

**T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**DİFERENSİYEL DENKLEMLERİN DİFERENSİYEL
KARESELLEŞTİRME METODUYLA ÇÖZÜMÜ**

GİZEM TEMELCAN

**YÜKSEK LİSANS TEZİ
MATEMATİK MÜHENDİSLİĞİ ANABİLİM DALI
MATEMATİK MÜHENDİSLİĞİ PROGRAMI**

**DANIŞMAN
PROF. DR. MUSTAFA SİVRİ**

İSTANBUL, 2013

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DİFERENSİYEL DENKLEMLERİN DİFERENSİYEL
KARESELLEŞTİRME METODUYLA ÇÖZÜMÜ

Gizem TEMELCAN tarafından hazırlanan tez çalışması 08.07.2013 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Matematik Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Prof. Dr. Mustafa SİVRİ
Yıldız Teknik Üniversitesi

Eş Danışman

Yrd. Doç. Dr. Ali ŞAHİN
Fatih Üniversitesi

Jüri Üyeleri

Prof. Dr. Mustafa SİVRİ
Yıldız Teknik Üniversitesi

Yrd. Doç. Dr. Ali ŞAHİN
Fatih Üniversitesi

Prof. Dr. Mustafa BAYRAM
Yıldız Teknik Üniversitesi

Prof. Dr. Müfit GİRESUNLU
İstanbul Üniversitesi

Doç. Dr. İbrahim KARATAY
Fatih Üniversitesi

ÖNSÖZ

Tez çalışmamım hazırlanmasında sayın hocam Prof. Dr. Mustafa Sivri' ye, yüksek lisans çalışmalarım boyunca bana destek olan, anlayış gösteren ve makaleleri temin etmemi sağlayan sayın hocam Yrd. Doç. Dr. Ali Şahin' e teşekkürlerimi sunarım.

Ayrıca her zaman maddi ve manevi desteğini esirgemeyen aileme ve arkadaşlarıma teşekkür ederim.

Temmuz, 2013

Gizem TEMELCAN

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	vii
KISALTMA LİSTESİ	viii
ŞEKİL LİSTESİ	ix
ÇİZELGE LİSTESİ	x
ÖZET	xii
ABSTRACT	xiii
BÖLÜM 1	
GİRİŞ	1
1.1 Literatür Özeti	1
1.2 Tezin Amacı	4
1.3 Hipotez	4
BÖLÜM 2	
DİFERENSİYEL KARESELLEŞTİRME METODUNDA KULLANILAN TEMEL KAVRAMLAR	5
2.1 Diferensiyel Kareselleştirme Metodu ile Türevsel Yaklaşım	5
2.1.1 İntegral Kareselleştirme	5
2.1.2 Diferensiyel Kareselleştirme	7
2.2 Lineer Vektör Uzayları	8
2.2.1 Lineer Vektör Uzayı Özellikleri	10
2.3 Yüksek Mertebeden Polinomal Yaklaşım	13
BÖLÜM 3	
POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME METODU ..	19
3.1 Birinci Mertebeden Türevler için Ağırlık Katsayılarının Hesaplanması ..	20
3.1.1 Bellman Yaklaşımları	21
3.1.1.1 Bellman'ın Birinci Yaklaşımı	21
3.1.1.2 Bellman'ın İkinci Yaklaşımı	23

3.1.2	Quan ve Chang Yaklaşımı	26
3.1.3	Shu Genel Yaklaşımı	28
3.2	İkinci ve Yüksek Mertebeden Türevler için Ağırlık Katsayılarının Hesaplanması	34
3.2.1	İkinci Mertebeden Türev için Ağırlık Katsayıları	34
3.2.1.1	Quan ve Chang Yaklaşımı	34
3.2.1.2	Shu Genel Yaklaşımı	35
3.2.2	Yüksek Mertebeden Türev için Shu Rekürans Formülü	37
3.3	Matris Çarpımı Yaklaşımı	40
3.4	Hata Analizi	41
3.4.1	Fonksiyon Yaklaşımı	41
3.4.2	Türev Yaklaşımı	45
3.5	Grid Nokta Dağılımları için Belirlenen Seçimler	49
3.5.1	Düzgün (Uniform) Grid Nokta Seçimi	49
3.5.2	Düzgün Olmayan (Non-Uniform) Grid Nokta Seçimi	50
3.5.3	Chebyshev-Gauss-Lobatto Grid Nokta Seçimi	51

BÖLÜM 4

ADİ DİFERENSİYEL DENKLEMLERİN POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME METODU İLE ÇÖZÜMÜ

4.1	Birinci Mertebeden Değişken Katsayılı Diferensiyel Denklemlerin PDK Metodu ile Çözümü	52
4.2	İkinci Mertebeden Değişken Katsayılı Sınır Değer Probleminin PDK Metodu ile Çözümü	63
4.3	İkinci Mertebeden Değişken Katsayılı Başlangıç Değer Probleminin PDK Metodu ile Çözümü	67
4.4	Yüksek Mertebeden Değişken Katsayılı Diferansiyel Denklemlerin PDK Metodu ile Çözümü	72

BÖLÜM 5

POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME METODU VE SONLU FARK METODU ARASINDAKİ İLİŞKİ

5.1	Sonlu Fark Metodunun Oluşumu	77
5.1.1	Sonlu Fark Formülleri	79
5.1.1.1	Birinci Mertebeden Türev için Yaklaşımlar	79
5.1.1.2	İkinci Mertebeden Türev için Yaklaşımlar	80
5.2	Polinom Tabanlı Diferensiyel Kareselleştirme ve Yüksek Mertebeden Sonlu Fark Metodu Arasındaki İlişki	80

BÖLÜM 6

KİSMİ TÜREVLİ DİFERENSİYEL DENKLEMLERİN POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME METODU İLE ÇÖZÜMÜ

6.1	Dirichlet Sınır Koşulları Altındaki Homojen Parabolik (Isı) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü	87
6.2	Dirichlet Sınır Koşulları Altındaki Homojen Olmayan Parabolik (Isı) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü	90

6.3	Neumann Sınır Koşulları Altındaki Homojen Parabolik (Isı) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü	93
6.4	Dirichlet Sınır Koşulları Altındaki Homojen Hiperbolik (Dalga) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü	96
6.5	Başlangıç Koşulları Altındaki Homojen Olmayan Hiperbolik (Dalga) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü	100

BÖLÜM 7

SONUÇ VE ÖNERİLER.....	103
KAYNAKLAR.....	104
EK-A	
ADİ DİFERENSİYEL DENKLEMİN ÇÖZÜMÜNE AİT İŞLEM BASAMAKLARI	106
ÖZGEÇMİŞ	116

SİMGE LİSTESİ

a_{ij}	Birinci mertebeden türevin ağırlık katsayıları
$ E $	Mutlak Hata
$E(f)$	Hata yaklaşım fonksiyonu
$E_D^{(m)}(f)$	m .mertebeden türevin hata yaklaşım fonksiyonu
I	Birim Matris
$K_N(x)$	N . dereceden ötelenmiş Legendre polinomu
$L(x)$	Lineer operatör
$L_N(x)$	N . dereceden Legendre polinomu
$r_k(x)$	Taban polinomu
$\mathbf{V}_N$	N boyutlu lineer vektör uzayı
$w_{ij}^{(m)}$	m . mertebeden türevin ağırlık katsayıları
δ_{ij}	Kronecker operatörü

KISALTMA LİSTESİ

CGL	Chebyshev Gauss Lobatto
DKM	Diferensiyel Kareselleştirme Metodu
PDKM	Polinom Tabanlı Diferensiyel Kareselleştirme Metodu

ŞEKİL LİSTESİ

	Sayfa
Şekil 2. 1 $f(x)$ fonksiyonunun $[a,b]$ aralığındaki belirli integrali.....	6
Şekil 3. 1 Düzgün dağılım ile $N=10$ nokta için grid noktaların dağılımının gösterimi	49
Şekil 3. 2 Düzgün olmayan dağılım ile $N=10$ nokta için grid noktaların dağılımının gösterimi	50
Şekil 3. 3 CGL dağılımı ile $N=10$ nokta için grid noktaların dağılımının gösterimi...	51

ÇİZELGE LİSTESİ

	Sayfa
Çizelge 3. 1 Düzgün dağılım ile $N=10$ nokta için grid noktaların dağılımı.....	49
Çizelge 3. 2 Düzgün olmayan dağılım ile $N=10$ nokta için grid noktaların dağılım .	50
Çizelge 3. 3 CGL dağılımı ile $N=10$ nokta için grid noktaların dağılımı	51
Çizelge 4. 1 Düzgün dağılım ile $N=5$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	57
Çizelge 4. 2 Düzgün dağılım ile $N=9$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	58
Çizelge 4. 3 Düzgün dağılım ile $N=13$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	59
Çizelge 4. 4 Düzgün dağılım ile farklı N değerlerindeki ortak x noktalarının kesin ve yaklaşık değerleri	59
Çizelge 4. 5 Düzgün olmayan dağılım ile $N=5$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	60
Çizelge 4. 6 Düzgün olmayan dağılım ile $N=13$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	61
Çizelge 4. 7 CGL dağılımı ile $N=5$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	62
Çizelge 4. 8 CGL dağılımı ile $N=13$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	62
Çizelge 4. 9 Düzgün dağılım ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	66
Çizelge 4. 10 Düzgün olmayan dağılım ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	66
Çizelge 4. 11 CGL dağılımı ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	67
Çizelge 4. 12 Düzgün dağılım ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	70
Çizelge 4. 13 Düzgün olmayan dağılım ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	71
Çizelge 4. 14 CGL dağılımı ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	71
Çizelge 4. 15 Düzgün dağılım ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	75
Çizelge 4. 16 Düzgün olmayan dağılım ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	75
Çizelge 4. 17 CGL dağılımı ile $N=10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar	76

Çizelge 6. 1	Örnek 6.1 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar	90
Çizelge 6. 2	Örnek 6.2 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar	93
Çizelge 6. 3	Örnek 6.3 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar	96
Çizelge 6. 4	Örnek 6.4 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar	99
Çizelge 6. 5	Örnek 6.5 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar	102

DİFERENSİYEL DENKLEMLERİN DİFERENSİYEL KARESELLEŞTİRME METODUYLA ÇÖZÜMÜ

Gizem TEMELCAN

Matematik Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi

Tez Danışmanı: Prof. Dr. Mustafa SİVRİ
Eş Danışman: Yrd. Doç.Dr. Ali ŞAHİN

Bu çalışmada nümerik bir metot olan DKM incelenmiştir. Çalışmada lineer adi ve lineer kısmi diferensiyel denklemler DKM kullanılarak çözülmüştür.

Bu çalışma yedi bölümden oluşmaktadır. İlk bölüm giriş olarak ele alınmıştır. İkinci bölümde DKM'nin temel kavramları olan lineer vektör uzayı ve polinomal yaklaşım üzerinde durulmuştur. Üçüncü bölümde PDKM hakkında bilgi verilmektedir. Dördüncü bölümde lineer adi diferensiyel denklemler PDKM ile çözülmüş ve elde edilen değerler kesin değerlerle karşılaştırılmıştır. Beşinci bölümde PDKM'nin Sonlu Fark Metodu ile arasındaki ilişkiden söz edilmiştir. Altıncı bölümde lineer kısmi türevli diferensiyel denklemler PDKM ile çözülmüş ve elde edilen değerler kesin değerler ile karşılaştırılmıştır. Son bölümde ise sonuçlara ve önerilere yer verilmiştir.

Anahtar Kelimeler: Polinom Tabanlı Diferensiyel Kareselleştirme Metodu, Adi Diferensiyel Denklemlere Uygulama, Kısmi Türevli Diferensiyel Denklemlere Uygulama

ABSTRACT

SOLUTION OF DIFFERENTIAL EQUATIONS USING DIFFERENTIAL QUADRATURE METHOD

Gizem TEMELCAN

Department of Mathematical Engineering

MSc. Thesis

Advisor: Prof. Dr. Mustafa SİVRİ

Co-Advisor: Assist. Prof. Dr. Ali ŞAHİN

This thesis presents one of the numerical discretization technique which is known as Differential Quadrature Method (DQM). In this thesis, linear ordinary and linear partial differential equation problems were solved by using DQM. In addition to DQM, Finite Difference Method (FDM) was also investigated for comparing DQM and FDM with each other. This thesis consists of seven chapters. First chapter is named as introduction. In the second chapter, mathematical fundamentals of DQM, which are linear vector space analysis and function approximation, were given. Third chapter gives information about Polynomial-based Differential Quadrature Method (PDQM). In the fourth chapter, PDQM was applied to linear ordinary differential equations and approximate solutions obtained were compared with the exact solutions. In the fifth chapter the relationship between PDQM and FDM were discussed. In the sixth chapter, PDQM was applied to linear partial differential equations and approximate solutions obtained were compared with the exact solutions. Finally, in the seventh chapter results obtained were discussed.

Keywords: Polynomial-based Differential Quadrature Method, Application to the Ordinary Differential Equations, Application to the Partial Differential Equations.

YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

1.1 Literatürün Özeti

DKM, yeterince düzgün fonksiyonların türev değerlerinin hesaplanmasında kullanılan nümerik bir metottur. Bu metot 1971 yılında Bellman ve Casti [1] tarafından Gauss Kareselleştirme İntegrasyon Metodu esas alınarak geliştirilmiştir. Gauss Kareselleştirme Metodu, gauss noktalarındaki integrant değerlerinin ağırlıklı toplamlarıyla yakınsayan bir integral elde edilerek oluşturulmuştur. Bu metot geliştirilerek, yeterince düzgün fonksiyonların herhangi mertebeden türev değerlerinin hesaplanmasında kullanılarak DKM elde edilmiştir. Buna göre DKM, düzgün bir fonksiyonunun türevinin herhangi bir grid noktadaki değerine tüm grid noktadaki fonksiyon değerlerinin ağırlıklı lineer toplamı ile yakınsamasıdır.

Sonlu Fark Metodu ve Sonlu Elemanlar Metoduna göre daha az grid nokta kullanılarak daha yakınsak sonuçlar elde edilebilen DKM lineer ve/veya lineer olmayan başlangıç ve/veya sınır değer problemleri için kullanılmaktadır [2-3]. DKM, lineer vektör uzayı analizine ve fonksiyon yaklaşımına dayanmaktadır. PDKM’de kısmi türevli diferensiyel denklemin çözümüne yüksek dereceden bir polinomla yaklaşılmaktadır. Bu yaklaşım polinomu ile birlikte polinomun ağırlık katsayıları da belirlenmektedir. Böylece belirlenen ağırlık katsayıları ile yaklaşım polinomu yazılabilmekte ve kısmi türevli diferensiyel denklemin çözümüne yakın bir çözüm polinomu elde edilebilmektedir.

PDKM’nin mühendislikteki ilk uygulamalarında ağırlık katsayılarını belirlemek üzere Bellman’ın birinci yaklaşımı kullanılmıştır. Bu yaklaşımda grid noktalar keyfi olarak seçilebilmektedir. Oluşan lineer denklem sisteminin tek bir çözümü vardır ki bu çözüme ait katsayılar matrisi Vandermonde matrisi yapısındadır. Ancak grid noktaların sayısı arttığında oluşan katsayılar matrisi istenilen sonucu veremeyecektir. Bu yüzden

Bellman'ın birinci yaklaşımı grid nokta sayısı en fazla 13 iken uygulanmaktadır. Bellman'ın ikinci yaklaşımında ise test fonksiyonu olarak ötelenmiş Legendre polinomları, grid noktalar için ise ötelenmiş Legendre polinomlarının kökleri kullanılmaktadır. Fakat grid noktaların seçimindeki kısıtlama nedeniyle bu yaklaşım pek tercih edilmemektedir. Bu yüzden pratik uygulamalarda Bellman'ın birinci yaklaşımı daha fazla tercih edilmektedir [4].

Grid noktaların sayısı arttığında ağırlık katsayılarını belirlemek zorlaştığından bu zorluğu ortadan kaldırmak üzere birçok çalışma yapılmıştır. Quan ve Chang [5-6], Lagrange interpolasyon polinomunu test fonksiyonu olarak alıp birinci ve ikinci mertebeden türevler için ağırlık katsayılarını hesaplayan formüller geliştirmişlerdir. Shu ve Richards [7] PDKM'de ağırlık katsayılarını hesaplayan tüm yaklaşımların lineer vektör uzayındaki taban vektörlerin seçiminden oluşturulabileceğini bulmuşlardır. Diğer bir ifadeyle, lineer vektör uzayındaki taban vektörlerinin her birinin seçimiyle ağırlık katsayılarını hesaplamakta birçok yol bulunmaktadır. Bu farklı yollarla hesaplanan ağırlık katsayıları sonuçta aynı değerleri vermektedir. Bu fikir, lineer vektör uzayı özelliklerinden yararlanılarak ortaya atılmıştır ve oldukça kullanışlıdır.

Yücel ve Sarı, DKM'yi lineer ve lineer olmayan yapıdaki tekil, iki noktalı sınır değer problemleri için incelemişlerdir. Çalışmalarının sonucunda DKM'nin minimum işlem gücü kullanılarak yakınsak sonuçlar verdiği görülmüştür [8]. Yücel, polinom tabanlı ve Fourier açılımı tabanlı DK metotlarını kullanarak özel bir başlangıç-sınır değer problemi olan Sturm-Liouville probleminin öz değerlerini hesaplamıştır. Periyodik durum gösteren problemlerde Fourier açılımı tabanlı DKM, PDKM'ye göre daha iyi sonuç verdiği için bu çalışmada her ikisi bir arada kullanılmıştır [9].

Girgin, lineer olmayan adi diferensiyel denklemleri çözmek üzere DKM ile Simülasyon Tekniği kullanılan yaklaşımları (Matlab/Simulink ve benzeri) karşılaştırarak yeni bir metot geliştirmiştir [10]. DKM ve Simülasyon Tekniğinin dezavantajlarının yok edilmesiyle oluşturulan bu metot Karşılaştırma Metodu olarak adlandırılmıştır. Bu metot kullanılarak dördüncü mertebeden sınır değer problemi, Duffing tipi başlangıç değer problemi, pendulum salınımı başlangıç değer problemi ve ikinci mertebeden başlangıç değer problemi çözülmüştür.

Gürarslan ve Sarı yaptıkları çalışmada lineer ve lineer olmayan difüzyon denklemlerinin kesin çözümlerini ele almışlardır [11]. Bu denklemleri çözebilmek üzere konum için

PDKM kullanılarak kısmi türevli diferansiyel denklem adi diferansiyel denklem sistemine indirgenmiş ve oluşan zamana bağlı adi diferansiyel denklem sistemi de Runge-Kutta Metodu ile çözülmüştür. Metodun doğruluğunu göstermek amacıyla elde edilen sonuçlar analitik çözümden elde edilen sonuçlarla karşılaştırılmıştır. Lineer olmayan denklemler ise dönüşüm uygulanmadan ve doğrusallaştırılmadan DKM ile çözülmüştür. Elde edilen sonuçlardan bu metodun güvenilir olduğu görülmüş ve gerçeğe yakın fiziksel modeller elde etmede diğer lineer olmayan kısmi türevli diferansiyel denklemlerde de kullanılabileceği gösterilmiştir.

Meral ve Tezer-Sezgin, kimya, biyoloji ve eczacılıkta önemli uygulamalarda kullanılan lineer olmayan bir ve iki boyutlu reaksiyon-difüzyon denklemini PDKM ile çözmüşlerdir [12]. Yine Meral, ayrıca lineer olmayan bir ve iki boyutlu ısı ve kütle transferi denklemlerini de bu metodu kullanarak çözmüştür [13]. Tanaka ve Chen de zamana bağlı difüzyon denklemlerinin çözümünde DKM'yi kullanmıştır [14]. Öğüt, uniform iki sabit ısı kaynağıyla ısıtılmış eğik kare bir kapalı bölgede doğal konveksiyonla ısı transferini DKM'yi kullanarak incelemiştir [15]. Korkmaz ve Dağ, lineer olmayan Burgers' denklemini PDKM ile çözümünü araştırmışlardır [16].

Ersoy ve diğerleri, hazırladıkları çalışmada DKM'nin titreşim problemlerine uygulanabilirliğini göstermişler ve metodu farklı problemlere uygulamışlardır. Burada problem olarak dairesel ve boşluklu dairesel zarların serbest titreşim hesabı seçilmiştir. Elde edilen sonuçlara göre düğüm nokta sayısının artması, sonucun hassasiyetini arttırmış ve eşit olmayan düğüm noktalarının sayısı ile DKM, eşit düğüm nokta dağılımı kullanılarak uygulanan DKM'ye göre daha iyi sonuç vermiştir [17]. Korkmaz, lineer olmayan Korteweg-de Vries (KdVE) dalga denklemini hem polinom tabanlı hem Fourier açılımı tabanlı DKM'leri kullanarak çözmüştür [18]. Böylece iki farklı (tabanlı) DKM'leri kendi aralarında karşılaştırmış ve her ikisinden de yakın sonuçlar elde etmiştir. Shu ve Xue, iki boyutlu Helmholtz denklemini polinom tabanlı ve Fourier açılımı tabanlı DKM ile çözmüştür [19]. Elde edilen sonuçlar değerlendirildiğinde, Helmholtz denkleminin çözümü için Fourier açılımı tabanlı DKM'nin daha uygun olacağı görülmüştür.

DKM, hidrolik mühendisliği alanında da kullanılmaktadır. Bu metot kullanılarak Kaya ve Ülke, taşkınların ötelenmesi üzerine bir çalışma yapmışlardır [20]. Taşkınların ötelenmesi olarak tanımlanan hesaplarda St. Venant denklemleri kullanılmaktadır. St. Venant denklemleri matematiksel olarak bazı kabullerle doğrusallaştırılabilmekte, farklı

dalga yaklaşımları kullanılsa da difüzyon dalga denklemi şeklinde ifade edilebilmektedir. Yapılan çalışmada difüzyon dalga denkleminin çözümünde PDKM kullanılmıştır. Elde edilen sonuçlar Sonlu Fark metotlarından elde edilen sonuçlarla karşılaştırılmıştır. Bu karşılaştırmada Sonlu Fark metotlarına nazaran daha iyi sonuç verdiği gözlemlenmiştir.

1.2 Tezin Amacı

Bu çalışmada, adi ve kısmi türevli diferensiyel denklemlerin nümerik bir metot olan DKM ile çözülmesi amaçlanmıştır.

1.3 Hipotez

Bu çalışmada, adi ve kısmi türevli diferensiyel denklemlerin nümerik çözümlerinde DKM'nin diğer nümerik metotlar gibi etkili bir metot olduğu; DKM'nin taban polinomlarının seçimine bağlı değil, grid noktaların dağılımına bağlı olduğu ve DKM ile Sonlu Fark Metodu arasındaki ilişki gösterilecektir.

DİFERENSİYEL KARESELLEŞTİRME METODUNDA KULLANILAN TEMEL KAVRAMLAR

DKM, türevlerin yaklaşımında kullanılan nümerik bir metottur. Bu metot, basit integral kareselleştirmeden yararlanılarak ortaya çıkarılmıştır.

Bu bölümde DKM’ de kullanılacak bazı tanımlamalara ve teoremlere yer verilecektir.

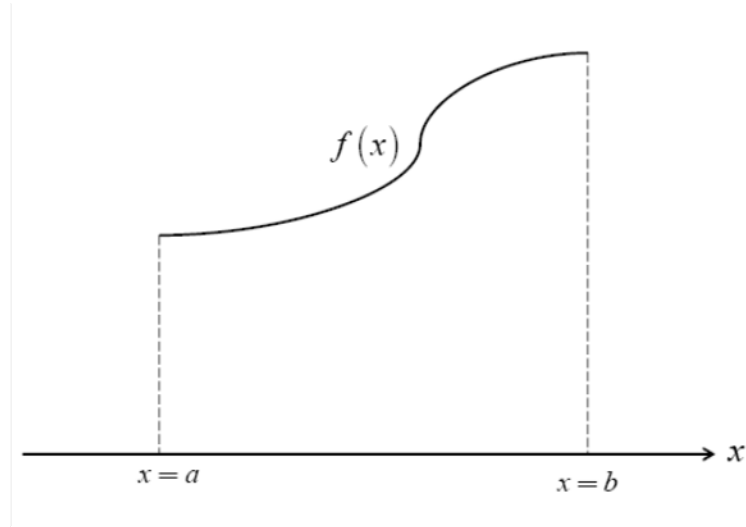
2.1 Diferensiyel Kareselleştirme Metodu ile Türevsel Yaklaşım

2.1.1 İntegral Kareselleştirme

$f(x)$ fonksiyonu, $[a, b]$ aralığında sürekli bir fonksiyon olsun. Belirli integral hesabın temel teoremine göre eğer $\frac{dF}{dx} = f$ olacak şekilde bir $F(x)$ fonksiyonu mevcut ise

$$\int_a^b f(x) dx = F(b) - F(a) \quad (2.1)$$

olacaktır. Eğer $f(x)$ fonksiyonunun değerleri ayrık kümelerden seçilmiş noktalarda tanımlanmış ise $F(x)$ fonksiyonu için kesin bir sonuç elde etmek zor olabilir. Böyle durumlarda nümerik yaklaşımlara gerek duyulmaktadır.



Şekil 2.1 $f(x)$ fonksiyonunun $[a, b]$ aralığındaki belirli integrali

Aynı zamanda $\int_a^b f(x)dx$ integrali, Şekil 2.1’ deki gibi $f(x)$ eğrisi altında kalan alanı göstermektedir. Böylece integralin değeri, eğri altında kalan alanın değerine eşit olacaktır. Bu özelliğe dayanarak birçok nümerik analiz teknikleri geliştirilmiştir.

$w_1, w_2, w_3, \dots, w_n$ ağırlık katsayıları, $a = x_1, x_2, x_3, \dots, x_n = b$ grid noktalar ve $f_1, f_2, \dots, f_n$ sırasıyla bu grid noktadaki fonksiyon değerleri olmak üzere $\int_a^b f(x)dx$ integrali

$$\int_a^b f(x)dx = w_1 f_1 + w_2 f_2 + \dots + w_n f_n = \sum_{k=1}^n w_k f_k \quad (2.2)$$

şeklinde hesaplanır. Bu eşitlik, integral kareselleştirme olarak adlandırılır. Buna göre integral kareselleştirme, sonlu aralıkta tanımlanan bir integrale yakınsamak için integralin tanım bölgesindeki tüm fonksiyon değerlerini kullanmaktadır.

Grid noktalar genellikle düzgün dağılım verecek şekilde seçilmektedir. Örneğin, h adım uzunluğu olmak üzere $i = 1, 2, 3, \dots, n$ için $x_i = x_{i-1} + h$ olsun. Buna göre tüm basit kareselleştirme formülleri, (2.2) eşitliğinin formunda yazılabilir. Aşağıda integral kareselleştirme eşitliğine ait iki özel durum verilmiştir:

Yamuk (Trapezoidal) Kuralı :

$$\int_a^b f(x)dx = \frac{h}{2} [f(a) + 2f(a+h) + \dots + 2f(a+(n-1)h) + f(b)] + E \quad (2.3)$$

şeklindedir. Bu integral yaklaşımındaki hata E ile gösterilmektedir. Buradaki ağırlık katsayıları

$$w_1 = \frac{h}{2}, w_2 = w_3 = \dots = w_{n-1} = h, w_n = \frac{h}{2}$$

olacak şekilde tanımlanır [21].

Simpson Kuralı :

$$\int_a^b f(x) dx = \frac{h}{3} [f_1 + 4f_2 + 2f_3 + 4f_4 + \dots + 4f_{n-2} + 2f_{n-1} + f_n] + E \quad (2.4)$$

şeklindedir. Bu integral yaklaşımındaki hata E ile gösterilmektedir. Buradaki ağırlık katsayıları da

$$w_k = \begin{cases} \frac{h}{3}, & k = 1, n \\ \frac{4h}{3}, & k = 2, 4, 6, \dots, n-2 \\ \frac{2h}{3}, & k = 3, 5, 7, \dots, n-1 \end{cases}$$

ile ifade edilir [21].

2.1.2 Diferensiyel Kareselleştirme

$f(x)$ düzgün bir fonksiyon ve x_i , karelere ayrılmış bölgedeki grid noktalar olsun. Buna göre, $f(x)$ fonksiyonunun x 'e göre birinci mertebeden türevinin x_i noktasındaki değeri, tanım bölgesindeki tüm fonksiyon değerlerinin lineer toplamına yakınsayacaktır. Dolayısıyla a_{ij} ağırlık katsayıları ve N tanım kümesinde karelere bölünmüş bölgedeki grid noktaların sayısı olmak üzere

$$f'(x_i) = \left. \frac{df}{dx} \right|_{x_i} = \sum_{j=1}^N a_{ij} f(x_j), i = 1, 2, \dots, N \quad (2.5)$$

eşitliği DKM olarak adlandırılır. Buradaki a_{ij} ağırlık katsayıları, x_i noktalarının farklı yerlerinde farklı değerler almaktadır. Böylece DKM' deki esas, a_{ij} ağırlık katsayılarının belirlenmesi olacaktır. Daha detaylı olarak DKM gelecek bölümde ele alınacaktır.

2.2 Lineer Vektör Uzayları

DKM ile ağırlık katsayıları hesaplanırken lineer vektör uzayından ve özelliklerinden faydalanılacaktır. Bu kısımda lineer vektör uzayı ve özellikleri ile ilgili tanımlar hatırlatılacaktır.

Tanım 2.1 F , skaler elemanlardan oluşan bir küme, “+” toplam ve “ $\times$ ” çarpım operatörü olsun. Aşağıdaki özellikleri sağlayan F kümesi bir cisimdir.

(1) Her $a, b \in F$ için $a + b \in F$ ve $a \times b \in F$ olur.

(2) Her $a, b \in F$ için toplama ve çarpma işlemlerinde $a + b = b + a$ ve $a \times b = b \times a$ şeklinde değişme özelliği mevcuttur.

(3) Her $a, b, c \in F$ için toplama ve çarpma işlemlerinde $(a + b) + c = a + (b + c)$ ve $(a \times b) \times c = a \times (b \times c)$ olacak şekilde birleşme özelliği mevcuttur.

(4) Her $a, b, c \in F$ için çarpma işleminin toplama işlemi üzerinde $a \times (b + c) = (a \times b) + (a \times c)$ şeklinde dağılma özelliği mevcuttur.

(5) Her $a \in F$ için $a + 0 = a$ olacak şekilde bir $0 \in F$ ve $1 \times a = a$ olacak şekilde bir $1 \in F$ mevcuttur.

(6) Her $a \in F$ için $a + b = 0$ olacak şekilde bir $b \in F$ mevcuttur.

(7) $a \neq 0$ olmak üzere her $a \in F$ için $a \times c = 1$ olacak şekilde bir $c \in F$ mevcuttur.

Bu özelliklere dayanarak DKM’deki a_{ij} ağırlık katsayıları, reel sayılarda bulunacaktır.

Tanım 2.2 V , elemanları vektör olan bir küme ve $F \subset V$ olsun. V kümesi vektör toplama ve skaler çarpım operatörlerine sahip ise aşağıdaki özellikleri sağlayan V uzayına lineer vektör uzayı denir.

(1) Her $\alpha, \beta \in V$ için $\alpha + \beta \in V$ olur.

(2) Her $\alpha, \beta \in V$ için toplama işleminin $\alpha + \beta = \beta + \alpha$ şeklinde değişme özelliği vardır.

(3) Her $\alpha, \beta, \gamma \in V$ için toplama işleminin $(\alpha + \beta) + \gamma = \alpha + (\beta + \gamma)$ şeklinde birleşme özelliği vardır.

- (4) Her $\alpha \in \mathbf{V}$ için $0 + \alpha = \alpha$ olacak şekilde bir $0 \in \mathbf{V}$ sıfır vektörü vardır.
- (5) Her $\alpha \in \mathbf{V}$ için $\alpha + \beta = 0$ olacak şekilde bir $\beta \in \mathbf{V}$ vardır.
- (6) Her $c \in \mathbf{F}$ ve her $\alpha \in \mathbf{V}$ için $c\alpha \in \mathbf{V}$ elemanı vardır ve bu elemana “ c ile α ’nın skaler çarpımı” adı verilir.
- (7) Her $a, b \in \mathbf{F}$ ve her $\gamma \in \mathbf{V}$ için skaler çarpımın $a(b\gamma) = (ab)\gamma$ şeklinde birleşme özelliği vardır.
- (8) Her $a \in \mathbf{F}$ ve her $\beta, \gamma \in \mathbf{V}$ için skaler çarpımın vektör toplamı üzerinde $a(\beta + \gamma) = a\beta + a\gamma$ şeklinde dağılma özelliği vardır.
- (9) Her $a, b \in \mathbf{F}$ ve her $\gamma \in \mathbf{V}$ için skaler çarpımın skalerlerin toplamı üzerine $(a + b)\gamma = a\gamma + b\gamma$ şeklinde dağılma özelliği vardır.
- (10) Her $\alpha \in \mathbf{V}$ için $1 \times a = a$ olacak şekilde $1 \in \mathbf{F}$ vardır.

$P_n(x)$, derecesi n ’den küçük ve c_i reel katsayılı tüm polinomların bir kümesi olsun. Bu tanımlara ve özelliklere dayanarak $P_n(x)$ polinomlar kümesinin bir vektör uzayı olduğu gösterilebilir.

$$P_n(x) = \sum_{i=0}^{n-1} c_i x^i \quad (2.6)$$

olmak üzere vektör toplamı ve skaler çarpım

$$\sum_{i=0}^{n-1} c_i x^i + \sum_{i=0}^{n-1} d_i x^i = \sum_{i=0}^{n-1} (c_i + d_i) x^i \quad (2.7)$$

$$a \left(\sum_{i=0}^{n-1} c_i x^i \right) = \sum_{i=0}^{n-1} (ac_i) x^i \quad (2.8)$$

şeklinde tanımlanırsa $P_n(x)$ kümesinin bir lineer vektör uzayı olduğu gösterilebilir.

2.2.1 Lineer Vektör Uzayı Özellikleri

Tanım 2.3 (Lineer Bağımsızlık) $c_i \in \mathbb{R}$ ve $\alpha_i \in V$ olsun. Eğer her $c_i = 0$ için

$$\sum_{i=1}^n c_i \alpha_i = 0 \quad (2.9)$$

oluyorsa α_i vektörleri lineer bağımsızdır. Lineer bağımsızlık sadece vektörler kümesine veya uzayına bağlı değildir. Eğer α_i vektörleri lineer bağımlı ise bu durumda vektörlerden en az biri, diğerlerinin lineer kombinezonu şeklinde yazılabilmelidir.

Tanım 2.4 (Lineer Vektör Uzayının Boyutu) V lineer vektör uzayı ve S, V 'nin bir alt kümesi olmak üzere

- S lineer bağımsız,
- Her $\alpha \in V$, S 'deki sonlu sayıda elemanın bir lineer birleşimi

ise S, V vektör uzayının bir bazıdır. Buna göre bir V vektör uzayının bir S bazındaki vektörlerin sayısına V uzayının boyutu denir [22].

Tanım 2.5 (Lineer Vektör Uzayının Tabanı) V , bir lineer vektör uzayı olsun. Eğer V uzayındaki her vektör, uzaydaki lineer bağımsız vektörlerin tek bir lineer kombinezonu olarak ifade edilebiliyorsa, bu lineer bağımsız vektörlerin kümesine V uzayının bir tabanı denir.

Lineer vektör uzayının boyutu n ise taban tanımını aşağıdaki teoremle yapabilmek mümkündür:

Teorem 2.1 n boyutlu bir V_n lineer vektör uzayında, elemanları lineer bağımsız vektörler olan her küme bir taban niteliğindedir.

İspat: V_n uzayında, $\alpha_1, \alpha_2, \dots, \alpha_n$ n tane lineer bağımsız vektör ve α , keyfi bir vektör olsun. Vektör uzayının boyutu (Tanım 2.4) tanımından, n sayısı, uzaydaki lineer bağımsız vektörlerin sayısı olduğu sürece $(n+1)$ elemanlı vektör kümesinin

$\alpha, \alpha_1, \alpha_2, \dots, \alpha_n$ vektörleri lineer bağımlı olacaktır. Aynı zamanda, $\mathbf{F}$ cisminde hepsi birden sıfır olmayan $c_0, c_1, c_2, \dots, c_n$ elemanları olsun ve

$$c_0\alpha + c_1\alpha_1 + c_2\alpha_2 + \dots + c_n\alpha_n = 0 \quad (2.10)$$

şeklinde yazılsın. Bu eşitlikte $c_0 \neq 0$ dır. Eğer $c_0 = 0$ olursa lineer bağımlılık olmayacaktır. $\alpha_1, \alpha_2, \dots, \alpha_n$ vektörleri lineer bağımsız olduğundan $c_1 = c_2 = \dots = c_n = 0$ dır. $c_0 = 0$ olursa bu durumda $\alpha, \alpha_1, \alpha_2, \dots, \alpha_n$ vektörleri lineer bağımsız olacaktır ki bu da lineer bağımlılık tanımına ters düşecektir. Bu yüzden $c_0 \neq 0$ olmalıdır. $i = 1, 2, \dots, n$ için $d_i = -\frac{c_i}{c_0}$ olmak üzere (2.10) eşitliği

$$\alpha = d_1\alpha_1 + d_2\alpha_2 + \dots + d_n\alpha_n \quad (2.11)$$

şeklinde yazılsın. Böylece, $\mathbf{V}_n$ uzayındaki her α vektörü, $\alpha_1, \alpha_2, \dots, \alpha_n$ vektörlerinin lineer kombinezonu ile ifade edilebilir. Şimdi bu lineer kombinezonunun tekliğini gösterelim.

Kabul edelim ki α vektörünün

$$\alpha = \overline{d}_1\alpha_1 + \overline{d}_2\alpha_2 + \dots + \overline{d}_n\alpha_n \quad (2.12)$$

şeklinde başka bir lineer kombinezonu var olsun. (2.11) eşitliğinden (2.12) eşitliğini çıkararak

$$0 = (d_1 - \overline{d}_1)\alpha_1 + (d_2 - \overline{d}_2)\alpha_2 + \dots + (d_n - \overline{d}_n)\alpha_n \quad (2.13)$$

elde edilir. $\alpha_1, \alpha_2, \dots, \alpha_n$ lineer bağımsız vektörler olduklarından her birinin katsayısı sıfıra eşit olacaktır. Böylece $i = 1, 2, \dots, n$ için $d_i = \overline{d}_i$ bulunacaktır ve lineer kombinezonun tekliği gösterilmiş olacaktır.

Bu teoremden şu özellik elde edilebilir: eğer n boyutlu bir $\mathbf{V}_n$ lineer vektör uzayında bir taban seçilmişse bu uzaydaki her vektör, sadece $\mathbf{F}$ cismindeki $d_1, d_2, \dots, d_n$ skalerlerinin kümesi ile temsil edilebilir.

Tanım 2.6 (Taban Değişimi) n boyutlu bir V_n lineer vektör uzayında birçok taban kümesi mevcuttur ve her bir taban kümesi, sadece diğer taban kümeleri cinsinden ifade edilebilir. Örneğin, V_n uzayından $\alpha_1, \alpha_2, \dots, \alpha_n$ ve $\beta_1, \beta_2, \dots, \beta_n$ şeklinde iki taban vektör kümesi alınsın. Buna göre $c_{i,j}, d_{i,j} \in \mathbf{F}$ olmak üzere

$$\alpha_i = \sum_{j=1}^n c_{i,j} \beta_j, \quad i = 1, 2, \dots, n \quad (2.14)$$

$$\beta_i = \sum_{j=1}^n d_{i,j} \alpha_j, \quad i = 1, 2, \dots, n \quad (2.15)$$

şeklinde taban değişimi yapılabilir.

Tanım 2.7 (Lineer Operatör) Her $\alpha_1, \alpha_2 \in V_n$ ve her $c_1, c_2 \in \mathbf{F}$ için

$$L(c_1 \alpha_1 + c_2 \alpha_2) = c_1 L(\alpha_1) + c_2 L(\alpha_2) \quad (2.16)$$

yazılabiliyorsa L fonksiyonu bir lineer operatördür. Buna göre (2.5) eşitliği bir lineer operatördür:

$$\begin{aligned} \left. \frac{d(c_1 f_1 + c_2 f_2)}{dx} \right|_{x_i} &= \sum_{j=1}^N a_{ij} (c_1 f_1(x_j) + c_2 f_2(x_j)) \\ &= \sum_{j=1}^N c_1 a_{ij} f_1(x_j) + \sum_{j=1}^N c_2 a_{ij} f_2(x_j) \\ &= c_1 \sum_{j=1}^N a_{ij} f_1(x_j) + c_2 \sum_{j=1}^N a_{ij} f_2(x_j). \end{aligned} \quad (2.17)$$

Teorem 2.2 n boyutlu V_n lineer vektör uzayında $\alpha_1, \alpha_2, \dots, \alpha_n$ bir taban vektörleri kümesi olsun. Eğer her $\alpha_1, \alpha_2, \dots, \alpha_n$ taban vektörü bir lineer operatörü sağlarsa V_n uzayındaki herhangi bir vektör de bu lineer operatörü sağlayacaktır.

İspat V_n uzayında $\alpha_1, \alpha_2, \dots, \alpha_n$ bir taban vektörleri kümesi ve bu küme $i = 1, 2, \dots, n$ için

$$L(\alpha_i) = 0 \quad (2.18)$$

şeklinde bir lineer operatörü sağlasın. α , V_n uzayında keyfi bir vektör olmak üzere (2.11) ve (2.16) eşitliklerinden yararlanarak

$$L(\alpha) = L\left(\sum_{i=1}^n d_i \alpha_i\right) = \sum_{i=1}^n L(d_i \alpha_i) = \sum_{i=1}^n d_i L(\alpha_i) = 0 \quad (2.19)$$

elde edilir.

Teorem 2.3 Eğer n boyutlu V_n lineer vektör uzayındaki bir taban vektör kümesi bir lineer operatörü sağlarsa, diğer taban vektörleri kümesi de bu lineer operatörü sağlayacaktır.

İspat V_n uzayında $\alpha_1, \alpha_2, \dots, \alpha_n$ ve $\beta_1, \beta_2, \dots, \beta_n$ iki taban vektör kümesi olsun. $\alpha_1, \alpha_2, \dots, \alpha_n$ vektörleri (2.18) eşitliğindeki lineer operatörü sağlasın. Buna göre (2.15) ve (2.16) eşitliklerinden yararlanarak

$$L(\beta_i) = L\left(\sum_{j=1}^n d_{i,j} \alpha_j\right) = \sum_{j=1}^n L(d_{i,j} \alpha_j) = \sum_{j=1}^n d_{i,j} L(\alpha_j) = 0 \quad (2.20)$$

elde edilir. Bu eşitlikten, $\beta_1, \beta_2, \dots, \beta_n$ taban vektörleri kümesinin lineer operatörü sağladığı görülür. Bu özellik, gelecek bölümde DKM' de ağırlık katsayılarının hesaplanmasında kullanılacaktır.

2.3 Yüksek Mertebeden Polinomal Yaklaşım

$f(x)$, $[a, b]$ aralığında tanımlanmış, reel değerli ve sürekli bir fonksiyon olsun. Buna göre

$$\lim_{N \rightarrow \infty} P_N(x) = f(x) \quad (2.21)$$

olacak şekilde bir $P_N(x)$ polinomlar dizisi mevcuttur. Bu ifade limitin tanımı kullanılarak şu şekilde de yazılabilir: $f(x)$, $[a, b]$ aralığında tanımlı ve sürekli reel değerli fonksiyon olsun. Her $\varepsilon > 0$ için derecesi $N = N(\varepsilon)$ olan öyle bir $P_N(x)$ polinomu vardır ki bu polinom $[a, b]$ aralığında

$$|f(x) - P_N(x)| \leq \varepsilon \quad (2.22)$$

eşitsizliğini sağlar.

$f(x)$ fonksiyonu bir adi diferensiyel denklemin çözümü ise bu fonksiyona derecesi N den küçük bir polinomla yaklaşılabılır. Bu yaklaşım, c_k birer sabit olmak üzere

$$f(x) \approx P_N(x) = \sum_{k=0}^{N-1} c_k x^k \quad (2.23)$$

şeklinde ifade edilebilir. $P_N(x)$ polinomu vektör toplamına ve skaler çarpıma bağlı olarak N boyutlu bir $\mathbf{V}_N$ lineer vektör uzayı oluşturmaktadır. Dolayısıyla $\mathbf{V}_N$ lineer vektör uzayında, vektörler polinomlar olarak alınırsa, $1, x, x^2, \dots, x^{N-1}$ polinomları lineer bağımsızdır. Böylece $k = 1, 2, \dots, N$ için

$$s_k(x) = x^{k-1} \quad (2.24)$$

polinomu $\mathbf{V}_N$ uzayının bir tabanıdır.

Adi diferensiyel denklemlerin nümerik çözümleri için belirli grid noktadaki fonksiyon değerlerine ihtiyaç duyulmaktadır. Buna göre $[a, b]$ aralığında, $a = x_1, x_2, \dots, x_N = b$ koordinatlı N tane grid noktanın ve bu grid noktadaki fonksiyon değerlerinin $f(x_i)$ olduğunu kabul edelim. Dolayısıyla (2.23) yaklaşımındaki c_k sabitleri

$$c_0 + c_1 x_1 + c_2 x_1^2 + \dots + c_{N-1} x_1^{N-1} = f(x_1)$$

$$c_0 + c_1 x_2 + c_2 x_2^2 + \dots + c_{N-1} x_2^{N-1} = f(x_2)$$

.....

$$c_0 + c_1 x_N + c_2 x_N^2 + \dots + c_{N-1} x_N^{N-1} = f(x_N)$$

eşitlik sisteminden elde edilebilir. Yukarıdaki sistemden oluşacak katsayılar matrisi Vandermonde formundadır ve bu matris tekil değildir. Dolayısıyla bu matrisin tersi vardır ve $c_0, c_1, c_2, \dots, c_{N-1}$ sabitleri için tek bir çözüm verir. Buradan, önce sabitler

belirlenir, sonrasında polinomal yaklaşım elde edilir. Eğer N çok büyük bir sayı ise matrisin tersinin bulunması zorlaşacaktır. Buna bağlı olarak $c_0, c_1, c_2, \dots, c_{N-1}$ katsayılarının belirlenmesi de güçleşecektir. Bu zorluğu ortadan kaldırmak amacıyla, (2.23) yaklaşım polinomunu belirlemekte

$$M(x) = (x - x_1)(x - x_2) \dots (x - x_N) \quad (2.25)$$

$$M'(x_i) = (x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_N) = \prod_{k=1, k \neq i}^N (x_i - x_k) \quad (2.26)$$

olmak üzere

$$l_i(x) = \frac{M(x)}{M'(x_i)(x - x_i)} \quad (2.27)$$

Lagrange interpolasyon polinomu kullanılır. Buna göre,

$$P_N(x) = \sum_{i=1}^N f(x_i) l_i(x) \quad (2.28)$$

şeklinde dir. Buna göre, Lagrange interpolasyon polinomu kullanılırken grid noktalarındaki fonksiyon değerleri alınır ve yaklaşım polinomu elde edilir. (2.28) eşitliğindeki taban polinomunun derecesi N 'den küçüktür. Aynı zamanda Lagrange interpolasyon polinomu vektör toplamı ve skaler çarpım işlemlerini sağlamaktadır. Buna göre (2.28) eşitliğiyle verilen Lagrange interpolasyon polinomu, V_N polinom vektör uzayı içerisinde yer almaktadır. Dolayısıyla V_N uzayının başka bir tabanı $k = 1, 2, \dots, N$ için $l_k(x)$ ile gösterilebilir. Buradaki

$$l_k(x_i) = \begin{cases} 1, & k = i \\ 0, & k \neq i \end{cases} \quad (2.29)$$

şeklinde tanımlanır.

$d_1, d_2, \dots, d_n$, F cisminin elemanları ve $[a, b]$ aralığında

$$d_1 l_1(x) + d_2 l_2(x) + \dots + d_N l_N(x) = \sum_{k=1}^N d_k l_k(x) = 0 \quad (2.30)$$

eşitliği sağlansın. (2.29) tanımını kullanarak, (2.30) eşitliğine x_i noktalarının uygulanmasıyla $i = 1, 2, \dots, N$ için $d_i = 0$ bulunur. Dolayısıyla $l_1(x), l_2(x), \dots, l_N(x)$ polinomları kendi aralarında lineer bağımsız olacaklardır ve V_N uzayının bir tabanı olarak alınabileceklerdir.

$P_N(x)$ yaklaşım polinomunu Newton interpolasyon polinomu ile de belirlemek mümkündür. Newton interpolasyon polinomu

$$n_1(x) = 1, \quad n_k(x) = (x - x_{k-1})n_{k-1}(x), \quad k = 2, 3, \dots, N \quad (2.31)$$

olarak tanımlanır. Newton interpolasyon polinomu kullanılarak yaklaşım polinomu

$$P_N(x) = a_1 + \sum_{k=2}^N a_k (x - x_1)(x - x_2) \dots (x - x_{k-1}) \quad (2.32)$$

şeklinde yazılır. Newton interpolasyon polinomu, vektör toplamı ve skaler çarpım işlemlerini sağlamaktadır ve bu polinomun derecesi N 'den küçüktür. Buna göre Newton interpolasyon polinomu, V_N polinom vektör uzayının içerisinde ve $k = 1, 2, \dots, N$ için $n_k(x)$ polinomları, V_N uzayında bir polinomlar kümesidir. Bu polinomlar kümesinin elemanlarının lineer bağımsız olduğunu gösterelim. $d_1, d_2, \dots, d_N$ F cisminin elemanları ve

$$d_1 n_1(x) + d_2 n_2(x) + \dots + d_N n_N(x) = \sum_{k=1}^N d_k n_k(x) = 0 \quad (2.33)$$

olsun. Bu (2.33) eşitliği, $[a, b]$ aralığındaki her noktada sağlanmalıdır. Eğer (2.33) eşitliği sırasıyla $x_1, x_2, \dots, x_N$ grid noktalarında sağlanırsa $i = 1, 2, \dots, N$ için $d_i = 0$ elde edilir. Böylece $k = 1, 2, \dots, N$ için $n_k(x)$ polinomları kendi aralarında lineer bağımsız olur ve V_N uzayının bir tabanı olarak kabul edilir.

Newton interpolasyon polinomu ile oluşturulan yaklaşım polinomunun sabitleri, polinomun grid noktalarındaki değerleri ile ifade edilebilir. Buna göre (2.32) eşitliği ile verilen polinomun açılımı $m = 1, 2, \dots, N$ için

$$P_N(x) = a_1 + a_2(x - x_1) + a_3(x - x_1)(x - x_2) + \dots + a_m(x - x_1)(x - x_2) \dots (x - x_{m-1})$$

şeklindedir ve oluşan bu yaklaşım polinomu $(m-1)$. derecedendir. Bu polinom, $x_1, x_2, \dots, x_m$ noktalarındaki fonksiyon değerlerini sağlamaktadır. (2.28) eşitliğinde Lagrange interpolasyon polinomu ile oluşturulan yaklaşım polinomunun açılımı ise $m=1, 2, \dots, N$ için

$$P_N(x) = \frac{M(x)f(x_1)}{M'(x_1)(x-x_1)} + \frac{M(x)f(x_2)}{M'(x_2)(x-x_2)} + \dots + \frac{M(x)f(x_m)}{M'(x_m)(x-x_m)}$$

şeklinde olup $(m-1)$. dereceden olacaktır. Böylece Newton interpolasyon polinomuyla elde edilen yaklaşım polinomunun en yüksek dereceli teriminin katsayısı olan a_m , Lagrange interpolasyon polinomuyla elde edilen yaklaşım polinomunun en yüksek dereceli teriminin katsayısına eşit olacaktır. Dolayısıyla Lagrange interpolasyon polinomuyla oluşturulan yaklaşım polinomunun en yüksek dereceli teriminin katsayısı

$$\frac{f(x_1)}{M'(x_1)} + \frac{f(x_2)}{M'(x_2)} + \dots + \frac{f(x_m)}{M'(x_m)} = \sum_{k=1}^m \frac{f(x_k)}{M'(x_k)}$$

olup buradan $m=1, 2, \dots, N$ için

$$a_m = \sum_{k=1}^m \frac{f(x_k)}{M'(x_k)} \quad (2.34)$$

bulunur.

Yukarıdaki incelemelere bakılırsa bir V_N lineer vektör uzayında birden fazla taban vektörleri kümesi mevcuttur. Bu taban vektörleri kümelerinden üçü

$$r_k(x) = x^{k-1}, \quad k=1, 2, \dots, N \quad (2.35)$$

$$r_k(x) = \frac{M(x)}{M'(x_k)(x-x_k)}, \quad k=1, 2, \dots, N \quad (2.36)$$

$$r_1(x) = 1, \quad r_k(x) = (x-x_{k-1})r_{k-1}(x), \quad k=2, 3, \dots, N \quad (2.37)$$

şeklinde gösterilebilir.

Yaklaşım polinomu başka formlarda ifade edilmek istenirse farklı taban vektörleri kümesi elde edilmelidir. Bunlardan biri, Legendre açılımı kullanılarak yaklaşım polinomu yazabilmektir.

Legendre polinomu $L_k(x)$, $[-1,1]$ aralığında tanımlı, k . dereceden bir polinom ve $L_0(x)=1$ olmak üzere

$$L_k(x) = \frac{1}{2^k k!} \frac{d^k}{dx^k} (x^2 - 1)^k \quad (2.38)$$

şeklinde tanımlanmaktadır ve

$$L_{k+1}(x) = \frac{2k+1}{k+1} x L_k(x) - \frac{k}{k+1} L_{k-1}(x) \quad (2.39)$$

rekürans bağıntısını sağlamaktadır. Buna göre Legendre açılımı, $L_k(x)$ k . dereceden Legendre polinomu ve c_k birer sabit olmak üzere

$$P_N(x) = \sum_{k=0}^{N-1} c_k L_k(x) \quad (2.40)$$

şeklindedir. Burada, $k = 0, 1, \dots, N-1$ için $L_k(x)$ bir taban vektörleri kümesidir. (2.36) eşitliğindeki taban polinomuna bakılırsa, yaklaşım polinomun grid noktalarının dağılımına göre değişebilmektedir. Eğer grid noktalar $[-1,1]$ aralığında N . dereceden Legendre polinomunun kökleri olarak alınırsa (2.36) eşitliği $k = 0, 1, \dots, N$ için

$$r_k(x) = \frac{L_N(x)}{L'_N(x_k)(x - x_k)} \quad (2.41)$$

şekline dönüşecektir.

POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME METODU

DKM, lineer vektör uzayı analizine ve fonksiyon yaklaşımına dayanmaktadır. PDKM’de adi diferensiyel denklemin çözümüne yüksek dereceden bir polinomla yaklaşılmaktadır. Bu yaklaşım polinomu ile birlikte polinomun ağırlık katsayıları da belirlenmektedir. Böylece belirlenen ağırlık katsayıları ile yaklaşım polinomu yazılabilecek ve adi diferensiyel denklemin çözümüne yakın bir çözüm polinomu elde edilebilecektir.

PDKM ile ağırlık katsayıları ilk kez Bellman ve Casti [1] tarafından hesaplanmıştır. Birinci mertebeden türevlerin ağırlık katsayılarının hesaplanmasında iki fikir ortaya atılmıştır. Bunlardan ilki bir cebirsel denklem sisteminin çözümünü içermektedir. Diğerinde ise basit bir cebirsel formül kullanılmaktadır. Ancak bu formüldeki grid noktalar ötelenmiş Legendre polinomunun köklerinden seçilmektedir.

PDKM’nin mühendislikteki ilk uygulamalarında ağırlık katsayılarını belirlemek üzere Bellman’ın birinci yaklaşımı kullanılmıştır. Çünkü bu yaklaşımda grid noktalar keyfi olarak seçilebilmektedir. Ancak grid noktaların sayısı arttıkça cebirsel denklem sistemindeki denklem sayısı da artmaktadır. Dolayısıyla katsayılar matrisini istenilen sonucu veremeyecektir. Buna göre grid noktaların sayısı fazla olduğunda ağırlık katsayılarını belirlemek oldukça zorlaşacaktır. Bu zorluğu ortadan kaldırmak üzere birçok çalışma yapılmıştır. Quan ve Chang [5-6], Lagrange interpolasyon polinomunu test fonksiyonu olarak alıp birinci ve ikinci mertebeden türevler için ağırlık katsayılarını hesaplayacak formüller geliştirilmişlerdir. Shu ve Richards [7] PDKM’de ağırlık katsayılarını hesaplayan tüm yaklaşımların lineer vektör uzayındaki taban vektörlerin seçiminden oluşturulabileceğini bulmuşlardır. Diğer bir ifadeyle, lineer vektör uzayındaki taban vektörlerinin her birinin seçimiyle ağırlık katsayılarını hesaplamakta

birçok yol bulunmaktadır. Bu farklı yollarla hesaplanan ağırlık katsayıları sonuçta aynı değerleri vermektedir. Bu fikir, lineer vektör uzayı özelliklerinden yararlanılarak ortaya atılmıştır ve oldukça kullanışlıdır. Shu birinci mertebeden türevlerin ağırlık katsayılarını hesaplamakta kullanılacak basit bir cebirsel formül geliştirmiştir. Bu formülde grid noktaların seçiminde hiçbir kısıtlama bulunmamaktadır. Yine Shu, ikinci ve yüksek mertebeden türevlerin ağırlık katsayılarının hesaplanması için bir rekürans bağıntısı oluşturulmuştur.

Bu bölümde birinci, ikinci ve yüksek mertebeden türevlerin ağırlık katsayılarının hesaplanmasındaki yaklaşımlar, hata analizi ve grid noktaların dağılımlarının seçimleri ele alınacaktır.

3.1 Birinci Mertebeden Türevler için Ağırlık Katsayılarının Hesaplanması

$[a, b]$ aralığında, koordinatları $a = x_1, x_2, \dots, x_N = b$ olan N tane grid nokta ele alınsın. $f(x)$, $[a, b]$ aralığında düzgün bir fonksiyon olmak üzere bu fonksiyonun birinci mertebeden türevine her x_i grid noktasında

$$f'(x_i) = \sum_{j=1}^N a_{ij} f(x_j), \quad i = 1, 2, \dots, N \quad (3.1)$$

formülü ile yakınsanacaktır. (3.1) eşitliğindeki a_{ij} , birinci mertebeden türevin ağırlık katsayılarıdır. (2.17) eşitliğinde gösterildiği üzere $f'(x)$ bir lineer operatördür.

PDKM'de a_{ij} ağırlık katsayılarının belirlenmesi esastır. $f'(x_i)$ türevlerini hesaplamakta, a_{ij} ağırlık katsayıları ile birlikte $f(x_i)$ fonksiyon değerleri kullanılmaktadır. Bunun için önce ağırlık katsayıları belirlenmelidir. Bu ağırlık katsayılarını belirlemekte birçok yaklaşım geliştirilmiştir. Bu yaklaşımlarda farklı taban polinomları kullanılmakta ancak uygulamanın temeli hepsinde aynıdır.

Yaklaşım polinomu $f(x)$ ve taban polinomu $r_k(x)$ olsun. Yaklaşım polinomunun taban polinomlarının lineer toplamı şeklinde yazılışı c_k birer sabit olmak üzere

$$f(x) = \sum_{k=1}^N c_k r_k(x) \quad (3.2)$$

şeklindedir. Bu yaklaşım polinomunun x 'e göre birinci mertebeden türevi ise

$$f'(x) = \sum_{k=1}^N c_k r'_k(x) \quad (3.3)$$

olacaktır. (3.2) ve (3.3) eşitlikleri (3.1) eşitliğiyle verilen DKM'de uygulansın:

$$\sum_{k=1}^N c_k r'_k(x_i) = \sum_{j=1}^N a_{ij} \left[\sum_{k=1}^N c_k r_k(x_j) \right]$$

$$\sum_{k=1}^N c_k r'_k(x_i) = \sum_{j=1}^N \sum_{k=1}^N a_{ij} c_k r_k(x_j)$$

$$\sum_{k=1}^N c_k r'_k(x_i) = \sum_{k=1}^N c_k \sum_{j=1}^N a_{ij} r_k(x_j).$$

Buna göre c_k değerleri terim terime eşitlendiğinde a_{ij} ağırlık katsayıları

$$r'_k(x_i) = \sum_{j=1}^N a_{ij} r_k(x_j) \quad (3.4)$$

eşitliği ile hesaplanabilecektir. (3.4) eşitliğine göre farklı taban polinomları kullanarak farklı yaklaşım polinomları elde etmeye çalışalım.

3.1.1 Bellman Yaklaşımları

(3.1) eşitliğindeki a_{ij} ağırlık katsayılarının belirlenmesi için Bellman iki yaklaşım önermiştir. Bu iki yaklaşımda farklı test fonksiyonları kullanılmaktadır.

3.1.1.1 Bellman'ın Birinci Yaklaşımı:

Bu yaklaşımda, test fonksiyonları $k = 0, 1, \dots, N-1$ için

$$r_k(x) = x^k \quad (3.5)$$

olarak seçilmiştir. Buna göre, N tane test fonksiyonu elde edilecektir. (3.1) eşitliğindeki a_{ij} ağırlık katsayıları için $i, j = 1, 2, \dots, N$ alınır. Dolayısıyla $N \times N$ tane ağırlık katsayısı oluşacaktır. Bu ağırlık katsayılarını elde etmek için N tane test

fonksiyonuna N tane grid nokta $(x_1, x_2, \dots, x_N)$ uygulanmalıdır. Sonuçta, a_{ij} ağırlık katsayıları için $N \times N$ tane cebirsel denklem elde edilecektir.

$k = 0$ için $r_0(x) = 1$ ve $\frac{d}{dx}r_0(x) = 0$ olmak üzere (3.1) eşitliğine göre

$$\left. \frac{d}{dx}r_0(x) \right|_{x_i} = \sum_{j=1}^N a_{ij}r_0(x_j)$$

$$\sum_{j=1}^N a_{ij} = 0 \quad (3.6)$$

elde edilir.

$k = 1$ için $r_1(x) = x$ olmak üzere $\frac{d}{dx}r_1(x) = 1$ ise

$$\sum_{j=1}^N a_{ij}x_j = 1 \quad (3.7)$$

elde edilir.

Benzer şekilde $k \geq 2$ için $r_k(x) = x^k$ olmak üzere $\frac{d}{dx}r_k(x) = kx^{k-1}$ ise

$$\sum_{j=1}^N a_{ij}x_j^k = kx_i^{k-1} \quad (3.8)$$

elde edilir. Böylece her bir $i = 1, 2, \dots, N$ için

$$a_{i1} + a_{i2} + a_{i3} + \dots + a_{iN} = 0$$

$$x_1a_{i1} + x_2a_{i2} + x_3a_{i3} + \dots + x_Na_{iN} = 1$$

$$x_1^2a_{i1} + x_2^2a_{i2} + x_3^2a_{i3} + \dots + x_N^2a_{iN} = 2x_i$$

$$x_1^3a_{i1} + x_2^3a_{i2} + x_3^3a_{i3} + \dots + x_N^3a_{iN} = 3x_i^2$$

$\vdots$

$$x_1^{N-1}a_{i1} + x_2^{N-1}a_{i2} + x_3^{N-1}a_{i3} + \dots + x_N^{N-1}a_{iN} = (N-1)x_i^{N-2}$$

bulunur. Oluşan denklem sistemi ise

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & \cdots & x_N \\ x_1^2 & x_2^2 & x_3^2 & \cdots & \cdots & x_N^2 \\ x_1^3 & x_2^3 & x_3^3 & \cdots & \cdots & x_N^3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ x_1^{N-1} & x_2^{N-1} & x_3^{N-1} & \cdots & \ddots & x_N^{N-1} \end{bmatrix} \begin{bmatrix} a_{i1} \\ a_{i2} \\ a_{i3} \\ \vdots \\ \vdots \\ a_{iN} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2x_i \\ 3x_i^2 \\ \vdots \\ (N-1)x_i^{N-2} \end{bmatrix} \quad (3.9)$$

şeklinde. Bu N bilinmeyenli denklem sisteminden a_{ij} ağırlık katsayıları hesaplanır. Görüldüğü üzere (3.9) denklem sisteminin katsayılar matrisi Vandermonde formundadır ve Vandermonde matrisinin tersi mevcut olduğundan bu denklem sisteminin tek bir çözümü vardır. Ancak N çok büyük bir sayı olduğunda matrisin tersini bulmak zorlaşacak ve kötü koşullandırılmış bir matrise dönüşecektir. Böylece matris istenilen sonucu veremeyecektir. Dolayısıyla bu yaklaşım, N sayısı 13'den küçük olduğu durumlarda uygulanmaktadır.

3.1.1.2 Bellman'ın İkinci Yaklaşımı:

Bu yaklaşımda test fonksiyonu olarak ötelenmiş Legendre polinomu kullanılmaktadır. Taban polinomu $k = 1, 2, \dots, N$ için

$$r_k(x) = \frac{K_N(x)}{(x - x_k) K'_N(x_k)} \quad (3.10)$$

olarak alınsın. Buradaki $K_N(x)$, N . dereceden ötelenmiş Legendre polinomudur. a_{ij} ağırlık katsayılarını elde etmek için x_k grid noktaları ötelenmiş Legendre polinomunun köklerinden seçilmektedir. N . dereceden Legendre polinomu $L_N(x)$ olmak üzere ötelenmiş Legendre polinomu $K_N(x) = L_N(1 - 2x)$ olarak tanımlanmaktadır [2]. Bu öteleme $[0, 1]$ aralığını $[-1, 1]$ aralığına birebir dönüştürmektedir. Buna göre (3.10) eşitliğindeki taban polinomunun birinci mertebeden türevi

$$r'_k(x) = \frac{K'_N(x)(x - x_k) - K_N(x)}{(x - x_k)^2 K'_N(x_k)} \quad (3.11)$$

olacaktır. (3.10) eşitliğindeki taban polinomu ve bu polinomun (3.11) eşitliğindeki birinci mertebeden türevi (3.1) eşitliğinde yerine yazılırsa

$$\frac{K'_N(x_i)(x_i - x_k) - K_N(x_i)}{(x_i - x_k)^2 K'_N(x_k)} = \sum_{j=1}^N a_{ij} \frac{K_N(x_j)}{(x_j - x_k) K'_N(x_k)}$$

$$\frac{K'_N(x_i)(x_i - x_k) - K_N(x_i)}{(x_i - x_k)^2} = \sum_{j=1}^N a_{ij} \frac{K_N(x_j)}{(x_j - x_k)} \quad (3.12)$$

elde edilir. $k = j \neq i$ olduğunda (3.12) eşitliğinin sağ tarafında $\frac{0}{0}$ belirsizliği oluşacaktır. Çünkü x_j noktaları ötelenmiş Legendre polinomunun kökleridir. Dolayısıyla $K_N(x_j) = 0$ olacaktır. Bu belirsizliği kaldırmak amacıyla (3.12) eşitliğinin sağ taraftaki fonksiyonuna L'Hospital kuralını uygulayalım.

$$\lim_{x \rightarrow x_k} \frac{K_N(x)}{(x - x_k)} = \lim_{x \rightarrow x_k} \frac{K'_N(x)}{1} = K'_N(x_k) = K'_N(x_j) \quad (3.13)$$

Bulunan (3.13) eşitliği (3.12) eşitliğinin sağ tarafında yazılırsa

$$\frac{K'_N(x_i)(x_i - x_k) - K_N(x_i)}{(x_i - x_k)^2} = \sum_{j=1}^N a_{ij} K'_N(x_j) \quad (3.14)$$

elde edilir. Ötelenmiş Legendre polinomunun kökleri x_i olup $K_N(x_i) = 0$ olacağından ve $k = j$ kabul edildiğinden $i \neq j$ için

$$\frac{K'_N(x_i)}{(x_i - x_j)} = \sum_{j=1}^N a_{ij} K'_N(x_j)$$

yazılır. Buna göre her bir ağırlık katsayısı $i \neq j$ için

$$a_{ij} = \frac{K'_N(x_i)}{(x_i - x_j) K'_N(x_j)} \quad (3.15)$$

ile hesaplanır. Eğer $k = j = i$ ise bu durumda (3.14) eşitliğinin sol tarafında $\frac{0}{0}$ belirsizliği meydana gelecektir. Bu yüzden sol taraftaki taban polinomunun birinci mertebeden türevine L'Hospital kuralı uygulamak gerekmektedir. Buna göre

$$\begin{aligned}\lim_{x \rightarrow x_k} \frac{K'_N(x)(x - x_k) - K_N(x)}{(x - x_k)^2} &= \lim_{x \rightarrow x_k} \frac{K''_N(x)(x - x_k) + K'_N(x) - K'_N(x)}{2(x - x_k)} \\ \lim_{x \rightarrow x_k} \frac{K''_N(x)(x - x_k) + K'_N(x) - K'_N(x)}{2(x - x_k)} &= \frac{K''_N(x_k)}{2} = \frac{K''_N(x_i)}{2}\end{aligned}\quad (3.16)$$

elde edilir. Bu (3.16) eşitliği (3.14) eşitliğinin sol tarafında yerine yazılırsa

$$\frac{K''_N(x_i)}{2} = \sum_{i=1}^N a_{ii} K'_N(x_i)$$

bulunur. Buna göre $i = j$ iken ağırlık katsayıları

$$a_{ii} = \frac{K''_N(x_i)}{2K'_N(x_i)} \quad (3.17)$$

şeklinde hesaplanır. (3.17) eşitliğini daha açıkça ifade etmek için

$$(1 - x^2)L''_N(x) - 2xL'_N(x) + N(N+1)L_N(x) = 0 \quad (3.18)$$

şeklinde tanımlanan Legendre diferensiyel denkleminde x yerine $1 - 2x$ yazalım. Buna göre

$$\begin{aligned}(1 - (1 - 2x)^2)L''_N(1 - 2x) - 2(1 - 2x)L'_N(1 - 2x) + N(N+1)L_N(1 - 2x) &= 0 \\ (4x - 4x^2)L''_N(1 - 2x) - 2(1 - 2x)L'_N(1 - 2x) + N(N+1)L_N(1 - 2x) &= 0\end{aligned}\quad (3.19)$$

olacaktır. $K'_N(x) = -2L'_N(1 - 2x)$ ve $K''_N(x) = 4L''_N(1 - 2x)$ olduğuna göre bu ifadeler (3.19) eşitliğinde yerine yazılarak

$$(x - x^2)K''_N(x) + (1 - 2x)K'_N(x) + N(N+1)K_N(x) = 0 \quad (3.20)$$

elde edilir. Bu (3.20) denklemi ötelenmiş Legendre diferensiyel denklemdir. Bu diferensiyel denklemde ötelenmiş Legendre polinomunun kökleri yazılırsa

$$(x_i - x_i^2)K_N''(x_i) + (1 - 2x_i)K_N'(x_i) + N(N+1)K_N(x_i) = 0$$

$$(x_i - x_i^2)K_N''(x_i) = -(1 - 2x_i)K_N'(x_i)$$

bulunur. Böylece $i = j$ iken ağırlık katsayıları

$$a_{ii} = \frac{1 - 2x_i}{2x_i(x_i - 1)} \quad (3.21)$$

eşitliği ile de elde edilebilir.

Bellman'ın ikinci yaklaşımı, birinci yaklaşımı kadar esnek değildir. Çünkü ikinci yaklaşımda grid noktalarının koordinatları keyfi seçilememektedir. Bunun yerine, N . dereceden ötelenmiş Legendre polinomunun kökleri seçilmektedir. Böylece ikinci yaklaşımda grid noktaların seçimindeki tutumdan dolayı pratik uygulamalarda genellikle birinci yaklaşım kullanılmaktadır.

3.1.2 Quan ve Chang Yaklaşımı

Bu yaklaşımda test fonksiyonu olarak (2.25) eşitliğinde tanımlanan N . dereceden $M(x)$ Lagrange interpolasyon polinomu kullanılmaktadır.

Taban polinomu $k = 1, 2, \dots, N$ için

$$r_k(x) = \frac{M(x)}{(x - x_k)M'(x_k)} \quad (3.22)$$

olarak seçilsin. (3.22) taban polinomunun birinci mertebeden türevi

$$r_k'(x) = \frac{M'(x)(x - x_k) - M(x)}{(x - x_k)^2 M'(x_k)} \quad (3.23)$$

şeklindedir. (3.22) taban polinomu ve (3.23) birinci mertebeden türevi (3.1) eşitliğinde yazılırsa

$$\frac{M'(x_i)(x_i - x_k) - M(x_i)}{(x_i - x_k)^2 M'(x_k)} = \sum_{j=1}^N a_{ij} \frac{M(x_j)}{(x_j - x_k)M'(x_k)}$$

$$\frac{M'(x_i)(x_i - x_k) - M(x_i)}{(x_i - x_k)^2} = \sum_{j=1}^N a_{ij} \frac{M(x_j)}{(x_j - x_k)} \quad (3.24)$$

meydana gelir. Eğer $k = j \neq i$ alınırsa (3.24) eşitliğinin sağ tarafındaki fonksiyonda $\frac{0}{0}$ belirsizliği oluşacaktır. Çünkü x_j noktaları Lagrange interpolasyon polinomunun kökleridir ve bu yüzden $M(x_j) = 0$ olacaktır. Bu belirsizliği ortadan kaldırmak amacıyla sağ taraftaki fonksiyona L'Hospital kuralı uygulamak gerekmektedir. Buna göre

$$\lim_{x \rightarrow x_k} \frac{M(x)}{(x - x_k)} = M'(x_k) = M'(x_j) \quad (3.25)$$

elde edilir. Bu (3.25) eşitliği (3.24) eşitliğinin sağ tarafında yazılırsa

$$\frac{M'(x_i)(x_i - x_k) - M(x_i)}{(x_i - x_k)^2} = \sum_{j=1}^N a_{ij} M'(x_j) \quad (3.26)$$

bulunur. x_i değerleri Lagrange interpolasyon polinomunun kökleri olduğundan ve $k = j$ kabul edildiğinden

$$\frac{M'(x_i)}{(x_i - x_j)} = \sum_{j=1}^N a_{ij} M'(x_j)$$

yazılabilir. Böylece $i \neq j$ iken her bir ağırlık katsayısı

$$a_{ij} = \frac{M'(x_i)}{(x_i - x_j)M'(x_j)} \quad (3.27)$$

eşitliği ile hesaplanabilir. Eğer $k = j = i$ alınırsa bu kez (3.26) eşitliğinin sol taraftaki fonksiyonunda $\frac{0}{0}$ belirsizliği meydana gelecektir. Bu fonksiyona da L'Hospital kuralı uygulanırsa

$$\begin{aligned} \lim_{x \rightarrow x_k} \frac{M'(x)(x - x_k) - M(x)}{(x - x_k)^2} &= \lim_{x \rightarrow x_k} \frac{M''(x)(x - x_k) + M'(x) - M'(x)}{2(x - x_k)} \\ \lim_{x \rightarrow x_k} \frac{M''(x)(x - x_k) + M'(x) - M'(x)}{2(x - x_k)} &= \frac{M''(x_k)}{2} = \frac{M''(x_i)}{2} \end{aligned} \quad (3.28)$$

elde edilir. Bu (3.28) eşitliği (3.26) eşitliğinin sol tarafında yazılırsa

$$\frac{M''(x_i)}{2} = \sum_{j=1}^N a_{ij} M'(x_j)$$

olur. $k = j = i$ olduğundan

$$a_{ii} = \frac{M''(x_i)}{2M'(x_i)} \quad (3.29)$$

elde edilir. Bulunan (3.27) ve (3.29) ağırlık katsayı eşitlikleri grid noktalarla ifade edilirse (2.26) eşitliğinden yararlanarak

$$a_{ij} = \frac{1}{x_j - x_i} \prod_{k=1, k \neq i, j}^N \frac{x_i - x_k}{x_j - x_k}, \quad j \neq i \quad (3.30)$$

$$a_{ii} = \sum_{k=1, k \neq i}^N \frac{1}{x_i - x_k} \quad (3.31)$$

cebirsel formülleri elde edilir ve a_{ij} ağırlık katsayıları hesaplanır. Bu formüller kullanılırken grid noktaların seçiminde hiçbir kısıtlama yoktur, keyfi grid nokta seçimi yapılabilmektedir.

3.1.3 Shu Genel Yaklaşımı

Shu genel yaklaşımı, Bellman'ın yaklaşımlarından yararlanılarak elde edilmiştir. Bu yaklaşım, Quan ve Chang yaklaşımı da dahil olmak üzere tüm yaklaşımları içermektedir. Bu yaklaşımın ortaya çıkması iki sorundan kaynaklanmaktadır. Bunlardan birincisi, katsayıların hesaplanmasında Bellman'ın iki yaklaşım kullanılıyor olmasıdır. Diğeri ise bu iki yaklaşımın aynı sonucu verebiliyor olmasıdır. Eğer bu iki yaklaşım aynı sonucu verebiliyorsa, katsayıları hesaplamakta başka yaklaşımlar da elde edilebilmelidir. Bu yaklaşımlar, lineer vektör uzayı analizi ve polinomal yaklaşımdan yararlanarak taban değişimi yapılmasıyla oluşturulabilir.

Bilindiği üzere bir adi diferensiyel denklemin çözümüne yüksek dereceden bir polinomla yaklaşılabilmektedir. Buna göre bu yaklaşım polinomunun derecesi $N - 1$ olarak alınsın. Bu yaklaşım polinomu, vektör toplamı ve skaler çarpım işlemlerini

içeren bir $\mathbf{V}_N$ lineer vektör uzayı oluşturmaktadır ve farklı şekillerde ifade edilebilmektedir. Bunlardan biri, c_k sabit olmak üzere

$$f(x) = \sum_{k=1}^N c_k x^{k-1} \quad (3.32)$$

fonksiyonu ile tanımlansın. $\mathbf{V}_N$ lineer vektör uzayında birçok taban vektörleri kümesi bulunmaktadır. Buradaki vektörler aslında birer polinomdur. Böylece taban vektörlerine taban polinomları da denilebilir. Bu taban polinomlarından bazıları,

$$r_k(x) = x^{k-1}, \quad k = 1, 2, \dots, N \quad (3.33)$$

$$r_k(x) = \frac{K_N(x)}{(x - x_k)K'_N(x_k)}, \quad k = 1, 2, \dots, N \quad (3.34)$$

$$r_k(x) = \frac{M(x)}{(x - x_k)M'(x_k)}, \quad k = 1, 2, \dots, N \quad (3.35)$$

$$r_1(x) = 1, \quad r_k(x) = (x - x_{k-1})r_{k-1}(x), \quad k = 2, 3, \dots, N \quad (3.36)$$

şeklinde. Bu taban polinomlarından (3.34) ve (3.35) eşitlikleri Lagrange interpolasyon polinomlarından, (3.36) eşitliği Newton interpolasyon polinomlarından elde edilmiştir. (3.34) ve (3.35) taban polinomları arasındaki fark, grid noktalarının dağılımından kaynaklanmaktadır. (3.34) taban polinomunun grid noktaları ötelenmiş Legendre polinomunun köklerinden oluşmaktadır. Ancak (3.35) polinomu için böyle bir seçim söz konusu olmamakla beraber keyfi grid nokta seçimi yapılabilmektedir. Eğer (3.35) polinomunda grid noktalar olarak ötelenmiş Legendre polinomunun kökleri seçilirse (3.34) polinomu elde edilecektir. Buna göre (3.34) polinomu (3.35) polinomunun özel bir durumudur.

(3.33) eşitliği Bellman'ın birinci yaklaşımının, (3.34) eşitliği Bellman'ın ikinci yaklaşımının test fonksiyonları ile benzerdir. Başka bir ifadeyle, Bellman yaklaşımlarında kullanılan test fonksiyonları, $\mathbf{V}_N$ uzayından iki taban polinomları kümesidir.

DKM'de kullanılan (3.1) eşitliği bir lineer operatör olduğuna göre lineer vektör uzayı özellikleri kullanılarak Teorem 2.3' e göre şu yazılabilir: eğer taban polinomları kümelerinden biri bir lineer operatörü sağlarsa aynı şekilde diğer taban polinomları

kümeleri de bu lineer operatörü sağlayacaktır. Buna göre tüm taban polinomları kümesi aynı ağırlık katsayılarını verecektir. Dolayısıyla ağırlık katsayıları, test fonksiyonlarının seçimine bağlı olmayacaktır.

Diğer taraftan Bellman yaklaşımlarındaki fark, sadece farklı test fonksiyonlarının kullanımına dayanmaktadır. Bu test fonksiyonları birer taban polinomlarıdır. Böylece ağırlık katsayıları hesaplanırken, farklı taban polinomları kümelerinin kullanımı farklı yaklaşımlarla sonuçlanacaktır. V_N lineer vektör uzayında birçok taban polinomları kümesi olduğu sürece ağırlık katsayılarını hesaplamakta birçok yaklaşımla karşılaşılabılır.

Lineer vektör uzayı özelliği, ağırlık katsayılarını bir adi diferensiyel denklemin ayrıştırılmasına uygulama olanağı sağlar. Adi diferensiyel denklemin çözümüne, V_N vektör uzayını oluşturan $(N-1)$. dereceden bir polinom ile yaklaşılr. V_N lineer vektör uzayında, taban polinomları kümesi, çözümden bağımsız olacak şekilde seçilebilir. Lineer vektör uzayı özelliğinden Teorem 2.3' den yararlanarak, eğer taban polinomları kümelerinden biri bir lineer operatörü sağlarsa uzaydaki diğer polinomlar da bu lineer operatörü sağlayacaktır. Bu özelliğe göre adi diferensiyel denklemin çözümü DKM'de kullanılan eşitliği sağlayacaktır. Diğer bir ifadeyle, taban polinomlarından elde edilen ağırlık katsayıları adi diferensiyel denklemin türevlerine ayrıştırılmasında kullanılabilir.

Taban polinomlar kümesi (3.33) yapısında verilirse, Bellman birinci yaklaşımındaki denklem sistemi gibi (3.9) sistemi elde edilir. Eğer taban polinomlar kümesi (3.34) eşitliğinden elde edilirse, Bellman ikinci yaklaşımındaki cebirsel formülle verilen (3.15) ve (3.21) ortaya çıkar. Bu yaklaşımda iki taban polinomlar kümesi kullanılacaktır. İlk taban polinomları kümesi olarak (3.35) Lagrange interpolasyon polinomunu ele alalım.

δ_{ij} Kronecker operatörü

$$\delta_{ij} = \begin{cases} 0 & , i \neq j \\ 1 & , i = j \end{cases} \quad (3.37)$$

ve

$$N(x_i, x_j) = M'(x_i) \delta_{ij} \quad (3.38)$$

olmak üzere $k = 1, 2, \dots, N$ için

$$M(x) = N(x, x_k)(x - x_k) \quad (3.39)$$

şeklinde tanımlansın. (3.39) eşitliğindeki $M(x)$ fonksiyonunu kullanarak (3.35) eşitliği

$$r_k(x) = \frac{M(x)}{(x - x_k)M'(x_k)} = \frac{N(x, x_k)(x - x_k)}{(x - x_k)M'(x_k)}$$

$$r_k(x) = \frac{N(x, x_k)}{M'(x_k)}, \quad k = 1, 2, \dots, N \quad (3.40)$$

ifadesine dönüşecektir. (3.40) eşitliğinin birinci mertebeden türevi

$$r'_k(x) = \frac{N'(x, x_k)}{M'(x_k)} \quad (3.41)$$

olmak üzere (3.40) ve (3.41) fonksiyonlarını (3.1) eşitliğindeki DKM eşitliğinde yerine yazarsak

$$\frac{N'(x_i, x_k)}{M'(x_k)} = \sum_{j=1}^N a_{ij} \frac{N(x_j, x_k)}{M'(x_k)}$$

$$N'(x_i, x_k) = \sum_{j=1}^N a_{ij} N(x_j, x_k) \quad (3.42)$$

elde edilir. (3.42) eşitliğinde (3.38) eşitliğinden yararlanarak her $k = j \neq i$ için

$$a_{ij} = \frac{N'(x_i, x_j)}{N(x_j, x_j)}$$

$$a_{ij} = \frac{N'(x_i, x_j)}{M'(x_j)\delta_{jj}}$$

$$a_{ij} = \frac{N'(x_i, x_j)}{M'(x_j)} \quad (3.43)$$

elde edilir. Bu (3.43) eşitliğindeki $M'(x_j)$ değeri (2.26) formülü ile hesaplanabilir. Ancak $N'(x_i, x_j)$ değerini hesaplamak için (3.39) eşitliğinin x 'e göre türetilmesi gerekmektedir. Buna göre $m = 1, 2, \dots, N-1$ ve $k = 1, 2, \dots, N$ için

$$\begin{aligned}
M(x) &= N(x, x_k)(x - x_k) \\
M'(x) &= N'(x, x_k)(x - x_k) + N(x, x_k) \\
M''(x) &= N''(x, x_k)(x - x_k) + 2N'(x, x_k) \\
&\vdots \\
M^{(m)}(x) &= N^{(m)}(x, x_k)(x - x_k) + mN^{(m-1)}(x, x_k)
\end{aligned} \tag{3.44}$$

şeklinde türetilir ve böylece (3.44) şeklinde bir rekürans formülü elde edilir. Bu rekürans formülü kullanılarak $N'(x_i, x_j)$ değeri hesaplanırsa

$$\begin{aligned}
M'(x_i) &= N'(x_i, x_j)(x_i - x_j) + N(x_i, x_j) \text{ eşitliğinden} \\
N'(x_i, x_j) &= \frac{M'(x_i) - N(x_i, x_j)}{x_i - x_j} \\
N'(x_i, x_j) &= \frac{M'(x_i) - M'(x_i)\delta_{ij}}{x_i - x_j} \\
N'(x_i, x_j) &= \frac{M'(x_i)(1 - \delta_{ij})}{x_i - x_j} \\
N'(x_i, x_j) &= \frac{M'(x_i)}{(x_i - x_j)}, i \neq j
\end{aligned} \tag{3.45}$$

elde edilir. Benzer şekilde $M''(x_i) = N''(x_i, x_i)(x_i - x_i) + 2N'(x_i, x_i)$ eşitliğinden

$$N'(x_i, x_i) = \frac{M''(x_i)}{2} \tag{3.46}$$

elde edilir. Bulunan (3.45) ve (3.46) eşitlikleri (3.43) eşitliğinde yerine yazılarak a_{ij} ağırlık katsayıları hesaplanırsa

$$a_{ij} = \frac{N'(x_i, x_j)}{M'(x_j)} = \frac{M'(x_i)}{(x_i - x_j)M'(x_j)}, i \neq j \quad (3.47)$$

$$a_{ii} = \frac{N'(x_i, x_i)}{M'(x_i)} = \frac{M''(x_i)}{2M'(x_i)} \quad (3.48)$$

bulunur.

Eğer x_i noktası verilmişse (2.26) eşitliğiyle $M'(x_i)$ değerini hesaplamak kolaydır, böylece $i \neq j$ için a_{ij} ağırlık katsayıları bulunabilir. Ancak a_{ii} ağırlık katsayılarının hesaplanmasında $M''(x_i)$ değeri gerekmektedir ve bu türevi hesaplamak zordur. Bu zorluk seçilen ikinci taban polinomları kümesi kullanılarak ortadan kaldırılabilir. Eğer taban polinomları kümelerinden biri bir lineer operatörü sağlarsa diğer taban vektörleri kümeleri de bu lineer operatörü sağlayacaktır. Buna göre Lagrange interpolasyon polinomlarının türetilmesiyle belirlenen a_{ij} ağırlık katsayıları, diğer taban polinomlarının türetilmesiyle de elde edilebilir. Kullanılacak diğer taban polinom kümesi (3.33) eşitliğiyle verilen x^{k-1} ($k=1,2,\dots,N$) olsun. Buna göre a_{ij} ağırlık katsayıları, $k=1$ iken bu taban polinomundan elde edilen (3.49) eşitliğini sağlayacaktır. Dolayısıyla

$$\sum_{j=1}^N a_{ij} = 0$$

veya

$$a_{ii} = - \sum_{j=1, j \neq i}^N a_{ij} \quad (3.49)$$

eşitliği kullanılarak $i = j$ için a_{ii} ağırlık katsayıları hesaplanacaktır.

Shu genel yaklaşımı ile ağırlık katsayıları hesaplanırken (3.47) ve (3.49) eşitlikleri kullanılacaktır. Bu eşitlikler, $\mathbf{V}_N$ lineer vektör uzayından seçilen iki taban polinomları kümesinden elde edilmiştir.

3.2 İkinci ve Yüksek Mertebeden Türevler için Ağırlık Katsayılarının Hesaplanması

Burada ikinci ve yüksek mertebeden türevlerin ağırlık katsayılarının hesaplanmasında kullanılan rekürans bağıntıları verilecektir.

3.2.1 İkinci Mertebeden Türevin Ağırlık Katsayıları

İkinci mertebeden türevin ayrıştırılmasında birinci yaklaşıma benzer olan

$$f''(x_i) = \sum_{j=1}^N w_{ij}^{(2)} f(x_j), i = 1, 2, \dots, N \quad (3.50)$$

eşitliği kullanılmaktadır. Buradaki $w_{ij}^{(2)}$ katsayıları, ikinci mertebeden türevin ağırlık katsayılarıdır. Bu yaklaşım bir lineer operatördür:

$$\begin{aligned} \frac{d^2(c_1 f_1 + c_2 f_2)}{dx^2} &= \sum_{j=1}^N w_{ij}^{(2)} (c_1 f_1(x_j) + c_2 f_2(x_j)) \\ &= \sum_{j=1}^N w_{ij}^{(2)} c_1 f_1(x_j) + \sum_{j=1}^N w_{ij}^{(2)} c_2 f_2(x_j) \\ &= c_1 f_1''(x_i) + c_2 f_2''(x_i). \end{aligned} \quad (3.51)$$

DKM eşitliğinin birinci ve ikinci mertebeden türevlerin seri açılımları birbirine benzemektedir. Aralarındaki tek fark, farklı ağırlık katsayılarının kullanılmasıdır.

3.2.1.1 Quan ve Chang Yaklaşımı

Bu yaklaşımda test fonksiyonları olarak Lagrange interpolasyon polinomları kullanılmaktadır. Bu yaklaşımdaki ağırlık katsayıları

$$w_{ij}^{(2)} = \frac{2}{x_j - x_i} \left(\prod_{k=1, k \neq i, j}^N \frac{x_i - x_k}{x_j - x_k} \right) \left(\sum_{l=1, l \neq i, j}^N \frac{1}{x_i - x_l} \right), i \neq j \quad (3.52)$$

$$w_{ii}^{(2)} = 2 \sum_{k=1, k \neq i}^{N-1} \left[\frac{1}{x_i - x_k} \left(\sum_{l=k+1, l \neq i}^N \frac{1}{x_i - x_l} \right) \right] \quad (3.53)$$

şeklinde hesaplanmaktadır.

3.2.1.2 Shu Genel Yaklaşımı

Shu genel yaklaşımı, birinci mertebeden türev yaklaşıma benzer şekilde, polinomal yaklaşıma ve lineer vektör uzayı analizine dayanmaktadır. Ağırlık katsayılarını hesaplamakta taban polinomları kümelerinden ikisi ((3.35) ve (3.33)) kullanılır. (3.40) eşitliği ve ikinci mertebeden türevi (3.50) eşitliğinde yerine yazılırsa

$$w_{ij}^{(2)} = \frac{N''(x_i, x_j)}{M'(x_j)} \quad (3.54)$$

elde edilir. Diğer taraftan (3.44) rekürans formülünden yararlanarak

$$M''(x_i) = N''(x_i, x_j)(x_i - x_j) + 2N'(x_i, x_j)$$

$$N''(x_i, x_j) = \frac{M''(x_i) - 2N'(x_i, x_j)}{(x_i - x_j)}, i \neq j \quad (3.55)$$

$$M'''(x_i) = N'''(x_i, x_i)(x_i - x_i) + 3N''(x_i, x_i)$$

$$N''(x_i, x_i) = \frac{M'''(x_i)}{3} \quad (3.56)$$

değerleri hesaplanır. Bu (3.55) ve (3.56) değerleri, (3.54) eşitliğinde yerine yazılırsa

$$w_{ij}^{(2)} = \frac{N''(x_i, x_j)}{M'(x_j)} = \frac{M''(x_i) - 2N'(x_i, x_j)}{(x_i - x_j)M'(x_j)}, i \neq j \quad (3.57)$$

$$w_{ii}^{(2)} = \frac{N''(x_i, x_i)}{M'(x_i)} = \frac{M'''(x_i)}{3M'(x_i)} \quad (3.58)$$

bulunur. Son olarak (3.47) ve (3.48) eşitliklerini sırasıyla (3.57) ve (3.58) katsayı eşitliklerinde yerine yazılmasıyla $i \neq j$ için

$$w_{ij}^{(2)} = \frac{M''(x_i)}{(x_i - x_j)M'(x_j)} - \frac{2N'(x_i, x_j)}{(x_i - x_j)M'(x_j)}$$

$$w_{ij}^{(2)} = \frac{2M'(x_i)M''(x_i)}{2M'(x_i)(x_i - x_j)M'(x_j)} - \frac{2}{(x_i - x_j)M'(x_j)} \frac{M'(x_i)}{(x_i - x_j)}$$

$$\begin{aligned}
w_{ij}^{(2)} &= 2 \frac{M''(x_i)}{2M'(x_i)(x_i - x_j)M'(x_j)} - \frac{2}{(x_i - x_j)} \frac{M'(x_i)}{M'(x_j)(x_i - x_j)} \\
w_{ij}^{(2)} &= 2a_{ii} \frac{M'(x_i)}{(x_i - x_j)M'(x_j)} - \frac{2}{(x_i - x_j)} \frac{M'(x_i)}{(x_i - x_j)M'(x_j)} \\
w_{ij}^{(2)} &= 2a_{ii}a_{ij} - \frac{2}{x_i - x_j} a_{ij} \\
w_{ij}^{(2)} &= 2a_{ij} \left(a_{ii} - \frac{1}{x_i - x_j} \right) \tag{3.59}
\end{aligned}$$

elde edilir.

$i \neq j$ olduğu sürece $w_{ij}^{(2)}$ ağırlık katsayıları (3.59) eşitliği ile kolayca hesaplanabilir. Ancak $w_{ii}^{(2)}$ ağırlık katsayısının hesaplanması için $M'''(x_i)$ değeri gerekmektedir ve üçüncü mertebeden türevin hesaplanması zordur. Bu zorluk lineer vektör uzayı özelliği kullanılarak ortadan kaldırılabilir. Lagrange interpolasyon polinomundan elde edilen (3.59) $w_{ij}^{(2)}$ eşitlik sistemi, diğer taban polinomları kümelerinden de elde edilebilmektedir. Kullanılacak diğer taban polinom kümesi (3.33) eşitliği olarak alınsın. Buna göre $w_{ij}^{(2)}$ ağırlık katsayıları, $k=1$ iken bu taban polinomundan elde edilen eşitliği sağlayacaktır.

$$\begin{aligned}
\sum_{j=1}^N w_{ij}^{(2)} &= 0 \\
w_{ii}^{(2)} &= - \sum_{j=1, j \neq i}^N w_{ij}^{(2)} \tag{3.60}
\end{aligned}$$

Böylece Shu genel yaklaşımı ile ikinci mertebeden türevlerin ağırlık katsayıları kullanılırken $w_{ij}^{(2)}$ ağırlık katsayıları hesaplanırken (3.59) eşitliği, $w_{ii}^{(2)}$ ağırlık katsayıları hesaplanırken (3.60) eşitliği kullanılmaktadır.

3.2.2 Yüksek Mertebeden Türevler için Shu Rekürans Formülü

Yüksek mertebeden türevlerin ayrıştırılmasında $i = 1, 2, \dots, N$ ve $m = 2, 3, \dots, N - 1$ için

$$f^{(m-1)}(x_i) = \sum_{j=1}^N w_{ij}^{(m-1)} f(x_j) \quad (3.61)$$

$$f^{(m)}(x_i) = \sum_{j=1}^N w_{ij}^{(m)} f(x_j) \quad (3.62)$$

lineer operatörleri kullanılmaktadır. Buradaki $w_{ij}^{(m)}$ ağırlık katsayıları, m . mertebeden türeve aittir. $w_{ij}^{(m)}$ ağırlık katsayılarını elde etmek için önceki Shu yaklaşımlarında olduğu gibi iki taban polinomları kümesi kullanılacaktır. Bu kümelerden biri, önceden (3.40) eşitliğiyle verilmiş olan $r_k(x)$ taban polinomlarıdır. Bu taban polinomu ve $(m-1)$. mertebeden türevi (3.61) eşitliğinde, m . mertebeden türevi (3.62) eşitliğinde yerlerine yazılırsa

$$w_{ij}^{(m-1)} = \frac{N^{(m-1)}(x_i, x_j)}{M'(x_j)} \quad (3.63)$$

$$w_{ij}^{(m)} = \frac{N^{(m)}(x_i, x_j)}{M'(x_j)} \quad (3.64)$$

katsayıları elde edilir. (3.63) eşitliği

$$N^{(m-1)}(x_i, x_j) = w_{ij}^{(m-1)} M'(x_j) \quad (3.65)$$

olarak yazılabilir. Bu (3.65) eşitliği her i ve j için geçerlidir. Diğer taraftan (3.44) ile verilen rekürans bağıntısından

$$M^{(m)}(x_i) = N^{(m)}(x_i, x_i)(x_i - x_i) + mN^{(m-1)}(x_i, x_i)$$

$$N^{(m-1)}(x_i, x_i) = \frac{M^{(m)}(x_i)}{m} \quad (3.66)$$

elde edilir. Aynı zamanda bu rekürans bağıntısından

$$\begin{aligned}
M^{(m)}(x_i) &= N^{(m)}(x_i, x_j)(x_i - x_j) + mN^{(m-1)}(x_i, x_j) \\
N^{(m)}(x_i, x_j) &= \frac{M^{(m)}(x_i) - mN^{(m-1)}(x_i, x_j)}{(x_i - x_j)}, i \neq j
\end{aligned} \tag{3.67}$$

bulunur. Eğer $i = j$ ise

$$\begin{aligned}
M^{(m+1)}(x_i) &= N^{(m+1)}(x_i, x_i)(x_i - x_i) + (m+1)N^{(m)}(x_i, x_i) \\
N^{(m)}(x_i, x_i) &= \frac{M^{(m+1)}(x_i)}{m+1}
\end{aligned} \tag{3.68}$$

olacaktır. (3.66) eşitliğini (3.67) eşitliğinde yerine yazalım. (3.66) eşitliğinden yararlanarak

$$\begin{aligned}
N^{(m)}(x_i, x_j) &= \frac{M^{(m)}(x_i) - mN^{(m-1)}(x_i, x_j)}{(x_i - x_j)} \\
&= \frac{mN^{(m-1)}(x_i, x_i) - mN^{(m-1)}(x_i, x_j)}{(x_i - x_j)} \\
N^{(m)}(x_i, x_j) &= \frac{m[N^{(m-1)}(x_i, x_i) - N^{(m-1)}(x_i, x_j)]}{(x_i - x_j)}, i \neq j
\end{aligned} \tag{3.69}$$

olacaktır. Elde edilen (3.69) eşitliği, (3.65) eşitliği kullanılarak

$$N^{(m)}(x_i, x_j) = \frac{m[w_{ii}^{(m-1)}M'(x_i) - w_{ij}^{(m-1)}M'(x_j)]}{(x_i - x_j)}, i \neq j \tag{3.70}$$

olarak basitleştirilebilir. Bu eşitlik (3.64) eşitliğinde yerine yazılır ve (3.47) eşitliği kullanılırsa

$$\begin{aligned}
w_{ij}^{(m)} &= \frac{N^{(m)}(x_i, x_j)}{M'(x_j)} = \frac{m[w_{ii}^{(m-1)}M'(x_i) - w_{ij}^{(m-1)}M'(x_j)]}{(x_i - x_j)M'(x_j)} \\
&= m \left[\frac{w_{ii}^{(m-1)}M'(x_i)}{(x_i - x_j)M'(x_j)} - \frac{w_{ij}^{(m-1)}M'(x_j)}{(x_i - x_j)M'(x_j)} \right]
\end{aligned}$$

$$w_{ij}^{(m)} = m \left(a_{ij} w_{ii}^{(m-1)} - \frac{w_{ij}^{(m-1)}}{x_i - x_j} \right), \quad i, j = 1, 2, \dots, N, \quad m = 2, 3, \dots, N-1 \quad (3.71)$$

elde edilir. $w_{ii}^{(m)}$ eşitliği, (3.68) eşitliğini (3.64) eşitliğinde yerine yazılarak elde edilebilir:

$$w_{ii}^{(m)} = \frac{N^{(m)}(x_i, x_i)}{M'(x_i)} = \frac{M^{(m+1)}(x_i)}{(m+1)M'(x_i)} \quad i, j = 1, 2, \dots, N, \quad m = 2, 3, \dots, N-1. \quad (3.72)$$

Önceki yaklaşımlarda olduğu gibi $i \neq j$ iken $w_{ij}^{(m)}$ ağırlık katsayılarını hesaplamak kolaydır ancak $w_{ii}^{(m)}$ için aynısı söylenemez. Çünkü $w_{ii}^{(m)}$ formülü $M(x)$ fonksiyonunun $(m+1)$. mertebeden türevini içermektedir ve bu türevi hesaplamak oldukça zordur. Bu durumda lineer vektör uzayı özelliklerinden yararlanmak gerekmektedir. Lagrange interpolasyon polinomu ile elde edilen $w_{ij}^{(m)}$ denklem sistemleri, aynı sonucu verecek şekilde diğer taban polinomları kümesiyle elde edilebilir. Taban polinomu olarak x^{k-1} ($k=1, 2, \dots, N$) alınırsa $k=1$ için elde edilen eşitliği $w_{ij}^{(m)}$ sağlayacaktır.

$$\sum_{j=1}^N w_{ij}^{(m)} = 0$$

$$w_{ii}^{(m)} = - \sum_{j=1, j \neq i}^N w_{ij}^{(m)} \quad (3.73)$$

3.3 Matris Çarpımı Yaklaşımı

Diferensiyel operatör tanımı

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial}{\partial x} \left(\frac{\partial f}{\partial x} \right) \quad (3.74)$$

olmak üzere, DKM eşitliği (3.74) eşitliğinin sol tarafına uygulanırsa

$$f''(x_i) = \sum_{j=1}^N w_{ij}^{(2)} f(x_j), \quad i = 1, 2, \dots, N \quad (3.75)$$

elde edilir. DKM eşitliği, (3.74) eşitliğinin sağ tarafına iki kez uygulanırsa

$$\begin{aligned} f''(x_i) &= \sum_{k=1}^N a_{ik} f'(x_k) = \sum_{k=1}^N a_{ik} \sum_{j=1}^N a_{kj} f(x_j), \quad i = 1, 2, \dots, N \\ &= \sum_{j=1}^N \left[\sum_{k=1}^N a_{ik} a_{kj} \right] f(x_j) \end{aligned} \quad (3.76)$$

olur. Buna göre (3.75) ve (3.76) eşitlikleri karşılaştırıldığında

$$w_{ij}^{(2)} = \sum_{k=1}^N a_{ik} a_{kj} \quad (3.77)$$

elde edilir.

$$\left[A^{(1)} \right] = \begin{pmatrix} a_{11} & a_{12} \cdots & a_{1N} \\ a_{21} & a_{22} \cdots & a_{2N} \\ \vdots & \vdots & \vdots \\ a_{N1} & a_{N2} \cdots & a_{NN} \end{pmatrix}, \quad \left[A^{(2)} \right] = \begin{pmatrix} w_{11}^{(2)} & w_{12}^{(2)} \cdots & w_{1N}^{(2)} \\ w_{21}^{(2)} & w_{22}^{(2)} \cdots & w_{2N}^{(2)} \\ \vdots & \vdots & \vdots \\ w_{N1}^{(2)} & w_{N2}^{(2)} \cdots & w_{NN}^{(2)} \end{pmatrix}$$

şeklinde tanımlanan matrislerle, (3.77) eşitliği

$$\left[A^{(2)} \right] = \left[A^{(1)} \right] \left[A^{(1)} \right] \quad (3.78)$$

olarak yazılabilir.

Böylece ikinci mertebeden türevin ağırlık katsayılarının, birinci mertebeden türevin ağırlık katsayıları matrislerinin çarpımıyla hesaplanabilir olduğu görülür. Bu formülü genelleştirmek gerekirse, $f(x)$ fonksiyonunun m . mertebeden türevi

$$\frac{\partial^{(m)} f}{\partial x^{(m)}} = \frac{\partial}{\partial x} \left(\frac{\partial^{(m-1)} f}{\partial x^{(m-1)}} \right) = \frac{\partial^{(m-1)}}{\partial x^{(m-1)}} \left(\frac{\partial f}{\partial x} \right) \quad (3.79)$$

şeklinde yazılabilir. $[A^{(m-1)}]$, $[A^{(m)}]$ ağırlık katsayı matrisleri sırasıyla $(m-1)$. ve m . mertebeden türevlere ait olsun. (3.79) eşitliğine DKM eşitliğinin uygulanmasıyla $m = 2, 3, \dots, N-1$ için

$$[A^{(m)}] = [A^{(1)}][A^{(m-1)}] = [A^{(m-1)}][A^{(1)}] \quad (3.80)$$

eşitliği elde edilecektir.

3.4 Hata Analizi

3.4.1 Fonksiyon Yaklaşımı

$f(x)$ fonksiyonuna, $(N-1)$. dereceden Lagrange interpolasyon polinomu

$$P_N f = \sum_{i=1}^N f(x_i) r_i(x) \quad (3.81)$$

ile yaklaşılsın. (3.81) eşitliğindeki Lagrange interpolasyon polinomu (3.35) eşitliğinde tanımlanmıştı. Buna göre $f(x)$ fonksiyonunun yaklaşım hatası

$$E(f) = f(x) - P_N f \quad (3.82)$$

olarak tanımlanır.

Eğer $f(x)$ fonksiyonunun N . mertebeden türevi K gibi bir sabit ise bu fonksiyon

$$f(x) = m_0 + m_1 x + m_2 x^2 + \dots m_{N-1} x^{N-1} + \frac{K x^N}{N!} \quad (3.83)$$

şeklinde ifade edilebilir. (3.81) eşitliği, derecesi $(N-1)$ 'e eşit veya küçük olan polinomlar tarafından sağlandığından $k = 0, 1, \dots, N-1$ için

$$E(x^k) = 0 \quad (3.84)$$

elde edilir. Buna göre (3.84) eşitliğini kullanılarak (3.83) eşitliğini (3.82) eşitliğinde yerine yazalım:

$$\begin{aligned}
E(f) &= f(x) - P_N f \\
&= m_0 + m_1 x + m_2 x^2 + \dots m_{N-1} x^{N-1} + \frac{Kx^N}{N!} - \sum_{i=1}^N f(x_i) r_i(x) \\
&= m_0 + m_1 x + \dots m_{N-1} x^{N-1} + \frac{Kx^N}{N!} - f(x_1) r_1(x) - f(x_2) r_2(x) - \dots - f(x_N) r_N(x) \\
&= m_0 + m_1 x + \dots m_{N-1} x^{N-1} + \frac{Kx^N}{N!} - \left\{ m_0 + m_1 x_1 + \dots m_{N-1} x_1^{N-1} + \frac{Kx_1^N}{N!} \right\} r_1(x) - \dots \\
&\quad \dots - \left\{ m_0 + m_1 x_N + m_2 x_N^2 + \dots m_{N-1} x_N^{N-1} + \frac{Kx_N^N}{N!} \right\} r_N(x) \\
&= m_0 \{1 - r_1(x) - r_2(x) - \dots - r_N(x)\} + m_1 \{x - x_1 r_1(x) - x_2 r_2(x) - \dots - x_N r_N(x)\} + \\
&\quad + m_2 \{x^2 - x_1^2 r_1(x) - x_2^2 r_2(x) - \dots - x_N^2 r_N(x)\} + \dots + \frac{K}{N!} \{x^N - x_1^N r_1(x) - x_2^N r_2(x) - \dots - x_N^N r_N(x)\}
\end{aligned}$$

$E(f)$ açılımında $m_0, m_1, \dots, m_{N-1}$ katsayı parantezlerine alınmış polinomların dereceleri $(N-1)$ dir. (3.84) eşitliğine göre $k = 0, 1, \dots, N-1$ dereceli polinomlar için $E(x^k) = 0$ olacaktır. Geriye sadece

$$E(f) = \frac{K}{N!} \{x^N - x_1^N r_1(x) - x_2^N r_2(x) - \dots - x_N^N r_N(x)\}$$

kalacaktır. Buna göre

$$E(x^N) = x^N - \sum_{i=1}^N x_i^N r_i(x)$$

olarak yazılırsa

$$E(f) = \frac{K}{N!} E(x^N) \quad (3.85)$$

elde edilir.

Diğer taraftan, $M(x)$ polinomunun derecesinin N olduğunu ve en yüksek dereceli teriminin katsayısının 1 olduğunu biliyoruz. Buna göre oluşacak $g(x) = x^N - M(x)$ polinomunun derecesi $(N-1)$ olacaktır. Bu $g(x)$ polinomu (3.82) eşitliğinde yazılırsa

$$E(g) = g(x) - P_N g$$

$$E(g) = x^N - M(x) - \sum_{i=1}^N g(x_i) r_i(x)$$

$$E(g) = x^N - M(x) - \sum_{i=1}^N [x_i^N - M(x_i)] r_i(x) \quad (3.86)$$

elde edilir. $i = 1, 2, \dots, N$ için x_i noktaları $M(x)$ polinomunun kökleri olduğundan $M(x_i) = 0$ olacaktır. Böylece

$$E(g) = x^N - M(x) - \sum_{i=1}^N x_i^N r_i(x)$$

elde edilir. $g(x)$ polinomu ve $r_i(x)$ taban polinomu $(N-1)$. derecedendir. (3.84) eşitliğine göre $E(g) = 0$ olacaktır. Buna göre

$$x^N - M(x) - \sum_{i=1}^N x_i^N r_i(x) = 0$$

$$x^N - \sum_{i=1}^N x_i^N r_i(x) = M(x) \quad (3.87)$$

olacaktır. Dolayısıyla (3.87) eşitliği (3.85) eşitliğinde yerine yazıldığında

$$E(f) = \frac{K}{N!} M(x) \quad (3.88)$$

elde edilecektir.

$f(x)$ fonksiyonunun N . mertebeden türevinin sabit olmadığını ancak sınırlı olduğu durumu ele alalım. Bu durumlarda $E(f)$ hata analizi için başka bir metot kullanılmaktadır.

$\phi(x) = P_N f$ olmak üzere $F(z)$ fonksiyonu

$$F(z) = f(z) - \phi(z) - cM(z) \quad (3.89)$$

şeklinde tanımlansın. $z = x_1, x_2, \dots, x_N$ olduğunda $F(z) = 0$ olacaktır. Eğer $F(x) = 0$ alınırsa

$$F(x) = f(x) - \phi(x) - cM(x) = 0$$

$$f(x) - \phi(x) = cM(x)$$

olup

$$E(f) = f(x) - P_N f = f(x) - \phi(x) = cM(x) \quad (3.90)$$

elde edilecektir. Böylece $F(z)$ fonksiyonunun tanım kümesinde $x, x_1, x_2, \dots, x_N$ şeklinde $(N+1)$ tane kökü vardır. Buna göre, Rolle Teoremi'nin tekrarlanarak uygulanmasıyla $F(z)$ fonksiyonunun N . mertebeden türevinin x_1 ve x_N noktaları arasında en az bir kökü bulunur. Böylece $F^{(N)}(\xi) = 0$ olacak şekilde en az bir $\xi \in (x_1, x_N)$ kökü vardır.

$\phi(z)$, $(N-1)$. dereceden bir polinom olduğuna göre (3.89) eşitliğinden

$$F(z) = f(z) - \phi(z) - cM(z)$$

$$F'(z) = f'(z) - \phi'(z) - cM'(z)$$

$$F''(z) = f''(z) - \phi''(z) - cM''(z)$$

$\vdots$

$$F^{(N-1)}(z) = f^{(N-1)}(z) - \phi^{(N-1)}(z) - cM^{(N-1)}(z)$$

türevleri elde edilir. $\phi(x) = P_N f$ olup, $(N-1)$. dereceden bir polinom olduğundan $\phi^{(N)}(z) = 0$ olacaktır. Buna göre

$$\begin{aligned} F^{(N)}(z) &= f^{(N)}(z) - \phi^{(N)}(z) - cM^{(N)}(z) \\ F^{(N)}(\xi) &= f^{(N)}(\xi) - cM^{(N)}(\xi) \end{aligned} \quad (3.91)$$

ve $F^{(N)}(\xi) = 0$ olduğundan

$$c = \frac{f^{(N)}(\xi)}{M^{(N)}(\xi)}$$

bulunur. $M^{(N)}(\xi) = N!$ bulunacağından

$$c = \frac{f^{(N)}(\xi)}{N!} \quad (3.92)$$

elde edilir. Böylece hata fonksiyonu $\xi = \xi(x)$ şeklinde bir fonksiyon olmak üzere

$$E(f) = \frac{f^{(N)}(\xi)M(x)}{N!} \quad (3.93)$$

olarak bulunur.

3.4.2 Türev Yaklaşımı

m . mertebeden türev yaklaşımı için hata

$$E_D^{(m)}(f) = \frac{\partial^m f}{\partial x^m} - \frac{\partial^m (P_N f)}{\partial x^m} = \frac{\partial^m f}{\partial x^m} - \frac{\partial^m \phi}{\partial x^m} \quad (3.94)$$

olarak tanımlanır. (3.93) eşitliği kullanılarak (3.94) eşitliği

$$E_D^{(m)}(f) = \frac{\partial^m f}{\partial x^m} - \frac{\partial^m \phi}{\partial x^m} = \frac{\partial^m (f - \phi)}{\partial x^m} = \frac{\partial^m (cM(x))}{\partial x^m}$$

$$E_D^{(m)}(f) = \frac{\partial^m \left(\frac{f^{(N)}(\xi)M(x)}{N!} \right)}{\partial x^m}$$

$$E_D^{(m)}(f) = \frac{1}{N!} \frac{\partial^m \left(f^{(N)}(\xi) M(x) \right)}{\partial x^m} \quad (3.95)$$

olarak yazılabilir. $\xi(x)$ bilinmeyen bir fonksiyon olduğundan (3.95) eşitliğini kullanarak $E_D^{(m)}(f)$ hatasını hesaplamak zordur. Özel bir durum olarak $f(x)$ 'in N . mertebeden türevi K gibi bir sabit olarak kabul edilsin. Buna göre (3.95) eşitliği

$$E_D^{(m)}(f) = \frac{K}{N!} M^{(m)}(x) \quad (3.96)$$

olacaktır. Bu durumda türev yaklaşımının hata dağılımı

$$E_D^{(m)}[f(x_i)] = \frac{K}{N!} M^{(m)}(x_i) \quad (3.97)$$

eşitliği ile elde edilecektir. (3.72) eşitliğinde bulunan m . mertebeden türevin diyagonal ağırlık katsayıları kullanılarak $M^{(m)}(x_i)$ değeri

$$M^{(m)}(x_i) = m w_{ii}^{(m-1)} M'(x_i) \quad (3.98)$$

olarak hesaplanır. Buna göre (3.98) eşitliği (3.97) eşitliğinde yerine yazılırsa

$$E_D^{(m)}[f(x_i)] = \frac{K m w_{ii}^{(m-1)} M'(x_i)}{N!}, \quad m = 1, 2, \dots, N-1 \quad (3.99)$$

elde edilir. Sonuçta, $f^{(N)}(\xi) = K$ sabiti olarak alındığında (3.99) eşitliği elde edilecektir. $f^{(N)}(\xi)$ fonksiyonunun sabit alınmadığı durumlarda ise fonksiyon yaklaşımında kullanılan metot kullanılmaktadır.

$g(z) = f(z) - \phi(z)$ fonksiyonunun tanım aralığında N tane kökü olsun. Rolle Teoremi'ne göre, $g^{(m)}(z)$ fonksiyonunun bu aralıkta $\overline{x_1}, \overline{x_2}, \dots, \overline{x_{N-m}}$ şeklinde $(N-m)$ tane kökü vardır. Buna göre $F^{(m)}(z)$ fonksiyonu

$$F^{(m)}(z) = g^{(m)}(z) - \overline{cM(z)}$$

$$F^{(m)}(z) = f^{(m)}(z) - \phi^{(m)}(z) - \overline{cM(z)} \quad (3.100)$$

olarak yazılabilir. Bu eşitlikteki $\overline{M(z)}$ fonksiyonu

$$\overline{M(z)} = (z - \overline{x_1})(z - \overline{x_2}) \dots (z - \overline{x_{N-m}})$$

şeklinde dir. $\overline{x} \neq \overline{x_1}, \overline{x_2}, \dots, \overline{x_{N-m}}$ olmak üzere eğer $F^{(m)}(\overline{x}) = 0$ ise $F^{(m)}(z)$ polinomunun $(N - m + 1)$ tane kökü olacaktır ve

$$F^{(m)}(\overline{x}) = f^{(m)}(\overline{x}) - \phi^{(m)}(\overline{x}) - \overline{cM}(\overline{x}) = 0$$

$$f^{(m)}(\overline{x}) - \phi^{(m)}(\overline{x}) = \overline{cM}(\overline{x})$$

$$E_D^{(m)}[f(\overline{x})] = f^{(m)}(\overline{x}) - \phi^{(m)}(\overline{x}) = \overline{cM}(\overline{x}) \quad (3.101)$$

olur. Rolle Teoremi $(N - m)$ kez uygulanırsa, $F^{(m)}(z)$ fonksiyonunun $(N - m)$. mertebeden türevi, en az bir $\overline{\xi}$ kökü bulundurur ve buradan

$$F^{(m)}(z) = f^{(m)}(z) - \phi^{(m)}(z) - \overline{cM}(z)$$

$$F^{(m)}(\overline{\xi}) = f^{(m)}(\overline{\xi}) - \phi^{(m)}(\overline{\xi}) - \overline{cM}(\overline{\xi})$$

$$F^{(m+1)}(\overline{\xi}) = f^{(m+1)}(\overline{\xi}) - \phi^{(m+1)}(\overline{\xi}) - \overline{cM'}(\overline{\xi})$$

$$F^{(m+2)}(\overline{\xi}) = f^{(m+2)}(\overline{\xi}) - \phi^{(m+2)}(\overline{\xi}) - \overline{cM''}(\overline{\xi})$$

⋮

$$F^{(m+N-m)}(\overline{\xi}) = f^{(m+N-m)}(\overline{\xi}) - \phi^{(m+N-m)}(\overline{\xi}) - \overline{cM^{(N-m)}}(\overline{\xi})$$

$$F^{(N)}(\overline{\xi}) = f^{(N)}(\overline{\xi}) - \phi^{(N)}(\overline{\xi}) - \overline{cM^{(N-m)}}(\overline{\xi})$$

bulunur. $\overline{M}(z)$ fonksiyonunun $(N - m)$ tane kökü olduğundan bu polinomun derecesi $(N - m)$ dir ve bu fonksiyonun $(N - m)$ kez türetilmesi sonucu

$$\overline{M^{(N-m)}}(z) = (N - m)!$$

elde edilecektir. $\phi(x)$ fonksiyonu $(N-1)$ dereceden bir fonksiyon olduğundan $\phi^{(N)}(x) = 0$ olur. $F^{(N-m)}(\bar{\xi}) = 0$ olduğu da bilindiğine göre

$$0 = f^{(N)}(\bar{\xi}) - 0 - \bar{c}(N-m)!$$

$$f^{(N)}(\bar{\xi}) = \bar{c}(N-m)!$$

$$\bar{c} = \frac{f^{(N)}(\bar{\xi})}{(N-m)!} \quad (3.102)$$

bulunur. Bu değer (3.101) eşitliğinde yazılarak

$$E_D^{(m)}[f(\bar{x})] = \frac{f^{(N)}(\bar{\xi})}{(N-m)!} \bar{M}(\bar{x}) \quad (3.103)$$

elde edilir. Bu eşitlik, türev yaklaşım hatasının hesaplanmasında kullanılmaktadır.

Eğer tüm koordinatlarının h aralığında ve $f(x)$ fonksiyonunun N . mertebeden türevinin sınırlı olduğu kabul edilirse C pozitif bir sabit olmak üzere

$$|f^{(N)}(\bar{\xi})| \leq C$$

$$|M^{(m)}(x)| \leq N(N-1)\dots(N-m+1)h^{N-m}$$

$$|\bar{M}(\bar{x})| \leq h^{N-m}$$

olur. Buna göre (3.97) ve (3.103) eşitlikleri

$$|E_D^{(m)}(f)| \leq \frac{Ch^{N-m}}{(N-m)!}, \quad m = 1, 2, \dots, N-1 \quad (3.104)$$

şeklini alır. Aynı zamanda türev yaklaşımının hata dağılımı (3.95) eşitliği kullanılarak da elde edilebilir.

3.5 Grid Nokta Dağılımları için Belirlenen Seçimler

Burada ağırlık katsayı formüllerini belirlemek üzere üç çeşit grid nokta seçiminden söz edilecektir.

3.5.1 Düzgün (Uniform) Grid Nokta Seçimi

Bu tür grid nokta seçiminde adım uzunluğu sabit alınmaktadır. $0 \leq x \leq a$ aralığında tanımlı olmak üzere $i = 1, 2, \dots, N$ için

$$x_i = \frac{i-1}{N-1}a \quad (3.105)$$

eşitliği ile grid noktalar hesaplanır. Buna göre iki grid nokta arası uzaklık

$$\Delta x = x_2 - x_1 = x_i - x_{i-1} = x_N - x_{N-1}$$

olmak üzere

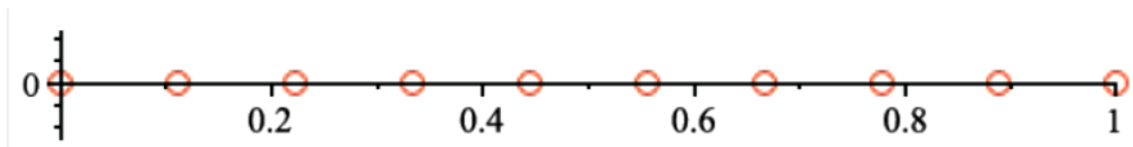
$$x_j - x_i = (j-i)\Delta x$$

şeklindedir.

Grid nokta sayısı $N = 10$ alındığı durumda düzgün nokta dağılımı Çizelge 3.1’ de ve bu noktaların gösterimi Şekil 3.1’de gösterilmiştir.

Çizelge 3.1 Düzgün dağılım ile $N = 10$ nokta için grid noktaların dağılımı

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0.0000	0.1111	0.2222	0.3333	0.4444	0.5556	0.6667	0.7778	0.8889	1.0000



Şekil 3.1 Düzgün dağılım ile $N = 10$ nokta için grid noktaların dağılımının gösterimi

3.5.2 Düzgün Olmayan (Non-Uniform) Grid Nokta Seçimi

$0 \leq x \leq a$ aralığında tanımlı olmak üzere grid noktalar $i = 1, 2, \dots, N$ için

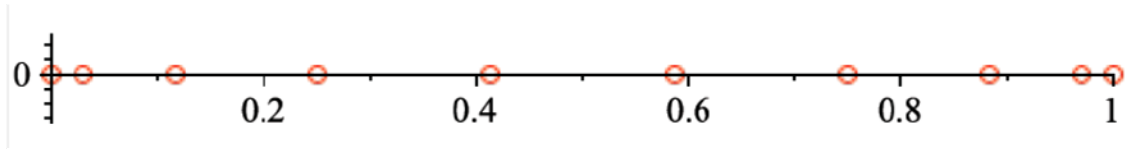
$$x_i = \frac{1 - \cos\left[\frac{(i-1)\pi}{N-1}\right]}{2} a \quad (3.106)$$

eşitliği ile hesaplanır [3].

Grid nokta sayısı $N = 10$ alındığı durumda düzgün olmayan nokta dağılımı Çizelge 3.2’ de ve bu noktaların gösterimi Şekil 3.2’ de gösterilmiştir.

Çizelge 3.2 Düzgün olmayan dağılım ile $N = 10$ nokta için grid noktaların dağılımı

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0.0000	0.0302	0.1170	0.2500	0.4132	0.5868	0.7500	0.8830	0.9698	1.0000



Şekil 3.2 Düzgün olmayan dağılım ile $N = 10$ nokta için grid noktalarının dağılımının gösterimi

3.5.3 Chebyshev-Gauss-Lobatto Grid Nokta Seçimi

Bu tür grid nokta seçiminde $i = 0, 1, \dots, N$ için

$$x_i = \cos\left(\frac{i\pi}{N}\right) \quad (3.107)$$

eşitliği kullanılır. Bu eşitlikteki x_i grid noktaları $[-1, 1]$ aralığında değer almaktadır.

Kabul edelim ki $[-1, 1]$ aralığındaki grid noktalar ξ_i olarak tanımlansın. Bu grid noktalar kullanılarak $[a, b]$ aralığındaki x_i noktalarını hesaplamak için

$$x_i = \frac{b-a}{2}(1-\xi_i) + a \quad (3.108)$$

koordinat dönüşümü kullanılır. Böylece $[-1, 1]$ aralığındaki ξ_i noktaları, $[a, b]$ aralığındaki x_i noktalarına dönüştürülmüş olur.

Grid nokta sayısı $N = 10$ alındığı durumda CGL nokta dağılımı Çizelge 3.3’ de ve bu noktaların gösterimi Şekil 3.3’ de gösterilmiştir.

Çizelge 3.3 CGL dağılımı ile $N = 10$ nokta için grid noktaların dağılımı

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}
0.0000	0.0245	0.0955	0.2061	0.3455	0.5000	0.6545	0.7939	0.9045	0.9755	1.0000



Şekil 3.3 CGL dağılımı ile $N = 10$ nokta için grid noktalarının dağılımının gösterimi

BÖLÜM 4

ADI DİFERENSİYEL DENKLEMLERİN POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME METODU İLE ÇÖZÜMÜ

Bu bölümde başlangıç ve sınır değer problemleri olmak üzere birinci, ikinci ve dördüncü mertebeden adi diferensiyel denklemler PDKM ile çözülecektir. Elde edilen yaklaşık çözümler kesin çözümlerle mutlak hata kullanılarak karşılaştırılacaktır. DKM yaklaşımlarından Shu genel yaklaşımı ve yüksek mertebeden türevlerin ağırlık katsayı matrislerinin hesaplanmasında matris çarpımı kullanılmıştır. Maple 13 programı kullanılarak yaklaşık değerler 16 digitte hesaplanmıştır. Çizelgelerde kullanılan değerlerde ise beş veya yedi basamağa yuvarlatma yapılmıştır.

4.1 Birinci Mertebeden Değişken Katsayılı Diferensiyel Denklemlerin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

Birinci mertebeden değişken katsayılı diferensiyel denklem $0 \leq x \leq \alpha$ olmak üzere

$$p(x)y'(x) + q(x)y(x) = f(x) \quad (4.1)$$

ve

$$y(0) = A \quad (4.2)$$

ise PDKM'yi kullanarak N nokta için yaklaşık çözümü elde edelim.

(4.1) eşitliğindeki birinci mertebeden değişken katsayılı diferensiyel denkleme $i = 1, 2, \dots, N$ olmak üzere x_i grid noktaları uygulanırsa

$$p(x_i)y'(x_i) + q(x_i)y(x_i) = f(x_i) \quad (4.3)$$

elde edilir. (4.3) eşitliğindeki diferansiyel denklemin birinci mertebeden türevi yerine (3.1) DKM eşitliği yazılırsa

$$p(x_i) \sum_{j=1}^N a_{ij} y(x_j) + q(x_i) y(x_i) = f(x_i) \quad (4.4)$$

olacaktır. $p(x_i) = p_i$, $q(x_i) = q_i$, $f(x_i) = f_i$ ve $y(x_i) = y_i$ olarak gösterilsin. Buna göre (4.4) eşitliği her bir $i, j = 1, 2, \dots, N$ için açıkça ifade edilirse

$$\begin{aligned} p_1 \{a_{11}y_1 + a_{12}y_2 + \dots + a_{1N}y_N\} + q_1y_1 &= f_1 \\ p_2 \{a_{21}y_1 + a_{22}y_2 + \dots + a_{2N}y_N\} + q_2y_2 &= f_2 \\ \vdots \\ p_N \{a_{N1}y_1 + a_{N2}y_2 + \dots + a_{NN}y_N\} + q_Ny_N &= f_N \end{aligned}$$

olmak üzere N bilinmeyenli N denklem elde edilir. Bu N bilinmeyenli N denklem matris ve vektör formunda

$$\begin{bmatrix} p_1a_{11} + q_1 & p_1a_{12} & \dots & p_1a_{1N} \\ p_2a_{21} & p_2a_{22} + q_2 & \dots & p_2a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ p_Na_{N1} & p_Na_{N2} & \dots & p_Na_{NN} + q_N \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \quad (4.5)$$

şeklinde yazılır. (4.5) eşitliğindeki lineer denklem sisteminin katsayılar matrisi, birinci mertebeden değişken katsayılı diferansiyel denklemler için daima aynı yapıda bulunmaktadır. Dolayısıyla bu genel yapıda verilen sisteme, diferansiyel denklemin koşullarının uygulanmasıyla denkleme ait yaklaşık çözümlere ulaşılabilir. Bu diferansiyel denklem için verilen koşul (4.2) eşitliğindeki $y(0) = A$ olduğuna göre $x_1 = 0$ olacağından

$$y(x_1) = y_1 = A \quad (4.6)$$

yazılabilir. Bu koşulu (4.5) eşitliğindeki sistemde yazmak gerekmektedir. Eğer koşuldan elde edilen eşitlik sisteme eklenirse N bilinmeyenli $N+1$ denklem olacaktır ve ağırlıklı katsayılar matrisi kare matris formunda olmayacaktır. Bu yüzden (4.5)

eşitliğindeki sistemden bir denklem çıkarılmalı ve çıkarılan bu denklemin yerine koşuldan elde edilen (4.6) eşitliği yazılmalıdır. Böylece

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ p_2 a_{21} & p_2 a_{22} + q_2 & \cdots & p_2 a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ p_N a_{N1} & p_N a_{N2} & \cdots & p_N a_{NN} + q_N \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} A \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \quad (4.7)$$

denklem sistemi elde edilir. Bu denklem sistemi çözülerek $i = 1, 2, \dots, N$ için y_i yaklaşık çözüm değerleri elde edilir.

Örnek 4.1

$$y'(x) = 1 + 2x \sin(x^2)$$

$$y(0) = 0$$

olarak verilen birinci mertebeden adi diferensiyel denklemi $0 \leq x \leq 1$ aralığında PDKM ile

- i) Düzgün (uniform) grid nokta dağılımını kullanarak $N = 5$, $N = 9$ ve $N = 13$ nokta sayılarının her biri için yaklaşık değerleri hesaplayalım ve bu yaklaşık değerleri kesin değerlerle karşılaştıralım.
- ii) Düzgün olmayan (non-uniform) grid nokta dağılımını kullanarak $N = 5$ ve $N = 13$ nokta sayıları için elde edilen yaklaşık değerleri kesin değerlerle karşılaştıralım.
- iii) CGL grid nokta dağılımını kullanarak $N = 5$ ve $N = 13$ nokta sayıları için elde edilen yaklaşık değerleri kesin değerlerle karşılaştıralım.

Çözüm

Verilen birinci mertebeden adi diferensiyel denklemi PDKM ile çözmek üzere denkleme $i = 1, 2, 3, \dots, N$ için x_i grid noktalarını uygulayalım:

$$y'(x_i) = 1 + 2x_i \sin(x_i^2).$$

Denklemin sol tarafına birinci mertebeden DKM eşitliğini yazarsak

$$\sum_{j=1}^N a_{ij} y(x_j) = 1 + 2x_i \sin(x_i^2)$$

elde edilecektir. Bu eşitlik her bir $i, j = 1, 2, 3, \dots, N$ için tek tek yazıldığında karşımıza N bilinmeyenli N tane denklem çıkacaktır. Bu denklemlerin lineer denklem sistemi olarak yazılabilmesi için bilinmeyenlerin katsayılarının matris formunda, sağ taraftaki bilinen değerlerin ise vektör formunda yazılması gerekmektedir. Böylece oluşacak lineer denklem sistemi

$$[a_{ij}]_{N \times N} [y(x_j)]_{N \times 1} = [1 + 2x_i \sin(x_i^2)]_{N \times 1}$$

şeklinde olacaktır. Başlangıç koşulu olarak verilen $y(0) = 0$ değeri ise

$$y(0) = y(x_1) = 0$$

olarak yazılır. Verilen bu başlangıç koşulu yardımıyla bilinmeyen sayısı $N - 1$ olurken, oluşan denklem sistemi de $(N - 1) \times (N - 1)$ boyutuna indirgenebilir. Ancak tüm eşitliklerdeki $y(x_1)$ değerini ağırlık katsayısıyla çarpıp sağ tarafa atmaktansa sisteme bir eşitlik gibi eklemek daha pratik olacaktır.

Yukarıdaki adi diferensiyel denklemin analitik çözümü

$$y(x) = x - \cos(x^2) + 1$$

şeklindedir. Buna göre N grid nokta sayısına bazı değerler vererek ve farklı grid nokta dağılımları kullanarak yaklaşık çözümler elde etmeye çalışalım ve bu yaklaşık çözümleri kesin çözümlerle karşılaştıralım.

- i) Öncelikle düzgün (uniform) grid nokta dağılımını kullanalım. Düzgün grid nokta dağılımında $i = 1, 2, \dots, N$ için Bölüm 3'te (3.105) eşitliğiyle verilen

$$x_i = \frac{i-1}{N-1}$$

eşitliği kullanılmaktadır. Buna göre $N = 5$ nokta alındığında grid noktalardan elde edilen ağırlıklı katsayılar matrisiyle oluşan denklem sistemi

$$\begin{bmatrix} -\frac{25}{3} & 16 & -12 & \frac{16}{3} & -1 \\ -1 & -\frac{10}{3} & 6 & -2 & \frac{1}{3} \\ \frac{1}{3} & -\frac{8}{3} & 0 & \frac{8}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 2 & -6 & \frac{10}{3} & 1 \\ 1 & -\frac{16}{3} & 12 & -16 & \frac{25}{3} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} 1 + 2x_1 \sin(x_1^2) \\ 1 + 2x_2 \sin(x_2^2) \\ 1 + 2x_3 \sin(x_3^2) \\ 1 + 2x_4 \sin(x_4^2) \\ 1 + 2x_5 \sin(x_5^2) \end{bmatrix}$$

şeklinde. Denklem sistemindeki ağırlıklı katsayılar matrisi Shu'nun genel yaklaşımında kullanılan Bölüm 3'teki (3.47) ve (3.49) eşitliklerinden elde edilmiştir. Bu denklem sistemindeki ağırlıklı katsayılar matrisinin determinantı sıfırdır. Bu matrisin determinantının sıfır olduğunu determinantın özelliklerinden söyleyebiliriz. Bu özelliği kısaca şu şekilde ifade edebiliriz: $j = 1, 2, \dots, N$ için β_j en az biri sıfırdan farklı olan skaler değerler olmak üzere eğer her $i = 1, 2, \dots, N$ için

$$\sum_{j=1}^N \beta_j a_{ij} = 0$$

koşulu sağlanıyorsa bu matrisin determinantı sıfıra eşittir. Bu koşul, ağırlıklı katsayılar matrisinin köşegen elemanları elde edilirken kullanılan Bölüm 3'teki (3.49) eşitliğine benzemektedir. (3.49) eşitliğinde $j = 1, 2, \dots, N$ için $\beta_j = 1$ değerlerini almaktadır. Buna göre yukarıdaki 5×5 boyutlu ağırlıklı katsayılar matrisinin determinantı sıfırdır. Dolayısıyla matrisin tersi yoktur. Bu denklem sisteminin tek bir çözümünün olabilmesi için başlangıç koşulunun sisteme eklenmesi gerekmektedir. Böylece matris tekil olmaktan çıkacak ve sistemin tek bir çözümü var olacaktır. Başlangıç koşulunu sisteme eklemek için sistemden bir eşitlik çıkarılmalıdır. Çıkarılacak bu eşitliği seçerken dikkat edilmesi gerekmektedir. Çünkü atılan her bir satır, yaklaşık çözümün kesin çözüme olan yakınsaklığını etkilemektedir. Buna göre başlangıç koşulunu denklem sistemine eklediğimizde

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & -\frac{10}{3} & 6 & -2 & \frac{1}{3} \\ \frac{1}{3} & -\frac{8}{3} & 0 & \frac{8}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 2 & -6 & \frac{10}{3} & 1 \\ 1 & -\frac{16}{3} & 12 & -16 & \frac{25}{3} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} 0.00000 \\ 1.03123 \\ 1.24740 \\ 1.79995 \\ 2.68294 \end{bmatrix}$$

elde edilir. Bu lineer denklem sisteminin çözülmesiyle elde edilen yaklaşık değerleri, analitik çözüm sonucu elde edilen değerlerle karşılaştıralım. Bu değerler ve mutlak hata Çizelge 4.1’ de verilmiştir.

Çizelge 4.1 Düzgün dağılım ile $N = 5$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.00	0.00000	0.00000	0.00000
0.25	0.25195	0.26638	0.01443
0.50	0.53109	0.54414	0.01305
0.75	0.90408	0.91811	0.01404
1.00	1.45970	1.47165	0.01196

Düzgün grid nokta dağılımını $N = 9$ nokta için uygulayalım. Buna göre ağırlıklı katsayılar matrisinin ilk satırını silip yerine başlangıç koşulunu eklediğimizde oluşacak denklem sistemi

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -\frac{446}{35} & 28 & -28 & \frac{70}{3} & -14 & \frac{28}{5} & -\frac{4}{3} & \frac{1}{7} \\
\frac{1}{7} & -\frac{16}{7} & -\frac{38}{5} & 16 & -10 & \frac{16}{3} & -2 & \frac{16}{35} & -\frac{1}{21} \\
-\frac{1}{21} & \frac{4}{7} & -4 & -\frac{18}{5} & 10 & -4 & \frac{4}{3} & -\frac{2}{7} & \frac{1}{35} \\
\frac{1}{35} & -\frac{32}{105} & \frac{8}{5} & -\frac{32}{5} & 0 & \frac{32}{5} & -\frac{8}{5} & \frac{32}{105} & -\frac{1}{35} \\
-\frac{1}{35} & \frac{2}{7} & -\frac{4}{3} & 4 & -10 & \frac{18}{5} & 4 & -\frac{4}{7} & \frac{1}{21} \\
\frac{1}{21} & -\frac{16}{35} & 2 & -\frac{16}{3} & 10 & -16 & \frac{38}{5} & \frac{16}{7} & -\frac{1}{7} \\
-\frac{1}{7} & \frac{4}{3} & -\frac{28}{5} & 14 & -\frac{70}{3} & 28 & -28 & \frac{446}{35} & 1 \\
1 & -\frac{64}{7} & \frac{112}{3} & -\frac{448}{5} & 140 & -\frac{448}{3} & 112 & -64 & \frac{761}{35}
\end{bmatrix}
\begin{bmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_5 \\
y_6 \\
y_7 \\
y_8 \\
y_9
\end{bmatrix}
=
\begin{bmatrix}
0.00000 \\
1.00391 \\
1.03123 \\
1.10512 \\
1.24740 \\
1.47596 \\
1.79995 \\
2.21273 \\
2.68294
\end{bmatrix}$$

şeklinde. Bu denklem sisteminin çözümüyle elde edilen yaklaşık değerlerin, analitik değerlerle karşılaştırılması Çizelge 4.2’ de verilmiştir.

Çizelge 4.2 Düzgün dağılım ile $N = 9$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.000	0.00000	0.00000	0.00000000
0.125	0.12512	0.12509	0.3117×10^{-4}
0.250	0.25195	0.25192	0.3011×10^{-4}
0.375	0.38487	0.38484	0.3035×10^{-4}
0.500	0.53109	0.53106	0.3023×10^{-4}
0.625	0.70033	0.70030	0.3033×10^{-4}
0.750	0.90408	0.90405	0.3011×10^{-4}
0.875	1.15405	1.15402	0.3051×10^{-4}
1.000	1.45970	1.45967	0.2899×10^{-4}

Düzgün grid nokta dağılımını $N = 13$ nokta için uygulayalım. Elde edilen yaklaşık değerlerin kesin değerlerle karşılaştırılması Çizelge 4.3’ de verilmiştir.

Çizelge 4.3 Düzgün dağılım ile $N = 13$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	0.0000000	0.0000000	0.000000000
0.0833	0.0833574	0.0833575	0.4024×10^{-7}
0.1666	0.1670524	0.1670525	0.3874×10^{-7}
0.2500	0.2519525	0.2519525	0.3926×10^{-7}
0.3333	0.3394998	0.3394999	0.3950×10^{-7}
0.4166	0.4316993	0.4316993	0.3886×10^{-7}
0.5000	0.5310876	0.5310876	0.3915×10^{-7}
0.5833	0.6406713	0.6406714	0.3946×10^{-7}
0.6666	0.7638170	0.7638170	0.3889×10^{-7}
0.7500	0.9040755	0.9040755	0.3912×10^{-7}
0.8333	1.0649240	1.0649240	0.3956×10^{-7}
0.9166	1.2494107	1.2494108	0.3911×10^{-7}
1.0000	1.4596977	1.4596977	0.3898×10^{-7}

Düzgün grid nokta dağılımını kullanarak farklı N değerleri için elde edilen ortak grid noktalarındaki yaklaşık değerleri kendi aralarında ve analitik değerlerle karşılaştıralım. Bu karşılaştırma Çizelge 4.4’ de verilmiştir.

Çizelge 4.4 Düzgün dağılım ile farklı N değerlerindeki ortak x noktalarının kesin ve yaklaşık değerleri

Grid Noktalar	Kesin Çözüm	Yaklaşık Çözüm		
		$N = 5$	$N = 9$	$N = 13$
$x = 0.25$	0.25195	0.26638	0.25192	0.25195
$x = 0.50$	0.53109	0.54414	0.53106	0.53109
$x = 0.75$	0.90408	0.91811	0.90405	0.90408
$x = 1.00$	1.45970	1.47165	1.45967	1.45970

Yukarıdaki Çizelge 4.4’ e bakıldığında Örnek 4.1 için grid nokta sayısı artarsa kesin çözüme yakınsaklığın artacağı söylenebilir.

ii) Düzgün olmayan (non-uniform) grid nokta dağılımını kullanarak $N = 5$ nokta için elde edilen yaklaşık değerleri kesin değerlerle karşılaştıralım.

Düzgün olmayan grid nokta dağılımında $i = 1, 2, \dots, N$ için Bölüm 3'te (3.106) eşitliğiyle verilen

$$x_i = \frac{1 - \cos\left[\frac{(i-1)\pi}{N-1}\right]}{2}$$

eşitliği kullanılmaktadır. Buna göre grid nokta sayısını $N = 5$ alıp başlangıç koşulunu sisteme eklediğimizde oluşacak denklem sistemi

$$\begin{bmatrix} 1.00000 & 0.00000 & 0.00000 & 0.00000 & 0.00000 \\ -3.41421 & 1.41421 & 2.82843 & -1.41421 & 0.58579 \\ 0.99999 & -2.82843 & -1.10^{-10} & 2.82843 & -0.99999 \\ -0.58579 & 1.41421 & -2.82843 & -1.41421 & 3.41421 \\ 1.00000 & -2.34314 & 4.00000 & -13.65685 & 11.00000 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} 0.00000 \\ 1.00628 \\ 1.24740 \\ 2.13658 \\ 2.68294 \end{bmatrix}$$

şeklinde dir. Bu denklem sisteminin çözümüyle elde edilen yaklaşık değerlerin, analitik değerlerle karşılaştırılması Çizelge 4.5' de verilmiştir.

Çizelge 4.5 Düzgün olmayan dağılım ile $N = 5$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.00000 #	0.00000 #	0.00000 #	0.000000000 #
0.14645 #	0.14668 #	0.15333 #	0.6653×10^{-2} #
0.50000 #	0.53109 #	0.53190 #	0.8128×10^{-3} #
0.85355 #	1.10742 #	1.11175 #	0.4335×10^{-2} #
1.00000 #	1.45970 #	1.46342 #	0.3723×10^{-2} #

Düzgün olmayan grid nokta dağılımını kullanılarak $N = 13$ nokta için yaklaşık çözümlerin kesin çözümlerle karşılaştırılması Çizelge 4.6' da verilmiştir.

Çizelge 4.6 Düzgün olmayan dağılım ile $N = 13$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.00000	0.00000000	0.00000000	0.0000000000
0.01704	0.01703713	0.01703713	0.1417×10^{-9}
0.06699	0.06699737	0.06699737	0.4029×10^{-11}
0.14645	0.14667658	0.14667658	0.8122×10^{-10}
0.25000	0.25195249	0.25195249	0.1052×10^{-9}
0.37059	0.38000643	0.38000643	0.1064×10^{-9}
0.50000	0.53108758	0.53108758	0.1627×10^{-9}
0.62941	0.70685855	0.70685855	0.3509×10^{-9}
0.75000	0.90407550	0.90407550	0.2589×10^{-9}
0.85355	1.10741507	1.10741507	0.2167×10^{-9}
0.93301	1.28857811	1.28857811	0.5621×10^{-9}
0.98296	1.41454614	1.41454614	0.1970×10^{-9}
1.00000	1.45969769	1.45969769	0.1989×10^{-9}

iii) CGL grid nokta dağılımını kullanarak $N = 5$ nokta için elde edilen yaklaşık değerleri analitik değerlerle karşılaştıralım.

CGL grid nokta dağılımında $i = 0, 1, \dots, N$ için Bölüm 3'te (3.107) eşitliğiyle verilen

$$x_i = \cos\left(\frac{i\pi}{N}\right)$$

eşitliği kullanılmaktadır. Buna göre CGL nokta dağılımını kullanarak $N = 5$ nokta için oluşan denklem sistemi

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -5.23607 & 2.34164 & 4.00000 & -1.78885 & 1.23607 & -0.55279 \\ 1.447214 & -3.99999 & 0.34164 & 3.23607 & -1.78885 & 0.76393 \\ -0.76393 & 1.78885 & -3.23607 & -0.34164 & 3.99999 & -1.44721 \\ 0.55279 & -1.23607 & 1.78885 & -4.00000 & -2.34164 & 5.23607 \\ -1.00000 & 2.21115 & -3.05573 & 5.78885 & -20.94427 & 16.99999 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} 0.00000 \\ 1.00174 \\ 1.08228 \\ 1.54376 \\ 2.32035 \\ 2.68294 \end{bmatrix}$$

şeklinde. Bu denklem sisteminin çözülmesiyle elde edilen yaklaşık değerlerin, kesin değerlerle karşılaştırılması Çizelge 4.7' de verilmiştir.

Çizelge 4.7 CGL dağılımı ile $N = 5$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.00000	0.00000	0.00000	0.000000000
0.09549	0.09553	0.09430	0.1233×10^{-2}
0.34549	0.35261	0.35246	0.1430×10^{-3}
0.65451	0.74487	0.74395	0.9187×10^{-3}
0.90451	1.22093	1.22036	0.5653×10^{-3}
1.00000	1.45970	1.45908	0.6156×10^{-3}

CGL grid nokta dağılımını kullanılarak $N = 13$ nokta için elde edilen yaklaşık değerlerin kesin değerlerle karşılaştırılması Çizelge 4.8’ de verilmiştir.

Çizelge 4.8 CGL dağılımı ile $N = 13$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.00000	0.0000000	0.0000000	0.000000000
0.01453	0.0145291	0.0145291	0.8285×10^{-9}
0.05727	0.0572774	0.0572774	0.6915×10^{-10}
0.12574	0.1258696	0.1258696	0.7361×10^{-9}
0.21597	0.2170552	0.2170552	0.1326×10^{-9}
0.32270	0.3281146	0.3281146	0.6451×10^{-9}
0.43973	0.4583683	0.4583683	0.4949×10^{-10}
0.56027	0.6091319	0.6091319	0.5609×10^{-9}
0.67730	0.7806907	0.7806907	0.2508×10^{-10}
0.78403	0.9670898	0.9670898	0.6407×10^{-9}
0.87426	1.152404	1.15240396	0.2156×10^{-9}
0.94273	1.3123344	1.3123344	0.3673×10^{-9}
0.98547	1.4211228	1.4211228	0.1874×10^{-9}
1.00000	1.4596977	1.4596977	0.1144×10^{-10}

4.2 İkinci Mertebeden Değişken Katsayılı Sınır Değer Probleminin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

İkinci mertebeden değişken katsayılı diferensiyel denklem

$$p(x)y''(x) + q(x)y'(x) + r(x)y(x) = f(x) \quad (4.8)$$

ve sınır koşulları $0 \leq x \leq 1$ olmak üzere

$$y(0) = A \text{ ve } y(1) = B \quad (4.9)$$

ise PDKM'yi kullanarak N nokta sayısı için yaklaşık çözümleri elde edelim.

(4.8) eşitliğindeki ikinci mertebeden değişken katsayılı diferensiyel denkleme $i = 1, 2, \dots, N$ için x_i grid noktaları uygulandığında diferensiyel denklem

$$p(x_i)y''(x_i) + q(x_i)y'(x_i) + r(x_i)y(x_i) = f(x_i) \quad (4.10)$$

şeklinde yazılır. Bu denklemin birinci ve ikinci mertebeden türevleri yerine DKM eşitliği uygulanırsa denklem

$$p(x_i) \sum_{j=1}^N w_{ij}^{(2)} y(x_j) + q(x_i) \sum_{j=1}^N a_{ij} y(x_j) + r(x_i) y(x_i) = f(x_i) \quad (4.11)$$

haline dönüşür. Bu eşitlik açıkça ifade edilirse

$$p_1 \{w_{11}^{(2)} y_1 + w_{12}^{(2)} y_2 + \dots + w_{1N}^{(2)} y_N\} + q_1 \{a_{11} y_1 + a_{12} y_2 + \dots + a_{1N} y_N\} + r_1 y_1 = f_1$$

$$p_2 \{w_{21}^{(2)} y_1 + w_{22}^{(2)} y_2 + \dots + w_{2N}^{(2)} y_N\} + q_2 \{a_{21} y_1 + a_{22} y_2 + \dots + a_{2N} y_N\} + r_2 y_2 = f_2$$

$\vdots$

$$p_N \{w_{N1}^{(2)} y_1 + w_{N2}^{(2)} y_2 + \dots + w_{NN}^{(2)} y_N\} + q_N \{a_{N1} y_1 + a_{N2} y_2 + \dots + a_{NN} y_N\} + r_N y_N = f_N$$

şeklinde dir. Bu denklemler düzenlenir ve oluşan N bilinmeyenli N denklem matris ve vektör formunda yazılırsa

$$\begin{bmatrix} p_1 w_{11}^{(2)} + q_1 a_{11} + r_1 & p_1 w_{12}^{(2)} + q_1 a_{12} & \cdots & p_1 w_{1N}^{(2)} + q_1 a_{1N} \\ p_2 w_{21}^{(2)} + q_2 a_{21} & p_2 w_{22}^{(2)} + q_2 a_{22} + r_2 & \cdots & p_2 w_{2N}^{(2)} + q_2 a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ p_N w_{N1}^{(2)} + q_N a_{N1} & p_N w_{N2}^{(2)} + q_N a_{N2} & \cdots & p_N w_{NN}^{(2)} + q_N a_{NN} + r_N \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \quad (4.12)$$

elde edilecektir. Bu lineer denklem sistemindeki katsayılar matrisi ikinci mertebeden değişken katsayılı diferensiyel denklemler için daima aynı yapıdadır. Dolayısıyla bu genel yapıda verilen sisteme, diferensiyel denklemin koşullarının uygulanmasıyla denkleme ait yaklaşık çözümlere ulaşılabilir. Buna göre verilen (4.9) sınır koşulları

$$y(0) = y(x_1) = A \quad \text{ve} \quad y(1) = y(x_N) = B \quad (4.13)$$

olarak yazılabilir. Bu koşulları matris sisteminde yazmak için sistemdeki ilk ve son eşitlik çıkarılıp yerine (4.13) koşul eşitlikleri yazılmalıdır. Böylece

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ p_2 w_{21}^{(2)} + q_2 a_{21} & p_2 w_{22}^{(2)} + q_2 a_{22} + r_2 & \cdots & p_2 w_{2N}^{(2)} + q_2 a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ p_{N-1} w_{(N-1)1}^{(2)} + q_{N-1} a_{(N-1)1} & p_{N-1} w_{(N-1)2}^{(2)} + q_{N-1} a_{(N-1)2} & \cdots & p_{N-1} w_{(N-1)N}^{(2)} + q_{N-1} a_{(N-1)N} \\ 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} A \\ f_2 \\ f_3 \\ \vdots \\ B \end{bmatrix}$$

denklem sistemi oluşacaktır. Bu denklem sisteminin çözülmesi sonucu $i = 1, 2, \dots, N$ için y_i yaklaşık çözüm değerleri elde edilecektir.

Örnek 4.2

$$xy''(x) - 3y'(x) = -8x$$

$$y(0) = 1, \quad y(1) = 4$$

İkinci mertebeden sınır değer probleminin yaklaşık değerlerini $0 \leq x \leq 1$ aralığında $N = 10$ nokta için PDKM ile hesaplayalım.

Çözüm

Yukarıdaki diferensiyel denkleme $i = 1, 2, \dots, N$ için x_i grid noktalarını uygulayalım:

$$x_i y''(x_i) - 3y'(x_i) = -8x_i.$$

Denklemin birinci ve ikinci mertebeden türevlerinin yerine DKM eşitliğini yazarsak

$$x_i \sum_{j=1}^N w_{ij}^{(2)} y(x_j) - 3 \sum_{j=1}^N a_{ij} y(x_j) = -8x_i$$

olacaktır. Bu eşitliği $i, j = 1, 2, \dots, N$ için matris ve vektör formunda yazalım:

$$\left[x_i w_{ij}^{(2)} - 3a_{ij} \right]_{N \times N} \left[y(x_j) \right]_{N \times 1} = [-8x_i]_{N \times 1}.$$

Buradaki ağırlıklı katsayılar matrisini kısaca $[b_{ij}] = [x_i w_{ij}^{(2)} - 3a_{ij}]$ olarak ifade edersek

$$[b_{ij}]_{N \times N} [y(x_j)]_{N \times 1} = [-8x_i]_{N \times 1}$$

yazılabilir. Verilen sınır koşullarını

$$y(0) = y(x_1) = 1$$

$$y(1) = y(x_N) = 4$$

olarak yazalım. Buna göre, denklem sistemine bu sınır koşullarının eklenmesi gerekmektedir. Sınır koşullarının yazılmasından sonra oluşacak sistem

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & b_{(N-1)3} & \ddots & b_{(N-1)N} \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} y(x_1) \\ y(x_2) \\ \vdots \\ y(x_{N-1}) \\ y(x_N) \end{bmatrix} = \begin{bmatrix} 1 \\ -8x_2 \\ \vdots \\ -8x_{N-1} \\ 4 \end{bmatrix}$$

şeklinde. Oluşan bu lineer denklem sisteminin çözülmesiyle $i = 1, 2, \dots, N$ için $y(x_i)$ yaklaşık değerleri elde edilecektir.

Denklemin analitik çözümü

$$y(x) = x^4 + 2x^2 + 1$$

olmak üzere düzgün nokta dağılımını kullanarak $N = 10$ nokta için kesin ve yaklaşık değerleri Çizelge 4.9’ da karşılaştıralım.

Çizelge 4.9 Düzgün dağılım ile $N = 10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	1.0000000	1.0000000	0.000000000
0.1111	1.0248438	1.0248437	0.7865×10^{-7}
0.2222	1.1012041	1.1012040	0.7493×10^{-7}
0.3333	1.2345679	1.2345678	0.7465×10^{-7}
0.4444	1.4340802	1.4340801	0.7378×10^{-7}
0.5556	1.7125438	1.7125438	0.7130×10^{-7}
0.6667	2.0864198	2.0864197	0.6580×10^{-7}
0.7778	2.5758269	2.5758268	0.5914×10^{-7}
0.8889	3.2045420	3.2045419	0.5031×10^{-7}
1.0000	4.0000000	4.0000000	0.000000000

Düzgün olmayan grid nokta dağılımını kullanılarak $N = 10$ nokta için kesin ve yaklaşık değerleri Çizelge 4.10’ de karşılaştıralım.

Çizelge 4.10 Düzgün olmayan dağılım ile $N = 10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	1.0000000	1.0000000	0.000000000
0.0302	1.0018193	1.0018193	0.2637×10^{-10}
0.1170	1.0275549	1.0275549	0.3763×10^{-9}
0.2500	1.1289063	1.1289063	0.2086×10^{-9}
0.4132	1.3705721	1.3705721	0.1982×10^{-9}
0.5868	1.8073106	1.8073106	0.1807×10^{-9}
0.7500	2.4414063	2.4414063	0.3842×10^{-9}
0.8830	3.1674326	3.1674326	0.2900×10^{-10}
0.9698	3.7659356	3.7659356	0.3149×10^{-9}
1.0000	4.0000000	4.0000000	0.000000000

CGL grid nokta dağılımını kullanılarak $N = 10$ nokta için kesin ve yaklaşık değerleri Çizelge 4.11’ de karşılaştıralım.

Çizelge 4.11 CGL dağılımı ile $N = 10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	1.0000000	1.0000000	0.0000000000
0.0245	1.0011981	1.0011981	0.1635×10^{-9}
0.0955	1.0183204	1.0183204	0.2526×10^{-9}
0.2061	1.0867651	1.0867651	0.5197×10^{-10}
0.3455	1.2529766	1.2529766	0.2633×10^{-9}
0.5000	1.5625000	1.5625000	0.9927×10^{-10}
0.6545	2.0402733	2.0402733	0.3967×10^{-9}
0.7939	2.6577656	2.6577656	0.5010×10^{-9}
0.9045	3.3056171	3.3056171	0.2425×10^{-9}
0.9755	3.8089587	3.8089587	0.3575×10^{-9}
1.0000	4.0000000	4.0000000	0.0000000000

4.3 İkinci Mertebeden Değişken Katsayılı Başlangıç Değer Probleminin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

İkinci mertebeden değişken katsayılı diferensiyel denklem

$$p(x)y''(x) + q(x)y'(x) + r(x)y(x) = f(x) \quad (4.14)$$

başlangıç koşulları

$$y(0) = A \text{ ve } y'(0) = B \quad (4.15)$$

olmak üzere PDKM'yi kullanarak N nokta sayısı için yaklaşık çözümleri hesaplayalım.

(4.14) eşitliğindeki ikinci mertebeden diferensiyel denkleme $i = 1, 2, \dots, N$ için x_i grid noktaları uygulanırsa

$$p(x_i)y''(x_i) + q(x_i)y'(x_i) + r(x_i)y(x_i) = f(x_i) \quad (4.16)$$

elde edilir. Bu denklemin birinci ve ikinci mertebeden türevleri yerine DKM eşitliği kullanılırsa denklem

$$p(x_i) \sum_{j=1}^N w_{ij}^{(2)} y(x_j) + q(x_i) \sum_{j=1}^N a_{ij} y(x_j) + r(x_i) y(x_i) = f(x_i) \quad (4.17)$$

şeklinde olacaktır. Bu eşitlik açık bir şekilde ifade edilirse

$$\begin{aligned}
p_1 \{w_{11}^{(2)} y_1 + w_{12}^{(2)} y_2 + \dots + w_{1N}^{(2)} y_N\} + q_1 \{a_{11} y_1 + a_{12} y_2 + \dots + a_{1N} y_N\} + r_1 y_1 &= f_1 \\
p_2 \{w_{21}^{(2)} y_1 + w_{22}^{(2)} y_2 + \dots + w_{2N}^{(2)} y_N\} + q_2 \{a_{21} y_1 + a_{22} y_2 + \dots + a_{2N} y_N\} + r_2 y_2 &= f_2 \\
\vdots \\
p_N \{w_{N1}^{(2)} y_1 + w_{N2}^{(2)} y_2 + \dots + w_{NN}^{(2)} y_N\} + q_N \{a_{N1} y_1 + a_{N2} y_2 + \dots + a_{NN} y_N\} + r_N y_N &= f_N
\end{aligned}$$

olacaktır. Oluşan N bilinmeyenli N denklem matris ve vektör formunda

$$\begin{bmatrix} p_1 w_{11}^{(2)} + q_1 a_{11} + r_1 & p_1 w_{12}^{(2)} + q_1 a_{12} \cdots & p_1 w_{1N}^{(2)} + q_1 a_{1N} \\ p_2 w_{21}^{(2)} + q_2 a_{21} & p_2 w_{22}^{(2)} + q_2 a_{22} + r_2 \cdots & p_2 w_{2N}^{(2)} + q_2 a_{2N} \\ \vdots & \vdots & \vdots \\ p_N w_{N1}^{(2)} + q_N a_{N1} & p_N w_{N2}^{(2)} + q_N a_{N2} \cdots & p_N w_{NN}^{(2)} + q_N a_{NN} + r_N \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{bmatrix}$$

şeklinde yazılır. Başlangıç koşullarından biri birinci mertebeden türevin bir grid noktasındaki değeri olduğundan bu koşula DKM eşitliği uygulanırsa

$$y(0) = y_1 = A \quad (4.18)$$

$$y'(0) = y'(x_1) = \sum_{j=1}^N a_{1j} y(x_j) = B \quad (4.19)$$

olacaktır. Bu başlangıç koşullarını denklem sistemine katmak üzere sistemdeki ilk ve son denklem çıkarılıp yerine (4.18) ve (4.19) koşul eşitlikleri yazılmalıdır. Buna göre

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ p_2 w_{21}^{(2)} + q_2 a_{21} & p_2 w_{22}^{(2)} + q_2 a_{22} + r_2 & \cdots & p_2 w_{2N}^{(2)} + q_2 a_{2N} \\ p_3 w_{31}^{(2)} + q_3 a_{31} & p_3 w_{32}^{(2)} + q_3 a_{32} & \cdots & p_3 w_{3N}^{(2)} + q_3 a_{3N} \\ \vdots & \vdots & \ddots & \vdots \\ p_{N-1} w_{(N-1)1}^{(2)} + q_{N-1} a_{(N-1)1} & p_{N-1} w_{(N-1)2}^{(2)} + q_{N-1} a_{(N-1)2} & \cdots & p_{N-1} w_{(N-1)N}^{(2)} + q_{N-1} a_{(N-1)N} \\ a_{11} & a_{12} & \cdots & a_{1N} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} A \\ f_2 \\ f_3 \\ \vdots \\ B \end{bmatrix} \quad (4.20)$$

denklemler sistemi oluşacaktır. Bu denklemler sistemi çözülerek $i = 1, 2, \dots, N$ için y_i yaklaşık çözüm değerleri elde edilecektir.

Örnek 4.3

$$y''(x) - y'(x) - 2y(x) = 4x^2$$

$$y(0) = 1, \quad y'(0) = 4$$

İkinci mertebeden başlangıç değer probleminin yaklaşık değerlerini $0 \leq x \leq 1$ aralığında $N = 10$ nokta için PDKM ile hesaplayalım.

Çözüm

Diferensiyel denkleme $i = 1, 2, \dots, N$ için x_i grid noktalarını uygulayalım:

$$y''(x_i) - y'(x_i) - 2y(x_i) = 4x_i^2.$$

Denklemin birinci ve ikinci mertebeden türevlerinin yerine DKM eşitliğini yazarsak

$$\sum_{j=1}^N w_{ij}^{(2)} y(x_j) - \sum_{j=1}^N a_{ij} y(x_j) - 2y(x_i) = 4x_i^2$$

elde edilir. Bu eşitliği $i, j = 1, 2, \dots, N$ için matris ve vektör formunda yazalım:

$$\left[w_{ij}^{(2)} - a_{ij} - 2I \right] \left[y(x_j) \right] = \left[4x_i^2 \right].$$

Buradaki ağırlıklı katsayılar matrisini kısaca $\left[b_{ij} \right] = \left[w_{ij}^{(2)} - a_{ij} - 2I \right]$ olarak ifade edersek

$$\left[b_{ij} \right] \left[y(x_j) \right] = \left[4x_i^2 \right]$$

elde edilir. Verilen başlangıç koşullarını

$$y(0) = y(x_1) = 1 \text{ ve}$$

$$y'(0) = y'(x_1) = \sum_{j=1}^N a_{1j} y(x_j) = 4$$

olarak yazalım. Buna göre sisteme bu başlangıç koşullarını girmek gerekmektedir. Bu koşulları ekleyebilmek için sistemden iki eşitlik atılmalı, bunların yerine başlangıç koşullarına ait eşitlikler yazılmalıdır. Koşulların yazılmasından sonra oluşacak sistem

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & b_{(N-1)3} & \ddots & b_{(N-1)N} \\ a_{11} & a_{12} & a_{13} & \cdots & a_{1N} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix} = \begin{bmatrix} 1 \\ 4x_2^2 \\ \vdots \\ 4x_{N-1}^2 \\ 4 \end{bmatrix}$$

şeklindedir. Oluşan bu lineer denklem sisteminin çözülmesiyle $i = 1, 2, \dots, N$ için y_i yaklaşık değerleri elde edilecektir.

Denklemin analitik çözümü

$$y(x) = 2e^{-x} + 2e^{2x} - 2x^2 + 2x - 3$$

şeklindedir. Bu analitik çözümünden elde edilen kesin değerlerle PDKM sonucu farklı dağılımlar kullanılarak elde edilen yaklaşık değerleri karşılaştıralım.

Düzgün grid nokta dağılımı ile $N = 10$ nokta için elde edilen kesin ve yaklaşık değerler Çizelge 4.12’ de verilmiştir.

Çizelge 4.12 Düzgün dağılım ile $N = 10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	1.00000	1.00000	0.000000000
0.1111	1.48491	1.48491	0.2667×10^{-6}
0.2222	2.06640	2.06640	0.7081×10^{-6}
0.3333	2.77298	2.77298	0.1212×10^{-5}
0.4444	3.64104	3.64104	0.1809×10^{-5}
0.5556	4.71680	4.71680	0.2525×10^{-5}
0.6667	6.05861	6.05862	0.3391×10^{-5}
0.7778	7.73997	7.73997	0.4438×10^{-5}
0.8889	9.85314	9.85315	0.5721×10^{-5}
1.0000	12.51387	12.51388	0.7360×10^{-5}

Düzgün olmayan grid nokta dağılımı ile $N = 10$ nokta için elde edilen kesin ve yaklaşık değerler Çizelge 4.13’ de verilmiştir.

Çizelge 4.13 Düzgün olmayan dağılım ile $N = 10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	1.00000	1.00000	0.000000000
0.0302	1.12341	1.12341	0.5086×10^{-8}
0.1170	1.51297	1.51297	0.1851×10^{-7}
0.2500	2.23004	2.23004	0.4241×10^{-7}
0.4132	3.37795	3.37795	0.7909×10^{-7}
0.5868	5.06464	5.06464	0.1254×10^{-6}
0.7500	7.28311	7.28311	0.1859×10^{-6}
0.8830	9.72900	9.72900	0.2530×10^{-6}
0.9698	11.72910	11.72910	0.3172×10^{-6}
1.0000	12.51387	12.51387	0.3315×10^{-6}

CGL grid nokta dağılımı ile $N = 10$ nokta için elde edilen kesin ve yaklaşık değerler Çizelge 4.14’ de verilmiştir.

Çizelge 4.14 CGL grid nokta dağılımı ile $N = 10$ nokta için elde edilen kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	1.00000	1.00000	0.000000000
0.0245	1.09972	1.09972	0.7814×10^{-9}
0.0955	1.41148	1.41148	0.3067×10^{-9}
0.2061	1.97506	1.97506	0.7942×10^{-9}
0.3455	2.85935	2.85935	0.2115×10^{-8}
0.5000	4.14962	4.14962	0.3811×10^{-9}
0.6545	5.89671	5.89671	0.2633×10^{-8}
0.7939	8.01722	8.01722	0.3505×10^{-8}
0.9045	10.19111	10.19111	0.1099×10^{-7}
0.9755	11.87397	11.87397	0.5726×10^{-8}
1.0000	12.51387	12.51387	0.5542×10^{-8}

4.4 Yüksek Mertebeden Değişken Katsayılı Diferensiyel Denklemlerin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

Yüksek mertebeden değişken katsayılı diferensiyel denklem $k = 0, 1, \dots, n$ için $a_k(x)$ birer katsayı olmak üzere

$$a_n(x)y^{(n)} + a_{n-1}(x)y^{(n-1)} + \dots + a_1(x)y' + a_0(x)y = f(x) \quad (4.21)$$

başlangıç koşulları $A_k \in \mathbb{R}$ olmak üzere

$$y(0) = A_0, y'(0) = A_1, y''(0) = A_2, \dots, y^{(n-1)}(0) = A_{n-1} \quad (4.22)$$

PDKM'yi kullanarak N nokta sayısı için yaklaşık çözümleri hesaplayalım.

(4.21) eşitliğindeki n . mertebeden diferensiyel denkleme $i = 1, 2, \dots, N$ için x_i grid noktaları uygulanırsa

$$a_n(x_i)y^{(n)} + a_{n-1}(x_i)y^{(n-1)} + \dots + a_1(x_i)y' + a_0(x_i)y = f(x_i) \quad (4.23)$$

elde edilir. Bu denklemdeki türevler yerine DKM eşitliği kullanılırsa denklem

$$a_n(x_i) \sum_{j=1}^N w_{ij}^{(n)} y(x_j) + a_{n-1}(x_i) \sum_{j=1}^N w_{ij}^{(n-1)} y(x_j) + \dots + a_1(x_i) \sum_{j=1}^N a_{ij} y(x_j) + a_0(x_i) y(x_i) = f(x_i)$$

$$\sum_{j=1}^N \left\{ a_n(x_i) w_{ij}^{(n)} + a_{n-1}(x_i) w_{ij}^{(n-1)} + \dots + a_1(x_i) a_{ij} + a_0(x_i) I \right\} y(x_j) = f(x_i) \quad (4.24)$$

şeklinde olacaktır. Oluşan N bilinmeyenli N denklem matris ve vektör formunda

$$\left[a_n(x_i) w_{ij}^{(n)} + a_{n-1}(x_i) w_{ij}^{(n-1)} + \dots + a_1(x_i) a_{ij} + a_0(x_i) I \right]_{N \times N} [y_j]_{N \times 1} = [f_i]_{N \times 1} \quad (4.25)$$

şeklinde yazılır. Başlangıç koşullarına DKM eşitliği uygulanırsa

$$y(0) = y_1 = A_0$$

$$y'(0) = y'(x_1) = \sum_{j=1}^N a_{1j} y(x_j) = A_1$$

$$y''(0) = y''(x_1) = \sum_{j=1}^N w_{1j}^{(2)} y(x_j) = A_2$$

⋮

$$y^{(n-1)}(0) = y^{(n-1)}(x_1) = \sum_{j=1}^N w_{1j}^{(n-1)} y(x_j) = A_{n-1}$$

olacaktır. Bu başlangıç koşullarını denklem sistemine katmak üzere sistemdeki N satırdan n tane satır silinip yerine başlangıç koşullarından elde edilen eşitlikler yazılmalıdır. Böylece yeni bir denklem sistemi oluşacaktır. Bu denklem sistemi çözülerek $i = 1, 2, \dots, N$ için y_i yaklaşık çözüm değerleri elde edilecektir.

Örnek 4.4

$$y^{(4)} - y = x^3$$

$$y(0) = 2, \quad y'(0) = -1, \quad y''(0) = 0, \quad y'''(0) = 5$$

şeklinde verilen dördüncü mertebeden başlangıç değer problemini PDKM ile $N = 10$ nokta için yaklaşık değerlerini bulalım.

Çözüm

Yukarıdaki diferensiyel denkleme $i = 1, 2, \dots, N$ için x_i grid noktalarını uygulayalım:

$$y^{(4)}(x_i) - y(x_i) = x_i^3.$$

Bu denklemin dördüncü mertebeden türevi yerine DKM eşitliğini yazarsak

$$\sum_{j=1}^N w_{ij}^{(4)} y(x_j) - y(x_i) = x_i^3$$

elde edilir. Bu eşitliği $i, j = 1, 2, \dots, N$ için matris ve vektör formunda yazalım:

$$\left[w_{ij}^{(4)} - I \right]_{N \times N} \left[y(x_j) \right]_{N \times 1} = \left[x_i^3 \right]_{N \times 1}.$$

Buradaki ağırlıklı katsayılar matrisini kısaca $[b_{ij}] = [w_{ij}^{(4)} - I]$ olarak ifade edersek

$$[b_{ij}][y(x_j)] = [x_i^3]$$

elde edilir. Verilen başlangıç koşullarında da DKM eşitliğini kullanırsak

$$y(0) = y(x_1) = 2$$

$$y'(0) = y'(x_1) = \sum_{j=1}^N a_{1j} y(x_j) = -1$$

$$y''(0) = y''(x_1) = \sum_{j=1}^N w_{1j}^{(2)} y(x_j) = 0$$

$$y'''(0) = y'''(x_1) = \sum_{j=1}^N w_{1j}^{(3)} y(x_j) = 5$$

olarak yazılabilir. Buna göre sisteme bu başlangıç koşullarının girilmesi gerekmektedir. Bu koşulları ekleyebilmek için sistemden dört eşitlik atılmalı, bunların yerine başlangıç koşullarına ait eşitlikler yazılmalıdır. Koşulların yazılmasından sonra oluşacak sistem

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & \cdots & 0 \\ a_{11} & a_{12} & a_{13} & a_{14} & \cdots & \cdots & a_{1N} \\ b_{31} & b_{32} & b_{33} & b_{34} & \cdots & \cdots & b_{3N} \\ b_{41} & b_{42} & b_{43} & b_{44} & & \cdots & b_{4N} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \cdots & \vdots \\ b_{(N-2)1} & b_{(N-2)2} & b_{(N-2)3} & b_{(N-2)4} & \ddots & & b_{(N-2)N} \\ w_{11}^{(2)} & w_{12}^{(2)} & w_{13}^{(2)} & w_{14}^{(2)} & \cdots & \ddots & w_{1N}^{(2)} \\ w_{11}^{(3)} & w_{12}^{(3)} & w_{13}^{(3)} & w_{14}^{(3)} & \cdots & & w_{1N}^{(3)} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \\ x_3^3 \\ x_4^3 \\ \vdots \\ x_{N-2}^3 \\ 0 \\ 5 \end{bmatrix}$$

şeklinde. Oluşan bu lineer denklem sisteminin çözülmesiyle $i = 1, 2, \dots, N$ için y_i yaklaşık değerleri elde edilecektir.

Denklemin analitik çözümü

$$y(x) = e^{-x} + \cos(x) + x^3$$

şeklinde. Buna göre PDKM'den elde edilecek yaklaşık çözüm değerlerini kesin çözüm değerleri ile karşılaştıralım.

Düzgün nokta dağılımı ile $N = 10$ nokta için kesin ve yaklaşık değerlerin karşılaştırılması Çizelge 4.15’ da verilmiştir.

Çizelge 4.15 Düzgün dağılım ile $N = 10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	2.00000	2.00000	0.000000000
0.1111	1.89004	1.89004	0.1121×10^{-7}
0.2222	1.78712	1.78712	0.6322×10^{-7}
0.3333	1.69853	1.69853	0.5822×10^{-6}
0.4444	1.63182	1.63183	0.6844×10^{-5}
0.5556	1.59483	1.59486	0.3618×10^{-4}
0.6667	1.59560	1.59574	0.1345×10^{-3}
0.7778	1.64241	1.64281	0.4021×10^{-3}
0.8889	1.74372	1.74475	0.1032×10^{-2}
1.0000	1.90818	1.91055	0.2364×10^{-2}

Düzgün olmayan grid nokta dağılımı ile $N = 10$ nokta için kesin ve yaklaşık değerlerin karşılaştırılması Çizelge 4.16’ de verilmiştir.

Çizelge 4.16 Düzgün olmayan dağılım ile $N = 10$ nokta için kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	2.00000	2.00000	0.000000000
0.0302	1.96987	1.96987	0.3939×10^{-9}
0.1170	1.88437	1.88437	0.5392×10^{-9}
0.2500	1.76334	1.76334	0.1561×10^{-6}
0.4132	1.64793	1.64794	0.4951×10^{-5}
0.5868	1.59087	1.59093	0.5723×10^{-4}
0.7500	1.62593	1.62625	0.3182×10^{-3}
0.8830	1.73687	1.73786	0.9974×10^{-3}
0.9698	1.85681	1.85873	0.1923×10^{-2}
1.0000	1.90818	1.91056	0.2382×10^{-2}

CGL grid nokta dağılımı ile $N = 10$ nokta için elde edilen kesin ve yaklaşık değerler Çizelge 4.17’ de verilmiştir.

Çizelge 4.17 CGL grid nokta dağılımı ile $N = 10$ nokta için elde edilen kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	y_i Kesin Çözüm	y_i Yaklaşık Çözüm	$ E $ Mutlak Hata
0.0000	2.00000	2.00000	0.000000000
0.0245	1.97554	1.97554	0.2226×10^{-10}
0.0955	1.90524	1.90524	0.5704×10^{-9}
0.2061	1.80134	1.80134	0.3234×10^{-7}
0.3455	1.69002	1.69002	0.1371×10^{-5}
0.5000	1.60911	1.60913	0.1851×10^{-4}
0.6545	1.59342	1.59355	0.1223×10^{-3}
0.7939	1.65352	1.65399	0.4729×10^{-3}
0.9045	1.76282	1.76400	0.1179×10^{-2}
0.9755	1.86609	1.86809	0.2001×10^{-2}
1.0000	1.90818	1.91056	0.2381×10^{-2}

**POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME
METODU VE SONLU FARK METODU ARASINDAKİ İLİŞKİ****5.1 Sonlu Fark Metodunun Oluşumu**

Birçok problemin analitik çözümünün zor bulunması veya analitik çözümünün olmadığı durumlarda nümerik metotlar tercih edilmektedir. Bu nümerik metotlardan biri olan Sonlu Fark Metodu iki şekilde elde edilmektedir. Bunlardan biri Taylor seri açılımına diğeri ise polinomal yaklaşıma dayanmaktadır.

Sonlu Fark Metodu, PDKM gibi nümerik bir metottur. Bu metotta, adi diferensiyel denklemdaki türevlere yaklaşmak üzere yerel ve komşu noktalarındaki fonksiyon değerleri kullanılmaktadır.

Tek değişkenli bir $f(x)$ fonksiyonunun $x_i + s$ noktasındaki değerine, Taylor seri açılımı kullanılarak $f(x)$ fonksiyonunun ve türevlerinin x_i noktasındaki değerleri ile yakınsanabilir.

$$f(x_i + s) = f(x_i) + f'(x_i)s + f''(x_i)\frac{s^2}{2!} + \dots + f^{(k)}(x_i)\frac{s^k}{k!} + \dots \quad (5.1)$$

Sonlu Fark Metodu'nda grid noktalarındaki fonksiyon değerleri bilinen değerler olarak kabul edilmektedir. Bu fonksiyon değerleri, bilinmeyen türevlere yaklaşımda kullanılmaktadır. Taylor seri açılımındaki $f'(x_i)$, $f''(x_i)$, $\dots$, $f^{(k)}(x_i)$ türevleri bilinmeyen değerlerdir. Burada dikkat edilmesi gereken, Taylor seri açılımının hangi terimde kesileceğini belirlemektir. Önce kesme terimine karar verilmeli ve sonrasında bilinmeyenlerin (türevlerin) sayısı belirlenmelidir. Örneğin, Taylor seri açılımı

$f^{(k)}(x_i) \frac{s^k}{k!}$ teriminden sonra kesilecekse bu durumda K tane bilinmeyen ($f'(x_i)$, $f''(x_i)$, ..., $f^{(K)}(x_i)$) olacaktır. Bu bilinmeyenleri bulmak için aynı sayıda K tane eşitliğe ihtiyaç vardır. K tane bilinmeyen olduğundan, Taylor seri açılımı K tane komşu noktaya uygulanmalıdır ve böylece K tane denklem elde edilmelidir. Çünkü Taylor seri açılımının bir grid noktada uygulaması bir denklem oluşturmaktadır. Buradaki K değeri, verilen türevin mertebesine ve gerekli doğruluk mertebesine bağlıdır.

n . mertebeden türevlenebilen bir $f(x)$ fonksiyonunun x_i grid noktasında p . mertebeden doğruluktaki sonlu fark eşitliği

$$f^{(n)}(x_i) = \text{fark ifadesi} + O(s^p) \quad (5.2)$$

şeklindedir. Buradaki grid noktaları genellikle düzensiz olarak kabul edilmektedir. Bu problemi çözerken Taylor seri açılımında $f^{(n)}(x_i)$ türevini içeren $f^{(n)}(x_i) \frac{s^n}{n!}$ terimi dikkate alınmalıdır. Buna göre $f^{(n)}(x_i)$ ifadesini elde etmek için Taylor seri açılımındaki her bir terim s^n değerine bölünmelidir.

$$f(x_i + s) = f(x_i) + f'(x_i)s + f''(x_i)\frac{s^2}{2!} + \dots + f^{(n)}(x_i)\frac{s^n}{n!}$$

$$\frac{f(x_i + s)}{s^n} = \frac{f(x_i)}{s^n} + \frac{f'(x_i)}{s^{n-1}} + \frac{1}{2!} \frac{f''(x_i)}{s^{n-2}} + \dots + \frac{f^{(n)}(x_i)}{n!}$$

Bu ifadeyi elde etmek için Taylor seri açılımındaki kesme hata teriminin mertebesi s^{n+p} olmalıdır. Buna göre s^{n+p} mertebeli terimin s^n ile bölünmesinden s^p mertebeli terim kalacaktır ki bu da doğruluk için gereken mertebedir. Dolayısıyla Taylor seri

açılımı $f^{(n+p-1)}(x_i) \frac{s^{n+p-1}}{(n+p-1)!}$ terimi dahil olmak üzere açılmalıdır. Bunun

sonucunda, Taylor seri açılımındaki bilinmeyenlerin sayısı $(n+p-1)$ olacaktır. Böylece Taylor seri açılımı $(n+p-1)$ komşu noktaya uygulanmalı ve $(n+p-1)$ eşitlik elde edilmelidir.

5.1.1 Sonlu Fark Formülleri

Tek değişkenli fonksiyonların türevlerinin sonlu fark formülleri Taylor seri açılımına dayanılarak bulunacaktır.

5.1.1.1 Birinci Mertebeden Türev için Yaklaşımlar

Bir $f(x)$ fonksiyonunun $(x+s)$ noktasındaki değeri (5.1) eşitliğindeki gibi Taylor seri açılımı ile yazılsın.

$$f(x_i + s) = f(x_i) + f'(x_i)s + f''(x_i)\frac{s^2}{2!} + \dots + f^{(k)}(x_i)\frac{s^k}{k!} + \dots \quad (5.3)$$

(5.3) eşitliği birinci mertebeden türevli terimden sonra kesilir ve bu eşitlikten birinci mertebeden türev çekilirse

$$\frac{df}{dx} = \frac{f(x+s) - f(x)}{s} + O(s) \quad (5.4)$$

yazılabilir. Bu ifade f fonksiyonunun birinci mertebeden türevi için yapılmış birinci dereceden bir yaklaşımdır ve birinci mertebeden ileri fark formülü olarak adlandırılır.

Eğer Taylor açılımı

$$f(x-s) = f(x) - s\frac{df}{dx} + \frac{s^2}{2!}\frac{d^2f}{dx^2} - \frac{s^3}{3!}\frac{d^3f}{dx^3} + \dots \quad (5.5)$$

şeklinde yazılırsa yukarıda yapılan işlemlere benzer şekilde

$$\frac{df}{dx} = \frac{f(x) - f(x-s)}{s} + O(s) \quad (5.6)$$

birinci mertebeden geri fark formülü elde edilir. (5.3) ve (5.5) eşitliklerindeki Taylor açılımlarının farkı alınır

$$\frac{df}{dx} = \frac{f(x+s) - f(x-s)}{2s} + O[s^2] \quad (5.7)$$

birinci mertebeden merkezi fark formülü elde edilir.

5.1.1.2 İkinci Mertebeden Türev için Yaklaşımlar

Taylor serisinin $(x+2s)$ ve $(x-2s)$ noktalarındaki açılımları kullanılarak birinci mertebedekine benzer şekilde sonlu fark yaklaşımları elde edilecektir. Bu formüller aşağıda tek tek verilmiştir.

İkinci mertebeden ileri fark formülü

$$\frac{d^2 f}{dx^2} = \frac{f(x+2s) - 2f(x+s) + f(x)}{s^2} + O(s) \quad (5.8)$$

İkinci mertebeden geri fark formülü

$$\frac{d^2 f}{dx^2} = \frac{f(x) - 2f(x-s) + f(x-2s)}{s^2} + O(s) \quad (5.9)$$

İkinci mertebeden merkezi fark formülü

$$\frac{d^2 f}{dx^2} = \frac{f(x+s) - 2f(x) + f(x-s)}{s^2} + O[s^2] \quad (5.10)$$

şeklindedir.

5.2 Polinom Tabanlı Diferensiyel Kareselleştirme ve Yüksek Mertebeden Sonlu Fark Metodu Arasındaki İlişki

Tanım kümesinde N tane grid nokta olduğunu kabul edelim. PDKM'yi kullanarak $f(x)$ fonksiyonunun birinci mertebeden türevinin bir x_i grid noktasındaki değerine, fonksiyonun N grid noktadaki fonksiyon değerlerinin ağırlık katsayılarıyla çarpımlarının lineer toplamı olarak yazılabiliriz.

$$f'(x_i) = \sum_{j=1}^N a_{ij} f(x_j), \quad i = 1, 2, \dots, N \quad (5.11)$$

Sonlu fark şeması, Taylor seri açılımına dayandığından, (5.11) eşitliğindeki ağırlık katsayılarını Taylor seri açılımı ile hesaplamaya çalışalım. (5.11) eşitliğindeki her bir $j = 1, 2, \dots, N$ için $f(x_j)$ değerine Taylor seri açılımı ile yakınsayabiliriz. Buna göre

$$\begin{aligned}
f(x_j) = & f(x_i) + f'(x_i)(x_j - x_i) + \dots + f^{(k)}(x_i) \frac{(x_j - x_i)^k}{k!} + \\
& + f^{(N-1)}(x_i) \frac{(x_j - x_i)^{N-1}}{(N-1)!} + R_N
\end{aligned} \tag{5.12}$$

yazılabilir. (5.12) seri açılımındaki R_N kesme hatası $\xi \in [x_i, x_j]$ olmak üzere

$$R_N = f^{(N)}(\xi) \frac{(x_j - x_i)^N}{N!} \tag{5.13}$$

şeklindedir. (5.12) eşitliğindeki seri, (5.11) eşitliğinde yerine yazılırsa

$$\begin{aligned}
f'(x_i) = & \sum_{j=1}^N a_{ij} \left[f(x_i) + f'(x_i)(x_j - x_i) + \dots + f^{(k)}(x_i) \frac{(x_j - x_i)^k}{k!} + f^{(N-1)}(x_i) \frac{(x_j - x_i)^{N-1}}{(N-1)!} \right] \\
f'(x_i) = & \sum_{j=1}^N [a_{ij} f(x_i)] + \sum_{j=1}^N [a_{ij} f'(x_i)(x_j - x_i)] + \dots + \sum_{j=1}^N \left[a_{ij} f^{(k)}(x_i) \frac{(x_j - x_i)^k}{k!} \right] + \\
& + \sum_{j=1}^N \left[a_{ij} f^{(N-1)}(x_i) \frac{(x_j - x_i)^{N-1}}{(N-1)!} \right] \\
f'(x_i) = & f(x_i) \sum_{j=1}^N a_{ij} + f'(x_i) \sum_{j=1}^N a_{ij} (x_j - x_i) + \dots + f^{(k)}(x_i) \sum_{j=1}^N a_{ij} \frac{(x_j - x_i)^k}{k!} + \\
& + f^{(N-1)}(x_i) \sum_{j=1}^N a_{ij} \frac{(x_j - x_i)^{N-1}}{(N-1)!}
\end{aligned} \tag{5.14}$$

elde edilir. Buna göre (5.14) eşitliğinin sağ tarafı sol tarafına terim terime eşitlendiğinde $i = 1, 2, \dots, N$ için

$$\sum_{j=1}^N a_{ij} = 0 \tag{5.15}$$

$$\sum_{j=1}^N a_{ij} (x_j - x_i) = 1 \tag{5.16}$$

$$\sum_{j=1}^N a_{ij} (x_j - x_i)^k = 0, \quad k = 2, 3, \dots, N-1 \quad (5.17)$$

cebirsel denklemleri elde edilir. Bu (5.15) - (5.17) eşitliklerinden oluşan denklem sistemi, birinci mertebeden türev için yüksek mertebeli sonlu fark şemasının ağırlık katsayılarını hesaplamakta kullanılır. Benzer şekilde, $f(x)$ fonksiyonunun m . mertebeden türevinin x_i noktasındaki değeri PDKM ile $i = 1, 2, \dots, N$ ve $m = 2, 3, \dots, N-1$ için

$$f^{(m)}(x_i) = \sum_{j=1}^N w_{ij}^{(m)} f(x_j) \quad (5.18)$$

şeklinde yazılır ve böylece $f(x)$ fonksiyonunun m . mertebeden türevi için yüksek mertebeli sonlu fark şemasının ağırlık katsayıları hesaplanabilir. (5.12) eşitliğindeki Taylor serisi m . mertebeden türeve kadar açılıp (5.18) eşitliğinde yerine yazılırsa

$$\begin{aligned} f^{(m)}(x_i) &= \sum_{j=1}^N w_{ij}^{(m)} \left[f(x_i) + f'(x_i)(x_j - x_i) + \dots + f^{(k)}(x_i) \frac{(x_j - x_i)^k}{k!} + f^{(m)}(x_i) \frac{(x_j - x_i)^m}{m!} \right] \\ f^{(m)}(x_i) &= \sum_{j=1}^N \left[w_{ij}^{(m)} f(x_i) \right] + \dots + \sum_{j=1}^N \left[w_{ij}^{(m)} f^{(k)}(x_i) \frac{(x_j - x_i)^k}{k!} \right] + \\ &\quad + \sum_{j=1}^N \left[w_{ij}^{(m)} f^{(m)}(x_i) \frac{(x_j - x_i)^m}{m!} \right] \\ f^{(m)}(x_i) &= f(x_i) \sum_{j=1}^N w_{ij}^{(m)} + \dots + f^{(k)}(x_i) \sum_{j=1}^N w_{ij}^{(m)} \frac{(x_j - x_i)^k}{k!} + \\ &\quad + f^{(m)}(x_i) \sum_{j=1}^N w_{ij}^{(m)} \frac{(x_j - x_i)^m}{m!} \end{aligned} \quad (5.19)$$

elde edilir. Bu (5.19) eşitliğinin sağ tarafı sol tarafa terim terime eşitlenirse

$$\sum_{j=1}^N w_{ij}^{(m)} = 0 \quad (5.20)$$

$$\sum_{j=1}^N w_{ij}^{(m)} (x_j - x_i)^m = m! \quad (5.21)$$

$$\sum_{j=1}^N w_{ij}^{(m)} (x_j - x_i)^k = 0, \quad k = 1, 2, \dots, N-1, k \neq m \quad (5.22)$$

cebirsel denklemleri elde edilir. Bu (5.20) - (5.22) eşitliklerinden oluşan denklem sistemi, yüksek mertebeli sonlu fark şemasındaki ikinci ve daha yüksek mertebeden türevlerin ağırlık katsayılarının hesaplanmasında kullanılmaktadır. Buna göre, yüksek mertebeli sonlu fark şemasındaki ağırlık katsayılarının PDKM ile hesaplanabileceği gösterilebilir. Diğer bir ifadeyle, PDKM'nin yüksek mertebeli sonlu fark şemasıyla eş değer olduğu söylenebilir.

Taban vektörleri kümelerinin birinden elde edilen ağırlık katsayıları, bir diğer taban vektörleri kümesinden elde edilen ağırlık katsayılarına denktir. Dolayısıyla ağırlık katsayılarını belirleyen denklem sistemleri de denk olacaktır. Buna göre Bölüm 3' te Bellman'ın birinci yaklaşımında $x^k, (k = 0, 1, \dots, N-1)$ polinomundan elde edilen eşitlikleri tekrar yazalım. $i = 1, 2, \dots, N$ için

$$\sum_{j=1}^N a_{ij} = 0 \quad (5.23)$$

$$\sum_{j=1}^N a_{ij} x_j = 1 \quad (5.24)$$

$$\sum_{j=1}^N a_{ij} x_j^k = k x_i^{k-1}, \quad k = 2, 3, \dots, N-1 \quad (5.25)$$

şeklinde. Buna göre PDKM ile (5.23) - (5.25) eşitliklerinden elde edilen denklem sisteminin, yüksek mertebeden sonlu fark şemasıyla verilen (5.15) - (5.17) eşitliklerinden elde edilen denklem sistemine denk olduğunu gösterelim.

Görüldüğü üzere her iki denklem sisteminin (5.15) ve (5.23) eşitlikleri birbirine eştir.

$$\sum_{j=1}^N a_{ij} = 0$$

Her iki denklem sisteminin (5.16) ve (5.24) eşitliklerinin birbirine eş olduğunu gösterelim.

$$\begin{aligned}\sum_{j=1}^N a_{ij} (x_j - x_i) - 1 &= \sum_{j=1}^N (a_{ij} x_j - a_{ij} x_i) - 1 \\ &= \sum_{j=1}^N a_{ij} x_j - x_i \sum_{j=1}^N a_{ij} - 1\end{aligned}$$

(5.23) eşitliğinden yararlanarak

$$\sum_{j=1}^N a_{ij} (x_j - x_i) - 1 = \sum_{j=1}^N a_{ij} x_j - 1 \quad (5.26)$$

elde edilir. Geriye iki denklem sisteminin son eşitliklerinin denk olduğunu göstermek kalmaktadır. Bu iki eşitliğin denk olduğunu tümevarımdan yararlanarak göstereceğiz. Buna göre (5.17) ve (5.25) eşitliklerinin ilk $p+1$ denklemin eş olduğunu kabul edelim. $k = 0, 1, \dots, p$ ve $i = 1, 2, \dots, N$ için

$$\sum_{j=1}^N a_{ij} (x_j - x_i)^k = \sum_{j=1}^N a_{ij} x_j^k - k x_i^{k-1} = 0 \quad (5.27)$$

ve $c_p^k = \binom{p}{k} = \frac{p!}{k!(p-k)!}$ olmak üzere

$$(a-b)^p = a^p - c_p^1 a^{p-1} b + \dots + (-1)^k c_p^k a^{p-k} b^k + \dots + (-1)^p b^p \quad (5.28)$$

binom açılımını ele alalım. Bu açılımda $a = b = 1$ alınırsa

$$\begin{aligned}-c_p^1 + \dots + (-1)^k c_p^k + \dots + (-1)^p &= -1 \\ c_p^1 - c_p^2 + \dots + (-1)^{k+1} c_p^k + \dots + (-1)^{p+1} &= 1\end{aligned} \quad (5.29)$$

elde edilir. (5.28) eşitliği kullanılarak $(p+2)$. denklem

$$(x_j - x_i)^{p+1} = x_j^{p+1} - c_{p+1}^1 x_j^p x_i + \dots + (-1)^k c_{p+1}^k x_j^{p+1-k} x_i^k + \dots + (-1)^{p+1} x_i^{p+1}$$

olmak üzere

$$\begin{aligned}\sum_{j=1}^N a_{ij} (x_j - x_i)^{p+1} &= \sum_{j=1}^N a_{ij} \left\{ x_j^{p+1} - c_{p+1}^1 x_j^p x_i + \dots + (-1)^k c_{p+1}^k x_j^{p+1-k} x_i^k + \dots + (-1)^{p+1} x_i^{p+1} \right\} \\ &= \sum_{j=1}^N \left(a_{ij} x_j^{p+1} \right) - \left(a_{ij} c_{p+1}^1 x_j^p x_i \right) + \left(a_{ij} c_{p+1}^2 x_j^{p-1} x_i^2 \right) + \dots + \left(a_{ij} (-1)^{p+1} x_i^{p+1} \right)\end{aligned}$$

$$\begin{aligned}\sum_{j=1}^N a_{ij} (x_j - x_i)^{p+1} &= \sum_{j=1}^N a_{ij} x_j^{p+1} - \\ &- c_{p+1}^1 x_i \left[\sum_{j=1}^N a_{ij} \left(x_j^p - \frac{c_{p+1}^2}{c_{p+1}^1} x_j^{p-1} x_i + \dots + (-1)^p \frac{1}{c_{p+1}^1} x_i^p \right) \right]\end{aligned}\quad (5.30)$$

elde edilir. Bu eşitlikte (5.27) ve (5.29) eşitlikleri kullanılırsa

$$\sum_{j=1}^N a_{ij} (x_j - x_i)^{p+1} = \sum_{j=1}^N a_{ij} x_j^{p+1} - (p+1) x_i^p \quad (5.31)$$

elde edilir. Bu eşitlik, her iki sistemdeki $(p+2)$. denklemlerin denk olduğunu göstermektedir. $p \leq N-2$ şeklinde bir keyfi tamsayı olduğu sürece verilen iki denklem sisteminin birbirine denk olduğu söylenebilir.

PDKM ile Bölüm 3'te bulunan yüksek mertebeden türevlerin ağırlık katsayı eşitliklerini tekrar yazalım. $i, j = 1, 2, \dots, N$ ve $m = 2, 3, \dots, N-1$ olmak üzere

$$w_{ij}^{(m)} = m \left(a_{ij} w_{ii}^{(m-1)} - \frac{w_{ij}^{(m-1)}}{x_i - x_j} \right) \quad (5.32)$$

$$\sum_{j=1}^N w_{ij}^{(m)} = 0 \quad (5.33)$$

eşitliklerinin (5.20), (5.21) ve (5.22) eşitlikleriyle denk olduğunu gösterelim. Görüldüğü üzere (5.20) ve (5.33) eşitlikleri birbirleriyle aynıdır.

$$\sum_{j=1}^N w_{ij}^{(m)} = 0$$

Buna göre $2 \leq m \leq N-1$ için $w_{ij}^{(m)}$ ağırlık katsayılarının (5.21) ve (5.22) eşitliklerini sağladığını gösterelim. (5.21) ve (5.22) eşitliklerini tek bir eşitlikle tanımlamak gerekirse

$$\sum_{j=1}^N w_{ij}^{(m-1)} (x_j - x_i)^k = \begin{cases} (m-1)!, & k = m-1 \\ 0, & k \neq m-1 \end{cases} \quad (5.34)$$

yazılabilir. Bu eşitlikte (5.32) eşitliği kullanılırsa $1 \leq k \leq N-1$ için

$$\begin{aligned} \sum_{j=1}^N w_{ij}^{(m)} (x_j - x_i)^k &= \sum_{j=1}^N \left[m a_{ij} w_{ii}^{(m-1)} - m \frac{w_{ij}^{(m-1)}}{x_i - x_j} \right] (x_j - x_i)^k \\ \sum_{j=1}^N w_{ij}^{(m)} (x_j - x_i)^k &= m w_{ii}^{(m-1)} \sum_{j=1}^N a_{ij} (x_j - x_i)^k + m \sum_{j=1}^N w_{ij}^{(m-1)} (x_j - x_i)^{k-1} \end{aligned}$$

elde edilir. Burada (5.27) eşitlikleri kullanılırsa

$$\sum_{j=1}^N w_{ij}^{(m)} (x_j - x_i)^k = m \sum_{j=1}^N w_{ij}^{(m-1)} (x_j - x_i)^{k-1} \quad (5.35)$$

haline dönüşecektir. Bu eşitlikte (5.34) eşitliği kullanılırsa

$$\sum_{j=1}^N w_{ij}^{(m)} (x_j - x_i)^k = \begin{cases} m!, & k = m \\ 0, & \text{diğer} \end{cases} \quad (5.36)$$

bulunur.

$2 \leq m \leq N-1$ aralığında keyfi bir tamsayı olduğu sürece PDKM'den bulunan $w_{ij}^{(m)}$ ağırlık katsayıları, yüksek mertebeden sonlu fark şemasından elde edilen denklem sistemini sağlayacaktır. Buna göre PDKM'nin Sonlu Fark Metodu'nun bir başka modeli olduğu söylenebilir. Bu metotlar hem iç noktalara hem de sınır noktalara uygulanabilmektedir. Ayrıca grid noktalarının düzenli veya düzensiz olması, metodun uygulanmasına engel olmamaktadır.

KİSMİ TÜREVLİ DİFERENSİYEL DENKLEMLERİN POLİNOM TABANLI DİFERENSİYEL KARESELLEŞTİRME METODU İLE ÇÖZÜMÜ

Bu bölümde kısmi türevli diferensiyel denklemleri PDKM'yi kullanarak çözmeye çalışacağız. Dirichlet ve Neumann sınır koşulları altındaki lineer ısı ve dalga denklemleri incelenecek ve bu denklemlerin zamana bağlı olan türevlerinde Sonlu Fark Metodu kullanılacaktır.

6.1 Dirichlet Sınır Koşulları Altındaki Homojen Parabolik (Isı) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

Örnek 6.1

$$U_t(x,t) = U_{xx}(x,t) , \quad 0 < x < 1 , \quad t > 0$$

Sınır koşulları

$$U(0,t) = U(1,t) = 0 , \quad t \geq 0$$

Başlangıç koşulu

$$U(x,0) = \sin(\pi x)$$

Dirichlet sınır koşulları altındaki homojen ısı denklemini PDKM ile çözelim [23].

Çözüm

$i = 1, 2, \dots, N$ için x_i grid noktalarını yukarıdaki ısı denkleminde uygulayalım:

$$U_t(x_i, t) = U_{xx}(x_i, t).$$

Fonksiyonun t 'ye göre olan türevinde (5.4) birinci mertebeden ileri sonlu fark formülünü, x 'e göre olan türevinde ise ikinci mertebeden DKM eşitliğini kullanalım:

$$\frac{U(x_i, t + \Delta t) - U(x_i, t)}{\Delta t} = \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t)$$

$$U(x_i, t + \Delta t) = U(x_i, t) + \Delta t \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t).$$

Bu eşitliği matris ve vektör gösterimi ile yazarsak

$$\left[U(x_i, t + \Delta t) \right]_{N \times 1} = \left[I + \Delta t w_{ij}^{(2)} \right]_{N \times N} \left[U(x_j, t) \right]_{N \times 1}$$

elde edilir. Buradaki ağırlıklı katsayılar matrisine kısaca $[b_{ij}]$ matrisi dersek

$$\left[U(x_i, t + \Delta t) \right]_{N \times 1} = [b_{ij}]_{N \times N} \left[U(x_j, t) \right]_{N \times 1}$$

olacaktır. Şimdi verilen sınır koşullarını grid noktalar cinsinden ifade edelim:

$$U(0, t) = U(x_1, t) = 0$$

$$U(1, t) = U(x_N, t) = 0.$$

Verilen başlangıç koşulunu da grid noktalar cinsinden yazarsak

$$U(x_i, 0) = \sin(\pi x_i)$$

olacaktır. Grid noktalar cinsinden elde edilen sınır koşulları yukarıdaki sisteme uygulanacak ve oluşan lineer denklem sistemi çözülecektir. Böylece bir sonraki zaman adımına ait sıcaklık değerleri elde edilecektir.

$t = 0$ anındaki sıcaklık değerlerini kullanarak $U(x_i, 0 + \Delta t)$ noktalarındaki değerleri hesaplayalım. Bunun için başlangıç koşulunu kullanarak elde edilecek sistem

$$\begin{bmatrix} U(x_1, 0 + \Delta t) \\ U(x_2, 0 + \Delta t) \\ \vdots \\ U(x_N, 0 + \Delta t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)(N-1)} & b_{(N-1)N} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} U(x_1, 0) \\ U(x_2, 0) \\ \vdots \\ U(x_N, 0) \end{bmatrix}$$

şeklinde olacaktır. Bu sistemde sağ taraftaki ağırlıklı katsayılar matrisi ve sütun vektörü bilinmektedir. Bu bilinenler kullanılarak

$$\begin{bmatrix} U(x_1, 0 + \Delta t) \\ U(x_2, 0 + \Delta t) \\ \vdots \\ U(x_N, 0 + \Delta t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)(N-1)} & b_{(N-1)N} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ \sin(\pi x_2) \\ \vdots \\ \sin(\pi x_{N-1}) \\ 0 \end{bmatrix}$$

elde edilir ve $i = 1, 2, \dots, N$ için $U(x_i, 0 + \Delta t)$ noktalarındaki sıcaklık değerleri hesaplanır.

Bu ısı denkleminin analitik çözümü

$$U(x, t) = e^{-\pi^2 t} \sin(\pi x)$$

olmak üzere $N = 10$ grid nokta için düzgün dağılımı kullanarak kesin ve yaklaşık değerlerin karşılaştırılması Çizelge 6.1' de verilmiştir.

Çizelge 6.1 Örnek 6.1 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	U_{Kesin}	U_{Yaklasik}	$ E $ Mutlak Hata
$\Delta t = 0.01$			
0.1111	0.30988	0.30826	0.1613×10^{-2}
0.2222	0.58238	0.57935	0.3030×10^{-2}
0.3333	0.78463	0.78055	0.4083×10^{-2}
0.4444	0.89225	0.88761	0.4642×10^{-2}
$\Delta t = 0.1$			
0.1111	0.12747	0.00445	0.1230
0.2222	0.23957	0.00838	0.2312
0.3333	0.32277	0.01129	0.3115
0.4444	0.36705	0.01284	0.3542
$\Delta t = 0.5$			
0.1111	0.00246	-1.3458	1.348
0.2222	0.00462	-2.5292	2.534
0.3333	0.00623	-3.4076	3.414
0.4444	0.00708	-3.8750	3.882

6.2 Dirichlet Sınır Koşulları Altındaki Homojen Olmayan Parabolik (Isı) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

Örnek 6.2

$$U_t(x, t) = U_{xx}(x, t) + \cos(x), \quad 0 < x < 1, \quad t > 0$$

Sınır koşulları

$$U(0, t) = 1 - e^{-t}$$

$$U(1, t) = \cos(1)(1 - e^{-t})$$

Başlangıç koşulu

$$U(x, 0) = 0$$

Dirichlet sınır koşulları altındaki homojen olmayan ısı denklemini PDKM ile çözelim [11].

Çözüm

$i = 1, 2, \dots, N$ için x_i grid noktalarını yukarıdaki ısı denklemine uygulayalım:

$$U_t(x_i, t) = U_{xx}(x_i, t) + \cos(x_i).$$

Fonksiyonun t 'ye göre olan türevinde (5.4) birinci mertebeden ileri sonlu fark formülünü, x 'e göre olan türevinde ise ikinci mertebeden DKM eşitliğini kullanalım:

$$\frac{U(x_i, t + \Delta t) - U(x_i, t)}{\Delta t} = \left(\sum_{j=1}^N w_{ij}^{(2)} U(x_j, t) \right) + \cos(x_i)$$

$$U(x_i, t + \Delta t) = U(x_i, t) + \left(\Delta t \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t) \right) + \Delta t \cos(x_i).$$

Bu eşitliği matris ve vektör kullanarak tekrar yazalım:

$$[U(x_i, t + \Delta t)]_{N \times 1} = [I + \Delta t w_{ij}^{(2)}]_{N \times N} [U(x_j, t)]_{N \times 1} + [\Delta t \cos(x_i)]_{N \times 1}.$$

Buradaki ağırlıklı katsayılar matrisine kısaca $[b_{ij}]$ dersek

$$[U(x_i, t + \Delta t)]_{N \times 1} = [b_{ij}]_{N \times N} [U(x_j, t)]_{N \times 1} + [\Delta t \cos(x_i)]_{N \times 1}$$

elde edilir. Şimdi verilen sınır koşullarını grid noktalar cinsinden ifade edelim:

$$U(0, t) = U(x_1, t) = 1 - e^{-t}$$

$$U(1, t) = U(x_N, t) = \cos(1)(1 - e^{-t}).$$

Verilen başlangıç koşulunun değeri sıfır olduğundan grid noktalar cinsinden ifade edilemeyecektir. Grid noktalar cinsinden elde edilen sınır koşulları yukarıdaki sisteme uygulanacak ve oluşan lineer denklem sistemi çözülecektir. Böylece bir sonraki zaman adımına ait sıcaklık değerleri elde edilecektir.

$t = 0$ anındaki sıcaklık değerlerini kullanarak $U(x_i, 0 + \Delta t)$ noktalarındaki değerleri hesaplayalım. Bunun için başlangıç koşulunu kullanarak elde edilecek sistem

$$\begin{bmatrix} U(x_1, 0 + \Delta t) \\ U(x_2, 0 + \Delta t) \\ \vdots \\ U(x_N, 0 + \Delta t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)(N-1)} & b_{(N-1)N} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} U(x_1, 0) \\ U(x_2, 0) \\ \vdots \\ U(x_N, 0) \end{bmatrix} + \begin{bmatrix} \Delta t \cos(x_1) \\ \Delta t \cos(x_2) \\ \vdots \\ \Delta t \cos(x_N) \end{bmatrix}$$

olmak üzere

$$\begin{bmatrix} U(x_1, 0 + \Delta t) \\ U(x_2, 0 + \Delta t) \\ \vdots \\ U(x_N, 0 + \Delta t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)N} \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 - e^0 \\ 0 \\ \vdots \\ 0 \\ \cos(1)(1 - e^0) \end{bmatrix} + \begin{bmatrix} \Delta t \cos(x_1) \\ \Delta t \cos(x_2) \\ \vdots \\ \Delta t \cos(x_N) \end{bmatrix}$$

şeklindedir. Bu lineer denklem sisteminin çözümünden $U(x_i, 0 + \Delta t)$ noktalarındaki sıcaklık değerleri bulunacaktır.

Bu ısı denkleminin analitik çözümü

$$U(x, t) = \cos(x)(1 - e^{-t})$$

olmak üzere $N = 10$ grid nokta için düzgün olmayan dağılım kullanarak kesin ve yaklaşık değerlerin karşılaştırılması Çizelge 6.2' de verilmiştir.

Çizelge 6.2 Örnek 6.2 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	U_{Kesin}	U_{Yaklasik}	$ E $ Mutlak Hata
$\Delta t = 0.01$			
0.0302	0.00995	0.00999	0.4981×10^{-4}
0.1170	0.00988	0.00993	0.4949×10^{-4}
0.8830	0.00632	0.00635	0.3163×10^{-4}
0.9698	0.00563	0.00565	0.2817×10^{-4}
$\Delta t = 0.1$			
0.0302	0.09512	0.09995	0.4835×10^{-2}
0.1170	0.09451	0.09931	0.4804×10^{-2}
0.8830	0.06041	0.06348	0.3071×10^{-2}
0.9698	0.05381	0.05654	0.2735×10^{-2}
$\Delta t = 0.5$			
0.0302	0.39329	0.49977	0.1064822
0.1170	0.39078	0.49658	0.1058026
0.8830	0.24978	0.31741	0.0676277
0.9698	0.22248	0.28271	0.0602352

6.3 Neumann Sınır Koşulları Altındaki Homojen Parabolik (Isı) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

Örnek 6.3

$$U_t(x,t) = U_{xx}(x,t) , \quad 0 < x < 1 , \quad t > 0$$

Sınır koşulları

$$U_x(0,t) = U_x(1,t) = 0 , \quad t \geq 0$$

Başlangıç koşulu

$$U(x,0) = x , \quad 0 \leq x \leq 1$$

Neumann sınır koşulları altındaki homojen ısı denklemini PDKM ile çözelim.

Çözüm

$i = 1, 2, \dots, N$ için x_i grid noktalarını yukarıdaki ısı denkleminde uygulayalım:

$$U_t(x_i, t) = U_{xx}(x_i, t).$$

Fonksiyonun t 'ye göre olan türevinde (5.4) birinci mertebeden ileri sonlu fark formülünü, x 'e göre olan türevinde ise ikinci mertebeden DKM eşitliğini kullanalım:

$$\frac{U(x_i, t + \Delta t) - U(x_i, t)}{\Delta t} = \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t)$$

$$U(x_i, t + \Delta t) = U(x_i, t) + \Delta t \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t).$$

Bu eşitliği matris ve vektör gösterimi ile tekrar yazalım:

$$[U(x_i, t + \Delta t)]_{N \times 1} = [I + \Delta t w_{ij}^{(2)}]_{N \times N} [U(x_j, t)]_{N \times 1}.$$

Buradaki ağırlıklı katsayılar matrisine kısaca $[b_{ij}]$ dersek

$$[U(x_i, t + \Delta t)]_{N \times 1} = [b_{ij}]_{N \times N} [U(x_j, t)]_{N \times 1}$$

elde edilir. Şimdi verilen sınır koşullarını grid noktalar cinsinden ifade edelim:

$$U_x(0, t) = U_x(x_1, t) = \sum_{j=1}^N a_{1j} U(x_j, t) = 0$$

$$U_x(1, t) = U_x(x_N, t) = \sum_{j=1}^N a_{Nj} U(x_j, t) = 0.$$

Verilen başlangıç koşulunun grid noktalar cinsinden yazılmasıyla

$$U(x_i, 0) = x_i$$

olacaktır. Grid noktalar cinsinden elde edilen sınır koşulları yukarıdaki sisteme uygulanacak ve oluşan lineer denklem sistemi çözülecektir. Böylece bir sonraki zaman adımına ait sıcaklık değerleri elde edilecektir.

$t = 0$ anındaki sıcaklık değerlerini kullanarak $U(x_i, 0 + \Delta t)$ noktalarındaki değerleri hesaplayalım. Bunun için başlangıç koşulunu kullanarak elde edilecek sistem

$$\begin{bmatrix} U(x_1, 0 + \Delta t) \\ U(x_2, 0 + \Delta t) \\ \vdots \\ U(x_N, t + \Delta t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1N} \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)(N-1)} & b_{(N-1)N} \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{NN} \end{bmatrix} \begin{bmatrix} U(x_1, 0) \\ U(x_2, 0) \\ \vdots \\ U(x_N, 0) \end{bmatrix}$$

olmak üzere

$$\begin{bmatrix} U(x_1, 0 + \Delta t) \\ U(x_2, 0 + \Delta t) \\ \vdots \\ U(x_N, t + \Delta t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1N} \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)(N-1)} & b_{(N-1)N} \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{NN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{N-1} \\ x_N \end{bmatrix}$$

şeklinde. Bu lineer denklem sisteminin çözümünden $U(x_i, 0 + \Delta t)$ noktalarındaki sıcaklık değerleri bulunacaktır. Ancak burada dikkat edilmesi gereken bir durum söz konusudur. Ağırlıklı katsayılar matrisine yazdığımız Neumann koşullarından elde edilecek değerler, verilen koşullar ile eş olmalıdır. Buna göre elde edilen $U(x_1, 0 + \Delta t)$ ve $U(x_N, 0 + \Delta t)$ değerleri soruda verilen koşula eş olmak üzere düzenlenmelidir. Bu soruda Neumann koşulları ağırlıklı katsayılar matrisinin ilk ve son satırlarına yazıldığından, sol taraftaki ilk ve son satırda değişiklik yapılmıştır.

Yukarıdaki ısı denkleminin analitik çözümü

$$U(x, t) = \frac{1}{2} - 4 \sum_{m=1}^{\infty} \frac{1}{[(2m-1)\pi]^2} e^{-[(2m-1)\pi]^2 t} \cos[(2m-1)\pi x]$$

olmak üzere $N = 10$ grid nokta için düzgün dağılımı kullanarak kesin ve yaklaşık değerlerin karşılaştırılması Çizelge 6.3' de verilmiştir.

Çizelge 6.3 Örnek 6.3 için farklı zaman adımları kullanarak belirli grid noktalarındaki kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	U_{Kesin}	U_{Yaklasik}	$ E $ Mutlak Hata
$\Delta t = 0.01$			
0.1111	0.14598	0.11111	0.3487×10^{-1}
0.2222	0.22925	0.22222	0.7031×10^{-2}
0.3333	0.33421	0.33333	0.8751×10^{-3}
0.4444	0.44451	0.44444	0.6194×10^{-4}
$\Delta t = 0.1$			
0.1111	0.35805	0.11111	0.24694
0.2222	0.38429	0.22222	0.16207
0.3333	0.42448	0.33333	0.09115
0.4444	0.47377	0.44444	0.02933
$\Delta t = 0.5$			
0.1111	0.49726	0.11111	0.38615
0.2222	0.49777	0.22222	0.27554
0.3333	0.49854	0.33333	0.16521
0.4444	0.49949	0.44444	0.05505

6.4 Dirichlet Sınır Koşulları Altındaki Homojen Hiperbolik (Dalga) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

Örnek 6.4

$$U_{tt}(x,t) = U_{xx}(x,t), \quad 0 < x < \pi, t > 0$$

Sınır koşulları

$$U(0,t) = U(\pi,t) = 0$$

Başlangıç koşulları

$$U(x,0) = 3 \sin(x)$$

$$U_t(x,0) = 0$$

Dirichlet sınır koşulları altındaki homojen dalga denklemini PDKM kullanarak hesaplayalım.

Çözüm

$i = 1, 2, \dots, N$ için x_i grid noktalarını yukarıdaki dalga denklemine uygulayalım:

$$U_{tt}(x_i, t) = U_{xx}(x_i, t).$$

Fonksiyonun t 'ye göre olan türevinde (5.8) ikinci mertebeden ileri sonlu fark formülünü, x 'e göre olan türevinde ise ikinci mertebeden DKM eşitliğini kullanalım:

$$\frac{U(x_i, t + 2\Delta t) - 2U(x_i, t + \Delta t) + U(x_i, t)}{(\Delta t)^2} = \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t)$$

$$U(x_i, t + 2\Delta t) - 2U(x_i, t + \Delta t) + U(x_i, t) = (\Delta t)^2 \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t).$$

Bu eşitliği matris ve vektör gösterimi ile tekrar yazalım:

$$\left[U(x_i, t + 2\Delta t) \right]_{N \times 1} = \left[(\Delta t)^2 w_{ij}^{(2)} - I \right]_{N \times N} \left[U(x_j, t) \right]_{N \times 1} + \left[2U(x_i, t + \Delta t) \right]_{N \times 1}.$$

Buradaki ağırlıklı katsayılar matrisine kısaca $[b_{ij}]$ dersek

$$\left[U(x_i, t + 2\Delta t) \right]_{N \times 1} = [b_{ij}]_{N \times N} \left[U(x_j, t) \right]_{N \times 1} + \left[2U(x_i, t + \Delta t) \right]_{N \times 1}$$

elde edilir. Şimdi verilen sınır koşullarını grid noktalar cinsinden ifade edelim:

$$U(0, t) = U(x_1, t) = 0$$

$$U(\pi, t) = U(x_N, t) = 0.$$

Verilen başlangıç koşullarının grid noktalar cinsinden yazılmasıyla

$$U(x_i, 0) = 3 \sin(x_i)$$

$$U_t(x_i, 0) = 0$$

olacaktır. Grid noktalar cinsinden elde edilen sınır koşulları yukarıdaki sisteme uygulanacak ve oluşan lineer denklem sistemi çözülecektir. Böylece bir sonraki zaman adımına ait yaklaşık değerler elde edilecektir.

$t = 0$ anındaki değerlerini kullanarak $U(x_i, 0 + 2\Delta t)$ noktalarındaki değerleri hesaplayalım.

$$[U(x_i, 0 + 2\Delta t)]_{N \times 1} = [b_{ij}]_{N \times N} [U(x_j, 0)]_{N \times 1} + 2[U(x_i, 0 + \Delta t)]_{N \times 1}$$

Bu eşitlikte sol taraftaki bilinmeyeni hesaplayabilmek için sağ taraftaki tüm vektörlerin bilinmesi gerekmektedir. Bu eşitlikteki $U(x_i, 0 + \Delta t)$ değerlerini bulabilmek için başlangıç koşullarından birini kullanmak gerekmektedir. Buna göre

$$U_t(x_i, 0) = \frac{U(x_i, 0 + \Delta t) - U(x_i, 0)}{\Delta t} = 0$$

$$U(x_i, 0 + \Delta t) = U(x_i, 0)$$

olacaktır. Böylece eşitliği tekrar yazarsak

$$[U(x_i, 0 + 2\Delta t)]_{N \times 1} = [b_{ij}]_{N \times N} [U(x_j, 0)]_{N \times 1} + 2[U(x_i, 0)]_{N \times 1}$$

elde edilecektir. Bu eşitlikte ağırlıklı katsayılar matrisine sınır koşullarını ve vektörlerde başlangıç koşulunu yerine yazarsak

$$\begin{bmatrix} U(x_1, 0 + 2\Delta t) \\ U(x_2, 0 + 2\Delta t) \\ \vdots \\ U(x_N, 0 + 2\Delta t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \vdots & & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & b_{(N-1)3} & \cdots & b_{(N-1)N} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} U(x_1, 0) \\ U(x_2, 0) \\ \vdots \\ U(x_N, 0) \end{bmatrix} + 2 \begin{bmatrix} U(x_1, 0) \\ U(x_2, 0) \\ \vdots \\ U(x_N, 0) \end{bmatrix}$$

olmak üzere

$$\begin{bmatrix} U(x_1, 0 + 2\Delta t) \\ U(x_2, 0 + 2\Delta t) \\ \vdots \\ U(x_N, 0 + 2\Delta t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2N} \\ \vdots & \vdots & \vdots & & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & b_{(N-1)3} & \cdots & b_{(N-1)N} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 3 \sin(x_2) \\ \vdots \\ 3 \sin(x_{N-1}) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 6 \sin(x_2) \\ \vdots \\ 6 \sin(x_{N-1}) \\ 0 \end{bmatrix}$$

elde edilir. Bu lineer denklem sisteminin çözümünden $U(x_i, 0 + 2\Delta t)$ noktalarındaki yaklaşık değerler bulunacaktır.

Yukarıdaki dalga denkleminin analitik çözümü

$$U(x, t) = 3 \cos(t) \sin(x)$$

olmak üzere $N = 10$ grid nokta için düzgün dağılımı kullanarak kesin ve yaklaşık değerlerin karşılaştırılması Çizelge 6.4’ de verilmiştir.

Çizelge 6.4 Örnek 6.4 için farklı zaman adımları kullanarak belirli grid noktadaki kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	U_{Kesin}	U_{Yaklasik}	$ E $ Mutlak Hata
$\Delta t = 0.01$			
0.3491	1.02586	1.02596	0.1026×10^{-3}
0.6981	1.92798	1.92817	0.1928×10^{-3}
1.0472	2.59756	2.59782	0.2598×10^{-3}
1.3963	2.95383	2.95413	0.2954×10^{-3}
$\Delta t = 0.1$			
0.3491	1.00561	1.01580	0.01019
0.6981	1.88992	1.90908	0.01916
1.0472	2.54629	2.57210	0.02581
1.3963	2.89553	2.92488	0.02935
$\Delta t = 0.5$			
0.3491	0.55438	0.76954	0.21516
0.6981	1.04190	1.44627	0.40437
1.0472	1.40375	1.94856	0.54481
1.3963	1.59628	2.21582	0.61954

6.5 Başlangıç Koşulları Altındaki Homojen Olmayan Hiperbolik (Dalga) Denklemin Polinom Tabanlı Diferensiyel Kareselleştirme Metodu ile Çözümü

Örnek 6.5

$$U_{tt}(x,t) = U_{xx}(x,t) - 1, t > 0$$

Başlangıç koşulları

$$U(x,0) = \sin(x)$$

$$U_t(x,0) = x$$

ile verilen homojen olmayan dalga denklemini PDKM kullanarak hesaplayalım [24].

Çözüm

$i = 1, 2, \dots, N$ için x_i grid noktalarını yukarıdaki dalga denklemine uygulayalım:

$$U_{tt}(x_i, t) = U_{xx}(x_i, t) - 1.$$

Fonksiyonun t 'ye göre olan türevinde (5.8) ikinci mertebeden ileri sonlu fark formülünü, x 'e göre olan türevinde ise ikinci mertebeden DKM eşitliğini kullanalım:

$$\frac{U(x_i, t + 2\Delta t) - 2U(x_i, t + \Delta t) + U(x_i, t)}{(\Delta t)^2} = -1 + \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t)$$

$$U(x_i, t + 2\Delta t) = 2U(x_i, t + \Delta t) - U(x_i, t) - (\Delta t)^2 + (\Delta t)^2 \sum_{j=1}^N w_{ij}^{(2)} U(x_j, t).$$

Bu eşitliği matris ve vektör gösterimi ile tekrar yazalım:

$$[U(x_i, t + 2\Delta t)] = [(\Delta t)^2 w_{ij}^{(2)} - I][U(x_j, t)] + [2U(x_i, t + \Delta t)] - [(\Delta t)^2].$$

Buradaki ağırlıklı katsayılar matrisine kısaca $[b_{ij}]$ dersek

$$[U(x_i, t + 2\Delta t)]_{N \times 1} = [b_{ij}]_{N \times N} [U(x_j, t)]_{N \times 1} + [2U(x_i, t + \Delta t)]_{N \times 1} - [(\Delta t)^2]_{N \times 1}$$

elde edilir. Verilen başlangıç koşullarını grid noktalar cinsinden yazarsak

$$U(x_i, 0) = \sin(x_i)$$

$$U_t(x_i, 0) = x_i$$

olacaktır.

$t = 0$ anındaki değerlerini kullanarak $U(x_i, 0 + 2\Delta t)$ noktalarındaki değerleri hesaplayalım.

$$[U(x_i, 0 + 2\Delta t)]_{N \times 1} = [b_{ij}]_{N \times N} [U(x_j, 0)]_{N \times 1} + [2U(x_i, 0 + \Delta t)]_{N \times 1} - [(\Delta t)^2]_{N \times 1}$$

Bu eşitlikte sol taraftaki bilinmeyeni hesaplayabilmek için sağ taraftaki tüm vektörlerin bilinmesi gerekmektedir. Bu eşitlikteki $U(x_i, 0 + \Delta t)$ değerlerini bulabilmek için başlangıç koşullarından birini kullanmak gerekmektedir. Buna göre

$$U_t(x_i, 0) = \frac{U(x_i, 0 + \Delta t) - U(x_i, 0)}{\Delta t} = x_i$$

$$U(x_i, 0 + \Delta t) = x_i \Delta t + U(x_i, 0)$$

olacaktır. Böylece eşitliği tekrar yazarsak

$$[U(x_i, 0 + 2\Delta t)]_{N \times 1} = [b_{ij}]_{N \times N} [U(x_j, 0)]_{N \times 1} + 2[x_i \Delta t + U(x_i, 0)]_{N \times 1} - [(\Delta t)^2]_{N \times 1}$$

elde edilecektir. Buna göre

$$\begin{bmatrix} U(x_1, 0 + 2\Delta t) \\ U(x_2, 0 + 2\Delta t) \\ \vdots \\ U(x_N, 0 + 2\Delta t) \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1N} \\ b_{21} & b_{22} & \cdots & b_{2N} \\ \vdots & \vdots & & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)N} \\ b_{N1} & b_{N2} & \cdots & b_{NN} \end{bmatrix} \begin{bmatrix} U(x_1, 0) \\ U(x_2, 0) \\ \vdots \\ U(x_N, 0) \end{bmatrix} + 2 \begin{bmatrix} x_1 \Delta t + U(x_1, 0) \\ x_2 \Delta t + U(x_2, 0) \\ \vdots \\ x_N \Delta t + U(x_N, 0) \end{bmatrix} - [(\Delta t)^2]_{N \times 1}$$

olmak üzere

$$\begin{bmatrix} U(x_1, 0 + 2\Delta t) \\ U(x_2, 0 + 2\Delta t) \\ \vdots \\ U(x_N, 0 + 2\Delta t) \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1N} \\ b_{21} & b_{22} & \cdots & b_{2N} \\ \vdots & \vdots & & \vdots \\ b_{(N-1)1} & b_{(N-1)2} & \cdots & b_{(N-1)N} \\ b_{N1} & b_{N2} & \cdots & b_{NN} \end{bmatrix} \begin{bmatrix} \sin(x_1) \\ \sin(x_2) \\ \vdots \\ \sin(x_N) \end{bmatrix} + 2 \begin{bmatrix} x_1 \Delta t + \sin(x_1) \\ x_2 \Delta t + \sin(x_2) \\ \vdots \\ x_N \Delta t + \sin(x_N) \end{bmatrix} - [(\Delta t)^2]_{N \times 1}$$

Bu lineer denklem sisteminin çözümünden $U(x_i, 0 + 2\Delta t)$ noktalarındaki yaklaşık değerler bulunacaktır.

Yukarıdaki dalga denkleminin analitik çözümü

$$U(x, t) = \frac{1}{2} [\sin(x + t) + \sin(x - t)] + xt - \frac{1}{2} t^2$$

olmak üzere $N = 10$ grid nokta için düzgün olmayan dağılımı kullanarak $[0, 1]$ aralığında kesin ve yaklaşık değerlerin karşılaştırılması Çizelge 6.5’ de verilmiştir.

Çizelge 6.5 Örnek 6.5 için farklı zaman adımları kullanarak belirli grid noktadaki kesin ve yaklaşık değerler ve mutlak hatalar

x_i Grid Noktaları	U_{Kesin}	U_{Yaklasik}	$ E $ Mutlak Hata
$\Delta t = 0.01$			
0.0000	-0.0002	-0.0001	0.9999×10^{-4}
0.4132	0.40950	0.40964	0.1402×10^{-3}
0.5868	0.56514	0.56530	0.1554×10^{-3}
1.0000	0.86110	0.86129	0.1841×10^{-3}
$\Delta t = 0.1$			
0.0000	-0.0200	-0.0100	0.9999×10^{-2}
0.4132	0.45615	0.47014	0.1399×10^{-1}
0.5868	0.64005	0.65555	0.1550×10^{-1}
1.0000	1.00470	1.02306	0.1836×10^{-1}
$\Delta t = 0.5$			
0.0000	-0.5000	-0.2500	0.2499
0.4132	0.13012	0.46432	0.3342
0.5868	0.38510	0.75211	0.3661
1.0000	0.95465	1.38110	0.4265

SONUÇ VE ÖNERİLER

Bu çalışmada, nümerik bir metot olan DKM'nin bir çeşidi PDKM, adi ve kısmi türevli diferensiyel denklemlere uygulanmıştır.

Lineer adi diferensiyel denkleme PDKM'nin uygulanmasıyla denklem cebirsel bir lineer denklem sistemine indirgenmiştir. Verilen başlangıç ve/veya sınır koşullarına da PDKM'nin uygulanmasıyla oluşan lineer denklem sistemi, bilinen yöntemlerden biri (Matrisin Ters Metodu, Cramer Metodu, Gauss Eliminasyon Metodu, Gauss-Jordan Metodu, LU Ayrıştırma Metodu ve benzeri) ile çözülebilmektedir.

Yüksek mertebeden sonlu fark şemasının ağırlık katsayıları PDKM ile hesaplanabilmektedir. Bir diğer ifade ile PDKM, yüksek mertebeden sonlu fark şemasıyla eş değerdir. m diferensiyel denklemin mertebesi ve N grid nokta sayısı olmak üzere eğer $2 \leq m \leq N-1$ aralığında ise PDKM ile bulunan $w_{ij}^{(m)}$ ağırlık katsayıları, yüksek mertebeden sonlu fark şemasıyla bulunan denklem sistemini sağlayacaktır. Buna göre PDKM, Sonlu Fark Metodu'nun genişletilmiş bir modelidir. Bu iki metot hem iç noktalara hem de sınır noktalara uygulanabilmektedir. Ayrıca grid noktaların düzenli veya düzensiz oluşu metodun uygulanmasına engel olmamaktadır.

PDKM'nin kısmi türevli diferensiyel denkleme uygulanmasıyla denklem, lineer adi diferensiyel denkleme indirgenmiştir. Verilen başlangıç ve/veya sınır koşullarına da PDKM'nin uygulanmasıyla oluşan lineer adi diferensiyel denklem, bilinen yöntemlerden biri (Runge-Kutta Metodu, Adams Metodu, Adams-Moulton Metodu, Sonlu Fark Metodu ve benzeri) ile çözülebilmektedir.

Sonuç olarak çalışmamızda kullanılan bu metot, benzer şekilde, lineer olmayan adi ve kısmi türevli diferensiyel denklemlere de uygulanabilir.

KAYNAKLAR

-
- [1] Zhang, Y. ve Zong, Z., (2009). Advanced Differential Quadrature Methods, CRC Press, USA.
 - [2] Bellman, R., Kashef, B.G. ve Casti, J., (1972). "Differential Quadrature: A Technique for the Rapid Solution of the Nonlinear Partial Differential Equations", Journal of Computational Physics, 10: 40-52.
 - [3] Bert, C.W. ve Malik, M., (1996). Differential Quadrature Method in Computational Mechanics: A Review, http://www.asme.org/terms/Terms_Use.cfm, 7 Şubat 2013.
 - [4] Shu, C., (2000). Differential Quadrature and Its Application in Engineering, Springer, Great Britain.
 - [5] Quan, J. ve Chang, C., (1989). "New Insights in Solving Distributed System Equations by the Quadrature Method –I. Analysis", Computers Chem. Engng, 13(7) : 779-788.
 - [6] Quan, J. ve Chang, C., (1989). "New Insights in Solving Distributed System Equations by the Quadrature Method –II. Numerical Experiments", Computers Chem. Engng, 13(9): 1017-1024.
 - [7] Shu, C. ve Richards, BE., (1990). "High resolution of natural convection in a square cavity by generalized differential quadrature", Proc of 3rd Conf on Adv in Numer Methods in Eng: Theory and Appl, Swansea, UK, 2: 978-985
 - [8] Yücel, U. ve Sarı, M., (2009). "Differential Quadrature Method (DQM) for a Class of Singular Two-point Boundary Value Problems", International Journal of Computer Mathematics, 86(3): 465-475.
 - [9] Yücel, U., (2006). "Approximations of Sturm-Liouville Eigenvalues Using Differential Quadrature (DQ) Method", Journal of Computational and Applied Mathematics, 192: 310-319.
 - [10] Girgin, Z., (2008). "Combining Differential Quadrature Method with Simulation Technique to Solve Non-linear Differential Equations", International Journal for Numerical Methods in Engineering, 75: 722-734.
 - [11] Gürarlan, G. ve Sarı, M., (2011). "Numerical Solutions of Linear and Nonlinear Diffusion Equations by a Differential Quadrature Method (DQM)", International Journal for Numerical Methods in Biomedical Engineering, 27: 69-77.
 - [12] Meral, G. ve Tezer-Sezgin, M., (2009). "Differential Quadrature Solution of Nonlinear Reaction-Diffusion Equation with Relaxation-type Time Integration", International Journal of Computer Mathematics, 86(3): 451-463.

- [13] Meral, G., (2013). "Differential Quadrature Solution of Heat- and Mass-transfer Equations" , Applied Mathematical Modelling, 37: 4350-4359.
- [14] Tanaka, M. ve Chen, W., (2001). "Coupling Dual Reciprocity BEM and Differential Quadrature Method for Time Dependent Diffusion Problems" , Applied Mathematical Modelling, 25: 257-268.
- [15] Büyük Ögüt, E., (2009). "İki Sabit Isı Kaynağıyla Isıtılmış Eğik Kare Bir Kapalı Bölgede Doğal Konveksiyonla Isı Transferi" , Trakya Univ J Sci, 10(1): 75-83.
- [16] Korkmaz, A. ve Dağ, İ., (2011). "Polynomial Based Differential Quadrature Method for Numerical Solution of Nonlinear Burgers' Equation" , Journal of the Franklin Institute, 348: 2863-2875.
- [17] Ersoy, H., Özpolat, L., Demir, Ç. ve Civalek, Ö., (2009). "Dairesel ve Boşluklu Dairesel Zarların Serbest Titreşim Hesabı" , Mühendis ve Makina, 50 (594) : 25-32.
- [18] Korkmaz, A., (2009), Numerical Algorithms for Solutions of Korteweg-de Vries Equation, wileyonlinelibrary.com , 19 Kasım 2012.
- [19] Shu, C. ve Xue, H., (1999). "Solution of Helmholtz Equation by Differential Quadrature Method" , Computer Methods in Applied Mechanics and Engineering, 175: 203-212.
- [20] Kaya, B. ve Ülke, A., (2012). "Diferansiyel Kuadratur Yöntemi Kullanılarak Difüzyon Dalga Yaklaşımı ile Taşkınların Ötellenmesi" ,Gazi Üniv. Müh. Mim. Fak. Der, 27(2): 313-322.
- [21] Evans, G., (1995). Practical Numerical Analysis, Wiley, Chichester, New York.
- [22] Hacısalıhoğlu, H.H.. Lineer Cebir, 1. Cilt (8.Baskı), Hacısalıhoğlu Yayınları.
- [23] Lam, C.Y., (1994). Applied Numerical Methods for Partial Differential Equations, Prentice Hall, Singapore.
- [24] Debnath, L. ve Myint-U, T., (2007). Linear Partial Differential Equations for Scientists and Engineers, Fourth Edition, Birkhauser, Boston.

ADI DIFERENSIYEL DENKLEMIN COZUMUNE AIT ISLEM BASAMAKLARI

```
> restart;
> with(LinearAlgebra):
> with(linalg):
> print("Ornek 4.1");
```

"Ornek 4.1" (1)

```
> print("Esit Aralikli (Uniform) Dagilim");
> print("Grid Nokta Sayisi"); N:=5;
> print("Sag Taraf Fonksiyonu"); f:=unapply(1+2*x*sin
(x^2),x);
> XX:=Vector(N):A1:=Matrix(N,N):
B:=Matrix(N,N,shape=identity):
```

"Esit Aralikli (Uniform) Dagilim"
"Grid Nokta Sayisi"
 $N := 5$
"Sag Taraf Fonksiyonu"
 $f := x \rightarrow 1 + 2x \sin(x^2)$ (2)

```
> print("Grid Noktalar");
> for k from 1 to N do X[k]:=(k-1)*(1/(N-1)) od;
> M:=unapply(mul((x-X[k]),k=1..N),x);
Mt:=unapply(diff(M(x),x),x):
```

"Grid Noktalar"

$$X_1 := 0$$

$$X_2 := \frac{1}{4}$$

$$X_3 := \frac{1}{2}$$

$$X_4 := \frac{3}{4}$$

$$X_5 := 1$$

$$M := x \rightarrow x \left(x - \frac{1}{4}\right) \left(x - \frac{1}{2}\right) \left(x - \frac{3}{4}\right) (x - 1)$$

(3)

```
> for i from 1 to N do
  for j from 1 to N do
```

```

        if i<>j then
            A1[i,j]:=Mt(X[i])/((X[i]-X[j])*Mt(X[j]))
        else A1[i,j]:=0
        end if
    od;
od;
> for i from 1 to N do a[i]:=-add(A1[i,j],j=1..N);
    for j from 1 to N do if i=j then A1[i,j]:=a[i] end
    if
        od:
    od:
> AA:=A1: for i from 1 to N do XX[i]:=evalf(f(X[i])) od:
> print("Agirlikli Katsayilar Matrisi"); AA;
print("Sag Taraf Vektoru"); XX;

```

"Agirlikli Katsayilar Matrisi"

$$\begin{bmatrix} -\frac{25}{3} & 16 & -12 & \frac{16}{3} & -1 \\ -1 & -\frac{10}{3} & 6 & -2 & \frac{1}{3} \\ \frac{1}{3} & -\frac{8}{3} & 0 & \frac{8}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 2 & -6 & \frac{10}{3} & 1 \\ 1 & -\frac{16}{3} & 12 & -16 & \frac{25}{3} \end{bmatrix}$$

"Sag Taraf Vektoru"

$$\begin{bmatrix} 1. \\ 1.031229659 \\ 1.247403959 \\ 1.799954010 \\ 2.682941970 \end{bmatrix}$$

(4)

```

> print("Degistirilecek satir numaralarini giriniz");
k1:=1;
for i from 1 to N do if i=k1 then AA[k1,i]:=1 else AA
[k1,i]:=0 end if od; XX[k1]:=0.;

```

"Degistirilecek satir numaralarini giriniz"

k1:=1

XX₁:=0.

(5)

```

> print("Baslangic Kosulu Eklenmis Agirlikli Katsayilar
Matrisi"); AA;
print("Baslangic Kosulu Eklenmis Sag Taraf Vektoru");

```

XX;

"Baslangic Kosulu Eklenmis Agirlikli Katsayilar Matrisi"

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & -\frac{10}{3} & 6 & -2 & \frac{1}{3} \\ \frac{1}{3} & -\frac{8}{3} & 0 & \frac{8}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 2 & -6 & \frac{10}{3} & 1 \\ 1 & -\frac{16}{3} & 12 & -16 & \frac{25}{3} \end{bmatrix}$$

"Baslangic Kosulu Eklenmis Sag Taraf Vektoru"

$$\begin{bmatrix} 0. \\ 1.031229659 \\ 1.247403959 \\ 1.799954010 \\ 2.682941970 \end{bmatrix}$$

(6)

```
> print("Yaklasik Cozumler");  
YC:=LinearSolve(AA,XX,method='LU');
```

"Yaklasik Cozumler"

$$YC:=\begin{bmatrix} 0. \\ 0.266381440666667 \\ 0.544140962250000 \\ 0.918114732750000 \\ 1.47165445966667 \end{bmatrix}$$

(7)

```
> print("Denklemin Cozumu");  
FF:=unapply(x-cos(x^2)+1,x);  
AC:=Vector(N);  
print("Kesin Cozumler");  
for i from 1 to N do AC[i]:=evalf(FF(X[i])) od: AC;
```

"Denklemin Cozumu"

$$FF:=x \rightarrow x - \cos(x^2) + 1$$

"Kesin Cozumler"

$$\begin{bmatrix} 0. \\ 0.2519524893 \\ 0.5310875783 \\ 0.9040755008 \\ 1.459697694 \end{bmatrix}$$

(8)

```
> print("Mutlak Hata"); abs(AC-YC);
```

"Mutlak Hata"

0.

0.0144289513666668

0.0130533839500002

0.0140392319500002

0.0119567656666668

(9)

```

> restart;
> with(LinearAlgebra):
> with(linalg):
> print("Ornek 4.1");
Digits:=6:
                                "Ornek 4.1"
(1)
> print("Esit Aralikli Olmayan (Non-Uniform) Dagilim");
    "Esit Aralikli Olmayan (Non-Uniform) Dagilim"
(2)
> print("Grid Nokta Sayisi"); N:=5;
> print("Sag Taraf Fonksiyonu");
f:=unapply(1+2*x*sin(x^2),x);
> XX:=Vector(N):A1:=Matrix(N,N):
B:=Matrix(N,N,shape=identity):
                                "Grid Nokta Sayisi"
                                N:=5
                                "Sag Taraf Fonksiyonu"
                                 $f:=x \rightarrow 1 + 2x \sin(x^2)$ 
(3)
> print("Grid Noktalar");
for k from 1 to N do X[k]:=evalf((1-cos((k-1)*Pi)/(
(N-1)))/2) od;
                                "Grid Noktalar"
                                 $X_1 := 0.$ 
                                 $X_2 := 0.146448$ 
                                 $X_3 := 0.500000$ 
                                 $X_4 := 0.853552$ 
                                 $X_5 := 1.$ 
(4)
> M:=unapply(mul((x-X[k]),k=1..N),x):
Mt:=unapply(diff(M(x),x),x):

> for i from 1 to N do
    for j from 1 to N do
        if i<>j then A1[i,j]:=Mt(X[i])/((X[i]-X[j])*Mt(X
[j]))
            else A1[i,j]:=0
        end if
    od;
od;
> for i from 1 to N do a[i]:=-add(A1[i,j],j=1..N);
    for j from 1 to N do
        if i=j then A1[i,j]:=a[i] end if
    od:
od:

> AA:=A1:
    for i from 1 to N do XX[i]:=evalf(f(X[i])) od:
> print("Agirlikli Katsayilar Matrisi"); AA;

```

```
print("Sag Taraf Vektoru"); XX;
```

"Agirlikli Katsayilar Matrisi"

-10.9999	13.6568	-4.00006	2.34317	-1.00000
-3.41415	1.41413	2.82846	-1.41422	0.585783
0.999986	-2.82842	-0.000004	2.82842	-0.999986
-0.585783	1.41422	-2.82846	-1.41413	3.41415
1.00000	-2.34317	4.00006	-13.6568	10.9999

"Sag Taraf Vektoru"

1.
1.00628
1.24740
2.13657
2.68294

(5)

```
> print("Degistirilecek satir numaralarini giriniz");
k1:=1;
for i from 1 to N do
  if i=k1 then AA[k1,i]:=1 else AA[k1,i]:=0
end if
od;
XX[k1]:=0.;
```

"Degistirilecek satir numaralarini giriniz"

$k1 := 1$

$XX_1 := 0.$

(6)

```
> print("Baslangic Kosulu Eklenmis Agirlikli Katsayilar
Matrisi");
AA;
print("Baslangic Kosulu Eklenmis Sag Taraf Vektoru");
XX;
```

"Baslangic Kosulu Eklenmis Agirlikli Katsayilar Matrisi"

1	0	0	0	0
-3.41415	1.41413	2.82846	-1.41422	0.585783
0.999986	-2.82842	-0.000004	2.82842	-0.999986
-0.585783	1.41422	-2.82846	-1.41413	3.41415
1.00000	-2.34317	4.00006	-13.6568	10.9999

"Baslangic Kosulu Eklenmis Sag Taraf Vektoru"

(7)

$$\begin{bmatrix} 0. \\ 1.00628 \\ 1.24740 \\ 2.13657 \\ 2.68294 \end{bmatrix}$$

(7)

```
> print("Yaklasik Cozumler");
YC:=LinearSolve(AA,XX,method='LU');
"Yaklasik Cozumler"
```

$$YC := \begin{bmatrix} 3.98348990280397 \cdot 10^{-16} \\ 0.153330371762685 \\ 0.531899893067614 \\ 1.11174620332467 \\ 1.46342086659910 \end{bmatrix}$$

(8)

```
> print("Denklemin Cozumu");
CF:=unapply(x-cos(x^2)+1,x);
"Denklemin Cozumu"
```

$$CF := x \rightarrow x - \cos(x^2) + 1$$

(9)

```
> AC:=Vector(N): print("Kesin Cozumler");
for i from 1 to N do AC[i]:=evalf(CF(X[i])) od:AC;
"Kesin Cozumler"
```

$$\begin{bmatrix} 0. \\ 0.146678 \\ 0.531088 \\ 1.10741 \\ 1.45970 \end{bmatrix}$$

(10)

```
> print("Mutlak Hata"); abs(AC-YC);
"Mutlak Hata"
```

$$\begin{bmatrix} 3.98348990280397 \cdot 10^{-16} \\ 0.00665237176268477 \\ 0.000811893067614422 \\ 0.00433620332466544 \\ 0.00372086659909798 \end{bmatrix}$$

(11)

```

> restart;
> with(LinearAlgebra):
> with(linalg):
> print("Ornek 4.1");
  Digits:=6:
                                "Ornek 4.1"

```

(1)

```

> print("Chebyshev Gauss Lobatto Dagilimi");
  "Chebyshev Gauss Lobatto Dagilimi"

```

(2)

```

> print("Grid Nokta Sayisi"); N:=5;Q:=N+1:
> print("Sag Taraf Vekturu"); f:=unapply(1+2*x*sin(x^2),
  x);
> XX:=Vector(Q): A1:=Matrix(Q,Q):
  B:=Matrix(Q,Q,shape=identity):
                                "Grid Nokta Sayisi"
                                N:=5
                                "Sag Taraf Vekturu"
                                f:=x→1+2x sin(x2)

```

(3)

```

> for k from 0 to N do Y[k+1]:=evalf(cos((k*Pi)/N)) od:
  print("Grid Noktalar"); for k from 0 to N do X[k+1]:=
    (1-Y[k+1])/2 od;
                                "Grid Noktalar"
                                X1:=0.
                                X2:=0.095492
                                X3:=0.345493
                                X4:=0.654507
                                X5:=0.904508
                                X6:=1.00000

```

(4)

```

> M:=unapply(mul((x-X[k]),k=1..Q),x):
> Mt:=unapply(diff(M(x),x),x):

> for i from 1 to Q do
  for j from 1 to Q do
    if i<>j then A1[i,j]:=Mt(X[i])/((X[i]-X[j])*Mt(X
      [j]))
    else A1[i,j]:=0
    end if
  od;
od;

> for i from 1 to Q do a[i]:=-add(A1[i,j],j=1..Q);
  for j from 1 to Q do if i=j then A1[i,j]:=a[i] end
  if
od:
od:

```

```

> AA:=A1: for i from 1 to Q do XX[i]:=evalf(f(X[i])) od:

```

```
> print("Agirlikli Katsayilar Matrisi"); AA;
print("Sag Taraf Vektoru"); XX;
```

"Agirlikli Katsayilar Matrisi"

-17.0000	20.9443	-5.78891	3.05578	-2.21115	1.00000
-5.23598	2.34157	4.00000	-1.78887	1.23606	-0.552782
1.44719	-3.99997	0.341596	3.23610	-1.78884	0.763924
-0.763924	1.78885	-3.23610	-0.34158	3.99994	-1.44719
0.552787	-1.23608	1.78888	-4.00003	-2.34159	5.23603
-1.00000	2.21117	-3.05578	5.78891	-20.9442	16.9999

"Sag Taraf Vektoru"

1.
1.00174
1.08228
1.54376
2.32035
2.68294

(5)

```
> print("Degistirilecek satir numaralarini giriniz");
k1:=1;
for i from 1 to Q do if i=k1 then AA[k1,i]:=1 else AA
[k1,i]:=0 end if od; XX[k1]:=0.;
```

"Degistirilecek satir numaralarini giriniz"

k1 := 1

XX₁ := 0.

(6)

```
> print("Baslangic Kosulu Eklenmis Agirlikli Katsayilar
Matrisi"); AA;
print("Baslangic Kosulu Eklenmis Sag Taraf Vektoru");
XX;
```

"Baslangic Kosulu Eklenmis Agirlikli Katsayilar Matrisi"

1	0	0	0	0	0
-5.23598	2.34157	4.00000	-1.78887	1.23606	-0.552782
1.44719	-3.99997	0.341596	3.23610	-1.78884	0.763924
-0.763924	1.78885	-3.23610	-0.34158	3.99994	-1.44719
0.552787	-1.23608	1.78888	-4.00003	-2.34159	5.23603
-1.00000	2.21117	-3.05578	5.78891	-20.9442	16.9999

"Baslangic Kosulu Eklenmis Sag Taraf Vektoru"

$$\begin{bmatrix} 0. \\ 1.00174 \\ 1.08228 \\ 1.54376 \\ 2.32035 \\ 2.68294 \end{bmatrix}$$

(7)

```
> print("Yaklasik Cozumler");
YC:=LinearSolve(AA,XX,method='LU');
"Yaklasik Cozumler"
```

$$YC := \begin{bmatrix} 2.91551277670959 \cdot 10^{-16} \\ 0.0943054732932635 \\ 0.352466152913785 \\ 0.743950428068123 \\ 1.22036017454195 \\ 1.45908382299088 \end{bmatrix}$$

(8)

```
> print("Denklemin Cozumu");
CF:=unapply(x-cos(x^2)+1,x);
"Denklemin Cozumu"
```

(9)

```
> AC:=Vector(Q):print("Kesin Cozumler");
for i from 1 to Q do AC[i]:=evalf(CF(X[i])) od:AC;
"Kesin Cozumler"
```

$$\begin{bmatrix} 0. \\ 0.095534 \\ 0.352609 \\ 0.744867 \\ 1.22092 \\ 1.45970 \end{bmatrix}$$

(10)

```
> print("Mutlak Hata"); abs(AC-YC);
"Mutlak Hata"
```

$$\begin{bmatrix} 2.91551277670959 \cdot 10^{-16} \\ 0.00122852670673647 \\ 0.000142847086214870 \\ 0.000916571931876775 \\ 0.000559825458049446 \\ 0.000616177009117758 \end{bmatrix}$$

(11)

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Gizem TEMELCAN
Doğum Tarihi ve Yeri : 01/01/1989 İstanbul
Yabancı Dili : İngilizce
E-posta : temelcan.gizem@gmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Lisans	Fen Edebiyat Fakültesi Matematik Bölümü	Yıldız Teknik Üniversitesi	2011
Lise	Sayısal	Bahçelievler YDA Lisesi	2007