



**T.C. İSTANBUL TİCARET  
ÜNİVERSİTESİ**

**FEN BİLİMLERİ ENSTİTÜSÜ**

**KARMAŞIK YÜZEYLERİN DÜZLEŞTİRMESİNE YÖNELİK  
YAKLAŞIM**

**Erdem YAVUZ**

**Danışman  
Prof. Dr. Rifat YAZICI**

**DOKTORA TEZİ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI  
İSTANBUL - 2018**

## KABUL VE ONAY SAYFASI

**Erdem YAVUZ** tarafından hazırlanan "**Karmaşık Yüzeylerin Düzleştirilmesine Yönelik Yaklaşım**" adlı tez çalışması 04/07/2018 tarihinde aşağıdaki jüri üyeleri önünde başarı ile savunularak, İstanbul Ticaret Üniversitesi Fen Bilimleri Enstitüsü **Bilgisayar Mühendisliği Anabilim Dalı**'nda **DOKTORA TEZİ** olarak kabul edilmiştir.

**Danışman**                      **Prof. Dr. Rıfat YAZICI**  
İstanbul Ticaret Üniversitesi



**Jüri Üyesi**                      **Prof. Dr. Selim AKYOKUŞ**  
Doğuş Üniversitesi



**Jüri Üyesi**                      **Doç. Dr. Serhat ÖZEKES**  
Üsküdar Üniversitesi



**Jüri Üyesi**                      **Dr. Öğr. Üyesi M. Cem KASAPBAŞI**  
İstanbul Ticaret Üniversitesi



**Jüri Üyesi**                      **Dr. Öğr. Üyesi Muhammed Ali AYDIN**  
İstanbul Üniversitesi



**Onay Tarihi : 23 TEMMUZ 2018**



**Prof. Dr. Necip ŞİMŞEK**  
Enstitü Müdürü

## AKADEMİK VE ETİK KURALLARA UYGUNLUK BEYANI

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

18.05.2018



**Erdem YAVUZ**

# İÇİNDEKİLER

## Sayfa

İÇİNDEKİLER.....	i
ÖZET .....	iii
ABSTRACT .....	v
TEŞEKKÜR.....	vii
ŞEKİLLER.....	viii
ÇİZELGELER.....	xii
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. LİTERATÜR ÖZETİ.....	6
2.1. Yüzey Projeksiyon Teknikleri .....	11
2.1.1. Klasik doku temsilindeki problem.....	12
2.1.2. Yüzey metrik distorsiyonu.....	13
2.1.3. Sabit-kenarlı izdüşüm algoritmaları.....	14
2.1.4. Serbest-kenarlı izdüşüm algoritmaları.....	16
2.1.4.1. En küçük karesel açı-korumalı izdüşüm.....	17
2.1.4.2. En olası benzerlik (ASAP) .....	17
2.1.4.3. En olası katılık (ARAP).....	18
2.1.4.4. Açı tabanlı düzleştirme (ABF).....	18
2.2. Delaunay Üçgenlemesi .....	19
3. MATERYAL ve YÖNTEM .....	24
3.1. Üçgensel Yüzeyin Parametrelenmesi.....	24
3.2. Barisentrik İzdüşüm .....	25
3.3. Enerji Tabanlı Yüzey Düzleştirme.....	29
3.3.1. Enerji fonksiyonunun tanımı .....	29
3.3.2. Enerji yayma.....	30
3.3.3. Ceza fonksiyonu .....	32
3.3.4. Doğruluk ölçümü .....	33
3.3.4.1 Alan doğruluğu.....	33
3.3.4.2 Biçim/Kenar doğruluğu.....	34
3.3.5. Enerji yayma algoritması.....	34
3.4. Dinamik Sinir Ağlar .....	35
3.4.1. Dinamik sinir ağ yapısı.....	36
3.4.2. Hopfield modeli.....	39
3.5. Önerilen Yüzey Düzleştirme Yaklaşımı .....	40
3.5.1. Sanal sınır poligonu belirleme .....	41
3.5.2. Başlangıç izdüşüm algoritması .....	44
3.5.2.1. Dinamik sinir ağı modeli .....	46
3.5.2.2. Yakınsama araştırması.....	48
3.5.3. Değişken adım büyüklüğüne dayalı enerji yayma algoritması.....	50
4. ARAŞTIRMA BULGULARI VE TARTIŞMA.....	58
4.1. Geometrik Düzleştirme Sonuçları .....	58
4.1.1. Aydınlatıcı örnekler .....	58
4.1.2. Sanal sınır düğüm sayısının izdüşüm sonucuna etkisi.....	75
4.1.3. Lambda komşuluk ağırlık matrisinin izdüşüm sonucuna etkisi .....	99
4.1.4. Başlangıç izdüşüm hızı .....	108
4.2. Mekanik Düzleştirme Sonuçları .....	109
4.2.1. Aydınlatıcı örnekler .....	109
4.2.2. Değişken adım büyüklüğünün enerji yayma süresine etkisi .....	120

5. SONUÇ VE ÖNERİLER.....	125
KAYNAKLAR .....	129
EK A. Denklemler .....	136
EK B. Şekiller .....	139
ÖZGEÇMİŞ.....	141



## ÖZET

**Doktora Tezi**

### **KARMAŞIK YÜZEYLERİN DÜZLEŞTİRİLMESİNE YÖNELİK YAKLAŞIM**

**Erdem YAVUZ**

**İstanbul Ticaret Üniversitesi  
Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalı**

**Danışman: Prof. Dr. Rifat YAZICI  
2018, 160 sayfa**

3B tarama teknolojisinin çok hızlı bir şekilde ilerlemesiyle birlikte serbest biçimli yüzeyler hızla gün yüzüne çıkmakta ve çok sayıda farklı uygulamalarda sayısız endüstriyel ürün tasarımında kullanılmaktadır. Serbest biçimli yüzeyler günümüze ait grafik uygulamalarında önemli bir rol oynamaktadır. Yüzey çizdirme, yüzey kaplama, yüzey örnekleme vb. serbest biçimli yüzeylerle ilgili uygulamaların sonuçları, yüzey parametrelere oldukça bağımlıdır. Bu uygulamalarda çoğunlukla parametrelere biçim koruyan türden olması ve hızlı bir şekilde yapılması arzulanan bir durumdur.

Bu çalışmada geometrik ve mekanik düzleştirme tekniklerini kullanarak konveks biçimli 3B yüzey yamalarının düzlemsel karşılıklarının etkili bir biçimde hesaplanmasına yönelik bir yaklaşım geliştirilmiştir. Bu yaklaşımda, 3B yüzey yamasının 2B parametrik düzlem izdüşümüne ilişkin koordinatların hesaplanması için barisentrik izdüşüm teorisine dayanan bir dinamik YSA (Yapay Sinir Ağı) modeli gerçekleştirilmiştir. Bu modelde, barisentrik izdüşüm teorisi yardımıyla yüzey yamasına ilişkin elde edilen yerel geometri bilgisi geri beslemeli dinamik sinir ağına aktarılır. Dinamik sinir ağı modeliyle, izdüşümü yapılacak yüzeye ilişkin noktalar parametrik uzayda denge durumuna ulaşıncaya kadar yinelemeli olarak hareket ettirilir. Geliştirilen yaklaşım, nispeten büyük (nokta sayısı 1.5K'dan daha fazla olan) yüzey yamalarının başlangıç izdüşümüne ilişkin parametrik uzaydaki koordinatlarını, ters matris hesaplamasını kullanan geleneksel yöntemle kıyasla etkili bir şekilde daha az sürede hesaplamaktadır. Önerilen yaklaşım, farklı boyutlardaki yüzey yamaları üzerinde test edilerek yöntemin etkinliği ortaya konulmuştur. Geometrik düzleştirme tekniğiyle elde edilen başlangıç izdüşümü mekanik düzleştirme tekniğiyle iyileştirilmiştir. İyileştirme süreci, düzlemde kurulan kütle-yay sistemiyle modellenerek değişken adım büyüklüğüne dayalı bir enerji yayma algoritmasıyla enerji tabanlı olarak gerçekleştirilmiştir.

**Anahtar Kelimeler:** Bařlangıç izdüşümü, dinamik sinir ađı, kütle-yay sistemi, parametreleme, üçgensel kafes, yüzey düzleřtirme.



## **ABSTRACT**

**Ph.D. Thesis**

### **AN APPROACH TO FLATTENING OF COMPLEX SURFACES**

**Erdem YAVUZ**

**Istanbul Commerce University  
Graduate School of Applied and Natural Sciences  
Department of Computer Engineering**

**Supervisor: Prof. Dr. Rifat YAZICI  
2018, 160 pages**

With the rapid advances in 3D scanning technology, free-form surfaces quickly come to life and are used in numerous industrial products design in many different applications. Free-form surfaces play an important role in contemporary graphics applications. Results of applications like surface rendering, surface tessellation, surface sampling, etc. related to freeform surfaces are highly dependent on the surface parametrization. In such applications, it is often desired that the parametrization be shape-preserving and performed quickly.

In this study, an approach has been developed for the effective computation of the planar equivalents of convex 3D surface patches using geometric and mechanical flattening techniques. In this approach, a dynamic ANN (artificial neural network) model based on barycentric mapping theory has been constructed and implemented in order to compute the coordinates of 2D parametric equivalent of 3D surface patch. In this model, the local geometry information about the surface patch is transferred to the recurrent ANN with the help of barycentric mapping theory. The dynamic neural network model iteratively moves the points related to the surface to be projected in the parameterized space until reaching the equilibrium state. The developed approach computes the coordinates of the parametric space of the initial mapping of relatively large surface patches (more than 1.5K vertices) more efficiently than the conventional method using inverse matrix computation. The proposed approach has been tested on surface patches of different sizes to demonstrate the effectiveness of the method. The initial projection obtained by the geometric flattening technique is improved by the mechanical flattening technique. The relaxation process is modeled by the mass-spring system established in the plane and is realized by energy releasing algorithm based on variable step size.



**Keywords:** Initial mapping, dynamic neural network, spring-mass system, parametrization, triangular mesh, surface flattening.



## TEŐEKKÜR

Doktora alıřmam boyunca bilgi ve tecrübesiyle bana rehberlik eden ve fikirleriyle hep aydınlatıcı olan ok deęerli danıřman hocam Prof. Dr. Rifat YAZICI'ya en samimi řükranlarımı sunarım.

Doktora sürecinde manevi desteklerini benden esirgemeyen deęerli alıřma arkadaşlarım Arař. Gör. Can EYÜPOęLU, Arař. Gör. Ufuk řANVER, Öğr. Gör. Fatma Nur AKI ve Arař. Gör. Dr. řeyda EYÜPOęLU'ya teőekkürlerimi sunarım.

Maddi ve manevi destekleri ve gösterdikleri sonsuz sabır ve sevgi için her zaman minnettar olduęum canım aileme en içten duygularıyla teőekkürlerimi sunarım.

Erdem YAVUZ  
İSTANBUL, 2018

## ŞEKİLLER

	Sayfa
Şekil 1.1. Yüzey tanımlama modeli (Abdul-Rahman vd., 2016) .....	3
Şekil 2.1. Öklid yüzey üzerinde doku gösterimi: (a) Orijinal yüzey üzerinde bir doku, (b) Düzlem yüzey üzerinde aynı doku (Abdul-Rahman vd., 2016) .....	12
Şekil 2.2. Öklid olmayan yüzeyde doku gösterimi: (a) Orijinal yüzey üzerindeki doku, (b) Düzlem yüzey üzerindeki doku (Abdul-Rahman vd., 2016) .....	13
Şekil 2.3. Sabit-kenarlı izdüşümde kullanılan yay modeli .....	15
Şekil 2.4. Bir üçgenin çevrel çemberi .....	19
Şekil 2.5. Boş çevrel çember özelliğini sağlayan üçgenler (Gartner ve Hoffmann, 2013) .....	20
Şekil 2.6. Delaunay koşulunu sağlamayan üçgenleme (De Loera vd., 2010) .....	21
Şekil 2.7. Ortak kenar değiştirme işlemi sayesinde Delaunay koşulunu sağlayan üçgenleme (De Loera vd., 2010) .....	21
Şekil 2.8. Delaunay üçgenleme (a) ile Voroni diyagramı (b) arasındaki ilişki .....	22
Şekil 3.1. 3B uzayda tanımlı bir yüzeyin (solda) 2B parametrik uzaydaki karşılığı (sağda) (Hormann vd., 2007) .....	24
Şekil 3.2. Düğümlerin yeniden etiketlenmesine ilişkin bir örnek .....	26
Şekil 3.3. Kütle-yay sistemi .....	29
Şekil 3.4. Ceza fonksiyonları (Wang vd., 2002) .....	32
Şekil 3.5. Bir yutucu sistemin yörüngesinin 2B çizimi (Gupta vd., 2004) .....	37
Şekil 3.6. Temel dinamik sinir ağ yapısı (Gupta vd., 2004) .....	38
Şekil 3.7. Hopfield ağı mimarisi (Du ve Swamy, 2006) .....	39
Şekil 3.8. Yüzey düzleştirme sürecinin akış diyagramı .....	41
Şekil 3.9. 3B (a) ve 2B (b) uzaydaki sanal sınır yörüngeleri .....	42
Şekil 3.10. Çeşitli çözünürlükte oluşturulan sanal sınır poligonları: Sınır düğümü sayısı (a) $n_v=4$ ; (b) $n_v=6$ ; (c) $n_v=8$ ; (d) $n_v=16$ .....	43
Şekil 3.11. Önerilen dinamik sinir ağı modeli .....	47
Şekil 3.12. İki-norm kalıntı hatası .....	49
Şekil 3.13. Değişken ve sabit adım büyüklüğü ile alan doğruluğunun değişimi .....	51
Şekil 3.14. Değişken ve sabit adım büyüklüğü ile kenar doğruluğunun değişimi .....	52
Şekil 3.15. Nihai değişken adım büyüklüğünün alan ve kenar doğruluklarına göre değişimi .....	53
Şekil 3.16. Nihai değişken adım büyüklüğünün aldığı değerler .....	55
Şekil 4.1. 7-köşeden oluşan örnek bir 3B yüzey yaması .....	58
Şekil 4.2. 7-köşeden oluşan örnek bir 3B yüzey yaması (farklı bir açıdan görünüşü) .....	59
Şekil 4.3. <i>Surf1</i> yüzey yamasına ( $n_v=8$ ) sanal sınır eklenmesiyle oluşan yüzey .....	60
Şekil 4.4. <i>Surf1</i> yüzey yamasına ( $n_v=8$ ) sanal sınır eklenmesiyle oluşan yüzey (farklı bir açıdan görünüşü) .....	60
Şekil 4.5. <i>Surf1</i> yüzey yaması için oluşturulan dinamik sinir ağının yapısı ..	63

Şekil 4.6. <i>Surf1</i> yüzey yamasının 3 iterasyon sonundaki düzlemsel izdüşümü.....	64
Şekil 4.7. <i>Surf1</i> yüzey yamasının 10 iterasyon sonundaki düzlemsel izdüşümü.....	64
Şekil 4.8. Sanal sınır düğümlerinin çıkarılmasıyla elde edilen düzlemsel izdüşüm sonucu ( <i>Surf1</i> yüzey yaması).....	65
Şekil 4.9. 15-köşeden oluşan örnek bir 3B yüzey yaması.....	66
Şekil 4.10. <i>Surf2</i> yüzey yamasına ( $n_v=8$ ) sanal sınır eklenmesiyle oluşan yüzey.....	68
Şekil 4.11. <i>Surf2</i> yüzey yamasının 50 iterasyon sonundaki düzlemsel izdüşümü.....	70
Şekil 4.12. Sanal sınır düğümlerinin çıkarılmasıyla elde edilen düzlemsel izdüşüm sonucu ( <i>Surf2</i> yüzey yaması).....	71
Şekil 4.13. 40-köşeden oluşan örnek bir 3B yüzey yaması.....	72
Şekil 4.14. <i>Surf3</i> yüzey yamasına ( $n_v=16$ ) sanal sınır eklenmesiyle oluşan yüzey.....	73
Şekil 4.15. <i>Surf3</i> yüzey yamasının 100 iterasyon sonundaki düzlemsel izdüşümü.....	74
Şekil 4.16. Sanal sınır düğümlerinin çıkarılmasıyla elde edilen düzlemsel izdüşüm sonucu ( <i>Surf3</i> yüzey yaması).....	75
Şekil 4.17. <i>Surf1</i> 'in $n_v=4$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	76
Şekil 4.18. <i>Surf1</i> 'in $n_v=6$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	77
Şekil 4.19. <i>Surf1</i> 'in $n_v=12$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	78
Şekil 4.20. <i>Surf1</i> 'in $n_v=16$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	79
Şekil 4.21. <i>Surf1</i> 'in $n_v=32$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	80
Şekil 4.22. <i>Surf1</i> 'in $n_v=64$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	81
Şekil 4.23. <i>Surf2</i> 'in $n_v=4$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	83
Şekil 4.24. <i>Surf2</i> 'in $n_v=6$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	84
Şekil 4.25. <i>Surf2</i> 'in $n_v=12$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	85
Şekil 4.26. <i>Surf2</i> 'in $n_v=16$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	86
Şekil 4.27. <i>Surf2</i> 'in $n_v=32$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	87
Şekil 4.28. <i>Surf2</i> 'in $n_v=64$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	88
Şekil 4.29. <i>Surf3</i> 'ün $n_v=4$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	90
Şekil 4.30. <i>Surf3</i> 'ün $n_v=6$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	91

Şekil 4.31. <i>Surf3</i> 'ün $n_v=8$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	92
Şekil 4.32. <i>Surf3</i> 'ün $n_v=12$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	93
Şekil 4.33. <i>Surf3</i> 'ün $n_v=32$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	94
Şekil 4.34. <i>Surf3</i> 'ün $n_v=64$ için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b).....	95
Şekil 4.35. <i>Surf1</i> yüzey yaması için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları.....	97
Şekil 4.36. <i>Surf2</i> yüzey yaması için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları.....	98
Şekil 4.37. <i>Surf3</i> yüzey yaması için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları.....	99
Şekil 4.38. <i>Surf1</i> yüzeyine ilişkin temel yöntem ve önerilen yönteme ait alan doğruluk farkı kıyaslaması (a) ve kenar doğruluk farkı kıyaslaması (b).....	102
Şekil 4.39. <i>Surf2</i> yüzeyine ilişkin temel yöntem ve önerilen yönteme ait alan doğruluk farkı kıyaslaması (a) ve kenar doğruluk farkı kıyaslaması (b).....	103
Şekil 4.40. <i>Surf3</i> yüzeyine ilişkin temel yöntem ve önerilen yönteme ait alan doğruluk farkı kıyaslaması (a) ve kenar doğruluk farkı kıyaslaması (b).....	104
Şekil 4.41. <i>Surf1</i> 'in $n_v=64$ için temel yöntem (a) ve önerilen yöntem (b) ile düzleme izdüşümü.....	105
Şekil 4.42. <i>Surf2</i> 'in $n_v=64$ için temel yöntem (a) ve önerilen yöntem (b) ile düzleme izdüşümü.....	106
Şekil 4.43. <i>Surf3</i> 'ün $n_v=64$ için temel yöntem (a) ve önerilen yöntem (b) ile düzleme izdüşümü.....	107
Şekil 4.44. Başlangıç izdüşüm sürelerinin görselleştirilmesi.....	109
Şekil 4.45. <i>Surf1</i> yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (1).....	111
Şekil 4.46. <i>Surf1</i> yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (2).....	112
Şekil 4.47. <i>Surf1</i> yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (3).....	113
Şekil 4.48. <i>Surf1</i> yüzeyine başlangıç izdüşüm ve enerji yayma algoritması sonuçları.....	114
Şekil 4.49. <i>Surf2</i> yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (1).....	116
Şekil 4.50. <i>Surf2</i> yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (2).....	117
Şekil 4.51. <i>Surf2</i> yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (3).....	118
Şekil 4.52. <i>Surf2</i> yüzeyine başlangıç izdüşüm ve enerji yayma algoritması sonuçları.....	119

Şekil 4.53. <i>Surf1</i> yüzeyine ilişkin sabit ve deęişken adimli enerji yayma yöntemlerinin performansları.....	123
Şekil 4.54. <i>Surf2</i> yüzeyine ilişkin sabit ve deęişken adimli enerji yayma yöntemlerinin performansları.....	123
Şekil B.1. <i>Surf2</i> yüzey yaması için oluşturulan dinamik sınır aęının yapısı...	139
Şekil B.2. <i>Surf3</i> yüzey yaması için oluşturulan dinamik sınır aęının yapısı...	140



## ÇİZELGELER

	<b>Sayfa</b>
Çizelge 3.1. Sabit ve değişken adım büyüklüğüne dayalı enerji yayma sürecindeki iterasyon sayıları kıyaslaması .....	54
Çizelge 4.1. Üç farklı yüzey için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları.....	96
Çizelge 4.2. <i>Surf1</i> örnek yüzey yaması için temel yöntem ve önerilen yöntem izdüşüm sonuçları .....	100
Çizelge 4.3. <i>Surf2</i> örnek yüzey yaması için temel yöntem ve önerilen yöntem izdüşüm sonuçları .....	100
Çizelge 4.4. <i>Surf3</i> örnek yüzey yaması için temel yöntem ve önerilen yöntem izdüşüm sonuçları .....	101
Çizelge 4.5. Çeşitli büyükte yüzeylerin başlangıç izdüşüm süreleri .....	108
Çizelge 4.6. <i>Surf1</i> yüzey yamasına ilişkin değişken adım büyüklüğüne dayalı enerji yayma algoritması parametreleri .....	110
Çizelge 4.7. Şekil 4.45 -4.47'deki enerji yayma sonuçlarına ilişkin alan ve kenar doğruluk fark değerleri .....	114
Çizelge 4.8. <i>Surf2</i> yüzey yamasına ilişkin değişken adım büyüklüğüne dayalı enerji yayma algoritması parametreleri .....	115
Çizelge 4.9. Şekil 4.49 -4.51'deki enerji yayma sonuçlarına ilişkin alan ve kenar doğruluk fark değerleri .....	119
Çizelge 4.10. <i>Surf1</i> yüzeyine ilişkin sabit ve değişken adım büyüklüğüne dayalı enerji yayma sürecindeki doğruluk değerleri .....	121
Çizelge 4.11. <i>Surf2</i> yüzeyine ilişkin sabit ve değişken adım büyüklüğüne dayalı enerji yayma sürecindeki ve doğruluk değerleri.....	122

## SİMGELER VE KISALTMALAR

<b>A</b>	Komşu iç düğümlerin ağırlıklarını içeren kare matris
<b>b<sub>1</sub>, b<sub>2</sub></b>	Sınır düğümlerinin ağırlıklı toplamını içeren vektör
<b>C</b>	Yay sabiti
<b>D</b>	Sönümlleme matrisi
<b>d<sub>i</sub></b>	<i>i</i> . düğümün komşu sayısı
<b>d<sub>i,j</sub></b>	3B uzayda <i>i</i> . köşe ile <i>j</i> . köşe arasındaki Öklid mesafesinin tersi
<b>E</b>	Bir yüzey yamasında birikmiş toplam enerji
<b>E<sub>C</sub></b>	Kenar doğruluk farkı
<b>E<sub>S</sub></b>	Alan doğruluk farkı
<b><math>\vec{f}</math></b>	Bir kütleyle etkileyen bileşke kuvvet
<b>f<sub>q</sub></b>	Modelin serbestlik dereceleriyle ilgili genel harici kuvvet
<b><math>\vec{F}_{\text{penalty}}</math></b>	Ceza fonksiyonu
<b>F</b>	Yüzey üçgenlerinin indislerini tutan matris
<b>g<sub>q</sub></b>	Yer çekimi kuvveti
<b>K</b>	Sınır düğüm sayısı
<b>m<sub>i</sub></b>	Kütle-yay sisteminde <i>i</i> . düğümün kütlesi
<b>M</b>	Yay kütle matrisi
<b>n</b>	İç düğüm sayısı
<b>n<sub>v</sub></b>	Sanal sınır poligonundaki düğüm sayısı
<b>N</b>	Toplam düğüm sayısı
<b>p</b>	Yüzeyin alan yoğunluğu
<b>P</b>	2B düzlemsel yüzey
<b>P*</b>	İleveli yüzeye ilişkin 2B düzlemsel yüzey
<b>P<sub>i</sub>, P<sub>j</sub></b>	Kütle-yay sistemindeki kütleler
<b>r<sub>v</sub></b>	Dinamik sanal sınır düğümlerine ev sahipliği yapan çemberin yarıçapı
<b>S</b>	3B Üçgensel kafes yüzey
<b>S*</b>	İleveli 3B üçgensel kafes yüzey
<b>X</b>	Yüzey kafesinin köşelerine karşılık gelen noktalar bulutu
<b>x<sub>i</sub></b>	3B uzaydaki nokta
<b>x<sub>1</sub></b>	İlk iç köşe
<b>x<sub>n</sub></b>	Sonuncu iç köşe
<b>x<sub>n+1</sub></b>	İlk sınır köşesi
<b>x<sub>N</sub></b>	Sonuncu sınır köşesi
<b>u<sub>i</sub>, u<sub>j</sub></b>	2B parametrik uzaydaki düğüm
<b>u<sub>1</sub></b>	İlk iç düğüm
<b>u<sub>n</sub></b>	Sonuncu iç düğüm
<b>u<sub>n+1</sub></b>	İlk sınır düğümü
<b>u<sub>N</sub></b>	Sonuncu sınır düğümü
<b>λ<sub>i,j</sub></b>	<i>i</i> . düğüm ile <i>j</i> . düğümü arasındaki ağırlık
<b>U<sub>b</sub></b>	Düzlemdeki sınır düğümlerini içeren matris
<b>α<sub>v</sub></b>	Sanal sınır çemberindeki adım açısı
<b>Λ</b>	Tüm komşu düğümlerin ağırlıklarını içeren matris
<b>Ω</b>	Düzlemsel parametrik yüzey



2B	İki Boyutlu
3B	Üç Boyutlu
ANN	Artificial Neural Network
ARAP	As-Rigid-As-Possible
ASAP	As-Similar-As-Possible
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CST	Sabit Gerilme Üçgenleri
FEM	Sonlu Eleman Yöntemi
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
IIR	Infinite Impulse Response
LSCM	Least Square Conformal Mapping
NURBS	Non-Uniform Rational B-Spline
ODE	Ordinary Differential Equations
RNN	Recurrent Neural Network
TDL	Tapped-Delay Line
VLSI	Very-Large-Scale Integration
YSA	Yapay Sinir Ağı

# 1. GİRİŞ

Eğrisel yüzeylerin düzlemsel karşılığının bulunması üretim alanında (örneğin gemi, oyuncak, giysi, ayakkabı ve mobilya endüstrisi gibi) yaygın olarak bilinen bir problemdir. Geleneksel yöntemde, üretim esnasında 2B parçalar yamultulur ve nihai ürünü inşa etmek üzere bir araya getirilir. İdealde yamultma ve bir araya getirmede gerilme durumu da hesaba katılmalıdır. Çünkü bu gerilme nihai üründe kaliteyi düşüren ve malzeme yorgunluğu yaratan bir elastik enerji doğurur. Bu endüstrilerde kullanılan geleneksel tasarım süreci deneme-yanılma yöntemine dayanır. Tasarımcı kağıt üzerinde 2B parçaları kabaca çizer ve istenen ürüne karşı düşecek bir prototip yapar. Sonuç tatmin edici değilse tasarımcı, kalıpları (pattern) kendi deneyimlerine dayanarak değiştirip başka bir prototip yapar. Bu prototip yapma ve modifikasyon adımları tekrar tekrar uygulanır. Bu da etkin olmayan, pahalı, yorucu ve oldukça zaman alan bir süreçtir (Wang, 2013).

Ürünlerin şekillerine ilişkin 3B yüzey yamalarının tasarlanması ve üretim için onların 2B kalıplarının yüzey düzleştirme yöntemiyle belirlenmesi günümüzde istenilen üretim kalitesini malzemelerdeki gerilmelerden dolayı verememektedir. Yalnız izometrik izdüşüm özelliğine sahip yüzey kafesleri için yüzey düzleştirme tekniği gerilmeden bağımsız olur (Carmo, 1976). Mevcut ticari CAD/CAM (Computer Aided Design/Manufacturing) sistemleri herhangi biçimli yüzeylerin modellenmesini sağlayacak fonksiyonlara sahip değildir ve literatürdeki geliştirilebilir yüzeylere ilişkin mevcut yaklaşımlar serbest biçimli (free-form) yüzeyleri modellemede yetersiz kalmaktadır (Wang, 2013).

Eğrisel yüzeylerin düzlemlerden kurulması (geliştirilmesi) problemiyle ilgili çok sayıda yaklaşım önerilmiştir. Bu problem yüzey sıfır eğriliğine sahip olduğu zaman anlamsızdır, ama yüzeyin eğriselliği arttığı zaman problem karmaşık hale gelir. Aynı yüzey için çok farklı sayıda düzlemsel geliştirme yaklaşımı vardır. Günümüzde üretilen karmaşık yüzeyli çok sayıda ürün, nokta-bulut (cloud of points) kullanılarak üretilebilmektedir. Düzlemsel geliştirmenin yapılmasında hangi yöntemin kullanılacağına seçimi hassas bir konudur. Çünkü

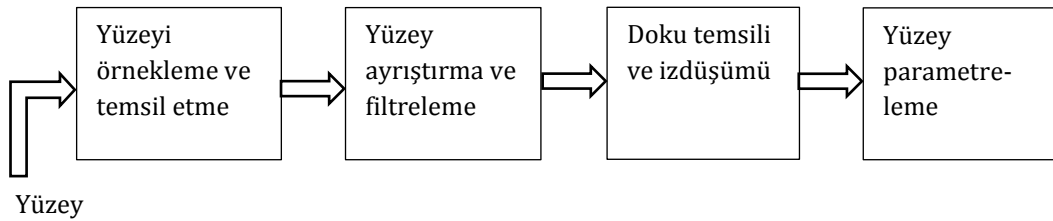
üretim yöntemi, malzeme özellikleri ve yüzey geometrisi gibi göz önüne alınması gereken çok sayıda sorun vardır.

Geleneksel Öklid düzlem yüzeyleri için yüzey doku karakteristiğine ilişkin formülasyonlar iyi kurulmuştur. Mesela Öklid yüzeyler, düzlemdeki her bir nokta için tek bir yükseklik değeriyle tanımlanır. Serbest biçimli yüzeyler ise doğası itibariyle daha karmaşık bir yapıya sahiptir (Abdul-Rahman vd., 2016). Bu çalışma alanı (domain) artık bir düzlem değildir ve sıfırdan farklı (non-zero) eğrilik (curvature) değerine sahip noktalar içerir. Dolayısıyla bu tip yüzeyler Öklid olmayan (non-Euclidean) yüzeyler olarak anılır. Gauss'un diferansiyel geometrideki teoremine göre, yani Egregium Teoremine göre, aynı eğriliğe sahip yüzeyler herhangi bir distorsiyona uğramadan birbirleri arasında izdüşüm yapılabilir. Buna karşın, farklı eğriliğe sahip yüzeyler arasında distorsiyonsuz izdüşüm yapılamaz. Mesela Dünya, düzlemsel bir yüzey üzerinde distorsiyonsuz görüntülenemez (Azariadis ve Sapidis, 2003). Dolayısıyla, serbest biçimli Öklid olmayan bir yüzey distorsiyonsuz veya bazı yüzey bilgilerinde kayıp olmadan bir düzleme izdüşürülemez. Sonuç olarak, serbest biçimli yüzeyler artık iki boyutlu bir ızgara üzerinde yükseklik değerleri olarak temsil edilemez. Başka bir örnek vermek gerekirse, bir yarım küre 2B ızgara (grid) üzerinde temsil etmeye çalışılırsa bu durum yüzey üzerindeki jeodezik mesafelerde distorsiyona sebep olur. Yani, 2B ızgara üzerine izdüşürüldüğünde gerçek yüzey üzerindeki farklı noktalar arasındaki jeodezik mesafeler distorsiyona uğrayacak ve küçülecektir (Abdul-Rahman vd., 2016). Verilen bir yüzey üzerindeki iki nokta arasındaki jeodezik mesafe, o yüzey üzerindeki iki nokta arasındaki en kısa mesafedir.

Serbest biçimli Öklid olmayan yüzeyler hızla gün yüzüne çıkmakta ve çok sayıda farklı uygulamalarda kullanılmaktadır. Örneğin serbest biçimli yüzeyler; optik, 3B yazıcılar, enerji verimli jet motorları, uçak türbin kanatları ve güvenilir uzun ömürlü implantlar gibi nesnelerin tasarımında kullanılmaktadır. Dolayısıyla, Öklid olmayan serbest biçimli yüzeyleri daha iyi karakterize edebilen güvenilir ve düzgün ölçme teknikleri geliştirmek önemli hale gelmektedir. Bu yüzden, yüzey dokusu tanımlama (surface texture characterisation) son on yılda önemli

değişime tanık olmuştur; kesit ölçümünden alan ölçümüne ve basit yüzeylerden karmaşık serbest biçimli yüzeylere kadar birçok tanımlama değişimi olmuştur (Jiang vd., 2007a, b; Jiang ve Whitehouse, 2012). Bu değişimler yeni evrim geçiren karmaşık serbest biçimli yüzeyleri daha iyi tanımlayabilen, temsil edebilen, ölçeklenen ve karakterize edebilen yeni algoritmalara ve tekniklere duyulan ihtiyacı tetiklemiştir (Jiang vd., 2011; Lou vd., 2012a, b; Wang vd., 2012; Abdul-Rahman, 2013; Jiang vd., 2013a, b; Liu vd., 2013; Liu vd., 2013; Zheleznyakova, 2015; Su vd., 2016).

Geleneksel olarak yüzey dokusunu tanımlama ve parametrelemenin dört ana adımda gerçekleştirildiği söylenebilir (Abdul-Rahman vd., 2016). Şekil 1.1’de görülen bu adımlar, yüzeyi örnekleme ve temsil etme (surface sampling and representation), yüzey ayrıştırma ve filtreleme (decomposition and filtration), doku temsili ve izdüşümü (texture representation and mapping) ve yüzey parametrelemedir (surface parameterization). Basit geometrilere kayma, Şekil 1.1’de verilen her bir adımdaki sadece Öklid yüzeyler için geçerli olan teknikleri ve algoritmaları yeniden düşünmeyi ve yeniden tasarlamayı gerektirir.



Şekil 1.1. Yüzey tanımlama modeli (Abdul-Rahman vd., 2016)

Serbest biçimli yüzey tanımlamada halledilmesi gereken önemli noktalardan birisi, böyle yüzeylerin dokusunun nasıl temsil edileceğidir. Başka bir deyişle yüzey biçimini filtreledikten sonra ortaya çıkan yüzey kalıntılarının (residues) nasıl temsil edileceğidir. Basit geometrik şekillerde varsayım şudur: Biçim Öklid’dir, yani düzlemsel yüzeydir. Dolayısıyla yüzey dokusunu düzleme izdüşürmek ve daha sonra kalıntıları analiz ederek yüzey parametrelerini elde etmek oldukça basittir. Bu izdüşüm herhangi bir distorsiyon getirmez. Çünkü basit geometrik şekillerde formun kendisi Öklid’dir. Buna karşın, aynı iş Öklid

olmayan geometriyi yüzeyler için yapılmaya çalışılırsa farklı türde distorsiyonlar ortaya çıkar. Bu distorsiyonlar, orijinal yüzeyin uzunluk (distance), açı (angle) veya alanında (area) kendini gösterir. Bundan dolayı yüzeyi analiz etmeden ve yüzeyin parametrelerini elde etmeden önce, doku temsil etme yöntemleri, bu distorsiyonları minimize etme bakımından çok önemlidir.

Serbest biçimli yüzeyler günümüze ait grafik uygulamalarında önemli bir rol oynamaktadır (Eigensatz vd., 2010). Yüzey gerçekleştirme (rendering), yüzey kaplama (surface tessellation), yüzey örnekleme (surface sampling) vb. serbest biçimli yüzeylerle ilgili uygulamaların sonuçları, yüzey parametrelere oldukça bağımlıdır. Bu uygulamalarda parametrelere biçim koruyan (conformal), yani parametre uzayındaki birim çemberi yüzey üzerindeki birim çembere izdüşüren türden olması arzulanan bir durumdur (Yang vd., 2016). Biçim koruyarak izdüşüm alanında oldukça kapsamlı çeşitli çalışmalar yapılmıştır. Başarılı çalışmaların büyük bir kısmı üçgensel kafeslerin (mesh) biçim koruyarak yapılan izdüşümlerinde karşımıza çıkmaktadır. Diğer taraftan karmaşık serbest biçimli yüzeylerin biçim koruyarak izdüşürülmesinde benzer başarılar henüz elde edilmemiştir (Yang vd., 2016). Biçim koruyarak yapılan parametrelere eksiklik, serbest biçimli yüzeyler kullanarak yüzey gerçekleştirme ve kaplama uygulamalarında yüksek kalitede sonuçlar elde etmede bir darboğaz oluşturur (Yang vd., 2016).

Yüzey düzleştirme değişik teknikler gerektirir. Bu teknikler; yüzeylerin geliştirilebilirliği için diferansiyel geometri, eğrisel yüzeyin düzlem üzerine izometrik izdüşümünü yaklaşımlandırmak için kullanılan afin transformasyonlar takımının özelliklerinin araştırılması, 3B-2B izdüşümündeki distorsiyonun minimizasyonu ve düzlemselleştirilecek 3B şekilleri işleyen biçim (shape) optimizasyon yöntemleridir.

Bu çalışmada yukarıda belirtilen tekniklerden yararlanılarak tamamen keyfi 3B cisimlerin uzunluk bilgisi mümkün olduğunca korunarak 2B düzlemsel karşılıklarının bulunmasını sağlayacak etkin bir teknik geliştirilmiştir. Bu teknik gerilme enerjisinin minimum olmasını sağlayarak şekillendirilecek yüzeyin

deformasyonunu olabildiğince minimum seviyeye indirmektedir. 3B yüzeyin parametrelenmesi, 2B düzlemsel yüzeye bire bir karşılık kuracak bir fonksiyon üretme çalışması olarak tanımlanabilir. Bu kavram, geometri işlemede önemli bir rol oynamaktadır. Çünkü bu, 3B modelleme problemlerini 2B uzaya dönüştürme yolunu açmayı sağlar ve burada işlem yapmak nispeten daha kolaydır. Karmaşık 3B yüzeyler, gelişmiş aralıklı tarama ve 3B baskı teknolojileri sayesinde yaygın olarak her yerde kullanıldığından, geniş kafesler için düşük parametrelene süreleri önem kazanmaktadır. Bu bağlamda, 3B üçgensel kafes yüzeylerin hızlandırılmış bir şekilde başlangıç izdüşümünün üretilmesi için bir dinamik sinir ağı modeli oluşturulmuştur. Bu çalışmada, geribeslemeli sinir ağı modeliyle konveks kombinasyon yaklaşımı birleştirilerek üçgensel kafes yüzeylerin başlangıç izdüşümü gerçekleştirilmiştir. Bu yeni yaklaşım, sinirsel ağların paralel-dağıtık işleme tabiatı ve barisentrik haritalama tekniğinin hızlı ve bire-bir izdüşüm özellikleri sayesinde nispeten büyük ağların projeksiyon için gereken hesaplama süresini önemli ölçüde azaltmaktadır.

Bu çalışmanın geri kalanı şu şekilde düzenlenmiştir: İkinci bölümde literatür özeti verilerek yüzey projeksiyon tekniklerine ve üçgenleme yöntemine değinilmiştir. Üçüncü bölümde üçgensel yüzeyin parametrelenmesine değinildikten sonra barisentrik izdüşüm teorisi ve enerji tabanlı düzleştirme tekniği verilerek önerilen yöntem açıklanmıştır. Önerilen yüzey düzleştirme yaklaşımına ilişkin deneysel sonuçlar ve araştırma bulgularına dördüncü bölümde yer verilmiştir. Son bölümde çalışmanın sonuçları ve ilerki çalışmalar yorumlanmıştır.

## 2. LİTERATÜR ÖZETİ

Yüzeyler, yassılaşılabilmek (malleability) bakımından geliştirilebilir (developable) yüzeyler ve geliştirilemeyen yüzeyler olarak ikiye ayrılabilir (Xu vd., 2008). Geliştirilebilir yüzeyler, her noktasında sıfır Gauss eğriliğine sahiptir, mesela koni, silindir, teğet düzlemi vs. Bu tür yüzeylerin düzleştirilmesi sonucunda ortaya çıkan yüzey ile önceki yüzey arasında izometrik bir karşılık vardır. Yine bu tür yüzeylerin düzleştirilmesi sonucunda ortaya çıkan düzleştirme sonucu tamamen doğrudur ve düzlem ile tamamen çakışır (Li vd., 2014). Geliştirilemeyen yüzeyler ise karmaşık yüzeyler olarak bilinir ve genellikle Bezier, NURBS (Non-Uniform Rational B-Spline) vb. yüzeylerdir.

Karmaşık bir yüzeyin düzleştirilmesi işlemi, 3B uzaydaki bir yüzeyin belirli kurallara (Zhan, 2007) göre düzlem yüzeyine izdüşürülmesi olayıdır. Günümüze kadar bu alanda pek çok çalışma yapılmıştır. Şimdiye kadar yapılan çalışmalara bakıldığında, karmaşık yüzeylerin düzleştirilmesi üç temel düzleştirme yaklaşımı altında toplanabilir: *Geometrik düzleştirme*, *mekanik düzleştirme* ve bu ikisinin birleşiminden oluşan *hibrit düzleştirme* (Xu vd., 2002).

Geometrik düzleştirme sınıfına giren bir çalışmada (Hinds vd., 1991), serbest biçimli yüzeyler bir dizi oval veya hiperbolik alt bölgelere bölünmek suretiyle Gauss eğriliği yaklaşımına dayanarak geliştirilebilir yüzeyler haline dönüştürülmüştür. Bu küçük alt bölgeler, kabul edilebilir hata ile düzleştirilerek tüm karmaşık yüzeyin yaklaşık düzleştirilmiş hali elde edilmiştir (Shin, 2011). Bu konuda 1991 yılında yayımlanan bir çalışma, ilerleyen zamanda yapılan çalışmalarda (Azariadis ve Aspragathos, 1997; Azariadis vd, 2002) izdüşüm yapılan yüzeylerde birbirine karşı düşen nokta çiftleri arasındaki Öklid mesafelerinin minimizasyonu yapılarak hata azaltma şeklinde iyileştirilmiştir. Bu dönemde Azariadis ve Aspragathos tarafından karmaşık yüzeyleri düzleştirmek için genel bir düzleştirme algoritması geliştirilmiştir (2000). Bu geliştirilen yöntemde yerel izdüşüm doğruluğunu kontrol etmek mümkündür. Adı geçen çalışmada önerilen yeni bir izdüşüm tekniğinin yanı sıra, izdüşüm kalitesini değerlendirmeye yönelik bir dizi metrik de önerilmiştir. Bodduluri ve

Ravani (1993) tarafından yapılan çalışmada serbest biçimli yüzeyler önce basit alt yüzeylere bölünmüştür ve daha sonra geliştirilebilir yüzeylere yaklaşılmıştır. Adı geçen çalışmada, rasyonel Bezier ve B-spline eğrilerine dayalı olarak geliştirilebilir yüzeylerin geometrik tasarımı için bir teknik geliştirilmiştir. Başka bir çalışmada (Cho vd., 1998) ızgara (grid) yaklaşımli düzleştirme algoritması önerilmiştir. Anılan çalışmada, izdüşüm hata fonksiyonu minimize edilerek, verilen bir parametrik yüzey yaması ile onun üçgensel alanı arasında yaklaşık bir yerel eş ölçülü izdüşüm ilişkisi kurulmuştur.

Konu ile ilgili bir diferansiyel geometri kitabında (Su vd., 1998), karmaşık yüzeylerin yaklaşık düzleştirilmesi, serbest (unconstrained) minimum problemi olarak kabul edilir ve temel ilkesi şöyledir: Düzleştirme öncesi ve sonrası arasında, eğrisel yüzeydeki her bir kenarın farkının karelerinin toplamı minimumdur. Ping (1997) tarafından karmaşık yüzeyler için bir düzleştirme algoritması sunulmuştur. Çalışmada metal levha bir dizi şerit alana bölünmüş ve sonra çizgili yüzeye benzetilmiştir. Daha sonra her bir çizgili yüzey, üçgensel kafese bölünmüştür ve her bir üçgensel eleman bir düzleme izdüşürülmüştür. Elde edilen sonuçlardan düzleştirmiş yüzeyin *çakışma* (overlaps) ve *aralıklar* (gaps) içerdiği görülmüştür. Aynı yazar başta olmak üzere bazı araştırmacılar ile birlikte ertesini yıl, çizgi alanlar arasındaki bu çakışma ve aralıkları giderecek bir çalışma (Ping vd., 1998) yapılmıştır. 2003 yılında yayımlanan bir çalışmada (Chen ve Wang) karmaşık yüzeyler için “quasi-rulings” kavramı ileri sürülerek adaptif segment tekniği yardımıyla yüzey düzleştirilmiştir (Li, 2014).

Geometrik düzleştirme çalışmalarına paralel olarak mekanik düzleştirme alanında da bazı çalışmalar yapılmıştır. MacCartney vd. (2000) düzleştirme işlemi esnasında enerji değişimini minimum yaparak karmaşık yüzeylerin düzleştirmesinde işe yarayan bir algoritma geliştirmişlerdir. Anılan çalışmadan, sıklıkla ham malzeme olarak dokuma kumaşların kullanılabildiği ve bu kumaşların doğası gereği izotropik olmayan bir yapıya sahip olması sebebiyle düzleştirme probleminin daha karmaşık hale gelebildiği anlaşılmaktadır. Bu tip kumaşlar, örgü yapılarına göre, mekanik özelliklerinde farklılıklar göstermektedir. Bir çalışmada (Lee ve Huh, 1998), sonlu elemanlar yaklaşımı



kullanılarak metal levha biçimlendirmedeki nihai şekillerden boş (blank) levhaların ve gerilme dağılımlarının tahmin edilmesi gerçekleştirilmiştir. Bu yaklaşım, sonlu elemanların geometrik uyumluluğu, plastik deformasyon teorisi ve kısıtlarla plastik işin minimizasyonu ve uygun bir başlangıç tahmini ile uğraşmıştır. Başka bir çalışmada (Lan vd., 2001) yine deformasyon teorisine dayanan, *sonlu elemanlar* yöntemiyle (finite element method-FEM) bir yüzey düzleştirme algoritması sunulmuştur. İdeal biçimlendirme teorisine dayanan *sonlu-elemanlar* ters yöntemi yardımıyla boş metal levha tasarımına yönelik bir dizi çalışma (Chung ve Richmond, 1992a, b; Chung vd., 2000) yapılmıştır. Bu yöntemin ters yöntem olarak anılmasının sebebi, metal levha parçasının nihai şeklinin önceden bilindiği durumda bu yöntem minimum gerilmeye sahip 2B boş (blank) başlangıç geometrisini üretmesidir (Chung vd., 2008).

Bir başka çalışmada (Zhang vd., 2003), karmaşık yüzeyler için evrensel bir düzleştirme algoritması geliştirilmiştir. Kafes yüzeyinin kenar ve düğümleri göz önüne alınarak yüzey düzleştirme geometrik problemi, mekanik yapı problemi haline dönüştürülmüştür. 2006 yılında yapılan çalışmada (Yoon vd.) ise hesaplama yükü çok yoğun olmayan bir düz tasarım modeli önerilmiştir. Anılan çalışma ideal biçimlendirme teorisine dayandırılarak hidrolik-biçimlendirmede (hydro-forming) düzlemsel olmayan ara yüzey tasarımında kullanılmıştır.

Bir başka çalışmada (Wang ve Wang, 2001) da yine karmaşık yüzey düzleştirmede enerji modeline dayalı bir algoritma önerilmiştir. Anılan çalışmada, yaylarla birbirine bağlı düğümlerden oluşan bir fiziksel model kurulmuştur. 3B yüzeyi düzleştirme esnasında denge durumu için enerji fonksiyonunun çözümü yapılarak 2B yüzeyler oluşturulmuştur. Geliştirilen yüzeyin doğruluğu yerel olarak kontrol edilebilmiştir. Ertesi yıl yaptıkları çalışmada Cai vd. (2002) iki koşula dayandırılan pozitif tanımlı hedef fonksiyon oluşturmuşlardır. Bu koşullar, malzemedeki deformasyonların eşit dağıtılması ve hacmin sabit kalmasıdır. Bu fonksiyonun tekrarlı bir yapıda minimizasyonu yapılarak düzlemsel yüzeyin konturu elde edilmiştir. Wang vd. (2002) tarafından kütle-yay modeline dayanan bir yaklaşım önerilmiştir. Anılan çalışmada düğümler kütle, kenarlar ise yay ile modellenmiştir.

Hibrit model olarak adlandırılabilir yaklaşımda ise ilk olarak karmaşık yüzey, geometrik düzleştirme yöntemi kullanılarak düzleştirilir. Fakat ortaya çıkan yüzeyde genelde örtüşen veya ayrık kalan parçalar görülür (Li vd., 2014). Daha sonra ise mekanik revizyon ile daha iyi ve doğru bir sonuç elde edilebilir. Bu yaklaşım kullanılarak yapılan bir çalışmada (Chung vd., 2008) ayakkabının üst kısmına ilişkin 3B yüzeyin düzlemsel karşılıklarının hesaplanması için yeni bir yöntem önerilmiştir. Anılan çalışmada 3B yüzey ilk önce, üçgensel elemanlar ile temsil edilmiş ve sonra kabaca 2B düzleme izdüşürülmüştür. Daha sonra bir iyileştirme tekniği ile geometrik hataları minimize edecek şekilde üçgensel elemanların düğüm noktalarının konumları yeniden hesaplanmıştır. Bu yöntemde, her ne kadar geometrik hataları minimize ederek nokta konumlarında iyileştirme yapılsa da ortaya çıkan 2B şekil, üzerinde gerilme enerjisi taşımaktadır ve bu enerji başka bir proses ile hafifletilmelidir (Chung vd., 2008). Anılan çalışmada bu gerilme enerjisi sonlu elemanlar ters yöntemi kullanılarak halledilmiştir. Shin (2011) tarafından yapılan çalışmada hibrit model benimsenerek izotropik olmayan malzemelerden oluşan serbest biçimli yüzeylerin düzlemselleştirilmesi (planar development) ile ilgili bir çalışma yayımlanmıştır. Anılan çalışmada önce, serbest biçimli yüzey üçgensel elemanlar kümesiyle temsil edilmiştir. Temsili üçgen kümesinde Euler operatörleri kullanılarak geometrik iyileştirme yapıldıktan sonra kabaca 2B düzleme izdüşüm yapılmıştır. Daha sonra, sabit gerilme üçgenleri (constant strain triangles-CST) kullanılarak gerilme enerjisi yöntemi ile en uygun (optimum) 2B konturlar elde edilmiştir. Anılan çalışmada ayrıca, elde edilen çeşitli 2B konturlarını kıyaslamak için hata kestirimine yönelik bir kontur hizalama tekniği önerilmiştir. Önerilen yöntem belirli bir örnek nesne üzerinde uygulanarak modelin geçerliliği kanıtlanmıştır.

Son dönemde yapılan bir çalışmada Liu vd. (2013) tarafından kafes yüzeylerin boş metal parçalarını düz desenlerine dönüştürmek için enerjiye dayalı düzleştirme algoritması geliştirilmiştir (Li vd., 2014). Fiziksel model olarak basitleştirilmiş enerji modeli önerilmiştir. Değişken adım büyüklüğüne dayanan yenilikçi bir enerji boşaltım süreci ortaya atılarak verimlilik artışı sağlanmıştır. Karmaşık yüzeyler için hiyerarşik düzleştirme metodu önerilmiştir ve bununla

düzleştirilen distorsiyonlar serbest bırakılmıştır. Geliştirilen yöntemin etkinliğini göstermek amacıyla enerji metodu ile klasik ARAP (as-rigid-as-possible) kafes (mesh) parametreleme yöntemi kıyaslanmıştır. Deneysel sonuçlara göre, boş metal parçaların düz kalıplarının üretiminde geliştirilen enerjiye dayalı düzeltirme metodu pratik ve etkin bir çözüm sunmaktadır (Li vd., 2014).

Son yıllarda Abdul-Rahman vd. (2016) tarafından yayınlanan bir çalışmada, serbest biçimli yüzey parametreleme için geliştirilen yeni bir model tanıtılmıştır. Bu yeni model, parametrelerin hesaplanmasından önce bir projeksiyon algoritmasının kullanımını önerir. Yapılan çalışmada, bilgisayar grafikleri alanındaki mevcut ağ (mesh) projeksiyon tekniklerinden farklı projeksiyon algoritmaları benimsenmiştir. Serbest biçimli yüzeyler 3B üçgensel kafeslerle temsil edilir ve dolayısıyla daha çeşitli karmaşık yüzeyleri temsil edebilir. Önerilen çalışmanın en önemli noktası, yüzeyin parametrelerini hesaplamadan önce yüzeyi bir projeksiyon algoritması ile 3B dünyadan 2B dünyaya minimum distorsiyon ile izdüşürmesidir. Bu çalışma, parametreleri hesaplamadan önce serbest biçimli Öklid olmayan yüzeylerin dokusunun nasıl temsil edileceğine ışık tutmuştur. Yüzey izdüşürüldükten sonra 2B uzaya bağlı olarak farklı yüzey parametreleri hesaplanabilir. Bu çalışmada dört tane genel istatistiksel parametre genişletilerek üçgensel kafeslerle temsil edilen yüzeylere uygun hale getirilmiştir. Bu dört parametre de, yüzey dokusunu en iyi tanımladığı düşünülen yükseklik (height) parametresidir. Anılan çalışmada yayınlanan sonuçlar; klasik projeksiyon metodunun, diğer projeksiyon tekniklerinden daha az kesin parametreler ürettiğini göstermiştir. Herhangi bir projeksiyon tekniğini işin içine sokmadan parametreleri doğrudan yüzey üzerinde hesaplamak en iyi yoldur iddiasında bulunulabilir. Bu gözlem doğrudur. Fakat yüzey parametrelerini doğrudan yüzey üzerinde hesaplamak çok zordur ve hala yeterince tanımlanmamıştır. Dolayısıyla anılan çalışmada önerilen model tüm parametreleri hesaplamada kullanılan bir ara aşamadır. Hangi projeksiyon türünün yüzeyi karakterize etmek için en iyi seçim olduğu hala açık değildir. Anılan çalışmada çeşitli teknikler uygulanmıştır, fakat yüzey

karakterize etmek veya belirli parametreler için en iyi projeksiyonu elde etmek için ilave çalışmalara ihtiyaç duyulmaktadır (Abdul-Rahman vd., 2016).

Aynı yıl Yand vd. (2016) tarafından yayınlanan çalışmada hiyerarşik serbest biçimli yüzeyler olarak adlandırılan serbest biçimli yüzey temsiline yönelik yeni bir çalışma önerilmiştir. Verilen NURBS yüzeylerin açı koruma özelliğini (konformalitesini) iyileştirmek için serbest biçim dönüşümlere dayanan bir optimizasyon algoritması önerilmiştir. Açı koruyan (conformal) sapmaları ölçen lineer olmayan enerji ve onun nümerik yaklaşımı formüle edilmiştir. Açı koruyan enerjiyi minimize etmek için başlangıçtaki serbest biçim dönüşüm, 3B ayrıklaştırılmış kafesin açı koruyan izdüşümüyle yaklaşımda bulunarak elde edilir. Bu da Ricci akış metoduyla hesaplanır. Başlangıçtaki serbest biçim dönüşümün açı koruma özelliğini iyileştirmek için lineer olmayan uyumluluk (conformality) optimizasyonu ve dönüşüm iyileştirme birbirine eklenerek açı koruma sapmasının kullanıcı tarafından belirlenen tolerans seviyesine kadar azaltılması sağlanır.

Serbest biçimli yüzeylerin uyumluluğu, yüzey gerçekleştirme ve yüzey kaplama uygulamaları için önemlidir. Aynı zamanda yüzeylerin eşit alanlı (equiareality) olması, yani 3B yüzeyin alanı ile 2B yüzeyin alanının eşit olması örnekleme/yüzey kaplama uygulamaları için önemlidir. Serbest biçimli yüzeylerin konformalitesi ve eşit alanlı olması arasında bir takas vardır. Anılan çalışmada yüzeylerin açı koruma özelliğine odaklanılmıştır. Serbest biçimli yüzeylerin eşit alanlı olmasının iyileştirilmesi gelecekte bu alanda yapılabilecek bir çalışma olarak durmaktadır (Yang vd., 2016).

## **2.1. Yüzey Projeksiyon Teknikleri**

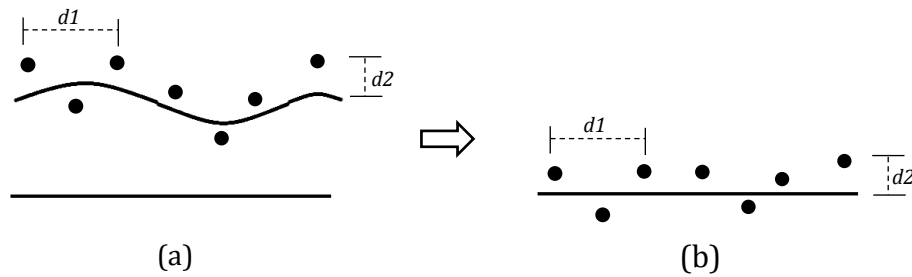
Serbest biçimli yüzeylerdeki dokuların (texture) temsili için kafes projeksiyon (izdüşüm) tekniklerinin bilinmesi gerekmektedir. Kafes projeksiyonu 3B uzaydaki bir kafesin düzlemsel 2B uzaydaki bir kafese izdüşürülmesi sürecidir (Abdul-Rahman vd., 2016). Projeksiyon algoritmaları bilgisayar grafikleri alanında popüler bir araştırma alanıdır ve şimdiye kadar bu alanda pek çok

algoritma (Liu vd., 2013, Hormann vd., 2007, Sheffer vd., 2007; Yoshizawa vd., 2004; Botsch vd., 2010) önerilmiştir.

Projeksiyon işlemi her zaman yüzey açıları, uzunlukları veya alanları gibi bir veya daha fazla yüzey parametresinde distorsiyona neden olur. İyi bir projeksiyon algoritması bu parametrelerden biri veya daha fazlasındaki distorsiyonları minimize edebilen algoritmadır. Herhangi bir projeksiyon algoritmasının temel amacı bu distorsiyon türlerinden en az birini minimize etmektir.

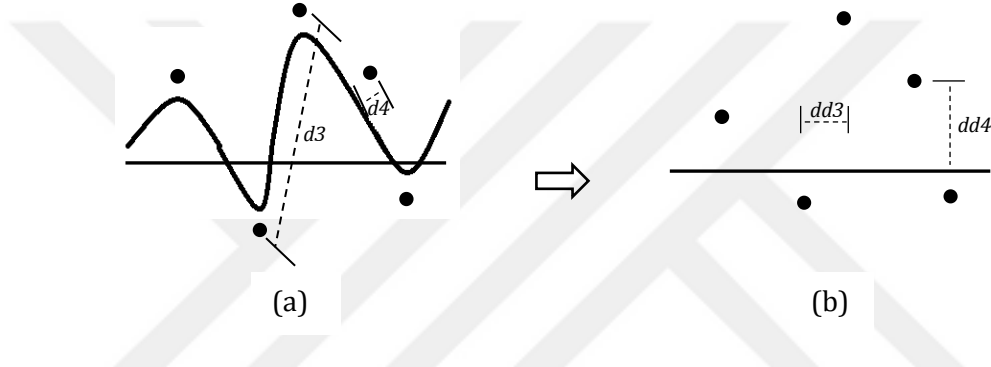
### 2.1.1. Klasik doku temsilindeki problem

Geleneksel olarak doku (texture), gerçek yüzeyden yüzey formu çıkarılarak temsil edilir (Abdul-Rahman vd., 2016). Öklid yüzeylerde yüzey formu bir düzlemdir veya düzleme çok yakındır. Dolayısıyla yüzey ayrıntıları, bir düzlem üzerinde herhangi bir distorsiyona uğramadan düzgün bir şekilde görüntülenebilir. Şekil 2.1'de basit bir Öklid yüzey verilmiştir ve bu yüzey özelliklerinin geleneksel temsil yöntemiyle nasıl distorsiyona uğramadan korunduğu gösterilmiştir. Şekilde görülen kavisli çizgi yüzey formunu, noktalar da yüzey dokusunu (özelliklerini) temsil etmektedir. Düz çizgi ise projeksiyon düzlemini temsil eder. Şekilden de anlaşılacağı gibi yüzey üzerindeki özellikler arasındaki mesafeler, projeksiyon düzlemi üzerindeki özellikler arasındaki mesafelerle aynıdır. Yani yüzey formu ve özellikler arasındaki mesafeler projeksiyon düzleminde korunur (Abdul-Rahman vd., 2016).



Şekil 2.1. Öklid yüzey üzerinde doku gösterimi: (a) Orijinal yüzey üzerinde bir doku, (b) Düzlem yüzey üzerinde aynı doku (Abdul-Rahman vd., 2016)

Diğer taraftan, Öklid olmayan yüzeyler söz konusu olduğunda, distorsiyon oluşacağından klasik projeksiyon yöntemi sıkıntılar doğuracaktır. Şekil 2.2’de, Öklid olmayan yüzeydeki ayrıntıların geleneksel yöntem kullanılarak düzlemsel uzaya projeksiyonu esnasında ortaya çıkabilecek distorsiyonu gösteren bir örnek görülmektedir. Şekilden kolayca anlaşılacağı gibi bu tip bir projeksiyonda yüzey özellikleri artık korunmaz. Mesela orijinal yüzeyde iki yüzey ayrıntısı arasındaki  $d3$  mesafesi, düzleme yapılan projeksiyon işlemi sonucunda distorsiyona uğrayarak çok daha küçük  $dd3$  değerini alabilir (Abdul-Rahman vd., 2016).



Şekil 2.2. Öklid olmayan yüzeyde doku gösterimi: (a) Orijinal yüzey üzerindeki doku, (b) Düzlem yüzey üzerindeki doku (Abdul-Rahman vd., 2016)

Klasik projeksiyon yöntemi, diğer projeksiyon yöntemlerine göre biraz farklı sonuçlar üretir. Çünkü dokudaki yükseklikler doğrudan düzlem üzerine izdüşürüldüğünden dolayı distorsiyona uğrayacaktır. Bunun sebebi şudur: Klasik projeksiyon dokudaki yükseklikleri, yüzeyin biçiminin normal doğrultusu yerine z-ekseninde dikkate alır. Bu örnekten de anlaşılacağı üzere serbest biçimli yüzeyler izdüşüm işleminden kaynaklanan yüzey metrik distorsiyonlarını en aza indirgeyecek projeksiyon algoritmalarına ihtiyaç duymaktadır (Abdul-Rahman vd., 2016).

### 2.1.2. Yüzey metrik distorsiyonu

Yüzeylerin çok farklı projeksiyon teknikleriyle izdüşürülmesi mümkündür. Verilen bir yüzeyin projeksiyonunun tek bir çözümü yoktur. Yani farklı projeksiyon teknikleri farklı özellikte izdüşüm sonuçları ortaya çıkarabilir.

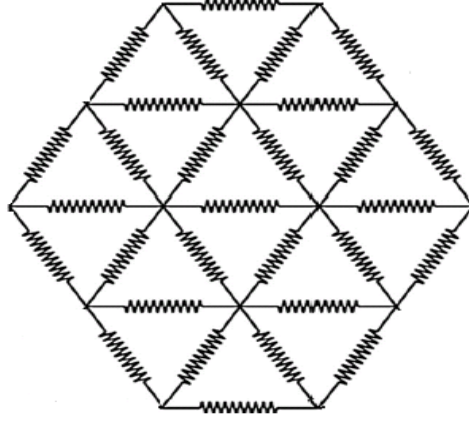
Mesela bir küre, *ortografik* veya *sterografik* projeksiyon gibi farklı yöntemlerle parametrelenebilir. Fakat, yüzeye ilişkin projeksiyon tekniğinden bağımsız olan bazı esas özellikler mevcuttur. Bu özellikler; yüzey normalleri, Gauss eğriliği, ortalama eğrilik gibi özelliklerdir (Hormann vd., 2007, Botsch vd., 2010). Diğer taraftan; yüzey üzerindeki açı, mesafe ve alan değerleri gibi temsil yöntemine bağlı olan bazı özellikler de mevcuttur. Mesela sadece sterografik projeksiyon kürenin açılarını muhafaza edebilirken ortografik projeksiyon açılarının distorsiyona uğramasına neden olur (Abdul-Rahman vd., 2016; Hormann vd., 2007).

Yüzey metrik distorsiyonunun mantığı temelde şudur: Genel anlamda 2B parametre uzayındaki birim çember gerçek 3B yüzey uzayında bir elipse izdüşürülür. Dairelerin elipse dönüştürülmesi, projeksiyon metodunun yerel metrik distorsiyonu olarak anılır. Bu distorsiyon elipsin iki temel eksenindeki uzunlukları olan  $\sigma_1$  ve  $\sigma_2$ 'ye kodlanır. Bu değerlere dayanarak projeksiyon algoritmalarının sahip olabileceği özellikler şu şekilde özetlenebilir (Abdul-Rahman vd., 2016:

- İzometrik (isometric) projeksiyon (*distance-preserving*), ( $\sigma_1 = \sigma_2 = 1$ )
- Açı koruyan (conformal) projeksiyon (*angle-preserving*), ( $\sigma_1 = \sigma_2$ )
- Alan koruyan (equiarreal) projeksiyon (*area-preserving*), ( $\sigma_1 \cdot \sigma_2 = 1$ )

### 2.1.3. Sabit-kenarlı izdüşüm algoritmaları

Sabit-kenarlı (fixed-boundary) izdüşüm barisentrik izdüşüm (barycentric mapping) olarak da bilinir ve en basit ve hızlı yöntemlerden birisidir (Abdul-Rahman vd., 2016; Hormann vd., 2007; Botsch vd., 2010). Bu teknikte 3B kafes yüzeyi yaylardan oluşan bir ağ ile modellenir. Bu teknikte cismin yüzeyini tanımlayan köşeler yaylarla birbirine bağlanır (Şekil 2.3).



Şekil 2.3. Sabit-kenarlı izdüşümde kullanılan yay modeli

Bir yüzeyin düzleme izdüşümünü gerçekleştirmek için yayların gerdirilmesinden faydalanılır. Yaylardaki enerji minimize edecek şekilde yüzeyi tanımlayan noktalar hareket ettirilir ve en verimli konfigürasyona ulaşıldığı durumda algoritma sonlandırılır. Yaylardaki toplam enerji şu formülle hesaplanır:

$$E = \frac{1}{2} \sum_{i=1}^n \sum_{j \in N_i} D_{ij} \|\mathbf{u}_i - \mathbf{u}_j\|^2. \quad (2.1)$$

Burada  $D_{ij}$  yay sabitini (ağırlıkları) temsil eder ve  $3B$  kafesden türetilir.  $i$ . köşenin parametrelenmiş konumu  $\mathbf{u}_i$  ile ve orijinal üçgensel kafesdeki  $i$ . köşenin birinci halka komşuları ise  $N_i$  ile gösterilmiştir. Yay modelindeki yay sabitlerinin, yani  $D_{ij}$  ağırlıklarının nasıl seçileceği bir soru işareti oluşturur. Basitçe bu ağırlıklar  $D_{ij}=1$  alınabilir veya üçgensel kafeste bunlara karşı düşen kenar uzunluklarıyla orantılı bir başlangıç ataması yapılabilir. Fakat bu basit tercihlerin her ikisi de daha önce bahsedilen metrikleri minimize etmez (Abdul-Rahman vd., 2016). Bu ağırlıkların belirlenmesinde açısallık ve uzunluk distorsiyonlarını minimize edecek türdeki yöntemlerden bazıları şunlardır:

- Ayrık harmonik izdüşüm (*Discrete harmonic mapping*): Bu teknikte ağırlıklar Laplace işlemi (Laplace-Beltrami) yaklaşımıyla türetilir. Açısallık distorsiyonunu minimize eder ve genellikle sınırların dışındaki yerlerde küçük açısallık distorsiyonlu iyi bir projeksiyon yapar.
- Ortalama-değerli izdüşüm (*Mean-value mapping*): Bu teknik de açısallık distorsiyonunu minimize eden bir yöntemdir.



- Düşük-uzamalı izdüşüm (*Low-stretch mapping*): Bu teknikte ağırlıklar iki aşamada hesaplanır. Birinci aşamada ağırlıklar, ayrık harmonik veya ortalama-değerli izdüşüm tekniklerin birisiyle hesaplanır. İkinci aşamada bu ağırlıklar birinci temel  $\sigma$  matrisinin (first fundamental matrix) özdeğerleri kullanılarak hesaplanır.

Sabit-kenarlı izdüşüm algoritmaları bazı metrik distorsiyonları minimize edecek hızlı ve güvenilir bir projeksiyon sağlar. Fakat bu projeksiyon tekniği kenarın (sınırın) konveks gövde üzerinde sabit olmasını zorunlu kılar. Bu da yöntemin kenarlarda büyük distorsiyonlar üretmesine neden olur (Hormann vd., 2007; Abdul-Rahman vd., 2016).

#### 2.1.4. Serbest-kenarlı izdüşüm algotirmaları

Sabit-kenarlı algoritmalar sınır noktalarının, dörtgen daire veya üçgen gibi, konveks biçim üzerinde sabitlenmiş olmasını gerektirir. Bu da kenar civarlarında muazzam distorsiyonlara sebep olur. Serbest-kenarlı (free-boundary) algoritmalar adından da anlaşılacağı gibi sınırların sabitlenmesini gerektirmeyen daha gelişmiş algoritmalar ve bu algoritmalar bahsedilen metriklerde olabildiğince en az distorsiyona neden olan projeksiyonu üretmeyi hedefler. Bilgisayar grafikleri alanı bu algoritmalar açısından oldukça zengindir. Bu algoritmaların en çok bilinen dört tanesi şunlardır (Lévy vd., 2002; Sheffer ve Sturler, 2001; Sheffer vd., 2005) :

- *En küçük karesel açı-korumalı izdüşüm* (Least square conformal mapping –LCSM)
- *En olası benzerlik* (As-similar-as-possible–ASAP)
- *En olası katılık* (As-rigid-as-possible–ARAP)
- *Açı tabanlı düzleştirme* (Angle based flattening–ABF)

Bu algoritmaların hepsi yapılan projeksiyondaki toplam distorsiyonu ölçen bir maliyet fonksiyonunu minimize etmeye çalışır. Toplam maliyet fonksiyonu, yerel (local) distorsiyonların tüm  $\Omega$  parametrik uzay üzerindeki ortalaması olarak tanımlanabilir. Yerel distorsiyonlar,  $\sigma_1$  ve  $\sigma_2$  özdeğerlerini kullanarak

her bir üçgen üzerinde ayrı ayrı hesaplanır (Abdul-Rahman vd., 2016). Toplam maliyet fonksiyonunun matematiksel olarak aşağıdaki gibi tanımlanır:

$$E = \frac{\sum_{t \in \Omega} E(\sigma_1, \sigma_2)}{\sum_{t \in \Omega} A(t)}. \quad (2.2)$$

Burada  $\sum_{t \in \Omega} E(\sigma_1, \sigma_2)$  belirli bir üçgende yerel distorsiyondur,  $A(t)$  ilgili üçgenin alanıdır ve  $\Omega$  ise tüm 2B parametrik uzaydır.

#### **2.1.4.1. En küçük karesel açı-korumalı izdüşüm**

Bu izdüşümde yerel distorsiyon iki özdeğer arasındaki farkın karesi olarak tanımlanır (Lévy vd., 2002) ve matematiksel ifadesi şöyledir:

$$E(\sigma_1, \sigma_1) = \frac{1}{2} (\sigma_1 - \sigma_2)^2. \quad (2.3)$$

Bu yerel distorsiyon ifadesi yukarıdaki genel toplam maliyet fonksiyonunda kullanılır ve dik-düşüş (gradient descent) gibi bir sayısal çözümleme metodu kullanılarak minimum değeri verecek bir sonuç üretilir. Formülden de anlaşılacağı üzere  $\sigma_1 = \sigma_2$  olması durumunda minimum sonucu veren çözüm ortaya çıkacaktır ve bu da açıları muhafaza eden (angle-preserving) bir projeksiyona karşı düşer. Yapılan çalışmalarda LSCM algoritmasının yerel alan distorsiyonu ürettiği ifade edilmiştir. Ayrıca bu algoritma yerel ve global bire-bir izdüşümü (bijectivity) garanti etmez ve uygun olmayan üçgenler ve çakışmalar üretebilir (Abdul-Rahman vd., 2016).

#### **2.1.4.2. En olası benzerlik (ASAP)**

Bu izdüşüm parametrik uzaydaki her bir üçgeni 3B uzaydaki karşılığına mümkün olabildiğince benzemeye zorlar. Bu teknik, açı koruyan izdüşümdeki aynı yerel distorsiyonu kullanır ve LCSM tekniğine çok yakın sonuçlar üretir.

### 2.1.4.3. En olası katılık (ARAP)

Parametrik uzaydaki her bir üçgeni, mümkün olabildiğince 3B uzaydaki karşılığı gibi sabit olmaya zorlar. Bu algoritma uzama (stretch) distorsiyonunu minimize etmeye çalışır ve yerel distorsiyonlar şu formülle ifade edilir:

$$E(\sigma_1, \sigma_2) = (\sigma_1 - 1)^2 + (\sigma_2 - 1)^2. \quad (2.4)$$

$\sigma_1 = \sigma_2 = 1$  olması durumunda bu ifadenin minimum değerli çözüme götüreceği açıktır. Bu teknik orijinal yüzeyin tüm metriklerini muhafaza etmeye çalışır (Abdul-Rahman vd., 2016).

### 2.1.4.4. Açı tabanlı düzleştirme (ABF)

Bu izdüşüm tekniği, Sheffer ve Sturler (2001) tarafından açı muhafaza eden projeksiyonda farklı bir yaklaşım benimsenerek geliştirilmiştir. Toplam maliyet fonksiyonunu, tüm yerel distorsiyonların ortalaması olarak tanımlamak yerine bu algortmada maliyet fonksiyonu 2B ile 3B uzaylarındaki açılar arasındaki fark olarak tanımlanır. ABF'de benimsenen maliyet fonksiyonunun matematiksel ifadesi aşağıdaki gibidir:

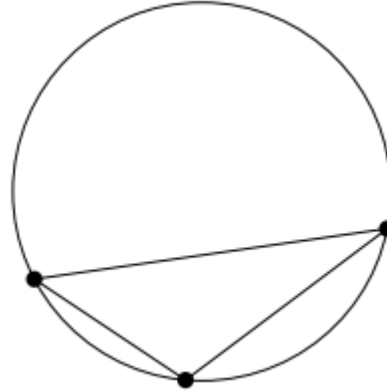
$$E_{ABF}(\alpha) = \sum_{t \in \Omega} \sum_{k=1}^3 (\alpha_i^k - \phi_i^k)^2 \omega_i^k. \quad (2.5)$$

Burada  $\alpha_i^k$ , 2B parametre uzayındaki bir açıyı;  $\phi_i^k$ , 3B üçgensel kafesteki açılardan türetilmiş optimal açıyı;  $\omega_i^k$  ise ağırlıklandırma çarpanını ifade eder. 2B parametre uzayındaki açıların geçerli bir üçgen tanımlayabilmesi için bazı kısıtların sağlanması gerekmektedir. Bu kısıtların en önemlilerinden biri: üçgenin iç açılarının toplamı  $180^\circ$  olması ve diğeri de bir köşenin etrafındaki açılarının toplamının  $360^\circ$  olmasıdır.

## 2.2. Delaunay Üçgenlemesi

Delaunay üçgenlemesi (Delaunay Triangulation), noktalar kümesinden birbiriyle üst üste gelmeyen ve süreklilik gösteren üçgenlerden oluşan bir kafes yaratmaya yarayan bir tekniktir (Li vd., 2004.). Matematikte ve hesaplamalı geometride, düzlemde verilen bir  $P$  ayrık noktalar kümesi için Delaunay üçgenlemesi,  $DT(P)$  olarak gösterilir. Ayrık noktalar kümesindeki hiçbir nokta bu üçgenlemede yer alan herhangi bir üçgenin çevrel çemberinin içerisinde yer almaz (De Loera vd., 2010; Hurtado vd., 1999). Delaunay üçgenlemesi, üçgenlemedeki üçgenlerin tüm açılarının en küçük açısını maksimuma çıkarır. Bu üçgenlemenin adı, 1934 yılından itibaren bu konudaki çalışmalarıyla bilinen Boris Delaunay'ın ismini (1934) almıştır.

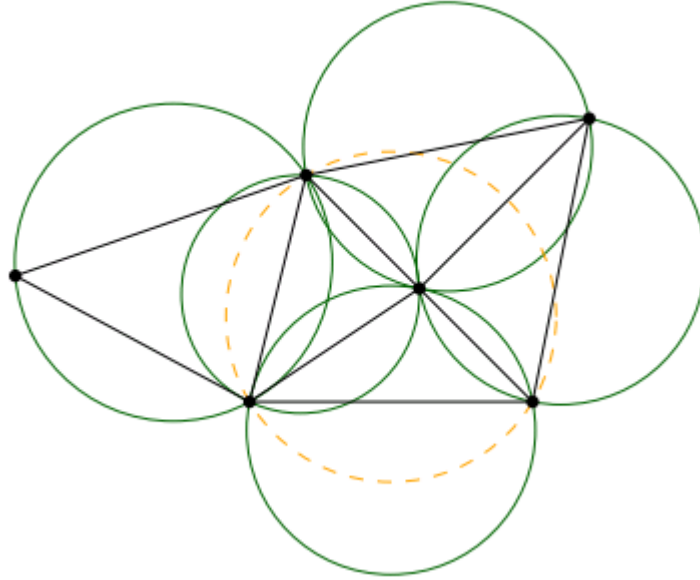
Bir üçgenin çevrel çemberi, üçgenin üç köşesinden de geçen eşsiz dairedir. Şekil 2.4'te bir üçgen ve onun çevrel çemberi görülmektedir.



Şekil 2.4. Bir üçgenin çevrel çemberi

Boş çevrel çember özelliği (empty circumcircle property) gereği,  $P \subset \mathbb{R}^2$  de tanımlı sınırlı bir nokta kümesinin Delaunay üçgenlenmesi olabilmesi için, her bir üçgenin çevrel çemberinin boş olması gerekir. Diğer bir ifadeyle,  $P$ 'deki noktalardan hiçbirinin bu çember içerisinde yer almaması gerekir. Şekil 2.5'teki örnekte, 6 noktadan oluşan bir kümenin Delaunay üçgenlenmesi görülmektedir. Şekilde görülen beş üçgenin de çevrel çemberleri boştur. Yani üçgenler boş çevrel çember özelliğini sağlamaktadır. Şekilde görülen kesikli çizgiyle

çizdirilmiş çember boş değildir. Fakat bu çember herhangi bir üçgenin çevrel çemberi olmadığından bir sorun teşkil etmez (Gartner ve Hoffmann, 2013).

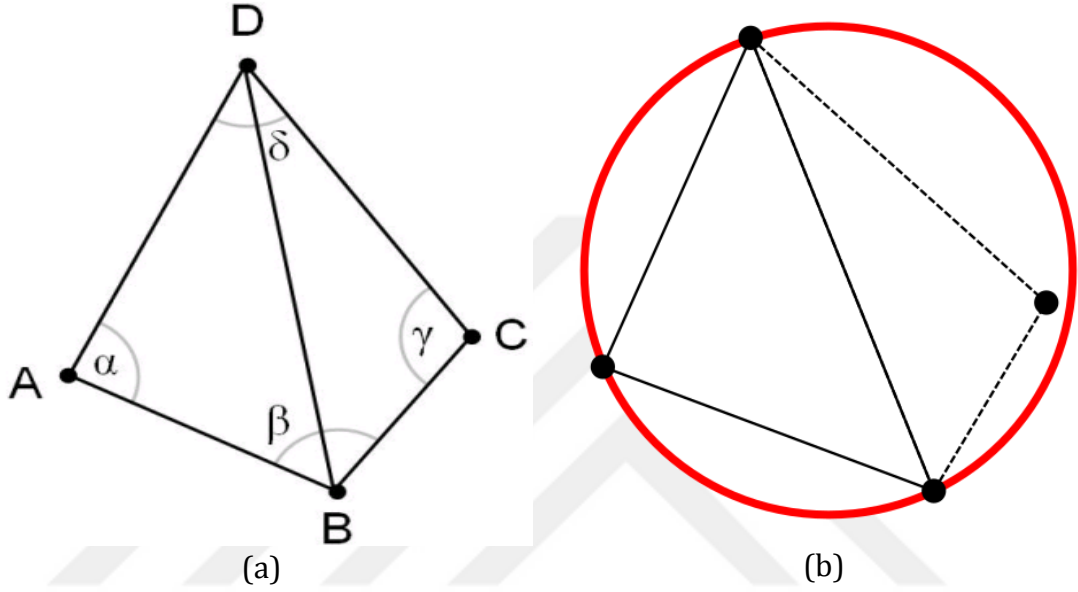


Şekil 2.5. Boş çevrel çember özelliğini sağlayan üçgenler (Gartner ve Hoffmann, 2013)

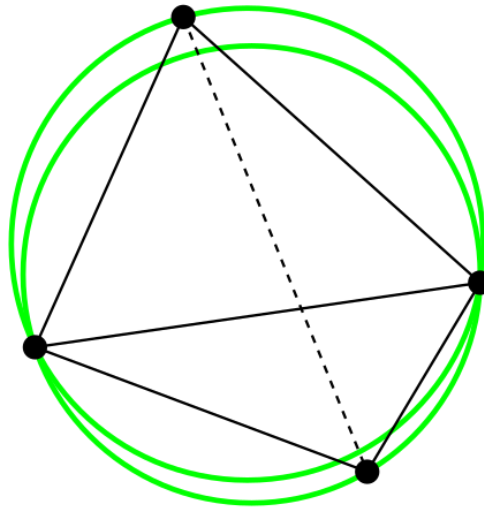
$n$  nokta sayısı ve  $d$  de boyut olmak üzere, Delaunay üçgenlemesinde düzlemde (yani  $d=2$  iken) dışbükey alanın (convex hull) üzerinde  $b$  adet nokta var ise o zaman bu noktaların herhangi bir üçgenlemesi en fazla  $m=2n-2-b$  adet üçgene sahip olur. Bir başka özellik olarak düzlemde, Delaunay üçgenlemesi en küçük açığı maksimuma çıkarır. Diğer üçgenlemelere kıyasla, Delaunay üçgenlemesinde en küçük açı, en az diğer üçgenlemelerdeki en küçük açı kadar büyüktür. Fakat Delaunay üçgenlemesi mutlaka en büyük açığı en aza indirmek zorunda değildir (Edelsbrunner vd., 1992). Delaunay üçgenlemesinin pek çok özelliği daha vardır. Bu özelliklere ayrıntılı olarak Gartner ve Hoffmann tarafından (2013) yazılan ders notlarında ve De Loera vd., tarafından (2010) yazılan kitapta yer verilmiştir.

Bahsedilen Delanuay özelliklerine bakıldığında önemli bir ayırt edici özellik ortaya çıkar. Şekil 2.6'da BD ortak kenarlı ABD ve BCD üçgenlerine bakıldığında, eğer  $\alpha$  ve  $\gamma$  açılarının toplamı  $180^\circ$ 'ye eşit veya daha küçük ise bu üçgenler Delanuay koşulunu sağlar (Edelsbrunner vd., 1992). Bu özellik, ortak kenarı

karşıtıyla-değiştirme (flipping) tekniğini kullanmaya olanak tanır. Eğer iki üçgen Delaunay koşulunu sağlamazsa, ortak kenar BD'yi karşıtıyla-değiştirip AC'yi seçmek Delaunay koşulunu sağlayan iki üçgen üretir. Bu işlem karşıtıyla-değiştirme olarak anılır. Bu işlemi yaparak Delaunay kriterini sağlayan üçgenleme örneği Şekil 2.7'de görülmektedir.

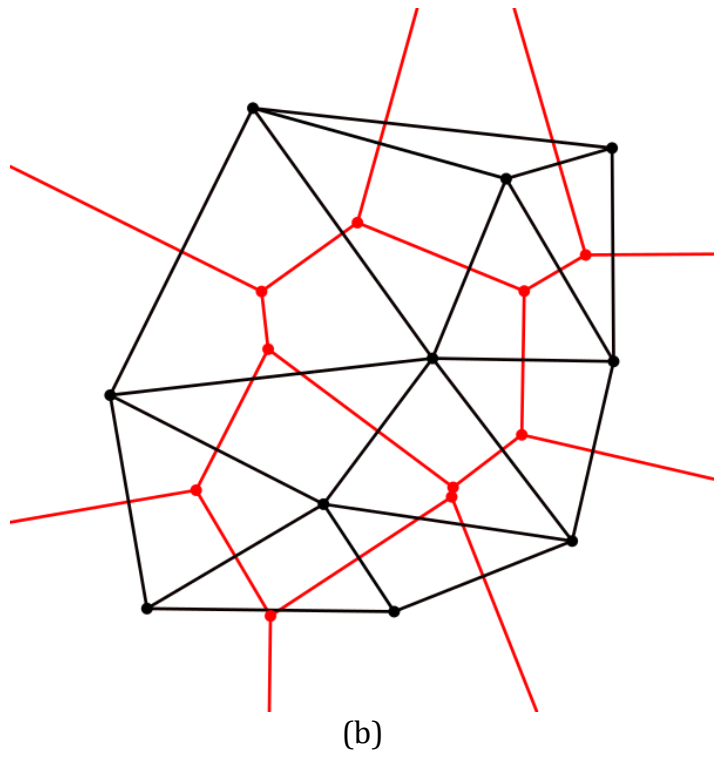
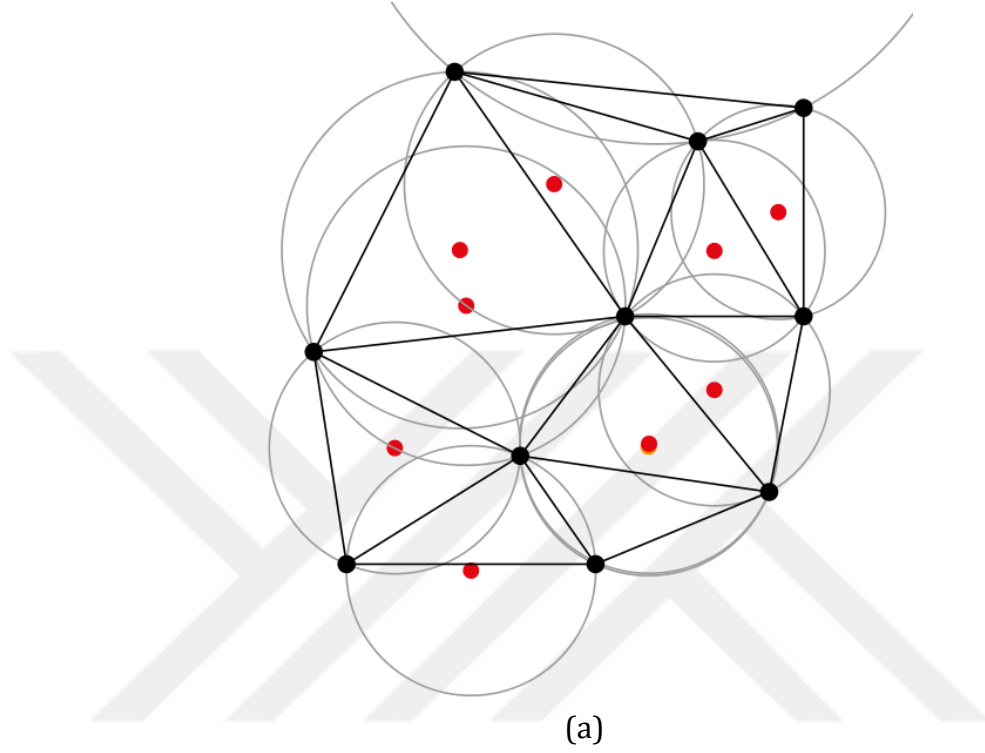


Şekil 2.6. Delaunay koşulunu sağlamayan üçgenleme (De Loera vd., 2010)



Şekil 2.7. Ortak kenar değiştirme işlemi sayesinde Delaunay koşulunu sağlayan üçgenleme (De Loera vd., 2010)

Ayrık P noktalar kümesinin genel pozisyondaki Delaunay üçgenlemesi, P için üretilen Voronoi diyagramının ikili grafına karşılık gelir. Şekil 2.8a'da örnek bir Delaunay üçgenlemesi görülmektedir.



Şekil 2.8. Delaunay üçgenleme (a) ile Voronoi diyagramı (b) arasındaki ilişki

Şekilde tüm çevrel çemberler gri renki olarak ve bu çemberlerin merkezleri kırmızı renkli nokta ile çizdirilmiştir. Bu şekildeki çevrel çemberlerin merkezleri birleştirmek ise Voroni diyagramını üretir. Voroni poligonu olarak da anılan bu diyagram Şekil 2.8b'de verilmiştir.

Delaunay üçgenlemesi, nokta kümesiyle ilgili uygulamalardaki tartışmasız en önemli üçgenlemedir. Bu konuda çok geniş bir literatür vardır (De Loera vd., 2010). Bir taraftan nokta kümesinin Voroni diyagramının ikili grafiği olduğundan, nokta kümesi üzerinde mesafe ile ilgili problemleri çözmek için kullanılacak faydalı bir araçtır. Mesela kenarlara ilişkin veri yapısı, nokta kümesine ait asgari Öklid tarama ağacı (minimum Euclidean spanning tree) içerir. Diğer taraftan Delaunay üçgenlemesi, nokta kümesinin en yeknesak (most uniform) üçgenlemesi olarak nitelendirilir. Yani bu, simplekslerinin düzenli simplekslere mümkün olduğunca yakın olduğu anlamına gelir. Bu da onu çözüm ağı oluşturma problemleri için iyi bir aday yapar. Arazi (terrain) ve nesnelere modellemede verilen örnek noktalar kümesi için Delaunay üçgenlemesi, düzgün bir üçgen kümesi verir ve bu üçgen kümesi modelde poligon olarak kullanılır. Delaunay üçgenleme özellikle dar açılı üçgenlerden kaçınmaya yarar. Ayrıca Delaunay üçgenlemeleri genellikle sonlu elemanlar yöntemi ve sonlu hacim fizik simülasyonu gibi alanla ayrıştırılmış çözücüler (space-discretised solvers) için kafesler oluşturmak amacıyla kullanılır. Çünkü bu üçgenleme yönteminde açı garantisi vardır ve hızlı üçgenleme algoritmaları mevcuttur.

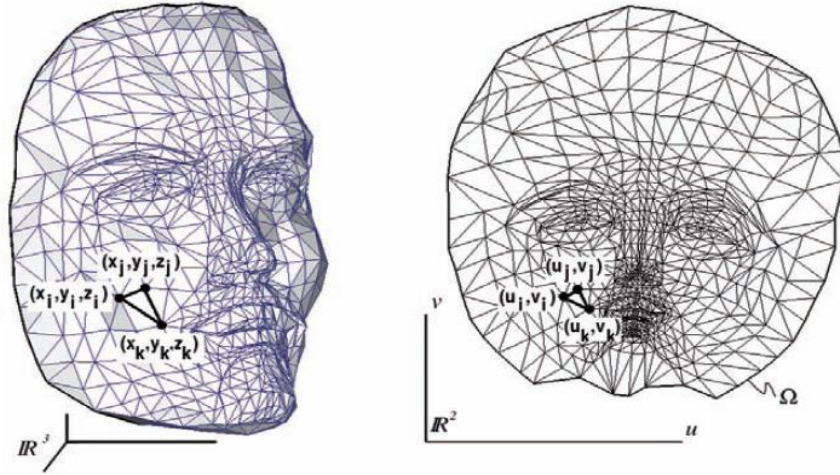


### 3. MATERYAL ve YÖNTEM

Bu bölümde, bu çalışmada benimsenen yüzey düzleştirme yaklaşımına temel oluşturan teknikler ve önerilen hibrit düzleştirme yöntemi açıklanmıştır.

#### 3.1. Üçgenel Yüzeyin Parametrenmesi

Üçgenlenmiş yüzeyler (triangulated surfaces), parça parça lineer fonksiyonlar kullanılarak parametreleştirilebilmektedir. Burada parçalar yüzeyi oluşturan üçgenlere karşı düşmektedir (Botsch vd., 2010). Üçgenlenmiş  $S$  yüzeyi, üç boyutlu uzaydaki  $\mathbf{x}_i=(x_i, y_i, z_i) \in \mathbb{R}^3$  köşe noktalarından ve  $F$  ile tanımlanan üçgenel yüzlerden oluşur.  $S$  yüzeyinin  $\mathbf{x}_i=(x_i, y_i, z_i)$  köşeleri (veya düğümleri),  $P$  düzlemsel üçgenlemesi  $S$  yüzeyine benzeyecek şekilde  $\mathbf{u}_i=(u_i, v_i) \in D$  noktalarına izdüşürülür. Burada  $D \subset \mathbb{R}^2$  ve konveks biçimdedir. Şekil 3.1’de 3B uzayda tanımlı bir yüzey (solda) ve bunun 2B parametrik uzaydaki karşılığı (sağda) gösterilmiştir.



Şekil 3.1. 3B uzayda tanımlı bir yüzeyin (solda) 2B parametrik uzaydaki karşılığı (sağda) (Hormann vd., 2007)

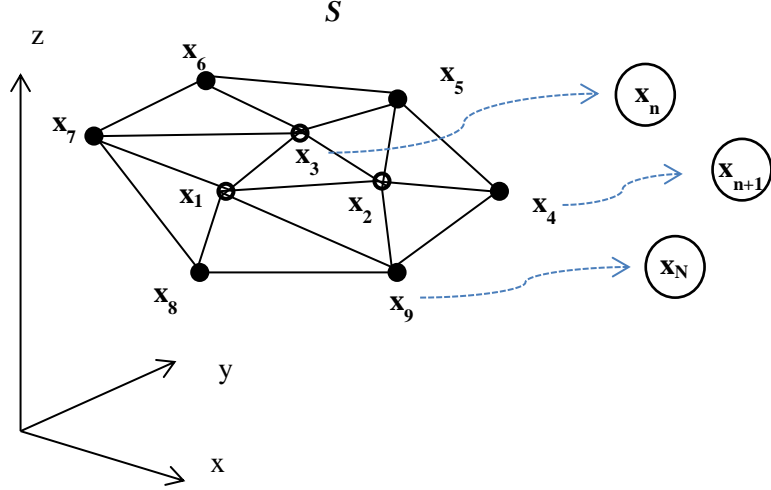
Parametrelemedeki temel mantık, her  $\mathbf{u}_i$  noktasının kendi komşularının konveks kombinasyonu olacak şekilde belirlenmesidir. Özetlemek gerekirse üçgenlenmiş yüzeyin parametrenmesi her bir  $i$  köşesi (vertex) ile ilişkili  $(u_i, v_i)$  koordinatının bulunması anlamına gelmektedir. Buna ilave olarak, yüzeyi

oluşturan üçgen yüzlere parametrik uzayda karşı düşen koordinatların kendileriyle kesişmemesi gerekmektedir. Bunun anlamı şudur: parametre uzayında herhangi iki üçgenin kesişimi; ya bir ortak kenar (common edge), ya bir ortak köşe (common vertex) ya da boş bir kümedir (Botsch vd., 2010).

### 3.2. Barisentrik İzdüşüm

Barisentrik izdüşüm (Barycentric mapping), üçgenlenmiş bir yüzeyin parametrelenmiş halini üretmede kullanılan en yaygın yöntemlerden birisidir (Botsch vd., 2010). Bu yöntem, graf teorisi alanındaki çalışmalarıyla bilinen Tutte'nin (1960) barisentrik izdüşüm teoremine dayanmaktadır ve yine bu alandaki etkili çalışmalarıyla bilinen Floater tarafından üçgenlenmiş yüzeyi parametreleme ve yaklaşımlandırma (Floater, 1997) çalışmalarına temel oluşturan bir yöntemdir. Bu teoreme göre diske benzer yapıda verilen bir üçgenlenmiş yüzey için eğer sınır düğümlerinin koordinatları  $(u,v)$  bir konveks poligonun üzerinde yer alırsa ve eğer iç düğümler komşularının konveks kombinasyonu ise bu durumda tüm  $(u,v)$  koordinatları geçerli bir parametreleme oluşturur.

Yapılmak istenen şey,  $S$  yüzeyinin biçimini yerel olarak koruyacak şekilde  $P$  düzlemsel şeklin, konveks kombinasyonlarının seçiminin belirlenmesini sağlayan algoritmayı oluşturmaktır.  $S(X, F)$  yüzeyi, köşe  $(X = \{ \mathbf{x}_i = (x_i, y_i, z_i) \}, 1 \leq i \leq N)$  ve  $F$  yüzlerinden (*Delaunay Triangulation* fonksiyonu ile üretilmiş üçgenlerin indislerini tutan matris) oluşan bir üçgenlenmiş yüzey olsun. 3B uzayındaki düğümleri yeniden etiketlemek sonraki formülasyonları daha anlaşılır kılacaktır. Bu amaçla,  $\mathbf{x}_1, \dots, \mathbf{x}_n$  iç düğümler (internal nodes) olarak ve  $\mathbf{x}_{n+1}, \dots, \mathbf{x}_N$  de sınır düğümleri (boundary nodes) olarak kabul edilebilir. Yüzeyin sınırlarındaki düğüm adedi  $K$  ile gösterilmek üzere bu indisler arasında  $K=N-n$  bağıntısı yazılabilir. Düğümlerin yeniden etiketlenmesine ilişkin bir örnek Şekil 3.2'de verilmiştir.



Şekil 3.2. Düzleminde bir poligonun köşelerinin yeniden etiketlenmesine ilişkin bir örnek

Bu bölümdeki ve Bölüm 3.1'deki tanımlamalar göz önünde bulundurulursa ağırlık merkezli izdüşüm algoritması aşağıdaki adımlardan oluşmaktadır:

1. İzdüşümün yapılacağı konveks poligonun koordinatları (düzlemsel yüzey üzerinde) belirlenir. Bu belirleme işlemi basitçe birim kare üzerinde, birim çember üzerinde veya daha karmaşık bir biçimde 3B uzaydaki yüzeyin kenarlarına oldukça benzer bir poligon üzerinde yapılabilir. Özetle,

$$\mathbf{U}_b = \mathbf{u}_{n+1}, \dots, \mathbf{u}_N. \quad (3.1)$$

$D \subset \mathbb{R}^2$  olan  $K$ -kenarlı konveks poligonun köşeleri (ve/veya kenarlarında tanımlı) olacak biçimde ve saat yönünün tersi sırada belirlenir.

2. Her bir  $i \in \{1, \dots, n\}$  için  $j=1, \dots, N$  olacak şekilde  $\lambda_{i,j}$  değerleri aşağıdaki şartları sağlayacak şekilde gerçek sayılardan seçilir (Botsch vd., 2010; Floater, 1997):

$$\left. \begin{array}{l} \lambda_{i,j} > 0 \text{ ve} \\ \lambda_{i,i} = -\sum_{j \neq i}^N \lambda_{i,j}, \\ \lambda_{i,j} = 0, \end{array} \right\} \begin{array}{l} \text{eğer } i. \text{ ve } j. \text{ düğümler bir kenar ile bağlı ise} \\ \text{diğer durumlarda} \end{array} \quad (3.2)$$

3.  $\mathbf{u}_1, \dots, \mathbf{u}_n$  ařađıdaki lineer sistem denklemlerinin çözümleri olarak tanımlanır:

$$\mathbf{u}_i = \sum_{j=1}^n \lambda_{i,j} \mathbf{u}_j \quad i=1, \dots, n. \quad (3.3)$$

Bu denklemler çözüldüğünde,

$$\mathbf{P} = \mathbf{P}(\mathbf{F}, \mathbf{U}_b, \Lambda), \quad (3.4)$$

elde edilir ve burada

$$\mathbf{U}_b = \{ \mathbf{u}_{n+1}, \dots, \mathbf{u}_N \} \text{ ve } \Lambda = (\lambda_{i,j})_{i=1, \dots, n; j=1, \dots, N} \text{ 'dir.} \quad (3.5)$$

Burada  $\mathbf{P}$  yüzeyi,  $\mathbf{S}(\mathbf{X}, \mathbf{F})$ 'nin  $\mathbb{R}^2$  deki yerleşimidir.  $\mathbf{u}_1, \dots, \mathbf{u}_N$  vektörleri birbirleriyle olan ilişkileri  $\mathbf{F}$  ile tanımlanan topolojiye göre düzlemde doğrularla (straight lines) bađlı düğümleri göstermektedir.

Tutte (1963) yıllar önce düzlemsel grafın düz çizgilerinin çizilmesiyle ilgili çalışmalar yapmıştır. Üçgenlenmiş geniş ölçekli graflar için Tutte, komşu ( $i=1, \dots, n$ ) düğümler için  $\lambda_{i,j}=1/d_i$  olması durumunda Denklem 3.3'ü çalıştırmayı önermiştir (yani,  $\mathbf{u}_i$  komşularının ağırlık merkezidir). Burada  $d_i$  parametresi,  $i$ . düğümün komşu sayısını gösterir.  $\mathbf{F}$  matrisi kafesin (grafın) topolojisini içerdiğinden, her bir düğümün komşu sayısı, bir halka (one-ring) komşuluđu için matris üzerinde bazı genel satır sıralama ve sayma işlemleri uygulayarak kolayca hesaplanır. Tutte bu özel şartta tekil bir çözümün olduğunu ve  $\mathbf{P}$ 'nin doğrulardan oluşan bir çizim olduğunu kanıtlamıştır (Floater, 1997). Diđer bir ifadeyle  $\mathbf{u}_1, \dots, \mathbf{u}_n$  eşsizdir ve farklıdır. Bu da  $\mathbf{P}$  düzleminde herhangi iki kenarın (üçgenlerin) ortak köşelerin haricinde çakışmayacağını gösterir.

Denklem 3.2'deki genel durum göz önünde bulundurularak Denklem 3.3'ün eşsiz (unique) çözüme sahip olduğu gösterilmek istenirse, bu denklemler aşağıdaki gibi yeniden yazılabilir (Floater, 1997; Botsch vd., 2010)

$$\mathbf{u}_i - \sum_{j=1}^n \lambda_{i,j} \mathbf{u}_j = \sum_{j=n+1}^N \lambda_{i,j} \mathbf{u}_j \quad i=1, \dots, n. \quad (3.6)$$

Buradaki mantık sınırdaki düğümleri konveks poligon üzerinde sabitlemek ve daha sonra iç düğümlerdeki koordinatları Denklem 3.2 ve Denklem 3.3'ü çözerek bulmaktır.  $\mathbf{u}_i$ 'nin  $u_i$  ve  $v_i$  bileşenleri ayrı ayrı düşünülürse, o zaman aşağıdaki iki matris denklemi eşdeğer bağıntı olarak yazılabilir:

$$\mathbf{A}\mathbf{u} = \mathbf{b}_1, \quad \mathbf{A}\mathbf{v} = \mathbf{b}_2. \quad (3.7)$$

Bu  $n$  (iç düğüm adedi) boyutundaki iki lineer sistem çözülür. Burada  $\mathbf{u}$  ve  $\mathbf{v}$  sütun vektörleri sırasıyla  $(u_1 \dots u_n)^T$  ve  $(v_1 \dots v_n)^T$  biçiminde olup bunlar iç düğümlerin  $u$  ve  $v$  koordinatlarını tutar. Denklemlerin sağ tarafındaki  $\mathbf{b}_1$  ve  $\mathbf{b}_2$  sınırdaki düğümlerin ağırlıklı koordinatlarını barındırır.  $\mathbf{A}$ -matrisi, elemanları aşağıdaki bağıntılarla verilen  $n \times n$  boyutunda bir kare matristir.

$$a_{i,i}=1, \quad a_{i,j}=-\lambda_{i,j}, \quad j \neq i, \quad (3.8)$$

$$\mathbf{A} = \begin{bmatrix} 1 & -\lambda_{1,2} & \dots & -\lambda_{1,n} \\ -\lambda_{2,1} & 1 & \dots & -\lambda_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ -\lambda_{n,1} & -\lambda_{n,2} & \dots & 1 \end{bmatrix}. \quad (3.9)$$

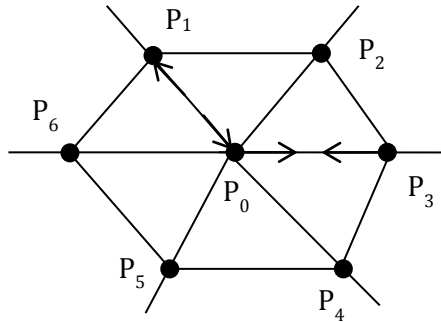
Floater'a (1997) göre Denklem 3.3'ün eşsiz çözümünün varlığı  $\mathbf{A}$  matrisinin tekil olmayan durumuna (*non-singularity*) denktir, yani  $\mathbf{A}$ 'nın tersi bulunarak elde edilebilir. Denklem 3.6'yı çözebilmek için pek çok yol vardır. Geniş ölçekli kafesler için en etkin çözümler "sparse iterative" ve "direct" yöntemleridir. Makul derecedeki küçük kafeslerde (örneğin 5K adet düğüme kadar) basit "Gauss-Seidel solver" kullanılabilir (Botsch vd., 2010).

### 3.3. Enerji Tabanlı Yüzey Düzleştirme

Enerji tabanlı yüzey düzleştirme süreci, kütle-yay sisteminin enerji fonksiyonu tarafından yönetilen bir (düzlemde) üçgenel kafes deformasyon süreci olarak tanımlanabilir (Wang vd., 2002). Düzlemsel üçgenel kütle-yay sistemi, üçgenel kafesin biçimini kontrol etmek ve benzetim (simulation) yapmak için kullanılır. Tüm model, komşuların yaylarla birbirine bağlı olduğu sanal kütlelerden oluşur (Liu vd., 2013).

#### 3.3.1. Enerji fonksiyonunun tanımı

Bu teknikte düzlemdeki üçgenel  $\Omega$ -kafesinin deformasyonu için bir kütle-yay sistemi kurulur.  $\Omega$  kafesi  $(\mathbf{K}, \mathbf{P})$  çiftiyle gösterilebilir. Burada  $\mathbf{K}$ , düğümlerin birbirleriyle olan bağlantısını, yani üçgenel kafesin topolojik yapısını,  $\mathbf{P}=\{P_1, \dots, P_m\}$  ise  $P_i \in \mathbb{R}^2$  kafesin biçimini tanımlayan düğümlerin konumlarını temsil eder. Şekil 3.1'de verilen 3B kafes yüzeyi,  $(\mathbf{K}, \mathbf{Q})$  çiftiyle gösterilebilir. Burada  $Q_i \in \mathbb{R}^3$  uzayındadır ve her bir  $Q_i$  düzlemdeki  $P_i$ 'ye karşılık gelir. Kütle-yay sisteminde pek çok parametre, onlara karşı düşen geometrik parametrelerden türetilir. Örneğin kuvvetler, elastik deformasyon enerjisi tarafından ve kütleler de, üçgenel kafes düğümlerinin arasındaki göreceli mesafeler ve üçgenlerin alanları tarafından belirlenir. Başlangıçtaki üçgenel kafes düğümlerinin konumları ile 2B yüzeyin son hali arasındaki fark düzlemsel kütle-yay sisteminde depolanan elastik deformasyon enerjisi olarak düşünülür. Örnek bir kütle-yay sistemi Şekil 3.3'te verilmiştir.



Şekil 3.3. Kütle-yay sistemi

Burada  $P_i$  ile gösterilen düğümler kütlelere ve  $P_i$ - $P_j$  kütleleri arasındaki bağlantılar ise yaylara karşı düşer. Deformasyon esnasında geliştirilen düzlemsel yüzey üzerindeki  $P_i$  ve  $P_j$  arasındaki mesafe, buna karşılık gelen uzaysal (3B) mesafeden daha büyük ise iki düğüm arasında çekme (attraction) kuvveti uygulanır (mesela Şekil 3.3'teki  $P_0$  ve  $P_3$  arasında olduğu gibi). Aksi takdirde ise düğümler arasında itici kuvvet olur (mesela  $P_0$  ve  $P_1$  arasında). Tek bir  $P_i$  kütlelerine ilişkin elastik deformasyon enerjisi ve gerilme kuvveti şu bağıntılarla hesaplanır (Wang vd., 2002):

$$E(P_i) = \sum_{j=1}^n \frac{1}{2} C(|P_i P_j| - d_j)^2, \quad (3.10)$$

$$\vec{f}(P_i) = \sum_{j=1}^n \frac{1}{2} C(|P_i P_j| - d_j) \vec{n}_{P_i P_j}. \quad (3.11)$$

Burada  $C$ , yay sabiti;  $|P_i P_j|$ , düzlemsel yüzeydeki  $P_i$  ve  $P_j$  arasındaki güncel mesafe;  $d_j$  uzunluğu  $Q_i$  ve  $Q_j$  arasındaki mesafe;  $\vec{n}_{P_i P_j}$  vektörü,  $P_i$ 'den  $P_j$ 'ye doğru giden birim vektör;  $n$  ise  $P_i$ 'ye komşuluğu olan düğüm sayısıdır. Özetle, kuvvetler düzlemsel yüzey ile verilen 3B yüzey arasındaki üçgen kenar farklarından üretilir.

Yüzey düzleştirmenin amacı, verilen 3B kafes yüzeye en uygun düzlemsel üçgen karşılığını ( $\Omega$ ) elde etmektir. Yani bulmak istenen şey, aşağıdaki enerji fonksiyonunu minimize edecek düzlemsel kafesi ( $\Omega = (\mathbf{K}, \mathbf{P})$ ) tanımlayan  $\mathbf{K}$  ve  $\mathbf{P}$ 'dir.

$$E(\Omega) = \sum_{i=1}^m E(P_i). \quad (3.12)$$

### 3.3.2. Enerji yayma

Enerji modelinde enerjii yaymak için genellikle aşağıda verilen Lagrange denkleminde faydalanılır (Metaxas, 1997).

$$Mq'' + Dq' + Kq = g_q + f_q. \quad (3.13)$$

Burada M, D ve K sırasıyla yay-kütlesi (spring mass), sönümlenme (damping) ve gerginlik (stiffness) matrisleridir.  $g_q$ , yer çekimi kuvveti ve  $f_q$  ise modelin serbestlik dereceleriyle ilgili genel harici kuvvettir. Bu iki kuvvet sıfıra eşitlenir ve sönümlenme elemanı ihmal edilirse kütle-yay sistemindeki Lagrange denklemi basitleştirilerek aşağıdaki denklem elde edilir:

$$Mq'' + Kq = 0. \quad (3.14)$$

Burada  $Kq$  teriminin değeri Denklem 3.11'deki  $-\vec{f}$  kuvvetine eşittir. Denklem 3.14'ün çözümünde Euler yöntemi kullanılır.  $\Delta t$  çok küçük ise  $P_i$  kütesinin ivmesi sabit kabul edilebilir. Tüm kütle-yay sisteminin dengesi, her bir düğümün bireysel dengelerinden oluşur. Her bir  $P_i$  düğümü için, Denklem 3.14 şu hale dönüştürülebilir (Wang vd., 2002):

$$m_i = \frac{p}{3} \sum A_k, \quad (3.15)$$

$$q_i''(t) = \frac{f_i(t)}{m_i}, \quad (3.16)$$

$$q_i'(t + \Delta t) = q_i'(t) + \Delta t q_i''(t), \quad (3.17)$$

$$q_i(t + \Delta t) = q_i(t) + \Delta t q_i'(t) + \frac{\Delta t^2}{2} q_i''(t). \quad (3.18)$$

Burada  $m_i$ ,  $P_i$ 'nin kütlesi;  $p$ , yüzeyin alan yoğunluğu;  $A_k$ ,  $P_i$  düğümünü içeren  $k$ . üçgenin model uzayındaki alanı;  $q_i''(t)$ ,  $P_i$  düğümünün  $t$  anındaki ivmesi;  $f_i(t)$ ,  $P_i$  düğümüne  $t$  anında etkiyen gerilme kuvveti;  $q_i'(t)$ ,  $P_i$  düğümünün  $t$  anındaki hızı ve  $q_i(t)$ ,  $P_i$  düğümünün  $t$  anındaki konumudur. Buradaki  $p$ , yüzeyin gerçek yoğunluğu değildir. Alan yoğunluğu  $p$  ve yay sabiti  $C$  ayrıca mekanik sistemin özellikleri test edilerek de belirlenebilir. Genellikle fizik tabanlı modellerin çoğunda  $p$  ve  $C$ , sadece deformasyonu daha uygun hale getiren ölçekleme katsayılarıdır (Wang vd., 2002; Metaxas, 1997). Örneğin  $p = 1/\min\{m_i\}$  ve  $C=0.5$  alınabilir. Çoğu



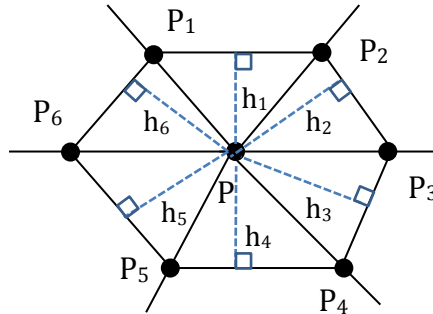
zaman adım büyüklüğü  $\Delta t=0.01$  alınır ve bu da her zaman kabul edilebilir bir hızda kararlı duruma ulaştırır (Wang vd., 2002).

### 3.3.3. Ceza fonksiyonu

Yüzey geliştirme esnasında örtüşme ve çakışma, yani üst üste binme (overlap) kolayca söz konusu olabilir. Çakışmayı önlemek için hareket eden kütlelere ceza (penalty) kuvveti eklenir. Provot'un (1995) yöntemi uyarlanarak P kütlesi için şu şekilde bir ceza fonksiyonu tanımlanabilir:

$$\vec{F}_{\text{penalty}} = - \sum_{j=1}^m C_{\text{penalty}} |h_j - h_j^*| \vec{n} \quad \left\{ \begin{array}{l} C_{\text{penalty}} = 1 \quad (h_j \leq h_j^*) \\ C_{\text{penalty}} = 0 \quad (h_j > h_j^*) \end{array} \right. \quad (3.19)$$

Burada  $h_j$ , düzlemsel yüzeyde P noktasından karşı kenara olan güncel mesafedir. (Örneğin Şekil 3.4'te P noktasından  $\overline{P_1P_2}$ ,  $\overline{P_2P_3}$ ,  $\overline{P_3P_4}$ ,  $\overline{P_4P_5}$ ,  $\overline{P_5P_6}$ ,  $\overline{P_6P_1}$  doğrularına olan mesafelerdir).  $h_j^*$ , orijinal yüzeyde Q ile j. karşı kenar arasındaki Öklidik mesafesidir. Q ise, 3B kafes üzerinde P düğümüne karşılık gelen köşedir.  $\vec{n}$ , düzlemsel yüzey üzerinde P noktasından karşı kenara doğru giden birim vektördür. m ise P'nin karşı kenarlarının sayısıdır.



Şekil 3.4. Ceza fonksiyonları (Wang vd., 2002)

$\vec{F}_{\text{penalty}}$  ceza fonksiyonu bir vektördür ve bu vektör,  $P_i$  kütlesi (düğümü) karşı kenara çok fazla yakın bir konuma doğru hareket ettiği zaman çakışmayı önlemek amacıyla bu ceza fonksiyonu  $P_i$  konumuna uygulanır:

$$q_{i_t}^* = q_{i_t} + \vec{F}_{\text{penalty}} \quad (3.20)$$

### 3.3.4. Doğruluk ölçümü

Geliştirilen düzlemsel yüzeyin doğruluğunu değerlendirmek için genellikle iki kriter kullanılır. Bunlar *alan doğruluğu* (area accuracy) ve *kenar/biçim doğruluğudur* (shape accuracy).

#### 3.4.4.1 Alan doğruluğu

Geliştirme süreci boyunca yüzey alanı değişim gösterebilir ve nihai göreceli alan farkı şu bağıntıyla hesaplanabilir (Wang vd., 2002):

$$E_s = \frac{\sum |A - A'|}{\sum A} \quad (3.21)$$

Burada  $A$ , geliştirme öncesindeki orijinal yüzey üzerindeki bir yamanın gerçek alanı ve  $A'$  ise geliştirme sonrasında ortaya çıkan düzlemsel yüzeydeki buna karşılık gelen yamanın alanıdır.  $S(u,v)$  ile tanımlı bir yüzeyin alan hesabı için iki katlı integral bağıntısının çözülmesi gerekir (Wang vd., 2002). Bu yaklaşımda, karmaşık bir yüzeyi sunmak için “yüzeycik” (facet) modeli kullanıldığından  $S$  yüzeyinin analitik formu belirlenemez. Bu sebeple  $A$ , yaklaşık olarak bu modeldeki her bir üçgenin alanı toplanarak basitçe şu bağıntıyla hesaplanabilir (Wang vd., 2002):

$$A = \sum_{i=0}^{n-1} A_i \quad (3.22)$$

Burada  $A_i$ , modeldeki  $i$ . üçgenin alanı ve  $n$  ise üçgenlerin sayısıdır.

### 3.3.4.2 Biçim/Kenar doğruluğu

Biçim hatası, orijinal yüzey üzerindeki eğrinin (curve) uzunluğu ile geliştirilen yüzeyde buna karşılık gelen kenarın (edge) uzunluğu arasındaki fark ile temsil edilir (Wang vd., 2002). Nihai göreceli kenar farkı şu bağıntıyla hesaplanabilir:

$$E_c = \frac{\sum |L-L'|}{\sum L}. \quad (3.23)$$

Burada  $L$ , geliştirme öncesindeki orijinal yüzey üzerindeki bir eğrinin gerçek uzunluğu ve  $L'$  ise geliştirme sonrasında ortaya çıkan düzlemsel yüzeyde  $L'$ 'ye karşı düşen kenarın uzunluğudur.  $S$  yüzeyi üzerindeki  $C=S(u(t),v(t))$  eğrisinde  $L$  eğri parçasının yer aldığı varsayılırsa  $L$  uzunluğu karmaşık bir integral çözümünü gerektirir (Wang vd., 2002).  $S$  yüzeyindeki benzer sebepten ötürü  $C$  eğrisinin analitik formu da belirlenemez. Dolayısıyla  $L$ , yüzeycik modelindeki her bir üçgenin kenar uzunlukları toplanmak suretiyle kabaca hesaplanabilir:

$$L = \sum_{i=0}^{m-1} L_i. \quad (3.24)$$

Burada  $L_i$ , modeldeki  $i$ . üçgenin kenarı ve  $m$  ise üçgenlerin sayısıdır.

### 3.3.5. Enerji yayma algoritması

Aşağıda ayrıntıları verilen *EnerjiYayma( $\Omega$ )* isimli algoritma,  $E(\Omega)$  enerji fonksiyonunu minimize eden  $\Omega=(\mathbf{K}, \mathbf{P})$  düzlemsel kafesini tanımlayan  $\mathbf{P}$  düğüm köşe koordinatlarını bulmak için Wang vd. (2002) tarafından önerilmiştir.

---

#### Algoritma-1 *EnerjiYayma( $\Omega$ )*

---

**Input:** Başlangıç izdüşümü yapılmış 2B noktalar kümesi  $\mathbf{P}=\{ P_0, P_1, \dots, P_{n-1}\}$  ve düğüm sayısı  $n$ .

**Output:**  $E(\Omega)$ 'i minimize eden  $\mathbf{P}$  noktalarının nihai konumları.

- 1: **for**  $i$  **from** 0 **to**  $n-1$  **do**
  - 2:     Denklem 3.15 kullanılarak  $P_i$  düğümünün kütlesi hesaplanır.
  - 3: **end for**
  - 4: **while** ((göreceli alan farkı  $E_s >$  izin verilen doğruluk **or** göreceli kenar uzunluk farkı  $E_c >$  izin verilen doğruluk) **and**  $E(\Omega)$  değişimi  $>$  izin verilen  $\epsilon$  yüzdesi) **and** iterasyon sayısı  $<$  izin verilen  $N$  sayısı ))
  - 5:     **for**  $i$  **from** 0 **to**  $n-1$  **do**
  - 6:         Denklem 3.11 kullanılarak  $P_i$  düğümü üzerine etkiyen gerilme kuvveti hesaplanır.
  - 7:         Denklemler 3.16 – 3.18 kullanılarak  $P_i$  düğümünün yeni konumu hesaplanır.
  - 8:         Denklemler 3.19 ve 3.20 kullanılarak  $P_i$  düğümüne uygulanacak ceza fonksiyonunu hesaplanır.
  - 9:          $P_i$  düğümünü yeni konumuna taşınır.
  - 10:     **end for**
  - 11:     Denklem 3.21 kullanılarak yeni  $E_s$  değerini hesaplanır.
  - 12:     Denklem 3.23 kullanılarak yeni  $E_c$  değerini hesaplanır.
  - 13:     Denklemler 3.10 ve 3.12 kullanılarak  $E(\Omega)$ 'nin yeni değeri hesaplanır.
  - 14: **end while**
  - 15: **return**  $\Omega$
- 

### 3.4. Dinamik Sinir Ağlar

Hesaplama açısından bakıldığında, durum geri-beslemesi içeren bir dinamik sinir yapısı, statik ağ yapısına göre daha çok hesaplama avantajı sağlayabilir. Genellikle, küçük bir geri-besleme sistemi çok daha büyük ve belki de sonsuz büyüklükte bir ileri-beslemeli sisteme eşdeğerdir (Hush ve Horne, 1993). Bunun en bilinen örneği, bir keyfi sonlu durum makinesini taklit etmek için sonsuz sayıda ileri-beslemeli lojik kapı gerekmesidir. Ayrıca, tek kutuplu sonsuz dürtü yanıtı (infinite impulse response-IIR) filtreyi taklit etmek için sonsuz dereceli bir sonlu dürtü yanıtı (finite impulse response-FIR) filtreye ihtiyaç duyulur. Doğrusal olmayan dinamik geri-beslemeli nöral ağ yapısı, biyolojik sinir sistemlerinde olduğu gibi dağıtık bilgi işleme kapasitesi sayesinde, özellikle sistem tanımlama, kontrol ve filtre uygulamaları için uygundur. Nöral ağlarda

geri-beslemenin yapıya dahil edilmesi, çeşitli denge noktalarına sahip bir dinamik sinir sistemini ortaya çıkarır (Gupta vd., 2004). Dinamik sinir sistemlerinin en genel özelliklerine aşağıdaki bölümde yer verilmiştir.

### 3.4.1. Dinamik sinir ağ yapısı

Bir dinamik sinir ağ yapısı sürekli zamanda şu şekilde tanımlanır:

$$\text{Durum denklemi: } \frac{dx(t)}{dt} = f(x(t), u(t), w) \quad (3.25)$$

$$\text{Çıkış denklemi: } y(t) = h(x(t), w) \quad (3.26)$$

Burada  $x \in R^n$  olacak şekilde durum vektörünü,  $u \in R^m$  olacak şekilde harici giriş vektörünü ve  $w \in R^l$  olacak şekilde nöral parametre vektörünü temsil eder. Nöral parametre vektörü, sinaptik bağlantı ağırlıklarını ve somatik operasyonel parametreleri içerir.  $f(\cdot)$ , nöral ağın yapısını temsil eden bir fonksiyondur.  $h(\cdot)$  ise, durum vektörü  $x(t)$  ile çıkış vektörü  $y(t)$  arasındaki ilişkiyi temsil eden bir fonksiyondur.

Benzer şekilde bir dinamik sinir ağ yapısının ayrık zamanda tanımlaması şu şekilde olur:

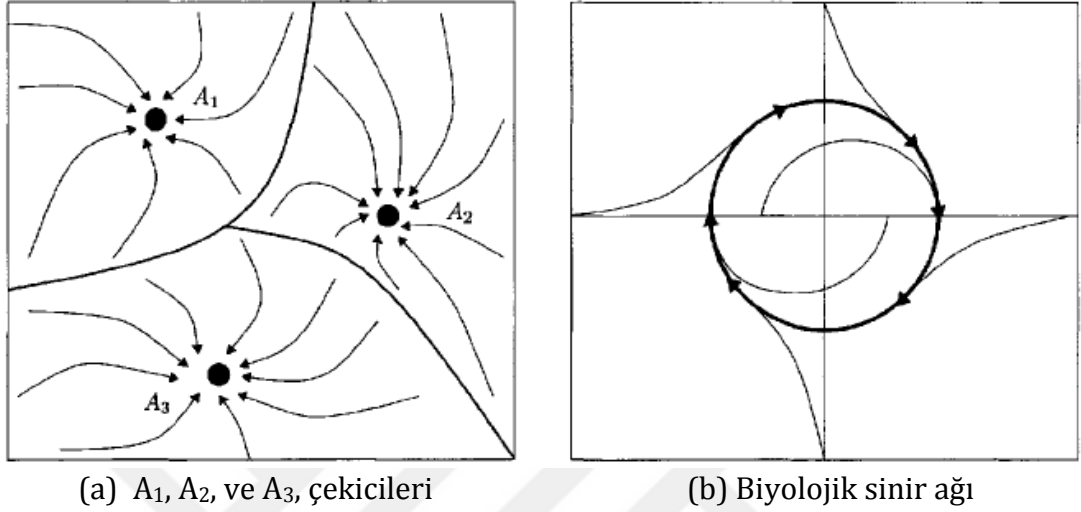
$$\text{Durum denklemi: } x(k+1) = f(x(k), u(k), w) \quad (3.27)$$

$$\text{Çıkış denklemi: } y(k) = h(x(k), w) \quad (3.28)$$

Bu dinamik sinir yapıları Pineda (1988) tarafından belirtilen üç genel karakteristiğe sahiptir (Gupta vd., 2004):

1. Dinamik sistemler genellikle çok serbestlik derecesine sahiptir.
2. Dinamik sinir ağları, doğrusal olmayan (nonlinear) yapılardır.
3. Son özellik olarak dinamik sinir ağları yutucudur. Yutucu sistemin karakteristiği şudur: Sistem geliştikçe, akış daha küçük boyuttaki bir

çekici üzerine yakınsama yapar. Genel yutucu sistemler, Şekil 3.5'te görüldüğü gibi, karmaşık davranış sergiler. Mesela, çekici (attractor) denen çok küçük boyuttaki kolektörlere veya periyodik yörüngeli tek boyutlu bir kolektöre yakınsayabilirler.



Şekil 3.5. Bir yutucu sistemin yörüngesinin 2B çizimi (Gupta vd., 2004)

Statik ileri-beslemeli sinir ağların gelişimine paralel olarak, dinamik geri-beslemeli sinir ağları ilk olarak çağrışimli bellek (associative memory) ve içeriği adreslenebilir bellek (content-addressable memory) problemleri (Hopfield, 1982; 1984; Kohonen, 1988) bağlamında örüntü tanıma için ortaya atılmıştır. Bozulmamış örüntü, kararlı denge noktası olarak ve bu örüntünün gürültülü versiyonları ise çekim noktası olarak kullanılır. Bu şekilde, bir örüntü kümesiyle ilişkilendirilmiş bir dinamik sinir sistemi oluşturulur. Eğer tüm çalışma uzayı düzgün bir şekilde böyle içeriği adreslenebilir bellek ile bölüntülenirse, o zaman örnek bir örüntüyü temsil eden herhangi bir başlangıç koşulu için bu sistem, bozulmamış örüntüye karşı düşen bir kararlı durum çözümüne sahip olmalıdır. Bir analog nöral ağın sürekli zamandaki modeli aşağıdaki doğrusal olmayan diferansiyel denklem sistemiyle tanımlanabilir.

$$\text{Durum denklemi: } \frac{dx_i(t)}{dt} = -\alpha_i(t) + \sum_{j=1}^n w_{ij}y_j(t) + s_i, \quad i=1, 2, \dots, n \quad (3.29)$$

$$\text{Çıkış denklemi: } y_i(t) = \sigma_i(x_i(t)), \quad i=1, 2, \dots, n \quad (3.30)$$

Burada  $x_i$ , i. sinirin durumunu temsil eder ve  $y_i$  de bu sinirin çıkışıdır.  $w_{ij}$ , i. ve j. sinirler arasındaki sinaptik bağlantıdır.  $s_i$ , sabit harici giriş ve  $\alpha_i$ , bir pozitif sabittir.  $\sigma_i(\cdot)$  ise monoton sigmoid bir fonksiyondur.

Şekil 3.6'da Denklem 3.29 ve 3.30 ile tanımlanan tek katmanlı bir dinamik sinir yapısı görülmektedir. Her dinamik sinir üç tip girişe sahiptir.

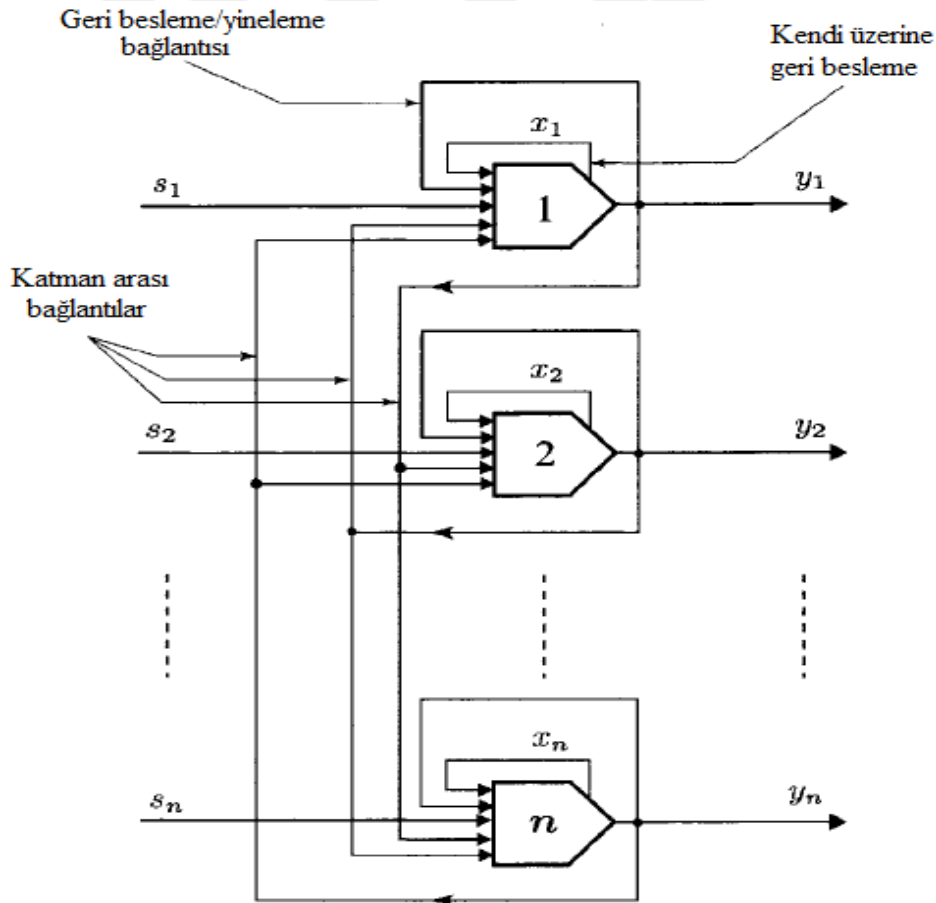
$s_i(t)$ : Dinamik işleme için harici bir giriş sinyali

$x_i(t)$ : Kendi üzerine geri-besleme bağlantısı (durum geri-besleme sinyali)

$y_i(t)$ : Katman arası bağlantılar (i. sinir dahil olmak üzere her sinirden bir çıkış sinyali)

olarak tanımlanır.

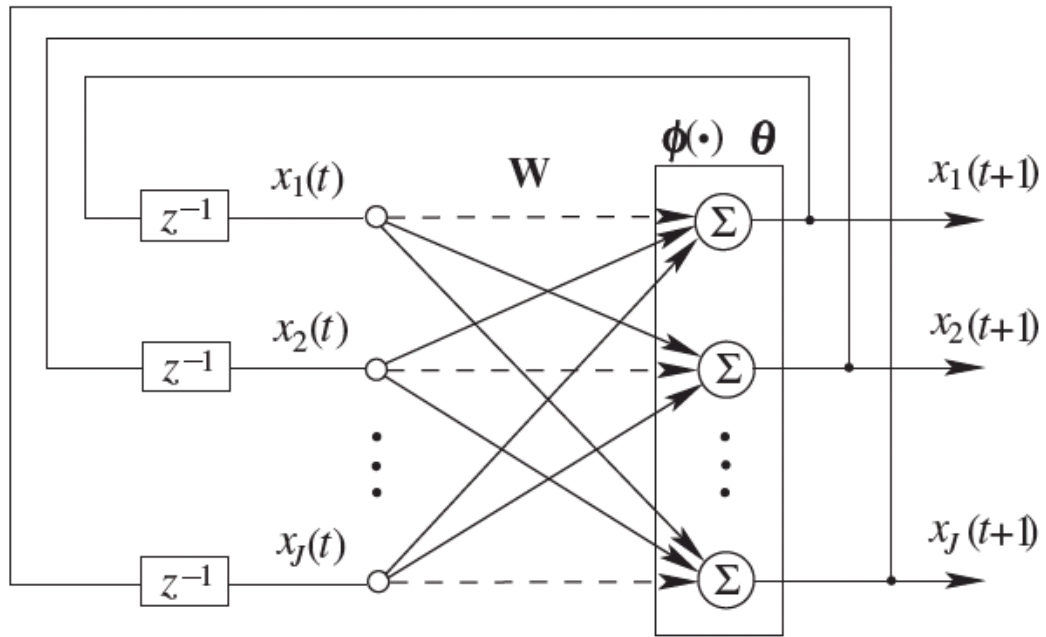
Hem yineleme, hem de katman arası bağlantıların başarısı sinaptik işlemleri gerektirir.



Şekil 3.6. Temel dinamik sinir ağ yapısı (Gupta vd., 2004)

### 3.4.2. Hopfield modeli

Hopfield modeli en popüler dinamik modeldir (Hopfield ve Tank, 1985) ve Şekil 3.6'dakine benzer bir topolojiye sahip kendi aralarında bağlı J adet McCulloch-Pitts (1943) sinirden oluşan bir yinelemeli sinir ağıdır (recurrent neural network-RNN). Hopfield modeli, Şekil 3.7'deki gibi genellikle J-J katmanlı mimari kullanılarak temsil edilir (Du ve Swamy, 2006). Giriş katmanı, sadece çıkış katmanından gelen geri-besleme sinyallerini toplar ve dağıtır. Bu ağın ağırlık matrisi sıfır-değerli bir köşegene sahip olup diğer ağırlıklar bu köşegene göre simetriktir. Yani  $w_{ij}=w_{ji}$  ve  $w_{ii} = 0$ . İkinci katmandaki her sinir, güncel net aktivasyonunu ( $net_i$ ) hesaplamak için diğer tüm sinirlerden gelen ağırlıklı girdileri toplar. Daha sonra bu  $net_i$  ağırlıklandırılmış toplam değeri bir aktivasyon fonksiyonuna sokular ve elde edilen sonuç diğer tüm sinirlere yayılır.



Şekil 3.7. Hopfield ağın mimarisi (Du ve Swamy, 2006)

Hopfield model danışmansız bir şekilde çalışır. Ağın dinamikleri doğrusal olmayan adi diferansiyel denklemler (ordinary differential equations-ODE) sistemi ile tanımlanmıştır. Bu ağın dinamikleri şu şekilde tanımlanır:



$$\text{net}_i(t+1) = \sum_{j=1}^J w_{ji} x_j(t) + \theta_i, \quad (3.31)$$

$$x_i(t+1) = \phi(\text{net}_i(t+1)). \quad (3.32)$$

Burada  $\text{net}_i$ , i. sinirin ağırlıklandırılmış net girişi;  $x_i(t)$ , sinirin çıkışı;  $\theta_i$ , sinirin eşik (bias) değeri ve  $\phi(\cdot)$  ise sigmoid fonksiyondur. Denklem 3.31 ve 3.32' deki ayrık zaman değişkeni  $t$ , sırayla 0,1,2, ... değerler alır.

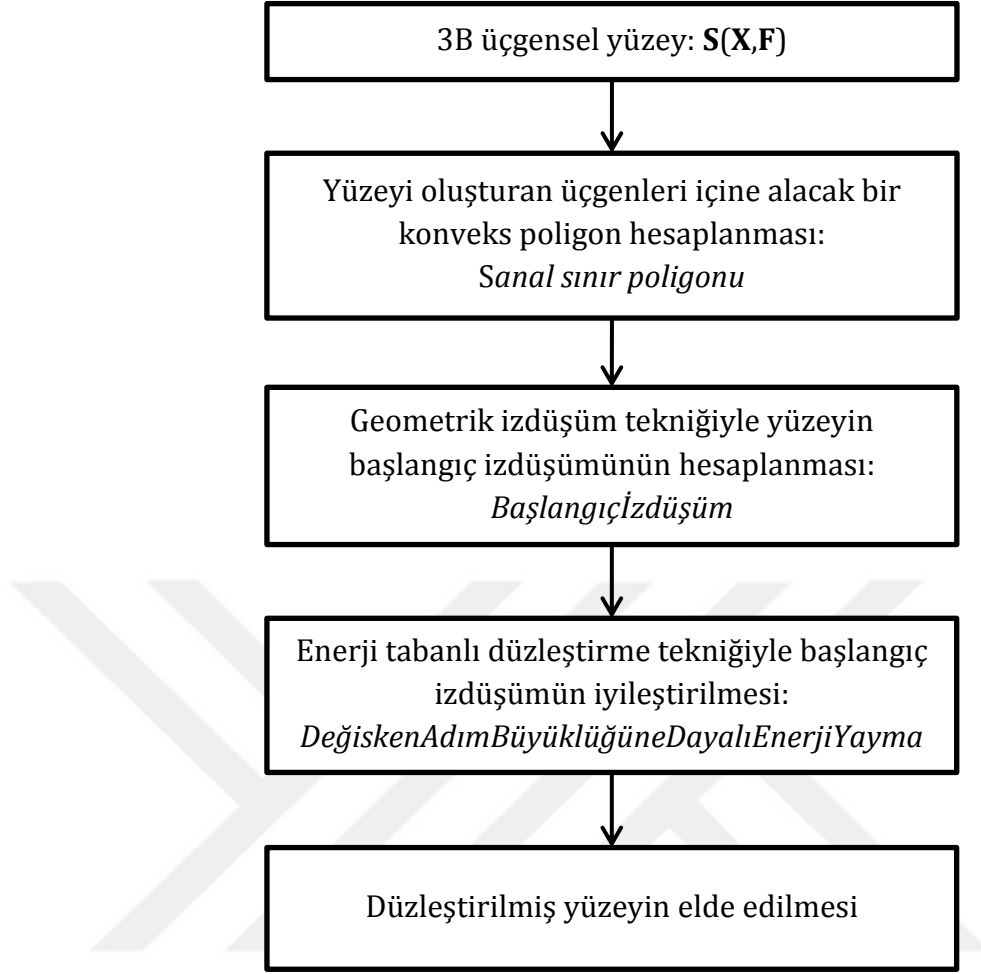
Ağın performansını karakterize etmek için enerji kavramı dâhil edilir ve aşağıdaki enerji fonksiyonu tanımlanır (Hopfield, 1982):

$$E = -\frac{1}{2} \sum_{i=1}^J \sum_{j=1}^J w_{ij} x_i x_j - \sum_{i=1}^J \theta_i x_i = -\frac{1}{2} \mathbf{x}^T \mathbf{W} \mathbf{x} - \mathbf{x}^T \boldsymbol{\theta}. \quad (3.33)$$

Burada  $\mathbf{x} = (x_1, x_2, \dots, x_J)^T$  giriş ve durum vektörü,  $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_J)^T$  ise sapma vektörüdür.

### 3.5. Önerilen Yüzey Düzleştirme Yaklaşımı

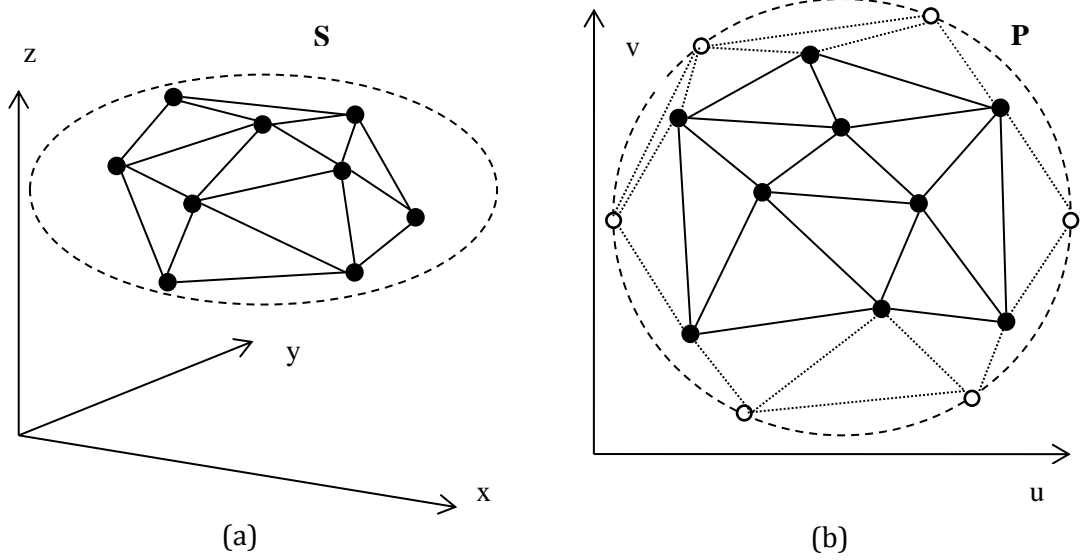
Bu çalışmanın amacı, konveks 3B yüzey yamalarının 2B karşılıklarının bulunmasıdır. Bu amaçla, yaygın olarak kullanılan geometrik ve mekanik düzleştirme yöntemlerinin birleşimi olan hibrit düzleştirme yaklaşımı benimsenmiştir. Böylece açı ve uzunluk bilgisi gibi geometrik boyutların korunumu sağlanacaktır. 3B'den 2B'ye izdüşüm sağlayan yöntemin verimliliği, düzleştirilmiş yüzeyin elde edildiği orijinal yüzeyi karşılama oranına bağlıdır. Önerilen yüzey düzleştirme yaklaşımının blok şeması Şekil 3.8'de verilmiştir.



Şekil 3.8. Yüzey düzleştirme sürecinin akış diyagramı

### 3.5.1. Sanal sınır poligonu belirleme

Konveks kombinasyon yaklaşımında (yani barisentrik izdüşümde), bir yüzey yamasının gerçek sınır düğümlerinin doğrudan bir çokgen üzerine sabitlenmesi, 2B parametrik alanın sınır yakınındaki üçgenlerin aşırı deforme ve çarpık olmasına neden olur (Lee vd., 2002). Bu nedenle 3B yüzey yamasının gerçek düğümlerini içine alan bir dizi sanal düğüm, dinamik bir şekilde oluşturulur. Hesaplama kolaylığı için sanal düğümler, bir çemberin yörüngesine yerleştirilir ve bu çemberin çapı, 3B yüzey yamasındaki üçgenlerin toplam alanına göre belirlenir. Bir yüzey yaması için oluşturulan sanal sınır yörüngesinin 3B ve 2B uzaydaki görünümleri Şekil 3.9'da verilmiştir.



Şekil 3.9. 3B (a) ve 2B (b) uzaydaki sanal sınır yörüngeleri

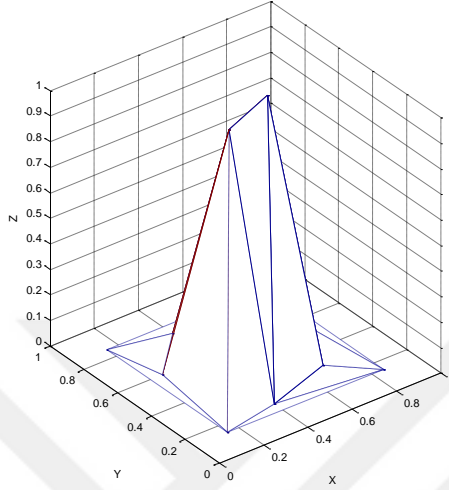
Dinamik sanal sınır düğümlerine ev sahipliği yapan çember aşağıdaki formüle göre hesaplanır:

$$r_v = \frac{\sum_{i=0}^{n-1} A_i}{\pi} \quad (3.34)$$

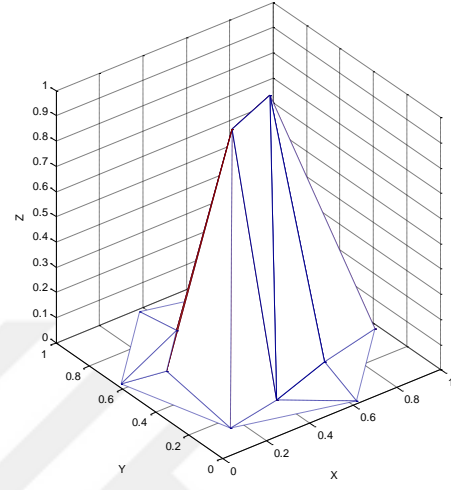
Burada  $A_i$ , yüzeycik modelindeki  $i$ . üçgenin alanı,  $n$  ise modeldeki üçgen sayısıdır. Barisentrik izdüşüm yöntemindeki sınır poligonu başlangıçta 3B yüzey yamasının düzlemsel eşdeğerinin sınır düğümlerini sabitlediğinden, bu sınır düğümleri düzlemsel gelişim süreci boyunca sabit kalır. Bundan dolayı, sınır poligonun belirlenmesi, geliştirilen düzlemsel yüzeyin orijinal 3B eşdeğerine yüksek oranda sadık kalması için kritik öneme sahiptir.

Çemberin yörüngesine yerleştirilecek sanal sınır düğümlerinin sayısı, geliştirilen 2B yüzeyin orijinal 3B yüzey yamasının şekline benzerliğini etkiler. Bu nedenle sanal sınır düğümlerinin sayısı yörüngede makul bir çözünürlük sağlayacak kadar fazla olmalıdır. Bu çözünürlük 3B yüzey yamasının toplam düğüm sayısı ile orantılı olarak belirlenir. Sanal sınır düğümlerinin sayısının belirlenmesinin başlangıçtaki düzleştirme sonucunu nasıl etkilediği, dördüncü bölümde çeşitli çözünürlük değerleri için incelenmiştir. Belirli bir yüzey yaması

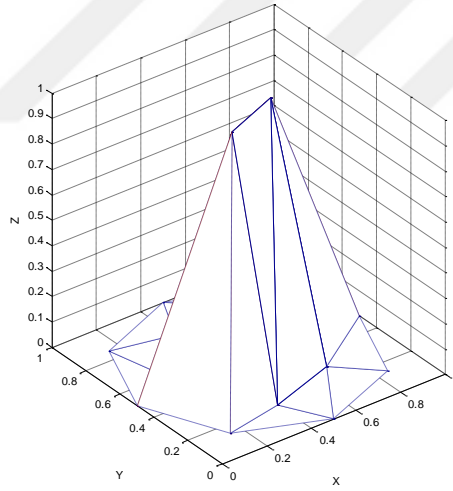
için çeşitli çözünürlüklerde üretilen sanal sınır poligonları Şekil 3.10'da verilmiştir. Şekilde (a)'dan (d)'ye kadar olan grafiklerde bir yüzey yaması için sınır düğümü sayısı  $n_v=4$ ,  $n_v=6$ ,  $n_v=8$  ve  $n_v=16$  olacak şekilde üretilen sanal sınır poligonları gözükmektedir.



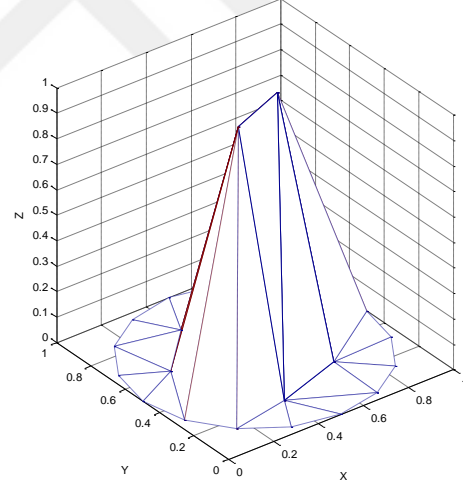
(a) 4



(b) 6



(c) 8



(d) 16

Şekil 3.10. Çeşitli çözünürlükte oluşturulan sanal sınır poligonları: Sınır düğümü sayısı (a)  $n_v=4$ ; (b)  $n_v=6$ ; (c)  $n_v=8$ ; (d)  $n_v=16$

Sınır düğümler, aşağıdaki denklemle belirlenen  $\alpha_v$  eşit açı aralıklarında olacak şekilde  $r_v$  yarıçaplı sanal sınır dairesinin yörüngesinde konumlandırılır.

$$\alpha_v = \frac{360^\circ}{n_v}. \quad (3.35)$$

### 3.5.2. Başlangıç izdüşüm algoritması

3B yüzeyin 2B parametre uzayındaki düzlemsel karşılığını bulmak için gerçekleştirilen başlangıç izdüşüm algoritması, Bölüm 3.2'de matematiksel altyapısı verilen Floater'ın ağırlık merkezli izdüşüm (barycentric mapping) yöntemine dayandırılmaktadır. Geçekleştirilen sanal sınır (virtual boundary) ekleme yaklaşımı (Lee vd., 2002) sayesinde 3B yüzeyin gerçek sınır düğümlerinin 2B parametre uzayında her seferinde sabit bir poligona izdüşürülmesi engellenir ve verilen karmaşık yüzeyin tamamını içine alacak yarıçapta bir sanal sınır çemberi ve poligon için içerisine sokulmaktadır. Bu yolla izdüşüm esnasında genellikle sınırlarda daha fazla deformasyona uğrayan üçgenlerin deformasyon miktarını azaltmak hedeflenmektedir. Bu şekilde başlangıç izdüşüm algoritmasının üreteceği izdüşüm sonucunun doğruluğu daha da iyileştirilmiş olur. Önerilen sanal sınır poligonlu başlangıç izdüşüm yaklaşımı Algoritma-2'de verilmiştir.

Genel olarak, önerilen algoritma  $\mathbf{X}$  ve  $\mathbf{F}$  olmak üzere iki argüman alır ve çıktı olarak  $\mathbf{S}(\mathbf{X}, \mathbf{F})$  yüzeyinin başlangıç düzlemsel karşılığı olan  $\mathbf{P}$ 'yi üretir.  $\mathbf{P}$ , u-v koordinat sistemindeki n adet düğümden, yani düzlemsel koordinatlardan oluşur.

Tutte (1963) tarafından kullanılan  $\lambda_{i,j}=1/d_i$  ifadesi, Algoritma-2'de 13. ve 14. adımlardaki ifadelerle değiştirilmiştir. Bu adımlar şunu ifade eder: 3B uzaydaki herbir komşu köşe arasındaki Öklid mesafesinin çarpmaya göre tersi 13. adım ile hesaplanır. 14. Adımda ise  $\lambda_{i,j}$ ,  $d_{ij}$ 'nin normalleştirilmiş halinin merkez köşenin komşularının toplam maliyetine bölümüyle hesaplanır. Bu yolla, üçgenlenmiş yüzeyin şeklinin, parametrik uzayda yerel olarak korunması hedeflenir. Bu durum Bölüm 4.1.3'te testlerle somut olarak gösterilmiştir.

---

## Algoritma-2 Başlangıç İzduşüm

---

**Input:** 3B uzayda verilen bir üçgensel kafes yüzeyi  $\mathbf{S}(\mathbf{X}, \mathbf{F})$ . Burada  $\mathbf{X}$ , 3B uzaydaki köşe koordinatlarını ve  $\mathbf{F}$  ise üçgenleri tanımlayan köşe indislerini tutar.

**Output:**  $\mathbf{S}$  yüzeyinin başlangıç 2B izduşüm sonucu  $\mathbf{P}$ .

- 1: Sanal sınır yörüngesini belirleyen çemberin  $r_v$  yarıçapı Denklem 3.34 kullanılarak hesaplanır.
  - 2: Orijinal yüzeydeki düğüm sayısına göre sanal sınır yörüngesindeki toplam  $n_v$  düğüm sayısı belirlenir.
  - 3: Sanal düğümlerin arasındaki adım  $\alpha_v$  açısı, Denklem 3.35 kullanılarak hesaplanır.
  - 4:  $z$  bileşenleri sıfır olacak şekilde sanal düğümlerin konumları belirlenir.
  - 6: Sanal düğümler  $\mathbf{S}(\mathbf{X}, \mathbf{F})$  orijinal yüzeye eklenerek  $\mathbf{S}^*(\mathbf{X}, \mathbf{F})$  ilaveli 3B yüzeyi elde edilir.
  - 7:  $\mathbf{S}^*(\mathbf{X}, \mathbf{F})$  ilaveli 3B kafes yüzeyi analiz edilerek sınır düğümler  $(n+1, \dots, N)$  tespit edilir.
  - 8:  $(N \times 3)$  boyutlu  $\mathbf{X}$  matrisinde sıralama yapılarak sınır düğümlerin iç düğümlerden sonra gelmesi sağlanır.
  - 9: Verilen karmaşık yüzeyin tamamını içine alacak ( $K$  adet köşeden oluşan) sanal sınır poligonu belirlenir. (Bölüm 3.5.1'e bakınız.)
  - 10:  $\mathbf{u}_{n+1}, \dots, \mathbf{u}_N$   $K$ -kenarlı ( $D \subset \mathbb{R}^2$ ) konveks poligonun köşelerinde olacak şekilde saat yönünün tersinde seçilir.
  - 11: **for i from 1 to n do**
  - 12:     **for j from 1 to N do**
  - 13:          $d_{i,j} = 1 / \|\mathbf{x}_{i,i} - \mathbf{x}_{i,j}\|$
  - 14:         Denklem 3.2 ile verilen kurala uyarak  $\lambda_{i,j} = d_{i,j} / \sum_{j=1}^N (d_{i,j})$  atamaları yapılır.
  - 15:     **end for**
  - 16: **end for**
  - 17: Ağırlık değerleri  $\lambda$  matrisine eşitlenerek bir ( $K$  harici girişli ve  $n$  çıkışlı) dinamik sinir ağı oluşturulur. Dinamik ağın  $n$  adet çıkışı, girişine geri-beslenir. Bunlar dahili girişler veya durumlar olarak adlandırılır.
  - 18: Dinamik sinir ağının başlangıç koşulları (initial states)  $\mathbf{S}^*(\mathbf{X}, \mathbf{F})$  yüzeyin  $x$ - $y$  koordinatlarına ( $z$  değerleri kullanılmadan) eşitlenir.
  - 19: Başlangıç koşulları bu değerlere ayarlanan dinamik ağın ürettiği değerler belirlenen eşiğin altında değişim gösterene kadar bu ağ çalıştırılır. (Convergence)
-

Bu çalışmada Bölüm 3.2'deki Denklem 3.6 ve 3.7'yi çözmek amacıyla yenilikçi bir şekilde bir dinamik sinir ağı modeli oluşturulmuştur. Bu denklemleri çözen sinir ağı modeli Algoritma-2'deki 17., 18. ve 19. adımlara karşı düşer. Bu sinir ağı modeli, Bölüm 3.2'nin sonunda bahsedilen ters matrisin hesaplanmasında ihtiyaç duyulan lineer denklem sisteminin çözümünün yerine geçer. Literatürdeki var olan daha önceki çalışmalar göz önünde bulundurulduğunda, dinamik sinir ağı modeli ve barisentrik izdüşüm teorisini kullanarak parametreleme problemine çözüm üreten bu türden başka bir çalışmaya rastlanmamıştır.

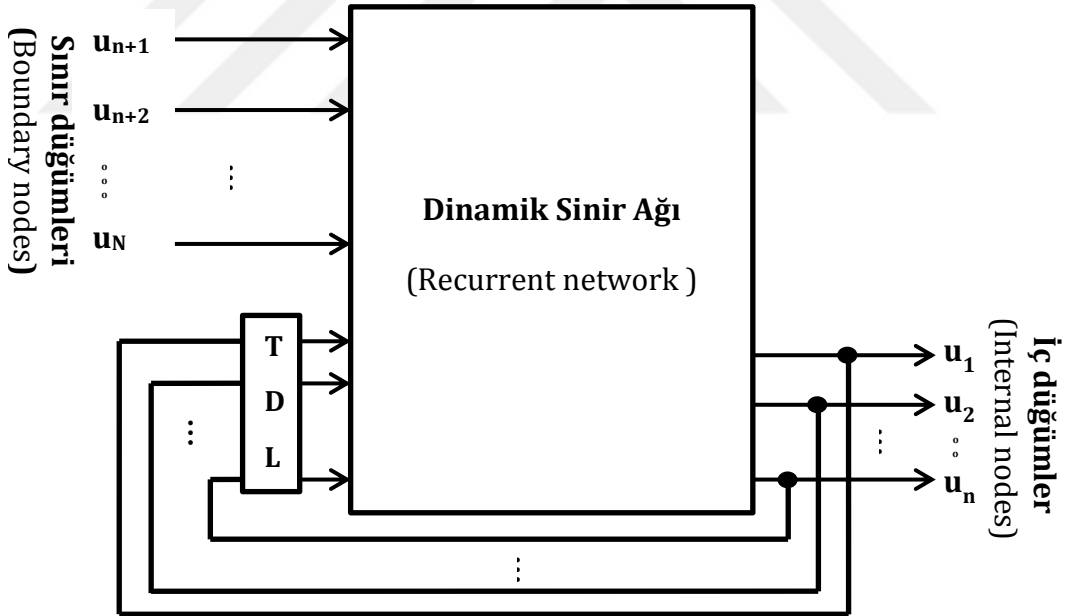
Çalışmada iç düğümlerin konumlarını etkili bir şekilde hesaplamak için benimsenen yinelemeli ağ mimarisi, her bir düğümün komşularının konveks birleşimi olarak ele alındığı barisentrik izdüşüm teorisine oldukça uyumludur. Önerilen model tüm çıkışları, parametreleştirilecek yüzey örgüsünün topolojisine göre belirlenen ilgili ağırlıklarla girişlere geri beslenmesini sağlayarak hesaplama zamanı bakımından ve kötü koşullu durumlarda etkili olan tekrarlı bir hesaplama yapısı sunar.

### **3.5.2.1. Dinamik sinir ağı modeli**

Dinamik sinir ağlarında, çıkış sadece o anda ağa uygulanan girişlere bağlı değildir. Bu ağlarda çıkışlar; aynı zamanda mevcut ve önceki girişlere, çıkışlara veya durumlara da bağlıdır. Bu ağların geri besleme bağlantıları vardır ve bunun anlamı şudur: Bu ağlarda o anki çıkış, geçmiş çıkışların bir fonksiyonudur (Hagan vd., 2014). Dinamik ağlar, gecikmeler içeren ve bir girdi dizisi üzerinde çalışan bir ağıdır. Başka bir deyişle, giriş dizisinin sırası ağın çalışması için çok önemlidir.

Dinamik ağlar hafızaya sahiptir, yani geçmişini hatırlar. Bu nedenle ağın herhangi bir andaki cevabı, yalnızca o anki girişlere değil aynı zamanda girdi sırasının geçmişine de bağlıdır (Hagan vd., 2014).

3B yüzeyin üçgenlenmiş düzlemsel biçimine ait düğümlerinin başlangıç koordinatlarını hesaplayan dinamik sınır ağı modeli Şekil 3.11'de verilmektedir. Şekilden anlaşılacağı üzere ağın girişindeki statik girişler, sınır düğümlerinin koordinatlarına karşı düşmektedir. Ağın çıkışları ise iç düğümlerin koordinatlarıdır ve bunlar dinamik sınır ağıyla yinelemeli bir yapıda hesaplanmaktadır. İç düğümlerin izdüşüm sürecinin en başındaki başlangıç koordinatları, orijinal 3B üçgensel kafesin köşelerinin x ve y bileşenlerine eşitlenerek düğümlerin K-köşeli sınır poligonunun içinde kalması sağlanır. Başlangıçta iç düğümlerin koordinatlarını orijinal 3B üçgensel kafesin x ve y koordinatlarına eşitleme işlemi paralel projeksiyon anlamına gelir. Önerilen dinamik ağ modelinde, iç düğümlerin koordinatları olan çıkışlar, Denklem 3.3'ü (Floater, 1997) sağlayan benzersiz bir çözüme yakınsaması için ağın girişlerine geri beslenir, Şekil 3.11. Ağ yinelemesini sonlandırma kriteri, iç düğümlerin konumlarında kayda değer bir değişiklik olmadığı durumdur.



Şekil 3.11. Önerilen dinamik sınır ağı modeli

Ağın başlangıç koşullarını, paralel projeksiyon ile orijinal 3B yüzey yamasının köşelerinin x ve y bileşenlerine eşitlemek, yakınsama sürecine kayda değer bir ivme kazandırır.



Dinamik ağlarda, genellikle bağlantıların bir kısmı tekrarlanan bir gecikme hattı (tapped-delay line-TDL) vasıtasıyla girişe geri beslenir (Hagan vd., 2014). Şekil 3.11'den görülebileceği gibi, iç düğümlerle ilgili sistem çıkışları, önerilen modelde bir-birimlik geciktirme hattı aracılığıyla girişe geri beslenir.

### 3.5.2.2. Yakınsama araştırması

Önerilen yöntemde, daha önce belirtildiği gibi, 3B yüzeyin düzlemsel koordinatlarının bulunması (n. dereceden) lineer (zamanla değişmeyen) cebirsel denklemlerin çözülmesini gerektirir.

$$\mathbf{Ax} = \mathbf{b}. \quad (3.36)$$

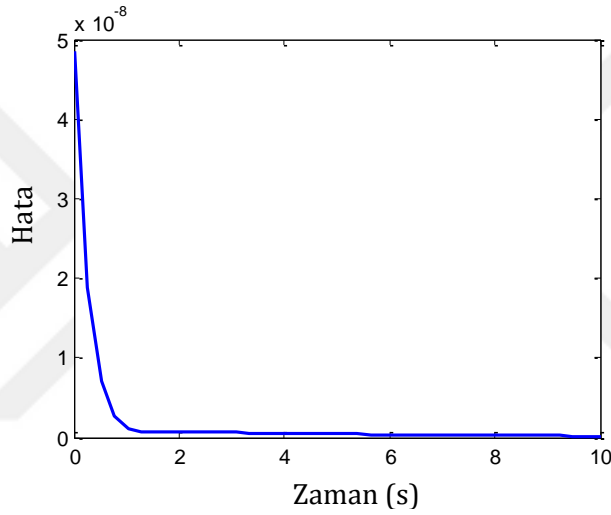
Burada,  $\mathbf{A}=[a_{ij}] \in \mathbb{R}^{n \times n}$  katsayı matrisi ve  $\mathbf{b}=[b_1, b_2, \dots, b_n]^T \in \mathbb{R}^n$  vektörü genellikle bilinen değişkenlerdir.  $\mathbf{x} \in \mathbb{R}^n$  ise çözülecek olan, yani bilinmeyen vektördür.  $\mathbf{A}$ 'nın sabit tekil olmayan (non-singular) katsayı matrisi olduğu varsayıldığında; eğer monoton olarak artan tek aktivasyon fonksiyonu kullanılırsa, o zaman dinamik sinir ağının  $\mathbf{x}(t)$  sinirsel (neural) durumu,  $\mathbf{x}(0) \in \mathbb{R}^n$  gibi herhangi bir başlangıç durumundan başlayarak lineer denklemin eşsiz çözümüne yakınsayacaktır (Madankan, 2010; Zhang vd., 2002; Zhang ve Ge, 2005 ).

Önerilen yöntemin yakınsama araştırması için iyi bilinen ve son derece kötü koşullu bir matris kullanılır. Örneğin  $\mathbf{A}$ 'nın bir Hilbert matris olması durumu düşünülebilir (Madankan, 2010; Zhang vd., 2002; Zhang ve Ge, 2005; Yu vd., 1992):

$$\mathbf{A} = \begin{bmatrix} 1 & \frac{1}{2} & \dots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \dots & \frac{1}{n+1} \\ \dots & \dots & \dots & \dots \\ \frac{1}{n} & \frac{1}{n+1} & \dots & \frac{1}{2n-1} \end{bmatrix} \quad (3.37)$$

$b_i = \sum_{j=1}^n e^{1/j} A(i,j)$  olacak şekilde  $\mathbf{b}=[b_1, b_2, \dots, b_n]^T$  seçilir (burada  $e=2.718281828$ 'dir). Açık bir şekilde teorik çözüm  $\mathbf{x}=(e, e^{1/2}, \dots, e^{1/n})^T$

sonucunu verir. Maalesef, bahsi geçen özel durum için  $n > 10$  iken Gauss-Jordan eliminasyonu veya LU faktörizasyon gibi normal doğrudan yöntemle kesin sayısal çözümü elde etmek imkansızdır (Zhang ve Ge, 2005; Yu vd., 1992). Aksine, Şekil 3.11’de görülen dinamik sinir ağın  $x(t)$  nöral durumu Denklem 3.37’nin eşsiz çözümüne yakınsayabilir. Bu durum  $n=15$  için Şekil 3.12’deki grafikte açıkça gösterilmiştir. Bu grafik, sistemin yakınsama davranışını gösteren iki-normlu kalıntı hatası (two norm residual error) barındırmaktadır. Şekildeki grafikten kolayca anlaşılacağı gibi, bu hata değeri sıfıra yakınsamaktadır.



Şekil 3.12. İki-norm kalıntı hatası

Eğer Denklem 3.37’deki kötü koşullu matris ile Denklem 3.36’yı çözmek için klasik Gauss-Jordan eliminasyon yöntemi kullanılırsa, önemli bir problem ortaya çıkar. Bu önemli problem şu şekilde açıklanabilir.  $\mathbf{b}$  vektörünün sensörler tarafından toplanan veriyi temsil ettiği ve  $\Delta \mathbf{b}$  gibi bir hata ile geldiği göz önünde bulundurulsun. Bu durumda probleme ilişkin geleneksel çözüm şöyle olurdu:

$$\mathbf{x} = \mathbf{A}^{-1}(\mathbf{b} + \Delta \mathbf{b}) = \mathbf{A}^{-1}\mathbf{b} + \mathbf{A}^{-1}\Delta \mathbf{b} = \mathbf{x}^* + \mathbf{A}^{-1}\Delta \mathbf{b}. \quad (3.38)$$

Burada  $\mathbf{x}^*$  gerçek çözümdür. Verinin eldesinden kaynaklanan hatasının sebep olduğu klasik çözüm hatası şu halde ifade edilir:

$$\Delta \mathbf{x} = \mathbf{x} - \mathbf{x}^* = \mathbf{A}^{-1} \Delta \mathbf{b}. \quad (3.39)$$

Bu ifadeden anlaşılacağı üzere,  $\Delta \mathbf{b}$  hata değeri  $\mathbf{A}^{-1}$  ile kuvvetlendirilerek  $\mathbf{x}$  değerinde çok büyük bir değişime sebep olabilir. Burada anlaşılmayı kolaylaştırmak adına  $\mathbf{A}$  değeri bir skaler olarak düşünülebilir. Eğer  $A$  çok küçük olursa, o zaman  $A$ 'nın tersi çok büyük olur. Dolayısıyla bu basitleştirmeden yola çıkılırsa, verideki küçük bir hata (örneğin yuvarlama hatası) bile  $A$ 'nın çok büyük olan tersi ile kuvvetlendirileceğinden çözümde devasa sapmalara yol açabilir.

### 3.5.3. Değişken adım büyüklüğüne dayalı enerji yayma algoritması

Enerji yayma algoritması nihai düzleştirilmiş yüzeyin elde edilmesinde büyük önem arz etmektedir. Bu algoritma düzleştirme doğruluğu ve verimliliğinde belirleyici rol oynamaktadır. Önceki bölümde ağırlık merkezli izdüşüm tekniğine dayalı başlangıç izdüşüm sonucu elde edildikten sonra bu sonuç, enerji yayma algoritması ile enerji boşaltım sürecine sokulur. Bu süreç izin verilen  $\emptyset_S$  alan doğruluğu ve  $\emptyset_C$  kenar doğruluğu şartları sağlanana kadar sürdürülebilir:

$$E_S \leq \emptyset_S, \quad (3.40)$$

$$E_C \leq \emptyset_C. \quad (3.41)$$

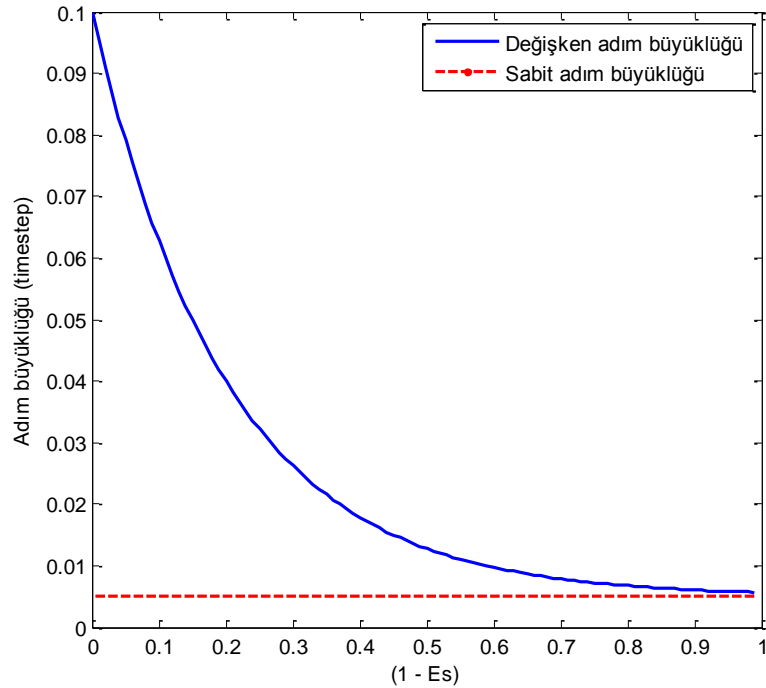
Bölüm 3.3'te matematiksel alt yapısı verilen enerji tabanlı düzleştirme tekniği benimsenerek Wang vd. (2002) tarafından ileri sürülen enerji boşaltım algoritması üzerinde bazı geliştirmeler/modifikasyonlar yapılmıştır. Bu geliştirmelerin başında sabit adım büyüklüğüne dayalı olarak Euler yöntemi ile hesaplanan bazı denklemlerdeki (Denklemler 3.16-3.18)  $\Delta t$  adım büyüklüğü, değişken adım büyüklüğüne dayalı bir yapıya dönüştürülmüştür. Değişken adım büyüklüğü, enerji boşaltım sürecinde geliştirilen 2B uzaydaki yüzeye ait değişime uğrayan alanların ve kenarların toplamının, orijinal yüzeydeki karşılıklarından olan farklarından türetilen bir fonksiyon tarafından

belirlenmiştir. Benimsenen değişken adım büyüklüğü fonksiyonunun ifadesi aşağıdaki gibidir:

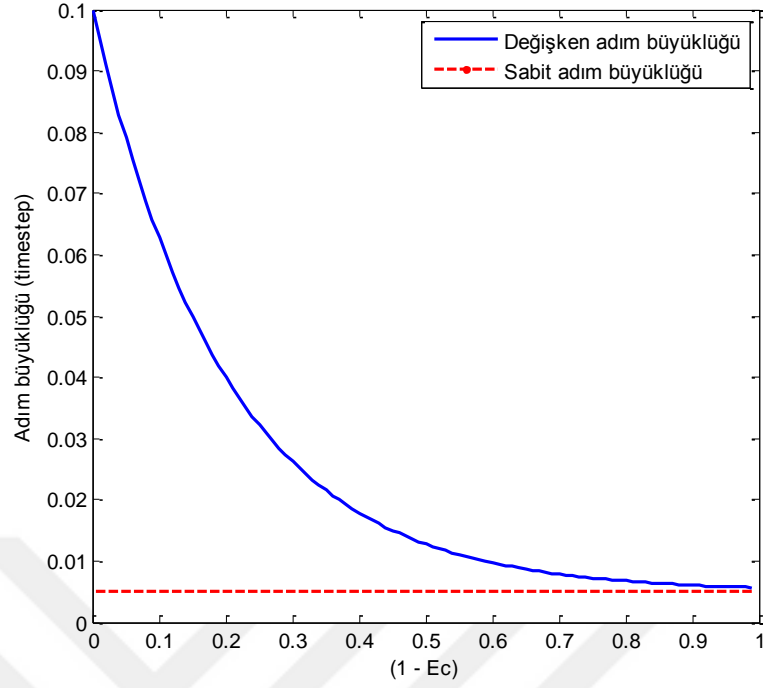
$$\Delta t_1 = 0.005 + 0.1e^{-5(1-E_s)}, \quad (3.42)$$

$$\Delta t_2 = 0.005 + 0.1e^{-5(1-E_c)}. \quad (3.43)$$

Bu üstel fonksiyonlardan birisi alan-doğruluğu yüzdesine, diğeri de kenar-doğruluğu yüzdesine bağlı olarak değişkenlik gösteren değerler üretir. Bu fonksiyonların grafikleri çizdirildiğinde ortaya çıkan adım büyüklüğünün değişimi Şekil 3.13 ve 3.14'te verilmiştir. Grafiklerde yatay eksenle sırasıyla alan-doğruluğunun ve kenar-doğruluğunun 1'den farkı vardır. Düşey eksenlerde ise adım büyüklüğü değerleri yer almaktadır. Denklem 3.42 ve 3.43'te verilen üstel ifadeler, büyük alan/kenar doğruluk yüzdesi farkları için büyük adım değeri üretmektedir. Doğruluk farkı değeri küçüldükçe bu fonksiyonun ürettiği adım büyüklüğü değeri de üstel olarak azalmaktadır. Grafikten de görüldüğü üzere fonksiyonun ürettiği değer, ofset değeri sayesinde doğruluk farkı değeri ne kadar küçük olursa olsun 0.005'in üstünde kalmaktadır.



Şekil 3.13. Değişken ve sabit adım büyüklüğü ile alan doğruluğunun değişimi

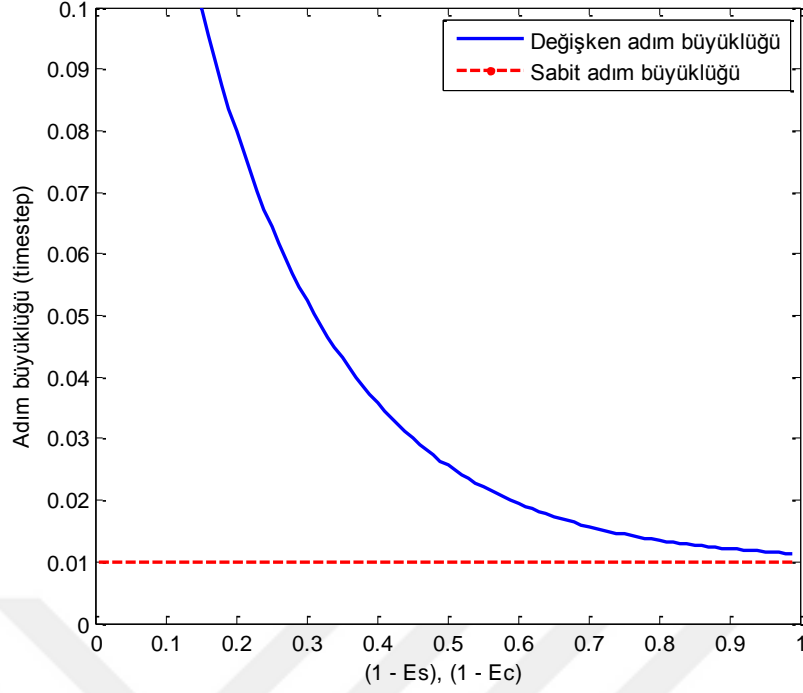


Şekil 3.14. Değişken ve sabit adım büyüklüğü ile kenar doğruluğunun değişimi

Bahsedilen iki fonksiyonun ifadesi toplanarak nihai  $\Delta t$  değişken adım büyüklüğü hesaplanmıştır:

$$\Delta t = \Delta t_1 + \Delta t_2. \quad (3.44)$$

Bu ifadede alan ve kenar doğruluğuna dayalı iki fonksiyondaki sabit terimler doğrudan toplandığından nihai adım büyüklüğünün 0.01 değerinin altına düşmemesi garanti edilmiştir. Bu durum, Şekil 3.15'teki grafikten net bir şekilde görülmektedir.



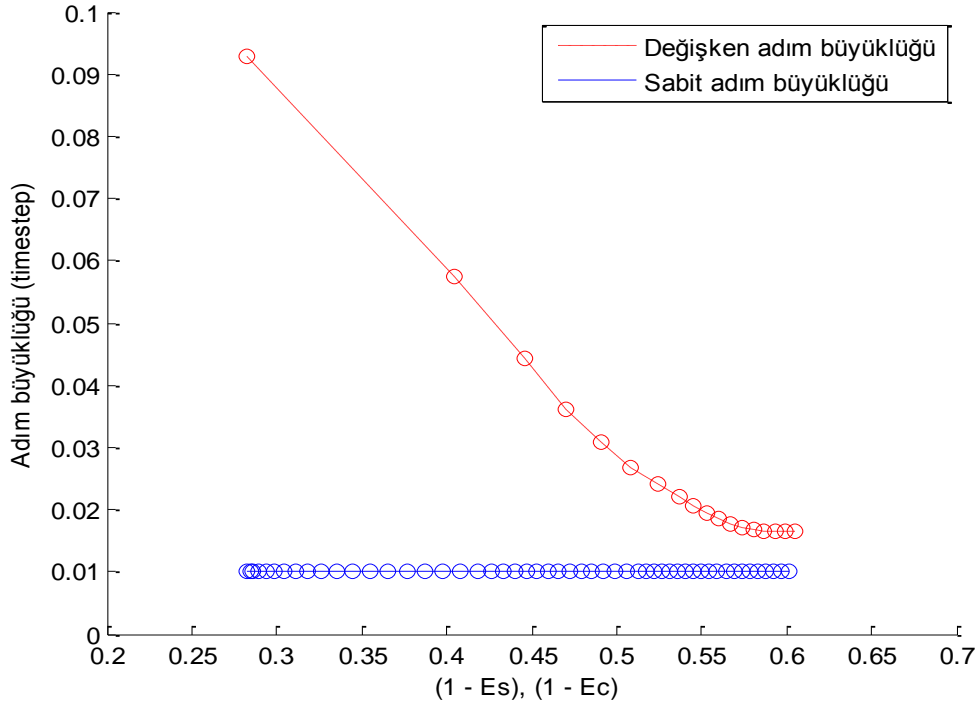
Şekil 3.15. Nihai değişken adım büyüklüğünün alan ve kenar doğruluklarına göre değişimi

Euler yöntemiyle çözülen 3.16-3.18 denklemlerindeki  $\Delta t$  adım büyüklüğü, Denklem 3.44 yardımıyla hesaplanır. Bu sayede sabit adım büyüklüğü ile ilerlendiği durumdaki enerji yayma sürecinde gereken iterasyon sayısının azaltılması hedeflenir. Belirli bir yüzey yaması için enerji yayma sürecinin sabit adımlı veya değişken adımlı olması durumunda gereken iterasyon sayısı ve buna bağlı olarak ortaya çıkan nihai göreceli alan ( $E_S$ ) ve ( $E_C$ ) kenar-doğruluk farkları Çizelge 3.1'de verilmiştir. Çizelgeden görüldüğü gibi, belirli bir yüzey yaması için izin verilen alan doğruluğu  $\phi_S=0.4$  ve kenar doğruluğu  $\phi_C=0.4$  eşiklerinin altında kalması şartlarını sağlayacak şekilde enerji yayma süreci sabit adım büyüklüğü ( $\Delta t=0.01$ ) ile işletildiğinde 53 iterasyonda istenen doğruluk oranları sağlanmıştır. Enerji yayma süreci aynı koşulları sağlayacak şekilde değişken adım büyüklüğüne dayalı olarak işletildiğinde ise 18 iterasyonda yaklaşık aynı sonuca ulaşılmaktadır. Bu süreç, değişken adım büyüklüğüne dayalı olarak işletildiğinde bahsedilen Euler yöntemiyle çözülen denklemlerdeki  $\Delta t$  adım büyüklüğünün aldığı değerlerin grafiği Şekil 3.16'da kırmızı renkle çizilmiştir.

Çizelge 3.1. Sabit ve değişken adım büyüklüğüne dayalı enerji yayma sürecindeki iterasyon sayıları kıyaslaması

Sabit adım büyüklüğü ( $\Delta t=0.01$ )		Değişken adım büyüklüğü	
İterasyon sayısı	Alan ve Kenar doğruluk farkı	İterasyon sayısı	Alan ve Kenar doğruluk farkı
1	$E_S=0.8928, E_C=0.7174$	1	$E_S=0.8928, E_C=0.7174$
2	$E_S=0.8926, E_C=0.7172$	2	$E_S=0.7858, E_C=0.5958$
3	$E_S=0.8917, E_C=0.7159$	3	$E_S=0.7101, E_C=0.5537$
4	$E_S=0.8900, E_C=0.7137$	4	$E_S=0.6415, E_C=0.5294$
5	$E_S=0.8877, E_C=0.7105$	5	$E_S=0.5781, E_C=0.5090$
6	$E_S=0.8847, E_C=0.7064$	6	$E_S=0.5191, E_C=0.4913$
7	$E_S=0.8809, E_C=0.7014$	7	$E_S=0.4636, E_C=0.4757$
8	$E_S=0.8765, E_C=0.6956$	8	$E_S=0.4106, E_C=0.4633$
9	$E_S=0.8713, E_C=0.6889$	9	$E_S=0.3596, E_C=0.4548$
10	$E_S=0.8655, E_C=0.6815$	10	$E_S=0.3098, E_C=0.4470$
11	$E_S=0.8589, E_C=0.6733$	11	$E_S=0.2609, E_C=0.4396$
12	$E_S=0.8515, E_C=0.6645$	12	$E_S=0.2126, E_C=0.4325$
13	$E_S=0.8435, E_C=0.6550$	13	$E_S=0.1647, E_C=0.4258$
14	$E_S=0.8347, E_C=0.6449$	14	$E_S=0.1466, E_C=0.4194$
15	$E_S=0.8251, E_C=0.6342$	15	$E_S=0.1324, E_C=0.4131$
16	$E_S=0.8148, E_C=0.6230$	16	$E_S=0.1393, E_C=0.4070$
17	$E_S=0.8038, E_C=0.6128$	17	$E_S=0.1571, E_C=0.4009$
18	$E_S=0.7920, E_C=0.6024$	<b>18</b>	<b><math>E_S=0.1756, E_C=0.3949</math></b>
19	$E_S=0.7794, E_C=0.5917$		
20	$E_S=0.7661, E_C=0.5821$		
21	$E_S=0.7520, E_C=0.5737$		
22	$E_S=0.7371, E_C=0.5663$		
23	$E_S=0.7215, E_C=0.5593$		
24	$E_S=0.7051, E_C=0.5531$		
25	$E_S=0.6879, E_C=0.5469$		
26	$E_S=0.6700, E_C=0.5406$		
27	$E_S=0.6513, E_C=0.5342$		
28	$E_S=0.6318, E_C=0.5277$		
.	.		
.	.		
.	.		
43	$E_S=0.2485, E_C=0.4453$		
44	$E_S=0.2171, E_C=0.4406$		
45	$E_S=0.1850, E_C=0.4359$		
46	$E_S=0.1698, E_C=0.4312$		
47	$E_S=0.1586, E_C=0.4266$		
48	$E_S=0.1474, E_C=0.4219$		
49	$E_S=0.1361, E_C=0.4172$		
50	$E_S=0.1327, E_C=0.4126$		
51	$E_S=0.1441, E_C=0.4080$		
52	$E_S=0.1558, E_C=0.4034$		
<b>53</b>	<b><math>E_S=0.1678, E_C=0.3988</math></b>		

Buna göre adım büyüklüğü başlangıçta, yani doğruluk oranı farkı yüksek iken büyük değerler almaktadır. Grafikte yatay eksen, bu doğruluk oranının 1'den farkına karşı düşmektedir. Yüzey geliştirmede enerji yayma sürecinin başında geliştirilmekte olan yüzeyin alan veya kenar-doğruluk farkı genellikle yüksek olur. Bunun 1'den farkı ise, bekleneceği üzere, tam tersi küçük bir değer olur. Örneğin 1. iterasyonda  $E_C=0.7174$  iken Denklem 3.34'deki ifadeye göre adım büyüklüğünün değeri yaklaşık  $\Delta t=0.092$  olarak hesaplanmıştır. Bu adım değerini, Şekil 3.15'teki grafikte, yatay eksendeki  $(1-E_C)$  değerini, yani 0.2826 noktasını, kırmızı renkli eğri üzerindeki yuvarlak nokta temsil etmektedir. Her iterasyon sonunda geliştirilmekte olan yüzeyin doğruluk farkı azaldıkça, yani doğruluk arttıkça adım büyüklüğü miktarının giderek üstel olarak azaldığı grafikten görülmektedir. Kırmızı renkli eğri üzerindeki nokta sayısı, Çizelge 3.1'in sağ sütündeki iterasyon sayısı kadar, yani 18 adettir. Diğer taraftan mavi çizgi, sabit adım büyüklüğüne dayalı süreç ile yüzey geliştirme esnasında kenar-doğruluk farkını göstermektedir. Bu yaklaşımda süreç boyunca adım büyüklüğü 0.01 olarak sabit kaldığından, çizgi üzerindeki nokta sayısı yine çizelgenin sol sütündeki iterasyon sayısı kadar, yani 53 adettir.



Şekil 3.16. Nihai değişken adım büyüklüğünün aldığı değerler



---

**Algoritma-3** *Değişken Adım Büyüklüğüne Dayalı Enerji Yayma ( $\Omega$ )*

---

**Input:** Başlangıç izdüşümü yapılmış 2B noktalar kümesi  $\mathbf{P}=\{P_0, P_1, \dots, P_{n-1}\}$  ve düğüm sayısı  $n$ .

**Output:**  $E(\Omega)$  enerjisini minimize eden  $\mathbf{P}$  noktalarının nihai konumları.

- 1: İzin verilen  $\emptyset_s$  alan-doğruluk farkı yüzdesi belirlenir.
  - 2: İzin verilen  $\emptyset_c$  kenar-doğruluk farkı yüzdesi belirlenir.
  - 3: İzin verilen  $\epsilon$  enerji değişim yüzdesi belirlenir.
  - 4: İzin verilen maksimum iterasyon sayısı  $N_{max}$
  - 5: **for**  $i$  **from** 0 **to**  $n-1$  **do**
  - 6: **for**  $i$  **from** 0 **to**  $n-1$  **do**
  - 7:     Denklem 3.15 kullanılarak  $P_i$  düğümünün kütlesi hesaplanır.
  - 8: **end for**
  - 9: **while** ((göreceli alan-doğruluk yüzdesi  $E_s > \emptyset_s$  **or** göreceli kenar-doğruluk yüzdesi  $E_L > \emptyset_L$ ) **and**  $E(\Omega)$  değişimi  $> \epsilon$  **and** iterasyon sayısı  $< N_{max}$ )
  - 10:     **for**  $i$  **from** 0 **to**  $n-1$  **do**
  - 11:         Denklem 3.11 kullanılarak  $P_i$  düğümü üzerine etkiyen gerilme kuvveti hesaplanır.
  - 12:         Denklemler 3.16 – 3.18 kullanılarak  $P_i$  düğümünün yeni konumu hesaplanır.
  - 13:         Denklemler 3.19 ve 3.20 kullanılarak  $P_i$  düğümüne uygulanacak ceza fonksiyonu hesaplanır.
  - 14:          $P_i$  düğümünü yeni konumuna taşınır.
  - 15:     **end for**
  - 16:     Denklem 3.21 kullanılarak yeni  $E_s$  değeri hesaplanır.
  - 17:     Denklem 3.23 kullanılarak yeni  $E_c$  değeri hesaplanır.
  - 18:     Denklemler 3.10 ve 3.12 kullanılarak  $E(\Omega)$ 'nin yeni değeri hesaplanır.
  - 20:      $\Delta t_1$  değeri Denklem 3.42'ye göre hesaplanır.
  - 21:      $\Delta t_2$  değeri Denklem 3.43'ye göre hesaplanır.
  - 22:     Nihai adım büyüklüğü  $\Delta t$ , Denklem 3.44 ile hesaplanır.
  - 23: **end while**
  - 24: **return**  $\Omega$
-

Başlangıç izdüşümü verilen 2B noktalar kümesi  $P$ 'yi girdi olarak alan *DeğişkenAdımBüyükliğüneDayalıEnerjiYayma( $\Omega$ )* isimli enerji tabanlı düzleştirme fonksiyonunun ayrıntıları Algoritma-3'te verilmiştir. Bu fonksiyon, başlangıç izdüşümü verilen yüzey yamasının 2B uzaydaki koordinatlarını (enerjiye dayalı olarak) değişken adımlı bir şekilde hareket ettirerek daha iyi bir düzleştirme sonucunun ortaya çıkmasını sağlamaktadır.



## 4. ARAŞTIRMA BULGULARI VE TARTIŞMA

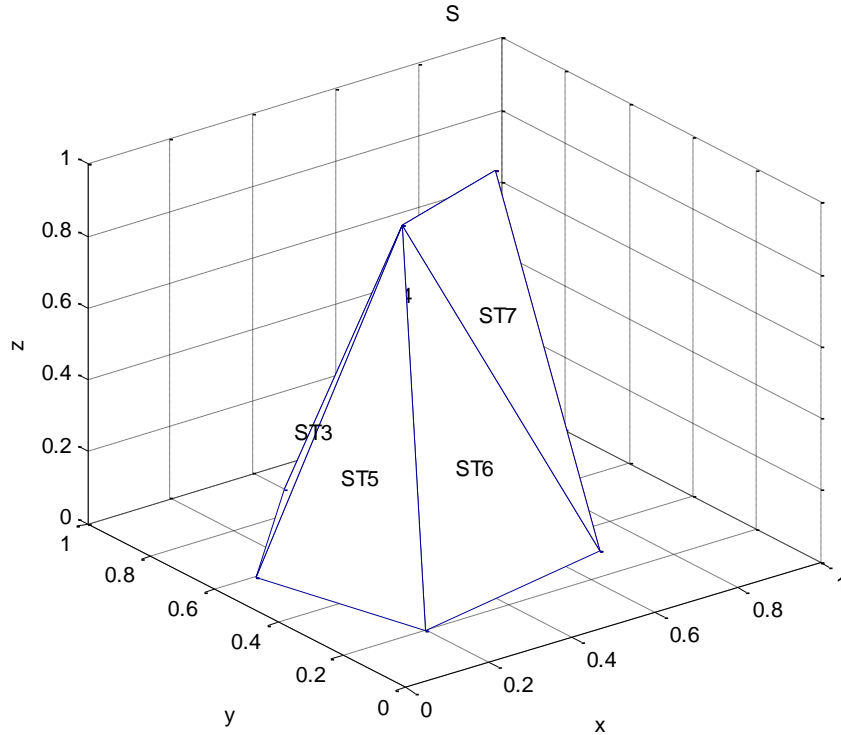
Bu bölümde karmaşık yüzeylerin düzleştirilmesi yönelik geliştirilen yaklaşıma ilişkin araştırma bulguları ve yorumlar yer almaktadır.

### 4.1. Geometrik Düzleştirme Sonuçları

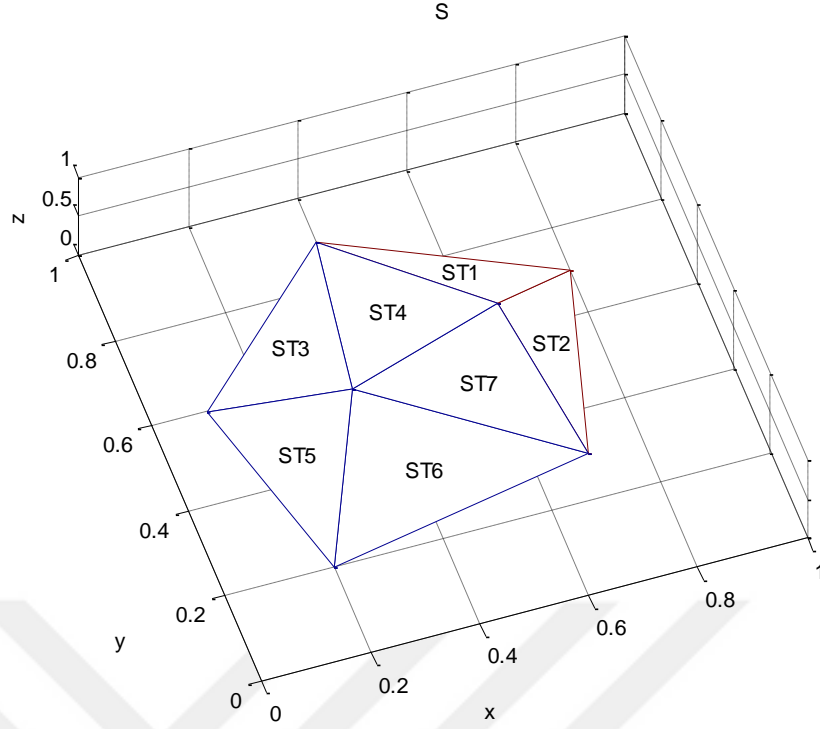
Barisentrik izdüşüm teorisi ve dinamik sinir ağı modeline dayalı olarak önerilen başlangıç izdüşüm algoritmasına ilişkin sonuçlar bu kısımda verilmiştir.

#### 4.1.1. Aydınlatıcı örnekler

Hızlıca anlaşılabilmesi ve kolayca izlenebilmesi için, önerilen başlangıç izdüşüm algoritması ilk olarak, yalnızca iki iç düğümü olan oldukça (toplam 7 köşeden oluşan *Surf1* isimli) basit bir yüzeye uygulanmıştır ve sonuçları verilmiştir. Bu amaçla, Şekil 4.1 ve Şekil 4.2'de iki farklı bakış açısına göre çizdirilen 3B yüzey, algoritmaya girdi olarak sunulmuştur ve tüm ara sonuçlar ayrıntılı bir şekilde gösterilmiştir.



Şekil 4.1. 7-köşeden oluşan örnek bir 3B yüzey yaması

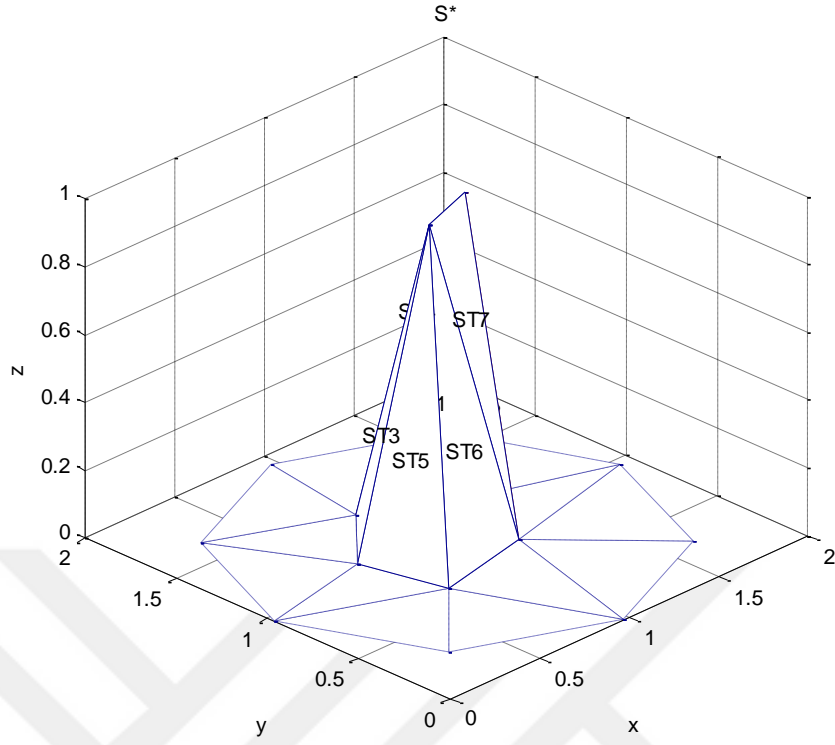


Şekil 4.2. 7-köşeden oluşan örnek bir 3B yüzey yaması (farklı bir açıdan görünüşü)

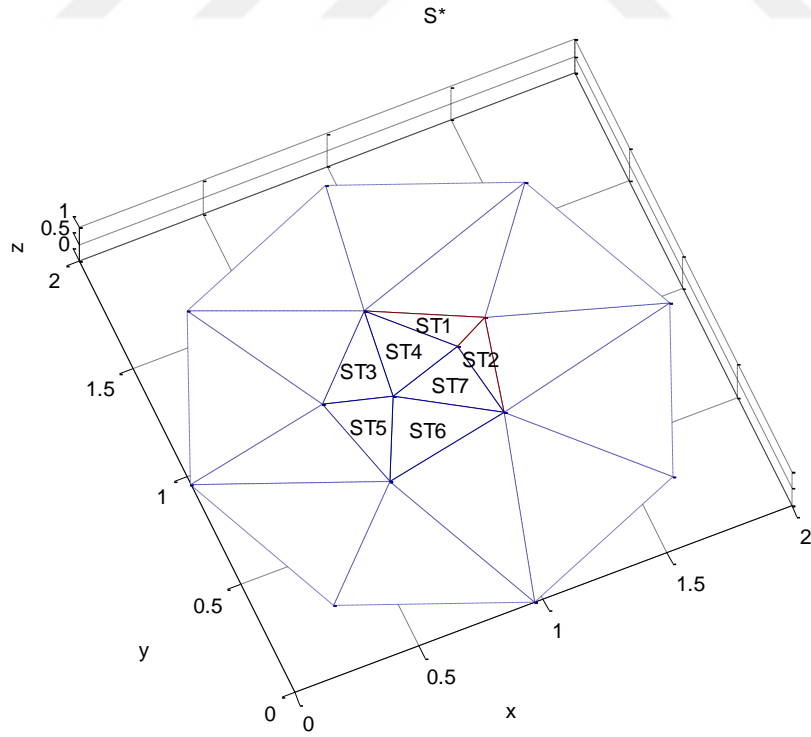
Yüzey kafesinin köşelerine karşılık gelen  $\mathbf{X}$  noktalar bulutu ve üçgen yüzleri açıklayan bilgiler, yani  $\mathbf{F}$  topoloji bilgisi aşağıdaki gibi verilir:

$$\mathbf{X} = \begin{bmatrix} 0.3 & 0.4 & 1 \\ 0.6 & 0.5 & 1 \\ 0.2 & 0.2 & 0 \\ 0.1 & 0.6 & 0 \\ 0.4 & 0.9 & 0 \\ 0.8 & 0.7 & 0 \\ 0.7 & 0.3 & 0 \end{bmatrix}, \mathbf{F} = \begin{bmatrix} 6 & 5 & 2 \\ 6 & 2 & 7 \\ 1 & 5 & 4 \\ 1 & 2 & 5 \\ 4 & 3 & 1 \\ 3 & 7 & 1 \\ 1 & 7 & 2 \end{bmatrix}. \quad (4.1)$$

Denklem 4.1'in tanımladığı 3B yüzey yaması için Bölüm 3.5.1'de detayları verilen sanal sınır poligonu belirleme yaklaşımıyla sanal poligon belirlenir. Bu poligon,  $n_v$  parametresine bağlı olarak çeşitli çözünürlüklerde belirlenebilmektedir. Bu parametrenin başlangıç izdüşümü sonucundaki doğruluk oranlarına olan etkisi Bölüm 4.1.2'de detaylandırılmıştır. Buradaki aydınlatıcı örnek için  $n_v=8$  olarak belirlenmiştir. Orijinal yüzey yamasına sanal sınır poligonu eklenmesiyle oluşan ilaveli yüzeyin ( $\mathbf{S}^*$ ) iki farklı açıdan görünüşü Şekil 4.3 ve Şekil 4.4'te verilmiştir.



Şekil 4.3. *Surf1* yüzey yamasına ( $n_v=8$ ) sanal sınır eklenmesiyle oluşan yüzey



Şekil 4.4. *Surf1* yüzey yamasına ( $n_v=8$ ) sanal sınır eklenmesiyle oluşan yüzey (farklı bir açıdan görünüşü)

Sınır düğümü sayısı  $n_v=8$  olacak şekilde sanal sınır poligonu eklenmesiyle oluşan, Şekil 4.3'te görülen  $S^*$  ilaveli yüzeyin köşelerine karşı düşen noktalar bulutu  $X^*$  ile, üçgen yüzeyleri tanımlayan bilgiler ise  $F^*$  ile verilmiştir (Denklem 4.2). Burada 3B uzaydaki köşe koordinatlarını tutan  $X^*$  matrisinde yeşil renkli olarak verilen koordinatlar, sanal sınır poligonu ekleme sonrasında orijinal yüzeye ilave edilen yeni noktalara aittir. (Bu örnek yüzey için Denklem 3.34'den yararlanarak  $r_x=r_y=r_v/2=0.48$  olarak bulunmuştur.)

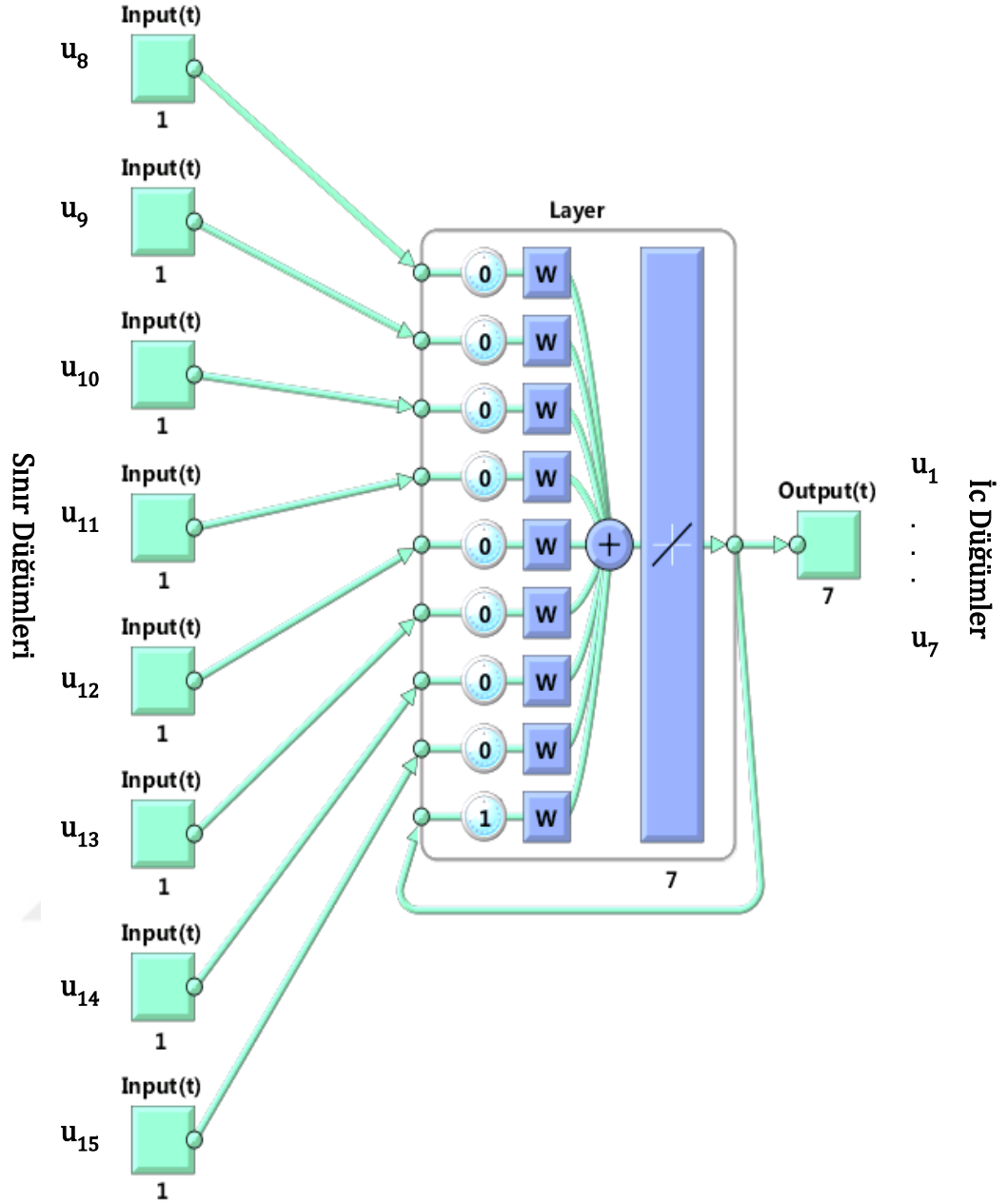
$$X^* = \begin{bmatrix} rx+0.3 & ry+0.4 & 1 \\ rx+0.6 & ry+0.5 & 1 \\ rx+0.2 & ry+0.2 & 0 \\ rx+0.1 & ry+0.6 & 0 \\ rx+0.4 & ry+0.9 & 0 \\ rx+0.8 & ry+0.7 & 0 \\ rx+0.7 & ry+0.3 & 0 \\ 0.28 & 0.28 & 0 \\ 0.96 & 0 & 0 \\ 1.65 & 0.28 & 0 \\ 1.93 & 0.96 & 0 \\ 1.65 & 1.65 & 0 \\ 0.96 & 1.93 & 0 \\ 0.28 & 1.65 & 0 \\ 0 & 0.96 & 0 \end{bmatrix}, F^* = \begin{bmatrix} 15 & 4 & 14 \\ 5 & 12 & 13 \\ 13 & 14 & 5 \\ 5 & 14 & 4 \\ 11 & 12 & 6 \\ 6 & 5 & 2 \\ 12 & 5 & 6 \\ 7 & 10 & 11 \\ 11 & 6 & 7 \\ 7 & 6 & 2 \\ 9 & 10 & 7 \\ 15 & 8 & 3 \\ 3 & 4 & 15 \\ 3 & 8 & 9 \\ 9 & 7 & 3 \\ 1 & 7 & 2 \\ 1 & 3 & 7 \\ 4 & 3 & 1 \\ 2 & 5 & 1 \\ 1 & 5 & 4 \end{bmatrix}. \quad (4.2)$$

Şekil 4.3'te görülen ve Denklem 4.2'de ayrıntıları verilen  $S^*(X, F)$  yüzeyi Algoritma-2'de işlendiğinde, 16. adımın sonunda elde edilen  $\Lambda$  matrisi aşağıdaki gibi olur.

$$\Lambda = \begin{bmatrix} -1 & 0.46 & 0.14 & 0.14 & 0.13 & 0 & 0.13 & 0 & 0.0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.53 & -1 & 0 & 0 & 0.15 & 0.16 & 0.16 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0 & -1 & 0.25 & 0 & 0 & 0.2 & 0.18 & 0.14 & 0 & 0 & 0 & 0 & 0 & 0.14 \\ 0.11 & 0 & 0.27 & -1 & 0.26 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.17 & 0.19 \\ 0.08 & 0.08 & 0 & 0.21 & -1 & 0.2 & 0 & 0 & 0 & 0 & 0.11 & 0.16 & 0.14 & 0 & 0 \\ 0 & 0.11 & 0 & 0 & 0.25 & -1 & 0.28 & 0 & 0 & 0 & 0.17 & 0.19 & 0 & 0 & 0 \\ 0.09 & 0.09 & 0.19 & 0 & 0 & 0.24 & -1 & 0 & 0.12 & 0.14 & 0.13 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (4.3)$$

$\Lambda$  matrisi bahsi geçen örnek yüzey için  $7 \times 15$  boyutunda olur.  $\Lambda$  matrisin boyutu, Bölüm 3.2'de anlatıldığı üzere, yüzeyi oluşturan iç düğüm sayısı ve toplam düğüm sayısı tarafından belirlenir. Denklem 4.2'de  $\mathbf{X}^*$  matrisinde siyah renkli olarak verilen 7 köşenin koordinatları ilaveli yüzey için iç düğüm olarak kabul edilir. Aynı matriste yeşil renkli olarak verilen 8 nokta ise yeni sınır düğümleridir. Daha sonra,  $\Lambda$  matrisinde ilk 7 sütun göz önünde alındığında ortaya çıkan kare matrisin köşegenlerinde gözüken “-1” değerine sahip elemanlar sıfıra eşitlenerek bu matris güncellenir. Güncellenen bu matris Şekil 4.5'te verilen dinamik sınır ağının ağırlık değerleri olarak kullanılır. Bu şekilde, 3B kafes yüzeyinin yerel geometri bilgisi dinamik sınır ağına aktarılmış olur.

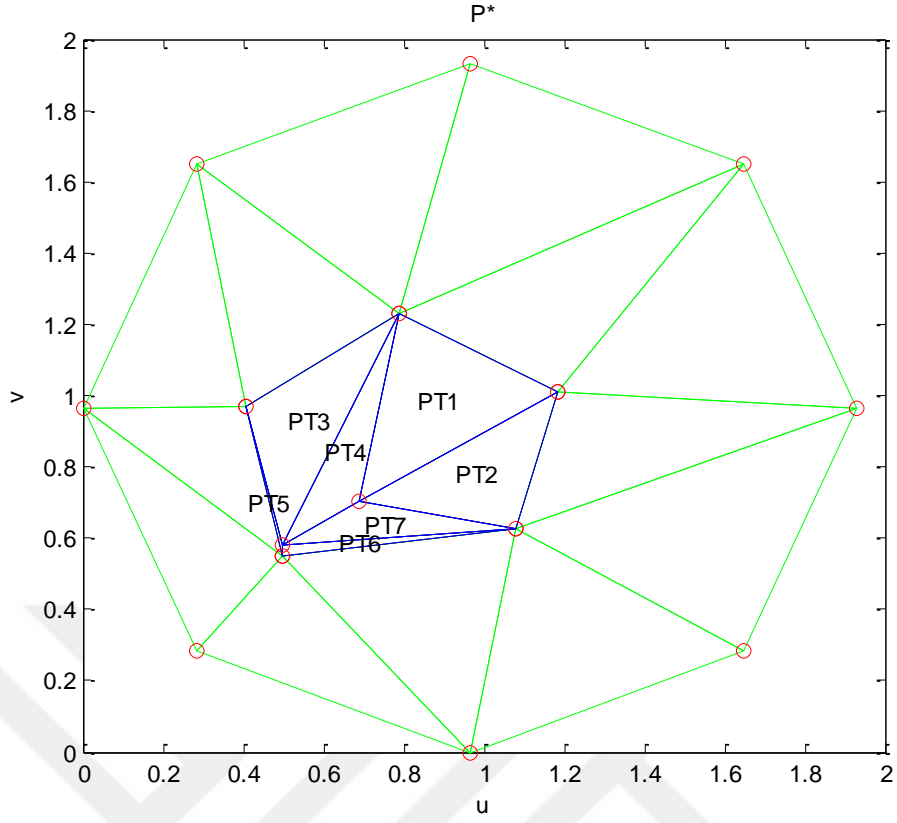




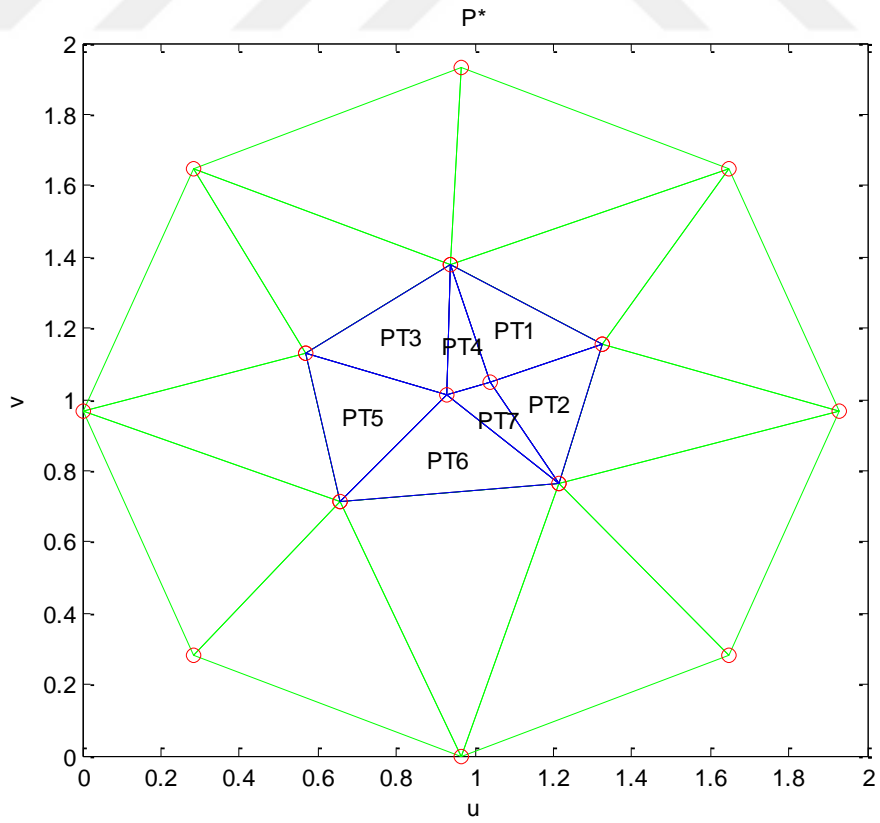
Şekil 4.5. *Surf1* yüzey yaması için oluşturulan dinamik sinir ağıının yapısı

Şekil 4.5'teki dinamik sinir ağı verilen örnek yüzey için çalıştırıldığında 3 ve 10 iterasyon sonundaki (u-v parametre uzayındaki) düzlemsel izdüşüm sonuçları sırasıyla Şekil 4.6 ve Şekil 4.7'de verilmiştir.



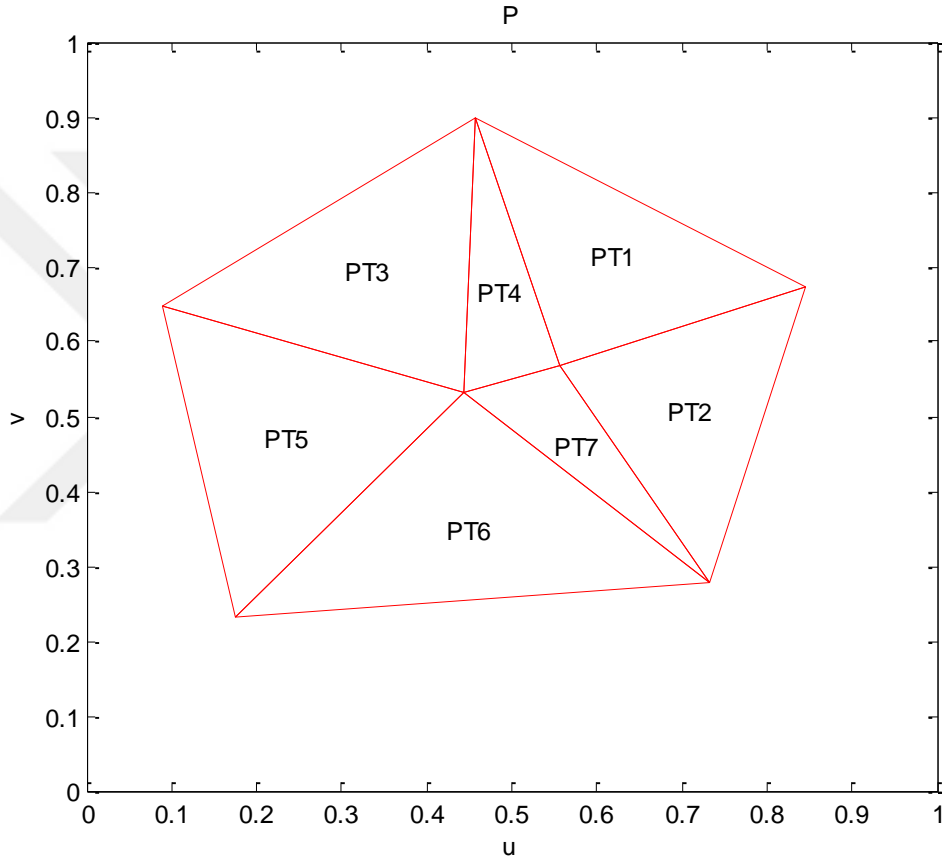


Şekil 4.6. *Surf1* yüzey yamasının 3 iterasyon sonundaki düzlemsel izdüşümü



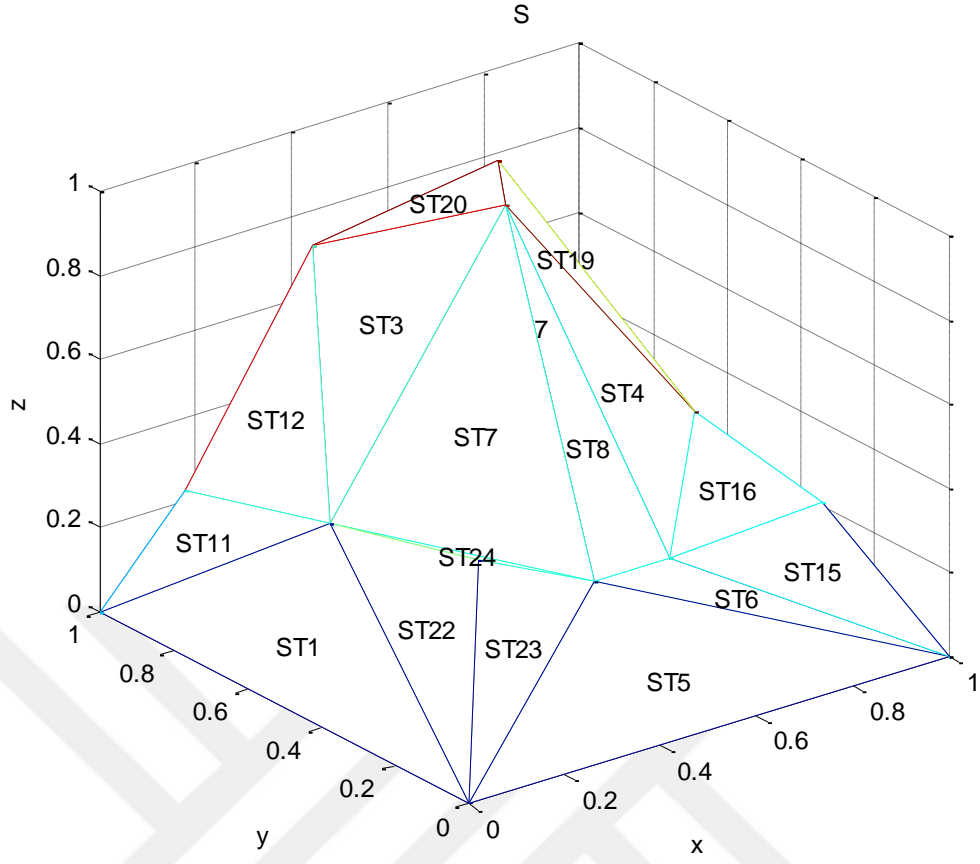
Şekil 4.7. *Surf1* yüzey yamasının 10 iterasyon sonundaki düzlemsel izdüşümü

Şekil 4.7’de 10 iterasyon sonucunda elde edilen izdüşüm sonucundan sanal sınır düğümleri çıkarılmasıyla başlangıçta verilen orijinal 3B yüzeye ilişkin başlangıç izdüşüm sonucu elde edilir. Bahsi geçen (Şekil 4.1’deki) yüzeyin başlangıç izdüşüm sonucu Şekil 4.8’de verilmiştir. Düzlemsel geliştirme sonucunda elde edilen başlangıç izdüşüm sonucunun alan-doğruluk farkı  $E_S = 0.7570$  ve kenar-doğruluk farkı  $E_C = 0.5825$  olarak bulunmuştur.



Şekil 4.8. Sanal sınır düğümlerinin çıkarılmasıyla elde edilen düzlemsel izdüşüm sonucu (*Surf1* yüzey yaması)

İkinci yüzey olarak, 15 köşeden oluşan (*Surf2* isimli) nispeten daha karmaşık bir yüzey yaması kullanılmıştır. Bu amaçla, Şekil 4.9’da görülen 3B yüzey, algoritmaya girdi olarak sunulmuştur ve tüm ara sonuçlar ayrıntılı bir şekilde gösterilmiştir.

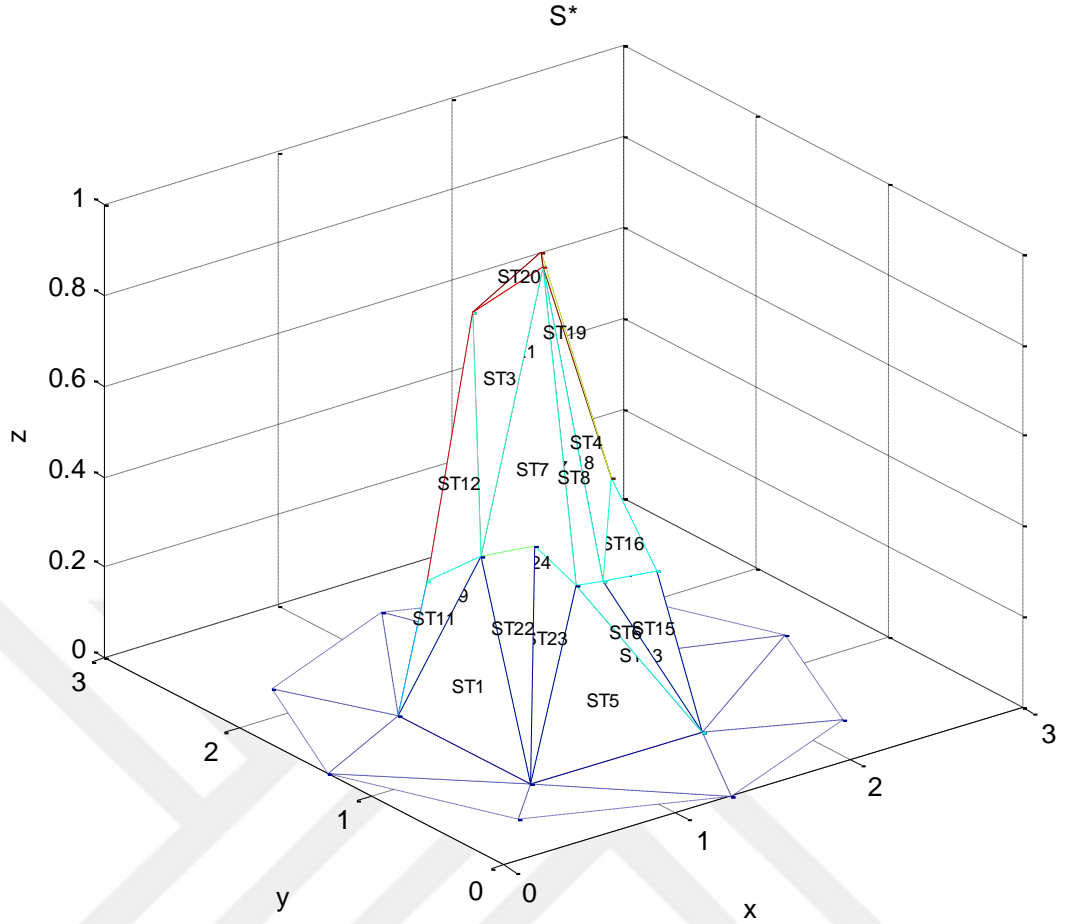


Şekil 4.9. 15-köşeden oluşan örnek bir 3B yüzey yaması

Bu yüzeyin köşelerini tanımlayan noktalar bulutunun koordinatları ve topoloji bilgisi aşağıda verilmiştir.

$$\mathbf{X} = \begin{bmatrix} 0.1 & 0.1 & 0.5 \\ 0.1 & 0.9 & 0.3 \\ 0.65 & 0.82 & 0.45 \\ 0.7 & 0.3 & 0.55 \\ 0.3 & 0.05 & 0.4 \\ 0.85 & 0.15 & 0.35 \\ 0.5 & 0.55 & 1 \\ 0.6 & 0.7 & 1 \\ 0.08 & 0.48 & 0.42 \\ 0.48 & 0.08 & 0.38 \\ 0.25 & 0.75 & 0.9 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \mathbf{F} = \begin{bmatrix} 9 & 13 & 12 \\ 14 & 13 & 3 \\ 11 & 9 & 7 \\ 7 & 10 & 4 \\ 12 & 15 & 5 \\ 15 & 10 & 5 \\ 7 & 9 & 5 \\ 5 & 10 & 7 \\ 2 & 3 & 13 \\ 2 & 11 & 3 \\ 13 & 9 & 2 \\ 9 & 11 & 2 \\ 6 & 15 & 14 \\ 14 & 4 & 6 \\ 6 & 10 & 15 \\ 6 & 4 & 10 \\ 14 & 3 & 8 \\ 8 & 4 & 14 \\ 7 & 4 & 8 \\ 8 & 11 & 7 \\ 3 & 11 & 8 \\ 1 & 9 & 12 \\ 12 & 5 & 1 \\ 1 & 5 & 9 \end{bmatrix}. \quad (4.4)$$

Denklem 4.4'ün tanımladığı 3B yüzey yaması için sanal poligon belirlenir. Buradaki aydınlatıcı örnek için yine  $n_v=8$  olarak belirlenmiştir. Orijinal yüzey yamasına sanal sınır poligonu eklenmesiyle oluşan ilaveli yüzeyin ( $\mathbf{S}^*$ ) görünüşü Şekil 4.10'da verilmiştir.



Şekil 4.10. *Surf2* yüzey yamasına ( $n_v=8$ ) sanal sınır eklenmesiyle oluşan yüzey

Sınır düğümü sayısı  $n_v=8$  olacak şekilde sanal sınır poligonu eklenmesiyle oluşan, Şekil 4.10'da görülen  $S^*$  ilaveli yüzeyin köşelerine karşı düşen noktalar bulutu  $X^*$  ile, üçgen yüzeyleri tanımlayan bilgiler ise  $F^*$  ile verilmiştir (Denklem 4.5). (Bu örnek yüzey için Denklem 3.34'den yararlanarak  $r_x=r_y=r_v/2=0.66$  olarak bulunmuştur.)

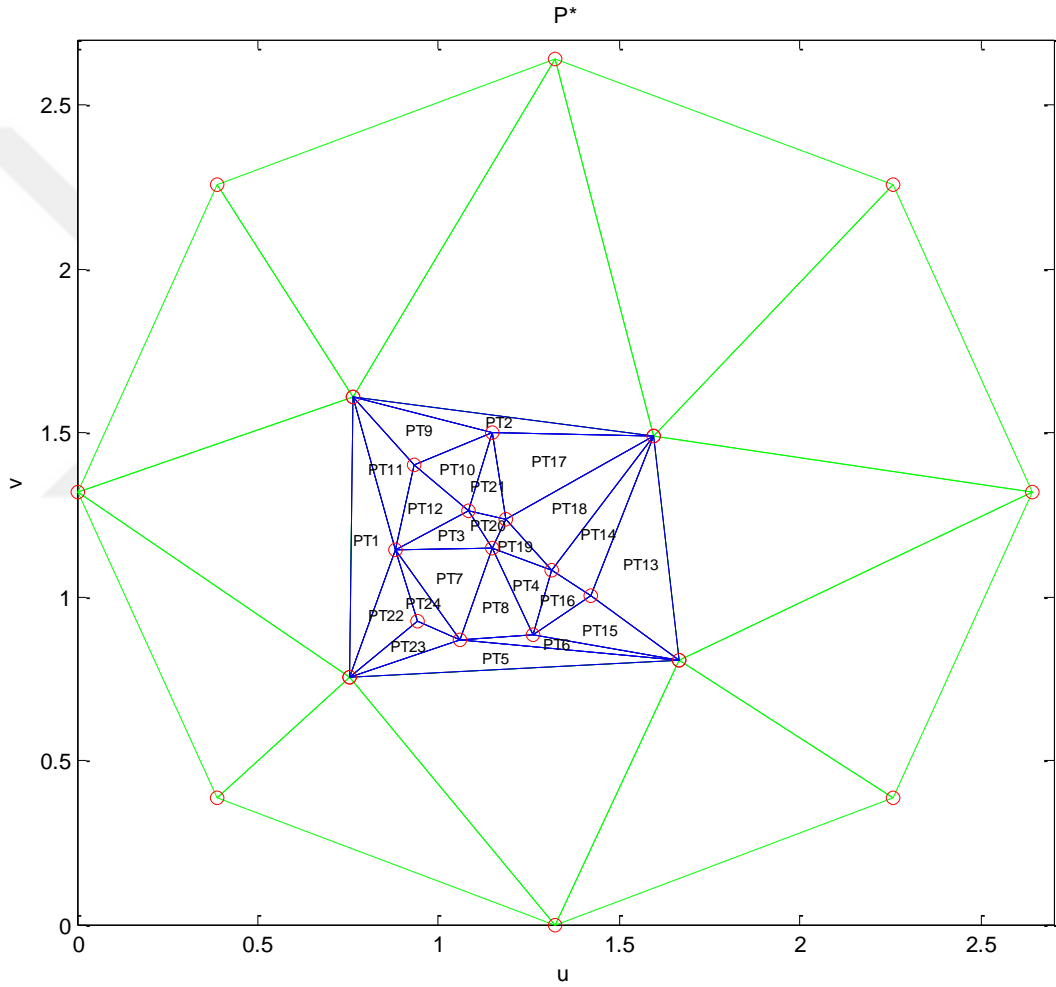
$$\mathbf{X}^* = \begin{bmatrix}
rx+0.1 & ry+0.1 & 0.5 \\
rx+0.1 & ry+0.9 & 0.3 \\
rx+0.65 & ry+0.82 & 0.45 \\
rx+0.7 & ry+0.3 & 0.55 \\
rx+0.3 & ry+0.05 & 0.4 \\
rx+0.85 & ry+0.15 & 0.35 \\
rx+0.5 & ry+0.55 & 1 \\
rx+0.6 & ry+0.7 & 1 \\
rx+0.08 & ry+0.48 & 0.42 \\
rx+0.48 & ry+0.08 & 0.38 \\
rx+0.25 & ry+0.75 & 0.9 \\
rx+0 & ry+0 & 0 \\
rx+0 & ry+1 & 0 \\
rx+1 & ry+1 & 0 \\
rx+1 & ry+0 & 0 \\
0.39 & 0.39 & 0 \\
1.32 & 0 & 0 \\
2.26 & 0.39 & 0 \\
2.64 & 1.32 & 0 \\
2.26 & 2.26 & 0 \\
1.32 & 2.64 & 0 \\
0.39 & 2.26 & 0 \\
0 & 1.32 & 0
\end{bmatrix}, \mathbf{F}^* = \begin{bmatrix}
13 & 22 & 23 \\
21 & 22 & 13 \\
19 & 20 & 14 \\
14 & 15 & 19 \\
14 & 20 & 21 \\
16 & 12 & 23 \\
17 & 12 & 16 \\
19 & 15 & 18 \\
18 & 15 & 17 \\
17 & 15 & 10 \\
23 & 12 & 9 \\
9 & 13 & 23 \\
15 & 14 & 6 \\
6 & 10 & 15 \\
21 & 13 & 3 \\
3 & 14 & 21 \\
8 & 14 & 3 \\
8 & 3 & 11 \\
1 & 9 & 12 \\
4 & 14 & 8 \\
4 & 6 & 14 \\
10 & 6 & 4 \\
13 & 9 & 2 \\
9 & 11 & 2 \\
2 & 3 & 13 \\
2 & 11 & 3 \\
9 & 1 & 5 \\
5 & 1 & 12 \\
5 & 12 & 17 \\
17 & 10 & 5 \\
10 & 4 & 7 \\
7 & 5 & 10 \\
9 & 5 & 7 \\
7 & 4 & 8 \\
8 & 11 & 7 \\
7 & 11 & 9
\end{bmatrix}. \quad (4.5)$$

Şekil 4.10'da görülen ve Denklem 4.5 ile ayrıntıları verilen  $\mathbf{S}^*(\mathbf{X}, \mathbf{F})$  yüzeyi Algoritma-2'de işlendiğinde, 15. adımın sonunda elde edilen  $\mathbf{\Lambda}$  matrisi Ek A'da Denklem A.1 ile verilmiştir.  $\mathbf{\Lambda}$  matrisi bahsi geçen örnek yüzey için 15x23 boyutundadır.

Daha sonra,  $\mathbf{\Lambda}$  matrisinde ilk 15 sütün göz önünde alındığında ortaya çıkan kare matrisin köşegenlerinde gözükten "-1" değerine sahip elemanlar sıfıra eşitlenerek bu matris güncellenir. Güncellenen bu matris ekler bölümünde Şekil

B.1’de verilen, bu yüzey için oluşturulan dinamik sınır ağının ağırlık değerleri olarak kullanılır. Bu şekilde, 3B kafes yüzeyinin yerel geometri bilgisi dinamik sınır ağına aktarılmış olur.

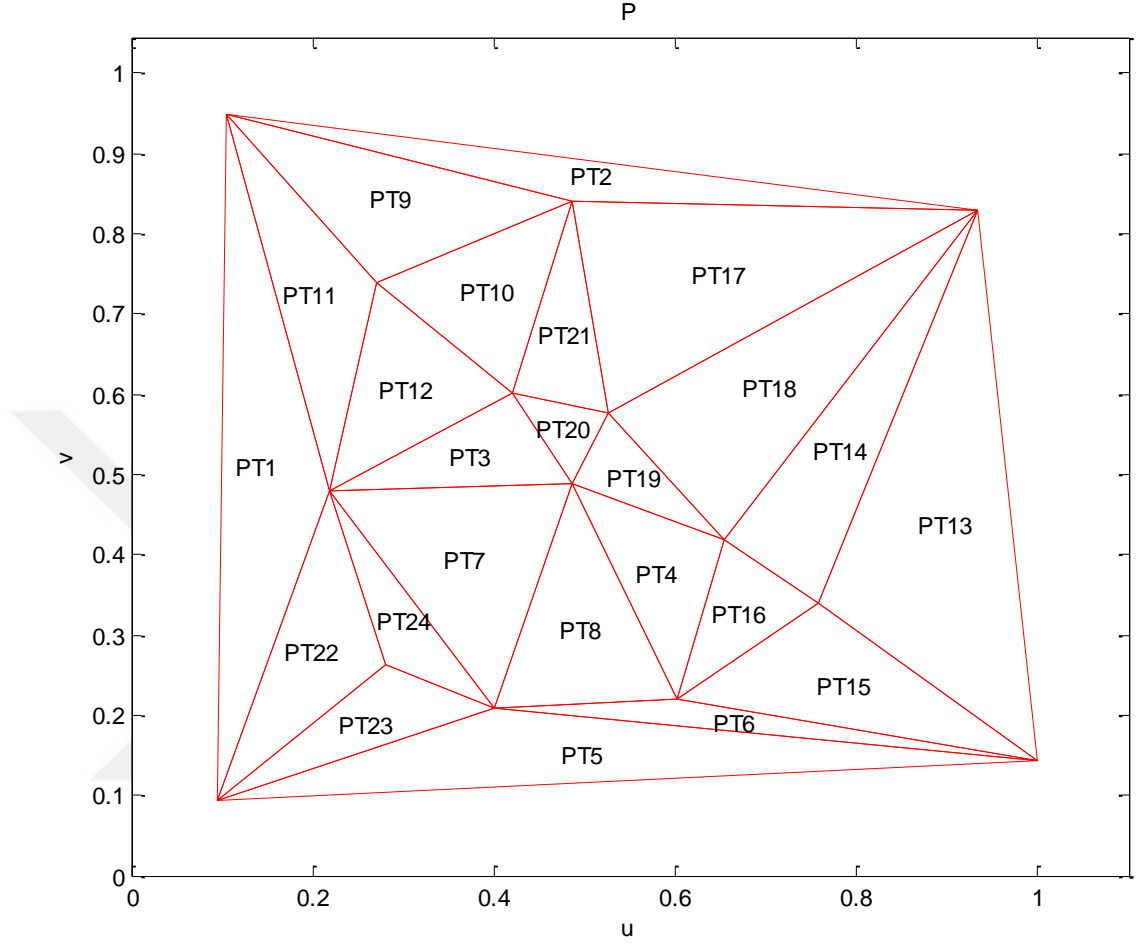
Şekil B.1’deki dinamik sınır ağı, verilen örnek yüzey için çalıştırıldığında 50 iterasyon sonundaki (u-v parametre uzayındaki) düzlemsel izdüşüm sonucu Şekil 4.11’de verilmiştir.



Şekil 4.11. *Surf2* yüzey yamasasının 50 iterasyon sonundaki düzlemsel izdüşümü

Şekil 4.11’deki 50 iterasyon sonucunda elde edilen izdüşüm sonucundan sanal sınır düğümlerinin çıkarılmasıyla ilk başta verilen orijinal 3B yüzeye ilişkin başlangıç izdüşüm sonucu elde edilir ve bu sonuç Şekil 4.12’de verilmiştir. Düzlemsel geliştirme sonucunda elde edilen başlangıç izdüşüm sonucunun alan-

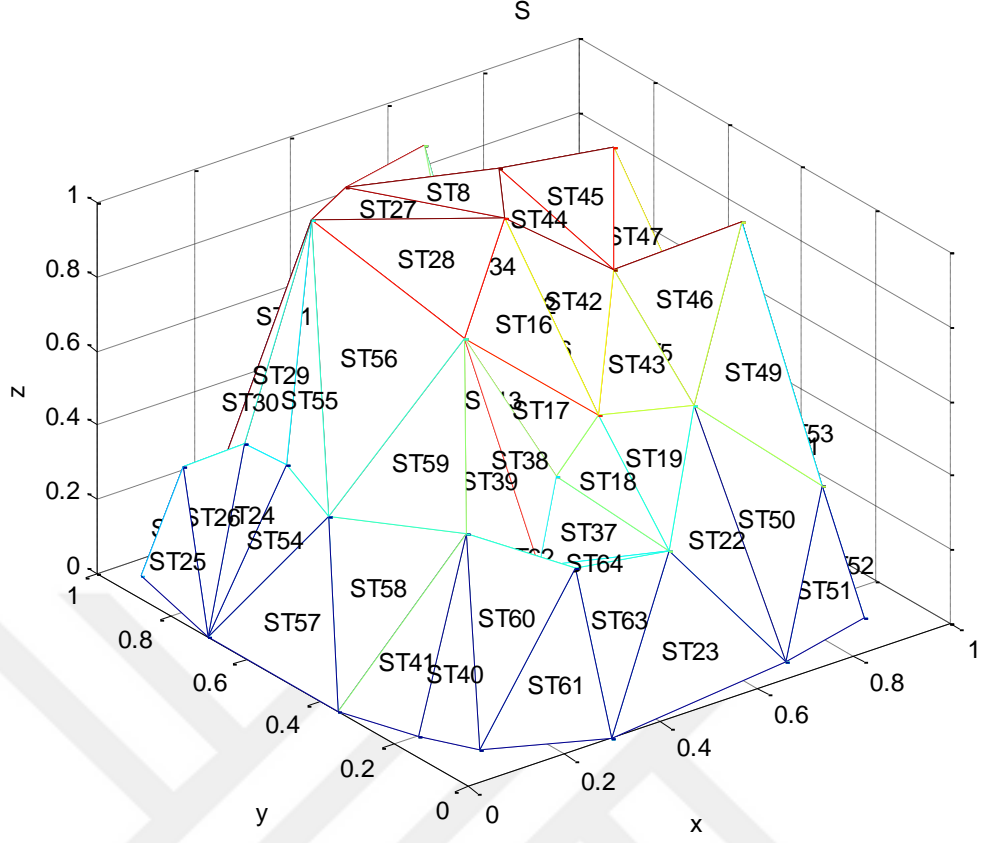
doğruluk farkı  $E_S = 0.7565$  ve kenar-doğruluk farkı  $E_C = 0.4543$  olarak bulunmuştur.



Şekil 4.12. Sanal sınır düğümlerinin çıkarılmasıyla elde edilen düzlemsel izdüşüm sonucu (*Surf2* yüzey yaması)

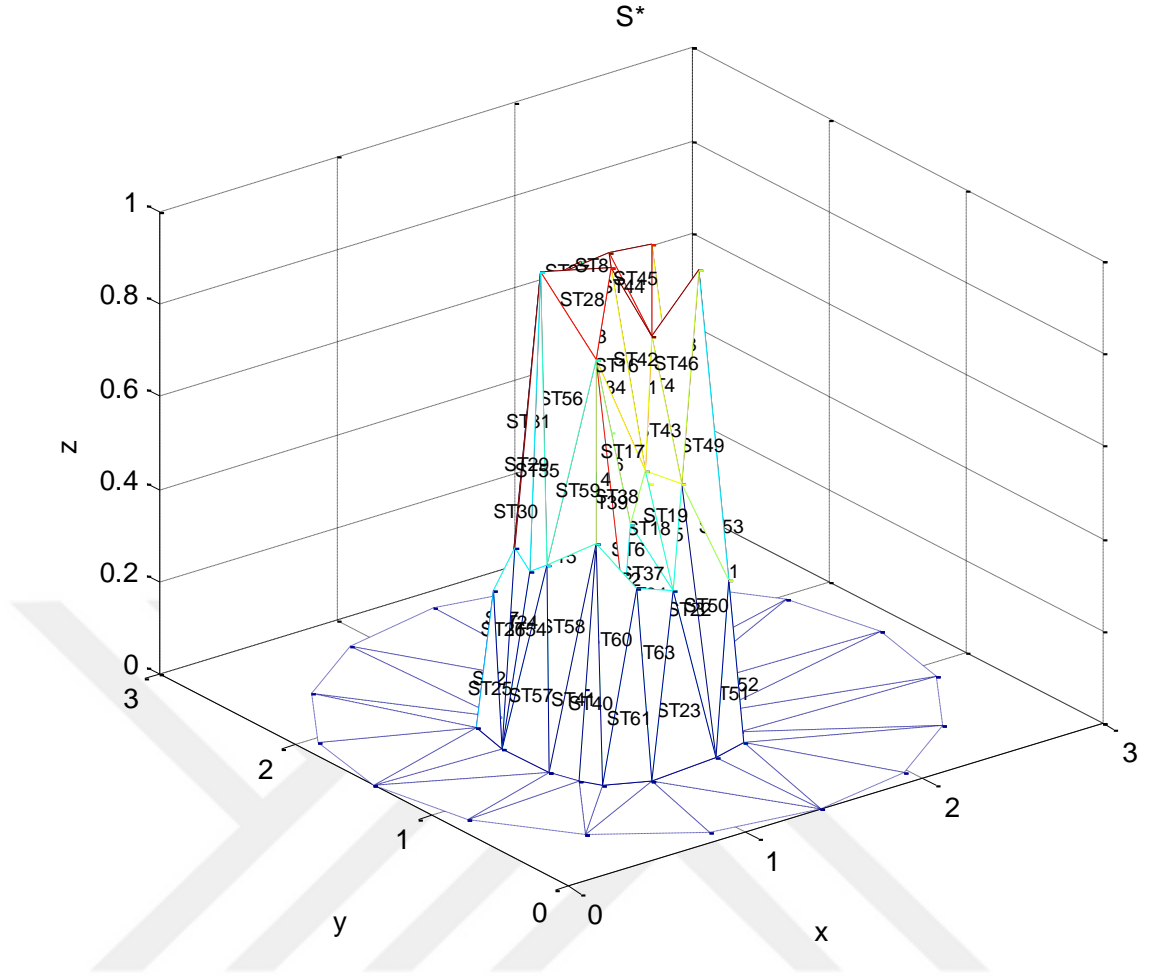
Üçüncü yüzey olarak, 40 köşeden oluşan bir karmaşık yüzey yaması kullanılmıştır. Bu amaçla, Şekil 4.13'de 3B görünüşü verilen yüzey, algoritmaya girdi olarak sunulmuştur ve tüm ara sonuçlar ayrıntılı bir şekilde gösterilmiştir.





Şekil 4.13. 40-köşeden oluşan örnek bir 3B yüzey yaması

Bu yüzeyin köşelerini tanımlayan noktalar bulutunun koordinatları ve topoloji bilgisi ekler bölümünde denklem A.2'de verilmiştir. Daha sonra, bu yüzey yaması için sanal sınır poligonu belirlenir. Buradaki aydınlatıcı örnek için bu kez  $n_v=16$  olarak belirlenmiştir. Orijinal yüzey yamasına sanal sınır poligonu eklenmesiyle oluşan ilaveli yüzeyin ( $S^*$ ) görünüşü Şekil 4.14'te verilmiştir.



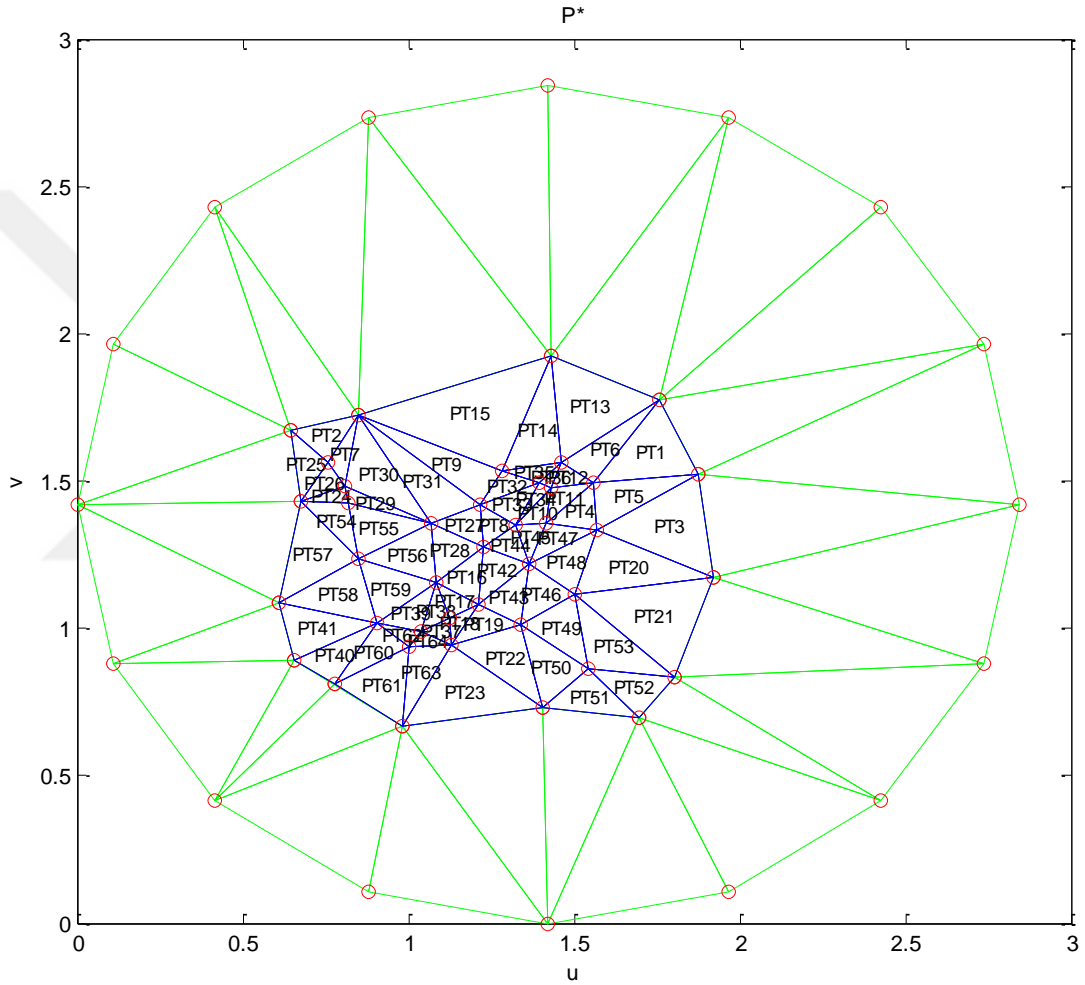
Şekil 4.14. *Surf3* yüzey yamasına ( $n_v=16$ ) sanal sınır eklenmesiyle oluşan yüzey

Sınır düğümü sayısı  $n_v=16$  olacak şekilde sanal sınır poligonu eklenmesiyle oluşan, Şekil 4.14'te görülen  $S^*$  ilaveli yüzeyin köşelerine karşı düşen noktalar bulutu  $X^*$  ile, üçgen yüzeyleri tanımlayan bilgiler ise  $F^*$  ile verilmiştir (Denklem A.3). (Bu örnek yüzey için denklem 3.34'den yararlanarak  $r_x=r_y=r_v/2=0.71$  olarak bulunmuştur.)

Şekil 4.14'te görülen ve Denklem A.3 ile ayrıntıları verilen  $S^*(X, F)$  yüzeyi Algoritma-2'de işlendiğinde  $\Lambda$  matrisi bahsi geçen örnek yüzey için  $40 \times 56$  boyutundadır. Daha sonra,  $\Lambda$  matrisde ilk 40 sütun göz önünde alındığında ortaya çıkan kare matrisin köşegenlerinde gözükten "-1" değerine sahip elemanlar sıfıra eşitlenerek bu matris güncellenir. Güncellenen bu matris ekler bölümünde Şekil B.2'de verilen, bu yüzey için oluşturulan dinamik sınır ağının

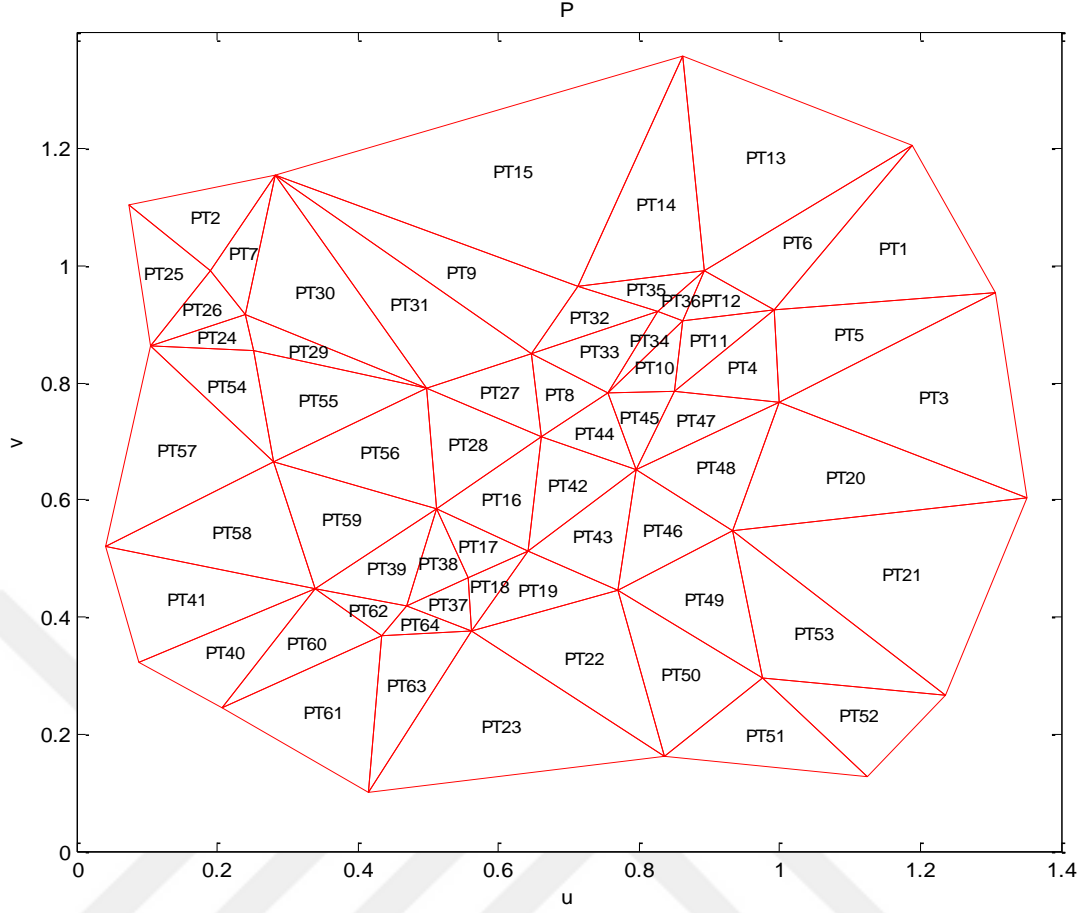
ağırlık değerleri olarak kullanılır. Bu şekilde, 3B kafes yüzeyinin yerel geometri bilgisi dinamik sinir ağına aktarılmış olur.

Şekil B.2'deki dinamik sinir ağı verilen örnek yüzey için çalıştırıldığında 100 iterasyon sonundaki (u-v parametre uzayındaki) düzlemsel izdüşüm sonucu Şekil 4.15'te verilmiştir.



Şekil 4.15. *Surf3* yüzey yamasının 100 iterasyon sonundaki düzlemsel izdüşümü

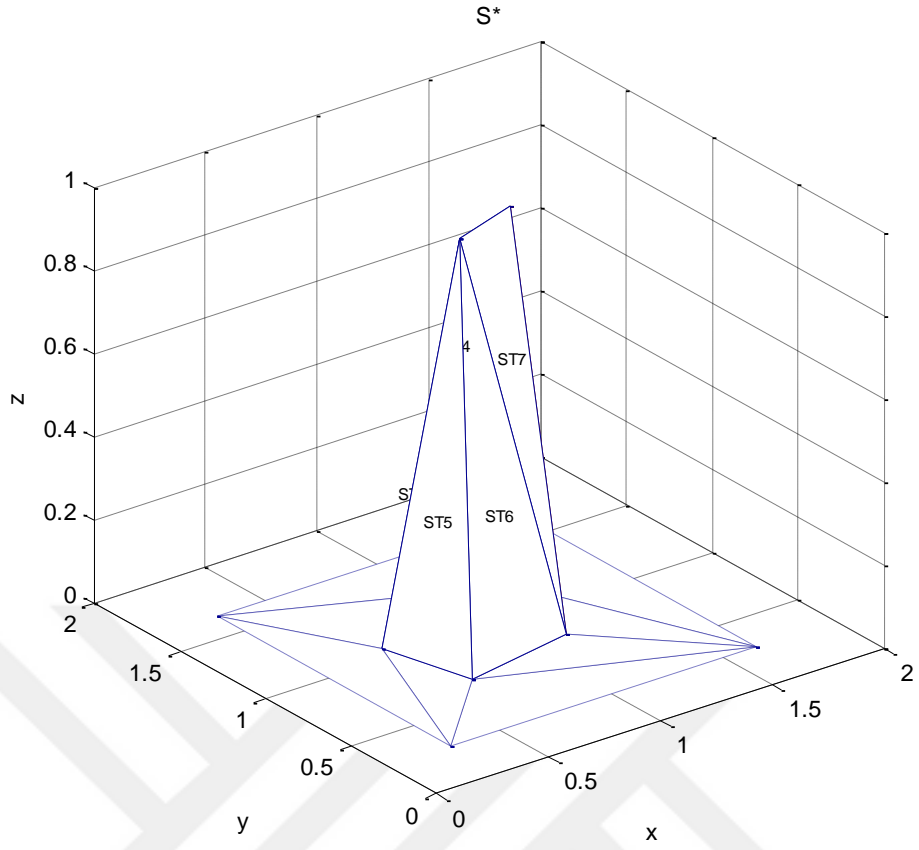
Şekil 4.15'teki 100 iterasyon sonucunda elde edilen izdüşüm sonucundan sanal sınır düğümleri çıkarılmasıyla başlangıçta verilen orijinal 3B yüzeye ilişkin başlangıç izdüşüm sonucu elde edilir ve bu sonuç Şekil 4.16'da verilmiştir. Düzlemsel geliştirme sonucunda elde edilen başlangıç izdüşüm sonucunun alan-doğruluk farkı  $E_s = 0.5911$  ve kenar-doğruluk farkı  $E_c = 0.4810$  olarak bulunmuştur.



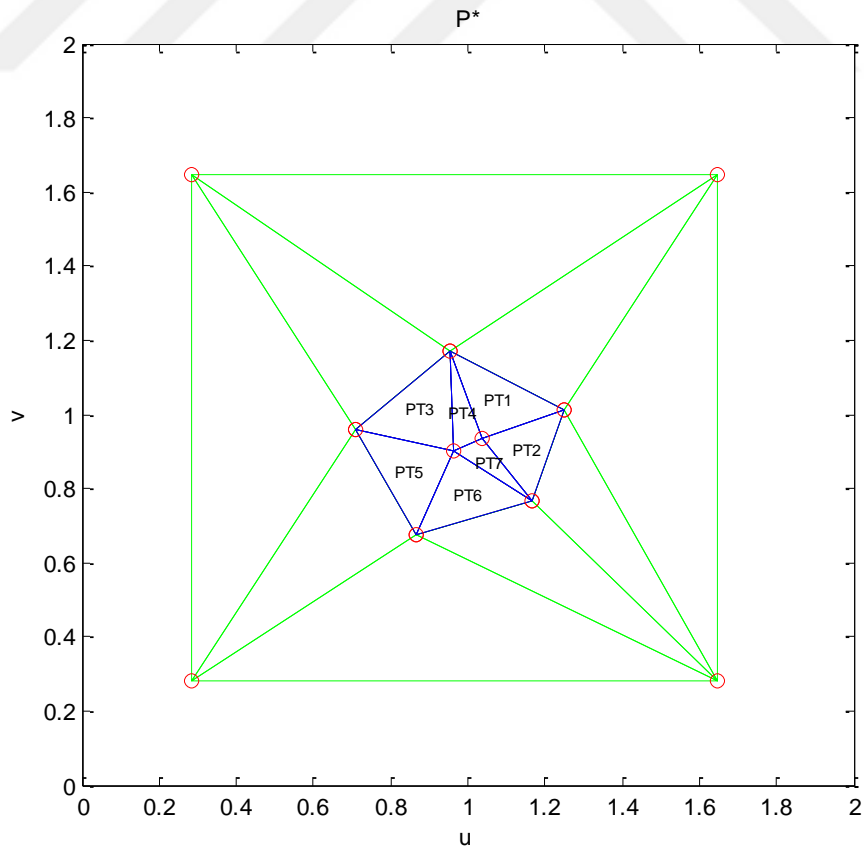
Şekil 4.16. Sanal sınır düğümlerinin çıkarılmasıyla elde edilen düzlemsel izdüşüm sonucu (*Surf3* yüzey yaması)

#### 4.1.2. Sanal sınır düğüm sayısının izdüşüm sonucuna etkisi

Bölüm 4.1.1’de örnek yüzeylerin başlangıç izdüşümleri yapılırken sanal sınır poligonundaki düğüm sayısı sabit tutulmuştur. Bu bölümde, önceki bölümde verilen örnek yüzeyler için farklı çözünürlükte sanal sınır poligonu oluşturulmuştur ve bu sanal sınır poligonundaki düğüm sayısının Bölüm 3.5.2’de verilen Algoritma-2’nin ürettiği başlangıç izdüşüm sonucunu nasıl etkilediği incelenmiştir. Birinci örnek yüzey yaması (*Surf1*) için çeşitli çözünürlüklerde sanal sınır poligonları üretilerek izdüşüm yapılmıştır. Anılan yüzey yaması için sanal düğüm sayıları  $n_v=4$ ,  $n_v=6$ ,  $n_v=12$ ,  $n_v=16$ ,  $n_v=32$  ve  $n_v=64$  olacak şekilde elde edilen ilaveli yüzey yamaları ve bunların düzlemsel karşılıkları Şekil 4.17-22’de verilmiştir. (Bu yüzey yaması için tüm izdüşüm sonuçları 10 iterasyon sonucunda elde edilmiştir.)

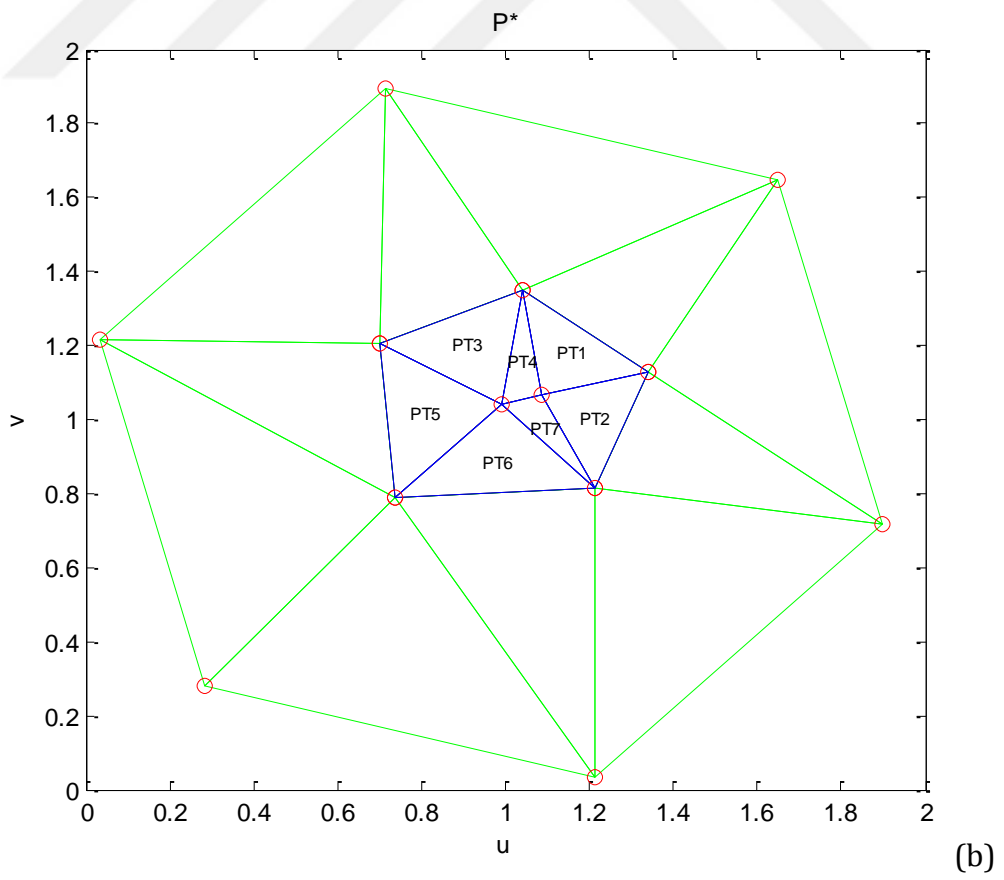
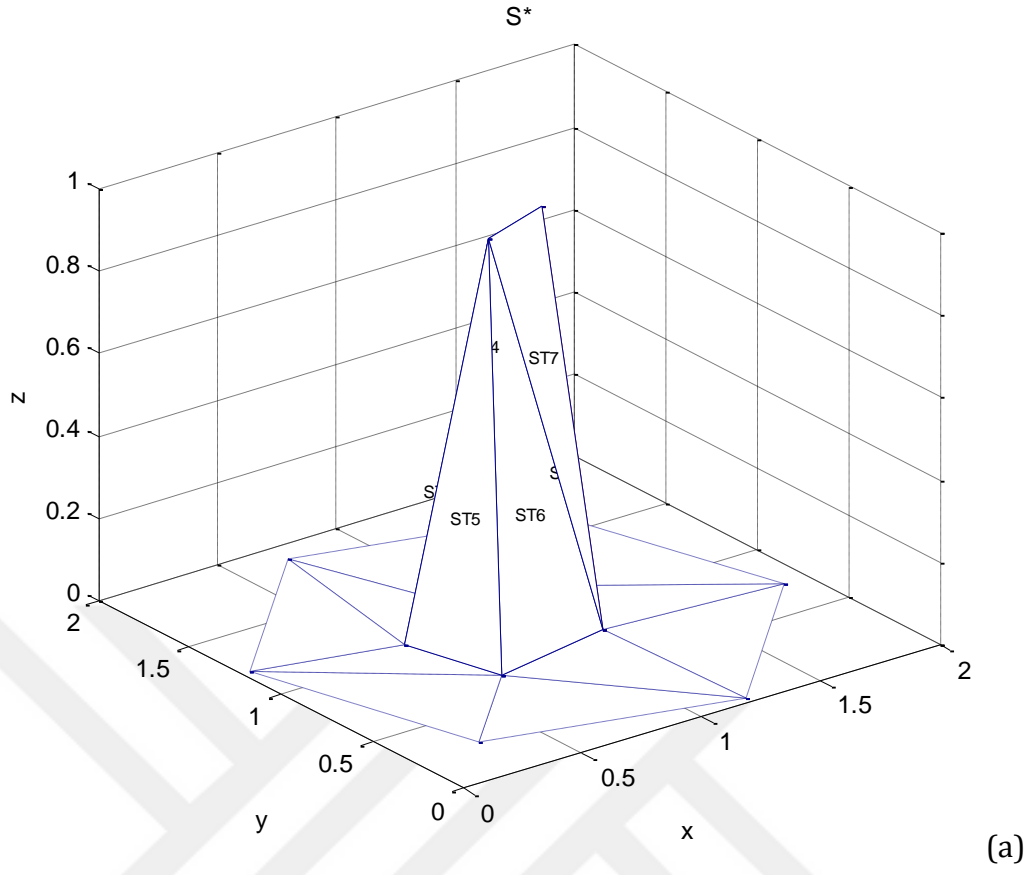


(a)

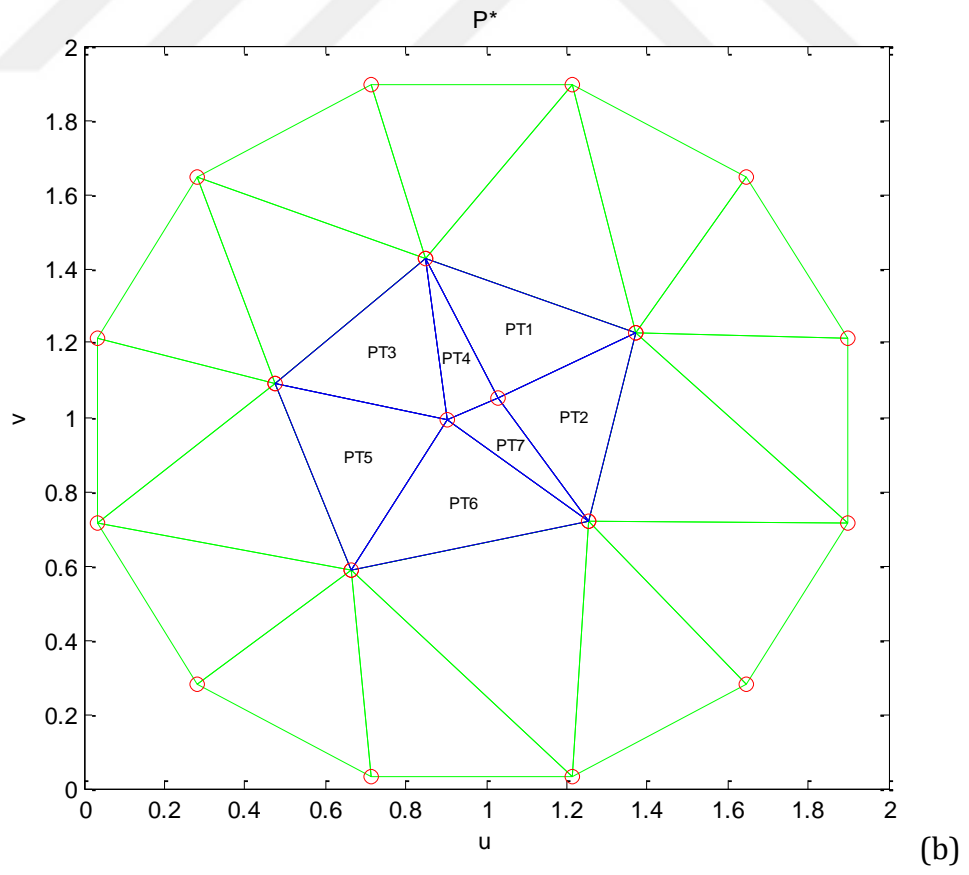
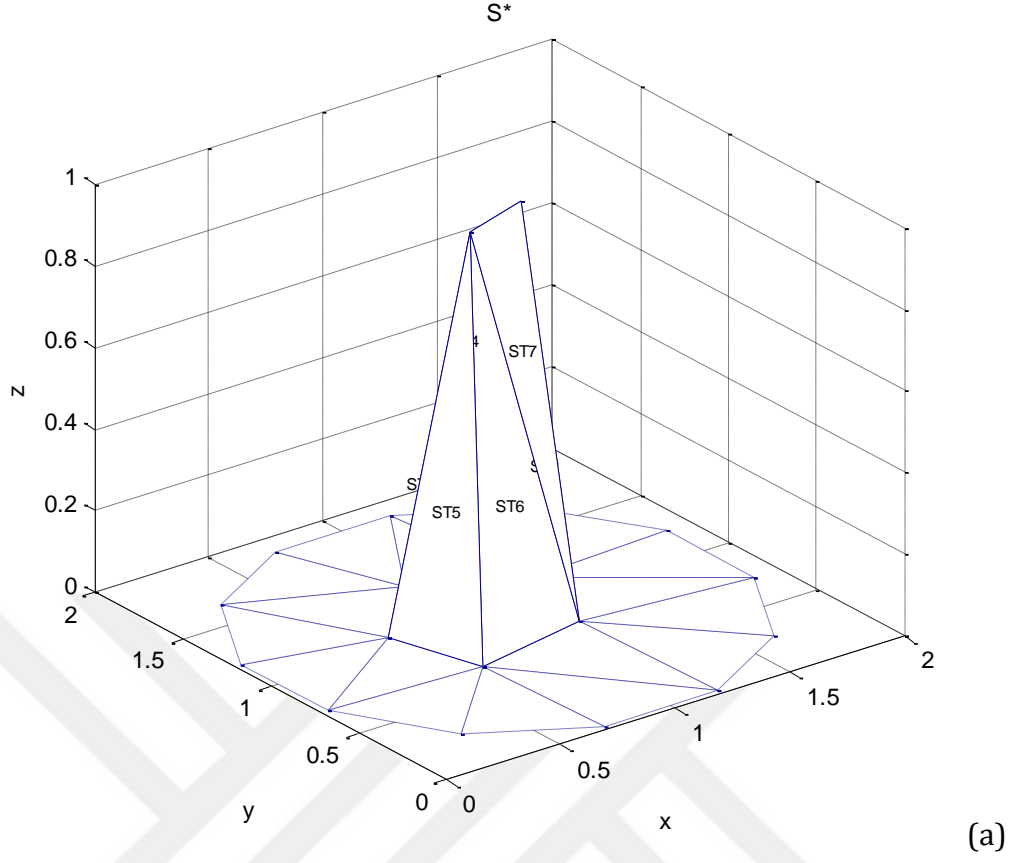


(b)

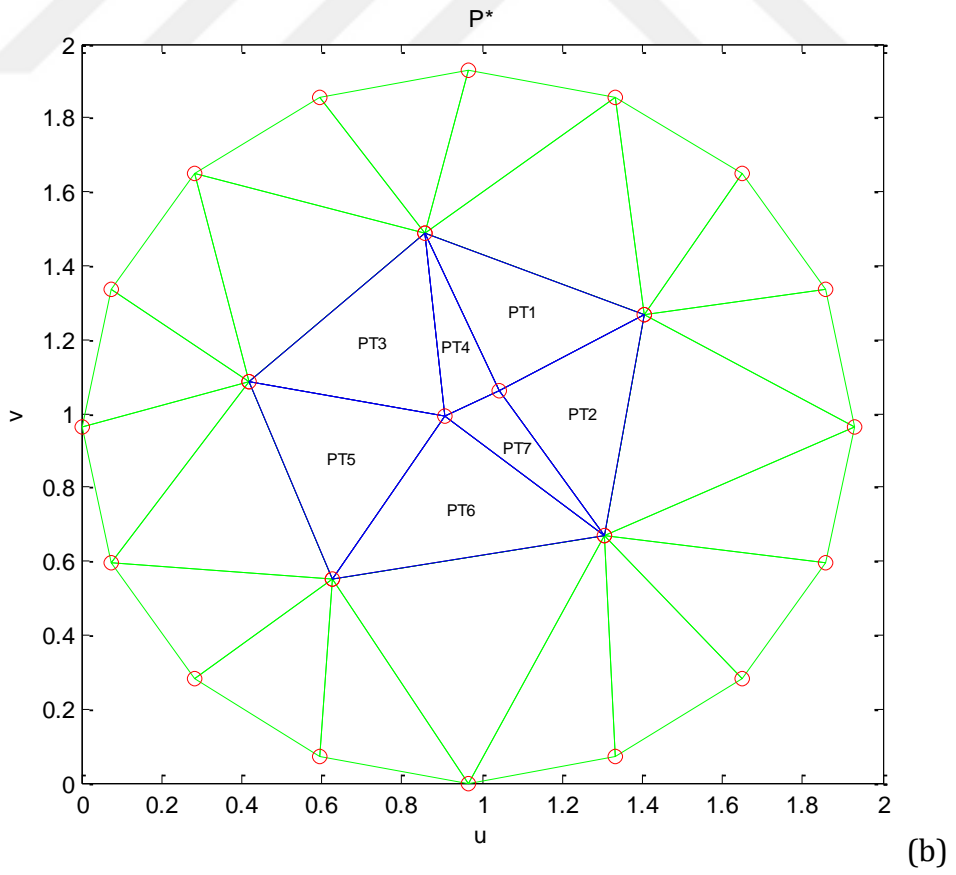
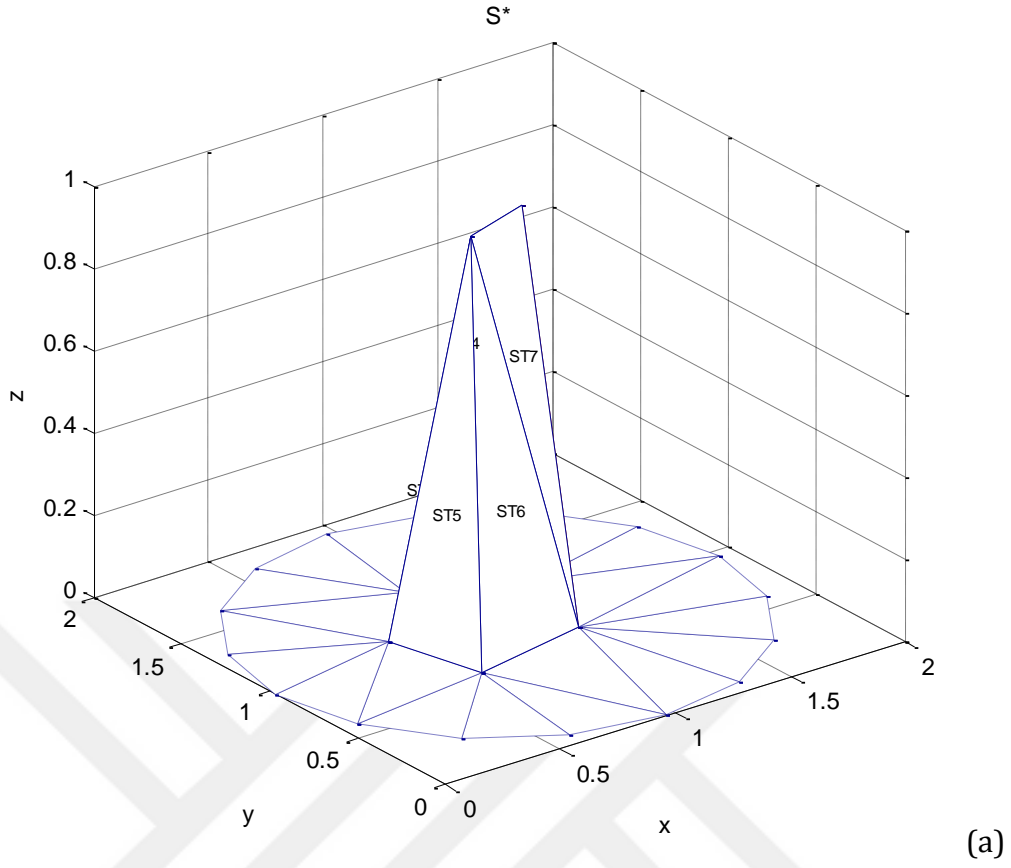
Şekil 4.17.  $Surf1$ 'in  $n_v=4$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)



Şekil 4.18. Surf1'in  $n_v=6$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

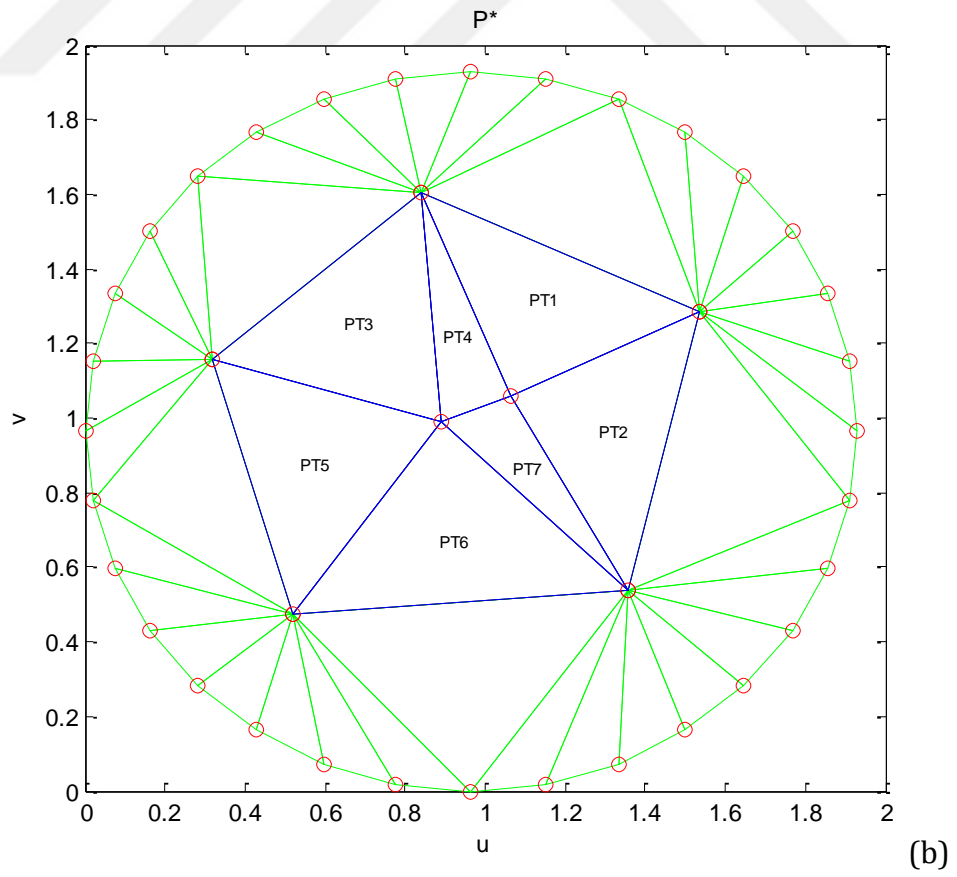
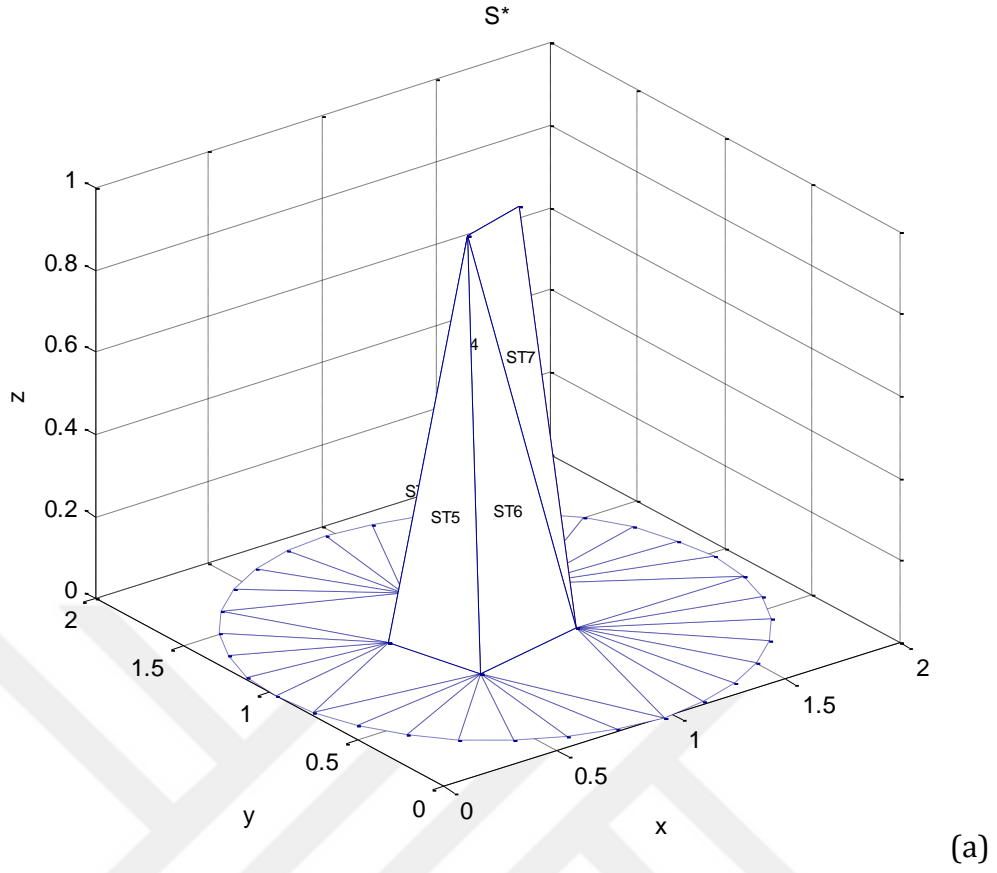


Şekil 4.19. *Surf1*'in  $n_v=12$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

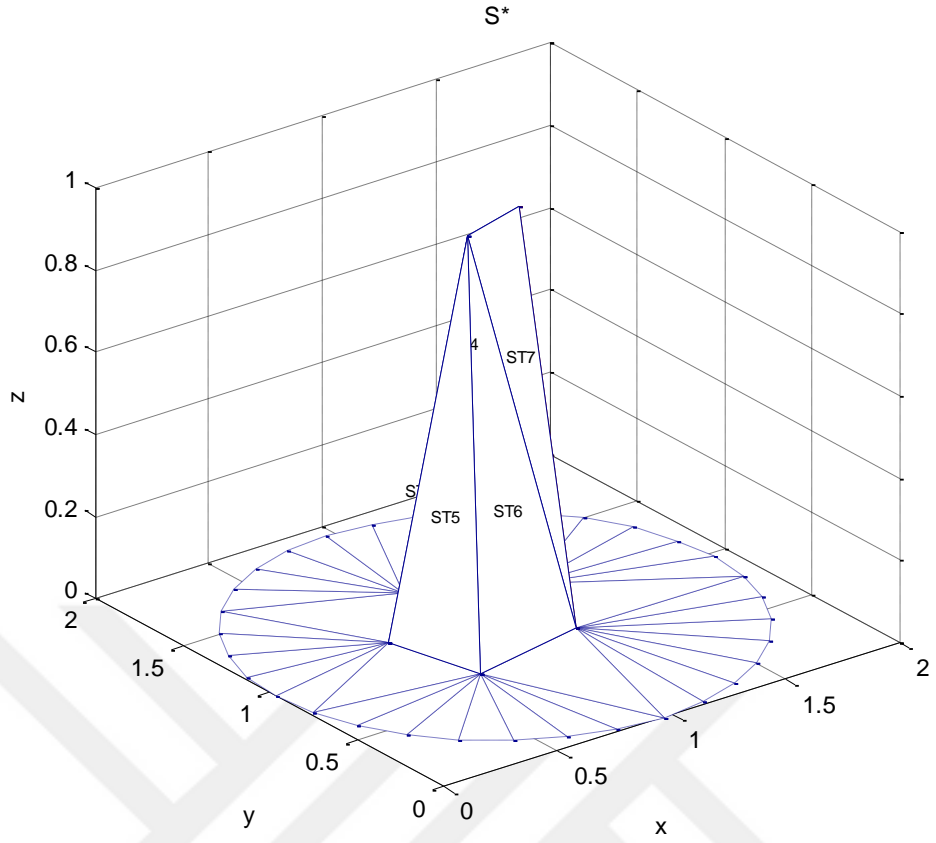


Şekil 4.20. *Surf1*'in  $n_v=16$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

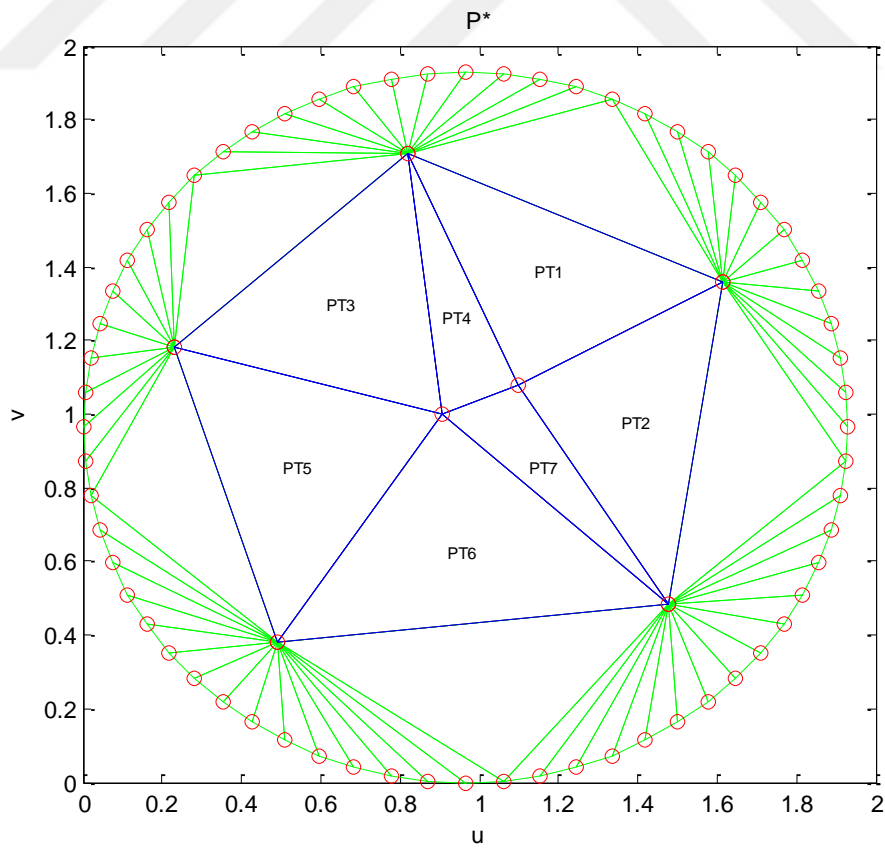




Şekil 4.21. *Surf1*'in  $n_v=32$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)



(a)

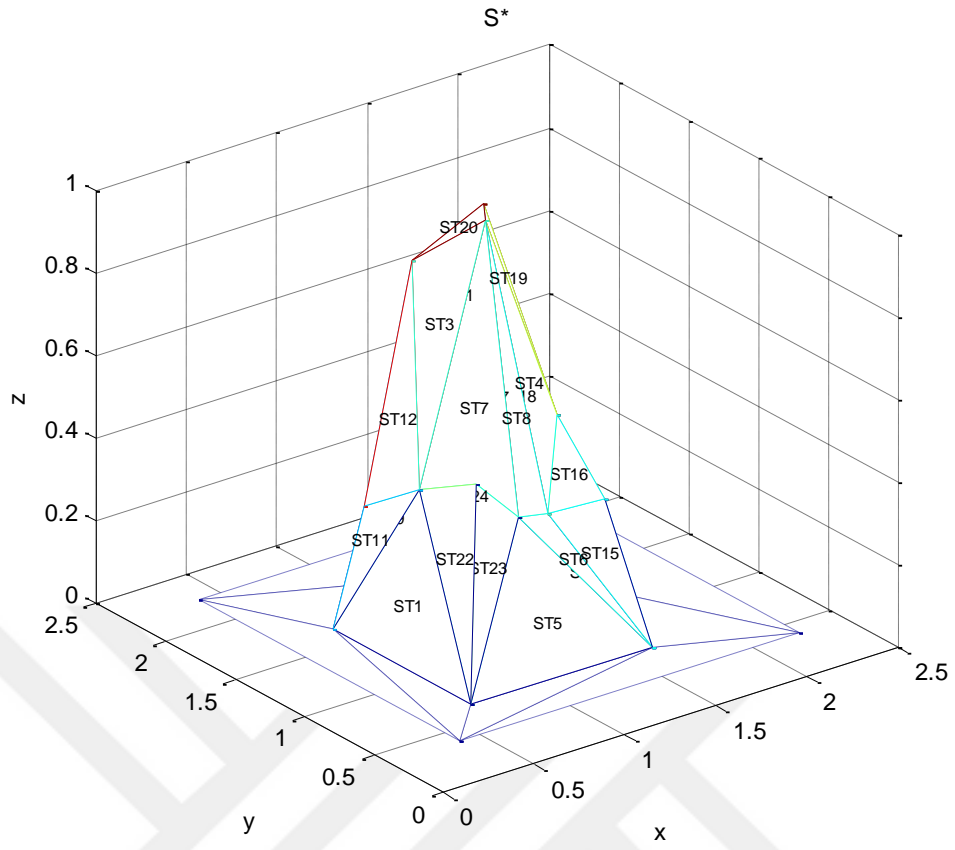


(b)

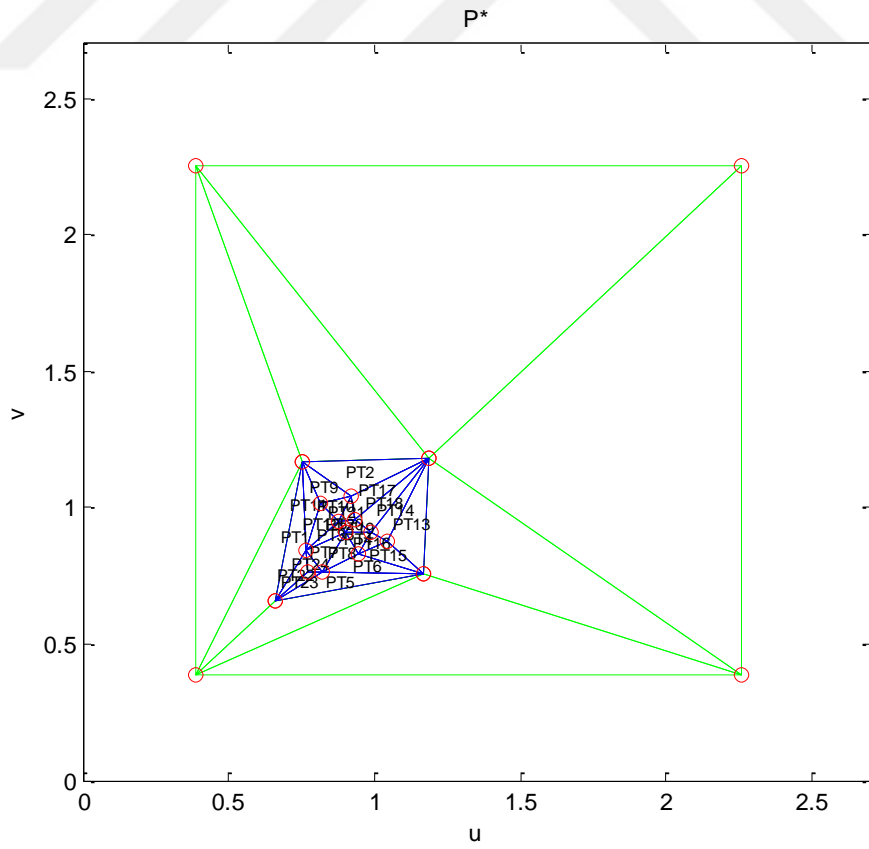
Şekil 4.22. *Surf1*'in  $n_v=64$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

İkinci örnek yüzey yaması (*Surf2*) için çeşitli çözünürlüklerde sanal sınır poligonları üretilerek izdüşüm yapılmıştır. Anılan yüzey yaması için sanal düğüm sayıları  $n_v=4$ ,  $n_v=6$ ,  $n_v=16$ ,  $n_v=32$  ve  $n_v=64$  olacak şekilde elde edilen ilaveli yüzey yamaları ve bunların düzlemsel karşılıkları Şekil 4.23-28'de verilmiştir. (Bu yüzey yaması için tüm izdüşüm sonuçları 50 iterasyon sonucunda elde edilmiştir.)



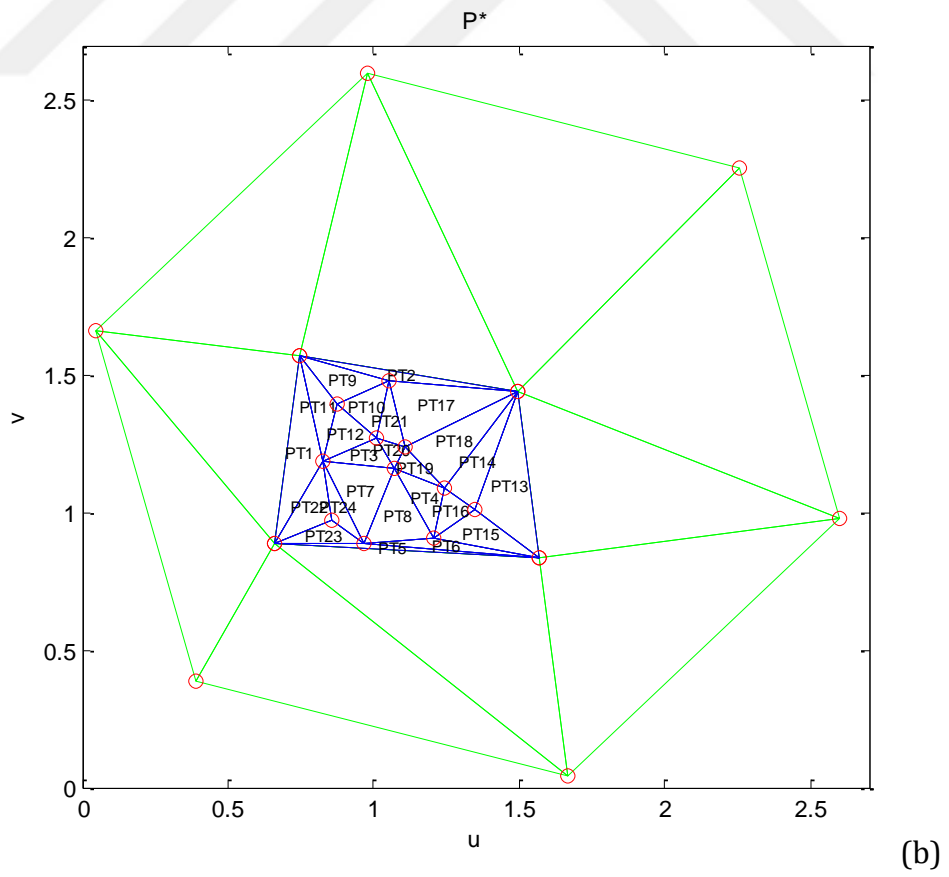
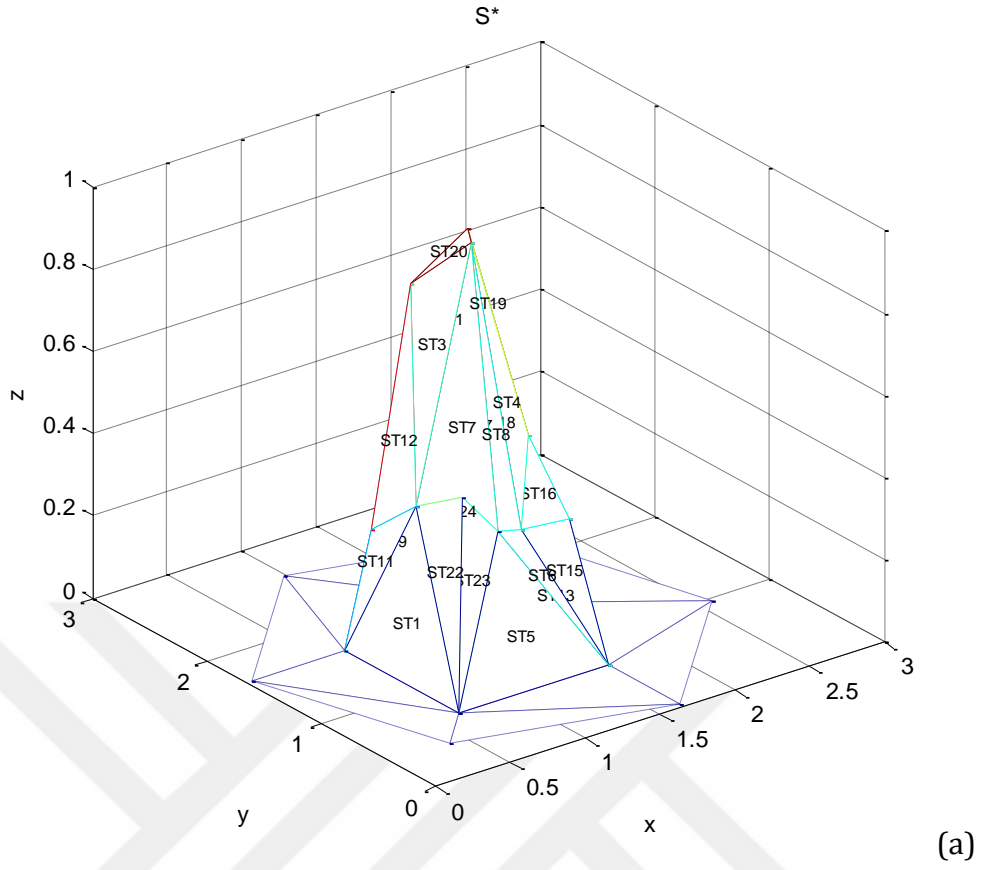


(a)

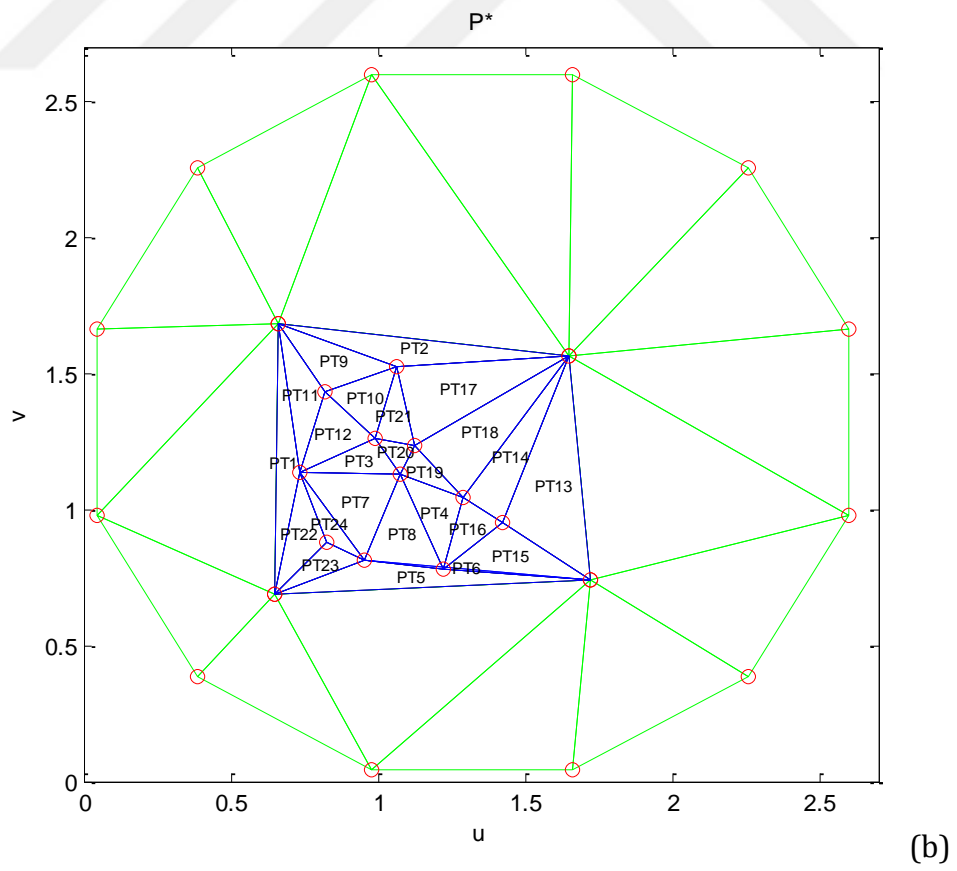
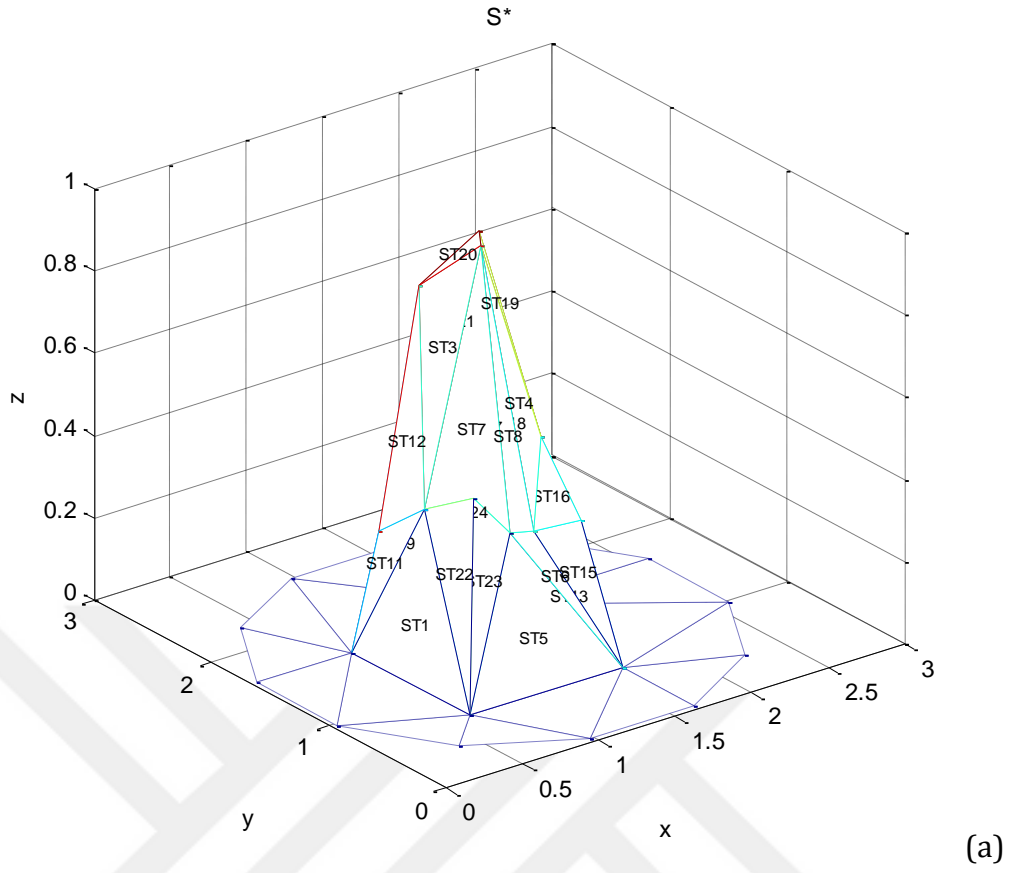


(b)

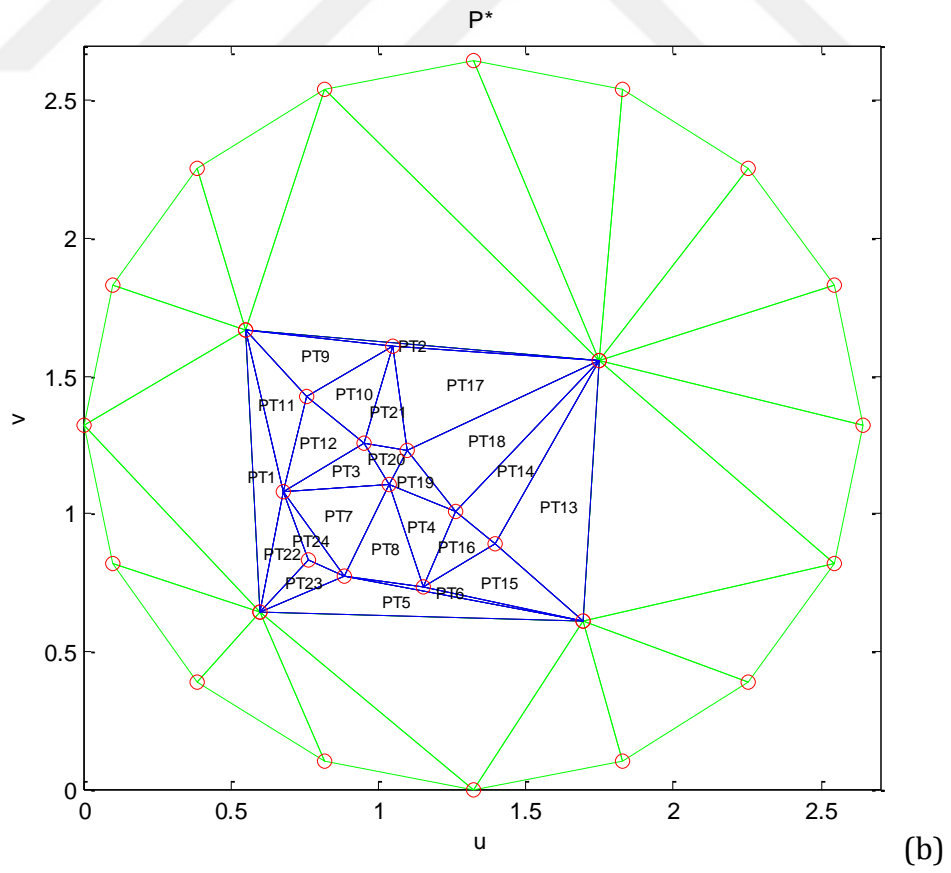
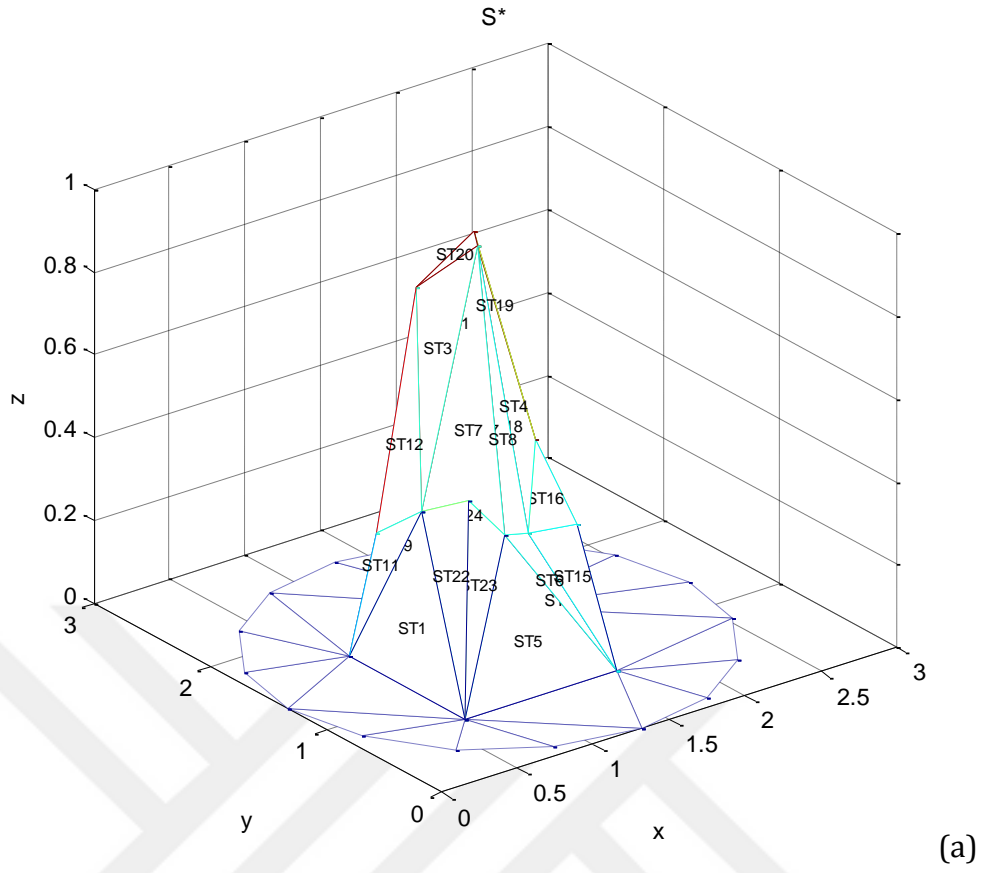
Şekil 4.23. *Surf2*'in  $n_v=4$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)



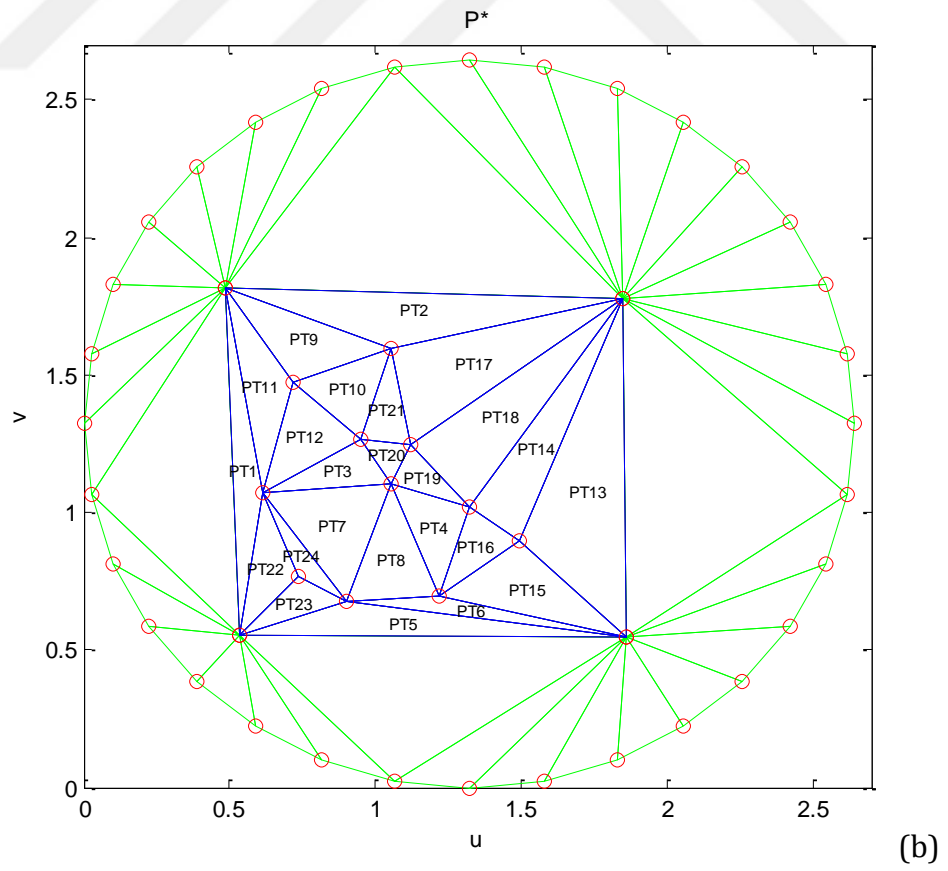
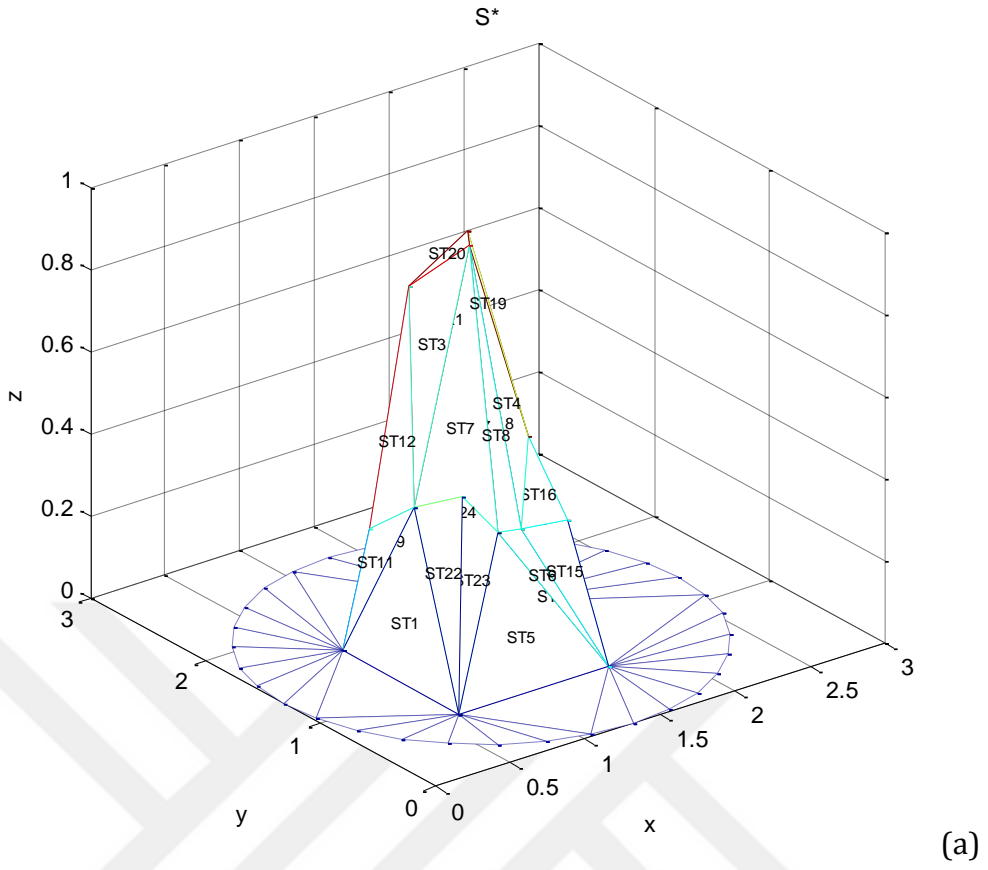
Şekil 4.24. *Surf2*'in  $n_v=6$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)



Şekil 4.25. *Surf2*'in  $n_v=12$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

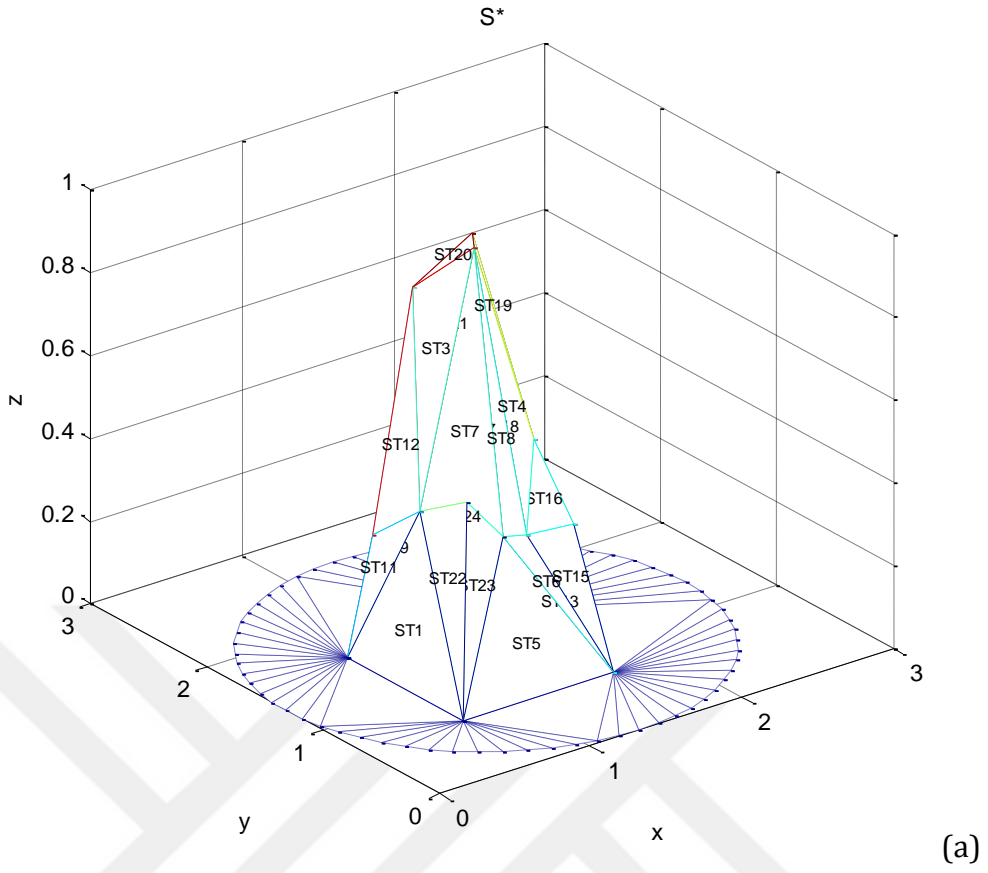


Şekil 4.26.  $Surf2$ 'in  $n_v=16$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

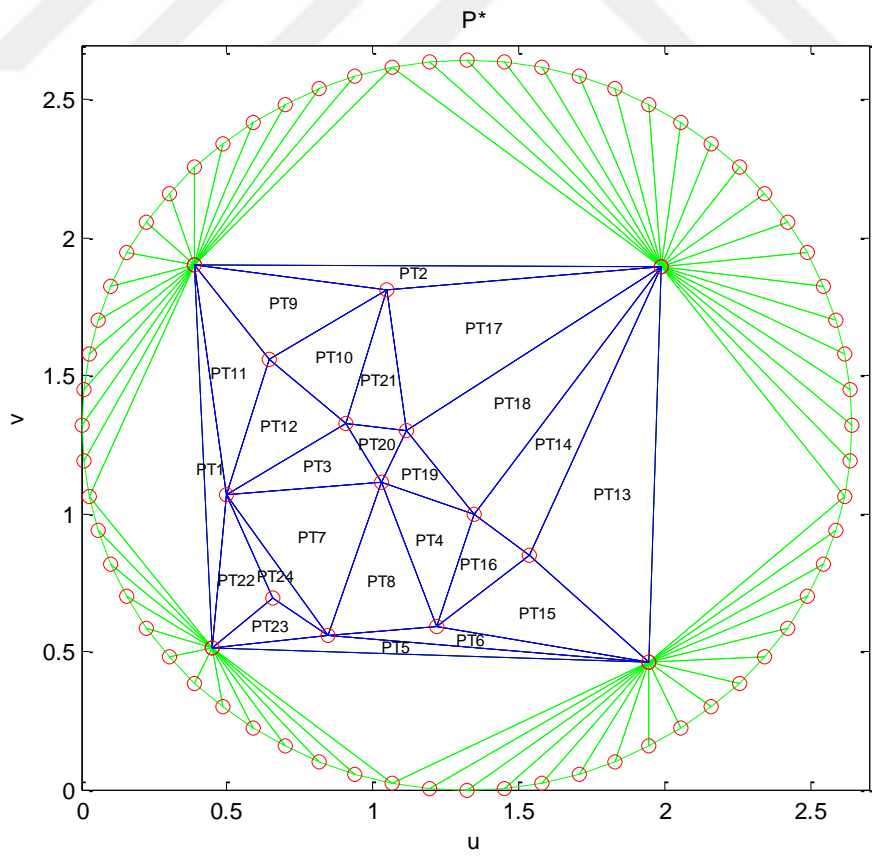


Şekil 4.27. *Surf2*'in  $n_v=32$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)





(a)

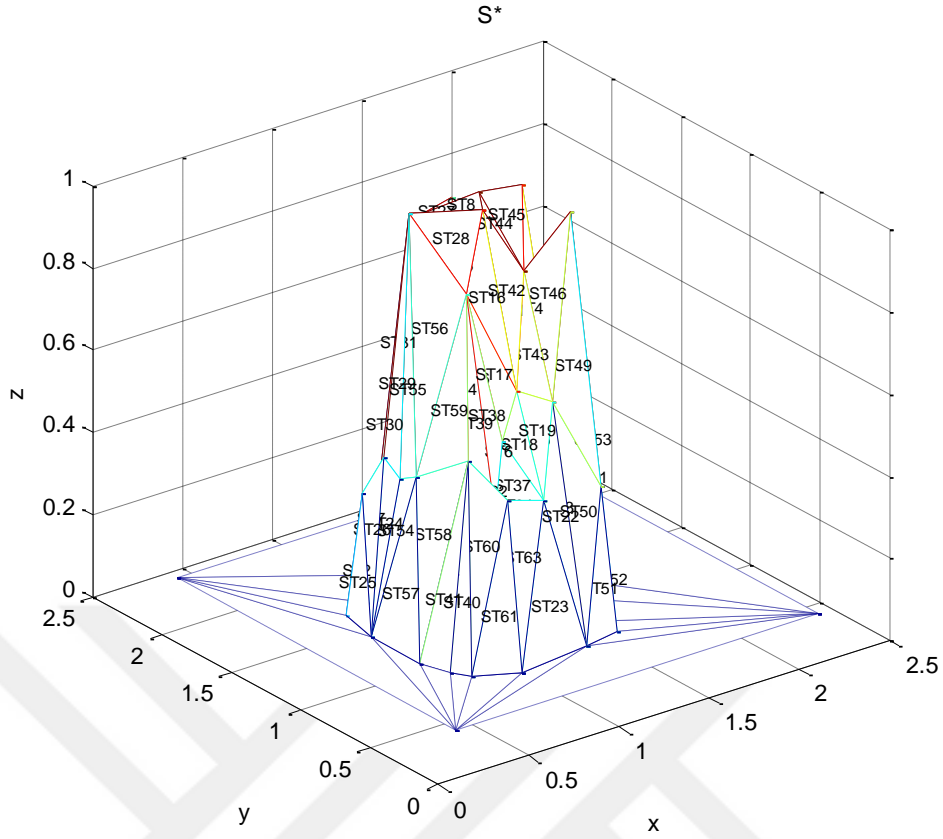


(b)

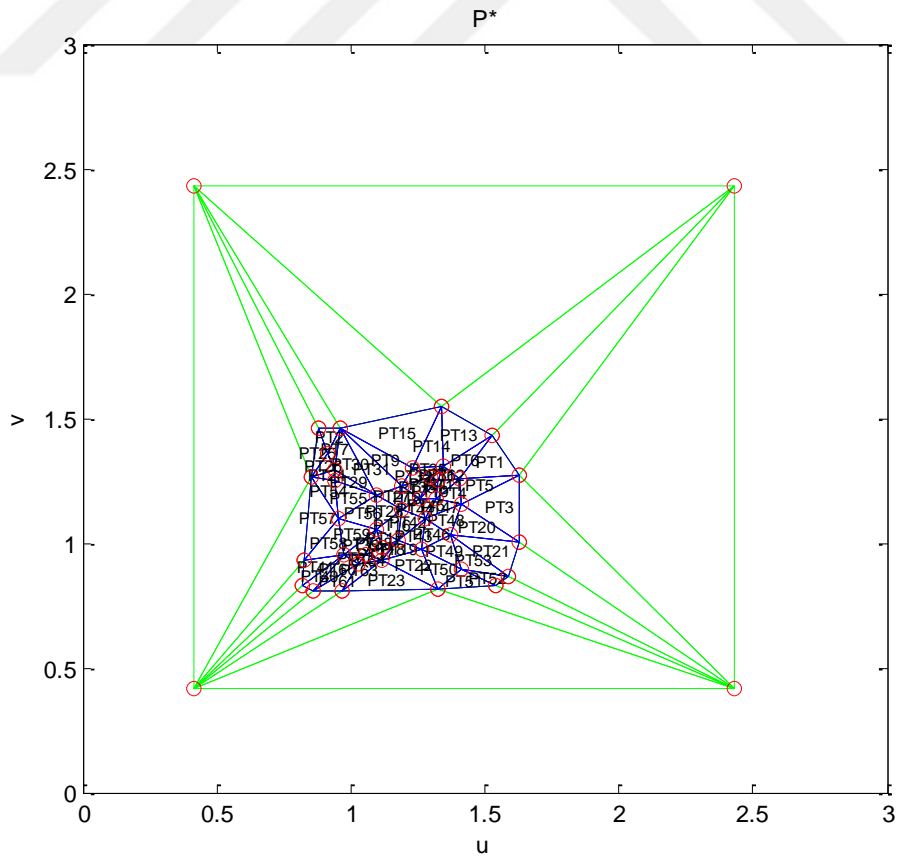
Şekil 4.28.  $Surf2$ 'in  $n_v=64$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

Üçüncü örnek yüzey yaması (*Surf3*) için çeşitli çözünürlüklerde sanal sınır poligonları üretilerek izdüşüm yapılmıştır. Anılan yüzey yaması için sanal düğüm sayıları  $n_v=4$ ,  $n_v=6$ ,  $n_v=8$ ,  $n_v=32$  ve  $n_v=64$  olacak şekilde elde edilen ilaveli yüzey yamaları ve bunların düzlemsel karşılıkları Şekil 4.29-34'te verilmiştir. (Bu yüzey yaması için tüm izdüşüm sonuçları 100 iterasyon sonucunda elde edilmiştir.)



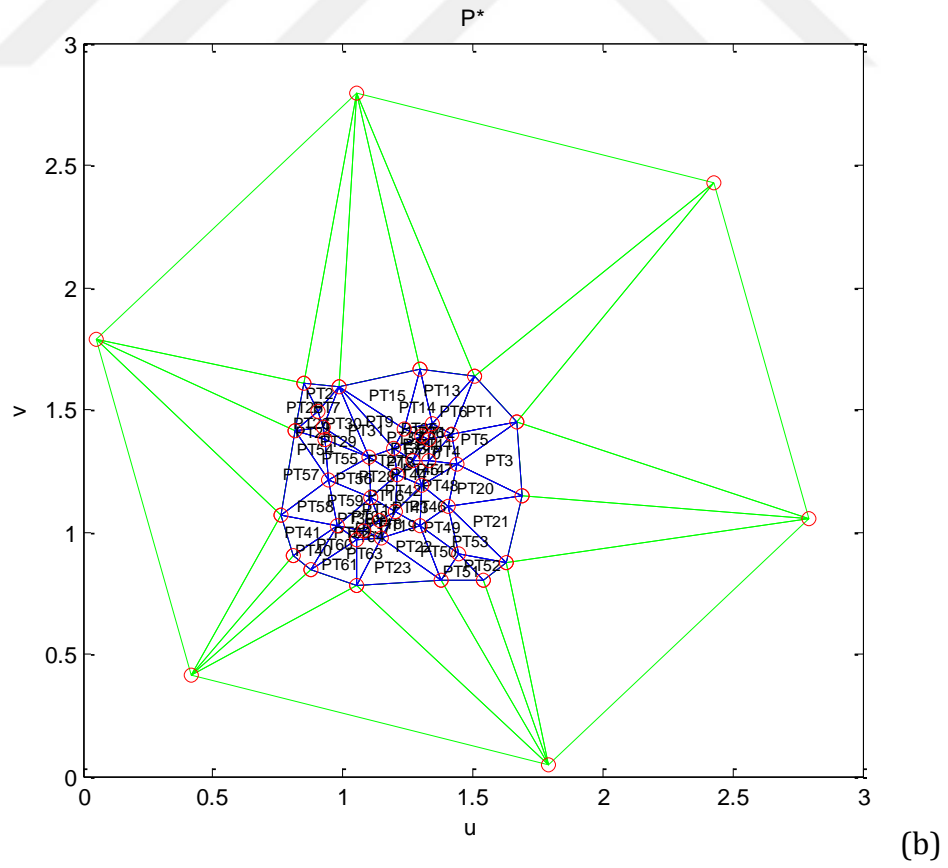
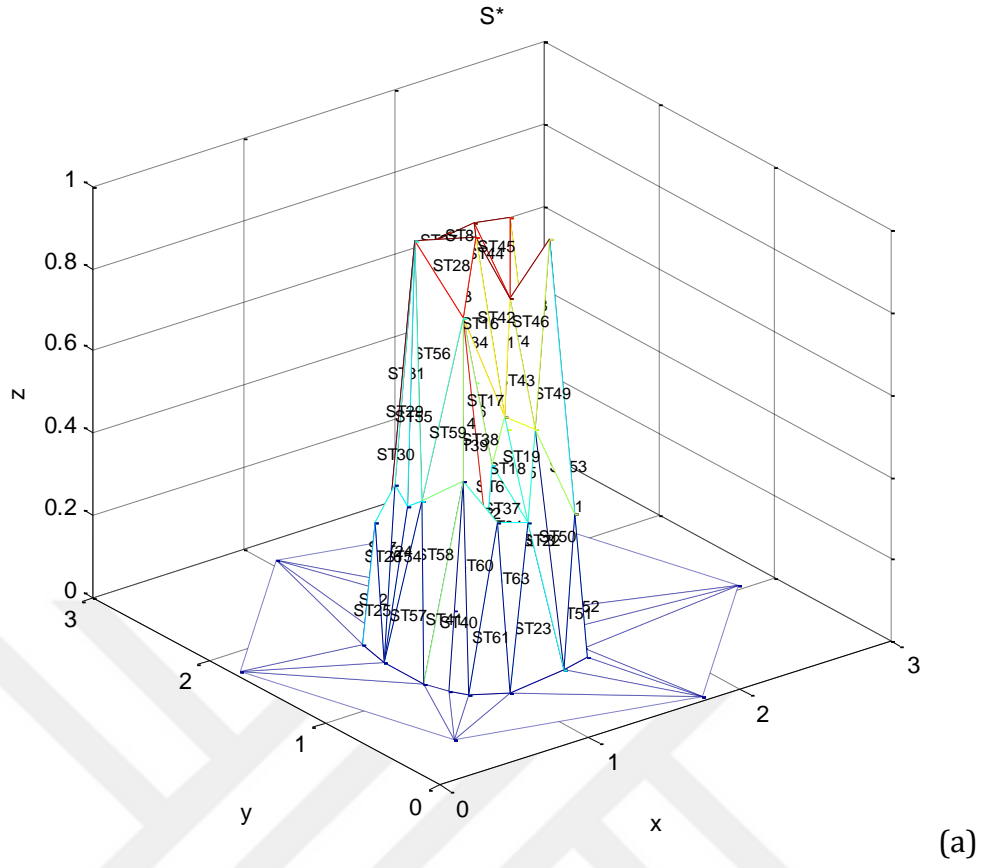


(a)

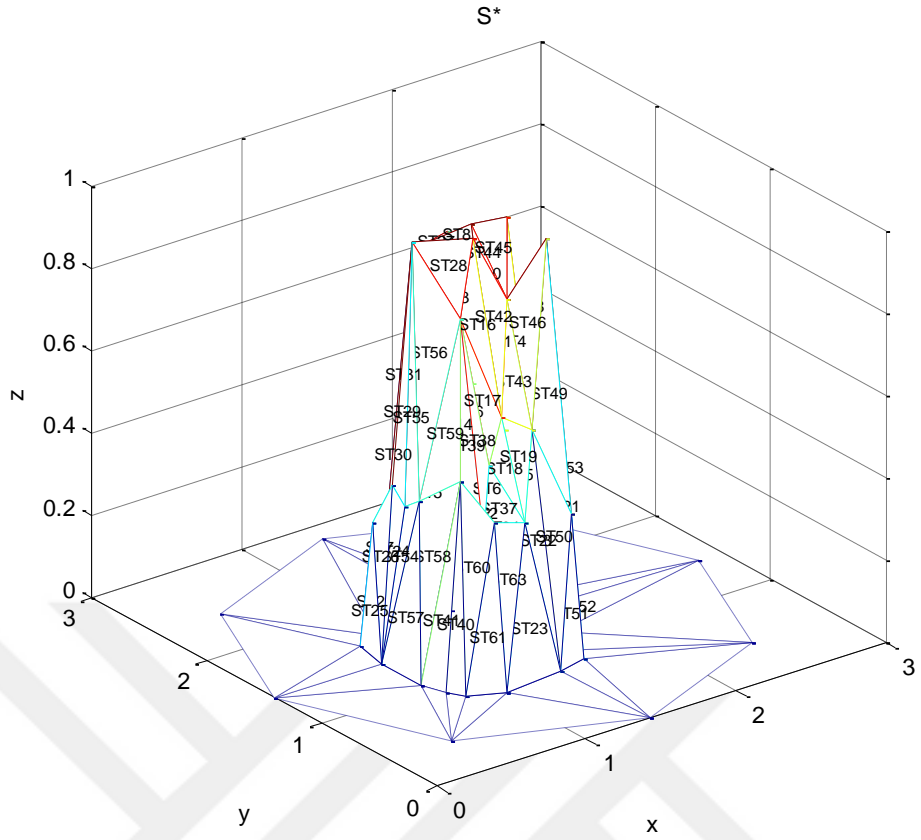


(b)

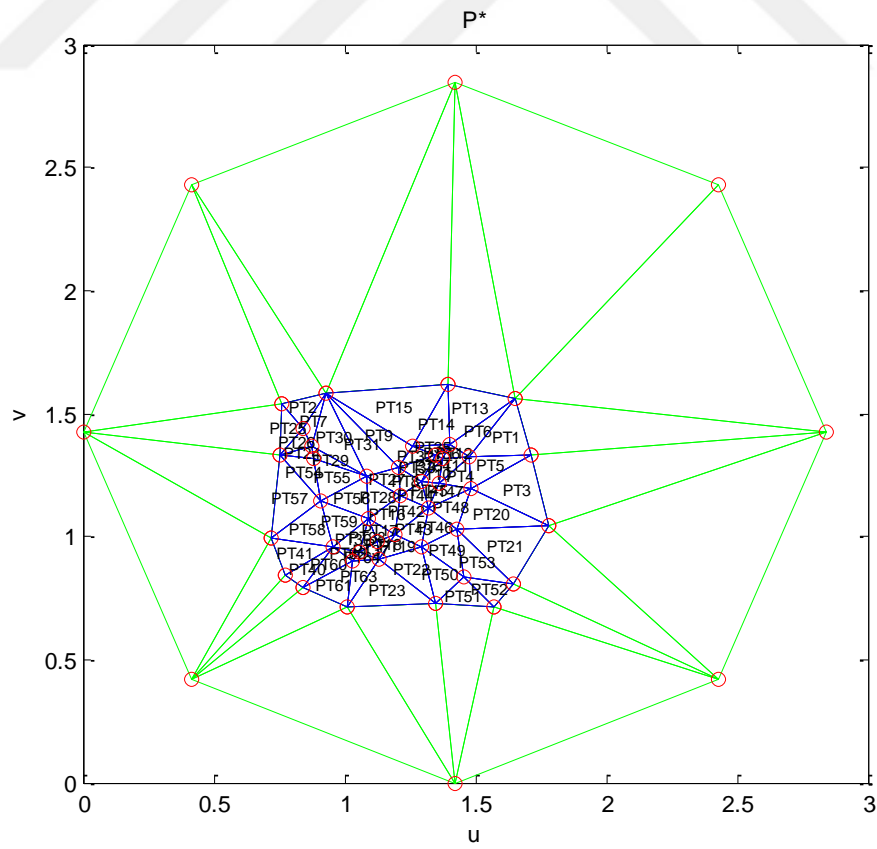
Şekil 4.29.  $Surf3$ 'ün  $n_v=4$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)



Şekil 4.30.  $Surf3$ 'ün  $n_v=6$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

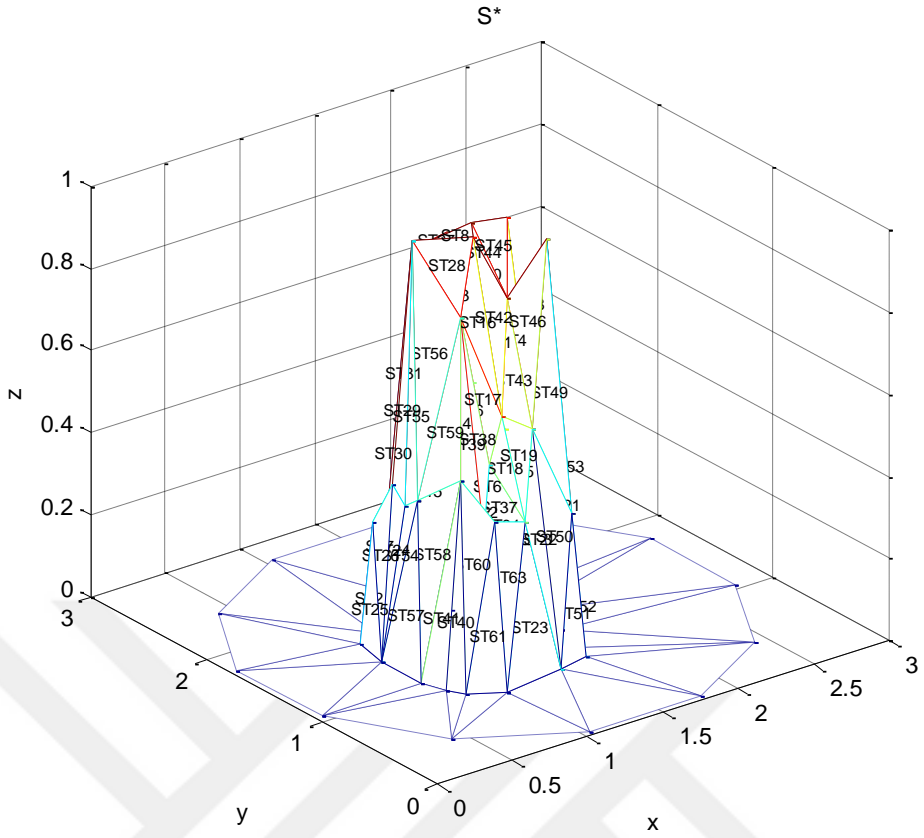


(a)

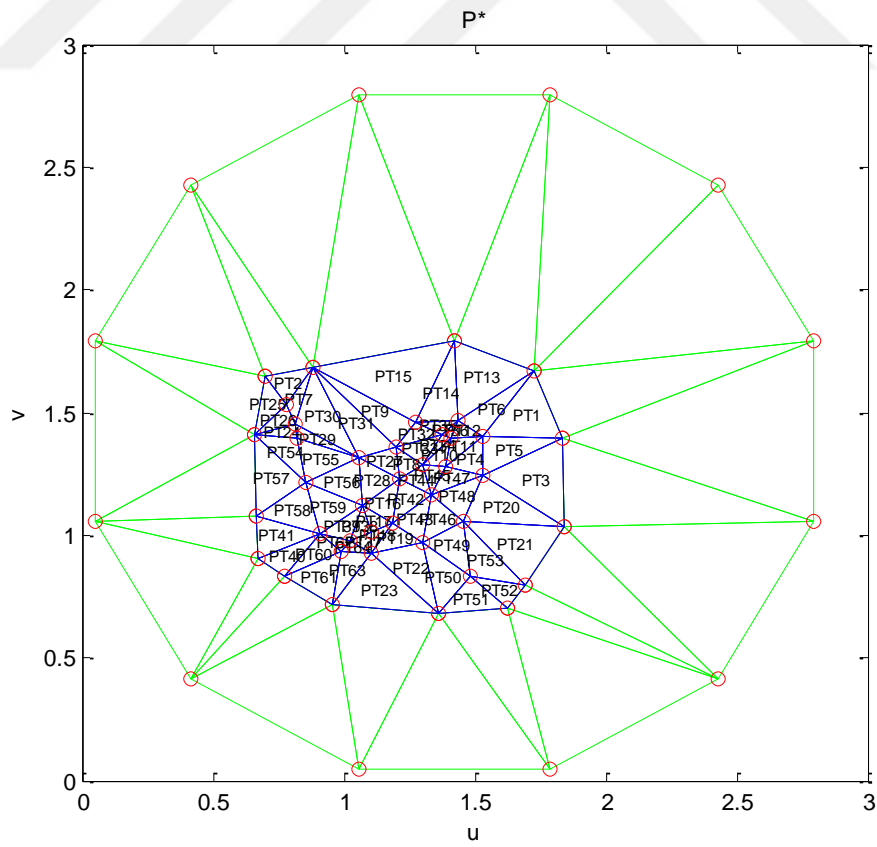


(b)

Şekil 4.31.  $Surf3'$ 'ün  $n_v=8$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)

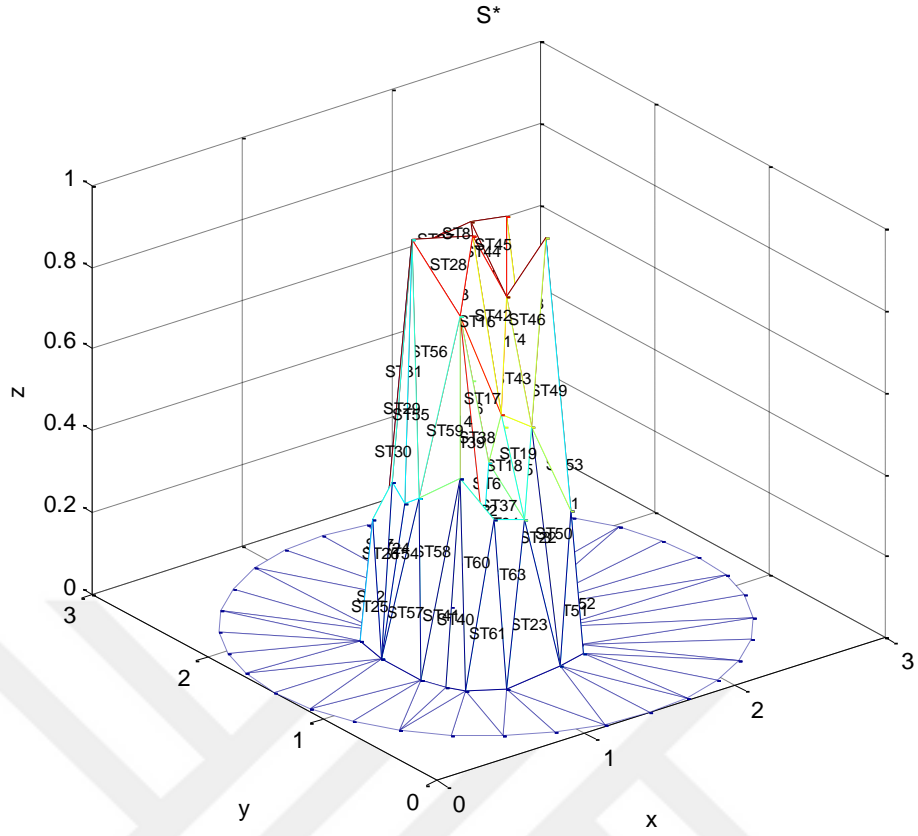


(a)

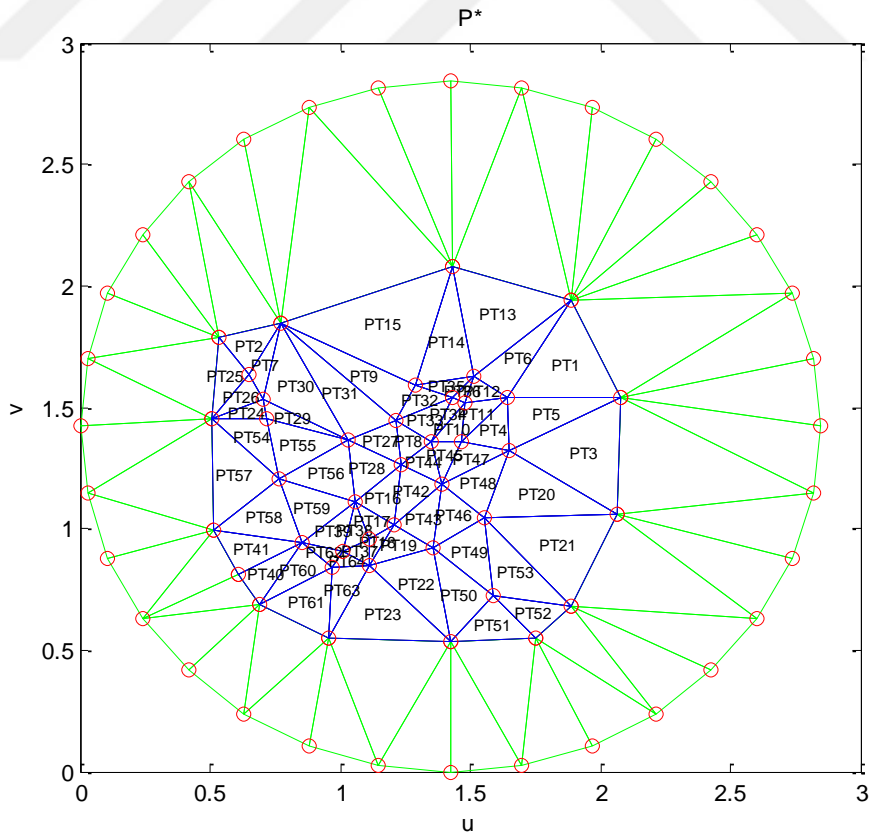


(b)

Şekil 4.32. *Surf3*'ün  $n_v=12$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)



(a)



(b)

Şekil 4.33.  $Surf3$ 'ün  $n_v=32$  için ilaveli yüzey yaması (a) ve düzlemsel karşılığı (b)



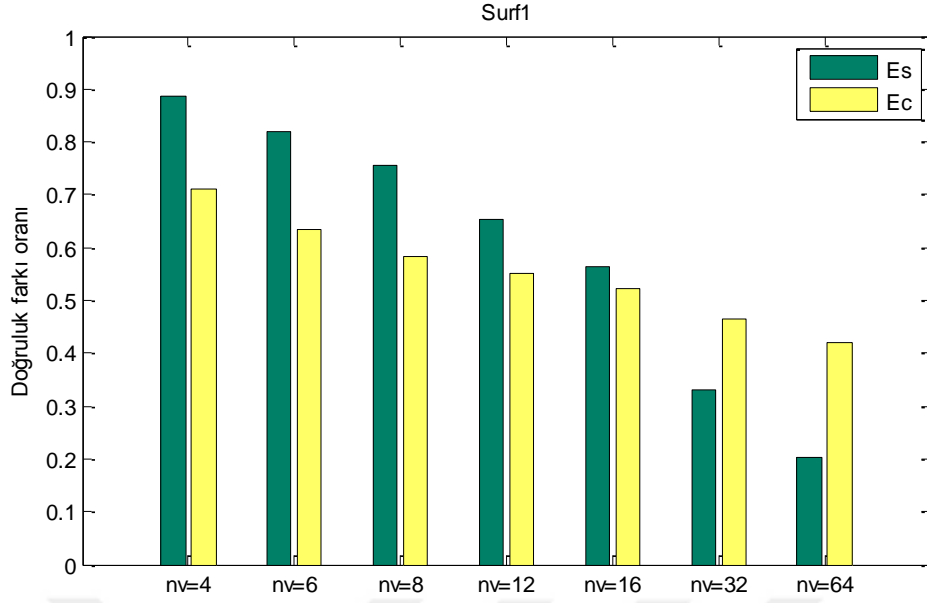


Şekil 4.17 ile 4.34 arasında izdüşüm sonuçları görsel olarak verilen (sanal düğüm ilaveli) yüzeylere ilişkin hesaplanan alan ve kenar-doğruluk fark değerleri Çizelge 4.1’de verilmiştir. Örnek yüzeylerin hepsi için sırasıyla 4, 6, 8, 12, 16, 32 ve 64 düğümlü sanal sınır poligonu oluşturularak bunların düzleme izdüşümleri gerçekleştirilmiştir.

Çizelge 4.1. Üç farklı yüzey için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları

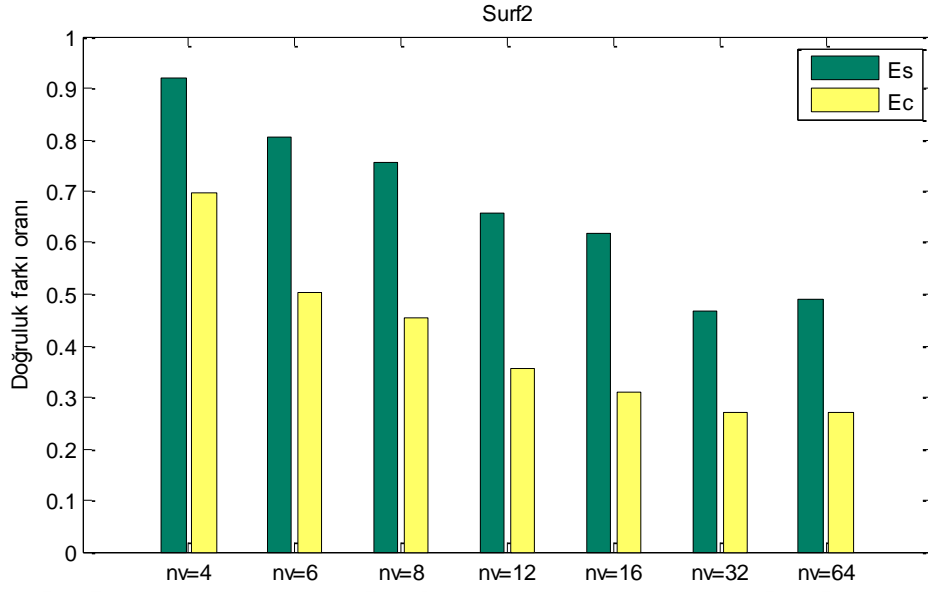
Yüzey	Surf1		Surf2		Surf3	
	Alan doğruluk farkı, $E_S$	Kenar doğruluk farkı, $E_C$	Alan doğruluk farkı, $E_S$	Kenar doğruluk farkı, $E_C$	Alan doğruluk farkı, $E_S$	Kenar doğruluk farkı, $E_C$
$n_v=4$	0.8875	0.7098	0.9211	0.6974	0.8400	0.6521
$n_v=6$	0.8183	0.6337	0.8056	0.5049	0.7825	0.6071
$n_v=8$	0.7570	0.5825	0.7565	0.4543	0.7420	0.5700
$n_v=12$	0.6544	0.5504	0.6579	0.3548	0.6513	0.5104
$n_v=16$	0.5627	0.5231	0.6178	0.3107	0.5911	0.4810
$n_v=32$	0.3312	0.4664	0.4685	0.2704	0.4341	0.3745
$n_v=64$	0.2038	0.4202	0.4906	0.2697	0.3351	0.2905

*Surf1* örnek yüzey yamasına ilişkin alan ve kenar-doğruluk farkları ayrıca Şekil 4.35’teki çubuk grafikte verilmiştir. Sanal sınır düğümü sayısı  $n_v=4$  iken alan-doğruluk farkı  $E_S=0.8875$  ve kenar-doğruluk farkı  $E_C=0.7098$  olarak hesaplanmıştır. Yüzeye ilişkin doğruluk farkları, sanal sınır düğümü sayısının artışıyla birlikte giderek azalarak  $n_v=64$  için  $E_S=0.2038$  ve  $E_C=0.4202$  olarak ölçülmüştür. Bu yüzeye ilişkin tüm sanal sınır düğümü değerleri için izdüşüm sonuçlarını hesaplamada kullanılan dinamik sınır ağı 10 iterasyon ile çalıştırılmıştır.



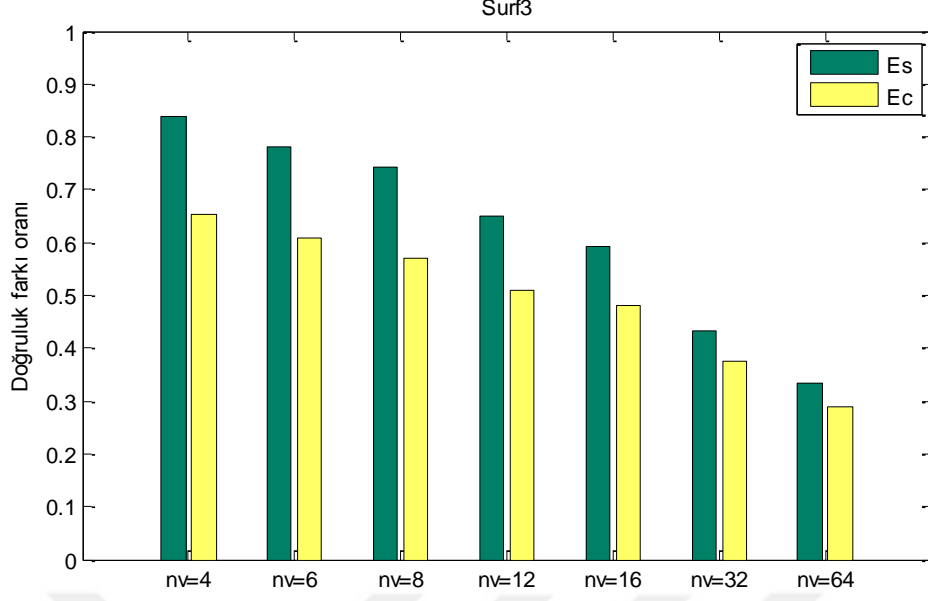
Şekil 4.35. *Surf1* yüzey yaması için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları

*Surf2* isimli örnek yüzeye ilişkin alan ve kenar-doğruluk farkları ayrıca Şekil 4.36'daki çubuk grafikte verilmiştir. Sanal sınır düğümü sayısı  $n_v=4$  iken alan-doğruluk farkı  $E_S=0.921$  ve kenar-doğruluk farkı  $E_C=0.6974$  olarak hesaplanmıştır. Yüzeye ilişkin doğruluk farkları, sanal sınır düğümü sayısının artışıyla birlikte giderek azalarak  $n_v=64$  için  $E_S=0.4906$  ve  $E_C=0.2697$  olarak ölçülmüştür. Bu yüzeye ilişkin tüm sanal sınır düğümü değerleri için izdüşüm sonuçlarını hesaplamada kullanılan dinamik sinir ağı 50 iterasyon ile çalıştırılmıştır.



Şekil 4.36. *Surf2* yüzey yaması için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları

*Surf3* isimli örnek yüzeye ilişkin alan ve kenar-doğruluk farkları ayrıca Şekil4.36'daki çubuk grafikte verilmiştir. Sanal sınır düğümü sayısı  $n_v=4$  iken alan-doğruluk farkı  $E_S=0.8400$  ve kenar-doğruluk farkı  $E_C=0.6521$  olarak hesaplanmıştır. Yüzeye ilişkin doğruluk farkları, sanal sınır düğümü sayısının artışıyla birlikte giderek azalarak  $n_v=64$  için  $E_S=0.3351$  ve  $E_C=0.2905$  olarak ölçülmüştür. Bu yüzeye ilişkin tüm sanal sınır düğümü değerleri için izdüşüm sonuçlarını hesaplamada kullanılan dinamik sinir ağı 100 iterasyon ile çalıştırılmıştır.



Şekil 4.37. *Surf3* yüzey yaması için çeşitli sanal düğüm sayıları kullanarak yapılan izdüşümlere ilişkin alan ve kenar doğruluk farkı oranları

#### 4.1.3. Lambda komşuluk ağırlık matrisinin izdüşüm sonucuna etkisi

Barisentrik izdüşüm teorisinin anlatıldığı bölümde (Bölüm 3.2) her bir düğüme ilişkin  $\lambda_{i,j}$  komşuluk ağırlık değerlerinin hesaplanma yönteminden bahsedilmiştir. Tutte tarafından graf çiziminde ortaya atılan ve daha sonra da Floater tarafından benimsenen temel yöntemde her bir düğüm için bu değer basitçe komşu sayısı ile ters orantılı olacak şekilde hesaplanır. Önerilen yöntemde Algoritma-2'nin 13. ve 14. adımda matematiksel ifadesi verilen değişikliklerle birlikte, bir düğüme ait  $\lambda_{i,j}$  değeri, onun komşu köşelerle arasındaki Öklid mesafesinin çarpmaya göre tersi kullanılarak hesaplanmıştır. (Bu konu ayrıntılı olarak Bölüm 3.5.2'de anlatılmıştır.) Bu sayede 3B uzaydaki yerel geometri bilgisi düzlemsel geliştirmenin yapıldığı parametrik uzaya etkin bir biçimde aktarılmış olur.

Bu amaçla önceki bölümde verilen örnek yüzeylerin hepsi için temel yöntem, önerilen yöntemle aynı şartlar altında icra edilmiştir ve sonuçlar Çizelge 4.2-Çizelge 4.4'te verilmiştir.

Çizelge 4.2. *Surf1* örnek yüzey yaması için temel yöntem ve önerilen yöntem izdüşüm sonuçları

<b>Yüzey: <i>Surf1</i></b>					
<b>10 iterasyon</b>		<b>Temel Yöntem</b>		<b>Önerilen Yöntem</b>	
	<b>Sanal düğüm sayısı</b>	<b>Alan doğruluk farkı, <math>E_S</math></b>	<b>Kenar doğruluk farkı, <math>E_C</math></b>	<b>Alan doğruluk farkı, <math>E_S</math></b>	<b>Kenar doğruluk farkı, <math>E_C</math></b>
	$n_v=4$	0.8903	0.6355	0.8875	0.7098
	$n_v=6$	0.8425	0.5678	0.8183	0.6337
	$n_v=8$	0.8212	0.5462	0.7570	0.5825
	$n_v=12$	0.7861	0.5146	0.6544	0.5504
	$n_v=16$	0.7649	0.5020	0.5627	0.5231
	$n_v=32$	0.7637	0.5122	0.3312	0.4664
	$n_v=64$	0.8247	0.5676	0.2038	0.4202

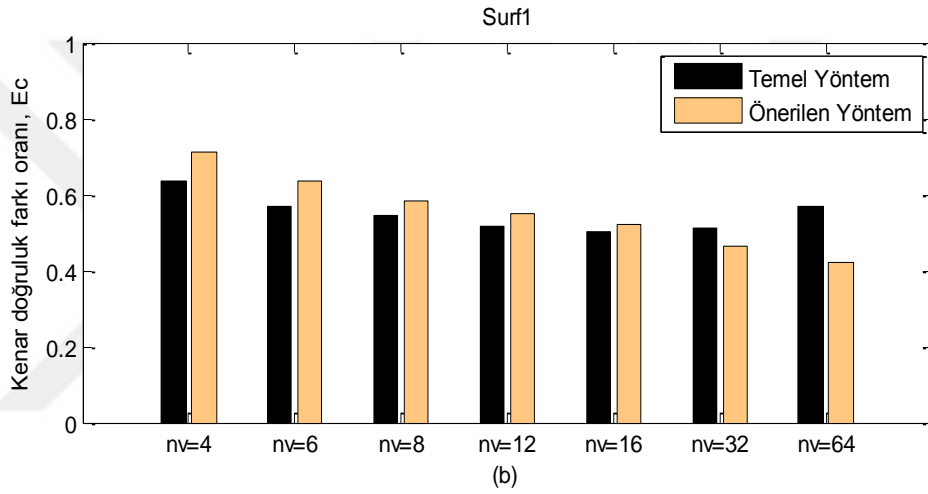
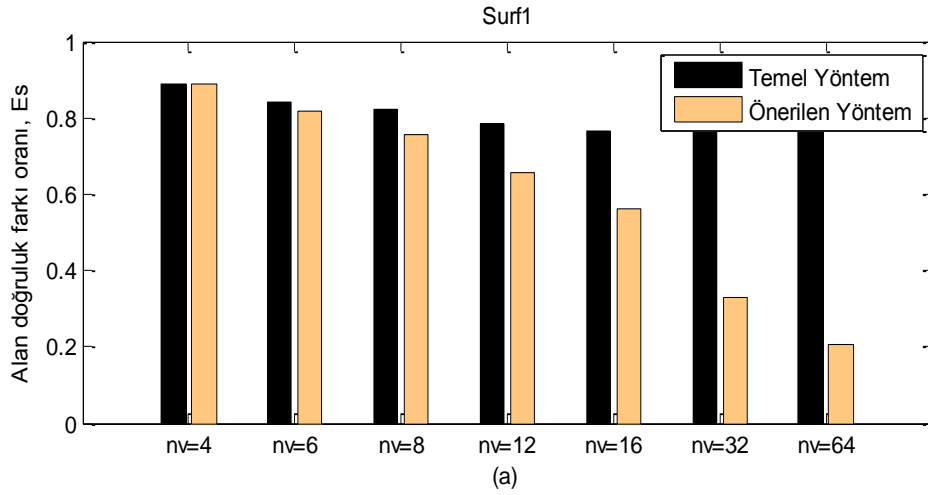
Çizelge 4.3. *Surf2* örnek yüzey yaması için temel yöntem ve önerilen yöntem izdüşüm sonuçları

<b>Yüzey: <i>Surf2</i></b>					
<b>50 iterasyon</b>		<b>Temel Yöntem</b>		<b>Önerilen Yöntem</b>	
	<b>Sanal düğüm sayısı</b>	<b>Alan doğruluk farkı, <math>E_S</math></b>	<b>Kenar doğruluk farkı, <math>E_C</math></b>	<b>Alan doğruluk farkı, <math>E_S</math></b>	<b>Kenar doğruluk farkı, <math>E_C</math></b>
	$n_v=4$	0.9164	0.6632	0.9211	0.6974
	$n_v=6$	0.7747	0.4653	0.8056	0.5049
	$n_v=8$	0.7382	0.4191	0.7565	0.4543
	$n_v=12$	0.6556	0.3634	0.6579	0.3548
	$n_v=16$	0.6597	0.3428	0.6178	0.3107
	$n_v=32$	0.5761	0.3739	0.4685	0.2704
	$n_v=64$	0.5825	0.3437	0.4906	0.2697

Çizelge 4.4. *Surf3* örnek yüzey yaması için temel yöntem ve önerilen yöntem izdüşüm sonuçları

Yüzey: <i>Surf3</i>					
	Sanal düğüm sayısı	Temel Yöntem		Önerilen Yöntem	
		Alan doğruluk farkı, $E_S$	Kenar doğruluk farkı, $E_C$	Alan doğruluk farkı, $E_S$	Kenar doğruluk farkı, $E_C$
100 iterasyon	$n_v=4$	0.8535	0.6141	0.8400	0.6521
	$n_v=6$	0.8148	0.5669	0.7825	0.6071
	$n_v=8$	0.7834	0.5456	0.7420	0.5700
	$n_v=12$	0.7403	0.5042	0.6513	0.5104
	$n_v=16$	0.7082	0.4866	0.5911	0.4810
	$n_v=32$	0.6665	0.4473	0.4341	0.3745
	$n_v=64$	0.6623	0.4253	0.3351	0.2905

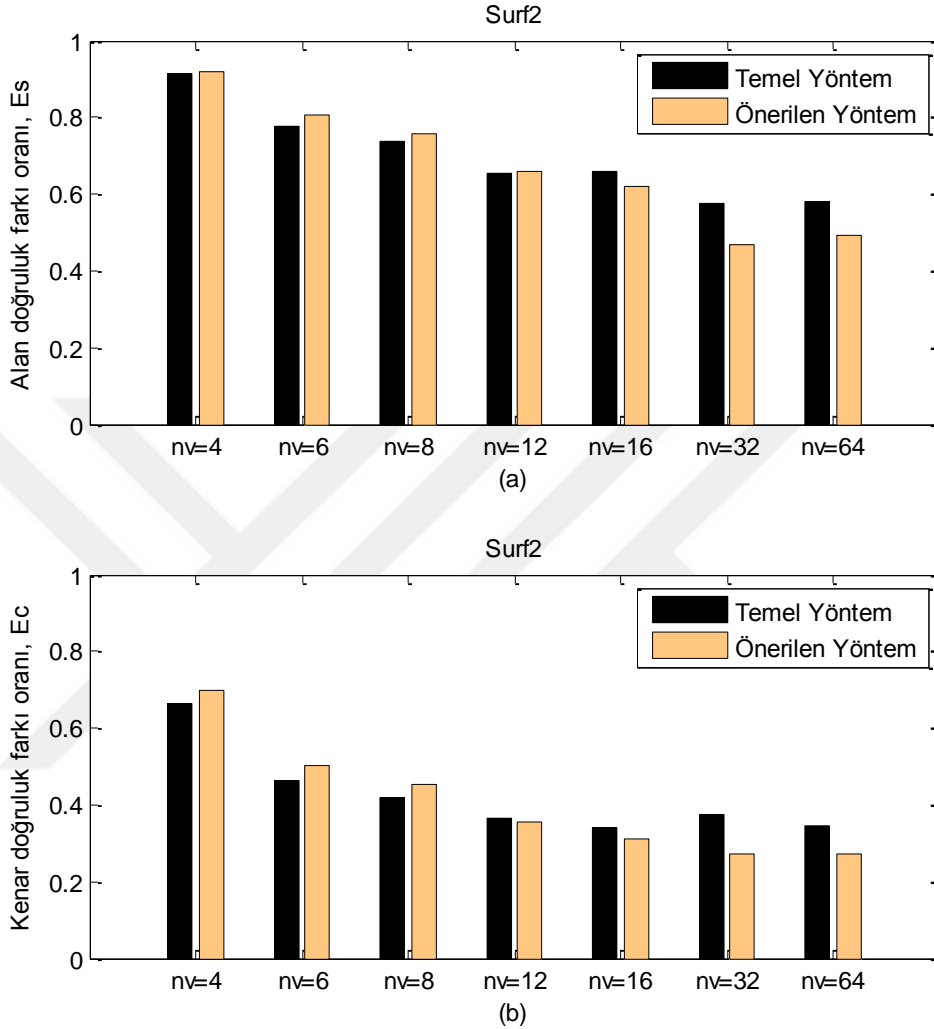
*Surf1* örnek yüzey yamasına ilişkin Çizelge 4.2’de verilen alan ve kenar-doğruluk farkları ayrıca Şekil 4.38’deki çubuk grafikte görsel olarak sunulmuştur. Bu yüzey yaması için  $n_v$  sanal sınır düğüm sayısı, 4’ten 64’e kadar belirli aralıklarla arttırılarak sanal sınır poligonunun çeşitli çözünürlükte olması sağlanmıştır. Her bir durumda bahse konu olan yüzey yaması için temel yöntemin ve önerilen yöntemin başlangıç izdüşüm sonuçları, alan ve kenar-doğruluk farkı cinsinden çubuk grafikte gözükmemektedir. Bu yüzey yaması için alan-doğruluk farkı ( $E_S$ ) temel yöntemde 0.8903 ile başlayıp 0.8247’e kadar gerilerken, önerilen yöntemde ise 0.8875 ile başlayıp 0.2038’ya kadar gerilemiştir. Benzer bir şekilde, kenar-doğruluk farkı ( $E_C$ ) temel yöntemde 0.6355 ile başlayıp 0.5676’e kadar gerilerken, önerilen yöntemde ise 0.7098 ile başlayıp 0.4202’ya kadar gerilemiştir. Bu yüzey yaması için kıyaslanan iki yöntem için de iterasyon sayısı 10 olarak alınmıştır.



Şekil 4.38. *Surf1* yüzeyine ilişkin temel yöntem ve önerilen yönteme ait alan doğruluk farkı kıyaslaması (a) ve kenar doğruluk farkı kıyaslaması (b)

*Surf2* örnek yüzey yamasına ilişkin Çizelge 4.3'te verilen alan ve kenar-doğruluk farkları ayrıca Şekil 4.39'daki çubuk grafikte görsel olarak sunulmuştur. Bu yüzey yaması için  $n_v$  sanal sınır düğüm sayısı, 4'ten 64'e kadar belirli aralıklarla artırılarak sanal sınır poligonunun çeşitli çözünürlükte olması sağlanmıştır. Her bir durumda bahse konu olan yüzey yaması için temel yöntemin ve önerilen yöntemin başlangıç izdüşüm sonuçları, alan ve kenar-doğruluk farkı cinsinden çubuk grafikte gözükmemektedir. Bu yüzey yaması için alan-doğruluk farkı temel yöntemde 0.9164 ile başlayıp 0.5825'e kadar gerilerken, önerilen yöntemde ise 0.9211 ile başlayıp 0.4906'ya kadar gerilemiştir. Benzer bir şekilde, kenar-doğruluk farkı temel yöntemde 0.6632 ile başlayıp 0.3437'e kadar gerilerken, önerilen yöntemde ise 0.6974 ile başlayıp 0.2697'ye kadar gerilemiştir. Bu

yüzey yaması için kıyaslanan iki yöntem için de iterasyon sayısı 50 olarak alınmıştır.

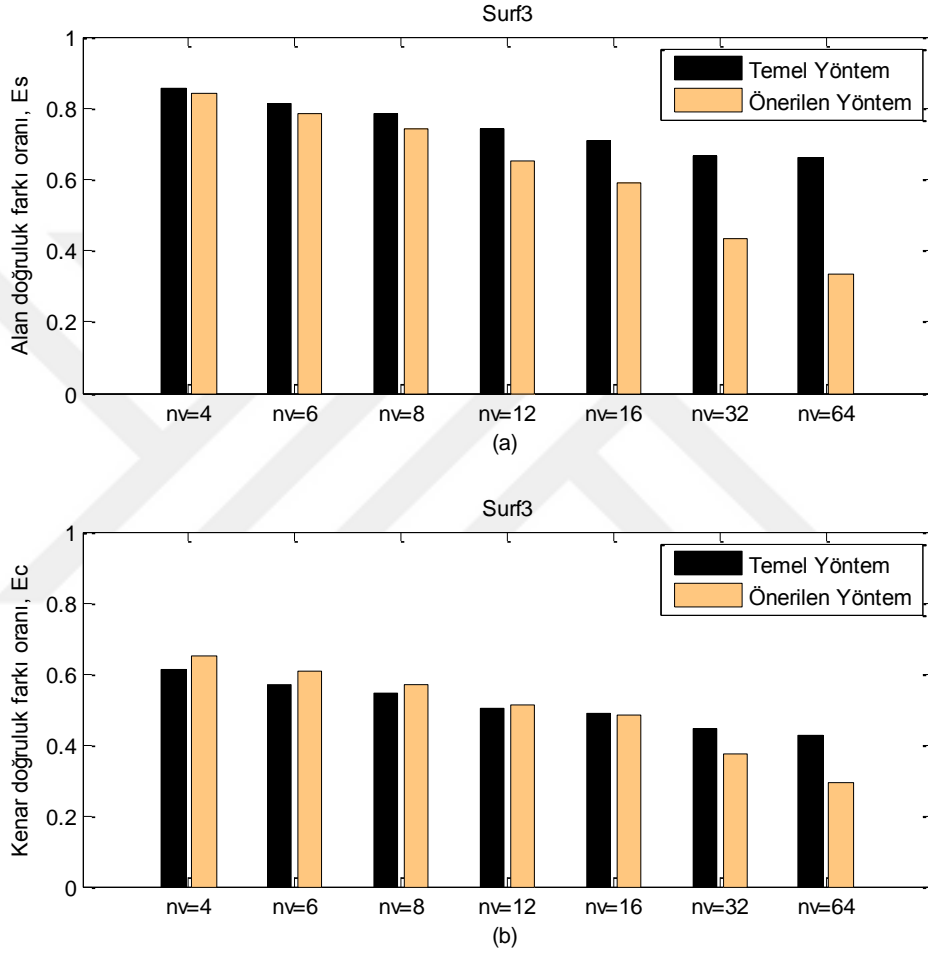


Şekil 4.39. *Surf2* yüzeyine ilişkin temel yöntem ve önerilen yöntemle ait alan doğruluk farkı kıyaslaması (a) ve kenar doğruluk farkı kıyaslaması (b)

*Surf3* örnek yüzey yamasına ilişkin Çizelge 4.4'te verilen alan ve kenar-doğruluk farkları ayrıca Şekil 4.40'daki çubuk grafikte görsel olarak sunulmuştur. Bu yüzey yaması için  $n_v$  sanal sınır düğüm sayısı, 4'ten 64'e kadar belirli aralıklarla arttırılarak sanal sınır poligonunun çeşitli çözünürlükte olması sağlanmıştır. Her bir durumda bahse konu olan yüzey yaması için temel yöntemin ve önerilen yöntemin başlangıç izdüşüm sonuçları, alan ve kenar-doğruluk farkı cinsinden çubuk grafikte gözükmemektedir. Bu yüzey yaması için alan-doğruluk farkı temel yöntemde 0.8535 ile başlayıp 0.6623'e kadar gerilerken, önerilen yöntemde ise

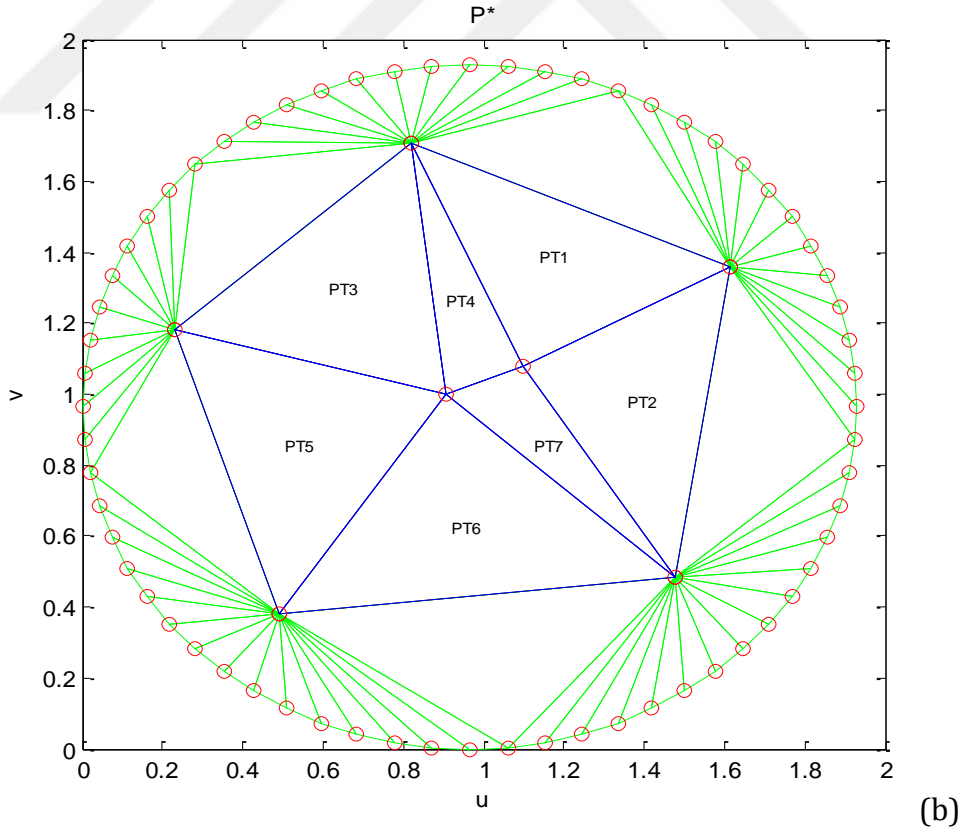
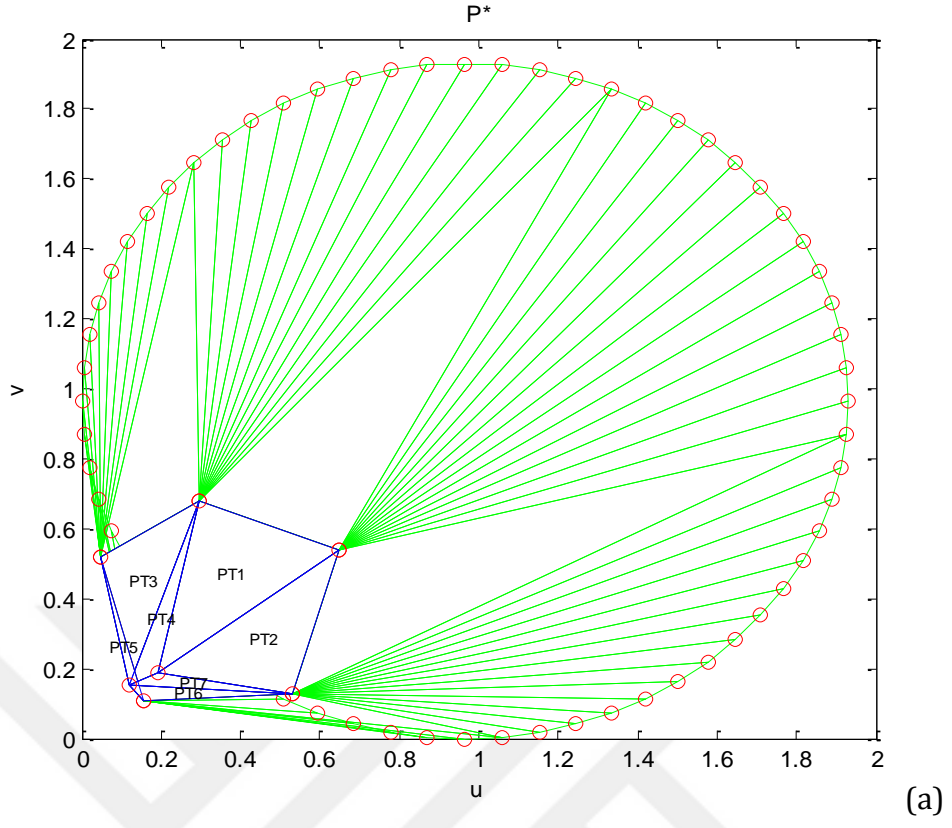


0.8400 ile başlayıp 0.3351'e kadar gerilemiştir. Benzer bir şekilde, kenar-doğruluk farkı temel yöntemde 0.6141 ile başlayıp 0.4253'e kadar gerilerken, önerilen yöntemde ise 0.6521 ile başlayıp 0.2905'ya kadar gerilemiştir. Bu yüzey yaması için kıyaslanan iki yöntem için de iterasyon sayısı 100 olarak alınmıştır.

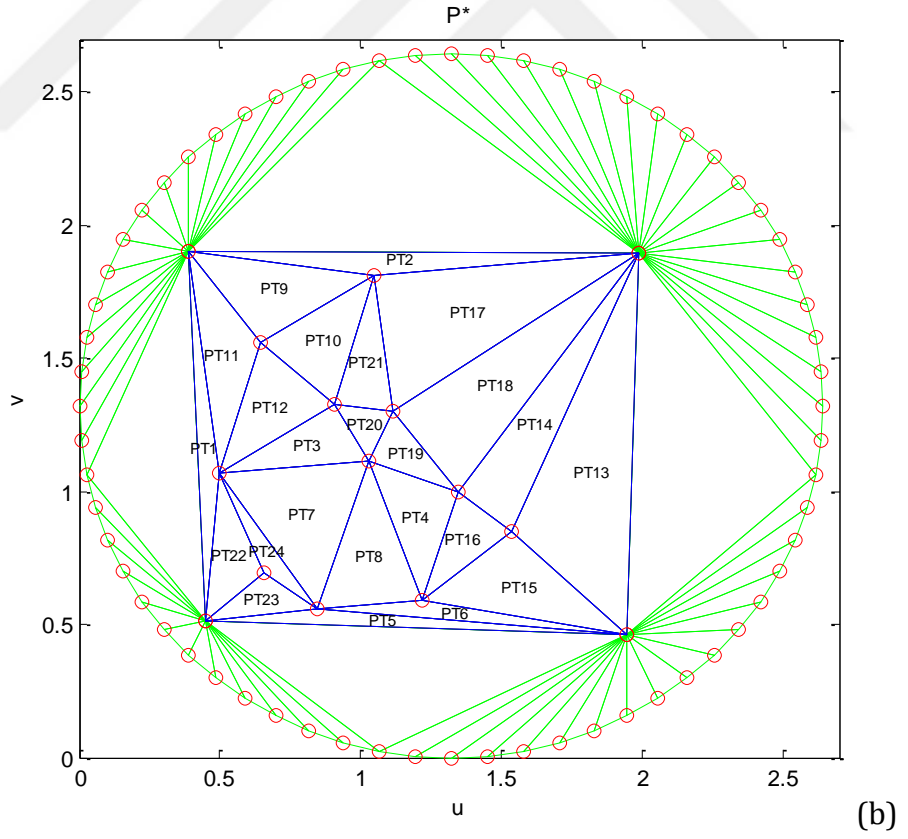
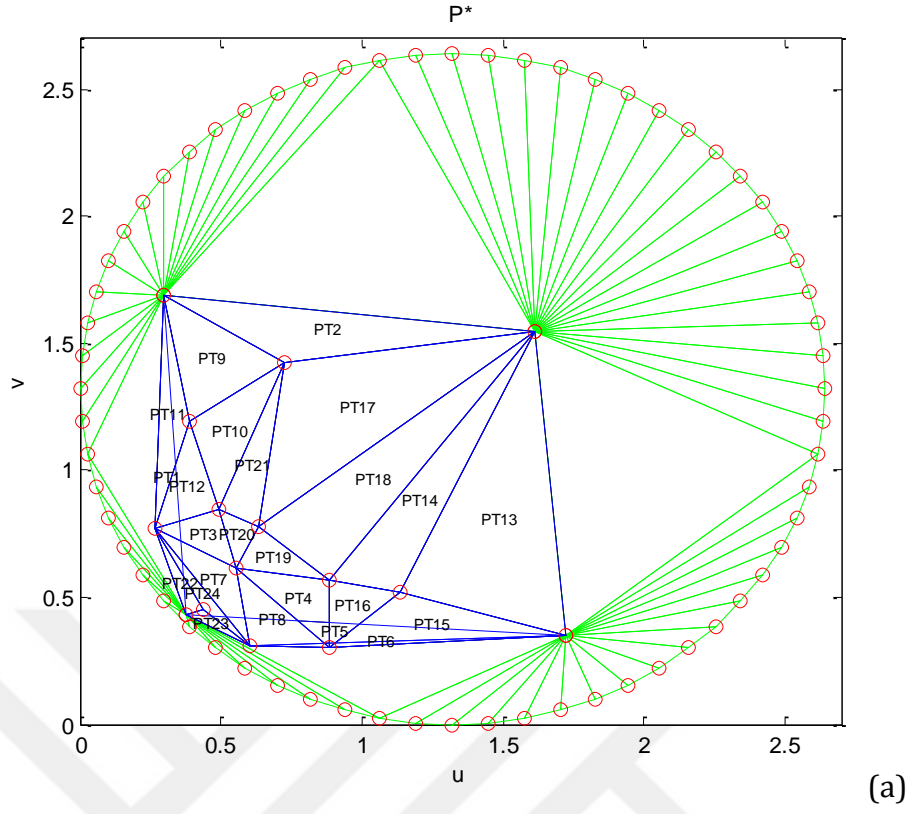


Şekil 4.40. *Surf3* yüzeyine ilişkin temel yöntem ve önerilen yönteme ait alan doğruluk farkı kıyaslaması (a) ve kenar doğruluk farkı kıyaslaması (b)

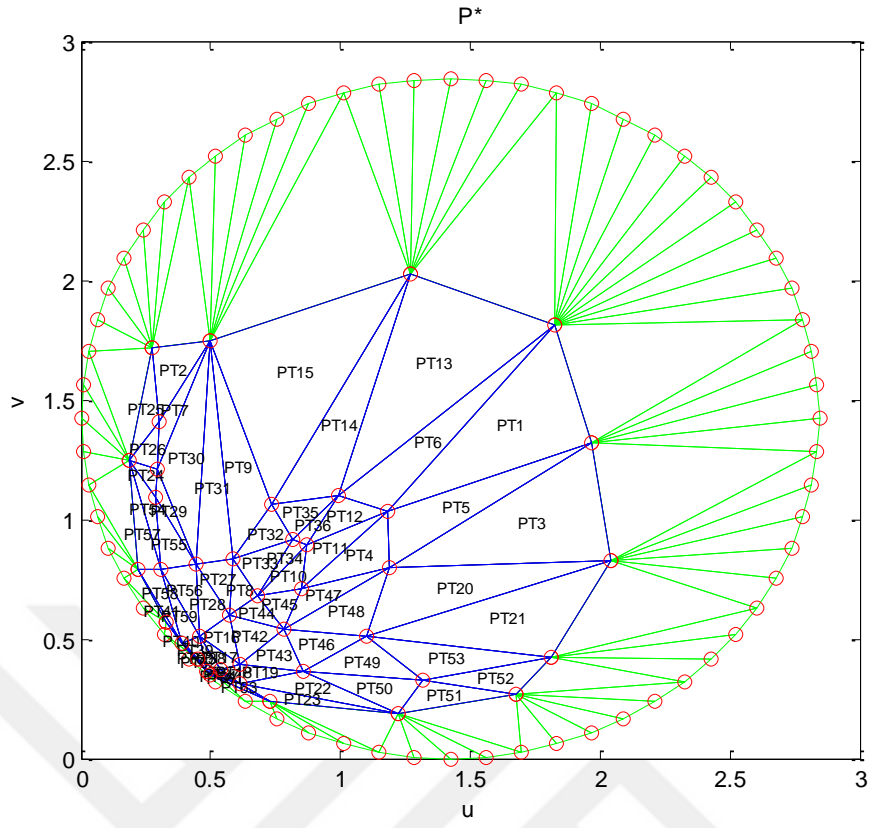
Bahsedilen üç örnek yüzeyin  $n_v=64$  için temel yöntem ve önerilen yöntem ile düzleme izdüşümü yapılarak u-v parametrik uzaydaki düğüm koordinatları Şekil 4.41-Şekil 4.43'te çizdirilmiştir.



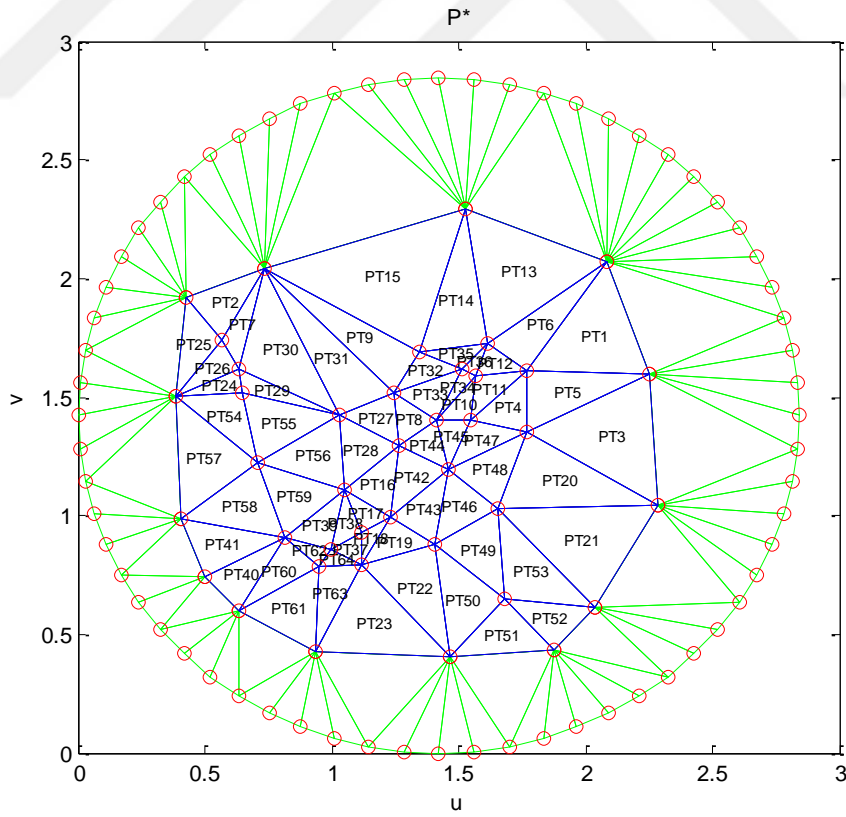
Şekil 4.41. *Surf1*'in  $n_v=64$  için temel yöntem (a) ve önerilen yöntem (b) ile düzleme izdüşümü



Şekil 4.42. *Surf2*'in  $n_v=64$  için temel yöntem (a) ve önerilen yöntem (b) ile düzleme izdüşümü



(a)



(b)

Şekil 4.43. *Surf3'*'ün  $n_v=64$  için temel yöntem (a) ve önerilen yöntem (b) ile düzleme izdüşümü

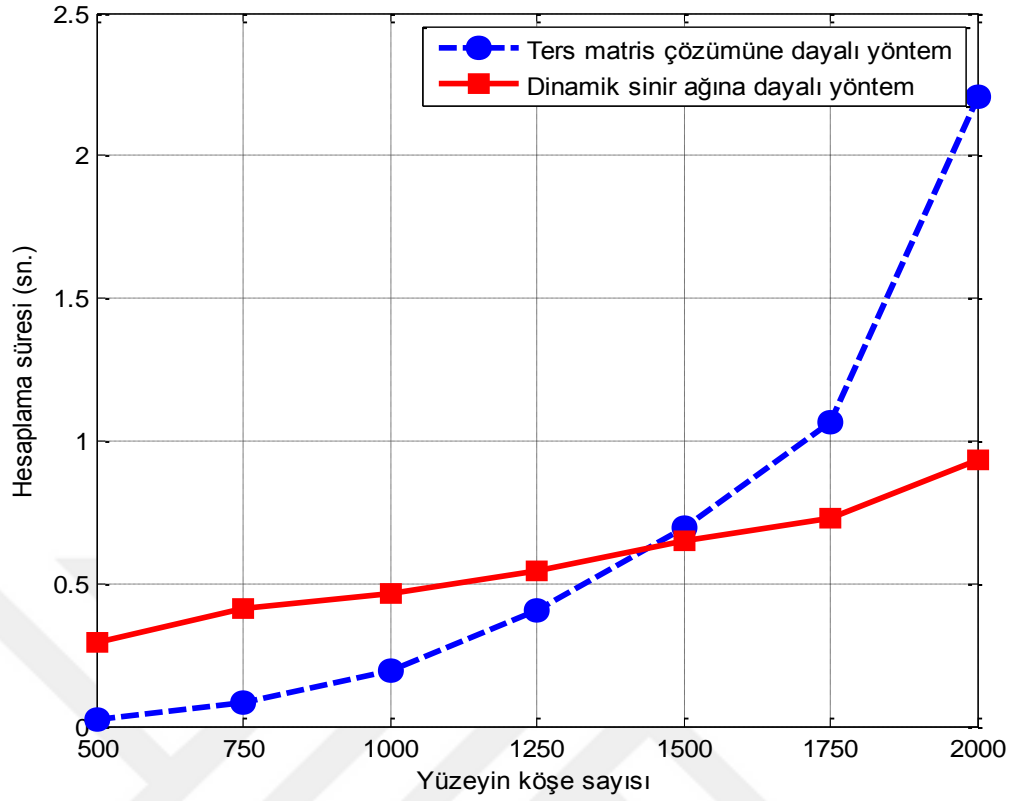
Elde edilen tüm bu sonuçlar önerilen yöntemin, temel yönteme göre (aynı iterasyon sayısında) daha iyi izdüşüm yaptığını göstermektedir. Önerilen yöntem ile elde edilen izdüşüm sonuçlarının orijinal yüzeyleri karşılama oranlarında ortaya çıkan farklar temel yönteme nazaran çok daha küçük çıkmıştır. Aynı yüzeyler için, önerilen yöntemde elde edilen izdüşüm başarısını yakalamak için temel yöntemdeki iterasyon sayısını arttırmak gerekmektedir. Fakat bu da izdüşüm süresinin artmasına sebep olur. Bu bakımdan önerilen yöntem aynı izdüşüm kalitesini sağlama noktasında, temel yönteme göre hız bakımından avantaj sağlamaktadır.

#### 4.1.4. Başlangıç izdüşüm hızı

Önerilen dinamik sinir ağına dayalı model ve Floater'ın ters matris çözümüne dayalı yöntemi çeşitli sayıda köşe içeren yüzeyler üzerinde test edilmiştir. Bu amaçla 4GB RAM belleğe ve Intel Core i5-3470 (3.2 Ghz) işlemciye sahip üzerinde Windows 7 İşletim Sistemi olan bir bilgisayar kullanılarak MATLAB R2016a (64-bit) hesaplama ortamında testler yapılmıştır. Kıyaslama için farklı boyutlarda rastgele üretilen 7 değişik yüzey kullanılmıştır. Kıyaslanan yöntemlerin çeşitli büyüklükteki yüzeyler için ölçülen izdüşüm süreleri Çizelge 4.5'te verilerek Şekil 4.44'teki grafikte görselleştirilmiştir.

Çizelge 4.5. Çeşitli büyükte yüzeylerin başlangıç izdüşüm süreleri

		<b>Floater'ın yöntemi (Ters matris çözümüne dayalı )</b>	<b>Önerilen yöntem (Dinamik sinir ağına dayalı)</b>
<b>Test Yüzeyi</b>	<b>Köşe sayısı</b>	<b>Hesaplama süresi(sn.)</b>	<b>Hesaplama süresi(sn.)</b>
1	500	0.0243	0.2933
2	750	0.0813	0.4134
3	1000	0.1956	0.4676
4	1250	0.4066	0.5431
5	1500	0.6988	0.6466
6	1750	1.0645	0.7296
7	2000	2.2041	0.9363



Şekil 4.44. Başlangıç izdüşüm sürelerinin görselleştirilmesi

Ters matris çözümüne dayalı yöntemin görece küçük boyutlu yüzeylerin (yani 1500'den daha az köşeye sahip) izdüşümünde daha hızlı olduğu grafikten görülmektedir. Fakat söz konusu, görece daha büyük yüzeyler olduğunda (yani 1500'den daha fazla köşeye sahip yüzeyler) önerilen dinamik sınır ağına dayalı yöntem diğer yöntemden çok daha kısa sürelerde izdüşüm gerçekleştirilmektedir.

## 4.2. Mekanik Düzleştirme Sonuçları

Geometrik düzleştirme tekniğiyle elde edilen başlangıç izdüşüm sonucunun kütle-yay modeliyle iyileştirilmesine ilişkin sonuçlar bu kısımda verilmiştir.

### 4.2.1. Aydınlatıcı örnekler

Önerilen başlangıç izdüşüm algoritmasıyla düzlemsel yüzeye projeksiyonu yapılan örnek yüzey yamalarının başlangıç izdüşüm sonuçlarında iyileştirme

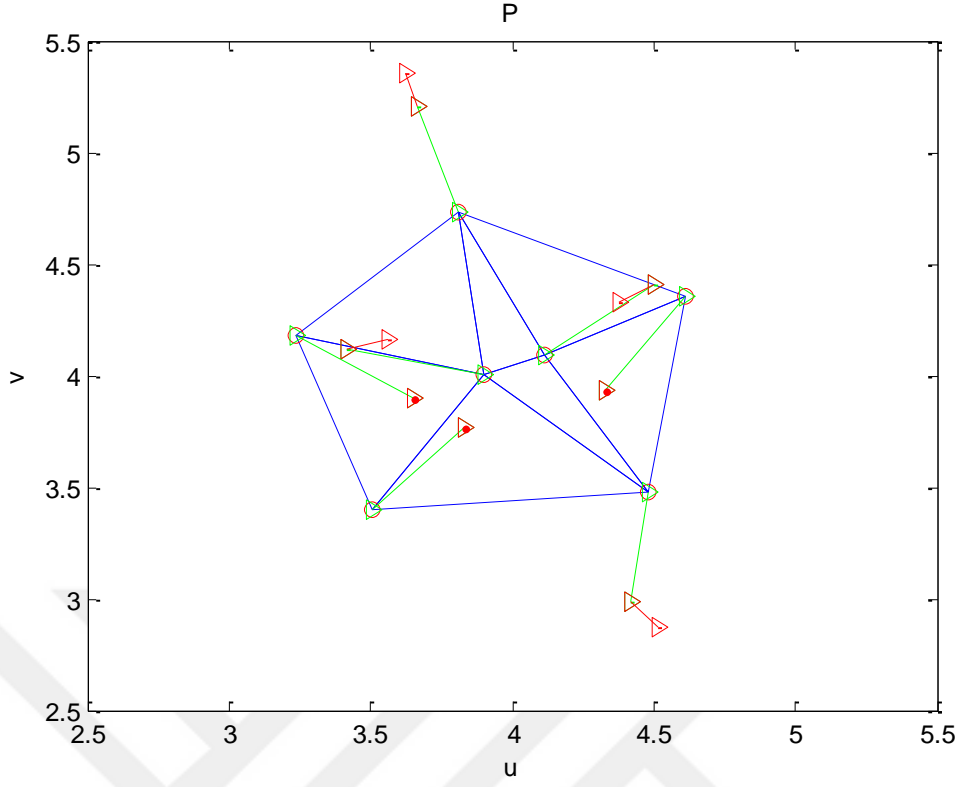
yapmak için Bölüm 3.5.3'teki değişken adım büyüklüğüne dayalı enerji yayma algoritması kullanılmıştır. Bu amaçla örnek yüzey yamalarına (*Surf1* ve *Surf2*) ilişkin önceki bölümde (Bölüm 4.1) elde edilen başlangıç izdüşüm sonuçları bu algoritmaya girdi olarak verilmiştir.

İlk olarak, *Surf1* yüzeyine ilişkin sanal sınır poligonu düğüm sayısı  $n_v=64$  iken üretilen başlangıç izdüşüm sonucu kullanılarak enerji yayma algoritmasının belirli koşullar altındaki ara adım sonuçları ve nihai sonuç verilmiştir. Değişken adım büyüklüğüne dayalı enerji yayma algoritması Çizelge 4.6'da değerleri verilen parametrelerle çalıştırılmıştır. Bu parametrelerden C ve p'nin seçiminin nasıl yapılacağına daha önce Bölüm 3.3'te değinilmiştir. Diğer parametreler ise geliştirilmekte olan düzlemsel yüzeyin türüne ve ihtiyaca göre değişiklik gösteren algoritmanın durma kriterini oluşturan parametrelerdir.

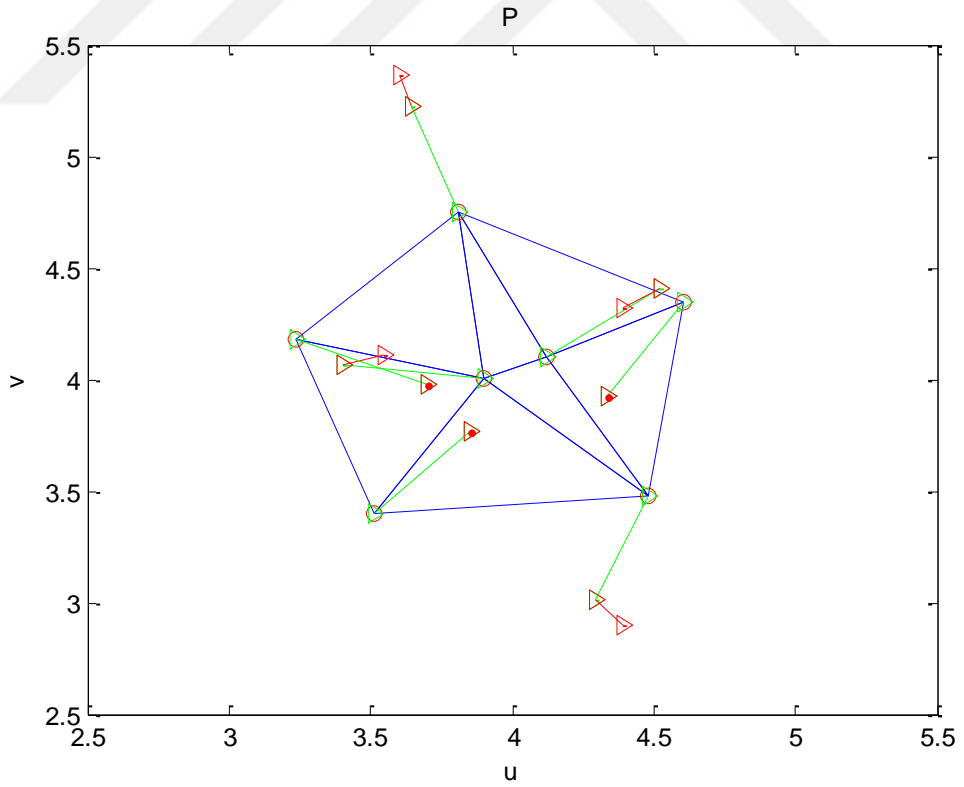
Çizelge 4.6. *Surf1* yüzey yamasına ilişkin değişken adım büyüklüğüne dayalı enerji yayma algoritması parametreleri

Parametreler		
Açıklaması	Sembol	Değeri
Yay sabiti	C	0.5
Alan yoğunluğu	p	2
İzin verilen alan-doğruluk farkı yüzdesi	$\emptyset_s$	0.16
İzin verilen kenar-doğruluk farkı yüzdesi	$\emptyset_c$	0.40
İzin verilen enerji değişim yüzdesi	$\epsilon$	0.05
İzin verilen maksimum iterasyon sayısı	Nmax	100

Çizelge 4.6'da verilen parametrelere göre çalıştırılan değişken adım büyüklüğüne dayalı enerji yayma algoritmasının (Algoritma-3) ara çıktılarından bazıları görsel olarak Şekil 4.45-4.47'de ve bunlara ilişkin sayısal sonuçlar Çizelge 4.7'de verilmiştir. Şekildeki yeşil renkli vektör kütleli hareket ettiren kuvvetin yönünü, kırmızı renkli vektör ise kütleli etkiyen ceza (fonksiyonu) kuvvetinin yönünü göstermektedir. Çizim ve gösterim kolaylığı bakımından kuvvet büyüklükleri aynı gösterilmiştir.



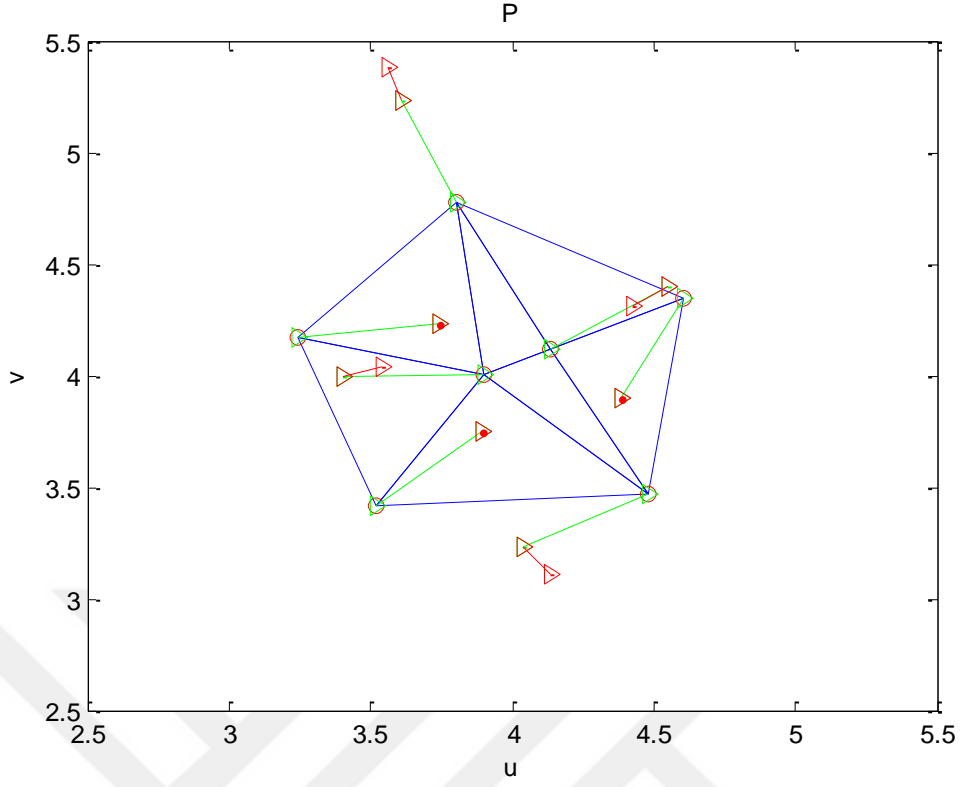
(a) iterasyon #1



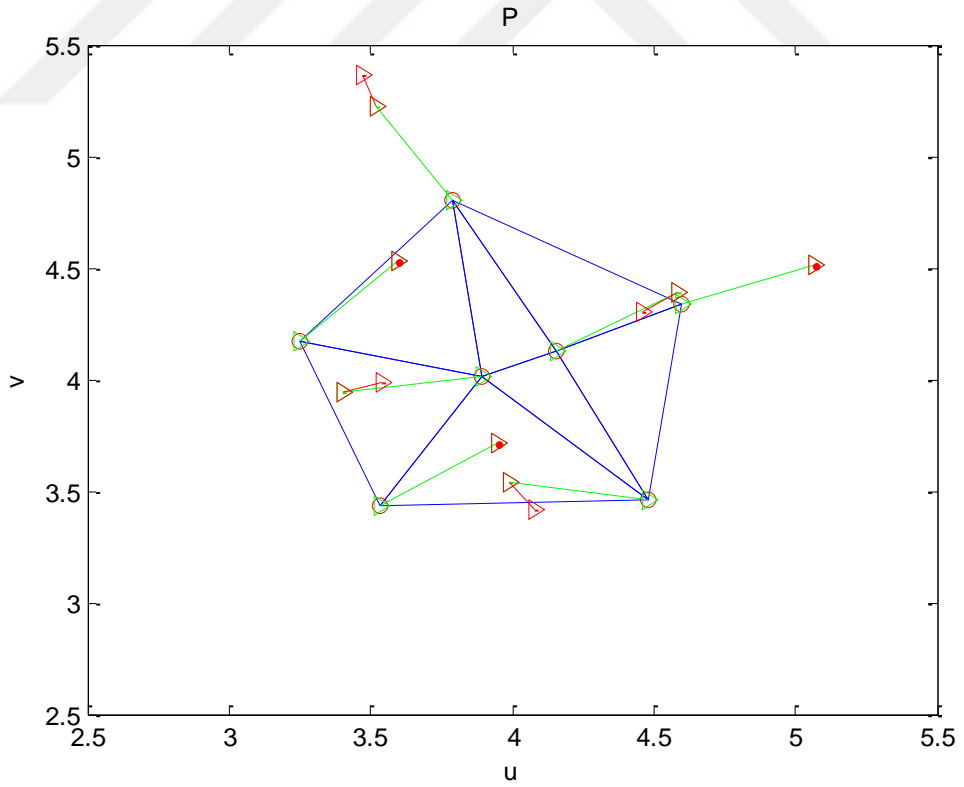
(b) iterasyon #5

Şekil 4.45. *Surf1* yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (1)



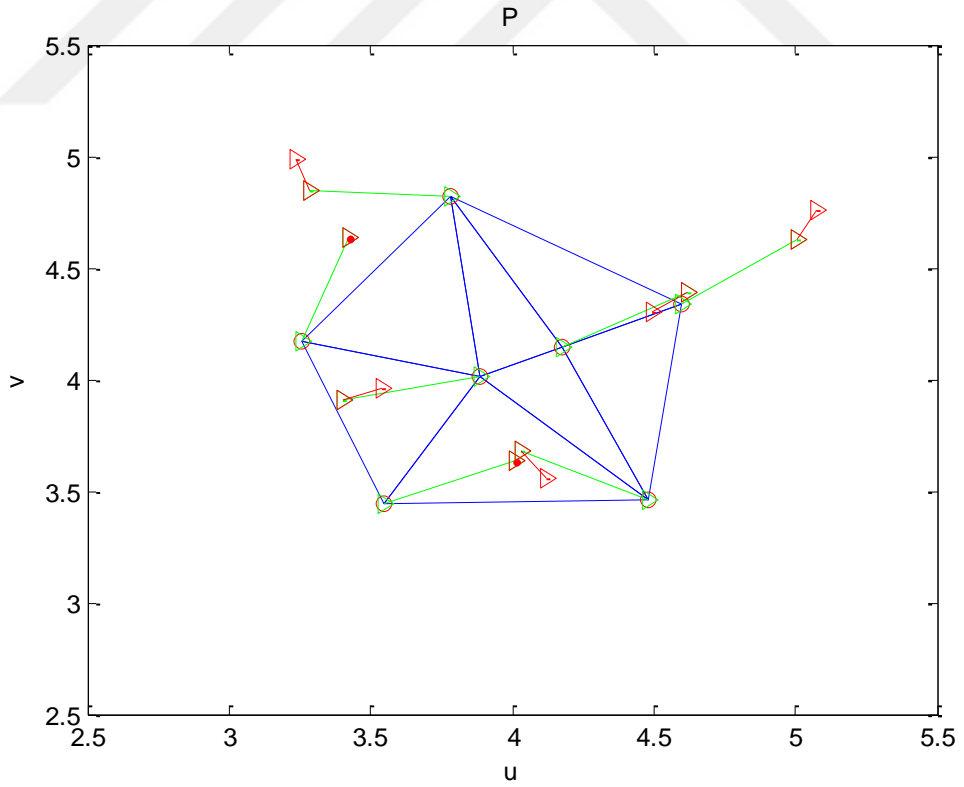
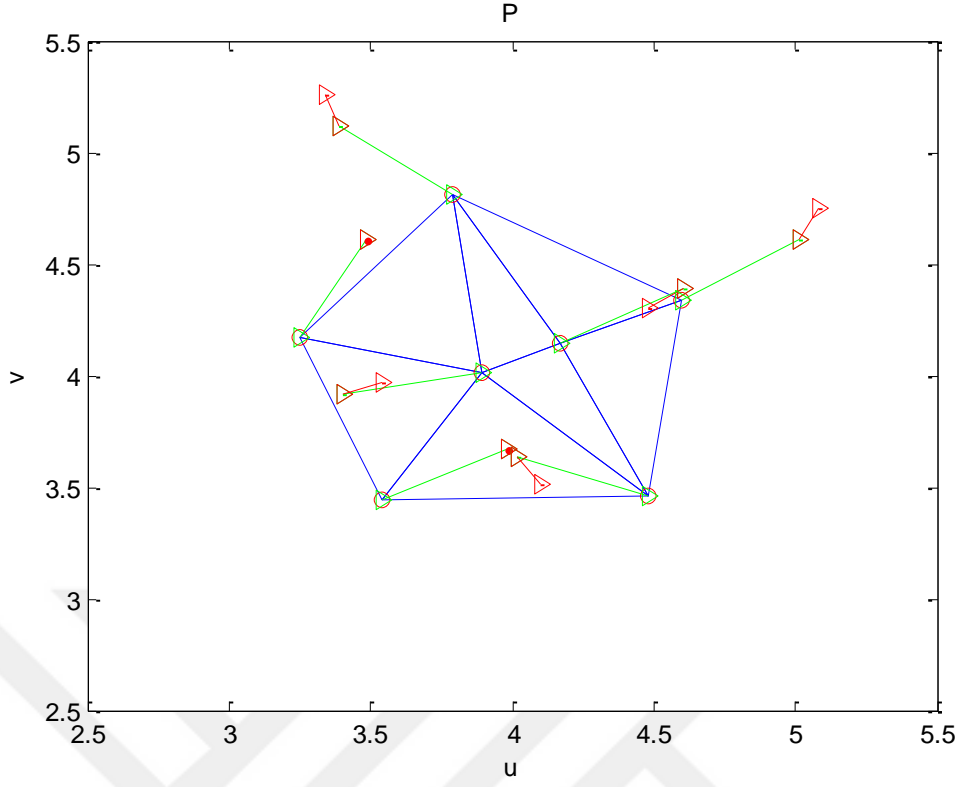


(a) iterasyon #10



(b) iterasyon #20

Şekil 4.46. *Surf1* yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (2)

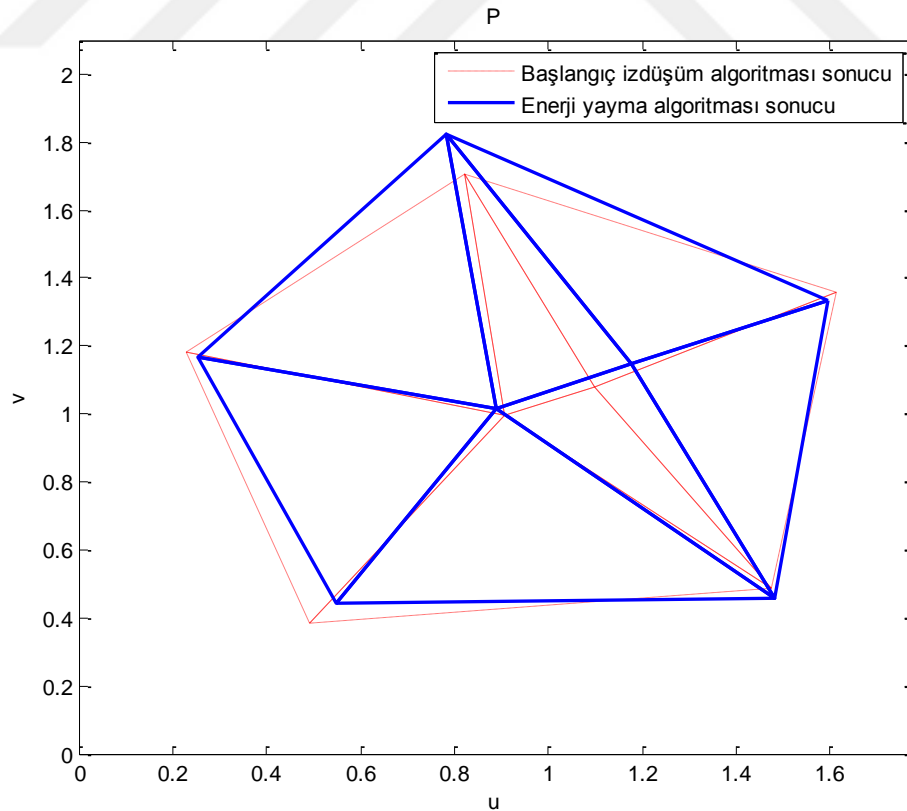


Şekil 4.47. *Surf1* yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (3)

Çizelge 4.7. Şekil 4.45 -4.47'deki enerji yayma sonuçlarına ilişkin alan ve kenar doğruluk fark değerleri

İterasyon #	Alan doğruluk farkı oranı, $E_s$	Kenar doğruluk farkı oranı, $E_c$
1	0.1968	0.4187
5	0.1919	0.4171
10	0.1857	0.4149
20	0.1743	0.4110
30	0.1599	0.4073
36	0.1472	0.3998

Alan-doğruluk fark değeri  $E_s=0.203$  ve kenar-doğruluk fark değeri  $E_c=0.4202$  ile (Çizelge 4.6'daki parametrelere göre çalıştırılan) enerji yayma algoritmasına sokulan *Surf1* yüzeyine ilişkin başlangıç izdüşüm sonucu daha da iyileştirilerek 36 iterasyon sonunda  $E_s=0.1472$  ve  $E_c=0.3998$  sonucu elde edilmiştir. Örnek yüzeye ilişkin başlangıç izdüşüm algoritmasının sonucu ve enerji yayma algoritmasının iyileştirme sonucu Şekil 4.48'deki grafikte üst üste çizdirilmiştir.



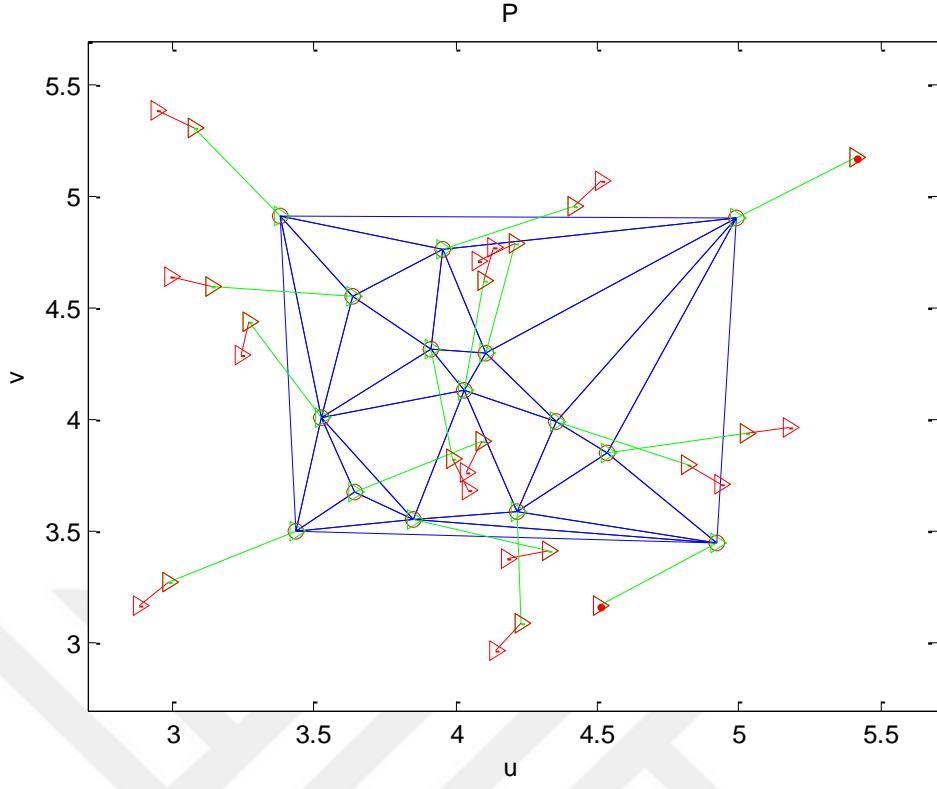
Şekil 4.48. *Surf1* yüzeyine başlangıç izdüşüm ve enerji yayma algoritması sonuçları

İkinci olarak, *Surf2* yüzeyine ilişkin sanal sınır poligonu düğüm sayısı  $n_v=64$  iken üretilen başlangıç izdüşüm sonucu kullanılarak enerji yayma algoritmasının belirli koşullar altındaki ara adım sonuçları ve nihai sonuç verilmiştir. Değişken adım büyüklüğüne dayalı enerji yayma algoritması Çizelge 4.8'de değerleri verilen parametrelerle çalıştırılmıştır.

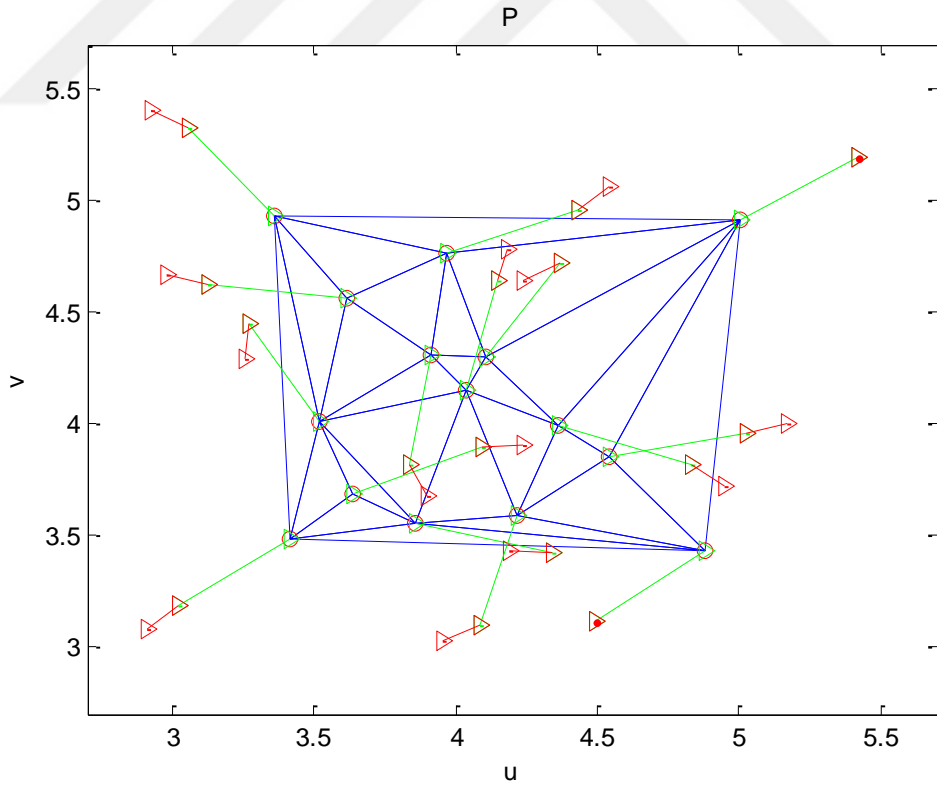
Çizelge 4.8. *Surf2* yüzey yamasına ilişkin değişken adım büyüklüğüne dayalı enerji yayma algoritması parametreleri

Parametreler		
Açıklaması	Sembol	Değeri
Yay sabiti	C	0.5
Alan yoğunluğu	p	2
İzin verilen alan-doğruluk farkı yüzdesi	$\phi_s$	0.30
İzin verilen kenar-doğruluk farkı yüzdesi	$\phi_c$	0.25
İzin verilen enerji değişim yüzdesi	$\epsilon$	0.05
İzin verilen maksimum iterasyon sayısı	Nmax	150

Çizelge 4.8'de verilen parametrelere göre çalıştırılan enerji yayma algoritmasının (Algoritma-3) ara çıktılarından bazıları görsel olarak Şekil 4.49-4.51'de ve bunlara ilişkin sayısal sonuçlar Çizelge 4.8'de verilmiştir.

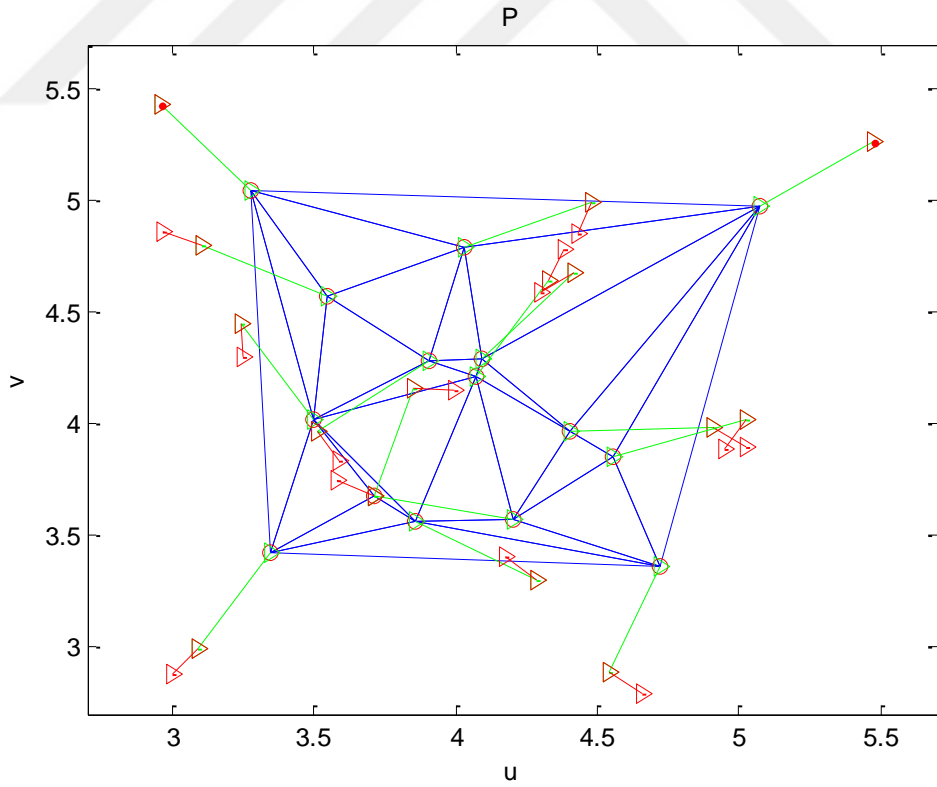
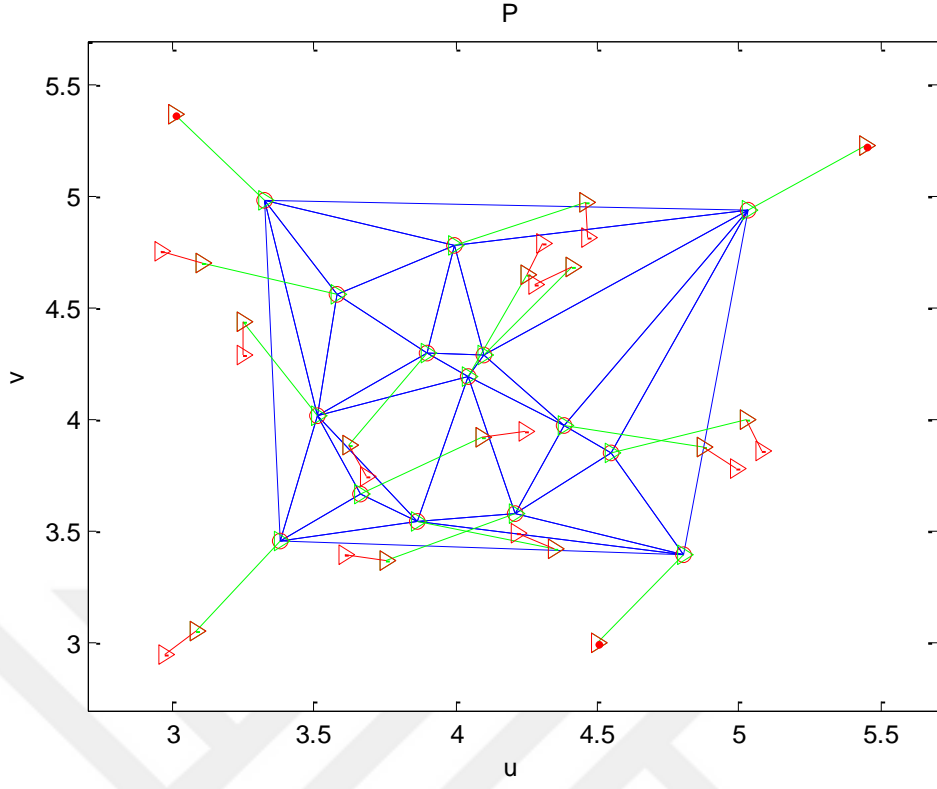


(a) iterasyon #1

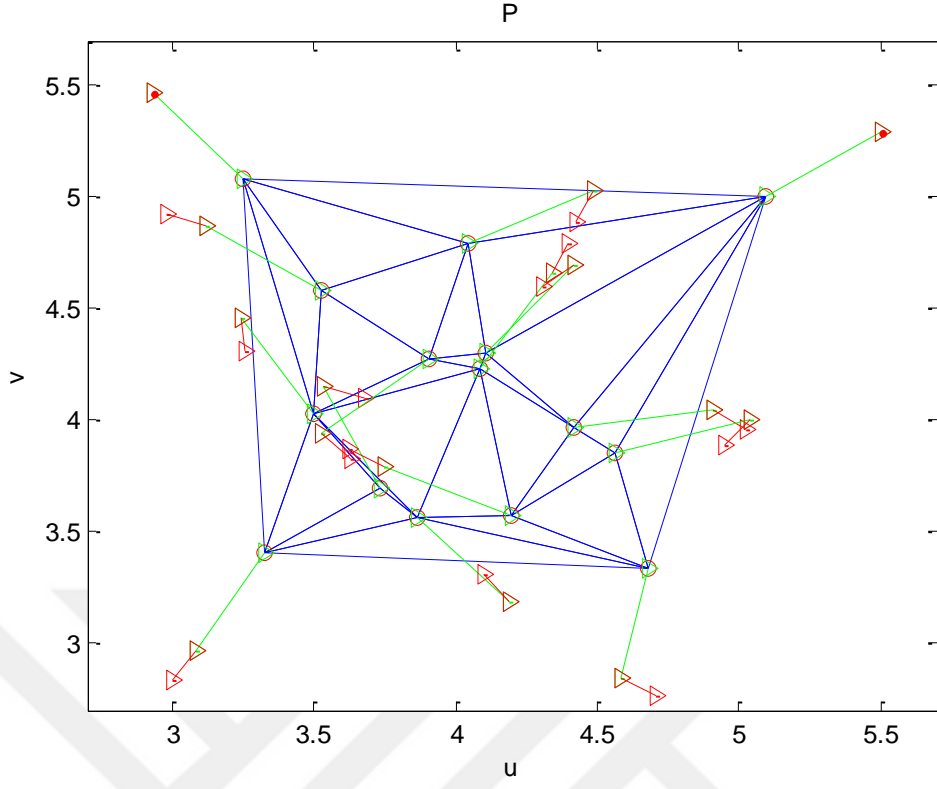


(b) iterasyon #10

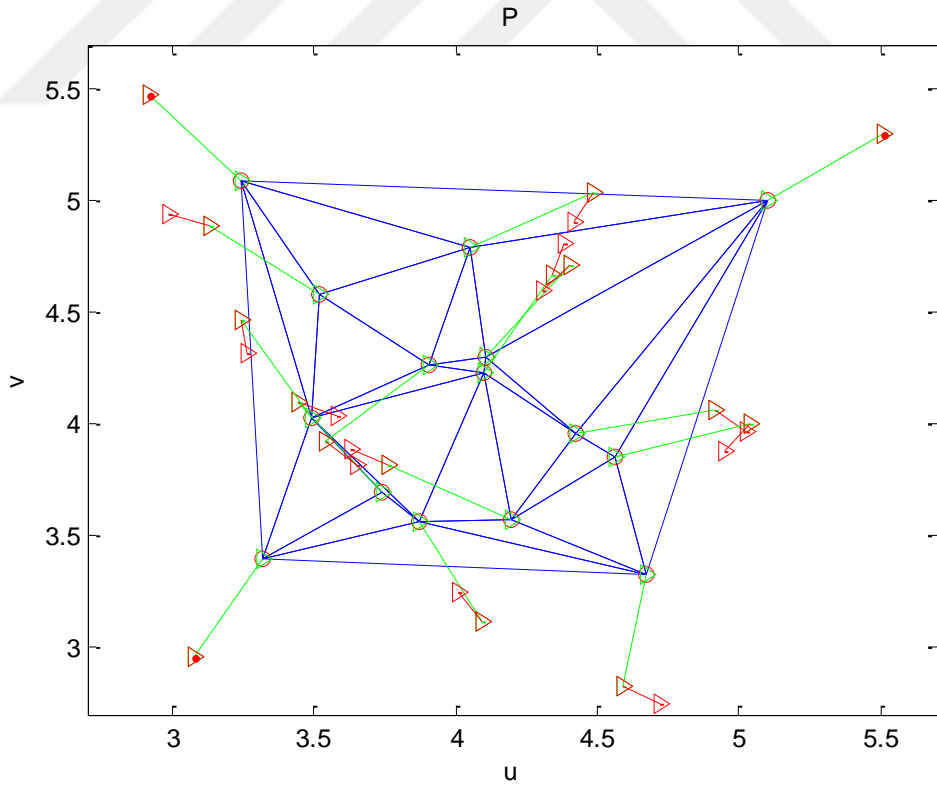
Şekil 4.49. *Surf2* yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (1)



Şekil 4.50. *Surf2* yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (2)



(a) iterasyon #60



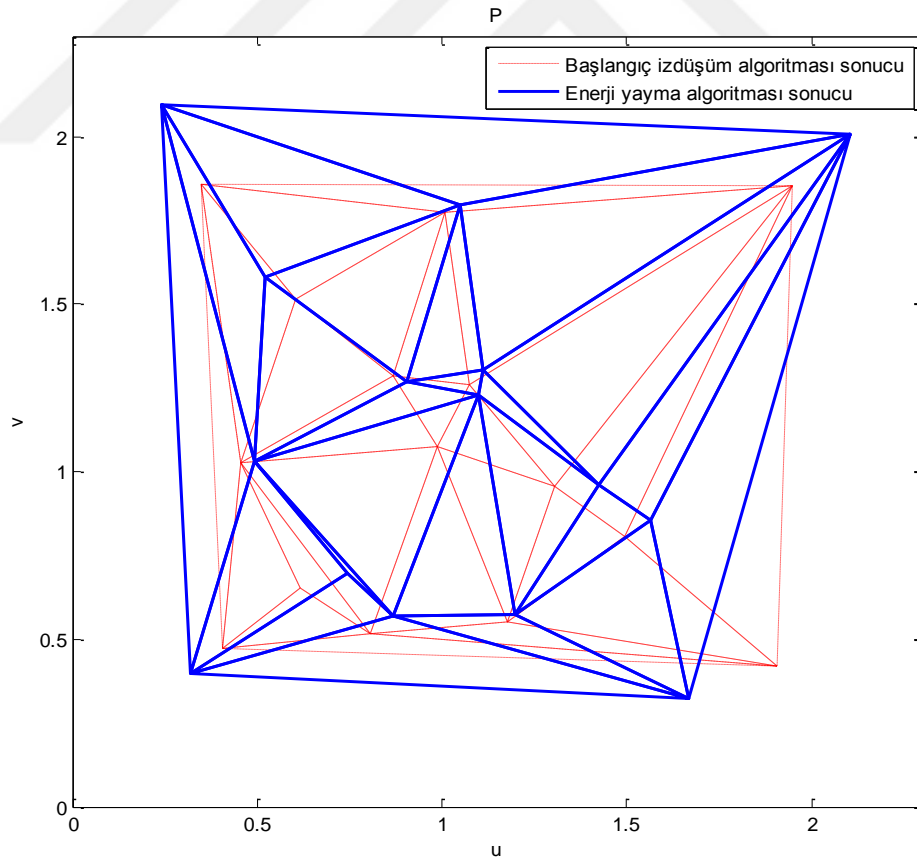
(b) iterasyon #70

Şekil 4.51. *Surf2* yüzeyine ilişkin enerji yayma algoritmasının çeşitli adımlardaki çıktıları (3)

Çizelge 4.9. Şekil 4.49 -4.51'deki enerji yayma sonuçlarına ilişkin alan ve kenar doğruluk fark değerleri

İterasyon #	Alan doğruluk farkı oranı, $E_S$	Kenar doğruluk farkı oranı, $E_C$
1	0.4794	0.2634
10	0.4595	0.2614
20	0.4366	0.2530
40	0.3861	0.2393
60	0.3295	0.2423
70	0.2982	0.2484

Alan-doğruluk fark değeri  $E_S=0.4906$  ve kenar-doğruluk fark değeri  $E_C=0.2697$  ile (Çizelge 4.8'deki parametrelere göre çalıştırılan) enerji yayma algoritmasına sokulan *Surf2* yüzeyine ilişkin başlangıç izdüşüm sonucu daha da iyileştirilerek 70 iterasyon sonunda  $E_S=0.2982$  ve  $E_C=0.2484$  sonucu elde edilmiştir. Örnek yüzeye ilişkin başlangıç izdüşüm algoritmasının sonucu ve enerji yayma algoritmasının iyileştirme sonucu Şekil 4.52'deki grafikte üst üste çizdirilmiştir.



Şekil 4.52. *Surf2* yüzeyine başlangıç izdüşüm ve enerji yayma algoritması sonuçları



#### 4.2.2. Değişken adım büyüklüğünün enerji yayma süresine etkisi

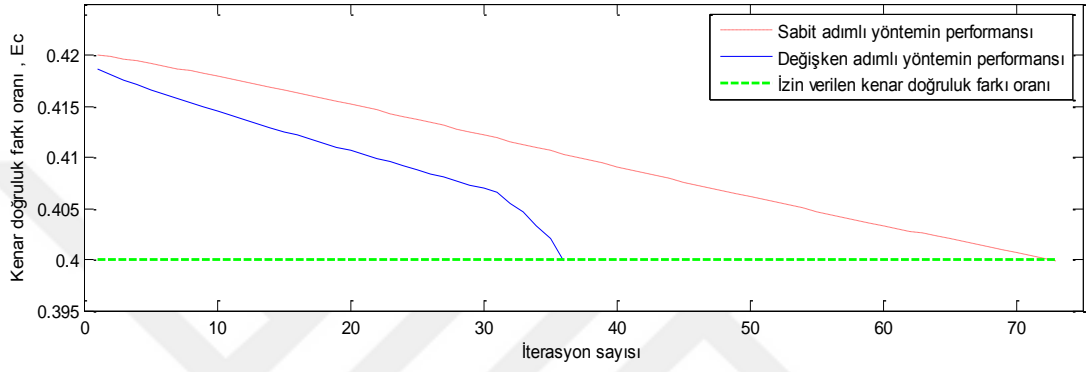
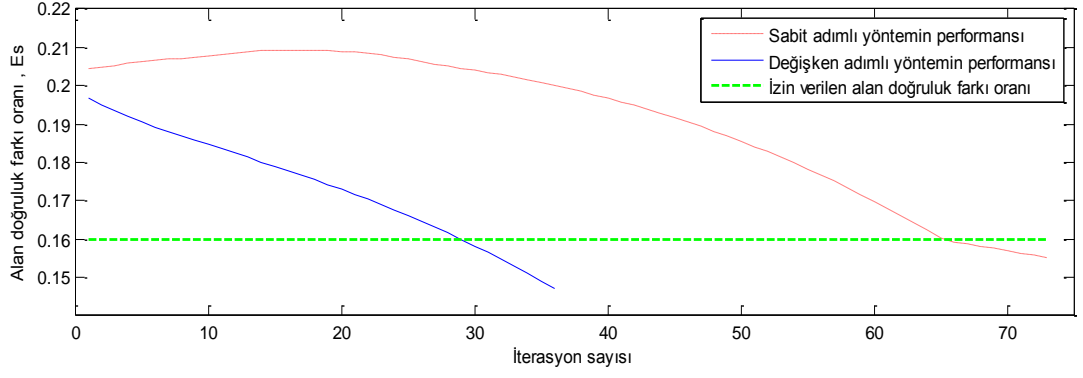
Enerji tabanlı düzleştirme yönteminde, daha önce Bölüm 3.5.3'te bahsedildiği gibi, Euler yöntemi ile denklem çözümünde  $\Delta t$  sabit adım büyüklüğü yerine değişken adım büyüklüğüne dayalı hale getirilmiştir. Bu yapı sayesinde enerji yayma süreci baz modeldekine göre daha hızlı bir şekilde gerçekleştirilmektedir. *Surf1* ve *Surf2* yüzey yamaları için enerji yayma sürecinin sabit adımlı veya değişken adımlı olması durumunda gereken iterasyon sayısı ve buna bağlı olarak ortaya çıkan nihai göreceli alan ( $E_S$ ) ve kenar ( $E_C$ ) farkları Çizelge 4.10 ve 4.11'de verilmiştir. Ayrıca örnek yüzey yamalarına ilişkin sabit ve değişken adımlı enerji yayma yöntemlerinin performansları *Surf1* için Şekil 4.53'te ve *Surf2* için Şekil 4.54'te verilmiştir.

Çizelge 4.10. *Surf1* yüzeyine ilişkin sabit ve değişken adım büyüklüğüne dayalı enerji yayma sürecindeki doğruluk değerleri

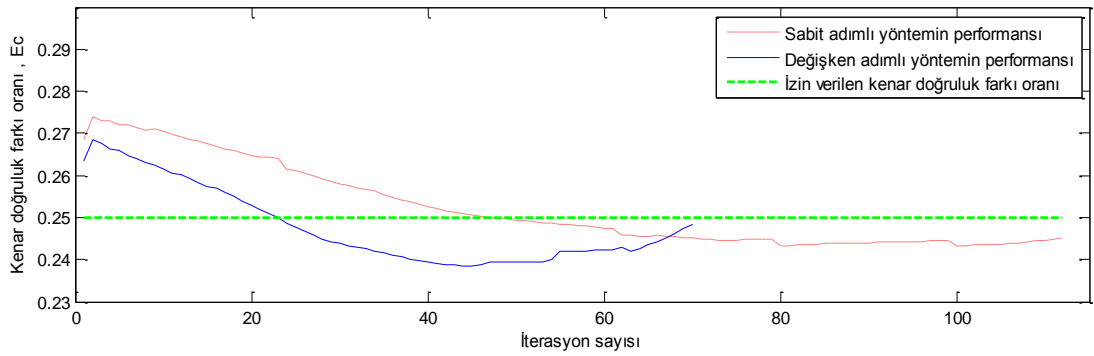
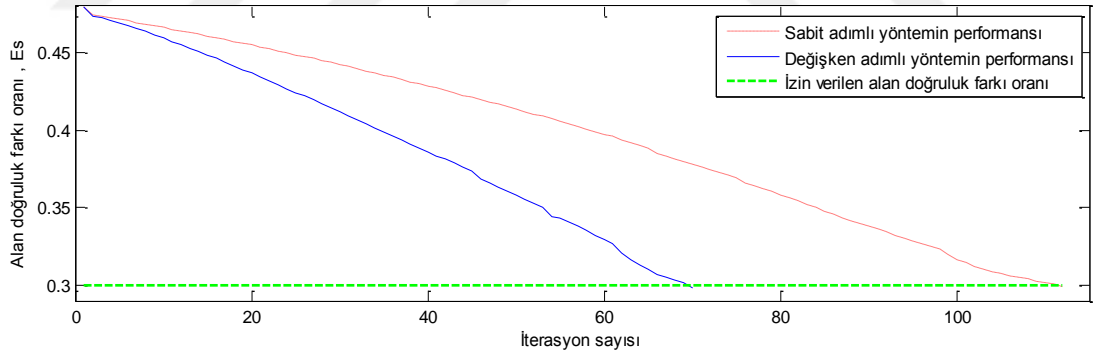
Sabit adım büyüklüğü ( $\Delta t=0.01$ )		Değişken adım büyüklüğü	
İterasyon sayısı	Alan ve kenar doğruluk farkı oranları	İterasyon sayısı	Alan ve kenar doğruluk farkı oranları
1	$E_S=0.2044, E_C=0.4200$	1	$E_S=0.1968, E_C=0.4187$
2	$E_S=0.2049, E_C=0.4198$	2	$E_S=0.1950, E_C=0.4181$
3	$E_S=0.2054, E_C=0.4196$	3	$E_S=0.1934, E_C=0.4176$
4	$E_S=0.2058, E_C=0.4194$	4	$E_S=0.1919, E_C=0.4171$
5	$E_S=0.2062, E_C=0.4192$	5	$E_S=0.1905, E_C=0.4166$
6	$E_S=0.2066, E_C=0.4189$	6	$E_S=0.1893, E_C=0.4162$
7	$E_S=0.2069, E_C=0.4187$	7	$E_S=0.1880, E_C=0.4157$
8	$E_S=0.2072, E_C=0.4185$	8	$E_S=0.1869, E_C=0.4153$
9	$E_S=0.2074, E_C=0.4182$	9	$E_S=0.1857, E_C=0.4149$
10	$E_S=0.2078, E_C=0.4180$	10	$E_S=0.1846, E_C=0.4145$
11	$E_S=0.2082, E_C=0.4177$	11	$E_S=0.1835, E_C=0.4141$
12	$E_S=0.2086, E_C=0.4174$	12	$E_S=0.1824, E_C=0.4137$
13	$E_S=0.2089, E_C=0.4172$	13	$E_S=0.1813, E_C=0.4133$
14	$E_S=0.2091, E_C=0.4169$	14	$E_S=0.1802, E_C=0.4129$
15	$E_S=0.2093, E_C=0.4166$	15	$E_S=0.1791, E_C=0.4125$
16	$E_S=0.2093, E_C=0.4163$	16	$E_S=0.1779, E_C=0.4121$
17	$E_S=0.2094, E_C=0.4161$	17	$E_S=0.1767, E_C=0.4118$
18	$E_S=0.2093, E_C=0.4158$	18	$E_S=0.1755, E_C=0.4114$
19	$E_S=0.2092, E_C=0.4155$	19	$E_S=0.1743, E_C=0.4110$
20	$E_S=0.2090, E_C=0.4152$	20	$E_S=0.1731, E_C=0.4106$
21	$E_S=0.2087, E_C=0.4149$	21	$E_S=0.1718, E_C=0.4103$
22	$E_S=0.2084, E_C=0.4146$	22	$E_S=0.1704, E_C=0.4099$
23	$E_S=0.2080, E_C=0.4143$	23	$E_S=0.1691, E_C=0.4095$
24	$E_S=0.2075, E_C=0.4140$	24	$E_S=0.1676, E_C=0.4092$
25	$E_S=0.2069, E_C=0.4137$	25	$E_S=0.1662, E_C=0.4088$
26	$E_S=0.2063, E_C=0.4134$	26	$E_S=0.1647, E_C=0.4084$
27	$E_S=0.2056, E_C=0.4131$	27	$E_S=0.1631, E_C=0.4080$
28	$E_S=0.2051, E_C=0.4128$	28	$E_S=0.1616, E_C=0.4077$
29	$E_S=0.2046, E_C=0.4125$	29	$E_S=0.1599, E_C=0.4073$
30	$E_S=0.2041, E_C=0.4122$	30	$E_S=0.1582, E_C=0.4069$
31	$E_S=0.2035, E_C=0.4119$	31	$E_S=0.1565, E_C=0.4065$
32	$E_S=0.2029, E_C=0.4116$	32	$E_S=0.1547, E_C=0.4055$
33	$E_S=0.2023, E_C=0.4113$	33	$E_S=0.1529, E_C=0.4047$
34	$E_S=0.2016, E_C=0.4109$	34	$E_S=0.1510, E_C=0.4033$
35	$E_S=0.2009, E_C=0.4106$	35	$E_S=0.1490, E_C=0.4021$
36	$E_S=0.2001, E_C=0.4103$	36	$E_S=0.1472, E_C=0.3998$
.	.		
.	.		
.	.		
71	$E_S=0.1564, E_C=0.4004$		
72	$E_S=0.1557, E_C=0.4001$		
73	$E_S=0.1550, E_C=0.3998$		

Çizelge 4.11. *Surf2* yüzeyine ilişkin sabit ve değişken adım büyüklüğüne dayalı enerji yayma sürecindeki ve doğruluk değerleri

Sabit adım büyüklüğü ( $\Delta t=0.01$ )		Değişken adım büyüklüğü	
İterasyon numarası	Alan ve kenar doğruluk farkı oranları	İterasyon numarası	Alan ve kenar doğruluk farkı oranları
1	$E_S=0.4797, E_C=0.2685$	1	$E_S= 0.4794, E_C=0.2634$
2	$E_S=0.4746, E_C=0.2739$	2	$E_S= 0.4738, E_C=0.2686$
3	$E_S=0.4735, E_C=0.2730$	3	$E_S= 0.4724, E_C=0.2675$
4	$E_S=0.4725, E_C=0.2729$	4	$E_S= 0.4708, E_C=0.2663$
5	$E_S=0.4715, E_C=0.2721$	5	$E_S= 0.4691, E_C=0.2660$
6	$E_S=0.4705, E_C=0.2722$	6	$E_S= 0.4673, E_C=0.2648$
7	$E_S=0.4695, E_C=0.2715$	7	$E_S= 0.4655, E_C=0.2642$
8	$E_S=0.4684, E_C=0.2709$	8	$E_S= 0.4636, E_C=0.2630$
9	$E_S=0.4673, E_C=0.2710$	9	$E_S= 0.4616, E_C=0.2625$
10	$E_S=0.4663, E_C=0.2705$	10	$E_S= 0.4595, E_C=0.2614$
11	$E_S=0.4652, E_C=0.2698$	11	$E_S= 0.4574, E_C=0.2605$
12	$E_S=0.4641, E_C=0.2692$	12	$E_S= 0.4553, E_C=0.2602$
13	$E_S=0.4630, E_C=0.2686$	13	$E_S= 0.4530, E_C=0.2592$
14	$E_S=0.4619, E_C=0.2681$	14	$E_S= 0.4508, E_C=0.2582$
15	$E_S=0.4607, E_C=0.2675$	15	$E_S= 0.4485, E_C=0.2572$
16	$E_S=0.4596, E_C=0.2670$	16	$E_S= 0.4462, E_C=0.2569$
17	$E_S=0.4584, E_C=0.2664$	17	$E_S= 0.4438, E_C=0.2560$
18	$E_S=0.4572, E_C=0.2659$	18	$E_S= 0.4415, E_C=0.2549$
19	$E_S=0.4561, E_C=0.2653$	19	$E_S= 0.4391, E_C=0.2539$
20	$E_S=0.4549, E_C=0.2647$	20	$E_S= 0.4366, E_C=0.2530$
21	$E_S=0.4537, E_C=0.2643$	21	$E_S= 0.4342, E_C=0.2519$
22	$E_S=0.4524, E_C=0.2645$	22	$E_S= 0.4318, E_C=0.2509$
		.	.
		.	.
		.	.
		66	$E_S= 0.3073, E_C= 0.2442$
.	.	67	$E_S= 0.3049, E_C= 0.2452$
.	.	68	$E_S= 0.3033, E_C= 0.2462$
.	.	69	$E_S= 0.3018, E_C= 0.2473$
		<b>70</b>	<b><math>E_S= 0.2982, E_C=0.2484</math></b>
104	$E_S=0.3088, E_C=0.2436$		
105	$E_S=0.3074, E_C=0.2437$		
106	$E_S=0.3062, E_C=0.2438$		
107	$E_S=0.3051, E_C=0.2439$		
108	$E_S=0.3040, E_C=0.2441$		
109	$E_S=0.3028, E_C=0.2444$		
110	$E_S=0.3016, E_C=0.2447$		
111	$E_S=0.3004, E_C=0.2450$		
<b>112</b>	<b><math>E_S=0.2992, E_C=0.2452</math></b>		



Şekil 4.53. *Surf1* yüzeyine ilişkin sabit ve değişken adımlı enerji yayma yöntemlerinin performansları



Şekil 4.54. *Surf2* yüzeyine ilişkin sabit ve değişken adımlı enerji yayma yöntemlerinin performansları

Şekildeki grafiklerden görüleceği üzere deęişken adım büyüklüğüne dayalı yöntem; iterasyon sayısını *Surf1* yüzey yaması için 70'den 36'ya, *Surf2* için ise 112'den 70'e kadar azaltmıştır.



## 5. SONUÇ VE ÖNERİLER

Serbest biçimli Öklid olmayan yüzeyler hızla gün yüzüne çıkmakta ve çok sayıda farklı uygulamalarda kullanılmaktadır. Örneğin serbest biçimli yüzeyler; optik, 3D yazıcılar, enerji verimli jet motorları, uçak türbin kanatları ve güvenilir uzun ömürlü implantlar gibi pek çok ürün tasarımında kullanılmaktadır. Bu anlamda serbest biçimli yüzeyleri daha iyi karakterize edebilen güvenilir ve düzgün ölçme teknikleri geliştirmek önemli hale gelmektedir. Serbest biçimli yüzeyler günümüz grafik uygulamalarında önemli bir rol oynamaktadır (Eigensatz vd., 2010). Yüzey çizdirme, yüzey kaplama, yüzey örnekleme vb. serbest biçimli yüzeylerle ilgili uygulamaların sonuçları, yüzey parametrelere oldukça bağlıdır.

3B yüzeyin parametrelenmesi, 2B düzlemsel yüzeye bire bir karşılık gelecek bir fonksiyon üretme çalışması olarak tanımlanabilir. Bu kavram geometri işlemede önemli bir rol oynamaktadır, çünkü 3B modelleme problemlerini 2B uzaya dönüştürme yolunu açmayı sağlar ve burada işlem yapmak nispeten daha kolaydır. Karmaşık 3B yüzeyler, gelişmiş aralıklı tarama ve 3B baskı teknolojileri sayesinde yaygın olarak her yerde kullanıldığından, geniş kafesler için düşük parametrelere süreleri önem kazanmaktadır.

Bu çalışmada, 3B üçgensel yüzey yamalarının 2B düzlemsel karşılıklarının üretilmesinde etkin bir yaklaşım önerilmiştir. Önerilen yaklaşım literatürde, geometrik düzleştirme ve mekanik düzleştirme tekniklerinin birleşiminden oluşan hibrit düzleştirme tekniği olarak nitelendirilmektedir. Bu çalışmada geometrik düzleştirme ile başlangıç izdüşümü elde edildikten sonra mekanik düzleştirme ile bu sonucun iyileştirilmesi yapılmaktadır.

Bu bağlamda, 3B üçgensel kafes yüzey yamalarının hızlandırılmış bir şekilde başlangıç izdüşümünün üretilmesi için bir dinamik sinir ağı modeli oluşturulmuştur. Bu çalışmada, geri-beslemeli sinir ağı modeli ile konveks kombinasyon yaklaşımı birleştirilerek üçgensel kafes yüzeylerin başlangıç izdüşümü gerçekleştirilmiştir. Bu yeni yaklaşım, yapay sinir ağlarının paralel-

dağıtık işleme tabiatı ve barisentrik haritalama tekniğinin hızlı ve bire bir izdüşüm özellikleri nedeniyle (nokta sayısı 1.5K'dan daha fazla olan) nispeten büyük yüzeylerin projeksiyonu için gereken hesaplama süresini önemli ölçüde azaltmaktadır.

Önerilen başlangıç izdüşüm algoritmasında düzleştirilmek istenen 3B yüzey yaması işlenmeden önce yüzey yamasına uygun bir sanal sınır poligonu üretme yaklaşımı benimsenmiştir. Belirlenen poligon üzerindeki noktalar (yani sanal sınır düğümleri) ele alınan yüzey yamasına eklenerek ilaveli yüzey yaması elde edilmiştir. Geometrik düzleştirme bu ilaveli yüzey yaması üzerinden gerçekleştirilmiştir. Böylelikle izdüşümde yüzey yamasının sınır noktalarına yakın üçgenlerin deformasyonu ve distorsiyonunun azaltılması sağlanmıştır. Bölüm 4.2.1'de verilen sanal sınır poligonundaki düğüm sayısının başlangıç izdüşüm sonucunun kalitesine etkisi örnek yüzey yamaları üzerinde ayrıntılı olarak test edilmiştir. Bu testlerden elde edilen bulgular sanal düğüm sayısını (ele alınan yüzey yamasının büyüklüğüne göre) belirli bir değere kadar artırmanın düzleştirme sonucunda elde edilen düzlemsel izdüşümün hem alan-doğruluk farkını hem de kenar-doğruluk farkını kayda değer ölçüde düşürdüğü gözlemlenmiştir.

Floater tarafından barisentrik izdüşüm teorisine dayalı olarak gerçekleştirilen izdüşüm yaklaşımında bir düğüme ilişkin komşuların ağırlık değerleri basitçe komşu sayısı ile ters orantılı olacak şekilde eşit olarak belirlenir. Önerilen başlangıç izdüşüm yaklaşımında bu ağırlık değerlerinin hesabı komşularla aradaki Öklid mesafesinin çarpmaya göre tersiyle orantılı olarak hesaplandıktan sonra normalizasyon değerine bölünerek daha hassas ve detaylı bir şekilde hesaplanmıştır. Bu sayede 3B uzaydaki yerel geometri bilgisi düzlemsel geliştirmenin yapıldığı parametrik uzaya etkin bir biçimde aktarılmış olur. Bu komşuluk ağırlık değerlerinin tutulduğu matrisin izdüşüm doğruluğuna etkisi Bölüm 4.1.3'te örnek yüzey yamaları üzerinde yapılan testlerde ortaya konulmuştur. Buna göre önerilen yaklaşım, aynı iterasyon miktarında alan-doğruluk ve kenar-doğruluk farkları bakımından temel yöntemle göre çok daha başarılı düzleştirme sonuçları üretmiştir.

Önerilen yöntem serbest biçimli üçgensel yüzey yamalarının, daha sonraki iyileştirme süreçlerinde girdi olarak kullanılabilen düzlemsel karşılıklarını hesaplayan başlangıç izdüşüm sonuçlarını üretir. Bölüm 4.1.4'te Floater'ın ters matris çözümüne dayalı olarak hesaplanan temel yöntemi ve bu çalışmada önerilen yöntemin izdüşüm süreleri çeşitli büyüklüklerdeki yüzey yamaları üzerinde test edilmiştir. Elde edilen sonuçlar önerilen yöntemin görece olarak büyük yüzeylerde (1.5K'dan daha fazla köşeye sahip) hesaplama süresi bakımından temel yönteme göre üstünlük sağladığını göstermiştir. Önerilen yaklaşımda kullanılan dinamik sinir ağı, ters matris hesaplamasında gereken yoğun işlemleri yapmadan düzlemsel yüzeyi oluşturan düğümleri komşularının ağırlık merkezlerine konumlanacak şekilde yinelemeli olarak hareket ettirir. Elde edilen kıyaslama sonuçları, gereken hesaplama yükü bakımından önerilen algoritmanın etkinliğini doğrulamaktadır.

Önerilen yaklaşımdaki dinamik sinir ağ yapısı Floater'ın temel yöntemindeki ters matris çözümünün yerini almaktadır. Ters matris çözümüne dayalı yöntemde, Hilbert matrisi gibi kötü koşullu matrisin belirli koşullar altındaki durumlarında, Bölüm 3.5.2.2'de anlatıldığı gibi Gauss-Jordan eliminasyonu veya LU faktörizasyon gibi normal doğrudan yöntemlerle kesin sayısal çözümü elde etmek imkânsızlaşır. Önerilen yaklaşımdaki dinamik sinir ağının sinirsel durumu bu kötü koşullu matrisin eşsiz çözümüne yakınsayabilmektedir.

Bu çalışmada mekanik düzleştirme sürecinde kullanılan yaklaşım Wang vd. (2002) tarafından benimsenen enerji modelidir. Temel enerji yayma algoritmasında sabit adım büyüklüğü ile hesaplanan ivme, hız ve konum denklemleri, bu çalışmada değişken adım büyüklüğüne dayalı olarak hesaplanmıştır. Önerilen değişken adım büyüklüğüne dayalı enerji yayma algoritmasının enerji yayma süresine etkisi Bölüm 4.2.2'de ayrıntılı olarak verilmiştir. Elde edilen sonuçlar değişken adım büyüklüğüne dayalı enerji yayma algoritmasının sabit adımlı yaklaşıma göre daha az sayıda iterasyon gerektirdiğini, yani daha hızlı bir şekilde enerji yayma işini yaptığını göstermiştir.



Konu ile ilgili gelecekteki arařtırmalar, algoritmanın FPGA (Field Programmable Gate Array) veya VLSI (Very-Large-Scale Integration) ile gereklemesi üzerine yoęunlařtırılabilir. Barisentrik izdüşüm teorisi, doęası gereęi kafes yüzeyin disk biçiminde olmasını gerekli kılar. Ancak bu teoriye uygun olmayan daha karmaşık bir örgü yüzey parametrelendirilecekse, o zaman karmaşık yüzey ileriki bir alıřmada geliştirilecek adaptif bir algoritma ile daha küçük alt yüzeylere ayrılabilir. Daha sonra alt yüzeyler ayrı ayrı disk biçiminde düzleřtirilebilir ve nihai olarak genel izdüşüme ulařmak için bu paralar bir araya getirilebilir.



## KAYNAKLAR

- Abdul-Rahman H.S., Jiang X.J., Scott P.J., 2013. Freeform surface filtering using the lifting wavelet transform. *Precision Engineering*, 37(1), 187-202.
- Abdul-Rahman H.S., Lou S., Zeng W., Jiang X., Scott P.J., 2016. Freeform texture representation and characterisation based on triangular mesh projection techniques, *Measurement*, 92, 172-182.
- Azariadis P., Aspragathos N., 1997. Design of plane developments of doubly curved surfaces, *Computer-Aided Design*, 29(10), 675-685.
- Azariadis, P.N., Aspragathos N.A., 2000. On using planar developments to perform texture mapping on arbitrarily curved surfaces. *Computers & Graphics*, 24(4), 539-554.
- Azariadis P.N., Nearchou A.C., Aspragathos N.A., 2002. An evolutionary algorithm for generating planar developments of arbitrarily curved surfaces, *Computers in Industry*, 47(3), 357-368.
- Azariadis P., Sapidis N., 2003. *Planar development of digital free-form surfaces*, *Advances in geometric modeling*, Wiley.
- Bodduluri R.M.C., Ravani B., 1993. Design of developable surfaces using duality between plane and point geometries, *Computer-Aided Design*, 25(10), 621-632.
- Botsch M., Kobbelt L., Pauly M., Alliez P., Lévy B., 2010. *Polygon mesh processing*. CRC press, 230p.
- Cai Z., Li M., Guo W., 2002. Calculating blank of shell-like parts based on triangular mesh system. *Engineering and Technology Edition*, 32(2).
- Carmo M.P., 1976. *Differential Geometry of Curves and Surfaces*, Prentice-Hall, 503, Englewood Cliffs.
- Chen D.R., Wang G.J., 2003. A complex surface adaptive segment and development algorithm based on its quasi-rulings, *Journal of Software*, 14(3), 660-665.
- Cho W., Patrikalakis N.M., Peraire J., 1998. Approximate development of trimmed patches for surface tessellation, *Computer-Aided Design*, 30(14), 1077-1087.
- Chung K., Richmond O., 1992. Ideal forming—I. Homogeneous deformation with minimum plastic work, *International Journal of Mechanical Sciences*, 34(7), 575-591.

- Chung K., Richmond O., 1992. Ideal forming—II. Sheet forming with optimum deformation. *International Journal of Mechanical Sciences*, 34(8), 617-633.
- Chung K., Yoon J.W., Richmond O., 2000. Ideal sheet forming with frictional constraints, *International Journal of Plasticity*, 16(6), 595-610.
- Chung W., Kim S.H., Shin K.H., 2008. A method for planar development of 3D surfaces in shoe pattern design. *Journal of Mechanical Science and Technology*, 22(8), 1510-1519.
- De Loera J.A., Rambau J., Santos F., 2010. *Triangulations Structures for Algorithms and Applications*. Springer-Verlag Berlin Heidelberg 2010.
- Du, K.L., Swamy, M.N., 2006. *Neural networks in a softcomputing framework*. Springer Science & Business Media.
- Delaunay B., 1934. Sur la sphère vide. *Bulletin de l'Académie des Sciences de l'URSS, Classe des sciences mathématiques et naturelles.*, 6: 793–800.
- Edelsbrunner H., Tan T.S., Waupotitsch R., 1992. An  $O(n^2 \log n)$  Time Algorithm for the Minmax Angle Triangulation. *SIAM Journal on Scientific and Statistical Computing*, 13(4), 994-1008.
- Eigensatz M., Kilian M., Schiftner A., Mitra N.J., Pottmann H., Pauly M., 2010. Paneling architectural freeform surfaces, *ACM transactions on graphics*, 29(4), 45.
- Floater M.S., 1997. Parametrization and smooth approximation of surface triangulations, *Computer Aided Geometric Design*, 14(3), 231-250.
- Gartner B., Hoffmann M., 2013. *Delaunay Triangulations*. Computational Geometry Lecture Notes HS, 2013.
- Hagan M.T., Demuth H.B., Beale, M.H., Jesus O.D., 2014. *Neural Network Design*, Boston Pws Pub, 1012p, Boston.
- Hinds B.K., McCartney J., Woods G., 1991. Pattern development for 3D surfaces, *Computer-aided design*, 23(8), 583-592.
- Hopfield, J.J., 1982. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554-2558.
- Hopfield, J.J., 1984. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences*, 81(10), 3088-3092.

- Hopfield, J.J., Tank, D.W., 1985. Neural computation of decisions in optimization problems. *Biological Cybernetics*, 52(3), 141-152.
- Hormann K., Lévy B., Sheffer A., 2007. *Mesh Parameterization: Theory and Practice*, ACM SIGGRAPPH, course notes.
- Hurtado F., Noy M., Urrutia J., 1999. Flipping edges in triangulations. *Discrete & Computational Geometry*, 22(3), 333-346.
- Hush, D.R., Horne, B.G., 1993. Progress in supervised neural networks. *IEEE Signal Processing Magazine*, 10(1), 8-39.
- Jiang X., Abdul-Rahman H.S., Scott P.J., 2013a. Lifting wavelet algorithm for freeform surface filtering using a Gaussian prediction operator, *International Journal of Precision Technology*, 3(3), 244-260.
- Jiang X.J., Abdul-Rahman H.S., Scott P.J., 2013b. Multi-scale freeform surface texture filtering using a mesh relaxation scheme. *Measurement Science and Technology*, 24(11), 115001.
- Jiang X., Cooper P., Scott P.J., 2011. Freeform surface filtering using the diffusion equation. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 467(2127), 841-859.
- Jiang X.J., Whitehouse D. J., 2012. Technological shifts in surface metrology, *CIRP Annals-Manufacturing Technology*, 61(2), 815-836.
- Jiang X., Scott P.J., Whitehouse D. J., Blunt L., 2007a. Paradigm shifts in surface metrology. Part I. Historical philosophy. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 463(2085), 2049-2070.
- Jiang X., Scott P.J., Whitehouse D. J., Blunt L., 2007b. Paradigm shifts in surface metrology. Part II. The current shift. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 463(2085), 2071-2099.
- Kohonen, T., 1988. An introduction to neural computing. *Neural Networks*, 1(1), 3-16.
- Lan J., Dong X, Li Z., 2001. Blank and strain estimates for sheet metal forming processes by an inverse approach of FE, *Journal of Plasticity Engineering*, 8, 60-62.
- Lee C.H., Huh H., 1998. Blank design and strain estimates for sheet metal forming processes by a finite element inverse approach with initial guess of linear deformation, *Journal of Materials Processing Technology*, 82(1-3), 145-155.

- Lee Y., Kim H.S., Lee S., 2002. Mesh parameterization with a virtual boundary. *Computers & Graphics*, 26(5), 677-686.
- Lévy B., Petitjean S., Ray N., Maillot J., 2002. Least squares conformal maps for automatic texture atlas generation, *ACM Transactions on Graphics*, 21(3), 362-371.
- Li S., Liu X., Zheng X., Lu H., 2014. An overview of flattening methods for complex surface. *Journal of Information & Computational Science*, 11(1), 323-333.
- Li Z., Zhu Q., Gold C., 2004. *Digital terrain modeling: principles and methodology*. CRC press.
- Liu C., Luo Z., Shi X., Liu F., Luo X., 2013. A fast mesh parameterization algorithm based on 4-point interpolatory subdivision, *Applied Mathematics and Computation*, 219(10), 5339-5344.
- Liu Q., Xi J., Wu Z., 2013. An energy-based surface flattening method for flat pattern development of sheet metal components, *The International Journal of Advanced Manufacturing Technology*, 68(5-8), 1155-1166.
- Lou S., Jiang X., Scott P.J., 2012a. Algorithms for morphological profile filters and their comparison. *Precision Engineering*, 36(3), 414-423.
- Lou S., Zeng W., Jiang X., Scott P.J., 2012b. Comparison of robust filtration techniques in geometrical metrology, 18th International Conference on Automation and Computing (ICAC), United Kingdom, 7-8 September, 1-6.
- Madankan A., 2010. Recurrent neural network for solving linear matrix equation, *International Conference On Electronics and Information Engineering (ICEIE)*, Vol. 2, pp. V2-70.
- McCartney J., Hinds B.K., Seow B.L., Gong D., 2000. An energy based model for the flattening of woven fabrics, *Journal of Materials Processing Technology*, 107(1-3), 312-318.
- McCulloch, W.S., Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115-133.
- Metaxas D.N., 1997. *Physics-based deformable models*. Kluwer Academic, 275p, Dordrecht.
- Pineda, F.J., 1988. Dynamics and architecture for neural computation. *Journal of Complexity*, 4(3), 216-245.
- Ping X.I., 1997. Geometric approach of 3D surface Development, *Chinese Journal of Computers*, 4, 315-322.

- Ping X., Zhang H., Xiong X., Wei G., 1998. Geometric arithmetic of development of sheet metal parts, *Journal of Engineering Graphics*, 3, 94-100.
- Provot X., 1995. Deformation constraints in a mass-spring model to describe rigid cloth behaviour. In *Graphics interface* (pp. 147-147). Canadian Information Processing Society.
- Shin K.H., 2011. A method for planar development of free-form surfaces made of anisotropic materials, *Journal of Mechanical Science and Technology*, 25(11), 2817-2825.
- Sheffer A., de Sturler E., 2001. Parameterization of faceted surfaces for meshing using angle-based flattening, *Engineering with Computers*, 17(3), 326-337.
- Sheffer A., Lévy B., Mogilnitsky M., Bogomyakov A., 2005. ABF++: fast and robust angle based flattening, *ACM Transactions on Graphics*, 24(2), 311-330.
- Sheffer A., Praun E., Rose K., 2007. Mesh parameterization methods and their applications, *Foundations and Trends in Computer Graphics and Vision*, 2(2), 105-171.
- Su K., Cui L., Qian K., Lei N., Zhang J., Zhang M., Gu X.D., 2016. Area-preserving mesh parameterization for poly-annulus surfaces based on optimal mass transportation, *Computer Aided Geometric Design*, 46, 76-91.
- Su B., Hua X., Xin Y., 1998. *An Introduction to Practical Differential Geometry*, Science Press, Beijing.
- Tutte W.T., 1960. Convex representations of graphs, *Proceedings of the London Mathematical Society*, 3(1), 304-320.
- Tutte W.T., 1963. How to draw a graph, *Proceedings of the London Mathematical Society*, 3(1), 743-767.
- Wang C.C.L., 2013. *Geometric Modeling and Reasoning of Human-Centered Freeform Products*, Springer-Verlag, 226, London.
- Wang C.C., Smith S.S., Yuen M.M., 2002. Surface flattening based on energy model, *Computer-Aided Design*, 34(11), 823-833.
- Wang J., Jiang X., Blunt L.A., Leach R.K., Scott P.J., 2012. Intelligent sampling for the measurement of structured surfaces. *Measurement Science and Technology*, 23(8), 085006.
- Wang H., Wang C.C., 2001. A general surface development algorithm based on energy model, *Journal of Computer-Aided Design and Computer Graphics*, 13(6), 556.

- Xu R., Liu X., Chen J., 2002. Development and status of surfaces flattening method, *Die and Mould Technology*, 5, 15-18.
- Xu J., Liu Y., Du T., Zhang Z., 2008. Algorithm for arbitrary complex surface flattening and its application in unfolding metal blank, *China Mechanical Engineering*, 19(2), 193-195.
- Yang Y.J., Zeng W., Meng X.X., 2016. Conformal freeform surfaces. *Computer-Aided Design*, 81, 48-60.
- Yoon J.W., Chung K., Pourboghrat F., Barlat F., 2006. Design optimization of extruded preform for hydroforming processes based on ideal forming design theory, *International Journal of Mechanical Sciences*, 48(12), 1416-1428.
- Yoshizawa S., Belyaev A., Seidel H.P., 2004. A fast and simple stretch-minimizing mesh parameterization. In *Shape Modeling Applications, Proceedings IEEE*, 200-208.
- Yu X., Loh N.K., Miller W.C., 1992. New recursive algorithm for solving linear algebraic equations, *Electronics Letters*, 28(22), 2069-2071.
- Zhan W., 2007. Research on Pattern Flattening for Parts with Complex Surfaces. Nanjing University, Aeronautics & Astronautics, M.Sc. Thesis.
- Zhang Y., Ge S.S., 2005. Design and analysis of a general recurrent neural network model for time-varying matrix inversion, *IEEE Transactions on Neural Networks*, 16(6), 1477-1490.
- Zhang Y., Jiang D., Wang J., 2002. A recurrent neural network for solving Sylvester equation with time-varying coefficients, *IEEE Transactions on Neural Networks*, 13(5), 1053-1063.
- Zhang Q.L., Luo X.Q., 2003. Finite element method for developing arbitrary surfaces to flattened forms, *Finite Elements in Analysis and Design*, 39(10), 977-984.
- Zheleznyakova A.L., 2015. Molecular dynamics-based triangulation algorithm of free-form parametric surfaces for computer-aided Engineering, *Computer Physics Communications*, 190, 1-14.

## **EKLER**

**EK A.** Denklemler

**EK B.** Şekiller





## EK A. Denklemler

$$\Lambda^T = \begin{bmatrix} -1 & 0 & 0 & 0 & 0.27 & 0 & 0 & 0 & 0 & 0.18 & 0 & 0 & 0.19 & 0 & 0 & 0 \\ 0 & -1 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0.16 & 0 & 0.15 & 0 & 0.32 & 0 & 0 \\ 0 & 0.2 & -1 & 0 & 0 & 0 & 0 & 0 & 0.14 & 0 & 0 & 0.15 & 0 & 0.13 & 0.19 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0.36 & 0.13 & 0.13 & 0 & 0.19 & 0 & 0 & 0 & 0 & 0.12 & 0 \\ 0.49 & 0 & 0 & 0 & -1 & 0 & 0.09 & 0 & 0.15 & 0.37 & 0 & 0.2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.32 & 0 & -1 & 0 & 0 & 0 & 0.18 & 0 & 0 & 0 & 0 & 0.12 & 0.28 \\ 0 & 0 & 0 & 0.17 & 0.08 & 0 & -1 & 0.44 & 0.1 & 0.09 & 0.28 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.2 & 0.15 & 0 & 0 & 0.39 & -1 & 0 & 0 & 0.26 & 0 & 0 & 0 & 0.1 & 0 \\ 0.29 & 0.27 & 0 & 0 & 0.13 & 0 & 0.1 & 0 & -1 & 0 & 0.16 & 0.15 & 0.16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.26 & 0.34 & 0.28 & 0.09 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0.18 \\ 0 & 0.18 & 0.19 & 0 & 0 & 0 & 0.21 & 0.22 & 0.12 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0.22 & 0 & 0 & 0 & 0.12 & 0 & 0 & 0 & 0.11 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0.35 & 0.14 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0.19 & 0.1 & 0 & 0.11 & 0 & 0.07 & 0 & 0 & 0 & 0 & 0 & -1 & 0.11 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0.1 & 0 & 0 & 0 & 0 & 0.11 & -1 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.07 & 0 & 0 & 0 & 0 & 0 & 0.08 & 0 & 0.1 & 0 & 0 & 0.15 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.18 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.11 & 0.1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.13 & 0 \\ 0 & 0 & 0.09 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.09 & 0.11 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.16 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.08 & 0 & 0 & 0 & 0.1 & 0.14 & 0 & 0 \end{bmatrix} \quad (\text{A.1})$$

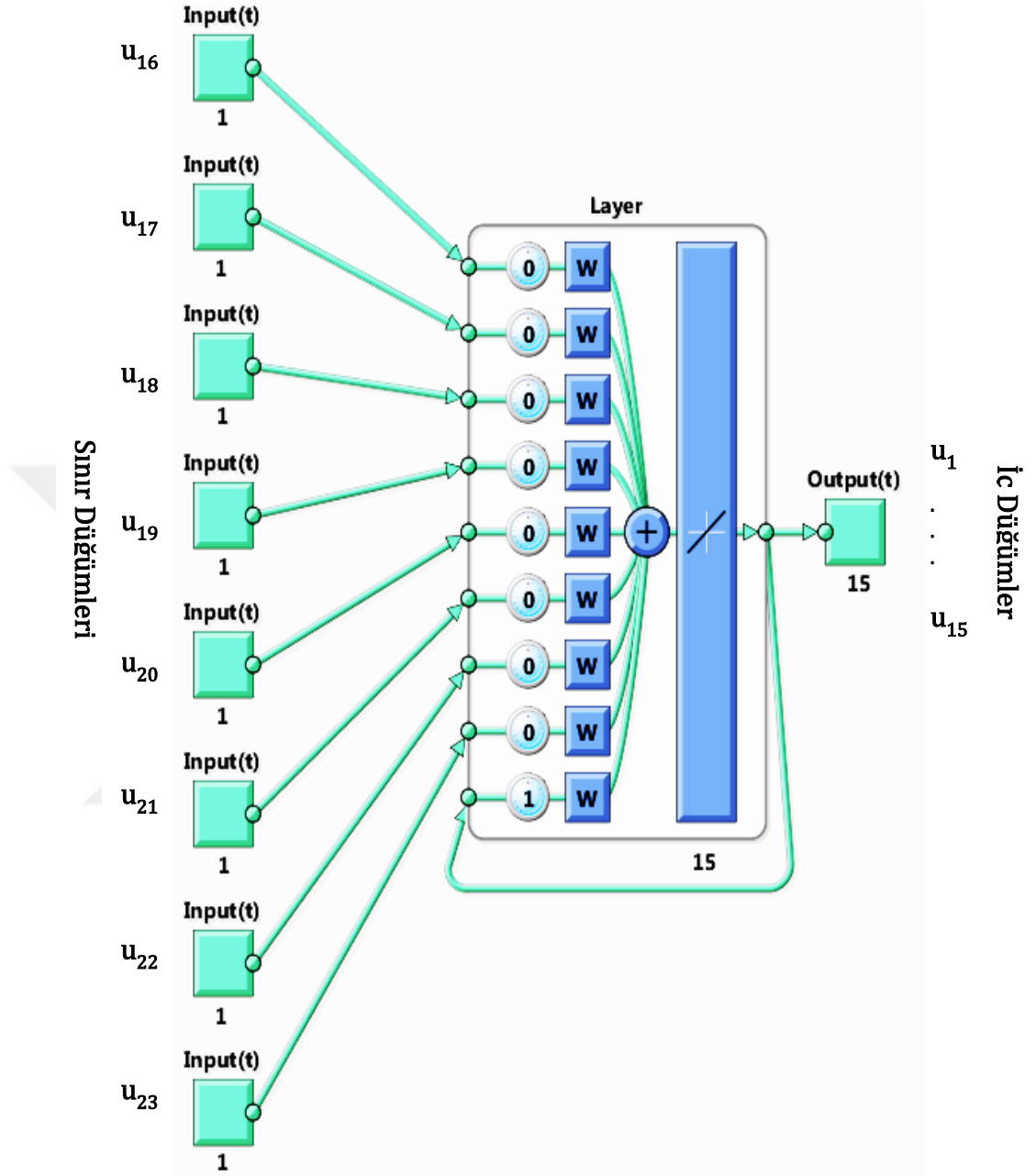
$$\mathbf{X} = \begin{bmatrix}
 0.15 & 0.2 & 0.5 \\
 0.1 & 0.9 & 0.3 \\
 0.15 & 0.8 & 0.4 \\
 0.2 & 0.75 & 0.35 \\
 0.7 & 0.9 & 0.45 \\
 0.7 & 0.3 & 0.55 \\
 0.6 & 0.8 & 0.5 \\
 0.65 & 0.75 & 0.6 \\
 0.3 & 0.1 & 0.4 \\
 0.3 & 0.2 & 0.35 \\
 0.85 & 0.15 & 0.35 \\
 0.5 & 0.55 & 1 \\
 0.6 & 0.7 & 1 \\
 0.6 & 0.9 & 0.95 \\
 0.08 & 0.48 & 0.42 \\
 0.48 & 0.08 & 0.38 \\
 0.25 & 0.75 & 0.99 \\
 0.4 & 0.85 & 0.95 \\
 0.8 & 0.65 & 1 \\
 0.3 & 0.4 & 0.85 \\
 0.8 & 0.3 & 1 \\
 0.4 & 0.28 & 0.5 \\
 0.9 & 0.6 & 0.6 \\
 0.9 & 0.8 & 0.45 \\
 0.51 & 0.31 & 0.6 \\
 0.65 & 0.45 & 0.85 \\
 0.1 & 0.1 & 0 \\
 0.05 & 0.95 & 0 \\
 0.95 & 0.95 & 0 \\
 0.9 & 0.1 & 0 \\
 0.3 & 0 & 0 \\
 0.7 & 0.05 & 0 \\
 0.2 & 1 & 0 \\
 0.7 & 1 & 0 \\
 0 & 0.35 & 0 \\
 0 & 0.7 & 0 \\
 1 & 0.45 & 0 \\
 1 & 0.75 & 0 \\
 0.95 & 0.2 & 0 \\
 0.05 & 0.2 & 0
 \end{bmatrix}, \mathbf{F} = \begin{bmatrix}
 24 & 38 & 29 \\
 33 & 28 & 2 \\
 37 & 38 & 23 \\
 23 & 24 & 19 \\
 38 & 24 & 23 \\
 5 & 24 & 29 \\
 33 & 2 & 3 \\
 12 & 13 & 18 \\
 18 & 14 & 33 \\
 8 & 13 & 19 \\
 19 & 24 & 8 \\
 19 & 24 & 8 \\
 24 & 5 & 8 \\
 34 & 5 & 29 \\
 14 & 5 & 34 \\
 33 & 14 & 34 \\
 12 & 20 & 25 \\
 25 & 20 & 22 \\
 25 & 22 & 16 \\
 16 & 6 & 25 \\
 37 & 23 & 21 \\
 21 & 39 & 37 \\
 32 & 6 & 16 \\
 16 & 31 & 32 \\
 4 & 3 & 36 \\
 36 & 2 & 28 \\
 36 & 3 & 2 \\
 12 & 18 & 17 \\
 17 & 20 & 12 \\
 17 & 3 & 4 \\
 33 & 3 & 7 \\
 17 & 18 & 33 \\
 14 & 18 & 7 \\
 7 & 18 & 13 \\
 13 & 8 & 7 \\
 7 & 5 & 14 \\
 7 & 8 & 5 \\
 16 & 22 & 10 \\
 22 & 20 & 10 \\
 20 & 1 & 10 \\
 27 & 1 & 40 \\
 40 & 1 & 35 \\
 12 & 25 & 26 \\
 26 & 25 & 6 \\
 26 & 13 & 12 \\
 19 & 13 & 26 \\
 6 & 21 & 26 \\
 26 & 23 & 19 \\
 26 & 21 & 23 \\
 11 & 21 & 6 \\
 6 & 32 & 11 \\
 11 & 32 & 30 \\
 30 & 39 & 11 \\
 39 & 21 & 11 \\
 4 & 36 & 15 \\
 15 & 17 & 4 \\
 20 & 17 & 15 \\
 15 & 36 & 35 \\
 35 & 1 & 15 \\
 15 & 1 & 20 \\
 9 & 1 & 27 \\
 27 & 31 & 9 \\
 9 & 10 & 1 \\
 9 & 31 & 16 \\
 16 & 10 & 9
 \end{bmatrix}$$

(A.2)

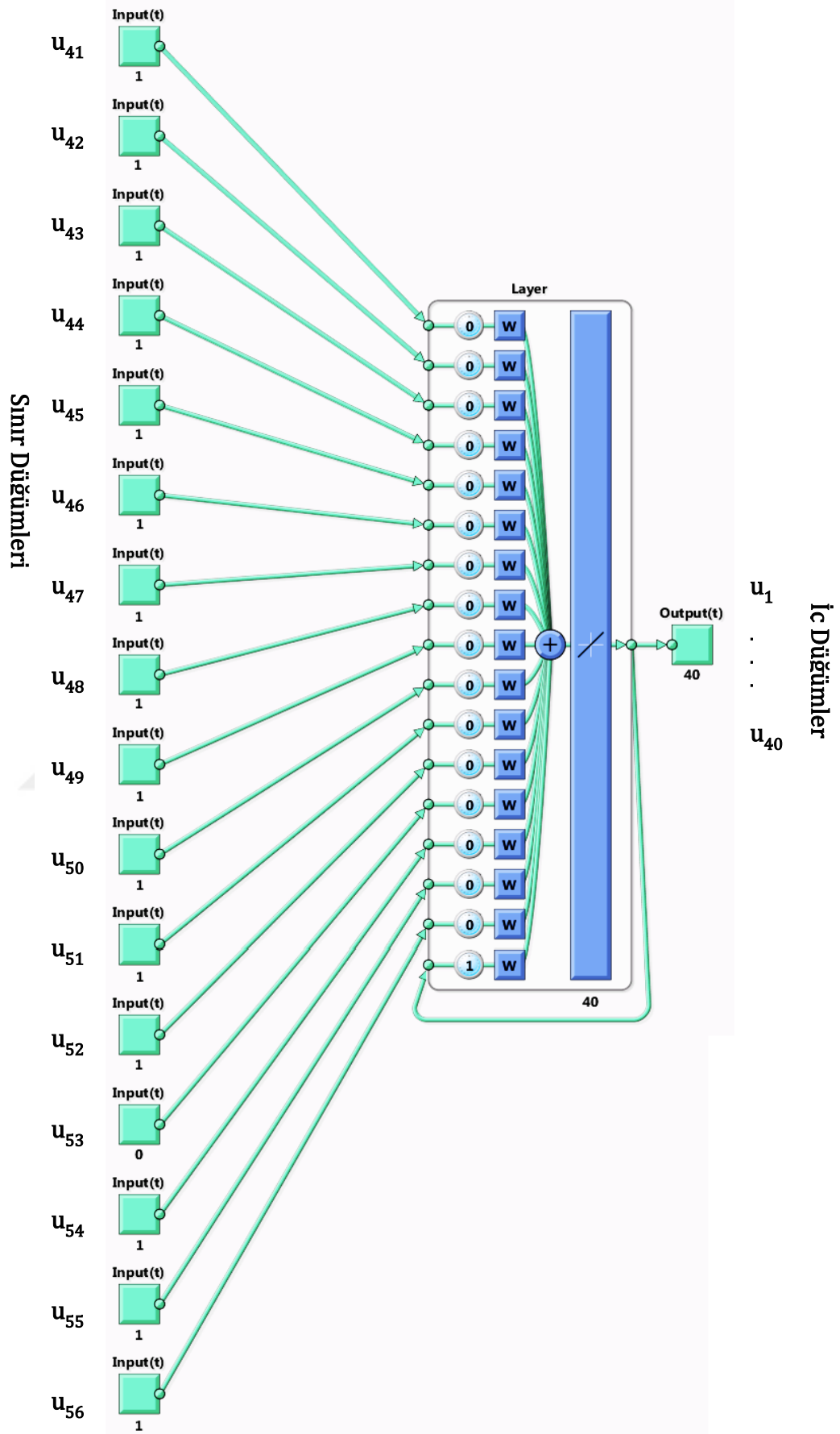
$$\mathbf{X}^* = \begin{bmatrix}
 rx+0.15 & ry+0.2 & 0.5 \\
 rx+0.1 & ry+0.9 & 0.3 \\
 rx+0.15 & ry+0.8 & 0.4 \\
 rx+0.2 & ry+0.75 & 0.35 \\
 rx+0.7 & ry+0.9 & 0.45 \\
 rx+0.7 & ry+0.3 & 0.55 \\
 rx+0.6 & ry+0.8 & 0.5 \\
 rx+0.65 & ry+0.75 & 0.6 \\
 rx+0.3 & ry+0.1 & 0.4 \\
 rx+0.3 & ry+0.2 & 0.35 \\
 rx+0.85 & ry+0.15 & 0.35 \\
 rx+0.5 & ry+0.55 & 1 \\
 rx+0.6 & ry+0.7 & 1 \\
 rx+0.6 & ry+0.9 & 0.95 \\
 rx+0.08 & ry+0.48 & 0.42 \\
 rx+0.48 & ry+0.08 & 0.38 \\
 rx+0.25 & ry+0.75 & 0.99 \\
 rx+0.4 & ry+0.85 & 0.95 \\
 rx+0.8 & ry+0.65 & 1 \\
 rx+0.3 & ry+0.4 & 0.85 \\
 rx+0.8 & ry+0.3 & 1 \\
 rx+0.4 & ry+0.28 & 0.5 \\
 rx+0.9 & ry+0.6 & 0.6 \\
 rx+0.9 & ry+0.8 & 0.45 \\
 rx+0.51 & ry+0.31 & 0.6 \\
 rx+0.65 & ry+0.45 & 0.85 \\
 rx+0.1 & ry+0.1 & 0 \\
 rx+0.05 & ry+0.95 & 0 \\
 rx+0.95 & ry+0.95 & 0 \\
 rx+0.9 & ry+0.1 & 0 \\
 rx+0.3 & ry+0 & 0 \\
 rx+0.7 & ry+0.05 & 0 \\
 rx+0.2 & ry+1 & 0 \\
 rx+0.7 & ry+1 & 0 \\
 rx+0 & ry+0.35 & 0 \\
 rx+0 & ry+0.7 & 0 \\
 rx+1 & ry+0.45 & 0 \\
 rx+1 & ry+0.75 & 0 \\
 rx+0.95 & ry+0.2 & 0 \\
 rx+0.05 & ry+0.2 & 0 \\
 0.42 & 0.42 & 0 \\
 0.88 & 0.11 & 0 \\
 1.42 & 0 & 0 \\
 1.97 & 0.11 & 0 \\
 2.43 & 0.42 & 0 \\
 2.74 & 0.88 & 0 \\
 2.84 & 1.42 & 0 \\
 2.74 & 1.97 & 0 \\
 2.43 & 2.43 & 0 \\
 1.97 & 2.74 & 0 \\
 1.42 & 2.84 & 0 \\
 0.88 & 2.74 & 0 \\
 0.42 & 2.43 & 0 \\
 0.11 & 1.97 & 0 \\
 0 & 1.42 & 0 \\
 0.11 & 0.88 & 0
 \end{bmatrix}, \mathbf{F}^* = \begin{bmatrix}
 55 & 28 & 54 \\
 54 & 28 & 53 \\
 49 & 29 & 48 \\
 50 & 29 & 49 \\
 36 & 28 & 55 \\
 55 & 35 & 36 \\
 56 & 35 & 55 \\
 33 & 52 & 53 \\
 53 & 28 & 33 \\
 38 & 37 & 47 \\
 47 & 48 & 38 \\
 38 & 48 & 29 \\
 47 & 37 & 46 \\
 39 & 45 & 46 \\
 46 & 37 & 39 \\
 41 & 42 & 31 \\
 27 & 41 & 31 \\
 31 & 42 & 43 \\
 43 & 32 & 31 \\
 43 & 44 & 30 \\
 30 & 32 & 43 \\
 30 & 44 & 45 \\
 30 & 45 & 39 \\
 40 & 56 & 41 \\
 40 & 41 & 27 \\
 35 & 56 & 40 \\
 51 & 52 & 34 \\
 52 & 33 & 34 \\
 34 & 50 & 51 \\
 29 & 50 & 34 \\
 28 & 36 & 2 \\
 2 & 33 & 28 \\
 24 & 38 & 29 \\
 39 & 37 & 21 \\
 27 & 31 & 9 \\
 15 & 36 & 35 \\
 25 & 20 & 22 \\
 22 & 20 & 10 \\
 1 & 40 & 27 \\
 27 & 9 & 1 \\
 1 & 9 & 10 \\
 10 & 20 & 1 \\
 35 & 40 & 1 \\
 1 & 15 & 35 \\
 20 & 15 & 1 \\
 25 & 26 & 12 \\
 12 & 20 & 25 \\
 29 & 34 & 5 \\
 5 & 24 & 29 \\
 8 & 24 & 5 \\
 5 & 7 & 8 \\
 19 & 24 & 8 \\
 32 & 30 & 11 \\
 11 & 30 & 39 \\
 39 & 21 & 11 \\
 6 & 21 & 26 \\
 6 & 26 & 25 \\
 32 & 11 & 6 \\
 6 & 11 & 21 \\
 16 & 31 & 32 \\
 16 & 9 & 31 \\
 25 & 22 & 16 \\
 \cdot & \cdot & \cdot \\
 \cdot & \cdot & \cdot \\
 \cdot & \cdot & \cdot
 \end{bmatrix}$$

(A.3)

## EK B. Şekiller



Şekil B.1. *Surf2* yüzey yaması için oluşturulan dinamik sinir ağının yapısı



Őekil B.2. Surf3 y zey yaması iin oluŐturulan dinamik sinir ađının yapısı

## ÖZGEÇMİŞ

Adı Soyadı : Erdem YAVUZ  
Doğum Yeri ve Yılı : İSTANBUL, 1984  
Yabancı Dili : İngilizce  
E-posta : eyavuz@ticaret.edu.tr



### Eğitim Durumu

Lise : Tuzla Teknik Lisesi, 2002  
Lisans : Marmara Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Anabilim Dalı, Bilgisayar ve Kontrol Teknolojisi Bölümü, 2008  
Yüksek Lisans : Marmara Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik-Bilgisayar Eğitimi Anabilim Dalı, Bilgisayar-Kontrol Eğitimi Bölümü, 2010

### Mesleki Deneyim

İstanbul Ticaret Üniversitesi,  
Mühendislik Fakültesi,  
Bilgisayar Mühendisliği Bölümü,  
Araştırma Görevlisi 2008-(devam ediyor)

### Yayınları

- Yavuz, E., Yazıcı, R., 2018. A dynamic neural network model for accelerating preliminary parameterization of 3D triangular mesh surfaces, Neural Computing & Applications, 1(0), 1-11. **(SCI-Expanded)**
- Yavuz, E., Topuz, V., 2018. A phoneme-based approach for eliminating out-of-vocabulary problem of Turkish speech recognition using Hidden Markov Model (ACCEPTED), Computer Systems Science and Engineering, 1(1), 0-0. **(SCI-Expanded)**
- Yavuz, E., Kasapbaşı, M.C., Eyüpoğlu, C., Yazıcı, R., 2018. An epileptic seizure detection system based on cepstral analysis and generalized regression neural network. Biocybernetics and Biomedical Engineering, 38(2), 201-216. **(SCI-Expanded)**

- Yavuz, E., Yazıcı, R., Kasapbaşı, M.C., Yamaç, E., 2016. A chaos-based image encryption algorithm with simple logical functions, *Computers & Electrical Engineering*, 54(1), 471-483. **(SCI-Expanded)**
- Yavuz, E., Şanver, U., Kasapbaşı, M.C., Yazıcı, R., 2015. An Implementation Of Vibration-Based Automatic Score-Keeping System For Table Tennis Game, *International Journal of Electronics; Mechanical and Mechatronics Engineering*, 4(1), 725-732.
- Özpınar, A., Yavuz, E., 2011. E-Commerce Training with Virtual Commerce Simulation, *International Journal of E-Adoption*, 3(2), 20-26.
- Şanver, U., Yavuz, E., Eyüpoğlu, C., Uzun, T., 2018. Design and Implementation of a Programmable Logic Controller Using PIC18F4580, 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus2018), St. Petersburg, 29 Ocak-1 Şubat, 231-235.
- Ünal, M.S., Ertürk, E., Kasapbaşı, M. C., Yavuz, E., Şanver, U., 2017. Real Time Remote Control of Robotic Hand System - TeleTouch, *International Conference on Engineering Technologies (ICENTE'17)*, Konya, 7-9 Aralık, 648-652.
- Yavuz, E., Eyüpoğlu, C., Şanver, U., Yazıcı, R., 2017. An Ensemble of Neural Networks for Breast Cancer Diagnosis, *IEEE 2nd International Conference on Computer Science and Engineering (UBMK'17)*, Antalya, 5-7 Ekim, 538-543.
- Şanver, U., Yavuz, E., Eyüpoğlu, C., 2017. An Image Processing Application to Detect Faulty Bottle Packaging, 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (2017 EIconRus), St. Petersburg, 1-3 Şubat, 986-989.
- Şanver, U., Yavuz, E., Kasapbaşı, M. C., Yazıcı, R., 2016. Modelling and Implementation of an Automatic Table-Tennis Scoreboarding System, 2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (2016 EIconRusNW), St.Petersburg, 2-3 Şubat, 325-329.
- Yavuz, E., Yazıcı, R., Kasapbaşı, M. C., Yamaç, E., 2014. Enhanced Chaotic Key-Based Algorithm for Low-Entropy Image Encryption, *IEEE 22nd Signal Processing and Communications Applications Conference (SIU 2014)*, Trabzon, 23-25 Nisan, 385-388.
- Yavuz, E., Topuz, V., 2010, Recognition of Turkish Vowels by Probabilistic Neural Networks Using Yule-Walker AR Method, *5th International Conference on Hybrid Artificial Intelligent Systems (HAIS'10)*, San Sebastián, 23-25 Haziran, 112-119.

Yavuz, E., Çintay, H., Kaçan, C., Erdal, H., 2008. Akıllı Ev Otomasyonu İçin Mikrodenetleyici Tabanlı Kablosuz Modül Tasarımı, , VII. Ulusal Öğrenci Sempozyumu (UOS 2008), Süleyman Demirel Üniversitesi, Isparta, 6 Mayıs.

