



IZGARA SİSTEMLERİN LRFD-AISC ŞARTNAMESİNE GÖRE PARÇACIK KÜME YÖNTEMİ KULLANILARAK OPTİMUM BOYUTLANDIRILMASI

E. Doğan*

Atatürk Üniversitesi,
Erzurum, Türkiye

F. Erdal†

Akdeniz Üniversitesi,
Antalya, Türkiye

S. Çarbaş‡

Orta Doğu Teknik Üniversitesi,
Ankara, Türkiye

M. P. Saka¹

Bahreyn Üniversitesi,
İsa Town, Bahrain

ÖZET

Bu çalışmada ızgara sistemler için parçacık küme optimizasyon yöntemini temel alan bir optimum tasarım algoritması sunulmaktadır. Parçacık küme optimizasyon yöntemi hayvanlarda rastlanan böcek kümelenmesi, kuş sürüleri ve balıkların toplu hareketleri gibi sosyal davranışlarının taklit edildiği bir sayısal optimizasyon tekniğidir. Sürü içindeki her bir birey optimum tasarım problemine aday bir çözüm ifade eder. Izgara sistemlerin optimum tasarımının gereği olarak; yapı elemanlarının W profiline uygun olması, LRFD-AISC tasarım şartnamelerindeki sınırlayıcıları sağlaması ve sistemin minimum ağırlığa sahip olması gerekmektedir. Bu amaçla burkulmayı da hesaba katan dayanım sınırlayıcıları göz önünde bulundurulmuş ve algoritmanın girişler için Amerikan Şartnamesindeki 272 kesit içerisinden seçim yapması sağlanmıştır. Her bir grup içindeki elemanlara aynı kesitin tayin edilebilmesi için eleman gruplandırılması yapılmıştır. Şartnamedeki sınırlayıcıları dikkate alarak minimum sistem ağırlığını hesaplayan bu algoritmanın etkinliği örnek problemlerle gösterilmiştir.

GİRİŞ

Yatay düzlemde birbirine rijit bağlı elemanlardan oluşan ızgara sistemler yapılarda köprü tabliyesi, gemi gövdesi, kat döşemesi gibi geniş alanların kaplanması amacıyla sıklıkla kullanılan yapı elemanlarıdır. Kendi düzlemlerine dik olarak yüklenmiş olan bu tür sistemlerin tasarımı geçmişten bu yana yapı mühendislerinin çözmek zorunda olduğu başlıca problemlerden biri olmuştur. Bu çalışmada sözkonusu yapı sistemlerinin stokastik optimizasyon teknikleri yardımıyla optimum boyutlandırılması konusu ele alınmıştır.

Stokastik sayısal optimizasyon arama tekniklerinde kaydedilen ilerlemelerin sonucu olarak etkili bir şekilde çalışan yeni boyutlandırma teknikleri yapı mühendislerinin kullanımına sunulmuştur. Bu tekniklerin kullanımıyla birlikte yapısal optimizasyon problemlerinin çözümü matematiksel programlama yöntemleri ile elde edilenlere göre çok daha etkin hale gelmiştir. Bu stokastik optimizasyon yöntemlerinin temelindeki mantık sürü zekası, erimiş metallerin soğutulma işlemi, en iyinin hayatta kalması gibi prensiplerin sayısal algoritmalar içinde taklit edilmesidir. Bu metodlar üretilmelerinde esinlenilen doğasal fenomene bağlı olarak isimlendirilirler.

*Dr., İnşaat Müh. Böl., E-posta: erkan.dogan@atauni.edu.tr

†Dr., İnşaat Müh. Böl., E-posta: eferhat@akdeniz.edu.tr

‡Arş. Gör., Mühendislik Bilimleri Böl., E-posta: carbas@metu.edu.tr

¹Prof. Dr., İnşaat Müh. Böl., E-posta: mpsaka@eng.uob.bh

Karınca kolonisi optimizasyon yöntemi karıncaların yuvaları ve yiyecekleri arasındaki en kısa yolu bulma çabalarını simule ederken, benzetimli tavlama erimiş metallerin soğutulma sürecindeki enerji minimizasyonunu kullanır. Bu stokastik arama teknikleri amaç fonksiyonu ve sınırlayıcıların türevlerine ihtiyaç duymadığı gibi, deterministik kurallar yerine olasılığa dayalı arama yollarını kullanırlar [1,2]. Tasarım mühendisleri son yıllarda alışılmışın dışında kalan bu etkili ve bir o kadarda güçlü tekniklerden esinlenerek bir çok optimum boyutlandırma algoritması geliştirmişlerdir. [3] de bu algoritmalar hakkında detaylı bilgi yer almaktadır.

Sürü zekası mantığına dayalı olarak işleyen parçacık sürü optimizasyonu tekniği bu tekniklerden biridir [4-5]. Doğada bulunan kuş, balık gibi hayvanlar sadece çoğalmak için değil aynı zamanda yiyecek bulmak ve avcılardan korunmak için de sürü oluştururlar. Kuş sürüleri ve balık topluluklarındaki her bir bireyin birbirlerine çarpmadan senkronize bir şekilde hareket edebilmeleri için çevresindekilerle uyumlu olması gerekir. Sürü içindeki her birey sürünün bir yerden başka bir yere rahatça ulaşabilmesi için çevresindeki bireylerle arasına optimum ölçüde bir mesafe koyar. Parçacık sürü optimizasyon tekniği kuşların hareketlerini temsil eden bu sosyal davranışın bir benzeşimi olarak tasvir edilebilir. Populasyon temelli bir algoritmadır. Populasyonu sürü olarak adlandırılırken sürü içindeki bireylere ise parçacık veya partikül ismi verilir. Sürü içerisindeki her bir parçacık optimum boyutlandırma problemi için birer aday çözümdür. Partiküller örnek uzaya doğru uçuşa eğilimindedirler ve bir zaman aralığı için pozisyonları, eski pozisyonları ve yenilenmiş olan hız vektörleri kullanılarak güncellenir. Tekniğin orijinalinde [6] değişkenler sürekli değer alabilen türde değişkenler olarak kabul edilmişlerdir. Başka bir ifadeyle boyutlandırma içinde yer alacak olan sayısal değerler, önceden belirlenmiş olan bir sayı kümesi içinden seçilmeksizin rastgele sayı seçme uygulamasından faydalanılarak elde edilen gerçek sayı değerleri ile ifade edilir. Literatürde yer alan bir çok yapısal optimizasyon probleminde bu kabulun yapıldığı parçacık sürü optimizasyon tekniği kullanılmıştır [7-9]. Ancak kirişlerin hazır kesit tablolarından seçildiği optimum çelik yapı boyutlandırma problemlerinde bu kabulun yapılması mümkün değildir. Çünkü hazır kesit tablolarından seçim yapabilmek için her profilin tablo içinde yer aldığı satır numarasının tasarım değişkeni olarak atanması gerekmektedir. Bunu yapabilmek ise ancak tamsayı değerlerinin seçimiyle mümkün olabilir. Literatürde sürekli sayıları tamsayı haline getirmek için kullanılan iki adet yaklaşım bulunmaktadır. Bunlardan birincisi ilk olarak Kennedy ve Eberhart [10] tarafından ortaya konulmuştur. Bu yaklaşımda sürekli değişkenler 0 ve 1 sayı çifti kullanılarak tamsayı haline getirilir. İkinci yöntem ise Liu tarafından önerilmiş olan küsüratlı sayıların en yakın tamsayılarla yuvarlanması şeklindedir [11]. Bu çalışmada gerçek sayılardan tamsayı elde etmek üzere ikinci yöntem olan yuvarlama işlemi kullanılmıştır. Bu çalışmada ızgara sistemin kirişleri için kullanılacak W- kesitleri LRFD- AISC [12] Amerikan kodundaki kesit listesi içinden seçen parçacık- küme optimizasyon yöntemi temelli bir optimum boyutlandırma algoritması geliştirilmiştir.

Tamsayı değişkenleri için Parçacık Küme Optimizasyon Yöntemi

Tamsayı değişkenli optimum boyutlandırma problemi aşağıdaki gibi tasvir edilebilir.

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

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

Parçacık küme optimizasyon yöntemi sürüyü temsil eden ve amaç fonksiyonu arama uzayı içinde rastgele seçilmiş olan parçacıklardan ibarettir. Orijinal parçacık küme optimizasyon yöntemi sürekli değişkenler için üretilmiştir. Tekniği tamsayı değerleri ile kullanabilmek için bazı değişiklikleri uygulamak gerekmektedir.

Öncelikle tamsayı değerlerini içeren X kümesindeki x_i değerleri artan sayılar şeklinde sıralanır. Daha sonra bu sayıların satır numaraları tasarım değişkeni olarak atanır. Örneğin 272 adet değer içeren tasarım setindeki 1 ile 272 arasındaki satır numaraları ana tasarım değişkenleridir. Tasarım döngüsünün herhangi bir adımında, algoritma tarafından satır numarasının üretilmesiyle birlikte bu satır numarasına karşılık gelen gerçek tasarım değişkeni değeri set içerisinden kolaylıkla alınabilir. Prosedür altı temel adımdan oluşur.

1. Adım. Bir partikül kümesi küme boyutunu temsil eden ve önceden belirlenmiş miktarda partikülden oluşur. Bu amaçla öncelikle sürü içindeki her bir partikül için bütün ilk pozisyonlar ve hızlar rastgele başlatma prensibiyle oluşturulur. Parçacıkların pozisyonları olan x_0^i değerleri, l_0^i satır numaraları olarak temsil edilir bununla birlikte her bir parçacık v_0^i hız vektörüne sahiptir. Bu değerler aşağıdaki formüller yardımıyla hesap edilir.

$$I_0^i = INT[I_{\min} + r(I_{\max} - I_{\min})] \quad (2)$$

$$v_0^i = [(I_{\min} + r(I_{\max} - I_{\min})) / \Delta t] \quad (3)$$

buradaki r değerinin tespiti için 0 ile 1 arasında değişen sayılar içinden rastgele sayı seçme uygulamasından faydalanılır. $I_{\min}$ 1 e eşittir ve $I_{\max}$ set içindeki toplam değer sayıdır. l_0^i in hesaplanmasıyla birlikte gerçek tasarım değişkeni olan x_0^i setten alınır.

2. Adım. Daha sonra tasarım değişkenlerini temsil eden pozisyon vektörleri x_k^i kullanılarak bütün partiküller için uyumluluk analizi yapılır ve amaç fonksiyonu değerleri $f(x_k^i)$ hesaplanır.

3. Adım. Üçüncü adımda partiküllerin en iyi değerinin ve küme içindeki en iyi partikül değerinin güncellenmesi işlemi yapılır: Bir partikülün o ana kadarki en iyi pozisyonu p_k^i (minimum amaç fonksiyonu $f(x_k^i)$ değerine sahip olan en iyi boyutlandırma) partikülün en iyi değeri olarak kabul edilir ve her bir partikülün en iyi değeri kaydedilir. Bunun yanı sıra işlemin başlangıcından itibaren herhangi bir partikül tarafından elde edilmiş olan en iyi pozisyon ise en iyi global pozisyon p_k^g olarak kaydedilir. Her iterasyon adımında her bir partikül ve sürü için en iyi pozisyon değerleri güncellenir.

4. Adım. Bu işlemten sonra her bir partikülün hız vektörü, partiküllerin mevcut pozisyonu, en iyi pozisyonu ve global en iyi pozisyon dikkate alınarak güncellenir.

$$v_{k+1}^i = wv_k^i + c_1r_1 \frac{(p_k^i - x_k^i)}{\Delta t} + c_2r_2 \frac{(p_k^g - x_k^i)}{\Delta t} \quad (4)$$

Burada r_1 ve r_2 , 0 ile 1 arasında yer alan rastgele sayılar, p_k^i o ana kadar bulunmuş en iyi pozisyon ve p_k^g ise k anında sürüde bulunan en iyi pozisyon değeridir. w algoritmanın keşif özelliklerini kontrol eden parçacık ataletini, c_1 ve c_2 ise parçacığın kendisine ve sürüye ne kadar bağlı kalacağını belirleyen güven parametrelerini gösterir.

5. Adım. Beşinci adımda bir önceki adımda güncellenen hız vektörü kullanılarak her bir partiküle ait satır numarası değeri yenilenir.

$$I_{k+1}^i = INT(I_k^i + v_{k+1}^i \Delta t) \quad (5)$$

Buradaki l_{k+1}^i , i numaralı parçacığın $k+1$ inci iterasyondaki x_{k+1}^i pozisyonuna karşılık gelen satır numarası, v_{k+1}^i hız vektörü ve Δt zaman adım aralığı değeridir.

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

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

Sınırlayıcıların kontrolü

Optimizasyon işlemi içinde değişkenler için belirlenen hazır profiller kullanılarak yapı analiz edilir ve sınırlayıcıları sağlayıp sağlamadığı incelenir. Bu durumun kontrolünü literatürde parçacık- sürü optimizasyon yöntemi için en etkin sonuçları verdiği belirlenmiş olan geri dönüş (fly-back) mekanizması üstlenir [8]. Bu mekanizmaya göre ilk olarak bütün parçacıklar analiz edilir ve sınırlayıcıları sağlayıp sağlamadığı kontrol edilir. Eğer bu parçacıklardan bir veya birkaçı sınırların dışında sonuç veriyorsa bunlar atılır ve yerlerine yenileri üretilir ve yeniden inceleme yapılır. Eğer sınırlardan sapma çok büyük değilse bu durumda bu parçacıklar iterasyon süresince sınırları sağlayabilen yeni parçacıklar üretebileceği düşünülerek çözüm içinde tutulur. Bunu yapabilmek için parçacıkların üretilmesi aşamasında hata payı büyük tutulur, parçacıkların üretilmesi işlemi tamamlanıp iterasyonlara geçildikten sonra adım adım küçültülerek son değeri olan 0.001 e ulaştırılır ve bu değer ile iterasyona devam edilir.

Izgara sistemlerin LRFD- AISC ye göre optimum boyutlandırılması

Izgara sistemlerin boyutlandırılmasındaki ana prensip kirişler için standart çelik kesit tablosundan kesit atamaktır. Bu sistemlerin optimum boyutlandırılmasının gereği olarak; yapı elemanlarının W profiline uygun olması, LRFD-AISC tasarım şartnamelerindeki sınırlayıcıları sağlaması ve sistemin minimum ağırlığa sahip olması gerekmektedir. LRFD- AISC sınırlayıcılarının tasarım problemi formülasyonunda kullanılmasıyla birlikte aşağıdaki programlama problemi elde edilir.

$$\text{Minimum } W = \sum_{k=1}^{N_d} m_k \sum_{i=1}^{N_k} l_i \quad (6)$$

$$\delta_j / \delta_{ju} \leq 1 \quad \text{ve} \quad j = 1, 2, \dots, p \quad (7)$$

$$M_{ur} / (\phi_b M_{nr}) \leq 1 \quad \text{ve} \quad r = 1, 2, \dots, nm \quad (8)$$

$$V_{ur} / (\phi_v V_{nr}) \leq 1 \quad \text{ve} \quad r = 1, 2, \dots, nm \quad (9)$$

Buradaki (6) denklemi ızgara sistemin ağırlığını gösterir. N_d sistemdeki toplam grup sayısını ifade ederken, m_k k numaralı gruba atanan hazır profil kesitinin birim ağırlığını, l_i k numaralı gruptaki elemanların uzunluğunu ve N_k k grubundaki toplam eleman sayısını sembolize eder. (7) numaralı denklem yatay öteleme sınırlayıcısıdır. Bu denklemdeki δ_j j numaralı birleşim noktasının ötelemesini ve δ_{ju} ise ötelemenin üst sınırını gösterir. Yatay ötelemeler ızgara sistemler için matris yöntemlerin kullanılmasıyla hesaplanır. (8) numaralı denklem LRFD-F2 şartnamesi uyarınca yatay olarak tutulmuş kirişler için dayanım sınırlayıcısıdır. Bu denklemdeki ϕ_b eğilmeye karşı direnç faktörüdür ve değeri 0.9 olarak verilir. M_{nr} sınır moment dayanımı M_{ur} ise r elemanına ait servis momentidir. (9) numaralı denklem LRFD-F2 ye göre kesme dayanımı sınırlayıcısıdır. Bu eşitsizlikteki ϕ_v kesme için direnç faktörü olup 0.9 olarak alınır. V_{nr} sınır kesme dayanımı V_{ur} ise r elemanı için servis kesme kuvvetidir.

Optimum boyutlandırma algoritması

(6)- (9) denklemlerinde verilen optimum boyutlandırma probleminin çözümü parçacık küme optimizasyonu yöntemi kullanılarak elde edilmiştir. Parçacık küme yöntemi ile hazırlanmış bu algoritma standart W-kesit listesinin satır numaralarını tasarım değişkenleri olarak kabul eder. Satır numarası seçilir seçilmez algoritma için gereken kesit özellikleri rahatlıkla bulunabilir hale gelir. Bu nedenle tasarım vektörü her bir grup için ayrı ayrı atanmış olan ve listenin satır numaralarına denk gelen bu tamsayı değerlerini içerir. Algoritma aşağıdaki adımlardan oluşur.

1. Parametre değerleri (parçacık ataleti w , parçacığın güven parametresi c_1 , sürünün güven parametresi c_2 , parçacığın maksimum hızı V_{max} , zaman adım aralığı Δt ve sürüdeki parçacık sayısı NPT) seçilir. Bu çalışmada kullanılan parametre değerleri Tablo 1 de verilmiştir. Bu değerlere örnekler üzerinde birçok deneme yapıldıktan sonra karar verilmiştir.
2. Parçacıkların üretilmesi: Izgara sistem içindeki her bir grup için listeden çelik kesit satır numarası seçilir.
3. Parçacıkların hızları rastgele seçme prensibiyle hesaplanır.
4. Bu kesitlerle sistem analiz edilir ve tasarım sınırlayıcılarını sağlayıp sağlamadığı kontrol edilir. Eğer sınırlayıcıları büyük ölçüde aşıyorsa bu durumda bu parçacıklar yok sayılarak yerine yenileri üretilir. Eğer sınırlayıcılar sağlanmıyor ama değerler sınırdan çok uzak değilse bu durumda sözkonusu parçacıklar sürü içinde tutulur.
5. Yeni parçacıkların kabul edilir olup olmadıkları irdelenir. Değilse bu durumda 4. adımdan tekrar işleme devam edilir.
6. Parçacıkların seçim işlemi tamamlandıktan sonra her bir parçacık için amaç fonksiyonu değeri hesap edilir. Bunlardan en iyi amaç fonksiyonu değerine sahip olan parçacık belirlenir.
7. Denklem 4 ve denklem 5 tekrar kullanılarak parçacıkların hız ve pozisyon vektörleri güncellenir.
8. Prosedürün başında seçilmiş olan maksimum iterasyon sayısına ulaşılan kadar 4. adımdan 7. adıma kadar olan işlemler tekrar edilir. Bu sayı amaç fonksiyonunda daha fazla iyileşme görülmeyeceği kanaatine varılabilecek derecede büyük seçilir.

Tablo 1. Parçacık- küme algoritması için kullanılan parametre seti

Parçacık- küme algoritması için kullanılan parametre setinin değerr aralıkları		
$NPT = 10 - 50$	$\Delta t = 1.0$	$c_1 = 1.0 - 2.0$
$w = 0.05 - 0.50$	$N_{ite} = 10000$	$c_2 = 1.0 - 2.0$

Boyutlandırma örnekleri

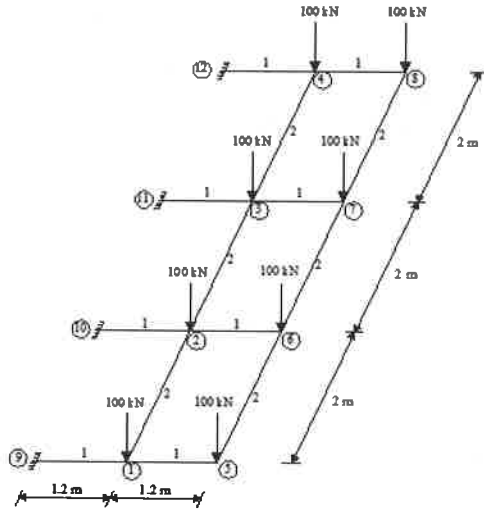
İki adet ızgara sistem örneği bir önceki bölümde bahsedilmiş olan parçacık küme optimizasyon yöntemi kullanılarak elde edilmiş optimum boyutlandırma algoritmasıyla optimize edilmiştir. Algoritmanın kullanacağı kesit tablosu W100x19.3mm den başlayarak W1100x499mm de son bulan 272 adet W kesit içermektedir.

14- elemanlı ankastre ızgara sistemi

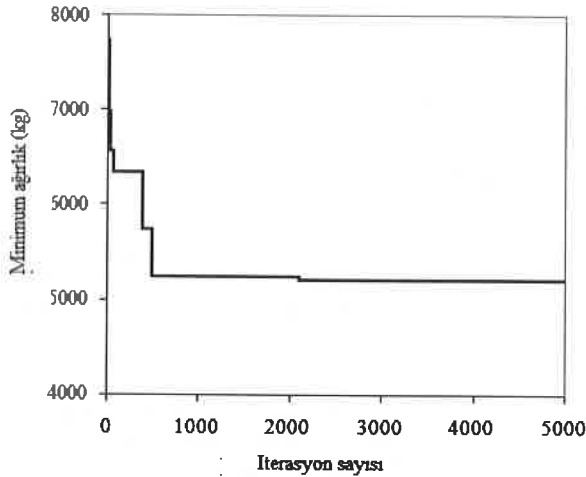
Şekil 1 de görülen 14 elemanlı basit ankastre ızgara sistem ilk boyutlandırma örneği olarak dikkate alınmıştır. Sistemin boyutları, eleman gruplandırılması ve yüklemeler de aynı şekil üzerinde gösterilmiştir. Moment ve kesme kuvveti sınırlayıcıları üstte açıklandığı gibi LRFD-AISC ye uygun olarak dikkate alınırken, 7 ve 8 numaralı birleşim noktalarına ait maksimum düşey yerdeğiştirme 20 mm ile sınırlandırılmıştır. Sistem elemanların iki farklı grup altında toplanması suretiyle tasarlanmıştır. Boyuna kirişler 1 no'lu ve enine kirişler 2 no'lu gruplar olarak kabul edilmiştir. Bu nedenle tasarım problemi sadece iki adet tasarım değişkeni kullanılarak çözülecektir.

Parçacık küme metodunun 1000 adım kadar iterasyonu sonucunda elde edilen kesitler birinci grup için W760X147 ve ikinci grup için W530X74 şeklinde elde edilmiştir ve bu kesitlerin kullanımıyla oluşan sistemin ağırlığı 5344.6 olarak hesaplanmıştır. Maksimum deplasman 7 numaralı birleşim

noktasında ve 15.5 mm şeklinde ölçülmüştür. Moment sınırlayıcısı için maksimum dayanım oranı 0.91 olup bu değer 1 numaralı elemanda görülmüştür. Bu değerler göstermiştir ki iterasyonlar bir miktar daha sürdürülürse daha iyi sonuçlar elde edilebilir. Bu doğrultuda iterasyonlar 2100 e kadar devam ettirilmiş ve sonuçlar Tablo 2 deki gibi kaydedilmiştir. Bu kez kesitler 1 ve 2 no'lu gruplar için sırasıyla W760X134 ve W360X79 olarak belirlenmiştir. Ağırlık bu kez 5305.7 olmuştur ki bu da daha iyi bir tasarım elde edildiğini gösterir. Bu tasarımda kaydedilen yerdeğişiklikler 7 ve 8 no'lu birleşim noktaları için sırasıyla 18.8 ve 7 mm olarak kaydedilmiştir. Maksimum dayanım sınırlayıcısı 1 nolu grup elemanlarında görülmüş olup bu değer 0.97 şeklinde hesaplanmıştır. Bu sonuçlar neticesinde dayanım sınırlayıcılarının bu boyutlandırma problemi için daha baskın olduğu kanısına varılır. Daha sonra problem tekrar 10000 adım için çözülmüş ve aynı sonuçların elde edildiği görülmüştür. Şekil 2 de minimum ağırlık- zaman grafiği verilmiştir.



Şekil 1. 14-elemanlı ızgara sistem



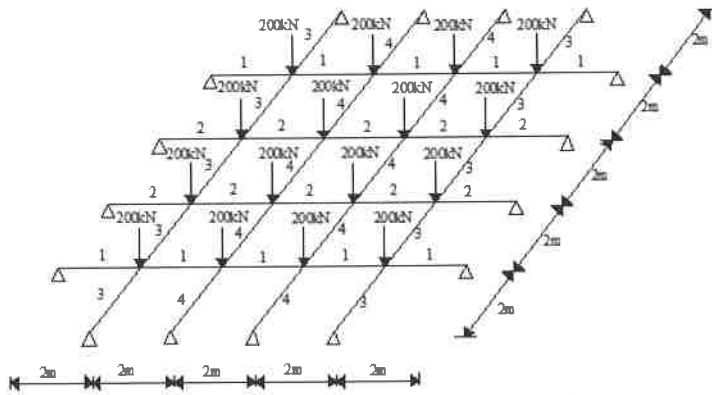
Şekil 2. 14-elemanlı ızgara sistem için minimum ağırlık zaman grafiği

Tablo 2. 14-elemanlı ızgara sistem için optimum boyutlandırma

W-Kesitler		δ_{max} (mm)	Maks. dayanım oranı	Min. ağırlık (kg)
Grup 1	Grup 2			
W760X134	W360X79	18.8	0.97	5305.7

40- elemanlı ızgara sistemi

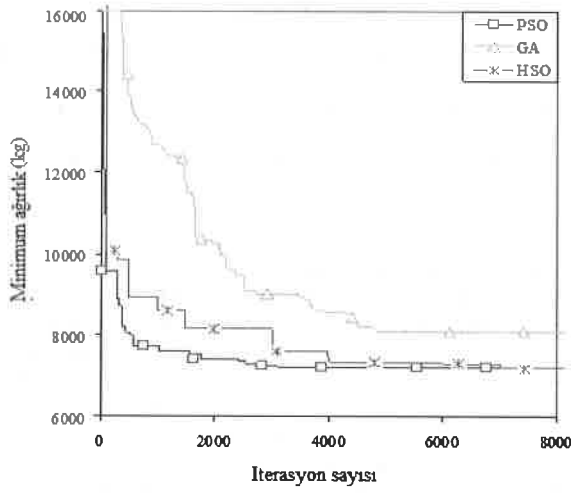
İkinci örnek olarak şekil 3 de görülen 40 elemanlı basit ızgara sistem ele alınmıştır. Sistemi oluşturan çelik yapı elemanları dört grup altında toplanmıştır. Dış ve iç boyuna kirişler birinci ve ikinci grup elemanları olarak düşünülürken, dış ve iç enine kirişler üçüncü ve dördüncü grup olarak hesaba katılmışlardır. Şekil 3 da gruplandırmalar ve yüklemeler gösterilmiştir. 6, 7, 10 ve 11 nolu birleşim noktalarındaki düşey deplasmanlar 25mm ile sınırlandırılmıştır. 40- elemanlı ızgara sistem parçacık küme, harmoni arama ve genetik algoritma yöntemleriyle ayrı ayrı çözülmüştür. Başlangıç popülasyonları ve maksimum iterasyon sayıları bütün yöntemler için eşit tutulmuştur. Bu üç yöntem için elde edilen optimum sonuçlar Tablo 3 de gösterilmiştir. Parçacık- küme yöntemi parametreleri için en uygun parametrelerin bulunması nedeniyle yapılan hassasiyet analizinin sonuçları Tablo 1 de verilmiştir. Tablodan NPT , $c1$, $c2$ w, V_{max} ve Δt için alınan 10, 1, 1, 0.08, 2 ve 1 söz konusu sistem için minimum ağırlığı vermiştir.



Şekil 3. 40-elemanlı ızgara sistem

Analizler sonucunda parçacık- küme yöntemi bu örnekte 1 no'lu grup için W410X46.1, 2 no'lu grup için W460X52 no'lu grup için W200X15 ve son olarak 4 no'lu grup için W1000X22 kesitlerini optimum kesitler olarak belirlemiştir. Bu kesitlerin kullanımıyla oluşan sistemin ağırlığı ise 7198.2 kg'dır. Maksimum deplasman değeri 23.2 mm olarak ölçülürken dayanım oranının 0.99 olduğu belirlenmiştir. Bu da dayanım sınırlayıcılarının tasarımında dominant olduğu anlamına gelir.

Aynı soru harmoni arama yöntemi ile de çözülmüş ve aynı sonuçlar elde edilmiştir. Fakat çözüm için genetik algoritma yöntemi kullanıldığında %12.4 daha ağır bir optimum tasarım elde edildiği görülmüştür. Minimum ağırlık- zaman grafiği şekil 4 de verilmiştir. Bu grafikten de anlaşılacağı üzere parçacık küme yöntemi en iyi yakınsamayı gösterir. Parçacık küme yaklaşık 5000 iterasyondan sonra optimum sonuca erişirken harmoni arama yöntemi bu sonuca ancak 7000 iterasyon sonunda ulaşmıştır. Genetik algoritma bu problem için en kötü değeri vermiştir.



Şekil 4. 40-elemanlı ızgara sistem için minimum ağırlık zaman grafiği

Tablo 3. 40-elemanlı ızgara sistem için optimum boyutlandırma

Metod	Optimum W-Kesit Grup No.		$\delta_{\max}$ (mm)	Maks. dayanım oranı	Minimum ağırlık (kg)
Parçacık küme algoritması	1	W410X46.1	23.2	0.99	7198.2
	2	W460X52			
	3	W200X15			
	4	W1000X222			
Harmoni arama	1	W410X46.1	23.2	0.99	7198.2
	2	W460X52			
	3	W200X15			
	4	W1000X222			
Genetik Algoritma	1	W150X13.5	24.7	0.84	8087.91
	2	W610X92			
	3	W410X46.1			
	4	W840X226			

SONUÇLAR

Bu çalışmada ızgara sistemlerin parçacık küme optimizasyonu yöntemi tabanlı optimum boyutlandırılmasını yapan bir algoritma geliştirilmiştir. Parçacık küme optimizasyonu yöntemi basit bir mantığa dayanmakla beraber optimizasyon problemlerinin çözümlerini bulmada oldukça etkin bir yöntemdir. Bu özelliğin doğruluğu bu çalışmada da kanıtlanmıştır. Geliştirilen optimum boyutlandırma yöntemi ile ızgara sistemler için 272 adet W profilinden LRFD-AISC şartnamesindeki sınırlayıcıları sağlayan ve aynı zamanda çerçeveyi minimum ağırlıklı yapan kesitleri kolaylıkla seçebilmiştir. 40 elemanlı ızgara sistem örneğinde parçacık sürü algoritmasının harmoni aramaya göre daha kısa sürede optimum sonuca eriştiği sergilenmiştir.

KAYNAKLAR

- [1] Kochenberger GA and Glover F "Handbook of Meta-Heuristics", Kluwer Academic Publishers, 2003
- [2] Dreco J, Petrowski A, Siarry P and Taillard E, "Meta-Heuristics for Hard Optimization", Springer-Verlag, Berlin, Heidelberg, 2006
- [3] Saka MP, "Optimum Design of Steel Frames Using Stochastic Search Techniques Based on Natural Phenomena: A Review "Civil Engineering Computations: Tools and Techniques", Ed. B. H. V. Topping, Saxe-Coburgh Publications, Chapter 6, pp: 105-147, 2007.
- [4] Bonabeau E, Dorigo M and Theraulaz G, "Swarm Intelligence: From Natural to Artificial Systems", Oxford University Press, U.K., 1999
- [5] Kennedy J, Eberhart R and Shi Y, "Swarm Intelligence", Morgan Kaufmann Publishers, 2001
- [6] Kennedy J, Eberhart RC. "Particle Swarm Optimization", In: Proceedings of IEEE International Conference on Neural Networks NJ: Piscataway; 1942-48, 1995.
- [7] Tasgetiren MF, Liang YC, Sevcli M, Gencyilmaz G. "Particle Swarm Optimization Algorithm for Makespan and Maximum lateness minimization in Permutation Flowshop Sequencing problem", In: Proceedings of the Fourth International Symposium on Intelligent Manufacturing Systems, Turkey: Sakarya; 2004. p. 431–41.
- [8] He S, Prempan E and Wu QH "An Improved Particle Swarm Optimizer for Mechanical Design Optimization Problems", Engineering Optimization, 36, 5, 585-605, 2004.
- [9] Arumugam MS, Rao MVC, Chandramohan A, "A new and Improved Version of Particle Swarm Optimization Algorithm with Global–Local Best Parameters", Knowl Inf. Syst. DOI 10.1007/s10115-007-0109-z.
- [10] Kennedy J, Eberhart RC, "A Discrete Binary Version of the Particle Swarm Algorithm", In Proceedings of the World Multi-Conference on Systemic, Cybernetics and Informatics. Pp: 4104-49, NJ: Piscataway; 1997.
- [11] Liu H, Shichang S, Ajith A, "Particle Swarm Approach to Scheduling Work-Flow Applications in Distributed Data-Intensive Computing Environments", Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06) 0-7695-2528-8/06.
- [12] LRFD-AISC, Manual of Steel Construction, "Load and Resistance Factor Design", Metric Conversion of the Second Edition, AISC, Vol. I&II, 1999
- [13] McGuire W, "Steel Structures", Prentice-Hall, 1968.
- [14] Pezeshk S, Camp CV, and Chen D, "Design of Nonlinear Framed Structures Using Genetic Optimization." J. Struct. Eng., 2000. 126(3), 382–388.
- [15] M.P. Saka, F. Erdal "Harmony search based algorithm for the optimum design of grillage systems to LRFD-AISC", Structural and Multidisciplinary Optimization, Vol. 38, pp. 25-41, March, 2009.