

ARAŞTIRMA MAKALESİ

Sosyal Örümcek Optimizasyon Algoritması Kullanılarak Düzlem Çelik Çerçevelerin Optimum Boyutlandırılması

Optimum Design of Plane Steel Frame Using the Social Spider Optimization Algorithm

Perihan Efe^{1,*}, Metin Yetkin²

¹İnşaat Mühendisliği Bölümü, Mühendislik Fakültesi, Balıkesir Üniversitesi, İstanbul, Türkiye.

²Ulaştırma Denizcilik ve Haberleşme Bakanlığı, Ankara, Türkiye.

Geliş / Received: 24.11.2024

Kabul / Accepted: 20.02.2025

*Sorumlu Yazar: Perihan Efe pefe@balikesir.edu.tr

ÖZ: Günümüzde doğa olaylarını taklit ederek geliştirilen algoritmalar araştırmacılar tarafından yoğun ilgi görmektedir. Bu tür algoritmalar birçok mühendislik alanlarında yaygın olarak kullanılmaktadır. Özellikle de yapı mühendisliğinde çelik yapıların optimum tasarımı konusunda bu tür algoritmalar kullanılarak yapılan pek çok çalışma mevcuttur. Bu çalışmada da, düzlem çelik çerçevelerin minimum ağırlıklı optimum tasarımını gerçekleştirmek için Sosyal Örümcek Optimizasyon (SSO) algoritması geliştirilmiştir. Sosyal Örümcek Optimizasyon algoritması dişi ve erkek örümceklerin doğadaki davranışları göz önüne alınarak geliştirilen bir yöntemdir. Geliştirilen bu algoritma visual basic applications (VBA) programlama dilinde yazılmış ve SAP2000 programıyla entegrasyonu sağlanmıştır. Düzlem çelik çerçevelerin optimum tasarımında Amerikan Çelik Yapılar Enstitüsü-Yük ve Dayanım Faktörü Tasarımı (AISC-LRFD) şartnamesi kullanılmıştır. Optimum tasarımda deplasman, göreceli kat ötelemesi, mukavemet kısıtlayıcıları ve ayrıca uygulanabilirlik açısından kesit boyut kısıtlayıcıları da dikkate alınmıştır. Çalışmada, algoritmanın performansını test etmek için literatürden seçilen ve farklı yöntemlerle optimum tasarımı gerçekleştirilen düzlem çelik çerçeve örnekleri göz önüne alınmış ve elde edilen sonuçlar karşılaştırılmıştır. Tasarım sonuçları SSO algoritmasının güçlü ve kullanılabilir bir algoritma olduğunu göstermiştir.

Anahtar Kelimeler: Optimizasyon, Optimum tasarım, Düzlem çelik çerçeve, Sosyal örümcek optimizasyonu, Yapı dizaynı

ABSTRACT: Algorithms developed by mimicking natural phenomena have attracted significant attention from researchers in recent years. These types of algorithms are widely used in many engineering. In particular, there are numerous studies in structural engineering where such algorithms are applied to the optimal design of steel structures. In this study, a Social Spider Optimization (SSO) algorithm has been developed to achieve the minimum weight optimal design of plane steel frames. Social spider optimization (SSO) algorithm is a method developed by imitating the behavior of male or female spiders in natural life. The developed SSO algorithm is coded at visual basic application (VBA) programming 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). The parameters as displacement, interstory drift and strength are considered at optimum design. Additionally, dimensions of sections are regarded for adaptedness. In the study, to test the effectiveness of the algorithm, plane steel frame examples selected from the literature and optimally designed with different methods and the results have been compared. These comparisons clearly demonstrate that the SSO is an effective and applicable algorithm for structural design problems.

Keywords: Optimization, Optimum Design, Plane Steel Frame, Social Spider Optimization, Structural Design

1. GİRİŞ

Yapı mühendislerinin en büyük amacı güvenlik şartlarından ödün vermeksizin ve en ekonomik olacak şekilde yapıları boyutlandırmaktır. Son zamanlarda Optimizasyon Yöntemleri bu amaca yönelik olarak çok büyük önem kazanmıştır. Optimizasyon, çok eskiden beri uygulanmakla birlikte, özellikle günümüzde bilgisayar teknolojileri ve yapay zekâ algoritmalarındaki gelişmeler sayesinde çok daha hızlı ve daha verimli çözümler sunmaktadır. Yapı mühendisliği gibi birçok disiplin de bu gelişmelerden faydalanmakta ve optimizasyon tekniklerinin sunduğu olanaklardan yararlanmaktadır [1]. Özellikle de çelik yapıların optimum tasarımında evrimsel optimizasyon teknikleri yaygın olarak kullanılmaktadır [2-6]. Evrimsel optimizasyon yöntemleri doğada gerçekleşen olayları taklit ederek geliştirilen yöntemlerdir. Doğada gerçekleşen bu olaylar bilgisayarda kodlanmak suretiyle problemin çözümüne ulaşılmaya çalışılmaktadır.

Çelik yapıların optimizasyonu konusunda farklı algoritmalar kullanılarak yapılan pek çok çalışma mevcuttur. [7] nolu çalışmada AISC (American Institute of Steel Construction) yönetmeliğinde belirtilen gerilme ve yer değiştirme kısıtlamaları dikkate alınarak, matematiksel bir algoritma ile çelik çerçeve sistemlerin boyutlandırılması yapılmıştır. Bu çalışma, amaç fonksiyonu ve kısıtların tanımlanarak algoritmalarla entegrasyonu konusunda birçok araştırmacıya rehberlik etmiştir. [8] nolu çalışmada TS648 standardı temel alınarak, düzlem çelik çerçeve sistemlerin boyutlandırılması genetik algoritma kullanılarak gerçekleştirilmiştir. [9] nolu çalışmada genetik algoritma kullanılarak düzlem kafes sistemler optimize edilmiştir. [10] nolu çalışmada Yarı rijit bağlantılara sahip çerçeveler için $P\Delta$ etkisi göz önünde bulundurularak, doğrusal olmayan bir yöntemle boyut optimizasyonu gerçekleştirilmiştir. [11] nolu çalışmada yarı rijit birleşimli doğrusal olmayan çelik çerçeveler genetik algoritma ile optimize edilmiştir. [12] nolu çalışmada, değişik algoritmalar kullanılarak düzlem çelik çerçevelerin optimizasyonu gerçekleştirilmiştir. [13] nolu çalışmada BS-5950 yönetmeliği kullanılarak Harmoni Arama Algoritması ile düzlem çelik çerçevelerin optimizasyonu gerçekleştirilmiştir.

Bu çalışmada da düzlem çelik çerçevelerin minimum ağırlıklı olacak şekilde boyutlandırılmaları için bir optimizasyon algoritması geliştirilmiştir. Bu amaçla yapı analizinde SAP2000 programı kullanılmış ve visual basic applications (VBA) dilinde kodlanan ve örümceklerin doğadaki davranışlarından esinlenerek, onların davranışlarını taklit eden Sosyal Örümcek Optimizasyon (SSO) algoritması geliştirilmiştir. Ağırlık optimizasyonu yapılacak düzlem çelik çerçevelerin tasarım değişkenleri için mevcut olan hazır profiller kullanılmıştır. Boyutlandırma kısıtlamaları olarak, mukavemet, deplasman ve yanal ötelenme değerleri dikkate alınmıştır. Mukavemet ve deplasman kısıtlamaları için Amerikan Çelik Yapılar Enstitüsü - Yük ve Dayanım Faktörü Tasarımı (AISC - LRFD) standardı referans alınmıştır [14]. Yanal ötelenme kısıtlayıcıları Visual Basic Applications (VBA) program dilinde kodlanmıştır. Katlar arası yanal öteleme üst sınırı “kat yüksekliği / 300”, çerçevenin en üst katının maksimum yanal ötelenme değeri ise “çerçeve yüksekliği / 300” alınmıştır.

Geliştirilen SSO algoritmasının etkinliğini göstermek amacıyla, literatürden seçilen iki örnek üzerinde sonuçlar karşılaştırılmış ve SSO algoritmasının güçlü ve kullanılabilir bir optimizasyon yöntemi olduğu gösterilmiştir.

2. SOSYAL ÖRÜMCEK OPTİMİZASYON ALGORİTMASI

Sosyal Örümcek Algoritması (SSO), son yıllarda geliştirilmiş sürü temelli bir algoritmadır [15]. Doğada birlikte yaşayan örümcek türlerinin davranışlarından esinlenerek geliştirilen bir optimizasyon yöntemidir. Doğada ortak örümcek ağını kullanan dişi ve erkek örümceklerden oluşan bir örümcek kolonisi bulunur. Kolonideki her bir örümceğin cinsiyetine bağlı olarak farklı bir görevi vardır. Örümcekler bu görevleri yerine getirmek için hareket, titreşim ve çiftleşme adı verilen operatörleri kullanırlar. Örümcekler birbirleriyle iletişim kurmak için titreşim oluştururlar. Titreşimin şiddeti örümceklerin performansına ve birbirlerine olan uzaklığına bağlıdır. Titreşimin etkisiyle örümceklerin birbirlerine yaklaşması veya uzaklaşması, onların hareket operatörünü tanımlar. Örümcekler titreşimin etkisiyle farklı konumlara hareket ederler [16, 17]. Kolonideki erkek örümcekler performans değerlerine bağlı olarak

baskın ve baskın olmayan erkek örümcekler olarak sınıflandırılmaktadır. Örümceklerin titreşim etkisiyle hareket ettikten sonra, eğer baskın bir erkek örümcek, çiftleşme etki alanında bir dişi örümcek bulursa, çiftleşme operatörü devreye girer. Bu çiftleşme sonucunda yeni bir örümcek dünyaya gelir. Yeni doğan örümceğin koloniyeye katılıp katılmaması, performansına bağlıdır. Eğer bu örümceğin performansı, başlangıçtaki kolonide yer alan en düşük performansa sahip örümceğin performansından daha iyiye, koloniyeye dahil edilir ve en kötü performansa sahip örümcek koloni dışına çıkarılır. Böylece koloni güncellenmiş olur. Yeni doğan örümceğin cinsiyeti ise, koloniden çıkarılan örümceğin cinsiyeti ile aynı olduğu kabul edilir. Böylece kolonideki başlangıçtaki dişi ve erkek örümceklerin sayısı korunmuş olur. Yeni koloni oluşturma işlemi, önceden belirlenmiş bir sayıda tekrarlanır. Nihai olarak, güncellenen koloni içinde en yüksek performansa sahip örümcek, optimizasyon probleminin en iyi çözümü olarak kabul edilir.

Sosyal Örümcek Optimizasyon algoritması başlangıçta örümcek sayısını belirleyerek işe başlar. Bu algoritmaya göre dişi örümceklerin sayısı, kolonideki toplam örümcek sayısının %65 ile %90'ı arasında değişmektedir [3, 4]. Örümcek sayısı belirlendikten sonra dişi ve erkek örümceklerin sayısı "1" ve "2" denklemleri ile hesaplanır;

$$N_f = N(0.90 - \text{rand } 0.25) \quad (1)$$

$$N_m = N - N_f \quad (2)$$

Denklemlerde, "N" kolonideki toplam örümcek sayısını, "N_f" dişi örümcek sayısını, "N_m" erkek örümcek sayısını, "rand" değeri ise 0 ile 1 arasında rastgele belirlenen bir sayıyı göstermektedir. Algoritma çalışmaya başladıktan sonra örümceklerin başlangıç pozisyonları belirlenir,

$$f_{ij}^0 = p_j^{\text{low}} + \text{rand}(0.1) (p_j^{\text{high}} - p_j^{\text{low}}) \quad (3)$$

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

Denklemden, "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 küçük değerini, " p_j^{high} " j. tasarım değişkeninin en büyük değerini göstermektedir.

$$m_{kj}^0 = p_j^{\text{low}} + \text{rand}(0.1) (p_j^{\text{high}} - p_j^{\text{low}}) \quad (4)$$

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

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

Sosyal Örümcek Optimizasyon algoritması tüm işlemleri üç esas operatör kullanarak yapar.

2.1 Titreşim operatörü

Koloni üyeleri, birbirlerine bilgi aktarımı için titreşimler gönderirler. Bu titreşimlerin büyüklüğü, örümceklerin performansına ve aralarındaki mesafeye bağlıdır. Birbirine yakın olan örümcekler, uzak olanlara göre daha güçlü titreşimler oluşturur. Bu titreşimler sayesinde örümcekler birbirlerini harekete geçirirler [16, 17]. Ayrıca, bir örümcek, öncekinden farklı bir pozisyona geçtiğinde yeni bir titreşim meydana getirir [18].

Sosyal Örümcek Optimizasyonunda titreşimin şiddeti "5" denklemi ile hesaplanır.

$$\text{Vib}_{ij} = w_j \cdot e^{-d_{ij}^2} \quad (5)$$

Denklemden "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. Bu mesafe "6" denklemi ile hesaplanır;

$$d_{ij} = \frac{\sqrt{\sum_{j=1}^n (S_{1j} - S_{2j})^2}}{d_{\text{max}}} \quad (6)$$

Denklemden, "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 uzun mesafe olup aşağıdaki gibi hesaplanmaktadır;

$$d_{\text{max}} = \sqrt{\sum_{j=1}^n (p_j^{\text{high}} - p_j^{\text{low}})^2} \quad (7)$$

Denklemden, p_j^{high} j. tasarım değişkeninin en yüksek değeri, p_j^{low} ise j. tasarım değişkeninin en düşük değeridir.

Performans değeri örümceğin optimizasyon probleminin çözümü olabilme potansiyelini gösterir ve şu şekilde hesaplanır;

$$w_j = \frac{1}{W_j} \quad (8)$$

Burada " w_j " j. örümceğin performans değeri, " W_j " ise j. örümceğin amaç fonksiyonudur.

Sosyal örümcek optimizasyonu dişi ve erkek örümcekler için farklı titreşim modelleri belirlemektedir. Bu modeller aşağıda verilmektedir.

2.1.1 Dişi örümcekler için titreşim modeli

2.1.1.1 "Vibc_i" Titreşim modeli

Bu modellemelerde performansı en yüksek olan dişi örümcek dikkate alınmaksızın, diğer dişi örümceklerin birbirlerine titreşim gönderdikleri kabul edilir. Her bir dişi örümcek ile bu örümceğe en yakın pozisyonda olan diğer dişi örümcek arasında titreşim şiddeti hesaplanır. Bu titreşim modelinde diğerlerinden farklı olarak titreşim şiddeti hesaplanırken performansı en iyi olan örümceğin performans değeri kullanılır [16, 17].

$$Vibc_i = w_c \cdot e^{-d^2_{ic}} \quad (9)$$

2.1.1.2 "Vibb_i" Titreşim modeli

$$f_{ij}^{k+1} = \begin{cases} f_{ij}^k + \alpha \cdot Vibc_i(s_c - f_{ij}^k) + \beta \cdot Vibb_i(s_b - f_{ij}^k) + \delta \left(\text{rand} - \frac{1}{2} \right) & 0 < \text{rand} \leq PF \\ f_{ij}^k - \alpha \cdot Vibc_i(s_c - f_{ij}^k) - \beta \cdot Vibb_i(s_b - f_{ij}^k) + \delta \left(\text{rand} - \frac{1}{2} \right) PF & PF < \text{rand} < 1 \end{cases} \quad (12)$$

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 ", $Vibc_i$ titreşim modelinde titreşimden etkilenen dişi örümceğin j. tasarım değişkenini, " s_b " ise $Vibb_i$ titreşim modelinde titreşimden etkilenen dişi örümceğin j. tasarım değişkenini göstermektedir

Bu modellemelerde kolonideki en yüksek performanslı dişi örümcek ile diğer dişi örümcekler arasında titreşim şiddeti hesaplanır.

$$Vibb_i = w_b \cdot e^{-d^2_{ib}} \quad (10)$$

2.1.2 Erkek örümcekler için Vibf_i titreşim modeli

Bu modellemelerde kolonide bulunan her bir erkek örümcek ile bu örümceğe en yakın olan dişi örümcek arasında titreşim şiddeti hesaplanır.

$$Vibf_i = w_f \cdot e^{-d^2_{if}} \quad (11)$$

2.2 Hareket operatörleri

2.2.1 Dişi örümceklerin hareket operatörü

Dişi örümcekler belirli bir çekiciliğe sahiptir ve cinsiyetleri ne olursa olsun diğer örümcekler üzerinde etkili olurlar. Dişi örümceklerin iş birliği davranışını simüle etmek için bir hareket operatörü tanımlanmıştır. Bu operatör, her dişi örümceğin konumunu dikkate alarak her "i." örümcek için tekrarlanır. Dişi örümcekler, yaydıkları titreşimler sayesinde diğer örümcekleri kendilerine çekebilir ya da uzaklaştırabilir. Örümceklerin birbirine yaklaşmasını veya uzaklaşmasını belirlemek amacıyla "PF" adlı bir parametre kullanılır. Örümcekler, bu parametreye dayanarak hareket ederler. "PF" değeri 0 ile 1 arasında değişen bir değer olup, problemin çözümüne geçmeden önce tanımlanması gerekmektedir [16].

2.2.2 Erkek örümceklerin hareket operatörü

Sosyal Örümcek Optimizasyonunda erkek örümceklerin hareket operatörleri baskın ve baskın olmayan örümcekler için farklı şekilde hesaplanmaktadır.

$$m_{ij}^{k+1} = \begin{cases} m_{ij}^k + \alpha \cdot \text{Vibf}_i(s_f - m_{ij}^k) + \delta(\text{rand} - \frac{1}{2}) & \text{if } w_{Nf+i} > w_{Nf+m} \\ m_{ij}^k - \alpha \cdot \left(\frac{\sum_{h=1}^{N_m} m_h^k \cdot w_{Nf+h}}{\sum_{h=1}^{N_m} w_{Nf+h}} - m_{ij}^k \right) & \text{if } w_{Nf+i} \leq w_{Nf+m} \end{cases} \quad (13)$$

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 " Vibf_i titreşim modelinde titreşimden etkilenen dişi örümceğin j. tasarım değişkenini, " w_{Nf+i} " i. erkek örümceğin performansını, " w_{Nf+m} " ortanca performanslı erkek örümceği ifade etmektedir.

2.3 Çiftleşme operatörleri

Sosyal örümcek kolonisinde çiftleşme, baskın erkek örümcekler ile dişi örümcekler arasında gerçekleşir. Şayet baskın karakterli erkek bir örümceğin etki alanında en az bir dişi örümcek bulunuyorsa, çiftleşme operatörü devreye girerek yeni bir örümcek dünyaya gelir. Yeni doğan örümceğin performans değeri hesaplanır ve yeni doğan örümceğin performansı ile başlangıçtaki en düşük performansa sahip örümceğin performansları karşılaştırılır. Eğer yeni doğan örümceğin performansı daha iyiye, en düşük performansa sahip örümcek koloniden çıkarılarak, onun yerine yeni doğan örümcek koloniye eklenir. Yeni örümceğin cinsiyeti, çıkarılan örümcekle aynı kabul edilir, böylece başlangıçtaki erkek ve dişi örümceklerin sayısı korunmuş olur. Baskın erkek örümceğin etki alanında birden fazla dişi örümcek bulunuyorsa, çiftleşme operatörü rulet yöntemiyle uygulanır.

Örümceklerin çiftleşme etki alanı "14" denklemi ile hesaplanır;

$$r = \frac{\sum_{j=1}^n (p_j^{\text{high}} - p_j^{\text{low}})}{2n} \quad (14)$$

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.

3. DÜZLEM ÇELİK ÇERÇEVELERİN OPTİMUM TASARIMI

Düzlem çelik çerçevelerin optimum boyutlandırma problemi LRFD-AISC Amerikan Şartnamesine göre "15" denklemi ile tanımlanmıştır.

Minimize edilecek fonksiyon:

$$W = \sum_{r=1}^{ng} m_r \sum_{s=1}^{tr} l_s \quad (15)$$

Burada " W " çerçevenin toplam ağırlığını göstermektedir. " m_r ", r çubuk grubu için "Standart Çelik Profilleri Tablosundan" seçilen profilin birim boyunun ağırlığıdır. " tr "; r grubundaki toplam çubuk sayısını, " ng " ise çerçevedeki toplam grup sayısını göstermektedir. " l_s " r grubuna ait s çubuğunun boyunun uzunluğunu ifade etmektedir.

Optimum tasarımda kullanılan kısıtlayıcı fonksiyonlar aşağıda verilmiştir.

Katlar arası yanal öteleme kısıtlayıcısı:

$$k_j = \frac{(\delta_j - \delta_{j-1})}{\delta_{ju}} - 1 \leq 0 \quad j = 1, \dots, ns \quad (16)$$

Denklemden, " δ_j " ve " δ_{j-1} " ardışık iki katın yanal öteleme değerleridir. " δ_{ju} " ise izin verilen yanal öteleme üst sınırı olup değeri "kat yüksekliği/300" olarak alınmıştır. " ns " çerçevedeki toplam kat sayıdır.

Maksimum katlar arası yanal öteleme kısıtlayıcısı:

$$k_t = \frac{\delta_t}{\delta_{tu}} - 1 \leq 0 \quad (17)$$

Denklemden, " δ_t " en üst katın maksimum yanal öteleme değeri, " δ_{tu} " ise izin verilen yanal öteleme üst sınırı olup değeri "çerçeve yüksekliği/300" olarak alınmıştır.

Mukavemet kısıtlayıcıları:

$$\frac{P_{uk}}{\phi P_{nk}} \geq 0,2 \text{ ise } k_k = \frac{P_{uk}}{\phi P_{nk}} + \frac{8}{9} \left(\frac{M_{uxk}}{\phi_b M_{nxxk}} \right) - 1 \leq 0 \quad k = 1, \dots, nc \quad (18)$$

$$\frac{P_{uk}}{\phi P_{nk}} < 0,2 \text{ ise } k_k = \frac{P_{uk}}{2\phi P_{nk}} + \frac{8}{9} \left(\frac{M_{uxk}}{\phi_b M_{nxxk}} \right) - 1 \leq 0 \quad k = 1, \dots, nc \quad (19)$$

(18) ve (19) no'lu denklemler LRFD-AISC şartnamesinden alınan mukavemet kısıtlayıcılarını göstermektedir. Her bir eleman eksenel kuvvet ve eğilme momentinin etkisi altındadır.

" P_{uk} " gereken kuvvet (çekme veya basınç), " P_{nk} " nominal kuvvet (çekme veya basınç), " ϕ " dayanım azaltma katsayısı olup, çekme durumunda " ϕ_t " olarak gösterilir ve değeri 0,90, basınç durumunda " ϕ_c " olarak gösterilir ve değeri 0,85 alınır. " ϕ_b " eğilme için dayanım azaltma katsayısı olup değeri 0,90 'dır. " M_{uxk} " ikinci mertebeden etkileri içeren kuvvetli eksenindeki eğilme dayanımı, " M_{nxxk} " kuvvetli eksenindeki nominal eğilme dayanımıdır.

Kesit boyut kısıtlayıcıları:

$$k_b = \frac{B_{sc}}{B_{sb}} - 1 \leq 0 \quad b = 1, \dots, nu \quad (20)$$

$$k_d = \frac{D_s}{D_{s-1}} - 1 \leq 0 \quad d = 1, \dots, nu \quad (21)$$

$$k_m = \frac{m_s}{m_{s-1}} - 1 \leq 0 \quad m = 1, \dots, nu \quad (22)$$

Denklemlerde " B_{sc} ", çerçevenin s. katındaki kolonun profil başlık genişliği, " B_{sb} " aynı kattaki kolona bağlanan kirişin profil başlık genişliği, " D_s " s. kattaki kolonun profil yüksekliği, " D_{s-1} " (s-1). kattaki kolonun profil yüksekliği, " m_s " s. kattaki kolonun profil birim boy ağırlığı, " m_{s-1} " (s-1). kattaki kolonun profil birim boy ağırlığı olarak ifade edilmektedir.

Cezalandırılmış amaç fonksiyonu şu şekilde tanımlanmıştır:

$$W_p = W (1 + P)^2 \quad (23)$$

$$P = \sum_{i=1}^m p_i \quad (24)$$

Denklem (24)' deki " p_i " ihlal etme katsayısı olup aşağıdaki gibi hesaplanır:

$$\begin{aligned} \text{Eğer } k_i > 0 \text{ ise } p_i &= k_i \\ \text{Eğer } k_i \leq 0 \text{ ise } p_i &= 0 \end{aligned}$$

4. SAYISAL UYGULAMA

Bu bölümde Sosyal Örumcek Optimizasyon (SSO) algoritmasının etkinliğini göstermek amacıyla literatürde mevcut olan ve değişik optimizasyon yöntemleri ile optimum tasarımları yapılmış düzlem çelik çerçeveler göz önüne alınmıştır [3, 18-20]. Bu çalışmada da bu çerçevelerden farklı kat ve açıklığa sahip iki adet düzlem çelik çerçeve incelenmiştir. Bu çerçeveler için SSO algoritması, Visual Basic Applications (VBA) programlama dili kullanılarak geliştirilmiştir. Oluşturulan program kodları, SAP2000 yazılımını çağırarak problem için belirlenen kesit özelliklerini ve sistem geometrisine dair bilgileri alır. Böylece yapının deplasman ve mukavemet değerleri hesaplanır. Yanal ötelenme sınırları ise VBA dili ile kodlanmıştır. Ayrıca görelî kat öteleme değerlerinin sağlanıp sağlanmadığı algoritma tarafından kontrol edilmektedir. Tablo 1 de sayısal uygulamada göz önüne alınan düzlem çelik çerçeveler için SSO parametreleri verilmektedir.

Tablo 1: Sayısal uygulamada kullanılan düzlem çelik çerçeveler için SSO parametreleri [22]

Göz önüne alınan düzlem çelik çerçeve	Örumcek Sayısı	PF	Max İterasyon
2 Açıklıklı 6 Katlı Çerçeve	50	0,125	2000
3 Açıklıklı 15 Katlı Çerçeve	50	0,100	6000

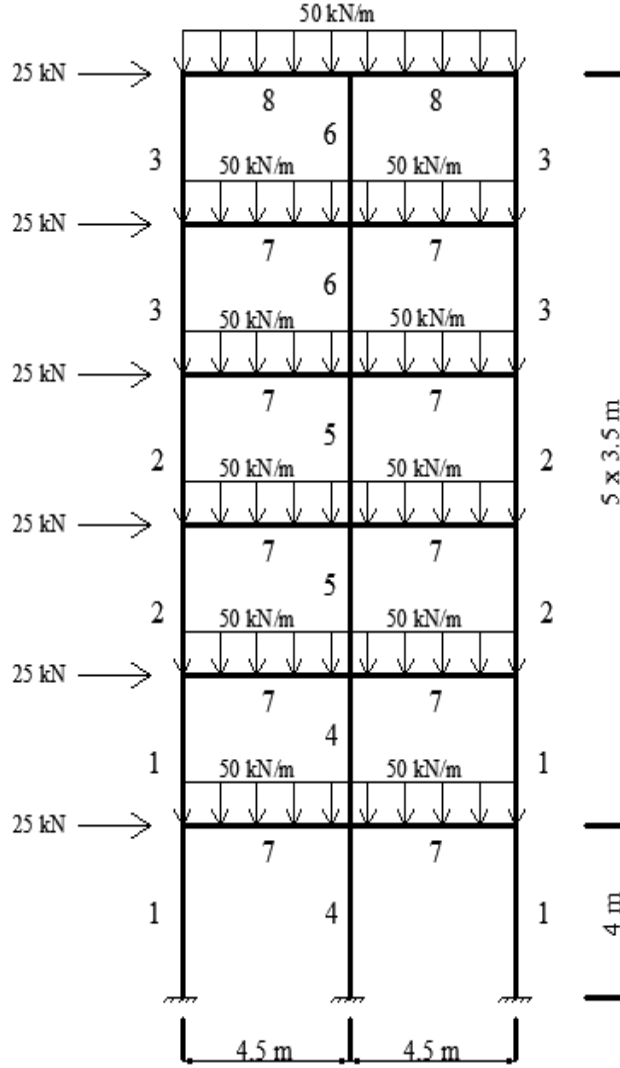
4.1 İki açıklıklı altı katlı çerçeve

Şekil 1' de otuz elemandan oluşan iki açıklıklı ve altı katlı düzlem çelik çerçevenin boyutları, yükleme durumu ve elemanların grup numaraları gösterilmektedir.

Çerçevenin mukavemet ve deplasman kısıtlayıcıları (AISC-LRFD) şartnamesinden alınmıştır. Katlar arası yanal öteleme üst sınırı ise

“kat yüksekliği/300” olarak sınırlandırılmıştır. Kullanılan malzemenin Elastisite Modülü 200000 N/mm², Akma Dayanımı 248,2 N/mm² dir. Kiriş ve kolon elemanları AISC-LRFD’ de yer alan standart 267 adet “W” kesitlerinden seçilmiştir.

Referans [4] de bu düzlem çelik çerçeve Parçacık Sürüsü Optimizasyonu (PSO) ve Harmoni Arama (HS) algoritması kullanılarak boyutlandırılmıştır. Bu çalışmada ise aynı çerçeve Sosyal Örümcek Optimizasyon algoritması kullanılarak optimize edilmiş ve sonuçlar Tablo 2’ de gösterilmiştir.



Şekil 1: İki açıklıklı altı katlı çerçevenin boyutları, yükleme durumu ve elemanların grup numaraları [22]

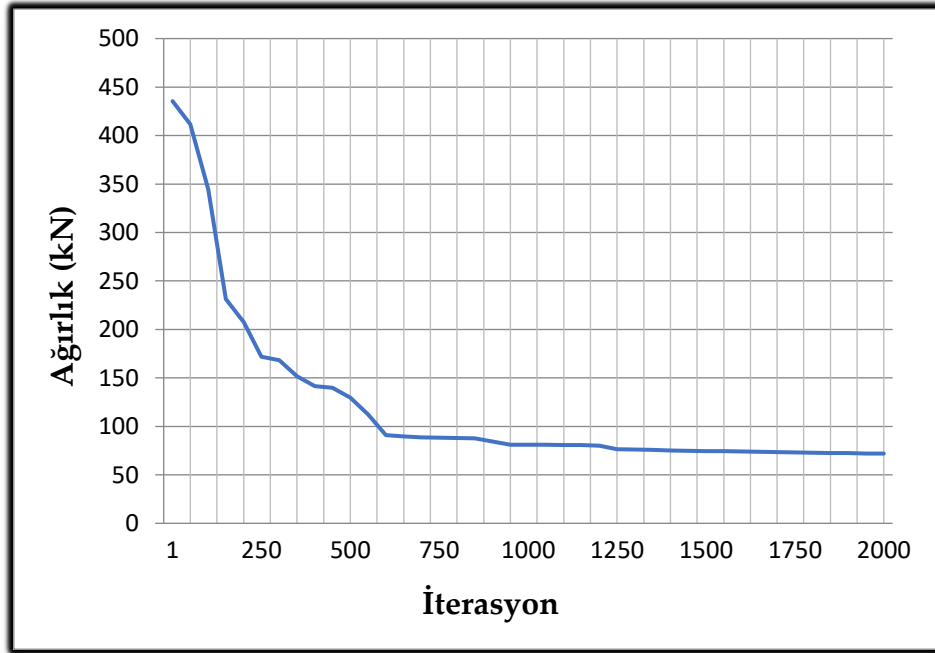
Şekil 1’de verilen iki açıklıklı altı katlı çerçevenin optimum tasarımı Sosyal Örümcek Optimizasyon (SSO) algoritması kullanılarak yapılmıştır [22]. Optimum tasarım sonuçları literatürde mevcut olan Parçacık Sürüsü Optimizasyonu (PSO) ve Harmoni Arama (HS) algoritması adı verilen optimizasyon yöntemleriyle yapılan çözümlerden elde edilen sonuçlarla karşılaştırılmıştır. Tablo 2’de görüldüğü üzere göz önüne alınan çerçevenin

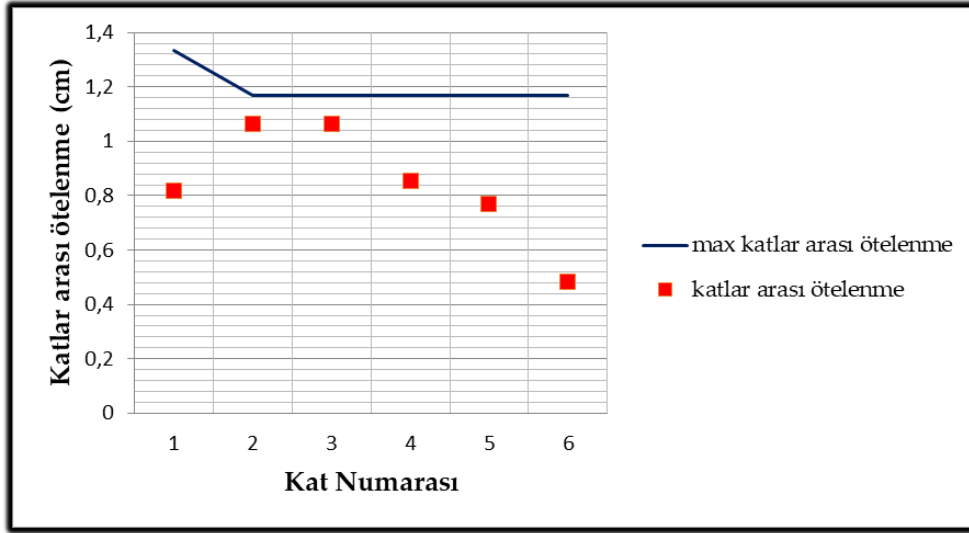
diğer çalışmalara nazaran daha az iterasyonla optimum boyutlandırılması yapılmış ve çerçevenin toplam ağırlığının [4] nolu referansta verilen Harmoni Arama (HS) algoritmasından %8,80, Parçacık Sürüsü Optimizasyonu (PSO) algoritmasından ise %4,68 daha hafif olduğu gözlenmiştir. Çerçeveye ait ağırlık iterasyon ilişkisi Şekil 2 de, katlar arası yanal ötelenmeler ise Şekil 3’de gösterilmektedir.

Tablo 2: İki açıklıklı altı katlı çerçevenin SSO algoritmasına göre optimum sonuçları

Grup No	Grup Adı	HS	PSO	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.

**Şekil 2:** İki açıklıklı altı katlı çerçevenin ağırlık iterasyon ilişkisi



Şekil 3: İki açıklıklı altı katlı çerçevenin x doğrultusundaki yanal ötelenmeleri

4.2 Üç açıklıklı onbeş katlı çerçeve

Şekil 4' de doksan elemandan oluşan üç açıklıklı ve on beş katlı düzlem çelik çerçevenin boyutları, yükleme durumu ve elemanların grup numaraları gösterilmektedir.

Çerçevenin mukavemet ve deplasman kısıtlayıcıları (AISC-LRFD) şartnamesinden alınmıştır. Çerçevenin en üst katının yanal ötelemesi 23,5 cm ile sınırlandırılmıştır. Kiriş ve kolon elemanları AISC-LRFD' de yer alan standart 267 adet "W" kesitlerinden seçilmiştir.

[19-21] nolu referanslarda bu düzlem çelik çerçeve Parçacık Sürüsü Optimizasyon Algoritması (PSO), Parçacık Sürüsü Optimizasyonu Pasif Toplanma Algoritması (PSOPC), Büyük Patlama-Büyük Çatırdama Algoritması (BB-BC) ve Heuristik Parçacık Sürüsü Karınca Kolonisi Optimasyonu

(HPSACO) Algoritması olmak üzere farklı algoritmalar kullanılarak boyutlandırılmıştır. Bu çalışmada ise aynı çerçevenin optimum tasarımı Sosyal Örümcek Optimizasyon algoritması kullanılarak yapılmıştır [22]. Sonuçlar farklı optimizasyon yöntemleri göz önüne alınarak yapılan çözümlerden elde edilen sonuçlarla karşılaştırılarak Tablo 3' de gösterilmiştir.

Tablo 3' de görüldüğü üzere göz önüne alınan çerçevenin diğer çalışmalara nazaran daha az iterasyonla optimum boyutlandırılması yapılmıştır. SSO algoritması ile optimum tasarımı yapılan çerçevenin ağırlığı PSO algoritmasından %18,73, PSOPC algoritmasından %7,52, BB-BC algoritmasından %3,88 ve HPSACO algoritmasından ise %1,92 daha az bulunmuştur.

Tablo 3: İki açıklıklı onbeş katlı çerçevenin SSO algoritmasına göre optimum sonuçları

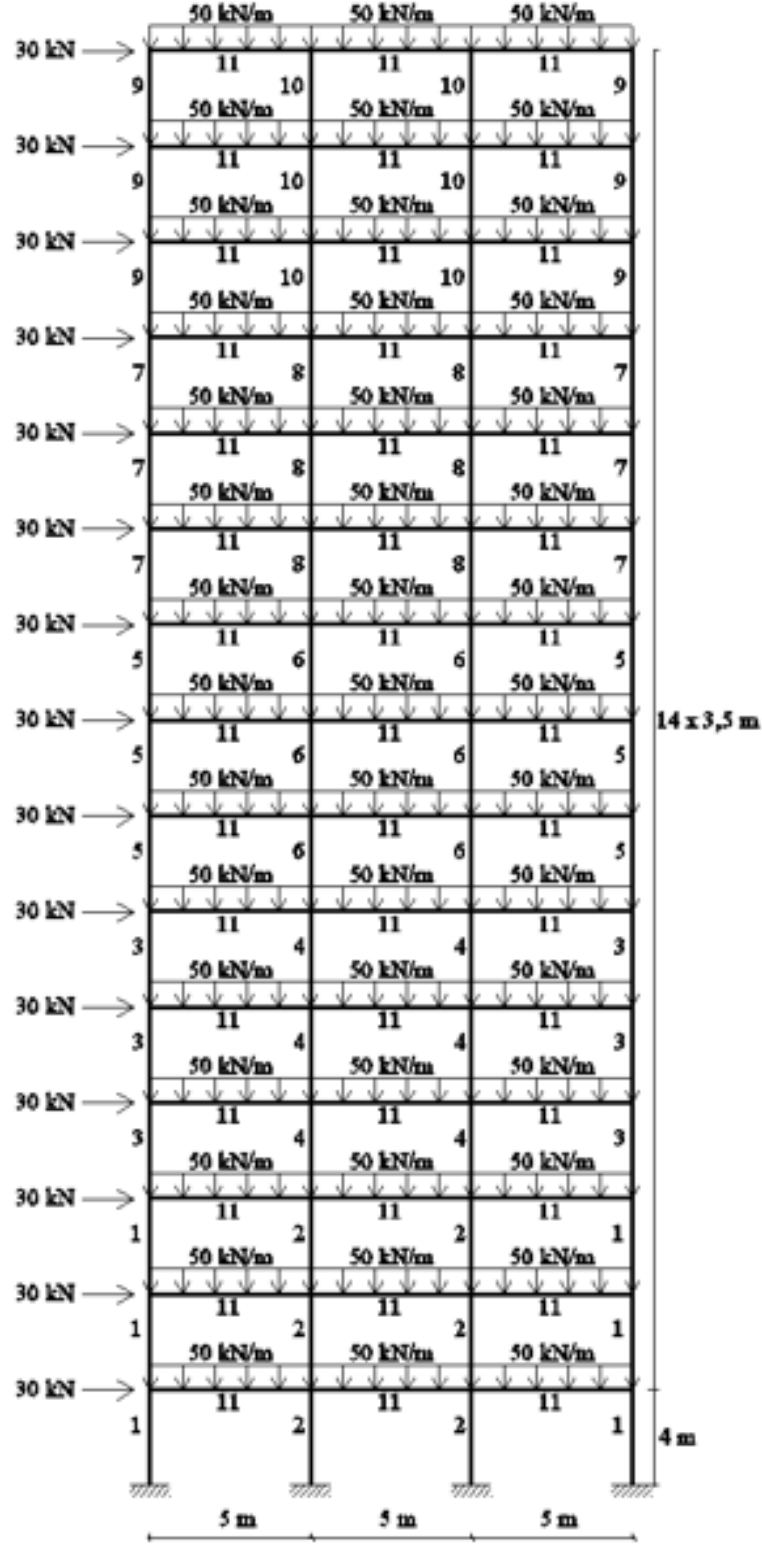
A. Kayeh & S. Talatahari						
Grup No	Grup Adı					Bu Çalışma
		PSO	PSOPC	BB-BC	HPSACO	(SSO)
1	Kolon	W33x118	W26x129	W24x117	W21x111	W14x120
2	Kolon	W33x263	W24x131	W21x132	W18x158	W14x159
3	Kolon	W24x76	W24x103	W12x95	W10x88	W33x118
4	Kolon	W36x256	W33x141	W18x119	W30x116	W21x111
5	Kolon	W21x73	W24x104	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.

5. SONUÇ VE ÖNERİLER

Bu çalışmada, ayırık değişkenli düzlem çelik çerçevelerin optimum tasarımı için sosyal örümcek optimizasyon (SSO) algoritması geliştirilmiştir. SSO algoritması, dişi ve erkek örümceklerin doğal yaşamdaki davranışları göz önüne alınarak geliştirilen bir optimizasyon yöntemidir. Algoritma, Visual Basic Applications (VBA) dilinde kodlanmış ve SAP2000 yazılımı ile entegre edilmiştir. Çerçevelerin mukavemet ve deplasman değerleri SAP2000 programıyla hesaplanırken, yanal ötelenme değerleri hazırlanan algoritma içinde kodlanmıştır. Çalışmada; literatürde mevcut olan ve değişik optimizasyon yöntemleri ile boyutlandırılan, ilki iki açıklıklı ve altı katlı diğeri ise üç açıklıklı ve onbeş katlı olan iki farklı düzlem çelik çerçeve göz önüne alınmıştır. Bu çerçeveler, çalışmada sunulan Sosyal Örümcek Optimizasyon algoritması ile boyutlandırılmış ve elde edilen

sonuçlar karşılaştırılmıştır. Yapılan karşılaştırmalar sonucunda diğer optimizasyon yöntemlerine nazaran Sosyal Örümcek Optimizasyon algoritması ile boyutlandırılan düzlem çelik çerçevelerin daha az iterasyon ile daha kısa sürede ve daha ekonomik olacak şekilde boyutlandırılabilceği görülmüştür. Yapılan çalışmadan elde edilen tecrübeler göre dişi örümceklerin sayısının fazla olması ve iterasyon sayısının artması SSO algoritmasının performansını olumlu yönde etkilediği görülmüştür. Ayrıca 0 ile 1 arasında değişen PF parametresinin 0,1 ve 0,125 değerlerinde daha iyi sonuçlar alındığı gözlenmiştir. Düzlem çelik çerçevelerin optimum tasarımında kullanılan SSO algoritmasının rekabetçi ve güçlü bir algoritma olduğu söylenebilir. Ayrıca bu algoritma ile literatürde mevcut olmayan farklı tipteki çelik çerçevelerin de en ekonomik olacak şekilde optimum boyutlandırılmaları mümkün olabilecektir.



Şekil 4: Üç açıklıklı onbeş katlı çerçevenin boyutları, yükleme durumu ve elemanların grup numaraları [22].

Yazarların Katkıları: Her iki yazar, literatür taraması, araştırma fikrini geliştirme, makale yazımı ve düzenlenmesi süreçlerine ortak katkı sağlamıştır.

Çıkar Çatışmaları: Bu çalışmanın yazarları herhangi bir kurum/kuruluş veya kişi ile çıkar çatışması olmadığını beyan etmektedir

6. KAYNAKLAR

- [1] S. Sezer and M. Hiçyılmaz, "Çelik yapıların kaynaklı birleşimlerinin metasezgisel yöntemlerle optimum tasarımı," *Dokuz Eylül Üniv. Müh. Fak. Fen ve Müh. Derg.*, vol. 24, no. 70, pp. 277–290, 2021.
- [2] S. O. Degertekin, "Optimum design of geometrically non-linear steel frames using artificial bee colony algorithm," *Steel Compos. Struct.*, vol. 12, no. 6, pp. 505–522, 2012.
- [3] M. P. Saka and E. Dogan, "Optimum design of unbraced steel frames to LRFD–AISC using particle swarm optimization," *Adv. Eng. Softw.*, vol. 46, no. 1, pp. 27–34, 2012.
- [4] S. Talatahari, E. Khalili, and S. M. Alavizadeh, "Accelerated particle swarm for optimum design of frame structures," *Math. Probl. Eng.*, vol. 2013, pp. 1–6, 2013.
- [5] M. R. Maheri and M. M. Narimani, "An enhanced harmony search algorithm for optimum design of side sway steel frames," *Comput. Struct.*, vol. 136, pp. 78–89, 2014.
- [6] P. Murren and K. Khandelwal, "Design-driven harmony search (DDHS) in steel frame optimization," *Eng. Struct.*, vol. 59, pp. 798–808, 2014.
- [7] M. P. Saka, "Optimum design of steel frames with stability constraints," *Comput. Struct.*, vol. 41, no. 6, pp. 1365–1377, 1991.
- [8] A. Daloğlu and M. Armutçu, "Genetik algoritma ile düzlem çelik çerçevelerin optimum tasarımı," *Teknik Dergi*, vol. 9, no. 42, pp. 1601–1615, 1998.
- [9] A. Daloğlu and Z. Aydın, "Kafes sistemlerin uygulamaya yönelik optimum tasarımı," *Pamukkale Üniv. Müh. Fak. Müh. Bil. Derg.*, vol. 5, no. 1, pp. 951–957, 1999.
- [10] E. S. Kameshki and M. P. Saka, "Optimum design of nonlinear steel frames with semi-rigid connections using a genetic algorithm," *Comput. Struct.*, vol. 79, no. 17, pp. 1593–1604, 2001.
- [11] M. S. Hayalioğlu and S. Ö. Değertekin, "Minimum cost design of steel frames with semi-rigid connections and column bases via genetic optimization," *Comput. Struct.*, vol. 83, no. 21–22, pp. 1849–1863, 2005.
- [12] A. Kaveh and S. A. Talatahari, "Discrete particle swarm ant colony optimization for design of steel frames," *Asian J. Civ. Eng. (Build. Hous.)*, vol. 9, pp. 563–575, 2007.
- [13] M. P. Saka, "Optimum design of steel sway frames to BS5950 using harmony search algorithm," *J. Constr. Steel Res.*, vol. 65, no. 1, pp. 36–43, 2009.
- [14] LRFD-AISC, *Manual of Steel Construction: Load and Resistance Factor Design*, vol. 1, American Institute of Steel Construction, 1994.
- [15] E. Baş, "Sosyal örümcek algoritmasının sürekli ve ayrık optimizasyon problemlerinde performans iyileştirmeleri," Doktora Tezi, Konya Teknik Üniversitesi, Fen Bil. Enstitüsü, Konya, 2020.
- [16] E. Cuevas and M. Cienfuegos, "A new algorithm inspired in the behavior of the social-spider for constrained optimization," *Expert Syst. Appl.*, vol. 41, no. 2, pp. 412–425, 2014.
- [17] E. Cuevas, M. Cienfuegos, D. Zaldivar, and M. Perez-Cisneros, "A swarm optimization algorithm inspired in the behavior of the social-spider," *Expert Syst. Appl.*, vol. 40, no. 16, pp. 6374–6384, 2013.
- [18] J. J. Q. Yu and V. O. K. Li, "A social spider algorithm for global optimization," *Appl. Soft Comput.*, vol. 30, pp. 614–627, 2015.
- [19] A. Kaveh and S. Talatahari, "Hybrid algorithm search, particle swarm and colony for structural design optimization," *Stud. Comput. Intell.*, vol. 239, pp. 159–198, 2009.
- [20] A. Kaveh and S. Talatahari, "An efficient hybrid algorithm based on harmony search, particle swarm and colony strategies for optimal design of structures," in *Harmony Search Algorithms for Structural Design Optimization*, pp. 159–198, 2009.
- [21] A. Kaveh and S. Talatahari, "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.
- [22] M. Yetkin, "Düzlem çelik çerçevelerin sosyal örümcek optimizasyonuna göre boyutlandırılması," Yüksek Lisans Tezi,

Balıkesir Üniversitesi, Fen Bilimleri Enstitüsü,
Balıkesir, 2015.