

Çizelge 4.4 27- elemanlı basit ızgara sistem için PSO, HA ve AAO yöntemlerine ait ağırlık sonuçları

Çözüm Numarası	PSO için ağırlık (kg)	HA için ağırlık (kg)	AAO için ağırlık (kg)
1	2064,9	2064,9	2064,9
2	2064,9	2064,9	2064,9
3	2064,9	2064,9	2064,9
4	2064,9	2064,9	2064,9
5	2064,9	2064,9	2064,9
6	2064,9	2064,9	2064,9
7	2064,9	2064,9	2064,9
8	2064,9	2064,9	2064,9
9	2064,9	2064,9	2064,9
10	2064,9	2064,9	2064,9

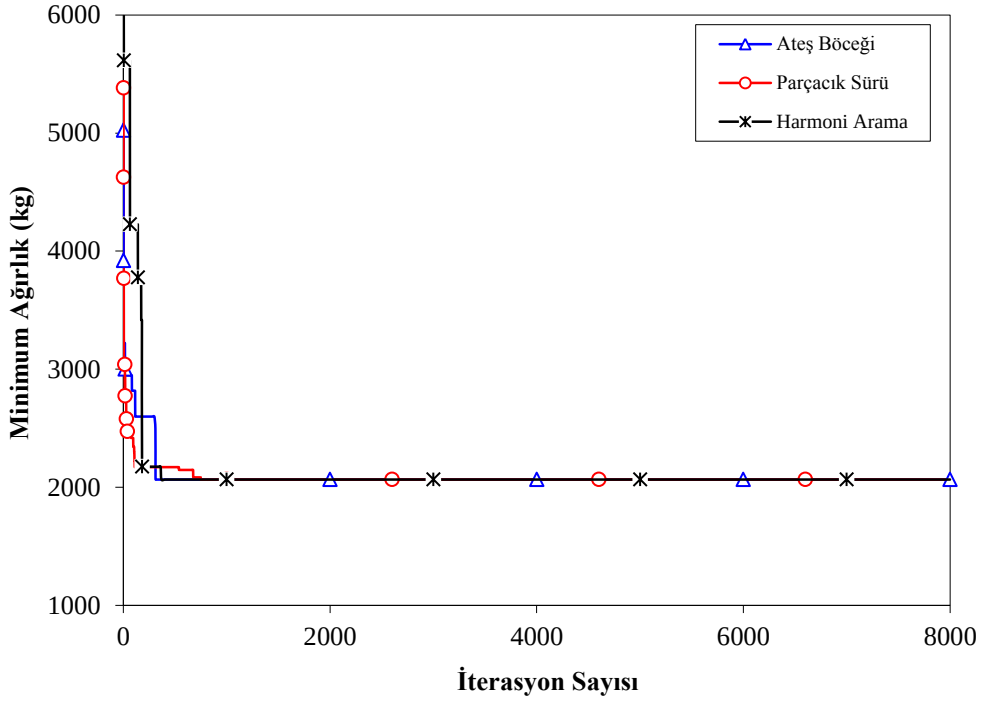
Çizelge 4.5'te görüldüğü üzere probleme ait optimum tasarım her üç yöntemde de bulunmuş ve bu tasarımın ağırlığı 2064.9 kg olarak kaydedilmiştir. Bu tasarım için her gruba ait kesit ataması da Çizelge 4.5'te gösterilmiş ve söz konusu tasarım problemi için optimum çözüm olarak kabul edilmiştir.

Çizelge 4.5 27-elemanlı ızgara sisteminin optimum tasarım sonuçları

Metod	Optimum W-Kesit Grup No.		$\delta_{\max}$ (mm)	Maks. dayanım oranı	Minimum ağırlık (kg)
Parçacık sürü optimizasyonu (PSO)	1 2 3	W150X13.5 W610X82 W310X23.8	12.6	0.98	2064.9
Harmoni arama (HA)	1 2 3	W150X13.5 W610X82 W310X23.8	12.6	0.98	2064.9
Ateş böceği algoritması (AAO)	1 2 3	W150X13.5 W610X82 W310X23.8	12.6	0.98	2064.9

Çizelge 4.5'teki sonuçlar irdelendiğinde ateş böceği algoritması yöntemi ile 27- elemanlı ızgara sistemde 1 no'lu grup için W150X13.5, 2 no'lu grup için W610X82 ve 3 no'lu grup için W310X23.8 optimum tasarıma ait kesitler olarak belirlenmiştir. Bu kesitlerin kullanımıyla oluşan sistemin ağırlığı ise 2064.9 kg dır. Maksimum deplasman 3 ve 8 numaralı birleşim noktalarında ve 12.6 mm olarak ölçülmüştür. Moment sınırlayıcısı için maksimum dayanım oranı 0.98 olup bu değer 1 numaralı elemanda görülmüştür. Bu sonuçlar doğrultusunda dayanım sınırlayıcılarının bu boyutlandırma problemi için daha etkin olduğu anlamına gelir.

Aynı soru parçacık sürü optimizasyon ve harmoni arama yöntemi ile de çözülmüş ve aynı sonuçlar elde edilmiştir. Minimum ızgara sistem ağırlığı için; parçacık sürü optimizasyon yöntemi, harmoni arama ve ateş böceği optimizasyonu olmak üzere üç farklı arama yöntemi kullanılarak elde edilen tasarımların yakınsama grafikleri Şekil 4.4'te verilmiştir.

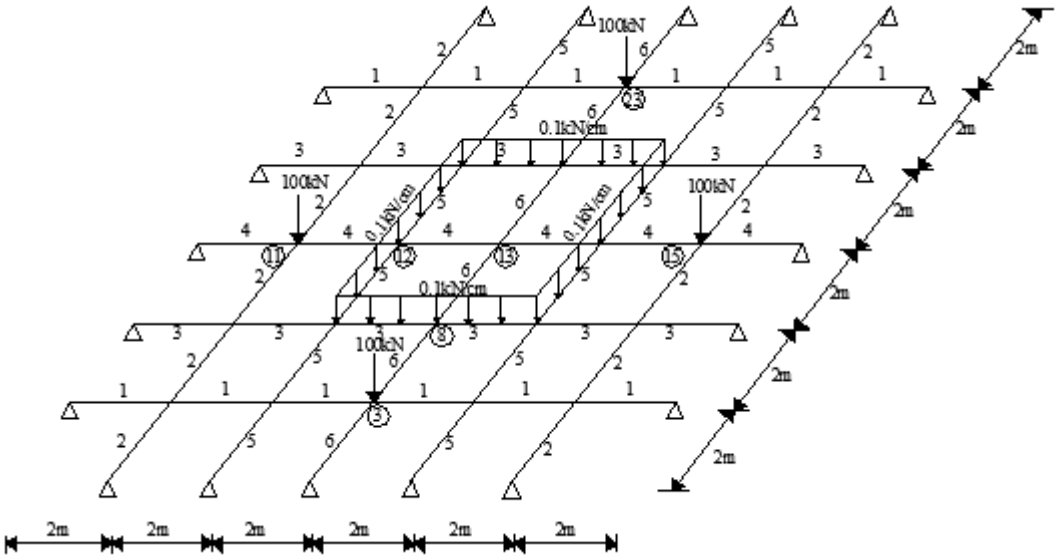


Şekil 4.4 27-elemanlı basit ızgara sistem için minimum ağırlık zaman grafiği

27- elemanlı ızgara sistem; harmoni arama yöntemiyle 27 dakikada, parçacık sürü optimizasyon yöntemiyle 50 dakikada ve ateş böceği optimizasyon yöntemiyle de 1 saat 7 dakikada 100000 iterasyona ulaşmıştır.

4.4.3 60- Elemanlı Basit Izgara Sistem

Üçüncü örnek olarak Şekil 4.5'te görülen 60 elemanlı basit ızgara sistem ele alınmıştır. Sistemi oluşturan çelik yapı elemanları altı grup altında toplanmıştır. Dış ve iç boyuna kirişler sırasıyla birinci, üçüncü ve dördüncü grup elemanları olarak düşünülürken, dış ve iç enine kirişler sırasıyla ikinci, beşinci ve altıncı grup olarak hesaba katılmışlardır. Kiriş açıklıkları, düşey yükler ve eleman grup numaraları şekil üzerinde gösterilmiştir. 0.1 kN/cm büyüklüğündeki yayılı yükler kiriş elemanlara ve 100 kN büyüklüğünde tekil yükler varsayılarak düğüm noktalarına etkitilmiştir. 3, 8, 11, 12, 13, 15 ve 23 nolu birleşim noktalarındaki düşey deplasmanlar 20 mm ile sınırlandırılmıştır.



Şekil 4.5 60- elemanlı basit ızgara sistem

60-elemanlı basit ızgara sistem, üç yöntemle de çözülmüş ve 10 farklı analiz sonucu elde edilen ağırlık değerleri Çizelge 4.6'de gösterilmiştir. En uygun sonuçlar ateş böceği optimizasyon yönteminde 31955 iterasyon sonunda elde edilirken, harmoni arama yöntemi 8662 iterasyondan sonra optimum sonuca erişmiş ve parçacık sürü optimizasyon yöntemi bu sonuca ancak 52137 iterasyon sonunda ulaşmıştır. Çizelge 4.7'te görüldüğü üzere probleme ait optimum tasarım PSO ve AAO yöntemi ile bulunmuş ve bu tasarımın ağırlığı 4169 kg olarak kaydedilmiştir. En kötü tasarım ise 4200.8 kg ile HA yöntemine aittir. Bu tasarım için her gruba ait kesit ataması da Çizelge 4.7'da gösterilmiş ve söz konusu tasarım problemi için optimum çözüm olarak kabul edilmiştir.

Çizelge 4.7'daki sonuçlar irdelendiğinde ateş böceği algoritması yöntemi ile 60- elemanlı ızgara sistemde 1 no'lu grup için W150X13.5, 2 no'lu grup için W150X13.5, 3 no'lu grup için W150X13.5, 4 no'lu grup için W920X201, 5 no'lu grup için W150X13.5 ve 6 no'lu grup için W410X38.8 optimum tasarıma ait kesitler olarak belirlenmiştir. Bu kesitlerin kullanımıyla oluşan sistemin ağırlığı ise 4169 kg dır. Maksimum deplasman 8 numaralı birleşim noktasında ve 19.8 mm olarak ölçülmüştür. Moment sınırlayıcısı için maksimum dayanım oranı 0.76 olup bu değer 3 numaralı elemanda görülmüştür. Bu sonuçlar doğrultusunda deplasman sınırlayıcılarının bu boyutlandırma problemi için daha dominant olduğu belirlenmiştir.

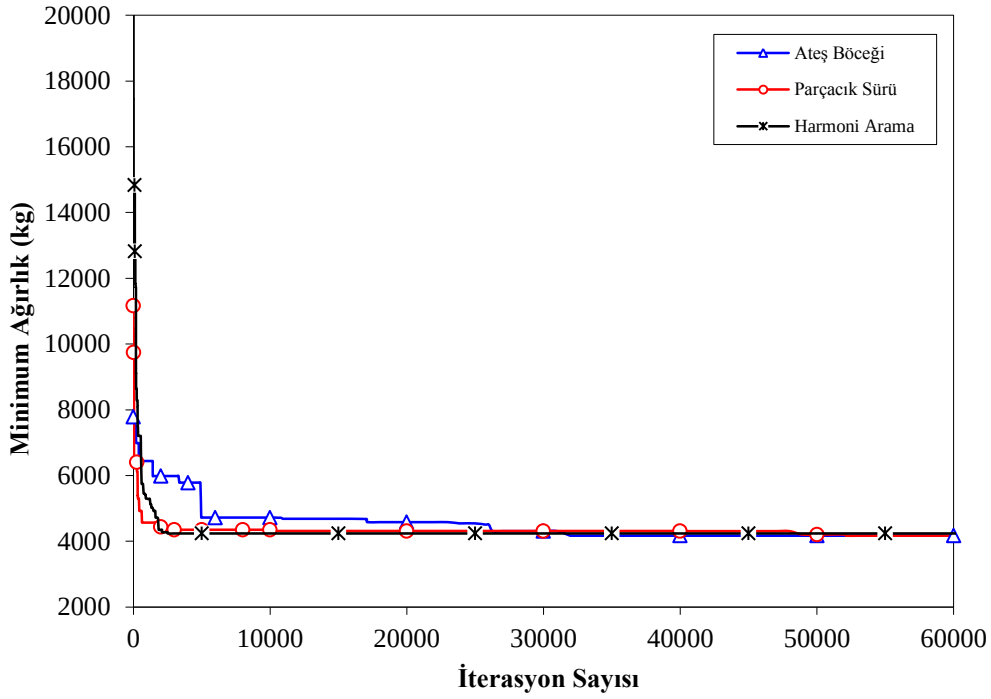
Çizelge 4.6 60- elemanlı basit ızgara sistem için PSO, HA ve AAO yöntemlerine ait ağırlık sonuçları

Çözüm Numarası	PSO için ağırlık (kg)	HA için ağırlık (kg)	AAO için ağırlık (kg)
1	4380,7	4200,8	4169
2	4311	4200,8	4200,8
3	4169	4200,8	4200,8
4	4564,3	4200,8	4169
5	4237,5	4200,8	4200,8
6	4200,8	4200,8	4200,8
7	4355	4200,8	4169
8	4200,8	4200,8	4169
9	4268,1	4200,8	4169
10	4169	4200,8	4169

Çizelge 4.7 60-elemanlı ızgara sisteminin optimum tasarım sonuçları

Metod	Optimum W-Kesit Grup No.		$\delta_{\max}$ (mm)	Maks. dayanım oranı	Minimum ağırlık (kg)
Parçacık sürü optimizasyonu (PSO)	1	W150X13.5	19.8	0.76	4169
	2	W150X13.5			
	3	W150X13.5			
	4	W360X38.8			
	5	W150X13.5			
	6	W920X201			
Harmoni arama (HA)	1	W150X13.5	19.9	0.47	4200.8
	2	W150X13.5			
	3	W150X13.5			
	4	W840X176			
	5	W150X13.5			
	6	W530X66			
Ateş böceği algoritması (AAO)	1	W150X13.5	19.8	0.76	4169
	2	W150X13.5			
	3	W150X13.5			
	4	W920X201			
	5	W150X13.5			
	6	W410X38.8			

Aynı soru parçacık sürü optimizasyon yöntemi ile de çözülmüş ve aynı sonuçlar elde edilmiştir. Fakat çözüm için harmoni arama yöntemi kullanıldığında %0.76 daha ağır bir optimum tasarım elde edildiği görülmüştür. Minimum ızgara sistem ağırlığı için; parçacık sürü optimizasyon yöntemi, harmoni arama ve ateş böceği optimizasyonu olmak üzere üç farklı arama yöntemi kullanılarak elde edilen tasarımların yakınsama grafikleri Şekil 4.6'da verilmiştir.

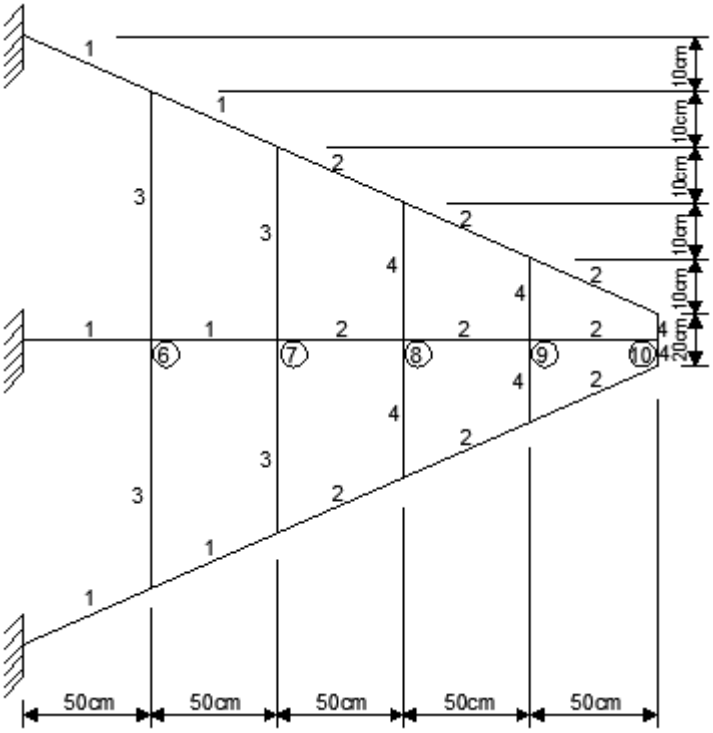


Şekil 4.6 60-elemanlı ızgara sistem için minimum ağırlık zaman grafiği

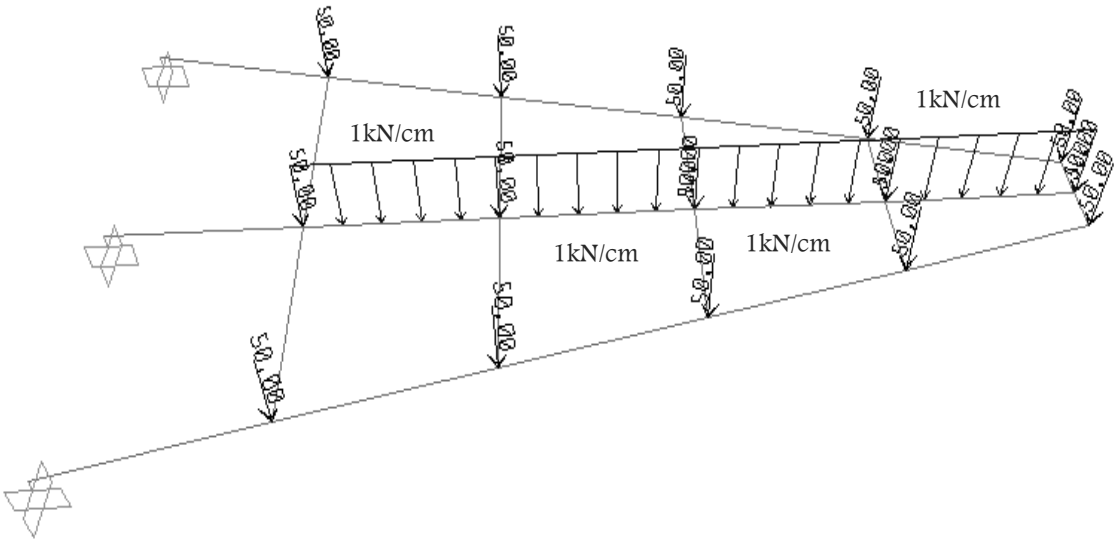
60- elemanlı ızgara sistem; harmoni arama yöntemiyle 1 saat 30 dakikada, parçacık sürü optimizasyon yöntemiyle 2 saat 36 dakikada ve ateş böceği optimizasyon yöntemiyle de 4 saatte 100000 iterasyona ulaşmıştır.

4.4.4 25- Elemanlı Ankastre Izgara Sistem

Şekil 4.7'de görülen 25 elemanlı ankastre ızgara sistem dördüncü boyutlandırma örneği olarak dikkate alınmıştır. Sistemi oluşturan çelik yapı elemanları dört grup altında toplanmıştır. Grup numaraları ve kiriş açıklıkları Şekil 4.7'de belirtildiği gibidir. Düşey yükler Şekil 4.8'de gösterilmiştir. Kirişlere 1 kN/cm büyüklüğünde yayılı yükler, düğüm noktalarına ise 50 kN büyüklüğünde tekil yükler uygulanmıştır. 6, 7, 8, 9 ve 10 nolu birleşim noktalarındaki düşey deplasmanlar 5 mm ile sınırlandırılmıştır.



Şekil 4.7 25- elemanlı ankastre ızgara sisteminin grup numaraları ve kiriş açıklıkları (plan)



Şekil 4.8 25- elemanlı ankastre ızgara sisteminin yükleme durumu

25- elemanlı ankastre ızgara sistem, bahsedilmiş olan üç optimizasyon tekniği ile ayrı ayrı on kere analiz edilmiş ve elde edilen tasarımlar Çizelge 4.8'te gösterilmiştir. En uygun sonuçlar parçacık sürü optimizasyon yönteminde 1218 iterasyon sonunda elde edilirken, harmoni arama yöntemi 5950 iterasyondan sonra optimum sonuca erişmiş, ateş böceği optimizasyon yöntemi bu sonuca ancak 4505 iterasyon sonunda ulaşmıştır.

Çizelge 4.8 25- elemanlı ankastre ızgara sistem için PSO, HA ve AAO yöntemlerine ait ağırlık sonuçları

Çözüm Numarası	PSO için ağırlık (kg)	HA için ağırlık (kg)	AAO için ağırlık (kg)
1	720,27	720,27	720,27
2	720,27	720,27	720,27
3	720,27	720,27	720,27
4	720,27	720,27	720,27
5	720,27	720,27	720,27
6	720,27	720,27	720,27
7	720,27	720,27	720,27
8	720,27	720,27	720,27
9	720,27	720,27	720,27
10	720,27	720,27	720,27

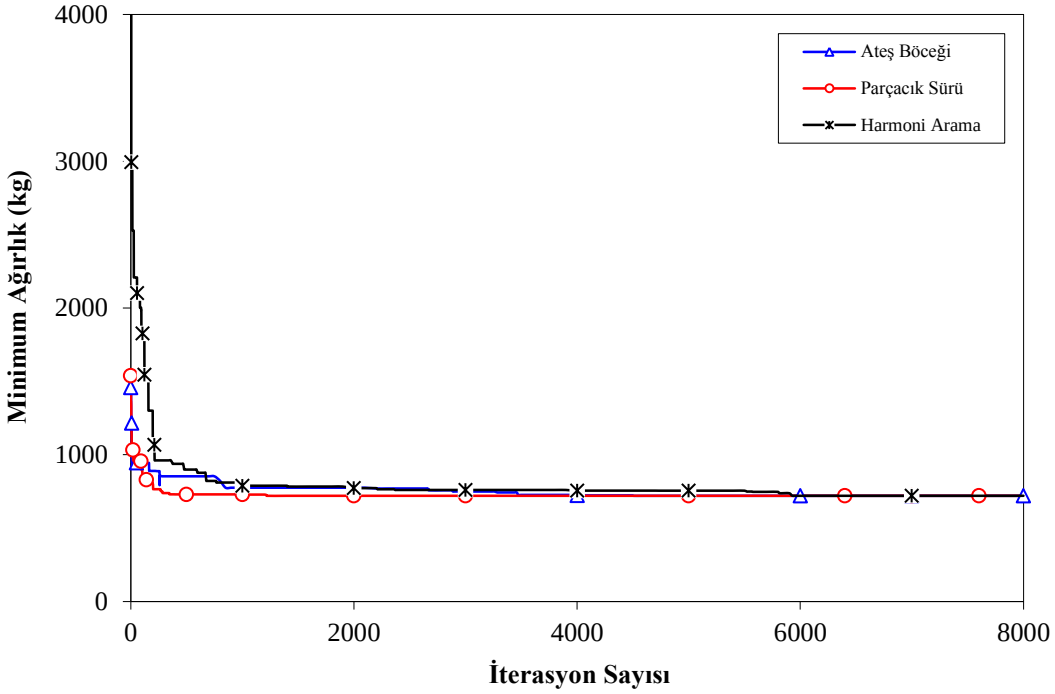
Üretilen bu en iyi tasarımlara ait sonuç değerleri Çizelge 4.9'da verilmiştir. Bu çizelgede görüldüğü üzere üç yöntem de optimum tasarım olarak aynı tasarımı üretmiş ve bu tasarımın ağırlığı 720.27 kg olarak kaydedilmiştir. Bu tasarım için her gruba ait kesit ataması da çizelgede gösterilmiş ve söz konusu tasarım problemi için optimum çözüm olarak kabul edilmiştir.

Çizelge 4.9 25- elemanlı ankastre ızgara sisteminin optimum tasarım sonuçları

Metod	Optimum W-Kesit Grup No.		$\delta_{\max}$ (mm)	Maks. dayanım oranı	Minimum ağırlık (kg)
Parçacık sürü optimizasyonu (PSO)	1 2 3 4	W760X134 W410X60 W150X13.5 W150X13.5	5	0.73	720.27
Harmoni arama (HA)	1 2 3 4	W760X134 W410X60 W150X13.5 W150X13.5	5	0.73	720.27
Ateş böceği algoritması (AAO)	1 2 3 4	W760X134 W410X60 W150X13.5 W150X13.5	5	0.73	720.27

Çizelge 4.9'daki sonuçlar irdelendiğinde ateş böceği algoritması yöntemi ile 25-elemanlı ankastre ızgara sistemde 1 no'lu grup için W760X134, 2 no'lu grup için W410X60, 3 no'lu grup için W150X13.5 ve son olarak 4 no'lu grup için W150X13.5 kesitleri optimum tasarıma ait kesitler olarak belirlenmiştir. Bu kesitlerin kullanımıyla oluşan sistemin ağırlığı ise 720.27 kg dır. Maksimum deplasman 10 numaralı birleşim noktasında ve 5 mm olarak ölçülmüştür. Moment sınırlayıcısı için maksimum dayanım oranı 0.73 olup bu değer 15 numaralı elemanda görülmüştür. Bu sonuçlar neticesinde deplasman sınırlayıcılarının bu boyutlandırma problemi için daha baskın olduğu kanısına varılır.

Aynı problem parçacık sürü optimizasyon ve harmoni arama yöntemi ile de çözülmüş ve aynı sonuçlar elde edilmiştir. Minimum ızgara sistem ağırlığı için; parçacık sürü optimizasyon yöntemi, harmoni arama ve ateş böceği optimizasyonu olmak üzere üç farklı arama yöntemi kullanılarak elde edilen tasarımların yakınsama grafikleri Şekil 4.9'da verilmiştir.

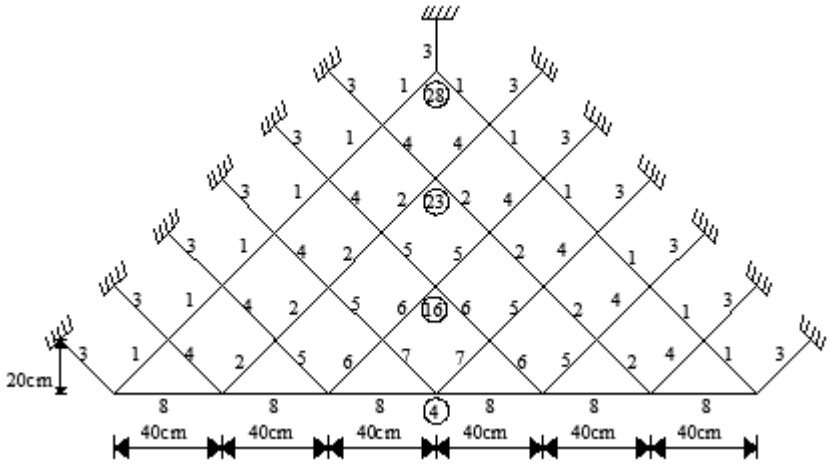


Şekil 4.9 25-elemanlı ankastre ızgara sistem için minimum ağırlık zaman grafiği

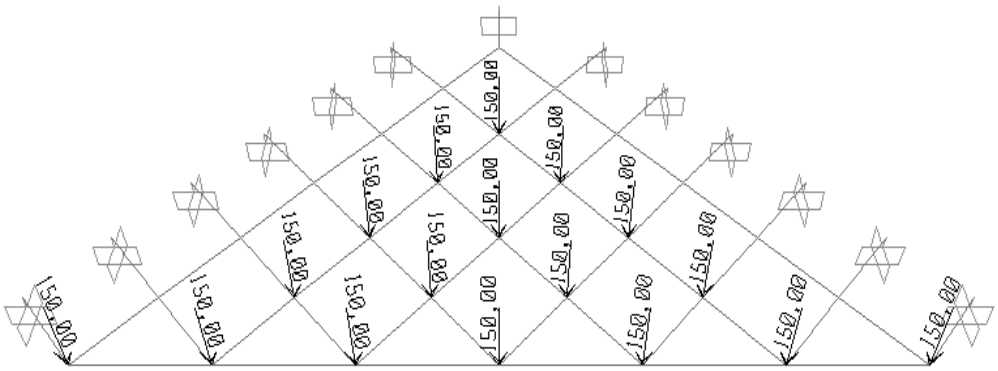
25- elemanlı ankastre ızgara sistem; harmoni arama yöntemiyle 55 dakikada, parçacık sürü optimizasyon yöntemiyle 1 saat 30 dakikada ve ateş böceği optimizasyon yöntemiyle de 2 saat 34 dakikada 100000 iterasyona ulaşmıştır.

4.4.5 61- Elemanlı Ankastre Izgara Sistem

Son tasarım örneği olarak Şekil 4.10'da görülen 61 elemanlı ankastre ızgara sistem ele alınmıştır. Sistemi oluşturan çelik yapı elemanları sekiz grup altında toplanmıştır. Grup numaraları ve kiriş açıklıkları Şekil 4.10'da belirtildiği gibidir. Düşey yükler Şekil 4.11'da gösterilmiştir. Düşey yük olarak 150 kN büyüklüğünde tekil yükler varsayılarak düğüm noktalarına etkitilmiştir. 4, 16, 24 ve 28 nolu birleşim noktalarındaki düşey deplasmanlar 4mm ile sınırlandırılmıştır.



Şekil 4.10 61- elemanlı ankastre ızgara sisteminin grup numaraları ve kiriş açıklıkları (plan)



Şekil 4.11 61- elemanlı ankastre ızgara sisteminin yükleme durumu

61- elemanlı ankastre ızgara sistem, bahsedilmiş olan üç optimizasyon tekniği ile ayrı ayrı on kere analiz edilmiş ve elde edilen en iyi değerler Çizelge 4.10'da gösterilmiştir. En uygun sonuçlar ateş böceği optimizasyon yönteminde 28121 iterasyon sonunda elde edilirken, harmoni arama yöntemi 40985 iterasyondan sonra optimum tasarımı üretmiş, parçacık sürü optimizasyon yöntemi bu sonuca ancak 93134 iterasyon sonunda ulaşmıştır. Çizelge 4.11'de görüldüğü üzere en hafif tasarımı 453.14 kg ile AAO yöntemi vermiştir ve söz konusu tasarım problemi için optimum çözüm olarak kabul edilmiştir. HA yöntemi ile bulunan sistemin ağırlığı ise 454.98 kg dır. En kötü tasarım ise 486.12 kg ile PSO yöntemine aittir. Bu üç yönteme ait tasarımlar için her gruba ait kesit ataması da bu çizelgede verilmiştir.

Çizelge 4.10 61- elemanlı ankastre ızgara sistem için PSO, HA ve AAO yöntemlerine ait ağırlık sonuçları

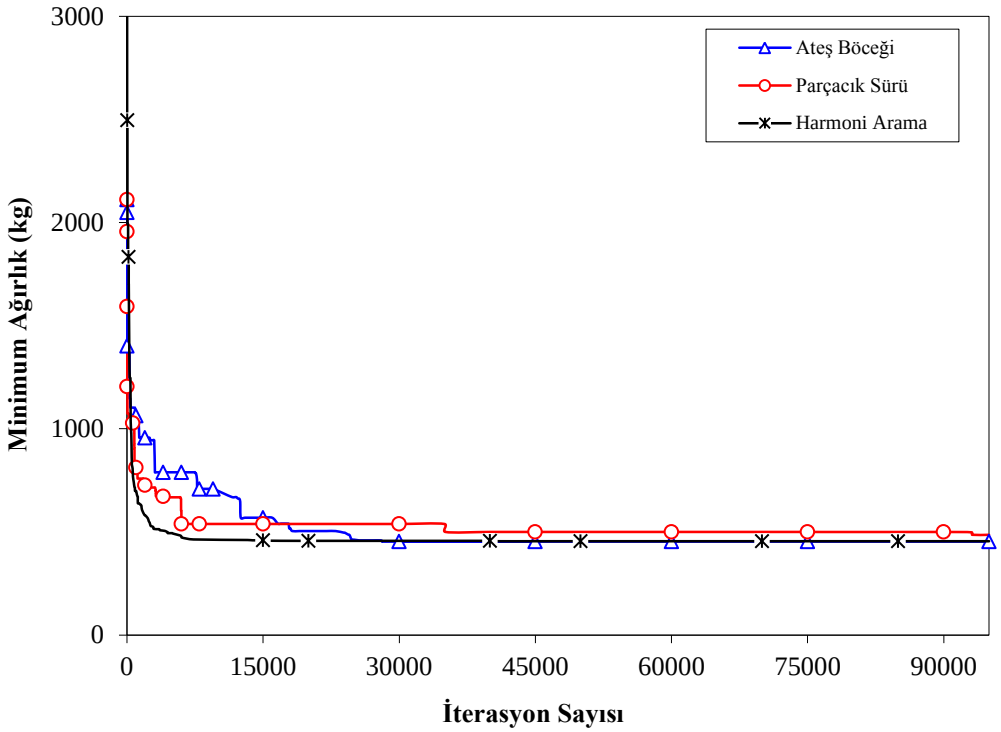
Çözüm Numarası	PSO için ağırlık (kg)	HA için ağırlık (kg)	AAO için ağırlık (kg)
1	533,63	454,98	458,21
2	511,8	463,74	499,91
3	505,35	463,74	455,4
4	519,18	454,98	453,14
5	489,74	463,74	455,4
6	486,12	463,74	456,71
7	521,51	454,98	463,74
8	507,24	454,98	458,92
9	568,69	463,74	453,25
10	504,89	463,74	455,4

Çizelge 4.11'daki sonuçlar irdelendiğinde ateş böceği algoritması yöntemi ile 61- elemanlı ankastre ızgara sistemde 1 no'lu grup için W150X13.5, 2 no'lu grup için W150X13.5, 3 no'lu grup için W410X46.1, 4 no'lu grup için W410X38.8, 5 no'lu grup için W250X17.9, 6 no'lu grup için W250X17.9, 7 no'lu grup için W150X13.5 ve son olarak 8 no'lu grup için W250X17.9 optimum tasarıma ait kesitler olarak belirlenmiştir. Bu kesitlerin kullanımıyla oluşan sistemin ağırlığı ise 453,14 kg dır. Maksimum deplasman 4 numaralı birleşim noktasında ve 2.2 mm olarak ölçülmüştür. Moment sınırlayıcısı için maksimum dayanım oranı 1.00 olup bu değer 51 numaralı elemanda görülmüştür. Bu sonuçlar neticesinde dayanım sınırlayıcılarının bu boyutlandırma problemi için daha etkin olduğu belirlenmiştir.

Çizelge 4.11 61-elemanlı ankastre ızgara sisteminin optimum tasarım sonuçları

Metod	Optimum W-Kesit Grup No.		$\delta_{\max}$ (mm)	Maks. dayanım oranı	Minimum ağırlık (kg)
Parçacık sürü optimizasyonu (PSO)	1	W250X25.3	1.6	0.83	486.12
	2	W250X25.3			
	3	W410X60			
	4	W410X46.1			
	5	W250X17.9			
	6	W250X17.9			
	7	W130X28.1			
	8	W310X21			
Harmoni arama (HA)	1	W150X13.5	2.1	1.00	454.98
	2	W150X13.5			
	3	W410X46.1			
	4	W410X38.8			
	5	W310X21			
	6	W200X15			
	7	W150X13.5			
	8	W250X17.9			
Ateş böceği algoritması (AAO)	1	W150X13.5	2.2	1.00	453.14
	2	W150X13.5			
	3	W410X46.1			
	4	W410X38.8			
	5	W250X17.9			
	6	W250X17.9			
	7	W150X13.5			
	8	W250X17.9			

Aynı örnek harmoni arama yöntemi ile de çözülmüş ve %0.4 daha ağır bir optimum tasarım elde edildiği görülmüştür. Fakat çözüm için parçacık sürü optimizasyon yöntemi kullanıldığında %7.2 daha ağır bir optimum tasarım elde edildiği görülmüştür. Minimum ızgara sistem ağırlığı için; parçacık sürü optimizasyon yöntemi, harmoni arama ve ateş böceği optimizasyonu olmak üzere üç farklı arama yöntemi kullanılarak elde edilen tasarımların yakınsama grafikleri Şekil 4.12'de verilmiştir.



Şekil 4.12 61- elemanlı ankastre ızgara sistem için minimum ağırlık zaman grafiği

61- elemanlı ankastre ızgara sistem; harmoni arama yöntemiyle 1 saat 40 dakikada, parçacık sürü optimizasyon yöntemiyle 2 saat 45 dakikada ve ateş böceği optimizasyon yöntemiyle de 4 saat 10 dakikada 100000 iterasyona ulaşmıştır.

5. SONUÇLAR VE ÖNERİLER

SONUÇLAR

Dört bölümden oluşan bu tez çalışmasında; güncel hayatta çeşitli alanlarda sıklıkla kullanılan bir yapı modeli olan ızgara sistemlerin, literatürde yer alan güçlü optimizasyon tekniklerinden biri olan ateş böceği algoritması tekniği kullanılarak optimum tasarımı üzerine araştırma yapılmıştır. Bu çalışmada ayrıca stokastik arama yöntemlerinden; parçacık sürü optimizasyon yöntemi, harmoni arama yöntemi ve ateş böceği optimizasyon yönteminin yapısal optimizasyon problemlerinin çözümünü bulmadaki performansları karşılaştırılmıştır.

Çalışmanın birinci bölümünde; optimizasyona giriş, yapısal optimizasyon kavramı, stokastik arama yöntemleri ve literatürde yer alan bu konularda yapılmış çalışmalar ile ilgili genel bilgilere yer verilmiştir.

Çalışmanın ikinci bölümünde; araştırmaya konu olan ateş böceğinin biyolojisine değinilmiştir. Ateş böceği algoritması optimizasyon yönteminin gelişimi ve kullanılan algoritma anlatılmıştır. Bu yöntem için FORTRAN programlama dili kullanılarak bir tasarım algoritması hazırlanmış ve bu algoritmanın performansını kontrol etmek üzere; literatürde yer alan Benchmark problemlerinin çözümleri araştırılmıştır. Bu Benchmark problemlerinde ateş böceği algoritması yöntemi ile elde edilen sonuçlar literatürden alınmış olan sonuçlarla mukayese edilmiş ve ateş böceği algoritmasının optimum çözümün elde edilmesi sürecindeki performansının oldukça iyi olduğu görülmüştür.

Çalışmanın üçüncü bölümünde; yatay düzlemde birbirine rijit bağlı elemanlardan oluşan ızgara sistemler ele alınmıştır. Yapısal analiz için kullanılan matris deplasman yöntemine de bu bölümde yer verilmiştir. Her iki ucu rijit elemanlar ile ilk ucu veya ikinci ucu mafsallı elemanlardan oluşan ızgara sistemlere ait rijitlik matrisinin oluşturulması anlatılıp, analitik modeli hazırlanmıştır. Izgara sistemlerin tasarımında esas alınan LRFD-AISC şartnamesindeki tasarım sınırlayıcıları hakkında bilgi verilmiştir.

Çalışmanın son bölümü olan dördüncü bölümde; ızgara sistemlerin optimum boyutlandırılmasında kullanılan matematiksel model anlatılmıştır. Ateş böceği algoritması optimizasyon yöntemini temel alarak ızgara sistemlerin minimum ağırlık tasarımını yapan algoritmanın işleyiş aşamaları altı adım altında toplanmış ve detaylandırılmıştır. LRFD-AISC'ye

göre tasarım sınırlayıcılarına uygun olarak yapı tasarımı yapılmıştır. Bu şartnameye göre boyutlandırma kriterlerini sağlayan çözümler içerisinde en hafif yapı olarak seçilecek olan çözüm için; ateş böceği optimizasyonu, parçacık sürü optimizasyonu yöntemi ile birlikte harmoni arama yöntemi yardımıyla ayrı ayrı arama yapılmıştır. 40- elemanlı basit ızgara sistem, 27-elemanlı basit ızgara sistem, 60- elemanlı basit ızgara sistem, 25- elemanlı ankastre ızgara sistem ve 61- elemanlı ankastre ızgara sistem yapıları örnek tasarım problemleri olarak kabul edilmiştir. Her bir optimizasyon yöntemi ile elde edilen sonuçların ağırlık kıyaslamaları yapılmıştır. Elde edilen sonuçlara göre her üç yöntem birbirine göre oldukça yakın sonuçlar vermektedir. Ele alınan ilk örnek olan 40- elemanlı basit ızgara sistemde en hafif tasarımı PSO ve AAO yöntemi vermiştir. En kötü tasarım ise HA yöntemine aittir. İkinci örnek olan 27-elemanlı basit ızgara sistem için yapılan optimizasyon işlemi için üç yöntemin elde ettiği tasarımların da aynı tasarımlar olduğu görülmüştür. Üçüncü boyutlandırma problemi olan 60-elemanlı basit ızgara sistemde en hafif tasarımı PSO ve AAO yöntemi vermiştir. En kötü tasarım ise HA yöntemine aittir. Dördüncü örnek olan 25- elemanlı ankastre ızgara sistem için yapılan optimizasyon işlemi için üç yöntemin elde ettiği tasarımların da aynı tasarımlar olduğu görülmüştür. Son örnek olan 61- elemanlı ankastre ızgara sistem için en hafif tasarımı AAO yöntemi vermiştir. En kötü tasarım ise PSO yöntemi tarafından üretilmiştir. Ayrıca harmoni arama, parçacık sürü optimizasyon yöntemi ve ateş böceği algoritması optimizasyonu yöntemlerinin 100000 iterasyona ulaşma süreleri belirlenmiştir. 100000 iterasyona en kısa sürede ulaşan harmoni arama yöntemi olmuştur. En uzun sürede ulaşanın ise ateş böceği algoritması optimizasyon yöntemi olduğu belirlenmiştir.

ÖNERİLER

Bu tez kapsamında, ızgara sistemler literatürde yer alan ateş böceği algoritması yöntemi yardımıyla optimum profil boyutlarına ve minimum ağırlığa sahip olacak şekilde modellenmiştir. İncelenen ateş böceği optimizasyon algoritması düzlemsel çelik çerçevelerin boyutlandırılmasında kullanılabilir.

Çalışmada kullanılan optimum tasarım problemleri, literatürde bulunan diğer bazı arama yöntemlerine uygulanmak suretiyle bu tekniklerin optimum çözümün bulunmasındaki performansları incelenebilir.

Optimum tasarımı yapılan ızgara sistem problemleri için kabul edilen eleman gruplandırmalarının değiştirilmesi suretiyle farklı eleman gruplandırmasının optimum ızgara sistem ağırlığına olan etkisi araştırılabilir.

KAYNAKLAR :

1. Akay, B., Nümerik optimizasyon problemlerinde yapay arı kolonisi (artificial bee colony) algoritmasının performans analizi, Doktora Tezi, Erciyes Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi, 2009.

2. Apostolopoulos, T., and Vlachos, A., Application of the firefly algorithm for solving the economic emissions load dispatch problem, Hindawi Publishing Corporation International Conference Journal of Combinatorics, Vol. 2011, 2011.

3. Arora, J.S., Optimization of structural and mechanical systems, World Scientific, Singapore, 59-195, 2007.

4. Babu, B.G., and Kannan, M., Lightning bugs, Resonance, 7(9), 49-55, 2002.

5. Bayar, G.Y., Çerçeve taşıyıcı sistemlerin optimum tasarımı, Dicle Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 2008.

6. Carlos, A., and Coello C., Theoretical and numerical constraint handling techniques used with evolutionary algorithm: A survey of the state of the art, Computer Methods in Applied Mechanics and Engineering 191:1245-1287, 2002.

7. Çağdaş, S., Yapı mekaniğinde nümerik metodlar ve matris-deplasman yöntemi, Türkmen Kitabevi, 2006.

8. Deb, K., Optimal design of a welded beam via genetic algorithms, American Institute of Aeronautics and Astronautics Journal 29 : 2013-2015, 1991.

9. Değertekin, S.Ö., Ülker, M., ve Hayalioğlu, M.S., Uzay çelik çerçevelerin tabu arama ve genetik algoritma yöntemleriyle optimum tasarımı, Technical Journal of Turkish Chamber of Civil Engineers, 13-3: 3917-3934, 2006.

10. Doğan, E., Optimum design of rigid and semi rigid steel sway frames including soil-structure interaction, A Thesis Submitted to The Graduate School Of Natural And Applied Sciences, METU, 2010.

11. Doğan, E., Erdal, F., Çarbaş, S. ve Saka, M.P., Izgara sistemlerin LRFD-AISC şartnamesine göre parçacık küme yöntemi kullanılarak optimum boyutlandırılması, 17. Ulusal Mekanik Kongresi, Fırat Üniversitesi, 5-9 Eylül, 2011.

12. Doğan, E., Erdal, F. and Saka, M.P., The effect of member grouping on the optimum design of grillages via search techniques, SDU International Journal of Technologic Sciences, Vol. 5, No 2, 64-77, December 2013.

13. Dorigo, M., Maniezzo, V. and Colorni, A., Positive feedback as a search strategy, 91-016 Politecnico di Milano, 1991.

14. Erdal, F., and Saka, M.P., Optimum design of grillage systems using harmony search algorithm, Proceedings of 8th International Conference of Computational Structures Technology, Las Palmas De Gran Canaria, Spain, Sept. 12-15, 2006.

15. Erdal, F., Optimum design of grillage systems by using harmony search algorithm, A Thesis Submitted to The Graduate School Of Natural And Applied Sciences, METU, 2007.

16. Erdal, F., and Saka, M.P., Effect of beam spacing in the harmony search based optimum design of grillages, Asian Journal Of Civil Engineering (Building And Housing) Vol. 9, No. 3, 215-228, 2008.

17. Erdal, F., Doğan, E., and Saka, M.P., An improved particle swarm optimizer for steel grillage systems, Structural Engineering and Mechanics, Vol. 47, No.4, 513-530, 2013.

18. Farahani, S.M., Nasiri, B., Abshouri, A.A., and Meybodi, M.R., An improved firefly algorithm with directed movement, IEEE, 978-1-61284-836-5/11, 2011.

19. Gandomi, A.H., Yang, X.S., and Alavi, A.H., Mixed variable structural optimization using firefly algorithm, Computers and Structures, Vol. 89, Issues 23–24, 2325–2336, 2011.

20. Glover, F., Tabu search, Kluwer academic Publications, MA, USA, 1-54, 1997.

21. Goldberg, D.E., Genetic algorithms in search, Optimization and Machine Learning, Addison-Wesley Publishing, INC, 1989.

22. Haupt, R.L., and Haupt, S.E., Practical genetic algorithms, 2nd Edition, John Wiley and Sons, Inc, 272p, 2004.

23. He, S., Prempain, E., and Wu, Q.H., An improved particle swarm optimizer for mechanical design optimization problems, Engineering Optimization 36: 585-605, 2004.

24. Holland, J., Genetic algorithms and the optimal allocation of trials, Society for Industrial and Applied Mathematics Journal of Computing, 2: 88-105, 1973.

25. Karaboğa, D., Yapay zekâ optimizasyon algoritmaları, Atlas Yayın Dağıtım, İstanbul, 2004.

26. Karaboğa, N., A new design method based on artificial bee colony algorithm for digital IIR filters, Journal of Franklin Institute, 346(1): 328–348, 2009.

27. Karaboğa, D., Yapay zeka optimizasyon algoritmaları, Nobel Yayın Dağıtım, 2011.

28. Kassimali, A., Structural analysis, Thomson, Ontario, Canada, 89-160, 2005.

29. Kennedy, J., and Eberhart, R. C., Particle swarm optimization, IEEE International Conference on Neural Networks, Piscataway, NJ, 1995.

30. Krishnanand, K.N., and Ghose, D., Detection of multiple source locations using a glowworm metaphor with applications to collective robotics, IEEE Swarm Intelligence Symposium, 84-91, 2005.

31. Krishnanand, K.N., and Ghose, D., Glowworm swarm optimisation: A new method for

optimisin multi-modal functions, International Journal of Computational Intelligence Studies, Vol. 1, no. 1, 2009.

32. Kurt, M., Stochastic optimization methods, Springer, Munich, Germany, 3-43, 2005.

33. Kwang, Y.L., and El-Sharkawi, M.A., Modern heuristic optimization techniques: theory and applications to power systems, Institute of Electrical and Electronics Engineers Press, NJ, USA, 43-147, 2008.

34. Khadwilard, A., Chansombat, S., Thepphakorn, T., Thapatsuwan, P., Chainate, W., and Pongcharoen, P., Investigation of firefly algorithm parameter setting for solving job shop scheduling problems, In proceeding of: ORNET 2011.

35. Lee, K.S., and Geem, Z.W., A new structural optimization method based on the harmony search algorithm, Computers and Structures, 82, 781-798, 2004.

36. LRFD-AISC, Load and resistance factor design, American Institute of Steel Construction, IL, USA, 62-564, 1999.

37. Lukasik, S., and Zak, S., Firefly algorithm for continuous constrained optimization tasks, Computational Collective Intelligence, Semantic Web, Social Networks and Multiagent Systems, Lecture Notes in Computer Science Vol. 5796, 97-106, 2009.

38. Mafrouz, S.Y., Design optimization of steelwork, Ph.D. Dissertation, Department of Civil Engineering University of Bradford, 1999.

39. Oftadeh, R., Mahjoob, M.J., and Shariatpanahi, M., A novel meta-heuristic optimization algorithm inspired by group hunting of animals hunting search, Computers Mathematics with Applications, Vol.60, 2087-2098, 2010.

40. Ragsdell, K.M., and Philips D.T., Optimal design of a class of welded structures using geometric programming, American Society of Mechanical Engineering Journal of Engineering for Industries 98 : 1021-1025, 1976.

41. Ray, T., and Liew, K.M., Society and civilization: An optimization algorithm based on the simulation of social behavior, Institute of Electrical and Electronics Engineers International Transactions on Evolutionary Computation 7: 386-396, 2003.

42. Saka, M.P., Optimum design of steel grillage systems using genetic algorithms, Computer-Aided Civil and Infrastructure Engineering, 13, 233-238, 1998.

43. Saka, M.P., Daloglu, A. and Malhas, F., Optimum spacing design of grillage systems using a genetic algorithm, Advances in Engineering Software, 31, 863-873, 2000.

44. Saka, M.P. and Erdal, F., Effect of beam spacing in the harmony search based optimum design of grillages, Asian Journal of Civil Engineering (building and housing) Vol. 9, No. 3, 215-228, 2008.

45. Saka, M.P. and Erdal, F., Harmony search based algorithm for the optimum design of grillage systems to LRFD-AISC, Structural and Multidisciplinary Optimization, Vol. 38, No:1, 25-41, 2009.

46. Lukasik, S. and Zak, S., Firefly algorithm for continuous constrained optimization tasks , In Proceeding of Springer, 169-178, 2009.

47. Şeker, S., Farklı tiplerde kolon-kiriş bağlantı noktalarına sahip yarı-rijit çelik çerçevelerin stokastik arama yöntemleri kullanılarak optimum boyutlandırılması, Celal Bayar Üniversitesi Fen Bilimleri Enstitüsü , Doktora Tezi , 2013.

48. Tereshko, V., Reaction-diffusion model of a honey bee colony's foraging behaviour, 6th International Conference on Parallel Problem Solving from Nature, 807-816, 3-540-41056-2, Springer-Verlag, London, UK, 2000.

49. Yang, X.S., Firefly algorithms formultimodal optimization, Proceedings of the Stochastic Algorithms: Foundations and Applications, Lecture Notes in Computing Sciences, Vol. 5792, 178-178, Springer, Sapporo, Japan, 2009.

50. Yang, X.S., Firefly algorithms for multimodal optimization, in: Stochastic algorithms: foundations and applications, Lecture Notes in Computer Sciences, Vol. 5792, 169-178, 2009.

51. Yang, X.S., Firefly algorithm, Levy flights and global optimization, Research and Development in Intelligent Systems, XXVI, 209-218, Springer, London, UK, 2010.

52. Yang, X.S., Firefly algorithm, Stochastic Test Functions and Design Optimization, Int. J. Bio-Inspired Computation, Vol. 2, No: 2, 78–84, 2010.

53. Yang, X.S., and He, X.,Firefly algorithm: Recent advances and applications, International Journal of Swarm Intelligence, Inderscience Publishers, Artificial Intelligence and Intelligent Systems, Vol. 1, 36-50, Science, Engineering and Technology, 2013.

ÖZGEÇMİŞ

Kişisel Bilgiler:

Adı : Fatma Esra

Soyadı : UZ

Doğum Tarihi ve Yeri : 1988 - İZMİR

Email : esrauz@hotmail.com

Eğitim Bilgileri:

Ortaöğretim : Vali Vecdi Gönül Lisesi (YDA) - İzmir

Lisans : Celal Bayar Üniversitesi İnşaat Mühendisliği Bölümü