



**BETONARME BİNALARIN
TASARIMINDA GÖRÜNTÜ İŞLEME
VE OPTİMİZASYON TABANLI
OTOMASYON YAKLAŞIMI**

Doktora Tezi

Talha ORHAN

Eskişehir 2025

**BETONARME BİNALARIN TASARIMINDA GÖRÜNTÜ İŞLEME VE
OPTİMİZASYON TABANLI OTOMASYON YAKLAŞIMI**

Talha ORHAN

DOKTORA TEZİ

İnşaat Mühendisliği Anabilim Dalı

Yapı Bilim Dalı

Danışman: Dr. Öğr. Üyesi Onur TUNABOYU

Eskişehir

Eskişehir Teknik Üniversitesi

Lisansüstü Eğitim Enstitüsü

Ağustos 2025

JÜRİ VE ENSTİTÜ ONAYI

Talha ORHAN'ın "BETONARME BİNALARIN TASARIMINDA GÖRÜNTÜ İŞLEME VE OPTİMİZASYON TABANLI OTOMASYON YAKLAŞIMI" başlıklı tezi 25/08/2025 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Eskişehir Teknik Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği"nin ilgili maddeleri uyarınca, İnşaat Mühendisliği Anabilim dalında Doktora Tezi olarak kabul edilmiştir.

Jüri Üyeleri

Unvan Adı Soyadı

İmza

Üye

Dr. Öğr. Üyesi Onur TUNABOYU

Üye

Prof. Dr. Özgür AVŞAR

Üye

Prof. Dr. Emre AKIN

Üye

Doç. Dr. Emre ÇİMEN

Üye

Dr. Öğr. Üyesi Uğur ALBAYRAK

Prof. Dr. Harun BÖCÜK

Lisansüstü Eğitim Enstitüsü Müdürü

25/08/2025

DANIřMAN ONAYI

Daniřmanlıęını yrttęm Doktora ęrencisi Talha ORHAN, BETONARME BİNALARIN TASARIMINDA GRNT İřLEME VE OPTİMİZASYON TABANLI OTOMASYON YAKLAřIMI bařlıklı tez alıřmasını tamamlamıřtır. Hazırlamıř olduęu tez tarafımca incelenmiř ve ęrencinin tez savunma sınavına alınması bilimsel ve etik aıdan uygun grlmřtir.

Tez Daniřmanı

Dr. ęr. yesi Onur TUNABOYU

ÖZET

BETONARME BİNALARIN TASARIMINDA GÖRÜNTÜ İŞLEME VE OPTİMİZASYON TABANLI OTOMASYON YAKLAŞIMI

Talha ORHAN

İnşaat Mühendisliği Anabilim Dalı

Yapı Bilim Dalı

Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Ağustos 2025

Danışman: Dr. Öğr. Üyesi Onur TUNABOYU

Bu çalışmada, mimari planlardan yola çıkılarak taşıyıcı sistem tasarımının görüntü işleme ve sezgisel optimizasyon yöntemleri ile otomasyonu amaçlanmıştır. İlk aşamada, dijital mimari planlar görüntü işleme teknikleri kullanılarak analiz edilmiş; yapısal aks sistemi ve potansiyel kolon yerleşim noktaları nesnel ve tekrarlanabilir bir biçimde belirlenmiştir. Bu veriler temel alınarak oluşturulan aks sistemi üzerinde kolonların yerleştirilmesi ile kolon ve kiriş kesitlerinin seçimi, yapı maliyeti ve yönetmelik gereklilikleri dikkate alınarak karma tamsayılı bir optimizasyon problemi olarak formüle edilmiştir. Çözüm uzayı büyük oranda ayırık yapıda olup yalnızca küçük bir bölümü geçerli çözümlerden oluştuğu için, klasik türev tabanlı yöntemler yerine sezgisel algoritmalar kullanılmıştır. Bu kapsamda farklı optimizasyon algoritmaları kullanılarak sistem test edilmiş, performansları karşılaştırılmış ve en iyi sonuçları veren algoritma belirlenmiştir. Her çözüm adayının değerlendirilmesi; algoritmik modelleme, sayısal analiz ve özel amaç fonksiyonu hesaplamalarını içermektedir. Bu işlemler çözüm adayları arasında bağımsız olarak yürütülebildiğinden, süreç algoritmanın kendisi paralel olarak çalışmasa da her adayın değerlendirilmesinde yüksek paralellik potansiyeli barındırmaktadır. Bu bağlamda, analizlerin çok çekirdekli işlemcilerde eşzamanlı yürütülmesini sağlayan paralel hesaplama yapısı geliştirilmiştir. Taşıyıcı sistem analizleri, yüksek hız avantajı nedeniyle OpenSEES platformunda gerçekleştirilmiş; değerlendirme ve ceza fonksiyonları ise MATLAB ortamında tanımlanmıştır. Geliştirilen sistem, farklı algoritmaların karşılaştırılması sonucunda optimum taşıyıcı sistem tasarımının elde edilmesine olanak sağlayan, kullanıcıdan minimum veri girişiyle çalışan uçtan uca bir otomasyon yaklaşımı sunmaktadır.

Anahtar Sözcükler: Betonarme bina tasarımı, Görüntü işleme, Topoloji optimizasyonu, Kesit optimizasyonu, Otomasyon.

ABSTRACT

IMAGE PROCESSING AND OPTIMIZATION BASED AUTOMATION APPROACH IN THE DESIGN OF REINFORCED CONCRETE BUILDINGS

Talha ORHAN

Department of Civil Engineering

Programme in Structure

Eskişehir Technical University, Institute of Graduate Programs, August 2025

Supervisor: Asst. Prof. Dr. Onur TUNABOYU

This study aims to automate the structural system design process from architectural plans by integrating image processing and heuristic optimization methods. In the initial stage, digital architectural plans are analyzed using image processing techniques, enabling the structural axis system and potential column locations to be identified in an objective and repeatable manner. Based on this data, the placement of columns along the generated axis system and the selection of column and beam cross-sections are formulated as an optimization problem, considering both code requirements and cost criteria. Since the solution space is largely discrete and only a small subset corresponds to valid solutions, heuristic algorithms are employed instead of classical derivative-based methods. Within this scope, different optimization algorithms were tested, their performances compared, and the algorithm providing the best results was identified. The evaluation of each candidate solution involves algorithmic modeling, numerical analysis, and the computation of a customized objective function. As these operations can be conducted independently across solutions, the process carries high parallelism potential at the evaluation level, even though the optimization algorithm itself does not operate in parallel. Accordingly, a parallel computing framework was developed to enable simultaneous evaluations on multi-core processors. Structural analyses were performed using the OpenSEES platform due to its high computational speed, while evaluation procedures and penalty functions were implemented within the MATLAB environment. The developed system provides an end-to-end automation approach that, through the comparison of different algorithms, enables the attainment of an optimum structural system design with minimal user input.

Keywords: Reinforced concrete building design, Image processing, Topology optimization, Section optimization, Automation.

TEŞEKKÜR

Tezimin her aşamasında bilgi ve tecrübeleriyle bana yol gösteren, kimi zaman da bana rağmen büyük bir sabır ve anlayışla yanımda duran; sadece akademik katkılarıyla değil, motivasyon verici tutumuyla da sürece yön veren değerli danışman hocam Dr. Öğr. Üyesi Onur TUNABOYU'na çok teşekkür ederim.

Tezimin değerlendirme aşamasında yapıcı eleştirileri ve yol gösterici önerileriyle çalışmama önemli katkılarda bulunan jüri üyelerim Prof. Dr. Özgür AVŞAR ve Doç. Dr. Emre ÇİMEN'e teşekkürü bir borç bilirim.

Doktora sürecim boyunca manevi desteği ve muhabbetiyle yanımda olan, doktorayı bitirebilme ihtimalime benden daha fazla inanarak motivasyonumu yüksek tutan kıymetli hocam Prof. Dr. Aynur ŞENSOY'a şükranlarımı sunarım.

Hayatım boyunca benden uzakta olsalar da, her zor zamanımda elimden tutacaklarına güvenebildiğim ve hiçbir zaman desteklerini esirgemeyen kıymetli annem ve ablama teşekkür ederim.

Savunma sunumum için hazırladıkları, benim de taşımakta zorlandığım envaiçeşit yiyeceklerle salonu adeta donatan; bununla da kalmayıp her daim manevi destekleriyle yanımda olmaya çalışan eşimin annesi ve anneannesine gönülden teşekkür ederim.

Bu uzun ve yorucu süreçte kimi zaman bir kahve, kimi zaman bir gülümseme, kimi zaman da 'hadi artık çalış' kamçılarıyla; kendi tezine koşturmasına rağmen bizzat vakit ayırıp bana yardım ederek sabrı, sevgisi ve desteğiyle her zaman yanımda olan pek kıymetli yeni ailem, biricik eşime kalpten minnettarım.

Talha ORHAN

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan “bilimsel intihal tespit programı”yla tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Talha ORHAN

İÇİNDEKİLER

	<u>Sayfa</u>
BAŞLIK SAYFASI	I
JÜRİ VE ENSTİTÜ ONAYI	II
DANIŞMAN ONAYI	III
ÖZET	IV
ABSTRACT.....	V
TEŞEKKÜR	VI
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ	VII
İÇİNDEKİLER	VIII
TABLolar DİZİNİ	XI
ŞEKİLLER DİZİNİ	XII
1. GİRİŞ	1
1.1. Amaç ve Kapsam	2
2. LİTERATÜR TARAMASI.....	4
2.1. Görüntü İşleme	4
2.2. Optimizasyon	8
2.3. Otomasyon.....	16
3. YÖNTEM.....	20
3.1. Genel Yaklaşım.....	20
3.2. Model Girdileri	23
3.2.1. Kullanıcı girdileri.....	23
3.2.1.1. Mimari plan	24
3.2.1.2. Yapı geometrisi	24
3.2.1.3. Kesit kütüphanesi	25
3.2.1.4. Malzeme özellikleri.....	26
3.2.1.5. Yükleme tanımları	27
3.2.1.6. Optimizasyon girdileri	28

3.2.1.7. Paralel işleme girdileri	28
3.2.2. Sistem girdileri	29
3.3. Kısıtlar ve Kabuller	30
3.3.1. Görüntü işleme tabanlı kabuller.....	30
3.3.2. Yapı modeli ve tasarım kısıtlamaları.....	31
3.3.3. Sonlu elemanlar modeline ilişkin teknik kabuller	31
3.3.4. Deprem yükü hesabına ilişkin kabuller	32
3.4. Görüntü İşleme Modülü.....	33
3.4.1. Görüntünün ikili formata dönüştürülmesi.....	35
3.4.2. Ölçek oranının belirlenmesi	37
3.4.3. Duvar çizgilerinin tespiti	38
3.4.4. Duvarların tespiti	42
3.4.5. Aksların belirlenmesi	44
3.4.6. Olası kolon yerlerinin belirlenmesi	47
3.4.7. Kiriş sisteminin belirlenmesi.....	50
3.4.8. Parametrik çıktıların düzenlenmesi	52
3.4.9. Görsel çıktıların gösterimi.....	54
3.5. Optimizasyon ile Yapı Tasarımı Modülü	55
3.5.1. Optimizasyon kurgusu.....	56
3.5.1.1. <i>Değişken vektörü yapısı</i>	57
3.5.1.2. <i>Amaç fonksiyonu kurgusu</i>	59
3.5.2. Kullanılan algoritmalar	61
3.5.2.1. <i>Adaptif parçacık sürü optimizasyonu algoritması (APSO)</i>	62
3.5.2.2. <i>Diferansiyel evrim algoritması (DE)</i>	63
3.5.2.3. <i>Tavlama benzetimi algoritması (SA)</i>	63
3.5.2.4. <i>JAYA algoritması</i>	64
3.5.2.5. <i>İyileştirilmiş aritmetik optimizasyon algoritması (IAOA)</i>	64
3.5.3. Modül tasarımı	66
3.5.3.1. <i>Optimizeasyona hazırlık</i>	67
3.5.3.2. <i>Optimizasyon döngüsü</i>	68
3.6. Paralel İşleme.....	86
3.7. Çıktıların İşlenmesi ve Görselleştirilmesi.....	87
4. ANALİZ SONUÇLARI VE TARTIŞMA.....	88

4.1. Görüntü İşleme	88
4.2. Optimizasyon	95
4.2.1. Paralel işleme.....	96
4.2.2. Hiper parametre optimizasyonu	97
4.2.3. Algoritma karşılaştırması ve seçimi	99
4.3. Geliştirilen Yöntemlerin Tutarlılığı ve Doğrulama Süreci.....	108
4.4. Sonuçların Kaydedilmesi	113
4.5. Görsel Çıktılar	117
4.6. Örnek Uygulamalar.....	121
5. SONUÇLAR VE ÖNERİLER	122
KAYNAKÇA	126
ÖZGEÇMİŞ	

TABLÖLER DİZİNİ

	<u>Sayfa</u>
Tablo 3.1. Beton için malzeme özellikleri tablosu.....	26
Tablo 3.2. Donatı çeliği için malzeme özellikleri tablosu	26
Tablo 3.3. Yazılımda kullanılan yük kombinasyonları	29
Tablo 4.1. Plan 1 için duvar çizgisi uzunlukları (ref: 100 cm)	90
Tablo 4.2. Plan 1 için duvar çizgisi uzunlukları (ref: 500 cm)	90
Tablo 4.3. Plan 2 için duvar çizgisi uzunlukları (ref: 100 cm)	91
Tablo 4.4. Plan 2 için duvar çizgisi uzunlukları (ref: 500 cm)	91
Tablo 4.5. Minimum ve maksimum farklar	92
Tablo 4.6. Algoritmaların hiper parametre sonuçları.....	99
Tablo 4.7. Algoritma test sonuçları.....	100
Tablo 4.8. Optimizasyon sonuçları (APSO)	102
Tablo 4.9. Optimizasyon sonuçları (DE)	103
Tablo 4.10. Optimizasyon sonuçları (IAOA).....	103
Tablo 4.11. Örnek bir kolona ait iç kuvvetlerin karşılaştırılması	109
Tablo 4.12. Bir düğüm noktası için deplasman değerleri	110
Tablo 4.13. Yapı periyotları ve modal kütle katılım oranlarına ait sonuçlar	110
Tablo 4.14. Bir kirişin donatı alanı sonuçları.....	111
Tablo 4.15. Optimizasyon sonucunda belirlenen kolon kesitleri	116
Tablo 4.16. Optimizasyon sonucunda belirlenen kiriş kesitleri	116
Tablo 4.17. Optimizasyonla elde edilen yapı modeline ait analiz sonuçları.....	117

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1.	Gri seviye görüntü ve piksel değerleri	5
Şekil 2.2.	a) Kesit optimizasyonu b) Şekil optimizasyonu c) Topoloji optimizasyonu (Gunwant and Misra, 2012).....	9
Şekil 2.3.	ACO optimizasyonu sonuçları a) Gürültü temizleme tekniği kullanılmadan ve b) Gürültü temizleme tekniği kullanılarak	11
Şekil 2.4.	Bütünleşik yapı için optimize edilmiş analiz (Qiao et al., 2016).....	12
Şekil 2.5.	Optimizasyon algoritmalarının türleri (Zakian and Kaveh, 2023).....	14
Şekil 2.6.	Seri ve paralel bağlı bilgisayarlardaki optimizasyon süreleri (Tsoulos et al., 2016)	16
Şekil 3.1.	Geliştirilmiş yazılımın genel akış diyagramı	22
Şekil 3.2.	Kesit kütüphanesine ilişkin hazırlanmış Excel dosyasında örnek bir kesit havuzu girdisi.....	25
Şekil 3.3.	Kolon ve kirişlerdeki lokal eksen yönü tanımlarının global eksenle gösterimi	32
Şekil 3.4.	a) Kullanıcı tarafından sağlanan örnek mimari plan, b) Görüntü işleme sonucunda elde edilen akslar ve olası kolon yerleri.....	34
Şekil 3.5.	Görüntü işleme modülü akış diyagramı	34
Şekil 3.6.	a) Orjinal görüntü-1, b) Gri seviye görüntü-1, c) İkili format görüntü-1, d) Orjinal görüntü-2, e) Gri seviye görüntü-2, f) İkili format görüntü-2....	36
Şekil 3.7.	Örnek bir çizginin, a) Düşük çözünürlüklü monitörden alınan, b) Yüksek çözünürlüklü	38
Şekil 3.8.	Düşey (üst sıra) ve yatay (alt sıra) çizgilere ait olası komşuluk varyasyonları.....	40
Şekil 3.9.	İkili formata çevrilmiş görüntünün, a) İlk hali, b) Çentik düzeltmesi sonrası	41
Şekil 3.10.	Köşe bölgelerinin belirlenmesi ve uzunluk hesaplamasında kullanılan orta nokta ölçüm stratejisi.....	42
Şekil 3.11.	Duvarların tespitinde komşuluk matrisi kullanılarak yapılan iyileştirmenin şematik gösterimi.....	43
Şekil 3.12.	Aks çizgilerinin belirlenmesi	44

Şekil 3.13. Aks çizgilerinin oluşturulmasında karşılaşılan sorunlu (kırmızı) ve doğru (yeşil) bölgelerin şematik gösterimi.....	45
Şekil 3.14. Birbirine yakın aksların tek bir aks olarak düzeltilmesi	46
Şekil 3.15. Sistematik olarak plandaki aksların belirlenmesi	46
Şekil 3.16. İlk oluşturulan kolonların akslarla beraber plan üzerinde gösterimi	47
Şekil 3.17. Ek kolon algoritmasının aşamaları.....	48
Şekil 3.18. Birleşen kolon çiftlerinin gösterimi	49
Şekil 3.19. Olası kolon yerleri tespiti sonucu	50
Şekil 3.20. a) Hatalı giriş sistemiği, b) Kullanılan giriş sistemiği.....	51
Şekil 3.21. Eksenlerin gösterimi a) Piksel cinsinden (px), b) Santimetre cinsinden (cm).....	53
Şekil 3.22. Görüntü işleme süreci a) gri seviye görüntü, b) ikili format görüntü, c) aksların oluşumu, d) olası kolon lokasyonlarının belirlenmesi	55
Şekil 3.23. Optimizasyon ile yapı tasarımı modülü akış diyagramı	66
Şekil 3.24. Maksimum maliyet tasarım (a) ile optimum tasarım (b) modellerin görsel kıyası	68
Şekil 3.25. Amaç fonksiyonu modülü akış diyagramı	69
Şekil 3.26. Birleşen kolonların yapısal analizde modellenmesi.....	70
Şekil 3.27. Örnek bir aday çözümde birleşen kolon çiftleri arası girişlerde eleme işlemi.....	71
Şekil 3.28. SAP2000 (a) ve OpenSEES'te (b) yapı modelleme sürecine ait kullanıcı arayüzü örnekleri	72
Şekil 3.29. Örnek bir yapıda kolon, giriş ve düğüm noktası isimlendirme sistemiği	73
Şekil 3.30. Eleman tanımı için gerekli bilgilerin hesaplanmasında kullanılan formüller	74
Şekil 3.31. Örnek bir modelin OpenSEES modal analiz çıktısında kütle katılım oranları	76
Şekil 3.32. Örnek bir modelin OpenSEES modal analiz çıktısında periyot ve frekans bilgileri.....	76
Şekil 3.33. Yapı analizinde OpenSEES ile MATLAB arasındaki veri alışverişi diyagramı	77

Şekil 3.34. Kiriş boyuna donatı tasarımı için pozitif yön kabulleri	78
Şekil 3.35. Kiriş etriye tasarımı için pozitif yön kabulleri	81
Şekil 3.36. Donatı yerleşim türünün gösterimi	84
Şekil 4.1. Görüntü işleme uygulamasında kullanılan iki farklı kat planı: a) Plan 1, b) Plan 2	89
Şekil 4.2. Plan 1 için görüntü işleme sürecine ait a) kullanıcı girdisi, b) ikili format, c) aks çizimi ve d) kolon tespiti çıktıları.....	93
Şekil 4.3. Plan 2 için görüntü işleme sürecine ait a) kullanıcı girdisi, b) ikili format, c) aks çizimi ve d) kolon tespiti çıktıları.....	94
Şekil 4.4. Paralel işleme için karşılaştırmalı performans analizi sonuçları.....	96
Şekil 4.5. Algoritmaların Bayes optimizasyonu yakınsama eğilimleri.....	98
Şekil 4.6. Değişken vektörü kurgularının APSO algoritması özelinde kıyaslanması	100
Şekil 4.7. Tüm konfigürasyonlardaki minimum maliyet sonuçları.....	105
Şekil 4.8. C25-K5-D1 konfigürasyonunda algoritmaların yakınsama eğilimleri	106
Şekil 4.9. C25-K9-D2 konfigürasyonunda algoritmaların yakınsama eğilimleri	107
Şekil 4.10. IAOA sonucu C25-K5-D1 için SAP2000 modeli.....	109
Şekil 4.11. OpenSEES ve SAP2000 programlarındaki pozitif yön kabulleri	110
Şekil 4.12. STA4CAD analizi sonucu yetersiz eleman listesi	112
Şekil 4.13. STA4CAD programında S109 kolonuna ait eksenel kuvvet değeri	113
Şekil 4.14. Optimizasyon sonucu kaydedilen tüm sonuçların MATLAB ortamında görüntülenmesi.....	115
Şekil 4.15. C25-K5-D1 konfigürasyonuna ait en iyi yapı modeli sonuçları, a) IAOA, b)APSO, c)DE.....	118
Şekil 4.16. IAOA ile gerçekleştirilmiş tüm parametrik kombinasyonların görsel çıktıları	119
Şekil 4.17. IAOA-C25-K5-D1 konfigürasyonu için en iyi yapı modeli sonuçları a) Plan1, b)Plan2	121

1. GİRİŞ

Yapı mühendisliğinde taşıyıcı sistemlerin tasarımı, yapının güvenliği, performansı, maliyeti ve yönetmeliklere uygunluğu açısından kritik öneme sahiptir. Bu tasarım süreci, yapı elemanlarının (kolon, kiriş vb.) konumlarının belirlenmesini, uygun kesitlerin seçilmesini ve yapısal sistemin genel düzeninin oluşturulmasını içeren çok aşamalı ve çok değişkenli bir karar problemidir. Geleneksel yöntemlerle yürütülen bu süreç, çoğunlukla mühendislik sezgisine, deneyime ve kullanıcı müdahalesi gerektiren CAD tabanlı analiz uygulamalarına dayanmakta; bu da süreci hem zaman alıcı hem de kişiye bağımlı hale getirmektedir. Ayrıca, özellikle büyük ve karmaşık yapılarda, tüm olası kolon yerleşim kombinasyonlarının ve kesit alternatiflerinin değerlendirilmesi pratik olarak mümkün olmadığından, genellikle optimal olmayan çözümlerle yetinilmektedir.

Öte yandan, son yıllarda mühendislik disiplinlerinde dijitalleşme ve otomasyon eğilimi belirgin biçimde artmış; görüntü işleme, yapay zeka ve optimizasyon tekniklerinin yapı mühendisliği tasarım süreçlerine entegre edilmesi yönünde çok sayıda çalışma gerçekleştirilmiştir. Bu gelişmeler doğrultusunda, taşıyıcı sistem tasarımının daha sistematik, tekrarlanabilir ve objektif biçimde yürütülebilmesi için geleneksel yaklaşımların yerini dijital modelleme ve algoritmik karar verme süreçlerine bırakması kaçınılmaz hale gelmiştir.

Bu bağlamda, bu tez çalışmasında taşıyıcı sistem tasarımının ilk adımı olan kolon yerleşiminin belirlenmesi, geleneksel yöntemlerden farklı olarak mimari planların görüntü işleme teknikleriyle otomatik olarak analiz edilmesiyle gerçekleştirilmiştir. Literatürdeki çoğu çalışma, kolon yerleşimlerini ya doğrudan kullanıcı tarafından belirlenmiş sabit noktalar üzerinden ele almakta ya da varsayımsal bir ızgara sistemi üzerinden çözümlemektedir. Bu çalışmada ise, mimari planlarda yer alan duvar çizgileri doğrudan dijital görüntüden tespit edilerek, gerçek tasarım kısıtlarını yansıtan bir kolon yerleşim uzayı elde edilmiştir. Böylece taşıyıcı sistem tasarımı, mimari ile uyumlu ve uygulamaya daha yakın bir zeminde başlatılmıştır.

Ayrıca, literatürdeki çalışmalardan farklı olarak kolonların hem konumsal yerleşimi (topolojisi) hem de kesit özellikleri aynı anda optimize edilerek, taşıyıcı sistemin genel yapısal etkinliği en üst düzeye çıkarılmaktadır. Bu yaklaşım, sadece kesit optimizasyonu yapan çalışmalardan farklı olarak, kolonların plan içindeki varlığı/yokluğu ve yerleri

üzerinde de karar vererek çözüm uzayını genişletmekte ve mühendislik açısından daha bütüncül bir tasarım olanağı sunmaktadır.

Çalışmanın en önemli özgün katkılarından biri, mimari plandan başlayan ve taşıyıcı sistemin nihai tasarımına kadar uzanan sürecin tamamen otomatik bir şekilde bütünleştirilmiş olmasıdır. Görüntü işleme ile elde edilen mimari veri, yapısal analiz ve yönetmelik kontrollerini içeren bir optimizasyon algoritmasına doğrudan entegre edilmiştir. Böylece kullanıcı müdahalesi olmadan, mimari plan üzerinden taşıyıcı sistemin mühendislik açısından geçerli ve optimize edilmiş hali elde edilebilmiştir.

1.1. Amaç ve Kapsam

Yapı tasarım problemleri, çoğunlukla ayırık (kesikli) değişkenlerden oluşan, yüksek boyutlu ve geçerli çözümlerin çözüm uzayında seyrek dağıldığı zorlu optimizasyon problemleri olarak karakterize edilir. Bu tür problemlerde, klasik türev tabanlı optimizasyon yöntemleri çoğu durumda yetersiz kalmaktadır; çünkü hem çözüm uzayı süreksizdir hem de amaç fonksiyonunun analitik olarak türevi bulunmamaktadır. Dahası, bu yöntemler çok sayıda yerel minimum içeren çözüm alanlarında global optimumdan uzak sonuçlar üretme eğilimindedir. Bu nedenle, sezgisel algoritmalar, yapı mühendisliğinde karmaşık optimizasyon problemleri için daha uygun bir çözüm yöntemi sunmaktadır.

Bu çalışma, görüntü işleme ve sezgisel optimizasyon tekniklerini birleştiren bütünleşik bir otomasyon yaklaşımı önererek, yapı mühendisliği tasarım süreçlerine katkı sağlamayı amaçlamaktadır. Geliştirilen sistem, mimari planlardan yola çıkarak aks sisteminin, kolon yerleşimlerinin ve kesit tiplerinin otomatik olarak belirlenmesini; ardından bu sistemin sayısal analizinin yapılarak performans ve maliyet açısından değerlendirilmesini kapsamaktadır. Süreç boyunca her çözüm adayının algoritmik olarak modellenmesi, sayısal olarak analiz edilmesi ve değerlendirme kriterlerine göre uygunluk fonksiyonunun hesaplanması sağlanmıştır. Bu işlemler çözüm adayları arasında birbirinden bağımsız şekilde yürütülebildiği için, sistem yüksek derecede paralel işlemeye elverişli bir yapıda kurgulanmıştır.

Yapı analizi için OpenSEES yazılımı tercih edilmiş; hesaplama verimliliği açısından analiz süreci çok çekirdekli işlemcilerde eşzamanlı olarak yürütülmüştür. Değerlendirme ve ceza fonksiyonları ise MATLAB ortamında kullanıcı tanımlı kurallar çerçevesinde gerçekleştirilmiştir. Bu sayede, optimizasyon sürecinde hem yapısal

geçerlilik hem de hesaplama performansı birlikte gözetilmiştir. Geliştirilen sistem, kullanıcıdan yalnızca mimari plan girişi gerektirmesi bakımından tasarım sürecinde minimum insan müdahalesi ile çalışan, uçtan uca bir otomasyon yapısı sunmaktadır.

Bu tez beş bölümden oluşmaktadır. Birinci bölümde, çalışmanın dayandığı kuramsal çerçeve sunulmuş; yapı mühendisliğinde taşıyıcı sistem tasarımının önemi, geleneksel yöntemlerin sınırlılıkları ve dijital yaklaşımların gerekliliği tartışılmıştır. Ayrıca çalışmanın amacı, kapsamı ve genel yaklaşımı açıklanmıştır. İkinci bölümde, inşaat mühendisliğinde görüntü işleme, optimizasyon ve otomasyon ile ilgili literatürde yer alan çalışmalar incelenmiş ve özetlenmiştir. Üçüncü bölümde, önerilen yöntemle ilişkin ayrıntılı bilgiler verilmiştir. Bu kapsamda, mimari planlardan elde edilen verilerin işlenmesi, taşıyıcı sistemin sonlu elemanlar modelinin oluşturulması, kullanılan optimizasyon algoritmalarının yapısı, amaç fonksiyonu kurgusu ve analiz sürecinde kullanılan paralel hesaplama yapısı ayrıntılı olarak açıklanmıştır. Dördüncü bölümde, geliştirilen sistemin farklı yapı senaryoları üzerindeki uygulamaları sunulmuş; elde edilen analiz ve optimizasyon sonuçları değerlendirilerek yöntem performansı yapısal uygunluk, maliyet etkinliği ve hesaplama verimliliği açısından tartışılmıştır. Beşinci ve son bölümde ise, çalışmadan elde edilen genel bulgular özetlenmiş ve gelecekte yapılabilecek iyileştirme ve genişletme çalışmaları için öneriler sunulmuştur.

2. LİTERATÜR TARAMASI

Günümüzde inşaat mühendisliği, geleneksel mühendislik uygulamalarının ötesine geçerek yapay zeka, görüntü işleme, optimizasyon ve otomasyon gibi çok disiplinli yaklaşımların entegrasyonuna yönelmektedir. Özellikle yapı mühendisliği alanında bu teknolojiler; tasarım, analiz, değerlendirme ve bakım süreçlerinde hem doğruluk hem de verimlilik açısından önemli katkılar sağlamaktadır. Yapıların giderek artan karmaşıklığı ve sürdürülebilirliğe yönelik beklentiler, mühendislik problemlerinin çözümünde geleneksel yöntemlerin sınırlarını zorlamaktadır. Bu nedenle, yenilikçi bilişim teknolojilerine olan ilgi gün geçtikçe artmaktadır.

Görüntü işleme yöntemleri sayesinde mimari planlar ve saha verileri sayısal olarak analiz edilerek taşıyıcı sistem elemanları otomatik biçimde tanımlanabilmekte, bu da yapısal modelleme sürecini önemli ölçüde hızlandırmaktadır. Optimizasyon algoritmaları ise yüksek boyutlu ve karmaşık tasarım problemlerinde hem maliyet etkin hem de yönetmeliklere uygun çözümler geliştirilmesine olanak sunmaktadır. Bu süreçlerin otomasyon ile bütünleştirilmesi, yapısal sistemlerin hızlı, sistematik ve hatasız biçimde modellenmesini sağlamaktadır.

Sonuç olarak, görüntü işleme, optimizasyon ve otomasyonun yapı mühendisliğine entegrasyonu, tasarım süreçlerinin daha akılcı, sistematik ve hızlı yürütülmesini sağlamaktadır. Bu entegrasyon yalnızca mühendislik etkinliklerinin verimliliğini artırmakla kalmamakta, aynı zamanda daha güvenli, ekonomik ve sürdürülebilir yapıların tasarımına olanak tanımaktadır.

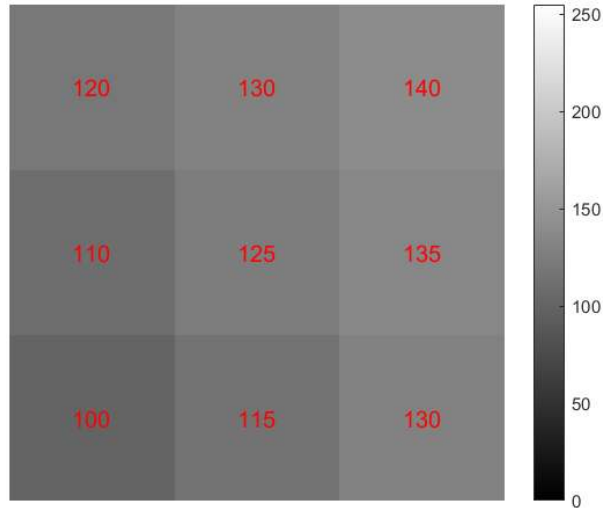
2.1. Görüntü İşleme

Görüntü işleme, dijital görüntülerin çeşitli matematiksel ve algoritmik yöntemler aracılığıyla işlenmesi ve bu sayede anlamlı bilgiye dönüştürülmesini amaçlayan disiplinler arası bir bilim ve mühendislik alanı olarak tanımlanmaktadır (Dastres and Soori, 2021). Bu alan kapsamında, görüntünün elde edilmesi, sayısallaştırılması, iyileştirilmesi, dönüştürülmesi ve analiz edilmesi gibi pek çok işlem bir arada yürütülmektedir (Çankaya et al., 2013). Görüntü işleme, bilgisayarlı görü, makine öğrenmesi, yapay zeka, sinyal işleme ve sayısal analiz gibi birçok farklı disiplini bünyesinde barındırmakta olup hem teorik hem de uygulamalı çalışmalarla desteklenmektedir.

Görüntü işleme süreci kapsamında, sayısal görüntüler üzerinde çeşitli filtreleme, öznitelik çıkarımı, kenar belirleme, segmentasyon, nesne tanıma ve sınıflandırma gibi işlemlerin gerçekleştirilmesi mümkün olmaktadır. Bu işlemler aracılığıyla, ham görüntü verisinden yüksek düzeyde bilgi çıkarımı sağlanabilmekte ve elde edilen sonuçlar farklı alanlarda kullanılabilir. Uygulama alanları arasında tıp, endüstri, savunma, güvenlik, tarım, robotik, uzaktan algılama ve yapı mühendisliği gibi çok sayıda disiplin yer almaktadır (Hoang, 2018; Prabakaran et al., 2020; Scholl et al., 2011; Zawad et al., 2021).

Dijital görüntüler, her biri belirli bir konuma ve parlaklık değerine sahip sonlu sayıda öğeden oluşmakta olup, bu öğeler piksel olarak adlandırılmaktadır (Gonzalez and Woods, 2017). Sayısal ortamda bir görüntü, iki boyutlu bir matris biçiminde temsil edilmekte ve her matris elemanı, görüntüdeki bir pikseli ifade etmektedir.

Gri seviye görüntülerde her piksel, genellikle 8 bitlik bir değerle (0–255 aralığında) tanımlanmakta ve bu değer pikselin parlaklık düzeyini göstermektedir (Şekil 2.1). Renkli görüntüler ise RGB (Red-Green-Blue) renk uzayında, her biri bir renk bileşenine karşılık gelen üç kanallı vektörler aracılığıyla ifade edilmektedir (Fitriyah and Wihandika, 2018; Padmavathi and Thangadurai, 2016). Görüntü işleme algoritmalarında, bu matris yapıları üzerinde çeşitli matematiksel ve sayısal işlemler uygulanarak, istenen dönüşümler sağlanmaktadır.



Şekil 2.1. Gri seviye görüntü ve piksel değerleri

İnşaat mühendisliğinde görüntü işleme yöntemleri, yapıların davranışının, hasar durumunun ve performansının tespit edilmesinde yaygın biçimde kullanılmaktadır. Bu

yöntemlerle betonarme ve çelik yapı elemanlarında oluşan deformasyon, çatlak ilerlemesi, yer değiştirme ve titreşim gibi parametrelerin tahribatsız biçimde ölçülmesi sağlanmaktadır. Ayrıca, şantiyelerde kalite kontrol süreçleri, inşaat ilerleme takibi uygulamaları, yapısal sağlık izleme sistemleri ve güvenlik değerlendirmesi gibi alanlarda görüntü işleme tekniklerinden yararlanılmaktadır.

(Mousa et al., 2021) tarafından, köprülerde meydana gelen yapısal davranışların tespiti amacıyla dijital görüntü korelasyonu (DIC) yöntemi kullanılmıştır. Beton, çelik ve asma köprüler üzerinde deformasyon, gerinim, titreşim, sapma ve dönme gibi parametreler incelenmiştir. Bu kapsamda, deney numuneleri kontrollü laboratuvar koşullarında hazırlanmış ve yüksek çözünürlüklü kameralar aracılığıyla görüntü verileri elde edilmiştir. Ayrıca termal görüntüleme yönteminden de yararlanılmıştır. Uygulamanın tahribatsız muayene esasına dayandığı ve köprü izleme süreçlerinde güvenilir bir teknik olarak değerlendirilebileceği ifade edilmiştir.

(Alemdar et al., 2018) tarafından yürütülen bir çalışmada, çelik köprülerde meydana gelen yer değiştirme ve şekil değiştirme davranışlarının incelenmesi amacıyla dijital fotogrametriye dayalı bir görüntü işleme yöntemi uygulanmıştır. Köprü elemanları üzerinde yerleştirilen işaretleyiciler yardımıyla deformasyonun mekansal dağılımı analiz edilmiş, elde edilen veriler SAP2000 yazılımı ile elde edilen hesaplamalarla karşılaştırılmıştır. Görüntü işleme tabanlı ölçüm yönteminin, daha düşük hata payıyla sonuç verdiği ve geleneksel hesaplama yöntemlerine kıyasla daha yüksek doğruluk sunduğu ortaya konmuştur.

Yang vd. (2019) yapı elemanlarındaki çatlak genişliklerinin hızlı ve doğru bir şekilde belirlenmesi amacıyla görüntü işleme temelli bir yöntem geliştirmiştir. Yapılan analizlerde, önerilen yöntemin ölçüm doğruluğu %99 seviyesinde hesaplanmış ve bu yöntemin özellikle risk taşıyan yapıların hasar tespit süreçlerinde etkin şekilde kullanılabileceği sonucuna varılmıştır.

Uygunoğlu ve Çetingül (2024) tarafından gerçekleştirilen başka bir çalışmada ise, betonarme yapılarda karbonatlaşma derinliğinin belirlenmesi amacıyla görüntü işleme tekniklerinden yararlanılmıştır. Karbonatlaşmaya maruz bırakılan beton numuneleri kırılarak iç yüzeylerine fenolftalein çözeltisi uygulanmıştır. Alkali ortamda renksiz kalan fenolftaleinin, karbonatlaşmış bölgelerde pembe-mor renk değişimi göstermesi esas alınarak renk ayrımı yapılmıştır. Elde edilen yüzey görüntüleri dijital ortamda analiz edilmiş ve RGB (kırmızı, yeşil, mavi) renk bileşenlerine dayalı sayısal değerlendirme

yapılmıştır. Bu yöntemle betonun karbonatlaşma potansiyeli görüntü işleme teknikleri ile ortaya konulmuştur.

Kim vd., (2013) çalışmasında, 4D CAD modellerinin sahadaki gerçek inşaat ilerlemesine göre otomatik güncellenmesini sağlayan bir görüntü işleme temelli yöntem geliştirilmiştir. Sahadan alınan görüntülerden yapı elemanlarının montaj durumu belirlenmiş ve bu bilgilerle bir zaman çizelgesi oluşturulmuştur. Çizelge, Microsoft Project ve JetStream yazılımları aracılığıyla 3D modelle entegre edilerek güncel bir 4D CAD modeli elde edilmiştir. %92 doğruluk oranıyla test edilen yöntem, büyük ölçüde otomasyon sağlamış ve 4D CAD uygulamalarında manuel süreci önemli ölçüde azaltmıştır.

McCormick ve Lord, (2012) tarafından gerçekleştirilen çalışmada, yapısal ölçümler için dijital görüntü korelasyonu (DIC) tekniğinin saha uygulamalarındaki potansiyeli incelenmiştir. DIC, bir yapının yükleme öncesi ve sonrası görüntüleri arasındaki iki boyutlu yer değiştirme ve şekil değişimlerini hesaplayarak tam alanlı deformasyon ölçümü yapılmasını sağlamaktadır. Çalışmada, bu teknik kullanılarak çatlak açılma tespiti, köprü sehim ölçümü, boru titreşimleri ve döner yapıların hareketlerinin izlenmesi gibi çeşitli saha uygulamaları sunulmuştur. Yüksek çözünürlüklü kameralar ve gelişmiş hesaplama algoritmaları kullanılarak DIC'nin, betonarme, çelik ve taş gibi farklı malzemelerde yüzey deformasyonlarının hassas biçimde izlenmesine olanak sağladığı gösterilmiştir. Ayrıca DIC sonuçlarının, klasik doğrusal değişimli diferansiyel transformatörler (LVDT) kullanılarak elde edilen ölçümlerle yüksek uyum gösterdiği belirtilmiştir. Çalışma, görüntü işleme temelli deformasyon izleme yöntemlerinin saha koşullarında güvenilir biçimde uygulanabileceğini ortaya koymuştur.

Ferrer vd., (2013) çalışmasında, yapı güvenlik sistemlerinin performansını değerlendirmek amacıyla yüksek hızlı kameralar ve görüntü işleme tekniklerine dayalı temassız bir ölçüm yöntemi geliştirilmiştir. 5 metreden düşürülen 75 kg'lık cismin hareketi analiz edilerek ağ deformasyonu ve ivme hesaplanmış; mevcut standartlardaki deformasyon sınırlarının tehlikeli ivmelere yol açabileceği gösterilmiştir. Çalışma, görüntü işleme yöntemlerinin yapı güvenliği değerlendirmelerinde etkili ve pratik bir alternatif sunduğunu ortaya koymuştur.

Chen ve Chen, (2020) tarafından gerçekleştirilen çalışmada, betonun iç yapısal kusurlarını tespit etmek amacıyla ultrasonik tomografi temelli, temassız bir görüntüleme yöntemi geliştirmiştir. Ultrasonik dalgaların hız dağılımları yapay zeka destekli

algoritmalarla analiz edilerek beton içindeki boşluk ve ayrışmalar başarıyla görselleştirilmiştir. Çalışma, betonarme yapılarda tahribatsız ve güvenilir kalite kontrol için uygulanabilir bir yöntem sunmaktadır.

2.2. Optimizasyon

Günümüzde mühendislik problemleri, artan sistem karmaşıklığı, çok sayıda etkileşimli tasarım parametresi ve çoğunlukla birbiriyle çelişen performans kriterleri nedeniyle geleneksel yöntemlerle çözülmesi güç çok sayıda alternatif çözümün değerlendirilmesini gerektirmektedir. Bu durum, özellikle yapı mühendisliği gibi disiplinlerde, teknik gerekliliklerin yanı sıra ekonomik, güvenlik ve yönetmelik temelli kısıtların da dikkate alınmasını zorunlu kılmaktadır. Ayrıca malzeme ve finansal kaynakların sınırlı olması, mevcut kaynakların en verimli şekilde kullanılmasını gerektirmekte; bu da tasarım sürecini çok amaçlı ve çok kısıtlı bir karar verme problemine dönüştürmektedir.

Bu bağlamda, optimizasyon algoritmaları, söz konusu karmaşık tasarım uzayında en iyi çözümlerin sistematik biçimde aranmasına olanak tanıyan temel araçlardan biri haline gelmiştir. Klasik deneme-yanılma yöntemleri, sınırlı sayıda alternatifi değerlendirmeye dayandığı için yüksek boyutlu çözüm uzaylarında hem zaman açısından verimsiz kalmakta hem de global optimuma ulaşma olasılığı düşmektedir. Oysa modern optimizasyon teknikleri, çok sayıda değişkenin ve kısıtın tanımlı olduğu problemlerde, geniş çözüm uzaylarını daha kısa sürede tarayarak daha etkili ve dengeli çözümler elde edebilme potansiyeline sahiptir. Bu nedenle, mühendislik tasarım süreçlerinde optimizasyon yaklaşımları, yalnızca performans artırımı açısından değil, aynı zamanda yönetmelik uyumluluğu, maliyet etkinliği ve kaynak verimliliği açısından da kritik bir rol üstlenmektedir.

Genel bir optimizasyon problemi, Denklem 2.1’de gösterildiği şekilde tanımlanabilir.

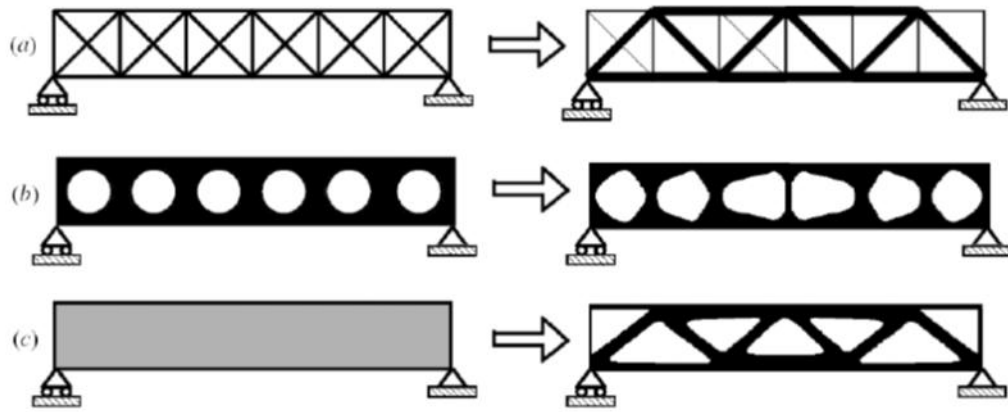
$$\begin{aligned} & \min f(x), & x \in R^n \\ \text{koşuluyla: } & g_i(x) < 0, & i = 1, \dots, q \\ & h_j(x) = 0, & j = 1, \dots, k \end{aligned} \quad (2.1)$$

Bu denklemde,

- x , optimize edilecek tasarım değişkenlerinin bulunduğu değişken vektörünü,
- $f(x)$, minimize edilmesi hedeflenen amaç fonksiyonunu,

- $g(x)$ eşitsizlik kısıtlarını,
- $h(x)$ eşitlik kısıtlarını,
- q eşitsizlik kısıtlarının sayısını,
- k eşitlik kısıtlarının sayısını belirtmektedir.

Yapı mühendisliği alanındaki optimizasyon uygulamaları, kapsamlı bir derleme çalışmasında da belirtildiği üzere (Kashani et al., 2022), genellikle kesit optimizasyonu, topoloji optimizasyonu ve şekil optimizasyonu olmak üzere üç ana kategoride sınıflandırılmaktadır. Bu üç yaklaşımın ortak amacı, genellikle toplam malzeme maliyetini veya yapının ağırlığını minimize etmektir. Şekil 2.2'de, basit bir kiriş elemanı üzerinden bu üç farklı optimizasyon türünün uygulanmasına ilişkin örnek sonuçlar gösterilmektedir (Gunwant and Misra, 2012).



Şekil 2.2. a) Kesit optimizasyonu b) Şekil optimizasyonu c) Topoloji optimizasyonu (Gunwant and Misra, 2012)

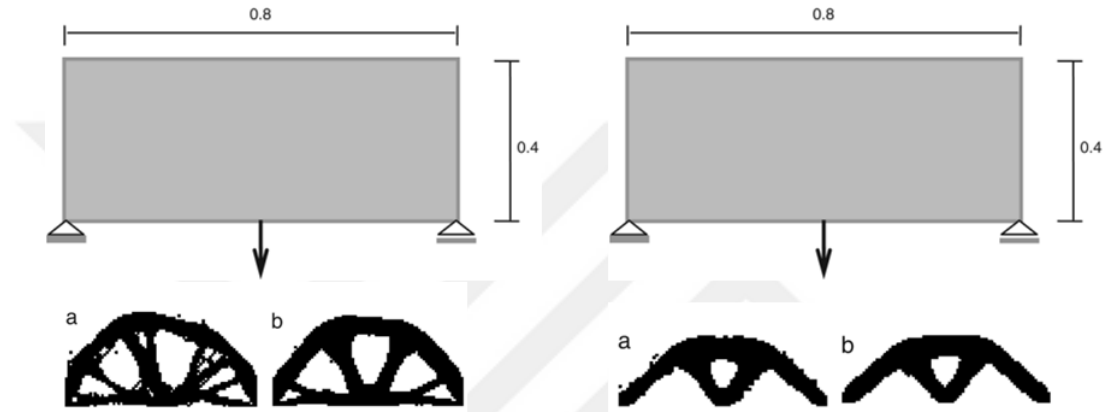
Bu kategoriler, yapının maruz kaldığı dış yükler ve yürürlükteki yönetmelik kısıtları doğrultusunda şu şekilde özetlenebilir:

Kesit optimizasyonu, yapının genel geometrik düzeninin (örneğin kolon ve kirişlerin konumları ve uzunlukları) sabit tutulduğu, ancak yapı elemanlarının kesit özelliklerinin değişken olarak kabul edildiği bir tasarım yaklaşımıdır. Yapı mühendisliği literatüründe bu alana yönelik çeşitli çalışmalar gerçekleştirilmiştir. Chisari ve Bedon, (2016) çerçevesi bir yapının sismik güçlendirme performansını değerlendirmek amacıyla lifli polimer (FRP) sargıların kullanımına odaklanmış ve güçlendirme tasarımını maliyet açısından etkin hale getirmek için, çerçevenin sünekliğini maksimize ederken FRP hacmini minimize eden çok amaçlı bir Genetik Algoritma (GA) yöntemi geliştirmiştir.

Bu yöntem, vaka analizi ile test edilerek pratikte uygulanabilirliği ortaya konmuş, çoklu amaç fonksiyonlarının ve farklı değişkenlerin bir arada kullanımıyla FRP ile güçlendirme uygulamalarında daha verimli ve kolay bir optimizasyon süreci sağlandığı görülmüştür. Benzer şekilde, Niaki vd., (2016) iki boyutlu bir çerçeve sistemin optimizasyonunu, Amerikan Beton Enstitüsü (ACI) yapı standardına uygun olarak genetik algoritma yöntemiyle gerçekleştirmiştir. Beton, çelik, kalıp ve işçilik maliyetlerinin toplamı amaç fonksiyonu olarak tanımlanmış; kısıtlar ise penaltı fonksiyonu yoluyla algoritmaya entegre edilmiştir. Popülasyon seçimi için turnuva seçimi ve kırımlar ile kolonlarda daha iyi yakınsama sağlamak amacıyla onarım operatörü kullanılarak algoritmanın performansı artırılmış, uygulama sonucunda yöntemin hem maliyet açısından etkin hem de hesaplama süresi bakımından verimli olduğu ortaya konulmuştur. Thomas ve Arulraj. G, (2017) ise, tek donatılı betonarme kırımların maliyetini minimize etmek amacıyla ardışık ikinci dereceden programlama (Sequential Quadratic Programming, SQP) algoritması kullanarak doğrusal olmayan kısıtlı bir optimizasyon yaklaşımı geliştirmiş; beton ve çelik sınıfı ile donatı çapının toplam maliyet üzerindeki etkilerini analiz etmişlerdir. Öte yandan, Melih Nigdeli ve Bekdaş, (2019) tarafından gerçekleştirilen bir başka çalışmada, betonarme kırımların maliyet optimizasyonuna ek olarak, döşeme kalınlığının kırımlar tasarımı üzerindeki etkisi incelenmiş ve analizler ACI 318 yönetmeliğine uygun olarak gerçekleştirilmiştir. Optimizasyon işlemi, Harmoni Arama (Harmony Search, HS) algoritması ile yürütülmüş; amaç fonksiyonu olarak ise kırımlar boyutları ve gerekli donatı miktarlarını içeren toplam maliyet esas alınmıştır. Dört farklı döşeme kalınlığı senaryosu üzerinden yapılan karşılaştırmalar, en uygun tasarım üzerindeki etkileri ortaya koymuş ve çalışmanın sonuçları, betonarme kırımlar tasarımı maliyet optimizasyonunun yanı sıra yapısal modellemenin doğru şekilde tanımlanmasının önemini vurgulamıştır.

Topoloji optimizasyonu, belirli bir tanımlı çözüm uzayı içinde hangi elemanların yapıda bulunması gerektiği ve hangilerinin elenmesi gerektiği kararının verildiği yöntemidir. Bu yöntem özellikle uzay kafes sistemleri gibi ayırık elemanlardan oluşan yapılarda sıkça kullanılmaktadır. Bu alanda literatürde çeşitli çalışmalar yer almaktadır. A Kaveh vd., (2008) iki ve üç boyutlu yapıların topoloji optimizasyonunda Karınca Kolonisi Optimizasyonu (ACO) algoritmasının kullanımını değerlendirmiştir. Çalışmanın temel amacı, sınırlı malzeme hacmi altında en rijit yapıyı elde etmektir. Bu doğrultuda, sonlu eleman yöntemi ile hesaplanan gerinim enerjisi değerleri kullanılarak

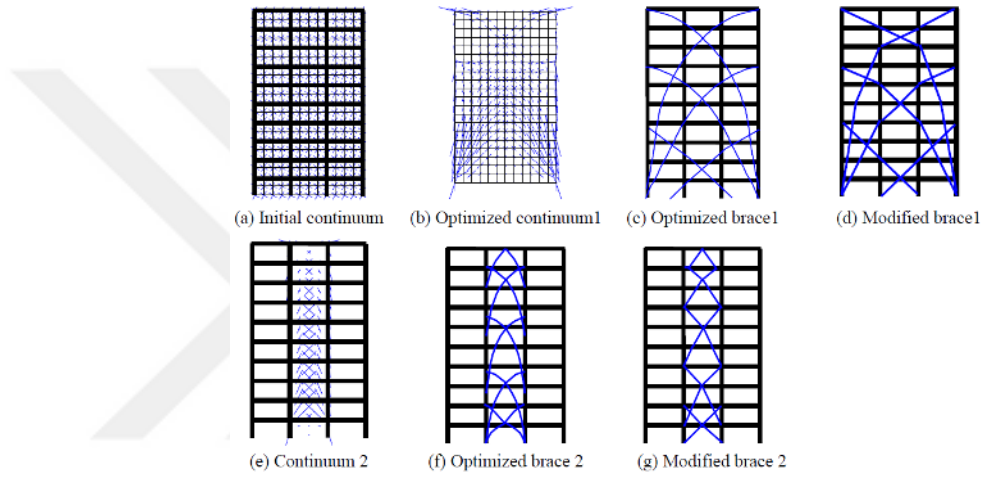
elemanların katkıları değerlendirilmekte ve her elemanın yapıda kalıp kalmayacağı ACO algoritmasıyla belirlenmektedir. Yöntem, ayrık (0/1) karar değişkenleriyle çalıştığından dolayı gri alan oluşumu engellenmekte, bu da mühendislik açısından doğrudan uygulanabilir çözümler elde edilmesini sağlamaktadır. Ayrıca, küçük ve işlevsiz elemanları önlemek amacıyla “gürültü temizleme” tekniği de algoritmaya entegre edilmiştir. İki ve üç boyutlu örneklerde yapılan uygulamalar, yöntemin farklı problem türlerinde etkin şekilde çalışabildiğini göstermiştir (Şekil 2.3).



Şekil 2.3. ACO optimizasyonu sonuçları a) Gürültü temizleme tekniği kullanılmadan ve b) Gürültü temizleme tekniği kullanılarak

Fu vd., (2025) çubuk, kiriş, kabuk veya bunların birleşiminden oluşan karmaşık yapıların topolojisi, malzemesi ve kiriş kesit türlerinin eşzamanlı olarak optimize edilmesine yönelik yeni bir mühendislik yöntemi geliştirmiştir. Bu yöntem, her konumda farklı kesitlere ve malzemelere sahip çoklu bileşenlerin yer aldığı genişletilmiş bir zemin yapısı üzerinde tanımlanmış bir optimizasyon problemine dayanmaktadır. Problem, ikili (0/1) değişkenler ve sürekli boyut değişkenlerini içeren karışık değişkenli yaklaşık problemler dizisi olarak formüle edilmiş; ikili değişkenler için genetik algoritma (GA), sürekli değişkenler için ise ikili yöntem kullanılmıştır. Yöntemin geçerliliği dört farklı yapısal örnekle test edilmiş ve her birinde topoloji, malzeme ve kesit seçimi birlikte optimize edilmiştir. Sonuçlar, önerilen yöntemin hem hesaplama verimliliği hem de yapısal performans açısından geleneksel yaklaşımlara kıyasla üstünlük sağladığını göstermiştir. Benzer şekilde, Qiao vd., (2016) deprem gibi yatay yükler altında yapıların dayanımını artırmak amacıyla çapraz elemanların konumlandırılmasına yönelik bir topoloji optimizasyonu uygulamıştır (Şekil 2.4). Çubuk-benzeri malzeme modeli ve gerilme kısıtlarıyla gerçekleştirilen bu optimizasyonda, yapının elastik davranış

gösterdiği düşük şiddetli deprem koşulları esas alınmıştır. Ancak, gerçekte yapılar çoğunlukla şiddetli depremlerde plastik şekil değiştirme gösterdiğinden, önerilen düzenlemelerin güvenilirliği için elastoplastik malzeme modeli ile sismik analizler de yapılmıştır. Perform-3D yazılımı aracılığıyla gerçekleştirilen bu analizler, optimizasyon sonucunda elde edilen çapraz yerleşimlerinin kuvvetli depremler altında yapısal performansa etkilerini değerlendirmek amacıyla kullanılmıştır. Bu çalışma, topoloji optimizasyonunun yalnızca elastik analizle sınırlı kalmaması, plastik davranışı da içeren doğrulamalarla desteklenmesi gerektiğini vurgulamaktadır.



Şekil 2.4. Bütünleşik yapı için optimize edilmiş analiz (Qiao et al., 2016)

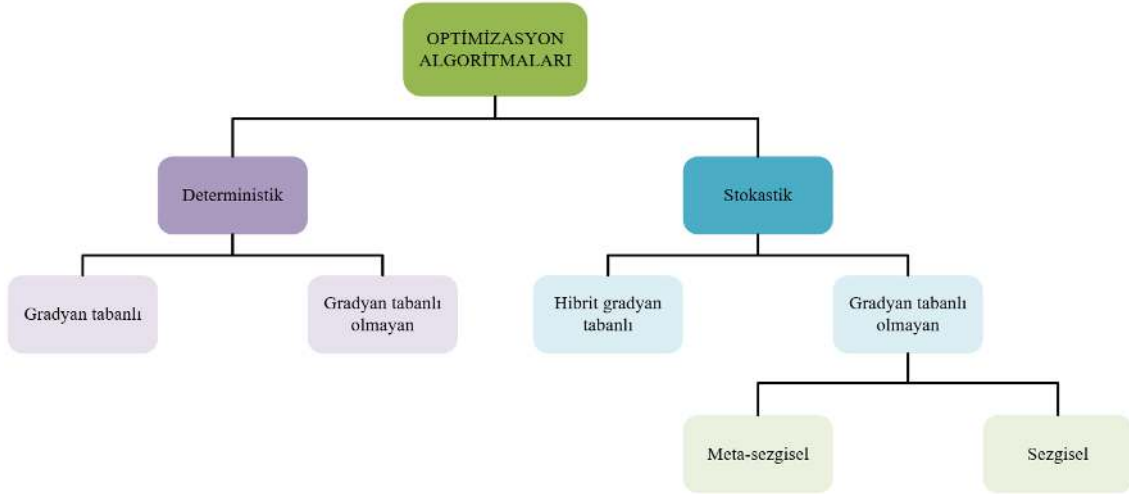
Lim vd., (2020) ise, uzay kafes sistemlerde burulma ve eğilme rijitliklerini maksimize ederken toplam kütleyi minimize etmeyi hedefleyen çok amaçlı bir topoloji optimizasyonu yöntemi önermiştir. Çalışmada, periyodik hücre yapıları kullanılarak uzay kafes sistemin parametrik tanımı yapılmış ve ardından çok amaçlı genetik algoritma (MOGA) ile şekil, boyut ve topoloji değişkenleri optimize edilmiştir. Optimizasyon sürecinde hem statik yükleme hem de dinamik titreşim kriterleri dikkate alınmış ve rijitlik ile hafiflik arasında optimal bir denge kurulmuştur. Elde edilen bulgular, periyodik hücre tabanlı parametrik modellerin yüksek performanslı uzay kafes sistemlerin tasarımı için etkin ve uygulanabilir bir yöntem sunduğunu ortaya koymuştur.

Şekil optimizasyonu ise yapısal eleman sayısının sabit kaldığı, ancak bu elemanların geometrik biçimlerinin değiştirildiği durumları ifade eder. Bu kapsamda, şekil optimizasyonu alanında literatürde farklı yöntemler geliştirilmiş ve çeşitli yapısal sistemlere uygulanmıştır. Cui ve Jiang, (2014) çalışmasında, uzaysal kısıtlar altında

çerçeve yapıların şekil ve topoloji optimizasyonu için bir morfogenez yaklaşımı geliştirmiştir. Yöntem, yapı elemanlarının doğrudan eklenmesi veya elenmesi ile düğüm noktalarının serbest ya da kısıtlı yer değişimleri üzerine kuruludur. Optimizasyon süreci, birim şekil değiştirme enerjisini minimize etmek amacıyla eleman ve düğüm duyarlılıklarına dayanmaktadır. Yöntem, tüm elemanların aynı kesitte olduğu varsayımıyla çalışır ve yapı genelinde eksenel kuvvet taşıma kapasitesini artırmaya yöneliktir. Elemanların katkısı azsa elenir, yüksek katkı gösteren bölgelerde ise yeni elemanlar eklenir. Düğümlerin konumu, şekil değiştirme enerjisini azaltacak şekilde iteratif olarak güncellenir. Bu süreçte yapının geometrik ve mimari kısıtları dikkate alınarak düğümler yalnızca tanımlı alanlar içinde hareket eder. Sayısal örnekler, yöntemin yapıdaki momentleri düşürdüğünü ve rijitliği artırdığını göstermektedir. Ayrıca, başlangıç geometrisine bağlı olarak farklı ve rasyonel yapı formlarının elde edilebildiği belirtilmiştir. Kociecki ve Adeli, (2015) karmaşık geometrilere sahip serbest biçimli çelik uzay çerçeve çatı yapılarının eş zamanlı boyutlandırma, topoloji ve şekil optimizasyonu için yeni bir evrimsel hesaplama yöntemi önermiştir. Amaç, yapının genel mimari formunu koruyarak hem yapı geometrisini hem de taşıyıcı elemanların kesitlerini ve topolojisini optimize ederek yapısal verimliliği artırmaktır. Mimari bütünlüğün korunması için şekil değişimlerine sezgisel sınırlar getirilmiştir. Yöntem, Kanada'nın Ottawa kentinde inşa edilen ve 2018'de tamamlanması planlanan Ottawa Hafif Raylı Sistemine ait iki farklı istasyon yapısına uygulanmış; %10–16 aralığında yapısal verim artışları elde edilmiştir. Martínez-Frutos vd., (2016) tarafından gerçekleştirilen çalışmada ise sürekli yapıların stokastik şekil optimizasyonu için seviye-küme tabanlı bir yöntem önerilmiştir. Yöntemin amacı, yapısal sağlamlık göstergeleri olarak beklenen uyumluluğun ve varyansının en aza indirilmesidir. Sürekli yapı davranışı, belirsiz yükleme ve malzeme özellikleri altında doğrusal elastisite denklemleriyle modellenmiştir. Belirsizlik, farklı olasılık dağılımlarına sahip rastgele değişkenler veya rassal alanlar aracılığıyla ifade edilmiştir. Uygun problem tanımı, izin verilen şekiller kümesi üzerindeki geometrik kısıtlar altında çözümün varlığının kanıtlanmasıyla sağlanmıştır. Önerilen yöntem, stokastik elastisite problemini zayıf formda ele almakta ve sürekli şekil türevleri için açık ifadeler sunmaktadır. Sayısal örnekler aracılığıyla yaklaşımın etkinliği ortaya konulmuştur.

Literatürde, çeşitli optimizasyon problemlerinin çözümüne yönelik olarak çok sayıda algoritma geliştirilmiştir. Ancak, bu algoritmalar arasından ele alınan probleme en

uygun yöntemin seçilmesi, çözümün başarısı açısından kritik öneme sahiptir. Şekil 2.5'te, optimizasyon algoritmalarının temel türleri genel hatlarıyla sınıflandırılarak gösterilmektedir.



Şekil 2.5. Optimizasyon algoritmalarının türleri (Zakian and Kaveh, 2023)

Optimizasyon problemlerinin çözümünde kullanılan algoritmalar genel olarak deterministik ve stokastik olmak üzere iki temel kategoriye ayrılmaktadır. Deterministik algoritmalar, tamamen matematiksel ilkelere dayalı olarak çalışmakta olup, aynı başlangıç koşullarında her zaman aynı sonuçları üretmektedirler. Buna karşılık, stokastik algoritmalar rastgelelik içerdiğinden, aynı başlangıç koşullarında bile farklı çalıştırmalarda farklı sonuçlar elde edilebilmektedir.

Deterministik yöntemler genellikle tek bir çözüm yolunu takip ederek iteratif biçimde ilerlerken, stokastik algoritmalar çok sayıda başlangıç noktası üzerinden, çok parçalı bir yapı ile çözüm uzayını daha geniş bir perspektiften tarar. Bu özellik, özellikle değişken sayısı yüksek ve doğrusal olmayan (nonlinear), konveks olmayan yapıya sahip fonksiyonların çözümünde belirleyici olmaktadır. Günümüz mühendislik problemleri, çoğunlukla bu tür karmaşık yapılar içerdiğinden deterministik algoritmalar, lokal optimumlara sıkışma eğilimi göstermekte ve global optimuma ulaşmakta yetersiz kalabilmektedir.

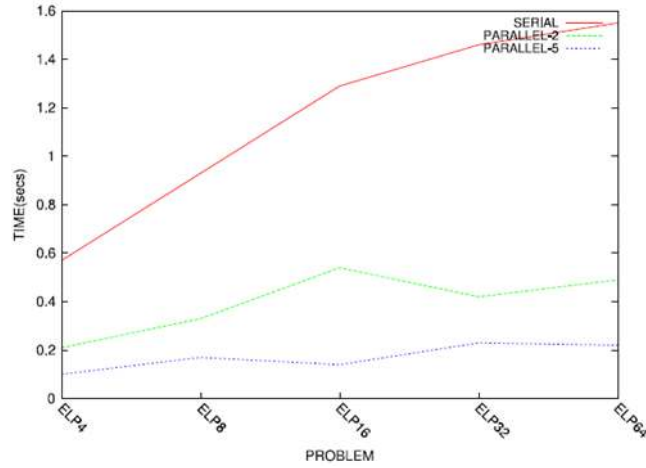
Ayrıca, deterministik yöntemlerin çoğu, gradyan ya da Hessian matrisi gibi türev bilgilerine ihtiyaç duymaktadır. Bu durum, karmaşık veya türevi alınamayan fonksiyonlar söz konusu olduğunda yöntemlerin uygulanabilirliğini sınırlamaktadır. Bu nedenlerle, stokastik algoritmalar yalnızca daha esnek ve uygulanabilir değil, aynı zamanda

mühendislik uygulamaları açısından daha verimli ve etkili çözümler sunma potansiyeline sahiptir (Saka et al., 2016).

Stokastik algoritmaların bir alt kümesi olan üstsezgisel (meta-sezgisel) algoritmalar üzerine literatürde çok sayıda çalışma mevcuttur ve bu algoritmalara dayalı olarak birçok optimizasyon yöntemi geliştirilmiştir. Bu algoritmalar arasında doğadan esinlenerek geliştirilen metafor tabanlı yaklaşımlar önemli bir yer tutmaktadır. Örnek olarak; Parçacık Sürü Optimizasyonu (PSO) (Kennedy and Eberhart, 1995a), Genetik Algoritmalar (GA) (Holland, 1992), Diferansiyel Evrim Algoritması (DE) (Storn and Price, 1995) ve Karınca Koloni Optimizasyonu (ACO) (Dorigo and Stützle, 2004) verilebilir. Bunun yanı sıra, Aritmetik Optimizasyon Algoritması (AOA) (Abualigah et al., 2021) gibi metafor içermeyen yaklaşımlar da literatürde yer almaktadır.

Son yıllarda, yeni algoritmalar geliştirmekten ziyade, mevcut üstsezgisel algoritmaların daha hızlı ve daha etkin çözümler üretebilecek biçimde iyileştirilmesine yönelik çalışmalar ön plana çıkmıştır. Bu bağlamda (Halim et al., 2021)'de vurgulandığı üzere, üstsezgisel algoritmaların rastlantısal doğası nedeniyle tek bir çalıştırma ile algoritma performansının değerlendirilmesi yeterli değildir. Yakınsama oranı, arama popülasyonlarının konum farklılıkları, hesaplama maliyeti, problem karmaşıklığı gibi ölçütlerin yanı sıra, sıralı çözümler veya birikimli dağılım gibi istatistiksel kriterler de değerlendirme sürecinde dikkate alınmalıdır.

Buna ek olarak, paralel işlem teknolojisinin gelişmesi ile birlikte optimizasyon algoritmalarında işlem sürelerinin azaltılması mümkün hale gelmiştir. Günümüzdeki çok çekirdekli işlemciler, paralel işleme yaklaşımı ile birden fazla çözüm adayının eşzamanlı değerlendirilmesine olanak tanımaktadır. Bu durum, toplam analiz süresini önemli ölçüde kısaltmakta ve özellikle çok sayıda yinelemeye ihtiyaç duyan algoritmalarda performansı artırmaktadır. Paralel işlemenin literatürdeki ilk sistematik tanımı, 1967 yılında Gene M. Amdahl'ın konferans bildirisiyle yapılmış ve bu yaklaşım daha sonra "Amdahl Kanunu" olarak anılmıştır (Amdahl, 2007). Tsoulos vd., (2016) tarafından sunulan grafik örneğinde, farklı sayıda bilgisayarın paralel bağlandığı durumlarda elde edilen hızlanmalar karşılaştırılmıştır (Şekil 2.6). Literatürde birçok üstsezgisel algoritma, bu avantajdan yararlanmak üzere paralel işlemeye uygun biçimde yeniden yapılandırılmıştır (Alba et al., 2004; Fan and Chang, 2009; Groenwold and Hindley, 2002; Yuan et al., 2015; Zhong and Pusheng, 1995).



Şekil 2.6. Seri ve parallel bağlı bilgisayarlardaki optimizasyon süreleri (Tsoulos et al., 2016)

Bu gelişmelerin etkisiyle, yapı mühendisliği dahil olmak üzere birçok mühendislik disiplininde karmaşık optimizasyon problemlerinin çözümü mümkün hale gelmiş ve bu alana yönelik akademik ilgi artmıştır. Yapısal optimizasyon alanındaki ilk çalışmalardan biri, 1976 yılında Dobbs ve Nelson tarafından gerçekleştirilmiş ve kafes sistemlerin kesit optimizasyonuna odaklanmıştır (Dobbs and Nelson, 1976). Bunu takiben, 1990 yılında Hajela tarafından yapılan çalışmada, bir kafes sistemin ağırlığı Genetik Algoritma ile minimize edilmeye çalışılmış; nodal deplasmanlar ise tasarım kısıtı olarak dikkate alınmıştır (Hajela, 1990). Sonraki yıllarda yapılan çok sayıda çalışma, bu temel yaklaşımların üzerine inşa edilmiş, eleman gerilmeleri ve nodal deplasmanlar tasarım kısıtları olarak kullanılmış ve farklı üstsezgisel algoritmaların performansları kıyaslanmıştır (Hasançebi et al., 2009; Pouriyanezhad et al., 2021; Sadollah et al., 2015; Serra and Venini, 2006; Talatahari et al., 2012).

2.3. Otomasyon

Otomasyon, insan müdahalesini en aza indirerek süreçlerin, sistemlerin veya makinelerin kendi kendine çalışmasını sağlayan teknolojik bir yaklaşımdır. Otomasyonun en büyük faydası; iş gücünden tasarruf sağlaması, enerji ve malzeme kullanımını azaltması, kaliteyi, doğruluğu ve hassasiyeti artırmasıdır (Mamodiya and Sharma, 2014). Bu sistemler; kontrol, izleme, karar verme ve uygulama gibi işlevleri otomatik olarak gerçekleştirerek hem verimliliği artırmakta hem de hata oranlarını düşürmektedir. Otomasyon kavramı, üretim endüstrisinden sağlık sektörüne kadar birçok alanda uygulanmakta olup, mühendislik disiplinlerinde de giderek artan bir öneme sahiptir.

İnşaat mühendisliği özelinde otomasyon, planlama, tasarım, yapım ve bakım gibi süreçlerin dijital sistemlerle desteklenmesi yoluyla süreçlerin daha hızlı, güvenilir ve maliyet etkin bir şekilde yürütülmesini amaçlamaktadır. Bu bağlamda, inşaatta güvenlik planlaması (H. Chen and Chan, 2021), bina bilgi modellemesi (BIM) (Vähä et al., 2013) ve 3B yazıcılarla yapı üretimi (Major and Gavardashvili, 2022) gibi teknolojiler inşaat sektöründe yaygın biçimde kullanılmaktadır. Otomasyon sayesinde proje yönetimi daha etkin hale gelmekte, iş gücü kaynaklı hatalar azalmakta ve iş sağlığı ve güvenliği açısından önemli kazanımlar elde edilmektedir. S. S. Chen ve Shirolé, (2006) inşaat mühendisliğinde otomasyonun özellikle köprü projelerinde bilgi akışını iyileştirme ve süreçleri hızlandırma potansiyelini ele almıştır. Mevcut uygulamalarda tasarım, üretim, inşaat ve bakım aşamaları arasında bilgi aktarımı büyük ölçüde parçalı, manuel ve tekrarlıdır; bu durum hem hatalara neden olmakta hem de zaman ve kaynak kaybına yol açmaktadır. Bu soruna çözüm olarak, yazarlar parametrik üç boyutlu köprü bilgi modellemesi (3B BIM) kavramını önermektedir. Tek ve merkezi bir 3B model aracılığıyla tüm proje paydaşlarının (tasarımcı, üretici, müteahhit vb.) eş zamanlı ve doğru bilgiye erişmesi sağlanarak, otomasyonun tüm proje sürecine entegre edilmesi mümkün olmaktadır. Çalışmada, 3B modelleme sayesinde analiz, malzeme metrajı, CNC üretim dosyaları oluşturma ve montaj planlaması gibi işlemlerin doğrudan model üzerinden otomatik biçimde yürütülebileceği vurgulanmıştır. Bu yönüyle çalışma, inşaat mühendisliğinde otomasyonun yalnızca üretim süreçlerini değil, aynı zamanda tasarım, veri yönetimi ve yaşam döngüsü yönetimi gibi bilişsel süreçleri de kapsamı gerektiğini göstermektedir. Pan vd., (2017) ise mobil vinçlerin şantiyelerde konumlandırılmasına yönelik bir otomasyon yöntemi geliştirmiştir. Yöntem, BIM tabanlı veri entegrasyonu ve 3B simülasyon kullanarak vinç yerleşim sürecini çevresel, operasyonel ve güvenlik kısıtları altında değerlendirmekte; uygun olmayan konumları otomatik olarak elemektedir. Son aşamada ise sistem, yeniden konumlandırma sayısını ve kaldırma süresini en aza indirerek optimum vinç konumunu belirlemektedir. Bu yaklaşım, kaldırma operasyonlarında güvenliği artırırken verimliliği yükseltmekte ve vinç planlamasını deneyime dayalı manuel süreçlerden çıkarıp tam otomatik bir karar destek sistemine dönüştürmektedir.

Yapı mühendisliği alanında ise otomasyon uygulamaları özellikle yapı tasarımı, analiz ve performans değerlendirme süreçlerinde öne çıkmaktadır. Literatürdeki çalışmalar, otomasyonun taşıyıcı sistemlerin boyutlandırılması, deprem performansı

analizi, hasar tespiti, yapı sağlığı izleme (SHM) ve optimizasyon gibi alanlarda etkin biçimde kullanıldığını göstermektedir (Farrar and Worden, 2007). Örneğin, bazı araştırmalarda yapay zeka tabanlı algoritmalar kullanılarak yapı elemanlarının en uygun kesitlerinin otomatik olarak belirlenmesi hedeflenmiş (Camp et al., 2003); diğer çalışmalarda ise görüntü işleme ve sensör verileri yardımıyla hasar tespiti ve bakım kararlarının otomatikleştirilmesi üzerinde durulmuştur (Kulambayev et al., 2024; Potenza et al., 2017). Son yıllarda, bu tür uygulamaların sayısı artmakta ve yapı mühendisliğinde otomasyonun hem bilimsel araştırmalarda hem de endüstriyel uygulamalarda giderek daha fazla yer bulduğu gözlemlenmektedir.

Yapı mühendisliğinde otomasyonun önemi giderek artmakta, bu alandaki teknolojik gelişmeler mühendislik uygulamalarında köklü dönüşümlere yol açmaktadır. Major ve Gavardashvili, (2022) yapı mühendisliğinde otomasyonun, insan müdahalesini azaltarak süreçlerin elektronik sistemler ve bilgi teknolojileriyle yürütülmesini sağladığını vurgulamaktadır. Özellikle toprak işleri, tünel kazıları ve yol yapımında kullanılan otomatik yönlendirme sistemleri sayesinde işlerin doğruluğu ve verimliliği artmakta, iş gücü ihtiyacı azalmaktadır. Ayrıca tasarım sürecinde BIM teknolojisi ve bilgisayar destekli hesaplama araçlarının kullanımı, karmaşık projelerin daha hızlı, hatasız ve optimize biçimde üretilmesine olanak tanımaktadır. Yazarlar, otomasyonun sadece uygulama değil, tasarım ve planlama aşamalarında da mühendislik süreçlerini köklü biçimde dönüştürdüğünü belirtmiştir.

Yapılan literatür taraması, görüntü işleme tekniklerinin farklı mühendislik disiplinlerinde yaygın biçimde kullanıldığını ortaya koymaktadır. Ancak yapı tasarımının otomasyonu amacıyla, görüntü işleme yöntemlerinin doğrudan uygulandığı özgün bir çalışmaya rastlanmamıştır. Bu bağlamda, mevcut tez çalışması literatüre üç temel açıdan yenilikçi katkı sunmaktadır:

(1) Mimari planlardan kolon yerleşimlerinin görüntü işleme yoluyla otomatik olarak belirlenmesi ve bu sayede yapısal analiz süreçlerine doğrudan uyumlu veri elde edilmesi;

(2) Betonarme binaların kolon topolojisinin (varlık-yokluk durumu ve konumu) ve kesit özelliklerinin eşzamanlı olarak optimize edilmesiyle taşıyıcı sistemin tüm yönleriyle bütüncül biçimde tasarlanması;

(3) Bu iki adımın uçtan uca bir otomasyon sistemi içinde entegre edilerek, yalnızca mimari plan giriřiyle başlayan sürecin yapısal modellemeyen analize kadar minimum kullanıcı müdahalesiyle tamamlanması.

Bu yönleriyle çalışma, mevcut literatürdeki yaklaşımlardan hem yöntem hem de uygulama kapsamı bakımından ayrılmakta ve yapı tasarımı sürecine otomasyon temelli yeni bir bakış açısı getirmektedir.



3. YÖNTEM

3.1. Genel Yaklaşım

Son yıllarda bilgi teknolojilerindeki hızlı gelişmeler, endüstriyel üretimden hizmet sektörüne kadar pek çok alanda otomasyon uygulamalarının yaygınlaşmasına öncülük etmiştir. Özellikle yapay zeka, makine öğrenmesi, görüntü işleme ve veri analitiği gibi alanlardaki ilerlemeler; insan müdahalesini azaltan, hata oranlarını düşüren ve süreçleri daha verimli kılan sistemlerin geliştirilmesini mümkün kılmıştır. Bu dönüşüm, yalnızca üretim odaklı sanayilerle sınırlı kalmayıp mühendisliğin çeşitli disiplinlerine de etki etmiştir.

İnşaat mühendisliği, geleneksel olarak yüksek oranda insan kararına ve deneyime dayanan bir meslek alanı olmasına rağmen, karmaşık karar süreçlerinin sistematik ve tekrarlanabilir şekilde ele alınması ihtiyacı doğrultusunda otomasyon tekniklerinin entegrasyonuna giderek daha fazla açık hale gelmiştir. Gerek tasarım gerekse uygulama safhasında, yapısal sistemlerin sayısal olarak modellenmesi, değerlendirilmesi ve optimize edilmesi süreçleri, manuel yöntemlerin sınırlılıkları nedeniyle giderek otomatikleştirilmeye başlanmıştır. Büyük ölçekli veri analizleri, çok değişkenli optimizasyon problemleri ve görsel veri işleme gereksinimleri, inşaat mühendisliğinde otomasyon temelli çözümleri yalnızca avantajlı değil, aynı zamanda kaçınılmaz hale getirmektedir.

Bu çalışmada geliştirilen yöntem, inşaat mühendisliği problemlerine otomasyon odaklı yaklaşımların nasıl entegre edilebileceğini göstermeyi amaçlamakta; özellikle yapısal tasarım süreçlerinde görüntü işleme teknikleri ve sezgisel optimizasyon algoritmalarının birlikte kullanımına dayanmaktadır. Geliştirilen sistemin amacı, insan girdisini asgariye indirerek, hem daha hızlı hem de daha sistematik bir tasarım süreci ortaya koymaktır.

Bu kapsamda, çalışmanın ilk aşamasında mimari planların dijital ortamda analiz edilerek taşıyıcı sistem elemanlarının yerleşimine temel oluşturacak yapısal aks sisteminin otomatik olarak belirlenmesi hedeflenmiştir. Günümüzde mimari projeler büyük ölçüde CAD tabanlı olarak hazırlanmakta ve bu projelerden aks sistemini manuel olarak elde etmek hem zaman alıcı hem de öznel kararlara bağlı bir süreç olmaktadır. Bu nedenle, çalışma kapsamında görüntü işleme teknikleri kullanılarak plan verisi sistematik biçimde analiz edilmiş; bu sayede akslar ve olası kolon yerleşim noktaları algoritmik

yöntemlerle belirlenebilmiştir. Böylece, taşıyıcı sistem tasarımına yönelik ilk ve en kritik kararların nesnel ve tekrar edilebilir biçimde üretilmesi sağlanmıştır.

Sonraki aşamada, elde edilen aks sistemi üzerine yerleştirilebilecek kolon konumları, yapı maliyeti ve yönetmelik kontrolleri doğrultusunda değerlendirilerek optimizasyon süreci gerçekleştirilmiştir. Bu işlemde, yapının hem taşıyıcı güvenliği hem de ekonomikliği dikkate alınmakta; kolon yerleşimleriyle birlikte uygun kolon ve kiriş kesitlerinin belirlenmesi çok değişkenli bir optimizasyon problemi olarak ele alınmıştır. Problem, yüksek boyutlu ve büyük ölçüde ayrık bir çözüm uzayına sahiptir; ayrıca geçerli çözümler bu uzayın yalnızca çok küçük bir alt kümesini oluşturmaktadır. Dahası, çözüm sürecinde değişken vektörü ile amaç fonksiyonu arasında doğrudan bir matematiksel bağıntı kurulamadığı için, klasik türev tabanlı optimizasyon yöntemlerinin uygulanması mümkün değildir. Çünkü her çözüm adayının değerlendirilmesi, değişken vektöründen yola çıkılarak yapı modelinin algoritmik olarak oluşturulmasını, bu modelin sayısal analizle çözümlenmesini ve elde edilen sonuçlardan amaç fonksiyonu bileşenlerinin türetilmesini gerektirmektedir. Bu nedenle, sezgisel optimizasyon yaklaşımları tercih edilmiştir.

Optimizasyon sürecinde, her bir çözüm adayının değerlendirilmesi bağımsız analizler gerektirdiğinden, işlem yükü doğal olarak yüksek bir paralellik potansiyeli taşımaktadır. Bu çalışmada, analizlerin çok sayıda çözüm adayı için eşzamanlı olarak yürütülebilmesini sağlamak amacıyla paralel işlemeye dayalı bir yapı benimsenmiştir. Geliştirilen sistem, çok çekirdekli işlemciler üzerinde aynı anda birden fazla analiz sürecini çalıştırarak toplam hesaplama süresini önemli ölçüde azaltmaktadır. Bu yapı sayesinde, özellikle üst düzey nüfus büyüklükleri ve çok iterasyonlu senaryolarda bile sistemin uygulanabilirliği ve pratikliği korunmaktadır. Paralel işleme stratejisi, OpenSEES (<http-1>) analizlerinin MATLAB (<http-2>) ortamından çağrıldığı noktada, her çözüm adayına ait yapı modelinin eşzamanlı oluşturulması ve analiz edilmesi prensibine dayanmaktadır. Böylece yalnızca algoritmanın yapısal etkinliği değil, hesaplama sürecinin zamansal verimliliği de gözetilmiştir.

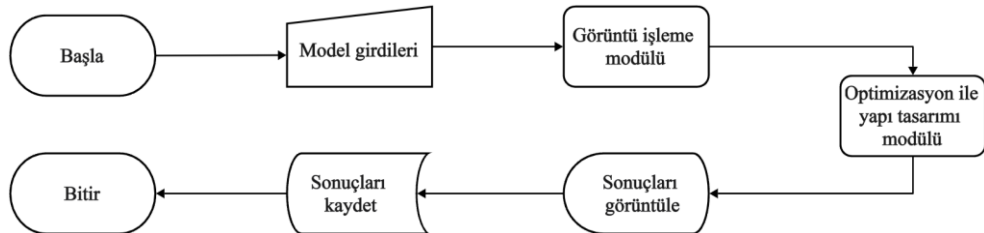
Son olarak, her bir çözüm adayına karşılık gelen taşıyıcı sistem modeli, sayısal analiz amacıyla OpenSEES yazılımına otomatik olarak aktarılmakta ve doğrusal statik analiz bu platformda gerçekleştirilmektedir. OpenSEES'in tercih edilme nedeni, benzer analiz yazılımlarına (SAP2000, ETABS, ProtaStructure vb.) kıyasla çok daha yüksek

analiz hızı sunmasıdır. Bu hız avantajı, özellikle optimizasyon sürecinde binlerce aday çözümün değerlendirilmesi gerektiği durumlarda belirleyici hale gelmektedir.

OpenSEES'in bu performansı, sonlu elemanlar analizinden sonra gerçekleştirilen sonuç işleme aşamasındaki farklılıktan kaynaklanmaktadır. Ticari yazılımlar, kullanıcıya zengin bir etkileşim ve görsel çıktı sağlamak amacıyla analiz sonrası çok sayıda büyüklüğü otomatik olarak hesaplayıp, arayüz üzerinden görüntülenmeye hazır halde bellekte saklamaktadır. Bu durum kullanıcı deneyimi açısından avantajlı olmakla birlikte, çok sayıda çözümün ardışık analizini gerektiren optimizasyon senaryolarında ciddi zaman kayıplarına yol açmaktadır. OpenSEES ise yalnızca kullanıcı tarafından açıkça tanımlanan büyüklükleri kayıt altına almakta, bunun dışındaki tüm hesaplamaları atlayarak analiz süresini önemli ölçüde kısaltmaktadır.

Her aday çözüm, kod içerisinde tanımlanan algoritmalar aracılığıyla OpenSEES model dosyasına dönüştürülmekte; analiz tamamlandıktan sonra elde edilen sınırlı ve hedef odaklı çıktılar MATLAB ortamında okunarak değerlendirme sürecine aktarılmaktadır. Taşıyıcı sistemin yer değiştirme sınırları, kat ötelemesi, kesit yeterliliği ve yönetmelik kontrollerine ilişkin değerlendirme, tamamen kullanıcı tanımlı ceza mekanizmaları ile MATLAB tarafında gerçekleştirilmektedir. Böylece her çözüm adayına ait uygunluk değeri, hem yapısal davranışa hem de tasarım kısıtlarına dayalı çok ölçütlü bir biçimde hesaplanmaktadır.

Özetle, geliştirilen yöntem; yalnızca mimari plana ait görsel veriyi kullanarak, kullanıcıdan minimum düzeyde bilgi girişi gerektiren ve taşıyıcı sistemin optimum maliyetle tasarlanmasına olanak tanıyan uçtan uca bir otomasyon süreci sunmaktadır. Bu bütünleşik yaklaşım, hem aks sisteminin elde edilmesi hem de kolon topolojisi ve kesit optimizasyonunun gerçekleştirilmesini tek bir çatı altında birleştirmekte; aynı zamanda benzer türdeki yapı tasarım problemlerine uyarlanabilir, genelleştirilebilir bir metodolojik çerçeve ortaya koymaktadır.



Şekil 3.1. Geliştirilmiş yazılımın genel akış diyagramı

Bu bölümde, geliştirilen yöntemin genel kapsamı ortaya konulmuş ve sürecin temel bileşenleri açıklanmıştır. Şekil 3.1’de sunulan akış diyagramı, yöntemin bütünsel işleyişini özetlemekte olup, diyagramda yer alan her bir bileşen izleyen alt bölümlerde sistematik biçimde ele alınacaktır.

3.2. Model Girdileri

Bu bölümde, geliştirilen yapısal tasarım modeline ilişkin temel girdiler ile modelleme sürecinde kabul edilen ön varsayımlar açıklanmıştır. Optimizasyon ve analiz süreçlerinin doğruluğu ve geçerliliği, büyük ölçüde kullanılan girdi parametrelerine ve bu parametrelerin tanımlanma biçimlerine bağlıdır. Bu kapsamda, taşıyıcı sistemin geometrik özellikleri, yükleme koşulları, malzeme özellikleri, kesit kütüphanesi, sınır koşulları ve analiz tipi gibi modelleme sürecini doğrudan etkileyen temel girdilere yer verilmiş; ayrıca hesaplama sürecinin sadeleştirilmesi amacıyla yapılan varsayımlar sistematik biçimde ortaya konulmuştur. İlgili girdiler, hem yapı sisteminin fiziksel temsili hem de optimizasyon algoritmasının çözüm uzayını belirlemesi açısından kritik öneme sahiptir. Bu doğrultuda, aşağıda sırasıyla yapısal modelde kullanılan parametreler ve bunlara ilişkin kabuller ayrıntılı biçimde sunulmuştur.

3.2.1. Kullanıcı girdileri

Geliştirilen otomasyon tabanlı yapı tasarım sisteminin yürütülebilmesi için bazı temel parametrelerin kullanıcı tarafından önceden tanımlanması gerekmektedir. Bu parametreler, modelin başlangıç geometrisini, yapısal elemanlara ilişkin tasarım seçeneklerini, analiz koşullarını ve optimizasyon sürecine ait sınırları doğrudan belirlemektedir. Kullanıcı girdileri yalnızca modelleme doğruluğunu etkilemekle kalmamakta, aynı zamanda sistemin farklı kat sayısına, aks düzenine veya yükleme senaryosuna sahip betonarme binalar için yeniden yapılandırılmasına olanak tanımaktadır. Bu kapsamda tanımlanması gereken başlıca girdiler; mimari plana ait bilgi girişi, kullanılabilir kesit türlerinin belirtildiği kesit kütüphanesi, betonarme sistemlere ait malzeme özellikleri, düşey ve yatay yükleme senaryolarının tanımı, sezgisel algoritmanın yapılandırılmasına yönelik optimizasyon parametreleri ve işlem yükünün dağıtımını etkileyen paralel işleme girdileridir. Bu alt başlıkta, söz konusu girdiler sistematik biçimde açıklanmış ve her birinin modelleme sürecindeki işlevi ortaya konulmuştur.

3.2.1.1. *Mimari plan*

Bu başlık altında tanımlanan kullanıcı girdileri, görüntü işleme sürecinin sağlıklı biçimde başlatılabilmesi için gereklidir. Mimari plan verisi, yazılımın taşıyıcı sistem akslarını doğru şekilde tespit edebilmesi açısından temel bir bileşen olup, bu doğrultuda kullanıcıdan aşağıdaki üç parametrenin tanımlanması beklenmektedir:

- 1- **Plan Görseli:** Herhangi bir CAD programında çizilmiş bir mimari planın, ekran alıntısı yoluyla oluşturulmuş bir görüntü dosyasıdır. Bu dosya, uzantısıyla birlikte tam adı belirtilerek sisteme tanıtılmaktadır.
- 2- **Kenar Belirleme Hassasiyeti:** Görüntüdeki duvar kenarlarının net biçimde algılanabilmesini sağlayan ve kullanıcı tarafından bir kaydırma çubuğu aracılığıyla ayarlanabilen tamsayı bir parametredir. Bu ayar, yazılımın görüntüyü işleme biçimini doğrudan etkilemekte ve kullanıcı, değişkenin etkisini arayüzde görüntünün yeniden oluşturulmuş hali üzerinden görsel olarak değerlendirebilmektedir.
- 3- **Görüntü İşleme Sürecinde Görsellerin Gösterimi:** Görüntü işleme süreci sırasında kullanıcıya kontrol imkanı sunmak amacıyla, gri seviye görüntü veya aks belirleme gibi adımlara ilişkin sonuçlar için görseller oluşturulmakta ve kullanıcıya sunulabilmektedir. Bu görsellerin gösterilip gösterilmeyeceği kullanıcı tercihiyle bırakılmıştır. Bu amaçla, görsel çıktılar bir parametreye bağlanmış; kullanıcı isterse tüm işlem süreci boyunca hiçbir görsel gösterilmeden işlem tamamlanabilmektedir.

Yukarıda belirtilen görüntü işleme kavramlarına ilişkin detaylı açıklamalar, ilgili algoritmalarla birlikte Bölüm 3.4’te sunulmuştur.

3.2.1.2. *Yapı geometrisi*

Yazılıma yalnızca mimari kat planı görüntüsü sağlandığı için, binaya ait kat sayısı ve kat yüksekliği bilgileri kullanıcı tarafından metre cinsinden manuel olarak tanımlanmalıdır. Yazılımın mevcut sürümünde, bina yüksekliği, Türkiye Bina Deprem Yönetmeliği 2018 (TBDY, 2018) Tablo 3.3’te tanımlanan Bina Yükseklik Sınıfı 5 - 8 (BYS 5-BYS 8) aralığıyla sınırlı tutulmuştur ($H_N \leq 28$). Bu sınırlama, yürürlükteki yönetmelik koşullarına uygunluk amacıyla getirilmiştir. İlerleyen sürümlerde, ilgili parametrelerin de otomasyon sürecine entegre edilmesiyle birlikte, bu tür sınırlamaların ortadan kaldırılması hedeflenmektedir.

3.2.1.3. Kesit kütüphanesi

Kesit kütüphanesinin tanımlanması için, kullanıcıya önceden hazırlanmış bir Excel dosyası sağlanması uygun görülmüştür. Bu tercih, hem veri girişinin standartlaştırılması hem de kullanım kolaylığı açısından önemlidir.

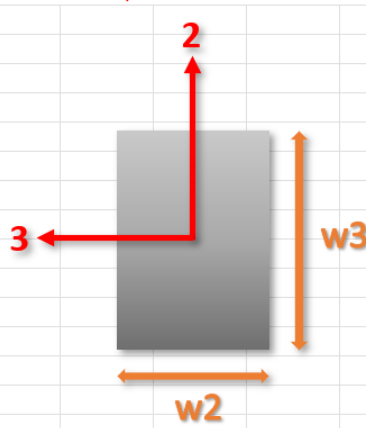
Kesitlerin tanımlanmasına yönelik olarak başlıca iki yöntem değerlendirilmiştir. İlk yöntemde, kullanıcı kesit boyutları için alt ve üst sınır değerlerini tanımlar ve sistem, bu sınırlar arasında tüm olası kesit kombinasyonlarını otomatik olarak üretir. Bu yaklaşım, kullanıcıya zaman kazandırmakla birlikte, girişlerin önceden belirli bir düzende artması veya azalması zorunluluğu nedeniyle esnekliği sınırlamaktadır. Nitekim, kullanıcı farklı eksenler için farklı artış aralıkları tanımlamak veya bazı kesit boyutlarını hariç tutmak isteyebilir.

İkinci yöntemde ise, kullanıcı tüm kesit boyutlarını doğrudan ve açık şekilde kendisi tanımlar. Bu yöntemin, özellikle özgül tasarım ihtiyaçlarını karşılamada ve farklı yönlerde göre değişen kesit geometrilerinin belirtilebilmesinde daha esnek olduğu görülmüştür. Bu nedenle, çalışmada ikinci yöntem tercih edilmiş ve kullanıcıdan, bina tasarımında kullanılabilir olmasını istediği tüm kesitleri, önceden oluşturulmuş Excel dosyasına eksiksiz biçimde girmesi beklenmiştir.

Son olarak, paspayı değeri de kullanıcı tarafından metre cinsinden tanımlanmalıdır. Bu parametre, yapı elemanlarının donatı alanı hesaplamalarında kullanılmakta olup, tüm elemanlar için sabit ve ortak bir değer olarak kabul edilmektedir.

Kolon (w2,w3)		Kiriş (w2,w3)							
0.3	0.3	0.3	0.4						
0.3	0.4	0.3	0.5						
0.3	0.5	0.3	0.6						
0.3	0.6	0.3	0.7						
0.4	0.3	0.4	0.4						
0.4	0.4	0.4	0.5						
0.4	0.5	0.4	0.6						
0.4	0.6	0.4	0.7						
0.5	0.3	0.5	0.5						
0.5	0.4	0.5	0.6						
0.5	0.5	0.5	0.7						
0.5	0.6								
0.6	0.3								
0.6	0.4								
0.6	0.5								
0.6	0.6								

Kesit boyutlarının birimi metredir.



Şekil 3.2. Kesit kütüphanesine ilişkin hazırlanmış Excel dosyasında örnek bir kesit havuzu girdisi

Kolon ve kiriş kesitleri ayrı ayrı tanımlanmakta olup, bu kesitlere dair örnek tablo Şekil 3.2’de sunulmuştur. Tabloda yer alan w_2 ve w_3 parametreleri, kolon ve kirişlerin sırasıyla yerel 2 ve 3 eksenlerine dik olan kesit boyutlarına karşılık gelmektedir. Bu eksenlerin global koordinat sisteminde hangi yönler karşılık geldiği ve modelleme açısından taşıdığı anlam, Bölüm 3.3’te ayrıntılı biçimde açıklanmıştır.

3.2.1.4. Malzeme özellikleri

Kesit kütüphanesinde olduğu gibi, kullanım kolaylığı sağlamak amacıyla malzeme özelliklerinin tanımlanabileceği önceden hazırlanmış bir Excel tablosu oluşturulmuştur. Sonlu elemanlar modelinin oluşturulabilmesi ve her eleman için gerekli donatı alanlarının hesaplanabilmesi için, beton ve donatı çeliğine ait bazı temel malzeme parametrelerinin kullanıcı tarafından tanımlanması gerekmektedir. Buna ek olarak, optimizasyon sürecinde maliyet hesabının yapılabilmesi için beton ve donatı çeliği birim fiyatları da tabloda girilmesi gereken parametreler arasındadır.

Bahsi geçen Excel tablosuna örnek olarak girilmiş bazı malzeme değerleriyle birlikte girilmesi gereken parametreler, Tablo 3.1 ve Tablo 3.2’de sunulmuştur.

Tablo 3.1. Beton için malzeme özellikleri tablosu

Beton Sınıfı	f_{ck} (kPa)	ν	E (kPa)	f_{ctk} (kPa)	ϵ_{cu}	k_1	k_3	Birim Fiyat (₺/m ³)
C30	30000	0.2	3.20E+07	1900	-0.003	0.82	0.85	2200

Tablo 3.2. Donatı çeliği için malzeme özellikleri tablosu

Çelik Sınıfı	f_y (kPa)	E (kPa)	ρ (ton/m ³)	Birim Fiyatlar (₺/ton)	
				Ø8- Ø10	Ø12- Ø32
B420C	420000	2.00E+08	7.856	26900	26350

Tablolarda yer alan parametreler aşağıda tanımlanmıştır:

- f_{ck} , betonun karakteristik basınç dayanımıdır.
- ν , betonun Poisson oranını ifade eder.
- E, betonun elastisite modülüdür.
- f_{ctk} , betonun karakteristik çekme dayanımıdır.
- ϵ_{cu} , betonun maksimum basınç birim şekil değiştirmesidir.

- k_1 ve k_3 , dikdörtgen kesitli elemanlarda sırasıyla eşdeğer basınç bloğunun derinliğini ve basınç gerilmesi katsayısını temsil eder (TS 500, 2000, Madde 7.1).
- f_y , donatı çeliğinin akma dayanımıdır.
- ρ , çeliğin yoğunluğunu ifade eder.

3.2.1.5. Yükleme tanımları

Yapı sisteminin sonlu elemanlar modeli üzerinden analiz edilebilmesi için, yükleme koşullarının doğru ve eksiksiz şekilde tanımlanması gerekmektedir. Bu bağlamda, kullanıcıdan düşey ve yatay yükleri temsil edecek bazı temel parametreleri önceden belirlemesi beklenmektedir.

Düşey yükler için, sabit ve hareketli olmak üzere iki farklı yükleme türü sistemde tanımlanmıştır. Sabit yükler için kullanıcıdan iki farklı parametrenin girişi istenmektedir. Birinci parametre, her katta yer alan kolon, kiriş ve döşeme elemanlarının birim alandaki ağırlığına karşılık gelen ve kN/m^2 cinsinden ifade edilen değerdir. İkinci parametre ise, kat düzleminde bulunan duvarların birim uzunluğuna karşılık gelen ağırlığını temsil etmekte olup kN/m cinsinden tanımlanmaktadır. Hareketli yük tanımı için ise kullanıcının, yapının kullanım amacına uygun olarak TS 498, (2021) Çizelge 7'ye dayanarak hareketli yük değerini kN/m^2 cinsinden girmesi yeterlidir.

Yatay yükler kapsamında yalnızca deprem etkisi dikkate alınmış ve bu amaçla eşdeğer deprem yükü yöntemi kullanılmıştır. Çalışma kapsamında bazı parametreler için sabit kabuller yapılmış olup, deprem yükünün hesaplanabilmesi için kullanıcı tarafından girilmesi gereken temel parametreler aşağıda sıralanmıştır:

- AFAD Türkiye Deprem Tehlike Haritaları internet sitesinden (<http-3>) binanın konumu seçilerek temin edilecek S_{DS} ve S_{D1} spektral ivme katsayıları,
- TBDY (2018) Tablo 4.3'e göre hareketli yük kütle katılım katsayısı (n),
- TBDY, (2018) Tablo 3.1'e göre bina önem katsayısı (I),
- TBDY, (2018) Tablo 4.1'e göre taşıyıcı sistem davranış katsayısı (R) ve dayanım fazlalığı katsayısı (D).

Sabit kabul edilen parametreler ve hesaplama sürecine ilişkin varsayımlar, ayrıntılı olarak Bölüm 3.3'te; tanımlanan tüm yüklerin analize nasıl aktarıldığına ilişkin açıklamalar ise Bölüm 3.5.3.2'de sunulmuştur.

3.2.1.6. Optimizasyon girdileri

Bu başlık altında, kullanıcıdan yalnızca popülasyon büyüklüğü ile maksimum iterasyon sayısını tanımlaması beklenmektedir. Optimizasyon sürecinde kullanılan algoritmalarından bazıları, performanslarını doğrudan etkileyen belirli kontrol parametrelerine – yani hiper parametrelere – duyarlıdır. Hiper parametreler, algoritmanın çalışma prensibini yönlendiren ve kullanıcı tarafından önceden belirlenen ayar değerleridir (örneğin, öğrenme katsayısı, sıcaklık düşüş oranı, çaprazlama oranı gibi). Bu nedenle, hiper parametre optimizasyonu süreci için ayrı bir çalışma gerçekleştirilmiş ve hiper parametre değerleri Bayes optimizasyonu yöntemiyle (Frazier, 2018) belirlenmiştir. Bayes optimizasyonu, yüksek maliyetli ve kapalı formda tanımlanamayan hedef fonksiyonların optimize edilmesinde kullanılan, kestirimsel bir model yardımıyla arama yapan bir yöntemdir. Her yeni deneme, mevcut bilgilere dayanarak seçilir ve bu sayede daha az hesaplama ile en uygun parametrelere ulaşılır. Çalışmada hiper parametre içeren algoritmalar için bu yöntemle belirlenen hiper parametreler sisteme sabitlenmiş ve kullanıcıdan ek giriş alınmaksızın kararlı bir çalışma sağlanmıştır.

Popülasyon büyüklüğü ve iterasyon sayısı gibi temel çalışma parametreleri de sabit bir değer olarak tanımlanabilir ve kullanıcıdan girdi alınmaksızın sistem tarafından atanabilirdi. Ancak, kullanıcıların bu değerlere ilişkin teknik bilgiye sahip olmama ihtimali göz önünde bulundurulsa da, yapı modelinde yer alan eleman sayısının artması durumunda sabit değerlerin yetersiz kalabileceği değerlendirilmiştir. Bu nedenle, sistemin farklı büyüklükteki yapı modellerine uyarlanabilirliğini korumak amacıyla bu iki parametrenin kullanıcı tarafından tanımlanması uygun görülmüştür.

3.2.1.7. Paralel işleme girdileri

Optimizasyon süreci kapsamında çok sayıda çözüm adayının birbirinden bağımsız analizlere tabi tutulması, paralel işlemeye uygun bir yapı oluşturmaktadır. Bu özelliğin etkin şekilde kullanılması, toplam hesaplama süresinin önemli ölçüde azaltılmasına olanak tanımaktadır. Bu nedenle, geliştirilen sistem paralel işlemeye uygun biçimde tasarlanmıştır.

Kullanıcıdan yalnızca iki parametrenin tanımlanması beklenmektedir. Bunlardan ilki, paralel işlemenin aktif hale getirilip getirilmeyeceğini belirleyen bir kontrol parametresidir. İkincisi ise kullanılacak işçi sayısını tanımlar. MATLAB ortamında işçi sayısı, fiziksel işlemci çekirdeği sayısından fazla olarak da tanımlanabilmekte ve belirli

bir sınıra kadar performans artışı sağlanabilmektedir. Ancak aşırı yüksek işçi sayıları, sistem kaynaklarının verimsiz kullanımına yol açarak performansın düşmesine neden olabilir. Bu nedenle, kullanıcıların kendi donanım yapılarına uygun optimum işçi sayısını deneme-yanılma yöntemiyle belirlemesi önerilmektedir.

3.2.2. Sistem girdileri

Yük kombinasyonları ve optimizasyon algoritmalarının hiper parametreleri, sistemin bütüncül işleyişine doğrudan etki eden sabit girdiler arasında yer almaktadır. Bu nedenle hem analiz güvenliğini hem de optimizasyon sürecinin istikrarını sağlamak amacıyla, bu parametreler yazılımın içerisine doğrudan gömülü olarak tanımlanmıştır.

Yük kombinasyonları, TBDY (2018) ve TS 498 (2021) standartlarına uygun olarak belirlenmiş ve yazılıma sabit matrisler biçiminde entegre edilmiştir. Kullanılan kombinasyonlar, sabit (G), hareketli (Q) ve iki yatay deprem etkisini (E_x ve E_y) temsil eden dört bileşenli katsayı dizilerinden oluşmaktadır. Kullanılmış olan yük kombinasyonlarına dair yük katsayıları Tablo 3.3'te sunulmuştur.

Tablo 3.3. Yazılımda kullanılan yük kombinasyonları

No	G	Q	E_x	E_y
1	1.40	1.60	0.00	0.00
2	1.00	1.00	1.00	0.30
3	1.00	1.00	1.00	-0.30
4	1.00	1.00	-1.00	0.30
5	1.00	1.00	-1.00	-0.30
6	1.00	1.00	0.30	1.00
7	1.00	1.00	-0.30	1.00
8	1.00	1.00	0.30	-1.00
9	1.00	1.00	-0.30	-1.00
10	0.90	0.00	1.00	0.30
11	0.90	0.00	1.00	-0.30
12	0.90	0.00	-1.00	0.30
13	0.90	0.00	-1.00	-0.30
14	0.90	0.00	0.30	1.00
15	0.90	0.00	-0.30	1.00
16	0.90	0.00	0.30	-1.00
17	0.90	0.00	-0.30	-1.00

Optimizasyon algoritmalarına ait hiper parametreler ise, doğrudan kullanıcı girdisine bırakılmamış, bunun yerine çalışmada ele alınan probleme özgü olarak Bayes

optimizasyon yöntemi ile ayrı bir ön analiz sürecinde belirlenmiştir. Böylece, her bir algoritma için en uygun parametre konfigürasyonu elde edilmiş ve bu değerler yazılımın içine sabit şekilde tanımlanarak, kullanıcıdan bağımsız kararlı bir optimizasyon süreci sağlanmıştır.

3.3. Kısıtlar ve Kabuller

Yazılımın henüz ilk sürümü olması nedeniyle, geliştirilen sistemde belirli teknik ve yapısal kabullerin yapılması zorunlu olmuştur. Bu kabuller, görüntü işleme altyapısının sınırlamaları ve yapısal analiz yaklaşımının sadeleştirilmiş biçimde modellenmesiyle doğrudan ilişkilidir. Bununla birlikte, getirilen bu sınırlamalar sistemin genel geçerliliğini ve işlevselliğini engellememekte; tersine, yapı tasarımı açısından anlamlı ve uygulanabilir çözümler üretebilecek bir otomasyon süreci ortaya koymaktadır.

Bu kapsamda geliştirilen sistem, geçerli tasarım çözümlerine yakınsayan, tutarlı ve tekrarlanabilir sonuçlar üretmektedir. Ancak mevcut haliyle, yalnızca belirli yapı türlerine ve sınırlı geometrik koşullara uygunluk göstermektedir. İlerleyen sürümlerde, hem görüntü işleme modülünün daha esnek hale getirilmesi hem de yapı modelinin farklı taşıyıcı sistem türlerini içerecek biçimde genişletilmesi hedeflenmektedir. Bu sayede, mevcut kısıtlamaların çoğu ortadan kaldırılarak, sistemin genel geçerliliği daha da artırılacaktır.

Aşağıda, bu sürümde sistemin işlevliliğini sağlayan başlıca kabuller ve getirilmiş olan teknik kısıtlamalar detaylı olarak sunulmuştur.

3.3.1. Görüntü işleme tabanlı kabuller

- Mimari planın, yalnızca duvar çizgilerini içermesi gerekmektedir; ölçülendirme, yazılar, merdiven, pencere, kapı gibi detaylar analiz sürecini bozacağından görüntüde yer almamalıdır.
- Görüntünün alt kısmında, kullanıcı tarafından belirlenmiş uzunlukta bir yatay referans çizgisi ile bu çizgiye karşılık gelen, uzunluk birimi belirtilmiş sayısal değer (örneğin “100 cm”) mutlaka yer almalıdır.
- Dış cephe geometrisi yalnızca dikdörtgen formda olmalıdır. Eğik veya kırıklı konturlar desteklenmemektedir.
- Bütün duvar çizgileri, referans çizgisi, ölçü rakamı ve birimi aynı renge sahip olmalıdır.

- Duvar çizgileri planda sadece yatay veya dikey doğrultuda olmalıdır. Bu versiyonda eğimli (diyagonal) duvarlar algılanamamaktadır.
- Planlarda kat çıkması (konsol çıkma) bulunmamalıdır.

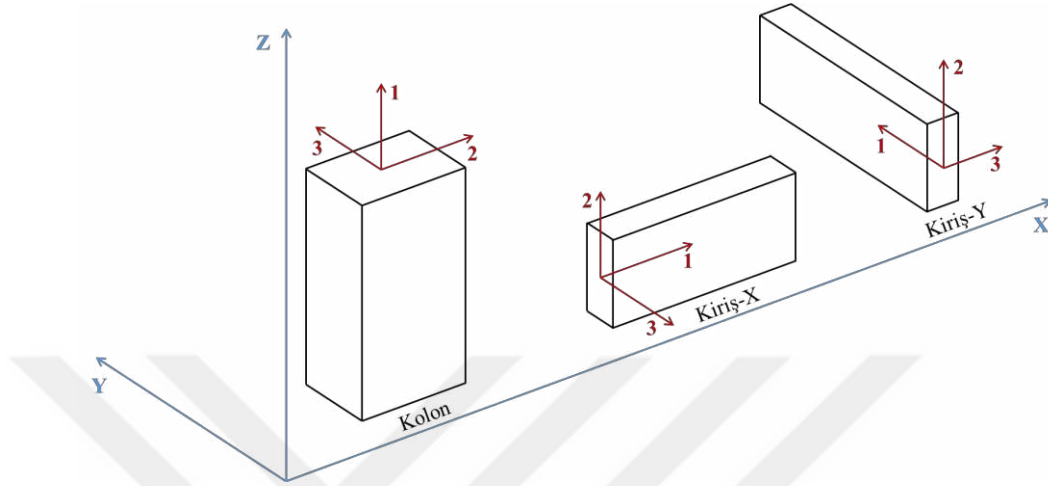
3.3.2. Yapı modeli ve tasarım kısıtlamaları

- Henüz bu versiyonda perde elemanlara ilişkin yönetmelik kontrolleri dahil edilmemiştir. Bu nedenle taşıyıcı sistem yalnızca kolon ve kiriş elemanlarıyla temsil edilmektedir.
- Birbirine belirli bir mesafeden daha yakın konumlanan kolonlar, birleşik kolon sistemi olarak ele alınmakta ve tek bir kolon şeklinde modellenmektedir. Ancak birleşme sonucunda oluşan kesit boyutları, perde eleman tanımı kapsamına girecek bir orana sahip olduğunda, kısa kenar boyutu sınır değerinin üstünde kalacak şekilde revize edilmektedir.
- Döşeme tasarımı henüz yapılmamaktadır. Sonlu elemanlar modeli bir çerçeve sistem olarak kurulmakta; döşeme yükleri doğrudan kiriş elemanlarına eşdeğer yük olarak aktarılmaktadır.
- Kolon–kiriş birleşim bölgesine ilişkin herhangi bir detaylı tasarım veya kontrol bu aşamada yapılmamaktadır.
- Yapı modeli oluşturulurken, plan düzleminde yalnızca dikdörtgen kesitli kolon ve kiriş elemanlarına izin verilmektedir. Sistem, bu geometrik sınırlamayı esas alarak gerekli kesit parametrelerini otomatik biçimde oluşturmaktadır.
- Program kapsamında yalnızca üst yapı elemanlarının tasarımı dikkate alınmakta; temel sistemi modellenmemekte ve temel tasarımına ilişkin herhangi bir analiz veya boyutlandırma işlemi gerçekleştirilmemekte, kolonların alt uçları ise ankastre mesnet koşulu ile temsil edilmektedir.
- Eleman bazında burulma tasarımı yapılmamaktadır.
- Mevcut aşamada, kodlamada pratikliği nedeniyle görelî kat ötelemesi kontrolü %2 sınır değeri esas alınarak uygulanmıştır.
- Narinlik etkisi bu aşamada dikkate alınmamıştır.

3.3.3. Sonlu elemanlar modeline ilişkin teknik kabuller

- Kolonların yerel eksenleri: Lokal 1, 2 ve 3 eksenleri sırasıyla global +Z, +X ve +Y yönlerine karşılık gelecek şekilde kodlanmıştır (Şekil 3.3).

- Kirişlerin yerel eksenleri: Lokal 1 eksen global +X ya da +Y yönünde olacak şekilde tanımlanırken, lokal 2 eksen her zaman +Z yönünde kabul edilmiştir (Şekil 3.3).



Şekil 3.3. Kolon ve kirişlerdeki lokal eksen yönü tanımlarının global eksenle gösterimi

- Yapının tüm taşıyıcı elemanlarında aynı beton sınıfı ve aynı donatı çeliği sınıfı kullanılmaktadır. Farklı elemanlarda farklı malzeme kullanımı desteklenmemektedir.
- Kullanıcı girdisi olan paspayı değeri, tüm yapı elemanlarında aynı kabul edilerek donatı tasarımı hesabında kullanılmaktadır.
- Yapı modellemesinde her katta rijit diyafram kabulü yapılmaktadır.
- Her kat için hesaplanan toplam kütle, ilgili kattaki düğüm noktalarına eşit şekilde dağıtılarak her düğüme kütle ataması yapılmaktadır.

3.3.4. Deprem yükü hesabına ilişkin kabuller

- Deprem yükü hesabı eşdeğer deprem yükü yöntemine dayanmaktadır.
- Bu versiyonda, deprem etkisi altındaki davranışın otomasyonuna tam olarak entegre edilmemesi nedeniyle, yapı konut tipi olarak düşünülmüş ve TBDY, (2018) Tablo 3.1'e göre Bina Kullanım Sınıfı (BKS) değeri 3 olarak sabitlenmiştir.
- Optimizasyon sürecinde oluşan modellerdeki kolon yerleşimleri düzensizlik içerebildiğinden, burulma düzensizliği oluşabileceği öngörülmüş ve buna uygun olarak TBDY (2018) Tablo 4.4'e göre Bina Yükseklik Sınıfı (BYS) sınırlandırılmıştır. Yazılım, modelleme aşamasında bina yüksekliği, kat adedi ya

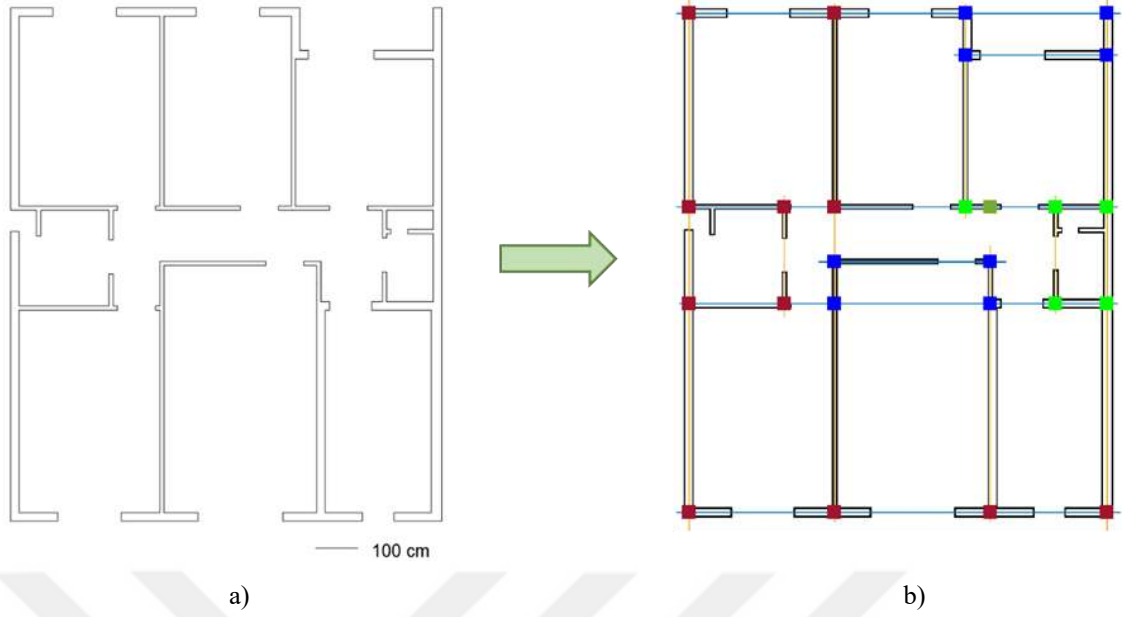
da eleman sayısına ilişkin herhangi bir sınırlama getirmeyecek biçimde tasarlanmıştır. Bununla birlikte, bu sürümde 28 metreden yüksek binaların modellenmesi yönetmeliğe aykırı sonuçlar doğurabilir.

3.4. Görüntü İşleme Modülü

Geleneksel yapı tasarım süreçlerinde taşıyıcı sistemin oluşturulması, genellikle mimari planların manuel olarak incelenmesiyle gerçekleştirilmekte; bu da sürecin hem zaman alıcı hem de öznel kararlara dayalı olmasına neden olmaktadır. Bu çalışmada geliştirilen sistemde ise yapı tasarımı süreci, yalnızca mimari kat planının dijital görüntüsünün analizine dayanan, insan müdahalesine ihtiyaç duymayan otomatik bir yaklaşımla başlatılmaktadır.

Görüntü işleme tekniklerinin yardımıyla mimari planlar sistematik biçimde değerlendirilmekte ve taşıyıcı sistemin oluşturulmasında temel teşkil edecek olan aks sistemi çıkarılmaktadır. Akslar yalnızca taşıyıcı kolon yerleşimlerinin potansiyel konumlarını belirlemekle kalmamakta; aynı zamanda kiriş sistemini sabit şekilde tanımlayan yapısal bir iskelet oluşturmaktadır. Çalışmada aks düzeni bir kez belirlendikten sonra sabitlenmekte, kiriş yerleşimi bu düzene göre tanımlanmakta ve tüm modelleme boyunca sabit kalmaktadır. Yalnızca kolon yerleşimi, daha sonra yürütülecek olan optimizasyon sürecinde değişken bir parametre olarak ele alınmaktadır.

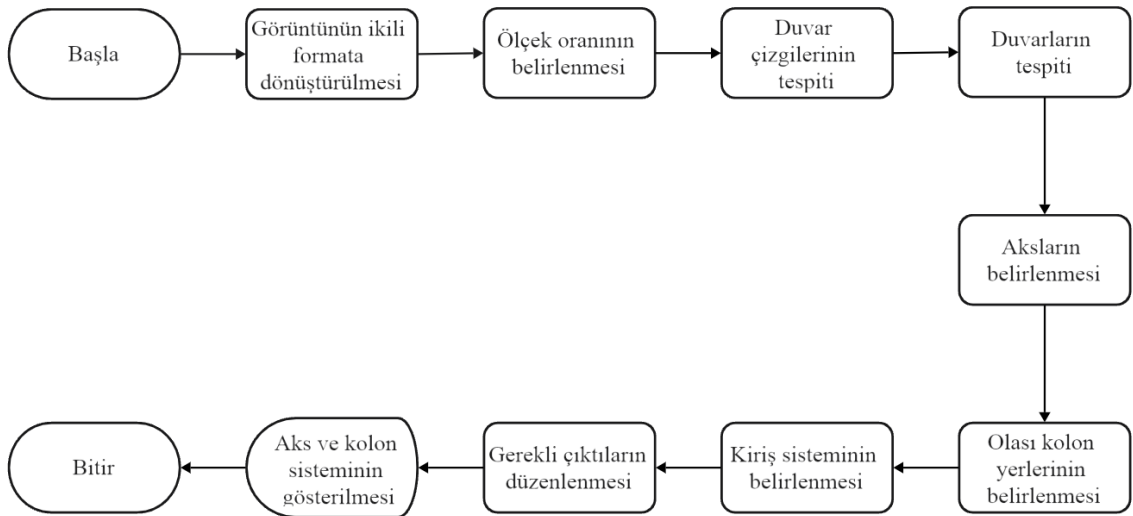
Kolon topolojisinin belirlenmesi sürecinde, çıkarılan yatay ve dikey aksların kesişim noktaları, olası kolon yerleşimleri olarak sistematik biçimde tanımlanmaktadır. Bu yöntem, yalnızca geometrik uygunluk açısından değil, aynı zamanda yapısal anlamlılık bakımından da tutarlı bir yerleşim ağı sunmakta; böylece kolon konfigürasyonunun optimizasyonu için anlamlı, sınırlı ve uygulanabilir bir arama uzayı oluşturulmaktadır. Bu durum, optimizasyon algoritmasının çözüm uzayını daha verimli taramasına ve fiziksel olarak uygulanabilir çözümlere yönelmesini kolaylaştırarak genel hesaplama etkinliğini artırmaktadır.



Şekil 3.4. a) Kullanıcı tarafından sağlanan örnek mimari plan, b) Görüntü işleme sonucunda elde edilen akslar ve olası kolon yerleri

Bu bölümde, mimari plan verisinin dijital olarak nasıl işlendiği, görüntülerin hangi aşamalardan geçirildiği ve bu süreç sonucunda aks sisteminin ve potansiyel kolon yerleşimlerinin nasıl belirlendiği ayrıntılı biçimde açıklanacaktır. Yazılımın görüntü işleme modülünde gerçekleştirilen işlemler sonucunda elde edilen giriş ve çıkış görüntülerine ilişkin bir örnek, Şekil 3.4’te sunulmaktadır.

Yazılımın görüntü işleme modülünde gerçekleştirilen aşamalara dair akış diyagramı Şekil 3.5’te sunulmuştur.



Şekil 3.5. Görüntü işleme modülü akış diyagramı

3.4.1. Görüntünün ikili formata dönüştürülmesi

Kullanıcıdan alınan ekran görüntüleri yaygın olarak JPEG, BMP ve PNG formatlarında olabilmektedir. Ancak bu çalışma kapsamında, görüntü işleme sürecinde PNG formatı tercih edilmiştir. Bu tercihin temel nedenleri arasında, kayıpsız sıkıştırma özelliği sayesinde veri kaybı olmadan görüntü kalitesinin korunması, alfa saydamlık desteği, geniş renk derinliği ve genellikle düşük dosya boyutu yer almaktadır. Ayrıca PNG formatı, çoğu ekran görüntüsü alma aracı tarafından doğal olarak desteklenmektedir. Bu özellikler sayesinde, görüntü verisi üzerinde gerçekleştirilecek sistematik analizler için güvenilir ve tutarlı bir temel sunmaktadır (Miano, 1999).

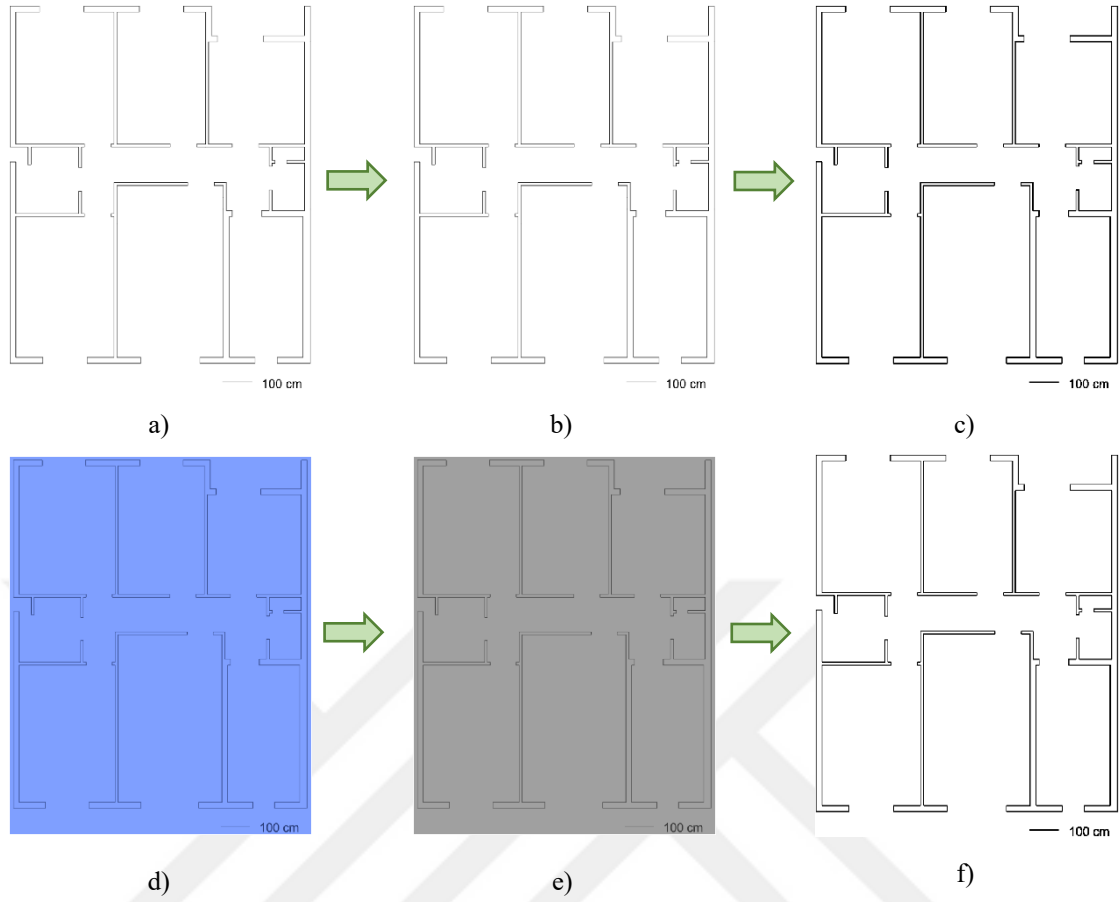
Alınan mimari plan görüntüsü, işlemeye başlanmadan önce MATLAB ortamında tanımlı bir fonksiyon aracılığıyla gri seviye formata dönüştürülmektedir. Bu dönüşüm sırasında kullanılan fonksiyon, ITU-R BT.601-7 ([http-4](http://4)) standardında tanımlanan ağırlıklı ortalama yöntemine dayalı dönüşüm formülünü (Denklem 3.1) esas almaktadır.

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \quad (3.1)$$

Formülde yer alan parametreler, her pikseldeki kırmızı (R), yeşil (G) ve mavi (B) renk bileşenlerini ifade etmekte olup, bu bileşenlerin ağırlıklı ortalaması alınarak ilgili pikselin gri seviye değeri (Y) hesaplanmaktadır.

Yazılımın görüntü işleme sürecinde, mimari planın ikili görüntüye dönüştürülmesi temel bir adımdır. Çünkü sonraki tüm analiz işlemleri bu ikili görüntü üzerinden yürütülmektedir. Bu nedenle, bu adımın güvenilir ve kararlı çalışması büyük önem taşımaktadır.

Mimari planlarda genellikle yüksek kontrastlı çizim renkleri kullanıldığından ve Bölüm 3.3.1’de belirtilen görsel kısıtlamalara uyulduğunda, gri seviye görüntü yalnızca arka plan ve çizim olmak üzere iki yoğunluk bölgesi içermektedir. Bu durumu avantajlı şekilde değerlendiren sistemde, kullanıcıdan eşik değeri alınmasına gerek kalmadan, otomatik bir eşikleme algoritması geliştirilmiştir. Gri seviye dağılım analizine dayalı bu algoritma, her görüntüye özgü en uygun eşik değerini belirleyerek kararlı bir binarizasyon işlemi gerçekleştirir. Bu sayede, farklı plan renklerine karşı dayanıklı ve net bir duvar planı elde edilmektedir. Farklı çizim renklerine sahip iki örnek görüntünün, sırasıyla gri seviye formata ve ardından ikili formata dönüştürülmesiyle elde edilen sonuçlar Şekil 3.6’da sunulmuştur.



Şekil 3.6. a) Orjinal görüntü-1, b) Gri seviye görüntü-1, c) İkili format görüntü-1, d) Orjinal görüntü-2, e) Gri seviye görüntü-2, f) İkili format görüntü-2

Gri seviye görüntüden ikili formata çevirimde kullanılmak üzere oluşturulmuş algoritma Denklem 3.2’de sunulmuştur.

$$B(x, y) = \begin{cases} 1, & \text{eğer } \bar{I} > T \text{ ve } \bar{I} \leq I(x, y) \leq I_{max} \\ 1, & \text{eğer } \bar{I} \leq T \text{ ve } I_{min} \leq I(x, y) \leq \bar{I} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (3.2)$$

Burada,

- I , gri seviye görüntü verisini,
- I_{min} , görüntüdeki minimum gri değeri,
- I_{max} , görüntüdeki maksimum gri değeri,
- $\bar{I}$, görüntünün ortalama değerini,
- T , referans eşik değerini (çalışmada 128 olarak alınmıştır),
- B , ikili format görüntüyü ifade etmektedir.

3.4.2. Ölçek oranının belirlenmesi

Mimari plan görüntüsünün analizine başlanmadan önce, uzunluk ölçümünün sağlıklı biçimde gerçekleştirilebilmesi için görüntüye ait ölçek oranının belirlenmesi gerekmektedir. Bu amaçla, mimari planın en alt kısmında, belirli bir uzunlukta yatay bir referans çizgisi ile bu çizgiye karşılık gelen uzunluğun sayısal değeri ve birimi birlikte yer almalıdır. Referans çizgi haricinde ölçü içermeyen bu görüntüde, yalnızca mimari duvar çizgileri bulunmaktadır. Görüntü işleme sürecinde elde edilen ikili görüntü matrisi, herhangi bir perspektif etkisinin bulunmaması nedeniyle, kare piksellerden oluşan düzenli bir ızgara sistemi olarak değerlendirilmektedir. Bu çerçevede, her pikselin yatay ve düşey yönde temsil ettiği gerçek uzunluk miktarının belirlenebilmesi için bir ölçek katsayısı hesaplanmaktadır.

Plan ölçeği olarak adlandırılan bu katsayı, referans uzunluk değeri ile referans çizginin görüntü üzerindeki piksel uzunluğu arasındaki orana bağlıdır. Katsayının hesaplanması Denklem 3.3 ile ifade edilmektedir.

$$Plan\ ölçeği = \frac{Referans\ ölçü}{Referans\ çizginin\ piksel\ uzunluğu} \quad (3.3)$$

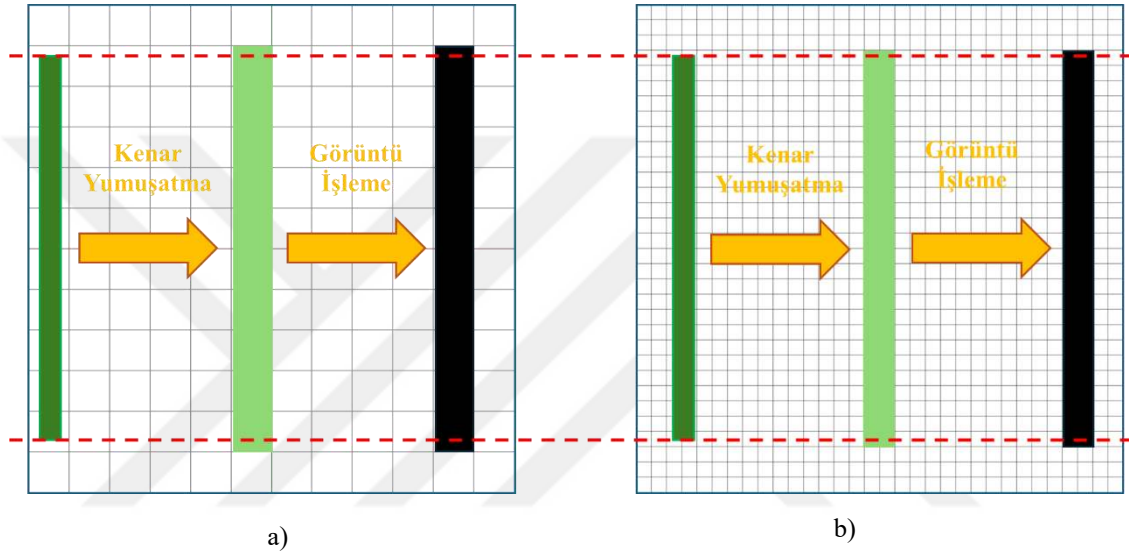
Sayısal değerın otomatik olarak okunabilmesi için görüntüye OCR (Optical Character Recognition) algoritması uygulanmakta ve bu sayede metin içeriği görüntüden çıkarılmaktadır.

Plan ölçeğinin doğruluğu üzerinde etkili olan birkaç temel unsur bulunmaktadır. Bunlardan ilki, görüntüye kaynaklık eden monitörün çözünürlüğüdür. Aynı fiziksel boyuttaki (örneğin 27 inç) ancak farklı çözünürlüklerdeki iki monitörden alınan ekran görüntüleri, farklı piksel yoğunlukları içerecektir. Daha yüksek çözünürlüklü monitörden alınan görüntü, aynı fiziksel alanı daha fazla piksel ile temsil ettiğinden, referans uzunluğun piksel cinsinden daha isabetli tespit edilmesini sağlar. Bu durum, ölçekleme katsayısının doğruluğunu artırmakta ve nihai uzunluk hesaplamasındaki hata payını azaltmaktadır.

Bir diğer hata kaynağı ise mimari çizimlerde sıkça karşılaşılan, çizgi kenarlarının bir pikselin tamamını doldurmaması durumudur. Bu gibi durumlarda, ekran kartları tarafından uygulanan kenar yumuşatma sistemleri devreye girmektedir. Kenar yumuşatma teknikleri, bir pikselin birden fazla renk bilgisi taşınması halinde, komşu piksellerle olan geçişleri yumuşatarak daha doğal ve kesintisiz bir görüntü sunmaktadır.

Ancak bu işlemler, piksel düzeyindeki renk değerlerini etkilediğinden, gri seviye ve ardından gelen ikili formata dönüşüm aşamalarında, çizgi kenarlarında çentiklenmeler veya fazladan pikselleşme gibi sapmalara neden olabilmektedir. Bu durum, duvar çizgilerinin sınırlarının tam olarak belirlenmesini güçleştirebilmekte ve dolaylı olarak uzunluk hesaplamalarında hata yaratabilmektedir.

Bahsi geçen her iki hata olasılığını da somutlaştırmak amacıyla hazırlanan şematik bir örnek Şekil 3.7’de sunulmaktadır.



Şekil 3.7. Örnek bir çizginin, a) Düşük çözünürlüklü monitörden alınan, b) Yüksek çözünürlüklü monitörden alınan sonuçları

Yapılan çeşitli denemelerde, farklı çözünürlüklerde alınan mimari plan görüntüleri üzerinde bu tür etkiler analiz edilmiş ve elde edilen hata paylarının yapı tasarımındaki sonuçları anlamlı şekilde etkilemediği gözlemlenmiştir. Ölçekleme sürecinde izlenen bu yaklaşım, sistemin otomasyon kabiliyetini artırmakta ve kullanıcı müdahalesine olan ihtiyacı ortadan kaldırmaktadır.

3.4.3. Duvar çizgilerinin tespiti

Duvar çizgilerinin tespitine başlamadan önce, görüntüde yalnızca çizgilerin kalmasını sağlamak amacıyla referans ölçüsünün görüntüden kaldırılması gerekmektedir. Bunun için öncelikle, plan üzerindeki çizgilerle referans ölçüsünün ortak renk değeri belirlenmelidir. Mimari plan çizimlerinde arka planın, çizgilere kıyasla çok daha fazla

piksel içerdiği gerçeğinden yararlanılarak, ikili formata dönüştürülmüş görüntüde çizgi rengine karşılık gelen piksel değeri Denklem 3.4'te verilen bağıntı ile belirlenmektedir.

$$r_{\text{çizgi}} = \begin{cases} 1, & \text{eğer } n_1 \leq n_0 \\ 0, & \text{diğer durumlarda} \end{cases} \quad (3.4)$$

Burada,

- n_1 , 1 değerine sahip piksel sayısını,
- n_0 , 0 değerine sahip piksel sayısını,
- $n_0 + n_1 = N$, ikili format görüntüdeki toplam piksel sayısını ifade etmektedir.

Sonrasında, OCR fonksiyonunun özelliklerinden yararlanılarak referans ölçünün bulunduğu bölge tespit edilmekte; bu bölgedeki piksellerin renk değeri, $1 - r_{\text{çizgi}}$ işlemi ile terslenerek değiştirilmekte ve böylece referans ölçü yazısı görüntüden etkin bir şekilde silinmektedir.

Referans ölçü yazısının görüntüden kaldırılmasının ardından, çizgilerin belirlenmesi sürecine geçilmektedir. Bu aşamada, görüntüdeki çizgilerin sistematik biçimde tespit edilmesi amacıyla hazır yazılım fonksiyonları değerlendirilmiştir. Ancak MATLAB ortamında bulunan ve çizgi saptamaya yönelik standart fonksiyonlar, mimari planlardaki çizgilerin yapısal çeşitliliği nedeniyle yeterince doğru ve tekrarlanabilir sonuçlar vermemiştir. Bu nedenle, çalışmanın ihtiyaçlarına özel olarak geliştirilmiş, doğrudan ikili görüntü üzerinde çalışan bir çizgi belirleme algoritmasının oluşturulması gerekli olmuştur.

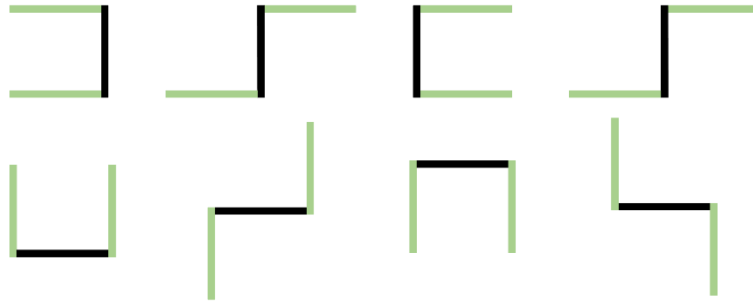
Geliştirilen algoritmada, iç içe döngüler kullanılarak görüntüdeki her piksel taranmakta ve yalnızca $r_{\text{çizgi}}$ değerine sahip olan pikseller dikkate alınmaktadır. İlk aşamada, yatay eksende arka arkaya gelen çizgi piksellerinin başlangıç ve bitiş koordinatları belirlenmekte ve her biri ayrı birer yatay çizgi olarak kaydedilmektedir. Bu nedenle, tek bir çizgiye ait piksel kümesi başlangıçta birden fazla yatay çizgi parçası olarak tanımlanmaktadır. Benzer şekilde, dikey doğrultuda yer alan çizgiler de bu yatay tarama mantığı nedeniyle ardışık satırlarda konumlanmış kısa yatay çizgiler şeklinde tespit edilmektedir. Dolayısıyla her iki doğrultuda da birleştirme işleminin gerçekleştirilmesi gerekmektedir.

Çizgilerin birleştirilmesi aşamasında, tanımlanmış tüm çizgiler birbirleriyle başlangıç ve bitiş koordinatları açısından kıyaslanmakta; birbirine yakın ve ardışık satırlarda yer alan çizgiler aynı çizgi grubu altında birleştirilmektedir. Ancak görüntüdeki

köşe noktalarında oluşan renk geçişleri, bu bölgelerde çizgi sürekliliğinin bozulmasına ve çentiklerin oluşmasına yol açmaktadır. Bu durum, aynı çizgiye ait parçalarda küçük koordinat farklarının ortaya çıkmasına neden olmaktadır. Bu sorunun üstesinden gelmek amacıyla, kullanıcı tarafından belirlenen bir kenar birleştirme hassasiyeti (Bölüm 3.2.1.1), başlangıç ve bitiş koordinatları için bir tolerans değeri olarak kullanılmakta ve bu tolerans aralığı içinde kalan çizgiler aynı grup altında tanımlanmaktadır.

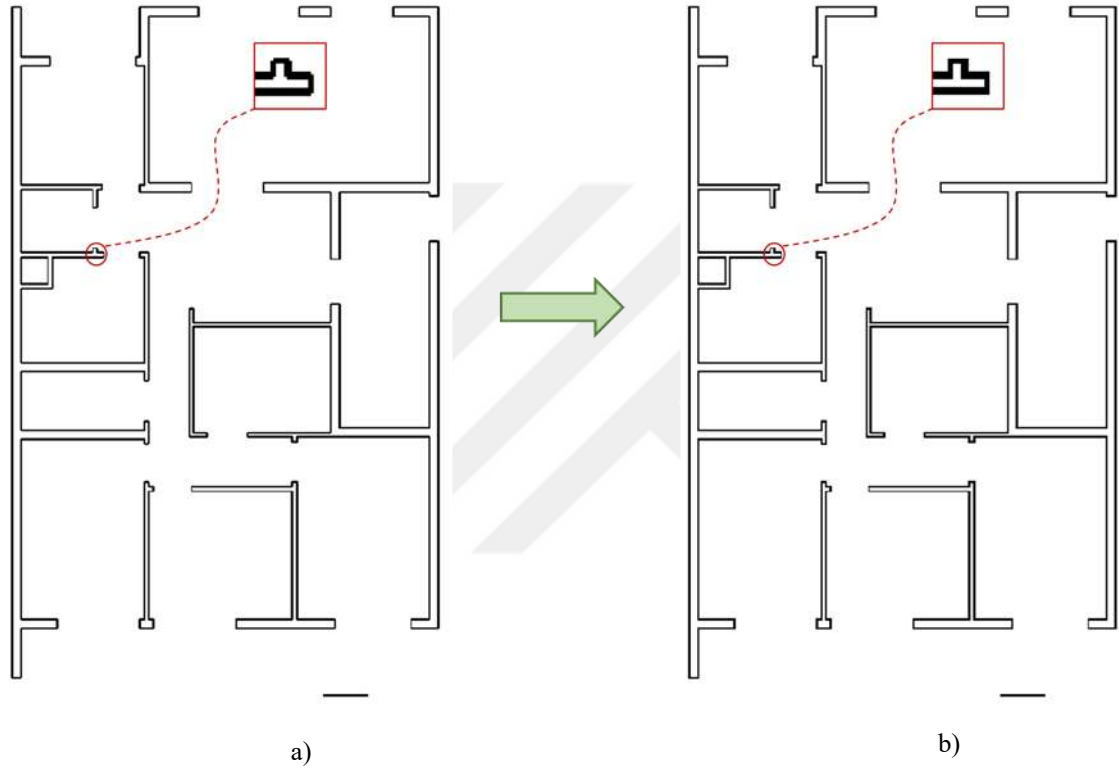
Yatay ve düşey çizgilerin farklı parametrelerde ayrı ayrı kaydedilmesi, yazılımın ilerleyen aşamalarında yapılacak işlemler açısından önemli bir kolaylık sağlamaktadır. Bu nedenle, tanımlanan her çizgi, piksel uzunluğu yalnızca yatay veya yalnızca düşey yöndeki uzantısına göre değerlendirilmekte ve ilgili parametreye atanarak sınıflandırılmaktadır. Ancak mimari planlarda, yatay ve düşey uzunlukları birbirine eşit olan kısa çizgilere sıklıkla rastlanabilmektedir. Bu tür çizgiler, yalnızca uzunluk kriteriyle sınıflandırıldığında yanlış bir gruba atanma riski taşımaktadır. Bu sorunu gidermek amacıyla bir komşuluk algoritması geliştirilmiştir. Algoritmanın temel mantığı, tüm duvarların kapalı birer poligon yapısına sahip olduğu ve sadece yatay ve düşey çizgilerden oluştuğu varsayımına (Bölüm 3.3.1) dayanmaktadır. Bu çerçevede, bir yatay çizginin her zaman iki düşey çizgi komşusuna, bir düşey çizginin ise iki yatay çizgi komşusuna sahip olması gerekliliği esas alınmaktadır. Belirtilen bu kurallara göre geliştirilen komşuluk algoritması sayesinde, tüm çizgiler yeniden denetlenmekte; yanlış sınıflandırılmış kısa çizgiler uygun parametreye aktarılmakta ve böylece veri yapısında tutarlılık sağlanmaktadır.

Düşey ve yatay çizgilere ait olası komşuluk durumlarını gösteren şematik çizimler Şekil 3.8’de sunulmuş olup, geliştirilen algoritmada bu geometrik varyasyonların tamamı dikkate alınarak kontrol mekanizması oluşturulmuştur.



Şekil 3.8. Düşey (üst sıra) ve yatay (alt sıra) çizgilere ait olası komşuluk varyasyonları

Bir sonraki adımda, çentiklerin giderilmesi ve çizgi sürekliliğinin sağlanması amacıyla her çizgi grubunda yer alan parçaların koordinatları yeniden düzenlenmektedir. Bu kapsamda, aynı gruptaki tüm çizgiler için başlangıç koordinatlarının en küçük, bitiş koordinatlarının ise en büyük değeri esas alınmakta; böylece her grup, sürekli ve düzgün bir çizgi haline getirilmektedir. Şekil 3.9’da örnek bir görüntünün ikili formata ilk çevrilmiş hali ve çentiklerin düzeltilmiş hali gösterilmektedir.

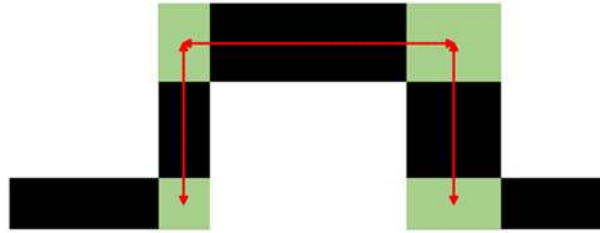


Şekil 3.9. İkili formata çevrilmiş görüntünün, a) İlk hali, b) Çentik düzeltilmesi sonrası

Son olarak, uzunluklarının olabildiğince düşük hata payı ile belirlenebilmesi amacıyla özel bir algoritma geliştirilmiştir. CAD tabanlı mimari planlarda yer alan çizgiler, yalnızca çizim kurallarına bağlı olarak farklı kalınlıklarda gösterilen, gerçekte fiziksel kalınlığı olmayan tek boyutlu elemanlardır. Ancak bu çalışma kapsamında kullanılan veriler, ekran alıntısı yoluyla elde edilen görüntüler, ikili formata dönüştürülmüş görüntü matrislerine dayandığı için, çizgiler piksel temsiline sahiptir ve kalınlıkları değişkenlik göstermektedir. Bir çizginin piksel cinsinden uzunluğu doğrudan plan ölçeğine bağlı olarak fiziksel uzunluğa çevrildiğinden, bu kalınlık farkları ölçüm doğruluğunu doğrudan etkilemektedir.

Çizgi tespiti algoritması, ilk aşamada yalnızca satır bazlı tarama yapacak şekilde yapılandırıldığından, başlangıçta yalnızca yatay çizgiler belirlenmektedir. Bu nedenle, çizgilerin kesişim noktalarında yer alan köşe pikseller yalnızca yatay çizgilerin veri yapısında bulunmaktadır. Düşey çizgiler ise daha sonra yatay çizgiler içerisinden ayrıştırıldıkları için, başlangıçta köşe bölgelerine ait piksel verisini içermemektedirler. Bu durum, düşey çizgilerin gerçek uzunluklarının eksik hesaplanmasına neden olabileceğinden, köşe bölgelerinin belirlenmesi ve bu etkiyi minimize edecek bir yöntem geliştirilmesi gerekliliğini doğurmuştur.

Bu amaçla, öncelikle komşuluk matrisi kullanılarak yatay ve düşey çizgilerin kesişim noktalarında ortak olması gereken köşe alanları tespit edilmektedir. Ardından, belirlenen bu köşe bölgeleri düşey çizgilere ait koordinat listelerine eklenmekte ve eksik konum bilgileri tamamlanmaktadır. Köşe alanlarının uzunluk hesaplamalarına dahil edilip edilmemesi durumlarına göre yapılan karşılaştırmalar sonucunda, en düşük hata payını veren yöntemin, ölçümün köşe bölgelerinin orta noktalarından alınması olduğu belirlenmiştir (Şekil 3.10). Buna göre, her bir çizginin uzunluğu, komşu çizgilerle paylaşılan köşe bölgelerinin orta noktalarına göre hesaplanmakta; elde edilen değer plan ölçeğiyle çarpılarak gerçek uzunluğa dönüştürülmekte ve ilgili parametre matrisine kaydedilmektedir.



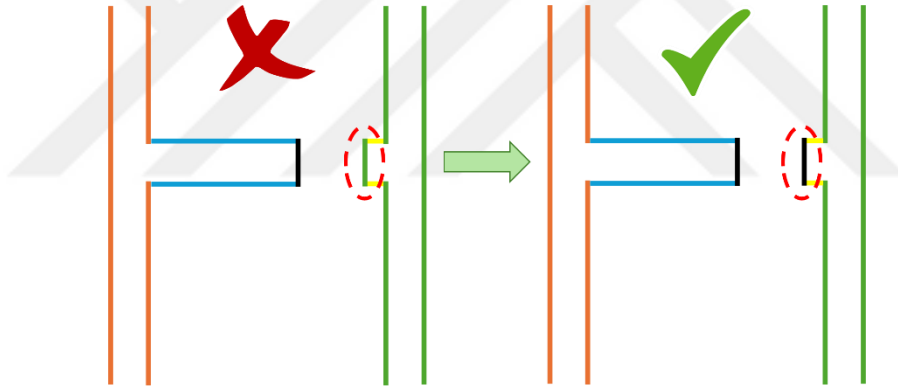
Şekil 3.10. Köşe bölgelerinin belirlenmesi ve uzunluk hesaplamasında kullanılan orta nokta ölçüm stratejisi

3.4.4. Duvarların tespiti

Bir önceki adımda görüntü üzerinden tespit edilen yatay ve düşey çizgiler, bu aşamada mimari plandaki duvarların tanımlanmasında kullanılmaktadır. Çalışma kapsamında her duvar, birbirine paralel ve belirli bir tolerans aralığında konumlanmış iki çizginin oluşturduğu kapalı bir alan olarak tanımlanmıştır. Bu tolerans aralığı, özellikle uzun bir çizginin kendisine paralel birden fazla çizgiyle eşleşmesi gereken durumlar için gereklidir. Çünkü yalnızca “en yakın paralel çizgi” kriterine göre yapılan eşleşmelerde, o

duvara ait olan diğer çizgilerin sistem tarafından tanınmadığı gözlemlenmiştir. Yapılan denemeler ve hata payı değerlendirmeleri sonucunda, çizgilerin birbirine dik uzaklıklarının 9 ila 28 cm arasında olması koşulu, bu tolerans değeri için uygun bir aralık olarak belirlenmiştir.

Ancak, bu yaklaşım uzun duvarlar için başarılı sonuçlar verirken, bu duvarlara kapı boşluğu amacıyla eklenmiş kısa çıkıntı duvarların tespitinde yetersiz kalmaktadır. Bu sorunun giderilmesi amacıyla komşuluk matrisi kullanılmış ve kısa çıkıntıların daha geniş bağlamda değerlendirilmesi sağlanmıştır. Şekil 3.11’de şematik olarak gösterilen örnek duvarlarda, turuncu ve mavi çizgilerle temsil edilen düşey ve yatay duvarlar tolerans kısıtıyla doğru şekilde tespit edilebilmektedir. Ancak aynı şekilde yeşil ve sarı çizgilerin temsil ettiği duvarlarda, yatay duvar olarak tanımlanması gereken sarı çizgiler, sistem tarafından eş zamanlı olarak düşey bir duvar segmenti şeklinde hatalı biçimde tanımlanabilmektedir.



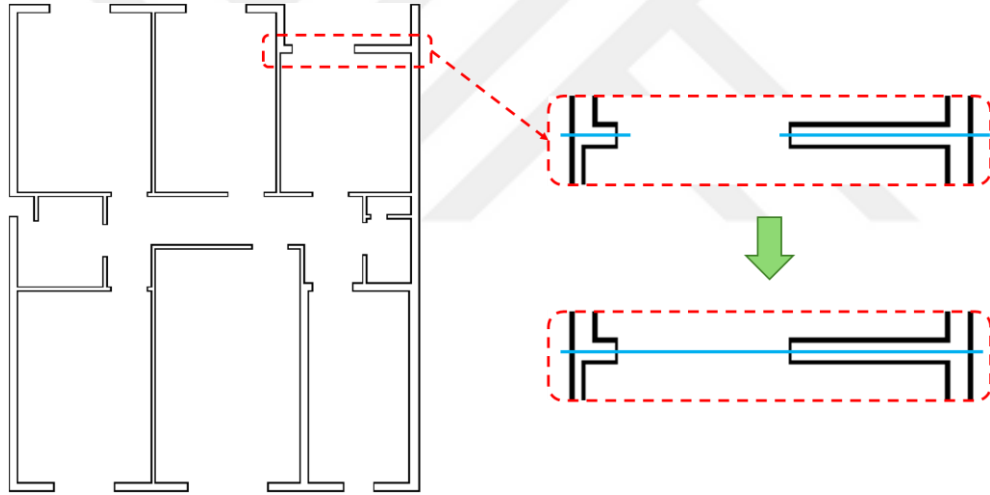
Şekil 3.11. Duvarların tespitinde komşuluk matrisi kullanılarak yapılan iyileştirmenin şematik gösterimi

Bu doğrultuda yatay çizgiler kendi aralarında, düşey çizgiler ise yine kendi yönlerinde ikili olarak değerlendirilmektedir. Her çizgi, konumu ve yönü dikkate alınarak potansiyel eşleşme adaylarıyla karşılaştırılmakta; başlangıç ve bitiş koordinatları belirli bir mesafe eşiği içinde kalan çizgi çiftleri, aynı duvara ait sınırlar olarak gruplanmaktadır. Algoritmanın temel varsayımı, duvarların yalnızca yatay ve düşey doğrultularda çizildiği ve her duvarın iki paralel çizgi ile temsil edildiğidir. Bu çerçevede her duvar, bir yatay çizgi çifti ya da bir düşey çizgi çifti ile tanımlanmakta ve sistematik biçimde numaralandırılarak duvar matrisine kaydedilmektedir.

3.4.5. Aksların belirlenmesi

Mimari projelerde taşıyıcı sistemlerin düzenlenmesinde temel referans elemanlardan biri olan aks çizgileri, yapı elemanlarının konumlandırılması ve hizalanmasında kritik rol oynamaktadır. Bu nedenle, duvar çizgilerinin tespitinden sonra, bu duvarlara karşılık gelen aks çizgilerinin oluşturulması gerekmektedir. Aksların doğru ve sistematik biçimde tanımlanması, sonraki yapısal modelleme ve analiz adımlarının doğruluğu açısından büyük önem taşımaktadır.

Çalışma kapsamında akslar, duvar kalınlıklarının orta noktasından geçen çizgiler olarak tanımlanmıştır. Bu amaçla geliştirilen algoritma, yatay ve düşey yönlerdeki duvar bilgilerini kullanarak her bir duvarın orta ekseninden geçen birer aks çizgisi oluşturmaktadır. Eğer aynı doğrultuda ve aynı hizada birden fazla duvar mevcutsa, bu duvarların ortasından geçen akslar tekil bir çizgi olarak ifade edilmektedir (Şekil 3.12).

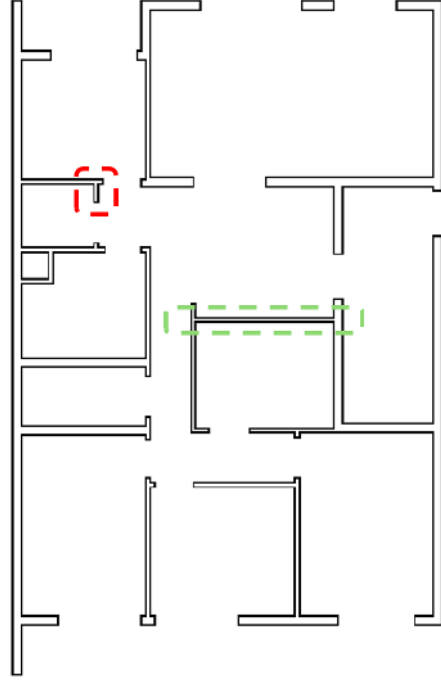


Şekil 3.12. Aks çizgilerinin belirlenmesi

Ancak bazı durumlarda, belirli bir doğrultuda yalnızca tek ve uzun bir duvarın bulunduğu durumlar da gözlemlenmiştir. Bu gibi durumlar, başlangıçta çoklu duvar varsayımına dayalı kurgulanan aks oluşturma algoritması açısından istisna teşkil etmektedir (Şekil 3.13, yeşil işaretli bölge). Bu nedenle algoritma, aynı hizada ve aynı doğrultuda herhangi bir duvarın varlığı durumunda oraya aks tanımlanacak şekilde yeniden yapılandırılmıştır.

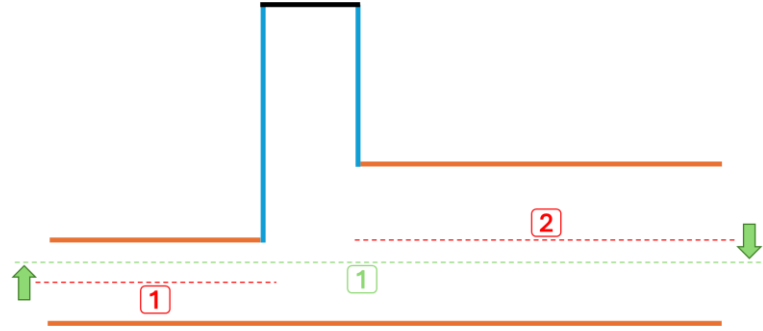
Bu revizyonla birlikte ortaya çıkan yeni bir durum, kapı açıklıkları gibi noktalarda bulunan kısa duvar parçalarının, tek başlarına dahi olsa bir aks tanımını tetikleyebilmesidir (Şekil 3.13, kırmızı işaretli bölge). Bu durum, yapısal analizde

gereksiz karmaşıya yol açabileceği için, aks oluşturulması için gerekli olan minimum duvar uzunluğuna ilişkin bir eşik tanımlanması gerekmiştir.



Şekil 3.13. Aks çizgilerinin oluşturulmasında karşılaşılan sorunlu (kırmızı) ve doğru (yeşil) bölgelerin şematik gösterimi

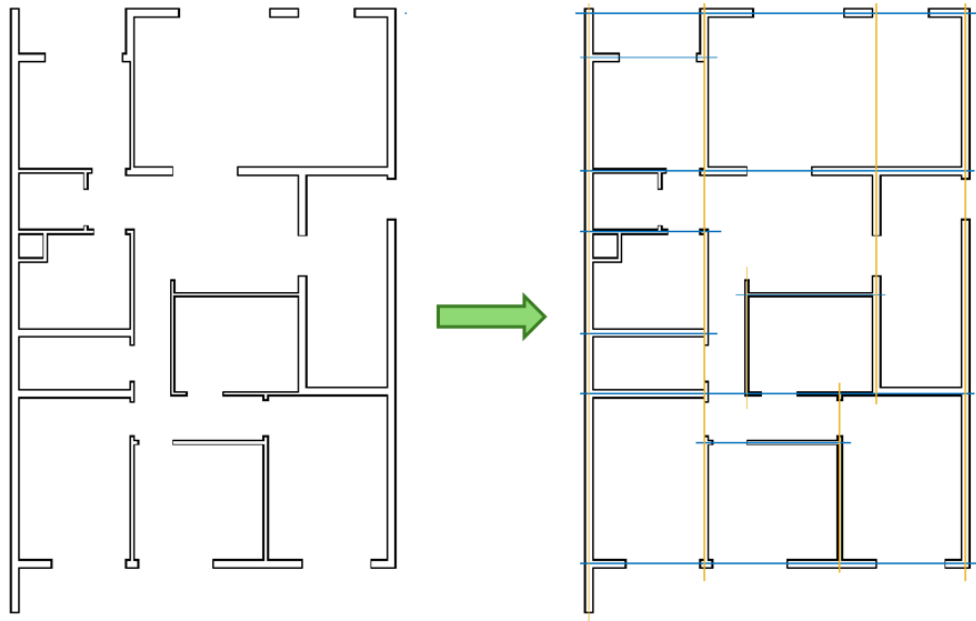
Bu eşik değerin belirlenmesinde, TBDY 2018 yönetmeliğinin 7.4.1 bölümündeki 1.b ve 1.c maddeleri temel alınmıştır. İlgili maddelere göre kiriş yüksekliği, döşeme kalınlığının üç katından ve aynı zamanda 300 mm'den az olamaz. Ayrıca kiriş yüksekliği serbest açıklığın dörtte birinden büyükse, kirişte ek gövde donatısı gerekmektedir. Yapısal karmaşıklık ve donatı maliyetini artırmamak amacıyla, bu gereklilikler temel alınarak minimum aks uzunluğu belirlenmiştir. Yaygın olarak kullanılan 12 cm'lik döşeme kalınlığı dikkate alındığında, en az kiriş yüksekliği 36 cm olmakta, gövde donatısı eklememek için gerekli serbest açıklık ise 144 cm olarak hesaplanmaktadır. Bu değere karşılık gelen mesafe, algorithmada aks oluşturulmasına izin verilen minimum duvar uzunluğu olarak kullanılmaktadır.



Şekil 3.14. Birbirine yakın aksların tek bir aks olarak düzeltilmesi

Duvarlara ait tüm akslar tanımlandıktan sonra, aynı hizada bulunan aks çizgileri birleştirilerek sadeleştirilmiş bir aks sistemi elde edilmektedir. Ancak bu aşamada karşılaşılan başlıca sorunlardan biri, uzun bir duvar boyunca herhangi bir noktada duvar kalınlığının değişmesi durumudur. Bu durumda aynı eksenle birden fazla aks oluşmakta ve aks sistemi gereksiz şekilde parçalanmaktadır. Bu sorunun giderilmesi amacıyla özel bir düzeltme algoritması geliştirilmiştir (Şekil 3.14). Geliştirilen algoritma, birbirine dik uzaklığı 25 cm'den az olan ve aynı doğrultuda bulunan aks çizgilerini birleştirmekte; yeni aksın konumu ise bu iki aks arasındaki dik mesafenin orta noktası olarak tanımlanmaktadır.

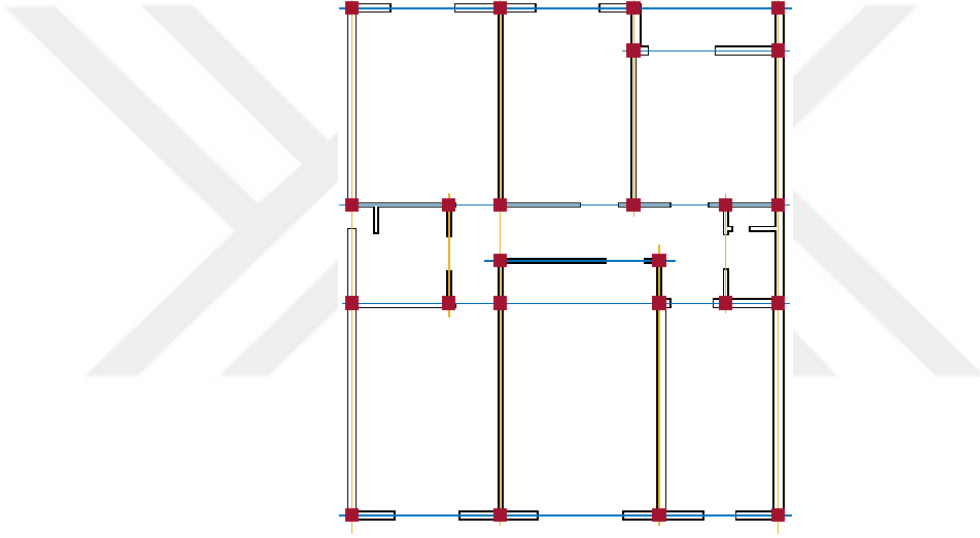
Sonuç olarak, yazılımın bu aşamasında elde edilen örnek bir aks yerleşimi çıktısı Şekil 3.15'te sunulmuştur.



Şekil 3.15. Sistematik olarak plandaki aksların belirlenmesi

3.4.6. Olası kolon yerlerinin belirlenmesi

Bu aşamada, mimari plandaki olası kolon noktalarının sistematik biçimde belirlenmesi hedeflenmiştir. Kolon yerlerinin doğru tespit edilmesi, hem taşıyıcı sistemin modellenmesi hem de optimizasyon ve analiz işlemleri için temel veri niteliğindedir. İlk olarak, yatay ve düşey aksların kesişim noktaları belirlenmekte; bu koordinatların her birinde kolon merkezi olacağı varsayımıyla bir kolon matrisi oluşturulmaktadır. Bu matris, her kolonun numaralandırılmış adını, görüntü verisi matrisindeki satır ve sütun koordinatlarını, bağlı olduğu yatay ve düşey aks numaralarını içermektedir. Oluşturulan bu kolonlar kullanıcıya görsel olarak sunulabilmektedir (Şekil 3.16).

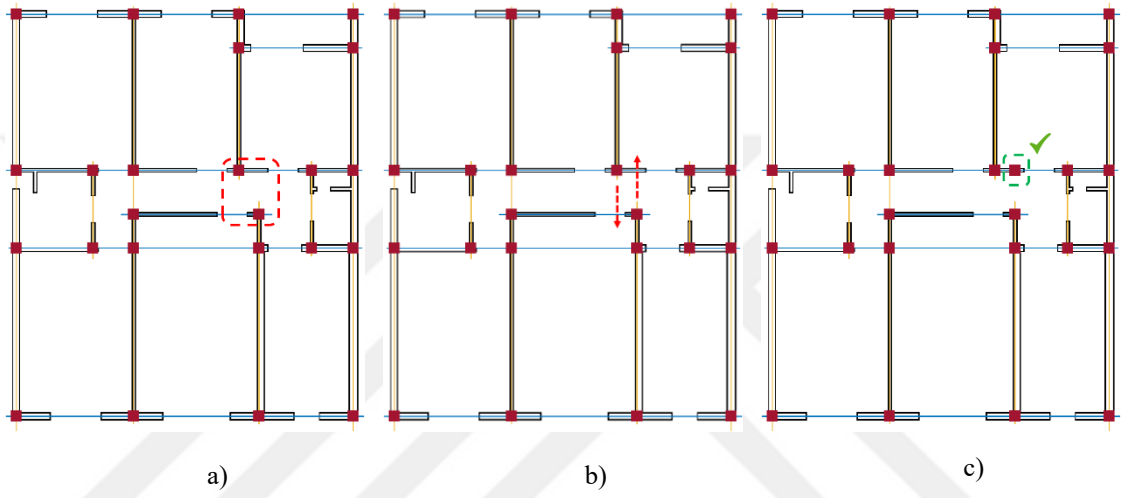


Şekil 3.16. İlk oluşturulan kolonların akslarla beraber plan üzerinde gösterimi

Ancak incelenen birçok mimari planın simetrik ve düzgün dörtgensel geometriye sahip olmaması nedeniyle, bazı akslar plan boyunca kesintili kalmaktadır (Şekil 3.17.a). Oysa taşıyıcı sistem tasarımı açısından aks sürekliliği büyük önem taşımaktadır. Bu problemi çözmek amacıyla, çalışma kapsamında ek kolon olarak adlandırılan ve normalde aks kesişimlerinde yer almayan kolon noktalarının sistematik biçimde tanımlanması üzerine bir algoritma geliştirilmiştir. Bu sayede, düzensiz akslar yapısal olarak birbirine bağlanmakta ve aks sürekliliği sağlanmaktadır.

Geliştirilen algoritma, öncelikle aynı doğrultuda (yatay ya da düşey) ardışık aksları taramakta; birbirine dik uzaklığı 144 cm'den az olan ve piksel düzeyinde sütun (yatay akslar için) veya satır (düşey akslar için) koordinatlarında ortak eleman içermeyen aks çiftlerini tespit etmektedir. Ancak bu koşul, her zaman ek kolon gereksinimini kesin

olarak belirlemek için yeterli olmamaktadır. Bu nedenle, birbirine yakın olan akslar belirlendikten sonra, hangisinin diğerine doğru uzatılarak bağlanmasının daha uygun olduğu değerlendirilmekte ve karar bu doğrultuda verilmektedir. Bu işlem sırasında, her iki aks da dik doğrultudaki en yakın akslara doğru uzatılmakta ve bağlanacak noktalarda duvar ya da boşluk olup olmadığı kontrol edilmektedir (Şekil 3.17.b). Eğer her iki bağlantı noktasında da duvar mevcutsa, bu iki noktaya ek kolonlar yerleştirilmekte ve kolon matrisi içine dahil edilmektedir (Şekil 3.17.c).



Şekil 3.17. Ek kolon algoritmasının aşamaları

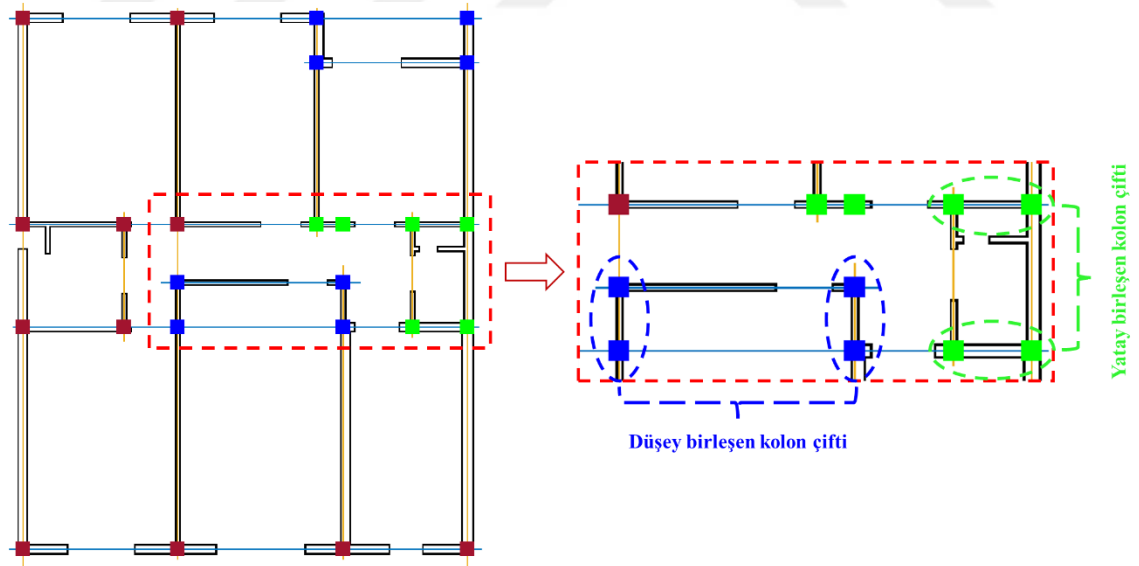
Ek kolonların sisteme dahil edilmesi, aks sürekliliğini artırmakla birlikte, bu kez minimum kiriş açıklığı koşulunun (Bölüm 3.4.5) ihlal edilmesine neden olabilmektedir. Bu koşulun ihlali, diğer ardışık ve paralel akslar arasındaki mesafenin kısa olmasından dolayı da meydana gelebilmektedir. Bu sorunun önüne geçmek amacıyla, birleşen kolonlar olarak adlandırılan yeni bir kavram tanımlanmıştır. Bu kapsamda, akslar arasında minimum açıklığın altında kalan kirişlerin uç noktalarındaki kolonlar, ilerleyen analiz aşamalarında birleştirilebilmek üzere işaretlenmektedir. Hangi kolonların birleştirileceğini belirlemek üzere algoritmaya özel bir kontrol yapısı eklenmiştir.

Bu işlemde tüm kolon matrisi yatay ve düşey doğrultuda ardışık kolonlar açısından taranmakta, aralarındaki mesafe 144 cm'den daha az olan kolon çiftleri belirlenmektedir. Ardından, bu kolonlar arasındaki bölgenin tamamında duvar bulunup bulunmadığı kontrol edilmektedir. Eğer bu aralık boyunca sürekli bir duvar mevcutsa, ilgili kolon çifti birleşen kolon olarak etiketlenmekte ve hangi kolonların birleştirileceği bilgisi ayrı bir

parametre olarak saklanmaktadır. Bu kontrol sayesinde, plan üzerinde boş alanların kolon birleştirme işlemiyle kapatılması da önlenmiş olmaktadır.

Son olarak, incelenen mimari planlarda aynı hizada ve paralel şekilde konumlanmış birleşen kolon çiftlerinin oluşabildiği gözlemlenmiştir. Bu durumda, söz konusu kolonlar arasında birbirine çok yakın iki kirişin yer alması mümkün olmaktadır. Ancak bu tür durumlar hem mimari açıdan (örneğin bir kirişin odanın ortasından geçmesini önlemek amacıyla) hem de taşıyıcı sistemin aks sürekliliği üzerinde bir etkisi bulunmadığı için, bu çiftler arasındaki kirişlerin sayısının bire düşürülmesi hedeflenmiştir.

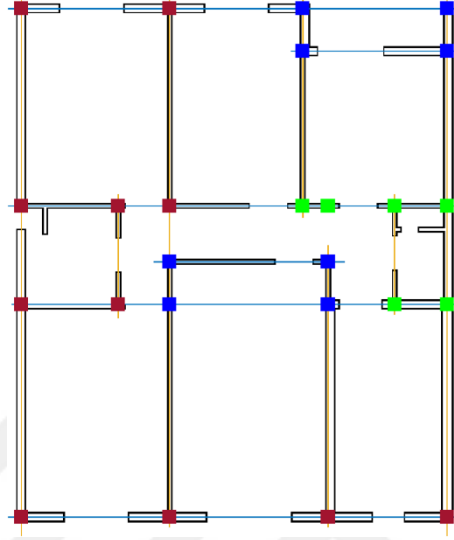
İlerleyen aşamalarda kiriş düzenlemelerinin doğru biçimde yapılabilmesi için, kolon matrisinde bu tür birleşen kolon çiftlerine ait kolonlara ilişkin ek bilgi alanı tanımlanmış ve bu işaretleme işlemini gerçekleştirecek bir algoritma geliştirilmiştir. Söz konusu algoritma, daha önce belirlenmiş olan birleşen kolonları teker teker diğer birleşen kolonlarla karşılaştırmakta; kolonların görüntü verisi matrisindeki konumlarına göre, düşey doğrultuda bulunan birleşen kolonlar için aynı yatay hizada, yatay doğrultudakiler için ise aynı düşey hizada bulunan çiftler birleşen kolon çifti olarak işaretlenmektedir (Şekil 3.18).



Şekil 3.18. Birleşen kolon çiftlerinin gösterimi

Sonuç olarak, bu aşamada mimari plandaki tüm olası kolon yerleri sistematik biçimde tespit edilmekte ve yapı analizi ile optimizasyon işlemlerinde kullanılmak üzere kolon bilgisi matrisi oluşturulmaktadır. Bu matris; aks kesişimlerinde yer alan kolonları, algoritma tarafından tanımlanan ek kolonları ve birleşme durumu işaretlenmiş kolonları

içermektedir. Şekil 3.19’da, yazılım tarafından kullanıcıya görsel çıktı olarak sunulan örnek bir kolon yerleşim düzeni gösterilmektedir. Bu örnekte açık yeşil renkli noktalar yatay ekseninde birleşen kolonları, lacivert renkli noktalar ise düşey ekseninde birleşen kolonları ifade etmektedir.



Şekil 3.19. Olası kolon yerleri tespiti sonucu

Bu bölümde yer alan görsellerdeki kolon kesitleri tamamen temsili olup, yalnızca mimari planda kolon yerleşimine uygun olabilecek muhtemel konumların gösterimi amacıyla kullanılmıştır.

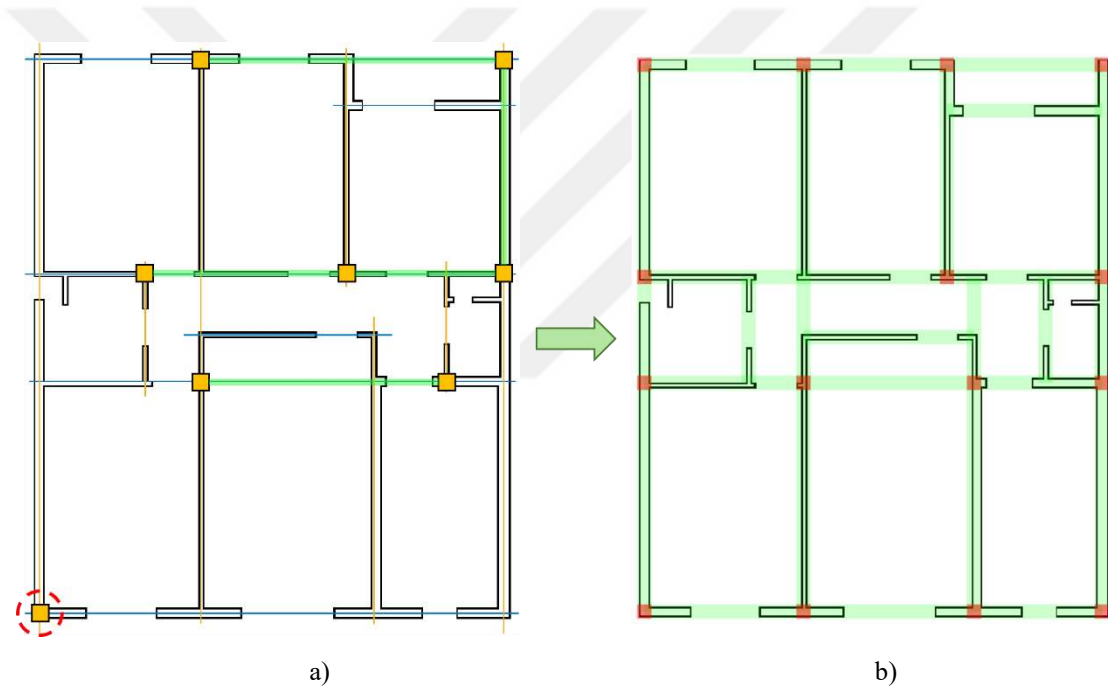
3.4.7. Kiriş sisteminin belirlenmesi

Bu aşamada, yapı analizinde kullanılmak üzere kirişlerin düğüm noktalarıyla birlikte tanımlanacağı bir olası kiriş matrisi oluşturulmaktadır. Bir önceki adımda, sonlu elemanlar (SE) modeli kapsamında düğüm noktası teşkil edebilecek tüm konumlar, olası kolon yerleri olarak belirlenmiştir. Bu nedenle, bu bölümde geliştirilen algoritma, kolon matrisinde yer alan konum bilgilerini kullanarak tüm potansiyel kiriş konfigürasyonlarını önceden sistematik biçimde tanımlamayı amaçlamaktadır.

Optimizasyon sürecinde kolon yerleşimi her iterasyonda farklılık gösterebildiğinden, herhangi bir aks üzerinde hangi düğüm noktaları arasında kiriş tanımlanacağı önceden bilinmemektedir. Bu belirsizliği ortadan kaldırmak amacıyla, algoritma her aks doğrultusundaki düğüm noktaları arasında tanımlanabilecek tüm ikili kombinasyonları üretmekte ve bunları olası kirişler olarak numaralandırarak kiriş

matrisine kaydetmektedir. Böylece, ilerleyen analiz aşamalarında mevcut kolon konfigürasyonuna uygun geçerli kirişler bu matris üzerinden kolaylıkla seçilebilmekte ve yapı modeli otomatik olarak kurulabilmektedir.

Çalışmanın ilk aşamalarında geliştirilen bu sistemin temel amacı, optimizasyon sürecinde yalnızca planda mevcut kolonlar arasında kirişlerin bulunmasını sağlamaktır. Ancak daha sonra yapılan optimizasyon denemelerinde, bazı iterasyonlarda planın belirli bölgelerinde yatay ya da düşey doğrultuda başka hiçbir kolona bağlanamayan tekil kolonların olduğu gözlemlenmiştir (Şekil 3.20.a). Bu durum, sonraki bölümlerde açıklanacak olan yük dağıtım mekanizması nedeniyle bu kolonlara yük iletilmemesine neden olmakta ve söz konusu kolonlar yapıya yapısal bir katkı sağlayamamaktadır.



Şekil 3.20. a) Hatalı kiriş sistemi, b) Kullanılan kiriş sistemi

Bu sorunu gidermek amacıyla, yapı modellemesinde sabit bir kiriş sistemine geçilmiştir (Şekil 3.20.b). Bu sistemde, plandaki her düğüm noktası çifti arasında kiriş tanımlanmakta ve bu sayede kolonların çevresel bağlantıları garanti altına alınmaktadır. Bu geçişe rağmen, başlangıçta oluşturulmuş olan ikili düğüm kombinasyonları matrisinin yapısı korunmuştur. Modelde tanımlanacak kirişler hala bu matris üzerinden çekilmekte, ancak yalnızca belirli özel durumlarda bazı kirişler analiz modelinden çıkartılmaktadır.

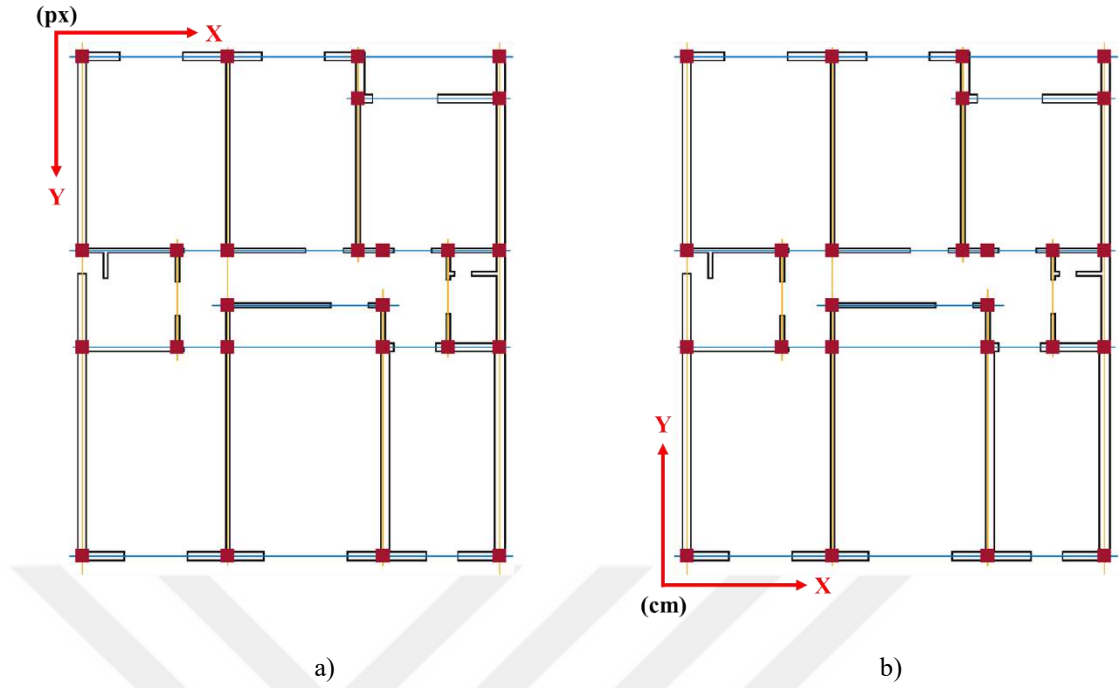
Bu özel durumlar, birleşen kolonlar veya birleşen kolon çiftleri ile ilgilidir. Birleşen kolonlar arasında kalan kirişler, ilerleyen aşamalarda yalnızca tek bir kolonun bırakılması

nedeniyle analiz modelinden kaldırılmaktadır. Benzer şekilde, birleşen kolon çiftlerine bağlanan ve aynı aks doğrultusunda bulunan iki kirişten yalnızca biri analizde tutulmakta, diğer kiriş sistem dışı bırakılmaktadır. Böylece hem yapısal anlamda gereksiz elemanların modele dahil edilmesi engellenmekte hem de analizde oluşabilecek çakışmaların önüne geçilmektedir.

3.4.8. Parametrik çıktıların düzenlenmesi

Görüntü işleme modülünün, yazılımın diğer modülleriyle entegre biçimde çalışabilmesi için bu modülden elde edilen parametrelerin uygun formatta ve sistematik şekilde düzenlenmesi gerekmektedir. Modül kapsamında, görüntü verisi üzerinde yapılan işlemler sonucunda sırasıyla duvar çizgileri, duvarlar, akslar, kolonlar ve kirişlere ait konum ve uzunluk bilgileri piksel cinsinden elde edilmektedir. Plan ölçeği yardımıyla bu uzunluklar santimetre cinsine çevrilerek gerçek ölçülerle uyumlu hale getirilmektedir.

Ancak yapı modeli oluşturulurken, yalnızca konumların santimetre cinsinden değerlerine değil, aynı zamanda eksen sistemine uygun olarak yeniden tanımlanmasına da ihtiyaç duyulmaktadır. Görüntü verisi satır ve sütunlar üzerinden okunduğunda, matris indekslleme sistemine bağlı olarak sol üst köşe orijin kabul edilmekte; yatay eksen sağa, düşey eksen ise aşağı yönde tanımlanmaktadır (Şekil 3.21.a). Bu sistem, görüntü işlemede işlevsel olsa da, yapının SE modellemesinde genellikle yatay eksenin sağa, düşey eksenin ise yukarı yönlü olduğu bir koordinat sistemi tercih edilmektedir (Şekil 3.21.b). Bu farkın giderilmesi amacıyla, eksen dönüşümünü gerçekleştiren bir algoritma geliştirilmiş ve elde edilen tüm konum bilgileri uygun şekilde bu sisteme çevrilmiştir.



Şekil 3.21. Eksenlerin gösterimi a) Piksel cinsinden (px), b) Santimetre cinsinden (cm)

Elde edilen konum verilerinin hem piksel hem de santimetre cinsinden tutulması, kullanıcıya görsel çıktı sunulması durumunda piksel bazlı gösterimlerin sağlanabilmesi açısından önemlidir. Bu nedenle, eksen dönüşümü ve uzunluk dönüşümü uygulanmış veriler ayrı matrisler olarak yapılandırılmış ve tümü modül çıktılarına dahil edilmiştir.

Bunun yanı sıra, yazılımın sonraki aşamalarında gerçekleştirilecek olan yük hesaplarında kullanılmak üzere kat alanı hesaplaması da yapılmıştır. Yazılımın mevcut sürümü yalnızca dikdörtgensel plan geometrilerini desteklediğinden, köşe kolonlarının konumları kullanılarak yaklaşık kat alanı belirlenmekte ve bir çıktı parametresi olarak tanımlanmaktadır.

Bu kapsamda, görüntü işleme modülünün sağladığı çıktı parametreleri şu şekildedir:

- Kolon matrisi (santimetre cinsinden, eksen dönüşümü yapılmış)
- Kolon matrisi (piksel cinsinden)
- Kiriş matrisi (santimetre cinsinden, eksen dönüşümü yapılmış)
- Kiriş matrisi (piksel cinsinden)
- Düzeltilmiş ikili format kat planı görüntü verisi
- Plan ölçeği
- Birleşen kolon matrisi

- Birleşen kolon çiftleri matrisi
- Yatay duvar çizgileri matrisi
- Düşey duvar çizgileri matrisi
- Yatay duvarlar matrisi
- Düşey duvarlar matrisi
- Yatay akslar matrisi
- Düşey akslar matrisi
- Kat alanı

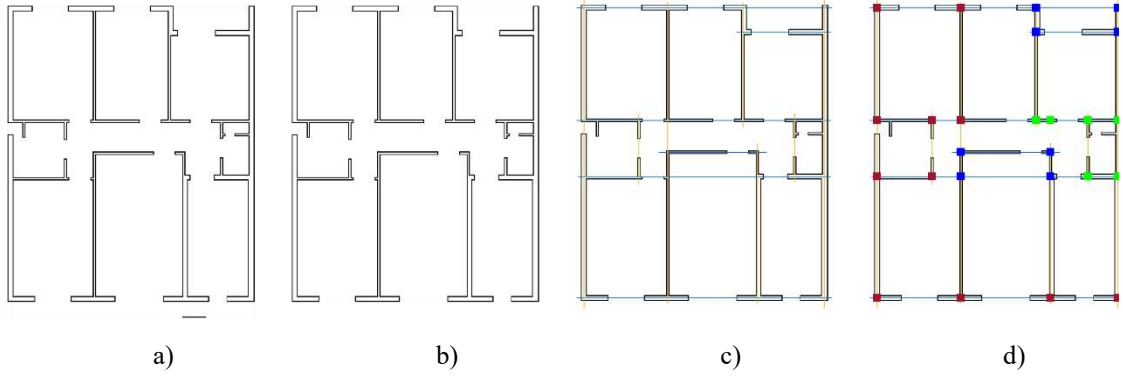
3.4.9. Görsel çıktıların gösterimi

Görüntü işleme süreci, mimari plan verisinin otomatik analizinde çok sayıda işlem adımını içermekte olup bu adımların doğruluğu, sistemin genel güvenilirliğini doğrudan etkilemektedir. Bu nedenle, kullanıcıya analiz sürecinde elde edilen ara çıktıları gözlemlene ve gerekli durumlarda müdahalede bulunma imkanı tanınması, hem hata tespiti hem de algoritmanın şeffaflığı açısından önemlidir. Bu bağlamda, sistemde kullanıcıya belirli aşamalarda görsel çıktıların gösterilip gösterilmeyeceğini seçme olanağı tanıyan bir parametre tanımlanmıştır.

Bölüm 3.2.1.1’de açıklanan kullanıcı girdisi doğrultusunda, görüntü işleme sürecinin belirli adımlarında elde edilen ara sonuçlar, kullanıcının tercihine bağlı olarak görselleştirilmekte ve kontrol amaçlı sunulabilmektedir. Bu kapsamda görselleştirilen kritik aşamalar Şekil 3.22’de sunulmakta olup; sırasıyla ikili format görüntü, belirlenen duvar çizgileri, aksların tespiti ve kolon yerlerinin belirlenmesi adımlarını içermektedir.

Bu görselleştirme sürecinde her adım, bir önceki işlem sonucuna bağlı olarak oluşturulduğundan, herhangi bir aşamadaki hata doğrudan sonraki adımları da etkilemektedir. Bu durum, hatalı bir analizin erken aşamalarda fark edilmesini kolaylaştırmakta ve böylece nihai kolon yerleşiminin doğruluğunun güvence altına alınmasına katkı sağlamaktadır.

Şekil 3.22’de, örnek bir mimari plan üzerinde görüntü işleme modülü tarafından oluşturulan görseller sunulmaktadır.



Şekil 3.22. Görüntü işleme süreci a) gri seviye görüntü, b) ikili format görüntü, c) aksların oluşumu, d) olası kolon lokasyonlarının belirlenmesi

3.5. Optimizasyon ile Yapı Tasarımı Modülü

Yapı mühendisliğinde taşıyıcı sistemlerin tasarımı, güvenlik ve ekonomik performans açısından çok boyutlu kararların birlikte değerlendirilmesini gerektiren karmaşık bir problemdir. Mimari planın sunduğu sınırlar dahilinde kolon ve kiriş yerleşimlerinin belirlenmesi, betonarme yapıların dayanımı ve yönetmeliklere uygunluğu açısından doğrudan belirleyici bir rol oynamaktadır. Bu çalışmada, görüntü işleme modülü ile otomatik olarak elde edilen mimari plan verileri temel alınarak, taşıyıcı sistemin yerleşimi ve boyutlandırılmasına yönelik bir optimizasyon modülü geliştirilmiştir.

Gerçekleştirilen bu optimizasyon süreci, çok sayıda ikili ve tamsayı karar değişkeni içermesi, çözüm uzayının yüksek boyutlu ve seyrek olması, aynı zamanda yönetmelik uyumunun katı kurallarla sağlanması zorunluluğu nedeniyle klasik yöntemlerle doğrudan çözülemeyecek ölçüde zorlayıcıdır. Bu nedenle, karar değişkenlerinin tanımlanması, amaç fonksiyonunun oluşturulması ve ceza terimlerinin belirlenmesi gibi konular ayrı alt başlıklar altında detaylandırılmıştır (Bölüm 3.5.1).

Optimizasyon sürecinde kullanılan karar değişkenleri, kolonların plana yerleştirilip yerleştirilmeyeceğini belirleyen ikili türde değişkenler ile, seçilen kolon ve kirişlerin kesit numaralarını tanımlayan tamsayı türde değişkenlerden oluşmaktadır. Bu yapı itibarıyla problem, literatürde karma tamsayılı karar problemleri (Mixed Integer Problems – MIP) olarak tanımlanan sınıfa girmektedir.

Amaç, hem yapı güvenliğini sağlayan hem de toplam yapı maliyetini minimize eden kolon–kiriş kombinasyonlarını algoritmik bir arama süreci ile elde etmektir. Bu kapsamda oluşturulan hedef fonksiyon, hem beton hem de donatı miktarlarına bağlı

olarak yapı maliyetini hesaba katmakta; yönetmelik koşullarını sağlamayan çözümler için ceza terimleri içermektedir. Bu sayede, uygun olmayan çözümlerin elenmesi veya düşük öncelikli hale getirilmesi sağlanmaktadır.

Çözüm uzayının genişliği ve türevlenebilir bir yapıdan uzak olması nedeniyle, amaç fonksiyonunun kapalı formda matematiksel bir ifadesi bulunmamaktadır. Bunun temel nedeni, karar değişkenlerinden elde edilen yapı konfigürasyonlarının, dışsal bir analiz modülüne (yapı analiz programı) aktarılması ve değerlendirme işlemlerinin yalnızca bu analizler aracılığıyla yapılabilmesidir. Dolayısıyla sistem, kara kutu (black-box) karakteristiği göstermekte ve çözüm arama sürecinde türev bilgisine ihtiyaç duymayan sezgisel algoritmalar tercih edilmektedir.

Bu bağlamda, farklı doğrudan arama temelli sezgisel algoritmalar karşılaştırmalı olarak çalıştırılmıştır. Uygulanan algoritmalar şunlardır:

- Adaptif Parçacık Sürü Optimizasyonu (APSO)
- Farklılaşım Evrimi (DE)
- Tavlama Benzetimi (SA)
- JAYA algoritması
- İyileştirilmiş Aritmetik Optimizasyon Algoritması (IAOA)

Bu algoritmaların detaylı tanımları ve parametre yapılarına Bölüm 3.5.2’de yer verilmiştir. Optimizasyonun yazılımsal modül tasarımı ise Bölüm 3.5.3’te optimizasyona hazırlık ve döngü aşamaları ile birlikte açıklanmıştır. Ayrıca optimizasyon işlemlerinde kullanılan paralel işlemeye dayalı hesaplama yapısı Bölüm 3.6’da, optimizasyon sonrası çıktıların işlenmesi ve görselleştirilmesi Bölüm 3.7’de, geliştirilen yöntemin doğruluğu ve tutarlılığına yönelik analizler ise Bölüm 3.8’de detaylandırılmıştır.

3.5.1. Optimizasyon kurgusu

Bu çalışmada geliştirilen yapı optimizasyon modülü, karma tamsayı yapısına sahip bir çözüm uzayı üzerinde çalışmakta ve her bir aday çözümü hem yapısal topoloji hem de kesit boyutlandırma açısından değerlendirmektedir. Bu bağlamda, algoritmanın etkin çalışabilmesi ve çözüm uzayının açık biçimde tanımlanabilmesi için, optimizasyon problemine ait değişken vektörü ve amaç fonksiyonu önceden dikkatlice kurgulanmıştır.

Değişken vektörü, yapısal sistemin hem kolon yerleşimini hem de kolon ve kiriş kesit boyutlarını temsil edecek şekilde üç parçalı bir yapıda tasarlanmıştır. Bu sayede her

aday çözüm, yapı modelinin farklı tasarım senaryolarını temsil edebilecek nitelikte bir veri kümesi olarak tanımlanabilmektedir.

Öte yandan, yapı analizi ve tasarım süreçleri ile entegre biçimde çalışan amaç fonksiyonu, her aday çözüm için maliyet ve yönetmelik uygunluğu kriterlerini birlikte değerlendiren bir yapıdadır. Bu fonksiyonun doğru tanımlanması, algoritmanın sadece düşük maliyetli değil, aynı zamanda yönetmeliklere uygun tasarımları tercih etmesini sağlamaktadır.

Bu başlık altında, geliştirilen optimizasyon kurgusunun temel bileşenleri olan değişken vektörü ve amaç fonksiyonunun nasıl oluşturulduğu açıklanmakta; söz konusu yapıların algoritma başarısına olan katkısı tartışılmaktadır.

3.5.1.1. Değişken vektörü yapısı

Bu çalışmada önerilen optimizasyon sistemi, taşıyıcı sistemin hem topolojik düzenini hem de kesitsel özelliklerini eşzamanlı olarak belirlemeyi hedeflemektedir. Bu nedenle, karar değişkenlerinin tanımı ve optimizasyon algoritmasına uygun şekilde yapılandırılması, çözüm kalitesini doğrudan etkileyen kritik bir adımdır. Bu nedenle çalışmada iki alternatif değişken vektörü kurgusu düşünülmüş ve teste tabi tutulmuştur. Değişken vektörü, çözüm uzayındaki her bireyin (yani her aday tasarımın) yapı modelini temsil ettiği bir veri yapısıdır.

Birinci değişken vektörü kurgusu (DV1), yapıdaki kolon yerleşimlerini, kolon kesitlerini ve kiriş kesitlerini ayrı ayrı tanımlayan bileşenlerden oluşmakta olup, hem ikili hem de ayrık tamsayılı değişkenler içeren karma bir yapıya sahiptir. Bu nedenle, değişken vektörü üç ana alt bileşene ayrılmaktadır.

İkinci değişken vektörü kurgusunda (DV2) ise, ilk kurguda yer alan ikili değişken bileşeni vektörden çıkarılmıştır. Bu yaklaşımla çözüm uzayının boyutu önemli ölçüde azaltılmak istenmiştir. Kolon topoloji bilgisi artık vektörde doğrudan bulunmadığından, bu eksiklik kolon kesit bilgilerini içeren parçada alt sınır değerinin sıfır olarak tanımlanması yoluyla giderilmeye çalışılmıştır. Bu şekilde, sıfır değeri kolonun o konumda bulunmadığını temsil etmekte, pozitif tamsayılar ise farklı kesit tiplerini tanımlamaktadır.

Kiriş kesitlerine ilişkin yapı modellemesi kurgusunda, aynı aks üzerindeki tüm kirişlerin aynı kesite sahip olması ilkesi benimsenmiştir. Bu yaklaşım, hem yapı tasarımı açısından aralarında kolon bulunmayan ve farklı kesit özelliklerine sahip kirişlerin yer

almasının önüne geçilmesini sağlamakta, hem de optimizasyon sürecindeki çözüm uzayının daha hedef odaklı biçimde daraltılmasına olanak tanımaktadır. Bu nedenle, değişken vektörlerinde giriş kesitlerini temsil eden kısım bu ilkeye uygun olarak yapılandırılmıştır.

- **Kolon topoloji bileşenleri**

İlk bölüm, yapıda yer alan her bir kolonun var olup olmadığını belirten ikili değişkenlerden oluşmaktadır (Denklem 3.5).

$$x_i^{bin} = \begin{cases} 1, & \text{kolon } i \text{ modelde mevcut ise} \\ 0, & \text{kolon } i \text{ modelde mevcut değil ise} \end{cases}, \quad i = 1, 2, \dots, n_c \quad (3.5)$$

Burada n_c , görüntü işleme modülü sonucu belirlenen toplam potansiyel kolon sayısını ifade eder.

- **Kolon kesit kimlikleri**

İkinci bölüm, mevcut olduğu belirlenen kolonlar için kesit boyutlarını tanımlamaktadır (Denklem 3.6). Bu değerler, kullanıcı tarafından tanımlanan kolon kesit havuzundaki indeks numaralarıdır.

$$x_i^{col} = \begin{cases} 1, 2, \dots, S_c & \text{eğer DV1 ise} \\ 0, 1, \dots, S_c & \text{eğer DV2 ise} \end{cases}, \quad i = 1, 2, \dots, n_c \quad (3.6)$$

Burada S_c kolon kesit havuzundaki toplam kesit sayısını göstermektedir. DV1 kurgusunda eğer $x_i^{bin} = 0$ ise, ilgili x_i^{col} değeri sonlu elemanlar modeli oluşturulması ve amaç fonksiyonu hesabında dikkate alınmamaktadır.

- **Kiriş kesit kimlikleri**

Üçüncü bölüm, modeldeki tüm girişlerin kesit tiplerini tanımlamaktadır (Denklem 3.7). Kat planındaki toplam aks sayısı n_{aks} olup, her aksta bulunan girişler kesit havuzundaki indekslerle temsil edilmektedir.

$$x_j^{beam} \in 1, 2, \dots, S_b, j = 1, 2, \dots, n_{aks} \quad (3.7)$$

Burada S_b giriş kesit havuzundaki toplam kesit sayısını göstermektedir.

- **Değişken vektörünün tam tanımı**

Bahsi geçen üç bölümün birleşimi ile DV1’de oluşan değişken vektörü (Denklem 3.8):

$$\mathbf{X} = [x_1^{bin}, \dots, x_{n_c}^{bin}, x_1^{col}, \dots, x_{n_c}^{col}, x_1^{beam}, \dots, x_{n_b}^{beam}] \quad (3.8)$$

şeklinde tanımlanmaktadır. Böylece $\mathbf{X} \in \mathbb{Z}^n$ olup, burada $n = 2 \cdot n_c + n_b$ olmak üzere toplam değişken sayısını göstermektedir.

Sonuç olarak DV2’de oluşan kurgu da Denklem 3.9’da sunulmaktadır.

$$\mathbf{X} = [x_1^{col}, \dots, x_{n_c}^{col}, x_1^{beam}, \dots, x_{n_b}^{beam}] \quad (3.9)$$

- **Çözüm uzayı**

Bu kurgu, hem yapısal topolojiyi hem de boyutlandırmayı aynı anda optimize etme imkanı sağlamaktadır. Çözüm uzayının boyutu DV1 ve DV2 için sırasıyla Denklem 3.10 ve Denklem 3.11’de sunulmaktadır.

$$|\Omega| = (2 \cdot S_c)^{n_c} \cdot S_b^{n_{aks}} \quad (3.10)$$

$$|\Omega| = (S_c + 1)^{n_c} \cdot S_b^{n_{aks}} \quad (3.11)$$

n_c ve n_{aks} arttıkça bu sayı üstel olarak büyümekte, çözüm uzayı hem geniş hem de seyrek bir hal almaktadır. Bu özellikler, deterministik yöntemlerin uygulanabilirliğini sınırlamakta ve sezgisel tabanlı algoritma tercih edilmesini gerekli kılmaktadır.

Sonuç olarak değişken vektörü kurgusu, bu çalışmanın bütün algoritmik yapısının temelini oluşturmaktadır. Vektörün hibrit yapısı, sistemin hem topoloji hem de kesit düzeyinde optimizasyon yapabilmesine olanak sağlamaktadır. Ayrıca her çözüm adayının, sayısal analiz öncesi yapı modeline çevrilebilmesini mümkün kılmaktadır. Bu yaklaşım, otomasyon sürecinin esnek, genişletilebilir ve sezgisel algoritmalarla tam entegre çalışmasına olanak tanımaktadır.

3.5.1.2. Amaç fonksiyonu kurgusu

Bir optimizasyon probleminin temel bileşenlerinden biri olan amaç fonksiyonu, her aday çözümün nicel olarak değerlendirilmesini sağlayarak arama sürecini yönlendiren en kritik unsurdur. Bu çalışmada geliştirilen optimizasyon yaklaşımında amaç, hem ekonomik hem de yapısal açıdan geçerli olan çözümlerin belirlenmesidir. Bu doğrultuda kullanılan amaç fonksiyonu, yapı sistemine ait toplam maliyetin minimize edilmesini hedeflemekte; aynı zamanda mühendislik tasarım ilkeleriyle uyumsuz çözümler için bir cezalandırma mekanizması içermektedir.

Optimizasyon sürecinde her aday çözüm, bir yapı modelini topolojik (kolonların konumu) ve boyutsal (kesit boyutları) açıdan tanımlayan bir değişken vektörü ile temsil edilmektedir. Bu vektör doğrultusunda oluşturulan yapısal model, analiz edilerek taşıyıcılık koşulları açısından değerlendirilmekte; ardından, çözümün toplam maliyeti hesaplanmaktadır. Ancak bazı aday çözümler, yürürlükteki tasarım yönetmeliklerine göre yetersiz olabilmektedir. Bu durumda, yapı analizi sonucu elde edilen performans

değerleri, yönetmelik sınırlarını sağlamıyorsa çözüm cezalandırılmakta ve amaç fonksiyonu değeri bu doğrultuda artırılmaktadır.

Bu durumda, yapı analizi sonucu elde edilen performans değerleri, yönetmelik sınırlarını sağlamıyorsa çözüm cezalandırılmakta ve amaç fonksiyonu değeri bu doğrultuda artırılmaktadır. Ancak bu ceza terimlerinin, toplam maliyet ile aynı büyüklük mertebesinde olmaması durumunda, amaç fonksiyonunda etkisiz kalması ya da aşırı baskın hale gelmesi mümkündür. Bu nedenle toplam maliyet değeri, her plan için önceden oluşturulan bir referans model üzerinden normalize edilmekte; böylece hem maliyet hem de ceza terimleri aynı mertebede değerlendirme ağırlığına sahip olmaktadır. Bu normalizasyon, algoritmanın hem yönetmelik açısından uygun çözümlere (geçerli çözümler) hem de bu çözümler arasında en düşük maliyetli olanlara yönelmesini sağlamaktadır.

Bu bağlamda kullanılan amaç fonksiyonu şu şekilde tanımlanabilir (Denklem 3.12):

$$f(x) = \frac{C(x)}{C_{maks}} + \sum_{i=1}^{n_p} P_i(x) \quad (3.12)$$

Burada:

- x : Aday çözüm vektörünü ifade etmektedir.
- $C(x)$: Aday çözüme karşılık gelen yapı modelinde, beton ve donatı miktarları esas alınarak hesaplanan toplam maliyettir.
- C_{maks} : İlgili kat planında tüm kolonların mevcut olduğu ve kesit havuzundaki en büyük kesitlerin atandığı referans modelin beton ve donatı miktarı esaslı toplam maliyetidir.
- n_p : Yazılım kapsamında tanımlı tüm yönetmelik kontrollerinin sayısını göstermektedir.
- $P_i(x)$: Aday çözümün sonlu elemanlar analizi sonrasında her bir yönetmelik kontrolü için elde edilen uygunluk değeridir. Bu değer, kontrolün sağlanması durumunda 0, sağlanamaması durumunda ise 1 olarak tanımlanan ceza katsayısını ifade etmektedir.

Bu kurgu sayesinde uygun çözümler kendi aralarında maliyetlerine göre sıralanabilirken cezalı çözümler sıralamada geriye düşmektedir. Bu yaklaşım, algoritmanın çözüm uzayında yapı mühendisliği açısından geçerli tasarımlara odaklanmasını sağlamakta, aynı zamanda ekonomikliği de gözetmektedir.

Sonu olarak ama fonksiyonu kurgusu, mhendislikte pratik uygulanabilirliđi yksek, ok kriterli bir deđerlendirme yaklařımı sunmakta; maliyet, gvenlik ve ynetmelik uygunluđunu eř zamanlı olarak ele almaktadır. Bu yapı, yalnızca en ekonomik zm deđil, aynı zamanda en uygulanabilir ve ynetmeliklere uygun zm semeyi garanti altına almaktadır. Bylece optimizasyon sreci, mhendislik kararlarını sadece sayısal olarak deđil, yapısal anlamda da geerli kılmaktadır.

3.5.2. Kullanılan algoritmalar

eřitli alıřmalar, optimizasyon srecinin geleneksel metodolojiyle elde edilen sonulardan daha iyi sonular verdiđi sonucuna varmıřtır (Eleftheriadis et al., 2018; Ghandi et al., 2017; Paya-Zaforteza et al., 2009; Sahab et al., 2004). Yıllar iinde geliřtirilen tm optimizasyon tekniklerinin farklı bir konsepti ve alıřma mantıđı vardır. Bu yntemlerden bazıları Paracık Sr Optimizasyonu (PSO), Genetik Algoritma (GA), Harmoni Arama (HS), Tavlama Benzetimi (SA), vb. řeklinde-dir. Tm bu meta-sezgisel teknikler, problem formlasyonunun trne, karřılanması gereken kısıtlamalara ve deđiřken trne bađlıdır. Her ne kadar eřitli yntemlerin kendi iinde farklı olumlu ve olumsuz ynleri olsa da probleme gre optimizasyon ynteminin seimi ok nemlidir.

Meta-sezgisel algoritmalar genel olarak keřif ve smr olmak zere iki temel ařamada alıřır. Keřif ařaması, arama uzayının geniř bir blmn arařtırarak global optimuma ulařma potansiyeli tařıyan blgeleri belirlemeye odaklanırken; smr ařaması, daha nce keřfedilen iyi zm blgelerinde yođunlařarak zm kalitesini artırmayı amalar. Etkili bir algoritma, bu iki ařama arasında dengeli bir geiř sađlayarak hem eřitliliđi korur hem de yakınsama bařarımını artırır.

Bu alıřma kapsamında, PSO (Kennedy and Eberhart, 1995b), SA (Kirkpatrick et al., 1983), DE (Storn and Price, 1995), JAYA (Rao, 2016) ve IAOA (Ali Kaveh and Hamedani, 2022) algoritmaları kullanılmıřtır. Literatrde uzun sredir yaygın olarak uygulanan ve performansları eřitli problemler zerinde kapsamlı biimde test edilmiř olan parametrik algoritmalar arasında yer alan SA ve DE'nin yanı sıra, alıřmada bařlangıta klasik PSO ile bařlanmıř, ardından parametre adaptasyonu ile APSO (adaptif PSO) formuna dnřtrlerek geliřtirilmiř versiyonu kullanılmıřtır. Ayrıca, son yıllarda nerilen ve parametre ayarı gerektirmeyen JAYA ile IAOA algoritmalarının sz konusu problem zerindeki bařarımı da karřılařtırmalı olarak incelenmiřtir.

Parametrelı algoritmalar iin hiper parametre ayarlamaları, Bayes optimizasyonu yntemi kullanılarak gerekleřtirilmiř ve her bir algoritma iin probleme zg parametre konfigrasyonları elde edilmiřtir.

Tm algoritmalar, ok sayıda zmn eřzamanlı deęerlendirilmesini mmkn kılmak amacıyla paralel iřlemeye uyumlu řekilde geliřtirilmiřtir. Ařaęıda, her bir algoritmaya zg olarak gerekleřtirilen yapısal iyileřtirmeler, hiper parametre tanımlama yaklařımları ve uygulama detayları alt bařlıklar altında sunulmaktadır.

3.5.2.1. Adaptif paracık sr optimizasyonu algoritması (APSO)

Algoritmada her bir paracık, zm uzayında bir aday zm temsil eder ve bu paracıklar belirli bir hızla hareket ederek, hem kendi gemiř deneyimlerinden (kiřisel en iyi) hem de topluluęun genel bařarısından (global en iyi) etkilenerek pozisyonlarını gncellerler. Bu mekanizma sayesinde algoritma, arama uzayında hem keřif hem de smr dengesini kurmaya alıřır.

PSO algoritmasının temel parametreleri řunlardır:

- Atalet katsayısı (w): nceki hızın korunma derecesini belirler, keřif ve smr arasındaki dengeyi kontrol eder.
- Biliřsel katsayı (c_1): Paracıęın kendi en iyi konumuna ynelme eęilimini belirler.
- Sosyal katsayı (c_2): Paracıęın srdeki en iyi konuma ynelme eęilimini belirler.
- Hız sınırı (v_{max}): Paracıkların bir iterasyonda yapabileceęi maksimum sıramayı sınırlandırır.

Bařlangıta algoritma, literatrdeki klasik haliyle test edilmiř ancak uygun zmler elde edilememiřtir. Bu nedenle ilk olarak atalet katsayısı, belirli iki sınır deęer arasında, anlık iterasyon sayısına baęlı olarak adaptif řekilde deęiřtirilmiřtir. Ardından, algoritmanın deęiřken vektr ikili ve ayrık tamsayı paralar olmakzere ikiye ayrılmıř ve tm hiper parametreler bu iki blm iin ayrık biimde tanımlanmıřtır. Bu dzenlemelerin ardından, hiper parametreler iin Bayes optimizasyonu uygulanmıř ve algoritma geerli zmlerzretmeye bařlamıřtır.

Ancak sonraki gzlemlerde, algoritmanın ok sayıda iterasyon boyunca aynı zmzzerinde herhangi bir iyileřme saęlayamadan takılı kaldıęı tespit edilmiřtir. Bu durum, zm uzayında yerel minimumlara sıkıřma olasılıęının yksek olduęunu gstermektedir. Bu problemi ařmak amacıyla algoritmaya bir stagnasyon kontrol mekanizması entegre edilmiřtir. Sz konusu mekanizma, her bir paracık iin ardıřık

iterasyonlarda çözümünün değişmediği durumları takip etmekte ve on iterasyon boyunca herhangi bir gelişme gözlenmeyen çözümler için müdahale mekanizmasını devreye sokmaktadır. Bu bağlamda, stagnasyon süresi belirli bir eşiği aştığında, o çözüm üzerinde rastgele bir mutasyon işlemi gerçekleştirilerek parçacığın bulunduğu konumdan uzaklaştırılması sağlanmıştır.

Uygulanan mutasyon işlemi, çözüm vektöründeki bazı bileşenlerin düşük olasılıklarla yeniden örneklenmesi prensibine dayanmaktadır. Özellikle ikili ve ayrık değişkenlerin bulunduğu bu problemde, mutasyon işlemi her bir değişken türüne uygun olacak şekilde yapılandırılmıştır. Böylece algoritmanın keşif yeteneği artırılmış, uzun süreli durağanlık durumlarında çeşitlilik yeniden kazandırılmış ve yerel minimumlara takılma riski azaltılmıştır. Bu iyileştirmeler sonucunda yapılandırılan algoritma, bu düzenlemeler ile analizlerde etkin biçimde kullanılmıştır.

3.5.2.2. Diferansiyel evrim algoritması (DE)

Bu çalışmada DE/rand/1/bin stratejisine dayalı bir Diferansiyel Evrim algoritması uygulanmıştır. Optimizasyon vektörü, ikili (kolon var/yok) ve ayrık tamsayı (kesit ID'leri) parçalardan oluşmaktadır.

- **Başlatma:** Bireyler, ikili kısım için rastgele 0/1; ayrık kısım için tanımlı sınırlar arasında tamsayılarla oluşturulmuştur.
- **Mutasyon:** Üç farklı birey seçilip klasik DE fark vektörü oluşturulmuş; ikili kısım için XOR işlemi, ayrık kısım için yuvarlanmış fark kullanılmıştır.
- **Çaprazlama:** Binom çaprazlama uygulanmış, hem ikili hem ayrık kısımlarda jrand dahil olacak şekilde yeni aday çözüm üretilmiştir.
- **Seçim:** Aday çözüm, mevcut çözümden daha iyi ise kabul edilmiştir.
- **Paralellik:** Hesaplama süresini azaltmak amacıyla algoritma paralel işlemeye uygun biçimde yapılandırılmıştır.

Adaptif parametre güncellemeleri veya mutasyon kontrolü gibi ek stratejiler bu versiyonda kullanılmamıştır.

3.5.2.3. Tavlama benzetimi algoritması (SA)

Bu tez kapsamında kullanılan SA algoritması, klasik Tavlama Benzetimi yönteminin çoklu çözümle (popülasyon tabanlı) ve paralel işlemeye uygun biçimde yapılandırılmış bir versiyonudur.

Algoritma parametreleri şunlardır:

- **Tmax**: Başlangıç sıcaklığı
- **Tmin**: Son sıcaklık
- **FlipProb**: İkili parçada bit çevirme (bit-flip) işleminin uygulanma olasılığı
- **DiscProb**: Ayırık parçada yeni değer üretme olasılığı her bir çözüm için komşu üretimi iki aşamalı olarak gerçekleştirilmiştir.
- **İkili Parça**: Her bir bit, FlipProb olasılığıyla terslenmiştir (xor işlemi).
- **Ayrık Parça**: Her bir değişken, DiscProb olasılığıyla kesit havuzundan rastgele yeni bir değer alacak şekilde değiştirilmiştir.

3.5.2.4. JAYA algoritması

Bu çalışmada kullanılan Jaya algoritması, klasik Jaya'nın ikili ve ayırık tamsayı değişkenlere sahip yapısal tasarım problemlerine uyarlanmış versiyonudur. Optimizasyon vektörü, taşıyıcı sistemdeki kolonların varlık bilgisi için ikili (0–1) değerlerden ve kolon ile kiriş kesit tanımlarını temsil eden ayırık tamsayı değerlerden oluşmaktadır.

Her iterasyonda her bireyin çözümü, popülasyondaki en iyi ve en kötü bireyler referans alınarak aşağıdaki şekilde güncellenmiştir (Denklem 3.13):

$$x_{new} = x + r_1 \cdot (x_{best} - |x|) - r_2 \cdot (x_{worst} - |x|) \quad (3.13)$$

Bu denklemde rastgele olarak belirlenen r_1 ve r_2 vektörleri, yeni bireyin yönünü belirlemede kullanılmıştır. Bu işlemi takiben:

- İkili değişkenler, hesaplanan x_{new} değerlerine göre olasılık temelli olarak 0–1 değerlerine dönüştürülmüştür.
- Ayırık tamsayı değişkenler ise yuvarlanarak ve sınırlandırılarak geçerli kesit ID'lerine dönüştürülmüştür.

3.5.2.5. İyileştirilmiş aritmetik optimizasyon algoritması (IAOA)

Bu çalışmada, literatürde sunulan Improved AOA (IAOA) algoritması temel alınarak, taşıyıcı sistemin ikili ve ayırık değişkenlerden oluşan karma yapısını optimize edebilecek biçimde uyarlanmış bir sürümü geliştirilmiştir. Algoritma, AOA (Arithmetic Optimization Algorithm) ailesine ait olup, keşif ve sömürü süreçlerini kontrol parametreleriyle dengeleyerek çalışmaktadır.

Optimizasyon süreci boyunca her çözüm, belirli olasılıklara göre iki temel güncelleme stratejisiyle yenilenir:

- Çarpan temelli keşif stratejisi: Rastgele seçilen bir katsayı ile mevcut çözüm büyütülüp küçültülerek uzayda geniş arama yapılır (Denklem 3.14).

$$x_{new} = x \cdot (1 + \alpha \cdot rand() \cdot MOP)^{\pm 1} \quad (3.14)$$

Burada:

- x : mevcut çözüm,
- α : rastgele ± 1 yönünde büyüklük kontrolü sağlayan işaret katsayısı,
- $rand()$: $[0,1]$ aralığında rastgele sayı üretir,
- MOP: iterasyona bağlı olarak güncellenen adaptif arama katsayısıdır.
- En iyi çözüm temelli sömürü stratejisi: En iyi çözüm yönünde bir kayma uygulanarak yerel arama gerçekleştirilir (Denklem 3.15):

$$x_{new} = x_{best} \pm MOP \cdot x_{best} \cdot rand() \quad (3.15)$$

Burada:

- x_{best} : o ana kadar bulunan en iyi çözüm,
- MOP: çözümün ne kadar uzağına/yanına gidileceğini belirleyen adaptif katsayı,
- $rand()$: her çözüm için farklılaştırma sağlayan rastgele sayı.

Her çözüm için yukarıdaki iki stratejiden biri, MOA (Math Optimizer Accelerated) adlı bir parametreye bağlı olasılıkla seçilmektedir. MOA değeri iterasyon boyunca doğrusal biçimde artmakta ve böylece algoritmanın başlangıçta daha keşif odaklı, ilerleyen aşamalarda ise daha sömürü odaklı çalışması sağlanmaktadır (Denklem 3.16):

$$MOA = MOP_{min} + \frac{i}{maks_{iter}} \cdot (MOP_{max} - MOP_{min}) \quad (3.16)$$

Burada:

- i : güncel iterasyon sayısı,
- $maks_{iter}$: maksimum iterasyon sayısı,
- MOP_{min} , MOP_{max} : adaptif katsayının minimum ve maksimum sınırlarıdır.

Ayrıca her çözüm için uygulanan güncellemelerde MOP (Math Optimizer Probability) parametresi aşağıdaki gibi adaptif biçimde belirlenmektedir (Denklem 3.17):

$$MOP = 1 - \left(\frac{i}{maksiter} \right)^r \quad (3.17)$$

Burada:

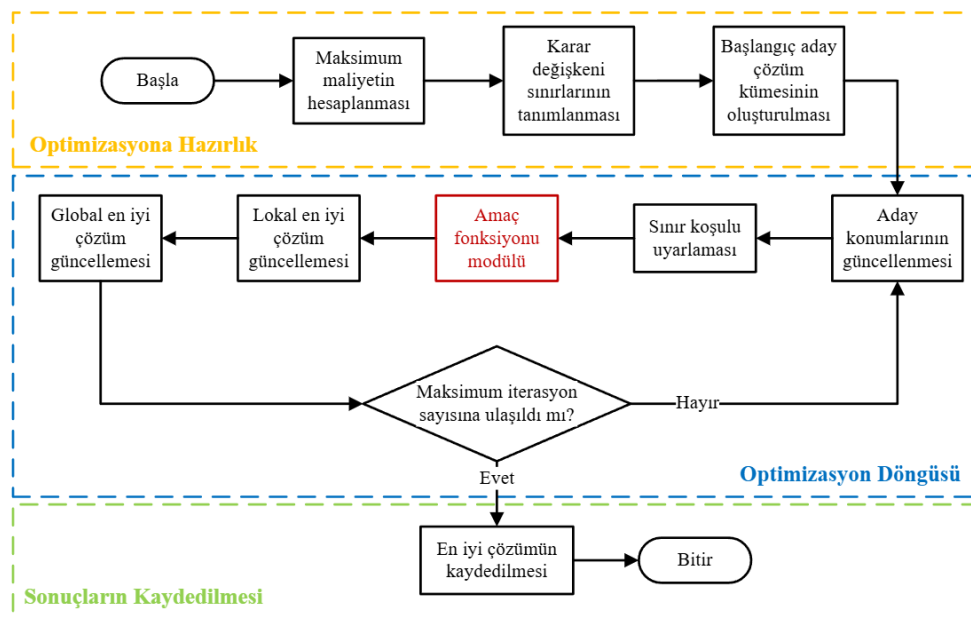
- r : her çözüm için rastgele seçilen bir üstel azalma katsayısıdır (genellikle 1 veya 2 olarak seçilir),

Bu tanım, algoritmanın başta daha geniş arama yapmasını, iterasyon ilerledikçe çözüm etrafında yoğunlaşmasını sağlar.

Bu mekanizma sayesinde IAOA, keşif ve sömürü süreçlerini zamana bağlı olarak kendiliğinden ayarlamakta, dolayısıyla çözüm uzayında dengeli bir şekilde ilerlemektedir.

3.5.3. Modül tasarımı

Geliştirilen yapı optimizasyon yazılımında, arama algoritmasının yapı modeliyle etkileşimi belirli bir sistematik dahilinde organize edilmiştir. Bu sistematik, çözüm uzayının taranması ve mühendislik açısından geçerli çözümlerin elde edilmesi sürecinde, algoritmanın ihtiyaç duyduğu tüm adımları belirli bir işlem sırasına bağlayan modüler bir tasarım mantığına dayanmaktadır. Şekil 3.23'te sunulan akış diyagramı, bu süreci üç temel aşamada tanımlamaktadır: Optimizasyona Hazırlık, Optimizasyon Döngüsü ve Sonuçların Kaydedilmesi.

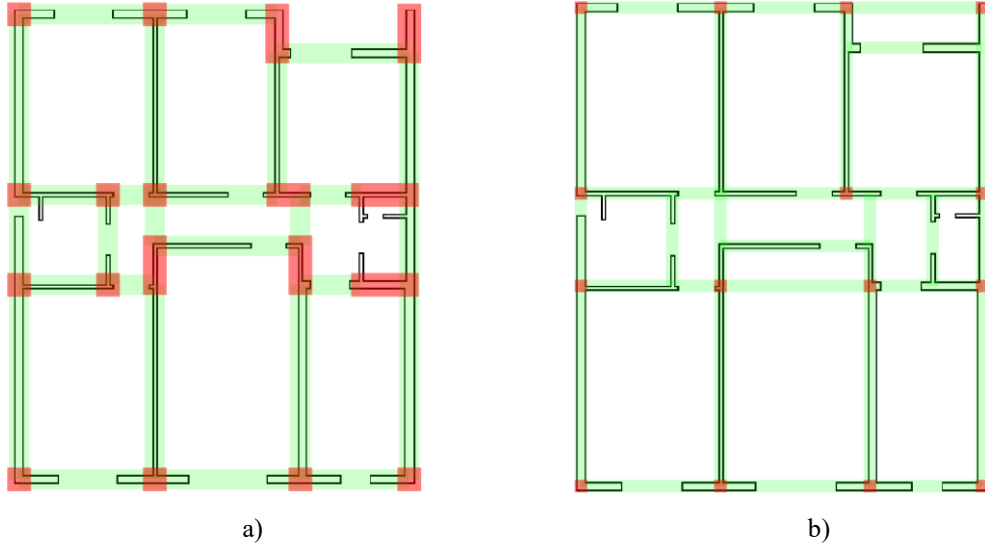


Şekil 3.23. Optimizasyon ile yapı tasarımı modülü akış diyagramı

Bu yapı sayesinde, algoritmanın çalışma süreci hem işlevsel olarak ayrıştırılmakta hem de yazılım modüllerinin bağımsız biçimde geliştirilmesine olanak tanınmaktadır. Her bir aşama, farklı görevleri yerine getiren alt modüllerden oluşmakta olup, bu modüller yazılımın geneline entegre biçimde çalışacak şekilde tasarlanmıştır. Özellikle aday çözümlerin değerlendirilmesi, güncellenmesi ve sonuçlarının yönetilmesi gibi kritik adımların birbirinden ayrılarak yapılandırılması, algoritmanın takip edilebilirliğini ve sürdürülebilirliğini artırmaktadır. Aşağıda her bir aşama detaylı olarak ele alınmıştır.

3.5.3.1. Optimizasyona hazırlık

Bu aşamada, görüntü işleme modülü ile elde edilen veriler kullanılarak optimizasyon sürecine hazırlık yapılmaktadır. İlk olarak, amaç fonksiyonu (3.12) hesabında kullanılmak üzere, üzerinde çalışılan kat planına ilişkin maksimum maliyet değeri belirlenmektedir. Bu değer, optimizasyon sürecinde tüm aday çözümlerin maliyet değerlerinin normalize edilmesinde referans noktası olarak kullanılmaktadır. Maksimum maliyetin belirlenmesi, çözüm uzayının en üst sınırını temsil eden teorik bir senaryonun modellenmesi ile gerçekleştirilmektedir. Bu amaçla, görüntü işleme sonucunda tespit edilen tüm kolon konumlarında kolonların mevcut olduğu kabul edilmekte ve her bir kolon ile giriş elemanına, ilgili yapı elemanına ait kesit havuzunda bulunan en büyük kesit alanına sahip kesit atanmaktadır. Bu konfigürasyon ile yapı modeli sonlu elemanlar yöntemi kullanılarak analiz edilmekte, analiz sonuçlarına göre donatı tasarımı yapılmakta ve nihai maliyet değeri hesaplanmaktadır. Örnek bir kat planına ait maksimum maliyet modeli ile optimizasyon süreci sonucunda elde edilen en uygun tasarım, Şekil 3.24'te birlikte sunulmaktadır.



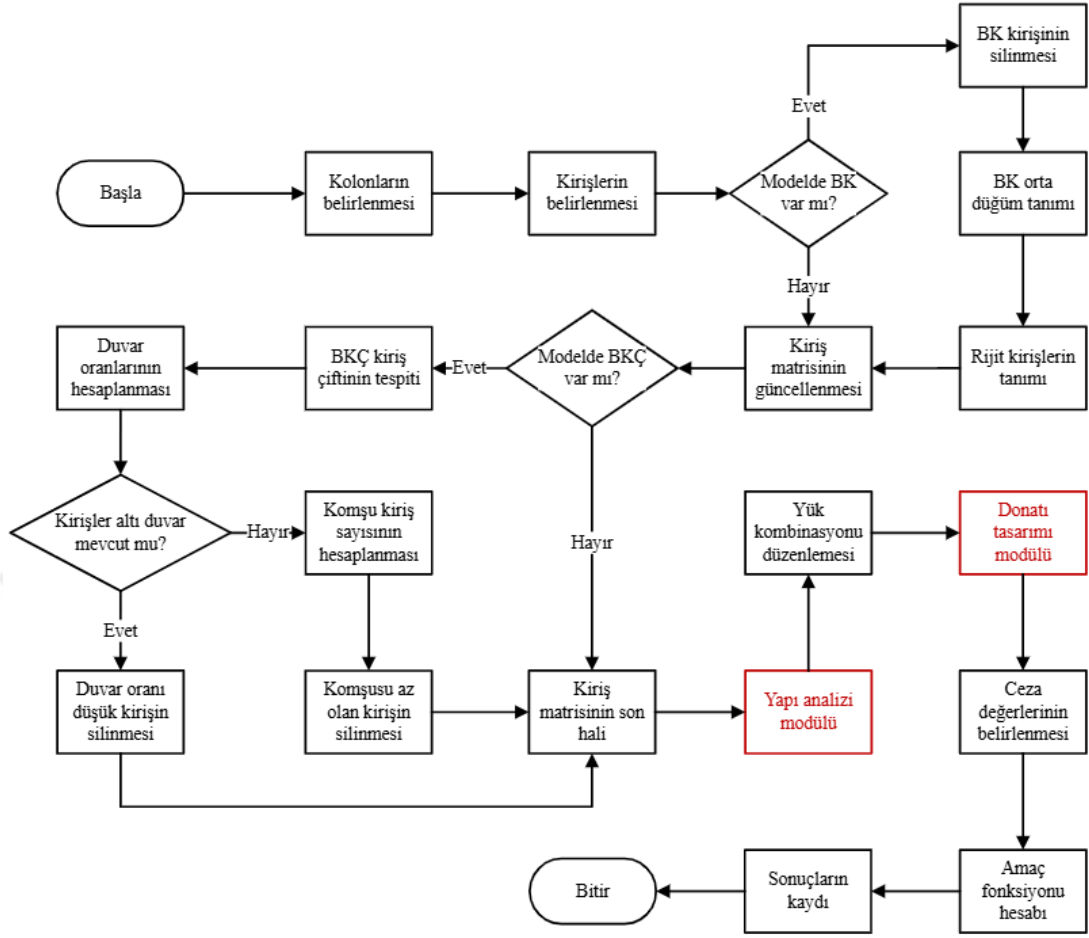
Şekil 3.24. Maksimum maliyet tasarım (a) ile optimum tasarım (b) modellerin görsel kıyası

Sonraki adımda, karar değişkenlerinin alabileceği minimum ve maksimum değerleri içeren sınır vektörleri oluşturulmaktadır. Bu vektörler, hem çözüm uzayının fiziksel ve geometrik kısıtlarını hem de kullanıcı tarafından sağlanan tasarım parametrelerini dikkate almaktadır. Böylece algoritmanın arama alanı, gereksiz veya fiziksel olarak imkansız kombinasyonlardan arındırılmakta, hesaplama yükü azaltılmakta ve optimizasyonun verimliliği artırılmaktadır.

Son olarak, belirlenen sınırlar dahilinde rastgele değerler kullanılarak başlangıç aday çözümler matrisi oluşturulmaktadır. Bu matris, optimizasyon algoritmalarının başlangıç popülasyonunu temsil etmekte ve sonraki iterasyonlarda iyileştirilecek çözüm adaylarının ilk versiyonlarını içermektedir. Böylece optimizasyon süreci, hem teorik hem de pratik olarak tanımlı bir başlangıç noktasından başlatılmaktadır.

3.5.3.2. Optimizasyon döngüsü

Bu aşama, algoritmanın çözüm uzayı içerisinde en uygun çözümü aradığı asıl yinelemeli süreci ifade etmektedir. Optimizasyon döngüsü, bir başlangıç popülasyonunun oluşturulmasının ardından, sezgisel algoritmaların karakteristik kurallarına bağlı olarak ardışık nesiller boyunca yeni çözüm adaylarının üretilmesi, değerlendirilmesi ve güncellenmesi şeklinde işlemektedir. Her algoritma için döngü yapısı farklılık gösterse de, temel mantık her yinelemede mevcut çözümlerden yola çıkarak daha iyi performansa sahip yeni çözümler oluşturmak üzerine kuruludur.

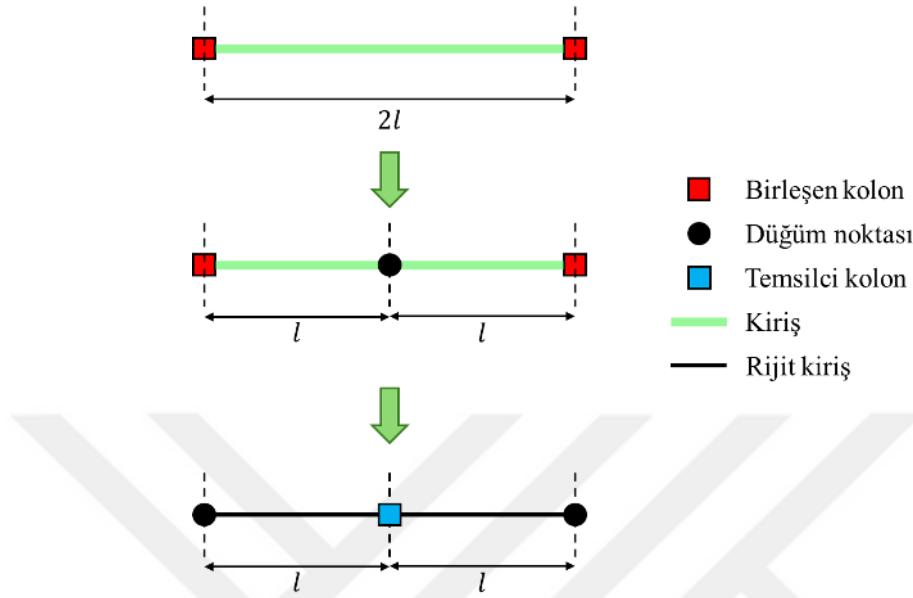


Şekil 3.25. Amaç fonksiyonu modülü akış diyagramı

Güncellenen aday çözümler, değerlendirme işlemleri için amaç fonksiyonu modülüne yönlendirilmektedir. Şekil 3.25'te bu modüle ait akış diyagramı sunulmakta olup, görselde birleşen kolonlar “BK”, birleşen kolon çiftleri ise “BKÇ” kısaltmaları ile gösterilmektedir. Bu aşamada ilk olarak, çözüm vektöründe yer alan kolon topolojisi bilgileri kullanılarak geçerli kolon matrisi, daha önce görüntü işleme ile elde edilmiş olası kolon matrisinden türetilmektedir. Ardından, bu kolonlara bağlı olan kirişler belirlenerek kiriş matrisi güncellenmektedir.

Ancak çözümde birleşen kolon veya birleşen kolon çifti bulunması durumunda, bu kiriş düzeninin yapısal modelleme açısından yeniden düzenlenmesi gerekmektedir. Çünkü birleşen kolonlar modelde doğrudan tanımlanmaz; bunlar, konumlarının geometrik merkezinde yer alan tek bir temsilci kolon ile temsil edilmekte, bu merkezden her bir birleşen kolon konumuna doğru tanımlanan rijit bağlantı elemanları aracılığıyla yapısal sisteme entegre edilmektedir (Şekil 3.26). Bu nedenle, birleşen kolon noktaları

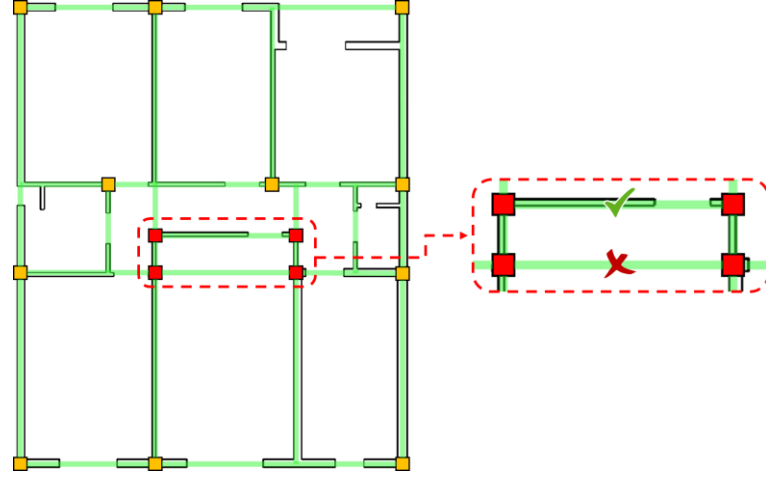
arasında daha önce tanımlanmış olabilecek bir kirişin modelde yer alması fiziksel olarak geçersiz hale gelmekte; dolayısıyla bu kiriş kiriş matrisinden silinmektedir.



Şekil 3.26. Birleşen kolonların yapısal analizde modellenmesi

Ayrıca oluşturulan temsilci kolon koordinatları ile rijit bağlantı noktalarına ilişkin bilgiler, yapı analizi modülünde kullanılmak üzere ayrı parametreler olarak kaydedilmekte ve modelin sonraki aşamalara sorunsuz biçimde aktarılması sağlanmaktadır.

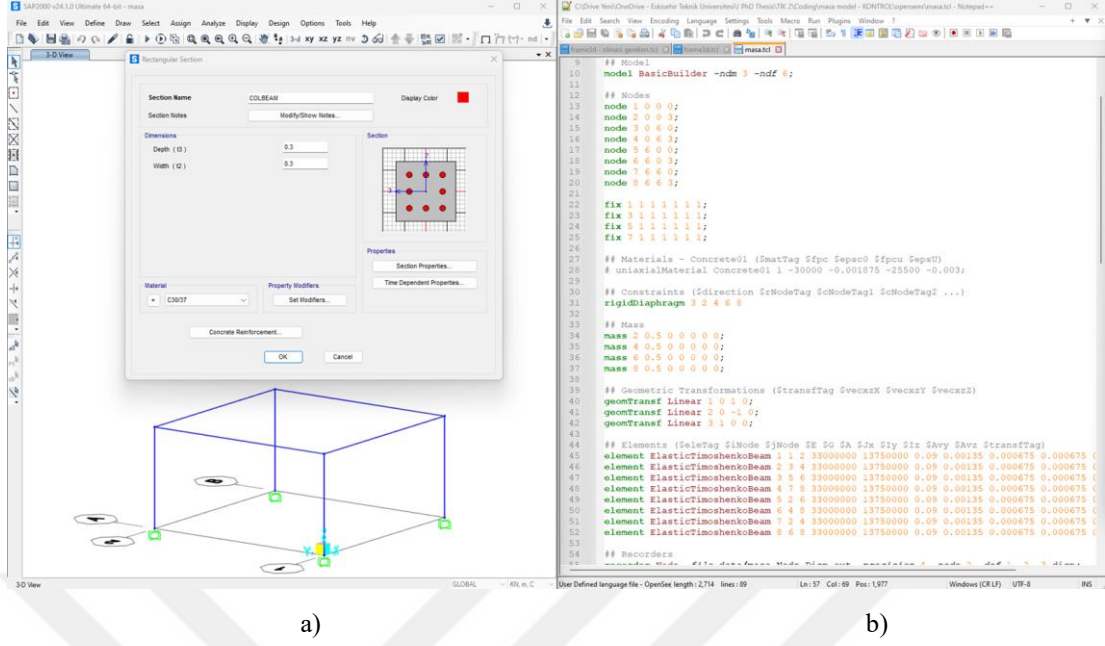
Öte yandan, birleşen kolon çiftleri arasında kalan kirişlerin tamamının yapıda yer alması istenmemektedir. Aksi halde model karmaşıklaşmakta ve gereksiz açıklıklar oluşmaktadır. Bu nedenle, söz konusu kirişlerden yalnızca birinin yapısal modelde yer alması hedeflenmiştir. Bu amaçla geliştirilen kontrol algoritmasında, birleşen kolon çiftleri arasında kalan kirişlerin altlarında duvar bulunup bulunmadığı incelenmekte; eğer en az birinin altında duvar varsa, bu duvarların uzunlukları üstlerindeki kirişlerin uzunluklarına oranlanmakta ve daha az orana sahip olan kiriş kiriş matrisinden çıkarılmaktadır (Şekil 3.27). Altında duvar bulunmayan iki kiriş söz konusu olduğunda ise, kirişlerin bağlandığı düğüm noktalarındaki komşu kiriş sayıları dikkate alınmakta ve daha az bağlantıya sahip olan kiriş sistemden elenmektedir. Bu işlemler sayesinde, yapısal anlamda geçerli ve sadeleştirilmiş bir model oluşturulması sağlanmakta, sonraki analiz sürecinde yalnızca mühendislik açısından anlamlı çözümler dikkate alınmaktadır.



Şekil 3.27. Örnek bir aday çözümde birleşen kolon çiftleri arası kirişlerde eleme işlemi

Tüm bu işlemlerin tamamlanmasının ardından geçerli kiriş matrisi oluşturulmakta, böylece aday çözümün sonlu elemanlar analizine uygun yapısal modeli oluşturulmuş olmaktadır. Bir sonraki aşamada bu model, yapısal analiz için ilgili modüle aktarılmaktadır.

Bu aşamada, önceki adımlarda oluşturulan kolon-kiriş sistemine ait tüm verilerin sonlu elemanlar modeline dönüştürülmesi gerekmektedir. Bu dönüşüm, yapısal analizlerin gerçekleştirileceği OpenSEES (Open System for Earthquake Engineering Simulation) yazılımının gereksinimlerine uygun olacak şekilde gerçekleştirilir. OpenSEES; SAP2000, ETABS ya da ProtaStructure gibi grafik arayüz tabanlı ticari yazılımlardan farklı olarak, tamamen metin tabanlı çalışan bir analiz platformudur. Bu yazılımda model tanımlamaları ve analiz komutları, Tcl (Tool Command Language) adı verilen bir programlama dili aracılığıyla yazılı komutlar şeklinde girilmektedir (Şekil 3.28).



Şekil 3.28. SAP2000 (a) ve OpenSEES'te (b) yapı modelleme sürecine ait kullanıcı arayüzü örnekleri

OpenSEES ile MATLAB tabanlı geliştirilen yazılım arasında veri aktarımını sağlamak üzere bir uygulama programlama arayüzü (Application Programming Interface, API) gereksinimi ortaya çıkmaktadır. Bu amaçla, açık kaynak kodları GitHub platformunda yayımlanmış olan OpenSeesAPI adlı MATLAB fonksiyonları entegrasyon sürecinde kullanılmıştır. Bu fonksiyonlar sayesinde, MATLAB ortamında oluşturulan yapı verileri, uygun Tcl sözdizimi ile otomatik olarak yazı formatına dönüştürülerek OpenSEES ortamında çalıştırılabilir hale getirilmektedir.

OpenSeesAPI'nin temel işlevi, kullanıcıdan aldığı sayısal giriş verilerine dayanarak bir Tcl betiği (script) oluşturmaktır. Sonlu elemanlar modeline ilişkin tüm bilgi girdileri (örneğin kesit boyutları, malzeme parametreleri, düğüm koordinatları, eleman bağlantıları vb.) bu fonksiyonlara sayı olarak verilmekte, daha sonra bu veriler Tcl dilinde yazı formatına dönüştürülmektedir. Bu süreçte sayısal verilerin yazıya çevriminde hassasiyet kritik bir rol oynamaktadır. Nitekim bazı küçük sayıların doğrudan sıfır olarak yazıya dönüştürülmesi sonucu model tanımlarında hatalar oluşabilmekte, bu da analiz sonuçlarının doğruluğunu doğrudan etkilemektedir. Bu sorunun önüne geçmek adına, çalışmada kullanılan bazı OpenSeesAPI fonksiyonları düzenlenmiş ve sayıdan yazıya çevrim süreci yeniden tasarlanmıştır. Bu sayede tüm modelleme adımları güvenilir biçimde yürütülmüştür.

The diagram illustrates a 3D lattice structure, likely a cubic lattice, with nodes and edges labeled with 3-digit numbers. The nodes are arranged in a 2x2x2 cube. The edges are labeled with 3-digit numbers, and the nodes are labeled with 3-digit numbers. A coordinate system (X, Y, Z) is shown at the bottom center.

Nodes (black dots) are labeled with 3-digit numbers:

- Bottom-left-front: 103
- Bottom-left-back: 101
- Bottom-right-front: 203
- Bottom-right-back: 201
- Top-left-front: 303
- Top-left-back: 301
- Top-right-front: 204
- Top-right-back: 302

Edges (lines) are labeled with 3-digit numbers:

- Vertical edges (left): 103, 203, 303
- Vertical edges (right): 102, 202, 302
- Horizontal edges (front): 1001, 2001, 3001
- Horizontal edges (back): 1002, 2002, 3002
- Diagonal edges (left): 1003, 2003, 3003
- Diagonal edges (right): 1004, 2004, 3004

A coordinate system (X, Y, Z) is shown at the bottom center, with X pointing right, Y pointing up, and Z pointing out of the page.

Yapının sonlu elemanlar modelinin oluşturulmasında, öncelikle sistematik isimlendirme yapısıyla birlikte, anlık değişken vektörü ve kullanıcı girdileri kullanılarak düğüm noktası matrisi oluşturulmaktadır. Ardından, global Z koordinatı en düşük olan düğüm noktaları tespit edilerek bu noktalara ankastre mesnet ataması yapılmakta ve buna uygun kısıt matrisi oluşturulmaktadır.

OpenSEES analiz sürecinde global rijitlik matrisinin doğru şekilde oluşturulabilmesi için her elemanın lokal eksen sisteminin global eksene göre tanımlanması gerekmektedir. OpenSEES, elemanın 1. lokal eksenini kendi içinde i-j düğüm noktaları arasında belirlerken, kullanıcıdan 3. lokal eksene karşılık gelen global vektörü istemektedir. Bu işlem, Bölüm 3.3.3'te detaylandırılan sistematığe uygun şekilde,

elemanın yönüne göre otomatikleştirilmiştir. Eksen kurgusu çeşitleri ayrıca numaralandırılarak eleman tanımlarında kullanılmaktadır.

Kesit özelliklerinin tanımlanabilmesi için kesit matrisi oluşturulmaktadır. Burada her bir kesit için gerekli geometrik ve mekanik parametreler hesaplanarak bu matrisle kaydedilmektedir. Aynı zamanda her elemanın eleman matrisine karşılık gelen kesit bilgisi ilişkilendirilmektedir.

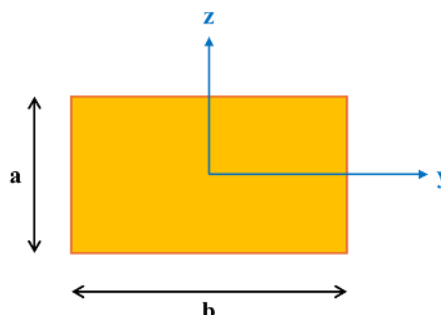
Birleşen kolonlar için özel bir kontrol mekanizması uygulanmaktadır. Uzun kenar/kısa kenar oranı 6'yı aştığında, perde tanımı oluşmaması için boyutlar otomatik olarak revize edilmektedir. Ayrıca bu kolonlar arasında tanımlanan rijit kirişlerin boyutları, kısa kenarı genişlik, kat yüksekliği ise derinlik olarak alınarak belirlenmektedir.

Yapıdaki tüm kolon ve kirişler OpenSEES'in ElasticTimoshenkoBeam (ETB) eleman modeli ile tanımlanmıştır. Bu model, kayma deformasyonlarını da içermesi nedeniyle SAP2000 gibi programlarla karşılaştırmalı analiz yapmaya olanak tanıyan gerçekçi bir çözüm sunar.

Her ETB eleman tanımı için aşağıdaki parametreler gereklidir:

- Eleman ismi
- i ve j düğüm noktalarının isimleri
- Elastisite modülü (E), kayma modülü (G), kesit alanı (A), burulma sabiti (J)
- Lokal eksenlere göre atalet momentleri (I_y , I_z) ve kesme alanları (A_{vy} , A_{vz})
- Lokal eksen sistemine ait isim

Tüm bu parametreler, oluşturulan düğüm, eleman ve kesit matrislerinden alınarak otomatik biçimde OpenSEES'e tanımlanır. Hesaplamalarda kullanılan formüller Şekil 3.30'da verilmiştir.



$$\begin{aligned}
 A &= ab \\
 A_{vy} &= A_{vz} = \frac{5}{6} A \\
 J &= a^3 b \left(\frac{1}{3} - \frac{0.21a}{b} \left(1 - \frac{a^4}{12b^4} \right) \right) \\
 I_z &= \frac{ab^3}{12}, I_y = \frac{ba^3}{12} \\
 G &= \frac{E}{2(1+\nu)}
 \end{aligned}$$

Şekil 3.30. Eleman tanımı için gerekli bilgilerin hesaplanmasında kullanılan formüller

Yükleme aşamasında, yapı modeline uygulanacak düşey ve yatay yüklerin tanımlanması gerçekleştirilmektedir. İlk olarak, kiriş elemanlarına yayılı yük şeklinde uygulanacak düşey yükler (sabit ve hareketli yükler) tanımlanmaktadır. Bu kapsamda, geliştirilen sistemde kullanıcıdan alınan alan yükü bilgileri, görüntü işleme modülü aracılığıyla elde edilen toplam kat alanı ile çarpılmakta; ardından bu değer, yapıda tanımlanan tüm kirişlerin toplam uzunluğuna bölünerek her bir kiriş için yayılı yük (kN/m) değeri elde edilmektedir. Böylece sabit ve hareketli yükler, OpenSEES modelinde doğrudan kirişlere yayılı yükler olarak aktarılmaktadır.

Yatay yükler kapsamında ise deprem etkileri dikkate alınmıştır. Bu amaçla öncelikle her bir katın kütesinin belirlenmesi gerekmektedir. Kat kütlesi hesaplamasında, standart olarak kullanılan $G + nQ$ formülü (TBDY 2018, Denklem 4.16) uygulanmaktadır. Burada:

- G : sabit yüklerin toplamı
- Q : hareketli yüklerin toplamı
- n : kullanıcı tarafından belirlenen hareketli yük kütle katılım katsayısı (TBDY 2018, Tablo 4.3)

Kat kütleleri, yukarıda belirtilen yük değerlerinin kat alanı ile çarpılması sonucu elde edilmekte ve yerçekimi ivmesine (g) bölünerek kütle birimine dönüştürülmektedir. Hesaplanan kat kütlesi, ilgili katta yer alan tüm düğüm noktalarına eşit olarak dağıtılarak her düğüme kütle ataması yapılmaktadır.

Yalnızca sabit yüklerin (G) tanımlandığı yükleme durumu altında yapı modeli için ayrıca bir modal analiz gerçekleştirilmiştir. Bu analizde, yapının her iki yatay yöndeki dinamik davranışını temsil eden modlar elde edilmekte; her mod için doğal frekans ve mod şekilleri hesaplanmaktadır.

Analiz sonuçlarından, her yatay yönde en yüksek kütle katılım oranına sahip modlar belirlenmekte (Şekil 3.31) ve bu modlara karşılık gelen hakim periyot (T) değerleri tespit edilmektedir (Şekil 3.32). Böylece yapı sisteminin dinamik özellikleri, özellikle deprem etkilerine karşı tasarım açısından kritik olan doğal periyotlar, sistematik şekilde elde edilmiştir.

* 9. MODAL PARTICIPATION MASS RATIOS (%):						
# The modal participation mass ratios (%) for each mode.						
#	MODE	MX	MY	MZ	RMX	RMZ
#						
1		83.935	0.00927847	0	0.00171727	14.5121
2		0.0748663	81.2605	0	14.9066	0.0119336
3		0.893477	2.85575	0	0.49912	0.124799
4		9.97432	0.000436475	0	0.00331557	68.8497
5		0.00597781	9.83475	0	64.7651	0.041335
6		0.0842051	0.343514	0	2.37425	0.6062
7		3.32478	2.62843e-05	0	2.57481e-05	8.12851
8		0.000965763	3.52021	0	8.43678	0.0028864
9		0.0269738	0.126264	0	0.303378	0.0931638
10		1.32762	0.000135504	0	0.000285401	6.51062
11		0.351546	4.59898e-05	0	0.000199408	1.11573
12		3.13147e-06	1.53983	0	7.00207	6.4558e-05

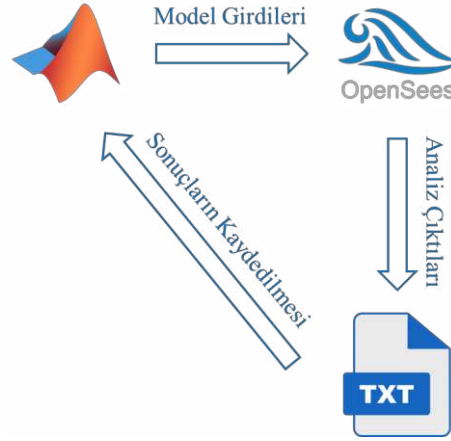
Şekil 3.31. Örnek bir modelin OpenSEES modal analiz çıktısında kütle katılım oranları

* 2. EIGENVALUE ANALYSIS:					
#	MODE	LAMBDA	OMEGA	FREQUENCY	PERIOD
#					
1		77.4101	8.7983	1.40029	0.714136
2		104.911	10.2426	1.63016	0.613437
3		132.241	11.4996	1.83022	0.546383
4		715.849	26.7554	4.25825	0.234838
5		995.953	31.5587	5.02273	0.199095
6		1226.27	35.0181	5.57331	0.179427
7		2060.9	45.3972	7.22518	0.138405
8		2995.26	54.7289	8.71038	0.114806
9		3538.89	59.4886	9.46791	0.10562
10		3934.11	62.7225	9.9826	0.100174
11		5731.36	75.7057	12.0489	0.0829948
12		6074.44	77.9387	12.4043	0.080617

Şekil 3.32. Örnek bir modelin OpenSEES modal analiz çıktısında periyot ve frekans bilgileri

Son aşamada, kullanıcıdan alınan deprem yükü parametreleri, belirlenen kat kütleleri ve modal analiz sonucu tespit edilen periyot değerleri kullanılarak TBDY (2018) Bölüm 4.7'deki eşdeğer deprem yükü yöntemiyle her kata uygulanacak toplam yatay kuvvetler hesaplanmaktadır. Bu kuvvetler, ilgili katın düğüm noktalarına eşit büyüklükte noktasal yatay yükler olarak dağıtılmakta ve model üzerine tanımlanmaktadır.

Yapı modellemesine dair gerekli tüm girdiler sağlandığında, oluşturulan Tcl dosyası, yine API içerisindeki bir başka fonksiyon aracılığıyla OpenSEES programında çalıştırılmaktadır. OpenSEES, diğer görsel yazılımlardan farklı olarak yalnızca kullanıcı tarafından özel olarak komut verilen çıktıları üretmektedir. Üstelik bu çıktılar, doğrudan analiz sonuçlarını içeren .txt formatındaki metin dosyalarıdır. Bu nedenle, analiz sonrası aşamada ihtiyaç duyulan veriler (örneğin iç kuvvetler, deplasmanlar vb.) MATLAB ortamına geri aktarılmakta ve ilgili veri işleme modüllerine yönlendirilmektedir (Şekil 3.33).



Şekil 3.33. Yapı analizinde OpenSEES ile MATLAB arasındaki veri alışverişi diyagramı

Yapı analizi optimizasyon sürecindeki her model için sadece G, Q, E_x ve E_y yükleri tanımlanarak ayrı ayrı dört analiz gerçekleştirilmekte ve sonuçlar kaydedilmektedir. Sonrasında Tablo 3.3'te verilmiş olan yük katsayıları ile farklı yükler altında oluşan düğüm noktası deplasmanları ve eleman iç kuvvetleri çarpılarak yük kombinasyonları sonuçları tespit edilmektedir.

Yapı analizi, optimizasyon sürecindeki her bir çözüm vektörü için yalnızca sabit yük (G), hareketli yük (Q) ve yatay deprem yükleri (E_x ve E_y) tanımlanarak ayrı ayrı dört yük durumu altında gerçekleştirilmiştir. Her analiz sonucunda oluşan düğüm noktası deplasmanları ve eleman iç kuvvetleri sistematik olarak kaydedilmiştir.

Bu veriler, TBDY (2018) uygun şekilde tanımlanmış olan yük kombinasyonları (Tablo 3.3) kapsamında değerlendirilmiştir. Bu aşamada, her yük birleşimi için ilgili yük durumu altında elde edilen iç kuvvet ve deplasman verileri belirli katsayılarla çarpılarak birleştirilmiş; böylece her yapı modeli için yönetmeliğe uygun yük kombinasyonları altındaki davranış elde edilmiştir.

Bu aşamada, yük kombinasyonlarına göre belirlenen analiz sonuçları kullanılarak TBDY (2018) Bölüm 4.9.1'de tanımlanan etkin görel kat ötelemeleri hesaplanmakta ve 4.9.2'de belirtilen ikinci mertebe etkilerin dikkate alınıp alınmayacağı kontrol edilmektedir. Gerekli şartların sağlanmaması durumunda, yapıdaki tüm iç kuvvetler ilgili yönetmelik hükmüne göre belirlenen oranda artırılmaktadır.

Buna ek olarak, yapıdaki burulma (A1) ve yumuşak kat (B2) düzensizlikleri TBDY 2018 Tablo 3.6 kapsamında değerlendirilmekte; tespit edilen düzensizlik türleri çıktı parametrelerine kaydedilerek kullanıcının kontrolüne sunulmaktadır.

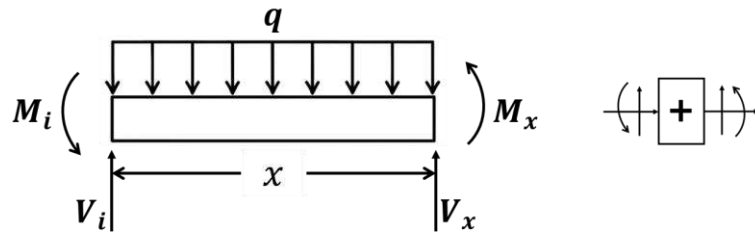
Bu süreç, yönetmelik kontrollerinin ve nihai taşıyıcı sistem performans değerlendirmelerinin doğru şekilde yapılabilmesini sağlamaktadır.

Sonraki aşamada, tüm yük kombinasyonlarına ait eleman iç kuvvetleri ile beton ve çelik malzeme bilgileri donatı tasarım modülüne iletilmekte; kolon ve kiriş donatı tasarımları bu modül kapsamında gerçekleştirilmektedir.

Kiriş donatı tasarımı süreci, üç aşamalı iç içe döngüden oluşan sistematik bir analiz yapısına sahiptir. En dıştaki döngü yapıdaki kat sayısı üzerinden ilerlemekte, ikinci döngü her katta yer alan tüm kiriş elemanlarını taramakta, en içteki döngü ise her bir kirişe ait tasarım momenti ve kesme kuvveti değerlerinin belirlenmesi amacıyla yük kombinasyonlarına karşılık gelen eleman iç kuvvetlerinin sırasıyla değerlendirilmesini kapsamaktadır. Bu kapsamda, her kiriş için yapı analizinden elde edilen uç noktalardaki moment ve kesme kuvveti değerleri esas alınarak gerekli tasarım girdileri oluşturulmaktadır.

Donatı tasarımı açısından, maksimum açıklık momenti bilgisi temel parametrelerden biri olarak değerlendirilmektedir. Ancak, kullanılan elastik analiz yaklaşımında OpenSEES yalnızca eleman uç noktalarındaki iç kuvvet ve moment değerlerini vermektedir. Bu nedenle, açıklık momentinin yaklaşık olarak belirlenebilmesi için, yapı analizi modülünde her bir yükleme durumu altında kirişlere etkiyen yayılı yük (q) değerleri kayıt altına alınmıştır.

Ayrıca, bu yükleme verileri ile yük kombinasyon katsayıları çarpılarak, her kombinasyon için temsili yayılı yük değerleri de hesaplanmış ve ayrı olarak saklanmıştır. Böylece yük kombinasyonlarına bağlı olarak açıklık momentinin analitik yöntemle hesaplanabilmesine olanak tanınmıştır.



Şekil 3.34. Kiriş boyuna donatı tasarımı için pozitif yön kabulleri

Şekil 3.34'te, bir kirişe düzgün yayılı yük uygulanması durumunda, kirişin i ucundan x mesafe uzaklıktaki bir kesitte oluşan kesme kuvveti ve momentin

hesaplanmasında esas alınan pozitif yön kabulleri gösterilmektedir. Bu yönler, OpenSEES yazılımının eleman iç kuvvetleri için tanımladığı pozitif yön kabulleri ile birebir uyumludur. Dolayısıyla, analitik hesaplamalar da bu kabullere göre yapılmakta ve elde edilen değerler, OpenSEES analiz sonuçlarıyla doğrudan karşılaştırılabilmektedir. Kiriş üzerinde düzgün yayılı yük (q) etkisinin var olduğu kabulüyle, herhangi bir kesitteki moment değeri, i ucundan olan mesafe x olmak üzere, kuvvet ve moment dengesi esas alınarak aşağıdaki şekilde ifade edilmektedir:

$$M(x) = V_i x - \frac{q x^2}{2} - M_i \quad (3.18)$$

Bu fonksiyonun türevi sıfıra eşitlenerek açıklıkta maksimum momentin olduğu nokta bulunmakta, elde edilen x değeri Denklem 3.18'de yerine yazıldığında Denklem 3.19'daki maksimum moment bağıntısına ulaşılmaktadır:

$$M_{max} = \frac{V_i^2}{2q} - M_i \quad (3.19)$$

Burada;

- x : kirişin i ucundan olan mesafedir,
- $M(x)$: mesafeye bağlı moment fonksiyonudur,
- V_i : kirişin i ucundaki kesme kuvvetidir (kN),
- M_i : i ucundaki momenttir (kNm),
- q : kirişe etkiyen yayılı yüküdür (kN/m).

Bu işlem, tüm yük kombinasyonu sonuçları taranarak yalnızca yayılı yükün sıfırdan farklı olduğu durumlar için gerçekleştirilmekte; yayılı yükün sıfır olduğu kombinasyonlarda açıklık momenti değeri sıfır olarak kabul edilmektedir.

Açıklık momentinin belirlenmesinin ardından, her bir kiriş için asal eksen boyu üç bölge tanımlanmakta (i ucu, açıklık, j ucu) ve bu bölgelerdeki moment değerleri sırasıyla M_i , M_{mid} ve M_j olarak hesaplanmaktadır. Bu hesaplamalar, tüm yük kombinasyonları dikkate alınarak gerçekleştirilmekte; her bölge için elde edilen maksimum moment değeri, ilgili bölgenin tasarım momenti olarak belirlenmektedir. Son aşamada ise, her bir kirişin bu üç bölgesine karşılık gelen üst, alt ve gövde donatı alanları, TS500:2000 ve TBDY 2018 yönetmeliklerinde tanımlanan kriterlere göre boyutlandırılmakta ve gerekli donatı alanı hesapları yapılmaktadır. Bu hesaplar (3.20)-(3.26)'da verilmektedir.

Topçu (2019)'ya göre kirişlerde aşırı sehim olmaması ve sehim hesabı istenmemesi durumu için,

$$\rho_l = \max(\rho - \rho') \quad (3.20)$$

olmak üzere,

$$\rho_1 = \rho_l = \frac{0.235 f_{cd}}{f_{yd}} \quad (3.21)$$

değeri uygun olmaktadır. Bu nedenle donatı oranını $\rho_l \leq 0.235 f_{cd} / f_{yd}$ ile sınırlandırarak ve TS 500:2000 Madde 13.2 deki kiriş yükseklik/açıklık oranlarına uyarak sehim hesabı yapılmayabilmektedir.

Buna göre,

$$M_1 = 0.235 f_{cd} b_w d^2 \left(1 - \frac{0.1175}{k_3} \right) \quad (3.22)$$

olarak belirlenmektedir. Bu durumda $M_1 \geq M_d$ ise tek donatılı tasarım, $M_1 < M_d$ ise çift donatılı tasarım gerçekleştirilmektedir.

$M_1 \geq M_d$ durumu için,

$$A_s = k_3 \frac{f_{cd}}{f_{yd}} a b_w \quad (3.23)$$

$$A'_s = 0 \quad (3.24)$$

ve $M_1 < M_d$ durumu için,

$$A_s = \rho_1 b_w d + \frac{M_d - M_1}{f_{yd}(d - d')} \quad (3.25)$$

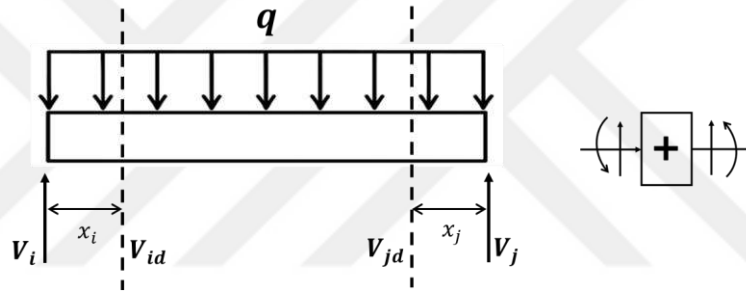
$$A'_s = \frac{M_d - M_1}{f_{yd}(d - d')} \cdot \frac{f_{yd}}{\sigma'_s} \quad (3.26)$$

olarak hesaplanmaktadır. Bu denklemlerde;

- ρ_1 : Çift donatılı kesite geçiş sınırı donatı oranını,
- f_{cd} : Beton tasarım basınç dayanımını,
- f_{yd} : Boyuna donatı tasarım akma dayanımını,
- M_1 : Çift donatılı kesite geçiş sınırı moment taşıma gücünü,
- b_w : Kiriş kesit genişliğini,
- d : Kiriş kesit derinliğini,
- d' : Kesit paspayını,
- k_3 : Eşdeğer basınç bloğu gerilme katsayısını,

- M_d : Kiriş tasarım momentini,
- A_s : Çekme bölgesindeki donatı alanını,
- A'_s : Basınç bölgesindeki donatı alanını,
- σ'_s : Basınç donatısındaki gerilmeyi ifade etmektedir.

Benzer şekilde, yapı analizi sonuçlarında yalnızca elemanların uç noktalarındaki kesme kuvvetleri elde edilebilmektedir. Ancak etriye donatısının boyutlandırılabilmesi için, kirişin bağlı olduğu kolonlardan itibaren $t/2+d$ uzaklıktaki kesme kuvvetlerinin bilinmesi gerekmektedir. Burada t , kolonun kiriş asal eksenine doğrultusundaki kesit boyutunu; d ise kirişin efektif derinliğini ifade etmektedir. Şekil 3.35'te, düzgün yayılı yük altındaki bir kirişte, kesme kuvveti hesaplarının yapıldığı noktalar ve bu hesaplarda kullanılan parametreler sunulmaktadır.



Şekil 3.35. Kiriş etriye tasarımı için pozitif yön kabulleri

Söz konusu noktalardaki kesme kuvvetlerini belirleyebilmek amacıyla, yalnızca düzgün yayılı yük etkisinde kalan kiriş elemanları için doğrusal eğilimli kesme diyagramı esas alınmış ve Denklem 3.27'de verilen formülasyona dayalı bir hesaplama yöntemi geliştirilmiştir.

$$\begin{aligned}
 m &= -\frac{V_i + V_j}{L_{kiriş}} \\
 V_{id} &= V_i + mx_i \\
 V_{jd} &= V_i + m(L_{kiriş} - x_j) \\
 V_d &= maks(|V_{id}|, |V_{jd}|)
 \end{aligned} \tag{3.27}$$

Burada;

- m : Kesme diyagramındaki eğimi,
- V_i : Kirişin i ucundaki kesme kuvvetini,
- V_j : Kirişin j ucundaki kesme kuvvetini,
- x_i : Kirişin i ucuna bağlı kolona göre $t/2+d$ mesafesini,

- x_j : Kirişin j ucuna bağlı kolona göre $t/2+d$ mesafesini,
- $L_{kiriş}$: Kiriş uzunluğunu,
- V_{id} : Kirişin i ucuna göre tasarım kesme kuvvetini,
- V_{jd} : Kirişin j ucuna göre tasarım kesme kuvvetini,
- V_d : Sonuç elde edilen kiriş tasarım kesme kuvvetini belirtmektedir.

Bu yöntem doğrultusunda, her yük kombinasyonu için ilgili mesafelerdeki kesme kuvvetleri hesaplanmakta ve her kiriş elemanı için en büyük değer tasarım kesme kuvveti olarak belirlenmektedir. Ayrıca, etriye tasarımında hem minimum hem de maksimum aralık değerleri sabit olarak tanımlanmıştır. Bu kapsamda, açıklık bölgelerinde maksimum etriye aralığı 20 cm, sıklaştırma bölgelerinde ise 10 cm olarak belirlenmiştir. Minimum etriye aralıkları ise açıklıkta 10 cm, sıklaştırma bölgelerinde 5 cm olarak sabitlenmiştir. Elde edilen bu değerler doğrultusunda, TS500:2000 ve TBDY 2018 yönetmelikleri uyarınca gerekli etriye alanları hesaplanarak donatı tasarımı tamamlanmıştır. Bu hesaplar Denklem 3.28 - 3.32'de verilmektedir.

$$V_{d,max} = 0.22 f_{cd} A_c \quad (3.28)$$

$$V_d \leq V_{d,max} \quad (3.29)$$

(3.29)'un sağlandığı durumda etriye tasarımına devam edilmekte; sağlanmadığı durumda ilgili kirişe ceza puanı uygulanmaktadır.

$$V_{cr} = 0.65 f_{ctd} A_c \left(1 + \gamma \frac{N_d}{A_c} \right) \quad (3.30)$$

Sonrasında (3.30) ile donatısız betonun kesme dayanımı hesaplanmaktadır. Kirişteki eksenel kuvvet (N_d) basınç ise $\gamma = 0.07$, çekme ise $\gamma = -0.3$ olarak alınmaktadır.

$V_d < V_{cr}$ durumu için,

$$\frac{A_{sw}}{s} = 0.3 \frac{f_{ctd}}{f_{ywd}} b_w \quad (3.31)$$

ve $V_d > V_{cr}$ durumu için,

$$\frac{A_{sw}}{s} = \frac{V_d}{f_{ywd} d} \quad (3.32)$$

denklemlerine göre etriye alanları hesaplanmaktadır. Bu denklemlerde;

- $V_{d,max}$: İzin verilen maksimum tasarım kesme kuvvetini,

- V_d : Tasarım kesme kuvvetini,
- f_{ctd} : Beton tasarım çekme dayanımını,
- f_{ywd} : Etriye tasarım akma dayanımını,
- A_{sw} : Etriye alanını,
- A'_s : Basınç bölgesindeki donatı alanını,
- A_c : Beton kesit alanını,
- s : Etriye aralığını ifade etmektedir.

Sonraki aşamada, taşıyıcı sistemde yer alan kolon elemanlarının donatı tasarımı gerçekleştirilmektedir. Bu işlem, eksenel yük ve moment etkisi altındaki betonarme kolon kesitlerinin donatı alanı gereksinimlerinin analitik çözümlerle doğrudan belirlenememesi nedeniyle, nümerik bir çözüm yöntemiyle yürütülmektedir. Bu kapsamda, literatürde Çakıroğlu ve Özer (1990) tarafından gerçekleştirilen çalışmada sunulan ve STA4CAD yazılımında uygulandığı bilinen yaklaşık hesap yöntemi, algoritmaya entegre edilmiştir. Söz konusu yöntem, her bir kolon elemanına ait eksenel kuvvet ve moment bileşenleri kullanılarak yaklaşık boyuna donatı alanını hesaplamakta ve böylece donatı tasarımı otomatik şekilde gerçekleştirilmektedir. Bu hesaplar Denklem 3.33 - 3.41'de sunulmaktadır.

$$n = \frac{N_d}{bh f_{ck}}, \quad m_x = \frac{M_{xd}}{bh^2 f_{ck}}, \quad m_y = \frac{M_{yd}}{b^2 h f_{ck}}, \quad \mu = \frac{A_s}{bh} \cdot \frac{f_{yk}}{f_{ck}} \quad (3.33)$$

olmak üzere, $n < 0.2$ ise,

$$m = m_x + km_y \quad (m_y \leq m_x) \quad (3.34)$$

$$k = (2 - 6,65n)m_y + 2,65n + 0,20 \quad (3.35)$$

$$\mu = 2,86m + 2,92n^2 - 1,48n \quad (3.36)$$

$n > 0.3$ ise,

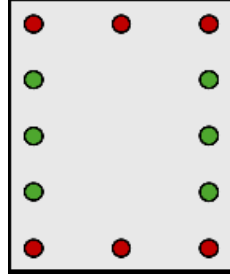
$$m = m_x + km_y \quad (m_y \leq m_x) \quad (3.37)$$

$$m = m_x + km_y \quad (m_y \leq m_x) \quad (3.38)$$

$$\mu = 2,92m + 0,62n^2 + 0,254n - 0,32 \quad (3.39)$$

olarak belirlenmektedir.

Bu çalışma kapsamında kolonların donatı tasarımı, kesitin dört kenarı boyunca eşit ve düzgün yayılmış tek bir donatı yerleşim türü esas alınarak gerçekleştirilmiştir (Şekil 3.36).



Şekil 3.36. Donatı yerleşim türünün gösterimi

Seçilen donatı yerleşim türüne göre k_1 sabiti Denklem 3.40'a göre belirlenmekte ardından kolon donatı alanı Denklem 3.41'e göre hesaplanmaktadır.

$$k_1 = 1.253885n^3 - 2.23048n^2 + 0.96059n + 1.144685 \quad (3.40)$$

$$A_s = k_1 \mu b h \frac{f_{ck}}{f_{yk}} \quad (3.41)$$

Kiriş elemanlarının tasarımından farklı olarak, kolonlarda donatı tasarımı yük kombinasyonlarının her biri için ayrı ayrı gerçekleştirilmekte; her kombinasyonda oluşan aksinel kuvvet ve moment çiftleri için ayrı tasarım yapılmaktadır. Sonuç olarak her kombinasyon için elde edilen donatı alanları arasında maksimum olan değer nihai tasarım değeri olarak kabul edilmekte ve ilgili kolona bu doğrultuda donatı ataması yapılmaktadır.

Kolon elemanlarında etriye alanı hesabı, kirişlerde izlenen hesaplama yöntemine benzer biçimde gerçekleştirilmiştir. Ancak kirişlerden farklı olarak, kolonlardaki etriye maliyeti hesaplanırken tüm kolon boyunun sıkılaştırma bölgesi kabul edildiği varsayılmış ve buna göre sabit, yoğun etriyeli bir donatı yerleşimi esas alınarak maliyet hesaplamaları yapılmıştır.

Yapı elemanlarına ait donatı ve etriye alanlarının belirlenmesinin ardından, o anki yapı modeline ait toplam maliyet hesaplaması gerçekleştirilmektedir. Beton maliyeti, her bir elemanın boyu ve kesit boyutları esas alınarak hacim hesabı yapılmakta ve birim hacim beton maliyeti (TL/m³) ile çarpılarak elde edilmektedir. Donatı ve etriye maliyetlerinin hesaplanmasında ise; donatılar için eleman uzunluğu, etriyeler için ise kesit boyutları ve pas payı dikkate alınarak toplam donatı uzunlukları belirlenmekte; ardından donatıların birim ağırlıkları ve birim ton başına maliyetleri (TL/ton) kullanılarak toplam donatı maliyetleri hesaplanmaktadır.

Amaç fonksiyonu modülünde, son adım olarak anlık yapı modeline ait toplam amaç fonksiyonu değerinin hesaplanabilmesi için, modelin maruz kaldığı ceza (penaltı)

durumlarının tespiti gerekmektedir. Bu çalışma kapsamında, yapı tasarımının güvenliğini ve yönetmeliklere uygunluğunu sağlamak amacıyla yedi farklı penaltı türü tanımlanmıştır. Bu penaltılar aşağıda sıralanmaktadır:

- **LongPen** (Longitudinal Penalty): Kiriş veya kolon elemanlarında boyuna donatı tasarımı sırasında, maksimum donatı oranı sınırının aşılması yani kesit boyutlarının yetersiz kalması durumunda ilgili elemandan dolayı bir ceza puanı uygulanır.
- **StirPen** (Stirrup Penalty): Etriye tasarımında, hesaplanan tasarım kesme kuvvetinin maksimum kesme kapasitesini aşması halinde ((3.29), kesitin yetersiz olduğunu gösteren bir ceza puanı devreye girer.
- **BDPen** (Beam Deflection Penalty): TS500:2000 Çizelge 13.1’de verilen minimum kesit yüksekliği/açıklık oranının sağlanmaması durumunda, elastik sehim hesabı yapılmasının zorunlu hale gelmesi nedeniyle, ilgili kirişe ceza puanı uygulanır.
- **CDPen** (Column Ductility Penalty): Kolon sünekliğini garanti altına almak amacıyla, TS500:2000 Denklem 7.7 ve TBDY 2018 Madde 7.3.1.2’ye göre sınırlandırılan eksenel yük oranının aşılması halinde ceza puanı verilir.
- **CBCPen** (Column-Beam Compatibility Penalty): Kolon kesitinin, bağlandığı kiriş kesitinden daha dar olması durumunda, birleşim bölgesinde oluşabilecek dayanım ve süneklik sorunlarını önlemek amacıyla kolon elemanına ceza puanı verilir.
- **DRPen** (Drift Ratio Penalty): Herhangi bir katta göreceli kat ötelemesinin %2’yi aşması durumunda, her ihlal için yapı modeline bir ceza puanı uygulanır.
- **SBPen** (Secondary Beam Penalty): Kullanıcı tarafından tanımlanan sınır değer üzerinde birincil bir kirişe bağlanan saplama (ikincil) kiriş sayısı bulunması halinde, sınırı aşan her bir saplama kiriş için bir ceza puanı verilir.

Her bir penaltı türü, ilgili yönetmelik maddeleri ve yapısal tasarım ilkeleri doğrultusunda ayrı ayrı kontrol edilmekte; herhangi bir ihlal durumunda, modelin toplam amaç fonksiyonu değerine belirli bir ceza katsayısı eklenmektedir. Bu sistematik yaklaşım, yalnızca ekonomik çözümlerin değil; aynı zamanda güvenli, yönetmeliklere uygun ve mühendislik açısından geçerli yapıların teşvik edilmesini sağlamaktadır.

Sonuç olarak, normalize edilmiş amaç fonksiyonu değeri, yapı modelinin toplam maliyeti ile yukarıda açıklanan ceza puanlarının toplanmasıyla elde edilmekte; böylece her birey (tasarım alternatifi) yapı güvenliği ve uygunluğu açısından değerlendirilmiş olmaktadır.

Optimizasyon süreci boyunca bu işlem her nesilde yinelenmekte; algoritma, her yinelemede daha uygun çözümler üretmeye yönlendirilmektedir. Bu döngüsel yapı sayesinde, maliyet, yönetmelik uygunluğu ve yapısal güvenlik gibi çok sayıda kısıt ve kriteri eşzamanlı dikkate alan çok kriterli bir tasarım süreci yürütülmektedir. Böylece geliştirilen optimizasyon algoritması, ayrık çözüm uzayı ve karmaşık karar değişkeni yapısına sahip olan bu problem için mühendislik bağlamında uygulanabilir, bütüncül ve optimum tasarımlar üretmektedir.

3.6. Paralel İşleme

Optimizasyon algoritmalarının paralel işlemeye uygun yapıda olmaları, çözüm sürecinde kullanılan bilgisayarın donanım özelliklerini doğrudan etkili hale getirmektedir. Bu bağlamda, işlemcinin çok çekirdekli paralel işlem performansı ile bellek kapasitesi ve erişim hızı, analiz süresi üzerinde önemli kazanımlar sağlamaktadır.

Bu çalışmada kullanılan bilgisayarın temel donanım özellikleri aşağıda verilmiştir:

İşlemci: AMD Ryzen 5 5600X 3.70 GHz 6 çekirdek

RAM kapasitesi ve hızı: 32 GB, 3200 MT/s

Çalışmada tüm optimizasyon algoritmaları, paralel işlemeye uyumlu olacak şekilde yapılandırılmıştır. Ayrıca, her bir algoritma için paralel işlemde kullanılan donanıma en uygun çekirdek sayısı, karşılaştırmalı bir performans analizi yoluyla belirlenmiştir. Bu analizler sonucunda, hızlandırma oranları ve analiz verimi hesaplanmıştır.

Hızlandırma oranı, seri analiz süresinin paralel analiz süresine oranı olarak tanımlanmakta; analiz verimi ise, hızlandırma oranının kullanılan çekirdek sayısına bölünmesiyle elde edilmektedir.

Bu parametre, paralel analizde kullanılan her bir çekirdeğin, analiz süresinin kısaltılmasına ne oranda katkı sağladığını ifade etmektedir. Bu sayede, donanımın verimli kullanım düzeyi analitik olarak değerlendirilebilmekte ve optimizasyon süreci donanım özelliklerine duyarlı hale getirilebilmektedir. Paralel işlemeye ilişkin tez kapsamında ulaşılan sonuçlar ise Bölüm 4.2.1’de ayrıntılı olarak sunulmaktadır.

3.7. Çıktıların İşlenmesi ve Görselleştirilmesi

Optimizasyon süreci sonucunda elde edilen verilerin sistematik biçimde değerlendirilmesi ve yorumlanabilmesi amacıyla çıktıların işlenmesi ve görselleştirilmesine yönelik çeşitli yöntemler uygulanmaktadır. Her bir analiz sonunda elde edilen yapı modeli, maliyet, ceza puanları, kolon ve kiriş kesit bilgileri, donatı alanları ve yapı analiz verileri gibi kritik parametreler düzenli biçimde kaydedilmektedir. Bu veriler hem sayısal tablolar halinde hem de grafiksel temsillerle kullanıcıya sunulurken, model performansının farklı kriterler açısından karşılaştırmalı biçimde incelenmesine olanak tanınmaktadır.

Ayrıca MATLAB ortamında geliştirilen özel arayüzler ile her bir optimizasyon çalışmasına ait çıktıların bütüncül biçimde görsel temsili oluşturulmuştur. Bu arayüzler; optimizasyonun seyrini, yapı modelinin geometrik yerleşimini, analiz çıktılarının dağılımını ve performans göstergelerini tek bir ortamda bir araya getirerek kullanıcıya tasarım kararlarını denetleme ve yeniden değerlendirme imkanı sağlamaktadır. Bu bağlamda elde edilen görselleştirmeler, yalnızca sonuçların yorumlanmasını kolaylaştırmakla kalmayıp, aynı zamanda optimizasyon algoritmalarının yapısal tasarım problemlerine uygulandığında nasıl bir çözüm ürettiğine dair detaylı içgörüler de sunmaktadır.

4. ANALİZ SONUÇLARI VE TARTIŞMA

Bu bölümde, görüntü işleme tabanlı otomatik kolon yerleşimi ile başlayan ve sezgisel optimizasyon algoritmalarıyla tamamlanan tümleştirilmiş sistemin farklı konfigürasyonlar altında ürettiği sonuçlar değerlendirilmiştir. Yapı analizinden tasarıma uzanan sürecin her adımında sistemin performansı, kararlılığı ve mühendislik açısından geçerliliği incelenmiştir.

Çalışma kapsamında, farklı beton sınıfları, kat sayıları ve deprem senaryoları ile çeşitli yapı modeli varyasyonları oluşturulmuş ve bu varyasyonlar üzerinde farklı optimizasyon algoritmaları denenmiştir. Kullanılan algoritmalar (APSO, DE, SA, JAYA ve IAOA) arasından en iyi performans gösterenler belirlenerek, yalnızca bu algoritmalar daha ileri analizler için seçilmiştir.

Analizler, yalnızca en iyi çözüme ulaşan algoritmaların sömürü kapasitesini değil; aynı zamanda bu çözümlere ulaşma süresini, yakınsama eğilimlerini ve yapı modeli karmaşıklığına (örneğin kat sayısı) karşı gösterdikleri dayanıklılığı da kapsamaktadır. Özellikle iterasyon eğrileri ve minimum maliyet çıktıları üzerinden yapılan değerlendirmeler, algoritmaların başlangıçtaki arama başarısı ile sonlardaki sömürü etkinliği arasındaki farkları göz önüne sermiştir.

Ayrıca, optimizasyon sonucu elde edilen yapı modellerinin taşıyıcı sistem tasarımı (kolon ve giriş kesitleri, donatı alanları), deprem performansı (görelî ötelemeler, iç kuvvet dağılımları, düzensizlik tespiti) ve yönetmelik uygunluğu gibi mühendislik parametreleri de analiz edilmiştir. Tüm bu bilgiler, kullanıcıya doğrudan erişebileceği şekilde sistem içinde yapılandırılmış ve kaydedilmiştir.

Son olarak, optimizasyon süreci ve sonuç yapıların görselleştirilmesi amacıyla sistemde hem yakınsama eğrisi hem de yapı planına gömülü kesit boyutlarını gösteren görsel çıktılar üretilmiştir. Bu görseller, kullanıcıların elde edilen yapısal tasarımları doğrudan değerlendirmesine olanak tanımakta, görsel kontrol yoluyla sezgisel doğrulama sağlamaktadır.

4.1. Görüntü İşleme

Çalışmanın görüntü işleme aşamasının tutarlı ve kararlı biçimde çalıştığının doğrulanabilmesi amacıyla iki farklı kat planı üzerinde uygulamalar gerçekleştirilmiştir. Bu planlar, Plan 1 ve Plan 2 olarak adlandırılmakta olup, her biri görüntü işleme modülünün işlevselliğini farklı mimari özellikler altında sınamak amacıyla seçilmiştir.

ve 500 cm uzunluklarında iki farklı referans çizgisi tanımlanmıştır. Bu sayede hem kısa hem de uzun çizgilerdeki hata sapmaları karşılaştırmalı olarak analiz edilmiştir.

Bu çerçevede, her iki planda hem yatay hem de düşey doğrultularda rastgele seçilen çeşitli uzunluktaki duvar çizgileri değerlendirmeye alınmış; bu çizgilerin yazılım tarafından hesaplanan uzunlukları ile çizim üzerindeki gerçek uzunlukları karşılaştırılmıştır. Elde edilen sonuçlar, Plan 1 ve Plan 2 için, iki farklı referans uzunluğu senaryosu altında detaylı olarak sunulmuştur (Tablo 4.1, Tablo 4.2, Tablo 4.3, Tablo 4.4). Bu tablolar, hesaplanan uzunlukların metrik farklarını ve yüzdesel hata oranlarını içermekte olup, görüntü işleme modülünün sayısal doğruluk düzeyi hakkında kapsamlı bilgi sağlamaktadır.

Tablo 4.1. Plan 1 için duvar çizgisi uzunlukları (ref: 100 cm)

Çizgi No	Yatay / Düşey	Gerçek Uzunluk (cm)	Hesaplanan Uzunluk (cm)	Metrik Fark (cm)	Hata Payı (%)
12	Yatay	30	29.23	0.77	2.57
34	Yatay	250	247.69	2.31	0.92
38	Yatay	10	10.00	0.00	0.00
42	Yatay	25	24.62	0.38	1.52
49	Yatay	140	139.23	0.77	0.55
2	Düşey	681	676.15	4.85	0.71
12	Düşey	70	69.23	0.77	1.10
23	Düşey	446	442.31	3.69	0.83
57	Düşey	45	44.62	0.38	0.84
59	Düşey	1207	1197.69	9.31	0.77

Tablo 4.2. Plan 1 için duvar çizgisi uzunlukları (ref: 500 cm)

Çizgi No	Yatay / Düşey	Gerçek Uzunluk (cm)	Hesaplanan Uzunluk (cm)	Metrik Fark (cm)	Hata Payı (%)
12	Yatay	30	29.33	0.67	2.23
34	Yatay	250	249.27	0.73	0.29
38	Yatay	10	10.26	0.26	2.60
42	Yatay	25	24.93	0.07	0.28
49	Yatay	140	139.30	0.70	0.50
2	Düşey	681	678.15	2.85	0.42
12	Düşey	70	69.65	0.35	0.50
23	Düşey	446	444.28	1.72	0.39
57	Düşey	45	44.72	0.28	0.62
59	Düşey	1207	1201.61	5.39	0.45

Tablo 4.3. Plan 2 için duvar çizgisi uzunlukları (ref: 100 cm)

Çizgi No	Yatay / Düşey	Gerçek Uzunluk (cm)	Hesaplanan Uzunluk (cm)	Metrik Fark (cm)	Hata Payı (%)
1	Yatay	20	19.83	0.17	0.85
2	Yatay	130	127.59	2.41	1.85
13	Yatay	370	362.07	7.93	2.14
57	Yatay	225	219.83	5.17	2.30
67	Yatay	60	59.48	0.52	0.87
1	Düşey	1490	1459.48	30.52	2.05
5	Düşey	60	58.62	1.38	2.30
32	Düşey	285	279.31	5.69	2.00
62	Düşey	20	19.83	0.17	0.85
68	Düşey	860	842.24	17.76	2.07

Tablo 4.4. Plan 2 için duvar çizgisi uzunlukları (ref: 500 cm)

Çizgi No	Yatay / Düşey	Gerçek Uzunluk (cm)	Hesaplanan Uzunluk (cm)	Metrik Fark (cm)	Hata Payı (%)
1	Yatay	20	20.25	0.25	1.25
2	Yatay	130	129.40	0.60	0.46
13	Yatay	370	369.72	0.28	0.08
57	Yatay	225	224.47	0.53	0.24
67	Yatay	60	59.86	0.14	0.23
1	Düşey	1490	1485.04	4.96	0.33
5	Düşey	60	59.86	0.14	0.23
32	Düşey	285	284.33	0.67	0.24
62	Düşey	20	20.25	0.25	1.25
68	Düşey	860	857.44	2.56	0.30

Referans çizgisi uzunluğunun, hesaplama doğruluğu üzerinde belirgin bir etkisi olduğu gözlemlenmiştir. Hem Plan 1 hem de Plan 2 için 500 cm uzunluğunda referans çizgisi kullanıldığında, metrik farklar ve hata oranları belirgin şekilde azalmıştır. Bu durum, daha uzun referans çizgilerinin piksel/cm oranını daha hassas belirlemesiyle açıklanabilir. Özellikle 10–60 cm aralığında olan kısa çizgilerdeki hata oranlarının 100 cm referansla %2.5 seviyesine kadar çıkabildiği; ancak 500 cm referansla bu oranın %0.3–0.5 seviyelerine düştüğü tespit edilmiştir.

Plan 2 örneğinde 370 cm uzunluğundaki bir yatay çizgi için hata oranı, 100 cm referansla %2.14 iken, 500 cm referansla %0.08'e düşmüştür. Benzer şekilde Plan 1'de 681 cm uzunluğundaki düşey çizgi için hata oranı, 100 cm referansla %0.71 iken, 500 cm referansla %0.42 seviyesine inmiştir. Bu veriler, uzun referansların dönüşüm katsayısını daha güvenilir şekilde tanımladığını ortaya koymaktadır.

Plan 1 ve Plan 2'nin hata dağılımı karşılaştırıldığında, referans çizgisi uzunluğuna karşı gösterdikleri hassasiyetin farklılaştığı görülmüştür. Her iki plan da yaklaşık olarak benzer piksel çözünürlüğünde elde edilmiş olmasına rağmen, geometrik yapılarındaki farklar sonucu etkilemiştir. Plan 1 daha kareye yakın bir düzende tasarlanmışken, Plan 2 dikey doğrultuda daha uzun ve dar bir yapıya sahiptir. Bu fark, özellikle düşey yönlü çizgilerdeki hata oranlarında kendini göstermektedir.

Plan 2'nin 100 cm referansla yapılan analizinde, 1490 cm uzunluğundaki düşey bir çizgide 30.52 cm metrik fark ve %2.05 hata payı elde edilmiştir. Aynı çizgi, 500 cm referansla analiz edildiğinde hata 4.96 cm'ye ve %0.33 oranına düşmüştür. Bu durum, Plan 2'nin referans çizgisi uzunluğu değişiminden daha fazla etkilendiğini ve uzun referans çizgilerine daha fazla ihtiyaç duyduğunu ortaya koymaktadır.

Her iki plan için referans çizgisi uzunluğu değiştiğinde oluşan maksimum ve minimum farklar Tablo 4.5'te özetlenmiştir.

Tablo 4.5. Minimum ve maksimum farklar

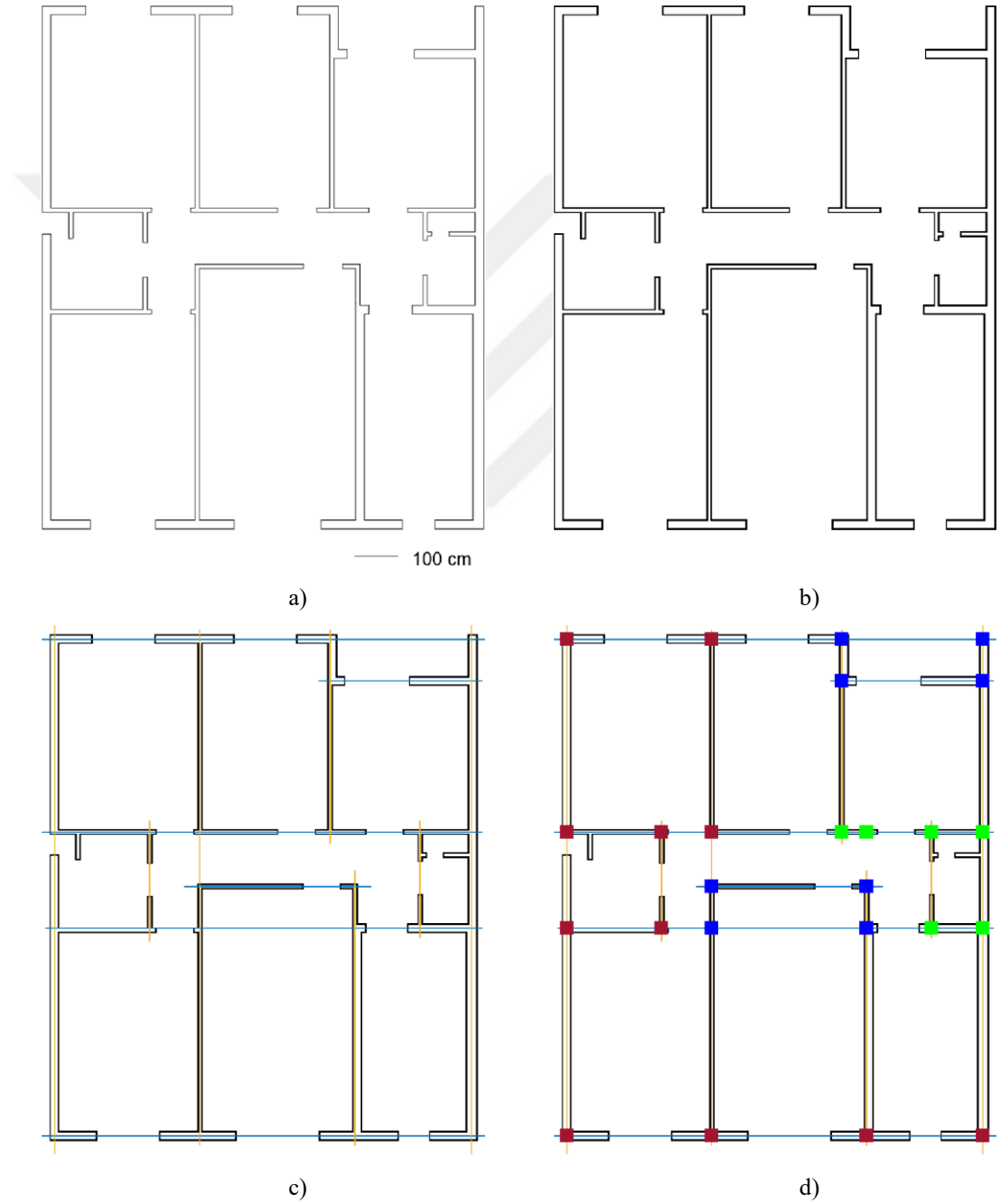
Plan	Referans	Metrik Fark Aralığı (cm)	Hata Payı Aralığı (%)
Plan 1	100 cm	0.00 – 9.31	0.00 – 2.57
Plan 1	500 cm	0.07 – 5.39	0.28 – 2.60
Plan 2	100 cm	0.17 – 30.52	0.85 – 2.30
Plan 2	500 cm	0.14 – 4.96	0.08 – 1.25

Plan 2'de 100 cm referansla yapılan ölçümlerde hataların daha yaygın ve yüksek olduğu; ancak 500 cm referansla birlikte bu hataların önemli ölçüde düştüğü görülmektedir. Bu durum, planın geometrik özelliğiyle birlikte çözünürlüğün yatay ve düşey doğrultuda farklı etkiler yaratabildiğini göstermektedir.

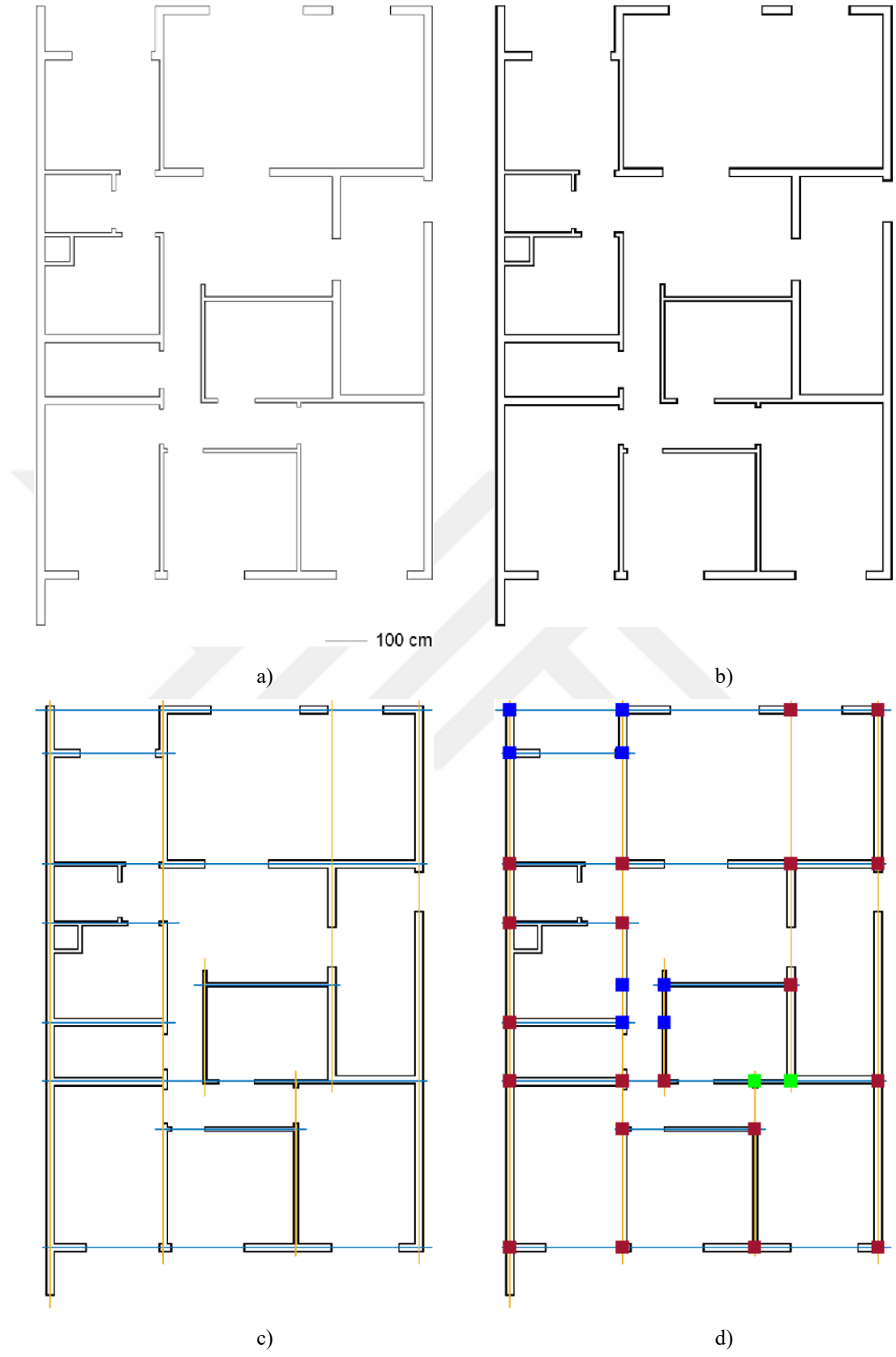
Bu farklılıkların ardından, yalnızca çizgi uzunluklarının değil, aynı zamanda çizgilerin oluşturduğu yapısal düzenin de doğru şekilde tanımlandığından emin olunabilmesi için görüntü işleme modülünün tüm işlem adımları detaylı biçimde incelenmiştir.

Bu kapsamda, analiz edilen Plan 1 ve Plan 2 üzerinde görüntü işleme süreci tam olarak uygulanmış ve her bir ara aşamanın çıktısı görsel olarak değerlendirilmiştir. Şekil 4.2 ve Şekil 4.3'te sunulan bu görseller, görüntü işleme modülünün işleyişini dört temel adım üzerinden temsil etmektedir: (a) kullanıcıdan alınan özgün plan görseli, (b) çizgi köşe düzeltmeleri uygulanmış ikili format, (c) tespit edilen aksların çizilmiş hali ve (d)

olası kolon konumlarının renkli noktalarla gösterildiği nihai durum. Kullanılan renk kodlamasında; kırmızı kareler normal kolonları, lacivert kareler düşey birleşen kolonları, açık yeşil kareler yatay birleşen kolonları ve koyu yeşil kareler algoritma tarafından aks sürekliliğini sağlamak amacıyla oluşturulan ek kolonları temsil etmektedir. Bu çok katmanlı görseller aracılığıyla görüntü işleme modülünün sadece çizgi bazlı değil, aynı zamanda yapısal eleman seviyesinde de doğru çalıştığı doğrulanmaktadır.



Şekil 4.2. Plan 1 için görüntü işleme sürecine ait a) kullanıcı girdisi, b) ikili format, c) aks çizimi ve d) kolon tespiti çıktıları



Şekil 4.3. Plan 2 için görüntü işleme sürecine ait a) kullanıcı girdisi, b) ikili format, c) aks çizimi ve d) kolon tespiti çıktıları

Çalışmanın görüntü işleme aşamasının tutarlı ve kararlı bir şekilde çalıştığının doğrulanabilmesi amacıyla, iki farklı kat planı üzerinde uygulamalar gerçekleştirilmiştir. Farklı mimari düzenlere sahip bu planlar üzerinden yürütülen analizlerde, kolon akslarının ve taşıyıcı sistem elemanlarının tespiti başarılı bir şekilde yapılmış; eksik ya da hatalı tanımlamalara rastlanmamıştır. Her iki plan için de görüntü işleme modülünün kolon yerleşimlerini doğru şekilde algıladığı ve ilgili matris yapılarını algoritmanın geri kalan adımlarıyla uyumlu olarak oluşturduğu gözlemlenmiştir. Bu sonuçlar, modülün farklı geometrik ve yapısal özelliklere sahip kat planlarında güvenilir biçimde çalışabildiğini ortaya koymakta; önerilen yöntemin esnekliğini ve genellenebilirliğini desteklemektedir.

4.2. Optimizasyon

Bu bölümde, geliştirilen sezgisel optimizasyon altyapısının farklı algoritmalarla uygulanması sonucu elde edilen performans verileri sunulmakta; algoritmaların arama davranışları, yakınsama eğilimleri ve maliyet minimizasyonundaki başarımları karşılaştırmalı olarak değerlendirilmektedir. Yapısal çözüm vektörünün hem ikili hem de ayrık tamsayı bileşenler içermesi nedeniyle kullanılan algoritmalar belirli yapısal uyarlamalara tabi tutulmuş ve problemin ayrık doğasına uygun şekilde yeniden yapılandırılmıştır. Bu kapsamda, Adaptif Parçacık Sürü Optimizasyonu (APSO), Diferansiyel Evrim (DE), Tavlama Benzetimi (SA), Jaya (JAYA) ve İyileştirilmiş Aritmetik Optimizasyon Algoritması (IAOA) gibi farklı arama stratejilerine sahip beş algoritma kullanılmıştır.

Parametre kontrollü algoritmalar için Bayes optimizasyonu yöntemiyle probleme özgü en uygun parametre setleri deneysel olarak belirlenmiş; tüm algoritmalar eşit popülasyon (100) ve iterasyon (300) sayılarıyla çalıştırılarak karşılaştırılabilir bir değerlendirme ortamı oluşturulmuştur. Optimizasyon algoritmalarının verimli ve güvenilir sonuçlar üretebilmesi yalnızca kendi iç yapılarına değil, aynı zamanda yürütüldükleri hesaplama ortamına da bağlı olduğundan; ilk olarak paralel işlem desteğinin analiz süresine etkisi test edilmiş ve uygun donanım yapılandırması belirlenmiştir.

Bu sürecin nihai amacı, probleme en uygun optimizasyon algoritmasını belirlemek olduğundan, her algoritma aynı kat planı ve yapısal parametreler (örneğin beton sınıfı, kat sayısı, deprem senaryosu) ile üçer kez çalıştırılmış ve her algoritmanın en iyi sonuçları

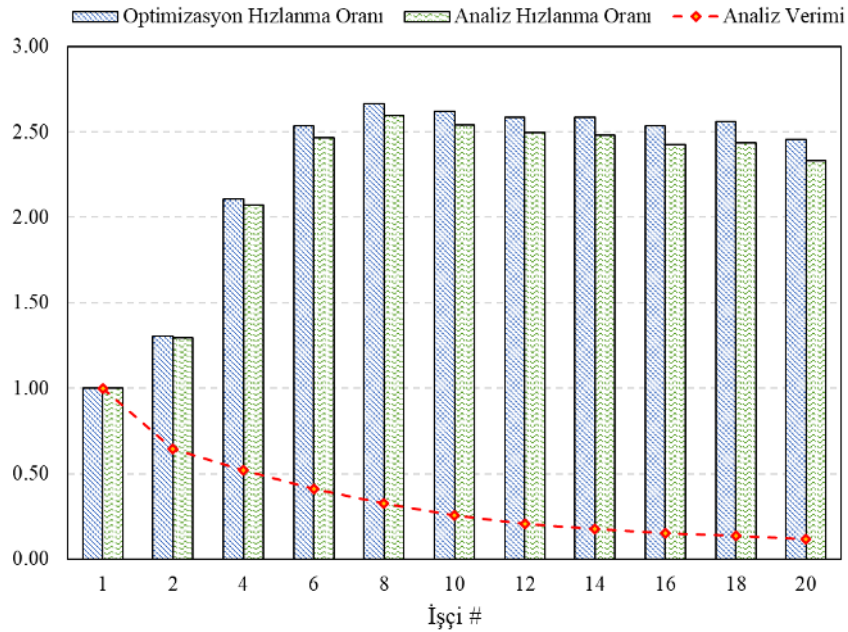
gözlemlenmiştir. Bu değerlendirme sonucunda başarılı bulunan algoritmalar, daha sonra gerçek yapı optimizasyonları için kullanılmıştır.

4.2.1. Paralel işleme

Çalışmada analizlerin yürütüleceği bilgisayarda bir karşılaştırmalı performans analizi gerçekleştirilmiştir. Farklı sayıda işçi (worker) ile aynı optimizasyon işlemi üçer kez çalıştırılmış ve her bir işçi sayısı için elde edilen sürelerin ortalaması alınarak, minimum analiz süresini sağlayan işçi sayısı belirlenmiştir.

Bu analizde, paralel işlem desteği yalnızca algoritmaların popülasyon döngülerinde uygulanmıştır (parfor yapıları). Gerçek analiz senaryosunu yansıtabilmesi amacıyla popülasyon büyüklüğü 100 birey olarak sabit tutulmuştur. Ancak toplam analiz süresinin makul düzeyde kalması ve bu benchmark testinin fazla zaman almaması amacıyla iterasyon sayısı 30 olarak sınırlandırılmıştır.

Buna ek olarak, seri analiz süresi referans alınarak, her işçi sayısında elde edilen paralel işlem süresiyle oranlanmak suretiyle analiz ve optimizasyon hızlanma oranları belirlenmiş; ayrıca, her işçi sayısı için analiz hızlanma oranları işçi sayılarına oranlanarak tanımlanan analiz verimi hesaplanmıştır. Elde edilen tüm bu sonuçlar, karşılaştırmalı olarak Şekil 4.4'te görsel biçimde sunulmuştur.



Şekil 4.4. Paralel işleme için karşılaştırmalı performans analizi sonuçları

Hızlanma oranları incelendiğinde, hem analiz hem de optimizasyon işlemleri seri analiz süresiyle karşılaştırıldığında, 6 ila 8 işçiye kadar artan işçi sayısı ile birlikte belirgin bir hızlanma göstermektedir. 8 işçide optimizasyon için hızlanma oranı 2.67, analiz için ise 2.60 olarak gözlemlenmiştir. Ancak bu noktadan sonra hızlanma oranı eğrileri yataylaşmakta, yani doygunluk seviyesine ulaşmaktadır. Bu durum, donanım kaynaklarının artırılmasının belirli bir noktadan sonra daha fazla kazanç sağlamadığını ve sistemin paralel işlem ölçeklenebilirlik sınırlarına yaklaşıldığını göstermektedir.

Ayrıca, işçi sayısı arttıkça optimizasyon ve analiz hızlanma oranları arasındaki fark da artmaktadır. Bu farkın temel nedeni, yalnızca işlemcinin işçilere görev dağıtımında daha fazla zaman harcaması değil; daha çok görevlerin paralel olarak planlanması, işçiler arası koordinasyonun sağlanması ve senkronizasyon işlemlerinin daha karmaşık hale gelmesidir. Özellikle parfor yapısında görevlerin tamamlanmasının beklenmesi gerektiğinden, senkronizasyon gecikmeleri hızlanma oranlarını sınırlayıcı bir etki yaratabilmektedir. Ayrıca, yük dengelemenin işçi sayısı arttıkça daha zor hale gelmesi de bu farklılaşmaya katkıda bulunmaktadır.

Öte yandan, analiz verimi metriği işçi sayısı arttıkça düşüş göstermektedir. Seri analizde verim 1.0 olarak kabul edilmiştir. Bu değer, 4 işçide 0.52, 10 işçide ise 0.25 seviyelerine düşmektedir. Bu durum, toplam analiz süresi azalsa bile kullanılan işlem kaynaklarının görece verimliliğinin düştüğünü; yani paralel işlemde elde edilen kazanımın marjinal olarak azaldığını göstermektedir. Verim düşüşü; işlemciler arası kaynak paylaşımı, eşzamanlılık kayıpları, veri aktarım gecikmeleri ve işçilerin boşta bekleme süreleri gibi paralel sistemlere özgü kısıtlamalardan kaynaklanmaktadır.

Sonuç olarak, 8 işçi sayısı hem hızlanma açısından en yüksek kazanımı sağlamakta hem de analiz veriminin henüz kabul edilebilir düzeyde kaldığı bir noktayı temsil etmektedir. Bu nedenle, yapı optimizasyon süreçlerinde kullanılacak paralel işlem konfigürasyonu olarak 8 işçi tercih edilmiştir.

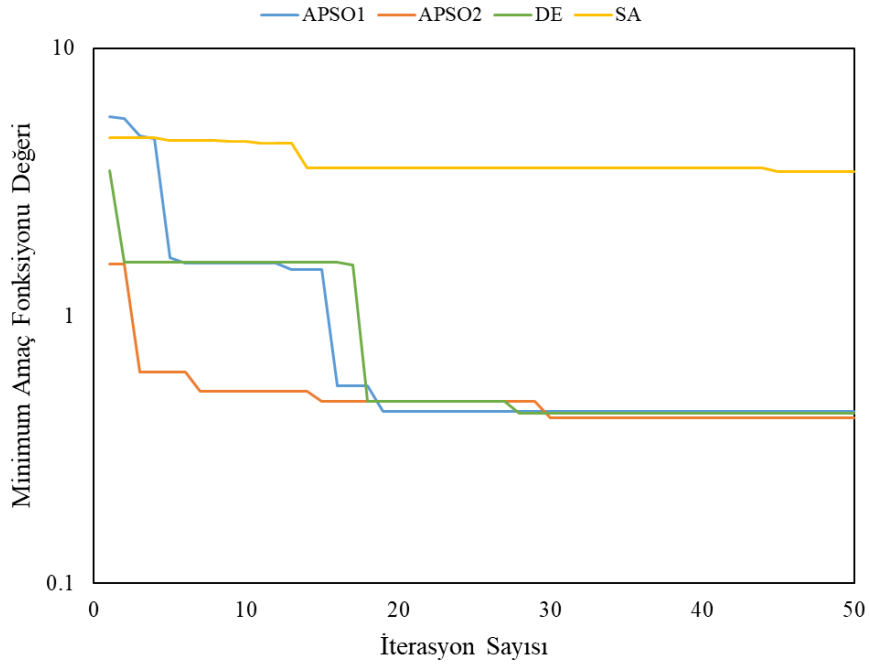
4.2.2. Hiper parametre optimizasyonu

Çalışma kapsamında kullanılan parametre kontrollü optimizasyon algoritmalarının (APSO, SA ve DE) probleme özgü başarımını artırmak ve yapı tasarımı açısından yönetmeliklere uygun çözümler elde edilmesini sağlamak amacıyla hiper parametre optimizasyonu gerçekleştirilmiştir. Bu amaçla, her bir algoritma için Bayes optimizasyonu yöntemi kullanılmıştır. APSO algoritması, iki farklı değişken vektörü

kurgusu (DV1 ve DV2) için ayrı ayrı uygulanmış (APSO1 ve APSO2) ve böylece her değişken vektörü yapısına özgü en uygun APSO parametreleri elde edilmiştir.

Bayes optimizasyonu sürecinde, her algoritma için 50 iterasyon üzerinden bir arama gerçekleştirilmiştir. Sürecin her bir iterasyonunda, ilgili algoritma 10 bireyden oluşan bir popülasyon ile 60 iterasyon boyunca çalıştırılmış ve elde edilen en iyi hedef fonksiyon değeri, Bayes optimizasyonu algoritmasına geri bildirim olarak verilmiştir. Böylece algoritmaların hiper parametre uzayında yönlendirilmiş ve verimli bir tarama sağlanmıştır.

Şekil 4.5'te, her bir algoritmanın Bayes optimizasyonu sürecine ait yakınsama eğrileri sunulmuştur. Bu grafiklerde, hiper parametre optimizasyon sürecinde her iterasyonda elde edilen en iyi hedef fonksiyon değerlerinin değişimi izlenerek, algoritmaların ayar hassasiyetleri ve genel yakınsama eğilimleri değerlendirilmektedir.



Şekil 4.5. Algoritmaların Bayes optimizasyonu yakınsama eğilimleri

Şekil 4.5 incelendiğinde, kullanılan algoritmaların hedef fonksiyon değeri açısından gösterdiği yakınsama eğilimleri karşılaştırmalı olarak görülmektedir. Deneme sürecinde popülasyon ve iterasyon sayılarının sınırlı tutulmuş olmasına rağmen, APSO1, APSO2 ve DE algoritmaları, yapı tasarımı açısından geçerli çözümlerin yer aldığı hedef

fonksiyon aralığına ulaşmayı başarmıştır. Buna karşın SA algoritması, Bayes optimizasyonu sürecinde bu bölgeye ulaşamamıştır.

Bu durum, algoritmaların probleme özgü yapıya gösterdiği uyum farkından kaynaklanabileceği gibi, SA algoritmasının Bayes optimizasyonundaki her denemede başlangıç konumlarının rastlantısal olarak daha düşük kaliteli bölgelerden seçilmiş olmasından da kaynaklanıyor olabilir.

Bayes optimizasyonu sonucunda elde edilen hiper parametre değerleri Tablo 4.6’da sunulmuştur. Bu parametreler, yazılımda kullanıcıdan bağımsız olacak şekilde sabitlenmiş ve tüm gerçek yapı optimizasyon testleri bu ayarlar altında gerçekleştirilmiştir.

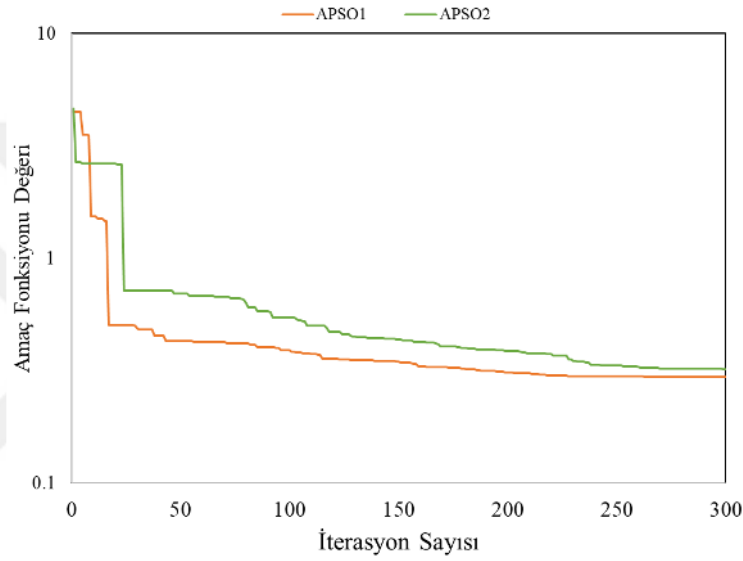
Tablo 4.6. Algoritmaların hiper parametre sonuçları

APSO1		APSO2		SA		DE	
$w_{b,1}$	1.20	w_1	0.92	T_{max}	2700.10	F	0.30
$w_{b,2}$	0.95	w_2	0.16	T_{min}	0.44	CR	0.13
$w_{d,1}$	1.07	c_1	1.10	FP	0.36		
$w_{d,2}$	0.18	c_2	2.01	DP	0.26		
$c_{1,b}$	1.02	v_{max}	6.00				
$c_{2,b}$	1.71	p_{mut}	0.03				
$c_{1,d}$	1.63						
$c_{2,d}$	1.37						
$v_{max,b}$	9.00						
$v_{max,d}$	1.00						
$p_{mut,b}$	0.01						
$p_{mut,d}$	0.01						

4.2.3. Algoritma karşılaştırması ve seçimi

Algoritmaların probleme özgü parametrelerle uyarlanması ardından, her biri aynı kullanıcı girdileriyle başlatılan optimizasyon süreçlerine tabi tutulmuştur. Bu kapsamda, değişken vektörü kurgularının sistem performansı üzerindeki etkisini değerlendirmek amacıyla APSO algoritması, iki farklı vektör yapısı olan DV1 (APSO1) ve DV2 (APSO2) ile ayrı ayrı uygulanmıştır. Her iki kurgu için üçer adet optimizasyon çalışması gerçekleştirilmiş ve elde edilen en iyi amaç fonksiyonu değeri APSO1 için 0.2968, APSO2 için ise 0.3210 olarak kaydedilmiştir. Buna ek olarak, Şekil 4.6’da görüldüğü üzere APSO1, APSO2’ye kıyasla daha hızlı bir yakınsama eğilimi sergilemiştir. Bu sonuçlar, çözüm uzayının tanımlanma biçiminin algoritma

performansını doğrudan etkilediğini göstermektedir. Kolon konumlarını ikili formatla temsil eden ve üç parçalı bir yapıya sahip olan DV1 kurgusu, yalnızca daha düşük amaç fonksiyonu değerleri üretmekle kalmamış, aynı zamanda çözüm geçerliliği ve hesaplama kararlılığı açısından da daha tutarlı bir performans sergilemiştir. DV2 yapısı genel başarı açısından DV1'in gerisinde kalmıştır. Bu nedenle, sistemin nihai uygulamasında DV1 yapısının kullanılması yönünde karar verilmiş ve tüm algoritmalar bu çözüm uzayı temel alınarak yapılandırılmıştır.



Şekil 4.6. Değişken vektörü kurgularının APSO algoritması özelinde kıyaslanması

APSO algoritmalarına ek olarak, diğer dört algoritmanın da aynı şekilde üçer tekrar ile çalıştırılmasıyla elde edilen en iyi amaç fonksiyonu değerleri Tablo 4.7'de sunulmuştur. Bu tabloda yer alan veriler, algoritmaların karşılaştırmalı performansını değerlendirmek ve en uygun olan üç algoritmayı belirlemek amacıyla kullanılmıştır.

Tablo 4.7. Algoritma test sonuçları

Beton Sınıfı	Kat #	Deprem	Algoritma Adı	Min AFD
C25	5	D1	JAYA	5.6606
			SA	0.4628
			APSO	0.2968
			DE	0.3254
			IAOA	0.2814

Tablo 4.7’de verilen test optimizasyon sonuçlarına göre APSO, DE ve IAOA algoritmaları, en düşük amaç fonksiyonu değerlerine ulaşarak probleme en uygun çözümleri üretmiştir. Bu üç algoritmanın, yapı tasarımı açısından hem yönetmeliklere uygunluk sağlayabilen hem de maliyeti minimize eden çözümler üretebildiği görülmektedir.

Öte yandan SA algoritması, cezasız çözümler bulabilmesine rağmen amaç fonksiyonu değeri bakımından daha yüksek sonuçlar üretmiş ve bu nedenle en iyi üç algoritma arasına girememiştir. JAYA algoritması ise testler boyunca çözüm uzayının cezalı bölgelerinden çıkamamış, dolayısıyla uygun ve geçerli yapı tasarımlarına ulaşamamıştır.

Gerçek yapı optimizasyonu çalışmaları, elde edilen test bulguları doğrultusunda yalnızca APSO, DE ve IAOA algoritmaları kullanılarak gerçekleştirilmiştir. Bu üç algoritma ile belirlenen bazı kullanıcı girdisi varyasyonları için ayrı ayrı analizler yapılmıştır. Kullanılan girdiler ve varyasyonları aşağıda özetlenmiştir:

- Beton sınıfı: C25 ve C40
- Deprem senaryosu: D1 (Eskişehir) ve D2 (Hatay)
- Bina kat sayısı: 5 kat (K5) ve 9 kat (K9)
- Optimizasyon algoritması: IAOA, DE, APSO

Geriye kalan tüm parametreler sabit tutulmuş ve yukarıda belirtilen her bir kombinasyon için algoritmalar üçer kez çalıştırılmıştır. Sabit tutulan diğer kullanıcı girdilerine ait değerler aşağıda listelenmiştir.

- Görüntü dosyası adı: Plan1.png
- Görüntü kenar hassasiyeti parametresi: 1
- Görüntü işleme sürecine ait görsel çıktıların gösterilmesi: false
- Optimizasyon popülasyon sayısı: 100
- Optimizasyon iterasyon sayısı: 300
- Paralel işleme parametresi: true
- Paralel işçi sayısı: 8
- Kesit bilgileri excel dosyası adı: pool_sec.xlsx
- Malzeme bilgileri excel dosyası adı: pool_mat.xlsx
- Paspayı: 0.05 (metre cinsinden)
- Bina kat yüksekliği: 3 (metre cinsinden)
- Bir kirişe bağlanabilecek saplama kiriş sayısı: 2

- Sabit yük: 5 (kN/m² cinsinden)
- Hareketli yük: 2 (kN/m² cinsinden)
- Duvar yükü: 6 (kN/m² cinsinden)
- Deprem analizi parametresi: true
- Hareketli yük kütle katılım oranı: 0.3
- Bina önem katsayısı: 1
- Taşıyıcı sistem davranış katsayısı: 8
- Dayanım fazlalığı katsayısı: 3

Bu parametre düzeniyle gerçekleştirilen analizlerin detayları ve elde edilen sonuçlar optimizasyon algoritmalarına göre Tablo 4.8, Tablo 4.9 ve Tablo 4.10’da verilmiştir.

Tablo 4.8. Optimizasyon sonuçları (APSO)

Beton Sınıfı	Kat #	Deprem	Maks Maliyet (₺)	Min Maliyet (₺)	Opt. Süresi (sn)	Min AFD
C25	5	D1	1,268,519	417,162	8852	0.3289
				416,699	9214	0.3285
				376,470	9072	0.2968
C40	5	D1	1,488,907	435,281	8784	0.2923
				480,283	9244	0.3226
				425,439	9062	0.2857
C25	5	D2	1,314,735	423,955	8738	1.3225
				509,295	8698	0.3874
				434,312	8939	0.3303
C40	5	D2	1,530,462	469,207	8634	0.3066
				456,167	9006	0.2981
				523,687	9062	0.3422
C25	9	D1	2,304,708	757,061	15022	0.3285
				733,739	15362	0.3184
				762,229	15630	0.3307
C40	9	D1	2,701,997	866,041	14905	0.3205
				860,154	15053	0.3183
				871,778	16274	0.3226
C25	9	D2	2,451,574	782,957	14516	1.3194
				831,135	15554	0.3390
				797,527	16046	0.3253
C40	9	D2	2,858,871	873,939	15685	0.3057
				837,939	15760	0.2931
				846,078	16510	0.2959

Tablo 4.9. Optimizasyon sonuçları (DE)

Beton Sınıfı	Kat #	Deprem	Maks Maliyet (₺)	Min Maliyet (₺)	Opt. Süresi (sn)	Min AFD
C25	5	D1	1,268,519	405,082	17944	0.3193
				412,833	16025	0.3254
				412,779	18487	0.3254
C40	5	D1	1,488,907	455,864	18967	0.3062
				455,318	18879	0.3058
				462,087	18215	0.3104
C25	5	D2	1,314,735	445,567	18481	0.3389
				442,990	18576	0.3369
				462,035	18484	0.3514
C40	5	D2	1,530,462	503,871	18205	0.3292
				482,425	18688	0.3152
				506,567	18698	0.3310
C25	9	D1	2,304,708	742,014	32220	0.3220
				760,731	32618	0.3301
				769,957	30985	0.3341
C40	9	D1	2,701,997	815,612	32504	0.3019
				812,867	32256	0.3008
				845,732	32268	0.3130
C25	9	D2	2,451,574	843,415	31502	0.3440
				827,320	32085	0.3375
				865,295	26897	0.3530
C40	9	D2	2,858,871	900,995	31805	0.3152
				899,700	32634	0.3147
				961,173	32396	0.3362

Tablo 4.10. Optimizasyon sonuçları (IAOA)

Beton Sınıfı	Kat #	Deprem	Maks Maliyet (₺)	Min Maliyet (₺)	Opt. Süresi (sn)	Min AFD
C25	5	D1	1,268,519	356,984	8817	0.2814
				360,894	8709	0.2845
				369,060	8548	0.2909
C40	5	D1	1,488,907	400,754	9012	0.2692
				400,754	9337	0.2692
				449,533	8790	0.3019
C25	5	D2	1,314,735	447,522	8671	0.3404
				365,159	8702	0.2777
				367,462	8777	0.2795
C40	5	D2	1,530,462	426,397	8903	0.2786
				444,808	8876	0.2906
				408,637	8880	0.2670
C25	9	D1	2,304,708	685,959	14020	0.2976
				660,980	14462	0.2868
				672,562	14706	0.2918
C40	9	D1	2,701,997	734,273	14738	0.2718
				759,814	14898	0.2812
				833,034	14887	0.3083
C25	9	D2	2,451,574	749,474	14616	0.3057
				737,260	14728	0.3007
				798,052	14746	0.3255

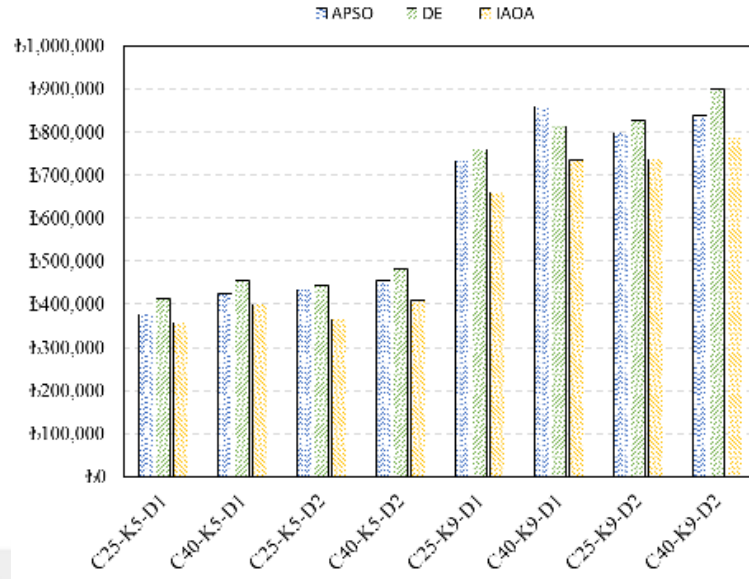
Tablo 4.10. (Devam) Optimizasyon sonuçları (IAOA)

C40	9	D2	2,858,871	884,177	14936	0.3093
				786,421	15053	0.2751
				808,983	14499	0.2830

Elde edilen sonuçlar doğrultusunda, algoritmaların performansı minimum maliyet, optimizasyon süresi ve minimum amaç fonksiyonu değeri kriterleri açısından karşılaştırıldığında IAOA algoritmasının genel olarak en başarılı sonuçları ürettiği gözlemlenmiştir. APSO, birçok durumda IAOA'ya oldukça yakın sonuçlar vermiş; ancak her zaman bir miktar daha geride kalmıştır. DE algoritması ise tüm senaryolarda hem çözüm kalitesi hem de maliyet açısından daha düşük performans göstermiş, yalnızca geçerli çözümler üretebilmiş fakat sömürü aşamasında yetersiz kalmıştır.

Optimizasyon süreleri açısından IAOA ve APSO benzer sürelerde çözüme ulaşırken, DE algoritması bu sürenin yaklaşık iki katı kadar işlem süresi gerektirmiştir. Bu farkın temel nedeni, algoritmalarda kullanılan paralel hesaplama (parfor) yapısının uygulanma biçimidir. IAOA ve APSO algoritmalarında yalnızca amaç fonksiyonu modülü hesaplamaları parfor yapısı içerisinde tanımlanmış ve işçilere gönderilen veri miktarı sınırlandırılarak zaman etkinliği sağlanmıştır. Buna karşılık, DE algoritmasında bazı vektörel işlemler de parfor içerisinde gerçekleştirilmiş ve bu durum işlem süresinin belirgin şekilde artmasına neden olmuştur.

Ayrıca, tablolar incelendiğinde 5 katlı ve 9 katlı yapı modelleri arasında belirgin bir optimizasyon süresi farkı olduğu görülmektedir. Bu farkın temel nedeni, yapı yüksekliği arttıkça taşıyıcı eleman sayısının artması ve buna bağlı olarak yapı analiz sürecinin uzamasıdır. Özellikle her analiz döngüsünde, artan eleman sayısına bağlı olarak daha fazla iç kuvvet, deplasman ve tasarım kontrolü verisinin üretilmesi, okunması ve işlenmesi gerekmektedir. Bu durum doğrudan optimizasyon süresine yansımaktadır.

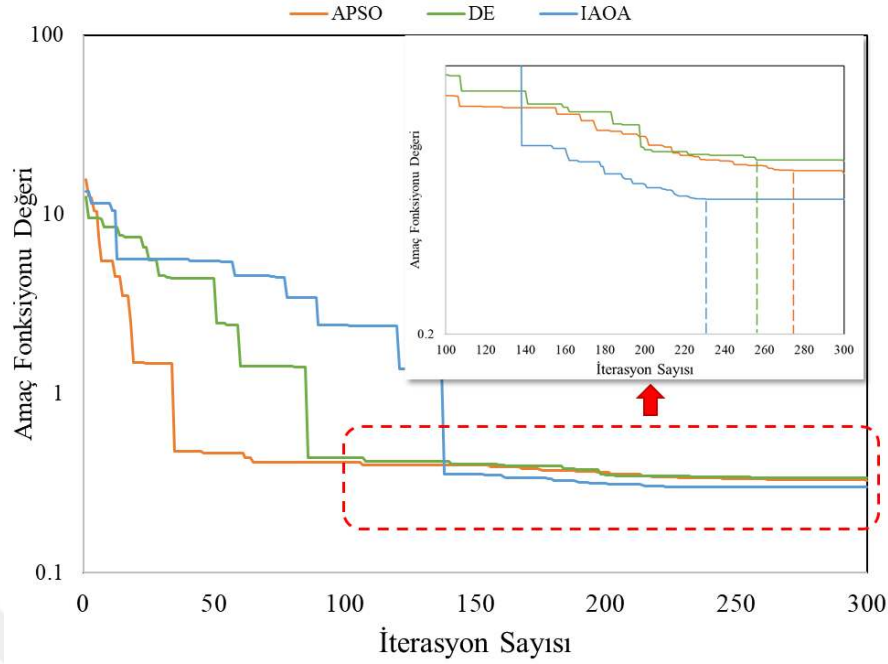


Şekil 4.7. Tüm konfigürasyonlardaki minimum maliyet sonuçları

Şekil 4.7’de farklı beton sınıfı (C25, C40), bina kat sayısı (5, 9) ve deprem senaryosu (D1: Eskişehir, D2: Hatay) kombinasyonları için APSO, DE ve IAOA algoritmalarıyla elde edilen en düşük maliyetli yapı tasarımları gösterilmektedir. Her bir grup, aynı yapı konfigürasyonu için algoritmalar arası performans farkını görsel olarak kıyaslamaya olanak tanımaktadır.

Grafikteki genel eğilim, IAOA algoritmasının tüm senaryolarda en düşük maliyetli çözümleri üreterek en başarılı performansı gösterdiğini ortaya koymaktadır. APSO da birçok durumda IAOA’ya yakın maliyet değerleriyle benzer bir başarı sergilemiştir. Buna karşılık, DE algoritması genel olarak daha yüksek maliyetli sonuçlar üretmiş ve çoğu senaryoda diğer iki algoritmanın gerisinde kalmıştır.

Ancak dikkat çekici bir istisna olarak, C40-K9-D1 konfigürasyonunda DE algoritması, APSO algoritmasından daha düşük maliyetli bir çözüm üretmiştir. Bu sonuç, genel algoritma başarımlarının eğilimlerinin dışına çıkan istisnai bir durumu temsil etmektedir. Sezgisel algoritmalar doğası gereği stokastik bir yapıya sahiptir; yani çözüm süreci, başlangıç popülasyonunun uzaydaki konumlarına ve bu konumlardan itibaren uygulanan arama stratejisinin yönelimine bağlı olarak değişkenlik gösterebilir. Bu bağlamda, söz konusu başarı, DE algoritmasının başlangıçta çözüm uzayının daha avantajlı bir bölgesine rastlantısal olarak yerleşmiş olması ve ardından uyguladığı sömürü adımlarının bu bölgedeki iyi çözümlere odaklanabilmiş olmasından kaynaklanmış olabilir.



Şekil 4.9. C25-K9-D2 konfigürasyonunda algoritmaların yakınsama eğilimleri

Yukarıdaki yakınsama grafiklerinde algoritmaların iterasyonlar boyunca amaç fonksiyonu değerinde gösterdikleri iyileşme performansı gözlemlenmektedir. İlk 100 iterasyon süresince AFD bakımından en hızlı düşüşü APSO sağlamış, onu DE takip etmiş, IAOA ise daha yavaş bir yakınsama eğilimi sergilemiştir. Ancak ileri iterasyonlarda IAOA, diğer algoritmalara kıyasla daha iyi çözümler üreterek optimuma daha hızlı ve etkili bir şekilde ulaşmıştır. Öte yandan, DE ve APSO hem daha kötü optimumlara ulaşmış hem de bu çözümlere daha geç ulaşabilmiştir. Ayrıca, yapı yüksekliğinin artmasına bağlı olarak çözüm uzayının boyutu büyüdükçe algoritmaların yakınsama süresinin uzadığı, yani optimuma ulaşmak için daha fazla iterasyon gerektiği açıkça görülmektedir. Bu durum, eleman sayısındaki artışla birlikte hem çözüm uzayının üstel olarak genişlemesi hem de ceza mekanizmasının daha fazla sayıda eleman üzerinden çalışmasıyla açıklanabilir.

Sonuç olarak, IAOA algoritması yalnızca geniş çözüm uzayını etkili biçimde tarayarak cezalı bölgelerden kaçınmakla kalmamış, aynı zamanda maliyet minimizasyonunu yüksek düzeyde başarıyla gerçekleştirmiştir. Özellikle sömürü aşamasında gösterdiği etkin performans sayesinde, tüm yapı konfigürasyonlarında en düşük maliyetli çözümleri üretmiştir. Bu nedenle, bu tez kapsamında tanımlanan karma yapısal optimizasyon problemi için en uygun algoritma olarak IAOA belirlenmiştir. Diğer algoritmalarından APSO da benzer bir performans sergilemiş olsa da, genel başarı düzeyi

IAOA'nın gerisinde kalmıştır. DE algoritması ise yalnızca geçerli çözümler üretebilmiş, ancak maliyet açısından yeterince verimli sonuçlar ortaya koyamamıştır. Bu karşılaştırmalar, algoritma seçiminin probleme özel yapılması gerektiğini ve çözüm uzayının yapısına göre bazı algoritmaların ciddi avantajlar sağlayabileceğini göstermektedir.

4.3. Geliştirilen Yöntemlerin Tutarlılığı ve Doğrulama Süreci

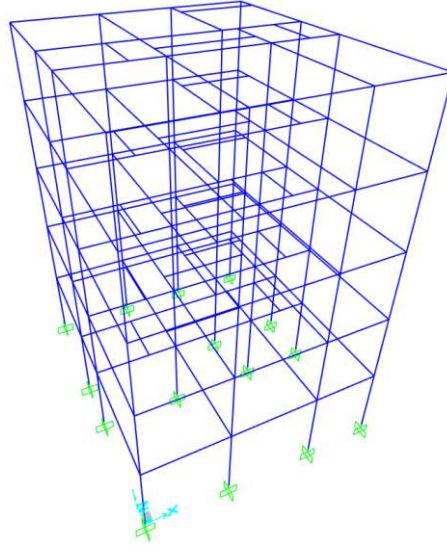
Tez kapsamında farklı amaçlara, tekniklere ve disiplinlere ait bilgilerin bir arada kullanıldığı çok yönlü bir otomasyon yazılımı geliştirilmesi hedeflenmiş ve bu doğrultuda gerçekleştirilen çalışmalarla küçük adımlarla da olsa sağlam temellere dayalı bir ilerleme sağlanmaya çalışılmıştır. Her yazılımın ilk sürümünde olduğu gibi, bu çalışmada geliştirilen yazılımda da bazı hataların bulunması olasıdır. Ancak yazılım bilinçli kullanıcılar tarafından kullanıldıkça ve bu kullanıcılardan geri bildirim alındıkça, varsa bu tür hataların tespit edilip giderilmesi mümkün olacaktır.

Tez süreci boyunca yazılımın bir modülü geliştirilirken, aynı zamanda önceki modüllerdeki olası hataların da tespiti ve giderilmesi amacıyla yoğun bir kontrol süreci yürütülmüştür. Bu çerçevede karşılaşılan her hata titizlikle analiz edilerek giderilmiş ve yazılım güncel halini almıştır.

Bu doğrulama süreci kapsamında, görüntü işleme modülü geliştirilirken her adımda görsel çıktılar alınmış; elde edilen duvar çizgileri ve akslar sayılarak, tanımlanan kurgunun beklenen şekilde işleyip işlemediği sürekli olarak denetlenmiştir.

Yapı analizi modülü için, oluşturulan sonlu eleman modeli birçok basit yapı örneği üzerinde hem geliştirilen yazılım aracılığıyla hem de SAP2000 (<http-5>) ile analiz edilmiştir. Her iki analizden elde edilen sonuçlar karşılaştırılarak tutarlılık sağlanana kadar hatalar ayıklanmış ve gerekli düzeltmeler yapılmıştır.

Ardından IAOA algoritması ile optimizasyon sonucu elde edilmiş örnek bir model (C25-K5-D1) SAP2000'de manuel olarak modellenmiş (Şekil 4.10) ve sonuçlar kıyaslanmıştır. Örnek bir kolon



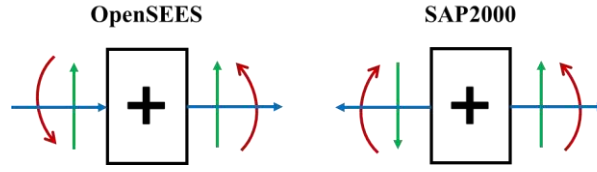
Şekil 4.10. LAOA sonucu C25-K5-D1 için SAP2000 modeli

Yapı modelinin doğruluğunu değerlendirmek amacıyla, SAP2000 ve geliştirilen otomasyon tabanlı yazılım ile elde edilen yapı periyotları, eleman iç kuvvetleri ve düğüm noktalarındaki yer değiştirmeler karşılaştırılmış ve Tablo 4.11, Tablo 4.12 ve Tablo 4.13'te sunulmuştur. Yapılan karşılaştırma sonucunda, analiz çıktılarının birebir örtüştüğü görülmüş; bu durum, yazılımın yapıyı eksiksiz ve hatasız biçimde modelleyebildiğini ortaya koymuştur.

Tablo 4.11. Örnek bir kolona ait iç kuvvetlerin karşılaştırılması

	ID	iP (kN)	iV2 (kN)	iV3 (kN)	iT (kNm)	iM2 (kNm)	iM3 (kNm)
Yazılım	118	705.62	-0.41	3.14	0.01	-3.11	-0.48
SAP2000	46	-705.62	0.41	-3.14	-0.01	-3.11	0.48
	ID	jP (kN)	jV2 (kN)	jV3 (kN)	jT (kNm)	jM2 (kNm)	jM3 (kNm)
Yazılım	118	705.62	0.41	-3.14	-0.01	-6.30	-0.74
SAP2000	46	-705.62	0.41	-3.14	-0.01	6.30	-0.74

Tablo 4.11'de sunulan iç kuvvet karşılaştırmalarında gözlenen pozitif–negatif farklar, yazılımda kullanılan yapı analiz programı OpenSEES ile karşılaştırma yapılan SAP2000 yazılımının pozitif yön kabullerinin farklı olmasından kaynaklanmaktadır. Her iki programda eksenel kuvvet, moment ve kesme kuvvetleri için kabul edilen yön kabulleri Şekil 4.11'de ayrıntılı olarak verilmiştir.



Şekil 4.11. OpenSEES ve SAP2000 programlarındaki pozitif yön kabulleri

Tablo 4.12. Bir düğüm noktası için deplasman değerleri

	ID	Deplasman	
		X (mm)	Y (mm)
Yazılım	601	11.11	0.69
SAP2000	116	11.11	0.69

Tablo 4.13. Yapı periyotları ve modal kütle katılım oranlarına ait sonuçlar

Mod #	SAP2000			Yazılım		
	Periyot (s)	Modal Kütle Katılım Oranları (%)		Periyot (s)	Modal Kütle Katılım Oranları (%)	
		X	Y		X	Y
1	0.8596	79.48	2.17	0.8596	79.48	2.17
2	0.8321	2.92	81.65	0.8321	2.92	81.65
3	0.6956	2.88	1.90	0.6956	2.88	1.90
4	0.2850	9.08	0.39	0.2850	9.08	0.39
5	0.2769	0.52	9.08	0.2769	0.52	9.08
6	0.2318	0.38	0.22	0.2318	0.38	0.21
7	0.1704	2.67	0.32	0.1704	2.67	0.32
8	0.1665	0.38	2.73	0.1665	0.38	2.73
9	0.1400	0.15	0.07	0.1400	0.15	0.07
10	0.1253	0.94	0.20	0.1253	0.94	0.20
11	0.1228	0.23	0.96	0.1228	0.23	0.96
12	0.1052	0.21	0.06	0.1052	0.21	0.06

Donatı tasarımı modülü geliştirilirken, benzer şekilde STA4CAD ([http-6](http://6)) yazılımı kullanılarak oluşturulan basit yapı modelleri aynı yükleme koşulları altında analiz edilmiştir. Geliştirilen yazılımın çıktıları ile STA4CAD sonuçları karşılaştırılmış; aradaki farkların, çalışmada kullanılan yük atama yaklaşımının STA4CAD yazılımındaki uygulamadan farklı olmasından kaynaklandığı tespit edilmiştir.

Yalnızca yükleme yöntemi farkının etkisini değerlendirmek amacıyla, IAOA ile elde edilmiş C25-K5-D1 çözümüne ait yapı modeli STA4CAD ortamında manuel olarak oluşturulmuş ve yapının toplam düşey yükleri eşitlenecek şekilde yükler tanımlanmıştır. Yapılan analiz sonucunda bir kirişin donatı alanlarına ilişkin değerler, geliştirilen yazılım ve STA4CAD arasında karşılaştırılmış ve sonuçlar Tablo 4.14'te sunulmuştur.

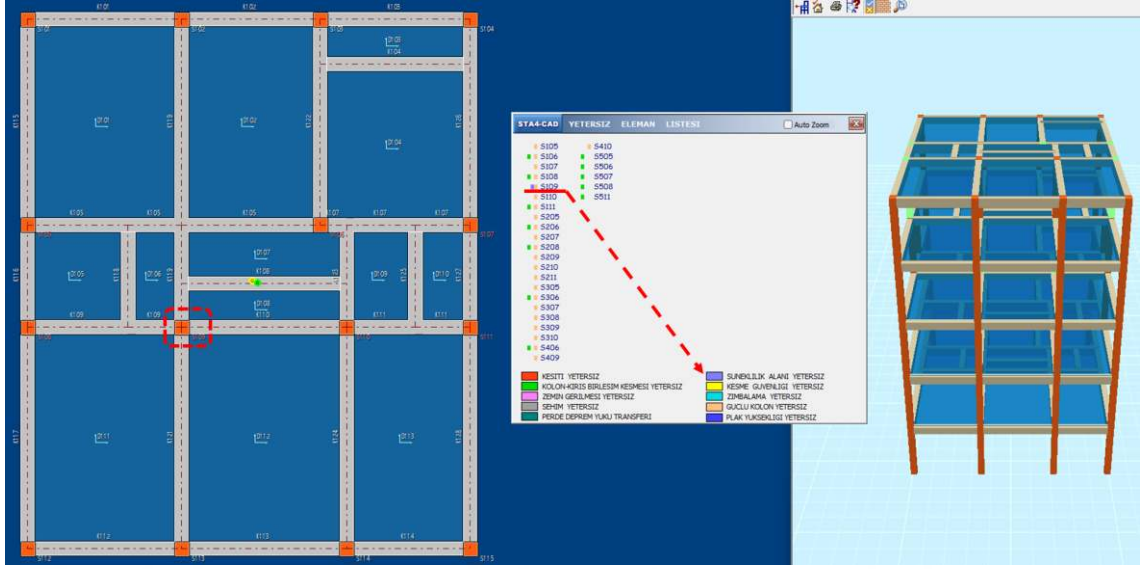
Tablo 4.14. Bir kirişin donatı alanı sonuçları

	Yazılım Kiriş Donatı Alanları (cm ²)			STA4CAD Kiriş Donatı Alanları (cm ²)		
	i	Açıklık	j	i	Açıklık	j
Üst	3.64	2.26	3.33	3.41	2.26	3.22
Alt	2.76	2.76	2.76	1.71	2.76	1.61

Tabloda görüldüğü üzere donatı alanlarına ait değerler yakın çıkmıştır. Alt donatılar arasındaki fark, yazılımda açıklıkta elde edilen donatı alanının kirişin iki ucuna uzatılması sonucu, uçlarda ek bir donatı gerekmemesi halinde açıklıkla aynı değere sahip olması gerektiğinden olmaktadır. STA4CAD’de bu işlem donatı alanlarının belirlenmesinden sonra bu alandan daha büyük olacak şekilde belirlenen adet ve çapta olan donatıların belirlenmesinden sonra gerçekleştirilmekte bu yüzden kiriş uç noktalarındaki alt donatı alanlarında fark görülmektedir.

Tabloda görüldüğü üzere, donatı alanlarına ait değerler birbirine oldukça yakındır. Alt donatılar arasındaki fark, yazılımda açıklık bölgesinde elde edilen donatı alanının kirişin iki ucuna uzatılması ve uç noktalarda ek bir donatı gerekmemesi durumunda bu alanın açıklıkla aynı kalması esasına dayanmaktadır. STA4CAD’de ise bu işlem, donatı alanları belirlendikten sonra bu alandan daha büyük olacak şekilde adet ve çap seçimi yapılarak gerçekleştirilmekte; ardından bu donatılar uçlara uzatılmaktadır. Bu nedenle, kiriş uç noktalarındaki alt donatı alanlarında fark oluşmaktadır. Öte yandan, üst donatı alanlarındaki küçük farklar ise iki yazılım arasındaki yükleme yöntemi farkından kaynaklanmaktadır.

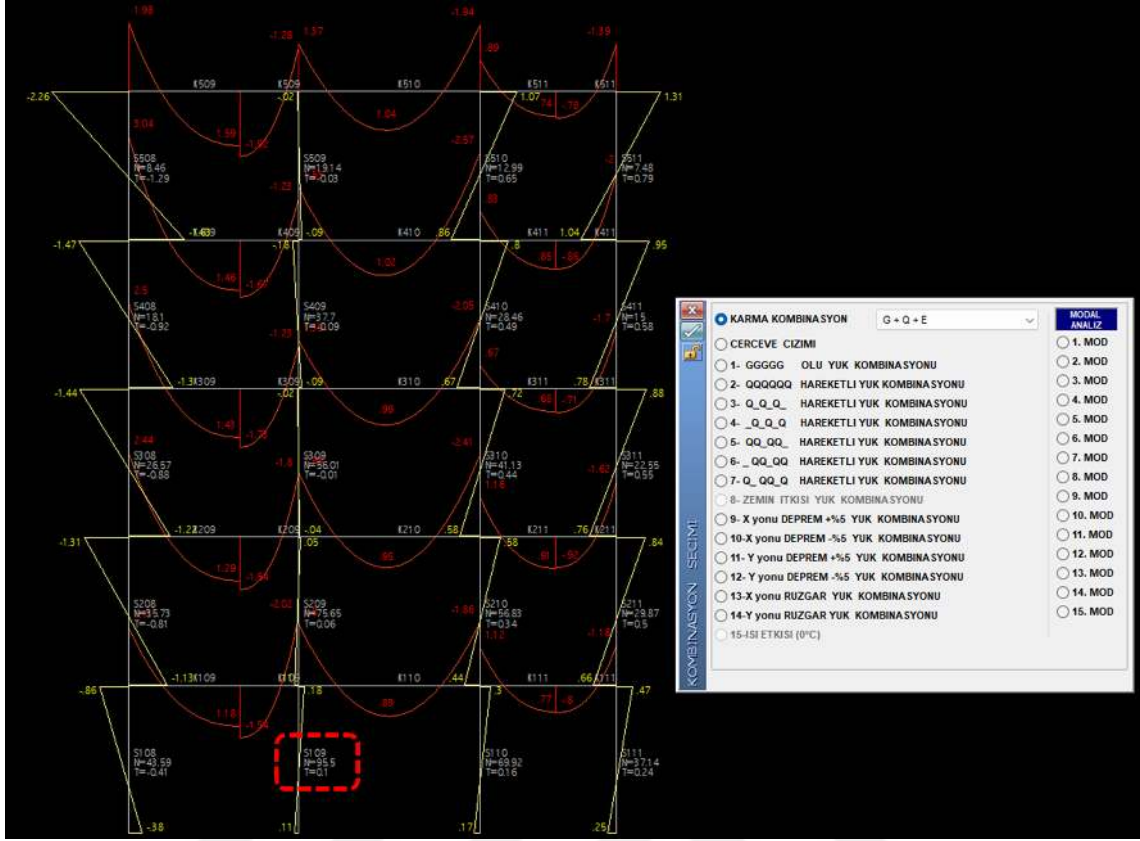
Buna ek olarak, yönetmelik kontrollerine ait penaltıların doğru şekilde çalışıp çalışmadığını değerlendirmek amacıyla, STA4CAD analizi sonucunda yönetmelik açısından yetersiz bulunan elemanlar incelenmiş ve ilgili bulgular Şekil 4.12’de sunulmuştur.



Şekil 4.12. STA4CAD analizi sonucu yetersiz eleman listesi

IAOA sonucu elde edilen yapıda, yazılımda tanımlanmış herhangi bir penaltı ihlali bulunmamasına rağmen, STA4CAD sonuçlarında bazı kolonlarda yetersizlikler tespit edilmiştir. Bu yetersizlikler incelendiğinde, büyük çoğunluğunun yazılımda henüz tanımlanmamış yönetmelik kontrollerine ait olduğu belirlenmiştir. Ancak bir kolona ait süneklik alanı yetersizliği (Şekil 4.12), yazılımda da mevcut olan CDPEN kontrolü kapsamında değerlendirildiği için, iki yazılım arasında ortaya çıkan farkın nedeni ayrıca araştırılmıştır.

STA4CAD’de S109, yazılımda ise 118 olarak tanımlanan kolona ait deprem yükü içeren yük kombinasyonları altındaki eksenel kuvvetler kıyaslandığında, bu kolonda penaltı oluşmaması için eksenel kuvvetin 900 kN’u aşmaması gerektiği, yazılım sonuçlarında ise bu kombinasyonlara ait maksimum eksenel kuvvetin 860 kN olduğu belirlenmiştir. Ancak, Şekil 4.13’te gösterildiği üzere, STA4CAD analizinde bu değer 955 kN olarak hesaplanmıştır. Bu nedenle, STA4CAD tarafında süneklik alanı yetersizliği oluşmuştur.



Şekil 4.13. STA4CAD programında S109 kolonuna ait eksenel kuvvet değeri

Bu durum, yükleme yöntemindeki farktan kaynaklanmaktadır. Sonuç olarak, yükleme yönteminin hem yönetmelik kontrolleri hem de donatı tasarımı üzerinde önemli bir etkisi olduğu açıkça görülmekte ve bu farkın yazılımın sonraki sürümlerinde giderilmesi gereken temel bir geliştirme alanı olduğu anlaşılmaktadır.

4.4. Sonuçların Kaydedilmesi

Optimizasyon sonucunda elde edilen en iyi yapı modeline ait tasarım ve analiz çıktıları, **outputs** adı verilen bir yapı altında toplanmıştır. Bu çıktılar, aşağıdaki temel bilgi gruplarını kapsamaktadır:

- **Yapı tasarımı bilgileri:** Kolon ve kiriş kesit boyutları, her bir eleman için donatı ve etriye alanları.
- **Yapı analizi sonuçları:**
 - Yapının X ve Y yönlerindeki hakim doğal periyotları,
 - Her bir elemanın iç kuvvetleri,
 - Kat kesme kuvvetleri,

- “Maksimum ve ortalama kat ötelemeleri,
 - Göreli kat öteleme oranları,
 - Tüm düğüm noktalarının deplasman değerleri,
 - Burulma düzensizliği ve yumuşak kat varlığı,
 - P–delta etkisinin hesaba katılıp katılmadığı bilgileri.
- **Optimizasyon çıktıları:**
 - Toplam maliyet ve ceza katkılarından oluşan amaç fonksiyonu değeri,
 - Kolon konumları,
 - Optimizasyon süresi.

Tüm bu bilgiler her analiz sonrası düzenli biçimde kaydedilmiş ve kullanıcının hem denetim hem de yeniden değerlendirme yapabilmesine olanak sağlayacak şekilde erişilebilir hale getirilmiştir.

Bunun haricinde, her bir optimizasyon çalışması sonucunda oluşan tüm tasarım ve analiz çıktılarının MATLAB ortamında görsel olarak bir arada sunulduğu Şekil 4.14 aşağıda verilmiştir. Bu şekil, elde edilen modelin maliyet, ceza, eleman yerleşimi ve analiz çıktıları gibi tüm verilerini kullanıcıya bütüncül biçimde sunarak detaylı incelemeye olanak sağlamaktadır.

Ayrıca, analiz sonuçları kapsamında örnek bir model (C25-K5-D1) için elde edilen kolon ve kiriş kesitlerine ait özet tablo ile yapı analizine ilişkin temel çıktılar Tablo 4.15, Tablo 4.16 ve Tablo 4.17’de sunulmuştur.

outputs x			
1x1 struct with 7 fields			
Field	Value	Size	Class
OptTime	8.8169e+03	1x1	double
Best_FF	0.2814	1x1	double
Best_Pos	1x63 double	1x63	double
Best_Model	1x1 struct	1x1	struct
Cost	3.5698e+05	1x1	double
Obj_Val	0.2814	1x1	double
STR	1x21 struct	1x21	struct
Script	[### Title: baba_deneme4\n### Author: Ta... -	-	-
LoadCase	["G","Q","Ex", ...]	-	-
EleLoad	[11.9150,2.3660,0, ...]	-	-
PeriodX	[0.8596,0.8596,0.8596, ...]	-	-
PeriodY	[0.8321,0.8321,0.8321, ...]	-	-
Columns	[5x7 cell,5x7 cell,5x7 cell, ...]	-	-
Beams	[5x8 cell,5x8 cell,5x8 cell, ...]	-	-
StoryShearX	[7.2000e-05,-2e-04,-4e-05,-6e-04,-4.2600e...	-	-
StoryShearY	[[-1.3323e-15,4.2000e-04,2.1000e-04,1.600...	-	-
FCIndex	[5x260 logical,5x260 logical,5x260 logical, ...]	-	-
FBIndex	[5x260 logical,5x260 logical,5x260 logical, ...]	-	-
MaxStoryDriftX	[2.6606e-05,4.2635e-05,2.6046e-05,-2.139...	-	-
MaxStoryDriftY	[[-1.2485e-05,1.4398e-05,3.5353e-05,4.903...	-	-
AvgStoryDriftX	[[-1.9809e-05,2.8202e-05,1.0486e-05,-2.223...	-	-
AvgStoryDriftY	[[-6.8865e-06,-2.8437e-07,1.9523e-05,3.27...	-	-
StoryWeight	[1.2385e+03,1.2385e+03,1.2385e+03,1.23...	-	-
TorsionInrX	[1.3431,1.5118,2.4838,9.6208,2.2061],[1.3...	-	-
TorsionInrY	[1.8130,-50.6316,1.8109,1.4972,1.8962],[1....	-	-
SoftStoreyInrX1	[0.7024,2.6894,-4.7155,-0.1653],[0.7024,2....	-	-
SoftStoreyInrX2	[1.4237,0.3718,-0.2121,-6.0498],[1.4236,0....	-	-
SoftStoreyInrY1	[24.2161,-0.0146,0.5960,1.7785],[24.2213,-...	-	-
SoftStoreyInrY2	[0.0413,-68.6518,1.6777,0.5623],[0.0413,-6...	-	-
StoryDriftRX	[0.0009,0.0014,0.0009,-0.0007,0.0010],[1.7...	-	-
StoryDriftRY	[[-0.0004,0.0005,0.0012,0.0016,0.0012],[-8...	-	-
ElemF	[260x13 double,260x13 double,260x13 dou...	-	-
NodeD	[125x3 double,125x3 double,125x3 double,...	-	-
DRPen	[0,0,0, ...]	-	-
ConCost	[1.3956e+05,1.3956e+05,1.3956e+05, ...]	-	-
CBCPen	[1x2 cell,1x2 cell,1x2 cell, ...]	-	-
PdeltaOn	[0,0,0, ...]	-	-
TorsionInrFlag	[1,1,1,1, ...]	-	-
SoftStoreyInrFlag	[1,1,1,1, ...]	-	-
SR	1x260 struct,1x260 struct,1x260 struct, ...]	-	-
Name	[1001,1004,1006, ...]	-	-
Index	[16,17,18, ...]	-	-
Length	[3.4300,3.0900,3.3400, ...]	-	-
Type	["Beam","Beam","Beam", ...]	-	-
As02	[3.6392e-04,2.2619e-04,3.3257e-04],[2.93...	-	-
As20	[2.7600e-04,2.7600e-04,2.7600e-04],[2.76...	-	-
LongPen	[0,0,0, ...]	-	-
As03	[0,0,0, ...]	-	-
As30	[0,0,0, ...]	-	-
StirPen	[0,0,0, ...]	-	-
As23	[1.0053e-04,1.0053e-04,1.0053e-04, ...]	-	-
CDPen	[0,0,0, ...]	-	-
CDNodes	[0,0,0, ...]	-	-
Cost	[782.3085,683.9625,738.9962, ...]	-	-
BDPen	[1x3 cell,1x3 cell,1x3 cell, ...]	-	-
SBPen	[1x3 cell,1x3 cell,1x3 cell, ...]	-	-
Penalty	0	1x1	double
PenInfo	7x4 cell	7x4	cell
Conv_curve	1x300 double	1x300	double
Max_Model	1x1 struct	1x1	struct
Cost	1.2685e+06	1x1	double
Obj_Val	1	1x1	double
STR	1x21 struct	1x21	struct
Penalty	0	1x1	double
PenInfo	7x4 cell	7x4	cell
Time	8.8652e+03	1x1	double

Şekil 4.14. Optimizasyon sonucu kaydedilen tüm sonuçların MATLAB ortamında görüntülenmesi

Tablo 4.15. Optimizasyon sonucunda belirlenen kolon kesitleri

KOLON KESİTLERİ		
Kolon ID	w2 (cm)	w3 (cm)
101	30	30
102	30	30
103	30	30
104	30	30
107	30	30
110	30	30
113	30	30
116	30	30
118	30	30
119	30	30
121	30	30
122	30	30
123	30	30
124	30	30
125	30	30

Tablo 4.16. Optimizasyon sonucunda belirlenen kiriş kesitleri

KİRİŞ KESİTLERİ		
Kiriş ID	w2 (cm)	w3 (cm)
1001	30	40
1004	30	40
1006	30	40
1007	30	40
1008	30	50
1014	30	50
1019	30	50
1023	30	50
1026	30	50
1028	30	50
1029	30	40
1030	30	40
1035	30	40
1039	30	40
1042	30	40
1044	30	40
1045	30	40
1048	30	40
1050	30	40
1051	30	50
1054	30	50
1056	30	50

Tablo 4.16. (Devam) Optimizasyon sonucunda belirlenen kiriş kesitleri

1057	30	40
1058	30	50
1062	30	50
1065	30	50
1067	30	50
1068	30	40
1070	30	40
1071	30	50
1074	30	50
1076	30	50
1077	30	40
1078	30	50
1082	30	50
1085	30	50
1087	30	50

Tablo 4.17. Optimizasyonla elde edilen yapı modeline ait analiz sonuçları

Yapı Periyodu (sn)		Görelî Kat Ötelemesi (%)		Donatı Tonajı (t)	Beton Metraжі (m ³)
X	Y	X	Y		
0.8596	0.8321	0.1008	0.0951	8.17	64.91

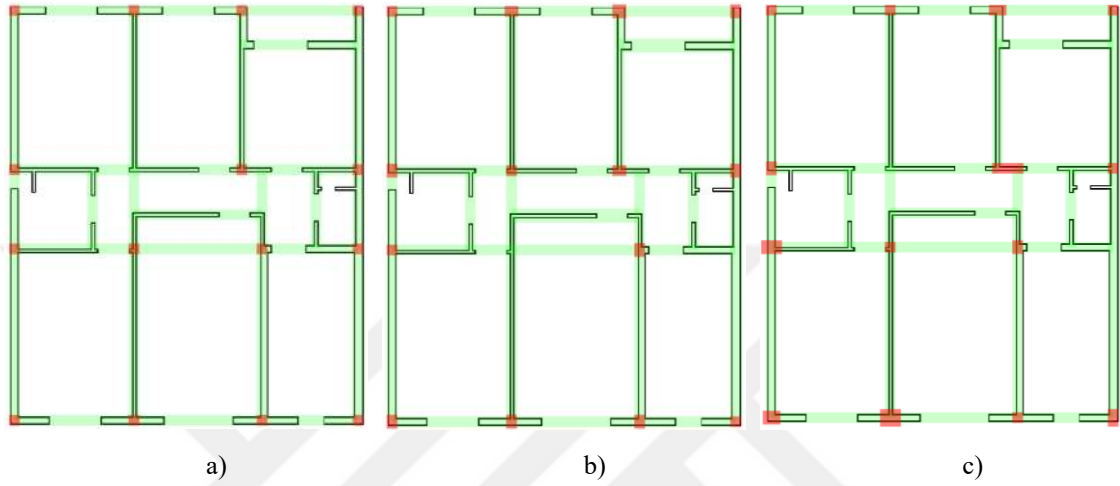
4.5. Görsel Çıktılar

Yazılımda, optimizasyon sonuçlarına ilişkin olarak görsel temsile sahip iki temel çıktı üretilmektedir. Bunlardan ilki, algoritmanın iteratif süreci boyunca amaç fonksiyonu değeriindeki değışimi gösteren yakınsama eğrisi; diğeri ise en iyi yapı modelinin kat planı üzerinde kesit boyutları ile birlikte gösterimidir.

Optimizasyon sonucunda elde edilen en iyi yapı modeline ait kat planı görselinde, kolon ve kiriş kesitlerinin metrik boyutları, plan çiziminin ölçeğı dikkate alınarak piksel karşılıklarına dönüştürülmüş ve bu bilgiler kullanılarak otomatik bir çizim algoritması geliştirilmiştir. Böylece, yapıdaki kesit dağılımı ve boyutlarına ilişkin görsel bir kontrol olanağı sağlanmış; aynı zamanda geometrik ceza koşullarının (örneğin minimum açıklık veya yerleşim mesafeleri) kullanıcı tarafından kolayca doğrulanabilmesine imkan tanınmıştır.

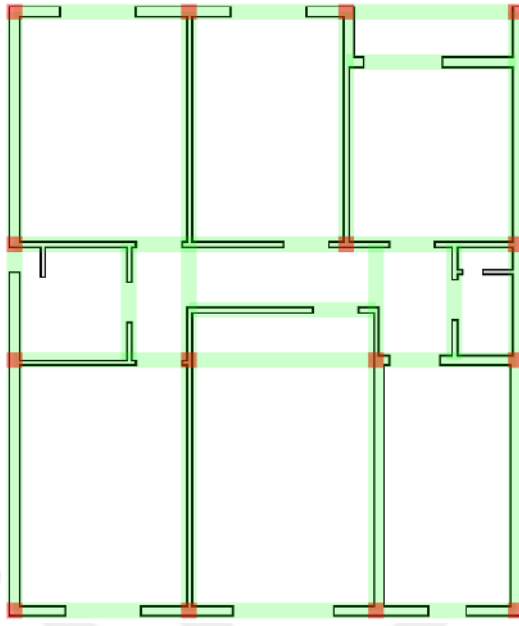
Şekil 4.15’te, kullanılan üç farklı optimizasyon algoritması olan IAOA, DE ve APSO ile elde edilen sonuçlardan, C25 beton sınıfı, 5 katlı yapı ve D1 deprem senaryosuna karşılık gelen konfigürasyonda en iyi yapı modeli çözümlerine ait kat planı görselleri sunulmuştur.

Bu görsellerde her algoritmanın optimize ettiği kolon ve kiriş yerleşimleri ile kesit boyutları, metrik ölçüler esas alınarak çizdirilmiş ve yapının genel geometrik düzeni karşılaştırmalı olarak değerlendirilmiştir. Böylece, algoritmaların aynı girdi koşulları altında ulaştığı yapısal düzen ve kesit seçimlerine ilişkin farklar doğrudan gözlemlenebilir hale getirilmiştir.

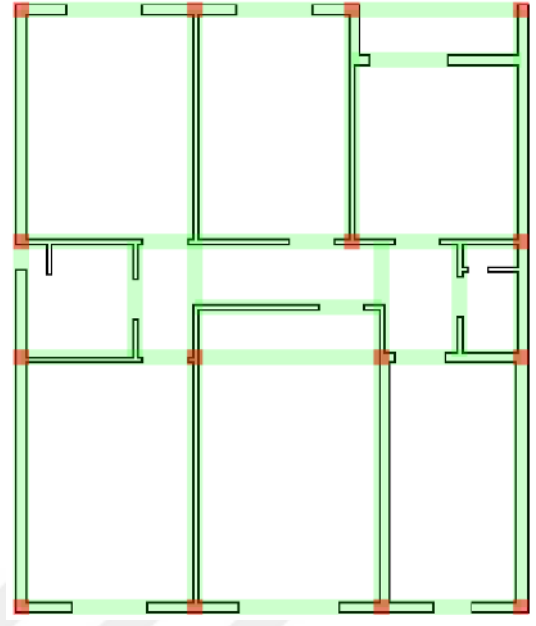


Şekil 4.15. C25-K5-D1 konfigürasyonuna ait en iyi yapı modeli sonuçları, a) LAOA, b) APSO, c) DE

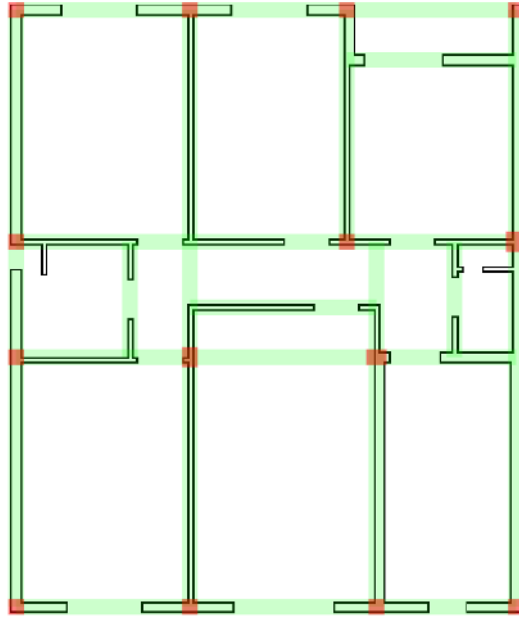
Ayrıca, optimizasyon problemine en uygun çözümlerin elde edildiği LAOA algoritması ile tüm parametre kombinasyonları için oluşturulan sonuç kat planlarına ait görseller, Şekil 4.16’da sunulmuştur.



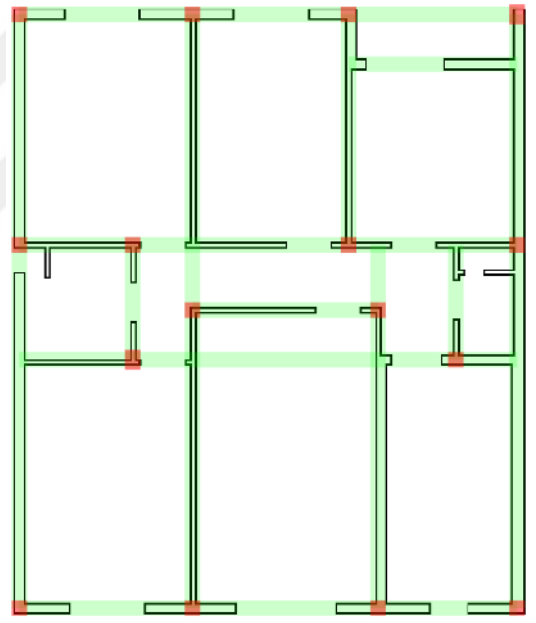
a) C25-K5-D1



b) C25-K5-D2

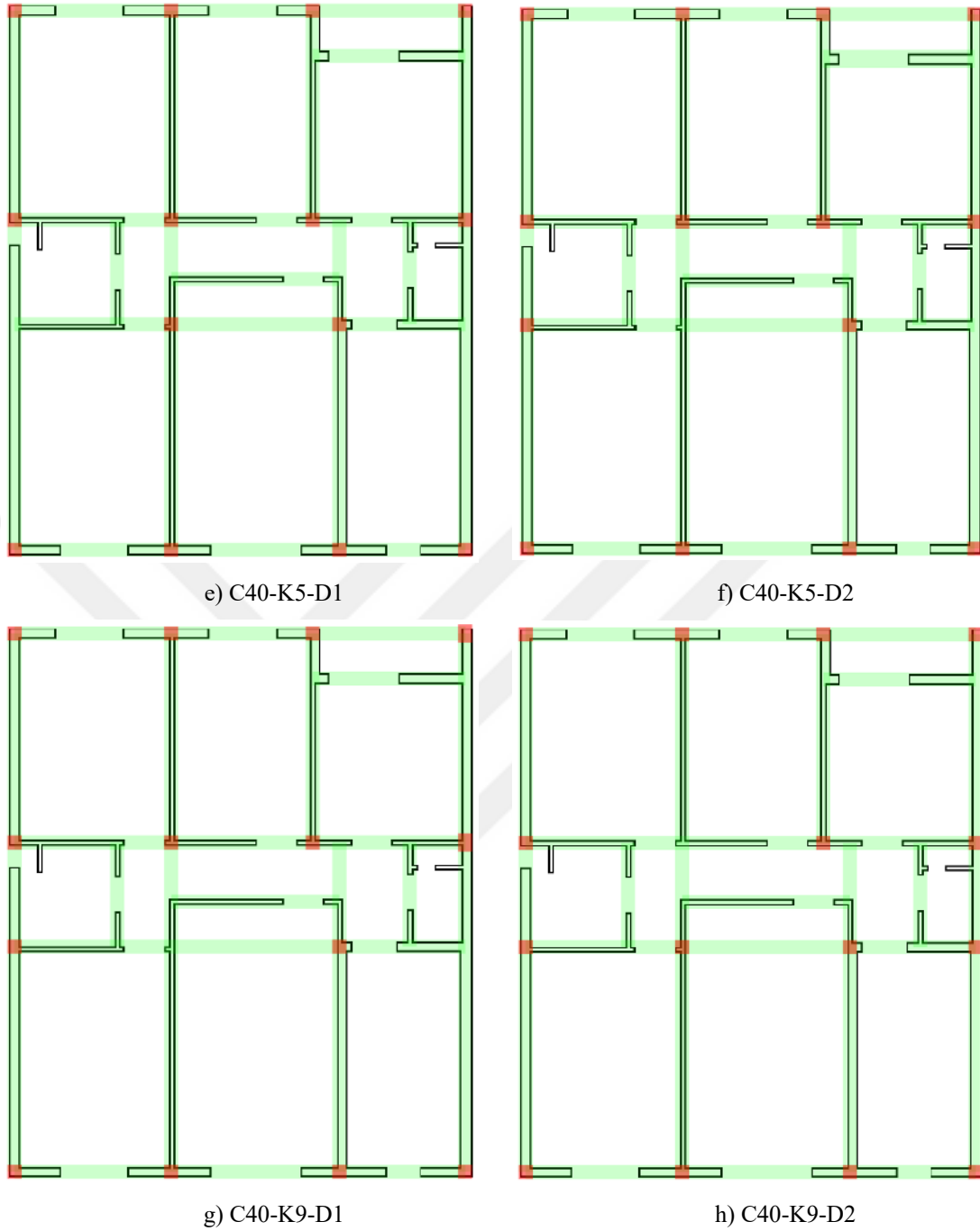


c) C25-K9-D1



d) C25-K9-D2

Şekil 4.16. *IAOA ile gerçekleştirilmiş tüm parametrik kombinasyonların görsel çıktıları*



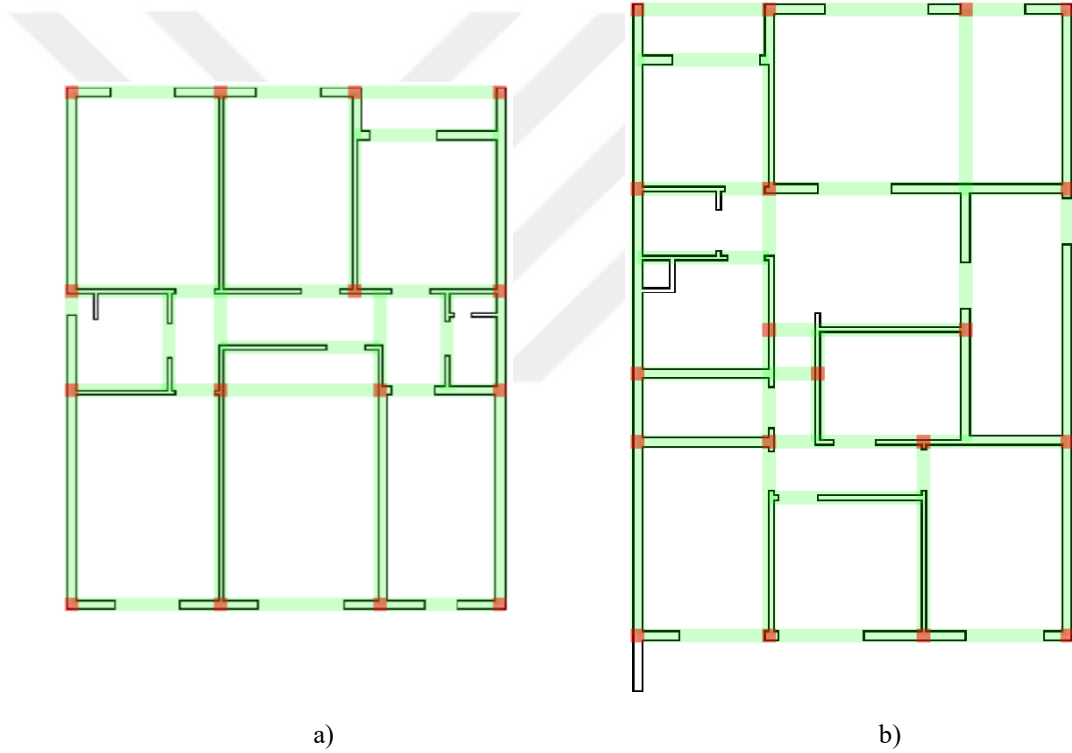
Şekil 4.16. (Devam) *IAOA ile gerçekleştirilmiş tüm parametrik kombinasyonların görsel çıktıları*

Bu görseller incelendiğinde, CBCPen (kiriş kesit genişliğinin kolon kesit genişliğinden büyük olması) cezasından tüm çözümlerin başarıyla kaçındığı görülmektedir. Bu durum, kolonların planda düzenli ve minimal sayıda yerleştirilmiş olmasının, cezanın oluşmasını önlemede etkili olduğunu ortaya koymaktadır. Öte yandan, 5 katlı modellerde D1 ve D2 deprem senaryoları arasında kolon yerleşimi bakımından belirgin bir fark gözlemlenmemekte; ancak 9 katlı modellerde, artan deprem yükleri

nedeniyle kolon sayısında anlamlı bir artış gözlemlenmektedir. Bu durum, yüksek katlı yapılarda rijitlik veya dayanım ihtiyacının arttığını ve bu ihtiyacın daha fazla kolon kullanımı ile karşılandığını göstermektedir.

4.6. Örnek Uygulamalar

Şimdiye kadar sunulan analizlerin ardından, son olarak, çalışmada kullanılan iki farklı kat planı, C25 beton sınıfı, 5 katlı ve D1 deprem senaryosu altında IAOA algoritması ile üçer kez optimize edilmiştir. Her plan için elde edilen en iyi yapı modelinin tasarımı Şekil 4.17'de görsel olarak sunulmuştur.



Şekil 4.17. IAOA-C25-K5-D1 konfigürasyonu için en iyi yapı modeli sonuçları a) Plan1, b)Plan2

Şekil 4.17'de sunulan iki farklı kat planına ait optimizasyon sonuçlarında görüldüğü üzere, kolon konumları seyreltilmiş ve kesit boyutları minimum değerlere indirgenmiştir. Her iki yapı için de geçerli tasarımlar elde edilmiş, yazılım kapsamında kontrol edilen hiçbir geometrik veya yapısal ceza unsuruna yakalanmaksızın optimum çözümler oluşturulmuştur.

5. SONUÇLAR VE ÖNERİLER

Bu çalışmada yapı mühendisliğinde optimizasyon temelli karar destek sağlayan bir sistem geliştirilmiş ve önerilen yaklaşım, çok boyutlu bir test süreciyle kapsamlı şekilde değerlendirilmiştir. Görüntü işleme, optimizasyon algoritmaları ve analiz süreçleri tek bir bütünlük çatı altında birleştirilerek, yapısal tasarımların hem otomatik hem de verimli biçimde gerçekleştirilmesi sağlanmıştır.

Görüntü işleme modülü, farklı plan geometrilerinde yüksek doğrulukla çalışmış; kolon konumlarını ve aks yapılarını istikrarlı şekilde algılayarak sistemin genellenebilirliğini desteklemiştir. Optimizasyon sürecinde ise test edilen algoritmalar arasında IAOA, en düşük maliyetli ve geçerli çözümleri üreterek öne çıkmıştır. APSO yakın performanslar sergilerken, DE daha az verimli ancak istikrarlı sonuçlar üretmiştir. Dolayısıyla, IAOA algoritması probleme özgü en yüksek performansı sergileyen yöntem olarak belirlenmiş ve bu doğrultuda, çalışma kapsamında geliştirilen yazılımın nihai sürümünde yalnızca bu algoritmanın kullanılmasına karar verilmiştir.

Çalışmada, optimizasyon problemini temsil etmek üzere iki farklı değişken vektörü kurgusu (DV1 ve DV2) test edilmiştir. Yapılan karşılaştırmalar sonucunda, DV1'in hem daha düşük maliyetli hem de yönetmelik açısından geçerli çözümleri daha hızlı ve istikrarlı şekilde ürettiği görülmüştür. Bu nedenle sistemin nihai sürümünde DV1 yapısı tercih edilmiştir.

Yapı yüksekliği ve deprem senaryosu gibi kullanıcı girdilerinin, hem çözüm kalitesi hem de hesaplama süresi üzerinde önemli etkiler oluşturduğu görülmüştür. Özellikle yüksek katlı yapılarda analiz süresi belirgin şekilde artmış, algoritmaların optimuma ulaşma süreci daha karmaşık hale gelmiştir.

Ayrıca, yapı üzerindeki yüklerin tanımlanma biçimi de tasarım sonuçlarını doğrudan etkilemiştir. Bu çalışmada, döşemelerin tanımlanmadığı bir modelleme sistemine geçici olarak yer verildiğinden, düşey yükler kirişlere yaklaşık yöntemlerle atanmıştır. Bu yaklaşım, özellikle açıklıklardaki iç kuvvet dağılımlarını etkileyerek donatı tasarımında STA4CAD gibi programlarla fark oluşmasına neden olmuştur. Ancak bu durum, sürecin otomasyonunu test etmeye yönelik bu ilk sürüm için kabul edilebilir düzeydedir.

Paralel hesaplama desteği, sistemin işlem süresini ciddi oranda azaltmış; özellikle 8 işçi ile yapılan analizler zaman açısından en verimli yapılandırmayı sunmuştur. Ayrıca,

geliştirilen sistemin çıktıları kullanıcıya hem metin hem de görsel olarak sunulurken, tasarım kararlarının denetlenebilir ve yorumlanabilir hale gelmesi sağlanmıştır.

Genel olarak, önerilen yöntem farklı senaryolarda tutarlı ve başarılı performanslar sergileyerek, hem mühendislik hesaplarını otomatikleştirme hem de yapısal tasarımı iyileştirme potansiyelini açıkça ortaya koymuştur.

Bu çalışmada geliştirilen otomasyon tabanlı yapı tasarımı sistemi, ilk sürüm itibarıyla belirli sınırlamalar içermekle birlikte, ilerleyen çalışmalarda aşağıda sıralanan başlıklarda yapılacak iyileştirmelerle hem teknik kapsam hem de kullanıcı deneyimi açısından önemli ölçüde geliştirilebilir:

- **Malzeme Kütüphanesi Entegrasyonu:** Yapıya ait beton ve donatı malzemelerinin tüm mekanik özelliklerini içeren bir malzeme kütüphanesi oluşturularak, kullanıcı tarafından yalnızca sınıf seçimi yapılması yeterli hale getirilebilir. Böylece veri girişi kolaylaştırılabilir ve standartlara uygunluk artırılabilir.
- **Yönetmelik Kontrollerinin Genişletilmesi:** Mevcut durumda yalnızca sınırlı sayıda yönetmelik kriterine göre yapılan kontrollerin, eksiksiz bir tasarım doğrulaması sağlayacak şekilde kapsamının genişletilmesi önerilmektedir. Buna ek olarak, farklı yönetmeliklerin (örneğin TBDY, ACI, Eurocode vb.) sisteme entegre edilmesiyle, kullanıcı tercihinine bağlı olarak yönetmelik bazlı doğrulama seçeneği sunulması mümkün hale gelebilir.
- **Kullanıcı Arayüzü Geliştirilmesi:** Yazılımın kod temelli çalışma yapısından arındırılarak grafiksel bir kullanıcı arayüzü (GUI) ile desteklenmesi, mühendislik bilgisi olan fakat yazılım bilgisi sınırlı kullanıcıların da sistemi etkin biçimde kullanmasını sağlayacaktır.
- **Kiriş Sisteminin İyileştirilmesi:** Kiriş yerleşimi ve açıklıklarının, otomatik olarak belirlenen kolon yerleşimlerine göre daha tutarlı şekilde ayarlanabilmesi için kiriş yerleşim algoritması geliştirilebilir.
- **Perde ve Döşeme Elemanlarının Entegrasyonu:** Sistem mimarisine perde duvarlar ve döşeme plakları da dahil edilerek, bu elemanların hem modelleme hem de yönetmelik kontrolleri açısından değerlendirilmesi sağlanabilir. Böylece yapı modellemesi daha bütüncül ve gerçekçi hale gelecektir.
- **Yükleme Modelinin Geliştirilmesi:** Eşdeğer deprem yüklemesi gibi idealize edilmiş yükleme yöntemlerinin ötesine geçilerek, zaman tanım alanında dinamik

analizlerin yapılabileceği gerçekçi deprem senaryoları sisteme entegre edilebilir. Ayrıca döşemeler tanımlandıktan sonra düşey yüklerin döşeme alanlarına yayılı yük olarak atanması ve ardından kirişlere otomatik zarf yükleriyle dağıtılması, diğer yapı analizi yazılımlarında olduğu gibi daha doğru bir yük aktarımı sağlayacaktır.

- **Görüntü İşleme Modülünün Zenginleştirilmesi:** Mimari planlardan bilgi çıkartmak amacıyla kullanılan görüntü işleme algoritmalarına alternatif işlevler ve farklı ön işleme teknikleri entegre edilerek, farklı plan türlerine daha esnek ve kararlı biçimde uyum sağlanabilir.
- **Kolon Konumlarının Kullanıcı Tarafından Kısıtlanabilmesi:** Gerçek proje uygulamalarında mimar veya mühendislerin bazı kolon konumlarını zorunlu ya da yasaklı olarak belirlemesi yaygındır. Bu gereksinimi karşılamak amacıyla, kullanıcıya belirli kolon yerlerini “kesin olacak” veya “kesin olmayacak” biçiminde işaretleme olanağı sunulabilir.
- **Görüntü İşleme veya DXF’ten Okuma Opsiyonu Sunulması:** Kullanıcıların mimari planlarını sisteme farklı formatlarda (örneğin görüntü dosyası veya DXF) tanıtabilmesini sağlayan çok formatlı veri girişi opsiyonu geliştirilebilir. Böylece farklı plan türlerinin kolayca sisteme entegre edilmesi mümkün hale gelir.

Sonuç olarak, bu çalışma kapsamında geliştirilen optimizasyon tabanlı yapı tasarımı sistemi, mimari planlardan başlayarak analiz ve değerlendirme aşamasına kadar uzanan süreci büyük ölçüde otomatikleştirmiş ve mühendislik tasarım sürecine sistematik bir karar destek altyapısı kazandırmıştır. Geliştirilen otomasyon sistemi sayesinde, kolon topolojisi ile kolon ve kiriş kesitleri eşzamanlı olarak optimize edilmiş, böylece taşıyıcı sistemin tüm bileşenleri bütüncül şekilde belirlenebilmiştir. Görüntü işleme modülü aracılığıyla, mimari planlardan muhtemel kolon yerleşimleri sistematik olarak çıkarılmış ve bu bilgiler doğrudan yapısal modelleme sürecine entegre edilmiştir. Optimizasyon sürecinde, birden fazla algoritma test edilerek probleme özgü en uygun algoritma belirlenmiş, bu sayede yalnızca geçerli çözümler değil, optimum maliyete sahip tasarımlar da elde edilmiştir. Farklı algoritmaların performanslarının karşılaştırmalı biçimde değerlendirilmesi, sistemin hem esnekliğini hem de genellenebilirliğini ortaya koyarken, paralel hesaplama ve kullanıcıya sunulan çıktılar sayesinde pratik kullanım potansiyeli de güçlendirilmiştir. Gerçekleştirilen bütünlük yaklaşım, geleneksel

yöntemlere kıyasla daha verimli ve sistematik bir yapı tasarımı süreci sunarak, dijitalleşen mühendislik uygulamaları için güçlü bir temel oluşturmuştur.



KAYNAKÇA

- Abualigah, L., Diabat, A., Mirjalili, S., Abd Elaziz, M., and Gandomi, A. H. (2021). The Arithmetic Optimization Algorithm. *Computer Methods in Applied Mechanics and Engineering*, 376, 113609.
- Alba, E., Luna, F., Nebro, A. J., and Troya, J. M. (2004). Parallel heterogeneous genetic algorithms for continuous optimization. *Parallel Computing*, 30, 699–719.
- Alemdar, Z. F., Bilici, S., and Alemdar, F. (2018). Deformation Measurement in a Steel Bridge System By Using Image Processing Method. *Sigma J Eng & Nat Sci* 9, 9(2), 149–156.
- Amdahl, G. M. (2007). Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities. *IEEE SolidState Circuits Newsletter*, 12(3), 19–20.
- Çakıroğlu, A., and Özer, E. (1990). Dikdörtgen ve daire betonarme kesitlerde taşıma gücü formülleri ve yaklaşıklık mertebeleri. *Teknik Dergi*, 1(1).
- Camp, C. V., Pezeshk, S., and Hansson, H. (2003). Flexural Design of Reinforced Concrete Frames by Genetic Algorithm. *Journal of Structural Engineering*, 129(1), 105–115.
- Çankaya, G., Arslan, M. H., and Ceylan, M. (2013). Görüntü işleme ve yapay sinir ağları yöntemleri ile betonun basınç dayanımının belirlenmesi. *Selçuk Üniversitesi Mühendislik, Bilim ve Teknoloji Dergisi*, 1(1), 1–12.
- Chen, H., and Chan, I. (2021). Automation in safety planning in construction: Insights from a literature review. *Computing in Civil Engineering*, 819–826.
- Chen, Q., and Chen, Y. (2020). Construction Material Concrete Quality Detection Method based on Ultrasonic Tomography Technology. In *2020 IEEE International Conference on Industrial Application of Artificial Intelligence (IAAI)* (pp. 64–69). IEEE.
- Chen, S. S., and Shirolé, A. M. (2006). Integration of information and automation technologies in bridge engineering and management: Extending the state of the art. *Transportation Research Record*, (1976), 3–12.
- Chisari, C., and Bedon, C. (2016). Multi-objective optimization of FRP jackets for improving the seismic response of reinforced concrete frames. *American Journal of Engineering and Applied Sciences*, 9(3), 669–679.
- Cui, C., and Jiang, B. (2014). A morphogenesis method for shape optimization of framed structures subject to spatial constraints. *Engineering Structures*, 77, 109–118.
- Dastres, R., and Soori, M. (2021). Advanced Image Processing Systems. *International Journal of Imaging and Robotics*, 21(1), 27–44.
- Dobbs, M. W., and Nelson, R. B. (1976). Application of optimality criteria to automated structural design. *AIAA Journal*, 14(10), 1436–1443.
- Dorigo, M., and Stützle, T. (2004). *Ant Colony Optimization*. The MIT Press.

- Eleftheriadis, S., Duffour, P., and Mumovic, D. (2018). BIM-embedded life cycle carbon assessment of RC buildings using optimised structural design alternatives. *Energy and Buildings*, 173, 587–600.
- Fan, S. K. S., and Chang, J. M. (2009). A parallel particle swarm optimization algorithm for multi-objective optimization problems. *Engineering Optimization*, 41(7), 673–697.
- Farrar, C. R., and Worden, K. (2007). An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1851), 303–315.
- Ferrer, B., Pomares, J. C., Irles, R., Espinosa, J., and Mas, D. (2013). Image processing for safety assessment in civil engineering. *Applied Optics*, 52(18), 4385–4390.
- Fitriyah, H., and Wihandika, R. C. (2018). An Analysis of RGB, Hue and Grayscale under Various Illuminations. In *3rd International Conference on Sustainable Information Engineering and Technology, SIET 2018 - Proceedings* (pp. 38–41). IEEE.
- Frazier, P. I. (2018). Recent Advances in Optimization and Modeling of Contemporary Problems. *Recent Advances in Optimization and Modeling of Contemporary Problems*, 255–278.
- Fu, J., Huang, H., Chen, S., Zhang, Y., and An, H. (2025). A topology , material and beam section optimization method for complex structures. *Engineering Optimization ISSN:*, 57(6), 1467–1485.
- Ghandi, E., Shokrollahi, N., and Nasrolahi, M. (2017). Optimum cost design of reinforced concrete slabs using cuckoo search optimization algorithm. *Iran Univ. Sci. Technol*, 7(4), 539–564.
- Gonzalez, R. C., and Woods, R. E. (2017). *Digital Image Processing. The Electrical Engineering Handbook* (4th ed.). Pearson.
- Groenwold, A. A., and Hindley, M. P. (2002). Competing parallel algorithms in structural optimization. *Structural and Multidisciplinary Optimization*, 24(5), 343–350.
- Gunwant, D., and Misra, A. (2012). Topology optimization of continuum structures using optimality criterion approach in ANSYS. *International Journal of Advances in Engineering & Technology*, 5(1), 470.
- Hajela, P. (1990). Genetic search - An approach to the nonconvex optimization problem. *AIAA Journal*, 28(7), 1205–1210.
- Halim, A. H., Ismail, I., and Das, S. (2021). *Performance assessment of the metaheuristic optimization algorithms: an exhaustive review. Artificial Intelligence Review* (Vol. 54). Springer Netherlands.
- Hasançebi, O., Çarbaş, S., Doğan, E., Erdal, F., and Saka, M. P. (2009). Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures. *Computers and Structures*, 87(5–6), 284–302.
- Hoang, N. D. (2018). Detection of Surface Crack in Building Structures Using Image Processing Technique with an Improved Otsu Method for Image Thresholding. *Advances in Civil Engineering*, 1, 3924120.

- Holland, J. H. (1992). *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.
- Kashani, A. R., Camp, C. V., Rostamian, M., Azizi, K., and Gandomi, A. H. (2022). *Population-based optimization in structural engineering: a review*. *Artificial Intelligence Review* (Vol. 55). Springer Netherlands.
- Kaveh, A., Hassani, B., Shojaee, S., and Tavakkoli, S. M. (2008). Structural topology optimization using ant colony methodology. *Engineering Structures*, 30, 2559–2565.
- Kaveh, Ali, and Hamedani, K. B. (2022). Improved arithmetic optimization algorithm and its application to discrete structural optimization. *Structures*, 35, 748–764.
- Kennedy, J., and Eberhart, R. (1995)a. Particle Swarm Optimization. In *ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942–1948).
- Kennedy, J., and Eberhart, R. (1995)b. Particle Swarm Optimization. In *ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942–1948).
- Kim, C., Kim, B., and Kim, H. (2013). 4D CAD model updating using image processing-based construction progress monitoring. *Automation in Construction*, 35, 44–52.
- Kirkpatrick, S., Gelatt Jr, C. D., and Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671–680.
- Kociecki, M., and Adeli, H. (2015). Engineering Applications of Artificial Intelligence Shape optimization of free-form steel space-frame roof structures with complex geometries using evolutionary computing. *Engineering Applications of Artificial Intelligence*, 38, 168–182.
- Kulambayev, B., Gleb, B., Katayev, N., Menglibay, I., and Momynkulov, Z. (2024). Real-Time Road Damage Detection System on Deep Learning Based Image Analysis. *International Journal of Advanced Computer Science and Applications*, 15(9), 1051–1061.
- Lim, J., You, C., and Dayyani, I. (2020). Multi-objective topology optimization and structural analysis of periodic spaceframe structures. *Materials and Design*, 190, 108552.
- Major, I., and Gavardashvili, G. (2022). Automation and robotization in civil engineering. *Zeszyty Naukowe Politechniki Częstochowskiej. Budownictwo*, 28, 55–60.
- Mamodiya, U., and Sharma, P. (2014). Review in Industrial Automation. *IOSR Journal of Electrical and Electronics Engineering*, 9(3), 33–38.
- Martínez-Frutos, J., Herrero-Pérez, D., Kessler, M., and Periago, F. (2016). Robust shape optimization of continuous structures via the level set method. *Computer Methods in Applied Mechanics and Engineering*, 305, 271–291.
- McCormick, N., and Lord, J. (2012). Digital image correlation for structural measurements. In *Proceedings of the Institution of Civil Engineers-Civil Engineering* (Vol. 165, pp. 185–190).
- Melih Nigdeli, S., and Bekdaş, G. (2019). The effect of the consideration of slab dimensions on optimum design of reinforced concrete beams. *European Journal of Engineering and Natural Sciences*, 3(2), 81–85.

- Miano, John. (1999). *Compressed image file formats : JPEG, PNG, GIF, XBM, BMP*. Addison Wesley.
- Mousa, M. A., Yussof, M. M., Udi, U. J., Nazri, F. M., Kamarudin, M. K., Parke, G. A. R., ... Ghahari, S. A. (2021). Application of digital image correlation in structural health monitoring of bridge infrastructures: A review. *Infrastructures*, 6(12).
- Niaki, M. I., Maheri, M. R., and Bagheri, M. (2016). Optimum Design of 2-D Reinforced Concrete Frames Using a Genetic Algorithm. *The Turkish Online Journal of Design, Art and Communication*, 6(AGSE), 1909–1919.
- Padmavathi, K., and Thangadurai, K. (2016). Implementation of RGB and grayscale images in plant leaves disease detection - Comparative study. *Indian Journal of Science and Technology*, 9(6), 4–9.
- Pan, Z., Guo, H., and Li, Y. (2017). Automated Method for Optimizing Feasible Locations of Mobile Cranes Based on 3D Visualization. *Procedia Engineering*, 196(June), 36–44.
- Paya-Zaforteza, I., Yepes, V., Hospitaler, A., and González-Vidoso, F. (2009). CO₂-optimization of reinforced concrete frames by simulated annealing. *Engineering Structures*, 31(7), 1501–1508.
- Potenza, F., Castelli, G., Gattulli, V., and Ottaviano, E. (2017). Integrated Process of Images and Acceleration Measurements for Damage Detection. *Procedia Engineering*, 199, 1894–1899.
- Pourriyanezhad, E., Rahami, H., and Mirhosseini, S. M. (2021). Truss optimization using eigenvectors of the covariance matrix. *Engineering with Computers*, 37(3), 2207–2224.
- Prabakaran, T., Periasamy, P., Mugendiran, V., and Ramanan. (2020). Studies on application of image processing in various fields: An overview. *IOP Conference Series: Materials Science and Engineering*, 961(1), 0–13.
- Qiao, S., Han, X., Zhou, K., and Ji, J. (2016). Seismic analysis of steel structure with brace configuration using topology optimization. *Steel and Composite Structures*, 21(3), 501–515.
- Rao, R. V. (2016). Jaya : A simple and new optimization algorithm for solving constrained and unconstrained optimization problems Jaya : A simple and new optimization algorithm for solving constrained and unconstrained optimization problems Bacteria Foraging Optimization (. *International Journal of Industrial Engineering Computations*, 7(1), 19–34.
- Sadollah, A., Eskandar, H., Bahreininejad, A., and Kim, J. H. (2015). Water cycle, mine blast and improved mine blast algorithms for discrete sizing optimization of truss structures. *Computers and Structures*, 149, 1–16.
- Sahab, M. G., Toropov, V. V., and Ashour, A. F. (2004). A hybrid genetic algorithm for structural optimization problems. *Asian Journal of Civil Engineering (Building And Housing)*, 5, 121–143.

- Saka, M. P., Hasançebi, O., and Geem, Z. W. (2016). Metaheuristics in structural optimization and discussions on harmony search algorithm. *Swarm and Evolutionary Computation*, 28, 88–97.
- Scholl, I., Aach, T., Deserno, T. M., and Kuhlen, T. (2011). Challenges of medical image processing. *Computer Science - Research and Development*, 26(1–2), 5–13.
- Serra, M., and Venini, P. (2006). On some applications of ant colony optimization metaheuristic to plane truss optimization. *Structural and Multidisciplinary Optimization*, 32(6), 499–506.
- Storn, R., and Price, K. (1995). Differential evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces. *Journal of Global Optimization*, 16.
- Talatahari, S., Kaveh, A., and Sheikholeslami, R. (2012). Chaotic imperialist competitive algorithm for optimum design of truss structures. *Structural and Multidisciplinary Optimization*, 46(3), 355–367.
- TBDY. (2018). *Türkiye Bina Deprem Yönetmeliği*. Afet ve Acil Durum Yönetimi Başkanlığı, Ankara.
- Thomas, S. M., and Arulraj. G, P. (2017). Optimization of Singly Reinforced Rc Beams. *International Journal of Research -GRANTHAALAYAH*, 5(2), 199–207.
- Topçu, A. (2019). *Betonarme I*. Eskişehir Osmangazi Üniversitesi.
- TS 498. (2021). *Yapı Elemanlarının Boyutlandırılmasında Alınacak Yüklerin Hesap Değerleri*. Ankara.
- TS 500. (2000). *Betonarme Yapıların Tasarım ve Yapım Kuralları*. Ankara.
- Tsoulos, I. G., Tzallas, A., and Tsalikakis, D. (2016). PDoublePop: An implementation of parallel genetic algorithm for function optimization. *Computer Physics Communications*, 209, 183–189.
- Uygunoğlu, T., and Çetingül, F. (2024). Görüntü İşleme Yöntemi ile Betonda Oluşan Karbonatlaşmanın Değerlendirilmesi. *Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 7(2), 802–817.
- Vähä, P., Heikkilä, T., Kilpeläinen, P., Järviluoma, M., and Gambao, E. (2013). Extending automation of building construction - Survey on potential sensor technologies and robotic applications. *Automation in Construction*, 36, 168–178.
- Yang, G., Wu, J., and Hu, Q. (2019). Rapid detection of building cracks based on image processing technology with double square artificial marks. *Advances in Structural Engineering*, 22(5), 1186–1193.
- Yuan, X., Dai, X., and Wu, L. (2015). A mutative-scale pseudo-parallel chaos optimization algorithm. *Soft Computing*, 19(5), 1215–1227.
- Zakian, P., and Kaveh, A. (2023). Seismic design optimization of engineering structures: a comprehensive review. *Acta Mechanica*, 234(4), 1305–1330.

- Zawad, M. R. S., Zawad, M. F. S., Rahman, M. A., and Priyom, S. N. (2021). A comparative review of image processing based crack detection techniques on civil engineering structures. *Journal of Soft Computing in Civil Engineering*, 5(3), 58–74.
- Zhong, C., and Pusheng, F. (1995). A parallel algorithm for constrained optimization problems. *Journal of Computational and Applied Mathematics*, 61(2), 225–230.
- http-1: <https://opensees.berkeley.edu/> (Eriřim tarihi: 21.07.2024)
- http-2: <https://www.mathworks.com/> (Eriřim tarihi: 07.01.2024)
- http-3: <https://tdth.afad.gov.tr/TDTH/main.xhtml> (Eriřim tarihi: 04.03.2025)
- http-4: https://www.itu.int/dms_pubrec/itu-r/rec/bt/r-rec-bt.601-7-201103-i!!pdf-e.pdf (Eriřim tarihi: 22.05.2024)
- http-5: <https://www.csiamerica.com/products/sap2000> (Eriřim tarihi: 17.05.2025)
- http-6: <https://www.sta4.net/staproperty14.aspx> (Eriřim tarihi: 12.04.2025)

ÖZGEÇMİŞ

ORCID NO: 0000-0002-3569-0297

Ad Soyad : Talha Orhan

Yabancı Dil : İngilizce

Eğitim ve Mesleki Geçmişi:

- Yüksek Lisans : Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, İnşaat Mühendisliği Anabilim Dalı, 2018
- Lisans : Gediz Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği Bölümü, 2015
- 2016, Araştırma Görevlisi, Eskişehir Teknik Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği Bölümü

Yayınları ve/veya Bilimsel/Sanatsal Faaliyetleri:

- Orhan, T., & Taşkın, K. (2021). Automated topology design of high-rise diagrid buildings by genetic algorithm optimization. The structural design of tall and special buildings, 30(9), e1853.
- Over, D., Ozbakan, N., Şamdan, F., Orhan, T., & Canbaz, M. (2025). An experimental and numerical study on the effects of cold joint location and angle in concrete. Journal of Building Engineering, 99, 111529.