

**T.C.
BALIKESİR ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
İNŞAAT MÜHENDİSLİĞİ ANABİLİM DALI**



**DÜZLEM ÇELİK ÇERÇEVELERİN SOSYAL ÖRÜMCEK
OPTİMİZASYONUNA GÖRE BOYUTLANDIRILMASI**

YÜKSEK LİSANS TEZİ

METİN YETKİN

BALIKESİR, MAYIS - 2015

**T.C.
BALIKESİR ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
İNŞAAT MÜHENDİSLİĞİ ANABİLİM DALI**



**DÜZLEM ÇELİK ÇERÇEVELERİN SOSYAL ÖRÜMCEK
OPTİMİZASYONUNA GÖRE BOYUTLANDIRILMASI**

YÜKSEK LİSANS TEZİ

METİN YETKİN

BALIKESİR, MAYIS - 2015

KABUL VE ONAY SAYFASI

Metin YETKİN tarafından hazırlanan “**DÜZLEM ÇELİK ÇERÇEVELERİN SOSYAL ÖRÜMCEK OPTİMİZASYONUNA GÖRE BOYUTLANDIRILMASI**” adlı tez çalışmasının savunma sınavı 29.05.2015 tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği / oy çokluğu ile Balıkesir Üniversitesi Fen Bilimleri Enstitüsü İnşaat Mühendisliği Anabilim Dalı Yüksek Lisans Tezi olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Danışman
Yrd. Doç. Dr. Perihan EFE

Eş Danışman
Yrd. Doç. Dr. İbrahim AYDOĞDU

Üye
Doç. Dr. Nevzat KIRAÇ

Üye
Yrd. Doç. Dr. Selim ŞENGEL

Üye
Yrd. Doç. Dr. Erkan KARAMAN

.....
.....
.....
.....
.....

Jüri üyeleri tarafından kabul edilmiş olan bu tez BAÜ Fen Bilimleri Enstitüsü Yönetim Kurulunca onanmıştır.

Fen Bilimleri Enstitüsü Müdürü

Doç. Dr. Necati Özdemir

.....

ÖZET

**DÜZLEM ÇELİK ÇERÇEVELERİN SOSYAL ÖRÜMCEK
OPTİMİZASYONUNA GÖRE BOYUTLANDIRILMASI
YÜKSEK LİSANS TEZİ
METİN YETKİN
BALIKESİR ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ
İNŞAAT MÜHENDİSLİĞİ ANABİLİM DALI
(TEZ DANIŞMANI: YRD. DOÇ. DR. PERİHAN EFE)
(EŞ DANIŞMAN: YRD. DOÇ. DR. İBRAHİM AYDOĞDU)
BALIKESİR, MAYIS - 2015**

Son yıllarda doğa olaylarını taklit ederek geliştirilen algoritmalar araştırmacılar tarafından büyük ilgi görmüştür. Bu algoritmalar birçok mühendislik alanında kullanılmaktadır. Yapı mühendisliğinde ise özellikle çelik yapıların optimum tasarlanması konusunda birçok çalışma mevcuttur.

Bu çalışmada düzlem çelik çerçevelerin optimum tasarımını yapan sosyal örümcek optimizasyon (SSO) algoritma tabanlı bir yazılım geliştirilmiştir. SSO algoritması visual basic applications (VBA) program dilinde kodlanmış ve SAP2000 programıyla entegre edilmiştir. Düzlem çelik çerçeveler Amerikan Çelik Yapılar Enstitüsü - Yük ve Dayanım Faktörü Tasarımı (AISC - LRFD) şartnamesi kullanılarak boyutlandırılmıştır. Geliştirilen SSO algoritmasının etkinliğini test etmek için literatürden seçilen düzlem çelik çerçeveler kullanılmıştır. Sonuçlar literatürde kullanılan diğer optimizasyon yöntemleriyle karşılaştırılmıştır. Karşılaştırma sonuçlarından SSO algoritmasının güçlü bir algoritma olduğu görülmüştür.

ANAHTAR KELİMELELER: Düzlem çelik çerçeve, Optimum tasarım, Sosyal örümcek optimizasyonu, AISC-LRFD, SAP2000

ABSTRACT

**OPTIMUM DESIGN OF PLANE STEEL FRAME ACCORDING TO SOCIAL
SPIDER OPTIMIZATION
MSC THESIS
METİN YETKİN
BALIKESİR UNIVERSITY INSTITUTE OF SCIENCE
CIVIL ENGINEERING
(SUPERVISOR: ASSIST. PROF. DR. PERİHAN EFE)
(CO-SUPERVISOR: ASSIST. PROF. DR. İBRAHİM AYDOĞDU)
BALIKESİR, MAY 2015**

In recent years, algorithms developed by mimicking natural events have attracted great attention to researchers. These algorithms are used in many engineering fields. There are many studies in structural engineering, especially optimum design of steel structures.

In this study, social spider optimization (SSO) algorithm based computer program has been developed for optimum design of the plane steel frames. SSO algorithm based computer program is written by using visual basic applications (VBA) language and integrated with the SAP2000 program. Plane steel frames are designed according to the American Institute of Steel Construction - Load and Resistance Factor Design (AISC - LRFD) specification design code. In order to test the effectiveness of the developed algorithm, design examples are selected from the literature studies. Obtained results have been compared with the results of literature studies. It is clearly seen from these comparisons that SSO is a powerful algorithm and can be used in structural design problems.

KEY WORDS: Plane steel frame, Optimum design, Social spider optimization, AISC-LRFD, SAP2000

İÇİNDEKİLER

Sayfa

ÖZET	i
ABSTRACT	ii
ŞEKİL LİSTESİ	v
TABLO LİSTESİ	vi
SEMBOL LİSTESİ	vii
KISALTMALAR	x
ÖNSÖZ	xi
1. GİRİŞ	1
1.1 Bu Çalışmanın Amacı ve Kapsamı.....	1
1.2 Literatür Çalışmaları.....	2
2. İNŞAAT MÜHENDİSLİĞİNDE OPTİMİZASYON	7
2.1 Optimizasyon Kavramı.....	7
2.1.1 Amaç Fonksiyonu.....	7
2.1.2 Tasarım Değişkenleri.....	8
2.1.3 Kısıtlayıcılar.....	8
2.2 Evrimsel Optimizasyon Yöntemleri.....	9
2.2.1 Genetik Algoritma.....	9
2.2.2 Harmoni Arama.....	9
2.2.3 Karınca Kolonisi Optimizasyonu.....	10
2.2.4 Yapay Arı Kolonisi.....	10
3. SOSYAL ÖRÜMCEK OPTİMİZASYONU	12
3.1 Titreşim Operatörü.....	14
3.1.1 Dişi Örümcekler için Titreşim Modeli.....	15
3.1.1.1 Vibci Titreşim Modeli.....	15
3.1.1.2 Vibbi Titreşim Modeli.....	16
3.1.2 Erkek Örümcekler için Titreşim Modeli.....	16
3.1.2.1 Vibfi Titreşim Modeli.....	16
3.2 Hareket Operatörleri.....	16
3.2.1 Dişi Örümceklerin Hareket Operatörü.....	16
3.2.2 Erkek Örümceklerin Hareket Operatörü.....	17
3.3 Çiftleşme Operatörü.....	18

4. DÜZLEM ÇELİK ÇERÇEVELERİN OPTİMUM TASARIM PROBLEMİ	20
4.1 Tek Katlı Tek Açıklıklı Bir Çerçevenin SSO ile Boyutlandırılması.....	27
5. SAYISAL ÖRNEKLER	46
Örnek 1. İki Açıklıklı Altı Katlı Çerçeve.....	48
Örnek 2. İki Açıklıklı Üç Katlı Çerçeve	53
Örnek 3. Tek Açıklıklı On Katlı Çerçeve	56
Örnek 4. Üç Açıklıklı On beş Katlı Çerçeve.....	61
6. SONUÇ VE ÖNERİLER	66
KAYNAKLAR	68
EKLER	71

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1	: Betonarme giriş kesiti.....	8
Şekil 3.1	: SSO algoritmasının akış şeması.....	20
Şekil 4.1	: Çerçevenin boyutları, yüklemeler ve grup numaraları.....	27
Şekil 4.2	: f1 örümceği ile sembolize edilen çerçeve ve 3D görünüşü.....	31
Şekil 4.3	: f2 örümceği ile sembolize edilen çerçeve ve 3D görünüşü.....	32
Şekil 4.4	: f3 örümceği ile sembolize edilen çerçeve ve 3D görünüşü.....	32
Şekil 4.5	: m1 örümceği ile sembolize edilen çerçeve ve 3D görünüşü.....	33
Şekil 4.6	: f1 örümceği ile sembolize edilen çerçevenin yanal öteleme değeri..	34
Şekil 4.7	: f2 örümceği ile sembolize edilen çerçevenin yanal öteleme değeri..	35
Şekil 4.8	: f3 örümceği ile sembolize edilen çerçevenin yanal öteleme değeri..	36
Şekil 4.9	: m1 örümceği ile sembolize edilen çerçevenin yanal öteleme değeri	36
Şekil 4.10	: 1. Tasarım değişkeni için örümceklerin çiftleşme etki alanı.....	40
Şekil 4.11	: 2. Tasarım değişkeni için örümceklerin çiftleşme etki alanı.....	42
Şekil 4.12	: snow örümceği ile sembolize edilen çerçeve ve 3D görünüşü.....	43
Şekil 4.13	: snow örümceği ile sembolize edilen çerçevenin yanal öteleme değeri.....	44
Şekil 5.1	: İki açıklıklı altı katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları.....	49
Şekil 5.2	: İki açıklıklı altı katlı çerçevenin optimum kesitleri.....	51
Şekil 5.3	: İki açıklıklı altı katlı çerçevenin ağırlık iterasyon ilişkisi.....	52
Şekil 5.4	: İki açıklıklı altı katlı çerçevenin x doğrultusundaki yanal ötelenmeleri.....	52
Şekil 5.5	: İki açıklıklı üç katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları.....	54
Şekil 5.6	: İki açıklıklı üç katlı çerçevenin optimum kesitleri.....	55
Şekil 5.7	: İki açıklıklı üç katlı çerçevenin ağırlık iterasyon ilişkisi.....	55
Şekil 5.8	: Tek açıklıklı on katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları.....	57
Şekil 5.9	: Tek açıklıklı on katlı çerçevenin optimum kesitleri.....	59
Şekil 5.10	: Tek açıklıklı on katlı çerçevenin ağırlık iterasyon ilişkisi.....	60
Şekil 5.11	: Tek açıklıklı on katlı çerçevenin x doğrultusundaki yanal ötelenmeleri.....	60
Şekil 5.12	: Üç açıklıklı on beş katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları.....	62
Şekil 5.13	: Üç açıklıklı on beş katlı çerçevenin optimum kesitleri.....	64
Şekil 5.14	: Üç açıklıklı on beş katlı çerçevenin ağırlık iterasyon ilişkisi.....	65

TABLO LİSTESİ

Sayfa

Tablo 4.1: Etkili uzunluk faktörü (TS648, AISC-LRFD)	25
Tablo 4.2: Profil Havuzu	28
Tablo 4.3: Örumceklerin başlangıç kesitleri	31
Tablo 4.4: Örumceklerin yanal öteleme değerleri ve yanal öteleme üst sınırı.....	35
Tablo 4.5: Örumceklerin cezalandırılmış ağırlıkları	35
Tablo 4.6: Örumceklerin birbirlerine uzaklıkları	37
Tablo 4.7: Örumceklerin yeni kesitleri	39
Tablo 5.1: Tasarım örneklerinde kullanılan SSO parametreleri	47
Tablo 5.2: İki açıklıklı altı katlı çerçevenin optimum sonuçları	50
Tablo 5.3: İki açıklıklı üç katlı çerçevenin optimum sonuçları.....	54
Tablo 5.4: Tek açıklıklı on katlı çerçevenin optimum sonuçları.....	58
Tablo 5.5: Üç açıklıklı on beş katlı çerçevenin optimum sonuçları.....	63

SEMBOL LİSTESİ

A_{gk}	: k çubuğunun en kesit alanı
b	: Kiriş tabla genişliği
b_w	: Kiriş genişliği
B_{sb}	: Kiriş profil başlık genişliği
B_{sc}	: Kolon profil başlık genişliği
d_{ij}	: i ve j örümcekleri arasındaki mesafe
d_{max}	: İki örümcek arasında olabilecek en büyük mesafe
D_s	: Kolon profil yüksekliği
e	: Logaritmik değer
E	: Elastisite modülü
f_{ij}^0	: i. dişi örümceğin j. tasarım değişkeninin başlangıç konumu
f_{ij}^{k+1}	: i. dişi örümceğin j. tasarım değişkeninin (k+1). konumu
$F(\vec{x})$: \vec{x} değişkenine bağlı amaç fonksiyonu
F_y	: Çelik akma dayanımı
$g(x)$: x değişkenine bağlı kısıtlayıcı fonksiyon
G_A	: Kolonun üst ucunun rijitlik dağıtım faktörü
G_B	: Kolonun alt ucunun rijitlik dağıtım faktörü
h_j	: j. katın yüksekliği
I	: Yanal olarak tutulmamış boy uzunluğu
I_b	: Göz önüne alınan kolonun üst ucuna rijit bağlı kirişin atalet momenti
I_b'	: Göz önüne alınan kolonun alt ucuna rijit bağlı kirişin atalet momenti
I_c	: Göz önüne alınan kolonun üst ucuna rijit bağlı kolonun atalet momenti
I_c'	: Göz önüne alınan kolonun alt ucuna rijit bağlı kolonun atalet momenti
K	: Etkili uzunluk faktörü
l_s	: r grubuna ait s çubuğunun boyunun uzunluğu
L_b	: Göz önüne alınan kolonun üst ucuna rijit bağlı kirişin uzunluğu
L_b'	: Göz önüne alınan kolonun alt ucuna rijit bağlı kirişin uzunluğu

- L_c** : Göz önüne alınan kolonun üst ucuna rijit bağlı kolonun boyu
- L_c'** : Göz önüne alınan kolonun alt ucuna rijit bağlı kolonun boyu
- m** : Toplam kısıtlayıcı sayısı
- m_{kj}^0** : k. erkek örümceğin j. tasarım değişkeninin başlangıç konumu
- m_{ij}^{k+1}** : i. erkek örümceğin j. tasarım değişkeninin (k+1). konumu
- m_r** : r çubuk grubu için seçilen profilin birim boyunun ağırlığı
- m_s** : Kolon profil birim boy ağırlığı
- M_{nxx}** : Kuvvetli eksenindeki nominal eğilme dayanımı
- M_{uxk}** : İkinci mertebeden etkileri içeren kuvvetli eksenindeki eğilme dayanımı
- n** : Tasarım değişkeni sayısı
- ng** : Çerçeveadaki toplam grup sayısı
- ns** : Çerçeveadaki toplam kat sayısı
- N** : Örümcek sayısı
- N_f** : Dişi örümcek sayısı
- N_m** : Erkek örümcek sayısı
- PF** : Dişi örümceklerin ileriye veya geriye hareket etmesini sağlayan 0 ile 1 arasında bir sayı
- p_j^{low}** : j. tasarım değişkeninin en düşük değeri
- p_j^{high}** : j. tasarım değişkeninin en yüksek değeri
- P_{nk}** : Nominal kuvvet
- P_{uk}** : Eksenel kuvvet
- $P(x)$** : Ceza fonksiyonu
- r** : Atalet yarıçapı
- r** : Çiftleşme etki alanı
- $rand$** : 0 ile 1 arasında bir sayı
- s_b** : $Vibb_i$ titreşim modelindeki titreşimden etkilenen dişi örümceğin j. tasarım değişkeni
- s_c** : $Vibc_i$ titreşim modelindeki titreşimden etkilenen dişi örümceğin j. tasarım değişkeni
- s_f** : $Vibf_i$ titreşim modelindeki titreşimden etkilenen dişi örümceğin j. tasarım değişkeni
- s_{1j}** : Titreşimi oluşturan örümceğin j. tasarım değişkeni

- s_{2j} : Titreşimden etkilenen örümceğin j. tasarım değişkeni
 t : Kiriş tabla kalınlığı
 t_r : r grubundaki toplam çubuk sayısı
 Vib_{ij} : i ve j örümcekleri arasında oluşan titreşim şiddeti
 v_s : İhlal etme katsayısı
 w_j : j. örümceğin performans değeri
 W_j : j. örümceğin amaç fonksiyonu
 w_{N_f+i} : i. erkek örümceğin performans değeri
 w_{N_f+m} : Ortanca performanslı erkek örümcek
 \vec{x} : Tasarım değişkeni
 α, β, δ : 0 ile 1 arasında bir sayı
 δ_j : j. katın yanal öteleme değeri
 δ_{ju} : İzin verilen yanal öteleme üst sınırı
 δ_{iu} : Kirişlerin izin verilen deplasman üst sınırı
 \emptyset : Dayanım azaltma katsayısı
 \emptyset_b : Eğilme için dayanım azaltma katsayısı

KISALTMALAR

ABC	: Yapay Arı Kolonisi
ACO	: Karınca Kolonisi Optimizasyonu
AISC	: Amerikan Çelik Yapılar Enstitüsü
APSO	: Hızlandırılmış Parçacık Sürüsü Optimizasyonu
BB-BC	: Büyük Patlama Büyük Çatırtı
BS	: İngiliz Şartnamesi
DDHS	: Tasarım Odaklı Harmoni Arama
EHS	: Geliştirilmiş Harmoni Arama
GA	: Genetik Algoritma
HPSACO	: Heuristik Parçacık Sürüsü Karınca Kolonisi Optimizasyonu
HS	: Harmoni Arama
IACO	: Geliştirilmiş Karınca Kolonisi Optimizasyonu
LRFD	: Yük ve Dayanım Faktörü
MMDGA	: Değiştirilmiş Çoklu Deme Genetik Algoritma
PSO	: Parçacık Sürüsü Optimizasyonu
PSOPC	: Parçacık Sürüsü Optimizasyonu Pasif Toplanma
SSO	: Sosyal Örümcek Optimizasyonu
TLBO	: Öğretme Öğrenme Tabanlı Optimizasyon
TS648	: Çelik Yapıların Hesap ve Yapım Kuralları

ÖNSÖZ

Çalışmalarım boyunca bana her zaman destek olan, değerli fikirleriyle beni aydınlatan tez danışmanın Yrd. Doç. Dr. Perihan EFE' ye, hiçbir zaman yardımlarını esirgemeyen, tezin hazırlanması sırasında karşılaştığım sorunlarla özel olarak ilgilenen ve algoritmanın yazılmasında büyük emeği geçen eş danışmanım Yrd. Doç. Dr. İbrahim AYDOĞDU' ya teşekkürlerimi sunarım.

Ayrıca bu çalışmayı bugünlere gelmemde büyük emeği geçen aileme ithaf ediyorum.

Balıkesir, 2015

Metin YETKİN

1. GİRİŞ

Dünya nüfusunun hızla artması insanları kullanışlı yapılar tasarlamaya yönlendirmektedir. Kullanışlı yapılar inşaat mühendisleri tarafından ekonomik, estetik ve dayanıklı olarak tasarlanmaktadır. İnşaat mühendisleri yapıları dış yüklere karşı boyutlandırırken ilk denemelerinde tahmini boyutlarda kesitler seçer. Bu kesitler çoğu zaman ekonomik olmaz. Seçilen kesitler dış yüklere karşı yeterli değilse boyutlar artırılır. Seçilen kesitler dış yüklere karşı çok büyükse bu kez kesitler küçültülmelidir. İnşaat mühendisi en uygun kesitleri bulana dek bu işlemleri tekrarlar. Bu şekilde tekrarlanarak en uygun çözümün bulunmasına optimizasyon denir.

İnşaat mühendislerinin en büyük amacı yapıları güvenlik şartlarından ödün vermeksizin en ekonomik olacak şekilde boyutlandırmaktır. Bu amacı gerçekleştirmek için optimizasyon yöntemleri kullanılır. En çok kullanılan optimizasyon yöntemleri doğadaki olayları taklit ederek geliştirilen yöntemlerdir.

1.1 Bu Çalışmanın Amacı ve Kapsamı

Bu çalışmanın amacı düzlem çelik çerçevelerin minimum ağırlıklı boyutlandırılması için bir optimizasyon algoritması geliştirmektir. Bu amaçla yapı analizinde SAP2000 programı kullanılmış ve visual basic applications (VBA) dilinde kodlanan sosyal örümcek optimizasyon (SSO) algoritması geliştirilmiştir. Ağırlık optimizasyonu yapılan düzlem çelik çerçevelerin tasarım değişkenleri için hazır profiller kullanılmıştır. Boyutlandırma kısıtlayıcıları olarak mukavemet, deplasman ve yanal ötelenme değerleri kullanılmıştır. Mukavemet ve deplasman kısıtları için Amerikan Çelik Yapılar Enstitüsü - Yük ve Dayanım Faktörü Tasarımı (AISC - LRFD) şartnamesi kullanılmıştır. Yanal ötelenme kısıtları ise VBA' da hazırlanmış

ve katlar arası maksimum yanal ötelenme değeri kat yüksekliği / 300, çerçevenin en üst katının maksimum yanal ötelenme değeri ise çerçeve yüksekliği / 300 alınmıştır.

Geliştirilen SSO algoritmasının kullanılabilir olduğunu göstermek için literatürden dört örnek seçilmiştir. Sonuçlar SSO algoritmasının güçlü bir yöntem olduğunu göstermiştir.

Hazırlanan bu çalışma altı bölümden oluşmaktadır. Birinci bölüm giriş bölümü olup bu çalışma hakkında genel bilgiler içermektedir. İkinci bölümde inşaat mühendisliğinde optimizasyon konusuna değinilmiştir. Üçüncü bölümde sosyal örümcek optimizasyonu detaylı bir şekilde açıklanmıştır. Dördüncü bölümde düzlem çelik çerçevelerin optimum tasarım problemi hakkında detaylar yer almaktadır. Beşinci bölümde sayısal örneklere yer verilmiştir. Altıncı bölüm ise sonuçların ve önerilerin verildiği bölümdür.

1.2 Literatür Çalışmaları

S. Pezeshk, C. V. Camp ve D. Chen 2000 yılında “Design of nonlinear framed structured using genetic optimization” isimli bir çalışma yapmışlardır [1]. Bu çalışmalarında 2 boyutlu doğrusal olmayan çelik çerçevelerin optimum tasarımı için genetik algoritma geliştirmişlerdir. Çerçevelerin tasarımında AISC – LRFD şartnamesi kullanılmıştır. Optimum tasarım için doğrusal analiz ve doğrusal olmayan analiz yöntemlerini kullanmışlardır. Analiz yöntemlerini karşılaştırmak için iki örnek kullanılmıştır.

E.S. Kameshki ve M.P. Saka 2001 yılında “Genetic algorithm based optimum bracing design of non-swaying tall plane frames” isimli bir çalışma yapmışlardır [2]. Bu çalışmada çok katlı yapıların optimum tasarımı için genetik algoritma yöntemini sunmuşlardır. Taşıyıcı elemanları boyutlandırmak için standart çelik profiller kullanılmıştır. Mukavemet ve deplasman sınırları İngiliz Şartnamesinden (BS 5950, 1990) alınmıştır.

C. V. Camp, B. J. Bichon ve S. P. Stovall 2005 yılında ‘‘Design of steel frames using ant colony optimization’’ isimli bir alıřma yapmıřlardır [3]. Bu alıřmada elik yapıların ayrıık optimizasyonu iin karınca kolonisi optimizasyonunu geliřtirmiřlerdir. elik yapıların minimum ağırlıklı boyutlandırılmasında mukavemet ve deplasman kısıtlayıcıları iin AISC – LRFD,2001 řartnamesi kullanılmıřtır. Literatürden setikleri ereveleri karınca kolonisi optimizasyonunu kullanarak optimize etmiřler ve sonuları genetik algoritma ve diđer optimizasyon yntemleriyle karřılařtırmıřlardır.

S. O. Degertekin, M.P. Saka ve M.S. Hayalioglu 2008 yılında ‘‘Optimal load and resistance factor design of geometrically nonlinear steel space frames via tabu search and genetic algorithm’’ isimli bir alıřma yapmıřlardır [4]. Bu alıřmalarında tabu arama ve genetik algoritma yntemlerini kullanarak dođrusal olmayan elik uzay ereveleri optimize etmiřlerdir. ereve ağırlıđını minimize etmek iin kullanılan kesitler Amerikan elik yapılar enstitüsünde (AISC) yer alan standart W kesitleri listesinden seilmiřtir. Hazırladıkları algoritmaları üç örnekte kullanmıřlardır. Tabu arama genetik algoritmaya göre daha iyi sonular vermiřtir.

S. O. Degertekin 2008 yılında ‘‘Optimum design of steel frames using harmony search algoritm’’ isimli bir alıřma yapmıřtır [5]. Degertekin bu alıřmasında elik erevelerin optimum tasarımı iin harmoni arama algoritmasını geliřtirmiřtir. Optimum tasarımda kullanılmak üzere elik profiller Amerikan elik Yapılar Yk Enstitüsünde (AISC) yer alan standart geniř bařlıklı (W) kesitlerden seilmiřtir. Optimum tasarım iin mukavemet kısıtlayıcıları AISC – LRFD řartnamesinden alınmıř ayrıca deplasman kısıtlayıcıları da kullanılmıřtır. Harmoni arama algoritmasının kullanılabilir olduđunu gsterrmek iin üç tane elik ereve kullanılmıř ve sonular genetik algoritma ve karınca kolonisi optimizasyon algoritması ile karřılařtırılmıřtır. Harmoni arama algoritması diđer yntemlere göre daha iyi sonular vermiřtir.

M.P. Saka 2009 yılında ‘‘Optimum design of steel sway frames to BS5950 using harmony search algorithm’’ isimli bir alıřma yapmıřtır [6]. M. P. Saka bu alıřmasında elik ereveleri İngiliz řartnamesini (BS 5950) kullanarak harmoni

arama yöntemiyle boyutlandırmıştır. Kirişler İngiliz yönetmeliğindeki 64 adet kesit listesinden kolonlar da 32 adet kesit listesinden seçilmiştir. Algoritmanın etkinliğini göstermek için genetik algoritma ile karşılaştırma yapılmıştır. Sonuçlar harmoni arama yönteminin daha iyi olduğunu ortaya koymuştur.

A. Kaveh ve S. Talatahari 2010 yılında “An improved ant colony optimization for the design of planar steel frames” isimli bir çalışma yapmışlardır [7]. Bu çalışmalarında düzlem çelik çerçevelerin tasarımı için geliştirilmiş karınca kolonisi optimizasyonu (IACO) sunmuşlardır. Algoritmanın etkinliğini görmek için daha önce standart karınca kolonisi optimizasyonu, genetik algoritma ve harmoni arama algoritmalarıyla optimizasyonu yapılan çerçeveler kullanılmıştır. Geliştirilmiş karınca kolonisi optimizasyonu hesaplama süresini önemli ölçüde azaltmış ve oldukça iyi sonuçlar vermiştir.

Vedat Toğan 2012 yılında “Design of planar steel frames using Teaching – Learning Based Optimization” isimli bir çalışma yapmıştır [8]. V. Toğan bu çalışmasında düzlemsel çelik çerçevelerin ayırık optimizasyonu için öğretim öğrenme tabanlı optimizasyon (TLBO) tekniğini kullanmıştır. Literatürden seçtiği örnekleri AISC-LRFD tasarım şartnamesine göre optimize etmiştir. Sonuçlar TLBO algoritmasının mühendislik problemleri için güçlü bir yöntem olduğunu ortaya koymuştur.

S.O. Değertekin 2012 yılında “Optimum design of geometrically non-linear steel frames using artificial bee colony algorithm” isimli bir çalışma yapmıştır [9]. Değertekin bu çalışmasında doğrusal olmayan çelik çerçevelerin tasarımı için yapay arı koloni (ABC) algoritması geliştirmiştir. Minimum ağırlığı elde etmek için mukavemet, yer değiştirme ve boyut sınırlayıcıları kullanılmıştır. Algoritmanın performansını değerlendirmek için literatürden seçilen üç örnek kullanılmıştır. ABC algoritması kısa sürede diğer optimizasyon yöntemlerinden daha iyi tasarımlar bulmuştur.

E. Dođan ve M.P. Saka 2012 yılında “Optimum design of unbraced steel frames to LRFD – AISC using particle swarm optimization” isimli bir alıřma yapmıřlardır [10]. Hazırladıkları bu alıřmada paracık sürs optimizasyonunu (PSO) kullanarak elik ereveleri AISC-LRFD' ye gre boyutlandırmıřlardır. Geliřtirdikleri algoritmayı literatrden setikleri ereveler zerinde uygulamıřlardır. PSO diđer optimizasyon yntemlerinden daha iyi sonular vermiřtir.

S. Talatahari, E. Khalili, ve S. M. Alavizadeh 2013 yılında “Accelerated particle swarm for optimum design of frame structures” isimli bir alıřma yapmıřlardır [11]. Bu alıřmalarında ereve yapıların optimum tasarımı iin hızlandırılmıř paracık sürs optimizasyonu (APSO) sunmuřlardır. APSO standart paracık sürs optimizasyonuna gre daha iyi sonular vermiřtir. APSO algoritmasının performansını grmek iin iki tane yapı erevesi rneđi kullanmıřlardır.

D. Safari, M. R. Maheri, ve A. Maheri 2013 yılında “On the performance of a modified multiple-deme genetic algorithm in LRFD design of steel frames” isimli bir alıřma yapmıřlardır [12]. Bu alıřmalarında deđiřtirilmıř oklu deme genetik algoritma (MMDGA) ile dzlemsel elik erevelerin AISC-LRFD řartnamesine gre optimum tasarımını gerekleřtirmiřlerdir. ereve ađırlıđını minimize etmek iin mukavemet, yer deđiřtirme ve inřaat ařamasında uygulanabilir olması aısından boyut kısıtlayıcıları kullanılmıřtır. Algoritma  rnek zerinde denenmiř ve diđer metaheuristik yntemlerle karřılařtırılmıřtır. Sonular algoritmanın gl olduđunu gstermiřtir.

Mahmoud R. Maheri ve M.M. Narimani 2014 yılında “An enhanced harmony search algorithm for optimum design of side sway steel frames” isimli bir alıřma yapmıřlardır [13]. Bu alıřmalarında geliřtirilmıř harmoni arama (EHS) algoritmasını kullanarak literatrden setikleri elik ereveleri optimize etmiřlerdir. Bu alıřma EHS algoritmasının klasik harmoni arama ve diđer metaheuristik optimizasyon yntemleriyle rekabet edebileceđini gstermiřtir.

Patrick Murren ve Kapil Khandelwal 2014 yılında “Design-driven harmony search (DDHS) in steel frame optimization” isimli bir çalışma yapmışlardır [14]. Bu çalışmalarında tasarım odaklı harmoni arama (DDHS) algoritmasını geliştirmişlerdir. Üç örnek üzerinde uyguladıkları algoritmanın verimli olduğu görülmüştür.

Özer Sevim ve Mustafa Sönmez 2014 yılında “Geliştirilmiş yapay arı koloni algoritması ile kafes ve düzlemsel çelik yapıların optimum tasarımı” isimli bir çalışma yapmışlardır [15]. Bu çalışmalarında kafes ve düzlemsel çelik yapıların ayırık değişkenli optimizasyonu için yapay arı koloni (ABC) algoritması hazırlamışlardır. Algoritma literatürden seçilen örnekler üzerinde kullanılmış ve sonuçlar karşılaştırılmıştır. Sonuçlar ABC algoritmasının kullanılabilir olduğunu göstermektedir

2. İNŞAAT MÜHENDİSLİĞİNDE OPTİMİZASYON

2.1 Optimizasyon Kavramı

Belirli amaçlar doğrultusunda ve koşullar altında tanımlanan problemin çözümleri arasından en uygun olanı bulma işine optimizasyon denir. En uygun çözüm minimum maliyet, minimum ağırlık ya da en kârlı yatırım olabilir. En uygun çözüm bulunurken bazı şartlar göz önünde bulundurulmalıdır. Örneğin çelik bir yapı minimum maliyetli olacak şekilde hazır profiller ile boyutlandırılmak istenirse, inşaat mühendisi bu yapıyı boyutlandırırken maliyeti göz önüne alırken aynı zamanda yapının şartnamelere uygun olmasını da sağlamak zorundadır.

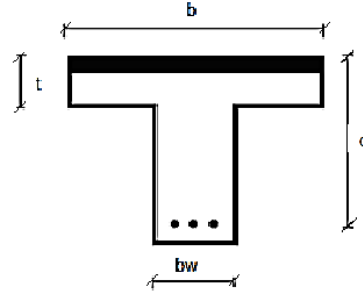
Optimizasyonun temel unsurları; amaç fonksiyonu, tasarım değişkenleri ve kısıtlayıcılarıdır.

2.1.1 Amaç Fonksiyonu

Bir optimizasyon probleminde tanımlanan amacın matematiksel fonksiyona dönüştürülmüş halidir. Tasarım probleminin optimize edilebilmesi için mutlaka amaç fonksiyonunun matematiksel ifadeye dönüştürülmesi gerekmektedir. Amaç fonksiyonu $F(\vec{x})$ şeklinde gösterilebilir. Amaca bağlı olarak amaç fonksiyonu da çeşitlilikler gösterebilir. Ağırlık, maliyet, verimlilik, performans en sık kullanılan amaç fonksiyonu çeşitleridir.

2.1.2 Tasarım Değişkenleri

Optimizasyon probleminin değerini değiştiren unsur değişkendir. $F(\vec{x})$ amaç fonksiyonundaki \vec{x} ifadesi tasarım değişkenlerinin vektörel ifadesidir. Betonarme kiriş tasarımı probleminde kiriş tablasının boyutları (b , t , b_w), kullanılacak betonun cinsi (C20, C25, C30), kullanılacak olan donatı sayısı ve çapları tasarım değişkeni olarak tanımlanabilir.



Şekil 2.1: Betonarme kiriş kesiti

2.1.3 Kısıtlayıcılar

Optimizasyon probleminde amaç fonksiyonlarının çözüm kümesi belirli şekilde kısıtlanabilir. Örneğin, çelik bir yapı mevcut şartnameleri sağlamak zorundadır. Bu gibi mevcut çözüm kümesini kısıtlayan fonksiyonlara kısıtlayıcı fonksiyonlar adı verilir. Kısıtlayıcı fonksiyonlar $g_1(x) = 0$, $g_2(x) < 0$, $g_3(x) > 0$ şeklinde ifade edilebilir. Örneğin çelik yapı tasarımında çelik profiller bileşik eğilme basınç altındaki taşıma kapasitesi uygulanan yüklerden büyük olması gerekmektedir.

$$\frac{P_{uk}}{\phi P_{nk}} \geq 0,2 \text{ ise } g_k(x) = \frac{P_{uk}}{\phi P_{nk}} + \frac{8}{9} \left(\frac{M_{uxk}}{\phi_b M_{nxk}} + \frac{M_{uyk}}{\phi_b M_{nyk}} \right) - 1 \leq 0 \quad (2.1)$$

$$\frac{P_{uk}}{\phi P_{nk}} < 0,2 \text{ ise } g_k(x) = \frac{P_{uk}}{2\phi P_{nk}} + \frac{8}{9} \left(\frac{M_{uxk}}{\phi_b M_{nxk}} + \frac{M_{uyk}}{\phi_b M_{nyk}} \right) - 1 \leq 0 \quad (2.2)$$

2.2 Evrimsel Optimizasyon Yöntemleri

Evrimsel optimizasyon yöntemleri doğada gerçekleşen olayları taklit ederek geliştirilmiştir. Doğada gerçekleşen bu olaylar bilgisayarda kodlanır ve problemin çözümüne ulaşılmaya çalışılır. Örneğin bu çalışmada örümceklerin doğadaki davranışlarından esinlenerek sosyal örümcek optimizasyonu adında bir algoritma geliştirilmiştir. Bu bölümde son yıllarda çok sık kullanılan optimizasyon yöntemlerinden bazıları verilmiş ve kısaca açıklanmıştır. Sosyal örümcek optimizasyonu ise bir sonraki bölümde detaylı olarak anlatılacaktır.

2.2.1 Genetik Algoritma

Genetik algoritma (GA), bireylerin doğa şartlarına uyum sağlayabilmesine dayanan bir yöntemdir. GA' ya göre doğa şartlarına uyum sağlayabilen canlı hayatta kalır. GA bireyleri bir araya getirip topluluk oluşturmak için şifreleme teknikleri kullanır. İki tabanına dayanan şifreleme tekniği en çok tercih edilen şifreleme tekniğidir. İki tabanına göre sayıların yan yana yazılmasıyla yeni bireyler, yeni bireylerin de bir araya gelmesiyle bir topluluk oluşur. Topluluktaki her bir birey kromozom ile ifade edilir. GA topluluktaki bireylerin kromozomlarını değiştirip yeni yavrular üretir [16]. Bu döngü belirli sayıda tekrarlanır. Topluluk tekrarlanan her döngünün ardından güncellenir. Son güncellemenin sonunda topluluktaki en iyi birey optimum sonuç olur. GA bu işlemleri üreme, çaprazlama ve mutasyon adı verilen evrimsel operatörleri kullanarak gerçekleştirir [17].

2.2.2 Harmoni Arama

Harmoni arama (HS), iyi bir ses uyumu bulmak için izlenen yolu taklit eden bir yöntemdir. İyi bir ses uyumu bulmak için notalar kullanılır. Do, re, mi, fa, sol, la, si müzisyenlerin kullandığı nota isimleridir. Müzisyenler bu notaları kullanarak birbiriyle uyumlu sesler elde etmeye çalışırlar. Örneğin herhangi bir müzik grubunda

gitarist, piyanist ve baterist olduğunu düşünelim. Piyanist {do, mi, sol} hafızasından “sol” , baterist {si, sol, re} hafızasından “si” , gitarist ise {la, fa, do} hafızasından “do” seçmiş olsun. Böylece üç müzisyen {sol, si, do} şeklinde harmoni oluşturur [18]. İyi bir ses uyumu elde edildiğinde harmoni arama bunları hafızaya alır. Bu işlemler harmoni arama hafızası dolduruluncaya kadar uygulanır. Daha iyi bir ses uyumu elde edilince hafızadaki en kötü ses uyumu çıkarılarak yerine yenisi dahil edilir. Bu işlemler önceden belirlenen sayıda tekrarlanarak hafıza güncellenir. Son güncellemenin sonunda hafızadaki en iyi ses uyumu optimum sonuç olur

2.2.3 Karınca Kolonisi Optimizasyonu

Karınca kolonisi optimizasyonu (ACO), karıncaların yuvaları ile yiyecek arasındaki en kısa yolu kullanmalarını taklit eden bir yöntemdir [16], [19]. Karıncalar göremedikleri halde feromon adı verilen kimyasal bir salgı maddesi sayesinde en kısa yolu kullanarak yiyeceğe ulaşırlar. Karıncalar bu işlemi koloni halinde yapar. Kolonideki karıncalardan herhangi biri ilk kez yiyecek aramaya çıktığında izlediği yol üzerine feromon salgılar. Kolonideki diğer karıncalar da bu salgıyı algıladıklarında bu yolu kullanmaya başlar [16]. Karıncaların salgıladığı feromon miktarı arttıkça kolonideki tüm karıncalar bu yolu kullanmaya başlar. Böylece karıncalar yuvaları ile yiyecek arasındaki en kısa yolu kullanır. Karınca kolonisi optimizasyonu bu şekilde geliştirilmiştir.

2.2.4 Yapay Arı Kolonisi

Yapay arı kolonisi (ABC) algoritması, koloni halinde yaşayan arıların doğadaki davranışlarını taklit ederek geliştirilmiş bir yöntemdir. Bir arı kolonisinde bir tane kraliçe arı, yüzlerce erkek arı ve binlerce dişi işçi arı bulunur [15], [20]. Buradaki arılardan her birinin farklı görevleri vardır. Erkek arıların görevi kraliçe arı ile çiftleşmektir. Kraliçe arıların görevi erkek arılar ile çiftleşerek yeni yavrular üretmektir. Dişi işçi arıların görevi ise arı kovanının temel ihtiyaçlarını karşılamaktır.

Temel ihtiyaçlardan en önemlisi diři işçi arıların yiyecek bulmasıdır. İşçi arılardan biri yeni bir yiyecek kaynağı bulduğunda bunu sallanma dansı ile diğeri işçi arılara bildirir.

3. SOSYAL ÖRÜMCEK OPTİMİZASYONU

Sosyal Örümcek Optimizasyonu (SSO) doğal hayattaki örümceklerin davranışlarını taklit ederek geliştirilen bir optimizasyon yöntemidir. Ortak örümcek ağını kullanan dişi ve erkek örümcekler bir araya gelerek koloni oluştururlar. Her bir örümcek sosyal örümcek olarak sembolize edilir ve problemin çözümüne göre bir performans değeri alır. Kolonide cinsiyete bağlı olarak her bir örümceğin farklı evrimsel operatörleri bulunur. Örümcekler, titreşim, hareket ve çiftleşme operatörlerini kullanır. Sosyal örümceklerin bu operatörleri uygulayabilmesi için etkileşim alanı olarak ortak bir örümcek ağını kullanırlar. Bu şekilde örümcekler bir araya gelerek bir koloni oluştururlar. Bir araya gelen örümcekler birbirleriyle iletişim kurmak için titreşim oluşturur. Titreşimin şiddeti örümceklerin performansına ve birbirlerine olan uzaklığa bağlıdır. Titreşimin etkisiyle örümcekler birbirlerine doğru yaklaşırlar ya da birbirlerinden uzaklaşırlar. Bu işlem örümceklerin hareket operatörünü ifade eder. Örümcekler hareket ettikten sonra yeni pozisyonlarında bulunurlar. Bu durumda baskın bir erkek örümceğin çiftleşme etki alanında bir dişi örümcek yer alırsa çiftleşme operatörü uygulanır. Çiftleşme sonucunda yeni bir örümcek doğar. Yeni doğan örümceğin koloniye alıp alınmaması performansına bağlıdır. Eğer yeni doğan örümceğin performansı başlangıçtaki kolonide yer alan en kötü performanslı örümcekten daha iyiyse koloniye dahil edilir. En kötü performanslı örümcek ise koloniden çıkarılır. Bu şekilde koloni güncellenir. Yeni doğan örümceğin cinsiyeti koloniden çıkarılan örümceğin cinsiyeti ile aynı kabul edilir. Böylece başlangıçtaki dişi ve erkek örümcek sayısı korunmuş olur. Yeni koloni elde etme işlemi önceden belirlenmiş sayıda tekrarlanır. En son güncellenen kolonideki performansı en yüksek olan örümcek optimizasyon probleminin optimum sonucu kabul edilir [21], [22].

Sosyal örümcek optimizasyon algoritması örümcek sayısını belirleyerek başlar. Sosyal örümcek optimizasyonuna göre dişi örümcek sayısı koloninin % 65 ile

% 90' ı arasındadır [21], [22]. Örümcek sayısı belirlendikten sonra dişi ve erkek örümcek sayıları aşağıdaki gibi hesaplanır;

$$N_f = N \cdot (0,90 - \text{rand} \cdot 0,25) \quad (3.1)$$

$$N_m = N - N_f \quad (3.2)$$

Burada, N kolonideki toplam örümcek sayısını, N_f dişi örümcek sayısını, N_m erkek örümcek sayısını, rand değeri ise 0 ile 1 arasında rastgele seçilen bir sayıyı göstermektedir.

Algoritma başladıktan sonra örümceklerin başlangıç pozisyonları belirlenmelidir,

$$f_{ij}^0 = p_j^{\text{low}} + \text{rand} \cdot (p_j^{\text{high}} - p_j^{\text{low}}) \quad (3.3)$$

$$i = 1,2,\dots,N_f \quad j = 1,2, \dots,n$$

Burada, i indisi i. dişi örümceği, j indisi i. dişi örümceğin tasarım değişkenini, f_{ij}^0 , i. dişi örümceğin başlangıç pozisyonunu, p_j^{low} j. tasarım değişkeninin en düşük değerini, p_j^{high} j. tasarım değişkeninin en yüksek değerini göstermektedir.

$$m_{kj}^0 = p_j^{\text{low}} + \text{rand} \cdot (p_j^{\text{high}} - p_j^{\text{low}}) \quad (3.4)$$

$$k = 1,2, \dots,N_m \quad j = 1,2, \dots,n$$

Burada, k indisi k. erkek örümceği, j indisi k. erkek örümceğin tasarım değişkenini, m_{kj}^0 , k. erkek örümceğin başlangıç pozisyonunu göstermektedir.

Sosyal örümcek optimizasyonu algoritması tüm işlemleri üç esas operatör kullanarak yapar.

3.1 Titreşim Operatörü

Koloni üyeleri birbirlerine bilgi iletmek amacıyla titreşim gönderirler. Titreşimler örümceklerin performansına ve birbirleri arasındaki uzaklığa bağlıdır. Birbirine yakın olan örümcekler uzaktaki örümceklere göre daha büyük titreşim oluşturur. Titreşim sonucunda örümcekler birbirlerini tahrik ederler [21], [22]. Bir örümcek bir öncekinden farklı bir pozisyona ulaştığında yeni bir titreşim oluşturacaktır [23].

Sosyal örümcek optimizasyonunda titreşimin şiddeti genel olarak şu şekilde hesaplanır.

$$Vib_{ij}=w_j \cdot e^{-d_{ij}^2} \quad (3.5)$$

Burada Vib_{ij} i ve j örümcekleri arasındaki titreşimin şiddetini, i titreşimi oluşturan örümceği, j titreşimden etkilenen örümceği, w_j j numaralı örümceğin performansını, d_{ij} i ve j örümcekleri arasındaki mesafeyi göstermektedir. d_{ij} aşağıdaki gibi hesaplanır;

$$d_{ij}=\sqrt{\sum_{j=1}^n (s_{1j} - s_{2j})^2} / d_{max} \quad (3.6)$$

Burada s_{1j} titreşimi oluşturan örümceğin j. tasarım değişkenini, s_{2j} titreşimden etkilenen örümceğin j. tasarım değişkenini, n tasarım değişkeni sayısını ifade etmektedir. d_{max} ise iki örümcek arasında olabilecek en büyük mesafe olup aşağıdaki gibi hesaplanmaktadır;

$$d_{max} = \sqrt{\sum_{j=1}^n (p_j^{high} - p_j^{low})^2} \quad (3.7)$$

Burada p_j^{high} j. tasarım deęişkeninin en yüksek deęeri, p_j^{low} ise j. tasarım deęişkeninin en düşük deęeridir.

Performans deęeri örümceęin optimizasyon probleminin çözümü olabilme potansiyelini gösterir ve řu řekilde hesaplanır;

$$w_j = \frac{1}{W_j} \quad (3.8)$$

Burada w_j j. örümceęin performans deęeri, W_j ise j. örümceęin amaç fonksiyonudur.

Sosyal örümcek optimizasyonu dişiler ve erkekler için farklı titreşim modelleri belirlemiştir.

3.1.1 Diři Örümcekler için Titreşim Modeli

3.1.1.1 Vibc_i Titreşim Modeli

Bu modellemede performansı en iyi olan diři örümcek dikkate alınmadan, diđer diři örümceklerin birbirlerine titreşim gönderdikleri kabul edilir. Burada her bir diři örümcek ile kendisine en yakın pozisyondaki başka bir diři örümcek arasında titreşim şiddeti hesaplanır. Bu titreşim modelinde diđerlerinden farklı olarak titreşim şiddeti hesaplanırken performansı daha iyi olan örümceęin performans deęeri kullanılır [21], [22].

$$Vibc_i = w_c \cdot e^{-d^2_{ic}} \quad (3.9)$$

3.1.1.2 Vibb_i Titreşim Modeli

Bu modellemede kolonideki performansı en iyi olan dişi örümcek (best female spider) ile diğer dişi örümcekler arasında titreşim şiddeti hesaplanır.

$$\text{Vibb}_i = w_b \cdot e^{-d^2_{ib}} \quad (3.10)$$

3.1.2 Erkek Örümcekler için Titreşim Modeli

3.1.2.1 Vibf_i Titreşim Modeli

Bu modellemede kolonide yer alan her bir erkek örümcek ile kendisine en yakın dişi örümcek arasında titreşim şiddeti hesaplanır.

$$\text{Vibf}_i = w_f \cdot e^{-d^2_{if}} \quad (3.11)$$

3.2 Hareket Operatörleri

3.2.1 Dişi Örümceklerin Hareket Operatörü

Dişi örümcekler belli bir cazibeye sahiptir ve cinsiyeti ne olursa olsun diğer örümcekler üzerinde bir etki bırakır. Dişi örümceklerin iş birliği davranışını taklit etmek amacıyla hareket operatörü tanımlanmıştır. Operatör her dişi örümceğin konumunu dikkate alarak her i . örümcekte tekrarlanır. Dişi örümcekler yaydıkları titreşime bağlı olarak diğer örümcekleri kendisine doğru çeker ya da kendisinden uzaklaştırır. Örümceklerin birbirlerine yaklaşmasını ya da birbirlerinden uzaklaşmasını belirlemek için PF adında bir parametre kullanılır. Örümcekler bu parametreye bağlı olarak hareket ederler. PF parametresi 0 ile 1 arasında bir değer olup problemin çözümüne geçmeden önce belirlenmelidir [21].

$$f_{ij}^{k+1} = \begin{cases} f_{ij}^k + \alpha \cdot \text{Vib}c_i \cdot (s_c - f_{ij}^k) + \beta \cdot \text{Vib}b_i \cdot (s_b - f_{ij}^k) + \delta \cdot \left(\text{rand} - \frac{1}{2}\right) & 0 < \text{rand} \leq \text{PF} \\ f_{ij}^k - \alpha \cdot \text{Vib}c_i \cdot (s_c - f_{ij}^k) - \beta \cdot \text{Vib}b_i \cdot (s_b - f_{ij}^k) + \delta \cdot \left(\text{rand} - \frac{1}{2}\right) & \text{PF} < \text{rand} < 1 \end{cases} \quad (3.12)$$

Burada, f_{ij}^{k+1} i. dişi örümceğin j. tasarım değişkeninin (k+1). konumunu, α , β , δ ve rand değerleri 0 ile 1 arasında rastgele seçilen bir sayıyı, s_c , $\text{Vib}c_i$ titreşim modelinde titreşimden etkilenen dişi örümceğin j. tasarım değişkenini, s_b ise $\text{Vib}b_i$ titreşim modelinde titreşimden etkilenen dişi örümceğin j. tasarım değişkenini göstermektedir.

3.2.2 Erkek Örümceklerin Hareket Operatörü

Sosyal örümcek optimizasyonunda erkek örümcekler, baskın ve baskın olmayan örümcekler olmak üzere ikiye ayrılır. Örümceklerin baskın olup olmaması performanslarına göre belirlenmektedir. Kolonide yer alan erkek örümcekler performanslarına göre sıralanır. Performansı ortanca performanslı erkek örümcektekenden fazla olanlar baskın, diğerleri ise baskın olmayan erkek örümcek sınıfına girer.

$$m_{ij}^{k+1} = \begin{cases} m_{ij}^k + \alpha \cdot \text{Vib}f_i \cdot (s_f - m_{ij}^k) + \delta \cdot \left(\text{rand} - \frac{1}{2}\right) & w_{N_f+i} > w_{N_f+m} \\ m_{ij}^k + \alpha \cdot \left(\frac{\sum_{h=1}^{N_m} m_h^k \cdot w_{N_f+h}}{\sum_{h=1}^{N_m} w_{N_f+h}} - m_{ij}^k \right) & w_{N_f+i} \leq w_{N_f+m} \end{cases} \quad (3.13)$$

Burada, m_{ij}^{k+1} i. erkek örümceğin j. tasarım değişkeninin (k+1). konumunu, α , δ ve rand değerleri 0 ile 1 arasında rastgele seçilen bir sayıyı, s_f , $\text{Vib}f_i$ titreşim modelinde titreşimden etkilenen dişi örümceğin j. tasarım değişkenini, w_{N_f+i} i. erkek örümceğin performansını, w_{N_f+m} ortanca performanslı erkek örümceği ifade etmektedir.

3.3 Çiftleşme Operatörü

Sosyal örümcek kolonisinde çiftleşmeler baskın erkek örümcekler ile dişi örümcekler arasında gerçekleşir. Herhangi bir baskın erkek örümceğin etki alanında en az bir tane dişi örümcek varsa çiftleşme operatörü uygulanır. Çiftleşme operatörü sonunda yeni bir örümcek doğar. Yeni doğan örümceğin performans değeri hesaplanır. Başlangıçtaki kolonide yer alan en kötü performanslı örümcek ile karşılaştırılır. Eğer yeni doğan örümceğin performansı daha iyiye en kötü performanslı örümcek koloniden çıkarılır ve yerine yeni doğan örümcek dahil edilir. Yeni doğan örümceğin cinsiyeti koloniden çıkarılan örümceğin cinsiyeti ile aynı kabul edilir. Böylece başlangıçtaki erkek ve dişi örümcek sayısı korunur.

Baskın erkek örümceğin çiftleşme etki alanında birden fazla dişi örümceğin bulunması durumunda çiftleşme operatörü rulet metoduna göre uygulanır.

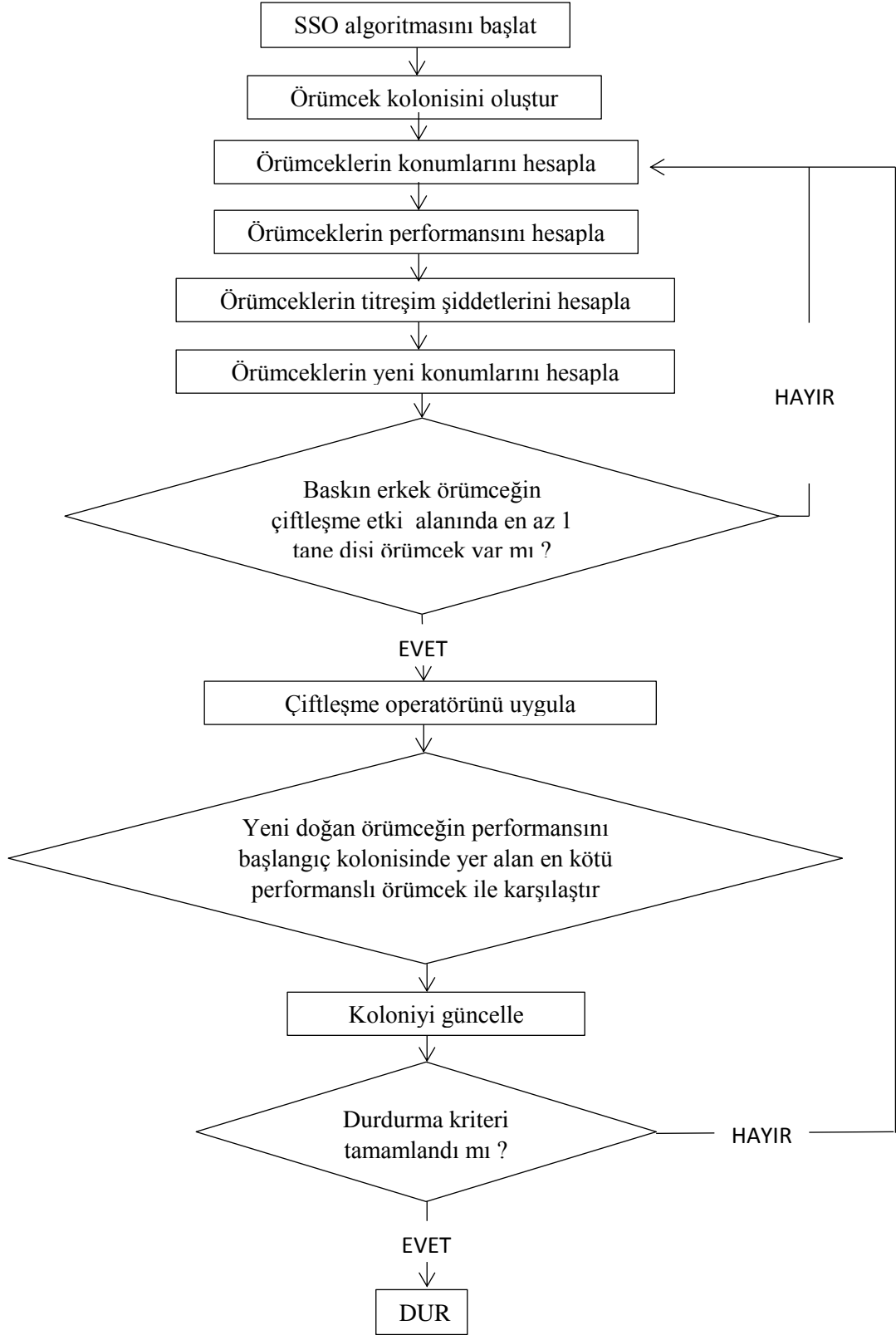
Örümceklerin çiftleşme etki alanı aşağıdaki gibi hesaplanır;

$$r = \frac{\sum_{j=1}^n (p_j^{\text{high}} - p_j^{\text{low}})}{2n} \quad (3.14)$$

Burada, r çiftleşme etki alanını, n tasarım değişkeni sayısını, p_j^{high} j . tasarım değişkeninin en yüksek değerini, p_j^{low} j . tasarım değişkeninin en düşük değerini göstermektedir.

Sosyal örümcek optimizasyonu algoritması aşağıdaki adımlardan oluşur.

1. Adım: Örümcek sayısını belirle. (3.1) ve (3.2) denklemlerini kullanarak dişi ve erkek örümcek sayısını hesapla.
2. Adım: Örümceklerin başlangıç pozisyonlarını belirle. Çiftleşme etki alanını hesapla.
3. Adım: Her bir örümceğin cezalandırılmış ağırlığını hesapla
4. Adım: Bütün örümceklerin performansını hesapla.
5. Adım: Bütün örümceklerin oluşturduğu titreşimlerin şiddetini hesapla
6. Adım: Dişi örümceklerin yeni pozisyonlarını belirle.
7. Adım: Erkek örümceklerin yeni pozisyonlarını belirle.
8. Adım: Herhangi bir baskın erkek örümceğin etki alanında dişi örümcek varsa çiftleşme operatörünü uygula yoksa 2.adıma dön.
9. Adım: Yeni doğan örümcek ile en kötü performanslı örümceği karşılaştırarak koloniyi güncelle
10. Adım: Durdurma kriteri sağlanırsa algoritmayı durdur. Aksi halde 2. Adıma dön.



Şekil 3.1: SSO algoritmasının akış şeması

4. DÜZLEM ÇELİK ÇERÇEVELERİN OPTİMUM TASARIM PROBLEMİ

Düzlem çelik çerçevelerin optimum boyutlandırma problemi LRFD-AISC (Load and Resistance Factor Design, Manual of Steel Construction, 1994) [24] Amerikan Şartnamesine göre şu şekilde tanımlanır;

Minimize edilecek fonksiyon:

$$W = \sum_{r=1}^{ng} m_r \sum_{s=1}^{t_r} l_s \quad (4.1)$$

Kısıtlayıcılar:

$$k_j = (\delta_j - \delta_{j-1}) / (\delta_{ju}) - 1 \leq 0 \quad j = 1, \dots, ns \quad (4.2)$$

$$k_t = (\delta_t / \delta_{tu}) - 1 \leq 0 \quad (4.3)$$

$$k_i = (\delta_i / \delta_{iu}) - 1 \leq 0 \quad i = 1, \dots, nd \quad (4.4)$$

$$\frac{P_{uk}}{\phi P_{nk}} \geq 0,2 \text{ ise } k_k = \frac{P_{uk}}{\phi P_{nk}} + \frac{8}{9} \left(\frac{M_{uxk}}{\phi_b M_{nxk}} \right) - 1 \leq 0 \quad k=1, \dots, nc \quad (4.5)$$

$$\frac{P_{uk}}{\phi P_{nk}} < 0,2 \text{ ise } k_k = \frac{P_{uk}}{2\phi P_{nk}} + \frac{8}{9} \left(\frac{M_{uxk}}{\phi_b M_{nxk}} \right) - 1 \leq 0 \quad k=1, \dots, nc \quad (4.6)$$

$$k_b = B_{sc} / B_{sb} - 1 \leq 0 \quad b = 1, \dots, nu \quad (4.7)$$

$$k_d = D_s / D_{s-1} - 1 \leq 0 \quad d = 1, \dots, nu \quad (4.8)$$

$$k_m = m_s / m_{s-1} - 1 \leq 0 \quad m = 1, \dots, nu \quad (4.9)$$

Burada denklem (4.1) çerçevenin ağırlığını göstermektedir. m_r , r çubuk grubu için standart çelik profilleri tablosundan seçilen profilin birim boyunun ağırlığıdır. t_r , r grubundaki toplam çubuk sayısını ve ng de çerçevedeki toplam grup sayısını göstermektedir. l_s ise r grubuna ait s çubuğunun boyunun uzunluğudur.

(4.2) denklemi çok katlı çerçeve katlarının göreceli yanal ötelenme kısıtlayıcılarını göstermektedir. δ_j ve δ_{j-1} ardışık iki katın yanal öteleme değerleridir. δ_{ju} izin verilen yanal öteleme üst sınırı olup değeri “kat yüksekliği/300” olarak alınmıştır. n_s çerçevedeki toplam kat sayısıdır.

(4.3) denklemi çerçevedeki en üst katın maksimum yanal ötelenme kısıtlayıcısını göstermektedir. δ_t en üst katın maksimum yanal ötelenme değeri, δ_{tu} ise izin verilen yanal ötelenme üst sınırı olup değeri “çerçeve yüksekliği/300” olarak alınmıştır.

(4.4) denklemi kirişlerin deplasmanlarına uygulanması gereken kısıtlayıcıları göstermektedir. δ_{iu} bir kirişin yapmış olduğu deplasman değeri, δ_{iu} ise kirişlerin deplasmanlarına uygulanan üst sınır değeri olup değeri “kiriş açıklığı/300” olarak alınmıştır. n_d çerçevedeki bu tür kısıtlayıcıların toplam sayısıdır.

(4.5) ve (4.6) denklemleri Amerikan Çelik Yapılar Enstitüsü – Yük ve Dayanım Faktörü Tasarımı (AISC-LRFD) şartnamesinden alınan mukavemet kısıtlayıcılarını göstermektedir. Her eleman eksenel kuvvet ve eğilme momentine maruzdur.

Eksenel kuvvetin çekme kuvveti olması durumunda (4.5) ve (4.6) denklemleri ile gösterilen terimler aşağıda açıklandığı gibi ifade edilirler;

P_{uk} gereken çekme kuvveti, P_{nk} nominal çekme dayanımı, ϕ dayanım azaltma katsayısı olup, çekme durumunda ϕ_t olarak gösterilir ve limit akma durumu için değeri 0,90, kırılma limit durumu için 0,75 alınır. ϕ_b eğilme için dayanım azaltma katsayısı olup değeri 0,90 ‘dır. M_{uxk} ikinci mertebeden etkileri içeren kuvvetli eksenindeki eğilme dayanımı, M_{nzk} kuvvetli eksenindeki nominal eğilme dayanımıdır.

Eksenel kuvvetin basınç kuvveti olması durumunda (4.5) ve (4.6) denklemleri ile gösterilen terimler aşağıda açıklandığı gibi ifade edilirler;

P_{uk} gereken basınç kuvveti, P_{nk} nominal basınç dayanımı, \emptyset dayanım azaltma katsayısı olup, basınç durumunda \emptyset_c olarak gösterilir ve değeri 0,85 alınır. (4.5) ve (4.6) denklemleri ile gösterilen terimlerden geriye kalanlar eksenel kuvvetin çekme olduğu durumundaki gibi alınır. n_c çerçevedeki toplam çubuk sayısıdır.

Eksenel kuvvetin nominal çekme dayanımı akma limit durumunda aşağıdaki gibi hesaplanır;

$$P_{nk} = F_y A_{gk} \quad (4.10)$$

Burada F_y çelik malzemenin akma dayanımı, A_{gk} ise k çubuğunun en kesit alanıdır.

Eksenel kuvvetin basınç olması durumunda elemanların taşıyabileceği maksimum kuvvet ise aşağıdaki gibi hesaplanır;

$$P_{nk} = F_{cr} A_{gk} \quad (4.11)$$

Burada F_{cr} , λ_c 'ye bağlı olarak hesaplanır. λ_c ise aşağıdaki denklem ile hesaplanabilir;

$$\lambda_c = \frac{KI}{r\pi} \sqrt{\frac{F_y}{E}} \quad (4.12)$$

Burada K k çubuğunun etkili uzunluk faktörünü, I k çubuğunun yanal olarak tutulmamış boyunu, E malzemenin Elastisite Modülünü ifade etmektedir.

$$\lambda_c \leq 1,5 \text{ için } F_{cr} = (0,658^{\lambda_c^2}) F_y \quad (4.13)$$

$$\lambda_c > 1,5 \text{ için } F_{cr} = \left(\frac{0,877}{\lambda_c^2} \right) F_y \quad (4.14)$$

k çubuğunun etkili uzunluk faktörü K yanal ötelenmesi önlenmemiş sistemler için aşağıdaki denklem ile hesaplanabilir;

$$K = \sqrt{\frac{1,6G_A G_B + 4(G_A + G_B) + 7,50}{G_A + G_B + 7,50}} \quad (4.15)$$

Burada kolonların üst ucu A, alt ucu B olmak üzere G_A ve G_B sırasıyla kolonların üst ve alt uçları için x-x ve y-y eksenindeki rijitlik dağıtım faktörleri olup aşağıdaki gibi hesaplanır;

$$G_A = \frac{\sum(I_c/L_c)}{\sum(I_b/L_b)} \quad , \quad G_B = \frac{\sum(I_c'/L_c')}{\sum(I_b'/L_b')} \quad (4.16)$$

Burada I_c ve I_c' göz önüne alınan kolonun sırasıyla üst ve alt ucuna rijit olarak bağlanan kolonların atalet momentleri, I_b ve I_b' göz önüne alınan kolonun üst ve alt ucuna rijit bağlı kirişlerin atalet momentleri, L_c ve L_c' göz önüne alınan kolonun üst ve alt ucuna rijit bağlı kolonların boyunu, L_b ve L_b' ise göz önüne kolonun üst ve alt ucuna rijit bağlı kirişlerin uzunluğunu göstermektedir. L_c kolon yüksekliği, I_b kiriş atalet momenti, L_b kiriş uzunluğudur. G ise burkulma boyu hesabında kullanılan katsayıdır.

Tablo 4.1 'de farklı mesnet şartlarına bağlı olarak etkili uzunluk faktörleri (K) verilmiştir.

Tablo 4.1: Etkili uzunluk faktörü (TS648, AISC-LRFD)

	(a)	(b)	(c)	(d)	(e)	(f)
Basınç çubuğunun burkulma şekli						
Teorik burkulma boyu çarpanı (K)	0.5	0.7	1.0	1.0	2.0	2.0
Tavsiye edilen boyu çarpanı (K)	0.65	0.80	1.2	1.0	2.10	2.0
Mesnet tanımları						
		Dönme ve ötelenme tutulu	Dönme serbest ve ötelenme tutulu	Dönme tutulu ve ötelenme serbest	Dönme ve ötelenme serbest	

(4.7) eşitsizliği herhangi bir s katında yer alan kolon kiriş birleşimindeki kiriş için seçilen profil başlık genişliğinin kolon için seçilen profil başlık genişliğinden daha büyük olamayacağını ifade etmektedir.

(4.8) eşitsizliği herhangi bir s katında üstteki kolon için seçilen profilin yüksekliğinin alttaki kolon için seçilen profil yüksekliğinden daha büyük olamayacağını ifade etmektedir.

(4.9) eşitsizliği herhangi bir s katında üstteki kolon için seçilen profilin birim boy ağırlığının alttaki kolon için seçilen profilin birim boy ağırlığından daha büyük olamayacağını ifade etmektedir.

(4.7), (4.8) ve (4.9) denklemlerindeki ν bu tür kısıtlayıcıların toplam sayısıdır.

Yukarıda (4.2) 'den (4.9)'a kadar olan denklemlerle açıklanan kısıtlayıcılar her zaman sağlanmayabilir. Böyle durumlar için ceza fonksiyonu geliştirilmiştir. Bu nedenle (4.1) denklemiyle verilen amaç fonksiyonu için ceza fonksiyonu $P(x)$ aşağıdaki gibi hesaplanır;

$$P(x) = W (1 + \sum_{s=1}^m v_s)^2 \quad (4.18)$$

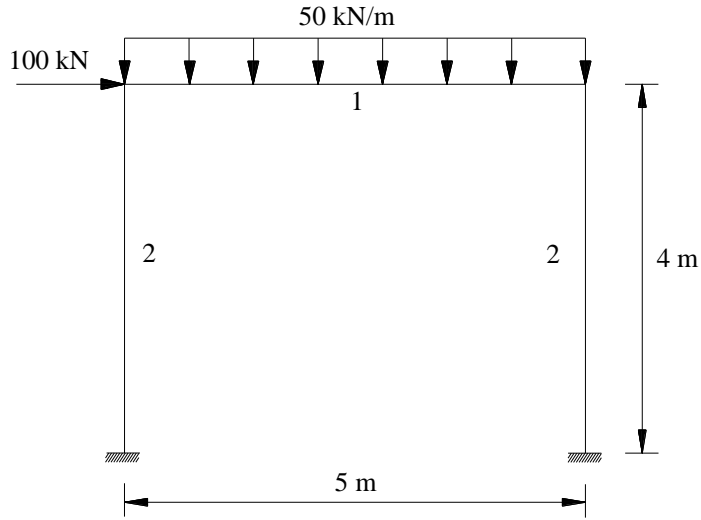
Burada W optimum tasarım problemindeki amaç fonksiyonu değeri, m tasarım problemi için verilen kısıtlayıcıların toplam sayısıdır. v_s ise ihlal etme katsayısıdır. v_s ihlal etme katsayısı aşağıdaki gibi hesaplanır;

$$\text{Eğer } k_s > 0 \text{ ise } v_s = k_s \quad s = 1, \dots, m$$

$$\text{Eğer } k_s \leq 0 \text{ ise } v_s = 0 \quad s = 1, \dots, m$$

Burada k_s kısıtlayıcı denklemleri ifade etmektedir (bkz. Denklem 4.2 - 4.9).

4.1 Tek Katlı Tek Açıklıklı Bir Çerçevenin SSO ile Boyutlandırılması



Şekil 4.1: Çerçevenin boyutları, yüklemeler ve grup numaraları

Bu örnekte şekil 4.1'de verilen tek katlı tek açıklıklı çelik bir çerçevenin optimum tasarımının sosyal örümcek optimizasyonu ile nasıl yapılacağı anlatılmıştır. Buradaki amacımız algoritmanın nasıl çalıştığını göstermek olduğu için parametre değerleri küçük seçilmiştir.

Optimum boyutlandırma problemi şu şekildedir;

$$W = \sum_{r=1}^{ng} m_r \sum_{s=1}^{tr} l_s \quad (4.19)$$

Sistemin yanal ötelenme kısıtlayıcısı ise şu şekilde alınmıştır;

$$k_j = (\delta_j - \delta_{j-1}) / (\delta_{ju}) - 1 \leq 0 \quad j = 1, \dots, ns \quad (4.20)$$

Bu örnekte kiriş için IPE, kolonlar için de IPBv profilleri kullanılacaktır. Kiriş ve kolonlar ayrı ayrı gruplandırılacaktır. Bunun için kiriş 1. grup, kolonlar ise 2. grup olacaktır. Çerçeveadaki her iki kolon için aynı profil kullanılacaktır. Algoritma, profilleri kiriş ve kolonlara aşağıdaki tabloda verilen profil havuzundan rastgele atayacaktır.

Tablo 4.2: Profil Havuzu

Sıra no	Profil	m	D	B	tw	tf	r	d	lx	ly	rx	ry	Zx	Zy	A
		kg/m	mm	mm	mm	mm	mm	mm	mm	cm ⁴	cm ⁴	cm	cm	cm ³	cm ³
1	IPE 600	122	600	220	12	19	24	514	92080	3390	24.3	4.65	3070	308	156
2	IPE 550	106	550	210	11.1	17.2	24	467	67120	2670	22.3	4.65	2440	254	134
3	IPE 500	90.7	500	200	10.2	16	21	426	48200	2140	20.4	4.31	1930	214	116
4	IPE 450	77.6	450	190	9.4	14.6	21	378	33740	1680	18.5	4.12	1500	176	98.8
5	IPE 400	66.3	400	180	8.6	13.5	21	331	23130	1320	16.5	3.95	1160	146	84.5
6	IPE 360	57.1	360	170	8	12.7	18	298	16270	1040	15	3.79	904	123	72.7
7	IPE 330	49.1	330	160	7.5	11.5	18	271	11770	788	13.7	3.55	713	98.5	62.6
8	IPE 300	42.2	300	150	7.1	10.7	15	248	8360	604	12.5	3.35	557	80.5	53.8
9	IPE 270	36.1	270	135	6.6	10.2	15	219	5790	420	11.2	3.02	429	62.2	45.9
10	IPE 240	30.7	240	120	6.2	9.8	15	190	3890	284	9.97	2.69	324	47.3	39.1
11	IPE 220	26.2	220	110	5.9	9.2	12	177	2770	205	9.11	2.48	252	37.3	33.4
12	IPE 200	22.4	200	100	5.6	8.5	12	159	1940	142	8.26	2.24	194	28.5	28.5
13	IPE 180	18.8	180	91	5.3	8	9	146	1320	101	7.42	2.05	146	22.2	23.9
14	IPE 160	15.8	160	82	5	7.4	9	127	869	68.3	6.58	1.84	109	16.7	20.1
15	IPE 140	12.9	140	73	4.7	6.9	7	112	541	44.9	5.74	1.65	77.3	12.3	16.4
16	IPE 120	10.4	120	64	4.4	6.3	7	93	318	27.7	4.9	1.45	53	8.65	13.2
17	IPE 100	8.1	100	55	4.1	5.7	7	74	171	15.9	4.07	1.24	34.2	5.79	10.3
18	IPE 80	6	80	46	3.8	5.2	5	59	80.1	8.49	3.24	1.05	20	3.69	7.64
1	IPBv1000	349	1008	302	21	40	30	868	722300	18460	40.3	6.45	14330	1220	444
2	IPBv900	333	910	302	21	40	30	770	570400	18450	36.7	6.6	12540	1220	424
3	IPBv800	317	814	303	21	40	30	674	442600	18630	33.1	6.79	10870	1230	404
4	IPBv700	301	716	304	21	40	27	582	329300	18800	29.3	7.01	9200	1240	383
5	IPBv650	293	668	305	21	40	27	534	281700	18980	27.5	7.13	8430	1240	374
6	IPBv600	285	620	305	21	40	27	486	237400	18980	25.6	7.22	7660	1240	364
7	IPBv550	278	572	306	21	40	27	438	198000	19160	23.6	7.35	6920	1250	354
8	IPBv500	270	524	306	21	40	27	390	161900	19150	21.7	7.46	6180	1250	344
9	IPBv450	263	478	307	21	40	27	344	131500	19340	19.8	7.59	5500	1260	335
10	IPBv400	256	432	307	21	40	27	298	104100	19340	17.9	7.7	4820	1260	326
11	IPBv360	250	395	308	21	40	27	261	84870	19520	16.3	7.83	4300	1270	319
12	IPBv340	248	377	309	21	40	27	243	76370	19710	15.6	7.9	4050	1280	316
13	IPBv320	245	359	309	21	40	27	225	68130	19710	14.8	7.95	3800	1280	312
14	IPBv300	233	340	310	21	39	27	208	59200	19400	14	8	3480	1250	303
15	IPBv280	189	310	288	18.5	33	24	196	39550	13160	12.8	7.4	2550	914	240
16	IPBv260	172	290	268	18	32.5	24	177	31310	10450	11.9	6.9	2160	780	220

Tablo 4.2: (devam)

17	IPBv240	157	270	248	18	32	21	164	24290	8150	11	6.39	1800	657	200
18	IPBv220	117	240	226	15.5	26	18	152	14600	5010	9.89	5.79	1220	444	149
19	IPBv200	103	220	206	15	25	18	134	10640	3650	9	5.27	967	354	131
20	IPBv180	88.9	200	186	14.5	24	15	122	7480	2580	8.13	4.77	748	277	113
21	IPBv160	76.2	180	166	14	23	15	104	5100	1760	7.25	4.26	566	212	97.1
22	IPBv140	63.2	160	146	13	22	12	92	3290	1140	6.39	3.77	411	157	80.6
23	IPBv120	52.1	140	126	12.5	21	12	74	2020	703	5.51	3.25	288	112	66.4
24	IPBv100	41.8	120	106	12	20	12	56	1140	399	4.63	2.74	190	75.3	53.2

Tablo 4.2’de görüleceği üzere kiriş için 18 adet IPE profili, kolonlar için 24 adet IPBv profili bulunmaktadır. Bunun anlamı 1. tasarım değişkeni 1 ile 18 arasında, 2. tasarım değişkeni ise 1 ile 24 arasında bir değer alacaktır. Bu profillerin sıra numaraları tasarım değişkeni olarak seçilmiştir. Algoritma 1. tasarım değişkeni için 1 ile 18 arasında, 2. tasarım değişkeni için 1 ile 24 arasında bir değer belirleyerek profil atayacaktır. Örneğin algoritma 1. tasarım değişkeni için 8 sıra numarasını, 2. tasarım değişkeni için 22 sıra numarasını seçsin. Bunun anlamı çerçevenin kirişi için tablo 4.2’de verilen IPE profilinden 8. sıradakini yani IPE300 ve kolonlar için IPBv profilinden 22. sıradakini yani IPBv140 profilini seçmiş olur.

Algoritma profil atama işlemini örümcek sayısı kadar yapar. Profil atama işlemi tamamlandıktan sonra çerçeveler oluşturulur. Her bir çerçeve sosyal örümcek olarak sembolize edilir. Her bir örümcek için şekil 4.1’ de verilen yüklemeler altında analiz yapılır ve çerçevenin yanal ötelenmesi hesaplanır. Böylece yanal ötelenme kısıtlayıcısının sağlanıp sağlanmadığı kontrol edilir. Eğer kısıtlayıcı sağlanıyorsa ceza uygulanmaz, sağlanmıyorsa ceza uygulanır.

Tek katlı olan çerçevede birinci katın yanal ötelenmesinin üst sınırı kat yüksekliği / 300 olarak belirlenmiştir. Yani her iki yanal ötelenme 1,33 cm’ den fazla olamaz. St37 çeliği olarak tanımlanan malzemenin elastisite modülü 2100000 kg/cm², akma gerilmesi 2400 kg/cm² alınmıştır. Çerçevenin analizi SAP2000 programıyla yapılmıştır. Formüllerde yer alan ‘‘rand’’ değeri 0 ile 1 arasında rastgele belirlenmiştir.

SSO algoritması örümcek sayısını belirleyerek başlar.

1. Adım: Örümcek sayısının belirlenmesi, dişi ve erkek örümcek sayısının hesaplanması:

Örümcek sayısı 4 olsun. $N = 4$

Dişi örümcek sayısı : $N_f = N \cdot (0,90 - \text{rand} \cdot 0,25)$

$$N_f = 4 \cdot (0,90 - 0,32 \cdot 0,25) = 3$$

Erkek örümcek sayısı : $N_m = N - N_f$

$$N_m = 4 - 3 = 1$$

2. Adım: Örümceklerin başlangıç pozisyonlarının belirlenmesi, çiftleşme etki alanının hesaplanması

Dişi örümceklerin başlangıç pozisyonları : $f_{ij}^0 = p_j^{\text{low}} + \text{rand} \cdot (p_j^{\text{high}} - p_j^{\text{low}})$

$$f_{11}^0 = 1 + 0.43 \cdot (18 - 1) = 8$$

$$f_{12}^0 = 1 + 0.93 \cdot (24 - 1) = 22$$

$$f_{21}^0 = 1 + 0.62 \cdot (18 - 1) = 11$$

$$f_{22}^0 = 1 + 0.14 \cdot (24 - 1) = 4$$

$$f_{31}^0 = 1 + 0.95 \cdot (18 - 1) = 17$$

$$f_{32}^0 = 1 + 0.44 \cdot (24 - 1) = 11$$

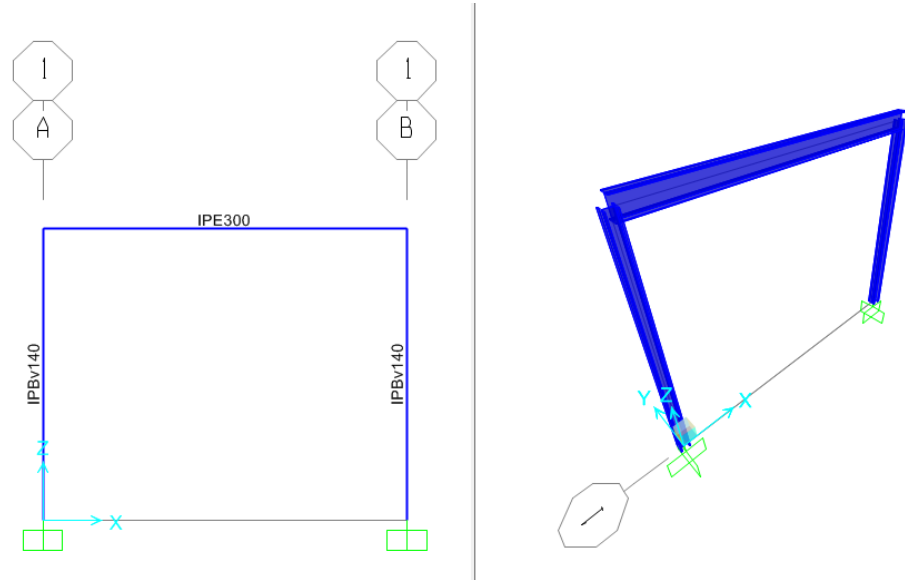
Erkek örümceklerin başlangıç pozisyonları : $m_{kj}^0 = p_j^{low} + rand . (p_j^{high} - p_j^{low})$

$$m_{11}^0 = 1 + 0.24 . (18 - 1) = 5$$

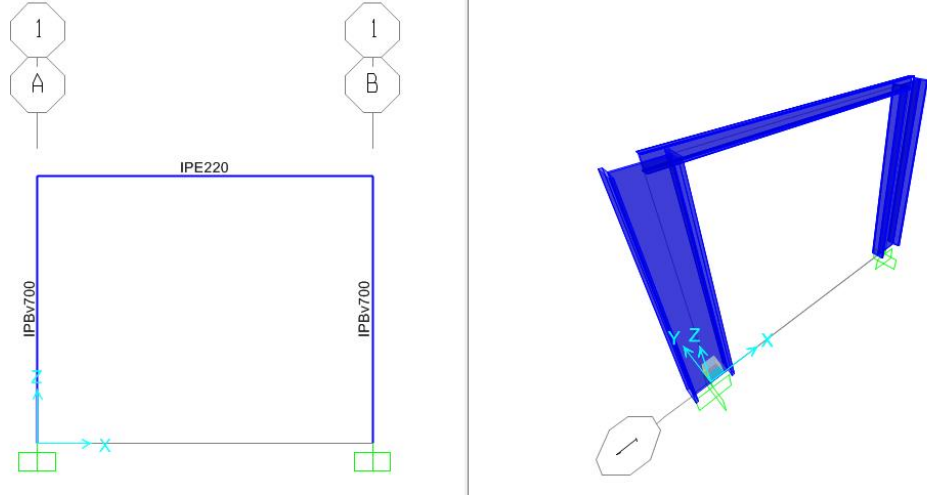
$$m_{12}^0 = 1 + 0.59 . (24 - 1) = 14$$

Tablo 4.3: Örümceklerin başlangıç kesitleri

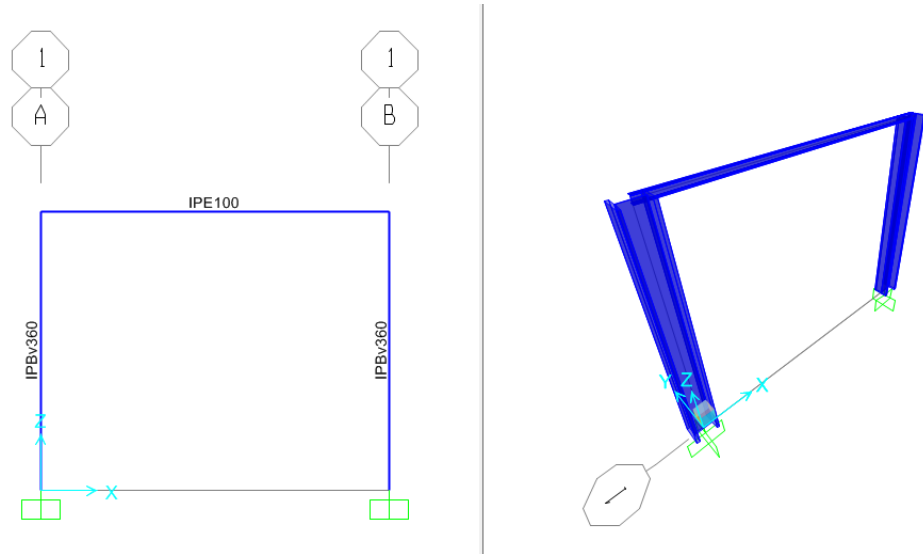
Örümcekler	1. Tasarım Değişkeni (Kiriş)	2. Tasarım Değişkeni (Kolon)
f_1	8 (IPE300)	22 (IPBv140)
f_2	11 (IPE220)	4 (IPBv700)
f_3	17 (IPE100)	11 (IPBv360)
m_1	5 (IPE400)	14 (IPBv300)



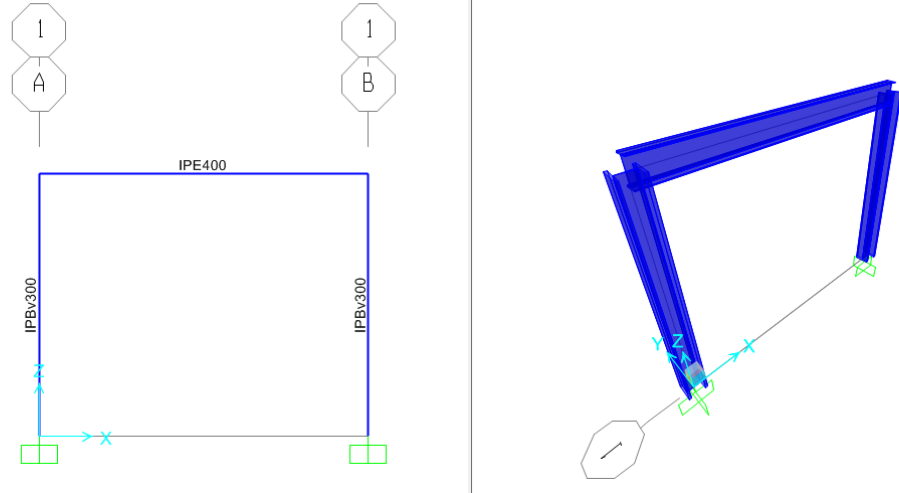
Şekil 4.2: f_1 örümceği ile sembolize edilen çerçeve ve 3D görünüşü



Şekil 4.3: f_2 örümceği ile sembolize edilen çerçeve ve 3D görünüşü



Şekil 4.4: f_3 örümceği ile sembolize edilen çerçeve ve 3D görünüşü

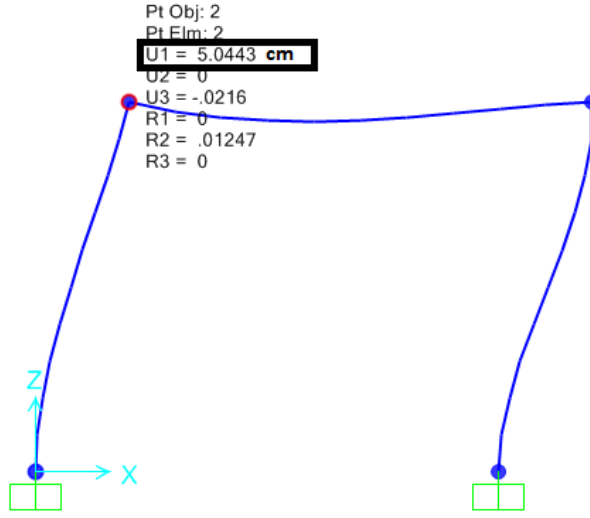


Şekil 4.5: m_1 örümceği ile sembolize edilen çerçeve ve 3D görünüşü

Çiftleşme etki alanı:
$$r = \frac{\sum_{j=1}^n (p_j^{\text{high}} - p_j^{\text{low}})}{2 \cdot n}$$

$$r = \frac{(18-1) + (24-1)}{2 \cdot 2} = 10$$

3. Adım: Örümceklerin cezalandırılmış ağırlıklarının hesaplanması



Şekil 4.6: f_1 örümceği ile sembolize edilen çerçevenin yanal öteleme değeri

Burada f_1 dişi örümceğinin cezalandırılmış ağırlığının nasıl hesaplandığı gösterilmiş, diğer örümceklerin cezalandırılmış ağırlık değerleri tablo 4.5' de verilmiştir.

f_1 dişi örümceği

$$W_{f_1} = 42,2 * 1 * 5 + 63,2 * 2 * 4 = 716,6 \text{ kg} = 0,72 \text{ ton}$$

$$\delta_j - \delta_{j-1} \leq (h_j / 300) \quad 5,04 \leq \frac{400}{300} \quad \text{sağlanmıyor, ceza fonksiyonu uygulanır}$$

$$P(x) = \frac{5,04}{(400/300)} - 1 = 2,80$$

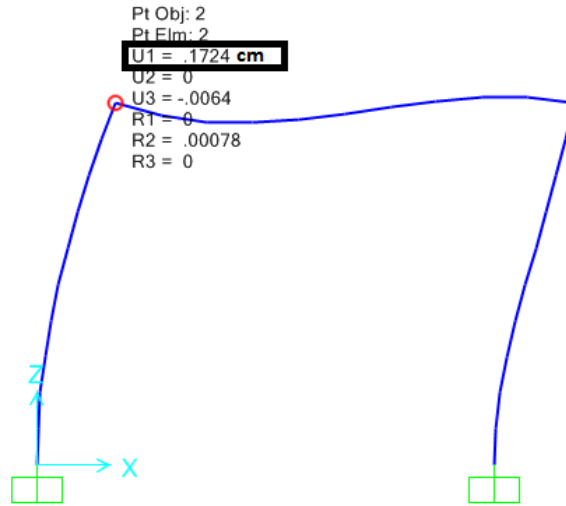
$$W_p = W_{f_1} * (1 + P(x))^2 = 0,72 * (1 + 2,80)^2 = 10,40 \text{ ton}$$

Tablo 4.4: Örümceklerin yanal öteleme değerleri ve yanal öteleme üst sınırı

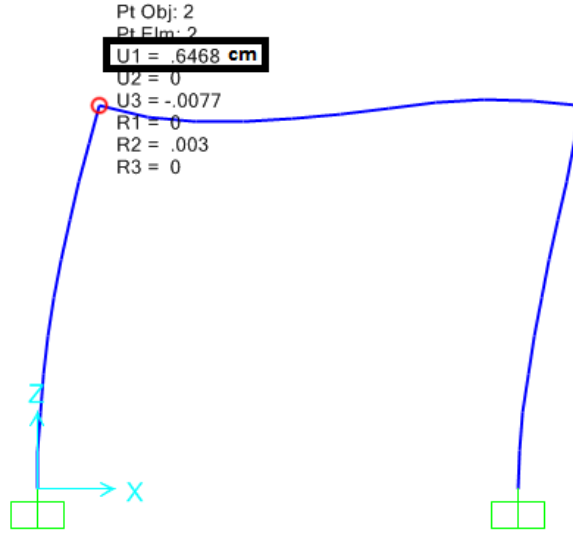
Örümcekler	Yanal öteleme değeri (cm)	Yanal öteleme üst sınırı (cm)
f_1	5,04	1,33
f_2	0,17	1,33
f_3	0,65	1,33
m_1	0,50	1,33

Tablo 4.5: Örümceklerin cezalandırılmış ağırlıkları

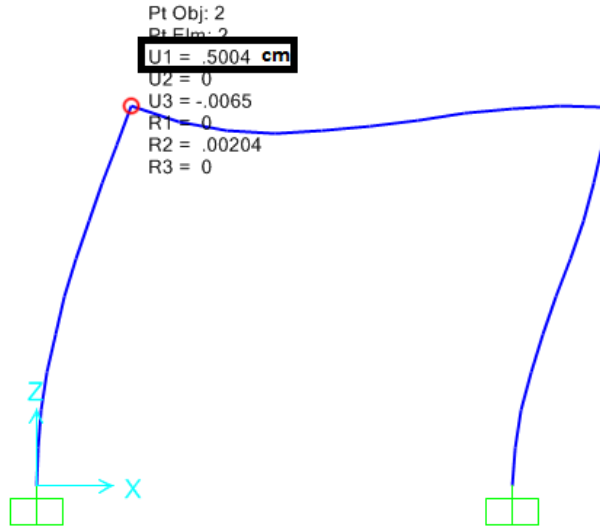
Örümcekler	Ağırlık (ton)	Ceza değeri P(x)	Cezalandırılmış ağırlık (ton)
f_1	0,72	2,80	10,40
f_2	2,54	0	2,54
f_3	2,04	0	2,04
m_1	2,20	0	2,20



Şekil 4.7: f_2 örümceği ile sembolize edilen çerçevenin yanal öteleme değeri



Şekil 4.8: f_3 örümceği ile sembolize edilen çerçevenin yanıl öteleme değeri



Şekil 4.9: m_1 örümceği ile sembolize edilen çerçevenin yanıl öteleme değeri

4. Adım: Örümceklerin performanslarının hesaplanması

$$\text{Performans: } w = \frac{1}{W}$$

$$w_1 = \frac{1}{w_p} = \frac{1}{10,40} = 0,10 \quad (\text{en kötü örümcek})$$

$$w_2 = \frac{1}{w_p} = \frac{1}{2,54} = 0,39$$

$$w_3 = \frac{1}{w_p} = \frac{1}{2,04} = 0,49 \quad (\text{en iyi dişi örümcek})$$

$$w_4 = \frac{1}{w_p} = \frac{1}{2,20} = 0,45$$

5. Adım: Örümceklerin titreşimlerinin hesaplanması

$$\text{Titreşim şiddeti : } \text{Vib}_{ij} = w_j \cdot e^{-d^2_{ij}}$$

Burada f_1 ile f_2 dişi örümcekleri arasındaki uzaklığın nasıl hesaplandığı gösterilmiş, diğer örümceklerin birbirlerine uzaklıkları tablo 4.6' da verilmiştir.

$$d_{ij} = \sqrt{\sum_{j=1}^n (s_{1j} - s_{2j})^2} / \sqrt{\sum_{j=1}^n (p_j^{\text{high}} - p_j^{\text{low}})^2}$$

$$d_{12} = \sqrt{(8 - 11)^2 + (22 - 4)^2} / \sqrt{(18 - 1)^2 + (24 - 1)^2} = 0,64$$

Tablo 4.6: Örümceklerin birbirlerine uzaklıkları

Örümcekler	f_1	f_2	f_3	m_1
f_1		0,64	0,50	0,30
f_2	0,64		0,32	0,41
f_3	0,50	0,32		0,43
m_1	0,30	0,41	0,43	

a) dişi örümcekler için $Vibc_i$ hesabı

f_1 dişi örümceği için;

f_1 'e en yakın dişi örümcek f_3 ' tür

$$Vibc_i = 0,49 * e^{-0.50^2} = 0,38$$

f_2 dişi örümceği için;

f_2 'ye en yakın dişi örümcek f_3 ' tür

$$Vibc_i = 0,49 * e^{-0.32^2} = 0,44$$

b) en iyi dişi örümcek için $Vibb_i$ hesabı

en iyi dişi örümcek f_3 ' tür.

f_3 ile f_1 arasındaki titreşim;

$$Vibb_i = 0,49 * e^{-0.50^2} = 0,38$$

f_3 ile f_2 arasındaki titreşim;

$$Vibb_i = 0,49 * e^{-0.32^2} = 0,44$$

c) erkek örümcekler için $Vibf_i$ hesabı

m_1 'e en yakın dişi örümcek f_1 ' dir.

$$Vibf_i = 0,45 * e^{-0.30^2} = 0,41$$

6. Adım: Dişi örümceklerin yeni pozisyonlarının belirlenmesi

$$f_{ij}^{k+1} = \begin{cases} f_{ij}^k + \alpha \cdot Vibc_i \cdot (s_c - f_{ij}^k) + \beta \cdot Vibb_i \cdot (s_b - f_{ij}^k) + \delta \cdot \left(\text{rand} - \frac{1}{2}\right) & 0 < \text{rand} \leq \text{PF} \\ f_{ij}^k - \alpha \cdot Vibc_i \cdot (s_c - f_{ij}^k) - \beta \cdot Vibb_i \cdot (s_b - f_{ij}^k) + \delta \cdot \left(\text{rand} - \frac{1}{2}\right) & \text{PF} < \text{rand} < 1 \end{cases}$$

$$f_{11}^1 = 8 + 0,25 * 0,38 * (17 - 8) + 0,73 * 0,38 * (17 - 8) + 0,16 * (0,65 - 0,50) = 11$$

$$f_{12}^1 = 22 + 0,45 * 0,38 * (11 - 22) + 0,51 * 0,38 * (11 - 22) + 0,66 * (0,46 - 0,50) = 17$$

$$f_{21}^1 = 11 - 0,47 * 0,44 * (17 - 11) - 0,23 * 0,44 * (17 - 11) + 0,91 * (0,85 - 0,50) = 9$$

$$f_{22}^1 = 4 + 0,85 * 0,44 * (11 - 4) + 0,97 * 0,44 * (11 - 4) + 0,25 * (0,56 - 0,50) = 9$$

Sosyal örümcek optimizasyonunda, en iyi dişi örümceğin başlangıç pozisyonunda kaldığı kabul edilmektedir.

7. Adım: Erkek örümceklerin yeni pozisyonlarının belirlenmesi

$$m_{ij}^{k+1} = \begin{cases} m_{ij}^k + \alpha \cdot \text{Vib}f_i \cdot (s_f - m_{ij}^k) + \delta \cdot \left(\text{rand} - \frac{1}{2}\right) & w_{N_{f+i}} > w_{N_{f+m}} \\ m_{ij}^k + \alpha \cdot \left(\frac{\sum_{h=1}^{N_m} m_h^k \cdot w_{N_{f+h}}}{\sum_{h=1}^{N_m} w_{N_{f+h}}} - m_{ij}^k\right) & w_{N_{f+i}} \leq w_{N_{f+m}} \end{cases}$$

$$m_{11}^1 = 5 + 0,36 * \left(\frac{5 * 0,45}{0,45} - 5\right) = 5$$

$$m_{12}^1 = 14 + 0,48 * \left(\frac{14 * 0,45}{0,45} - 14\right) = 14$$

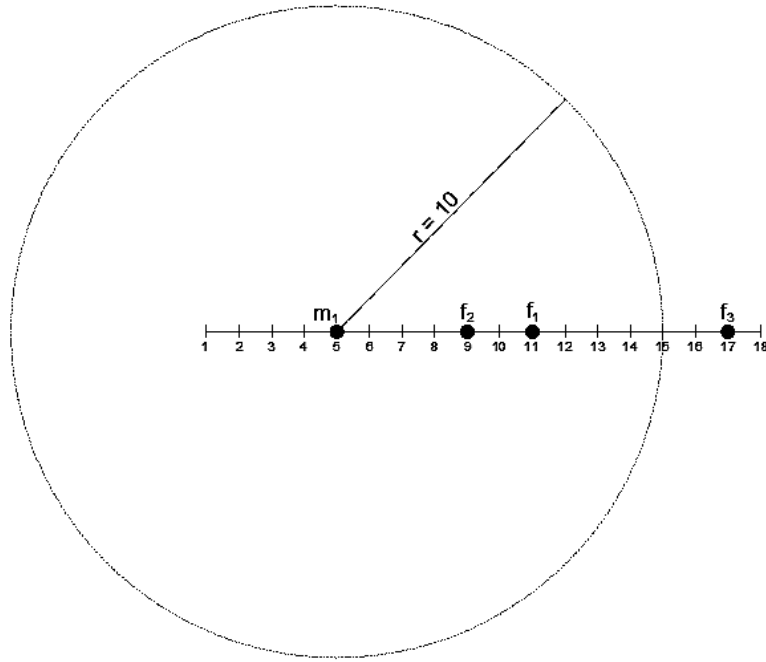
Tablo 4.7: Örümceklerin yeni kesitleri

Örümcekler	1. Tasarım Değişkeni (Kiriş)	2. Tasarım Değişkeni (Kolon)
f_1	11 (IPE220)	17 (IPBv240)
f_2	9 (IPE270)	9 (IPBv450)
f_3	17 (IPE100)	11 (IPBv360)
m_1	5 (IPE400)	14 (IPBv300)

8. Adım: Çiftleşme operatörünün uygulanması

Çiftleşme operatörünün uygulanabilmesi için kolonide bulunan baskın erkek örümceğin etki alanında dişi örümceğin yer alıp almadığına bakılır. Baskın erkek örümceğin etki alanında en az 1 tane dişi örümcek yer almak zorundadır.

1. tasarım değişkeni için ;



Şekil 4.10: 1. Tasarım değişkeni için örümceklerin çiftleşme etki alanı

çiftleşme etki alanında yer alan örümcekler;

$$m_1 : 5 \quad w = 0,45$$

$$f_2 : 9 \quad w = 0,39$$

$$f_1 : 11 \quad w = 0,10$$

Rulet Metodu;

$$P_i = \frac{w_i}{\sum w_i}$$

$$P_{m_1} = \frac{0,45}{0,45 + 0,39 + 0,10} = 0,48$$

$$P_{f_2} = \frac{0,39}{0,45 + 0,39 + 0,10} = 0,41$$

$$P_{f_1} = \frac{0,10}{0,45 + 0,39 + 0,10} = 0,11$$

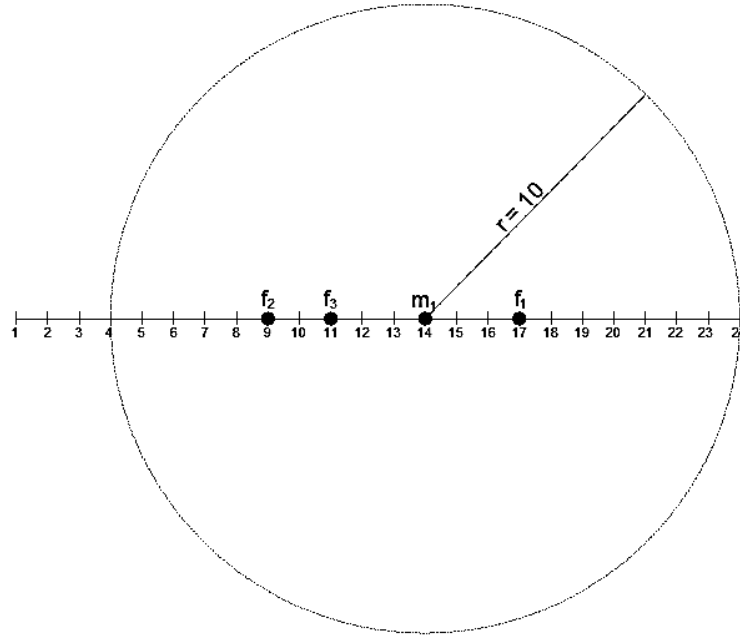
rand = 0,65 olsun

$0,65 > 0,48$ işleme devam edilir

$0,65 < 0,48 + 0,41$ olduğundan,

yeni doğan örümcek için 1. Tasarım değişkeni f_2 örümceğinden alınır. Yani yeni doğan örümceğin 1. Tasarım değişkeni 9 olarak alınır.

2. tasarım deęiřkeni için ;



Şekil 4.11: 2. Tasarım deęiřkeni için örümceklerin çiftleşme etki alanı

çiftleşme etki alanında yer alan örümcekler;

$$m_1 : 14 \quad w = 0,45$$

$$f_1 : 17 \quad w = 0,10$$

$$f_2 : 9 \quad w = 0,39$$

$$f_3 : 11 \quad w = 0,49$$

Rulet Metodu;

$$P_i = \frac{w_i}{\sum w_i}$$

$$P_{m_1} = \frac{0,45}{0,45 + 0,10 + 0,39 + 0,49} = 0,31$$

$$P_{f_1} = \frac{0,10}{0,45 + 0,10 + 0,39 + 0,49} = 0,06$$

$$P_{f_2} = \frac{0,39}{0,45 + 0,10 + 0,39 + 0,49} = 0,27$$

$$P_{f_3} = \frac{0,49}{0,45 + 0,10 + 0,39 + 0,49} = 0,34$$

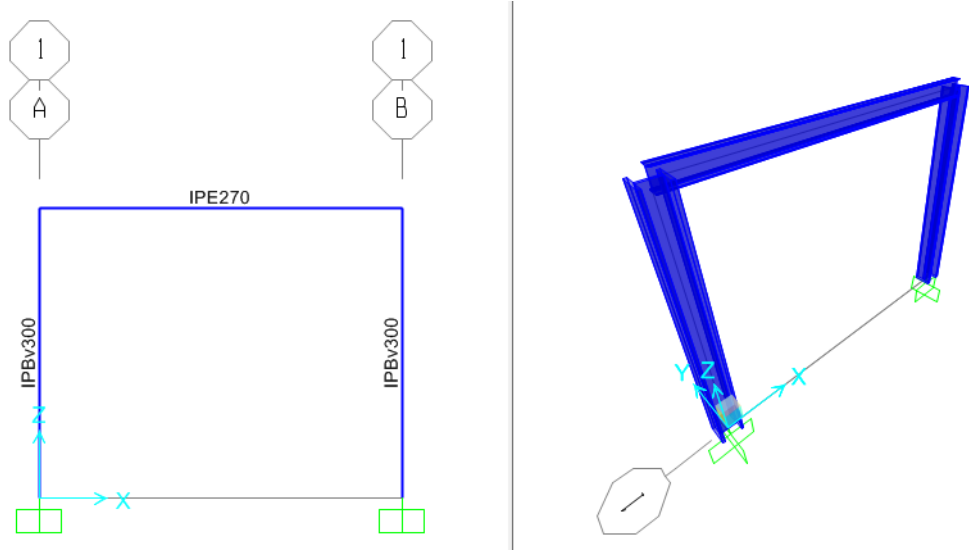
rand = 0,63 olsun

0,63 > 0,34 işleme devam edilir

0,63 < 0,34 + 0,31 olduğundan,

yeni doğan örümcek için 2. Tasarım değişkeni m_1 örümceğinden alınır. Yani yeni doğan örümceğin 2. Tasarım değişkeni 14 olarak alınır.

$s_{new} : [9 , 14]$



Şekil 4.12: s_{new} örümceği ile sembolize edilen çerçeve ve 3D görünüşü

s_{new} örümceği;

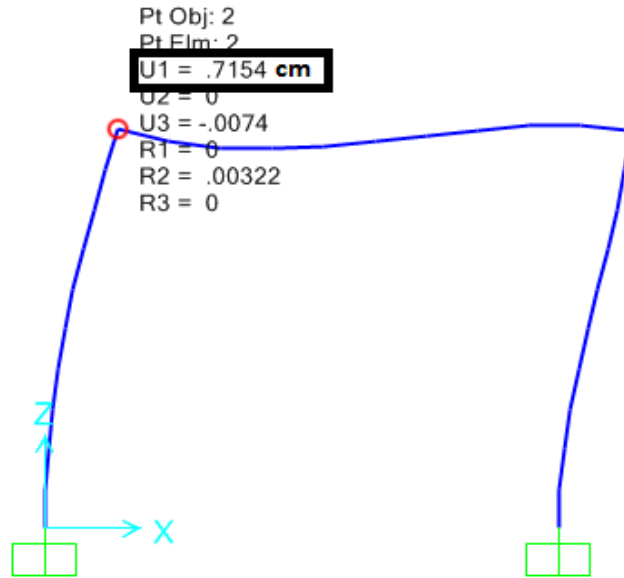
$$W_{s_{new}} = 36,1 * 1 * 5 + 233 * 4 * 2 = 2044,5 \text{ kg} = 2,05 \text{ ton}$$

$$\delta_j - \delta_{j-1} \leq (h_j / 300) \quad 0,72 \leq \frac{400}{300} \quad \text{sağlanıyor}$$

$$P(x) = 0$$

$$W_p = W_{s_{new}} = 2,05 \text{ ton}$$

$$w_{s_{new}} = \frac{1}{W_p} = \frac{1}{2,05} = 0,48$$



Şekil 4.13: s_{new} örümceği ile sembolize edilen çerçevenin yanal öteleme değeri

9. Adım: Koloninin güncellenmesi

Kolonide yer alan en kötü performanslı örümcek ile yeni doğan örümcek kıyaslanır, en kötü örümcek f_1 dişi örümceğidir.

$$w_{s_{new}} = 0,48$$

$$w_1 = 0,10$$

$w_{snew} > w_1$ olduğu için, f_1 dişi örümceği koloniden çıkarılarak yerine yeni doğan örümcek dahil edilir. Yeni doğan örümceğin cinsiyeti koloniden çıkarılan örümceğin cinsiyeti ile aynı olacağı için dişidir.

Güncellenen koloni şu şekildedir :

$$f_{snew} : [9 , 14] \quad W = 2044,5 \text{ kg}$$

$$f_2 : [9 , 9] \quad W = 2104 \text{ kg}$$

$$f_3 : [17 , 11] \quad W = 2040,5 \text{ kg}$$

$$m_1 : [5 , 14] \quad W = 2195,5 \text{ kg}$$

Burada W çerçeve ağırlığını göstermektedir.

10. Adım: iterasyon sayısı 1 tamamlandığı için algoritmayı durdur.

Optimum çerçeve ağırlığı güncellenen kolonide ağırlığı minimum olan örümcektir. Yani f_3 ile gösterilen örümcek optimum sonuçtur. Burada optimum tasarımda kiriş için IPE100, kolonlar için IPBv360 profilleri kullanılmıştır. Buna göre optimum çerçeve ağırlığı 2040,5 kg'dır.

5. SAYISAL ÖRNEKLER

Bu bölümde sosyal örümcek optimizasyon (SSO) algoritmasının performansını görmek amacıyla literatürlerden seçilen örnekler kullanılmıştır. SSO algoritması visual basic applications (VBA) program dili ile kodlanmıştır. Hazırlanan program kodları SAP2000 programını çağırarak problem için verilen kesit özelliklerini ve sistem geometrisine ait bilgileri alarak yapının deplasmanını ve mukavemet değerlerini hesaplamaktadır. Yanal ötelenme sınırları ise VBA dilinde kod olarak yazılmıştır.

1.örnekte [10] numaralı literatürden seçilen iki açıklıklı altı katlı çelik çerçeve SSO algoritması kullanılarak boyutlandırılmıştır. Tasarım sonuçları, literatürde mevcut yöntemlere ait sonuçlarla karşılaştırıldığında SSO algoritmasının güçlü bir yöntem olduğunu ortaya koymaktadır.

2.örnekte [3] ve [14] numaralı literatürlerden seçilen iki açıklıklı üç katlı çelik çerçeve kullanılmıştır. Bu çerçeve SSO algoritması kullanılarak boyutlandırılmıştır. SSO algoritması diğer yöntemlerle aynı sonuçları bulmuştur.

3.örnekte [9] numaralı literatürden seçilen tek açıklıklı on katlı çelik çerçeve kullanılmıştır. Tasarım sonuçları SSO algoritmasının rekabetçi bir algoritma olduğunu göstermiştir.

4.örnekte [25] ve [26] numaralı literatürlerden seçilen üç açıklıklı on beş katlı çelik çerçeve kullanılmıştır. Bu çerçeve SSO ile boyutlandırılmış ve sonuçlar diğer yöntemlerle karşılaştırılmıştır. SSO algoritması ilgili literatürlerde verilen diğer optimizasyon yöntemlerinden daha iyi sonuçlar vermiştir.

Geliştirilen algoritma programı Microsoft Windows 7 Ultimate 64 Bit işletim sistemli Intel (R) Core (TM)2 Duo CPU 2.53 GHz işlemci özelliklerine sahip ve 3,00 GB belleği olan kişisel bilgisayarda çalıştırılmıştır.

Program çalıştırılmadan önce örümcek sayısı, PF değeri ve maksimum iterasyon sayısı belirlenmelidir.

Tablo 5.1: Tasarım örneklerinde kullanılan SSO parametreleri

Tasarım Örnekleri	Örümcek Sayısı	PF	Max İterasyon
2 Açıklıklı 6 Katlı Çerçeve	50	0,125	2000
2 Açıklıklı 3 Katlı Çerçeve	50	0,100	100
1 Açıklıklı 10 Katlı Çerçeve	50	0,100	3000
3 Açıklıklı 15 Katlı Çerçeve	50	0,100	6000

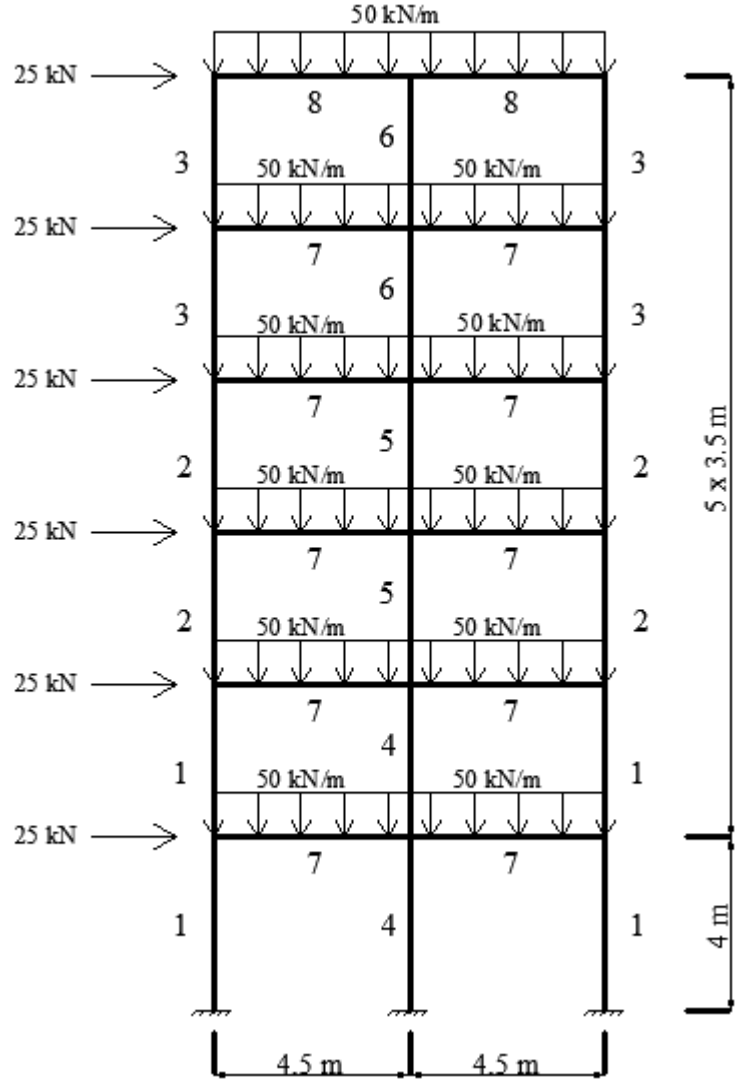
Örnek 1. İki Açıklıklı Altı Katlı Çerçeve

Şekil 5.1’ de otuz elemandan oluşan iki açıklıklı altı katlı düzlem çelik çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları gösterilmiştir.

Çerçevenin mukavemet ve deplasman kısıtlayıcıları Amerikan Çelik Yapılar Enstitüsü – Yük ve Dayanım Faktörü Tasarımı (AISC – LRFD 99) şartnamesinden alınmıştır. Katlar arası yanal öteleme üst sınırı “kat yüksekliği / 300” alınmıştır. Kirişler ve kolonlar LRFD’ de yer alan ve Ek A’ da verilen 267 adet W kesitlerinden seçilmiştir. Kullanılan malzemenin Elastisite Modülü 200000 N/mm², Akma Dayanımı 248,2 N/mm² alınmıştır.

Sistem kolonlar için 6, kirişler için 2 olmak üzere 8 gruptan oluşmaktadır. Kolonlar için dış ve iç kolonlar her iki katta aynı kesit olarak alınmıştır. Kirişler ise ilk beş kattaki tüm kirişler aynı kesit olarak belirlenmiştir. Aynı şekilde çatı katındaki tüm kirişler de aynı kesit olarak alınmıştır.

Bu düzlem çelik çerçeve daha önce M. P. Saka tarafından parçacık sürüsü optimizasyonu (PSO) ve harmoni arama algoritması (HS) kullanılarak boyutlandırılmıştır. Bu çalışmada ise aynı çerçeve sosyal örümcek optimizasyon algoritması kullanılarak optimize edilmiş ve sonuçlar tablo 5.2’ de gösterilmiştir.



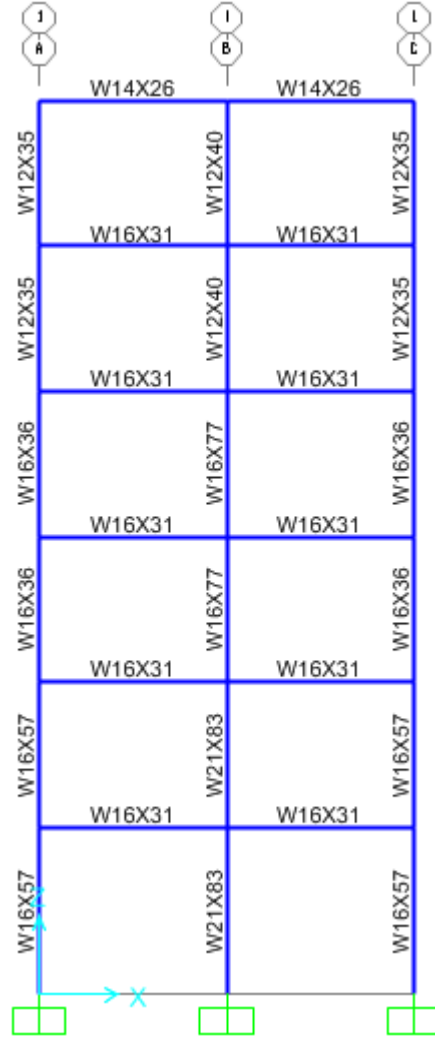
Şekil 5.1: İki açıklıklı altı katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları

Tablo 5.2: İki açıklıklı altı katlı çerçevenin optimum sonuçları

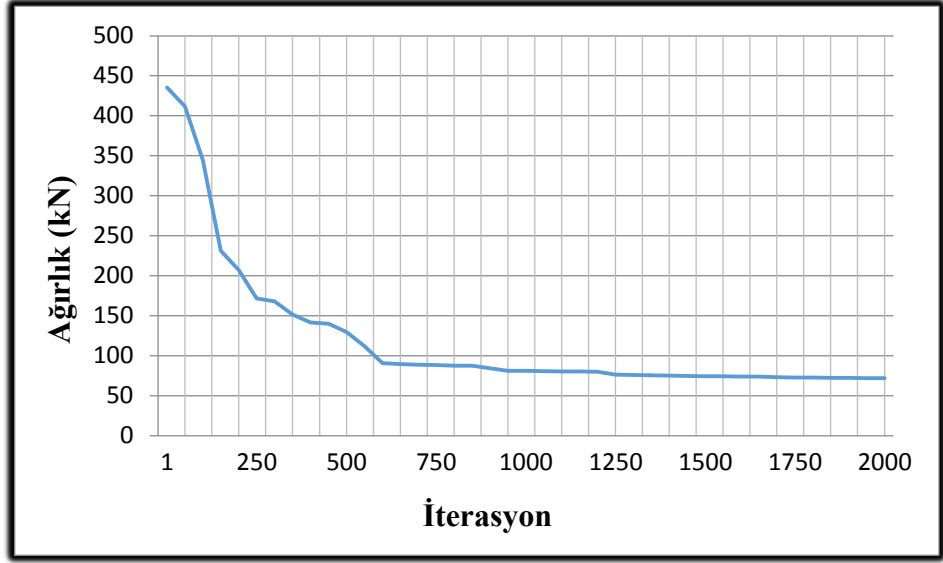
Grup No	Grup Adı	HS [10]	PSO [10]	Bu çalışma (SSO)
1	Kolon	W18x55	W16x57	W16x57
2	Kolon	W12x50	W16x40	W16x36
3	Kolon	W8x31	W10x39	W12x35
4	Kolon	W21x73	W24x62	W21x83
5	Kolon	W18x65	W24x62	W16x77
6	Kolon	W12x40	W8x40	W12x40
7	Kiriş	W16x40	W14x30	W16x31
8	Kiriş	W14x22	W18x65	W14x26
Ağırlık (kN)		76,78	73,87	70,57
En üst kat yanal ötelenme (cm)		*	5,521	5,056
Maksimum katlar arası yanal ötelenme (cm)		*	1,153	0,821
Maksimum gerilme oranı		*	0,96	0,93
Maksimum iterasyon sayısı		*	6500	2000

* Değerler önceki çalışmalarda verilmemiştir.

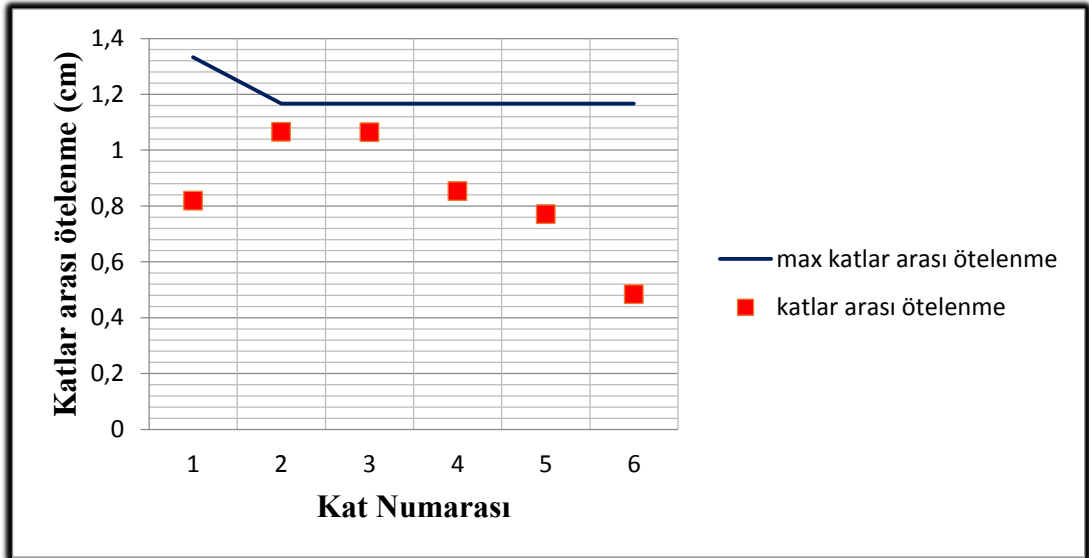
Bu örnekte iki açıklıklı altı katlı çerçevenin optimum tasarımı sosyal örümcek optimizasyon (SSO) algoritması kullanılarak yapılmıştır. Optimum tasarım sonuçları diğer optimizasyon yöntemleriyle karşılaştırılmış ve sonuçlar SSO algoritmasının çok etkili bir yöntem olduğunu göstermiştir. SSO algoritması ile optimum tasarımı yapılan çerçevenin ağırlığı HS algoritmasından %8,80, PSO algoritmasından ise %4,68 daha az bulunmuştur.



Şekil 5.2: İki açıklıklı altı katlı çerçevenin optimum kesitleri



Şekil 5.3: İki açıklıklı altı katlı çerçevenin ağırlık iterasyon ilişkisi



Şekil 5.4: İki açıklıklı altı katlı çerçevenin x doğrultusundaki yanal ötelenmeleri

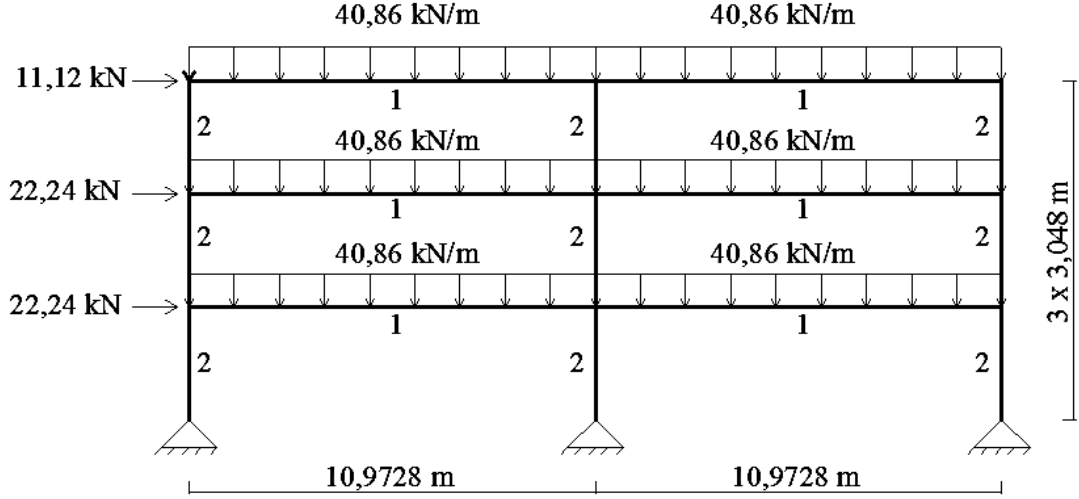
Örnek 2. İki Açıklıklı Üç Katlı Çerçeve

Şekil 5.5’ de on beş elemandan oluşan iki açıklıklı üç katlı düzlem çelik çerçevenin boyutları, yükleme şekli, elemanların grup numaraları gösterilmiştir.

Mukavemet ve deplasman kısıtlayıcıları Amerikan Çelik Yapılar Enstitüsü-Yük ve Dayanım Faktörü Tasarımı (AISC-LRFD, 1999) şartnamesinden alınmıştır. Kiriş elemanları LRFD’ de bulunan ve Ek A ‘da verilen 267 adet W kesitlerinden seçilmiştir, kolon elemanları ise 18 adet W10 kesitlerinden seçilmiştir. Kirişlerin etkili boy faktörü 0,167 alınmıştır. Kullanılan malzeme için Elastisite Modülü $E=200000$ Mpa, Akma Gerilmesi $f_y = 248,2$ Mpa alınmıştır. Bu çerçeve için yanal ötelenme sınırı uygulanmamıştır.

Sistem 2 gruptan oluşmaktadır. Kirişler 1.grup ve kolonlar 2.grup olarak belirlenmiştir. Sistem 2 açıklıktan oluşmaktadır ve her bir açıklık 10,9728 m uzunluğundadır. Her bir kat 3,048 m yüksekliğindedir. Sistem şekil 5.4’ de gösterildiği gibi tek tip yüklemeye maruzdur. Tüm kirişlerin üzerinde 40,86 kN/m değerinde düzgün yayılı yük mevcuttur. X doğrultusunda düğüm noktalarına en üst kat için 11,12 kN, birinci ve ikinci kat için 22,24 kN büyüklüğünde yatay tekil yük etki ettirilmiştir. Mesnetlenme tipi sabit mafasallı mesnettir.

Bu düzlem çelik çerçeve daha önce C. V. Camp tarafından karınca kolonisi optimizasyon algoritması (ACO ve P. Murren tarafından tasarım odaklı harmoni arama algoritması (DDHS) kullanılarak boyutlandırılmıştır. Bu çalışmada ise sosyal örümcek optimizasyon algoritması kullanılarak boyutlandırılmıştır. Sonuçlar diğer yöntemlerle karşılaştırılmış ve tablo 5.3’ te gösterilmiştir.



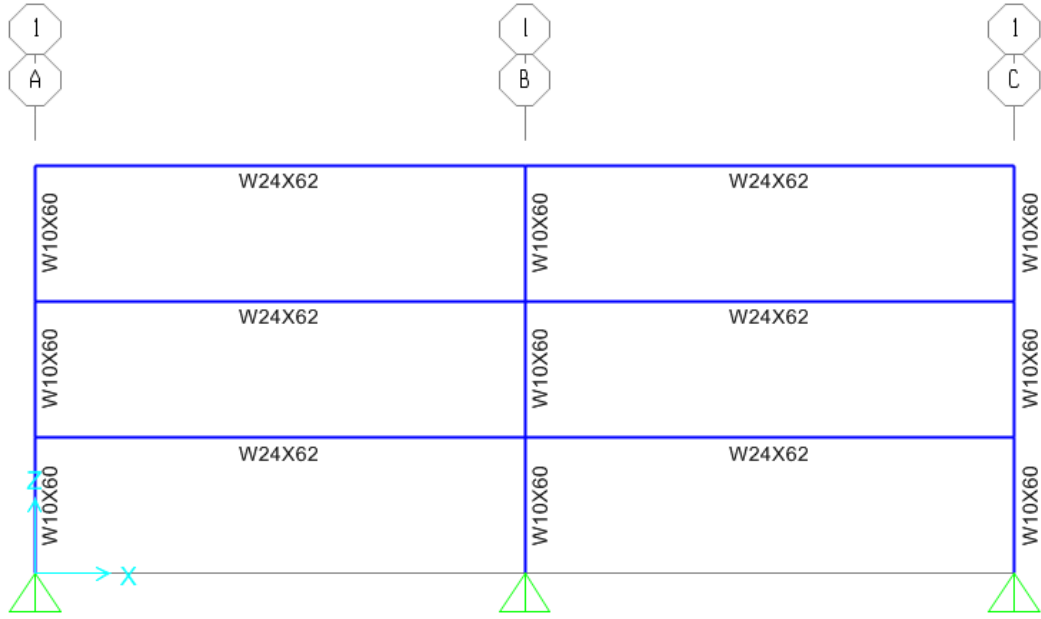
Şekil 5.5: İki açıklıklı üç katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları

Tablo 5.3: İki açıklıklı üç katlı çerçevenin optimum sonuçları

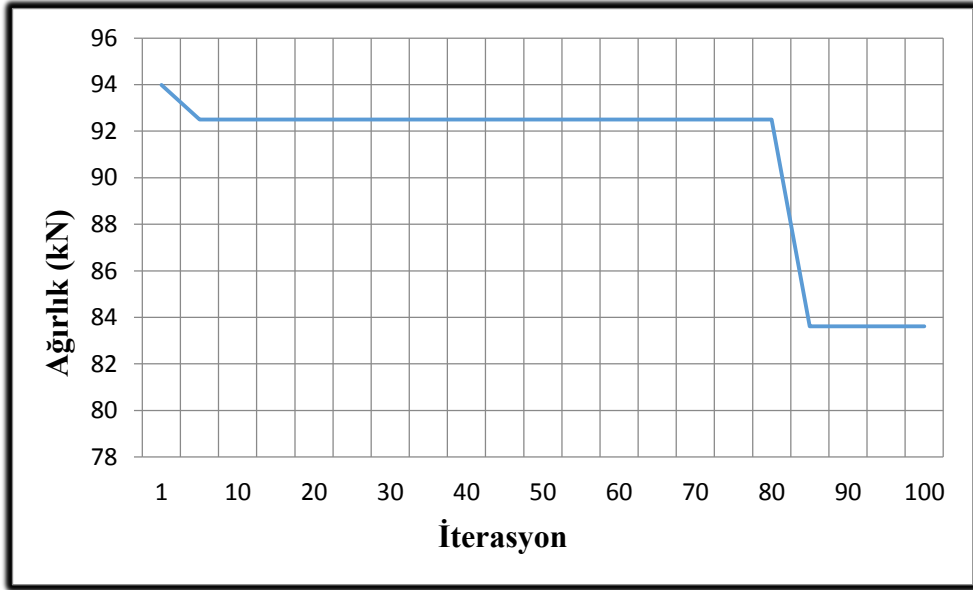
Grup No	Grup Adı	Camp [3]	P.Murren [14]	Bu çalışma (SSO)
1	Kiriş	W24X62	W24X62	W24X62
2	Kolon	W10X60	W10X60	W10X60
Ağırlık (kN)		83,587	83,587	83,587
Maksimum gerilme oranı		*	*	0,963
Maksimum iterasyon sayısı		*	*	100

* Değerler önceki çalışmalarda verilmemiştir

Bu örnekte iki açıklıklı üç katlı çerçeve sosyal örümcek optimizasyon (SSO) algoritması kullanılarak boyutlandırılmıştır. Tasarım sonuçları diğer yöntemlerle karşılaştırılmış ve SSO algoritması diğer yöntemlerle aynı sonuçları vermiştir.



Şekil 5.6: İki açıklıklı üç katlı çerçevenin optimum kesitleri



Şekil 5.7: İki açıklıklı üç katlı çerçevenin ağırlık iterasyon ilişkisi

Örnek 3. Tek Açıklıklı On Katlı Çerçeve

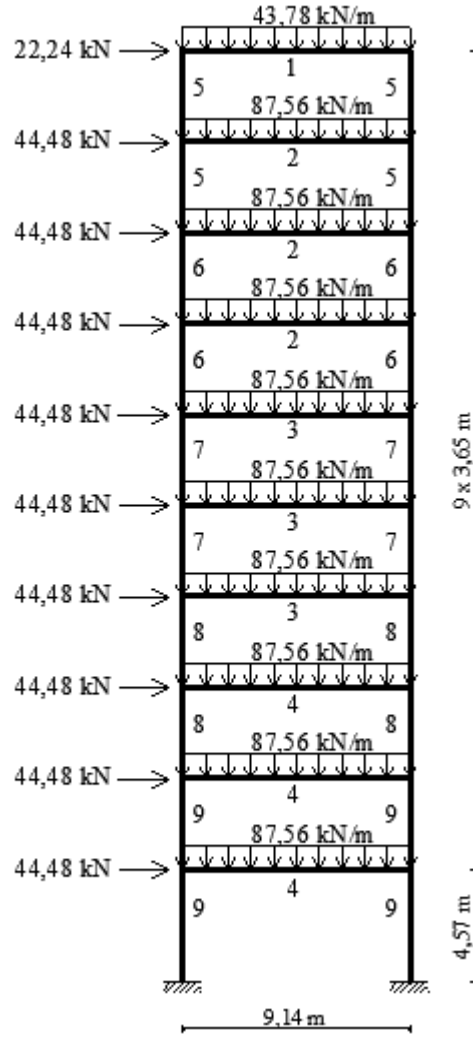
Şekil 5.8’ de tek açıklıklı on katlı düzlem çelik çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları gösterilmiştir.

Çerçevenin mukavemet ve deplasman kısıtlayıcıları Amerikan Çelik Yapılar Enstitüsü – Yük ve Dayanım Faktörü Tasarımı (AISC – LRFD 99) şartnamesinden alınmıştır. Katlar arası yanal öteleme üst sınırı “kat yüksekliği / 300” ve çerçevenin en üst katının yanal öteleme üst sınırı “çerçeve yüksekliği / 300” alınmıştır. Kirişler LRFD’ de yer alan ve Ek A’ da verilen 267 adet W kesitlerinden, kolonlar ise 66 adet W12 ve W14 kesitlerinden seçilmiştir. Kirişlerin etkili boy faktörü 0,2 alınmıştır.

Sistem kolonlar için 5, kirişler için 4 olmak üzere 9 gruptan oluşmaktadır. Buna göre 10. kat kirişi 1. grup, 9, 8 ve 7. kat kirişleri 2. grup, 6, 5 ve 4. kat kirişleri 3. grup, 3, 2 ve 1.kat kirişleri 4.grup, 10. ve 9. kat kolonları 5. grup, 8. ve 7. kat kolonları 6. grup, 6. ve 5. kat kolonları 7. grup, 4. ve 3. kat kolonları 8. grup 2. ve 1.kat kolonları 9. grup olarak belirlenmiştir.

Çerçeve 9,14 m genişliğindedir. 1.kat 4,57 m, diğer 9 kat 3,65 m yüksekliğindedir. Sistem tek tip yüklemeye maruzdur. 10.kat kiriş üzerinde 43,78 kN/m, diğer kat kirişleri üzerinde ise 87,56 kN/m değerinde düzgün yayılı yük bulunmaktadır. X doğrultusunda düğüm noktalarına en üst kat için 22,24 kN, diğer 9 kat için 44,48 kN büyüklüğünde yatay tekil yük etki ettirilmiştir. Mesnetlenme tipi ankastre mesnettir.

Bu düzlem çelik çerçeve daha önce Değertekin tarafından yapay arı kolonisi algoritması (ABC) kullanılarak boyutlandırılmıştır. Bu çalışmada SSO algoritması ile boyutlandırma yapılmış ve sonuçlar tablo 5.4’ te gösterilmiştir.

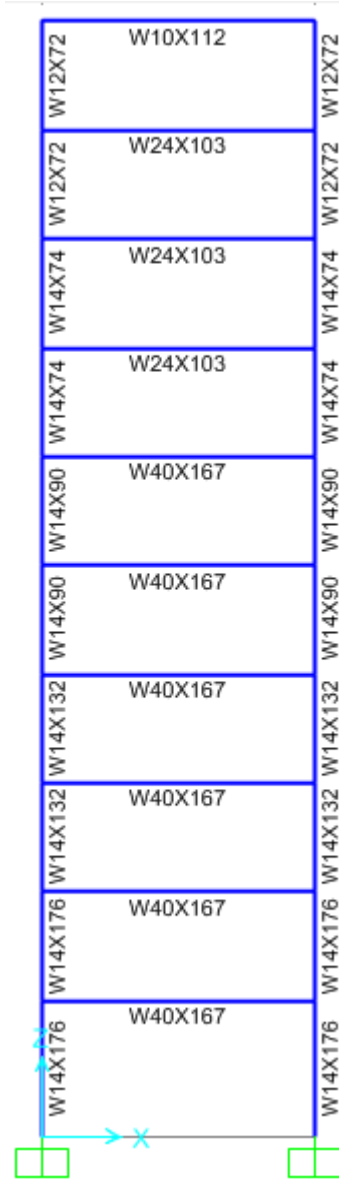


Şekil 5.8: Tek açıklıklı on katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları

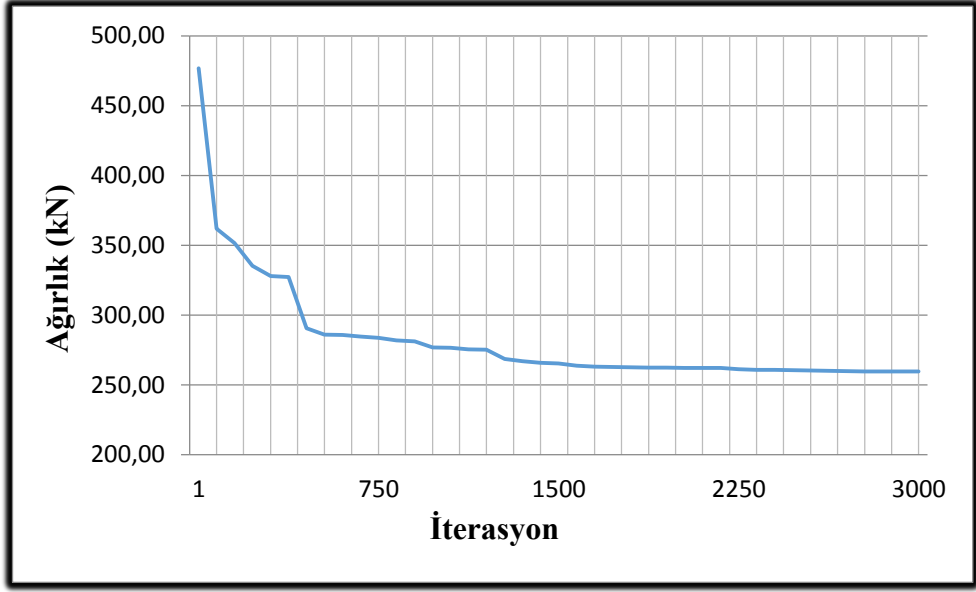
Tablo 5.4: Tek açıklıklı on katlı çerçevenin optimum sonuçları

Grup No	Grup Adı	Değertekin	Değertekin	Bu Çalışma	Bu Çalışma
		(ABC) [9]	(ABC) [9]	(SSO)	(SSO)
		En Kötü	En İyi	En Kötü	En İyi
1	Kiriş	W24x68	W24x68	W21x83	W10x112
2	Kiriş	W27x94	W27x84	W27x102	W24x103
3	Kiriş	W33x118	W33x118	W36x194	W40x167
4	Kiriş	W36x150	W36x150	W40x149	W40x167
5	Kolon	W14x68	W12x58	W14x68	W12x72
6	Kolon	W14x99	W14x99	W14x74	W14x74
7	Kolon	W14x159	W14x145	W14x109	W14x90
8	Kolon	W14x176	W14x176	W14x145	W14x132
9	Kolon	W14x233	W14x233	W14x176	W14x176
Ağırlık (kN)		317,595	308,68	316,22	310,927
Ortalama Ağırlık (kN)		314,153		313,574	
En üst kat yanal ötelenme (cm)		9,54	9,98	9,72	10,29
Maksimum katlar arası yanal ötelenme (cm)		1,23	1,23	1,49	1,44
Maksimum gerilme oranı		0,94	0,98	0,93	0,98
Maksimum iterasyon sayısı		2820	2880	3000	3000

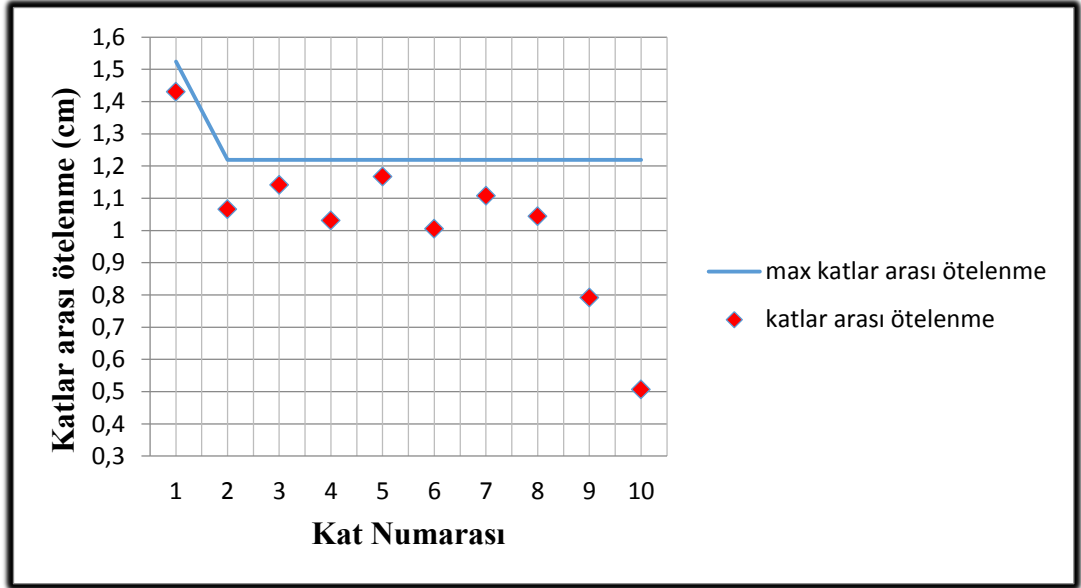
Bu örnekte tek açıklıklı on katlı çerçeve SSO algoritması kullanılarak boyutlandırılmıştır. Bu çerçeve Değertekin tarafından ABC algoritması kullanılarak boyutlandırılmıştır. Tablo 5.4' te hem Değertekin tarafından hem de bu çalışmadaki en iyi ve en kötü sonuçlar gösterilmiştir. Sonuçlar SSO algoritmasının rekabetçi bir algoritma olduğunu göstermektedir. SSO algoritması ile optimum tasarımı yapılan çerçevenin ortalama ağırlığı ABC algoritmasından %0,18 daha az bulunmuştur.



Şekil 5.9: Tek açıklıklı on katlı çerçevenin optimum kesitleri



Şekil 5.10: Tek açıklıklı on katlı çerçevenin ağırlık iterasyon ilişkisi



Şekil 5.11: Tek açıklıklı on katlı çerçevenin x doğrultusundaki yanal ötelenmeleri

Örnek 4. Üç Açıklıklı On beş Katlı Çerçeve

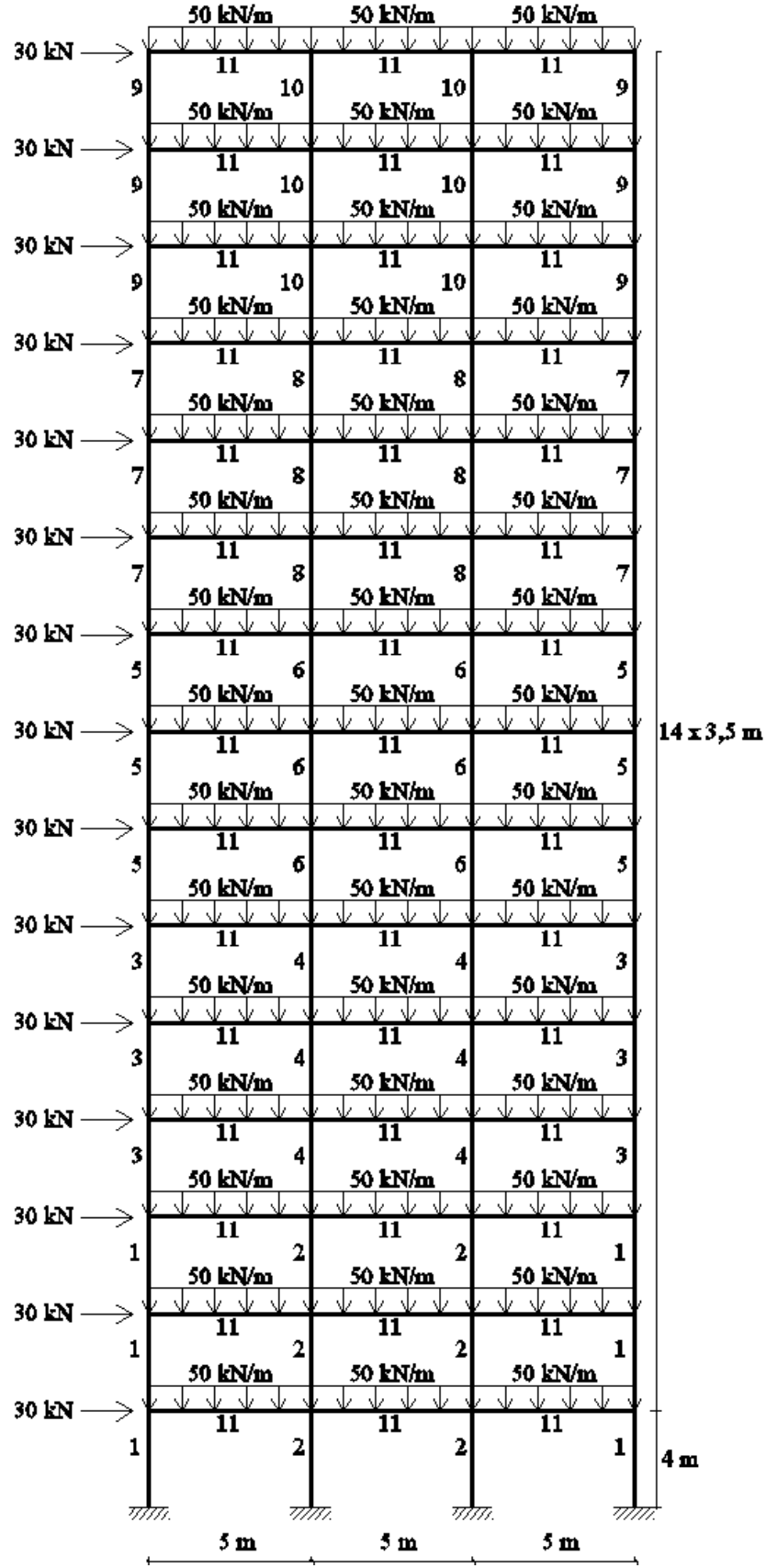
Şekil 5.12’ de üç açıklıklı on beş katlı düzlem çelik çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları gösterilmiştir.

Çerçevenin mukavemet ve deplasman kısıtlayıcıları Amerikan Çelik Yapılar Enstitüsü – Yük ve Dayanım Faktörü Tasarımı (AISC – LRFD 99) şartnamesinden alınmıştır. Çerçevenin en üst katının yanal ötelenmesi 23,5 cm ile sınırlandırılmıştır. Kirişler ve kolonlar LRFD’ de yer alan ve Ek A’ da verilen 267 adet W kesitlerinden seçilmiştir. Kirişlerin etkili boy faktörü 0,2 alınmıştır.

Sistem kolonlar için 10, kirişler için 1 olmak üzere 11 gruptan oluşmaktadır. Dış ve iç kolonlar her üç katta aynı kesit olarak alınmıştır. Çerçevdeki tüm kirişler ise aynı kesit olarak belirlenmiştir.

Çerçevde her bir kiriş açıklığı 5 m’ dir. 1.kat 4,00 m, diğer 14 kat 3,50 m yüksekliğindedir. Sistem tek tip yüklemeye maruzdur. Tüm kirişlerin üzerinde 50 kN/m değerinde düzgün yayılı yük bulunmaktadır. X doğrultusunda düğüm noktalarına 30 kN büyüklüğünde yatay tekil yük etki ettirilmiştir. Mesnetlenme tipi ankastre mesnettir.

Bu düzlem çelik çerçeve daha önce A. Kaveh tarafından parçacık sürüsü optimizasyonu (PSO) algoritması, parçacık sürüsü optimizasyonu pasif toplanma (PSOPC) algoritması, büyük patlama-büyük çatırdama (BB-BC) algoritması ve heuristik parçacık sürüsü karınca kolonisi optimizasyonu (HPSACO) algoritması kullanılarak boyutlandırılmıştır. Bu çalışmada ise SSO algoritması ile boyutlandırma yapılmış ve sonuçlar tablo 5.5’ de gösterilmiştir.



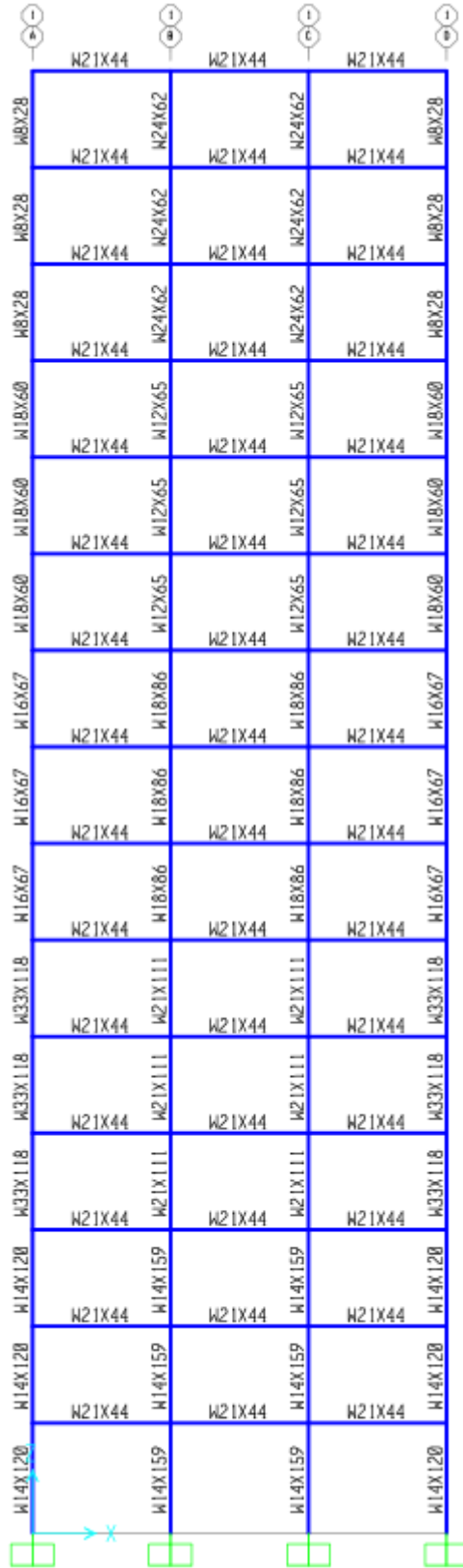
Şekil 5.12: Üç açıklıklı on beş katlı çerçevenin boyutları, yükleme şekli ve elemanların grup numaraları

Tablo 5.5: Üç açıklıklı on beş katlı çerçevenin optimum sonuçları

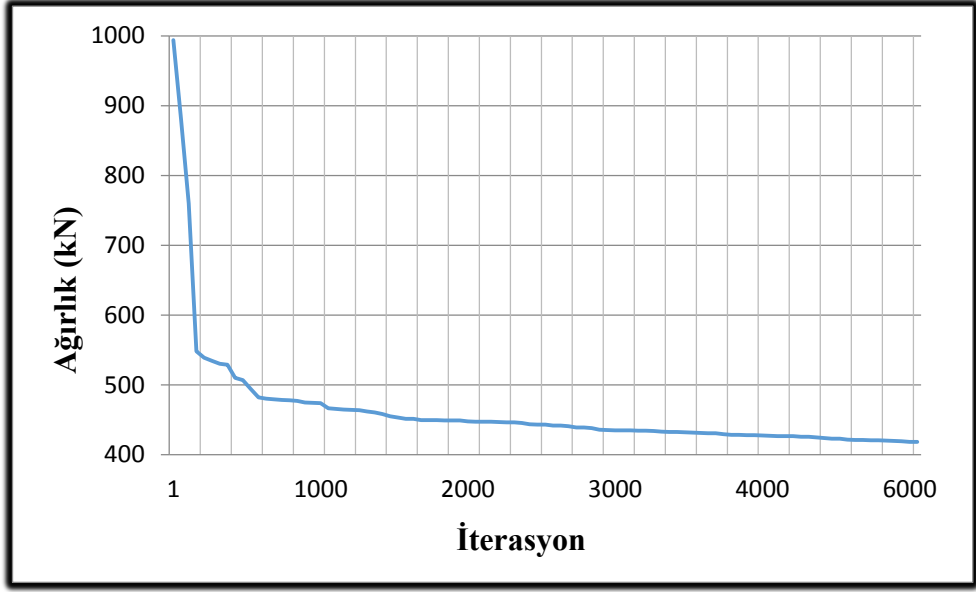
Grup No	Grup Adı	A.Kaveh & S.Talatahari				Bu Çalışma (SSO)
		PSO [25]	PSOPC [27]	BB-BC [26]	HPSACO [27]	
1	Kolon	W33x118	W26x129	W24x117	W21x111	W14x120
2	Kolon	W33x263	W24x131	W21x132	W18x158	W14x159
3	Kolon	W24x76	W24x103	W12x95	W10x88	W33x118
4	Kolon	W36x256	W33x141	W18x119	W30x116	W21x111
5	Kolon	W21x73	W24x104	W21x93	W21x83	W16x67
6	Kolon	W18x86	W10x88	W18x97	W24x103	W18x86
7	Kolon	W18x65	W14x74	W18x76	W21x55	W18x60
8	Kolon	W21x68	W26x94	W18x65	W26x114	W12x65
9	Kolon	W18x60	W21x57	W18x60	W10x33	W8x28
10	Kolon	W18x65	W18x71	W10x39	W18x46	W24x62
11	Kiriş	W21x44	W21x44	W21x48	W21x44	W21x44
Ağırlık (kN)		496,68	452,34	434,54	426,36	418,32
En üst kat Yanal ötelenme (cm)		10,42	11,36	11,63	11,57	14,61
Maksimum gerilme oranı		*	*	*	*	0,986
Maksimum iterasyon		*	*	9500	*	6000

* Değerler önceki çalışmalarda verilmemiştir.

Bu örnekte üç açıklıklı on beş katlı çerçeve SSO algoritması kullanılarak boyutlandırılmış ve sonuçlar tablo 5.5’ de gösterilmiştir. Sonuçlar SSO algoritmasının güçlü bir algoritma olduğunu göstermektedir. SSO algoritması ile optimum tasarımı yapılan çerçevenin ağırlığı PSO algoritmasından %18,73, PSOPC algoritmasından %7,52, BB-BC algoritmasından %3,88, HPSACO algoritmasından %1,92 daha az bulunmuştur.



Şekil 5.13: Üç açıklıklı on beş katlı çerçevenin optimum kesitleri



Şekil 5.14: Üç açıklıklı on beş katlı çerçevenin ağırlık iterasyon ilişkisi

6. SONUÇ VE ÖNERİLER

Bu çalışmada ayırık deęişkenli düzlem çelik çerçevelerin optimum tasarımı için sosyal örümcek optimizasyon (SSO) algoritması geliştirilmiştir. SSO algoritması diři ve erkek örümceklerin doğal hayattaki davranışlarını taklit ederek geliştirilen bir yöntemdir. SSO algoritması visual basic applications (VBA) dilinde kodlanmış ve SAP2000 programı ile entegre edilmiştir. Mukavemet ve deplasman deęerleri SAP2000 programıyla hesaplanmış, çerçevenin yanal ötelenme deęeri ise hazırlanan algoritmada kodlanmıştır.

SSO algoritmasının güçlü bir yöntem olduğunu göstermek için literatürden dört örnek seçilmiştir. Düzlem çelik çerçevelerin optimum tasarımı için SSO algoritması ilk kez bu çalışmada kullanılmasına rağmen sonuçlar algoritmanın güçlü bir yöntem olduğunu ortaya koymuştur. Bu çalışmada optimum tasarımları yapılan çerçeveler için aşağıdaki sonuçlar elde edilmiştir.

1.örnekte iki açıklıklı altı katlı çelik çerçevenin optimum tasarımı SSO algoritması kullanılarak gerçekleştirilmiştir. Analiz sonuçları SSO algoritmasının güçlü bir yöntem olduğunu ortaya koymaktadır.

2.örnekte literatürde çok sık kullanılan iki açıklıklı üç katlı çelik çerçeve kullanılmıştır. Bu çerçevenin optimum tasarımı SSO algoritması kullanılarak gerçekleştirilmiştir. SSO algoritması diğer yöntemlerle aynı sonuçları bulmuştur.

3.örnekte tek açıklıklı on katlı çelik çerçeve kullanılmıştır. Bu çerçevenin optimum tasarımı SSO algoritması kullanılarak gerçekleştirilmiştir. Analiz sonuçları SSO algoritmasının rekabetçi olduğunu göstermektedir.

4. örnekte üç açıklıklı on beş katlı çelik çerçeve kullanılmış ve SSO algoritması ile boyutlandırılmıştır. Sonuçlar diğer yöntemlerle karşılaştırılmış ve SSO algoritmasının çok güçlü bir algoritma olduğu görülmüştür.

Bu sonuçlar düzlem çelik çerçevelerin optimum tasarımında kullanılan SSO algoritmasının güçlü bir yöntem olduğunu göstermektedir. Bu çalışmadaki tecrübelerle göre dişi örümceklerin sayısının fazla olması SSO algoritmasının performansını olumlu yönde etkilemektedir. Ayrıca PF parametresi de 0,1 ve 0,125 değerlerinde daha iyi sonuçlar vermektedir. Farklı tipteki çerçeveler SSO algoritması kullanılarak boyutlandırılabilir.

KAYNAKLAR

- [1] Pezeshk, S., Camp, C. V., and Chen, D., “Design of Nonlinear Framed Structures Using Genetic Optimization,” *J. Struct. Eng.*, vol. 126, no. 3, pp. 382–388, (2000).
- [2] Kameshki, E. S., and Saka, M. P., “Genetic algorithm based optimum bracing design of non-swaying tall plane frames,” *J. Construtional Steel Res.*, vol. 57, pp. 1081–1097, (2001).
- [3] Camp, C. V., Bichon, B. J., and Stovall, S. P., “Design of Steel Frames Using Ant Colony Optimization,” *J. Struct. Eng.*, vol. 131, no. 3, pp. 369–379, Mar. (2005).
- [4] Degertekin, S. O., Saka, M. P., and Hayalioglu, M. S., “Optimal load and resistance factor design of geometrically nonlinear steel space frames via tabu search and genetic algorithm,” *Eng. Struct.*, vol. 30, no. 1, pp. 197–205, Jan. (2008).
- [5] Degertekin, S. O., “Optimum design of steel frames using harmony search algorithm,” *Struct Multidisc Optim.*, vol. 36, pp. 393–401, (2008).
- [6] Saka, M. P., “Optimum design of steel sway frames to BS5950 using harmony search algorithm,” *J. Construtional Steel Res.*, vol. 65, no. 1, pp. 36–43, (2009).
- [7] Kaveh, A., and Talatahari, S., “An improved ant colony optimization for the design of planar steel frames,” *Eng. Struct.*, vol. 32, no. 3, pp. 864–873, (2010).
- [8] Togan, V., “Design of planar steel frames using Teaching – Learning Based Optimization,” *Eng. Struct.*, vol. 34, pp. 225–232, (2012).
- [9] Degertekin, S. O., “Optimum design of geometrically non-linear steel frames using artificial bee colony algorithm,” *Steel Compos. Struct.*, vol. 12, no. 6, pp. 505–522, (2012).
- [10] Saka, M. P., and Dogan, E., “Optimum design of unbraced steel frames to LRFD – AISC using particle swarm optimization,” *Adv. Eng. Softw.*, vol. 46, pp. 27–34, (2012).

- [11] Talatahari, S., Khalili, E., and Alavizadeh, S. M., “Accelerated particle swarm for optimum design of frame structures,” *Math. Probl. Eng.*, vol. 2013, p. 6, (2013).
- [12] Safari, D., Maheri, M. R., and Maheri, A., “On the performance of a modified multiple-deme genetic algorithm in LRFD design of steel frames,” *Trans. Civ. Eng.*, vol. 37, no. C2, pp. 169–190, (2013).
- [13] Maheri, M. R., and Narimani, M. M., “An enhanced harmony search algorithm for optimum design of side sway steel frames,” *Comput. Struct.*, vol. 136, pp. 78–89, (2014).
- [14] Murren, P., and Khandelwal, K., “Design-driven harmony search (DDHS) in steel frame optimization,” *Eng. Struct.*, vol. 59, pp. 798–808, (2014).
- [15] Sevim, Ö., and Sönmez, M., “Geliştirilmiş yapay arı koloni algoritması ile kafes ve düzlemsel çelik yapıların optimum tasarımı,” *Niğde Üniversitesi Mühendislik Bilim. Derg.*, vol. 3, no. 2, pp. 38–51, (2014).
- [16] Aydoğdu, İ., “Betonarme ve çelik yapıların deprem yüklerinde göz önüne alınarak boyutlandırılması,” *Tübitak Projesi 106M490, Araştırmacı*, (2009).
- [17] Degertekin, S. O., and Hayalioglu, M. S., “Modern arama yöntemleriyle üç boyutlu çelik çerçevelerin optimizasyonu,” *Bilim. Mod. Yöntemler Sempozyumu*, (2010).
- [18] Lee, K. S., and Geem, Z. W., “A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice,” *Comput. Methods Appl. Mech. Eng.*, vol. 194, no. 36–38, pp. 3902–3933, Sep. (2005).
- [19] Colomi, A., Dorigo, M., Maniezzo, V., Elettronica, D., and Milano, P., “Distributed Optimization by Ant Colonies,” *Eur. Conf. Artif. Life, Paris*, pp. 134–142.
- [20] Sonmez, M., “Artificial Bee Colony algorithm for optimization of truss structures,” *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2406–2418, Mar. (2011).
- [21] Cuevas, E., and Cienfuegos, M., “A new algorithm inspired in the behavior of the social-spider for constrained optimization,” *Expert Syst. Appl.*, vol. 41, pp. 412–425, (2014).
- [22] Cuevas, E., Cienfuegos, M., Zaldívar, D., and Pérez-cisneros, M., “A swarm optimization algorithm inspired in the behavior of the social-spider,” *Expert Syst. Appl.*, vol. 40, no. 16, pp. 6374–6384, (2013).
- [23] Yu, J. J. Q., Member, S., and Li, V. O. K., “A Social Spider Algorithm for Global Optimization,” pp. 1–16, (2013).
- [24] “American Institute of Steel Construction. Manuel of steel construction: Load and resistance factor design,” vol. I, (1994).

- [25] Kaveh, A., and Talatahari, S., “Hybrid algorithm of harmony search, particle swarm and ant colony for structural design optimization,” *Stud. Comput. Intell.*, vol. 239, pp. 159–198, (2009).
- [26] Kaveh, A., and Talatahari, S., “A discrete big bang-big crunch algorithm for optimal design of skeletal structures,” *Asian J. Civ. Eng.*, vol. 11, no. 1, pp. 103–122, (2010).
- [27] Kaveh, A., and Talatahari, S., “An efficient hybrid algorithm based on harmony search, particle swarm and ant colony strategies for optimal design of structures”, Chapter 5 of the book titled *Harmony search algorithms for structural design*, Springer-Verlag Berlin Heidelberg, (2009.)

EKLER

EKLER

Ek A: 267 Adet W Kesit Listesi (AISC-LRFD)

W4X13	W10X100	W14X61	W16X100	W24X76	W30X124	W36X328
W5X16	W10X112	W14X68	W18X35	W24X84	W30X132	W36X359
W5X19	W12X14	W14X74	W18X40	W24X94	W30X148	W36X393
W6X8.5	W12X16	W14X82	W18X46	W24X103	W30X173	W36X439
W6X9	W12X19	W14X90	W18X50	W24X104	W30X191	W36X527
W6X12	W12X22	W14X99	W18X55	W24X117	W30X211	W36X650
W6X16	W12X26	W14X109	W18X60	W24X131	W30X235	W36X798
W6X15	W12X30	W14X120	W18X65	W24X146	W30X261	W40X149
W6X20	W12X35	W14X132	W18X71	W24X162	W30X292	W40X167
W6X25	W12X40	W14X145	W18X76	W24X176	W30X326	W40X183
W8X10	W12X45	W14X159	W18X86	W24X192	W30X357	W40X211
W8X13	W12X50	W14X176	W18X97	W24X207	W30X391	W40X235
W8X15	W12X53	W14X193	W18X106	W24X229	W33X118	W40X264
W8X18	W12X58	W14X211	W18X119	W24X250	W33X130	W40X278
W8X21	W12X65	W14X233	W18X130	W24X279	W33X141	W40X327
W8X24	W12X72	W14X257	W18X143	W24X306	W33X152	W40X331
W8X28	W12X79	W14X283	W18X158	W24X335	W33X169	W40X392
W8X31	W12X87	W14X311	W18X175	W24X370	W33X201	W40X199
W8X35	W12X96	W14X342	W21X44	W27X84	W33X221	W40X215
W8X40	W12X106	W14X370	W21X50	W27X94	W33X241	W40X249
W8X48	W12X120	W14X398	W21X57	W27X102	W33X263	W40X277
W8X58	W12X136	W14X426	W21X48	W27X114	W33X291	W40X297
W8X67	W12X152	W14X455	W21X55	W27X129	W33X318	W40X324
W10X12	W12X170	W14X500	W21X62	W27X146	W33X354	W40X362
W10X15	W12X190	W14X550	W21X68	W27X161	W33X387	W40X372
W10X17	W12X210	W14X605	W21X73	W27X178	W36X135	W40X397
W10X19	W12X230	W14X665	W21X83	W27X194	W36X150	W40X431
W10X22	W12X252	W14X730	W21X93	W27X217	W36X160	W40X503
W10X26	W12X279	W14X808	W21X101	W27X235	W36X170	W40X593
W10X30	W12X305	W16X26	W21X111	W27X258	W36X182	W44X230
W10X33	W12X336	W16X31	W21X122	W27X281	W36X194	W44X262

Ek A: (devam)

W10X39	W14X22	W16X36	W21X132	W27X307	W36X210	W44X290
W10X45	W14X26	W16X40	W21X147	W27X336	W36X232	W44X335
W10X49	W14X30	W16X45	W21X166	W27X368	W36X256	
W10X54	W14X34	W16X50	W21X182	W27X539	W36X230	
W10X60	W14X38	W16X57	W21X201	W30X90	W36X245	
W10X68	W14X43	W16X67	W24X55	W30X99	W36X260	
W10X77	W14X48	W16X77	W24X62	W30X108	W36X280	
W10X88	W14X53	W16X89	W24X68	W30X116	W36X300	

Ek B: Sosyal Örumcek Optimizasyonu Parametreleri ve Sonuçların Görüntülenmesi

SocialSpideSsp_newdriftf.xlsm - Microsoft Excel (Ünün Etkinleştirilmemiş)

Sosyal Örumcek Algoritma Parametreleri

Problem parametreleri	Değer
Örumcek Sayısı= 50	8
Pf değeri= 0,125	
Maksimum iter= 2000	
Number of Sec List 2	
02/03/15 18:32 → Analiz Başlama	
02/03/15 20:33 → Analiz Bitiş	
Dış Örumcek Sayısı 45	
Erkek Örumcek Sayısı 5	

2 Açıklıklı 6 Katlı Çerçeve

Anlık 71,96253 Optimum 71,96253

Ağırlık Ceza 0

Cezalı Ağırlık 71,96253

Grup No Kesit Liste No

Grup No	Kesit Liste No	W16X57	W16X36	W12X35	W21X83	W16X77	W12X40	W16X31	W14X26
1	2	70	42	40	96	93	46	36	30
2	2	42	42	40	96	93	46	36	30
3	2	40	40	40	96	93	46	36	30
4	2	40	40	40	96	93	46	36	30
5	2	40	40	40	96	93	46	36	30
6	2	40	40	40	96	93	46	36	30
7	1	40	40	40	96	93	46	36	30
8	1	40	40	40	96	93	46	36	30

Not: 1 = Kiriş Kesitleri
2 = Kolon Kesitleri

Profili Sıra Numarası

Kat Yükseldiği Kat Numarası