

Kafes Sistemlerin Yapım Kısıtları ile Optimal Tasarımı

Sedanur Balta

**YÜKSEK LİSANS TEZİ**

İnşaat Mühendisliği Anabilim Dalı

Temmuz 2022



Optimal Design of Trusses with Constructability Constraints

Sedanur Balta

**MASTER OF SCIENCE THESIS**

Department of Civil Engineering

July 2022

Kafes Sistemlerin Yapım Kısıtları ile Optimal Tasarımı

Sedanur Balta

Eskişehir Osmangazi Üniversitesi  
Fen Bilimleri Enstitüsü  
Lisansüstü Yönetmeliği Uyarınca  
İnşaat Mühendisliği Anabilim Dalı  
Mekanik Bilim Dalında  
YÜKSEK LİSANS TEZİ  
Olarak Hazırlanmıştır

Danışman: Doç. Dr. Hakan Özbaşaran

Temmuz 2022

## ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. Hakan ÖZBAŞARAN danışmanlığında hazırlamış olduğum "**Kafes Sistemlerin Yapım Kısıtları ile Optimal Tasarımı**" başlıklı tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik, ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik, ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik, ilke ve kurallara göre sunduğumu beyan ederim. 18/07/2022

Sedanur Balta

İmza

## ÖZET

Bilgisayar ile otomatik tasarım bir tasarımın bilgisayar tarafından insan müdahalesi olmadan gerçekleştirilmesidir. Eşzamanlı boyut, şekil ve topoloji optimizasyonu konusunda yapılan çalışmalar bilgisayar ile otomatik tasarımın, bilgisayar destekli tasarımın yerini almaya hazırlandığını göstermektedir. Eşzamanlı boyut, şekil ve topoloji optimizasyonu konusunda sunulan çoğu çalışma, önerdiği optimizasyon yönteminin (veya yöntem iyileştirmesinin) performansını popüler test problemleri ile ölçmektedir. Bu test problemleri (i) bazı düğüm noktalarının (belirli veya tüm yönlerde) hareket etmesine izin vermemek, (ii) bazı düğümlerin birlikte hareket etmesini zorunlu kılmak, (iii) bazı çubukların kullanılmasına ve/veya kaldırılmasına izin vermemek ve (iv) çubukların ait oldukları kesit gruplarını önceden belirlemek gibi basitleştirmeler kullanılarak tasarlanmıştır; bu sebeple gerçek dünya tasarım problemlerini temsil edememektedirler. Ek olarak, bu problemlerin çoğunda sadece yapısal kısıtlar göz önünde bulundurulur ve yapım kısıtları ihmal edilir. Yapısal kısıtlar kullanılan malzeme (gerilme, şekil değiştirme vb.) ve taşıyıcı sistem davranışı (düğüm yer değiştirme, global burkulma vb.) ile ilgilidir. Yapım kısıtları ise sistemde kesişen elemanların bulunmaması ve bir düğüme çok sayıda elemanın bağlanmaması gibi sistemin imal edilmesini mümkün kılan kısıtlardır. Bu çalışmada düzlem kafes sistemlerin bilgisayar ile otomatik tasarımının gerçek hesaplama yükü tartışılmıştır. Burada “gerçek hesaplama yükü” ifadesi ile anlatılmak istenen, gerçek dünya problemlerinde olmayan basitleştirici kısıtların bulunmadığı ve yapısal kısıtların yanında yapım kısıtlarının da hesaba katıldığı tasarım problemlerine yakın-optimal çözümler aramak için harcanan işlem gücüdür. Kafes sistemlerin optimizasyonu için uygun olduğu daha önce yapılan çalışmalarla gösterilmiş parametresiz bir metasezgisel algoritma kullanılarak sayısal deneyler yapılmış ve sonuçlar tartışılmıştır.

**Anahtar Kelimeler:** Otomatik tasarım, Kafes sistem, Optimizasyon, Metasezgisel, Yapım kısıtları

## SUMMARY

Computer-automated design is the realization of a design by a computer without human intervention. Studies on simultaneous size, shape, and topology optimization show that computer-automated design is preparing to replace computer-aided design. Most studies presented on simultaneous size, shape, and topology optimization measure the performance of their proposed optimization method (or method improvement) with popular test problems. These test problems (i) do not allow some nodes to move (in certain or all directions), (ii) require some nodes to move together, (iii) do not allow some members to be used and/or removed, and (iv) the members belong to designed using simplifications such as predetermining cross-section groups; therefore, they cannot represent real-world design problems. In addition, in most of these problems, only structural constraints are considered, and construction constraints are neglected. Structural constraints are related to the material used (stress, strain, etc.) and the behavior of the structural system (node displacement, global buckling, etc.). On the other hand, construction constraints are the constraints that make it possible to manufacture the system, such as the absence of intersecting elements in the system and not connecting many elements to a node. In this study, the real computational load of computer-automated design of plane truss systems is discussed. What is meant by the expression “real computational load” here is the processing power spent in searching for near-optimal solutions to design problems where there are no simplistic constraints that are not found in real-world problems and where construction constraints are taken into account as well as structural constraints. Numerical experiments were carried out using a parameterless metaheuristic algorithm, which has been shown by previous studies to be suitable for the optimization of truss systems, and the results are discussed.

**Keywords:** Automatic design, Truss system, Optimization, Metaheuristic, Construction constraints

## İÇİNDEKİLER

	<u>Sayfa</u>
<b>ÖZET</b> .....	<b>vi</b>
<b>SUMMARY</b> .....	<b>vii</b>
<b>TEŞEKKÜR</b> .....	<b>viii</b>
<b>İÇİNDEKİLER</b> .....	<b>ix</b>
<b>ŞEKİLLER DİZİNİ</b> .....	<b>xi</b>
<b>ÇİZELGELER DİZİNİ</b> .....	<b>xiii</b>
<b>1. GİRİŞ VE AMAÇ</b> .....	<b>1</b>
<b>2. LİTERATÜR ARAŞTIRMASI</b> .....	<b>6</b>
2.1 2016 Yılına Ait Çalışmalar .....	6
2.2 2017 Yılına Ait Çalışmalar .....	7
2.3 2018 Yılına Ait Çalışmalar .....	9
2.4 2019 Yılına Ait Çalışmalar .....	10
2.5 2020 Yılına Ait Çalışmalar .....	12
2.6 2021 Yılına Ait Çalışmalar .....	13
2.7 İmalat Kısıtlarını Göz Önünde Bulunduran Çalışmalar .....	15
<b>3. YÖNTEM</b> .....	<b>21</b>
3.1 Geliştirme Platformu.....	22
3.2 Kullanılan Optimizasyon Algoritması .....	24
3.2.1 Sayısal örnek.....	26
<b>4. SAYISAL ÖRNEKLER</b> .....	<b>31</b>
4.1 Baz Yapı Yöntemi.....	31
4.2 Kısıtlar.....	32
4.2.1 İmalat kısıtları .....	35
4.3 Amaç ve Ceza Fonksiyonları .....	39
4.4 Kullanılan Test Problemleri .....	42

**İÇİNDEKİLER (devam)**

	<b><u>Sayfa</u></b>
4.4.1 4 düğüm noktalı test problemi .....	43
4.4.2 6 düğüm noktalı test problemi .....	44
4.4.3 8 düğüm noktalı test problemi .....	46
<b>5. TASARIM VEKTÖRÜ .....</b>	<b>48</b>
<b>6. BULGULAR VE TARTIŞMA .....</b>	<b>51</b>
6.1 4 Düğüm Noktalı (Düğüm Noktaları Sabit) Test Problemi Sonuçları .....	52
6.2 6 Düğüm Noktalı (Düğüm Noktaları Sabit) Test Problemi Sonuçları .....	55
6.3 6 Düğüm Noktalı (Düğüm Noktaları Hareketli) Test Problemi Sonuçları .....	59
6.4 8 Düğüm Noktalı (Düğüm Noktaları Sabit) Test Problemi Sonuçları .....	63
6.5 8 Düğüm Noktalı (Düğüm Noktaları Hareketli) Test Problemi Sonuçları .....	67
<b>7. SONUÇ VE ÖNERİLER .....</b>	<b>73</b>
<b>KAYNAKLAR DİZİNİ .....</b>	<b>75</b>

## ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
1.1. Boyut Optimizasyonu (a) boyut optimizasyonu öncesi, (b) boyut optimizasyonu sonrası .....	1
1.2. Şekil Optimizasyonu (a) şekil optimizasyonu öncesi, (b) şekil optimizasyonu sonrası.....	2
1.3. Topoloji Optimizasyonu (a) topoloji optimizasyonu öncesi, (b) topoloji optimizasyonu sonrası .....	2
1.4. Kafes sistem .....	3
2.1. Sadece yapısal kısıtları sağlayan bir kafes sistem yapısı .....	15
3.1. Jaya algoritmasının akış şeması .....	25
4.1. Örtüşen ve kesişen eleman kısıtı .....	35
4.2. Elemanlar arasındaki açı kısıtı a) her iki eleman için ortak olan bir düğüm noktasında b) elemanların kesiştiği bir noktada küçük açı.....	36
4.3. Bir düğüme bağlı eleman kısıtı .....	37
4.4. İki düğüm arasındaki minimum mesafe kısıtı .....	37
4.5. 4 düğüm noktalı kafes sistem için baz yapısı.....	43
4.6. 6 düğüm noktalı kafes sistem için baz yapısı.....	44
4.7. 8 düğüm noktalı kafes sistem için baz yapısı.....	46
5.1. 4 düğüm noktalı kafes sistem için baz yapısı.....	48
5.2. 4 düğüm noktalı kafes sistem için bulunan en iyi tasarım.....	48
6.1. 4 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı .....	53
6.2. 4 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım .....	53
6.3. 4 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım .....	54
6.4. 6 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı .....	56
6.5. 6 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım .....	57
6.6. 6 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım .....	57

## ŞEKİLLER DİZİNİ (devam)

<u>Sekil</u>	<u>Sayfa</u>
6.7. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı.....	60
6.8. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım.....	61
6.9. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım.....	61
6.10. 8 düğüm noktalı test problemi (düğüm noktaları sabit) için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı.....	64
6.11. 8 düğüm noktalı test problemi (düğüm noktaları sabit) için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım.....	65
6.12. 8 düğüm noktalı test problemi (düğüm noktaları sabit) için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım.....	65
6.13. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı.....	69
6.14. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım.....	69
6.15. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım.....	70

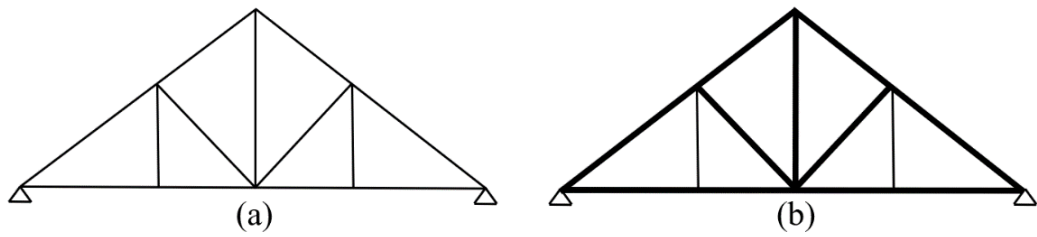
## ÇİZELGELER DİZİNİ

<u>Cizelge</u>	<u>Sayfa</u>
2.1. Son 5 yıldaki çalışmaların dikkate aldığı kısıtlar .....	13
2.2. Literatürde dikkate alınan imal edilebilirlik kısıtları.....	19
3.1. Python kodunda yararlanılan kütüphaneler.....	22
4.1. Optimizasyonda kullanılan eşit kollu korniyer alanları.....	32
4.2. Çalışmada dikkate alınan imalat kısıtları .....	38
5.1. 53 ve 60 numaralı profillere ait veriler.....	49
6.1. İzin verilen analiz sayısı.....	51
6.2. 4 düğüm noktalı (düğüm noktaları sabit) test problemi optimizasyon sonuçları .....	52
6.3. 4 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait veriler .....	54
6.4. 4 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait eleman kesit alanları .....	55
6.5. 6 düğüm noktalı (düğüm noktaları sabit) test problemi optimizasyon sonuçları .....	55
6.6. 6 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait veriler .....	58
6.7. 6 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarımın eleman kesit alanları. 58	58
6.8. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) optimizasyon sonuçları.....	59
6.9. 6 düğüm noktalı (düğüm noktaları hareketli) test problemi en iyi tasarıma ait veriler.....	62
6.10. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) en iyi tasarımın eleman kesit alanları.....	62
6.11. 8 düğüm noktalı test problemi (düğüm noktaları sabit) optimizasyon sonuçları .....	63
6.12. 8 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait veriler .....	66
6.13. 8 düğüm noktalı test problemi (düğüm noktaları sabit) en iyi tasarımın eleman kesit alanları .....	67
6.14. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) optimizasyon sonuçları.....	68
6.15. 8 düğüm noktalı (düğüm noktaları hareketli) test problemi en iyi tasarıma ait veriler .....	70
6.16. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) en iyi tasarımın eleman kesit alanları.....	71

## 1. GİRİŞ VE AMAÇ

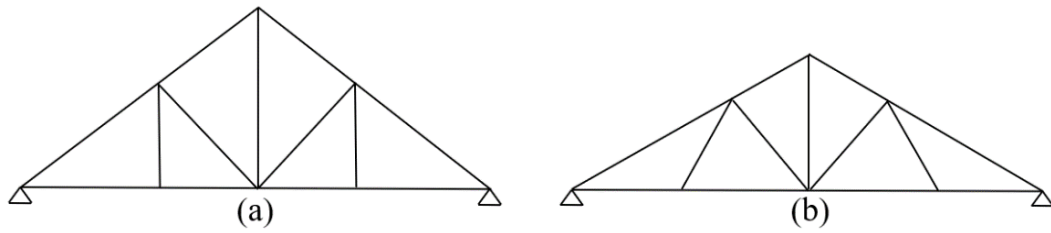
Yapısal optimizasyon, farklı yükleme koşulları ve belirlenen kısıtlar altında en iyi çözümü arama işlemidir. Son yıllarda, bu alan giderek daha fazla gelişmiş ve bilimsel araştırmaların yanı sıra pratik mühendislik uygulamalarında da olağanüstü başarılar elde etmiştir. Kafes sistem, levha ve kiriş gibi birçok yapı bileşenine başarıyla uygulanmıştır. Bu yapı bileşenleri arasından kafes sistemler, Bellagamba ve Yang'ın (1981) frekans kısıtları altında kafes sistemlerin boyut ve şekil optimizasyonu ile ilgili öncü makaleyi yayınlamasından sonra birçok araştırmacının dikkatini çekmiştir (Lieu, 2022).

Literatürde mevcut olan yapısal optimizasyon türleri üç gruba ayrılabilir: (1) boyut optimizasyonu, (2) şekil optimizasyonu ve (3) topoloji optimizasyonu. Bu optimizasyon türlerinden ilki, yapıyı oluşturan bileşenlerin optimum değerlerini bulmakla ilgilenen boyut optimizasyonudur (Şekil 1.1). Örneğin bir çatı kafes sisteminde açıklığın kaç metre olacağı ve ne kadar yük taşıyacağı problemin en başında bilinmektedir ama bu sistemde kullanılacak olan elemanların kesit alanları ve uzunluklarının optimum olacak şekilde belirlenebilmesi için boyut optimizasyonu yapılması gereklidir.



Şekil 1.1. Boyut Optimizasyonu (a) boyut optimizasyonu öncesi, (b) boyut optimizasyonu sonrası

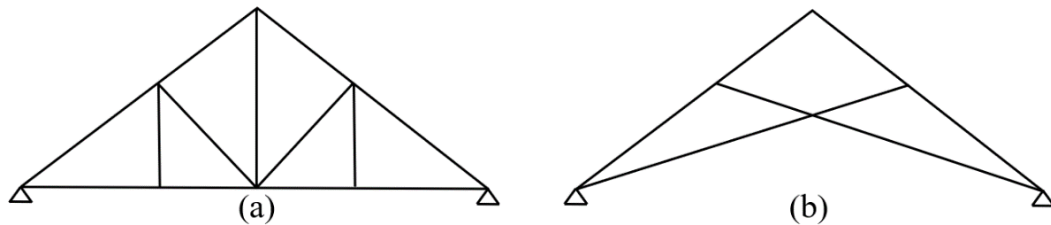
Şekil optimizasyonu ise bir yapının mümkün olan en iyi geometrisini bulmayı amaçlar (Şekil 1.2). Şekil optimizasyonunda düğüm noktalarının koordinatları, optimizasyon sırasında değiştirilecek olan tasarım değişkenleri olarak kabul edilir (Bakhtary vd., 1996). Şekil tasarım değişkenleri belirli bir aralıktaki herhangi bir değeri



Şekil 1.2. Şekil Optimizasyonu (a) şekil optimizasyonu öncesi, (b) şekil optimizasyonu sonrası

alabileceğinden, şekil optimizasyonu genellikle sürekli tasarım değişkenleri içerir (Dillen vd., 2021).

Topoloji optimizasyonu, elemanların sistemden çıkarılması ya da yeni elemanların sisteme eklenmesi ile gerçekleştirilir (Şekil 1.3). Şekil optimizasyonuna kıyasla boyut ve topoloji optimizasyonları genellikle ayrık tasarım değişkenleri içerir.



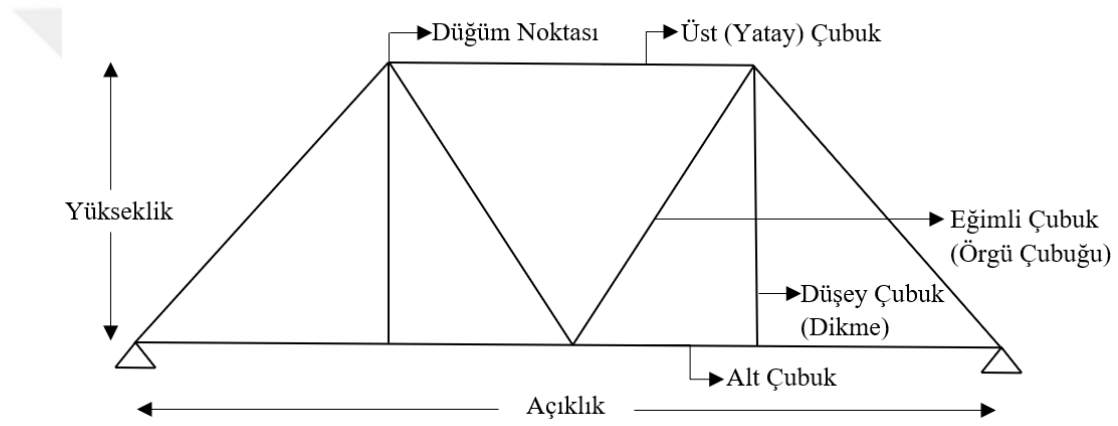
Şekil 1.3. Topoloji Optimizasyonu (a) topoloji optimizasyonu öncesi, (b) topoloji optimizasyonu sonrası

Yukarıdaki üç optimizasyon türünden ayrı ayrı bahsedilse de kafes sistem yapılarını en iyi şekilde tasarılmanın en etkili yolu, üç optimizasyon yönteminin de aynı anda ele alındığı eşzamanlı optimizasyon (tek aşamalı optimizasyon) yöntemidir.

Kafes sistemler, sürekli yükler altında düğümlere bağlanan, çekme veya basınca maruz kalan doğrusal elemanlardan meydana gelen iki veya üç boyutlu yapılardır. Kafes sistemlerin oluşturulmasında kullanılan temel geometrik yapı üçgendir. Bunun sebebi mekanik mukavemeti ve stabilitesinin yüksek olması, aynı zamanda kuvvet etkisi altında

formunun bozulmamasıdır. Kafes sistemleri oluşturan doğrusal elemanlar (çubuklar) narindirler ve eksenine dik doğrultudaki yükleri taşıyamazlar. Bu yüzden sistem üzerindeki yükler doğrusal elemanların birleşim yerlerine (düğüm noktalarına) aktarılarak tüm yapıya dağıtılır. Bu da demektir ki kafes sistemlerin tasarımları istikrarlı bir ağırlık dağılımına izin verir.

Tüm elemanlar ve uygulanan kuvvetler aynı düzlemde olduğunda, sistem bir düzlem veya 2 boyutlu kafes sistem olarak adlandırılır (Şekil 1.4).



Şekil 1.4. Kafes sistem

Taşıyıcı sistemlerin açıklıkları büyüdükçe betonarme elemanların kesitleri de büyür ve çözüm ekonomik olmamaya başlar. Bu sebeple betonarme taşıyıcı sistemler yerini kafes ve çerçeve sistemlerine bırakırlar. Ayrıca kafes sistemler büyük yüklere ve yüksek basınçlara karşı direnebilen yapılardır. Kafes sistemlerin bir diğer güzel özelliği ise tasarımcıların, yapıların boyutlarını genişletmesine ve çeşitli şekiller oluşturmaya izin vermesidir. Gerçek dünya yapılarında kafes sistemler en çok köprüler, havaalanı terminaleri, endüstriyel binalar, stadyum çatıları, oditoryumlar ve diğer eğlence binaları gibi çok uzun açıklıklara gereksinim olan yapılarda kullanılırlar. Ayrıca kafes sistemlerin vinçler ve asansörler gibi birçok makinede ve uçak kanat ve gövdelerinde kullanıldığı da bilinmektedir.

Yapı mühendisliğinde optimizasyonun amacı belirli kısıtlar altında uygulanabilir tasarımlar içerisindeki en iyi (optimum) tasarıma ulaşabilmektir. Yapı mühendisliği problemleri tek bir olurlu çözümü olacak şekilde kısıtlandırılmamıştır; bu sebeple optimale yakın çözümlere ulaşabilmek için çok sayıda tasarım adayının değerlendirilmesine ihtiyaç vardır. Tatmin edici çözümlere ulaşabilmek için çoğunlukla binlerce tasarımın üretilip analiz edilmesi gerekmektedir. Kullanıcının bir yazılımın modelleme arayüzünü kullanarak taşıyıcı sistem adayları oluşturması ve bu adayların yazılım ile değerlendirilip kısıtları ne kadar ihlal ettiğinin belirlenmesi şeklinde ilerleyen bir bilgisayar destekli tasarım süreci ile optimale yakın bir tasarımın keşfedilmesi imkansızca çok yakındır. Bu sebeple bilgisayar destekli tasarımın yerini bilgisayar ile otomatik tasarıma bırakacağı öngörülmektedir. Bilgisayar ile otomatik tasarım bir tasarımın bilgisayar tarafından insan müdahalesi olmadan gerçekleştirilmesidir.

Kafes sistemlerin yapısal kısıtlar altında optimizasyonu gerilme, yer değiştirme, burkulma, frekans, kinematik stabilite, global stabilite ve yerel stabilite gibi kısıtlar göz önünde bulundurularak gerçekleştirilebilir. Literatürde kafes sistemlerin yapısal kısıtlar altında boyut, şekil ve topoloji optimizasyonu konusunda çok sayıda çalışma bulunmaktadır. Bu çalışmada bu konunun ne kadar ilgi gördüğünden bahsedilmeyecektir. Diğer taraftan, imalat (yapım) kısıtları ile kafes sistem optimizasyonu konusunda ulaşılabilen çalışma sayısı dikkat çekici derecede azdır. Araştırmacıların kafes sistemlerin optimizasyonu hakkında yaptıkları çalışmalarda imalat kısıtı olarak çoğunlukla örtüşen (Kanno 2018 a; Kanno 2018 b) ve kesişen elemanlar (Cui vd. 2018) dikkate alınmıştır. Yakın zamanda yapılmış bazı çalışmalarda ise örtüşen ve kesişen elemanların yanında yapı elemanı uzunluğu, düğüm sayısı ve elemanlar arası açı sınırları da (Fairclough ve Gilbert 2020) imalat kısıtı olarak göz önünde bulundurulmuştur.

Mevcut çalışmaların geneli klasik optimizasyon problemlerini kullanarak sadece yapısal kısıtlar altında optimizasyon problemiyle ilgilenmiştir. Bu yüzden gerçek dünya optimizasyon problemlerinin hesaplama yükü hakkında herhangi bir fikir vermezler.

Bu çalışmada kafes sistemlerin bilgisayar ile otomatik tasarımının hesaplama yükü araştırılmıştır. Burada “bilgisayar ile otomatik tasarım” ifadesinden anlaşılması gereken “kafes sistemlerin hiçbir basitleştirici kural kullanmadan, yapısal ve imalat kısıtları altında

eşzamanlı boyut, şekil ve topoloji optimizasyonu”dur. Konu ile ilgili çalışmalarda tanıtılan test problemlerini kullanmak yerine gerçek işlem yükünü tespit edebilmek için yazar tarafından üretilen problemler üzerinde çalışılmıştır. Çubuk gerilmeleri ve düğüm yer değiştirmeleri yapısal kısıtları ve en fazla kesit çeşidi, örtüşen elemanlar, kesişen elemanlar, bir düğüme bağlı eleman sayısı, yapı elemanı uzunluğu, en fazla düğüm sayısı ve elemanlar arasındaki en küçük açı imalat kısıtları dikkate alınarak problemler çözülmüş ve sonuçlar tartışılmıştır. Aynı zamanda bir kafes sistemin imal edilebilir olması için gereken tüm kısıtların dikkate alınması halinde ne gibi durumlar ile karşılaşılacağı, hesaplama yükü ve yakınsama analizi hakkında edinilen bilgilere yer verilmiştir. Yapısal kısıtlara imalat kısıtları da eklendiğinde problemin ne kadar zorlaştığı ve ne kadar işlem gücü gerektiği konusunda araştırmacılara fikir verilmesi ve sonraki araştırmalar için bir ön çalışma yapılması amaçlanmıştır.

## 2. LİTERATÜR ARAŞTIRMASI

Kafes sistemlerin yapısal kısıtlar altında optimizasyonu gerilme, yer değiştirme, burkulma, frekans, kinematik stabilite, global stabilite ve yerel stabilite gibi kısıtlar göz önünde bulundurularak gerçekleştirilebilir. Literatürde kafes sistemlerin yapısal kısıtlar altında boyut, şekil ve topoloji optimizasyonu konusunda çok sayıda çalışma bulunmaktadır. Son 5 yıla ait kafes sistem optimizasyonu hakkında bulunan çalışmalardan aşağıda bahsedilmiştir.

### 2.1 2016 Yılına Ait Çalışmalar

Savsani vd. (2016), kafes sistemlerin eş zamanlı boyut ve topoloji optimizasyonu için MS-TLBO (Modified Subpopulation Teaching Learning Based Optimization) algoritmasını kullanarak gerilme, yer değiştirme, burkulma, frekans ve kinematik stabilite kısıtları altında düzlem ve uzay kafes kirişlerin optimum tasarımlarını incelemiştir.

Mortazavi ve Toğan (2016) yaptıkları çalışmada, eşzamanlı boyut, şekil ve topoloji optimizasyonunu gerçekleştirmek için iPSO (integrated Particle Swarm Optimization) adı verilen bir algoritma önermişlerdir. Çalışmada düğüm koordinatları, eleman kesitleri, yer değiştirme ve gerilme kısıtları dikkate alınmış ve uygulanmıştır. Önerilen optimizasyon yöntemi çoklu yük koşullu problemleri çözme ve standart tasarım kodlarına yani AISC'ye (American Institute of Steel Construction) göre kısıtları işleme yeteneği açısından test edilmiştir.

Milajić vd. (2016), belirli bir kafes yapısının optimum tasarımında kullanılmak üzere çok sayıda değişkenle başa çıkabilecek verimli ve etkili bir optimizasyon aracı bulmak için BB-BC (Big Bang - Big Crunch) algoritmasının farklı varyantlarının uygulanabilirliğini ve performansını araştırmışlardır.

Maheri vd. (2016), kafeslerin boyut ve topolojisini aynı anda optimize etmek için gerilme ve yer değiştirme kısıtları altında GA (Genetic Algorithm) ve PSO (Particle Swarm Optimization) algoritmasını birleştirerek oluşturdukları bir hibrit algoritmayı kullanmışlardır.

Fiore vd. (2016), DE (Differential Evolution) algoritmasını kullanarak düzlemsel içi boş kesitli çelik kafes kirişlerin boyut, şekil ve topoloji optimizasyonunu ele almışlardır. Bu çalışmanın temel amacı, pratik amaçlar için uygulanabilen, boru şeklindeki düzlemsel kafes kirişler için uygulanabilir bir optimizasyon problemi formüle etmektir. Tasarım kısıtları, Eurocode 3 yönergelerine uygun olarak çelik yapı tasarım kurallarına dayanmaktadır.

Ahrari ve Deb (2016), karmaşık kafes yapılarının gerilme, yer değiştirme, burkulma ve narinlik kısıtları altında eşzamanlı boyut, şekil ve topoloji optimizasyonu için FSD-ES (Fully Stressed Design Based on Evolution Strategies) algoritmasının FSD-ES-II adı verilen geliştirilmiş bir versiyonunu kullanmışlardır.

## 2.2 2017 Yılına Ait Çalışmalar

Madah ve Amir (2017), kafes sistemlerin gerilme ve yer değiştirme kısıtı altında boyut ve topoloji optimizasyonu için burkulmaya dair herhangi bir kısıt olmadan tüm burkulma hususlarını hesaba katabilen bir formülasyon geliştirmiştir. Geliştirilen formülasyonda burkulma, çok sayıda kısıt uygulamak yerine geometrik doğrusal olmayan bir analiz kullanılarak dolaylı olarak dikkate alınmıştır.

Zhou vd. (2017), hem sürekli hem de ayrık tasarım değişkenlerini dikkate alarak, belirsiz parametrelerin varlığında düzlem ve uzay kafes sistemlerin boyut, topoloji ve geometri optimizasyonunu ele almışlardır. Genetik algoritma, sürekli ve ayrık tasarım değişkenlerini eşzamanlı olarak ele almak için geliştirilmiştir. Ayrıca optimizasyon sürecinde gerilme ve yer değiştirme kısıtları da dikkate alınmıştır.

Li ve Khandelwal (2017) bu çalışmada geometrik olarak doğrusal olmayan kafes sistemlerin topoloji optimizasyonu, kritik yük kısıtları olan ve olmayan optimizasyon formülasyonları kullanılarak gerçekleştirilmiştir. Aynı zamanda kafes sistemlerin stabilite sorunlarını ele almak için bazı yaklaşımlar önerilmiştir.

Kaveh ve Zolghadr (2017) bu makalede, frekans kısıtları altındaki kafes sistemlerin boyut ve şekil optimizasyonu için CPA (Cyclical Parthenogenesis Algorithm) adı verilen yeni geliştirilmiş bir metasezgisel yöntem kullanmışlardır.

Ohsaki ve Hayashi (2017) bu çalışmalarında, düzlem ve uzay kafes sistemlerin gerilme kısıtı altında optimizasyonu için yeni bir eşzamanlı geometri ve topoloji optimizasyonu yöntemi sunmuşlardır. Eksenel kuvvetin, eleman uzunluğuna oranı olan kuvvet yoğunluğu tasarım değişkeni olarak dikkate alınmış ve yakın aralıklı düğümlerinin varlığından kaynaklanan zorluklar önlenmeye çalışılmıştır.

Assimi vd. (2017) kafes yapıların eşzamanlı boyut ve topoloji optimizasyonu için bir genetik programlama yaklaşımı sunmuşlardır. Sayısal örnekler, yapının kinematik stabilite, elemanlarda izin verilen maksimum gerilme ve düğümlerde izin verilen maksimum yer değiştirme kısıtlamalarına tabi tutulmuştur.

Chen vd. (2017) düzen optimizasyonunu, kafes yapılarını boyut, şekil ve topoloji değişkenlerinin birleşik etkisi altında optimize etmek için bir prosedürdür diyerek tanımlamışlardır. Gerilme ve yer değiştirme kısıtları altında kafes sistemlerin ağırlığını en aza indirmek için sürekli şekil değişkenleri ve şekil hassasiyetlerini kullanan IGATA (Improved Genetic Algorithm with Two-Level Approximation) algoritmasını kullanmışlardır. Sürekli boyut/şekil değişkenleri ve ayrık topoloji değişkenlerini içeren aynı zamanda şekil değişkenleri değerlerinin düğüm koordinatlarına eşit olduğu kapsamlı bir model oluşturulmuştur.

Mortazavi vd. (2017), kafes sistemlerin boyut ve düzen optimizasyonu üzerinde iPSO algoritmasının performansını araştırmışlardır. İPSO, optimizasyon kısıtlarının üstesinden gelebilmek için standart PSO algoritmasının geliştirilmiş halidir. Son algoritmanın performansı, çeşitli kısıtlar (yani düğüm yer değiştirmeleri, eleman

gerilmeleri ve burkulma) altında, karışık tasarım arama uzayları (yani hem ayrık hem de sürekli değişkenlerle) dahil olmak üzere bir dizi iyi bilinen kafes sistem problemleri üzerinde test edilmiştir.

Savsani vd. (2017) ayrık tasarım değişkenlerine sahip kafes sistemlerin boyut ve topoloji optimizasyonunu çözmek için dört temel (Teaching-Learning-Based Optimization (TLBO), Heat Transfer Search (HTS), Water Wave Optimization (WWO), Passing Vehicle Search (PVS)) ve dört değiştirilmiş (Modified TLBO (MTLBO), Modified HTS (MHTS), Modified WWO (MWWO), Modified PVS (MPVS)) metasezgisel optimizasyon algoritmasının performansını karşılaştırmışlardır. Tüm problemler gerilme, yer değiştirme, burkulma, frekans ve kinematik stabilite kısıtları dikkate alınarak incelenmiştir.

Mortazavi ve Toğan (2017) kafes sistemlerin boyut ve düzen optimizasyonu için geliştirmiş oldukları iPSO algoritmasını kullanarak frekans, gerilme, yer değiştirme ve stabilite kısıtları altında kafes sistemlerin optimum tasarımlarını incelemiştir.

### **2.3 2018 Yılına Ait Çalışmalar**

Kaveh ve Zolghadr (2018) çalışmalarında, frekans kısıtları altında yapısal optimizasyon sorununun farklı örneklerini ele almak için kullanılan yirmi farklı metasezgisel optimizasyon algoritmasını incelemiştir.

Panagant ve Bureerat (2018) kafes sistemlerin gerilme, burkulma ve yer değiştirme kısıtları altında eşzamanlı boyut, şekil ve topoloji optimizasyonu için yeni bir hibrit optimizasyon algoritması geliştirmişlerdir. Yeni algoritma, GWO (Grey Wolf Optimization) algoritmasının, DE (Differential Evolution) algoritmasına entegre edilmesiyle elde edilirken, FSD'de (Fully Stressed Design) geliştirilen bu yeni algoritma ile birleştirilmiştir.

Tejani vd. (2018) düzlem ve uzay kafes sistemlerin gerilme, yer değiştirme ve kinematik stabilite kısıtları altında eş zamanlı boyut, şekil ve topoloji optimizasyonunu

dört temel metasezgisel (TLBO, HTS, WWO ve PVS) ve dört geliştirilmiş metasezgisel (Improved Teaching–Learning-Based Optimization (ITLBO), Improved Heat Transfer Search (IHTS), Improved Water Wave Optimization (IWWO), Improved Passing Vehicle Search (IPVS)) algoritma kullanarak incelemişlerdir.

Degertekin vd. (2018) kafes sistemlerin gerilme, burkulma ve yer değiştirme kısıtları altında boyut ve düzen optimizasyonu için JA (Jaya Algorithm) adı verilen metasezgisel bir yöntem kullanmışlardır. JA'nın temel özelliği, algoritmaya özgü parametrelerin ayarlanmasını gerektirmemesidir. Jaya, popülasyondaki en iyi tasarıma yaklaşmaya ve aynı anda en kötü tasarımdan uzaklaşmaya çalışan parametresiz bir algoritmadır.

#### **2.4 2019 Yılına Ait Çalışmalar**

Yancang ve Zhen (2019) kafes sistem optimizasyonu için daha etkili bir yöntem bulmak amacıyla, geliştirilmiş BA (Bat Algorithm) algoritmasını önermişlerdir. Önerilen bu algoritma kafes sistemlerin topoloji optimizasyonu için kullanılmış ve optimizasyon sonuçları diğer algoritmalarından elde edilen sonuçlarla karşılaştırılmıştır. Ayrıca sayısal örnekler gerilme ve yer değiştirme kısıtları altında incelenmiştir.

Madah ve Amir (2019) bu çalışmalarında burkulmaya dirençli kafes sistemlerin boyut, şekil ve topoloji optimizasyonunu ele almaktadırlar. Önerilen optimizasyon problemi formülasyonunda burkulma, çok sayıda kısıt empoze etmek yerine, geometrik doğrusal olmayan bir analiz kullanılarak dolaylı olarak hesaba katılmıştır.

Degertekin vd. (2019) gerilme, yer değiştirme, burkulma ve stabilite kısıtları altında kafes sistemlerin ayırık boyutlandırma, düzen ve topoloji optimizasyonu için Jaya algoritmasının DAJA (Discrete Advanced Jaya Algorithm) olarak adlandırılan yeni bir formülasyonunu geliştirmişlerdir.

Tejani vd. (2019), kafes sistemlerin optimizasyonu için PVS algoritmasının üç değiştirilmiş versiyonunu önermişlerdir. Önerilen algoritmalar gerilme, yer değiştirme, burkulma, doğal frekans ve kinematik stabilite kısıtları altında 2 ve 3 boyutlu kafes sistemlerin eşzamanlı boyut ve topoloji optimizasyonu için test edilmiştir.

Techasen vd. (2019) kafes sistemlerin gerilme kısıtı altında eşzamanlı boyut, şekil ve topoloji optimizasyonunu çözmek için bir dizi çok amaçlı evrimsel algoritma kullanarak elde edilen çözümleri karşılaştırmışlardır. Aynı zamanda bu çalışma kafeslerin güvenilirlik optimizasyonu ve eşzamanlı topoloji, şekil ve boyutlandırma optimizasyonunun birleşimi için bir ilk çalışma olarak kabul edilmiştir.

Assimi vd. (2019), MSOGP (Multi-objective Structural Optimization Genetic Programming) adı verilen bir yaklaşımı kullanarak kafes sistemlerin gerilme, yer değiştirme ve kinematik stabilite kısıtları altında eşzamanlı boyut ve topoloji optimizasyonunu incelemişlerdir.

Kaveh ve Mahjoubi (2019) frekans kısıtları altındaki kafes sistemlerin boyut ve düzen optimizasyonu için SPO (Spiral Optimization) algoritmasının geliştirilmiş bir sürümü olan HSPO (Hypotrochoid Spiral Optimization) algoritmasını önermişlerdir.

Hayashi ve Ohsaki (2019) başka bir çalışmalarında önerdikleri FDM (Force Density Method) yöntemine dayalı olarak kafes sistemlerin topolojisi ve geometrisinin eşzamanlı optimizasyonu için bir Grasshopper (Rhinoceros yazılımı için bir algoritmik modelleme eklentisi) bileşeni geliştirmişlerdir. Amaç ve kısıtlama fonksiyonları yalnızca kuvvet yoğunluğunun açık fonksiyonları olarak ifade edilmiştir. Düğüm konumlarını kontrol etmek için serbest biçimli bir tasarım yüzeyi sunmuşlardır.

Weldeyesus vd. (2019) doğrusal olmayan kinematik uyumluluk kısıtını gevşeterek doğrusal yarı-tanımlı programlama ile global stabilite kısıtları altında kafes sistemlerin düzen optimizasyonu problemini çözmüşlerdir.

## 2.5 2020 Yılına Ait Çalışmalar

Weldeyesus vd. (2020) bu çalışmalarında, doğrusal olmayan yarı tanımlı bir programlama problemi olarak, koyulan global stabilite kısıtları formülasyonu ile mevcut bir kafes sistem topoloji optimizasyonuna geometri optimizasyonunu dahil ederek incelemiştir.

Mortazavi (2020)a bu çalışmasında metasezgisel optimizasyon tekniklerinin, yapısal boyut ve topoloji optimizasyon problemlerinin üstesinden gelme kabiliyetini geliştirmek için yeni bir yardımcı bulanık karar mekanizmasının tanıtılmasıyla ilgilidir. Sayısal örnekler gerilme, yer değiştirme ve frekans kısıtları altında incelenmiştir.

Mortazavi'nin (2020)b daha önceden geliştirdiği IFSA (Interactive Fuzzy Search Algorithm) adı verilen bir metasezgisel algoritma bu çalışmada, birden fazla doğal frekans kısıtına sahip kafes yapılarının boyut ve düzen optimizasyonunu çözmek için ilk kez kullanılmıştır. IFSA, kendi kendine uyarlanabilir ve parametresiz bir arama algoritmasıdır.

Kumar vd. (2020) düzlem ve uzay kafes sistemlerin gerilme, frekans, yer değiştirme, burkulma ve kinematik stabilite kısıtları altında eşzamanlı boyut ve topoloji optimizasyonunu çözmek için IALO (Improved Ant Lion Optimizer), IDA (Improved Dragonfly Algorithm), IWOA (Improved Whale Optimization Algorithm), IHTS (Improved Heat Transfer Search) ve ITLBO (Improved Teaching Learning Based Optimization) olarak adlandırılan beş metasezgisel algoritmanın geliştirilmiş sürümlerini kullanmışlardır.

Reintjes ve Lorenz (2020) matematiksel optimizasyon, Bilgisayar Destekli Tasarım (CAD), Sonlu Elemanlar Analizi (FEA) ve Katmanlı Üretim (AM)'in algoritma odaklı bir ürün tasarım sürecinde sistematik olarak nasıl birleştirileceğini araştırmıştır. Sayısal örnekler gerilme kısıtları altında incelenmiştir.

Kaveh ve Seddighian (2020), iki ve üç boyutlu kafes sistemlerin stabilite kısıtı altında eş zamanlı olarak malzeme, düğüm koordinatı, kesit alanı boyutu ve eleman bağlanabilirliğini optimize etmek için FA (Firefly Algorithm) algoritmasının yeni bir zenginleştirilmiş versiyonunu olan EFA (Enriched Firefly Algorithm) algoritmasını önermişlerdir.

## 2.6 2021 Yılına Ait Çalışmalar

El Bouzouiki vd. (2021) bu makalede, aynı olmayan hücelere ve değiştirilmiş bir FSD (Fully Stressed Design) / FUD (Fully Utilized Design) yaklaşımına dayanan yeni bir CA (Cellular Automata) algoritmasını sunmuşlardır. Yeni algoritma hem gerilme hem de yer değiştirme kısıtları dahil olmak üzere kafes giriş yapılarının minimum ağırlık optimizasyonu problemini çözmek için önerilmiştir.

Lemonge vd. (2021) hem sürekli hem de ayrık tasarım değişkenlerini dikkate alarak, gerilme, yer değiştirme, frekans ve stabilite kısıtları altında kafes sistemlerin boyut, şekil ve düzen optimizasyonunu incelemiştir. Optimizasyon için GDE3 (Third Evolution Step of Generalized Differential Evolution) adı verilen bir algoritma kullanmışlardır.

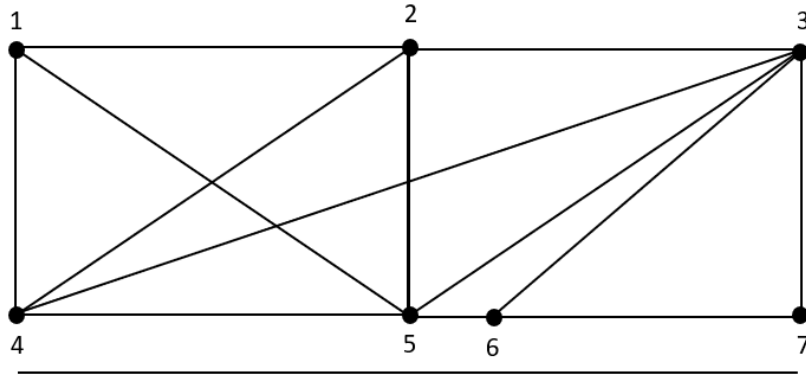
Çizelge 2.1. Son 5 yıldaki çalışmaların dikkate aldığı kısıtlar

	Gerilme	Yer Değiştirme	Burkulma	Frekans	Kinematik Stabilite	Stabilite (Global-Yerel)
Savsani vd. (2016)						
Mortazavi ve Toğan (2016)						
Maheri vd. (2016)						
Ahrari ve Deb (2016)						

Çizelge 2.1. Son 5 yıldaki çalışmaların dikkate aldığı kısıtlar (devam)

Madah ve Amir (2017)						
Zhou vd. (2017)						
Kaveh ve Zolghadr (2017)						
Ohsaki ve Hayashi (2017)						
Assimi vd. (2017)						
Chen vd. (2017)						
Mortazavi vd. (2017)						
Savsani vd. (2017)						
Mortazavi ve Toğan (2017)						
Kaveh ve Zolghadr (2018)						
Panagant ve Bureerat (2018)						
Tejani vd. (2018)						
Degertekin vd. (2018)						
Yancang ve Zhen (2019)						
Madah ve Amir (2019)						
Degertekin vd. (2019)						
Tejani vd. (2019)						
Techasen vd. (2019)						
Assimi vd. (2019)						
Kaveh ve Mahjoubi (2019)						
Weldeyesus vd. (2019)						
Weldeyesus vd. (2020)						
Mortazavi (2020)a						
Mortazavi (2020)b						
Kumar vd. (2020)						
Reintjes ve Lorenz (2020)						
Kaveh ve Seddighian (2020)						
El Bouzouiki vd. (2021)						
Lemonge vd. (2021)						

Çizelge 2.1’de görüldüğü üzere son 5 yılda yapılan çalışmalar gerilme, yer değiştirme, frekans vb. kısıtlar altında optimizasyonla çalışmıştır fakat imalat (yapım) kısıtları ile kafes sistem optimizasyonunu inceleyen makale sayısı son derece azdır. Sadece yapısal kısıtlar altında kafes sistemlerin imal edilebilmesi çoğunlukla imkansızdır.



Şekil 2.1. Sadece yapısal kısıtları sağlayan bir kafes sistem yapısı

Yukarıdaki şekilde (Şekil 2.1) üst üste gelen elemanlar fark edilebilmesi açısından birbirinden biraz uzağa çizilmiştir; aslında altta ayrı şekilde çizilmiş olan çubuk eleman 4 nolu ve 7 nolu düğüm noktaları arasındadır. Bu tasarımda yukarıda bahsedilen yapısal kısıtların tümü sağlansa dahi tez metninin ileriki kısımlarında detaylı olarak açıklanacak kesişen elemanlar, bir düğüme bağlanmış çok sayıda eleman, üst üste binen elemanlar, çok yakın düğümlerin bulunması gibi durumlardan kaynaklı olarak bu kafes sistemin imal edilebilmesi mümkün değildir. Bu çalışmalar teorik önem arz etmekle beraber optimizasyon sonucunda ortaya çıkan kafes sistem imal edilebilir değildir. Bir kafes sistemin imal edilebilmesi için hangi kısıtlar uygulanmalı ve literatürde hangi kısıtlar dikkate alınmıştır öğrenmek için yazarın imalat kısıtları ile ilgili ulaştığı makaleler aşağıda özetlenmiştir.

## 2.7 İmalat Kısıtlarını Göz Önünde Bulunduran Çalışmalar

Ayrık tasarım değişkenleri (kesit alanı, düğüm koordinatı ve topoloji bilgisi), eleman gerilme kısıtları, düğüm yer değiştirme kısıtları imalat kısıtından sayılmamıştır. Bu sebeple imalat kısıtı olarak söz edilen kısıtları öneren yazılar literatür taramasına alınmamıştır. Ayrıca global stabilite, lokal stabilite ve kinematik stabilite bozuklukları da imalat kısıtından sayılmamış ve kullanılan eleman çeşiti de imalat kısıtı olarak algılanmamalıdır.

Çoğu yayın kafes sistem topolojisini üyelerin bir kombinasyonu olarak ifade eder. Ancak bu yöntemlerin dezavantajı, ortaya çıkan topolojinin gereksiz veya diğer üyelerle örtüşen elemanlar içermesi olabilir. Bu sorunlara ek olarak, üretilen yapılar, yapısal olarak kararlı değildir. Kawamura vd. (2002), kafes sistem topolojisini bir üçgen kombinasyonu olarak ifade ederek bu sorunları çözen yeni bir yöntem önermişlerdir. Bununla birlikte, mevcut yöntem tasarım parametreleri olarak düğüm noktalarının dağılımını değil, üçgen öğeleri esas almaktadır. Olağan yöntemlerle karşılaştırıldığında yazarlar tarafından önerilen yöntem ile üretilen tasarımların gereksiz ve örtüşen elemanlara sahip olmadığı ve yapıların her zaman kararlı olduğu doğrulanmıştır.

Hamza vd. (2003) belirli bir sınıf düzlem kafes sistemleri (N-şekilli) ele almıştır. Çok sayıda güvenlik kısıtı kullanılan bu çalışmada imalat kısıtı olarak değerlendirilebilecek olan "kirişin tüm üst ve alt çubuk elemanlarının tek bir düz çizgi oluşturması" ve "aşıkların belirli bir aralıkta olması" kısıtlarını da değerlendirmişlerdir.

Ohsaki ve Katoh (2005) topoloji optimizasyonu problemini, gerilme kısıtları altında düğümlerin ve elemanların varlığını gösteren tamsayı değişkenleri kullanarak bir Karışık Tamsayı Programlama (Mixed Integer Programming) problemi olarak formüle etmişlerdir. Aynı zamanda kafes sistemlerin optimal topolojisinin üst ve alt sınır çözümlerini elde etmek için yazarlar tarafından bir dal ve sınır yöntemi önerilmiştir. Gerçekçi çözümler elde edebilmek için kesişen eleman kısıtı uygulanmış ve kesişen elemanların bulunmadığı çözümler elde edilmiştir.

Dominguez vd. (2006), genel olarak kafes sistem yapıları bazı temel bölümlerinin çoğaltılmasıyla elde edildiğini bu sebepten dolayı da tüm yapıyı optimize etmek yerine temel bölmelerin özelliklerini optimize ederek nihai optimum tasarıma ulaşılabileceğini savunmuşlardır. Gerçekçi optimal tasarımlar elde edebilmek için standart profillerin kesit alanları Amerikan Çelik Yapı Enstitüsü standartlarına uygun olacak şekilde alınmıştır.

Wang ve Ohmori (2010) yaptıkları bu araştırmada, gerilme, yer değiştirme ve düğüm noktasına bağlı elemanlar arasındaki açılı kısıtları altında GA kullanarak kafes optimizasyon şemasının gerçek yapısal tasarıma uygulanabilirliğini geliştirmişlerdir. İmal edilebilirlik açısından kesişen elemanların varlığından kaçınılmıştır. Bağlantı elemanları

(küre), yapının toplam ağırlığının belirli bir bölümünü oluşturduğundan dolayı optimizasyonda toplam ağırlığın tam olarak hesaplanabilmesi için bağlantı elemanlarının ağırlıkları da hesaplamalarda ayrıca dikkate alınmıştır. Yazarlar ayrıca küresel bağlantı elemanlarına çubuk elemanların birbirlerine temas etmeden bağlanabilmesi durumunu da göz önünde almışlardır.

Frans ve Arfiadi (2014) bu makalede çatı kafes sistemlerin boyut, şekil ve topoloji optimizasyonları için kesit alanları ve her düğüme bağlı minimum eleman sayısı ikili (binary) kodlu genetik algoritmalar kullanılarak optimize edilirken, düğüm noktalarının koordinatları gerçek kodlu genetik algoritmalar kullanılarak belirlenir. Optimizasyon sürecinde, çatı eğim açısı sabit tutulmuştur.

Mela (2014) bu çalışmasında kafes sistemlerin topoloji optimizasyonunda üye burkulması ile ilgili çeşitli konuları ele almıştır. Zemin yapısının paralel ardışık üyeleri zincirler olarak tanımlanmış ve örtüşen elemanlar, bir zincirin her bir düğüm çifti arasındaki zemin yapısına eklenmiştir. Doğru burkulma uzunluğu, bir zincire ait olan her bir düğüm çifti arasına örtüşen üyeler eklenerek ve eklenen tüm üyeler için burkulma kısıtları yazarak yakalanır. Örtüşen ve kesişen elemanlar içeren topolojilerden kaçınmak için probleme bir dizi kısıtlama da eklenmiştir.

Hooshmand ve Campbell (2016) bu çalışmasında, kablolu kafes sistemlerin boyut, şekil ve topoloji optimizasyonu için yeni bir yaklaşım sunmuştur. Bu çalışmada aşağıdaki kısıtlar kullanılmıştır: (1) gerilme, (2) yer değiştirme, (3) burkulma kuvveti, (4) tasarım alanını sınırlandırmak için dış uzamsal sınırlar, (5) minimum yapı elemanı uzunluğu ve (6) maksimum yapı elemanı uzunluğu (5 ve 6 numaralı kısıtlar burkulma kısıtının yanı sıra üretim kısıtı veya estetik hususlar nedeniyle getirilmiştir).

Fu vd. (2017) bu makalede, yapısal düğümlerin konumlarında küçük belirsizliklerle kafes sistemlerin topoloji optimizasyonu problemini çözmeye yönelik bir yaklaşım sunmuşlardır. Duyarlılık hesaplarından kaçınmak için bu çalışmada gradyansız bir yöntem olan PTO (Proportional Topology Optimization) yöntemi kullanılmıştır. Kafes sistemlerin düğüm konumlarının rastgele olduğu varsayılmış ve buradaki belirsizlikle başa çıkabilmek için olasılık yöntemi kullanılmıştır. İmalat kısıtı olarak örtüşen elemanlardan kaçınılmıştır.

Kanno (2018 a) yaptığı bu çalışmada kendi ağırlık yükü ve düğüm sayısı üzerindeki üst sınır kısıtları altında kafes yapıların topoloji optimizasyonu için yeni bir MISOCP (Mixed Integer Second Order Cone Programming) adı verilen bir yaklaşım sunmuştur. İkinci kısıt, üretim maliyeti açısından pratik olarak önemlidir. Sayısal deneylerle, düğüm sayısı üzerindeki kısıtın genellikle üretilebilirlik açısından tercih edilen, basit bir kafes tasarımı sağladığı gösterilmiştir. Ayrıca örtüşen elemanların varlığını yasaklayan kısıtlar da getirilmiştir.

Kanno (2018 b) harici yükteki belirsizliği göz önünde bulundurarak topoloji optimizasyonu için bir konveks programlama yaklaşımı geliştirmiştir. Sorunun kombinatoryal yönleri tamamlayıcılık kısıtları çerçevesinde ele alınmıştır. Yapım kısıtı olarak üst üste binen elemanlar sınırlandırılmıştır.

Cui vd. (2018) daha önce gerilme ve yer değiştirme kısıtları altında kafes sistemlerin topoloji optimizasyonu için geliştirilmiş olan iki seviyeli yaklaşım yöntemini bu çalışmada, eleman burkulma kısıtlarını dikkate almak ve topoloji optimizasyonu için elemanların kesişmesi ve üst üste binmesini kısıtlamak için geliştirmiş ve genişletmiştir.

Xiao vd. (2018) kafes sistemlerin tasarımı için eklemeli üretim ve topoloji optimizasyonunu kullanmışlardır. Farklı yükleme modlarına karşılık gelen üç kafes yapısı FCC (Face Center Cube), VC (Vertex Cube) ve ECC (Edge Center Cube) topoloji optimizasyonu ile tasarlanmış ve ardından SLM (Selective Laser Melting) ile üretilmiştir.

Shahabsafa vd. (2020) kinematik stabiliteyi dikkate alarak ayrık kafes topoloji tasarımı ve boyutlandırma optimizasyonu problemi için yeni bir MILO (Mixed Integer Linear Optimization) adı verilen bir model geliştirmişlerdir. Yapım kısıtı olarak üst üste binen ve kesişen elemanlardan kaçınılmıştır.

Fairclough ve Gilbert (2020) bu çalışmalarında kafes sistem düzen optimizasyonu problemlerinin ölçeğini temel inşa edilebilirlik kısıtlarıyla genişletmeyi amaçlamışlardır. Önceki formülasyon, bir kafes yapısının pratikliğini ve inşa edilebilirliğini artırmak için tasarlanmış kısıtlar da dahil olmak üzere çok çeşitli kısıtlar uygulayabilen esnek bir yöntem sağlamak için MILP (Mixed Integer Linear Programming) kullanılarak genişletilmiştir.



Çizelge 2.2’de de imalat kısıtlarından her makalenin ne anladığı listelenmiştir. Çelik yapılar üzerine çalışmalar ve imalatını yapan firmaların sitelerinde imalat kısıtı olabilecek farklı herhangi bir kısıta rastlanmamıştır.

Çizelge 2.2’de görüldüğü üzere literatürde imalat kısıtlarının tümünün dikkate alındığı bir çalışma mevcut değildir. Ama imal edilebilir bir kafes sistem üretilebilmesi için örtüşen elemanlar, kesişen elemanlar, bir düğüme bağlı eleman sayısı, yapı elemanı uzunluğu, en fazla düğüm sayısı ve elemanlar arasındaki en küçük açı kısıtlarının hepsini dikkate almak gerekir.



### 3. YÖNTEM

İnsanlar çok eski zamanlardan bu yana bir problemin tek bir olurlu çözümünü olmadığını fark etmiş ve bu çözümlerden en iyi (ideal) olanını bulma arayışına girmişlerdir. Bu en iyiyi arama prosedürünü ise “optimizasyon” olarak adlandırmışlardır.

Bilim, mühendislik, teknoloji, matematik, felsefe ve daha birçok alandaki gelişmeler ve ilerlemelerde, her zaman daha iyi performansla daha iyi ürünler veya sonuçlar elde etmek için bir çaba sarf edilir (Sinha, G., R., 2019). Günümüzde çoğu optimizasyon çalışması, tasarım sürecinde yer alan kısıt ve belirsizliklere göre uygun matematiksel programlama teknikleri kullanılarak en iyi (optimum) çözümü bulmakla ilgilenebilir. Elle ya da bilgisayar destekli tasarım ile yürütülen tasarım süreçlerinde (bilgisayar destekli tasarımdan anlaşılacak şey bilgisayarın mühendisin önerdiği tasarımın ne derece uygun olduğunu döndürdüğü bir işleyiş biçimidir) bir problemin optimum çözümüne ulaşmak neredeyse imkansızdır.

Bulduğumuz dönem, dinamik olarak değişmeye devam eden teknolojik gelişmelerin ve endüstriyel devrimlerin çağıdır. Bu gelişmelere bağlı olarak optimizasyon süreçleri ve tasarım prosedürleri de yapay zeka yöntemleri kullanılarak insanların tasarım süreçlerine çok az dahil olduğu ya da dahil olmasına gerek duyulmadığı, tüm aşamaların bilgisayar ile otomatik bir şekilde gerçekleştirildiği “Bilgisayar ile Otomatik Tasarım” (BOT) olarak adlandırılan bir tasarım sürecine evrilmektedir.

Bilgisayar ile otomatik tasarımda tüm aşamalar insan müdahalesi olmadan bilgisayar tarafından gerçekleştirildiğinden binlerce hatta milyonlarca farklı tasarım üretilebilir ve bunların analizi gerçekleştirilebilir. Bu sayede optimale yakın veya optimum çözümlere ulaşmak mümkün olmaktadır. Bu makalede ise “Bilgisayar ile Otomatik Tasarımdan” ifadesinden anlaşılması gereken kafes sistemlerin yapısal ve imalat kısıtlarını sağlayan en hafif çözümünün (optimallik kriteri minimum ağırlık olan) bulunması için tüm tasarım ve optimizasyon aşamalarının herhangi bir kullanıcı müdahalesi olmadan Python programlama dili kullanılarak yazılan bir program ile gerçekleştirilmesidir.

### 3.1 Geliştirme Platformu

Geliştirme platformu olarak nesne yönelimli, yorumlamalı, birimsel (modüler) ve etkileşimli yüksek seviyeli bir dil olan Python tercih edilmiştir. Bu tercihin sebebi Python için çok sayıda kütüphanenin bulunması ve hemen her ihtiyaç için hazırlanmış fonksiyonlara ulaşılabilmesidir. Aynı zamanda Python ile yazılmış kodlar ticari yazılımlar için özgürce kullanılabilir ve sahip olduğu basit söz dizimi ile diğer programlama dillerine göre daha kolay bir şekilde öğrenilir, anlaşılabilir ve üzerinde geliştirmeler yapılabilir. Bu çalışma kapsamında hazırlanan Python kodu, random, math, numpy, matplotlib ve openseespy kütüphanelerinden faydalanmaktadır. Kodda yararlanılan kütüphanelerin özellikleri Çizelge 3.1’de verilmiştir.

Çizelge 3.1. Python kodunda yararlanılan kütüphaneler

Random	Random, yerleşik bir Python kütüphanesidir. Bu kütüphane, rastgele sayılar üretmek, bir liste veya dizi için rastgele bir değer yazdırmak gibi tüm rastgele eylemleri gerçekleştirmek için kullanılabilir.
Math	Python, matematiksel hesaplamalarla başa çıkmak için bir matematik kütüphanesi içerir. Math, toplama(+), çıkarma(-), çarpma(*), bölme (/) gibi temel işlemler ve trigonometrik, logaritmik, üstel fonksiyonlar gibi gelişmiş işlemlerin gerçekleştirilebileceği bir kütüphanedir.
Numpy	NumPy (Numerical Python), hesaplamaları hızlı bir şekilde yapmamızı sağlayan kapsamlı bir sayısal yöntemler kütüphanesidir. Numpy’ın temelini numpy dizileri oluşturur. Numpy dizileri python listelerine benzer fakat hız ve işlevsellik açısından python listelerinden daha kullanışlıdır. (Bayram Durna, 2019 )
Matplotlib	Matplotlib, Python ve NumPy ile çalışanlar için faydalı olan çok güçlü bir çizim kütüphanesidir. Çoğu problemde verileri istatistiksel olarak değerlendirebilmek için görselleştirmek oldukça önemlidir ve Matplotlib bu amaç için oldukça kullanışlı bir kütüphanedir.

Resmi sitesinde verilen tanımı ile OpenSees (the Open System for Earthquake Engineering Simulation), yapısal ve jeoteknik sistemlerin sismik tepkisini simüle etmek için bir yazılım çerçevesidir. OpenSees, Pasifik Deprem Mühendisliği Araştırma Merkezi'nde (Pacific Earthquake Engineering Research (PEER) Center) performansa dayalı deprem mühendisliği araştırmaları için hesaplama platformu olarak geliştirilmiştir (Opensees, 2006).

- Yetenekleri: OpenSees, çok çeşitli malzeme modelleri, öğeler ve çözüm algoritmaları kullanan sistemlerin doğrusal olmayan tepkisini modellemek ve analiz etmek için gelişmiş yeteneklere sahiptir. Her ne kadar deprem üzerinde özelleşmiş bir yazılım olarak tanıtılsa da genel bir sonlu elemanlar çözücüsü olarak da kullanılabilir (Opensees, 2006).
- Modelleme: OpenSees, yapısal ve jeoteknik modeller için kiriş-kolon elemanları ve sürekli elemanlar sağlar. Kiriş ve kolonlar için çok çeşitli tek eksenli malzeme ve kesit modelleri mevcuttur (Opensees, 2006).
- Analiz: Doğrusal olmayan analiz, çok çeşitli algoritmalar ve çözüm yöntemleri gerektirir. OpenSees, doğrusal olmayan statik ve dinamik yöntemler, denklem çözücüler ve kısıtlamaları işleme yöntemleri sağlar (Opensees, 2006).

OpenSeesPy ise OpenSees sonlu elemanlar analizi yazılımının Python kütüphanesi haline getirilmiş biçimidir. Kullanıcılar, gelişmiş sonlu eleman analizi modellerinden ve algoritmalarından yararlanırken, Python dilinin özelliklerini kullanan uygulamalar geliştirebilirler.

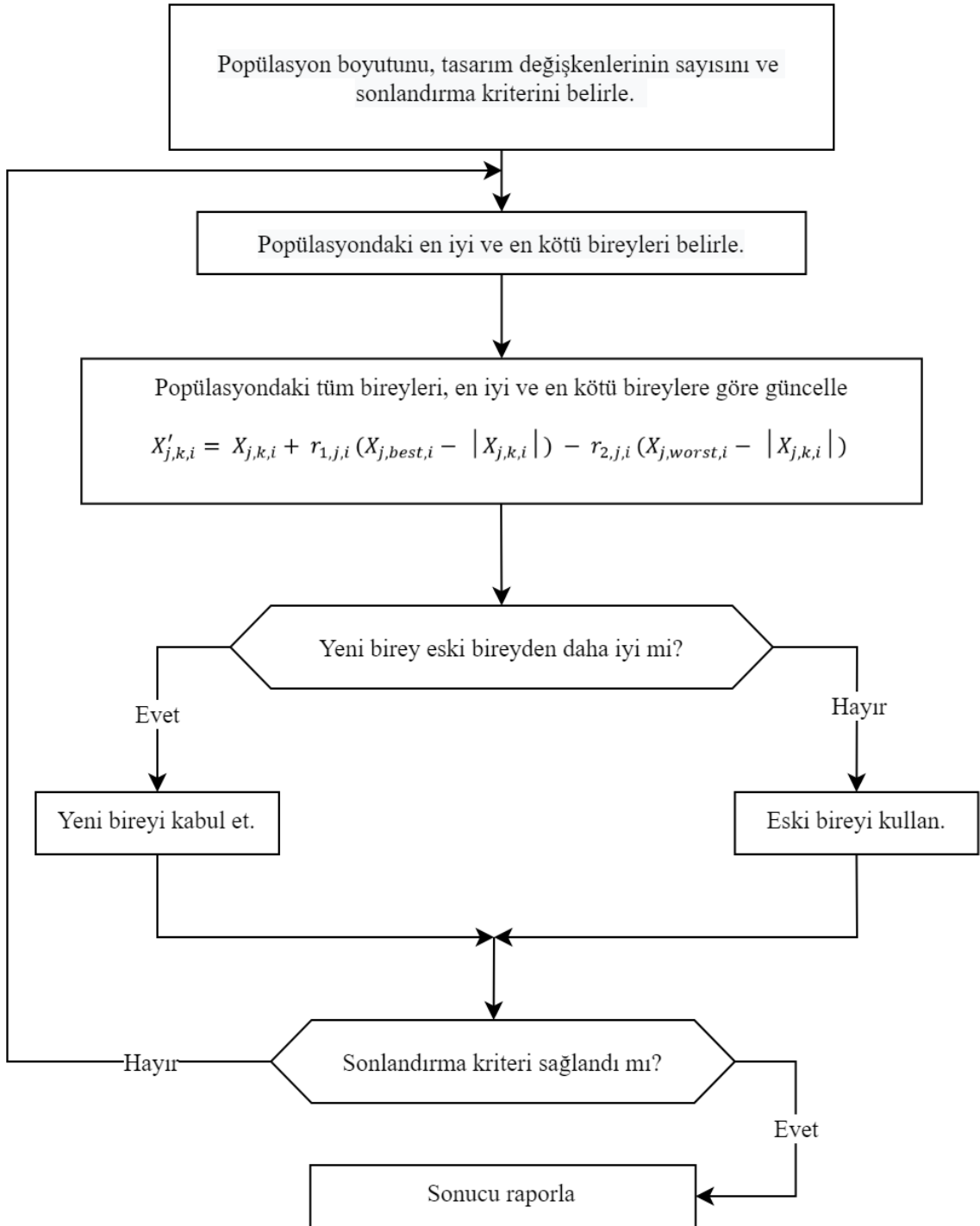
### 3.2 Kullanılan Optimizasyon Algoritması

Metasezgisel algoritmalar, problemleri çözmek için kullanılacak güçlü ve sağlam optimizasyon teknikleridir. Bu yöntemler problemten bağımsız, doğadan ilham alan, genellikle amaç fonksiyonu ve kısıtlar hakkında herhangi bir gradyan bilgisi gerektirmeyen popülasyon tabanlı optimizasyon teknikleridir. Ayrıca geleneksel matematiksel yöntemlerden farklı olarak nihai çözümün kalitesi genellikle başlangıç noktasına bağlı değildir. Birçok temel özelliği paylaşmasına rağmen, farklı metasezgisel yöntemler farklı kural setleri kullanır ve bu nedenle çözümler farklı nihai yapılarla sonuçlanır. Bu algoritmalar genellikle, daha doğru ve sağlam sonuçlar elde ederken daha az hesaplama çabası ile sorunu çözmeyi amaçlar (Kaveh ve Zolghadr, 2018).

Bu çalışmada optimizasyon için metasezgisel bir algoritma olan Jaya algoritması kullanılmıştır. Bu algoritma, bir aday çözümün iyileştirilebilmesi için popülasyondaki en iyi çözüme doğru ilerlemesi ve en kötü çözümden uzaklaşması gerektiği fikrine dayanmaktadır. Jaya algoritması algoritmaya özgü herhangi bir kontrol parametresi içermez yalnızca ortak (genel) kontrol parametrelerini ayarlamak yeterlidir. Jaya algoritmasının tek satırlık hareket operatörü Denklem 3.1’de verilmiştir.

$$X'_{j,k,i} = X_{j,k,i} + r_{1,j,i} (X_{j,best,i} - |X_{j,k,i}|) - r_{2,j,i} (X_{j,worst,i} - |X_{j,k,i}|) \quad (3.1)$$

Denklem 3.1’de  $X_{j,k,i}$ ; i’nci iterasyondaki k’inci tasarım için j’inci tasarım değişkenini temsil eder.  $X'_{j,k,i}$  yeni adayın tasarım değişkenini,  $r_{1,j,i}$  ve  $r_{2,j,i}$  i’nci iterasyondaki j’inci tasarım değişkeni için [0,1] aralığında rastgele üretilen reel sayıları,  $X_{j,best,i}$  ve  $X_{j,worst,i}$  i’nci iterasyondaki en iyi ve en kötü tasarımların j’inci tasarım değişkenini göstermektedir. Denklemdeki  $r_{1,j,i} (X_{j,best,i} - |X_{j,k,i}|)$  ifadesi mevcut tasarımın en iyi tasarıma yaklaşmasını,  $-r_{2,j,i} (X_{j,worst,i} - |X_{j,k,i}|)$  ifadesi ise mevcut tasarımın en kötü tasarımdan uzaklaşmasını sağlamaktadır.  $r_{1,j,i}$  ve  $r_{2,j,i}$  reel sayıları tasarım uzayının daha etkili araştırılmasını sağlarken,  $|X_{j,k,i}|$  mutlak değeri arama kabiliyetinin güçlendirilmesini sağlamaktadır (Venkata Rao, 2016).



Şekil 3.1. Jaya algoritmasının akış şeması

### 3.2.1 Sayısal örnek

Jaya algoritması aslında çok temel ve basit bir mantığa dayanır: öncelikle en iyi ve en kötü adaylar belirlenir, belirlenen adaylar kullanılarak her aday en iyi çözüme yaklaştırılırken en kötü çözümden de uzaklaştırılır. Jaya algoritmasının mantığının daha iyi anlaşılabilmesi için algoritma adımları bir sayısal örnek üzerinde gösterilmiştir.

$$f(x) = x_1^2 - x_1 * x_2 + x_2^2 - 4 * x_1 + 2 * x_2 + 3 \quad (3.2)$$

$$-3 \leq x_1, x_2 \leq 3 \quad (3.3)$$

Denklem 3.2’de verilen fonksiyon için  $x_1$  ve  $x_2$  değişkenlerinin alabileceği değerler Denklem 3.3’te gösterildiği gibi sınırlandırılmıştır. Denklem 3.2’de verilen fonksiyonun verilen koşullar altındaki minimum değerinin -1 olduğu bilinmektedir ve fonksiyonun minimum değerini almasını sağlayan  $x_1$  ve  $x_2$  değişkenlerinin değerleri ise sırasıyla (2, 0) olarak bulunmuştur.

Denklem 3.2’de verilen fonksiyonun Jaya algoritması kullanılarak belirlenen sınır koşulları altında alabileceği minimum değerinin nasıl bulunduğu aşağıda gösterilmiştir. Örnek için popülasyon büyüklüğü 5, iterasyon sayısı 20 olarak alınmıştır.

#### Adım 1

Öncelikle  $x_1$  ve  $x_2$  değişkenlerine -3 ila 3 değerleri arasında rastgele değerler vermek için Denklem 3.4’ te verilen formülasyon kullanılır.

$$U \leq x \leq L$$

$$x = L + r * (U - L)$$
(3.4)

Denklem 3.4'te U ve L kullanıcı tarafından belirlenen üst ve alt sınır değerler, r ise 0 ila 1 arasında rastgele bir sayıdır.

$$\begin{bmatrix} x_1 & x_2 \\ -2.427 & 2.328 \\ -1.992 & 2.544 \\ -0.405 & -2.468 \\ 0.990 & -0.793 \\ 1.198 & -2.419 \end{bmatrix}; \begin{bmatrix} f(x) \\ 34.324 \\ 31.564 \\ 4.940 \\ -0.152 \\ 3.555 \end{bmatrix}$$
(3.5)

Denklem 3.4'te verilen formülasyon kullanılarak  $x_1$  ve  $x_2$  değişkenlerinin her biri için -3 ila 3 arasında virgülden sonra hassasiyetleri 4 basamak olacak şekilde rastgele 5 sayı üretilmiştir. Üretilen bu rastgele sayılar denklemde yerine konularak  $f(x)$  fonksiyonunun aldığı değerler bulunmuş ve hem  $x_1$  ve  $x_2$  hem de  $f(x)$  değerleri Denklem 3.5'te verilmiştir. Problemden denklemin alabileceği en küçük değer arandığı için bulunan  $f(x)$  değerlerinden; en büyük olan değere karşılık gelen  $x_1$  ve  $x_2$  değerleri en kötü çözüm, en küçük olan değere karşılık gelen  $x_1$  ve  $x_2$  değerleri ise en iyi çözüm olarak kabul edilmiş ve Denklem 3.6'da verilmiştir.

$$[x_{best,1}, x_{best,2}] = [0.990, -0.793]$$

$$[x_{worst,1}, x_{worst,2}] = [-2.427, 2.328]$$
(3.6)

## Adım 2

Sırasıyla  $x_1$  ve  $x_2$  değerleri Jaya algoritması kullanılarak güncellenir. Bulunan değerler denklemde yerine koyularak yeni  $f(x)$  değeri bulunur ve eski çözüme ait olan  $f(x)$  değeri ile karşılaştırılır. Güncellenmiş değerlerle bulunan  $f(x)$  değeri, eski  $f(x)$  değerinden daha küçükse ve  $x_1$  ve  $x_2$  değerleri sınır koşullarını sağlıyorsa bulunan yeni  $f(x)$  değeri kabul edilir ve eski çözüme ait  $x_1$  ve  $x_2$  değerleri yeni bulunan  $x_1$  ve  $x_2$  değerleri ile güncellenir. Ancak güncellenmiş değerlerle bulunan  $f(x)$  değeri, eski  $f(x)$  değerinden büyükse güncelleme yapılmaz ve eski değerler sabit kalır.

Denklem 3.7’de Jaya algoritması kullanılarak güncelleme işlemi yapılmıştır;

$$r_1 = 0.541, \quad r_2 = 0.625$$

$$\begin{aligned} x_1 &= -2.427 + 0.541(0.990 - |-2.427|) - 0.625(-2.427 - |-2.427|) \\ &= -0.171 \end{aligned}$$

$$\begin{aligned} x_2 &= 2.328 + 0.541(-0.793 - |2.328|) - 0.625(2.328 - |2.328|) \\ &= 0.640 \end{aligned} \quad (3.7)$$

$$x_{\text{new}} = [-0.171, 0.640] \in [-3, 3]$$

$$\begin{aligned} f(x_{\text{new}}) &= -0.171^2 + 0.171 * 0.640 + 0.640^2 - 4 * -0.171 + 2 \\ &\quad * 0.640 + 3 = 5.509 \quad (\text{Yeni çözüm}) \end{aligned}$$

$$f(x) = 34.324 \quad (\text{Eski çözüm}) \quad (3.8)$$

Gerçekleştirilen işlemler sonucunda  $f(x_{\text{new}})$  değeri elde edilir.  $f(x_{\text{new}}) < f(x)$  ise  $x_1$ ,  $x_2$  ve  $f(x)$  değerleri güncellenir.

$$\begin{bmatrix} x_1 & x_2 \\ -0.171 & 0.640 \\ -1.992 & 2.544 \\ -0.405 & -2.468 \\ 0.990 & -0.793 \\ 1.198 & -2.419 \end{bmatrix}; \begin{bmatrix} f(x) \\ 5.512 \\ 31.564 \\ 4.940 \\ -0.152 \\ 3.555 \end{bmatrix} \quad (3.9)$$

Yeni ve eski çözüm karşılaştırıldığında yeni çözüm değeri daha küçük olduğu için kabul edilmiş ve  $x_1$ ,  $x_2$  ve  $f(x)$  değerleri Denklem 3.9'da gösterildiği gibi güncellenmiştir.

### Adım 3

Belli bir güncelleme işleminden sonra  $x_1$  ve  $x_2$  değişkenleri  $f(x)$  fonksiyonunu minimum yapan değerlerine ulaşır ve  $f(x)$  değeri artık değişmez olur.  $x_1$  ve  $x_2$ 'nin -3 ila 3 arasında alacağı değerlerle  $f(x)$  fonksiyonunun minimum olduğu nokta bulunmuş olur.  $f(x)$  fonksiyonunun minimum olduğu nokta Denklem 3.11'de gösterilmiştir.

### 10.İterasyon

$$\begin{bmatrix} x_1 & x_2 \\ 1.924 & -0.055 \\ 1.807 & -0.480 \\ 1.764 & -0.308 \\ 1.969 & -0.015 \\ 2.163 & -0.013 \end{bmatrix}; \begin{bmatrix} f(x) \\ -0.995 \\ -0.825 \\ -0.922 \\ -0.999 \\ -0.971 \end{bmatrix} \quad (3.10)$$

## 20. Iterasyon

$$\begin{array}{cc|c} x_1 & x_2 & f(x) \\ \hline 2.01 & -0.005 & -0.99983 \\ 2.012 & 0.010 & -0.99988 \\ 1.995 & -0.002 & -0.99998 \\ \hline 2.000 & 0.000 & -1.0 \\ \hline 2.016 & -0.003 & -0.99968 \end{array} ; \quad (3.11)$$



## 4. SAYISAL ÖRNEKLER

### 4.1 Baz Yapı Yöntemi

Kafes sistemlerin topoloji optimizasyonunda optimal yapılar elde etme imkanı sağlayan yöntemler arasındaki en iyi yöntem, uygulamadaki basitliği ve kullanılabilirliği nedeniyle baz yapı (temel yapı veya zemin yapısı da denir) yöntemidir. Bu yönteme göre tüm aday düğümleri ve düğümler arasındaki bağlantıları (çubuk, eleman) içeren başlangıç yapısına baz yapı denir (Topping, 1983). Baz yapının bazı elemanları çıkartılarak, düğüm noktalarının yerleri değiştirilerek ve eleman kesitleri ile oynanarak optimum yapıya ulaşılabileceği düşünülür. Bu nedenle, baz yapıda ne kadar çok eleman varsa, en uygun çözümün potansiyel bağlantılar içinde olma olasılığı o kadar artar. Optimizasyon sırasında, yeterince küçük alanlara sahip elemanlar zemin yapısından silinir, böylece sistemin topolojisi değişir ve kalan elemanlar en uygun yapıyı oluşturur (Gao vd., 2017). Genel olarak bu eleme, problem kısıtları göz önünde bulundurularak gerçekleştirilir (Deb ve Gulati, 2001). Aynı zamanda, elemanları silmek kararsız mekanizmalar üretebileceğinden, kabul edilebilir yapısal topolojilerin elde edilmesi için ekstra geometrik kısıtlar karşılanmalıdır (Mortazavi ve Toğan, 2016).

Zemin yapısı, nihai topoloji yapısı üzerinde büyük bir etkiye sahip olduğundan zemin yapısını oluşturmak için kullanılan yöntem kafes sistemlerin topoloji optimizasyonunda oldukça büyük bir öneme sahiptir. Bir yöntem yalnızca birkaç düğüm ve çubukla iyi tanımlanmış bir zemin yapısı oluşturabilirse, iyi bir topoloji çözümü elde etmeye ve hesaplama maliyetini azaltmaya yardımcı olabilir (Gao vd., 2017).

## 4.2 Kısıtlar

Bu çalışmada yapısal kısıtlar olarak çubuk gerilmeleri ve düğüm yer değiştirmeleri, imalat kısıtları olarak ise en fazla kesit çeşidi, örtüşen elemanlar, kesişen elemanlar, bir düğüme bağlı eleman sayısı, yapı elemanı uzunluğu, en fazla düğüm sayısı ve elemanlar arasındaki en küçük açı dikkate alınmış ve ayrık tasarım değişkenleri kullanılarak kafes sistemlerin optimizasyonu gerçekleştirilmiştir. Ayrık tasarım değişkenleri bir imalat kısıtı olarak kabul edilmemiş ve Çizelge 4.1’de verilen eşit kollu korniyer alanları kullanılmıştır.

Çizelge 4.1. Optimizasyonda kullanılan eşit kollu korniyer alanları

Numara	Eşit Kollu Korniyer Boyutları	Alan (mm <sup>2</sup> )
1	L 20 x 20 x 3	112
2	L 25 x 25 x 3	142
3	L 25 x 25 x 4	185
4	L 30 x 30 x 3	174
5	L 30 x 30 x 4	227
6	L 35 x 35 x 4	267
7	L 40 x 40 x 4	308
8	L 40 x 40 x 5	379
9	L 45 x 45 x 4	349
10	L 45 x 45 x 4,5	390
11	L 45 x 45 x 5	430
12	L 50 x 50 x 4	389
13	L 50 x 50 x 5	480
14	L 50 x 50 x 6	569
15	L 50 x 50 x 7	656
16	L 55 x 55 x 5	532
17	L 55 x 55 x 6	631
18	L 60 x 60 x 4	471
19	L 60 x 60 x 5	582
20	L 60 x 60 x 6	691
21	L 60 x 60 x 8	903
22	L 65 x 65 x 6	753
23	L 65 x 65 x 7	870
24	L 65 x 65 x 8	985
25	L 70 x 70 x 6	813
26	L 70 x 70 x 7	940
27	L 70 x 70 x 8	1070
28	L 70 x 70 x 9	1190
29	L 75 x 75 x 4	593

Çizelge 4.1. Optimizasyonda Kullanılan Eşit Kollu Korniyer Alanları (devam)

30	L 75 x 75 x 5	734
31	L 75 x 75 x 6	873
32	L 75 x 75 x 7	1010
33	L 75 x 75 x 8	1140
34	L 75 x 75 x 10	1410
35	L 80 x 80 x 5	786
36	L 80 x 80 x 6	935
37	L 80 x 80 x 7	1080
38	L 80 x 80 x 8	1230
39	L 80 x 80 x 10	1510
40	L 90 x 90 x 6	1050
41	L 90 x 90 x 7	1220
42	L 90 x 90 x 8	1390
43	L 90 x 90 x 9	1550
44	L 90 x 90 x 10	1710
45	L 90 x 90 x 11	1870
46	L 100 x 100 x 6	1180
47	L 100 x 100 x 7	1370
48	L 100 x 100 x 8	1550
49	L 100 x 100 x 10	1920
50	L 100 x 100 x 12	2270
51	L 110 x 110 x 8	1710
52	L 110 x 110 x 10	2120
53	L 110 x 110 x 12	2510
54	L 120 x 120 x 8	1870
55	L 120 x 120 x 10	2320
56	L 120 x 120 x 11	2540
57	L 120 x 120 x 12	2750
58	L 120 x 120 x 13	2970
59	L 120 x 120 x 15	3390
60	L 120 x 120 x 16	3600
61	L 130 x 130 x 10	2520
62	L 130 x 130 x 12	3000
63	L 130 x 130 x 13	3230
64	L 130 x 130 x 14	3470
65	L 130 x 130 x 16	3930
66	L 150 x 150 x 10	2930
67	L 150 x 150 x 12	3480
68	L 150 x 150 x 13	3760
69	L 150 x 150 x 14	4030
70	L 150 x 150 x 15	4300
71	L 150 x 150 x 16	4570
72	L 150 x 150 x 18	5100
73	L 150 x 150 x 20	5630
74	L 160 x 160 x 14	4320
75	L 160 x 160 x 15	4610

Çizelge 4.1. Optimizasyonda Kullanılan Eşit Kollu Korniyer Alanları (devam)

76	L 160 x 160 x 16	4900
77	L 160 x 160 x 17	5180
78	L 180 x 180 x 13	4550
79	L 180 x 180 x 14	4880
80	L 180 x 180 x 15	5210
81	L 180 x 180 x 16	5540
82	L 180 x 180 x 17	5870
83	L 180 x 180 x 18	6190
84	L 180 x 180 x 19	6510
85	L 180 x 180 x 20	6830
86	L 200 x 200 x 15	5810
87	L 200 x 200 x 16	6180
88	L 200 x 200 x 17	6550
89	L 200 x 200 x 18	6910
90	L 200 x 200 x 19	7270
91	L 200 x 200 x 20	7630
92	L 200 x 200 x 21	7990
93	L 200 x 200 x 22	8350
94	L 200 x 200 x 23	8710
95	L 200 x 200 x 24	9060
96	L 200 x 200 x 25	9410
97	L 200 x 200 x 26	9760
98	L 250 x 250 x 20	9630
99	L 250 x 250 x 21	10100
100	L 250 x 250 x 22	10600
101	L 250 x 250 x 23	11000
102	L 250 x 250 x 24	11500
103	L 250 x 250 x 25	11900
104	L 250 x 250 x 26	12400
105	L 250 x 250 x 27	12800
106	L 250 x 250 x 28	13300
107	L 250 x 250 x 35	16300

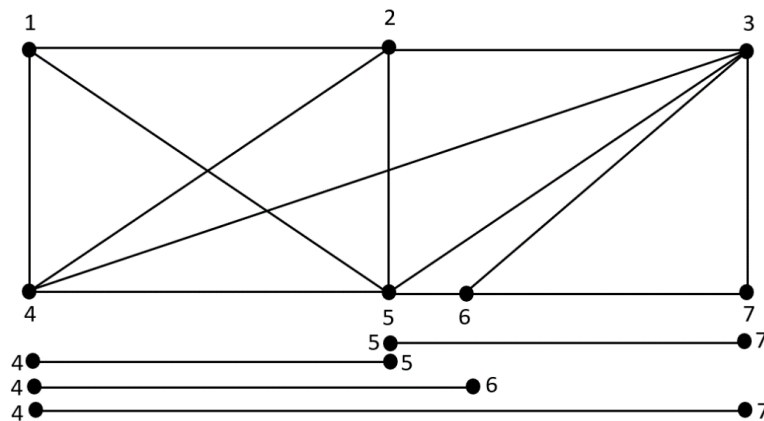
Gerçek hayatta kafes sistemler fabrikalarda standart olarak üretilen profiller kullanılarak inşa edilirler. Bu çalışmada ayırık tasarım değişkenleri uygulanmış ama bir imalat kısıtı olarak ele alınmamıştır.

### 4.2.1 İmalat kısıtları

Bir kafes sistem yapısında mimari biçimin korunması önemli bir imalat kısıtıdır. Örneğin kemer biçimli bir yapıyı optimize ederken yapının şekil parametrelerinin (düğüm noktası koordinatları) değişimi mimari formu bozmayacak şekilde gerçekleştirilmelidir.

İmalat sırasında bir taşıyıcı sistemde çok sayıda farklı kesitin kullanılması montaj hatası ihtimalini oldukça yükseltir. Bunun yanı sıra yapı çeliği siparişlerinde belli bir miktarın altında malzemeyi tedarik etmek son derece maliyetli olmaktadır. Bu yüzden yapılan çalışmada yazar en fazla kesit çeşidini bir imalat kısıtı olarak kabul etmiş ve uygulamışlardır.

Tasarım alanını tam olarak keşfetmek ve optimum bir tasarım elde etme olasılığını artırmak için, baz yapı her zaman izin verilen herhangi iki düğümün birleştirilmesi ile inşa edilir ve bu nedenle, bu tür yapılar kaçınılmaz olarak kesişen ve örtüşen elemanlar (Şekil 4.1) içerir. Doğrudan mühendislikte kullanılacak elverişli kafes sistem tasarımları üretebilmek için optimizasyon sürecinde örtüşen ve kesişen elemanların varlığını yasaklayan kısıtlar uygulanması gerekir (Cui vd., 2018). Ayrıca literatürde doğru burkulma uzunluğunu elde edebilmek için baz yapıya örtüşen elemanların dahil edildiği görülmektedir. Örtüşen elemanların varlığı, optimizasyon problemine ilave yük getirir çünkü genelde birbirleriyle örtüşen iki elemandan birini seçme zorunluluğu ile karşılaşılır (Kanno, 2018 a).

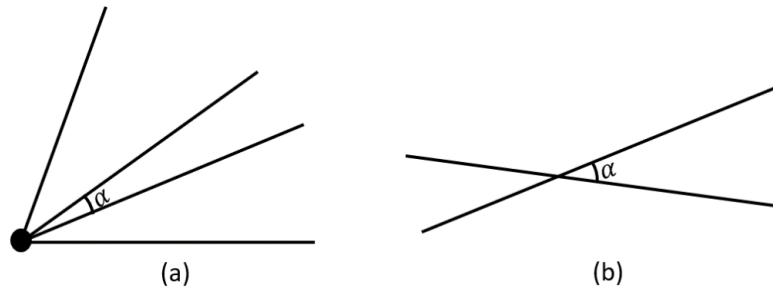


Şekil 4.1. Örtüşen ve kesişen eleman kısıtı

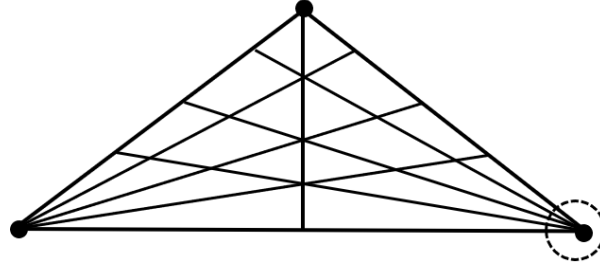
Yukarıdaki şekilde (Şekil 4.1) örtüşen ve kesişen elemanlar gösterilmiştir. Örtüşen eleman kısıtı uygulandığında 4-7, 4-6 ve 5-7 elemanları yerine 4-5-6-7 elemanı seçilerek 4-7, 4-6 ve 5-7 elemanları sistemden kaldırılır. 1-5, 2-4 ve 1-5, 3-4 elemanlarının kesişen eleman kısıtını sağlayabilmesi için kesişen elemanlardan birinin sistemden kaldırılması gerekir. Ancak Şekil 4.1'de gösterilen örtüşen ve kesişen elemanlardan herhangi biri optimizasyon işlemi sırasında kaldırılamazsa imal edilebilir bir çözüm elde edilemez.

Düğüm sayısı üzerindeki kısıtlamalar genellikle imal edilebilirlik açısından tercih edilen basit bir kafes sistem tasarımının üretilmesini sağlar. Aynı zamanda bir kafes sistemin hem imalat maliyeti hem de hesaplama yükü düğüm sayısı arttıkça artar. Bu durumlardan dolayı düğüm sayısı üzerinde bir üst sınır kısıtı uygulamak hesaplama maliyetini düşürmek ve üretilebilir çözümler elde edebilmek için oldukça verimlidir.

Kafes sistemlerin üretilmesini zorlaştıran bir diğer problem elemanlar arasında küçük açılarının bulunmasıdır. Burada dikkat edilmesi gereken bir nokta ise açının her iki eleman için ortak olan bir düğüm noktasında (Şekil 4.2(a)) ya da elemanların uzunluklarından biri veya her ikisi boyunca kısmen kesiştiği bir noktada (Şekil 4.2(b)) oluşabileceğidir (Fairclough ve Gilbert, 2020). Aynı zamanda bir düğüme ne kadar çok eleman bağlanırsa açı kısıtını sağlamak bir o kadar zorlaşacaktır. Elemanlar arasındaki açı kısıtı sağlansa dahi imal edilebilir çözümler elde edebilmek için bir düğüme bağlanan eleman sayısı (Şekil 4.3) da kısıtlanmalıdır çünkü bir sistem imal edilirken bir düğüm noktasına belirli sayıda eleman bağlanabilir. Ayrıca 2 boyutlu bir sistemde serbest (mesnetsiz) bir düğüm noktasına en az 2, 3 boyutlu bir sistemde ise en az 3 eleman bağlanmalıdır eğer bağlanmamışsa o düğüm sistemden kaldırılır.



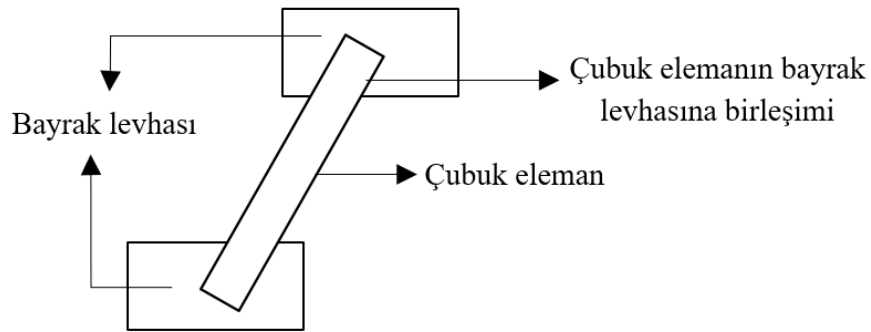
Şekil 4.2. Elemanlar arasındaki açı kısıtı a) her iki eleman için ortak olan bir düğüm noktasında b) elemanların kesiştiği bir noktada küçük açı



Şekil 4.3. Bir düğüme bağlı eleman kısıtı

Kafes sistemler, düğüm noktası olarak adlandırılan noktalara bağlanan çubuk elemanlardan oluşur ve bu çubuk elemanlar yalnızca aksenal kuvvetlere (basınç veya çekme) maruz kalırlar. Basınç kuvveti altında kalan uzun çubuk elemanlar burkulmaya oldukça meyillilerdir. Sistemin burkulma yükleri altında kalmadan imal edilebilmesi için eleman uzunluk kısıtına ihtiyaç duyulur. Ayrıca gerçek dünyada kafes sistemleri oluşturan çubuk elemanlar mükemmel değildir ve bir çubuk elemandaki kusurlar (eksen eğrilikleri, kesit değişiklikleri vb.) boyları uzadıkça imalatı engelleyecek düzeye gelebilir.

Yapı sistemlerinin çiziminde nokta olarak görülen düğüm noktalarını oluşturabilmek için bir çalışma alanına ihtiyaç vardır. Örnek olarak 20 cm'lik kaynaklar ile oluşturulacak bayrak levhali bir birleşim için (Şekil 4.4) yaklaşık olarak 50 cm (çubukların 2 yönden de bayrak levhasına bağlandığı düşünülerek) kadar bir alana ihtiyaç vardır. Bu şekilde iki birleşimin yan yana imal edilebilmesi için birleşim noktaları arasında  $25+25=50$  cm mesafe bulunması gerekir.



Şekil 4.4. İki düğüm arasındaki minimum mesafe kısıtı

Çizelge 4.2. Çalışmada dikkate alınan imalat kısıtları

İmalat Kısıtları	Açıklama
Mimari biçimin korunması	Kafes sistemin kullanım alanına göre tasarlanan biçiminin korunması imal edilebilmesi açısından önemli bir kısıttır.
En fazla kesit çeşidi	Montaj işlemi sırasında imalat hatalarını en aza indirmek ve maliyeti düşürmek amacıyla uygulanan bir imalat kısıtıdır.
Örtüşen elemanlar	Optimum tasarım elde etme olasılığını arttırmak ve doğru burkulma uzunluğunu elde edebilmek için kafes sistemler örtüşen elemanlar içerir ancak örtüşen elemanların imal edilebilmesi mümkün olmadığından örtüşen elemanlar bir imalat kısıtı olarak dikkate alınmıştır.
Kesişen elemanlar	Optimum tasarım elde etme olasılığını arttırmak için kafes sistemler kesişen elemanlar içerir ancak kesişen elemanların imal edilmesi mümkün olmadığından kesişen elemanlar bir imalat kısıtı olarak dikkate alınmıştır.
En fazla düğüm sayısı	Kafes sistemlerde hesaplama maliyetini düşürmek ve üretilebilir çözümlere ulaşılabilmesi amacıyla uygulanan bir imalat kısıtıdır.
Elemanlar arasındaki en küçük açı	Çubuk elemanlar düğüm noktalarına bağlanırken imalat sırasında bağlantı noktalarının ve elemanların çakışmaması için elemanlar arasındaki en küçük açı kısıtına ihtiyaç duyulmuştur.
Bir düğüme bağlı eleman sayısı	Bir kafes sistem imal edilirken bir düğüm noktasına belli sayıda eleman bağlanabilir bu yüzden bir düğüme bağlı eleman sayısı problemlerde imalat kısıtı olarak uygulanmıştır.
Yapı elemanı uzunluğu	Basınç kuvveti altındaki uzun elemanlar burkulmaya oldukça meyillidirler. Aynı zamanda çubuk elemanların kusurları boyları uzadıkça artar bu yüzden yapı elemanı uzunluğu bir imalat kısıtı olarak dikkate alınmıştır.
İki düğüm arasındaki minimum mesafe	İmal edilirken çubuk elemanların düğüm noktalarına bağlanabilmeleri için bir çalışma alanına ihtiyaç vardır bu yüzden iki düğüm arasındaki minimum mesafe bir imalat kısıtı olarak ele alınmıştır.

Dikkate alınan imalat kısıtları yazar tarafından belirlenmiş ve Çizelge 4.2’de çalışmada dikkate alınan tüm imalat kısıtları kısaca özetlenmiştir.

### 4.3 Amaç ve Ceza Fonksiyonları

Optimizasyon işlemi sırasında en iyi adayın tespit edilebilmesi için aşağıdaki yol izlenmiştir:

1. Yapısal kısıtların ihlali sebebiyle oluşan ceza değeri ( $pS$ ) en küçük olan aday(lar) listelenir.
2. İlk maddede listelenen adayların içerisinde imalat kısıtları sebebiyle oluşan ceza değeri ( $pC$ ) en küçük olan aday(lar) ayrılır.
3. İkinci maddedeki işlem sonucu elde edilen listede ağırlığı en küçük olan aday (eğer aynı ağırlığa sahip birden fazla aday varsa herhangi biri) en iyi aday olarak belirlenir.

Benzer şekilde optimizasyon işlemi sırasında en kötü adayın tespiti için aşağıdaki yol izlenmiştir:

1. Yapısal kısıtların ihlali sebebiyle oluşan ceza değeri ( $pS$ ) en büyük olan aday(lar) listelenir.
2. İlk maddede listelenen adayların içerisinde imalat kısıtları sebebiyle oluşan ceza değeri ( $pC$ ) en büyük olan aday(lar) ayrılır.
3. İkinci maddedeki işlem sonucu elde edilen listede ağırlığı en büyük olan aday (eğer aynı ağırlığa sahip birden fazla aday varsa herhangi biri) en kötü aday olarak belirlenir.

Yapısal kısıtların ihlali sebebiyle oluşan ceza değeri (pS) aşağıdaki şekilde hesaplanır.

$$pS = \left( \frac{\sum \left( \frac{str}{\limStr} - 1 \right)}{mbrLen} + \frac{\sum \left( \frac{dsp}{\limDsp} - 1 \right)}{nodLen} \right) \quad (4.1)$$

Her çubuk eleman için gerilme ve her düğüm noktası için yer değiştirme kısıtı yukarıdaki gibi hesaplanır. Denklem 4.1’de str ifadesi bir elemanın gerilme değerini, dsp ifadesi ise bir düğüm noktasının yer değiştirme değerini temsil eder. limStr ve limDsp değerleri sırasıyla gerilme ve yer değiştirme sınır değerlerini ifade etmektedir. mbrLen ifadesi çubuk elemanların toplam sayısını gösterirken, nodLen ifadesi ise tüm düğüm noktalarının toplam sayısını göstermektedir.

İmalat kısıtları sebebiyle oluşan ceza değeri (pC) aşağıda verilen tüm imalat kısıtı cezalarının toplanması ile elde edilmektedir. Kafes sisteme ait her çubuk eleman için eleman uzunluk kısıtı dikkate alınmış ve kısıta uymayan elemanların ceza değeri aşağıdaki şekilde hesaplanmıştır:

$$pL = \frac{\sum \left( \frac{lng}{\limLng} - 1 \right)}{mbrLen} \quad (4.2)$$

Denklem 4.2’de lng, elemanın uzunluğunu temsil eder. limLng eleman uzunluk sınır değerini, mbrLen değeri çubuk elemanların toplam sayısını göstermektedir.

Kafes sisteme ait her çubuk eleman için örtüşen ve kesişen elemanlar dikkate alınmış ve imal edilebilir sistemler üretilebilmesi için örtüşen ve kesişen eleman bulunan tasarımlar cezalandırılmıştır. Örtüşen ve kesişen elemanların ceza değeri aşağıdaki şekilde hesaplanmıştır:

$$pO = \frac{\sum(\text{ovrMem})}{\text{allMem}} \quad (4.3)$$

Denklem 4.3'te ovrMem, örtüşen elemanların örtüştüğü uzunluğu temsil eder. allMem değeri ise tüm elemanların uzunluklarının toplamını göstermektedir.

$$pI = \frac{\text{intMem}}{\text{mbrLen}} \quad (4.4)$$

Denklem 4.4'te intMem, her kesişen eleman için ceza değerine 1 eklenerek bulunur. mbrLen değeri ise çubuk elemanların toplam sayısını göstermektedir.

Kafes sistemlerde birbirine bağlı elemanlar arasındaki açı, imal edilebilirlik açısından kısıtlanmış ve minimum açı kısıtına uymayan her açı için ceza değeri aşağıdaki şekilde hesaplanmıştır:

$$pA = \frac{\sum \left( 1 - \frac{\text{ang}}{\text{limAng}} \right)}{(2 \times \text{mbrLen})} \quad (4.5)$$

Denklem 4.5'te ang, iki eleman arasındaki açıyı temsil eder. limAng açı sınır değerini, mbrLen değeri ise çubuk elemanların toplam sayısını göstermektedir.

İmal edilebilir sistemler elde edebilmek için bir düğüme bağlanan eleman sayısı da kısıtlanmalıdır çünkü bir sistem imal edilirken bir düğüm noktasına belirli sayıda eleman bağlanabilir. Bir düğüme bağlı eleman sayısı kısıtının sağlanamadığı her bir düğüm noktası için ceza değeri aşağıdaki şekilde hesaplanmıştır:

$$pD = \frac{\sum \left( \frac{\text{nCon}}{\text{limMem}} - 1 \right)}{\text{nodLen}} \quad (4.6)$$

Denklem 4.6’da nCon, bir düğüme bağlı tüm çubuk eleman sayısını temsil eder. limMem bir düğüme bağlanabilecek eleman sınır değerini, nodLen değeri ise tüm düğüm noktalarının sayısını göstermektedir.

İmal edilebilir sistemlerde çubuk elemanların düğüm noktalarına bağlanabilmeleri için çalışma alanına ihtiyaç duyulur. İki düğüm arasındaki mesafe bu yüzden bir imalat kısıtı olarak ele alınmış ve kısıtı sağlamayan tüm düğüm noktaları için ceza değeri aşağıdaki şekilde hesaplanmıştır:

$$pN = \frac{\sum \left(1 - \frac{\text{nod}}{\text{limNod}}\right)}{\text{nodLen}} \quad (4.7)$$

Denklem 4.7’de nod, iki düğüm arasındaki uzaklık değerini temsil eder. limNod iki düğüm noktası arasındaki uzunluk sınır değerini, nodLen değeri ise tüm düğüm noktalarının sayısını göstermektedir.

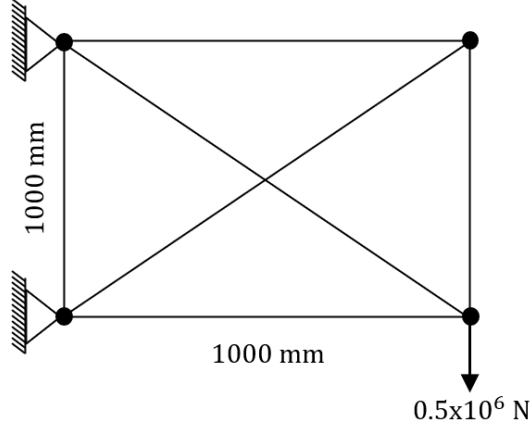
İmalat kısıtları sebebiyle oluşan tüm cezaların toplamı incelenen sistemin imalat cezasını (pC) verir (Denklem 4.8).

$$pC = pL + pO + pI + pA + pD + pN \quad (4.8)$$

#### 4.4 Kullanılan Test Problemleri

Literatürde kafes sistemlerin optimizasyonu için tanıtılan birden çok test problemi bulunmaktadır. Kafes sistemlerin yapısal ve imalat kısıtlarını sağlayan ve ağırlığı en az olan çözümlerine boyut, şekil ve topoloji optimizasyonu sonucunda ulaşabilmek için literatürde tanıtılan test problemleri yetersiz kalacaktır. Bu yüzden yapılan çalışmada konu ile ilgili tanıtılan test problemlerini kullanmak yerine gerçek işlem yükünü tespit edebilmek için yazar tarafından üretilen 3 farklı test problemi üzerinde çalışılmıştır.

#### 4.4.1 4 düğüm noktalı test problemi



Şekil 4.5. 4 düğüm noktalı kafes sistem için baz yapı

Şekil 4.5'te ilk örneğin temel yapısı (muhtemel tüm düğümlerin ve elemanların bulunduğu konfigürasyon) görülmektedir. Bu basit örnek en elverişli durumlardaki hesaplama maliyetini ortaya koyabilmek için çözülmüştür. Tasarım sürecini daha da basitleştirmek için bu örneğin düğüm noktalarının hareketine izin verilmemiştir. Kullanılan malzemenin birim hacim ağırlığı  $\rho = 7850 \text{ kgf/m}^3 \approx 7.7 \times 10^{-5} \text{ N/mm}^3$ , elastisite modülü  $E = 200000 \text{ N/mm}^2$  olarak alınmıştır. Düğüm noktaları önce üst düğümler sonra alt düğümler olacak şekilde 1'den başlayarak numaralandırılır. İzin verilen çubuk gerilmeleri en fazla  $\pm 199.75 \text{ N/mm}^2$  ve düğüm yer değiştirmeleri en fazla  $\pm 20 \text{ mm}$  olacak şekilde sınırlandırılmıştır. Sistemin imal edilebilirliği açısından sistemde örtüşen ve kesişen elemanların bulunmasına izin verilmemiştir. Aynı düğüme bağlanan elemanlar arasındaki en küçük açı  $15^\circ$ , eleman uzunluk sınırı  $1500 \text{ mm}$ , bir düğüme bağlı en fazla eleman sayısı 4, iki düğüm arasındaki izin verilen en küçük mesafe  $200 \text{ mm}$  olacak şekilde sınırlandırılmıştır.

Baz (temel) yapı oluşturulurken hem yatay düzlem üzerindeki iki düğüm noktası arasındaki mesafe hem de düşey düzlem üzerindeki iki düğüm noktası arasındaki mesafe  $1000 \text{ mm}$  olarak verilmiştir. Bu test probleminde tüm düğüm noktalarının yatay koordinat

değerleri sabit tutulmuş, düşey koordinat değerleri ise istenilen mimari formun korunabilmesi için verilen formüller ile hesaplanmıştır.

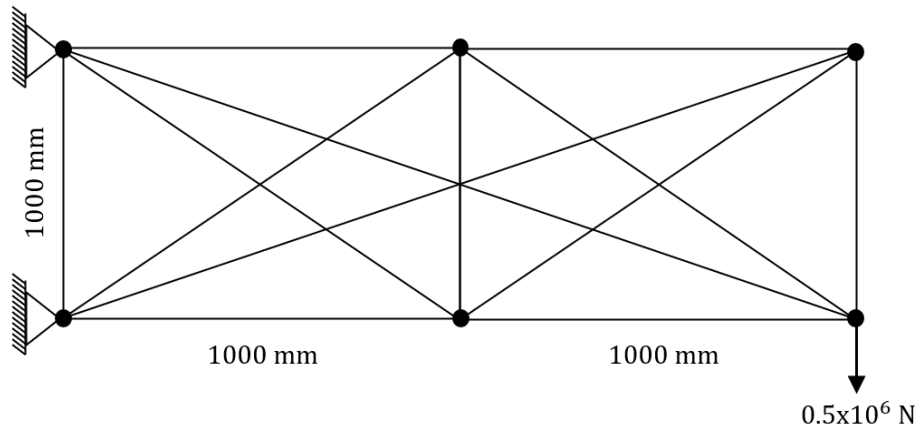
Üst düğüm noktalarının y koordinat değerini hesaplamak için Denklem 4.9 kullanılır:

$$f_{yTop}(x) = 1000.0 \quad (4.9)$$

Alt düğüm noktalarının y koordinat değerini hesaplamak için Denklem 4.10 kullanılır:

$$f_{yBottom}(x) = 0.0 \quad (4.10)$$

#### 4.4.2 6 düğüm noktalı test problemi



Şekil 4.6. 6 düğüm noktalı kafes sistem için baz yapı

Şekil 4.6’da 6 düğüm noktalı kafes sistem örneğinin temel yapısı (muhtemel tüm düğümlerin ve elemanların bulunduğu konfigürasyon) görülmektedir. İkinci tasarım örneğinde de (Şekil 4.6) tüm yapım ve imalat kısıtları birinci tasarım örneği ile aynı olacak şekilde uygulanmıştır.

Baz (temel) yapı oluşturulurken hem yatay düzlem üzerindeki iki düğüm noktası arasındaki mesafe hem de düşey düzlem üzerindeki iki düğüm noktası arasındaki mesafe 1000 mm olarak verilmiştir. Bu test probleminde düğüm noktalarının sabitlendiği ve serbest bırakıldığı iki farklı durum incelenmiştir.

Düğüm noktalarının sabitlendiği durumda tüm düğüm noktalarının yatay koordinat değerleri sabit tutulmuş, düşey koordinat değerleri ise istenilen mimari formun korunabilmesi için verilen formüller ile hesaplanmıştır.

Düğüm noktalarının hareketli olduğu durumda ise sadece sabit yükün etki ettirildiği düğüm noktası ve mesnet bağlanan düğüm noktalarının yatay koordinat değerleri sabit tutulmuş olup diğer tüm düğüm noktalarının yatay koordinat değerlerinin değişimine izin verilmiştir. Düğüm noktalarının düşey koordinatları istenilen mimari formun korunabilmesi için verilen formüller ile hesaplanmıştır.

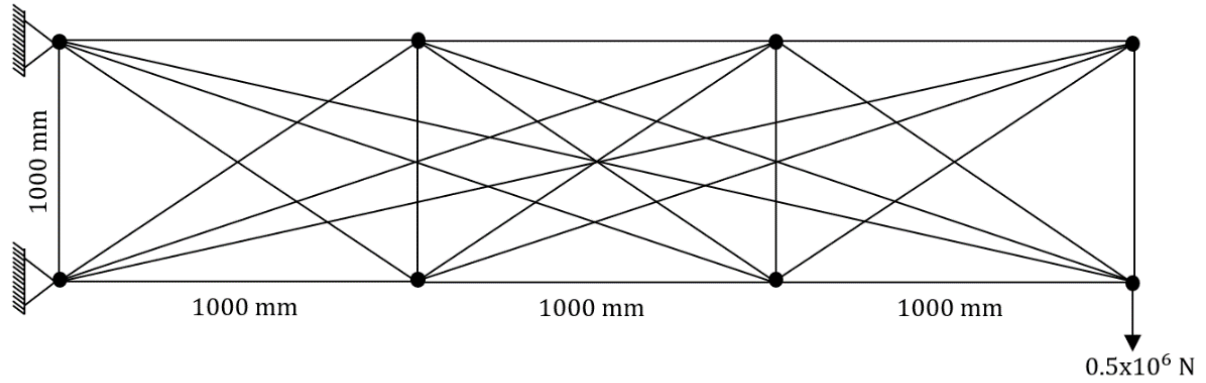
Üst düğüm noktalarının y koordinat değerini hesaplamak için Denklem 4.11 kullanılır:

$$fy_{Top}(x) = 1000.0 \quad (4.11)$$

Alt düğüm noktalarının y koordinat değerini hesaplamak için Denklem 4.12 kullanılır:

$$fy_{Bottom}(x) = 0.0 \quad (4.12)$$

#### 4.4.3 8 düğüm noktalı test problemi



Şekil 4.7. 8 düğüm noktalı kafes sistem için baz yapı

Şekil 4.7’de 8 düğüm noktalı kafes sistem örneğinin temel yapısı (muhtemel tüm düğümlerin ve elemanların bulunduğu konfigürasyon) görülmektedir. Üçüncü tasarım örneğinde de (Şekil 4.7) tüm yapım ve imalat kısıtları birinci tasarım örneği ile aynı olacak şekilde uygulanmıştır. Düğüm noktalarının sabitlendiği ve serbest bırakıldığı iki farklı durum incelenmiştir.

Bu test probleminde de 6 düğüm noktalı test probleminde olduğu gibi baz (temel) yapı aynı şekilde oluşturulmuş ve düğüm noktaları sabit ve hareketli olmak üzere iki farklı durum incelenmiştir.

Üst düğüm noktalarının y koordinat değerini hesaplamak için Denklem 4.13 kullanılır:

$$f_{yTop}(x) = 1000.0 \quad (4.13)$$

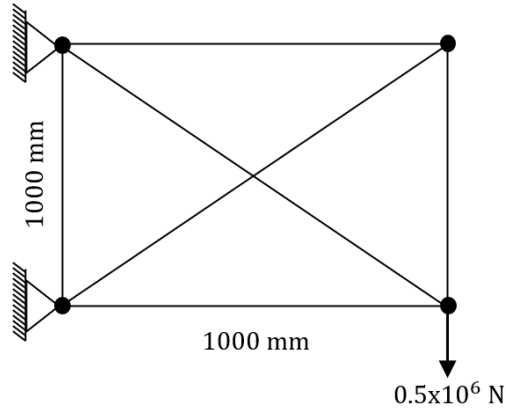
Alt düğüm noktalarının y koordinat değerini hesaplamak için Denklem 4.14 kullanılır:

$$fy_{Bottom}(x) = 0.0 \quad (4.14)$$

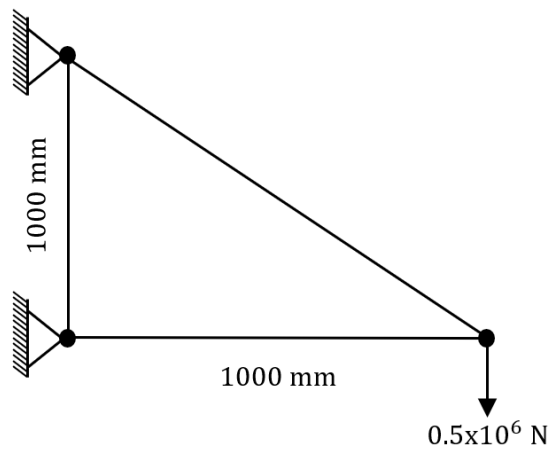


## 5. TASARIM VEKTÖRÜ

Tasarım vektörü, bir optimizasyon problemine ait çözümlerin elde edilmesini sağlayan tüm giriş parametrelerini içinde barındıran bir liste olarak tanımlanabilir. Giriş parametreleri sürekli ya da ayrık değişkenler olarak alınabilir. Optimizasyon aşamasında girdi türünü belirtmek önemlidir çünkü çoğu optimizasyon algoritması optimum tasarımı ararken girdi türlerini farklı şekilde ele alır.



Şekil 5.1. 4 düğüm noktalı kafes sistem için baz yapı



Şekil 5.2. 4 düğüm noktalı kafes sistem için bulunan en iyi tasarım

4 düğüm noktalı kafes sistem için tasarım vektörü şu şekilde verilmiştir:

$$[52.080, 58.523, -2.499, -0.572, 1.499, -2.308, -2.499, -0.186, 0.0, 1000.0, 0.0, 1000.0] \quad (5.1)$$

Grup Numarası (nGr)
Çubuk Elemanlar
Düğüm Noktaları  
x Koordinatları

Yazar tarafından bir imalat kısıtı olarak kabul edilen grup sayısı kısıtı bu örnek için 2 olarak alınmış olup, ilk iki değer eleman gruplarını göstermektedir. Şekil 5.1’de 4 düğüm noktalı test problemi için baz yapısı verilmiştir. Tasarım vektöründe (Denklem 5.1), baz yapısında bulunan tüm çubuk elemanlar için var olup olmadığına dair bir sayısal değer tanımlanmıştır. Bu sayısal değerlere göre (-) değer alan elemanlar sistemden kaldırılır. 0 ve 1 değerlerine doğru giden elemanlar ise aldığı değere göre (seçilen programlama dilinin yaptığı kabullerden dolayı 0 değeri için 1, 1 değeri için ise 2 numaralı grup anlaşılmalıdır.) o gruba ait profil kullanılarak sistemde yerini alır. Mesela elemanlar bölümüne bakıldığında ilk çubuk elemana ait değer -2.499 olduğu için baz yapıdaki 1 numaralı çubuk eleman sistemden kaldırılmıştır ancak üçüncü çubuk elemana ait değere bakıldığında değer 1.499 olduğu görülür; bu değer 1 tamsayısına yuvarlanır ve ilgili çubuğun 2 numaralı eleman grubuna ait bir çubuk eleman olduğuna işaret eder. 2 numaralı grup (58.52189  $\approx$  59) 59 numaralı profili temsil eder. Ancak daha önceden de söz edildiği üzere, seçilen programlama dilinde numaralandırmalar 0’dan başladığı için 60. profile ait bilgilerin alınması gerekir. Optimizasyon sonrasında elde edilen tasarım Şekil 5.2’de verilmiştir. Denklem 5.1’de verilen grup numaralarına denk gelen profiller Çizelge 5.1’de verilmiştir.

Çizelge 5.1. 53 ve 60 numaralı profillere ait veriler

Profil No	Eşit Kollu Korniyer Boyutları	Alan (mm <sup>2</sup> )
53	L 110 x 110 x 12	2510
60	L 120 x 120 x 16	3600

Tasarım vektörünün son bölümünde ise düğüm noktalarının x koordinatları verilmiştir. Bu test probleminde düğüm noktaları sabit tutulduğu için x koordinatları yazar tarafından verilen değerler ile aynı olup düğüm noktaları hareketli test problemlerindeki x koordinatları değişkenlik gösterebilmektedir.



## 6. BULGULAR VE TARTIŞMA

Bu çalışmada imal edilebilir kafes sistemler elde edebilmek için hem yapısal hem de imal edilebilirliğe dair kısıtlar altında boyut, şekil ve topoloji optimizasyonu gerçekleştirilmiştir. Optimizasyon sürecinde, 2016 yılında Ravipudi Venkata Rao tarafından ‘Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems’ isimli makalede tanıtılan parametresiz bir algoritma olan Jaya algoritması kullanılmış ve Python programlama dili kullanılarak üretilen yazılım ile problemler çözülmüştür. Tasarım ve analiz aşaması, dışarıdan hiçbir şekilde kullanıcı müdahalesi olmadan yani bilgisayar ile otomatik tasarım süreci ile gerçekleştirilmiştir.

Optimizasyon sürecinde, yazar tarafından üretilen her bir örnek için izin verilen analiz sayısı Çizelge 6.1’de verilmiştir:

Çizelge 6.1. İzin verilen analiz sayısı

Test Problemi	Analiz Sayısı
4 Düğüm Noktalı (Düğüm Noktaları Sabit)	$25 \times 750 = 18750$
6 Düğüm Noktalı (Düğüm Noktaları Sabit)	$25 \times 1000 = 25000$
6 Düğüm Noktalı (Düğüm Noktaları Hareketli)	$25 \times 1500 = 37500$
8 Düğüm Noktalı (Düğüm Noktaları Sabit)	$25 \times 3000 = 75000$
8 Düğüm Noktalı (Düğüm Noktaları Hareketli)	$25 \times 4000 = 100000$

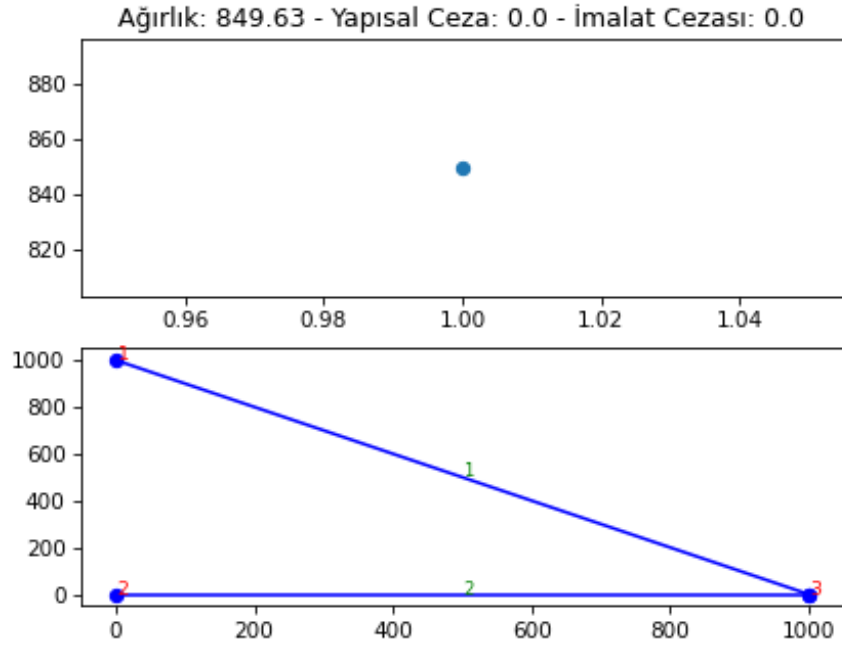
Tasarımlarda 2 farklı kesit kullanılmasına izin verilmiştir. Tüm test problemleri 20 kez çözülmüş olup çözümler ortalama olarak sırasıyla 3 dakika, 4 dakika, 8 dakika, 27 dakika ve 43 dakika sürmüştür. Uygulamada durma kriteri olarak ‘izin verilen en büyük iterasyonu’ kullanmak çoğu zaman iyi bir fikir olmamasına karşın bir optimizasyon sürecinin testi için son derece kullanışlıdır.

### 6.1 4 Dügüm Noktalı (Dügüm Noktaları Sabit) Test Problemi Sonuçları

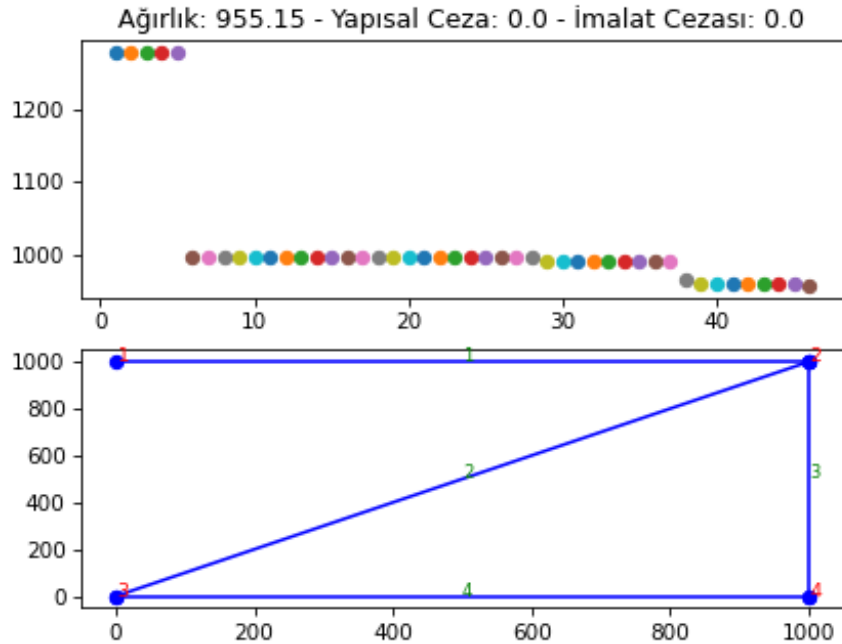
4 düğüm noktalı (düğüm noktaları sabit) test problemi için popülasyon büyüklüğü 25 ve izin verilen iterasyon sayısı 750 olarak seçilmiştir. Dügüm noktaları sabit bu test problemi için analiz sayısı  $25 \times 750 = 18750$  olarak alınmıştır. Ardışık 20 optimizasyon denemesinin sonuçları Çizelge 6.2’de verilmiştir.

Çizelge 6.2. 4 düğüm noktalı (düğüm noktaları sabit) test problemi optimizasyon sonuçları

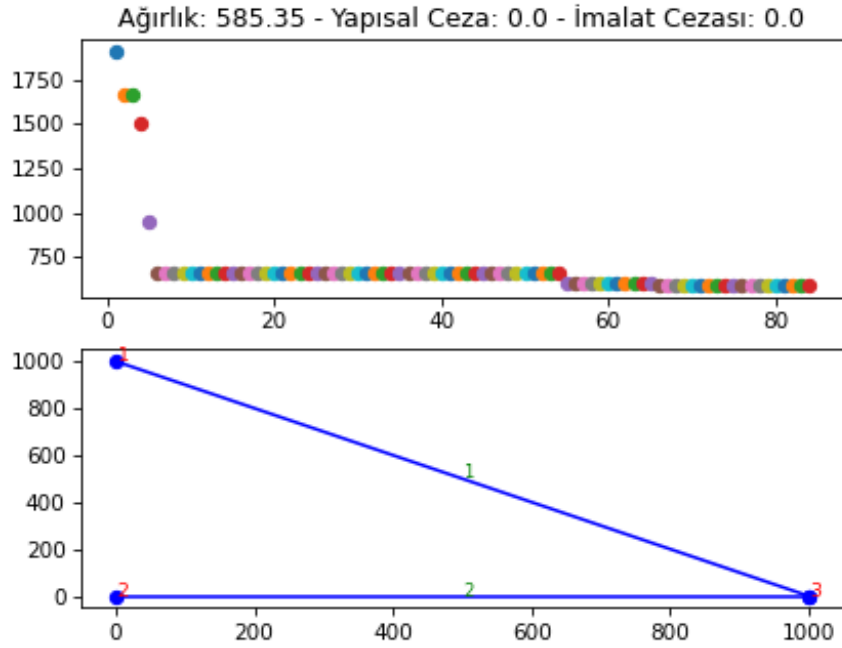
Tasarım No.	Ağırlık (N)	Yapısal Ceza	İmalat Cezası
1	585.35	0	0
2	955.15	0	0
3	585.35	0	0
4	585.35	0	0
5	686.54	0	0
6	669.29	0	0
7	669.29	0	0
8	585.35	0	0
9	669.29	0	0
10	669.29	0	0
11	699.04	0	0
12	585.35	0	0
13	585.35	0	0
14	585.35	0	0
15	585.35	0	0
16	669.29	0	0
17	585.35	0	0
18	585.35	0	0
19	955.15	0	0
20	669.29	0	0



Şekil 6.1. 4 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı



Şekil 6.2. 4 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım



Şekil 6.3. 4 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım

Çizelge 6.3. 4 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait veriler

Düğüm No.	(x) Koordinatı
1	0.0
2	1000.0
3	0.0
4	1000.0

Düğümünlerin y koordinatlarının neden verilmediğine dair açıklama için lütfen Bölüm 4.4.1'e bakınız.

Grup No	Kesit No
1	53
2	60

Çubuk No	Grup No
1-2	Kaldırıldı
1-3	Kaldırıldı
1-4	60
2-3	Kaldırıldı
2-4	Kaldırıldı
3-4	53

Çizelge 6.4. 4 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait eleman kesit alanları

Eleman No.	Kullanılan Profil	Alan (mm <sup>2</sup> )
1	L 120 x 120 x 16	3600
2	L 110 x 110 x 12	2510

Çizelge 6.2 incelendiğinde, yapısal ve imalat kısıtlarının tam olarak sağlandığı ve ağırlığın 669.29 N olduğu noktada bir lokal minimum olduğu görülmektedir. Söz edilen optimizasyon denemesine ait en iyi başlangıç tasarımının yapı ağırlığı 849.63 N olarak bulunmuştur ve tasarım Şekil 6.1’de verilmiştir. En iyi tasarıma ait yapı 1,3,4,8,12,13,14,15,17 ve 18. optimizasyon denemesinde bulunmuş olup Şekil 6.3’te gösterilmiştir. En iyi tasarıma ait yapı ağırlığı 585.35 N olarak bulunmuştur. Çizelge 6.3 ve 6.4’te en iyi tasarıma ait veriler paylaşılmıştır. Çözüme ait en kötü tasarım ise Şekil 6.2’de gösterilmiş ve ağırlığı 955.15 N olarak bulunmuştur. Çizelge 6.2’den anlaşılacağı üzere çoğu optimizasyon denemesinde ağırlık 585.35 N olarak hesaplanmıştır. Her ne kadar bir mühendislik sezisi olarak 585.35 N’luk çözümün optimal olduğu düşünülse de kesin bir şey söylemek doğru olmayacaktır.

## 6.2 6 Düğüm Noktalı (Düğüm Noktaları Sabit) Test Problemi Sonuçları

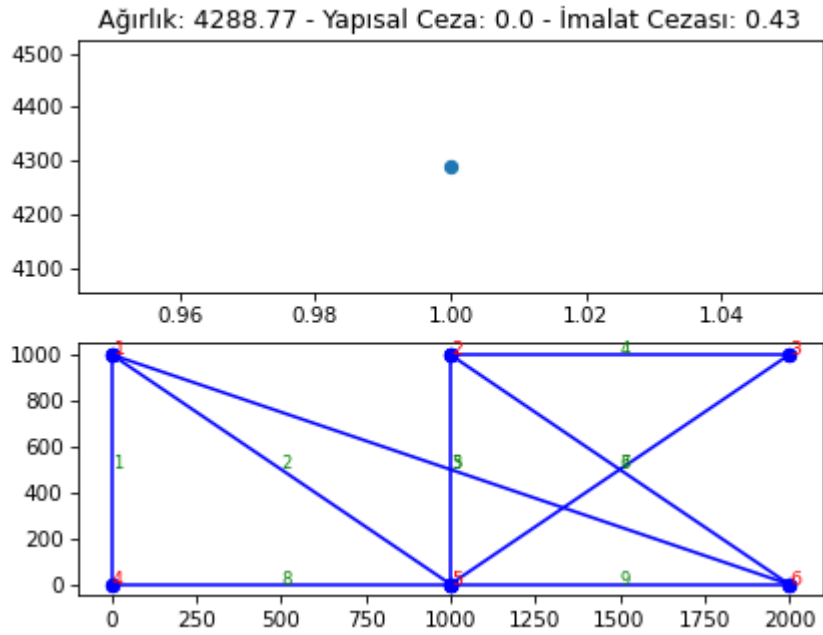
6 düğüm noktalı (düğüm noktaları sabit) test problemi için popülasyon büyüklüğü 25, izin verilen iterasyon sayısı ise 1000 olarak seçilmiştir. Düğüm noktaları sabit bu test problemi için analiz sayısı  $25 \times 1000 = 25000$  olarak alınmıştır. Ardışık 20 optimizasyon denemesinin sonuçları düğüm noktalarının sabit olduğu örnek için Çizelge 6.5’te verilmiştir.

Çizelge 6.5. 6 düğüm noktalı (düğüm noktaları sabit) test problemi optimizasyon sonuçları

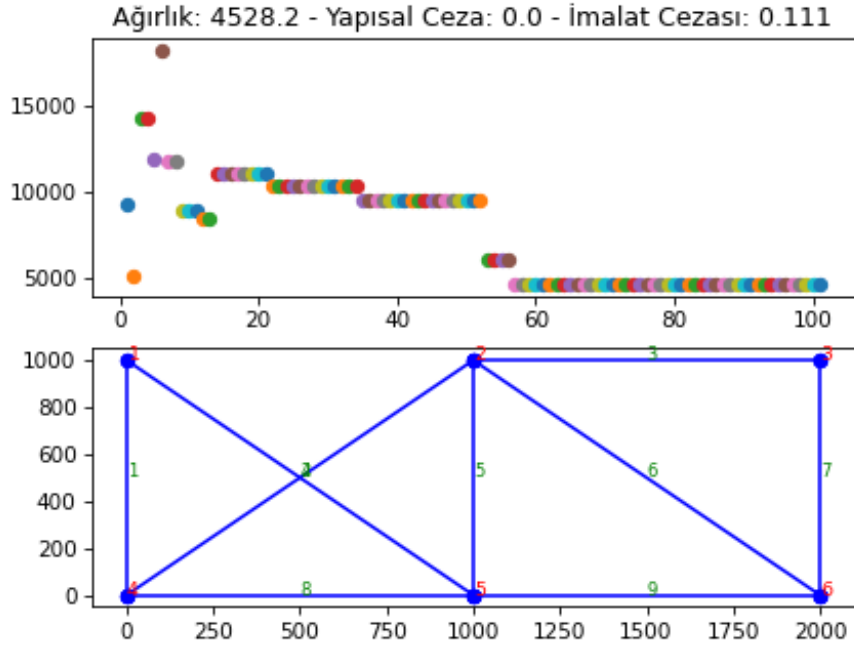
Tasarım No.	Ağırlık (N)	Yapısal Ceza	İmalat Cezası
1	1954.19	0	0.0545
2	2681.82	0	0
3	2820.43	0	0

Çizelge 6.5. 6 düğüm noktalı (düğüm noktaları sabit) test problemi optimizasyon sonuçları (devam)

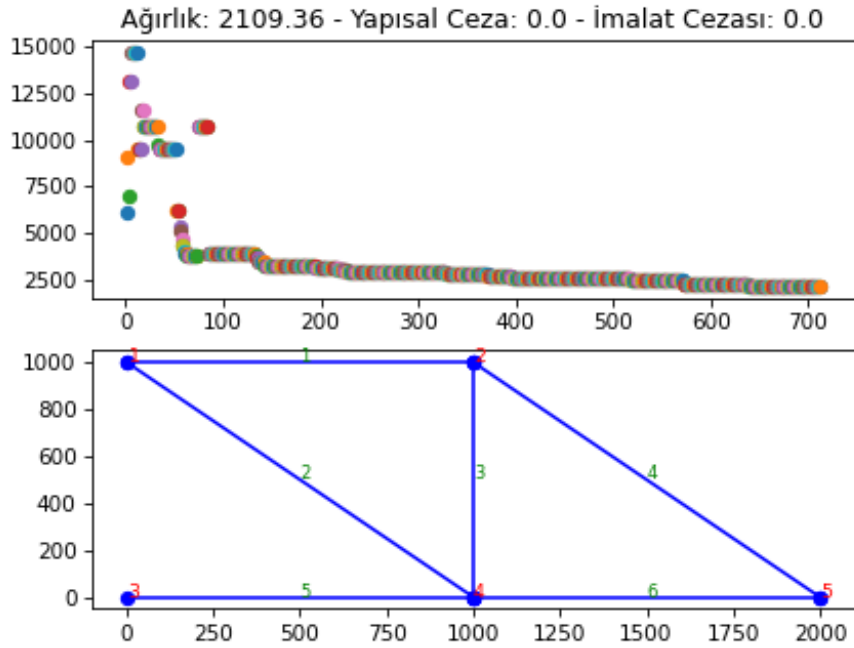
4	2995.95	0	0.0545
5	2997.08	0	0
6	2314.95	0	0.0417
7	2121.19	0	0
8	2680.28	0	0
9	3608.09	0	0
10	3299.77	0	0
11	2669.50	0	0
12	2811.19	0	0
13	2109.36	0	0
14	2497.48	0	0
15	2282.92	0	0
16	2669.50	0	0
17	4528.20	0	0.1111
18	3025.07	0	0
19	3071.48	0	0
20	3083.19	0	0



Şekil 6.4. 6 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı



Şekil 6.5. 6 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım



Şekil 6.6. 6 düğüm noktalı (düğüm noktaları sabit) test problemi için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım

Çizelge 6.6. 6 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait veriler

Düğüm No.	(x) Koordinatı
1	0.0
2	1000.0
3	2000.0
4	0.0
5	1000.0
6	2000.0
Düğümlerin y koordinatlarının neden verilmediğine dair açıklama için lütfen bölüm 4.4.2'ye bakınız.	
Grup No	Kesit No
1	61
2	77
Çubuk No	Grup No
1-2	Kaldırıldı
1-3	Kaldırıldı
1-4	Kaldırıldı
1-5	2
1-6	Kaldırıldı
2-3	Kaldırıldı
2-4	Kaldırıldı
2-5	1
2-6	2
3-4	Kaldırıldı
3-5	Kaldırıldı
3-6	Kaldırıldı
4-5	2
4-6	Kaldırıldı
5-6	Kaldırıldı

Çizelge 6.7. 6 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarımın eleman kesit alanları

Eleman No.	Kullanılan Profil	Alan (mm <sup>2</sup> )
1	L 130 x 130 x 10	2520
2	L 160 x 160 x 17	5180
3	L 130 x 130 x 10	2520
4	L 160 x 160 x 17	5180
5	L 160 x 160 x 17	5180
6	L 130 x 130 x 10	2520

6 düğüm noktalı (düğüm noktaları sabit) test problemi için tüm kısıtları sağlayan en hafif tasarım 13. optimizasyon denemesinde bulunmuştur ve ağırlığı 2109.36 N'dur (Şekil 6.6). Çizelge 6.6 ve 6.7'de en iyi tasarıma ait veriler paylaşılmıştır. En iyi başlangıç tasarımına ait yapı ağırlığı 4288.77 N olarak bulunmuş ve Şekil 6.4'te verilmiştir. Çözüme ait en kötü tasarım ise Şekil 6.5'te gösterilmiş ve ağırlığı 4528.2 N olarak bulunmuştur. Çizelge 6.5'te görüleceği üzere tüm optimizasyon denemelerinde yapısal cezası olmayan tasarımlar elde edilebilirken denemelerin dördünde imalat cezası olmayan bir tasarım adayı elde edilememiştir.

### 6.3 6 Düğüm Noktalı (Düğüm Noktaları Hareketli) Test Problemi Sonuçları

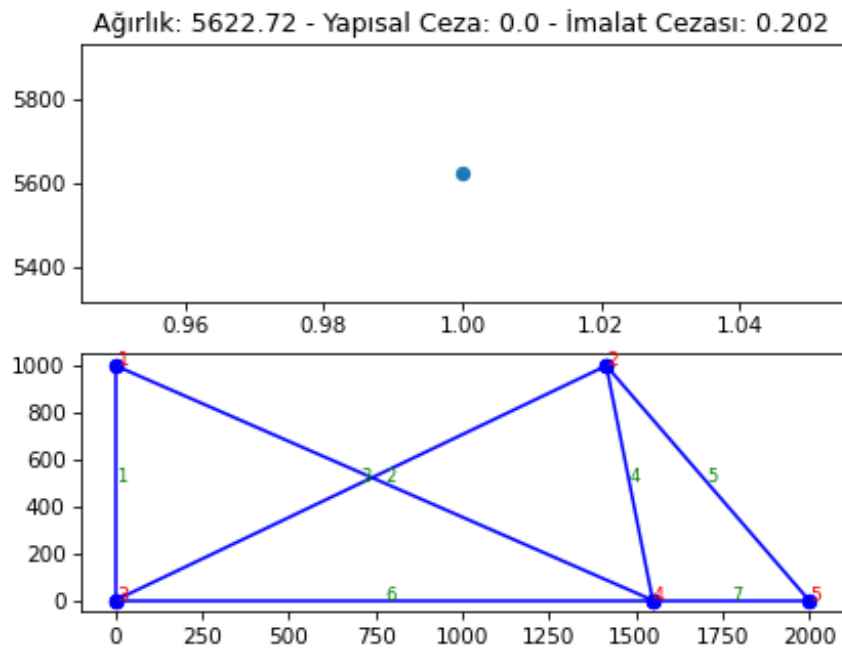
6 düğüm noktalı (düğüm noktaları hareketli) test problemi için popülasyon büyüklüğü 25, iterasyon sayısı ise 1500 olarak seçilmiştir. Düğüm noktaları hareketli bu test problemini için analiz sayısı  $25 \times 1500 = 37500$  olarak alınmıştır. Ardışık 20 optimizasyon denemesinin sonuçları düğüm noktalarının hareketli olduğu örnek için Çizelge 6.8'de verilmiştir.

Çizelge 6.8. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) optimizasyon sonuçları

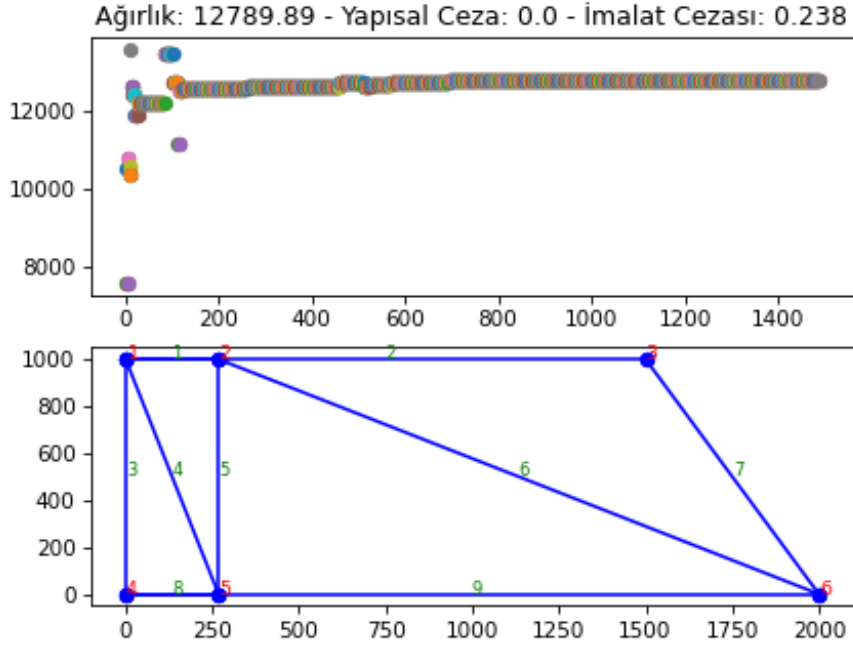
Tasarım No.	Ağırlık (N)	Yapısal Ceza	İmalat Cezası
1	3656.25	0	0.1030
2	5941.82	0	0
3	1947.23	0	0.1030
4	2338.35	0	0
5	2227.67	0	0
6	12789.89	0	0.2378
7	2323.72	0	0
8	2097.35	0	0
9	2708.94	0	0
10	2244.42	0	0
11	2235.55	0	0
12	2598.77	0	0

Çizelge 6.8. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) optimizasyon sonuçları (devam)

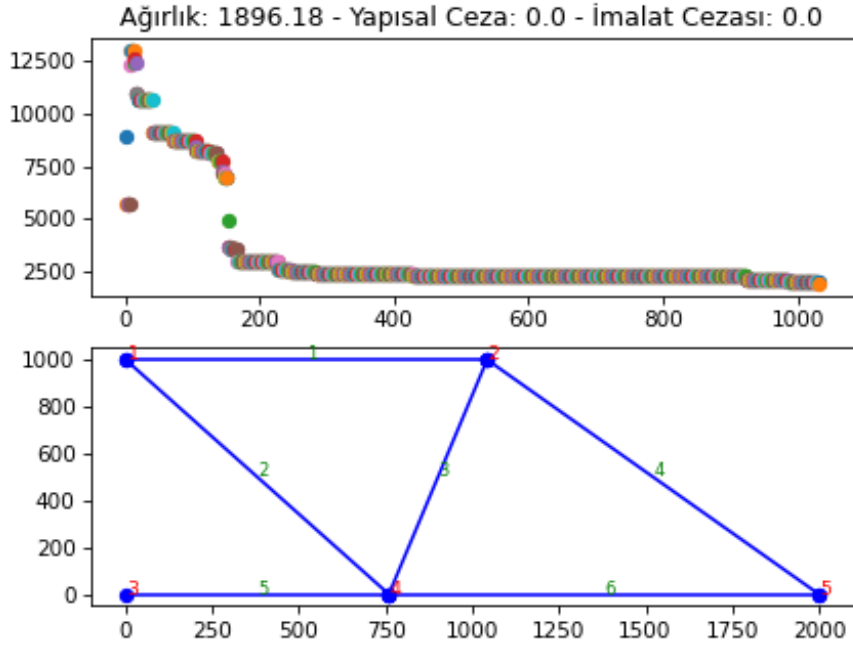
13	2664.07	0	0.1030
14	3612.53	0	0
15	2253.28	0	0
16	2656.25	0	0.0476
17	2288.50	0	0
18	2212.55	0	0
19	2781.45	0	0
20	1896.18	0	0



Şekil 6.7. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı



Şekil 6.8. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım



Şekil 6.9. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım

Çizelge 6.9. 6 düğüm noktalı (düğüm noktaları hareketli) test problemi en iyi tasarıma ait veriler

Düğüm No.	(x) Koordinatı
1	0.0
2	1388.187
3	1042.783
4	0.0
5	759.314
6	2000.0
Düğümlerin y koordinatlarının neden verilmediğine dair açıklama için lütfen bölüm 4.4.2'ye bakınız.	
Grup No	Kesit No
1	72
2	67
Çubuk No	Grup No
1-2	Kaldırıldı
1-3	1
1-4	Kaldırıldı
1-5	1
1-6	Kaldırıldı
2-3	Kaldırıldı
2-4	Kaldırıldı
2-5	Kaldırıldı
2-6	Kaldırıldı
3-4	Kaldırıldı
3-5	1
3-6	1
4-5	2
4-6	Kaldırıldı
5-6	1

Çizelge 6.10. 6 düğüm noktalı test problemi (düğüm noktaları hareketli) en iyi tasarımın eleman kesit alanları

Eleman No.	Kullanılan Profil	Alan (mm <sup>2</sup> )
1	L 150 x 150 x 18	5100
2	L 150 x 150 x 18	5100
3	L 150 x 150 x 18	5100
4	L 150 x 150 x 18	5100
5	L 150 x 150 x 12	3480
6	L 150 x 150 x 18	5100

6 düğüm noktalı (düğüm noktaları hareketli) test problemi için tüm kısıtları sağlayan en hafif tasarımın ağırlığı 1896.18 N olarak bulunmuştur (Şekil 6.9). Çizelge 6.9 ve 6.10'da en iyi tasarıma ait veriler paylaşılmıştır. En iyi başlangıç tasarımına ait yapı ağırlığı ise 5622.72 N olarak bulunmuş ve Şekil 6.7'de verilmiştir. Çözüme ait en kötü tasarım ise Şekil 6.8'de gösterilmiş ve ağırlığı 12789.89 N olarak bulunmuştur. Çizelge 6.8'de görüleceği üzere tüm optimizasyon denemelerinde yapısal cezası olmayan tasarımlar elde edilebilirken denemelerin beşinde imalat cezası olmayan bir tasarım adayı elde edilememiştir.

#### 6.4 8 Düğüm Noktalı (Düğüm Noktaları Sabit) Test Problemi Sonuçları

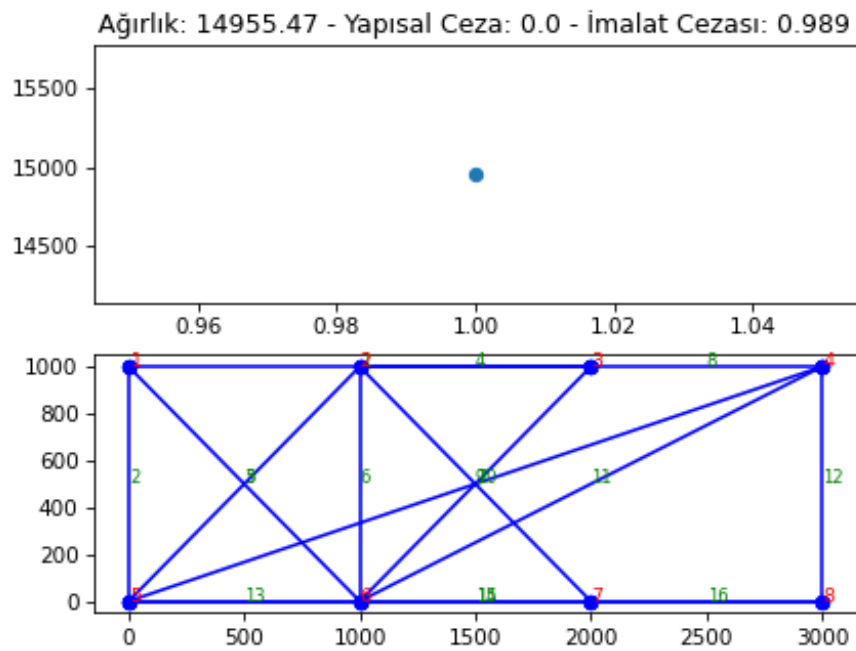
8 düğüm noktalı (düğüm noktaları sabit) test problemi için popülasyon büyüklüğü 25, iterasyon sayısı ise 3000 olarak seçilmiştir. Düğüm noktaları sabit bu test problemi için analiz sayısı  $25 \times 3000 = 75000$  olarak alınmıştır. Ardışık 20 optimizasyon denemesinin sonuçları düğüm noktalarının sabit olduğu örnek için Çizelge 6.11'de verilmiştir.

Çizelge 6.11. 8 düğüm noktalı test problemi (düğüm noktaları sabit) optimizasyon sonuçları

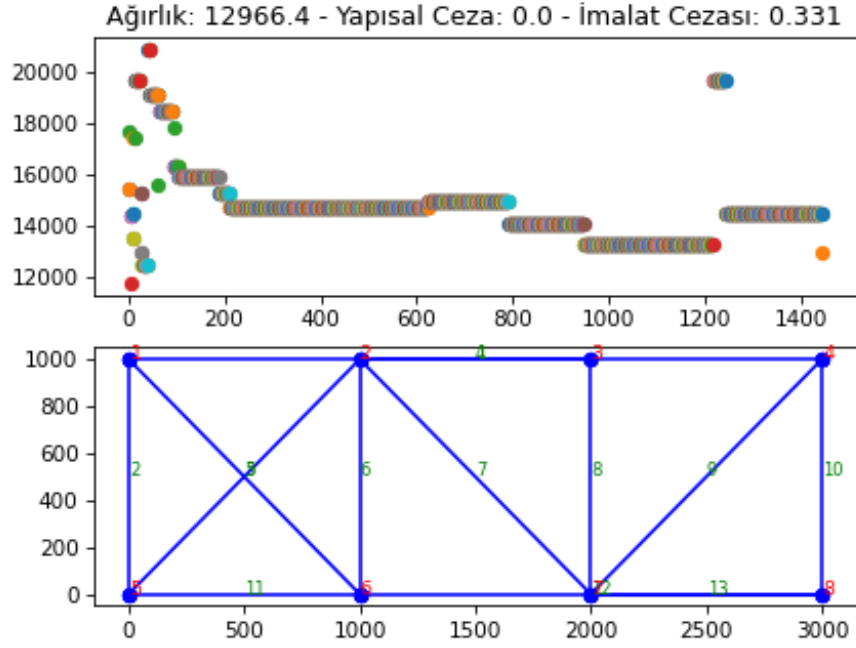
Tasarım No.	Ağırlık (N)	Yapısal Ceza	İmalat Cezası
1	7118.24	0	0.1002
2	8465.14	0	0.2250
3	9953.21	0	0.0660
4	5972.22	0	0
5	12966.40	0	0.3312
6	6554.39	0	0.2868
7	11645.05	0	0
8	6502.70	0	0
9	8050.30	0	0.2836
10	8141.33	0	0.1026
11	7436.81	0	0
12	9940.89	0	0
13	14624.32	0	0.0313

Çizelge 6.11. 8 düğüm noktalı test problemi (düğüm noktaları sabit) optimizasyon sonuçları (devam)

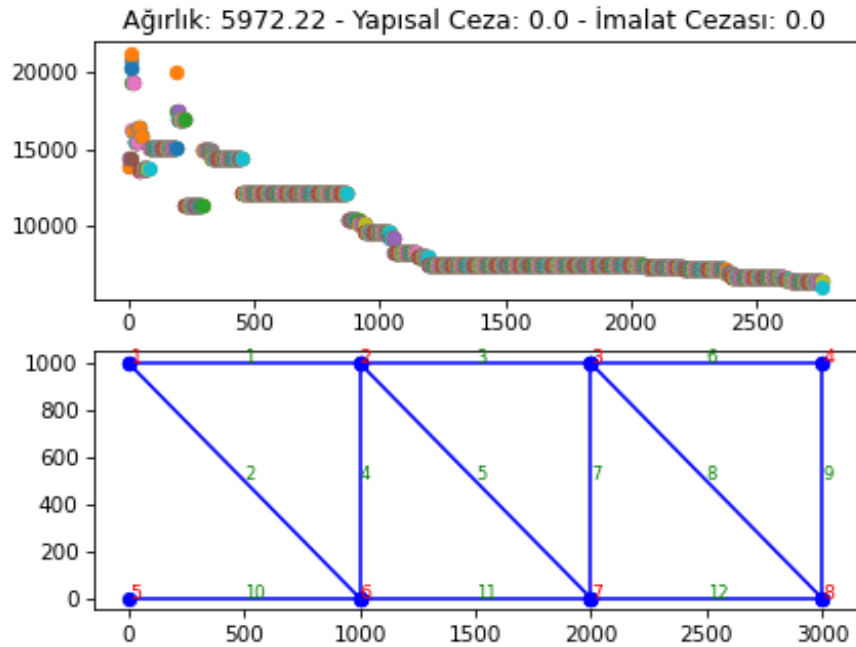
14	6837.41	0	0.0721
15	12246.98	0	0.2051
16	4993.04	0	0.1297
17	6411.40	0	0.0690
18	6046.81	0	0
19	8802,95	0	0.1757
20	9832.23	0	0.1403



Şekil 6.10. 8 düğüm noktalı test problemi (düğüm noktaları sabit) için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı



Şekil 6.11. 8 düğüm noktalı test problemi (düğüm noktaları sabit) için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım



Şekil 6.12. 8 düğüm noktalı test problemi (düğüm noktaları sabit) için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım

Çizelge 6.12. 8 düğüm noktalı (düğüm noktaları sabit) test problemi en iyi tasarıma ait veriler

Düğüm No.	(x) Koordinatı
1	0.0
2	1000.0
3	2000.0
4	3000.0
5	0.0
6	1000.0
7	2000.0
8	3000.0
Düğümlerin y koordinatlarının neden verilmediğine dair açıklama için lütfen bölüm 4.4.3'e bakınız.	
Grup No	Kesit No
1	91
2	60
Çubuk No	Grup No
1-2	2
1-3	Kaldırıldı
1-4	Kaldırıldı
1-5	Kaldırıldı
1-6	2
1-7	Kaldırıldı
1-8	Kaldırıldı
2-3	1
2-4	Kaldırıldı
2-5	Kaldırıldı
2-6	1
2-7	1
2-8	Kaldırıldı
3-4	1
3-5	Kaldırıldı
3-6	Kaldırıldı
3-7	2
3-8	1
4-5	Kaldırıldı
4-6	Kaldırıldı
4-7	Kaldırıldı
4-8	2
5-6	2
5-7	Kaldırıldı
5-8	Kaldırıldı
6-7	2
6-8	Kaldırıldı
7-8	2

Çizelge 6.13. 8 düğüm noktalı test problemi (düğüm noktaları sabit) en iyi tasarımın eleman kesit alanları

Eleman No.	Kullanılan Profil	Alan (mm <sup>2</sup> )
1	L 120 x 120 x 16	3600
2	L 120 x 120 x 16	3600
3	L 200 x 200 x 20	7630
4	L 200 x 200 x 20	7630
5	L 200 x 200 x 20	7630
6	L 200 x 200 x 20	7630
7	L 120 x 120 x 16	3600
8	L 200 x 200 x 20	7630
9	L 120 x 120 x 16	3600
10	L 120 x 120 x 16	3600
11	L 120 x 120 x 16	3600
12	L 120 x 120 x 16	3600

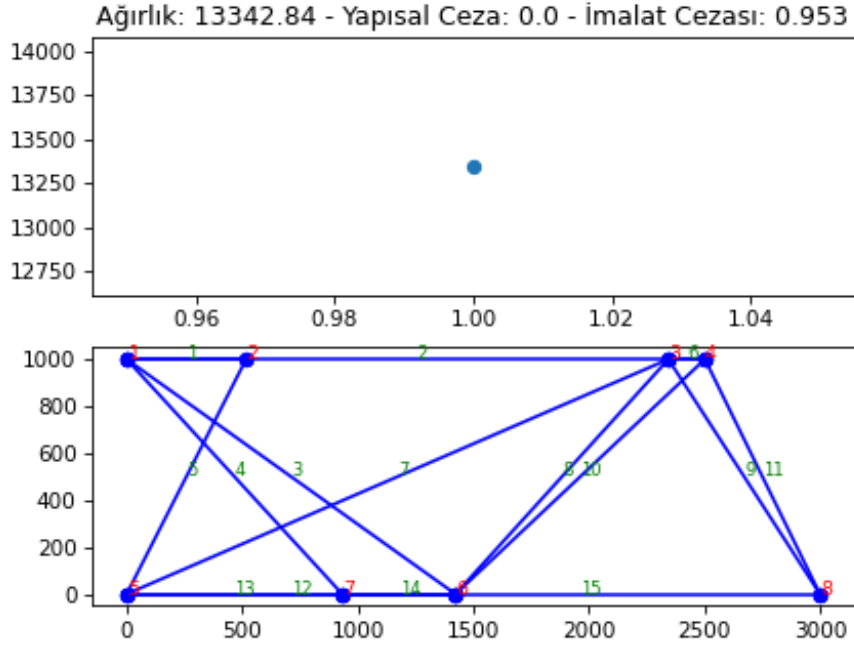
8 düğüm noktalı (düğüm noktaları sabit) test problemi için tüm kısıtları sağlayan en hafif tasarım 4. optimizasyon denemesinde bulunmuştur ve ağırlığı 5972.22 N'dur (Şekil 6.12). Çizelge 6.12 ve 6.13'te en iyi tasarıma ait veriler paylaşılmıştır. En iyi başlangıç tasarımına ait yapı ağırlığı 14955.47 N olarak bulunmuş ve Şekil 6.10'da verilmiştir. En iyi başlangıç tasarımında yapı kısıtları tam olarak sağlansa da imalat kısıtları tam olarak sağlanamamıştır. Çözüme ait en kötü tasarım ise Şekil 6.11'de gösterilmiş ve ağırlığı 12966.4 N olarak bulunmuştur. Çizelge 6.11'de görüleceği üzere tüm optimizasyon denemelerinde yapısal cezası olmayan tasarımlar elde edilebilirken denemelerin on dördünde imalat cezası olmayan bir tasarım adayı elde edilememiştir.

### 6.5 8 Düğüm Noktalı (Düğüm Noktaları Hareketli) Test Problemi Sonuçları

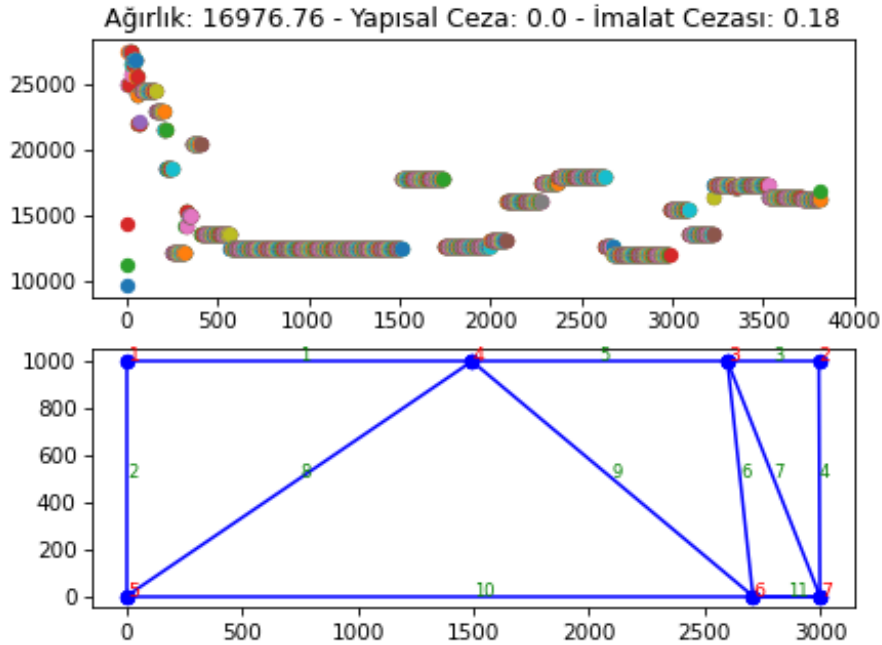
8 düğüm noktalı (düğüm noktaları hareketli) test problemi için popülasyon büyüklüğü 25, iterasyon sayısı ise 4000 olarak seçilmiştir. Düğüm noktaları hareketli bu test problemi için analiz sayısı  $25 \times 4000 = 100000$  olarak alınmıştır. Ardışık 20 optimizasyon denemesinin sonuçları düğüm noktalarının sabit olduğu örnek için Çizelge 6.14'te verilmiştir.

Çizelge 6.14. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) optimizasyon sonuçları

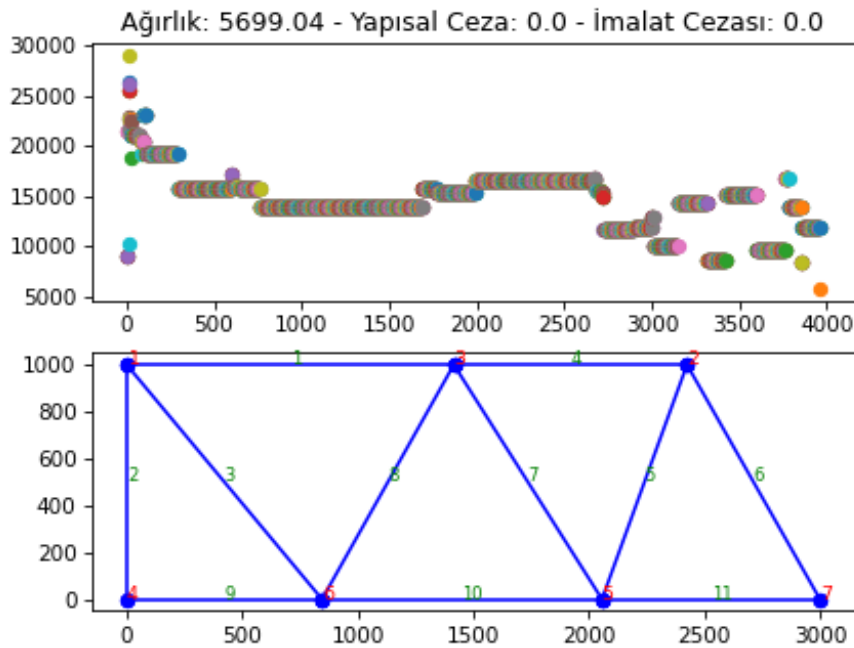
Tasarım No.	Ağırlık (N)	Yapısal Ceza	İmalat Cezası
1	15411.71	0	0.0986
2	18173.51	0	0.0468
3	17623.90	0	0.0573
4	7048.98	0	0
5	13086.00	0	0.0313
6	5914.58	0	0
7	16498.33	0	0.0747
8	6227.81	0	0
9	6520.38	0	0
10	17101.77	0	0.1455
11	13300.32	0	0.0313
12	13885.29	0	0
13	16976.76	0	0.1800
14	17268.74	0	0.0403
15	13921.59	0	0.0221
16	7111.23	0	0.0313
17	17016.51	0	0.1667
18	5699.04	0	0
19	6784.48	0	0
20	12849.49	0	0



Şekil 6.13. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi başlangıç tasarımı



Şekil 6.14. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en kötü tasarım



Şekil 6.15. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) için yapılan ardışık 20 optimizasyon denemesine ait en iyi tasarım

Çizelge 6.15. 8 düğüm noktalı (düğüm noktaları hareketli) test problemi en iyi tasarıma ait veriler

Düğüm No.	(x) Koordinatı
1	0.0
2	2957.753
3	2424.806
4	1415.795
5	0.0
6	2056.872
7	845.431
8	3000.0
Düğümlerin y koordinatlarının neden verilmediğine dair açıklama için lütfen bölüm 4.4.3'e bakınız.	
Grup No	Kesit No
1	65
2	91
Çubuk No	Grup No
1-2	Kaldırıldı
1-3	Kaldırıldı
1-4	2

Çizelge 6.15. 8 düğüm noktalı (düğüm noktaları hareketli) test problemi en iyi tasarıma ait veriler (devam)

1-5	1
1-6	Kaldırıldı
1-7	2
1-8	Kaldırıldı
2-3	Kaldırıldı
2-4	Kaldırıldı
2-5	Kaldırıldı
2-6	Kaldırıldı
2-7	Kaldırıldı
2-8	Kaldırıldı
3-4	2
3-5	Kaldırıldı
3-6	1
3-7	Kaldırıldı
3-8	2
4-5	Kaldırıldı
4-6	1
4-7	1
4-8	Kaldırıldı
5-6	Kaldırıldı
5-7	2
5-8	Kaldırıldı
6-7	2
6-8	1
7-8	Kaldırıldı

Çizelge 6.16. 8 düğüm noktalı test problemi (düğüm noktaları hareketli) en iyi tasarımın eleman kesit alanları

Eleman No.	Kullanılan Profil	Alan (mm <sup>2</sup> )
1	L 200 x 200 x 20	7630
2	L 130 x 130 x 16	3930
3	L 200 x 200 x 20	7630
4	L 200 x 200 x 20	7630
5	L 130 x 130 x 16	3930
6	L 200 x 200 x 20	7630
7	L 130 x 130 x 16	3930
8	L 130 x 130 x 16	3930
9	L 200 x 200 x 20	7630
10	L 200 x 200 x 20	7630
11	L 130 x 130 x 16	3930

8 düğüm noktalı (düğüm noktaları hareketli) test problemi için tüm kısıtları sağlayan en hafif tasarım 18. optimizasyon denemesinde bulunmuştur ve ağırlığı 5699.04 N'dur (Şekil 6.15). Çizelge 6.15 ve 6.16'da en iyi tasarıma ait veriler paylaşılmıştır. En iyi başlangıç tasarımına ait yapı ağırlığı 13342.84 N olarak bulunmuş ve Şekil 6.13'te verilmiştir. En iyi başlangıç tasarımında yapı kısıtları tam olarak sağlansa da imalat kısıtları tam olarak sağlanamamıştır. Çözüme ait en kötü tasarım ise Şekil 6.14'te gösterilmiş ve ağırlığı 16976.76 N olarak bulunmuştur. Çizelge 6.14'te görüleceği üzere tüm optimizasyon denemelerinde yapısal cezası olmayan tasarımlar elde edilebilirken denemelerin on ikisinde imalat cezası olmayan bir tasarım adayı elde edilememiştir.



## 7. SONUÇ VE ÖNERİLER

Bu çalışmada düzlem kafes sistemlerin imalat kısıtları altında bilgisayar ile otomatik tasarımının hesaplama yükü konusu irdelenmiştir. Elde edilen sonuçlar aşağıda maddeler halinde özetlenmiştir.

- İlk örnek (Bkz. Şekil 4.5) için yapılan 20 ardışık optimizasyon denemesinin 10'unda algoritma 585.35 N noktasında bir minimum değere ulaşmıştır. Bu değer global minimum olduğu düşünülmektedir.
- Bir optimizasyon prosedürünün başarı ölçütlerinden biri de yapılan ardışık denemelerde bulunan sonuçların birbirine ne kadar yakın olduğudur. 6 düğüm noktalı örneğin (Bkz. Şekil 4.6) düğüm noktaları sabit durum için 25000 analize izin verilmesine rağmen bulunan çözümlerin ağırlıkları 2109.36 N ile 4528.20 N arasında değişmektedir. Benzer şekilde aynı örneğin düğüm noktaları hareketli durumu için 37500 analize izin verilmiştir ve 1896.18 N ile 12789.89 N arasında değişen ağırlıklara sahip çözümler elde edilmiştir. Aynı zamanda düğüm noktaları sabit örnekte 4, düğüm noktaları hareketli örnekte 5 çözümde imalat kısıtları sağlanamamıştır. İncelenen örneğin basitliği göz önüne alındığında elde edilen çözümler arasındaki fark kabul edilemez seviyededir.
- 6 düğüm noktalı ve 8 düğüm noktalı problemlerin düğüm noktalarının hareketine izin vermeden yürütülen optimizasyon süreçlerinin sonucunda (boyut ve topoloji optimizasyonu) minimum yapı ağırlıkları sırasıyla 2109.36 N ve 5972.22 N olarak bulunmuştur. Aynı örnekler için düğüm noktalarının hareketine izin verilmesi durumunda (boyut, şekil ve topoloji optimizasyonu) bulunan minimum yapı ağırlıkları ise sırasıyla 1896.18 N ve 5699.04 N olarak bulunmuştur. Beklendiği üzere düğüm noktalarının hareketine izin verilmesi görece daha iyi çözümlere ulaşılabilmesini sağlamıştır.
- 8 düğüm noktalı düğüm noktaları sabit örneğin bulunabilen en iyi tasarımında (Bkz. Şekil 6.12) 6 ve 9 numaralı elemanların ve 8 düğüm noktalı düğüm noktaları hareketli örneğin bulunabilen en iyi tasarımında da (Bkz. Şekil 6.15) 2 numaralı elemanın

gereksiz olduđu düşünölmektedir. Sonuç olarak, 8 düğüm noktalı (düğüm noktaları sabit ve hareketli) örneklerde optimal topolojiye ulaşamadığı düşünölmektedir.

- Elde edilen tüm verilere bakıldığında, yapısal ve imalat kısıtları altında kafes sistemlerin bir metasezgisel algoritma kullanılarak yapılan eşzamanlı boyut, şekil ve topoloji optimizasyonunun hesaplama yükünün oldukça yüksek olduđu görölmüşür.
- Büyük ölçekli kafes sistemlerin yapısal ve imalat kısıtları altında bilgisayar ile otomatik tasarımı için sadece bir metasezgisel algoritma ile bir ceza yöntemini birleştirerek oluşturulacak süreçler ile ev ve işyeri bilgisayarlarında (şimdilik) makul sürelerde optimale yakın çözümler elde edilemeyeceği fikri oluşmuştur.

## KAYNAKLAR DİZİNİ

- Ahrari, A., Deb, K., 2016, An improved fully stressed design evolution strategy for layout optimization of truss structures, *Computers and Structures*, 164, 127–144. <https://doi.org/10.1016/j.compstruc.2015.11.009>
- Assimi, H., Jamali, A., Nariman-zadeh, N., 2017, Sizing and topology optimization of truss structures using genetic programming, *Swarm and Evolutionary Computation*, 37, 90–103. <https://doi.org/10.1016/j.swevo.2017.05.009>
- Assimi, H., Jamali, A., Nariman-zadeh, N., 2019, Multi-objective sizing and topology optimization of truss structures using genetic programming based on a new adaptive mutant operator, *Neural Computing and Applications*, 31(10), 5729–5749. <https://doi.org/10.1007/s00521-018-3401-9>
- Bakhtiary, N., Allinger, P., Friedrich, M., Mulfinger, F., Sauter, J., Puchinger, M., 1996, A New Approach for Sizing, Shape and Topology Optimization, *Journal of Materials & Manufacturing*, 105, 745-761. <https://doi.org/10.4271/960814>
- Bayram Durna, M., Veri Bilimi İçin Temel Python Kütüphaneleri-1 : Numpy, <https://medium.com/bili%C5%9Fim-hareketi/veri-bilimi-i-CC%87%C3%A7in-temel-python-k%C3%BCt%C3%BCphaneleri-1-numpy-750429a0d8e5>, erişim tarihi: 25.03.2022
- Bellagamba, L., Yang, T. Y., 1981, Minimum-mass truss structures with constraints on fundamental natural frequency, *AIAA Journal*, 19(11), 1452–1458. <https://doi.org/10.2514/3.7875>
- Chen, S., Shui, X., Huang, H., 2017, Improved genetic algorithm with two-level approximation using shape sensitivities for truss layout optimization, *Structural and Multidisciplinary Optimization*, 55(4), 1365–1382. <http://doi.org/10.1007/s00158-016-1583-2>

**KAYNAKLAR DİZİNİ (devam)**

- Cui, H., An, H., Huang, H., 2018, Truss topology optimization considering local buckling constraints and restrictions on intersection and overlap of bar members, *Structural and Multidisciplinary Optimization*, 58(2), 575–594. <http://doi.org/10.1007/s00158-018-1910-x>
- Deb, K., Gulati, S., 2001, Design of truss-structures for minimum weight using genetic algorithms, *Finite Elements in Analysis and Design*, 37, 447-465. [https://doi.org/10.1016/S0168-874X\(00\)00057-3](https://doi.org/10.1016/S0168-874X(00)00057-3)
- Degertekin, S. O., Lamberti, L., Ugur, I. B., 2018, Sizing, layout and topology design optimization of truss structures using the Jaya algorithm, *Applied Soft Computing Journal*, 70, 903–928. <https://doi.org/10.1016/j.asoc.2017.10.001>
- Degertekin, S. O., Lamberti, L., Ugur, I. B., 2019, Discrete sizing/layout/topology optimization of truss structures with an advanced Jaya algorithm, *Applied Soft Computing Journal*, 79, 363–390. <https://doi.org/10.1016/j.asoc.2019.03.058>
- Dillen, W., Lombaert, G., Schevenels, M., 2021, A hybrid gradient-based/metaheuristic method for Eurocode-compliant size, shape and topology optimization of steel structures, *Engineering Structures*, 239, 112137. <https://doi.org/10.1016/j.engstruct.2021.112137>
- Dominguez, A., Stiharu, I., Sedaghati, R., 2006, Practical design optimization of truss structures using the genetic algorithms, *Research in Engineering Design*, 17(2), 73–84. <https://doi.org/10.1007/s00163-006-0020-8>
- El Bouzouiki, M., Sedaghati, R., Stiharu, I., 2021, A non-uniform cellular automata framework for topology and sizing optimization of truss structures subjected to stress and displacement constraints, *Computers and Structures*, 242, 106394. <https://doi.org/10.1016/j.compstruc.2020.106394>

**KAYNAKLAR DİZİNİ (devam)**

- Fairclough, H., Gilbert, M., 2020, Layout optimization of simplified trusses using mixed integer linear programming with runtime generation of constraints, *Structural and Multidisciplinary Optimization*, 61(5), 1977–1999. <https://doi.org/10.1007/s00158-019-02449-7>
- Fiore, A., Marano, G. C., Greco, R., Mastromarino, E., 2016, Structural optimization of hollow-section steel trusses by differential evolution algorithm, *International Journal of Steel Structures*, 16(2), 411–423. <https://doi.org/10.1007/s13296-016-6013-1>
- Frans, R., Arfiadi, Y., 2014, Sizing, shape, and topology optimizations of roof trusses using hybrid genetic algorithms, *Procedia Engineering*, 95, 185–195. <https://doi.org/10.1016/j.proeng.2014.12.178>
- Fu, Z., Wang, C., Zhao, J., 2017, Truss topology optimization under uncertain nodal locations with proportional topology optimization method, *Mechanics Based Design of Structures and Machines*, 45(2), 190–206. <https://doi.org/10.1080/15397734.2016.1163640>
- Gao, G., Liu, Z., Li, Y., Qiao, Y., 2017, A new method to generate the ground structure in truss topology optimization, *Engineering Optimization*, 49(2), 235–251. <https://doi.org/10.1080/0305215X.2016.1169050>
- Hamza, K., Mahmoud, H., Saitou, K., 2003, Design optimization of N-shaped roof trusses using reactive taboo search, *Applied Soft Computing Journal*, 3(3), 221–235. [https://doi.org/10.1016/S1568-4946\(03\)00036-X](https://doi.org/10.1016/S1568-4946(03)00036-X)
- Hayashi, K., Ohsaki, M., 2019, FDMopt: Force density method for optimal geometry and topology of trusses, *Advances in Engineering Software*, 133(April), 12–19. <https://doi.org/10.1016/j.advengsoft.2019.04.002>

**KAYNAKLAR DİZİNİ (devam)**

- Hooshmand, A., Campbell, M. I., 2016, Truss layout design and optimization using a generative synthesis approach, *Computers and Structures*, 163, 1–28. <http://doi.org/10.1016/j.compstruc.2015.09.010>
- Kanno, Y., 2018a, Mixed-integer second-order cone programming for truss topology optimization with self-weight load and limitation on number of nodes, *IEEE International Conference on Industrial Engineering and Engineering Management*, 2017-December, 1009–1012. <http://doi.org/10.1109/IEEM.2017.8290044>
- Kanno, Y., 2018b, Robust truss topology optimization via semidefinite programming with complementarity constraints: a difference-of-convex programming approach, *Computational Optimization and Applications*, 71(2), 403–433. <https://doi.org/10.1007/s10589-018-0013-3>
- Kaveh, A., Mahjoubi, S., 2019, Hypotrochoid spiral optimization approach for sizing and layout optimization of truss structures with multiple frequency constraints, *Engineering with Computers*, 35(4), 1443–1462. <http://doi.org/10.1007/s00366-018-0675-6>
- Kaveh, A., Seddighian, M. R., 2020, Simultaneously multi-material layout, and connectivity optimization of truss structures via an Enriched Firefly Algorithm, *Structures*, 27(June), 2217–2231. <https://doi.org/10.1016/j.istruc.2020.07.029>
- Kaveh, A., Zolghadr, A., 2017, Cyclical parthenogenesis algorithm for layout optimization of truss structures with frequency constraints, *Engineering Optimization*, 49(8), 1317–1334. <http://doi.org/10.1080/0305215X.2016.1245730>
- Kaveh, A., Zolghadr, A., 2018, Meta-heuristic methods for optimization of truss structures with vibration frequency constraints, *Acta Mechanica*, 229(10), 3971–3992. <https://doi.org/10.1007/s00707-018-2234-z>

**KAYNAKLAR DİZİNİ (devam)**

- Kawamura, H., Ohmori, H., Kito, N., 2002, Truss topology optimization by a modified genetic algorithm, *Structural and Multidisciplinary Optimization*, 23(6), 467–472. <https://doi.org/10.1007/s00158-002-0208-0>
- Kumar, S., Tejani, G. G., Pholdee, N., Bureerat, S., 2020, Improved metaheuristics through migration-based search and an acceptance probability for truss optimization, *Asian Journal of Civil Engineering*, 21(7), 1217–1237. <https://doi.org/10.1007/s42107-020-00271-x>
- Lemonge, A. C. C., Carvalho, J. P. G., Hallak, P. H., Vargas, D. E. C., 2021, Multi-objective truss structural optimization considering natural frequencies of vibration and global stability, *Expert Systems with Applications*, 165, 113777. <https://doi.org/10.1016/j.eswa.2020.113777>
- Li, L., Khandelwal, K., 2017, Topology optimization of geometrically nonlinear trusses with spurious eigenmodes control, *Engineering Structures*, 131, 324–344. <http://doi.org/10.1016/j.engstruct.2016.11.001>
- Lieu, Q. X., 2022, A novel topology framework for simultaneous topology, size and shape optimization of trusses under static, free vibration and transient behavior, *Engineering with Computers*. <https://doi.org/10.1007/s00366-022-01599-5>
- Madah, H., Amir, O., 2017, Truss optimization with buckling considerations using geometrically nonlinear beam modeling, *Computers and Structures*, 192, 233–247. <http://doi.org/10.1016/j.compstruc.2017.07.023>
- Madah, H., Amir, O., 2019, Concurrent structural optimization of buckling-resistant trusses and their initial imperfections, *International Journal of Solids and Structures*, 162, 244–258. <https://doi.org/10.1016/j.ijsolstr.2018.12.007>

**KAYNAKLAR DİZİNİ (devam)**

- Maheri, M. R., Askarian, M., Shojaee, S., 2016, Size and topology optimization of trusses using hybrid genetic-particle swarm algorithms, *Iranian Journal of Science and Technology - Transactions of Civil Engineering*, 40(3), 179–193. <https://doi.org/10.1007/s40996-016-0023-2>
- Mela, K., 2014, Resolving issues with member buckling in truss topology optimization using a mixed variable approach, *Structural and Multidisciplinary Optimization*, 50(6), 1037–1049. <https://doi.org/10.1007/s00158-014-1095-x>
- Milajić, A., Beljaković, D., Solovev, V., Murgul, V., 2016, Multi-objective Truss Optimization Using Different Types of the BB-BC Algorithm, *Procedia Engineering*, 165, 947–953. <http://doi.org/10.1016/j.proeng.2016.11.804>
- Mortazavi, A., 2020a, A new fuzzy strategy for size and topology optimization of truss structures, *Applied Soft Computing Journal*, 93, 106412. <https://doi.org/10.1016/j.asoc.2020.106412>
- Mortazavi, A., 2020b, Size and layout optimization of truss structures with dynamic constraints using the interactive fuzzy search algorithm, *Engineering Optimization*, 53(3), 369-391. <https://doi.org/10.1080/0305215X.2020.1726341>
- Mortazavi, A., Toğan, V., 2016, Simultaneous size, shape, and topology optimization of truss structures using integrated particle swarm optimizer, *Structural and Multidisciplinary Optimization*, 54(4), 715–736. <https://doi.org/10.1007/s00158-016-1449-7>
- Mortazavi, A., Toğan, V., 2017, Sizing and layout design of truss structures under dynamic and static constraints with an integrated particle swarm optimization algorithm, *Applied Soft Computing Journal*, 51, 239–252. <https://doi.org/10.1016/j.asoc.2016.11.032>

**KAYNAKLAR DİZİNİ (devam)**

- Mortazavi, A., Toğan, V., Nuhoğlu, A., 2017, Weight minimization of truss structures with sizing and layout variables using integrated particle swarm optimizer, *Journal of Civil Engineering and Management*, 23(8), 985–1001. <https://doi.org/10.3846/13923730.2017.1348982>
- Ohsaki, M., Katoh, N., 2005, Topology optimization of trusses with stress and local constraints on nodal stability and member intersection, *Structural and Multidisciplinary Optimization*, 29(3), 190–197. <https://doi.org/10.1007/s00158-004-0480-2>
- Ohsaki, M., Hayashi, K., 2017, Force density method for simultaneous optimization of geometry and topology of trusses, *Structural and Multidisciplinary Optimization*, 56(5), 1157–1168. <https://doi.org/10.1007/s00158-017-1710-8>
- OpenSees, 2006, The Open System for Earthquake Engineering Simulation, <https://opensees.berkeley.edu/OpenSees/home/about.php>, erişim tarihi: 25.03.2022.
- Panagant, N., Bureerat, S., 2018, Truss topology, shape and sizing optimization by fully stressed design based on hybrid grey wolf optimization and adaptive differential evolution, *Engineering Optimization*, 50(10), 1645–1661. <https://doi.org/10.1080/0305215X.2017.1417400>
- Reintjes, C., Lorenz, U., 2020, Bridging mixed integer linear programming for truss topology optimization and additive manufacturing, *Optimization and Engineering*, 22, 849-893. <https://doi.org/10.1007/s11081-020-09541-8>
- Savsani, V. J., Tejani, G. G., Patel, V. K., 2016, Truss topology optimization with static and dynamic constraints using modified subpopulation teaching–learning-based optimization, *Engineering Optimization*, 48(11), 1990–2006. <https://doi.org/10.1080/0305215X.2016.1150468>

**KAYNAKLAR DİZİNİ (devam)**

- Savsani, V. J., Tejani, G. G., Patel, V. K., Savsani, P., 2017, Modified meta-heuristics using random mutation for truss topology optimization with static and dynamic constraints, *Journal of Computational Design and Engineering*, 4(2), 106–130. <https://doi.org/10.1016/j.jcde.2016.10.002>
- Shahabsafa, M., Fakhimi, R., Lei, W., He, S., Martins, J. R. R. A., Terlaky, T., Zuluaga, L. F., 2020, Truss topology design and sizing optimization with guaranteed kinematic stability, *Structural and Multidisciplinary Optimization*, 63(1), 21-38. <https://doi.org/10.1007/s00158-020-02698-x>
- Sinha, G., R., 2019, Introduction and background to optimization theory, e-kitap, 1-18, <https://iopscience.iop.org/book/978-0-7503-2404-5/chapter/bk978-0-7503-2404-5ch1#bk978-0-7503-2404-5ch1fn1>, erişim tarihi: 20.03.2022.
- Techasen, T., Wansasueb, K., Panagant, N., Pholdee, N., Bureerat, S., 2019, Simultaneous topology, shape, and size optimization of trusses, taking account of uncertainties using multi-objective evolutionary algorithms, *Engineering with Computers*, 35(2), 721–740. <http://doi.org/10.1007/s00366-018-0629-z>
- Tejani, G. G., Savsani, V. J., Bureerat, S., Patel, V. K., Savsani, P., 2019, Topology optimization of truss subjected to static and dynamic constraints by integrating simulated annealing into passing vehicle search algorithms, *Engineering with Computers*, 35(2), 499–517. <http://doi.org/10.1007/s00366-018-0612-8>
- Tejani, G. G., Savsani, V. J., Patel, V. K., Savsani, P. V., 2018, Size, shape, and topology optimization of planar and space trusses using mutation-based improved metaheuristics, *Journal of Computational Design and Engineering*, 5(2), 198–214. <https://doi.org/10.1016/j.jcde.2017.10.001>

**KAYNAKLAR DİZİNİ (devam)**

- Topping, B. H. V, 1983, Shape Optimization of Skeletal Structures: A Review, *Journal of Structural Engineering*, 109(8), 1933-1951. [https://doi.org/10.1061/\(ASCE\)0733-9445\(1983\)109:8\(1933\)](https://doi.org/10.1061/(ASCE)0733-9445(1983)109:8(1933))
- Venkata Rao, R., 2016, Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems, *International Journal of Industrial Engineering Computations*, 7(1), 19–34. <https://doi.org/10.5267/j.ijiec.2015.8.004>
- Wang, H., Ohmori, H., 2010, Truss optimization using genetic algorithm, considering construction process, *International Journal of Space Structures*, 25(4), 205–215. <https://doi.org/10.1260/0266-3511.25.4.205>
- Weldeyesus, A. G., Gondzio, J., He, L., Gilbert, M., Shepherd, P., Tyas, A., 2019, Adaptive solution of truss layout optimization problems with global stability constraints, *Structural and Multidisciplinary Optimization*, 60(5), 2093–2111. <https://doi.org/10.1007/s00158-019-02312-9>
- Weldeyesus, A. G., Gondzio, J., He, L., Gilbert, M., Shepherd, P., Tyas, A., 2020, Truss geometry and topology optimization with global stability constraints, *Structural and Multidisciplinary Optimization*, 62(4), 1721–1737. <https://doi.org/10.1007/s00158-020-02634-z>
- Xiao, Z., Yang, Y., Xiao, R., Bai, Y., Song, C., Wang, D., 2018, Evaluation of topology-optimized lattice structures manufactured via selective laser melting, *Materials and Design*, 143, 27–37. <https://doi.org/10.1016/j.matdes.2018.01.023>
- Yancang, L., Zhen, Y., 2019, Application of Improved Bat Algorithm in Truss Optimization, *KSCE Journal of Civil Engineering*, 23(6), 2636–2643. <https://doi.org/10.1007/s12205-019-2119-2>

**KAYNAKLAR DİZİNİ (devam)**

Zhou, P., Du, J., Lü, Z., 2017, Interval analysis based robust truss optimization with continuous and discrete variables using mix-coded genetic algorithm, *Structural and Multidisciplinary Optimization*, 56(2), 353–370. <https://doi.org/10.1007/s00158-017-1668-6>

