

ZONGULDAK BÜLENT ECEVİT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BÉZİER EĞRİLERİ İLE İLGİLİ PYTHON UYGULAMALARI

MATEMATİK ANABİLİM DALI

YÜKSEK LİSANS TEZİ

RUMEYSA AKALIN

ZONGULDAK

OCAK 2024

ZONGULDAK BÜLENT ECEVİT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BÉZİER EĞRİLERİ İLE İLGİLİ PYTHON UYGULAMALARI

MATEMATİK ANABİLİM DALI

YÜKSEK LİSANS TEZİ

Rumeysa AKALİN

DANIŞMAN : Doç. Dr. Mehmet GÜMÜŞ

İKİNCİ DANIŞMAN : Dr. Öğr. Üyesi Seda KARATEKE

ZONGULDAK

Ocak 2024

KABUL:

Rumeysa AKALIN tarafından hazırlanan “Bézier Eğrileri İle İlgili Python Uygulamaları” başlıklı bu çalışma jürimiz tarafından değerlendirilerek Zonguldak Bülent Ecevit Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Anabilim Dalında Yüksek Lisans Tezi olarak oybirliğiyle kabul edilmiştir. 19/01/2024

Danışman: Doç. Dr. Mehmet GÜMÜŞ

Zonguldak Bülent Ecevit Üniversitesi, Fen Fakültesi, Matematik Bölümü

İkinci Danışman: Dr. Öğr. Üyesi Seda KARATEKE

İstanbul Aydın Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü

Üye: Prof. Dr. Metin ZONTUL

Sivas Bilim ve Teknoloji Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü

Üye: Doç. Dr. İsmet GÖLGELEYEN

Zonguldak Bülent Ecevit Üniversitesi, Fen Fakültesi, Matematik Bölümü

Üye: Doç. Dr. Nazmiye GÖNÜL BİLGİN

Zonguldak Bülent Ecevit Üniversitesi, Fen Fakültesi, Matematik Bölümü

ONAY:

Yukarıdaki imzaların, adı geçen öğretim üyelerine ait olduğunu onaylarım.

....../....../20....

Prof. Dr. Fikret GÖLGELEYEN
Fen Bilimleri Enstitüsü Müdürü



“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Rumeysa AKALİN

ÖZET

Yüksek Lisans Tezi

BÉZIER EĞRİLERİ İLE İLGİLİ PYTHON UYGULAMALARI

Rumeysa AKALIN

Zonguldak Bülent Ecevit Üniversitesi

Fen Bilimleri Enstitüsü

Matematik Anabilim Dalı

Tez Danışmanı: Doç. Dr. Mehmet GÜMÜŞ

İkinci Tez Danışmanı: Dr. Öğr. Üyesi Seda KARATEKE

Ocak 2024, 91 sayfa

Bilgisayar Destekli Tasarımın (BDT) gelişmesiyle teorik bilginin bilgisayar teknolojisi ile görselleştirilmesinin önemi giderek artmakta ve popülerliğini korumaktadır. Eğriler ve yüzeylerin bilgisayar sistemlerinde çizdirilebilmesinde en önemli öncüllerinden biri Bézier eğrileridir. Bu tezin amacı, Bilgisayar Destekli Geometrik Tasarım (BDGT)'da sıklıkla kullanılan Bézier eğrilerini teorik olarak araştırmak; açık kaynak ve öğrenmesi daha kolay ve esnek olan ve bu sebeplerden yapay zekâ alanında en çok tercih edilen dillerden biri olan Python 3.10 bilgisayar programlama dilinde Bézier eğrilerinin uygulamalarını sunmaktır. Yapılan uygulamalardan bazıları yapay zekânın alt dalı olarak bilinen Makine Öğrenmesi (MÖ) uygulamaları eğri uydurma (curve fitting) yöntemi ile ilgili olacaktır. Tez beş bölümden oluşmaktadır: Birinci bölümde; makine öğrenmesi (MÖ) ve Bézier eğrileri hakkında okuyucuyu motive edici bilgiler verilmiştir. İkinci bölümde; tezin geriye kalan kısmında kullanılacak olan Öklid uzayı, Minkowski uzayı ve Bézier eğrileri ile ilgili temel tanım ve teoremlere yer verilmiştir. Ayrıca temel Bézier eğrilerinin ve bazı örneklerinin grafiklerinin Python 3.10 bilgisayar programlama dilinde grafikleri çizdirilerek kullanılan kodlar tez

ÖZET (devam ediyor)

sonunda ek olarak sunulmuştur. Üçüncü ve dördüncü bölümlerde ise üç boyutlu Minkowski uzayında Bézier eğrilerinin Serret-Frenet çatılarıyla ilgili ispatlara ve bunlarla ilgili örneklerle yer verilmiştir. Beşinci bölümde ise dört adet farklı kontrol noktasına sahip Bézier eğrilerinin kontrol noktaları değiştirilerek elde edilen ve verilen parametre aralığındaki artış miktarı değiştirilerek oluşan görüntülerdeki farklılıklar Python 3.10 (64 bit) bilgisayar programlama dilinde çizdirilerek incelenmiştir.

Anahtar Kelimeler: Yapay Zekâ (YZ), Bézier eğrileri, Bernstein temel fonksiyonu, Minkowski uzayı, Serret-Frenet çatısı, eğri uydurma, python 3.10 uygulamaları.

Bilim Kodu: 403.06.00

ABSTRACT

M. Sc. Thesis

PYTHON APPLICATIONS ON BÉZIER CURVES

Rumeysa AKALIN

**Zonguldak Bülent Ecevit University
Graduate School of Natural and Applied Sciences
Department of Mathematics**

Thesis Advisor : Assoc. Prof. Dr. Mehmet GÜMÜŞ

Co-Advisor : Assist. Prof. Dr. Seda KARATEKE

January 2024, 91 page

With the development of Computer Aided Design (CAD), the importance of visualizing theoretical knowledge with computer technology is increasing and it maintains its popularity. One of the most important pioneer tools to draw curves and surfaces on computer systems is Bézier curves. The aim of this thesis is to theoretically investigate Bézier curves, which are frequently used in Computer Aided Geometric Design (CAGD), and to implement Bézier curves in the computer programming language Python 3.10, which is one of the most preferred languages in the field of Artificial Intelligence (AI) thanks to flexibility of its codes and, also being open source. Some of the presented applications are related to the curve fitting method of Machine Learning (ML) applications, known as the sub-branch of Artificial Intelligence (AI). The thesis consists of five chapters: In the first part of the thesis, motivating information is given to the reader about Machine Learning (ML) and Bézier curves. In the second chapter, basic definitions and theorems about Euclidean space, Minkowski space and Bézier curves, which will be used in the rest of the thesis, are given. In addition, the graphs of basic Bézier

ABSTRACT (continued)

curves and some of their examples are plotted employing computer programming language Python 3.10 and their codes are given as an appendix at the end of the thesis. In the third and fourth chapters, proofs of Serret-Frenet frameworks of Bézier curves in three-dimensional Minkowski space and examples are presented. In the fifth chapter, the image obtained by changing the control points of Bézier curves with four different control points and the differences in the images obtained by changing the amount of increase in the given parameter range are examined drawing them in the computer programming language Python 3.10 (64 bit).

Keywords: Artificial Intelligence (AI), Bézier curves, Bernstein basis function, Minkowski space, Serret-Frenet frame, curve fitting, python 3.10 applications.

Science Code: 403.06.00

TEŞEKKÜR

Yüksek lisans danışmanlığımı üstlenip özenle çalışmalarımı takip eden, her zaman engin bilgi ve deneyimleriyle yolumu açan, ilgi ve desteğini esirgemeyen değerli hocalarım sayın Doç. Dr. Mehmet GÜMÜŞ ve Dr. Öğr. Üyesi Seda KARATEKE' ye ve çalışmalarım boyunca desteğine ihtiyaç duyduğumuz her zaman bize tüm bilgi birikimiyle destek olan değerli hocam Prof. Dr. Metin ZONTUL' a en samimi duygularım ile sonsuz teşekkürlerimi sunarım.

Hayatımın her alanında tecrübeleri ve teşvikleriyle her zaman yanımda olan kıymetli abim Doç. Dr. Murat BEYTUR' a ve beni yetiştiren, bugüne gelmemde çok emeği olan maddi-manevi hiçbir desteği eksik etmeyen sevgili anneme, babama ve kıymetli ablalarım en samimi duygularımla teşekkürlerimi sunar ve onları çok sevdiğimi belirtmek isterim.

Son olarak da varlıklarıyla beni her zaman mutlu eden, neşe kaynağım olan yeğenlerim Ebrar ve Furkan' a sonsuz sevgilerimi sunarım.



İÇİNDEKİLER

	<u>Sayfa</u>
KABUL:	ii
ÖZET	iii
ABSTRACT	v
TEŞEKKÜR	vii
İÇİNDEKİLER.....	ix
ŞEKİLLER DİZİNİ.....	xiii
EK AÇIKLAMALAR DİZİNİ.....	xv
SİMGELER VE KISALTMALAR DİZİNİ.....	xvii
 BÖLÜM 1 GİRİŞ.....	 1
 BÖLÜM 2 TEMEL KAVRAMLAR	 5
 2.1 ÖKLİD UZAYINDA EĞRİLER.....	 5
 2.2 BÉZIER EĞRİLERİ İLE İLGİLİ KAVRAMLAR	 8
 2.2.1 Bézier Eğri Parçasının Oluşturulması	 10
 2.2.2 Bézier Eğrisinin Türevleri.....	 22
 2.3 MINKOWSKI UZAYI İLE İLGİLİ KAVRAMLAR	 25
 BÖLÜM 3 ÜÇ BOYUTLU MINKOWSKI UZAYINDA BÉZIER EĞRİLERİ.....	 31

İÇİNDEKİLER (devam ediyor)

Sayfa

3.1 TIMELIKE BÉZIER EĞRİLERİNİN SERRET-FRENET ÇATISI, EĞRİLİĞİ VE BURULMASI.....	31
3.2 SPACELIKE BÉZIER EĞRİLERİNİN SERRET-FRENET ÇATISI, EĞRİLİĞİ VE BURULMASI.....	41
3.2.1 Spacelike Normale Sahip Spacelike Bézier Eğrilerinin Serret-Frenet Çatısı, Eğriliği ve Burulması	41
BÖLÜM 4 E^3 ' de TIMELIKE VE SPACELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET FRENET ELEMANLARINA İLİŞKİN UYGULAMALAR	53
4.1 E^3 ' DE ANA NORMALİ TIMELIKE OLAN TIMELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET-FRENET ELEMANLARI UYGULAMALARI.....	53
4.2 E^3 ' DE ANA NORMALİ SPACELIKE OLAN SPACELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET-FRENET ELEMANLARI UYGULAMALARI.....	57
4.3 E^3 ' DE ANA NORMALİ TIMELIKE OLAN SPACELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET-FRENET ELEMANLARI UYGULAMALARI.....	60
BÖLÜM 5 EĞRİ UYDURMA	65
BÖLÜM 6 SONUÇ.....	73
KAYNAKLAR.....	75
EK AÇIKLAMALAR.....	83

İÇİNDEKİLER (devam ediyor)

	<u>Sayfa</u>
ÖZGEÇMİŞ	91





ŞEKİLLER DİZİNİ

<u>No</u>	<u>Sayfa</u>
Şekil 2.1 Birinci dereceden Bernstein polinomunun Python 3.10 görüntüsü.	11
Şekil 2.2 p_0, p_1 kontrol noktalarının oluşturduğu lineer Bézier eğrisi.	11
Şekil 2.3 $p_0 = (1,1), p_1 = (2,0)$ kontrol noktalarından geçen lineer Bézier eğrisinin Python 3.10 görüntüsü.	12
Şekil 2.4 p_0, p_1, p_2 kontrol noktalarının oluşturduğu kuadratik Bézier eğrisi.	12
Şekil 2.5 İkinci dereceden Bernstein polinomunun Python 3.10 görüntüsü.	13
Şekil 2.6 $p_0 = (0,0), p_1 = (2, -1), p_2 = (3,4)$ kontrol noktalarından geçen kuadratik Bézier eğrisinin Python 3.10 görüntüsü.	14
Şekil 2.7 Üçüncü dereceden Bernstein polinomunun Python 3.10 görüntüsü.	15
Şekil 2.8 p_0, p_1, p_2, p_3 kontrol noktalarının oluşturduğu kübik Bézier eğrisinin Python 3.10 görüntüsü.	15
Şekil 2.9 $p_0 = (0,0), p_1 = (1,0), p_2 = (2,3), p_3 = (-1,0)$ kontrol noktalarından geçen kübik Bézier eğrisinin Python 3.10 görüntüsü.	16
Şekil 2.10 $p_0 = (0,0,0), p_1 = (1,0,1), p_2 = (2,3,0), p_3 = (-1,1,1)$ kontrol noktalarından geçen düzlemsel kübik Bézier eğrisinin Python 3.10 görüntüsü.	17
Şekil 2.11 Minkowski düzleminde vektörler.	26
Şekil 5.1(a, b, c, d): dört adet kontrol noktası ile verilen kübik Bézier eğrisinin kontrol noktalarının değişmesiyle oluşan harflerin Python 3.10 görüntüsü.	66
Şekil 5.2 Şekil 5.1.a 'nın Python 3.10 kodu.	66
Şekil 5.3 Şekil 5.1.b 'nın Python 3.10 kodu.	67
Şekil 5.4 Şekil 5.1.c 'nın Python 3.10 kodu.	67
Şekil 5.5 Şekil 5.1.d 'nın Python 3.10 kodu.	68
Şekil 5.6 (a, b, c): $t = 0,1, t = 0,01, t = 0,001$ artış miktarına göre dört adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 görüntüsü.	68
Şekil 5.7 Şekil 5.6.a $t = 0,1$ artış miktarına göre dört adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 kodu.	69

ŞEKİLLER DİZİNİ (devam ediyor)

No

Sayfa

Şekil 5.8 Şekil 5.6.b $t = 0,01$ artış miktarına göre dört adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 kodu.	70
Şekil 5.9 Şekil 5.6.c $t = 0,001$ artış miktarına göre dört adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 kodu.....	71



EK AÇIKLAMALAR DİZİNİ

Sayfa

EK A Birinci dereceden Bernstein polinomunun Python 3.10 kodu ve grafiği.	83
EK B $p_0 = (1,1), p_1 = (2,0)$ kontrol noktalarından geçen lineer Bézier eğrisinin Python 3.10 görüntüsü	84
EK C İkinci dereceden Bernstein polinomunun Python 3.10 kodu ve grafiği	85
EK D $p_0 = (0,0), p_1 = (2, -1), p_2 = (3,4)$ kontrol noktalarından geçen kuadratik Bézier eğrisinin Python 3.10 kodu ve grafiği	86
EK E Üçüncü dereceden Bernstein polinomunun Python 3.10 kodu ve grafiği	87
EK F $p_0 = (0,0), p_1 = (1,0), p_2 = (2,3), p_3 = (-1,0)$ kontrol noktalarından geçen kübik Bézier eğrisinin Python 3.10 kodu ve grafiği.....	88
EK G $p_0 = (0,0,0), p_1 = (1,0,1), p_2 = (2,3,0), p_3 = (-1,1,1)$ kontrol noktalarından geçen kübik Bézier eğrisinin Python 3.10 kodu ve grafiği.....	89



SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

$\mathbb{R}$	: Reel sayı
$\langle , \rangle$	: İç çarpım
$\mathbb{E}$	: Öklid uzay
$\mathbb{E}^2$	: İki boyutlu Öklid uzayı
$\mathbb{E}^3$	: Üç boyutlu Öklid uzay
T	: Serret-Frenet çatısının teğet vektörü
B	: Serret-Frenet çatısının binormal vektörü
N	: Serret-Frenet çatısının ana normal vektörü
κ	: Eğrinin eğriliği
τ	: Eğrinin burulması
$\binom{n}{k}$	: n 'nin k 'lı kombinasyonu
p_i	: Bézier eğrisi üzerindeki kontrol noktası
$\mathfrak{B}$	: n . dereceden Bézier eğrisi
B_i^n	: n . dereceden Bernstein temel fonksiyonu
$\frac{dB_i^n}{dt}$	: n . dereceden Bernstein temel fonksiyonunun türevi
$\frac{d\mathfrak{B}}{dt}$	: n . dereceden Bézier eğrisinin türevi
$\mathfrak{B}^r$	: n . dereceden Bézier eğrisinin r - inci türevi
Δ	: İleri fark operatörü
$\wedge_{IL}$	: Minkowski uzayında vektörel çarpım
R_1^3	: Üç boyutlu Minkowski uzayı
$\ x\ _{IL}$	: Minkowski uzayında vektörel norm

SİMGELER VE KISALTMALAR DİZİNİ (devam ediyor)

KISALTMALAR

AI	: Artificial Intelligence
BDT	: Bilgisayar Destekli Tasarım
BDGT	: Bilgisayar Destekli Geometrik Tasarım
CAD	: Computer Aided Design
CAGD	: Computer Aided Geometric Design
İHA	: İnsansız Hava Aracı
ML	: Machine Learning
MÖ	: Makine Öğrenmesi
YZ	: Yapay Zekâ
JOGL	: Java Open Graphics Library

BÖLÜM 1

GİRİŞ

21. yüzyıla yaklaşırken internet kullanımının artışı ve teknolojinin gelişmesiyle birlikte “bilişim çağı” denilen bir zaman dilimine geçilmiştir. Zamanla insanların ihtiyaçlarına göre şekillenen bu teknolojik gelişmeler, insanlardan edinilen her türlü bilgi ile onların günlük yaşamını kolaylaştırmak için yapay zekâ sistemleri oluşturulmaya başlanmıştır. Yapay zekâ sistemlerinin oluşturulabilmesi için verilere ihtiyaç duyulmaktadır. Bu verilerin en önemli kaynağı da insanlardır. Yapay zekâ kavramı mühendislik, tıp, biyoloji, psikoloji, sosyoloji, hukuk, eğitim-öğretim, ticaret, animasyon gibi farklı alanlarda kullanılabilir. Kullanım alanlarına göre yapay zekâ bazen felsefi bazen de teknik olarak ifade edilebilir. Teknik olarak yapay zekânın en yaygın tanımlarından biri insanların ‘zekâ’ olarak nitelendirdiği öğrenme, karar verme, problem çözme, yorumlama gibi faaliyetlerin bir makine tarafından gerçekleştirilebilmesi olarak kabul edilmektedir. Birçok farklı alanda kullanılabilmesi ve 1950’li yıllardan sonra internet kullanımının artış göstermesiyle birlikte hayatımıza hızla giriş yapan makine öğrenmesi kavramı her geçen gün daha da gelişerek güncelliğini ve popüleritesini korumaktadır. Geçmişte makinelere öğretilen algoritmalar sayesinde “Makinelerin bilgiyi işleyebilir, hesaplayabilir ve sonuçlandırabilir yapılarda olduğundan ve aynı zamanda tüm bunları yüksek hızda yaptığından makinelerin düşünebileceğini belirten” ifadelerde bulunulmuş ve tüm bunlardan dolayı bu büyük makinelerin “Beyne benzer karmaşık yapılardan oluştuklarından” bahsedilmiştir [1-5].

Makine öğrenmesi, insanların hayatını kolaylaştırmak için çeşitli algoritma, modelleme ve teknikleri bilgisayar sistemlerine işleyerek makinelerin insanlar gibi öğrenme, yorumlama ve sonuçlandırma yeteneklerinin olması için kullanılan, yapay zekânın alt dalı olan bilimsel çalışma alanıdır. Bu çalışma alanı başlangıçta imkânsız gibi düşünülse de yapay zekâ, matematik ve istatistik alanlarının kesişmesi ile mümkün hale gelmiştir. Örneğin, üretim alanında (tedarikten lojistiğe kadar birçok farklı alanda), sağlık alanında (kanser tümörü algılamadan, göz hastalıkları tespitine kadar her türlü hastalıkta), finansal hizmetlerde (borsa hareketleri, dolandırıcılık şüphesi taşıyan kişilerin tespiti vs.), medya ve eğlence sektöründe, e-ticarette ve hukuk gibi birçok farklı alanda uygulama alanı vardır [1-15].

Toplumsal/insani ihtiyaları karřılayan ve yapay zekâ literatürünü önemli ölçüde zenginleřtiren seilmiş bazı alıřmalar ařağıdaki gibi özetlenmiřtir. “Diagnosis of Diseases through Eye Images using Artificial Intelligence,” adlı alıřmada göz görüntüleri yoluyla hastalıkların tespiti ve teřhisinde yeni yapay zekâ algoritmaları geliřtirmiş ve test etmiřtir. Ayrıca göz hastalıklarının teřhisinde de yapay zekâ kullanımı önerilmektedir [16]. “Ortodonti Kliniğine Bařvuran Hastaların Okluzal (alt ve üst diřler temas halindeyken diřler arasındaki iliřki) Fotoğraflarından Yapay Zekâ Kullanılarak aprařıklık řiddetinin Belirlenmesi,” adlı uzmanlık alıřmasında yapay zekânın diř hekimliğı dalında kullanım alanını geniřletmek, geliřtirilen yapay zekâ modeli ile doėru teřhis ve tedavi sürecine katkıda bulunmak ve bu sürecin hızlanması konusunda ilerleme kaydedilmesi amalanmıřtır [17, 18]. “Tarihi Yapılarda Yapay Zekâ Destekli Hasar Tespiti İçin Bir Model Önerisi,” adlı alıřmada tarihi yapılardaki bozulmaların oldukça büyük kaynak ve zaman gerektiren geleneksel yöntemler ile oluřturulan bozulma haritaları yerine yapay zekâ destekli geliřtirilen hasar tespit modeli ile daha hızlı ve doėru tespitin yapılabil-diğı görülmüřtür [19]. “İnsan Kaynakları Yönetiminde Yapay Zekâ ve Türkiye Uygulaması,” adlı alıřma sayesinde insan kaynakları sürecinde iře alım sürecinde aday bulma ve seme için ayrılan süreç büyük oranda azalmıřtır. Ayrıca, performans deėerlendirme sürecinde de daha objektif ve yeterli performansı sergileyemeyen alıřanlara performanslarını iyileřtirmeleri için seenekler önerilmektedir [20]. “Novel Models for Credit Card Fraud Detection,” adlı alıřmada kredi kartı sahtekarlıklarının tespiti için eřitli yapay zekâ algoritmaları önerilmiřtir [21]. “Avukatlık Mesleğinde Yapay Zekâ Kullanımı,” adlı alıřmada gün getike artan teknolojiyle beraber mesleėe dahil olan robot avukatların meslekteki yerine, faydasına ve zararlarına deėinilmiřtir [22].

Bahsi geen alıřmalar yapay zekanın ok farklı alanlarda kullanıldığını ve ne kadar önemli bir alıřma sahası olduėunu göstermektedir.

Bu alanların bazı örneklerinde teorik bilgiyi bilgisayar sistemlerine öğretebilmek için bazı temel parametrik eğrilere ve yüzeylere ihtiyaç duyulmaktadır. Eğriler ve yüzeyler, bilgisayar destekli geometrik tasarımın temelini oluřturmaktadır. Bu yüzden makine öğrenmesinden bahsedilince ilk akla gelen kavramlardan biri eğrilerdir. Eğriler sanat, endüstri, mimari ve otomotiv gibi birok alanda kullanım alanı bulmuřtur. Özellikle temel bir örnek vermek gerekirse elektronik ortamlarda kullanılan eřitli yazı tiplerinin oluřmasında eğriler kullanılır. Ölekleme yapılarak farklı boyutlarda, öteleme, dönüřtürme ve derinleřtirme gibi farklı uygulamalar yapılarak özel yazı tipleri elde edilebilir. Bilgisayar grafiklerinin temelinde eğriler olduėundan dolayı olarak bu uygulamaların biroėunun temeli matematik ile atılmıřtır. Bunlar

arasında Bilgisayar Destekli Tasarım (BDT), geometrik modelleme, robotik, photoshop, sinema efektleri gibi çok çeşitli uygulama alanı vardır [23-30]. Bu tezde, eğrilerin ve yüzeylerin bilgisayar sistemlerinde çizdirilebilmesinde yaygın olarak kullanılan Bézier eğrileri ve bazı ilginç Python 3.10 (64 bit) uygulamaları çalışılmıştır. Bézier eğrileri başlangıç ve bitiş noktaları olan, sonlu sayıda kontrol noktalarından oluşur. Dolayısıyla bu kontrol noktaları ve parametrenin değer aralığı değiştirilerek çeşitli görüntülere ulaşılabilir. Böylece Bézier eğrilerinin tıptan mimariye, mimariden otomotiv sanayisine, otomotiv sanayisinden çeşitli mühendislik alanlarına kadar birçok kullanım alanı vardır [23-57].

Bu kullanım alanları ile ilgili bazı çalışmalar aşağıdaki gibi özetlenmiştir ve bu çalışmalardan bazıları yapay zekâ alanını da kapsamaktadır.

“Kısıtlara Bağlı Matematiksel Modelleme ile İnsansız Hava Aracı İçin Yumuşatılmış Rota Planlaması” adlı çalışmada kısıtlara bağlı matematiksel modelleme ile İnsansız Hava Araçları (İHA) için yumuşatılmış rota planlama problemi ele alınmıştır. Yumuşatma yöntemleri olarak Bézier Eğrileri, B-Spline Eğrileri ve Dubins Yolu kullanımı iki örnek problem üzerinde incelenmiş ve elde edilen sonuçlar karşılaştırılmıştır [15]. “Kübik Bézier Eğrileri ile Yüz İfadesi Tanıma” adlı çalışmada gülen, şaşkın, korkmuş, kızgın, iğrenmiş ve doğal olmak üzere yedi farklı yüz ifadesi tespitinde kübik Bézier eğrileri kullanılarak resimlerden yüz ifadesi tanıma işlemi gerçekleştirilmiştir. Resimlerdeki yüz görsellerinden duygusal ifadelerin tanınabilmesi için eğri uydurma yöntemine bağlı olarak yeni bir algoritma geliştirilmiştir [58]. “ $\mathbb{R}^3$ ’ deki Yüzey Eğrilerinin Bézier Eğrileri ve Matlab Uygulamaları” adlı çalışmada yüzey üzerinde bir eğri ve bu eğrinin ikiden fazla noktası alınarak buna ait Bézier eğrisi hesaplanmıştır. $\mathbb{R}^3$ ’ deki yüzey eğrilerinin Bézier eğrileri incelenip MATLAB uygulamaları yapılmıştır [59]. “Öklid Düzleminde ve Öklid Uzayında Bézier Eğrileri” adlı çalışmada ise bilgisayar bilimleri ve uygulamalı matematikte kullanılan temel bir algoritma yardımıyla iki boyutlu ve üç boyutlu Öklid düzleminde ve Öklid uzayında Bézier eğrilerinin rolü incelenmiştir [60]. “On the Cubic Bézier Curves in $\mathbb{E}^3$ ” adlı çalışmada $\mathbb{E}^3$ ’ te kübik Bézier eğrileri matris formunda kontrol noktalarına bağlı olarak incelenmiştir [61]. “Otonom Araçlarda Hareket Planlaması” başlıklı çalışmada otonom araçlarda mevcut yol ve trafik durumunda çalışma döngüsündeki hareket planlama safhası üzerine yapılan çalışmalar ve teknikler incelenmiş, kıyaslanmış ve geleceğe dair bilgiler verilmiştir [62]. “Interactive 3D Visualization of Bézier Curves using Java Open Graphics Library (JOGL)” de 3D Bézier eğrilerinin etkileşimli görselleştirilmesi ve işlenmesi için ve kübik Bézier eğrileri aracılığıyla grafik tasarımda harf oluşturma sürecinin görsel gösterimi için Java uygulamaları ve JOGL gibi en güncel görselleştirme teknikleri kullanılarak

yeni bir program aracı sunulmuştur [63]. “A Path Generation For Automed Vehicle Based On Bézier Curve And Via-Points” isimli makalede Bézier eğrisine dayalı olarak otomatik araçlarda navigasyon sistemi kullanımında düzgün yolun türetilmesi için bir yol oluşturma yöntemi sunulmaktadır [64].

Bahsi geçen çalışmalar Bézier eğrilerinin çok farklı alanlarda kullanıldığını ve bilgisayar destekli tasarımda ve bununla birlikte yapay zekâ alanında ne kadar önemli bir çalışma konusu olduğunu göstermektedir.

Bu tez beş bölümden oluşmaktadır: Tezin birinci bölümünde; yapay zekâ, makine öğrenmesi ve Bézier eğrileri hakkında kısaca bilgiler ve literatürde bulunan alanlardan çeşitli örnekler verilmiştir. İkinci bölüm; tezin kalan kısmında kullanılacak Öklid uzayı, Minkowski uzayı ve Bézier eğrileri ile ilgili temel tanım ve teoremlere yer verilmiştir. Ayrıca temel Bézier eğrilerinin ve bazı örneklerinin Python 3.10 (64 bit) programlama dilinde grafikleri çizdirilerek kodları ek olarak tezin sonunda verilmiştir. Üçüncü ve dördüncü bölümde ise üç boyutlu Minkowski uzayında Bézier eğrilerinin Serret-Frenet çatılarıyla ilgili ispatlar yapıp bunlarla ilgili örneklere yer verilmiştir. Beşinci bölümde ise dört farklı kontrol noktasına sahip Bezier eğrilerinin kontrol noktaları değiştirilerek elde edilen görüntü ve verilen parametre aralığındaki artış miktarı değiştirilerek oluşan görüntüdeki farklılıklar Python 3.10 (64 bit) programlama dilinde çizdirilerek incelenmiştir.

BÖLÜM 2

TEMEL KAVRAMLAR

Bu bölüm üç alt bölümden oluşmaktadır. Birinci alt bölümde Öklid uzayı; ikinci alt bölümde Bézier eğrileri ve son alt bölümde de Minkowski uzayı hakkında tez boyunca kullanılacak olan temel kavramlara yer verilmiştir.

2.1 ÖKLİD UZAYINDA EĞRİLER

Bu alt bölümde, tezde gerekli olan bazı önemli tanımlara yer verilmiştir.

Tanım 2.1 $I \subset \mathbb{R}$ bir açık aralık olacak şekilde (I, α) koordinat komşuluğu verilsin.

$$\alpha : I \rightarrow \mathbb{R}^n$$

için $u \in I$ tanımlı olmak üzere $\alpha(u) = (\alpha_1(u), \alpha_2(u), \dots, \alpha_n(u))$ şeklinde tanımlı olan $\alpha(u)$ 'ya $\mathbb{R}^n$ 'de bir *eğri* denir. Burada I aralığına α eğrisinin parametre aralığı, $u \in I$ ya α eğrisinin parametresi denir [66, 67].

Tanım 2.2 $\mathbb{R}^n$ de ve (I, α) koordinat komşuluğunda bir M eğrisi verilsin.

$$\alpha : I \rightarrow \mathbb{R}^n$$

olmak üzere $u \in I$ için $\alpha(u) = (\alpha_1(u), \alpha_2(u), \dots, \alpha_n(u))$ ifadesinin türevi olan $\alpha'(u) = (\alpha'_1(u), \alpha'_2(u), \dots, \alpha'_n(u))$ ile tanımlı olan $\alpha'(u)$ vektörüne M eğrisinin *hız vektörü* denir [66, 67].

Tanım 2.3 $\mathbb{R}^n$ de (I, α) koordinat komşuluğunda bir M eğrisi verilsin. $|\alpha'| : I \rightarrow \mathbb{R}$ fonksiyonu tanımlansın. $u \in I$ için

$$|\alpha'| (u) = |\alpha'(u)|$$

eşitliği sağlanıyorsa bu fonksiyona (I, α) koordinat komşuluğuna göre M eğrisinin skaler hız fonksiyonu, $|\alpha'(u)| \in \mathbb{R}$ ifadesine de *skaler hız* denir [66, 67].

Tanım 2.4 $\mathbb{R}^n$ de (I, α) koordinat komşuluğunda bir M eğrisi verilsin. $u \in I$ için,

$$|\alpha'(u)| = 1 \quad (2.1)$$

oluyorsa (I, α) koordinat komşuluğunda M eğrisine *birim hızlı eğri* denir [66, 67].

Tanım 2.5 Her noktasındaki hız vektörü sıfırdan farklı olan eğriye regüler eğri denir. Yani,

$$\alpha: I \rightarrow \mathbb{R}^n$$

olmak üzere her $u \in I$ için $\alpha'(u) \neq 0$ ise α eğrisi *regüler eğridir* [66, 67].

Tanım 2.6 A bir afin reel uzay olsun. Afin reel uzayların birleşmesiyle tanımlanan vektör uzayı da V olsun. V uzayında iç çarpım

$$\langle \cdot, \cdot \rangle: V \times V \rightarrow \mathbb{R}$$

$$(x, y) \rightarrow \langle x, y \rangle = \sum_{i=1}^n x_i y_i \quad \{x = (x_1, x_2, \dots, x_n), y = (y_1, y_2, \dots, y_n)\} \quad (2.2)$$

şeklinde tanımlanırsa A afin uzayına n boyutlu *Öklid uzayı* denir. Bu n boyutlu Öklid uzayı $\mathbb{E}^n$ ile gösterilir [66, 67].

Sonuç 2.1 İki boyutlu ve üç boyutlu Öklid uzayları sırasıyla $\mathbb{E}^2$ ve $\mathbb{E}^3$ ile gösterilir. Aynı zamanda sırasıyla $\mathbb{E}^2$ ve $\mathbb{E}^3$ 'de iç çarpımları

$$\langle x, y \rangle = \langle (x_1, x_2), (y_1, y_2) \rangle = x_1 y_1 + x_2 y_2 \quad (2.3)$$

$$\langle x, y \rangle = \langle (x_1, x_2, x_3), (y_1, y_2, y_3) \rangle = x_1 y_1 + x_2 y_2 + x_3 y_3 \quad (2.4)$$

olarak tanımlanır.

Tanım 2.7 p ve q , $\mathbb{E}^2$ 'de sıfırdan farklı iki vektör olsun.

$$\cos \theta = \frac{\langle p, q \rangle}{\|p\| \|q\|} \quad \text{ve} \quad \sin \theta = \frac{\langle p, Jq \rangle}{\|p\| \|q\|} \quad (2.5)$$

şeklinde tanımlanır. Burada θ ; p ve q arasında $0 \leq \theta \leq 2\pi$ olacak şekilde benzersiz bir yönelim açısıdır [68].

Tanım 2.8 $\mathbb{R}^2$ de bir eğri $\alpha(t) = (\alpha_1(t), \alpha_2(t))$ şeklinde verilsin. t yay parametresi olarak tanımlanacak olursa bu eğrinin Frenet çatısı

$$T = (\alpha'_1(t), \alpha'_2(t)) \quad (2.6)$$

$$N = (-\alpha'_2(t), \alpha'_1(t)) \quad (2.7)$$

ile tanımlanır [66, 68].

Tanım 2.9 $r(t)$ parametrik düzlemsel eğri olsun. Böylece $r(t)$ eğrisinin eğriliği

$$\kappa = \frac{r''(t) \cdot J r''(t)}{\|r'(t)\|^3} \quad (2.8)$$

olarak tanımlanır [69].

Tanım 2.10 $\alpha : I \rightarrow E^3$ birim olmayan bir hız eğrisi olsun. α 'nın Serret-Frenet çerçevesi $\{T, N, B\}$, eğriliği κ ve burulması τ ;

$$T(t) = \frac{\alpha'(t)}{\|\alpha'(t)\|} \quad (2.9)$$

$$N(t) = B(t) \times T(t) \quad (2.10)$$

$$B(t) = \frac{\alpha'(t) \times \alpha''(t)}{\|\alpha'(t) \times \alpha''(t)\|} \quad (2.11)$$

$$\kappa(t) = \frac{\|\alpha'(t) \times \alpha''(t)\|}{\|\alpha'(t)\|^3} \quad (2.12)$$

$$\tau(t) = \frac{\det(\alpha'(t), \alpha''(t), \alpha'''(t))}{\|\alpha'(t) \times \alpha''(t)\|^2} \quad (2.13)$$

denklemleri ile tanımlanır [69].

2.2. BÉZIER EĞRİLERİ İLE İLGİLİ KAVRAMLAR

Bu bölümde, öncelikle Bézier eğrileri hakkında genel bilgilere daha sonra da Bézier eğrileri ve Bernstein polinomları ile ilgili tezde kullanılacak temel kavramlara yer verilmiştir.

Bézier eğrileri 1950-1960 yılları arasında Renault araba firmasında mühendis olarak çalışan Fransız matematikçi Pierre Bézier tarafından keşfedilmiştir ve onun adıyla anılmaya başlanmıştır. Fakat Pierre Bézier keşfetmeden önce bu eğrilerin temeli 1912 yılında keşfedilen Bernstein polinomlarına kadar uzanmaktadır. Bundan yaklaşık 50 yıl sonra Citroën araba firmasında çalışmakta olan Paul de Casteljau tarafından “De Casteljau” adında Bézier eğrileri ile aynı sonuca ulaşan bir algoritma geliştirilmiştir. Fakat bu algoritmanın patentini almış olsa bile hiçbir yerde yayınlamamıştır. Daha sonra yaklaşık aynı yıllarda birbirlerinden habersiz şekilde P. Bézier’de bu eğrilerle ilgili çalışmalarını yapmıştır. Otomotiv firmasında çalışan Pierre Bézier, otomotiv gövdelerinin tasarlanması ile ilgili çalışırken bu konuyu bilgisayar grafiklerine taşımak amacıyla, 1912 yılında yaklaşımlar teorisi üzerine yaptığı araştırmada Rus Matematikçi Sergei Natanovich Bernstein tarafından geliştirilen ve kendi adıyla anılan Bernstein polinomlarını kullanarak bu eğriler yardımıyla yeni bir geometrik modelleme elde etmiştir. Böylece Bernstein polinomu yardımıyla eğri ve yüzeylerin Bernstein formunda ifadesiyle tamamen gelişen bu yeni teori Pierre Bézier’in adıyla anılmaya başlanmıştır. Daha sonra bilgisayar grafiklerinde sıklıkla tercih edilmiştir. Başlangıç ve bitiş noktalarına sahip olan n . dereceden Bézier eğrilerinin $n + 1$ tane kontrol noktası vardır. Başlangıç ve bitiş noktalarının arasında da sonlu sayıda kontrol noktası olmalıdır. Kontrol noktalarının sırasıyla birleşmesiyle oluşan görüntüye kontrol çokgeni denir. Bézier eğrilerinin kontrol noktaları çokgeninin başlangıç ve bitiş (uç noktalarından) noktalarından geçmek zorundadır. Fakat diğer noktalar eğri üzerinde bulunmak zorunda değildir. Bu sebeple Bézier eğrileri kontrol noktaları çokgenine sadece uç noktalarda bağlıdır. Bu kontrol noktaları istenen eğriye göre değiştirilerek istenen şekle ulaşılabilir. Böylece Bézier eğrilerinin kontrol noktalarına bağlı ve derecesiyle kontrol noktaları arasında doğrudan bir ilişki olduğu anlaşılır. Bu nedenle Bézier eğrilerinin şekli aslında kontrol noktalarının dolayısıyla kontrol çokgeninin değişimiyle doğrudan ilgilidir. Bilgisayar Destekli Tasarım (BDT) programları ve bilgisayar grafikleri çizimlerinde kontrol

noktaları değiştirilerek uygun eğri modelleri çıkarılabilir. Bu da Bézier eğrilerinin birçok alanda kullanılmasını sağlamaktadır [23-27, 31-64, 70-72]

Şimdi Bézier eğrileri ile ilgili temel kavramlar verilecektir.

Tanım 2.11 $A_1, A_2, \dots, A_k$ ($k \geq 3$) düzlemdeki sıralı noktalar olsun. Bu noktaları ardışık olarak birbirine ve sonuncusunu da birincisine doğru parçalarıyla bağlayarak elde edilen geometrik şekle düzlemde bir k -kenarlı *poligon* denir [65, 66].

Tanım 2.12 $x, y \in \mathbb{R}$ ve $n \in \mathbb{N}$ için $(x + y)^n$ binom açılımı,

$$(x + y)^n = \sum_{k=0}^n \binom{n}{k} x^k y^{(n-k)} \quad (2.14)$$

olarak tanımlanır. Buradaki binom açılımı İslamın Altın Çağı'nın önemli matematikçilerinden biri olan Ömer Hayyam (12. ve 13. y.y.) döneminden beri bilinmektedir ve katsayıları $\binom{n}{k}$ olan Pascal üçgeni olarak bilinen algoritma ile verilebilir. x ve y nin olasılık değişkenleri olması durumunda; Bernstein bu açılıma, istatistiksel bir tanım kazandırmıştır. Bu olaya ait olasılıklar x ve y ise;

$$x + y = 1$$

şeklindedir. Burada, $y = 1 - x$ olarak tanımlanırsa, $x + y = x + (1 - x)$ toplamı olarak yazılabilir. Böylece bu olaya ait tek değişkenli olasılık açılımı,

$$1 = (x + (1 - x))^n \quad (2.15)$$

$$= \sum_{k=0}^n \binom{n}{k} x^k (1 - x)^{n-k} \quad (2.16)$$

olarak tanımlanır. Eş. (2.16) açılımı Bernstein temel fonksiyonu ve bu fonksiyona eşlik eden her bir

$$\binom{n}{k} x^k (1 - x)^{n-k} \quad (2.17)$$

terimi Bernstein temel fonksiyonunun terimleri olarak isimlendirilir. Bernstein polinomları, polinomlar halkasında önemli bir role sahiptir ve Bézier eğrilerinin temelini oluşturur [59].

Tanım 2.13 p_i ($i = 0, 1, \dots, n$) 'ler $(n + 1)$ tane kontrol noktası olmak üzere,

$$\mathfrak{B}^n(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i p_i = \sum_{i=0}^n B_i^n(t) p_i, \quad 0 \leq t \leq 1 \quad (2.18)$$

şeklinde tanımlı eğriye *Bézier eğrisi* denir. Burada

$$B_i^n(t) = \begin{cases} \binom{n}{i} (1-t)^{n-i} t^i, & 0 \leq i \leq n \\ 0, & \text{diğer durumlarda} \end{cases} \quad (2.19)$$

ifadesine *Bernstein polinomu* denir. Burada $\binom{n}{i} = \frac{n!}{(n-i)!i!}$ olarak tanımlı binom katsayısıdır [27, 30, 38, 67].

2.2.1 Bézier Eğri Parçasının Oluşturulması

Tanım 2.14 İki ve üç boyutlu $\{E^n (n = 2, 3)\}$ Öklid uzayında p_0, p_1 kontrol noktaları verildiğinde $\overline{p_0 p_1}$ doğru parçası üzerindeki herhangi bir nokta $\mathfrak{B}_0^1(t)$ ile gösterilirse;

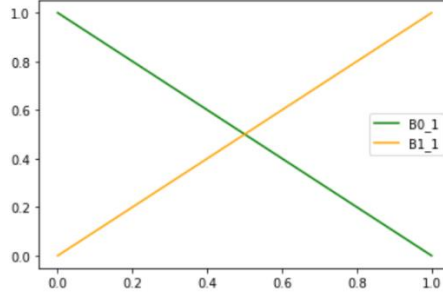
$$\mathfrak{B}_0^1(t) = B_0^1(t) p_0 + B_1^1(t) p_1 = (1-t) p_0 + t p_1 \quad ; \quad t \in [0, 1]$$

denklemini bir lineer eğri parçasının bir parametrizasyonunu oluşturur. Burada;

$$B_0^1(t) = \binom{1}{0} t^0 (1-t)^{1-0} = (1-t) \quad (2.20)$$

$$B_1^1(t) = \binom{1}{1} t^1 (1-t)^{1-1} = t \quad (2.21)$$

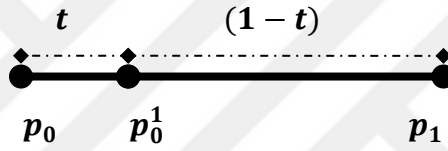
eşitlikleri birinci dereceden Bernstein temel fonksiyonu ve p_0, p_1 kontrol noktaları olarak adlandırılır [27, 28, 38, 59]. Birinci dereceden Bernstein temel fonksiyonunun $0 \leq t \leq 1$ için grafiği aşağıdaki gibi çizilebilir (bkz. Şekil 2.1).



Şekil 2.1 Birinci dereceden Bernstein temel fonksiyonunun Python 3.10 görüntüsü [59].

Yukarıdaki Şekil 2.1'in Python 3.10 kodu EK A' da verilmiştir [38, 73, 74].

Bernstein temel fonksiyonu ve kontrol noktalarının birleşmesiyle oluşan denklem lineer Bézier eğrisini ifade etmektedir (bkz. Şekil 2.2).



Şekil 2.2 p_0, p_1 kontrol noktalarının oluşturduğu lineer Bézier eğrisi [59].

Örnek 2.1 $\mathbb{E}^2$ de $p_0 = (1,1)$, $p_1 = (2,0)$ kontrol noktalarından geçen $\overline{p_0 p_1}$ doğru parçası;

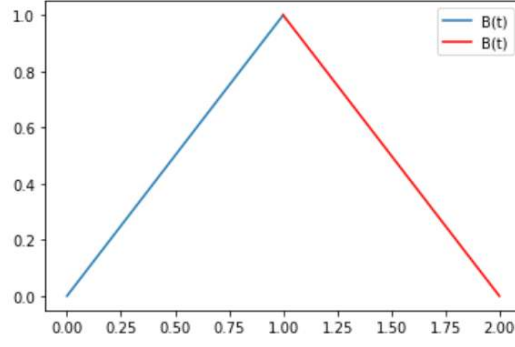
$$\mathfrak{B}^1(t) = \sum_{i=0}^1 B_i^1(t) p_i \quad ; \quad t \in [0,1] \quad (2.22)$$

$$= B_0^1(t) p_0 + B_1^1(t) p_1$$

$$= (1-t)(1,1) + t(2,0)$$

$$= (1+t, 1-t)$$

dir (bkz. Şekil 2.3).



Şekil 2.3 $p_0 = (1,1)$, $p_1 = (2,0)$ kontrol noktalarından geçen lineer Bézier eğrisi Python 3.10 görüntüsü [59].

Şekil 2.3' in Python 3.10 kodu EK B' de verilmiştir [38, 73, 74].

Tanım 2.15 $\mathbb{E}^3$ Öklid uzayında verilen p_0, p_1 ve p_2 noktaları için $t \in [0,1]$ olmak üzere

$$P_0^1(t) = (1-t)p_0 + tp_1$$

$$P_1^1(t) = (1-t)p_1 + tp_2$$

$$P_0^2(t) = (1-t)P_0^1(t) + tP_1^1(t)$$

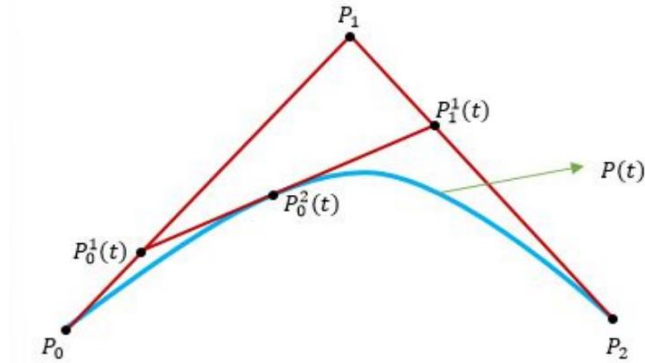
denklemleri yardımıyla

$$\mathfrak{B}(t) = P_0^2(t) = \sum_{i=0}^2 B_i^2(t)p_i ; \quad t \in [0,1] \quad (2.23)$$

$$= B_0^2(t)p_0 + B_1^2(t)p_1 + B_2^2(t)p_2$$

$$= (1-t)^2p_0 + 2(1-t)tp_1 + t^2p_2 \quad (2.24)$$

$\mathfrak{B}(t)$ eğrisine *kuadratik Bézier eğrisi* denir ve kontrol noktalarının oluşturduğu doğru parçaları ile elde edilen üç boyutlu şekle *kontrol poligonu* denir (bkz. Şekil 2.3) [27, 28, 59].



Şekil 2.4 p_0, p_1, p_2 kontrol noktalarının oluşturduğu kuadratik Bézier eğrisi [59].

Şekil 2.4' den görüldüğü gibi; $\overline{p_0 p_1}$ doğru parçası üzerinde herhangi bir nokta $p_0^1(t)$, $\overline{p_1 p_2}$ doğru parçası üzerinde herhangi bir nokta $p_1^1(t)$, $\overline{p_0^1(t) p_1^1(t)}$ doğru parçası üzerinde herhangi bir nokta $p_0^2(t)$ ile gösterilmiştir.

(2.15) eşitliğinden ikinci dereceden Bernstein temel fonksiyonunun terimleri,

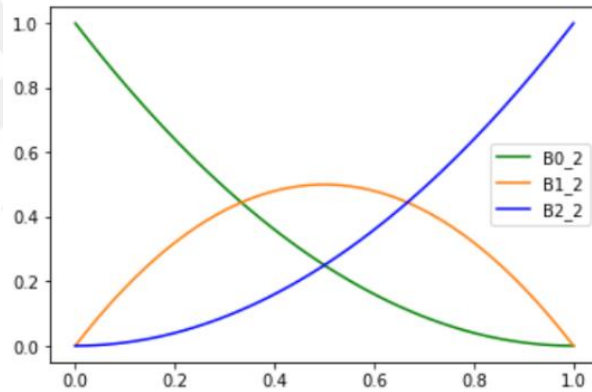
$$B_0^2(t) = \binom{2}{0} t^0 (1-t)^{2-0} = (1-t)^2,$$

$$B_1^2(t) = \binom{2}{1} t^1 (1-t)^{2-1} = 2(1-t)t,$$

$$B_2^2(t) = \binom{2}{2} t^2 (1-t)^{2-2} = t^2$$

şeklinde yazılır. Bu fonksiyonlar ikinci dereceden Bernstein temel fonksiyonu olarak adlandırılır [27, 28, 59].

İkinci dereceden Bernstein temel fonksiyonunun $0 \leq t \leq 1$ için grafiği aşağıdaki gibi çizilebilir.



Şekil 2.5 İkinci dereceden Bernstein temel fonksiyonunun Python 3.10 görüntüsü [59].

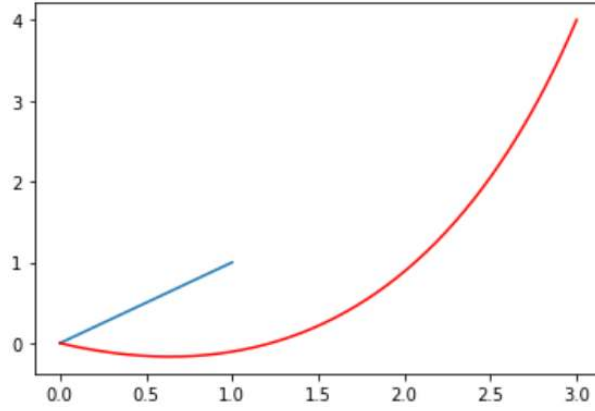
Şekil 2.5'in Python 3.10 kodu EK C' de verilmiştir [38, 73, 74].

Örnek 2.2 $p_0 = (0,0), p_1 = (2,-1), p_2 = (3,4)$ kontrol noktalarının oluşturduğu $\mathfrak{B}(t)$ düzlemsel kuadratik Bézier eğrisinin parametrik denklemi (2.4) eşitliği yardımıyla

$$\begin{aligned} \mathfrak{B}(t) &= \sum_{i=0}^2 B_i^2(t) p_i = (1-t)^2 p_0 + 2(1-t)t p_1 + t^2 p_2 \\ &= (1-t)^2 (0,0) + 2(1-t)t (2,-1) + t^2 (3,4) \\ &= (0,0) + (4t(1-t), -2(1-t)t) + (3t^2, 4t^2) \\ &= (0 + 4(t-t^2) + 3t^2, -2t + 2t^2 + 4t^2) \end{aligned}$$

$$= (4t - t^2, 6t^2 - 2t); \quad t \in [0,1]$$

dir (bkz. Şekil 2.6).



Şekil 2.6 $p_0 = (0,0), p_1 = (2,-1), p_2 = (3,4)$ kontrol noktalarının oluşturduğu düzlemsel kuadratik Bézier eğrisinin Python 3.10 görüntüsü [59].

Şekil 2.6'nın Python 3.10 kodu EK D' de verilmiştir [38, 73, 74].

Tanım 2.16 $\mathbb{E}^n$ ($n = 2,3$) Öklid uzayında p_0, p_1, p_2 ve p_3 noktaları verildiğinde $t \in [0,1]$ olmak üzere

$$P_0^1(t) = (1-t)p_0 + tp_1$$

$$P_1^1(t) = (1-t)p_1 + tp_2$$

$$P_2^1(t) = (1-t)p_2 + tp_3$$

$$P_0^2(t) = (1-t)p_0^1(t) + tp_1^1(t)$$

$$P_1^2(t) = (1-t)p_1^1(t) + tp_2^1(t)$$

$$P_0^3(t) = (1-t)p_0^2(t) + tp_1^2(t)$$

dır. Yukarıdaki denklemler yardımıyla

$$\mathfrak{B}(t) = P_0^3(t) = \sum_{i=0}^3 B_i^3(t)p_i \quad ; \quad t \in [0,1] \quad (2.25)$$

$$\begin{aligned} &= B_0^3(t)p_0 + B_1^3(t)p_1 + B_2^3(t)p_2 + B_3^3(t)p_3 \\ &= (1-t)^3p_0 + 3(1-t)^2tp_1 + 3(1-t)t^2p_2 + t^3p_3 \end{aligned} \quad (2.26)$$

kübik Bézier eğrisi tanımlanır. Burada;

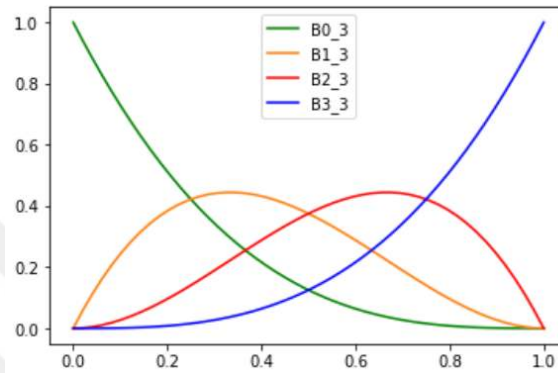
$$B_0^3(t) = \binom{3}{0} t^0 (1-t)^{3-0} = (1-t)^3$$

$$B_1^3(t) = \binom{3}{1}t^1(1-t)^{3-1} = 3(1-t)^2t$$

$$B_2^3(t) = \binom{3}{2}t^2(1-t)^{3-2} = 3(1-t)t^2$$

$$B_3^3(t) = \binom{3}{3}t^3(1-t)^{3-3} = t^3$$

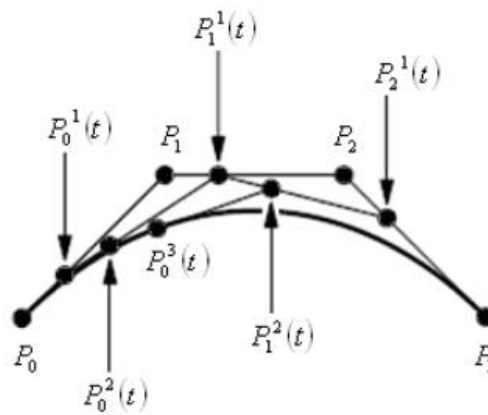
ifadeleri üçüncü dereceden Bernstein temel fonksiyonunu ifade eder [27, 28, 59]. Üçüncü dereceden Bernstein temel fonksiyonunun $0 \leq t \leq 1$ için görüntüsü aşağıdaki gibi çizdirilebilir (bkz. Şekil 2.7).



Şekil 2.7 Üçüncü dereceden Bernstein temel fonksiyonunun Python 3.10 görüntüsü [59].

Şekil 2.7'nin Python 3.10 kodu EK E' de verilmiştir [38, 73, 74].

$B_0^3, B_1^3, B_2^3, B_3^3$ Bernstein temel fonksiyonunun terimleri ve p_0, p_1, p_2, p_3 de kontrol noktalarının birleşmesiyle kübik Bézier eğrisi oluşmaktadır (bkz. Şekil 2.8).



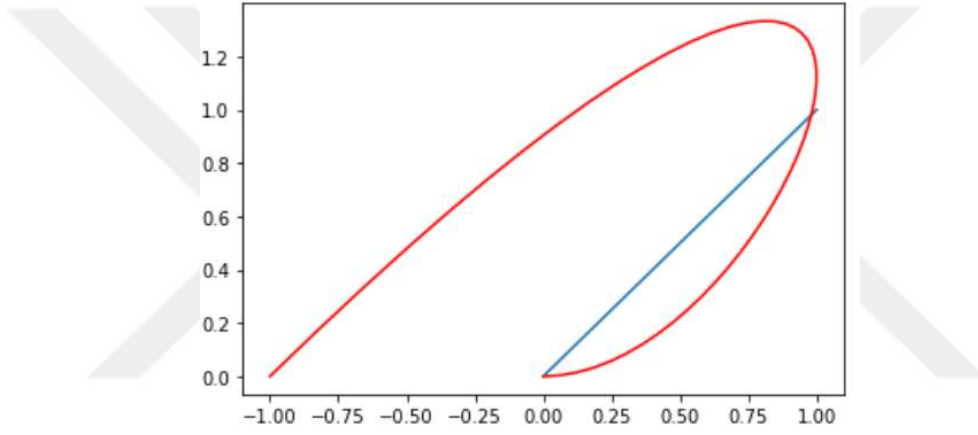
Şekil 2.8 p_0, p_1, p_2 ve p_3 kontrol noktalarının oluşturduğu kübik Bézier eğrisi [59].

Şekil 2.8'de görüldüğü üzere p_0, p_1, p_2 ve p_3 kontrol noktalarının doğru parçaları ile birleştirilmesi ile elde edilen poligona *kontrol çokgeni (poligonu)* denir [27, 28, 59].

Örnek 2.3 $\mathbb{E}^2$ 'de $p_0 = (0,0), p_1 = (1,0), p_2 = (2,3)$ ve $p_3 = (-1,0)$ kontrol noktalarının oluşturduğu düzlemsel kübik Bézier eğrisinin parametrik denklemi (2.7) eşitliği yardımıyla

$$\begin{aligned}
\mathfrak{B}(t) &= \sum_{i=0}^3 B_i^n(t) p_i \\
&= (1-t)^3 p_0 + 3(1-t)^2 t p_1 + 3(1-t) t^2 p_2 + t^3 p_3 \\
&= (1-t)^3 (0,0) + 3(1-t)^2 t (1,0) + 3(1-t) t^2 (2,3) + t^3 (-1,0) \\
&= (0,0) + (3t(1-t)^2, 0) + (6(1-t)t^2, 9(1-t)t^2) + (-t^3, 0) \\
&= (0,0) + (3t - 6t^2 + 3t^3, 0) + (6t^2 - 6t^3, 9t^2 - 9t^3) + (-t^3, 0) \\
&= (3t - 4t^3, 9t^2 - 9t^3); \quad t \in [0,1]
\end{aligned}$$

dir (bkz. Şekil 2.9).



Şekil 2.9 $p_0 = (0,0), p_1 = (1,0), p_2 = (2,3)$ ve $p_3 = (-1,0)$ kontrol noktalarının oluşturduğu kübik Bézier eğrisinin Python 3.10 görüntüsü [59].

Şekil 2.9'un Python 3.10 kodu EK F' da verilmiştir [38, 73, 74].

Tanım 2.17 E^3 öklid uzayında $(n+1)$ kontrol noktası verilsin. Bu kontrol noktalar $p_0 = (x_0, y_0, z_0), p_1 = (x_1, y_1, z_1), p_2 = (x_2, y_2, z_2), \dots, p_n = (x_n, y_n, z_n)$ olsun. Bu noktaların oluşturduğu

$$\mathfrak{B}(t) = \sum_{i=0}^n B_i^n(t) p_i \quad (2.27)$$

$\mathfrak{B}(t)$ eğrisine n . dereceden Bézier eğrisi denir. Burada;

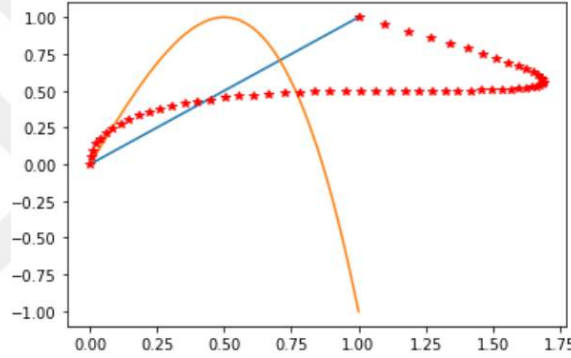
$$B_i^n(t) = \binom{n}{i} (1-t)^{n-i} t^i; \quad 0 \leq i \leq n, \quad (2.28)$$

ifadesi ile tanımlı $B_i^n(t)$ ifadesi n . dereceden Bernstein temel fonksiyonudur [27, 28, 38, 59].

Örnek 2.4 $\mathbb{E}^3$ 'de $p_0 = (0,0,0), p_1 = (1,0,1), p_2 = (2,3,0)$ ve $p_3 = (-1,1,1)$ kontrol noktalarının oluşturduğu düzlemsel kübik Bézier eğrisinin parametrik denklemi (2.7) eşitliği yardımıyla

$$\begin{aligned}
\mathfrak{B}(t) &= \sum_{i=0}^3 B_i^n(t) p_i \\
&= (1-t)^3 p_0 + 3(1-t)^2 t p_1 + 3(1-t)t^2 p_2 + t^3 p_3 \\
&= (1-t)^3 (0,0,0) + 3(1-t)^2 t (1,0,1) + 3(1-t)t^2 (2,3,0) + t^3 (-1,1,1) \\
&= (0,0) + (3t(1-t)^2, 0) + (6(1-t)t^2, 9(1-t)t^2) + (-t^3, 0) \\
&= (0,0) + (3t - 6t^2 + 3t^3, 0) + (6t^2 - 6t^3, 6t^2 - 6t^3) + (-t^3, 0) \\
&= (3t - 4t^3, 9t^2 - 9t^3); \quad t \in [0,1]
\end{aligned}$$

dır (bkz. Şekil 2.10).



Şekil 2.10 $p_0 = (0,0,0), p_1 = (1,0,1), p_2 = (2,3,0)$ ve $p_3 = (-1,1,1)$ kontrol noktalarının oluşturduğu düzlemsel kübik Bézier eğrisinin Python 3.10 görüntüsü [59].

Şekil 2.10'un Python 3.10 kodu EK G'da verilmiştir [38, 73, 74].

Teorem 2.1 Bernstein polinomu aşağıdaki özellikleri sağlar:

1. *Anti-negatiftir.* $B_i^n(t)$ fonksiyonlarının tamamı $0 \leq t \leq 1$ aralığında anti-negatiftir.

$t \in [0,1]$ için $B_k^n(t) \geq 0$ dır. Bu özellik;

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} > 0, t \geq 0 \text{ ve } (1-t \geq 0) \text{ olduğundan}$$

$$B_k^n(t) = \binom{n}{k} (1-t)^{n-k} t^k \geq 0 \quad (2.29)$$

olur.

2. *Birimin parçalanması* özelliğini sağlar.

$$\sum_{i=0}^n B_i^n(t) = 1 \quad (2.30)$$

eşitliği sağlanır. Bernstein polinomu $B_i^n(t)$ için, Eş. (2.15) binom teoreminden açıkça

$$(x + y)^n = \sum_{i=0}^n \binom{n}{i} x^i y^{n-i}$$

şeklinde yazılır. $x + y = 1$ olursa ve $x = t$ ise o halde $y = 1 - t$ dir. $n \in \mathbb{Z}^+$ için

$$1 = 1^n = [t + (1 - t)]^n = \sum_{i=0}^n \binom{n}{i} t^i (1 - t)^{n-i} = \sum_{i=0}^n B_i^n(t) \quad (2.31)$$

olduğu görülür.

3. *Simetriktir.*

$$B_i^n(t) = B_{n-i}^n(1 - t) \quad (2.32)$$

dir. Bernstein temel fonksiyonu $B_i^n(t)$ için;

$$B_{n-i}^n(t) = B_i^n(1 - t) \quad (2.32)$$

eşitliği sağlanır. Burada, n . dereceden bir Bernstein temel fonksiyonu (2.15) eşitliği ile verilmişti.

$$B_{n-i}^n(t) = \binom{n}{n-i} (1 - t)^i t^{n-i}$$

$$= \frac{n!}{(n-i)!i!} (1 - t)^i t^{n-i}$$

$$= \frac{n!}{i!(n-i)!} (1 - t)^i t^{n-i} = B_i^n(1 - t) \quad (2.33)$$

olur. Böylece, (2.33) ve (2.34) eşitliklerinin birbirine eşit olduğu görülür. Bu nedenle,

$$B_{n-i}^n(t) = B_i^n(1-t)$$

olur. Böylece, $B_{n-i}^n(t)$ fonksiyonunun grafiği, $B_i^n(1-t)$ fonksiyonunun yansımasıdır.

4. Yinelemelidir.

n . dereceden Bernstein temel fonksiyonları $B_i^n(t)$ $i = 0, 1, \dots, n$ için $(n-1)$. dereceden fonksiyonlarla

$$B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t) \quad (2.34)$$

eşitliği gibi ifade edilebilir.

Burada; $B_{-1}^{n-1}(t) = 0$ ve $B_n^{n-1}(t) = 0$ dır. n . dereceden bir Bernstein temel fonksiyonu (2.2.3) eşitliğini kullanarak

$$B_i^{n-1}(t) = \binom{n-1}{i} (1-t)^{n-1-i} t^i \quad (2.35)$$

ve

$$B_{i-1}^{n-1}(t) = \binom{n-1}{i-1} (1-t)^{n-i} t^{i-1} \quad (2.36)$$

eşitlikleri elde edilir. Böylece $i = 0$ için, $B_{-1}^{n-1}(t)$ yardımıyla

$$B_0^n(t) = (1-t)^n = (1-t)B_0^{n-1}(t) + tB_{-1}^{n-1}(t) \quad (2.37)$$

bulunur. Benzer şekilde $i = n$ için, $B_n^{n-1}(t) = 0$ yardımıyla

$$B_n^n(t) = t^n = (1-t)B_n^{n-1}(t) + tB_{n-1}^{n-1}(t) \quad (2.38)$$

yazılabilir. Genel halde, $1 \leq i \leq n$ için (2.36) ve (2.37) eşitlikleri sırasıyla $(1-t)$ ve t ile çarpılırsa Bernstein temel fonksiyonu tanımından

$$(1-t).B_i^{n-1}(t) + t.B_{i-1}^{n-1}(t)$$

$$\begin{aligned}
&= (1-t) \cdot \binom{n-1}{i} (1-t)^{n-1-i} t^i + t \cdot \binom{n-1}{i-1} (1-t)^{n-i} t^{i-1} \\
&= \binom{n-1}{i} (1-t)^{n-i} t^i + \binom{n-1}{i-1} (1-t)^{n-i} t^i \\
&= (1-t)^{n-i} t^i \cdot \left[\binom{n-1}{i} + \binom{n-1}{i-1} \right] \\
&= (1-t)^{n-i} t^i \cdot \left[\frac{(n-1)!}{(n-1-i)! i!} + \frac{(n-1)!}{(n-i)! (i-1)!} \right] \\
&= (1-t)^{n-i} t^i \cdot \left[\frac{(n-i)+i}{n} \right] \cdot \frac{n!}{i! (n-1)!} \\
&= \binom{n}{i} (1-t)^{n-i} t^i \\
&= B_i^n(t)
\end{aligned}$$

dir.

5. *Türevlenebilirdir.*

$$\frac{dB_i^n(t)}{dt} = n[B_{i-1}^{n-1}(t) - B_i^{n-1}(t)] \quad (2.40)$$

dir. Eş. (2.15)'nin her iki tarafının t 'ye göre türevi alınır,

$$\begin{aligned}
\frac{dB_i^n(t)}{dt} &= \frac{n!}{i! (n-i)!} [-(n-i)(1-t)^{n-1-i} t^i + i(1-t)^{n-i} t^{i-1}] \\
&= -\frac{n!}{i!(n-1-i)!} (1-t)^{n-1-i} t^i + \frac{n!}{(i-1)!(n-i)!} (1-t)^{n-i} t^{i-1} \\
&= n \left[\frac{(n-1)!}{(i-1)!(n-1)!} (1-t)^{n-i} t^{i-1} - \frac{(n-1)!}{i!(n-1-i)!} (1-t)^{n-1-i} t^i \right] \\
&= n \left[\binom{n-1}{i-1} (1-t)^{n-i} t^{i-1} - \binom{n-1}{i} (1-t)^{n-1-i} t^i \right]
\end{aligned}$$

$$= n[B_{i-1}^{n-1}(t) - B_i^{n-1}(t)]$$

eşitliği elde edilir [27-30].

Teorem 2.2 $\mathfrak{B}(t)$ ile tanımlanan Bézier eğrisi aşağıdaki özellikleri sağlar:

1. *Uç nokta şartları sağlanır.* Yani Bézier eğrisi kontrol çokgeninin $t = 0$ başlangıç noktasından geçerken $t = 1$ noktasında da eğri bitiş noktasından geçer.
2. *Uç noktalarda teğet olma şartları sağlanır.* $\mathfrak{B}^n(t)$, kontrol noktaları $p_0, p_1, \dots, p_n$ olan n . dereceden bir Bézier eğrisinin $t = 0$ başlangıç noktasındaki ve $t = 1$ bitiş noktasındaki teğeti sırasıyla

$$\left. \frac{d\mathfrak{B}^n(t)}{dt} \right|_{t=0} = n(p_1 - p_0) \text{ ve } \left. \frac{d\mathfrak{B}^n(t)}{dt} \right|_{t=1} = n(p_n - p_{n-1}) \text{ dır.}$$

3. *Geometrik invarianttır.* T bir afin dönüşüm olarak tanımlansın. O halde,

$$T(\sum_{i=0}^n p_i B_i^n(t)) = \sum_{i=0}^n T(p_i) B_i^n(t)$$

eşitliği sağlanır. Yani, Bernstein temel fonksiyonunun birimin parçalanması özelliğinden eğrinin şekli, koordinat sisteminin dönme ve ötelemesi altında değişmezdir.

4. *Konveks kafes özelliğine sahiptir.* n . dereceden bir $\mathfrak{B}^n(t)$ Bézier eğrisi $p_0, p_1, \dots, p_n$ kontrol noktaları ile oluşturulan konveks kafesin tamamen içerisinde kalmaktadır ve böylece kontrol noktaları değiştirilerek eğri istenilen şekilde değiştirilebileceği için bu durum tasarımcı için büyük kolaylık sağlamaktadır.

5. *Simetri özelliği sağlanır.*

$$\sum_{i=0}^n P_i B_i^n(t) = \sum_{i=0}^n P_{n-i} B_i^n(1-t) \quad (2.41)$$

dir [27-30].

2.2.2 Bézier Eğrisinin Türevleri

Teğetleri ve normalleri belirleme gibi eğrileri kapsayan birçok uygulama sırasında, türev hesaplamaya ihtiyaç duyulur. Bézier eğrisi tanımının temelinde Bernstein temel fonksiyonu yer aldığından Bézier eğrisinin türevi hesaplanırken Bernstein temel fonksiyonu türevinden yararlanılır. Bu yüzden öncelikle n . dereceden Bernstein temel fonksiyonunun türevleri aşağıda ele alınmıştır.

Teorem 2.3 $\mathbb{E}^n$ Öklid uzayında (2.27) denklemi ile verilen bir Bézier eğrisinin türevine ulaşabilmek için n . dereceden Bernstein temel fonksiyonlarının türevlerini inceleyelim.

Birinci türevi,

$$B'_{i,n}(t) = \frac{i-nt}{t(1-t)} B_i^n(t) \quad (2.42)$$

ve ikinci türevi

$$B''_{i,n}(t) = \left(\frac{i(i-1)-2i(n-1)t+n(n-1)t^2}{t^2(1-t)^2} \right) B_i^n(t) \quad (2.43)$$

ile ifade edilir. Ayrıca, Bernstein temel fonksiyonunun birinci türevini

$$B'_{i,n}(t) = n(B_{i-1}^{n-1}(t) - B_i^{n-1}(t)) \quad (2.44)$$

eşitliği ile göstermek mümkündür [27, 30, 76].

İspat: (2.28) eşitliği ile verilen Bernstein temel fonksiyonuna çarpımın türevi kuralı uygulanırsa

$$\begin{aligned} B'_{i,n}(t) &= \binom{n}{i} (i(1-t)^{n-i}t^{i-1} - (n-i)(1-t)^{n-i-1}t^i) \\ &= \left(\frac{i-nt}{t(1-t)} \right) \binom{n}{i} (1-t)^{n-i}t^i \\ &= \left(\frac{i-nt}{t(1-t)} \right) B_i^n(t) \end{aligned}$$

elde edilir. (2.42) eşitliğine tekrar türev alma işlemi uygulanırsa

$$\begin{aligned}
B''_{i,n}(t) &= \left(\frac{i-nt}{t(1-t)} \right)' B_i^n(t) + \left(\frac{i-nt}{t(1-t)} \right) B'_{i,n}(t) \\
&= \left(\frac{2it-nt^2-i}{t^2(1-t)^2} \right) B_i^n(t) + \left(\frac{i-nt}{t(1-t)} \right)^2 B_i^n(t) \\
&= \left(\frac{i(i-1)-2i(n-1)t+n(n-1)t^2}{t^2(1-t)^2} \right) B_i^n(t)
\end{aligned}$$

olarak bulunur. Ayrıca,

$$\begin{aligned}
B'_{i,n}(t) &= -\binom{n}{i}(n-i)(1-t)^{n-i-1}t^i + \binom{n}{i}i(1-t)^{n-i}t^{i-1} \\
&= -\frac{n!}{(n-i)!i!}(n-i)(1-t)^{n-i-1}t^i + \frac{n!}{(n-i)!i!}i(1-t)^{n-i}t^{i-1} \\
&= -\frac{n(n-1)!}{(n-i-1)!i!}(1-t)^{n-i-1}t^i + \frac{n(n-1)!}{(n-i)!(i-1)!}(1-t)^{n-i}t^{i-1} \\
&= n\left(-\binom{n-i}{i}(1-t)^{n-i-1}t^i + \binom{n-1}{i-1}(1-t)^{n-i}t^{i-1}\right) \\
&= n(B_{i-1}^{n-1}(t) - B_i^{n-1}(t))
\end{aligned}$$

elde edilir. ■

Teorem 2.3 yardımıyla n . dereceden Bézier eğrilerinin türevleri için aşağıda verilen teorem ve sonuçlar elde edilir.

Teorem 2.4 $\mathbb{E}^n$ Öklid uzayında (2.27) denklemi ile verilen n . dereceden bir Bézier eğrisinin birinci türevi

$$\mathfrak{B}'(t) = n \sum_{i=0}^{n-1} B_i^{n-1}(t) \Delta^1 p_i \quad (2.45)$$

şeklindeir. Burada, p_i ($i = 0, 1, 2, \dots, n$) kontrol noktaları olmak üzere

$$\Delta^1 p_i = p_{i+1} - p_i \quad (2.46)$$

olup *ileri fark operatörü* olarak adlandırılır [27, 30, 76].

İspat: Bernstein temel fonksiyonunun rekürsiyon özelliğinden $B_{-1}^{n-1}(t) = B_n^{n-1}(t) = 0$ olmak üzere (2.44) eşitliği de kullanılarak,

$$\begin{aligned}\mathfrak{B}'(t) &= \sum_{i=0}^n B'_{i,n}(t)p_i = \sum_{i=0}^n n \left(B_{i-1}^{n-1}(t) - B_i^{n-1}(t) \right) p_i \\ &= \sum_{i=0}^n n B_{i-1}^{n-1}(t)p_i - \sum_{i=0}^n n B_i^{n-1}(t)p_i \\ &= \sum_{i=0}^n n B_{i-1}^{n-1}(t)p_i - \sum_{i=0}^{n-1} n B_i^{n-1}(t)p_i\end{aligned}$$

elde edilir. Son ifadenin toplamaları tekrar düzenlenirse

$$\begin{aligned}\mathfrak{B}'(t) &= \sum_{i=0}^{n-1} n B_i^{n-1}(t)p_{i+1} - \sum_{i=0}^{n-1} n B_i^{n-1}(t)p_i \\ &= \sum_{i=0}^{n-1} n B_i^{n-1}(t)(p_{i+1} - p_i) = n \sum_{i=0}^{n-1} B_i^{n-1}(t)\Delta^1 p_i\end{aligned}$$

sonucu elde edilir.

Bézier eğrisinin birinci türev formülüne tekrar türev uygulaması yapılarak ikinci ve daha yüksek mertebeden türevlere ulaşılır.

Sonuç 2.2 $\mathbb{E}^n$ Öklid uzayında (2.27) denklemi ile verilen n . dereceden bir Bézier eğrisinin ikinci türevi Teorem 2.4' e tekrar türev formülünün uygulanmasıyla

$$\mathfrak{B}''(t) = n(n-1) \sum_{i=0}^{n-2} B_i^{n-2}(t)\Delta^2 p_i \quad (2.47)$$

şeklinde elde edilir. Burada, Δ ileri fark operatörünün ikinci türevi

$$\Delta^2 p_i = (\Delta^1 p_{i+1} - \Delta^1 p_i) = (p_{i+2} - 2p_{i+1} + p_i) \quad (2.48)$$

ifadesine eşittir.

Sonuç 2.3 $\mathbb{E}^n$ öklid uzayında (2.27) denklemi ile verilen n . dereceden bir Bézier eğrisinin r . türevi,

$$\mathfrak{B}^r(t) = \frac{n!}{(n-r)!} \sum_{i=0}^{n-r} B_i^{n-r}(t)\Delta^r p_i \quad (2.49)$$

şeklindedir. Burada, Δ ileri fark operatörünün r . türevi

$$\Delta^r p_i = \Delta^{r-1} p_{i+1} - \Delta^{r-1} p_i \quad (2.50)$$

ifadesine eşittir [27, 30, 76].

2.3 Minkowski Uzayı İle İlgili Kavramlar

Bu bölümde Minkowski geometrisinde ve tezin geriye kalan kısımlarında sıklıkla kullanılacak bazı temel kavramlara yer verilecektir.

$\mathbb{R}_1^3 = (\mathbb{E}^3, \langle \cdot, \cdot \rangle)$ ile gösterilen Lorentz ya da Minkowski uzayı üç boyutlu Öklid uzayı üzerine kurulmuş bir metrik uzaydır. Bu metriğin Öklid iç çarpımından farklı olmasının nedeni son teriminin negatif işaretli olmasıdır. Böylece çıkan sonuca göre bu uzayda iççarpım üç sınıfa ayrılarak incelenebilir. Bu sınıflandırma timelike (zamansal), spacelike (uzaysal) ve lightlike (ışıksal) olarak yapılabilir. [76]

Tanım 2.18 $\mathbb{R}^2$ uzayında standart reel vektör uzayı üzerinde

$$\therefore \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$$

$$(x, y) \rightarrow \langle x, y \rangle = x_1 y_1 + x_2 y_2 ; x = (x_1, x_2), y = (y_1, y_2) \quad (2.51)$$

Öklid iç çarpımı tanımlanırsa, $\mathbb{R}^2$ afin uzayına, Öklidyen 2-uzay denir. $\mathbb{R}^2$ ile gösterilir [77, 78].

Tanım 2.19 $\mathbb{R}^2$ uzayında standart reel vektör uzayı üzerinde

$$\therefore \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$$

$$(x, y) \rightarrow \langle x, y \rangle = x_1 y_1 - x_2 y_2 ; x = (x_1, x_2), y = (y_1, y_2) \quad (2.52)$$

Minkowski iç çarpımı tanımlanırsa, $\mathbb{R}^2$ afin uzayı, iki boyutlu Minkowski uzayı veya iki boyutlu Lorentziyen uzayı olarak isimlendirilir ve $\mathbb{R}_1^2$ ile gösterilir [77, 78].

Sonuç 2.4 $\mathbb{R}^3$ uzayında standart reel vektör uzayı üzerinde

$$\therefore \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}$$

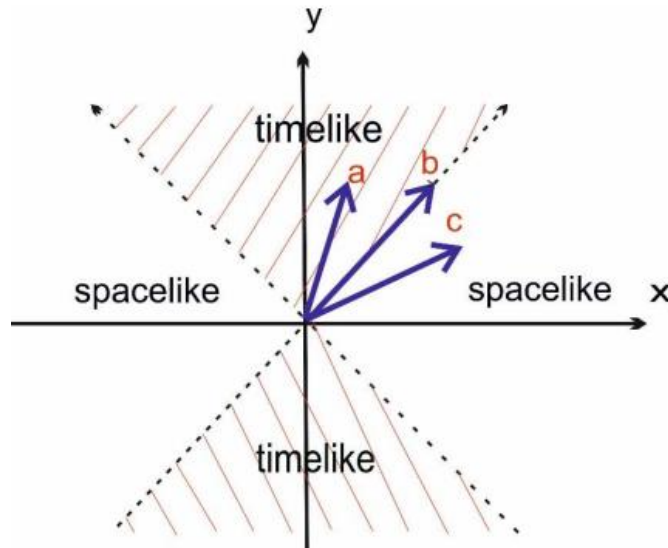
$$(x, y) \rightarrow \langle x, y \rangle = x_1 y_1 + x_2 y_2 - x_3 y_3 ; x = (x_1, x_2, x_3), y = (y_1, y_2, y_3) \quad (2.53)$$

Minkowski iç çarpımı tanımlanırsa, $\mathbb{R}^3$ afin uzayı, üç boyutlu Minkowski uzayı olarak isimlendirilir ve $\mathbb{R}_1^3$ ile gösterilir [76, 78-82].

Tanım 2.20 $\mathbb{R}_1^2$ uzayındaki $x = (x_1, x_2)$ herhangi bir vektör olmak üzere

1. Metrik, pozitif tanımlı yani $\langle x, x \rangle > 0$ veya $x = 0$ ise x' e spacelike (uzaysal) vektör,
2. Metrik, negatif tanımlı yani $\langle x, x \rangle < 0$ ise x' e timelike (zamansal) vektör,
3. Metrik, dejenere metrik yani $\langle x, x \rangle = 0, x \neq 0$ ise x' e lightlike (veya null) vektör

denir [78, 81]. $\mathbb{R}_1^2$ Minkowski düzleminde 1. ve 2. açkırtaylar üzerinde lightlike vektörler (b), iki açkırtay arasında timelike vektörler (a) ve açkırtayların dışında da spacelike vektörler (c) bulunur.



Şekil 2.11 Minkowski düzleminde vektörler [79]

Tanım 2.21 Bir $x \in \mathbb{R}_1^2$ vektörünün normu $\|x\|_{IL} = \sqrt{|\langle x, x \rangle|}$ olarak tanımlanır ve

1. $\|x\|_{IL} > 0$,
2. $\|x\|_{IL} = 0 \Leftrightarrow x$ bir lightlike (null) vektördür.
3. x bir timelike vektör ise $\|x\|_{IL}^2 = -\langle x, x \rangle$,
4. x bir spacelike vektör ise $\|x\|_{IL}^2 = \langle x, x \rangle$

özellikleri sağlanır [77, 78, 80, 81].

Tanım 2.22 Üç boyutlu Minkowski $(\mathbb{R}_1^3)$ uzayında $x = (x_1, x_2, x_3)$, $y = (y_1, y_2, y_3)$ iki vektör olmak üzere,

$$\langle x, y \rangle = x_1 y_1 + x_2 y_2 - x_3 y_3 \quad (2.54)$$

$$x \wedge_{IL} y = (x_3 y_2 - x_2 y_3, x_1 y_3 - x_3 y_1, x_1 y_2 - x_2 y_1) \quad (2.55)$$

çarpımları sırasıyla iç çarpım ve vektörel (dış) çarpımı olarak tanımlanır. Minkowski-3 uzayının baz vektörleri $e_i = (\delta_{i1}, \delta_{i2}, \delta_{i3})$, $i = 1, 2, 3$ için Minkowski vektörel çarpımının determinant ile gösterimi

$$x \wedge_{IL} y = -\det \begin{bmatrix} e_1 & e_2 & e_3 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix}$$

olarak verilir, burada $\delta_{ij} = \begin{cases} 1 & , i = j \\ 0 & , i \neq j \end{cases}$ dir.

Baz vektörleri $e_1 \wedge_{IL} e_2 = e_3$, $e_2 \wedge_{IL} e_3 = -e_1$, $e_3 \wedge_{IL} e_1 = -e_2$ vektörel çarpım koşullarını sağlamaktadır [76-81].

Teorem 2.5 Üç boyutlu R_1^3 uzayında üç vektör $x = (x_1, x_2, x_3)$, $y = (y_1, y_2, y_3)$ ve $z = (z_1, z_2, z_3)$ olsun. Bu durumda,

$$1. \quad \langle x \wedge_{IL} y, z \rangle = -\det(x, y, z) \quad (2.56)$$

$$2. \quad (x \wedge_{IL} y) \wedge_{IL} z = -\langle x, z \rangle y + \langle y, z \rangle x \quad (2.57)$$

$$3. \quad \langle x \wedge_{IL} y, x \rangle = 0 \text{ ve } \langle x \wedge_{IL} y, y \rangle = 0 \quad (2.58)$$

$$4. \quad \langle x \wedge_{IL} y, x \wedge_{IL} y \rangle = -\langle x, x \rangle \langle y, y \rangle + (\langle x, y \rangle)^2 \quad (2.59)$$

eşitlikleri sağlanır [78, 79, 81].

İspat:

1. (2.55)'dan $x \wedge_{IL} y = (x_3y_2 - x_2y_3, x_1y_3 - x_3y_1, x_1y_2 - x_2y_1)$ dir.

$$\langle x \wedge_{IL} y, z \rangle = x_3y_2z_1 - x_2y_3z_1 + x_1y_3z_2 - x_3y_1z_2 - x_1y_2z_3 + x_2y_1z_3 \quad (2.60)$$

elde edilir.

$$\begin{aligned} -\det(x, y, z) &= -\begin{vmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{vmatrix} \\ &= -x_1y_2z_3 + x_1y_3z_2 + x_2y_1z_3 - x_2y_3z_1 - x_3y_1z_2 + x_3y_2z_1 \end{aligned} \quad (2.61)$$

bulunur. (2.60) ve (2.55)' den

$$\langle x \wedge_{IL} y, z \rangle = -\det(x, y, z)$$

dir.

$$\begin{aligned} 2. \quad (x \wedge_{IL} y) \wedge_{IL} z &= -\begin{vmatrix} e_1 & e_2 & e_3 \\ x_3y_2 - x_2y_3 & x_1y_3 - x_3y_1 & x_1y_2 - x_2y_1 \\ z_1 & z_2 & z_3 \end{vmatrix} \\ &= (-x_1y_3z_3 + x_3y_1z_3 + x_1y_2z_2 - x_2y_1z_2)e_1 + (x_3y_2z_3 - x_2y_3z_3 - x_1y_2z_1 + \\ &\quad x_2y_1z_1) + (x_3y_2z_2 - x_2y_3z_2 - x_1y_3z_1 + x_3y_1z_1)e_3 \end{aligned}$$

elde edilir. $(x \wedge_{IL} y) \wedge_{IL} z$ vektörünün birinci bileşenine $x_1y_1z_1$, ikinci bileşenine $x_2y_2z_2$ ve üçüncü bileşenine $x_3y_3z_3$ ifadeleri bir eklenip bir çıkarılacak olursa

$$\begin{aligned} (x \wedge_{IL} y) \wedge_{IL} z &= (-x_1y_3z_3 + x_3y_1z_3 + x_1y_2z_2 - x_2y_1z_2 - x_1y_1z_1 + x_1y_1z_1)e_1 \\ &\quad + (x_3y_2z_3 - x_2y_3z_3 - x_1y_2z_1 + x_2y_1z_1 + x_2y_2z_2 - x_2y_2z_2)e_2 \end{aligned}$$

$$\begin{aligned}
& +(x_3y_2z_2 - x_2y_3z_2 - x_1y_3z_1 + x_3y_1z_1 + x_3y_3z_3 - x_3y_3z_3)e_3 \\
& = (-x_1z_1 - x_2z_2 + x_3z_3)(y_1e_1 + y_2e_2 + y_3e_3) \\
& + (y_1z_1 + y_2z_2 - y_3z_3)(x_1e_1 + x_2e_2 + x_3e_3) \\
& = -\langle x, z \rangle y + \langle y, z \rangle x
\end{aligned}$$

elde edilir.

3. (2.56)'den $\langle x \wedge_{IL} y, x \rangle = -\det(x, y, x) = 0$ ve $\langle x \wedge_{IL} y, y \rangle = -\det(x, y, y) = 0$ dır.

4. (2.56), (2.57) ve determinant fonksiyonunun özelliklerinden

$$\begin{aligned}
\langle x \wedge_{IL} y, x \wedge_{IL} y \rangle &= -\det(x, y, x \wedge_{IL} y) \\
&= \det(x, x \wedge_{IL} y, y) \\
&= -\det(x \wedge_{IL} y, x, y) \\
&= \langle (x \wedge_{IL} y) \wedge_{IL} x, y \rangle \\
&= \langle -\langle x, x \rangle y + \langle y, x \rangle x, y \rangle \\
&= -\langle x, x \rangle \langle y, y \rangle + (\langle x, y \rangle)^2
\end{aligned}$$

bulunur [79]. ■



BÖLÜM 3

ÜÇ BOYUTLU MINKOWSKI UZAYINDA BÉZIER EĞRİLERİ

Bu bölümde, bilgisayar destekli geometrik tasarımda sıklıkla kullanılan kübik Bézier eğrilerinin öncelikle genel tanımları verilmiş, daha sonra da özel olarak $t = 0$ başlangıç ve $t = 1$ bitiş noktalarındaki Serret-Frenet çatısı, eğriliği ve burulması incelenmiştir.

3.1 TIMELIKE BÉZIER EĞRİLERİNİN SERRET-FRENET ÇATISI, EĞRİLİĞİ VE BURULMASI

Bu alt başlıkta timelike Bézier eğrilerinin Serret-Frenet çerçevesi, eğriliği ve burulması incelenecektir.

$\mathfrak{B}(t)$ birim hızlı olmayan timelike Bézier eğrisi olsun. $\mathfrak{B}(t)$ Bézier eğrisi timelike ise Serret-Frenet çatısındaki T teğet vektörü de timelike olmalıdır. $\mathfrak{B}(t)$ timelike Bézier eğrisinin $t = 0$ başlangıç noktasındaki T teğet vektörü timelike, B binormal vektörü ve N asli normal spacelike için $\{T, N, B\}|_{t=0}$ ortonormal çatısı tanımlansın. Böylece,

$$\left. \begin{aligned} \langle T, T \rangle &= -1, & \langle N, N \rangle &= 1, & \langle B, B \rangle &= 1 \\ \langle T, N \rangle &= 0, & \langle T, B \rangle &= 0, & \langle N, B \rangle &= 0 \end{aligned} \right\} \quad (3.1)$$

koşulları sağlanır [78].

Teorem 3.1 Δp_i ve Δp_j timelike (zamansal) vektörler ise

$$|\langle \Delta p_i, \Delta p_j \rangle|_{IL} = -\|\Delta p_i\|_{IL} \|\Delta p_j\|_{IL} \cosh \theta \quad (3.2)$$

$$\|\Delta p_i \wedge_{IL} \Delta p_j\|_{IL} = \|\Delta p_i\|_{IL} \|\Delta p_j\|_{IL} \sinh \theta \quad (3.3)$$

olur ve böylece Δp_i ve Δp_j timelike vektörleri arasındaki θ açısı da timelike vektördür [78, 81].

Teorem 3.2 $\mathfrak{B}(t)$ timelike Bézier eğrisi ve $p_j \in R^n$ 'ler kontrol noktaları olsun. $\mathfrak{B}(t)$ 'nin Serret-Frenet çerçevesi T, N, B , eğriliği κ ve burulması τ ;

$$T(t) = \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\left(-\sum_{j,i=0}^{m-1} B_j^{m-1}(t) B_i^{m-1}(t) \langle \Delta p_j, \Delta p_i \rangle \right)^{\frac{1}{2}}} \quad (3.4)$$

$$N(t) = - \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-1} B_j^{m-1}(t) B_i^{m-2}(t) B_k^{m-1}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \wedge_{IL} \Delta p_k}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL} \left\| \sum_{k=0}^{m-1} B_k^{m-1}(t) \Delta p_k \right\|_{IL}} \quad (3.5)$$

$$B(t) = \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL}} \quad (3.6)$$

$$\kappa(t) = \frac{m-1}{m} \frac{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL}}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}^3} \quad (3.7)$$

$$\tau(t) = - \frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(t) B_i^{m-2}(t) B_k^{m-3}(t) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL}^2} \quad (3.8)$$

denklemleri ile verilir. Burada $\Delta p_j = p_{j+1} - p_j$, $\Delta^2 p_j = \Delta p_{j+1} - \Delta p_j$ ve $\Delta^3 p_j = \Delta^2 p_{j+1} - \Delta^2 p_j$ dir [68, 78, 81].

İspat: Δp_j vektörleri timelike (zamansal) vektörler olduğundan, $t \in [0,1]$ için Δp_j 'nin normu

$$\|\Delta p_j\|_{IL} = \sqrt{-\langle \Delta p_j, \Delta p_j \rangle} \quad (3.9)$$

dır. Böylece Serret-Frenet çatısının, teğet vektörü Eş. (2.1.9)'dan

$$\begin{aligned} T(t) &= \frac{\mathfrak{B}'(t)}{\|\mathfrak{B}'(t)\|_{IL}} = \frac{m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\left\| m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}} \\ &= \frac{m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{|m| \left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}} \end{aligned}$$

$$\begin{aligned}
&= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\sqrt{-\langle \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j, \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \rangle}} \\
&= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\sqrt{-\sum_{j=0}^{m-1} \sum_{j=0}^{m-1} B_j^{m-1}(t) B_j^{m-1}(t) \langle \Delta p_j, \Delta p_j \rangle}} \\
&= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\sqrt{-\left(-\left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}^2\right)}} \\
&= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}}
\end{aligned}$$

denklemini ile elde edilir. Binormal vektörü (2.11) eşitlikten;

$$\begin{aligned}
B(t) &= \frac{\mathfrak{B}'(t) \Delta_{IL} \mathfrak{B}''(t)}{\|\mathfrak{B}'(t) \Delta_{IL} \mathfrak{B}''(t)\|_{IL}} \\
&= \frac{m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \wedge_{IL} m(m-1) \sum_{i=0}^{m-2} B_i^{m-2}(t) \Delta^2 p_j}{\left\| m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \wedge_{IL} m(m-1) \sum_{i=0}^{m-2} B_i^{m-2}(t) \Delta^2 p_j \right\|_{IL}} \\
&= \frac{m^2(m-1) \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| m^2(m-1) \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\
&= \frac{m^2(m-1) \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{|m^2(m-1)| \left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\
&= \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}}
\end{aligned}$$

denklemini ile elde edilir. Timelike Bézier eğrisinin T teğet vektörü timelike, N ana normal vektörü ve B binormal vektörü spacelike vektörler olduğundan ana normal vektörü olan N (2.10) eşitliğinden

$$\begin{aligned}
N(t) &= -B(t) \wedge_{IL} T(t) \\
&= -\frac{\mathfrak{B}'(t) \Delta_{IL} \mathfrak{B}''(t)}{\|\mathfrak{B}'(t) \Delta_{IL} \mathfrak{B}''(t)\|_{IL}} \wedge_{IL} \frac{\mathfrak{B}'(t)}{\|\mathfrak{B}'(t)\|_{IL}}
\end{aligned}$$

$$\begin{aligned}
&= \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \wedge_{IL} \frac{\sum_{k=0}^{m-1} B_k^{m-1}(t) \Delta p_k}{\left\| \sum_{k=0}^{m-1} B_k^{m-1}(t) \Delta p_k \right\|_{IL}} \\
&= - \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-1} B_j^{m-1}(t) B_i^{m-2}(t) B_k^{m-1}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \wedge_{IL} \Delta p_k}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL} \left\| \sum_{k=0}^{m-1} B_k^{m-1}(t) \Delta p_k \right\|_{IL}}
\end{aligned}$$

denklemini ile elde edilir. Eş. (2.12) kullanılarak timelike Bézier eğrisinin eğriliği;

$$\begin{aligned}
\kappa(t) &= \frac{\|\mathfrak{B}'(t) \Delta_{IL} \mathfrak{B}''(t)\|_{IL}}{\|\mathfrak{B}'(t)\|_{IL}^3} \\
&= \frac{\left\| m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \wedge_{IL} m(m-1) \sum_{i=0}^{m-2} B_i^{m-2}(t) \Delta^2 p_i \right\|_{IL}}{\left\| m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}^3} \\
&= \frac{|m^2(m-1)| \left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}}{|m^3| \left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}^3} \\
&= \frac{m-1}{m} \frac{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}^3}
\end{aligned}$$

denklemini ile elde edilir ve Eş. (2.13)'den timelike Bézier eğrisinin burulması;

$$\begin{aligned}
\tau(t) &= \frac{\langle \mathfrak{B}'(t) \Delta_{IL} \mathfrak{B}''(t), \mathfrak{B}'''(t) \rangle}{\|\mathfrak{B}'(t) \Delta_{IL} \mathfrak{B}''(t)\|_{IL}^2} \\
&= \frac{\langle m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \wedge_{IL} m(m-1) \sum_{i=0}^{m-2} B_i^{m-2}(t) \Delta^2 p_i, m(m-1)(m-2) \sum_{k=0}^{m-3} B_k^{m-3}(t) \Delta^3 p_k \rangle}{\left\| m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \wedge_{IL} m(m-1) \sum_{i=0}^{m-2} B_i^{m-2}(t) \Delta^2 p_i \right\|_{IL}^2} \\
&= - \frac{m^3(m-1)^2(m-2) \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(t) B_i^{m-2}(t) B_k^{m-3}(t) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{m^4(m-1)^2 \left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2} \\
&= - \frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(t) B_i^{m-2}(t) B_k^{m-3}(t) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2}
\end{aligned}$$

denklemini ile elde edilir.

Teorem 3.1 ve Teorem 3.2' den Sonuç 3.1 ve Sonuç 3.2 elde edilebilir.

Sonuç 3.1 $\mathfrak{B}(t)$ timelike Bézier eğrisi ve p_j 'ler kontrol noktası olsun. $\mathfrak{B}(t)$ nin $t = 0$ 'daki Serret-Frenet çerçevesi T, N, B , eğriliği κ ve burulması τ :

$$T(0) = \frac{\Delta p_0}{\sqrt{-\langle \Delta p_0, \Delta p_0 \rangle}} \quad (3.10)$$

$$N(0) = \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \coth \theta - \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta \quad (3.11)$$

$$B(0) = \frac{\Delta p_0 \wedge_{IL} \Delta p_1}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta} \quad (3.12)$$

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL} \sinh \theta}{\|\Delta p_0\|_{IL}^2} \quad (3.13)$$

$$\tau(0) = -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}^2} \quad (3.14)$$

eşitlikleri ile verilir ve burada θ , Δp_0 ve Δp_1 arasındaki açıdır [78, 81].

İspat: (2.28) denklemindeki Bernstein temel fonksiyonunun tanımından;

$$B_0^{m-1}(t) = \binom{m-1}{0} (1-t)^{m-1} t^0 = \frac{(m-1)!}{(m-1)!0!} (1-t)^{m-1} t^0 = (1-t)^{m-1}$$

$$B_1^{m-1}(t) = \binom{m-1}{1} (1-t)^{m-2} t^1 = \frac{(m-1)!}{(m-2)!1!} (1-t)^{m-2} t^1 = (m-1)(1-t)^{m-2} \cdot t$$

$$B_{m-2}^{m-1}(t) = \binom{m-1}{m-2} (1-t)^1 t^{m-2} = \frac{(m-1)!}{1!(m-2)!} (1-t)^1 t^{m-2} = (m-1)(1-t) t^{m-2}$$

$$B_{m-1}^{m-1}(t) = \binom{m-1}{m-1} (1-t)^0 t^{m-1} = \frac{(m-1)!}{0!(m-1)!} (1-t)^0 t^{m-1} = t^{m-1}$$

$$B_0^{m-2}(t) = \binom{m-2}{0} (1-t)^{m-2} t^0 = \frac{(m-2)!}{(m-2)!0!} (1-t)^{m-2} t^0 = (1-t)^{m-2}$$

$$B_1^{m-2}(t) = \binom{m-2}{1} (1-t)^{m-3} t^1 = \frac{(m-2)!}{(m-3)!1!} (1-t)^{m-3} t^1 = (m-2)(1-t)^{m-3} \cdot t$$

$$B_{m-2}^{m-2}(t) = \binom{m-2}{m-2} (1-t)^0 t^{m-2} = \frac{(m-2)!}{0!(m-2)!} (1-t)^0 t^{m-2} = t^{m-2}$$

ifadeleri elde edilir. Böylece $\mathfrak{B}(t)$ 'nin $t = 0$ noktasındaki Serret-Frenet çatısı T, N, B , eğriliği κ ve burulması τ aşağıdaki gibi elde edilir. Teğet vektörü;

$$\begin{aligned}
T(0) &= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(0) \Delta p_j}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(0) \Delta p_j \right\|_{IL}} \\
&= \frac{B_0^{m-1}(0) \Delta p_0 + B_1^{m-1}(0) \Delta p_1 + \dots + B_{m-2}^{m-1}(0) \Delta p_{m-2} + B_{m-1}^{m-1}(0) \Delta p_{m-1}}{\left\| B_0^{m-1}(0) \Delta p_0 + B_1^{m-1}(0) \Delta p_1 + \dots + B_{m-2}^{m-1}(0) \Delta p_{m-2} + B_{m-1}^{m-1}(0) \Delta p_{m-1} \right\|_{IL}} \\
&= \frac{(1-0)^{m-1} \Delta p_0 + (m-1)(1-0)^2 \cdot 0 \Delta p_1 + \dots + (m-1)(1-0) \cdot 0^2 \Delta p_{m-2} + 0^{m-1} \Delta p_{m-1}}{\left\| (1-0)^{m-1} \Delta p_0 + (m-1)(1-0)^2 \cdot 0 \Delta p_1 + \dots + (m-1)(1-0) \cdot 0^2 \Delta p_{m-2} + 0^{m-1} \Delta p_{m-1} \right\|_{IL}} \\
&= \frac{\Delta p_0}{\left\| \Delta p_0 \right\|_{IL}} = \frac{\Delta p_0}{\sqrt{-\langle \Delta p_0, \Delta p_0 \rangle}}
\end{aligned}$$

denklemleri ile hesaplanır. Binormal vektörü;

$$\begin{aligned}
B(0) &= \frac{\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)}{\left\| \mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0) \right\|_{IL}} \\
&= \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\
&= \frac{(B_0^{m-1}(0) + B_1^{m-1}(t) + \dots + B_{m-1}^{m-1}(0)) (B_0^{m-2}(0) + B_1^{m-2}(0) + \dots + B_{m-2}^{m-2}(0)) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| (B_0^{m-1}(0) + B_1^{m-1}(t) + \dots + B_{m-1}^{m-1}(0)) (B_0^{m-2}(0) + B_1^{m-2}(0) + \dots + B_{m-2}^{m-2}(0)) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\
&= \frac{B_0^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + B_0^{m-1}(0) B_1^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_0^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{\left\| B_0^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + B_0^{m-1}(0) B_1^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_0^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle \right\|_{IL}} \\
&\quad + \frac{B_1^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + B_1^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{B_1^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + B_1^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle} \\
&\quad + \frac{B_{m-1}^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + B_{m-1}^{m-1}(0) B_1^{m-2}(0) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle}{B_{m-1}^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + B_{m-1}^{m-1}(0) B_1^{m-2}(0) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle} \\
&\quad + \frac{\dots + B_{m-1}^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle}{\dots + B_{m-1}^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle} \Big\|_{IL} \\
&= \frac{(1-0)^{m-1} (1-0)^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + (1-0)^{m-1} (m-2) (1-0)^{m-3} \cdot 0 \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 0^{m-1} 0^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{\left\| (1-0)^{m-1} (1-0)^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + (1-0)^{m-1} (m-2) (1-0)^{m-3} \cdot 0 \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 0^{m-1} 0^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle \right\|_{IL}}
\end{aligned}$$

$$\begin{aligned}
& + \frac{(m-1)(1-0)^{m-2} \cdot 0 \cdot (1-0)^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + (m-1)(1-0)^{m-2} \cdot 0 \cdot 0^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{(m-1)(1-0)^{m-2} \cdot 0 \cdot (1-0)^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + (m-1)(1-0)^{m-2} \cdot 0 \cdot 0^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle} \\
& + \frac{0^{m-1}(1-0)^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + 0^{m-1}(m-2)(1-0)^{m-3} \cdot 0 \cdot \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 0^{m-1} 0^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle}{0^{m-1}(1-0)^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + 0^{m-1}(m-2)(1-0)^{m-3} \cdot 0 \cdot \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 0^{m-1} 0^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle} \parallel_{IL} \\
& = \frac{\langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle}{\parallel \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle \parallel_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} (\Delta p_1 - \Delta p_0) \rangle}{\parallel \langle \Delta p_0 \wedge_{IL} (\Delta p_1 - \Delta p_0) \rangle \parallel_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle - \langle \Delta p_0 \wedge_{IL} \Delta p_0 \rangle}{\parallel \langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle - \langle \Delta p_0 \wedge_{IL} \Delta p_0 \rangle \parallel_{IL}} \\
& = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\parallel \langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \parallel_{IL}}
\end{aligned}$$

denklemini ile hesaplanır. Ana normal vektörü ise

$$\begin{aligned}
N(0) & = -B(0) \wedge_{IL} T(0) \\
& = -\frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\parallel \langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \parallel_{IL}} \wedge_{IL} \frac{\Delta p_0}{\parallel \Delta p_0 \parallel_{IL}} \\
& = -\frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \wedge_{IL} \Delta p_0}{\parallel \langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \parallel_{IL} \parallel \Delta p_0 \parallel_{IL}} = -\left(\frac{\langle \Delta p_0, \Delta p_0 \rangle \Delta p_1 - \langle \Delta p_1, \Delta p_0 \rangle \Delta p_0}{\parallel \langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \parallel_{IL} \parallel \Delta p_0 \parallel_{IL}} \right) \\
& = -\frac{\parallel \Delta p_0 \parallel_{IL} \parallel \Delta p_0 \parallel_{IL} \Delta p_1 \cosh \theta}{\parallel \Delta p_0 \parallel_{IL} \parallel \Delta p_0 \parallel_{IL} \parallel \Delta p_1 \parallel_{IL} \sinh \theta} + \frac{\parallel \Delta p_1 \parallel_{IL} \parallel \Delta p_0 \parallel_{IL} \Delta p_0}{\parallel \Delta p_0 \parallel_{IL} \parallel \Delta p_0 \parallel_{IL} \parallel \Delta p_1 \parallel_{IL} \sinh \theta} \\
& = \frac{\Delta p_0}{\parallel \Delta p_0 \parallel_{IL}} \coth \theta - \frac{\Delta p_1}{\parallel \Delta p_1 \parallel_{IL}} \operatorname{csch} \theta
\end{aligned}$$

denklemini ile hesaplanır. Eğriliği,

$$\begin{aligned}
\kappa(0) & = \frac{\parallel \mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0) \parallel_{IL}}{\parallel \mathfrak{B}'(0) \parallel_{IL}^3} \\
& = \frac{m-1}{m} \frac{\parallel \langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \parallel_{IL}}{\parallel \Delta p_0 \parallel_{IL}^3} = \frac{m-1}{m} \frac{\parallel \Delta p_0 \parallel_{IL} \parallel \Delta p_1 \parallel_{IL} \sinh \theta}{\parallel \Delta p_0 \parallel_{IL}^3} \\
& = \frac{m-1}{m} \frac{\parallel \Delta p_1 \parallel_{IL} \sinh \theta}{\parallel \Delta p_0 \parallel_{IL}^2}
\end{aligned}$$

denklemini ile elde edilir. Burulması,

$$\begin{aligned}
\tau(0) &= \frac{\langle \mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0), \mathfrak{B}'''(0) \rangle}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}} \\
&= -\frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(0) B_i^{m-2}(0) B_k^{m-3}(0) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2} \\
&= -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta^2 p_1, \Delta^3 p_2)}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}^2}
\end{aligned}$$

denklemini ile elde edilir. Buradaki θ açısı Δp_1 ve Δp_0 vektörleri arasındaki açıdır.

Sonuç 3.2 $\mathfrak{B}(t)$ timelike Bézier eğrisi ve p_j 'ler kontrol noktası olsun. $\mathfrak{B}(t)$ nin $t = 1$ 'daki Serret-Frenet çerçevesi T, N, B , eğriliği κ ve burulması τ :

$$T(1) = \frac{\Delta p_{m-1}}{\sqrt{-\langle \Delta p_{m-1}, \Delta p_{m-1} \rangle}} \quad (3.15)$$

$$N(1) = \frac{\Delta p_{m-2}}{\|\Delta p_{m-2}\|_{IL}} \operatorname{csch} \theta - \frac{\Delta p_{m-1}}{\|\Delta p_{m-1}\|_{IL}} \operatorname{coth} \theta \quad (3.16)$$

$$B(1) = -\frac{\Delta p_{m-1} \wedge_{IL} \Delta p_{m-2}}{\|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-2}\|_{IL} \sinh \theta} \quad (3.17)$$

$$\kappa(1) = \frac{m-1}{m} \frac{\|\Delta p_{m-2}\|_{IL} \sinh \theta}{\|\Delta p_{m-1}\|_{IL}^2} \quad (3.18)$$

$$\tau(1) = -\frac{m-2}{m} \frac{\det(\Delta p_{m-1}, \Delta p_{m-2}, \Delta p_{m-3})}{\|\Delta p_{m-1} \wedge_{IL} \Delta p_{m-2}\|_{IL}^2} \quad (3.19)$$

ile verilir ve burada θ , Δp_{m-2} ve Δp_{m-1} arasındaki açıdır [78, 81].

İspat: Eş. (2.28) Bernstein polinomunun tanımı yardımıyla $\mathfrak{B}(t)$ 'nin $t = 1$ noktasındaki Serret-Frenet çatısı T, N, B , eğriliği κ ve burulması τ aşağıdaki gibi elde edilir. Teğet vektörü;

$$\begin{aligned}
T(1) &= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(1) \Delta p_j}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(1) \Delta p_j \right\|_{IL}} \\
&= \frac{B_0^{m-1}(1) \Delta p_0 + B_1^{m-1}(1) \Delta p_1 + \dots + B_{m-2}^{m-1}(1) \Delta p_{m-2} + B_{m-1}^{m-1}(1) \Delta p_{m-1}}{\left\| B_0^{m-1}(1) \Delta p_0 + B_1^{m-1}(1) \Delta p_1 + \dots + B_{m-2}^{m-1}(1) \Delta p_{m-2} + B_{m-1}^{m-1}(1) \Delta p_{m-1} \right\|_{IL}}
\end{aligned}$$

$$\begin{aligned}
&= \frac{(1-1)^{m-1}\Delta p_0 + (m-1)(1-1)^2.1\Delta p_1 + \dots + (m-1)(1-1).1^2\Delta p_{m-2} + 1^{m-1}\Delta p_{m-1}}{\|(1-1)^{m-1}\Delta p_0 + (m-1)(1-1)^2.1\Delta p_1 + \dots + (m-1)(1-1).1^2\Delta p_{m-2} + 1^{m-1}\Delta p_{m-1}\|_{IL}} \\
&= \frac{\Delta p_{m-1}}{\|\Delta p_{m-1}\|_{IL}} = \frac{\Delta p_{m-1}}{\sqrt{-\langle \Delta p_{m-1}, \Delta p_{m-1} \rangle}}
\end{aligned}$$

eşitliği ile ifade edilir. Binormal vektörü;

$$\begin{aligned}
B(1) &= \frac{\mathfrak{B}'(1) \wedge_{IL} \mathfrak{B}''(1)}{\|\mathfrak{B}'(1) \wedge_{IL} \mathfrak{B}''(1)\|_{IL}} \\
&= \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(1) B_i^{m-2}(1) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(1) B_i^{m-2}(1) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\
&= \frac{(B_0^{m-1}(1) + B_1^{m-1}(1) + \dots + B_{m-1}^{m-1}(1)) (B_0^{m-2}(1) + B_1^{m-2}(1) + \dots + B_{m-2}^{m-2}(1)) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\|(B_0^{m-1}(1) + B_1^{m-1}(1) + \dots + B_{m-1}^{m-1}(1)) (B_0^{m-2}(1) + B_1^{m-2}(1) + \dots + B_{m-2}^{m-2}(1)) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle\|_{IL}} \\
&= \frac{B_0^{m-1}(1) B_0^{m-2}(1) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + B_0^{m-1}(1) B_1^{m-2}(1) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_0^{m-1}(1) B_{m-2}^{m-2}(1) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{\|B_0^{m-1}(1) B_0^{m-2}(1) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + B_0^{m-1}(1) B_1^{m-2}(1) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_0^{m-1}(1) B_{m-2}^{m-2}(1) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle\|_{IL}} \\
&+ \frac{B_1^{m-1}(1) B_0^{m-2}(1) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + B_1^{m-1}(1) B_{m-2}^{m-2}(1) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{B_1^{m-1}(1) B_0^{m-2}(1) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + B_1^{m-1}(1) B_{m-2}^{m-2}(1) \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle} \\
&+ \frac{B_{m-1}^{m-1}(1) B_0^{m-2}(1) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + B_{m-1}^{m-1}(1) B_1^{m-2}(1) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_{m-1}^{m-1}(1) B_{m-2}^{m-2}(1) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle}{B_{m-1}^{m-1}(1) B_0^{m-2}(1) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + B_{m-1}^{m-1}(1) B_1^{m-2}(1) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_{m-1}^{m-1}(1) B_{m-2}^{m-2}(1) \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle\|_{IL}} \\
&= \frac{(1-1)^{m-1} (1-1)^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + (1-1)^{m-1} (m-2) (1-1)^{m-3}.1 \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 1^{m-1} 1^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{\|(1-1)^{m-1} (1-1)^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + (1-1)^{m-1} (m-2) (1-1)^{m-3}.1 \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 1^{m-1} 1^{m-2} \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle\|_{IL}} \\
&+ \frac{(m-1) (1-1)^{m-2}.1 (1-1)^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + (m-1) (1-1)^{m-2}.1.1^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{(m-1) (1-0)^{m-2}.1 (1-1)^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_0 \rangle + \dots + (m-1) (1-1)^{m-2}.1.1^{m-2} \langle \Delta p_1 \wedge_{IL} \Delta^2 p_{m-2} \rangle} \\
&+ \frac{1^{m-1} (1-1)^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + 1^{m-1} (m-2) (1-1)^{m-3}.1 \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 1^{m-1} 1^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle}{1^{m-1} (1-1)^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_0 \rangle + 1^{m-1} (m-2) (1-1)^{m-3}.1 \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_1 \rangle + \dots + 1^{m-1} 1^{m-2} \langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle\|_{IL}} \\
&= \frac{\langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle}{\|\langle \Delta p_{m-1} \wedge_{IL} \Delta^2 p_{m-2} \rangle\|_{IL}} = \frac{\langle \Delta p_{m-1} \wedge_{IL} (\Delta p_{m-1} - \Delta p_{m-2}) \rangle}{\|\langle \Delta p_{m-1} \wedge_{IL} (\Delta p_{m-1} - \Delta p_{m-2}) \rangle\|_{IL}} = \frac{\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-1} \rangle - \langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle}{\|\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-1} \rangle - \langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle\|_{IL}} \\
&= \frac{\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle}{\|\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle\|_{IL}} = - \frac{\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle}{\|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-2}\|_{IL} \sinh \theta}
\end{aligned}$$

eşitliği ile ifade edilir. Ana normal vektörü;

$$N(1) = -B(1) \wedge_{IL} T(1)$$

$$\begin{aligned}
&= -\frac{\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle}{\|\Delta p_{m-1} \wedge_{IL} \Delta p_{m-2}\|_{IL}} \wedge_{IL} \frac{\Delta p_{m-1}}{\|\Delta p_{m-1}\|_{IL}} = -\frac{\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle \wedge_{IL} \Delta p_{m-1}}{\|\Delta p_{m-1} \wedge_{IL} \Delta p_{m-2}\|_{IL} \|\Delta p_{m-1}\|_{IL}} \\
&= \frac{\langle \Delta p_{m-1}, \Delta p_{m-1} \rangle \Delta p_{m-2} - \langle \Delta p_{m-2}, \Delta p_{m-1} \rangle \Delta p_{m-1}}{\|\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle\|_{IL} \|\Delta p_{m-1}\|_{IL}} \\
&= \frac{\|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-1}\|_{IL} \Delta p_{m-2}}{\|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-2}\|_{IL} \sinh \theta} - \frac{\|\Delta p_{m-2}\|_{IL} \|\Delta p_{m-1}\|_{IL} \Delta p_{m-1} \cosh \theta}{\|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-2}\|_{IL} \sinh \theta} \\
&= \frac{\Delta p_{m-2}}{\|\Delta p_{m-2}\|_{IL}} \operatorname{csch} \theta - \frac{\Delta p_{m-1}}{\|\Delta p_{m-1}\|_{IL}} \coth \theta
\end{aligned}$$

eşitliği ile ifade edilir. Eğriliği;

$$\begin{aligned}
\kappa(1) &= \frac{\|\mathfrak{B}'(1) \wedge_{IL} \mathfrak{B}''(1)\|_{IL}}{\|\mathfrak{B}'(1)\|_{IL}^3} = \frac{m^2(m-1)}{|m|^3} \frac{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(1) B_i^{m-2}(1) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(1) \Delta p_j \right\|_{IL}^3} \\
&= \frac{m-1}{m} \frac{\|\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle\|_{IL}}{\|\Delta p_{m-1}\|_{IL}^3} = \frac{m-1}{m} \frac{\|\Delta p_{m-1}\|_{IL} \|\Delta p_{m-2}\|_{IL} \sinh \theta}{\|\Delta p_{m-1}\|_{IL}^3} \\
&= \frac{m-1}{m} \frac{\|\Delta p_{m-2}\|_{IL} \sinh \theta}{\|\Delta p_{m-1}\|_{IL}^2}
\end{aligned}$$

eşitliği ile ifade edilir ve son olarak burulması;

$$\begin{aligned}
\tau(1) &= \frac{\langle \mathfrak{B}'(1) \wedge_{IL} \mathfrak{B}''(1), \mathfrak{B}''(1) \rangle}{\|\mathfrak{B}'(1) \wedge_{IL} \mathfrak{B}''(1)\|_{IL}} \\
&= \frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(1) B_i^{m-2}(1) B_k^{m-3}(1) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(1) B_i^{m-2}(1) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2} \\
&= \frac{m-2}{m} \frac{\det(\Delta p_{m-1}, \Delta^2 p_{m-2}, \Delta^3 p_{m-3})}{\|\langle \Delta p_{m-1} \wedge_{IL} \Delta p_{m-2} \rangle\|_{IL}^2}
\end{aligned}$$

eşitliği ile ifade edilir ve burada Δp_{m-2} ve Δp_{m-1} arasındaki açı θ açısıdır.

3.2. SPACELIKE BÉZIER EĞRİLERİNİN SERRET-FRENET ÇATISI, EĞRİLİĞİ VE BURULMASI

Bu bölümde spacelike Bézier eğrilerinin Serret-Frenet çerçevesi, eğriliği ve burulması spacelike ve timelike normallerle incelenecektir.

3.2.1 Spacelike Normale Sahip Spacelike Bézier Eğrilerinin Serret-Frenet Çatısı, Eğriliği ve Burulması

$\mathfrak{B}(t)$ spacelike eğrisini ele alalım. Birim hızlı olmayan $\mathfrak{B}(t)$ spacelike Bézier eğrisinin kontrol noktaları $p_0, p_1, \dots, p_{n-1} \in R_1^3$ olsun. $\mathfrak{B}(t)$ Bézier eğrisi spacelike ise T teğet vektörü spacelike olmalıdır. Asli normalinin spacelike olduğu durumda $\mathfrak{B}(t)$ spacelike Bézier eğrisine “*asli normal spacelike olan spacelike Bézier eğrisi*” denir. T teğet vektörü spacelike, N normal vektörü spacelike, B binormal vektörü timelike olmak üzere $\mathfrak{B}(t)$ spacelike Bézier eğrisinin $t = 0$ başlangıç noktasındaki $\{T, N, B\}|_{t=0}$ ortonormal çatısı tanımlandığında

$$\left. \begin{array}{lll} \langle T, T \rangle = 1, & \langle N, N \rangle = 1, & \langle B, B \rangle = -1 \\ \langle T, N \rangle = 0, & \langle T, B \rangle = 0, & \langle N, B \rangle = 0 \end{array} \right\} \quad (3.20)$$

koşulları sağlanır. Spacelike Bézier eğrisi aynı zamanda $\langle \frac{d\mathfrak{B}^n(t)}{dt}, \frac{d\mathfrak{B}^n(t)}{dt} \rangle > 0$ koşulunu sağlar [78].

Teorem 3.3 x ve y Minkowski-3 uzayında iki vektör olsun.

- 1) Eğer x ve y spacelike vektörleri $|\langle x, y \rangle|_{IL} < \|x\|_{IL} \|y\|_{IL}$ koşulunu sağlıyorsa $x \times y$ vektörü bir spacelike vektör için

$$|\langle x, y \rangle|_{IL} = \|x\|_{IL} \|y\|_{IL} \cos \theta \quad (3.21)$$

$$\|x \times y\|_{IL} = \|x\|_{IL} \|y\|_{IL} \sin \theta \quad (3.22)$$

eşitliği sağlanır. Burada θ ; x ve y spacelike vektörleri arasındaki spacelike açıdır.

- 2) Eğer x ve y spacelike vektörleri $|\langle x, y \rangle|_{IL} > \|x\|_{IL} \|y\|_{IL}$ koşulunu sağlıyorsa $x \times y$ vektörü bir timelike vektör için

$$|\langle x, y \rangle|_{IL} = -\|x\|_{IL}\|y\|_{IL} \cosh \theta \quad (3.23)$$

$$\|x \times y\|_{IL} = \|x\|_{IL}\|y\|_{IL} \sinh \theta \quad (3.24)$$

eşitliği sağlanır. Burada θ ; x ve y spacelike vektörleri arasındaki hiperbolik açıdır.

- 3) Eğer x ve y spacelike vektörleri $|\langle x, y \rangle|_{IL} = \|x\|_{IL}\|y\|_{IL}$ eşitliğini sağlıyorsa $x \times y$ vektörü lightlike vektördür [78, 81].

Teorem 3.4 $\mathfrak{B}(t)$ spacelike normale sahip spacelike Bézier eğrisi olsun ve p_j kontrol noktaları olsun. $\mathfrak{B}(t)$ 'nin Serret-Frenet çerçevesi T, N, B , eğriliği κ ve burulması τ :

$$T(t) = \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\left(\sum_{j=0}^{m-1} B_j^{m-1}(t) B_j^{m-1}(t) \langle \Delta p_j, \Delta p_j \rangle \right)^{\frac{1}{2}}},$$

$$N(t) = -\frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-1} B_j^{m-1}(t) B_i^{m-2}(t) B_k^{m-1}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \wedge_{IL} \Delta p_k}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL} \left\| \sum_{k=0}^{m-1} B_k^{m-1}(t) \Delta p_k \right\|_{IL}},$$

$$B(t) = \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL}},$$

$$\kappa(t) = \frac{m-1}{m} \frac{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL}}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}^3},$$

$$\tau(t) = -\frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(t) B_i^{m-2}(t) B_k^{m-3}(t) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(t) B_i^{m-2}(t) (\Delta p_j \wedge_{IL} \Delta^2 p_i) \right\|_{IL}^2}$$

eşitlikleri ile verilir ve burada Δp_j aynı koni içinde yer alır [79, 82].

İspat: Δp_j vektörleri spacelike vektörler olduğundan, $t \in [0,1]$ için Δp_j 'nin normu

$$\|\Delta p_j\|_{IL} = \sqrt{\langle \Delta p_j, \Delta p_j \rangle} \quad (3.25)$$

dır. Böylece Serret-Frenet çatısının, teğet vektörü şu şekilde hesaplanır:

$$\begin{aligned}
T(t) &= \frac{\mathfrak{B}'(t)}{\|\mathfrak{B}'(t)\|_{IL}} = \frac{m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\left\| m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}} \\
&= \frac{m \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{|m| \left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}} = \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \right\|_{IL}} \\
&= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\sqrt{\langle \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j, \sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j \rangle}} \\
&= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(t) \Delta p_j}{\sqrt{\sum_{j=0}^{m-1} \sum_{j=0}^{m-1} B_j^{m-1}(t) B_j^{m-1}(t) \langle \Delta p_j, \Delta p_j \rangle}}
\end{aligned}$$

ispatın kalan kısımları Teorem 3.1 ispatı ile aynıdır.

Sonuç 3.3 $\mathfrak{B}(t)$ asli normali spacelike olan spacelike Bézier eğrisi için $p_i \in E^n$ kontrol noktaları olsun. Δb_i ler aynı konide yer alan vektörler alındığında $t = 0$ başlangıç noktasındaki Serret-Frenet çatısının teğet vektörü

$$T(0) = \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} = \frac{\Delta p_0}{\sqrt{\langle \Delta p_0, \Delta p_0 \rangle}}$$

eşitliği ile ifade edilir. Binormal vektörü ve Asli Normal vektörü iki durumda incelenir:

1. Durum: $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 vektörleri için N asal normali ve B binormalı (3.21) ve (3.22) eşitlikleri kullanılarak aşağıdaki şekilde

$$B(0) = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta}$$

$$N(0) = \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \csc \theta - \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \cot \theta$$

2. Durum: Δp_0 ve Δp_1 vektörleri için $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} > \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa N asal normali ve B binormalı (3.23) ve (3.24) eşitlikleri kullanılarak aşağıdaki şekilde

$$B(0) = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta}$$

$$N(0) = \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta - \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \operatorname{coth} \theta$$

ifade edilir [78, 81].

İspat: $\mathfrak{B}(t)$ spacelike Bézier eğrisi olduğu için T teğet vektörü de spacelike olmalıdır. Bu yüzden Δp_0 'ın normu $\|\Delta p_0\|_{IL} = \sqrt{\langle \Delta p_0, \Delta p_0 \rangle}$ ifadesine eşittir. Böylece $t = 0$ başlangıç noktasındaki teğet vektörü

$$\begin{aligned} T(0) &= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(0) \Delta p_j}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(0) \Delta p_j \right\|_{IL}} \\ &= \frac{B_0^{m-1}(0) \Delta p_0 + B_1^{m-1}(0) \Delta p_1 + \dots + B_{m-2}^{m-1}(0) \Delta p_{m-2} + B_{m-1}^{m-1}(0) \Delta p_{m-1}}{\left\| B_0^{m-1}(0) \Delta p_0 + B_1^{m-1}(0) \Delta p_1 + \dots + B_{m-2}^{m-1}(0) \Delta p_{m-2} + B_{m-1}^{m-1}(0) \Delta p_{m-1} \right\|_{IL}} \\ &= \frac{(1-0)^{m-1} \Delta p_0 + (m-1)(1-0)^2 \cdot 0 \Delta p_1 + \dots + (m-1)(1-0) \cdot 0^2 \Delta p_{m-2} + 0^{m-1} \Delta p_{m-1}}{\left\| (1-0)^{m-1} \Delta p_0 + (m-1)(1-0)^2 \cdot 0 \Delta p_1 + \dots + (m-1)(1-0) \cdot 0^2 \Delta p_{m-2} + 0^{m-1} \Delta p_{m-1} \right\|_{IL}} \\ &= \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} = \frac{\Delta p_0}{\sqrt{\langle \Delta p_0, \Delta p_0 \rangle}} \end{aligned}$$

eşitliği ile ifade edilir. $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulunu sağlayan Δp_0 ve Δp_1 vektörleri spacelike vektör ise spacelike açının iç çarpım ve vektörel (dış) çarpım özelliklerinin kullanılmasıyla birim hızlı olmayan $\mathfrak{B}(t)$ Bézier eğrisinin $t = 0$ başlangıç noktasındaki binormal vektörü:

$$\begin{aligned} B(0) &= \frac{\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}} \\ &= \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\ &= \frac{(B_0^{m-1}(0) + B_1^{m-1}(t) + \dots + B_{m-1}^{m-1}(0)) (B_0^{m-2}(0) + B_1^{m-2}(0) + \dots + B_{m-2}^{m-2}(0)) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| (B_0^{m-1}(0) + B_1^{m-1}(t) + \dots + B_{m-1}^{m-1}(0)) (B_0^{m-2}(0) + B_1^{m-2}(0) + \dots + B_{m-2}^{m-2}(0)) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\ &= \frac{B_0^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + B_0^{m-1}(0) B_1^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_0^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle}{\left\| B_0^{m-1}(0) B_0^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle + B_0^{m-1}(0) B_1^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_1 \rangle + \dots + B_0^{m-1}(0) B_{m-2}^{m-2}(0) \langle \Delta p_0 \wedge_{IL} \Delta^2 p_{m-2} \rangle \right\|_{IL}} \end{aligned}$$

$$\begin{aligned}
& + \frac{B_1^{m-1}(0)B_0^{m-2}(0)\langle\Delta p_1\wedge_{IL}\Delta^2 p_0\rangle+\dots+B_1^{m-1}(0)B_{m-2}^{m-2}(0)\langle\Delta p_1\wedge_{IL}\Delta^2 p_{m-2}\rangle}{B_1^{m-1}(0)B_0^{m-2}(0)\langle\Delta p_1\wedge_{IL}\Delta^2 p_0\rangle+\dots+B_1^{m-1}(0)B_{m-2}^{m-2}(0)\langle\Delta p_1\wedge_{IL}\Delta^2 p_{m-2}\rangle} \\
& + \frac{B_{m-1}^{m-1}(0)B_0^{m-2}(0)\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_0\rangle+B_{m-1}^{m-1}(0)B_1^{m-2}(0)\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_1\rangle+\dots+B_{m-1}^{m-1}(0)B_{m-2}^{m-2}(0)\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_{m-2}\rangle}{B_{m-1}^{m-1}(0)B_0^{m-2}(0)\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_0\rangle+B_{m-1}^{m-1}(0)B_1^{m-2}(0)\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_1\rangle+\dots+B_{m-1}^{m-1}(0)B_{m-2}^{m-2}(0)\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_{m-2}\rangle}\|_{IL} \\
& = \frac{(1-0)^{m-1}(1-0)^{m-2}\langle\Delta p_0\wedge_{IL}\Delta^2 p_0\rangle+(1-0)^{m-1}(m-2)(1-0)^{m-3}.0\langle\Delta p_0\wedge_{IL}\Delta^2 p_1\rangle+\dots+0^{m-1}0^{m-2}\langle\Delta p_0\wedge_{IL}\Delta^2 p_{m-2}\rangle}{\|(1-0)^{m-1}(1-0)^{m-2}\langle\Delta p_0\wedge_{IL}\Delta^2 p_0\rangle+(1-0)^{m-1}(m-2)(1-0)^{m-3}.0\langle\Delta p_0\wedge_{IL}\Delta^2 p_1\rangle+\dots+0^{m-1}0^{m-2}\langle\Delta p_0\wedge_{IL}\Delta^2 p_{m-2}\rangle} \\
& + \frac{(m-1)(1-0)^{m-2}.0.(1-0)^{m-2}\langle\Delta p_1\wedge_{IL}\Delta^2 p_0\rangle+\dots+(m-1)(1-0)^{m-2}.0.0^{m-2}\langle\Delta p_1\wedge_{IL}\Delta^2 p_{m-2}\rangle}{(m-1)(1-0)^{m-2}.0.(1-0)^{m-2}\langle\Delta p_1\wedge_{IL}\Delta^2 p_0\rangle+\dots+(m-1)(1-0)^{m-2}.0.0^{m-2}\langle\Delta p_1\wedge_{IL}\Delta^2 p_{m-2}\rangle} \\
& + \frac{0^{m-1}(1-0)^{m-2}\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_0\rangle+0^{m-1}(m-2)(1-0)^{m-3}.0.\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_1\rangle+\dots+0^{m-1}0^{m-2}\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_{m-2}\rangle}{0^{m-1}(1-0)^{m-2}\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_0\rangle+0^{m-1}(m-2)(1-0)^{m-3}.0.\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_1\rangle+\dots+0^{m-1}0^{m-2}\langle\Delta p_{m-1}\wedge_{IL}\Delta^2 p_{m-2}\rangle}\|_{IL} \\
& = \frac{\langle\Delta p_0\wedge_{IL}\Delta^2 p_0\rangle}{\|\langle\Delta p_0\wedge_{IL}\Delta^2 p_0\rangle\|_{IL}} = \frac{\langle\Delta p_0\wedge_{IL}(\Delta p_1-\Delta p_0)\rangle}{\|\langle\Delta p_0\wedge_{IL}(\Delta p_1-\Delta p_0)\rangle\|_{IL}} = \frac{\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle-\langle\Delta p_0\wedge_{IL}\Delta p_0\rangle}{\|\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle-\langle\Delta p_0\wedge_{IL}\Delta p_0\rangle\|_{IL}} \\
& = \frac{\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle}{\|\Delta p_0\wedge_{IL}\Delta p_1\|_{IL}} = \frac{\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle}{\|\Delta p_0\|_{IL}\|\Delta p_1\|_{IL}\sin\theta}
\end{aligned}$$

eşitliği ile hesaplanır. T teğet vektörü ve N ana normal vektörü spacelike, B binormal vektörü timelike ise $N = -B \wedge_{IL} T$ alınacaktır. Burada Δp_0 spacelike vektör olduğu için (3.25) denklemden $\|\Delta p_0\|_{IL}^2 = \langle\Delta p_0, \Delta p_0\rangle$ dır. $\mathfrak{B}(t)$ Bézier eğrisinin $t = 0$ başlangıç noktasındaki ana normal vektörü;

$$N(0) = -B(0) \wedge_{IL} T(0)$$

$$\begin{aligned}
& = -\frac{\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle}{\|\Delta p_0\wedge_{IL}\Delta p_1\|_{IL}} \wedge_{IL} \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \\
& = -\frac{\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle\wedge_{IL}\Delta p_0}{\|\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle\|_{IL}\|\Delta p_0\|_{IL}} = -\left(\frac{-\langle\Delta p_0, \Delta p_0\rangle\Delta p_1 + \langle\Delta p_1, \Delta p_0\rangle\Delta p_0}{\|\langle\Delta p_0\wedge_{IL}\Delta p_1\rangle\|_{IL}\|\Delta p_0\|_{IL}}\right) \\
& = \frac{\|\Delta p_0\|_{IL}\|\Delta p_0\|_{IL}\Delta p_1 \cos\theta}{\|\Delta p_0\|_{IL}\|\Delta p_0\|_{IL}\|\Delta p_1\|_{IL}\sin\theta} - \frac{\|\Delta p_1\|_{IL}\|\Delta p_0\|_{IL}\Delta p_0}{\|\Delta p_0\|_{IL}\|\Delta p_0\|_{IL}\|\Delta p_1\|_{IL}\sin\theta} \\
& = \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \csc\theta - \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \cot\theta
\end{aligned}$$

eşitliği ile ifade edilir. Burada Δp_0 ve Δp_1 spacelike vektörleri arasındaki açı θ dır.

$|\langle\Delta p_0, \Delta p_1\rangle|_{IL} > \|\Delta p_0\|_{IL}\|\Delta p_1\|_{IL}$ koşulunu sağlayan Δp_0 ve Δp_1 vektörleri spacelike vektör

ise hiperbolik açının iç çarpım vektörel çarpım özelliklerinin kullanılmasıyla $\mathfrak{B}(t)$ birim hızlı olmayan Bézier eğrisinin $t = 0$ başlangıç noktasındaki binormal vektörü:

$$\begin{aligned}
B(0) &= \frac{\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}} \\
&= \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\
&= \frac{\langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle}{\|\langle \Delta p_0 \wedge_{IL} \Delta^2 p_0 \rangle\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} (\Delta p_1 - \Delta p_0) \rangle}{\|\langle \Delta p_0 \wedge_{IL} (\Delta p_1 - \Delta p_0) \rangle\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle - \langle \Delta p_0 \wedge_{IL} \Delta p_0 \rangle}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle - \langle \Delta p_0 \wedge_{IL} \Delta p_0 \rangle\|_{IL}} \\
&= \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta}
\end{aligned}$$

eşitliği ile ifade edilir. T teğet vektörü ve N ana normal vektörü spacelike, B binormal vektörü timelike ise $N(t) = -B(t) \wedge_{IL} T(t)$ alınacaktır. Burada Δp_0 spacelike vektör olduğu için $\|\Delta p_0\|_{IL}^2 = \langle \Delta p_0, \Delta p_0 \rangle$ dır. $\mathfrak{B}(t)$ birim hızlı olmayan Bézier eğrisinin $t = 0$ başlangıç noktasındaki nna normal vektörü:

$$\begin{aligned}
N(0) &= -B(0) \wedge_{IL} T(0) \\
&= -\frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} \wedge_{IL} \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \\
&= -\frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \wedge_{IL} \Delta p_0}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL} \|\Delta p_0\|_{IL}} = -\left(\frac{-\langle \Delta p_0, \Delta p_0 \rangle \Delta p_1 + \langle \Delta p_1, \Delta p_0 \rangle \Delta p_0}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL} \|\Delta p_0\|_{IL}} \right) \\
&= \frac{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \Delta p_1 \cosh \theta}{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta} - \frac{\|\Delta p_1\|_{IL} \|\Delta p_0\|_{IL} \Delta p_0}{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta} \\
&= \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta - \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \coth \theta
\end{aligned}$$

olarak hesaplanır. Burada θ , Δp_0 ve Δp_1 spacelike vektörleri arasındaki hiperbolik açıdır.

Sonuç 3.4 Kontrol noktaları spacelike vektörler olan $p_0, p_1, \dots, p_n$ olan ve Δp_i spacelike vektörlerden oluşan n . dereceden asli normali spacelike olan spacelike $\mathfrak{B}(t)$ Bézier eğrisinin $t = 0$ başlangıç noktasında eğrilik ve burulması iki durum için aşağıdaki gibi hesaplanır:

1. Durum: $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 spacelike vektörleri için

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL} \sin \theta}{\|\Delta p_0\|_{IL}^2} \text{ ve } \tau(0) = -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}^2}$$

2. Durum: $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} > \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 spacelike vektörleri için

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL} \sinh \theta}{\|\Delta p_0\|_{IL}^2} \text{ ve } \tau(0) = -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}^2}$$

olarak bulunur [78, 81].

İspat: Birinci durumda olduğu gibi $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 spacelike vektörleri için $\mathfrak{B}(t)$ spacelike Bézier eğrisinin $t = 0$ başlangıç noktasındaki eğriliği κ ve burulması τ , (3.21) ve (3.22) eşitlikleri yardımıyla

$$\begin{aligned} \kappa(0) &= \frac{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}}{\|\mathfrak{B}'(0)\|_{IL}^3} = \frac{m-1}{m} \frac{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}}{\|\Delta p_0\|_{IL}^3} \\ &= \frac{m-1}{m} \frac{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta}{\|\Delta p_0\|_{IL}^3} = -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}^2} \\ &= \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL} \sin \theta}{\|\Delta p_0\|_{IL}^2}, \\ \tau(0) &= \frac{\langle \mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0), \mathfrak{B}'''(0) \rangle}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}} \\ &= -\frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(0) B_i^{m-2}(0) B_k^{m-3}(0) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2} \\ &= -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta^2 p_1, \Delta^3 p_2)}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}^2} \end{aligned}$$

ifadesi ile elde edilir. İkinci durumda olduğu gibi $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} > \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 spacelike vektörleri için spacelike vektörler için $\mathfrak{B}(t)$ spacelike

Bézier eğrisinin $t = 0$ başlangıç noktasındaki eğriliği κ ve burulması τ ise sırasıyla (3.23) ve (3.24) eşitlikleri kullanılarak

$$\begin{aligned}\kappa(0) &= \frac{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}}{\|\mathfrak{B}'(0)\|_{IL}^3} = \frac{m-1}{m} \frac{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}}{\|\Delta p_0\|_{IL}^3} \\ &= \frac{m-1}{m} \frac{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta}{\|\Delta p_0\|_{IL}^3} = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL} \sinh \theta}{\|\Delta p_0\|_{IL}^2},\end{aligned}$$

$$\begin{aligned}\tau(0) &= \frac{\langle \mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0), \mathfrak{B}'''(0) \rangle}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}} \\ &= -\frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(0) B_i^{m-2}(0) B_k^{m-3}(0) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2} \\ &= -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta^2 p_1, \Delta^3 p_2)}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}^2}\end{aligned}$$

ifadeleri elde edilir.

Sonuç 3.5 Ana normal spacelike olan $\mathfrak{B}(t)$ spacelike Bézier eğrisi için $b_i \in E^3$ kontrol noktaları olsun. Aynı konide yer alan Δb_i vektörleri alındığında $t = 0$ başlangıç noktasındaki Serret-Frenet çatısının teğet vektörü

$$T(0) = \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} = \frac{\Delta p_0}{\sqrt{\langle \Delta p_0, \Delta p_0 \rangle}}$$

dir. Binormal vektörü ve Asli Normal vektörü iki durumda incelenir:

1. Durum: $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 vektörleri için N asal normal ve B binormali (3.21) ve (3.22) eşitlikleri yardımıyla;

$$B(0) = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta}$$

$$N(0) = -\frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \csc \theta + \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \cot \theta$$

2. Durum: $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} > \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 vektörleri için N asal normali ve B binormali (3.23) ve (3.24) eşitlikleri yardımıyla;

$$B(0) = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta}$$

$$N(0) = -\frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta - \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \coth \theta$$

ifadeleri ile hesaplanır [78, 81].

İspat: $\mathfrak{B}(t)$ spacelike Bézier eğrisi olduğu için teğet vektörü de spacelike olmalıdır. Bu yüzden $\|\Delta p_0\|_{IL} = \sqrt{\langle \Delta p_0, \Delta p_0 \rangle}$ olur. $t = 0$ başlangıç noktasındaki teğet vektörü

$$\begin{aligned} T(0) &= \frac{\sum_{j=0}^{m-1} B_j^{m-1}(0) \Delta p_j}{\left\| \sum_{j=0}^{m-1} B_j^{m-1}(0) \Delta p_j \right\|_{IL}} \\ &= \frac{B_0^{m-1}(0) \Delta p_0 + B_1^{m-1}(0) \Delta p_1 + \dots + B_{m-2}^{m-1}(0) \Delta p_{m-2} + B_{m-1}^{m-1}(0) \Delta p_{m-1}}{\left\| B_0^{m-1}(0) \Delta p_0 + B_1^{m-1}(0) \Delta p_1 + \dots + B_{m-2}^{m-1}(0) \Delta p_{m-2} + B_{m-1}^{m-1}(0) \Delta p_{m-1} \right\|_{IL}} \\ &= \frac{(1-0)^{m-1} \Delta p_0 + (m-1)(1-0)^2 \cdot 0 \Delta p_1 + \dots + (m-1)(1-0) \cdot 0^2 \Delta p_{m-2} + 0^{m-1} \Delta p_{m-1}}{\left\| (1-0)^{m-1} \Delta p_0 + (m-1)(1-0)^2 \cdot 0 \Delta p_1 + \dots + (m-1)(1-0) \cdot 0^2 \Delta p_{m-2} + 0^{m-1} \Delta p_{m-1} \right\|_{IL}} \\ &= \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} = \frac{\Delta p_0}{\sqrt{\langle \Delta p_0, \Delta p_0 \rangle}} \end{aligned}$$

eşitliği ile elde edilir. $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulunu sağlayan Δp_0 ve Δp_1 vektörleri spacelike vektör ise spacelike açının iç çarpım ve vektörel çarpım özelliklerinin kullanılmasıyla birim hızlı olmayan $\mathfrak{B}(t)$ Bézier eğrisinin $t = 0$ başlangıç noktasındaki binormal vektörü (3.21) ve (3.22) denklemleri yardımıyla

$$\begin{aligned} B(0) &= \frac{\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}} \\ &= \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta} \end{aligned}$$

olduğu kolayca görülür. T teğet vektörü ve N ana normal vektörü spacelike, B binormal vektörü timelike ise $N = B \wedge_{IL} T$ alınacaktır. Burada Δp_0 spacelike vektör olduğu için $\|\Delta p_0\|_{IL}^2 = \langle \Delta p_0, \Delta p_0 \rangle$ dır. Asli normal vektörü (3.21) ve (3.22) denklemleri yardımıyla;

$$\begin{aligned}
N(0) &= B(0) \wedge_{IL} T(0) = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} \wedge_{IL} \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \\
&= \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle \wedge_{IL} \Delta p_0}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL} \|\Delta p_0\|_{IL}} = \left(\frac{-\langle \Delta p_0, \Delta p_0 \rangle \Delta p_1 + \langle \Delta p_1, \Delta p_0 \rangle \Delta p_0}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL} \|\Delta p_0\|_{IL}} \right) \\
&= \frac{-\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \Delta p_1}{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta} + \frac{\|\Delta p_1\|_{IL} \|\Delta p_0\|_{IL} \Delta p_0}{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta} \\
&= -\frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \csc \theta + \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \cot \theta
\end{aligned}$$

olduğu görülür. Burada θ , Δp_0 ve Δp_1 spacelike vektörleri arasındaki açıdır.

$|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} > \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulunu sağlayan Δp_0 ve Δp_1 vektörleri spacelike vektör ise hiperbolik açının iç çarpım ve vektörel çarpım özelliklerinin kullanılmasıyla birim hızlı olmayan $\mathfrak{B}(t)$ Bézier eğrisinin $t = 0$ başlangıç noktasındaki binormal vektörü (3.23) ve (3.24) denklemleri yardımıyla:

$$\begin{aligned}
B(0) &= \frac{\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}} \\
&= \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}} \\
&= \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta}
\end{aligned}$$

eşitliği elde edilir. T teğet vektörü spacelike, N ana normal vektörü timelike, B binormal vektörü spacelike ise $N(t) = B(t) \wedge_{IL} T(t)$ alınacaktır. Burada Δp_0 spacelike vektör olduğu için $\|\Delta p_0\|_{IL}^2 = \langle \Delta p_0, \Delta p_0 \rangle$ dır. Ana normal vektörü (3.23) ve (3.24) denklemleri yardımıyla:

$$\begin{aligned}
N(0) &= B(0) \wedge_{IL} T(0) \\
&= \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} \wedge_{IL} \frac{\Delta p_0}{\|\Delta p_0\|_{IL}}
\end{aligned}$$

$$\begin{aligned}
&= \frac{\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle_{IL} \Delta p_0}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle_{IL}\|_{IL} \|\Delta p_0\|_{IL}} = \frac{-\langle \Delta p_0, \Delta p_0 \rangle \Delta p_1 + \langle \Delta p_1, \Delta p_0 \rangle \Delta p_0}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle_{IL}\|_{IL} \|\Delta p_0\|_{IL}} \\
&= -\frac{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \Delta p_1}{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta} + \frac{\|\Delta p_1\|_{IL} \|\Delta p_0\|_{IL} \Delta p_0 \cosh \theta}{\|\Delta p_0\|_{IL} \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta} \\
&= -\frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta + \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \coth \theta
\end{aligned}$$

olarak hesaplanır. Burada θ , Δp_0 ve Δp_1 spacelike vektörleri arasındaki hiperbolik açıdır.

Sonuç 3.6 Kontrol noktaları spacelike vektörler $(p_0, p_1, \dots, p_n)$ olan ve Δp_i spacelike vektörlerden oluşan n . dereceden asli normal timelike olan spacelike $\mathfrak{B}(t)$ Bézier eğrisinin $t = 0$ başlangıç noktasında eğriliği (κ) ve burulması (τ) iki durum için aşağıdaki gibi hesaplanır:

1. Durum: $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulunu sağlayan Δp_0 ve Δp_1 spacelike vektörleri için

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL}}{\|\Delta p_0\|_{IL}^2} \sin \theta \quad \text{ve} \quad \tau(0) = -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}^2}$$

2. Durum: $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} > \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulunu sağlayan Δp_0 ve Δp_1 spacelike vektörleri için

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL}}{\|\Delta p_0\|_{IL}^2} \sinh \theta \quad \text{ve} \quad \tau(0) = -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}^2}$$

olarak bulunur [78, 81].

İspat: Birinci durumda olduğu gibi $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} < \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulu sağlanıyorsa Δp_0 ve Δp_1 spacelike vektörleri için $\mathfrak{B}(t)$ spacelike Bézier eğrisinin $t = 0$ başlangıç noktasındaki eğriliği (κ) ve burulması (τ) sırasıyla (3.21) ve (3.22) denklemleri yardımıyla

$$\begin{aligned}
\kappa(0) &= \frac{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}}{\|\mathfrak{B}'(0)\|_{IL}^3} = \frac{m-1}{m} \frac{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle_{IL}\|_{IL}}{\|\Delta p_0\|_{IL}^3} \\
&= \frac{m-1}{m} \frac{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta}{\|\Delta p_0\|_{IL}^3}
\end{aligned}$$

$$= \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL} \sin \theta}{\|\Delta p_0\|_{IL}^2},$$

$$\tau(0) = \frac{\langle \mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0), \mathfrak{B}'''(0) \rangle}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}}$$

$$= -\frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(0) B_i^{m-2}(0) B_k^{m-3}(0) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2}$$

$$= -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta^2 p_1, \Delta^3 p_2)}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}^2} = -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}^2}$$

ifadeleri ile elde edilir. İkinci durumda olduğu gibi $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} > \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL}$ koşulunu sağlayan Δp_0 ve Δp_1 spacelike vektörleri için $\mathfrak{B}(t)$ spacelike Bézier eğrisinin $t = 0$ başlangıç noktasındaki eğriliği $\kappa(0)$ ve burulması $\tau(0)$ sırasıyla (3.23) ve (3.24) denklemleri yardımıyla

$$\kappa(0) = \frac{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}}{\|\mathfrak{B}'(0)\|_{IL}^3}$$

$$= \frac{m-1}{m} \frac{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}}{\|\Delta p_0\|_{IL}^3} = \frac{m-1}{m} \frac{\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta}{\|\Delta p_0\|_{IL}^3}$$

$$= \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL}}{\|\Delta p_0\|_{IL}^2} \sinh \theta,$$

$$\tau(0) = \frac{\langle \mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0), \mathfrak{B}'''(0) \rangle}{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}}$$

$$= -\frac{m-2}{m} \frac{\sum_{j=0}^{m-1} \sum_{i=0}^{m-2} \sum_{k=0}^{m-3} B_j^{m-1}(0) B_i^{m-2}(0) B_k^{m-3}(0) \det(\Delta p_j, \Delta^2 p_i, \Delta^3 p_k)}{\left\| \sum_{j=0}^{m-1} \sum_{i=0}^{m-2} B_j^{m-1}(0) B_i^{m-2}(0) \langle \Delta p_j \wedge_{IL} \Delta^2 p_i \rangle \right\|_{IL}^2}$$

$$= -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta^2 p_1, \Delta^3 p_2)}{\|\langle \Delta p_0 \wedge_{IL} \Delta p_1 \rangle\|_{IL}^2}$$

eşitlikleri elde edilir. ■

BÖLÜM 4

$\mathbb{E}^3$ 'te TIMELIKE VE SPACELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET-FRENET ELEMANLARINA İLİŞKİN UYGULAMALARI

4.1 $\mathbb{E}^3$ 'TE ANA NORMALİ TIMELIKE OLAN TIMELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET FRENET ELEMANLARI UYGULAMALARI

Örnek 4.1 $\mathfrak{B}(t)$ kontrol noktaları $p_0 = (1,3,-3)$, $p_1 = (3,0,4)$, $p_2 = (2,1,6)$, $p_3 = (2,5,5)$ olan timelike kübik Bézier eğrisi olsun. O halde timelike konveks çokgeni,

$$\Delta p_0 = p_1 - p_0 = (2, -3, 7),$$

$$\Delta p_1 = p_2 - p_1 = (-1, 1, 2),$$

$$\Delta p_2 = p_3 - p_2 = (0, 3, 1)$$

vektörleri ile verilir. Buradaki vektörler $\langle \Delta p_i, \Delta p_i \rangle < 0$ koşulunu sağlayan her bir vektörden dolayı timelike vektördür. Böylece $\mathfrak{B}(t)$ timelike kübik Bézier eğrisi olduğu gerçekleşmiş olur. Bu örnekte asli normal timelike olan timelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Serret-Frenet çatısı, eğriliği ve burulması aşağıdaki gibi hesaplanır.

Sonuç (2.3)'daki (2.49) denkleminden yararlanarak

$$\mathfrak{B}(t) = \sum_{i=0}^3 B_i^3(t) p_i$$

ifadesinin $t = 0$ başlangıç noktasındaki birinci ikinci ve üçüncü türevleri

$$\mathfrak{B}'(0) = 3 \sum_{i=0}^2 B_i^2(0) \Delta p_i = 3 \cdot \Delta p_0 = 3(2, -3, 7) = (6, -9, 21)$$

$$\mathfrak{B}''(0) = 3 \cdot 2 \sum_{i=0}^1 B_i^1(0) \Delta^2 p_i = 3 \cdot 2 \cdot \Delta^2 p_0 = 3 \cdot 2 \cdot (\Delta p_1 - \Delta p_0)$$

$$= 3.2[(-1,1,2) - (2,-3,7)] = 3.2.(-3,4,-5) = (-18,24,-30)$$

$$\mathfrak{B}'''(0) = 3.2.1 \sum_{i=0}^0 B_i^0(0) \Delta^3 p_i = 3.2.1 \Delta^3 p_0 = 3.2.1(\Delta p_2 - 2\Delta p_1 + \Delta p_0)$$

$$= 3.2.1[(0,3,1) - 2(-1,1,2) + (2,-3,7)] = 6(4,-2,4) = (24,-12,24)$$

şeklinde hesaplanır. Burada Δp_0 'ın normu,

$$\|\Delta p_0\|_{IL} = \sqrt{-\langle \Delta p_0, \Delta p_0 \rangle} = \sqrt{-(2^2 + (-3)^2 - 7^2)} = \sqrt{-(-36)} = \sqrt{36} = 6$$

ile hesaplanır. Ana normali timelike olan timelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki teğet vektörü (3.10) denkleminde yararlanarak

$$T(0) = \frac{\mathfrak{B}'(0)}{\|\mathfrak{B}'(0)\|_{IL}} = \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} = \frac{(2,-3,7)}{\sqrt{36}} = \left(\frac{2}{6}, \frac{-3}{6}, \frac{7}{6}\right)$$

olarak elde edilir. Burada,

$$\langle T(0), T(0) \rangle = \left(\frac{2}{6}\right)^2 + \left(\frac{-3}{6}\right)^2 - \left(\frac{7}{6}\right)^2 = \frac{4}{36} + \frac{9}{36} - \frac{49}{36} = -1$$

olduğundan dolayı T teğet vektörü timelike vektördür. Δp_1 ve Δp_0 arasındaki θ açısını hesaplamak için $|\langle \Delta p_0, \Delta p_1 \rangle|_{IL} = -\|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \cosh \theta$ ifadesi kullanılacaktır. Buradan,

$$|\langle (2,-3,7), (-1,1,2) \rangle|_{IL} = -\|(2,-3,7)\|_{IL} \|(-1,1,2)\|_{IL} \cosh \theta$$

$$\Rightarrow |-2 - 3 - 14|_{IL} = -\sqrt{-(2^2 + (-3)^2 - 7^2)} \sqrt{-((-1)^2 + 1^2 - 2^2)} \cosh \theta$$

$$\Rightarrow |19|_{IL} = -\sqrt{36} \sqrt{2} \cosh \theta$$

$$\Rightarrow 19 = -6\sqrt{2} \cosh \theta$$

$$\Rightarrow \cosh \theta = \frac{19}{6\sqrt{2}}$$

olarak bulunur. Vektörel çarpımı, (2.3.5) denklemden

$$\Delta p_0 \wedge_{IL} \Delta p_1 = (7 \cdot (-1) - (-3) \cdot 2, 2 \cdot 2 - 7 \cdot (-1), 2 \cdot 1 - (-3) \cdot (-1)) = (13, 11, -1)$$

olarak hesaplanır. Ayrıca vektörel çarpımın normu,

$$\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL} = \sqrt{13^2 + 11^2 - (-1)^2} = \sqrt{289} = 17$$

dir. Buradan $\|\langle \Delta p_0, \Delta p_1 \rangle\|_{IL} = \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta$ ifadesini kullanarak

$$17 = \|(2, 3, -7)\|_{IL} \|(-1, 1, 2)\|_{IL} \sinh \theta$$

$$\Rightarrow 17 = \sqrt{-(2^2 + (-3)^2 - 7^2)} \sqrt{-((-1)^2 + 1^2 - 2^2)} \sinh \theta$$

$$\Rightarrow 17 = \sqrt{36} \sqrt{2} \sinh \theta$$

$$\Rightarrow \sinh \theta = \frac{17}{6\sqrt{2}}$$

olarak bulunur. Timelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Binormal vektörü,

$$B(0) = \frac{\Delta p_0 \wedge_{IL} \Delta p_1}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{(13, 11, -1)}{17} = \left(\frac{13}{17}, \frac{11}{17}, -\frac{1}{17}\right)$$

olarak elde edilir. Burada $\langle B(0), B(0) \rangle = \left(\frac{13}{17}\right)^2 + \left(\frac{11}{17}\right)^2 - \left(-\frac{1}{17}\right)^2 = 1$ olduğundan binormal vektörün spacelike vektör olduğu açıkça görülür. Asli normal vektörü ise (3.12) denklemden yararlanarak

$$\begin{aligned} N(0) &= -B(0) \wedge_{IL} T(0) = \frac{1}{17} (-13, -11, 1) \wedge_{IL} \frac{1}{6} (2, -3, 7) \\ &= \frac{1}{17 \cdot 6} (1 \cdot (-3) - (-11) \cdot 7, (-13) \cdot 7 - 1 \cdot 2, (-13) \cdot (-3) - (-11) \cdot 2) \\ &= \frac{1}{102} (74, -93, 61) \end{aligned}$$

sonucu elde edilir. İkinci bir yol olarak (3.2) ve (3.3) denklemleri kullanılarak aşağıdaki şekilde bir hesaplama yapılırsa

$$N(0) = -B(0) \wedge_{IL} T(0)$$

$$\begin{aligned} &= \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \coth \theta - \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta \\ &= \frac{(2,-3,7)}{6} \frac{\frac{19}{6\sqrt{2}}}{\frac{17}{6\sqrt{2}}} - \frac{(-1,1,2)}{\sqrt{2}} \frac{1}{\frac{17}{6\sqrt{2}}} = \frac{(2,-3,7)}{6} \frac{19}{6\sqrt{2}} \frac{6\sqrt{2}}{17} - \frac{(-1,1,2)}{\sqrt{2}} \frac{6\sqrt{2}}{17} \\ &= \frac{(38,-57,133)}{17.6} - \frac{(-36,36,72)}{17.6} = \frac{1}{102} (74, -93, 61) \end{aligned}$$

yine aynı sonuç elde edilmiş olur. Böylece timelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Serret-Frenet çatısı,

$$T|_{t=0} = \left(\frac{2}{6}, \frac{-3}{6}, \frac{7}{6}\right), B|_{t=0} = \left(\frac{13}{17}, \frac{11}{17}, \frac{-1}{17}\right), N|_{t=0} = \left(\frac{74}{102}, \frac{-93}{102}, \frac{61}{102}\right)$$

olarak elde edilir. Ayrıca timelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki eğriliği (3.13) denklem kullanılarak,

$$\begin{aligned} \kappa(0) &= \frac{\|\mathfrak{B}'(0) \wedge_{IL} \mathfrak{B}''(0)\|_{IL}}{\|\mathfrak{B}'(0)\|_{IL}^3} = \frac{3.3.2}{3^3} \frac{\|\Delta p_0 \wedge_{IL} \Delta^2 p_0\|_{IL}}{\|\Delta p_0\|_{IL}^3} \\ &= \frac{2}{3} \frac{\|(2,-3,7) \wedge_{IL} (-3,4,-5)\|_{IL}}{\|(2,-3,7)\|_{IL}^3} = \frac{2}{3} \frac{\|(13,11,-1)\|_{IL}}{\|(2,-3,7)\|_{IL}^3} = \frac{2}{3} \frac{17}{6^3} = \frac{17}{18^2} \end{aligned}$$

bulunur ve yine aynı zamanda (3.2) ve (3.3) denklemde yerine yazarsak

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL}}{\|\Delta p_0\|_{IL}^2} \sinh \theta = \frac{3-1}{3} \frac{\sqrt{2}}{6^2} \frac{17}{6\sqrt{2}} = \frac{17}{18^2}$$

bulunur. Yukarıdaki denklemlerden görüldüğü üzere iki farklı yöntem kullanıldığında da yine aynı sonuç elde edilir. Son olarak timelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki burulması ise Sonuç (3.1)'da (3.14) denklemde yerine yazıldığında,

$$\tau(0) = -\frac{3-2}{3} \frac{\begin{vmatrix} 2 & -3 & 7 \\ -1 & 1 & 2 \\ 0 & 3 & 1 \end{vmatrix}}{\|(13,11,-1)\|_{IL}^2} = -\frac{1}{3} \frac{-34}{17^2} = \frac{2}{51}$$

sonucu elde edilir [78]. ■

4.2 $\mathbb{E}^3$, TE ANA NORMALİ SPACELIKE OLAN SPACELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET-FRENET ELEMANLARI UYGULAMALARI

Örnek 4.2 $\mathfrak{B}(t)$ kontrol noktaları $p_0 = (1,0,1)$, $p_1 = (3,1,2)$, $p_2 = (4,0,1)$, $p_3 = (3,0,2)$ olan spacelike kübik Bézier eğrisi olsun. O halde spacelike konveks çokgeni

$$\Delta p_0 = p_1 - p_0 = (2,1,1),$$

$$\Delta p_1 = p_2 - p_1 = (1, -1, -1),$$

$$\Delta p_2 = p_3 - p_2 = (-1,0, -1)$$

vektörleri ile oluşur. Buradaki her bir vektör $\langle \Delta p_i, \Delta p_i \rangle > 0$ koşulunu sağladığından dolayı spacelike vektördür. Böylece $\mathfrak{B}(t)$ spacelike kübik Bézier eğrisi olur. Bu örnekte asli normali spacelike olan spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Serret-Frenet çatısı, eğriliği ve burulması aşağıdaki gibi hesaplanır.

Sonuç 2.3' daki (2.49) denkleminde yararlanarak

$$\mathfrak{B}(t) = \sum_{i=0}^3 B_i^3(t) p_i$$

ifadesinin $t = 0$ başlangıç noktasındaki birinci, ikinci ve üçüncü türevleri

$$\mathfrak{B}'(0) = 3 \sum_{i=0}^2 B_i^2(0) \Delta p_i = 3. \Delta p_0 = 3(2,1,1) = (6,3,3)$$

$$\mathfrak{B}''(0) = 3.2 \sum_{i=0}^1 B_i^1(0) \Delta^2 p_i = 3.2. \Delta^2 p_0 = 3.2. (\Delta p_1 - \Delta p_0)$$

$$= 3.2[(1, -1, -1) - (2,1,1)] = 3.2. (-1, -2, -2) = (-6, -12, -12)$$

$$\mathfrak{B}'''(0) = 3.2.1 \sum_{i=0}^0 B_i^0(0) \Delta^3 p_i = 3.2.1 \Delta^3 p_0 = 3.2.1 (\Delta p_2 - 2\Delta p_1 + \Delta p_0)$$

$$= 3.2.1[(-1,0, -1) - 2(1, -1, -1) + (2,1,1)] = 6(-1,3,2) = (-6,18,12)$$

şeklinde hesaplanır. Burada Δp_0 'ın normu,

$$\|\Delta p_0\|_{LL} = \sqrt{\langle \Delta p_0, \Delta p_0 \rangle} = \sqrt{(2^2 + 1^2 - 1^2)} = \sqrt{4} = 2$$

ile hesaplanır. Asli normali spacelike olan spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki T teğet vektörü Sonuç 3.3'den yararlanarak

$$T(0) = \frac{\mathfrak{B}'(0)}{\|\mathfrak{B}'(0)\|_{IL}} = \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} = \frac{(2,1,1)}{\sqrt{4}} = \left(\frac{2}{2}, \frac{1}{2}, \frac{1}{2}\right) = \left(1, \frac{1}{2}, \frac{1}{2}\right)$$

olarak elde edilir. Burada,

$$\langle T(0), T(0) \rangle = \left(\frac{2}{2}\right)^2 + \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2 = \frac{4}{4} + \frac{1}{4} - \frac{1}{4} = 1$$

olduğundan T teğet vektörü spacelike vektördür. Δp_1 ve Δp_0 spacelike vektörler olduğundan bu vektörler arasındaki θ açısını hesaplamak için (3.21) ve (3.22) denklemleri kullanılır. Buradan,

$$|\langle (2,1,1), (1,-1,-1) \rangle|_{IL} = \|(2,1,1)\|_{IL} \|(1,-1,-1)\|_{IL} \cos \theta$$

$$\Rightarrow |2 - 1 - (-1)|_{IL} = \sqrt{(2^2 + (1)^2 - 1^2)} \sqrt{(1^2 + (-1)^2 - (-1)^2)} \cos \theta$$

$$\Rightarrow |2|_{IL} = \sqrt{4} \sqrt{1} \cos \theta$$

$$\Rightarrow 2 = 2.1 \cos \theta$$

$$\Rightarrow \cos \theta = \frac{2}{2} = 1$$

olarak bulunur. Vektörel çarpımı, (2.55) denklemden

$$\Delta p_0 \wedge_{IL} \Delta p_1 = (1.(-1) - 1.1, 2.1 - 1.1, 2.(-1) - 1.1) = (-2, 1, -3)$$

olarak hesaplanır. Ayrıca vektörel çarpımın normu,

$$\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL} = \sqrt{(-2)^2 + 1^2 - (-3)^2} = \sqrt{4} = 2$$

dir. Buradan $\|\langle \Delta p_0, \Delta p_1 \rangle\|_{IL} = \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sin \theta$ ifadesi kullanılarak

$$2 = \|(2,1,1)\|_{IL} \|(1,-1,-1)\|_{IL} \sin \theta$$

$$\Rightarrow 2 = \sqrt{(2^2 + 1^2 - 1^2)}\sqrt{(1^2 + (-1)^2 - (-1)^2)} \sin \theta$$

$$\Rightarrow 2 = 2.1 \sin \theta$$

$$\Rightarrow \sin \theta = \frac{2}{2} = 1$$

olarak bulunur. Spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Binormal vektörü,

$$B(0) = \frac{\Delta p_0 \wedge_{IL} \Delta p_1}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{(-2, 1, -3)}{2} = \left(\frac{-2}{2}, \frac{1}{2}, \frac{-3}{2}\right)$$

olarak elde edilir. Burada $\langle B(0), B(0) \rangle = \left(\frac{-2}{2}\right)^2 + \left(\frac{1}{2}\right)^2 - \left(\frac{-3}{2}\right)^2 = -1$ olduğundan binormal vektörün spacelike vektör olduğu açıkça görülür. Ana normal vektörü ise Sonuç 3.3'den yararlanarak,

$$\begin{aligned} N(0) &= -B(0) \wedge_{IL} T(0) \\ &= \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta - \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \coth \theta \\ &= \frac{(1, -1, -1)}{1} \cdot \frac{1}{1} - \frac{(2, 1, 1)}{2} \cdot \frac{1}{1} = (-1, -2, -2) \end{aligned}$$

bulunur. Böylece spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Serret-Frenet çatısı,

$$T|_{t=0} = \left(1, \frac{1}{2}, \frac{1}{2}\right), \quad B|_{t=0} = \left(-1, \frac{1}{2}, \frac{-3}{2}\right), \quad N|_{t=0} = (-1, -2, -2)$$

olarak elde edilir. Ayrıca spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki eğriliği Sonuç 3.3 den yararlanılarak,

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL}}{\|\Delta p_0\|_{IL}^2} \sin \theta = \frac{3-1}{3} \frac{1}{2^2} \cdot 1 = \frac{1}{6}$$

bulunur. Son olarak spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki burulması ise Sonuç 3.3 de yerine yazıldığında,

$$\begin{aligned}\tau(0) &= -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}^2} = -\frac{3-2}{3} \frac{\begin{vmatrix} 2 & 1 & 1 \\ 1 & -1 & -1 \\ -1 & 0 & -1 \end{vmatrix}}{\|(-2,1,-3)\|_{IL}^2} \\ &= -\frac{1}{3} \frac{3}{2^2} = -\frac{1}{4}\end{aligned}$$

sonucu elde edilir [78]. ■

4.3 $\mathbb{E}^3$, TE ANA NORMALİ TIMELIKE OLAN SPACELIKE KÜBİK BÉZIER EĞRİLERİNİN SERRET FRENET ELEMANLARI UYGULAMALARI

Örnek 4.3 $\mathfrak{B}(t)$ kontrol noktaları $p_0 = (-3, -2, -1)$, $p_1 = (0, 0, 2)$, $p_2 = (3, 2, 4)$, $p_3 = (2, 3, 5)$ olan spacelike kübik Bézier eğrisi olsun. O halde spacelike konveks çokgeni

$$\Delta p_0 = p_1 - p_0 = (3, 2, 3),$$

$$\Delta p_1 = p_2 - p_1 = (3, 2, 2),$$

$$\Delta p_2 = p_3 - p_2 = (-1, 1, 1)$$

vektörleri ile oluşur. Buradaki her bir vektör $\langle \Delta p_i, \Delta p_i \rangle > 0$ koşulunu sağladığından dolayı spacelike vektördür. Böylece $\mathfrak{B}(t)$ spacelike kübik Bézier eğrisi olur. Bu örnekte asli normalî timelike olan spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Serret-Frenet çatısı, eğriliği ve burulması aşağıdaki gibi hesaplanır.

Sonuç 2.3'daki (2.49) denkleminde yararlanarak

$$\mathfrak{B}(t) = \sum_{i=0}^3 B_i^3(t) p_i$$

ifadesinin $t = 0$ başlangıç noktasındaki birinci, ikinci ve üçüncü türevleri

$$\mathfrak{B}'(0) = 3 \sum_{i=0}^2 B_i^2(0) \Delta p_i = 3 \cdot \Delta p_0 = 3(3, 2, 3) = (9, 6, 9)$$

$$\mathfrak{B}''(0) = 3 \cdot 2 \sum_{i=0}^1 B_i^1(0) \Delta^2 p_i = 3 \cdot 2 \cdot \Delta^2 p_0 = 3 \cdot 2 \cdot (\Delta p_1 - \Delta p_0)$$

$$= 3 \cdot 2 \cdot [(3, 2, 2) - (3, 2, 3)] = 3 \cdot 2 \cdot (0, 0, -1) = (0, 0, -1)$$

$$\begin{aligned}\mathfrak{B}'''(0) &= 3.2.1 \sum_{i=0}^0 B_i^0(0) \Delta^3 p_i = 3.2.1 \Delta^3 p_0 = 3.2.1 (\Delta p_2 - 2\Delta p_1 + \Delta p_0) \\ &= 3.2.1 [(-1, 1, 1) - 2(3, 2, 2) + (3, 2, 3)] = 6(-4, -1, 0) = (-24, -6, 0)\end{aligned}$$

şeklinde hesaplanır. Burada Δp_0 'ın normu,

$$\|\Delta p_0\|_{IL} = \sqrt{\langle \Delta p_0, \Delta p_0 \rangle} = \sqrt{(3^2 + 2^2 - 3^2)} = \sqrt{2^2} = 2$$

ile hesaplanır. Ana normali timelike olan spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki T teğet vektörü Sonuç 3.1 denkleminde yararlanarak

$$T(0) = \frac{\mathfrak{B}'(0)}{\|\mathfrak{B}'(0)\|_{IL}} = \frac{\Delta p_0}{\|\Delta p_0\|_{IL}} = \frac{(3, 2, 3)}{\sqrt{4}} = \left(\frac{3}{2}, \frac{2}{2}, \frac{3}{2}\right) = \left(\frac{3}{2}, 1, \frac{3}{2}\right)$$

olarak elde edilir. Burada,

$$\langle T(0), T(0) \rangle = \left(\frac{3}{2}\right)^2 + \left(\frac{2}{2}\right)^2 - \left(\frac{3}{2}\right)^2 = \frac{9}{4} + \frac{4}{4} - \frac{9}{4} = \frac{4}{4} = 1$$

olduğundan dolayı T teğet vektörü spacelike vektördür. Δp_1 ve Δp_0 spacelike vektörler olduğundan bu vektörler arasındaki θ açısını hesaplamak için (3.23) ve (3.24) denklemleri kullanılır. Buradan,

$$|\langle (3, 2, 3), (3, 2, 2) \rangle|_{IL} = -\|(3, 2, 3)\|_{IL} \|(3, 2, 2)\|_{IL} \cosh \theta$$

$$\Rightarrow |9 + 4 - 6|_{IL} = -\sqrt{(3^2 + (2)^2 - 3^2)} \sqrt{(3^2 + (2)^2 - (2)^2)} \cosh \theta$$

$$\Rightarrow |7|_{IL} = -\sqrt{2^2} \sqrt{3^2} \cosh \theta$$

$$\Rightarrow 7 = -2.3 \cosh \theta$$

$$\Rightarrow \cos \theta = -\frac{7}{2.3}$$

olarak bulunur. Vektörel çarpımı, (2.55) denklemden

$$\Delta p_0 \wedge_{IL} \Delta p_1 = (3, 2, 3) \wedge_{IL} (3, 2, 2) = (3.2 - 2.2, 3.2 - 3.3, 3.2 - 3.1) = (2, -3, 3)$$

olarak hesaplanır. Ayrıca vektörel çarpımın normu alındığında,

$$\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL} = \sqrt{(2)^2 + (-3)^2 - (3)^2} = \sqrt{4} = 2$$

dir. Buradan $\|\langle \Delta p_0, \Delta p_1 \rangle\|_{IL} = \|\Delta p_0\|_{IL} \|\Delta p_1\|_{IL} \sinh \theta$ ifadesini kullanarak

$$2 = \|(3,2,3)\|_{IL} \|(3,2,2)\|_{IL} \sinh \theta$$

$$\Rightarrow 2 = \sqrt{(3^2 + 2^2 - 3^2)} \sqrt{(3^2 + (2)^2 - (2)^2)} \sinh \theta$$

$$\Rightarrow 2 = 2.3 \sinh \theta$$

$$\Rightarrow \sinh \theta = \frac{1}{3}$$

olarak bulunur. Spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Binormal vektörü,

$$B(0) = \frac{\Delta p_0 \wedge_{IL} \Delta p_1}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}} = \frac{(2, -3, 3)}{2} = \left(\frac{2}{2}, \frac{-3}{2}, \frac{3}{2}\right)$$

olarak elde edilir. Burada $\langle B(0), B(0) \rangle = \left(\frac{2}{2}\right)^2 + \left(\frac{-3}{2}\right)^2 - \left(\frac{3}{2}\right)^2 = 1$ olduğundan binormal vektörün spacelike vektör olduğu açıkça görülür. Ana normal vektörü ise Sonuç 3.1 deki 3.2 denkleminde yararlanarak

$$N(0) = -B(0) \wedge_{IL} T(0) = -\frac{\Delta p_0}{\|\Delta p_0\|_{IL}} \coth \theta - \frac{\Delta p_1}{\|\Delta p_1\|_{IL}} \operatorname{csch} \theta$$

$$= -\frac{(3,2,3)}{2} \cdot \frac{-\frac{7}{6}}{\frac{1}{3}} - \frac{(3,2,2)}{3} \cdot \frac{1}{\frac{1}{3}} = -\frac{(3,2,3)}{2} \cdot \left(-\frac{7}{6}\right) \frac{3}{1} - \frac{(3,2,2)}{3} \cdot \frac{3}{1}$$

$$= \frac{(9, -1, 13)}{4}$$

bulunur. Böylece spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki Serret-Frenet çatısı,

$$T|_{t=0} = \left(\frac{3}{2}, 1, \frac{3}{2}\right), B|_{t=0} = \left(\frac{2}{2}, \frac{-3}{2}, \frac{3}{2}\right), N|_{t=0} = \frac{(9, -1, 13)}{4}$$

olarak elde edilir. Ayrıca spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki eğriligi Sonuç 3.1 deki 3.1.13 denklemden yararlanılarak,

$$\kappa(0) = \frac{m-1}{m} \frac{\|\Delta p_1\|_{IL}}{\|\Delta p_0\|_{IL}^2} \sin \theta = \frac{3-1}{3} \frac{3}{2^2} \cdot \frac{1}{3} = \frac{1}{6}$$

bulunur. Son olarak spacelike kübik Bézier eğrisinin $t = 0$ başlangıç noktasındaki burulması ise Sonuç 3.1’ da 3.14 denklemde yerine yazıldığında,

$$\begin{aligned} \tau(0) &= -\frac{m-2}{m} \frac{\det(\Delta p_0, \Delta p_1, \Delta p_2)}{\|\Delta p_0 \wedge_{IL} \Delta p_1\|_{IL}^2} \\ &= -\frac{3-2}{3} \frac{\begin{vmatrix} 3 & 2 & 3 \\ 3 & 2 & 2 \\ -1 & 1 & 1 \end{vmatrix}}{\|(2, -3, 3)\|_{IL}^2} = -\frac{1}{3} \frac{5}{2^2} = -\frac{5}{12} \end{aligned}$$

sonucu elde edilir [78]. ■



BÖLÜM 5

EĞRİ UYDURMA (CURVE FITTING)

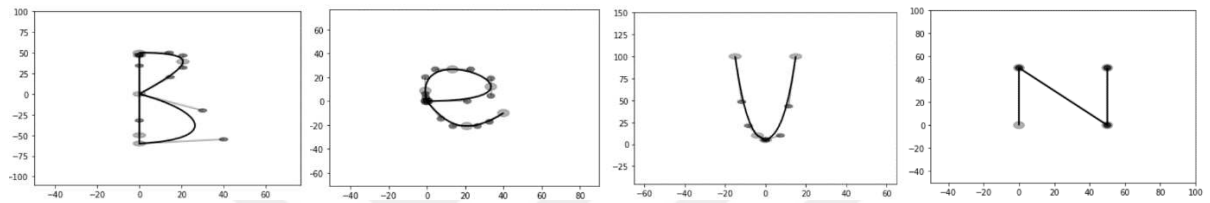
Bu bölümde, ikinci bölümde teorik bilgisi verilen Bézier eğrileri ve makine öğrenmesinin eğri uydurma (curve fitting) yöntemi kullanılarak Python 3.10 (64 bit) bilgisayar programlama dili yardımıyla Bézier eğrileri iki farklı bakış açısı ile ele alınacaktır. Bunlardan birincisi kontrol noktaları değiştirilerek oluşacak yeni görüntülerin incelenmesi; ikincisi de kontrol noktalarını değiştirmeden parametre aralığının artış miktarı değiştirildiğinde görüntüdeki değişimin incelenmesidir.

Doğada bilim insanları tarafından matematiksel olarak temsil edilebilecek ve sonlu sayıda eğrinin birleşmesiyle oluşturulabilecek sonsuz çoklukta şekil yer almaktadır. İstenen şekil tam olarak tasvir edilemese bile en yakın görüntüyü elde edebilmek için yaklaşım eğrisi veya serbest biçimli eğri kullanılır. Eğri uydurma yöntemi, bir dizi veri noktası ile en iyi uyumu sağlayan istenen nesnenin görüntüsünü oluşturma sürecini ifade eder. Eğri uydurma algoritmasının amacı, görüntünün ana hatlarına minimum değişimle yaklaşmak için minimum sayıda kübik eğri parçası kullanmaktır.

Bu kavrama çok yakın olan interpolasyon kavramı genellikle aynı anlamı taşıyor gibi görünse de birbirinden ayrı kavramlardır. Eğriler arasında iki yöntemde de fonksiyon girdi ve çıktılarıyla alakalı benzer bir ilişki olduğu düşünülebilir. İnterpolasyon yönteminde genellikle veri noktaları girildiğinde çıkan sonuç mutlaka eğri üzerinde bulunmalıdır. Ancak eğri uydurma yönteminde interpolasyon yönteminin tam aksine veri noktaları girildiğinde eğrinin bu noktalardan geçmesinin zorunlu olmadan en uygun görüntüye ulaşılması hedeflenir. Bu iki yöntem arasındaki en temel fark, veri noktalarının ulaşılacak istenilen görüntünün üzerinde olup olmadığıdır. Eğri uydurma iki veya daha fazla değişken arasındaki ilişki hakkında örnek verilere dayalı olarak veri görselleştirmeye yardımcı olabilir. Bézier eğrileri kontrol noktalarına bağlı olarak değiştirme özellikleri ve derecesi ile kontrol noktaları arasında bağlantı olması ve çeşitli özellikleri sağlaması nedeniyle eğri uydurma işlemlerinde yaygın olarak tercih edilen eğri çeşitlerindendir. “An algorithm for automatic capturing of the font outlines” isimli

makalede özellikle yazı tiplerinin ana hatlarını yakalamak için yararlı olan, optimum eğri tasarımı için basit ve etkili bir yöntem sunulması bu konunun temsillerinden sadece bir tanesidir [86]. Aşağıdaki örneklerle eğri uydurma yönteminde iki boyutlu ve üç boyutlu Bézier eğrilerinin kontrol noktalarının değiştirilmesi ve parametre aralığındaki artış hızının değiştirilmesiyle ortaya çıkan görüntüler Python 3.10 programlama dilinde çizdirilerek incelenmiştir [82-89, 92].

Örnek 5.1. 4 kontrol noktası ile verilen kübik Bézier eğrilerinin kontrol noktalarının değişmesiyle oluşan harflerin Python 3.10 (64 bit) görüntüleri ve algoritmaları incelenmektedir [90].



Şekil 5.1 (a, b, c, d): 4 kontrol noktası ile verilen kübik Bézier eğrilerinin kontrol noktalarının değişmesiyle oluşan harflerin Python 3.10 görüntüsü.

```
In [38]: from beziers.point import Point
from beziers.path import BezierPath
from beziers.cubicbezier import CubicBezier
b1 = CubicBezier(
    Point(0.0,50.0), Point(30.0,50.0), Point(25.0,35.0), Point(0.0,0.0)
)
b2 = CubicBezier(
    Point(60.0,-20.0), Point(30.0,-20.0), Point(40.0,-55.0), Point(0.0,-60.0)
)
b3 = CubicBezier(
    Point(0.0,0.0), Point(0.0,120.0), Point(0.0,-25.0), Point(0.0,-50.0)
)
path = BezierPath.fromSegments([b1,b2,b3])
path.closed = False
path.addExtremes()
path.balance()
path.translate(Point(0.0,0.0))

import matplotlib.pyplot as plt
fig, ax = plt.subplots()
path.addExtremes()
path.plot(ax)
plt.show()
```

Şekil 5.2 Şekil 5.1.a Python 3.10 kodu (B)

```
In [1]: from beziers.point import Point
from beziers.path import BezierPath
from beziers.cubicbezier import CubicBezier
b1 = CubicBezier(
    Point(40.0,-10.0), Point(20.0,-30.0), Point(10.0,-20.0), Point(0.0,0.0)
)
b2 = CubicBezier(
    Point(0.0,0.0), Point(0.0,0.0), Point(0.0,0.0), Point(0.0,0.0)
)
b3 = CubicBezier(
    Point(0.0,0.0), Point(-10.0,60.0), Point(80.0,0.0), Point(0.0,0.0)
)
path = BezierPath.fromSegments([b1,b2,b3])
path.closed = False
path.addExtremes()
path.balance()
path.translate(Point(0.0,0.0))

import matplotlib.pyplot as plt
fig, ax = plt.subplots()
path.addExtremes()
path.plot(ax)
plt.show()
```

Şekil 5.3 Şekil 5.1.b Python 3.10 kodu (E)

```
In [2]: from beziers.point import Point
from beziers.path import BezierPath
from beziers.cubicbezier import CubicBezier
b1 = CubicBezier(
    Point(-15.0,100.0), Point(-10.0,20.0), Point(-6.0,15.0), Point(-4.0,10.0)
)
b2 = CubicBezier(
    Point(0.0,5.0), Point(0.0,5.0), Point(0.0,5.0), Point(0.0,5.0)
)
b3 = CubicBezier(
    Point(4.0,10.0), Point(6.0,10.0), Point(10.0,20.0), Point(15.0,100.0)
)
path = BezierPath.fromSegments([b1,b2,b3])
path.closed = False
path.addExtremes()
path.balance()
path.translate(Point(0.0,0.0))

import matplotlib.pyplot as plt
fig, ax = plt.subplots()
path.addExtremes()
path.plot(ax)
plt.show()
```

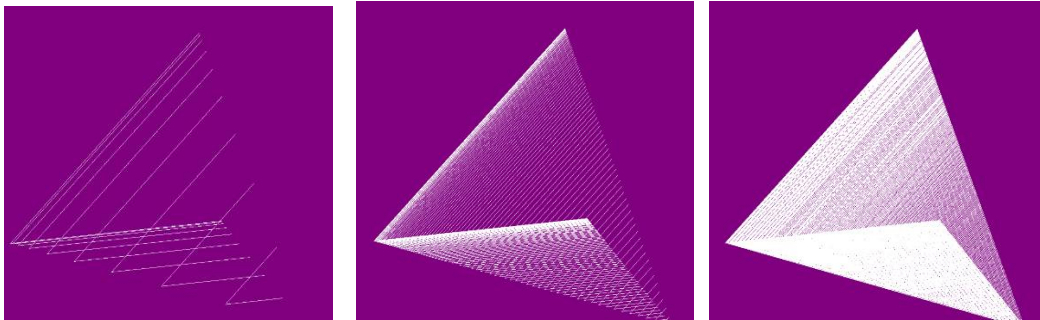
Şekil 5.4 Şekil 5.1.c Python 3.10 kodu (U)

```
In [14]: from beziers.point import Point
from beziers.path import BezierPath
from beziers.cubicbezier import CubicBezier
b1 = CubicBezier(
    Point(0.0,0.0), Point(0.0,50.0), Point(0.0,50.0), Point(0.0,50.0)
)
b2 = CubicBezier(
    Point(50.0,0.0), Point(50.0,0.0), Point(50.0,0.0), Point(50.0,0.0)
)
b3 = CubicBezier(
    Point(50.0,50.0), Point(50.0,50.0), Point(50.0,50.0), Point(50.0,50.0)
)
path = BezierPath.fromSegments([b1,b2,b3])
path.closed = False
path.addExtremes()
path.balance()
path.translate(Point(0.0,0.0))

import matplotlib.pyplot as plt
fig, ax = plt.subplots()
path.addExtremes()
path.plot(ax)
plt.show()
```

Şekil 5.5 Şekil 5.1.d Python 3.10 kodu (N)

Örnek 5.2 4 adet kontrol noktası verilerek oluşturulan kübik Bézier eğrisinin Python 3.10 (64 bit) algoritması ve grafiği aşağıdaki gibidir. $0 < t < 1$ olmak üzere sırasıyla ($t = 0,1$), ($t = 0,01$), ($t = 0,001$) artış miktarına göre verilen kontrol noktası değerlerinin değişen şekillerini incelenmiştir [91].



Şekil 5.6 (a, b, c) $t = 0,1, t = 0,01, t = 0,001$ artış miktarına göre 4 adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 görüntüsü

```

In [121]: import tkinter
import math

class Bezier(tkinter.Canvas):
    # Because Canvas doesn't support simple pixel plotting,
    # we need to help us out with a line with length 1 in
    # positive x direction.
    def plot_pixel(self, x0, y0, z0, color):
        self.create_line(x0, y0, z0, x0+1, y0+1, z0, fill=color)

    # Calculates the quadratic Bézier polynomial for
    # the n+1=3 coordinates.
    def quadratic_bezier_sum(self, t, w):
        t2 = t * t
        mt = 1-t
        mt2 = mt * mt
        return int(w[0]*mt2 + w[1]*2*mt*t + w[2]*t2)

    # Calculates the cubic Bézier polynomial for
    # the n+1=4 coordinates.
    def cubic_bezier_sum(self, t, w):
        t2 = t * t
        t3 = t2 * t
        mt = 1-t
        mt2 = mt * mt
        mt3 = mt2 * mt
        return int(w[0]*mt3 + 3*w[1]*mt2*t + 3*w[2]*mt*t2 + w[3]*t3)

    def draw_quadratic_bez(self, p1, p2, p3, color):
        t = 0
        while (t < 1):
            x = self.quadratic_bezier_sum(t, (p1[0], p2[0], p3[0]))
            y = self.quadratic_bezier_sum(t, (p1[1], p2[1], p3[1]))
            z = self.quadratic_bezier_sum(t, (p1[2], p2[2], p3[2]))
            # self.plot_pixel(math.floor(x), math.floor(y))
            self.plot_pixel(x, y, z, color)
            t += 0.1 # 1000 iterations. If you want the curve to be really
                    # fine grained, consider "t += 0.0001" for ten thousand iterations.

    def draw_cubic_bez(self, p1, p2, p3, p4, color):
        t = 0
        while (t < 1):
            x = self.cubic_bezier_sum(t, (p1[0], p2[0], p3[0], p4[0]))
            y = self.cubic_bezier_sum(t, (p1[1], p2[1], p3[1], p4[1]))
            z = self.cubic_bezier_sum(t, (p1[2], p2[2], p3[2], p4[2]))
            self.plot_pixel(x, y, z, color)
            t += 0.1

In [122]: master = tkinter.Tk()
w = Bezier(master, width=1000, height=1000, bg='purple')
w.pack()

---# Finally draw some Bézier curves :)
#w.draw_quadratic_bez((600, 548, 100), (600, 548, 100), (600, 548, 100), 'pink')
w.draw_cubic_bez((600, 548, 100), (600, 548, 100), (600, 548, 100), (800, 800, 800), 'white')
tkinter.mainloop()

```

Şekil 5.7 Şekil 5.6.a t = 0,1 artış miktarına göre 4 adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 kodu

```

In [118]: import tkinter
import math

class Bezier(tkinter.Canvas):
    # Because Canvas doesn't support simple pixel plotting,
    # we need to help us out with a line with length 1 in
    # positive x direction.
    def plot_pixel(self, x0, y0, z0, color):
        self.create_line(x0, y0, z0, x0+1, y0+1, z0, fill=color)

    # Calculates the quadratic Bézier polynomial for
    # the n+1=3 coordinates.
    def quadratic_bezier_sum(self, t, w):
        t2 = t * t
        mt = 1-t
        mt2 = mt * mt
        return int(w[0]*mt2 + w[1]*2*mt*t + w[2]*t2)

    # Calculates the cubic Bézier polynomial for
    # the n+1=4 coordinates.
    def cubic_bezier_sum(self, t, w):
        t2 = t * t
        t3 = t2 * t
        mt = 1-t
        mt2 = mt * mt
        mt3 = mt2 * mt
        return int(w[0]*mt3 + 3*w[1]*mt2*t + 3*w[2]*mt*t2 + w[3]*t3)

    def draw_quadratic_bez(self, p1, p2, p3, color):
        t = 0
        while (t < 1):
            x = self.quadratic_bezier_sum(t, (p1[0], p2[0], p3[0]))
            y = self.quadratic_bezier_sum(t, (p1[1], p2[1], p3[1]))
            z = self.quadratic_bezier_sum(t, (p1[2], p2[2], p3[2]))
            # self.plot_pixel(math.floor(x), math.floor(y))
            self.plot_pixel(x, y, z, color)
            t += 0.01 # 1000 iterations. If you want the curve to be really
                    # fine grained, consider "t += 0.0001" for ten thousand iterations.

    def draw_cubic_bez(self, p1, p2, p3, p4, color):
        t = 0
        while (t < 1):
            x = self.cubic_bezier_sum(t, (p1[0], p2[0], p3[0], p4[0]))
            y = self.cubic_bezier_sum(t, (p1[1], p2[1], p3[1], p4[1]))
            z = self.cubic_bezier_sum(t, (p1[2], p2[2], p3[2], p4[2]))
            self.plot_pixel(x, y, z, color)
            t += 0.01

In [120]: master = tkinter.Tk()
w = Bezier(master, width=1000, height=1000, bg='purple')
w.pack()

—># Finally draw some Bézier curves :)
#w.draw_quadratic_bez((600, 548, 100), (600, 548, 100), (600, 548, 100), 'pink')
w.draw_cubic_bez((600, 548, 100), (600, 548, 100), (600, 548, 100), (800, 800, 800), 'white')
tkinter.mainloop()

```

Şekil 5.8 Şekil 5.6.b $t = 0,01$ artış miktarına göre 4 adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 kodu

```

In [121]: import tkinter
import math

class Bezier(tkinter.Canvas):
    # Because Canvas doesn't support simple pixel plotting,
    # we need to help us out with a line with length 1 in
    # positive x direction.
    def plot_pixel(self, x0, y0, z0, color):
        self.create_line(x0, y0, z0, x0+1, y0+1, z0, fill=color)

    # Calculates the quadratic Bézier polynomial for
    # the n+1=3 coordinates.
    def quadratic_bezier_sum(self, t, w):
        t2 = t * t
        mt = 1-t
        mt2 = mt * mt
        return int(w[0]*mt2 + w[1]*2*mt*t + w[2]*t2)

    # Calculates the cubic Bézier polynomial for
    # the n+1=4 coordinates.
    def cubic_bezier_sum(self, t, w):
        t2 = t * t
        t3 = t2 * t
        mt = 1-t
        mt2 = mt * mt
        mt3 = mt2 * mt
        return int(w[0]*mt3 + 3*w[1]*mt2*t + 3*w[2]*mt*t2 + w[3]*t3)

    def draw_quadratic_bez(self, p1, p2, p3, color):
        t = 0
        while (t < 1):
            x = self.quadratic_bezier_sum(t, (p1[0], p2[0], p3[0]))
            y = self.quadratic_bezier_sum(t, (p1[1], p2[1], p3[1]))
            z = self.quadratic_bezier_sum(t, (p1[2], p2[2], p3[2]))
            # self.plot_pixel(math.floor(x), math.floor(y))
            self.plot_pixel(x, y, z, color)
            t += 0.001 # 1000 iterations. If you want the curve to be really
                        # fine grained, consider "t += 0.0001" for ten thousand iterations.

    def draw_cubic_bez(self, p1, p2, p3, p4, color):
        t = 0
        while (t < 1):
            x = self.cubic_bezier_sum(t, (p1[0], p2[0], p3[0], p4[0]))
            y = self.cubic_bezier_sum(t, (p1[1], p2[1], p3[1], p4[1]))
            z = self.cubic_bezier_sum(t, (p1[2], p2[2], p3[2], p4[2]))
            self.plot_pixel(x, y, z, color)
            t += 0.001

In [122]: master = tkinter.Tk()
w = Bezier(master, width=1000, height=1000, bg='purple')
w.pack()

—># Finally draw some Bézier curves :)
#w.draw_quadratic_bez((600, 548, 100), (600, 548, 100), (600, 548, 100), 'pink')
w.draw_cubic_bez((600, 548, 100), (600, 548, 100), (600, 548, 100), (800, 800, 800), 'white')
tkinter.mainloop()

```

Şekil 5.9 Şekil 5.6.c $t = 0,001$ artış miktarına göre 4 adet kontrol noktası verilen kübik Bézier eğrisinin Python 3.10 kodu



BÖLÜM 6

SONUÇ

Bu tezde eğriler ve yüzeylerin bilgisayar sistemlerinde çizdirilebilmesinde rol oynayan öncü eğri çeşitlerinden biri olan Bézier eğrileri ile ilgili çalışmalar yapılmıştır. Bézier eğrileri başlangıç ve bitiş noktalarına sahip olmakla birlikte bu noktalar arasında sonlu sayıda kontrol noktasına sahiptir. Dolayısıyla bu kontrol noktaları ve ilgili parametrenin değer aralığı değiştirilerek çeşitli görüntülere ulaşılmıştır. Özetle, tezin birinci bölümünde Yapay Zekâ (YZ), Makine Öğrenmesi (MÖ) ve Bézier eğrileri ve bu üçünün kesişim kümesinde yer alan önemli motive edici çalışmalara vurgu yapılmıştır. İkinci bölümde hem Öklid uzayında hem de üç boyutlu Minkowski uzayında Bézier eğrisinden bahsedilip, bu eğrilerin ve bazı örneklerinin grafiklerinin Python 3.10 (64 bit) bilgisayar programlama dili vasıtasıyla grafikleri çizdirilerek kodları ek olarak tezin sonunda verilmiştir. Üçüncü ve dördüncü bölümlerde ise üç boyutlu Minkowski uzayında Bézier eğrilerinin Serret-Frenet çatılarıyla ilgili ispatlar verilir; önce $p_0 = (1,3,-3)$, $p_1 = (3,0,4)$, $p_2 = (2,1,6)$, $p_3 = (2,5,5)$ kontrol noktaları için ana normali timelike olan timelike kübik Bézier eğrilerinin Serret-Frenet elemanları, eğriliği ve burulmasının sonuçları, daha sonra da $p_0 = (1,0,1)$, $p_1 = (3,1,2)$, $p_2 = (4,0,1)$, $p_3 = (3,0,2)$ kontrol noktaları için ana normali spacelike olan spacelike kübik Bézier eğrilerinin Serret-Frenet elemanları, eğriliği ve burulması ile ilgili örneklerle yer verilmiştir. Son bölümde ise yapay zekânın alt dalı olan makine öğrenmesinin eğri uydurma (curve fitting) yöntemi ile ilgili bilgilere yer verilip bununla ilgili farklı kontrol noktalarına sahip Bézier eğrilerinin kontrol noktaları elde edilmek istenen görüntüye göre keyfi olarak değiştirildiğinde elde edilen görüntüleri ve bir diğer örnekte de yine dört kontrol noktasına sahip Bézier eğrilerinin bu sefer de verilen parametre aralığındaki artış miktarı değiştirilerek elde edilen görüntüleri Python 3.10 (64 bit) programlama dilinde çizdirilerek sunulmuştur. Ele alınan örneklerde kontrol noktalarına keyfi değerler verildiğinde bu eğrilerin kontrol noktalarına bağlı olarak değiştiği ve istenen görüntünün elde edilebildiği gözlemlenmiştir. Ayrıca; bu çalışma Python 3.10 (64 bit) programlama dilinin ve matematik kütüphanelerinin zenginleşmesine de katkıda bulunabilir. Gelecek çalışmalarda ise; Python 3.10 (64 bit) bilgisayar programlama dilinin etkili ve pratik

kullanımıyla, Bézier eğrilerinin özel bir hali olan rasyonel Bézier eğrileri veya Bézier yüzeyleri ile ilgili nümerik örnekler ve görseller ilginç gerçek hayat uygulamalarında kullanılabilir.



KAYNAKLAR

- [1] **Özdemir R** (2021) Makine Öğrenmesindeki Sınıflandırma Yöntemlerinin Karşılaştırılması ve E-ticaret Üzerinde Bir Uygulama. *Yüksek Lisans Tezi*, T.C. İstanbul Ticaret Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 94 s.
- [2] **Kurtboğan H** (2023) Yeni Dünyada Yapay Zekâ Metaforu ve Yapay Zekanın Çalışan Performansına Etkisi. *Yüksek Lisans Tezi*, Karamanoğlu Mehmetbey Üniversitesi Sosyal Bilimler Enstitüsü, 173 s.
- [3] **Dereli T** (2020) Birey ve Toplum Güvenliği, Yapay Zekâ ve İnsanlık. *Bilişim Teknolojileri ve İletişim Dergisi*, 18 s.
- [4] **Öztürk K ve Şahin M E** (2018) Yapay Sinir Ağları ve Yapay Zekaya Genel Bir Bakış. *Takvim-i Vekayi*, 6(2): 25-36.
- [5] **Alpaydın E** (2010) *Introduction to Machine Learning*. ISBN 978-0-262-02818-9 Massachusetts Institute of Technology Press, United States of America, 640 s.
- [6] **Kalaycı T T** (2006) Yapay Zekâ Teknikleri Kullanan Üç Boyutlu Grafik Yazılımları İçin “Extensible 3D” (X3d) ile Bir Altyapı Oluşturulması ve Gerçekleştirimi, *Yüksek Lisans Tezi*, Ege Üniversitesi Fen Bilimleri Enstitüsü, İzmir, Sf. 138.
- [7] **Hsu S C, Huang H-H And Huang C-L** (2017) Facial Expression Recognition for Human-Robot Interaction. *2017 First IEEE International Conference on Robotic Computing (IRC)*, Taiwan, pp. 1-7.
- [8] **Khan M I and Bhuiyan M A A** (2009) Facial Expression Recognition for Human-Robot Interface. *International Journal of Computer Science and Network Security*, vol. 9, no. 4, pp. 300-306.
- [9] **Danışman T** (2008) Multi-modal Emotion Recognition In Video. *PhD Thesis*, Dokuz Eylül University Graduate School Of Natural And Applied Sciences, İzmir, 159 s.
- [10] **Sarwar A, Ali M, Suri J And Sharma V** (2015) Performance Evaluation Of Machine Learning Techniques For Screening Of Cervical Cancer, *In 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 880-886.
- [11] **Arı İ ve Akarun L** (2009) Yüz Özniteliklerinin Takibi ve İşaret Dili İçin İfade Tanıma. *Sinyal İşleme ve Uygulamaları Konferansı (SIU2009)*, Antalya, 2009.
- [12] **Coşkun F** (2010) Video Görüntülerde Gerçek Zamanlı Trafik Şeridi Tespit ve Takip Sistemi. *Yüksek Lisans Tezi*, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü. İstanbul, sf. 133.

KAYNAKLAR (devam ediyor)

- [13] **Zainudin M L, Yahya Z R, Muhamad W Z A W and Hasan Z A** (2021) Curve reconstruction by cubic ball on Arabic fonts using butterfly optimization algorithm. *Journal of Theoretical and Applied Information Technology*, 99(23):5700–5706.
- [14] **Hacıoğlu A** (2005) Lüle Tasarımında Yapay Zekâ Kullanımı. *Havacılık ve Uzay Teknolojileri Dergisi*, 2(1): 1-7.
- [15] **Buran B A** (2019) Kısıtlara Bağlı Matematiksel Modelleme ile İnsansız Hava Aracı İçin Yumuşatılmış Rota Planlaması. *Doktora Tezi*, T.C. İstanbul Kültür Üniversitesi Lisansüstü Eğitim Enstitüsü, sf. 181.
- [16] **Samakeh M** (2023) Diagnosis Of Diseases Through Eye Images Using Artificial Intelligence. *Master Thesis*, T.C. İstanbul Aydın University, Institute Of Graduate Studies, Department Of Software Engineering, İstanbul, 78 s.
- [17] **İlhan Mavi BN** (2023) Ortodonti Kliniğine Başvuran Hastaların Okluzal Fotoğraflarından Yapay Zekâ Kullanılarak Çapraşıklık Şiddetinin Belirlenmesi. *Uzmanlık Tezi*, T. C. Sivas Cumhuriyet Üniversitesi Diş Hekimliği Fakültesi Ortodonti Anabilim Dalı, Sivas, Sf. 79.
- [18] **Çalikkocaoğlu S** (1998) Tam Protezler. *İstanbul: Protez Akademisi ve Gnatoloji Derneği 2. Bilimsel Yayını*, 484-500.
- [19] **Ararat M** (2023) Tarihi Yapılarda Yapay Zekâ Destekli Hasar Tespiti İçin Bir Model Önerisi. *Doktora Tezi*, T.C. Hasan Kalyoncu Üniversitesi Lisansüstü Eğitim Enstitüsü Mimarlık Anabilim Dalı, Gaziantep, Sf. 115.
- [20] **Sayan Yüksel Z K** (2023), İnsan Kaynakları Yönetiminde Yapay Zekâ ve Türkiye Uygulaması. *Yüksek Lisans Tezi*, T.C. İstanbul Nişantaşı Üniversitesi Lisansüstü Eğitim Enstitüsü, İstanbul, Sf. 79.
- [21] **Kültür Y** (2017) Novel Models For Credit Card Fraud Detection. *PhD Thesis*, Boğaziçi University, Graduate Program in Computer Engineering, İstanbul, 107 s.
- [22] **İpçi Ö** (2021) Avukatlık Mesleğinde Yapay Zekâ Kullanımı. *Yüksek Lisans Tezi*, T.C. Trakya Üniversitesi Sosyal Bilimler Enstitüsü Kamu Yönetimi Anabilim Dalı, Edirne, Sf. 153.
- [23] **Dirim T** (2022) Öklid Uzayında Bézier Eğrileri ve Yüzeyleri. *Yüksek Lisans Tezi*, Bursa Uludağ Üniversitesi Fen Bilimleri Enstitüsü, 80 s.
- [24] **Yılmaz A** (2009) Bézier eğrileri ve Bézier Yüzeyleri. *Yüksek Lisans Tezi*, Akdeniz Üniversitesi Fen Bilimleri Enstitüsü, Matematik Anabilim Dalı, 57 s.
- [25] **İncesu M** (2003) Bézier eğrileri, Bézier yüzeyleri ve Matlab ile Sayısal Algoritmalar. *Yüksek Lisans Tezi*, Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü, Matematik Anabilim Dalı, 116 s.

KAYNAKLAR (devam ediyor)

- [26] **Farin G** (2002) *A History of Curves and Surfaces in CAGD*. Handbook on Computer Aided Geometric Design, pp. 1-21.
- [27] **Marsh D** (2005) *Applied Geometry for Computer Graphics and CAD*. Second Edition, Springer, 361 s.
- [28] **Kaplan C S and Mann S** (2016) *Introduction to Computer Graphics*. Lecture notes CS 488/688 of University of Waterloo, 327 s.
- [29] **Saxena A and Sahay B** (2002) *Computer Aided Engineering Design*. ISBN:1-4020-2555-6, Springer, New York, 410 s.
- [30] **Farin G** (2002) *Curves and Surfaces for CAGD: A Practical Guide*. Fifth Edition, Morgan Kaufmann Publishers, 447 s.
- [31] **Şimşek B** (2016) Bezier Eğrileri ile Deforme El Modelinin Geliştirilmesi. *Yüksek Lisans Tezi*, T.C. Akdeniz Üniversitesi Sağlık Bilimleri Enstitüsü Biyoistatistik ve Tıbbi Bilişim Anabilim Dalı, Antalya, sf. 53.
- [32] **Weiss T, Suresh B and Behl M** (2020) Bezier Curve Based End-To-End Trajectory Synthesis For Agile Autonomous Driving. *Proc. NeurIPS Mach. Learn. Auton. Driving Workshop*, 10 s.
- [33] **Safaruddin M S and Misro M Y** (2021) Multi-Objectives Path Planning using Bézier Curve. *Malaysian Journal of Mathematical Sciences* 15(1): 45-59.
- [34] **Özmen G ve Kandemir R** (2012) Haar Dalgacıkları ve Kübik Bezier Eğrileri ile Yüz İfadesi Tespiti. *ELECO '2012 Elektrik-Elektronik ve Bilgisayar Mühendisliği Sempozyumu, 29 Kasım- 01 Aralık 2012, Bursa*.
- [35] **Bezier P E** (1971) Example of an Existing System in the Motor Industry: The Unisurf System. *Proceedings The Royal Society Lond. A*, 321, 207-218.
- [36] **Hacıoğlu A and Özkol İ** (2003) Kanat profili tasarımında genetik algoritma kullanımı. *İTÜDERGİSİ/d*, Cilt:2, Sayı:6, 73-81.
- [37] **Akalin R, Karateke S, Gümüş M, and Zontul M** (2022) A Review on Bézier Curves Applications in Computer Aided Geometric Design (CAGD). *International Graduate Research Symposium IGRS'22* (Oral Presentation), 02 Haziran 2022, İstanbul, Turkey.
- [38] **Karateke S, Akalin R, and Gümüş M** (2023) The Evolution of Bézier Curves in Computer-Aided Geometric Design (CAGD): A Systematic Review. In: *Akpınar, A. (ed.), Research on Mathematics and Science- II. Özgür Publications*, 67-81.
- [39] **Simsek B and Yardimci A** (2016) Using Bezier Curves in Medical Applications. *Filomat* 30:4, 937–943.
- [40] **Abdel-Aziz H S, Zanaty E A, Ali H A and Saad M K** (2021) Generating Bézier curves for medical image reconstruction. *Results in Physics*, 23.

KAYNAKLAR (devam ediyor)

- [41] **Zain SAAAS, Misro M Y and Miura K T** (2023) Generalized Riemann-Liouville Fractional Bézier Curve And Its Applications In Engineering Surface. *Alexandria Engineering Journal*, 65, 585-606.
- [42] **Schumaker L L and Yu A** (2022) Approximation by polynomial splines on curved triangulations. *Computer Aided Geometric Design*, Volume 92, 102050.
- [43] **Juhász I** (2021) A NURBS transition between a Bézier curve and its control polygon. *Journal of Computational and Applied Mathematics*, 396, 113626.
- [44] **Altunordu Ş** (2021) Bezier Eğrileri ve Bazı Uygulamaları. *Yüksek Lisans Tezi*, T.C. Ordu Üniversitesi, Fen Bilimleri Enstitüsü, sf. 111.
- [45] **Bayrakdar S, Akgün D and Yücedağ İ** (2017) Video dosyaları üzerinde yüz ifade analizi için hızlandırılmış bir yaklaşım, *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 23(5), 602-613.
- [46] **Hasanvand A** (2021) Bézier Curve. 13 s.
- [47] **Farin G** (2006) Class A Bézier Curves. *Computer Aided Geometric Design* 23, pp. 573-581.
- [48] **Ch'ng C K, Chan C S & Liu C L** (2020) Total-Text: toward orientation robustness in scene text detection. *IJDAR* 23, 31–52.
- [49] **Tetik Y E ve Bolat B** (2011) Gürültülü Ortamlarda Konuşma Tespiti İçin Yeni bir Öznitelik Çıkarım Yöntemi, *Elektrik-Elektronik ve Bilgisayar Sempozyumu*, Fırat Üniversitesi- Elazığ, 86-89.
- [50] **Sofu M M** (2019) Bozuk Kalıp Elemanın Tersine Mühendislik Uygulaması İle İmalatı, *Mühendislik Bilimleri ve Tasarım Dergisi* 7(2), 272 – 281, 2019 e-ISSN: 1308-6693.
- [51] **Nakaura T, Higaki T, Awai K and Ikeda O** (2020) A Primer For Understanding Radiology Articles About Machine Learning And Deep Learning. *Diagnostic and Interventional Imaging*, Volume 101, Issue 12, Pages 765-770.
- [52] **Sealfon R S G, Mariani L H, Kretzler M and Troyanskaya O G** (2020) Machine Learning, The Kidney And Genotype–Phenotype Analysis. *Kidney International*, Volume 97, Issue 6, June 2020, pages 1141-1149.
- [53] **Bulut V** (2016) Bilgisayar Destekli Geometrik Tasarım ve Uygulamaları. *Doktora Tezi*, Ege Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Anabilim Dalı, sf. 174.
- [54] **CH Chu, CCL Wang, and CR Tsai** (2008) Computer aided geometric design of strip using developable Bézier patches. *Computers in Industry*, Elsevier, Volume 59, pp. 601-611.

KAYNAKLAR (devam ediyor)

- [55] Yang L and Zeng X M (2009) Bézier curves and surfaces with shape parameters. *Int. J. Comput. Math.*, 86 (7) 1253–1263.
- [56] Ibrahim A and Albergail N (2019) The Cubic Bezier-Ball Like curve with Shape Parameters. *Journal of Alasmarya University: Basic and Applied Sciences*, ISSN: 2706-9524 (Print), 4(1), 12 s.
- [57] Raad A, Ayache M, Abboud, Permezel A, Merzouki R and Lartigau E (2014) Target Evolution Modeling for Robotized Adaptive Radiotherapy. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, Besançon, France, pp 118-124.
- [58] Özmen G (2012) Kübik Bézier Eğrileri ile Yüz İfadesi Tanıma. *Yüksek Lisans Tezi*, T.C. Trakya Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Ana Bilim Dalı, Edirne, sf. 56.
- [59] Yılmaz Luzum C (2018) R^3 'deki Yüzey Eğrilerinin Bézier Eğrileri ve Matlab Uygulamaları. *Yüksek Lisans Tezi*, Van Yüzüncü Yıl Üniversitesi, Fen Bilimleri Enstitüsü, Van, sf. 66.
- [60] Erkan E (2019) Öklid Düzleminde ve Öklid Uzayında Bézier Eğrileri. *Doktora Tezi*, T.C. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Anabilim Dalı, İstanbul, Sf. 163.
- [61] Kılıçoğlu Ş and Şenyurt S (2019) On The Cubic Bézier Curves In $\mathbb{E}^3$, *Ordu Üniversitesi Bilim ve Teknoloji Dergisi*, 9 (2): 83-97, e-ISSN: 2146-6459.
- [62] Nennioğlu A K ve Köroğlu T (2018) Otonom Araçlarda Hareket Planlaması, *Artibilim: Adana Bilim ve Teknoloji Üniversitesi Fen Bilimleri Dergisi*, 1(2), pp. 20-29.
- [63] Vlachkova K and Boikova (2011) Interactive 3D Visualization Of Bézier Curves Using Java Open Graphics Library (JOGL). *Serdica J. Computing*, PP. 323-332.
- [64] Kawabata K, Ma Liang, Xue J, Zhu C and Zheng N (2015) A path generation for automated vehicle based on Bézier curve and via-points. *Robotics and Autonomous Systems*, 74 (2015) 243–252.
- [65] Karlığa B (2002) *Düzlem Analitik Geometri*. Ankara.
- [66] Hacısalihoğlu H H (1993) *Diferansiyel Geometri*. Ankara Üniversitesi Yayınevi.
- [67] Levent A (2019) Bilgisayar Destekli Geometrik Tasarımda Geçiş Eğrileri. *Doktora Tezi*, T. C. İnönü Üniversitesi Fen Bilimleri Enstitüsü Matematik Anabilim Dalı, 77 s.
- [68] Gray A, Abbena E and Salamon S (2016) *Modern Differential Geometry of Curves and Surfaces with Mathematica*, USA.
- [69] Blaga P A (2005) *Lectures on the Differential Geometry of Curves and Surfaces*.

KAYNAKLAR (devam ediyor)

- [70] **Laurent P J and Sablonnoiere P** (2001) Pierre Bézier: An Engineer and a Mathematician. *Computer Aided Geometric Design*, 18: 609-617.
- [71] **Rabut C** (2002) On Pierre Bézier's life and motivations. *Computer-Aided Design*, 34: 493-510.
- [72] **Rogers D F** (2002) Pierre Etienne Bézier (1910-1999). *Computer-Aided Design*, 34: 489-491.
- [73] <<https://sogrekci.com/ders-notu/python-ve-bilimsel-hesaplama/42-grafik-cizimi/>>, Ziyaret Tarihi: 01.06.2023.
- [74] <<https://pybilim.wordpress.com/2014/01/27/matplotlib-3-bitisik-sekiller-yazi-ve-diger-suslemeler/>>, Ziyaret tarihi: 01.06.2023.
- [75] **Salamon D** (2006) *Curves and Surfaces for Computer Graphics*. Springer Science+Business Media, USA.
- [76] <<https://evrimagaci.org/lorentzminkowski-uzayinda-egri-ve-yuzeylerin-incelenmesi-9133>>, Ziyaret tarihi: 28.01.2024.
- [77] **H. H. Uğurlu ve A. Çalışkan** (2012) *Darboux Ani Dönme Vektörleri ile Spacelike ve Timelike Yüzeyler Geometrisi* Celal Bayar Üniversitesi Yayınları, Manisa.
- [78] **Çelik S** (2017) Bilgisayar Destekli Geometrik Tasarım Elemanlarından Bézier Eğrileri ve Yüzeylerinin Minkowski Uzayında İncelenmesi. *Yüksek Lisans Tezi*, Bitlis Eren Üniversitesi Fen Bilimleri Enstitüsü, Bitlis, sf. 152.
- [79] **Turgut A** (1995) Üç Boyutlu Minkowski Uzayında Spacelike ve Timelike Regle Yüzeyler. *Doktora Tezi*, Ankara Üniversitesi Fen Bilimleri Enstitüsü, Ankara, sf. 104.
- [80] **R Lopez** (2014) *Differential Geometry of Curves and Surfaces in Lorents-Minkowski space*, 64 s.
- [81] **Yılmaz Ceylan A** (2023) The Geometry of Bézier Curves in Minkowski 3-Space. *Universal Journal of Mathematics and Applications*, 6(1): 7-14.
- [82] **Shao L And Zhou H** (1996) Curve Fitting With Bézier Cubics. *Graphical Models And Image Processing* Vol. 58, No. 3, May, Pp. 223–232.
- [83] **Zain SAAASM, Misro MY and Miura KT** (2022) Curve Fitting Using Generalized Fractional Bézier Curve. *Computer-Aided Design and Applications*, 350-363.
- [84] **Kiusalaas J** (2013) *Interpolation and Curve Fitting In: Numerical Methods in Engineering with Python 3*. 1st edition, ISBN 978-1-107-03385-6, Cambridge University Press, United States of America, 438 pp/s.

KAYNAKLAR (devam ediyor)

- [85] **Yang Y and Gordon S P** (2014) Interpolation and Polynomial Curve Fitting. *The Mathematics Teacher*, 108(2):132-141.
- [86] **Sarfraz M and Razzak M F A** (2002) An algorithm for automatic capturing of the font outlines. *Computers & Graphics*, 26: 795–804.
- [87] **Li H (PhD)** (2023) *Curve Fitting and Interpolation. In: Numerical Methods Using Kotlin*. Apress. Berkeley, CA. Online ISBN 978-1-4842-8826-9, pp 169-196.
- [88] <<https://aws.amazon.com/tr/compare/the-difference-between-artificial-intelligence-and-machine-learning/#:~:text=tahminleri%20dahil%20olabilir,-,Temel%20farklar%3A%20yapay%20zeka%20ve%20makine%20%C3%B6%C4%9Frenimi,%C3%A7e%C5%9Fitli%20stratejiler%20ve%20teknolojiler%20i%C3%A7erir>>, Ziyaret tarihi: 09.10.2023.
- [89] <https://primer-computational-mathematics.github.io/book/c_mathematics/numerical_methods/1_Interpolation_curve_fitting.html#interpolation-and-curve-fitting>, Ziyaret tarihi: 15.12.202.
- [90] <<https://pypi.org/project/beziers/>>, Ziyaret tarihi: 26.09.2022.
- [91] <https://github.com/NikolaiT/CunningCaptcha/blob/master/python_tests/bezier.py.>, Ziyaret tarihi: 02.12.2023.
- [92] **Karateke S, Zontul M, Mishra V N et al.** (2024) On the Approximation by Stancu-Type Bivariate Jakimovski–Leviatan–Durrmeyer Operators. *La Matematica*.



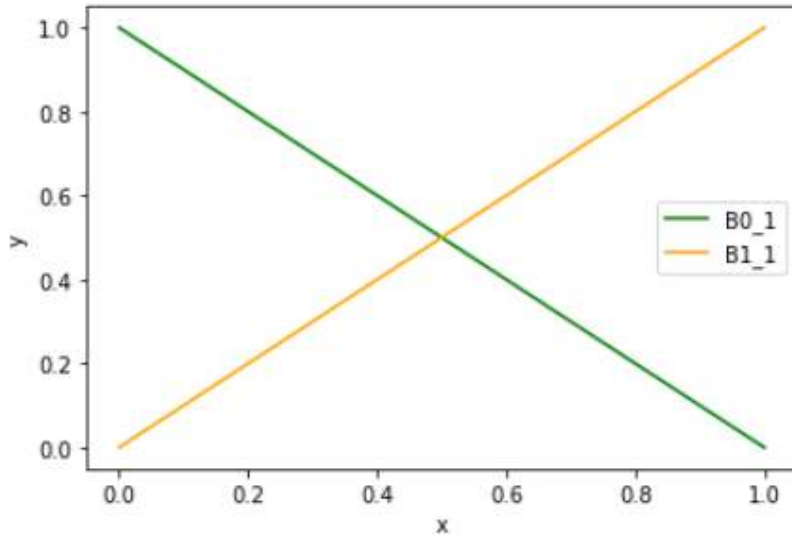
EK AÇIKLAMALAR

EK A: Şekil 2.1 Birinci dereceden Bernstein polinomunun Python 3.10 kodu ve grafiği:

```
In [79]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
#Bi_n=sum(i,n,x)(n,i)*(1-t)**(n-i)*t**i
x = np.linspace(0,1)
B0_1=1-x
plt.plot(x,B0_1,"g", label="B0_1")

y = np.linspace(0, 1)
B1_1=x
plt.plot(x,B1_1,"orange", label="B1_1")
#plt.plot(x,y,(B0_1,B1_1),"r", label="B0_1")

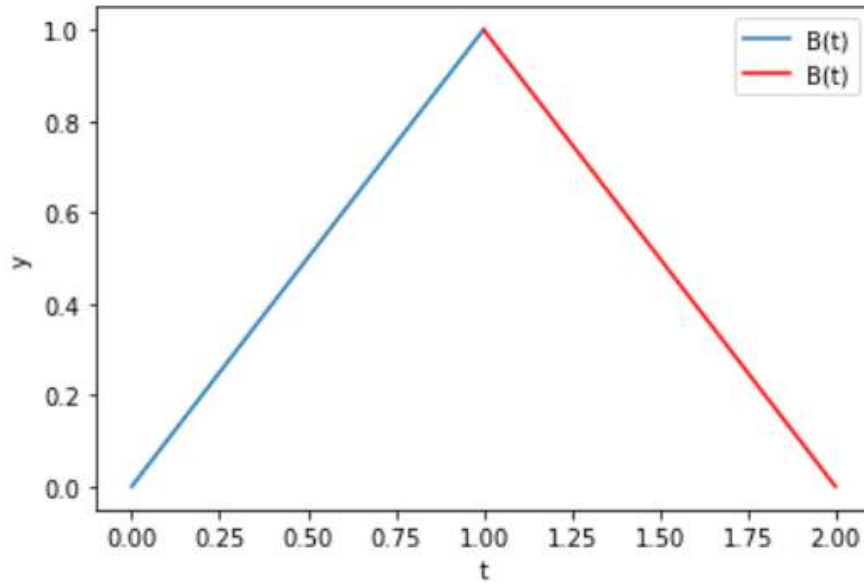
plt.xlabel("x")
plt.ylabel("y")
#plt.suptitle("Birinci dereceden Bernstein polinomu")
plt.legend(loc="best")
plt.show()
```



EK B: Şekil 2.3 $p_0 = (1,1), p_1 = (2,0)$ kontrol noktalarından geçen lineer Bézier eğrisinin Python 3.10 kodu ve grafiği:

```
In [47]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
#B(t)=sum(i,n,t)(n,i)*(1-t)**(n-i)*t**i*p_i
t = np.linspace(0,1)
y = np.linspace(0,1)
B0=1+t
B1=1-t
plt.plot(t,y,B0,B1,"r", label="B(t)")

plt.xlabel("t")
plt.ylabel("y")
#plt.suptitle("İki boyutlu kübik bezier eğrisi")
plt.legend(loc="best")
plt.show()
```



EK C: Şekil 2.5 İkinci dereceden Bernstein polinomunun Python 3.10 kodu ve grafiği:

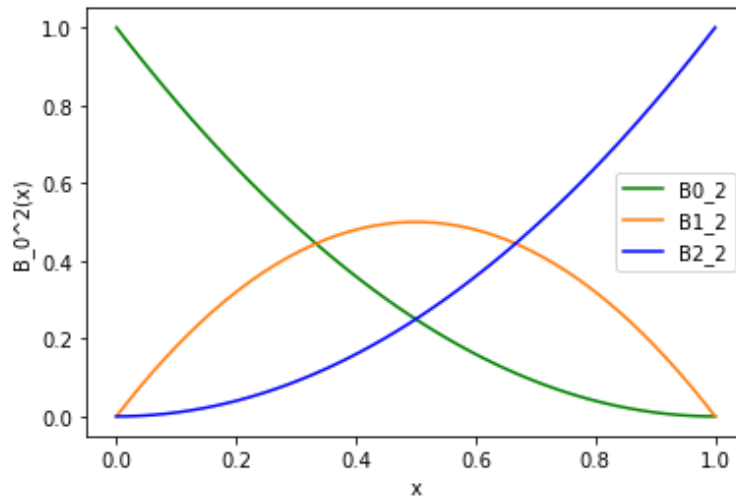
```
In [5]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
#Bi_n=sum(i,n,x)(n,i)*(1-t)**(n-i)*t**i

x = np.linspace(0, 1)
B0_2=(1-x)**2
plt.plot(x,B0_2,"g", label="B0_2")

x = np.linspace(0, 1)
B1_2=2*(1-x)*x
plt.plot(x,B1_2,"tab:orange", label="B1_2")

x = np.linspace(0, 1)
B2_2=x**2
plt.plot(x,B2_2,"b", label="B2_2")

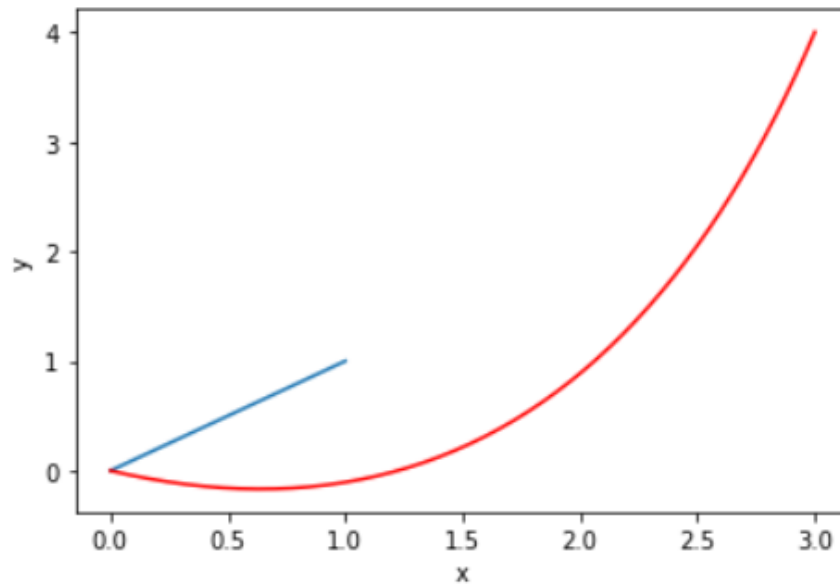
plt.xlabel("x")
plt.ylabel("B_0^2(x)")
#plt.suptitle("The Bernstein Polynomial: The quadratic case")
plt.legend(loc="best")
plt.show()
```



EK D: Şekil 2.6 $p_0 = (0,0)$, $p_1 = (2,-1)$, $p_2 = (3,4)$ kontrol noktalarının oluşturduğu düzlemsel kuadratik Bézier eğrisinin Python 3.10 kodu ve grafiği:

```
In [51]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
#B(t)=sum(i,n,x)(n,i)*(1-t)**(n-i)*t**i*p_i
x = np.linspace(0,1)
y = np.linspace(0,1)
B0=4*x-x**2
B1=6*x**2-2*x
plt.plot(x,y,B0,B1,"r", label="B(x)")

plt.xlabel("x")
plt.ylabel("y")
#plt.suptitle("İki boyutlu kübik bezier eğrisi")
#plt.legend(loc="best")
plt.show()
```



EK E: Şekil 2.7 Üçüncü dereceden Bernstein polinomunun Python 3.10 kodu ve grafiği:

```
In [93]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
#Bi_n=sum(i,n,x)(n,i)*(1-t)**(n-i)*t**i

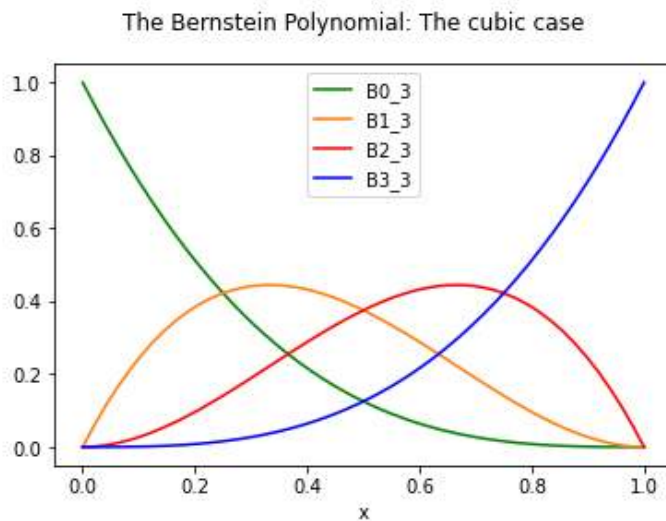
x = np.linspace(0, 1)
B0_3=(1-x)**3
plt.plot(x,B0_3,"g", label="B0_3")

x = np.linspace(0, 1)
B1_3=3*(1-x)**2*x
plt.plot(x,B1_3,"tab:orange", label="B1_3")

x = np.linspace(0, 1)
B2_3=3*(1-x)*x**2
plt.plot(x,B2_3,"r", label="B2_3")

x = np.linspace(0, 1)
B3_3=x**3
plt.plot(x,B3_3,"b", label="B3_3")

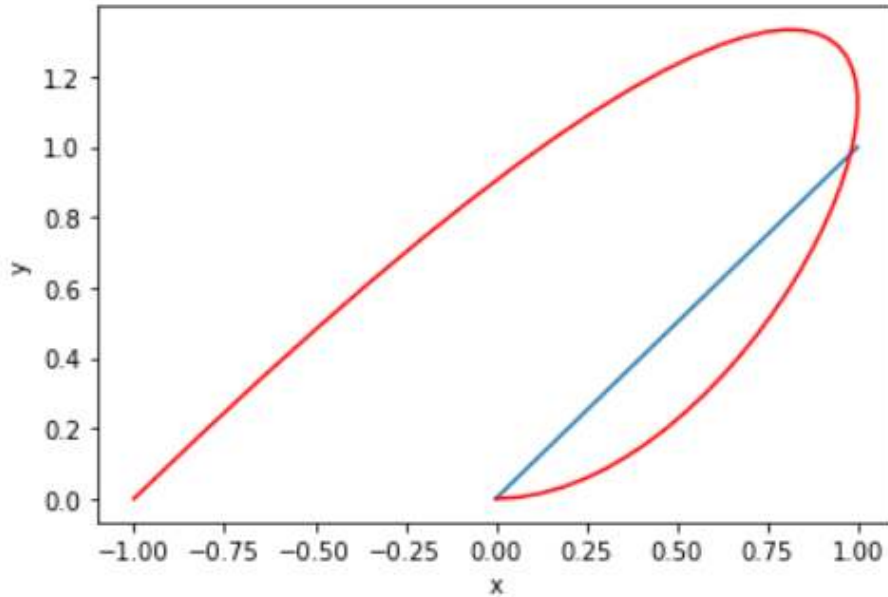
plt.xlabel("x")
#plt.ylabel("B_0^3(x)")
plt.suptitle("The Bernstein Polynomial: The cubic case")
plt.legend(loc="best")
plt.show()
```



EK F: Şekil 2.9 $p_0 = (0,0), p_1 = (1,0), p_2 = (2,3), p_3 = (-1,0)$ kontrol noktalarının oluşturduğu düzlemsel kuadratik Bézier eğrisinin Python 3.10 kodu ve grafiği:

```
In [36]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
#B(t)=sum(i,n,x)(n,i)*(1-t)**(n-i)*t**i*p_i
x = np.linspace(0,1)
y = np.linspace(0,1)
B0=3*x-4*x**3
B1=9*x**2-9*x**3
plt.plot(x,y,B0,B1,"r", label="B(x)")

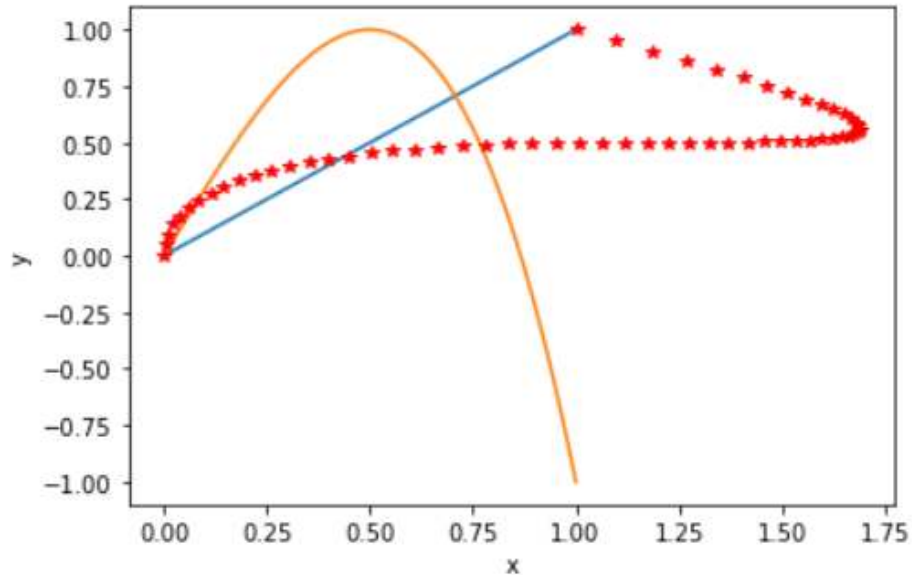
plt.xlabel("x")
plt.ylabel("y")
#plt.suptitle("İki boyutlu kübik bezier eğrisi")
#plt.legend(loc="best")
plt.show()
```



EK G: Şekil 2.10 $p_0 = (0,0,0), p_1 = (1,0,1), p_2 = (2,3,0), p_3 = (-1,1,1)$ kontrol noktalarının oluşturduğu düzlemsel kuadratik Bézier eğrisinin Python 3.10 kodu ve grafiği:

```
In [37]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
#B(t)=sum(i,n,x)(n,i)*(1-t)**(n-i)*t**i*p_i
x = np.linspace(0,1,60)
y = np.linspace(0,1,60)
z = np.linspace(0,1,60)
B0=3*x-4*x**3 #turuncu
B1=9*x**2-8*x**3 #kırmızı
B2=4*x**3-6*x**2+3*x #mavi
plt.plot(x,y,z,B0,B1,B2,"r*", label="B(x)")

plt.xlabel("x")
plt.ylabel("y")
#plt.suptitle("üç boyutlu kübik bezier eğrisi")
#plt.legend(loc="best")
plt.show()
```





ÖZGEÇMİŞ

Rumeysa AKALİN, ilk ve orta öğrenimini İstanbul’da tamamladı. 2013 yılında Zonguldak Bülent Ecevit Üniversitesi’nde lisans eğitimine başladı ve 2017 yılında Matematik bölümünü 3. olarak bitirdi. 2017 yılında yine Zonguldak Bülent Ecevit Üniversitesi, Ereğli Eğitim Fakültesi’nden Pedagojik Formasyon Sertifikasını aldı. 2021 yılında Zonguldak Bülent Ecevit Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Anabilim Dalı programında yüksek lisans programına başladı. Aynı zamanda Anadolu Üniversitesi’nde Yönetim Bilişim Sistemleri lisans eğitimine devam etmektedir.