

**T.C.  
MANİSA CELAL BAYAR ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**DOKTORA TEZİ  
MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI  
MEKANİK BİLİM DALI**

**GECİKMELİ DİFERANSİYEL DENKLEMLERİN  
PERTURBASYON-İTERASYON METODU İLE ANALİZİ**

**Muhammet Mustafa BAHŞI**

**Danışman  
Prof. Dr. Mehmet ÇEVİK**



**MANİSA-2019**

## TEZ ONAYI

Muhammet Mustafa BAŞI tarafından hazırlanan "GECİKMELİ DİFERANSİYEL DENKLEMLERİN PERTURBASYON-İTERASYON METODU İLE ANALİZİ" adlı tez çalışması 08/11/2019 tarihinde aşağıdaki jüri üyeleri önünde Celal Bayar Üniversitesi Fen Bilimleri Enstitüsü Makine Mühendisliği Anabilim Dalı'nda DOKTORA TEZİ olarak başarı ile savunulmuştur.

Danışman

Prof. Dr. Mehmet ÇEVİK  
İzmir Kâtip Çelebi Üniversitesi



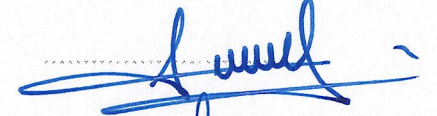
Jüri Üyesi

Prof. Dr. Mehmet SEZER  
Manisa Celal Bayar Üniversitesi



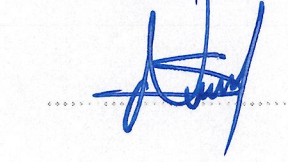
Jüri Üyesi

Doç. Dr. B. Burak ÖZHAN  
Manisa Celal Bayar Üniversitesi



Jüri Üyesi

Prof. Dr. Nurcan BAYKUŞ  
SAVAŞANERİL  
Dokuz Eylül Üniversitesi



Jüri Üyesi

Dr. Öğr. Üyesi Ebubekir ATAN  
İzmir Kâtip Çelebi Üniversitesi



## **TAAHHÜTNAME**

Bu tezin Manisa Celal Bayar Üniversitesi Mühendislik Fakültesi Makine Mühendisliği Bölümü'nde, akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

**Muhammet Mustafa BAŞI**



## İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER .....	I
SİMGELER VE KISALTMALAR DİZİNİ .....	III
ŞEKİLLER DİZİNİ.....	V
TABLO DİZİNİ .....	VII
TEŞEKKÜR.....	VIII
ÖZET.....	IX
ABSTRACT.....	X
1. GİRİŞ .....	1
2. GECİKMELİ DİFERANSİYEL DENKLEMLER .....	7
2.1. Zaman Gecikmeli Diferansiyel Denklemler.....	7
2.2. Pantograf Tipi Gecikmeli Diferansiyel Denklemler .....	8
2.3. Geçmiş Fonksiyonuna Dayalı Gecikmeli Diferansiyel Denklemler .....	9
3. PERTURBASYON-İTERASYON METODU İLE ÇÖZÜM .....	13
3.1. Zaman Gecikmeli Diferansiyel Denklemler için Perturbasyon-İterasyon Metodu.....	14
3.1.1. PIA(1,1) Algoritması .....	15
3.1.2. PIA(1,2) Algoritması .....	16
3.2. Pantograf Tipi Gecikmeli Diferansiyel Denklemler için Perturbasyon-İterasyon Metodu.....	18
3.2.1. PIA(1,1) Algoritması .....	19
3.2.2. PIA(1,2) Algoritması .....	20
3.3. Geçmiş Fonksiyonuna Dayalı Gecikmeli Diferansiyel Denklemler için Perturbasyon-İterasyon Metodu .....	22
3.3.1. PIA(1,1) Algoritması .....	25
4. PERTURBASYON-İTERASYON METODUNUN ETKİNLİĞİNİN SAYISAL ÖRNEKLERLE DOĞRULANMASI .....	29
4.1. Zaman Gecikmeli Diferansiyel Denklemler için Sayısal Örnekler.....	29
4.1.1. Sayısal Örnek 1 .....	30
4.1.2. Sayısal Örnek 2.....	33
4.1.3. Sayısal Örnek 3 .....	36
4.2. Pantograf Tipi Gecikmeli Diferansiyel Denklemler için Sayısal Örnekler .....	40
4.2.1. Sayısal Örnek 1 .....	40
4.2.2. Sayısal Örnek 2.....	43
4.2.3. Sayısal Örnek 3.....	46
4.2.4. Sayısal Örnek 4.....	48
4.2.5. Sayısal Örnek 5.....	53
4.2.6. Sayısal Örnek 6.....	55
4.3. Geçmiş Fonksiyonuna Dayalı Gecikmeli Diferansiyel Denklemler için Sayısal Örnekler .....	58
4.3.1. Sayısal Örnek 1 .....	58
4.3.2. Sayısal Örnek 2.....	62
4.3.3. Sayısal Örnek 3 .....	68
5. PERTURBASYON-İTERASYON METODUNUN MÜHENDİSLİK PROBLEMLERİNE UYGULANMASI.....	71
5.1. Gecikmeli Mathieu Denkleminin Perturbasyon-İterasyon Metodu ile Çözümü .....	71

5.2. Gecikmeli Sönümlü Mathieu Denkleminin Perturbasyon-İterasyon Metodu ile Çözümü .....	83
5.3. Frezeleme Makinesinde Kesme İşlemi Denkleminin Perturbasyon-İterasyon Metodu ile Çözümü .....	86
6. SONUÇ VE ÖNERİLER .....	95
KAYNAKLAR .....	97
EKLER .....	106
EK A. (Bazı Problemlerin MATLAB Kodları ve Çıktıları).....	106
EK A.1. (Problem (2.11)'in MATLAB Kodları ve Çıktıları).....	106
EK A.2. (Bölüm 4.1 Sayısal Örnek 1'in MATLAB Kodları ve Çıktıları)	107
EK A.3. (Bölüm 4.2 Sayısal Örnek 1'in MATLAB Kodları ve Çıktıları)	109
EK A.4. (Bölüm 4.3 Sayısal Örnek 1'in MATLAB Kodları ve Çıktıları)	110
EK B. (Bazı Problemlerin Çözüm Fonksiyonlarının MATLAB Çıktıları)....	111
EK B.1. (Gecikmeli Mathieu Denkleminin $u_2^{D_2}(t)$ Fonksiyonunun MATLAB Çıktısı) .....	111
EK B.2. (Gecikmeli Sönümlü Mathieu Denkleminin $u_2^{D_j}(t)$ Fonksiyonunun MATLAB Çıktısı) .....	112
ÖZGEÇMİŞ .....	116

## SİMGELER VE KISALTMALAR DİZİNİ

### Simgeler

$\mathbf{A}^{D_j}$	$P_{k-1}^{D_j}(t)$ polinomunun katsayılar matrisi
$a_d$	Kesme oranının radyal derinliği
$a_p$	Kesme derinliği
$D$	Problemin tanım aralığı
$D_j$	$D$ tanım bölgesinin $j$ . alt aralığı
$e_n(t)$	$u_n(t)$ çözümünün hata fonksiyonu
$E_n$	$u_n(t)$ çözümünün maksimum mutlak hata değeri
$F^{D_j}$	$F$ denkleminin $D_j$ alt aralığındaki formu
$g(\phi_j(t))$	Anahtarlama fonksiyonu (key function)
$h(t)$	Problemin geçmiş fonksiyonu (history function)
$h^{D_j}(t)$	$D_j$ alt aralığındaki geçmiş fonksiyonu
$J$	$D$ tanım aralığının alt aralık sayısı
$K_n$	Lineer normal kesme katsayısı
$K_s(t)$	Özel kesme kuvveti katsayısı
$K_t$	Lineer teğet kesme katsayısı
$m_m$	Aletin modal kütlesi
$n$	İterasyon Sayısı
$P_{k-1}^{D_j}(t)$	$(k - 1)$ . mertebeden $D_j$ alt aralığında tanımlı polinom
$u(t)$	Deplasman
$(u_c)_n(t)$	$n$ . PIA çözümünün düzeltme terimi
$u_n(t)$	$n$ . PIA çözümü
$u_n^{D_j}(t)$	Problemin $D_j$ alt aralığındaki $n$ . iterasyon çözümü
$u_n^{(k)}(t)$	$n$ . iterasyonun $k$ . mertebeden türevi
$u_0^{D_j}(t)$	$D_j$ alt aralığındaki başlangıç fonksiyonu
$u_\lambda$	Pantograf terimi
$u_\tau$	Gecikme terimi
$z_n$	Takım diş sayısı
$\varepsilon$	Perturbasyon parametresi
$\zeta$	Sönüm katsayısı

$\lambda$	Pantograf sabiti
$\tau$	Gecikme
$\phi_{ex}$	Parçanın dişliden çıkış açısı
$\phi_j(t)$	$j$ . dişin açısal konumu
$\phi_{st}$	Parçanın dişliye giriş açısı
$\omega_n$	Açısal doğal frekans

### Kısaltmalar

DAM	Direkt Adams-Moulton Metodu
DDE	Gecikmeli Diferansiyel Denklem (Delay Differential Equation)
EMHPM	Çok Kademeli Homotopi Perturbasyon Metodu (Enhanced Multistage Homotopy Perturbation Method)
HAM	Homotopi Asimtotik Metod
MSLP	Çok Ölçekli Lindstedt–Poincaré Metodu (Multi Scale Lindstedt–Poincaré Method)
OHAM	Optimal Homotopi Asimtotik Metod
PIA	Perturbasyon İterasyon Algoritması
VIM	Varyasyonel İterasyon Metodu
VMM	Değişken Çokadım Metodu (Variable Multistep Method)

## ŞEKİLLER DİZİNİ

	Sayfa
Şekil 2.1. Geçmiş fonksiyonuna dayalı (2.11) gecikmeli diferansiyel denkleminin $k = -1$ ve $\tau = 1$ için analitik çözümü ile $k = -1$ için (2.8) adi diferansiyel denkleminin analitik çözümünün karşılaştırılması.....	12
Şekil 4.1. Problem (4.3)'ün PIA(1,1) algoritması ile elde edilmiş birinci iterasyon çözümünün mutlak hata fonksiyonu.....	32
Şekil 4.2. Problem (4.3)'ün PIA(1,1) algoritması ile elde edilmiş ikinci iterasyon çözümünün mutlak hata fonksiyonu.....	32
Şekil 4.3. Problem (4.3)'ün PIA(1,1) algoritması ile elde edilmiş üçüncü iterasyon çözümünün mutlak hata fonksiyonu.....	33
Şekil 4.4. Problem (4.10)'un PIA(1,1) algoritması ile elde edilmiş birinci ve ikinci iterasyon çözümleri ile problemin tam çözümünün karşılaştırılması .....	36
Şekil 4.5. Problem (4.17)'nin PIA(1,1) algoritması ve VIM ile elde edilmiş birinci iterasyon çözümlerinin problemin tam çözümü ile karşılaştırılması.....	38
Şekil 4.6. Problem (4.17)'nin PIA(1,1) algoritması ve VIM ile elde edilmiş ikinci iterasyon çözümlerinin problemin tam çözümü ile karşılaştırılması .....	39
Şekil 4.7. Problem (4.17)'nin PIA(1,1) algoritması ile ve VIM ile elde edilmiş ikinci iterasyon çözümlerinin mutlak hata fonksiyonlarının karşılaştırılması ....	39
Şekil 4.8. Problem (4.31)'in PIA(1,1) beşinci iterasyon çözümü ile MATLAB ddsd fonksiyonu ile elde edilen sayısal çözümün karşılaştırılması.....	45
Şekil 4.9. Problem (4.39)'un PIA(1,1) ile elde edilen ilk üç iterasyon çözümü ile problemin tam çözümünün karşılaştırılması .....	48
Şekil 4.10. Problem (4.45)'in ikinci iterasyon PIA(1,1) ve VIM çözümleri ile tam çözümün karşılaştırılması .....	50
Şekil 4.11. Problem (4.45)'in üçüncü iterasyon PIA(1,1) ve VIM çözümleri ile tam çözümün karşılaştırılması .....	51
Şekil 4.12. Problem (4.45)'in PIA(1,2) algoritması ile elde edilen birinci iterasyon çözümünün VIM ile elde edilen ikinci ve üçüncü iterasyon çözümleri ve problemin tam çözümün karşılaştırılması .....	53
Şekil 4.13. Problem (4.63)'ün PIA(1,1) algoritmasının ilk üç iterasyon çözümü ile problemin tam çözümünün karşılaştırılması .....	57
Şekil 4.14. Problem (4.63)'ün PIA(1,1) algoritmasının ilk üç iterasyon çözümünün mutlak hata fonksiyonlarının grafiklerinin karşılaştırılması .....	58
Şekil 4.15. Problem (4.70)'in PIA(1,1) algoritmasının ilk üç iterasyon çözümü ile problemin tam çözümünün karşılaştırılması .....	61
Şekil 4.16. Problem (4.89)'un PIA(1,1) algoritmasının beşinci iterasyon çözümü ile MATLAB ddsd fonksiyonu ile çözümünün karşılaştırılması.....	70
Şekil 5.1. Düşey doğrultuda periyodik hareketli mesnetli matematiksel sarkaç problemi [55].....	72
Şekil 5.2. Gecikmeli Mathieu denkleminin PIA(1,1) algoritmasının ikinci ve üçüncü iterasyon çözümü ile MATLAB (dde23) sayısal çözümünün karşılaştırılması ( $\delta = 18.0$ , $\varepsilon = 1$ , $\zeta = 0.5$ , ve $\tau = 2\pi$ ) .....	78
Şekil 5.3. Gecikmeli Mathieu denkleminin gecikme teriminin katsayısının farklı değerleri $\zeta = 0, 0.5$ ve $1$ için PIA(1,1) dördüncü iterasyon çözümleri ( $\delta = 18$ , $\varepsilon = 1$ , $\tau = 2\pi$ ).....	79
Şekil 5.4. Gecikmeli Mathieu denkleminin PIA(1,1) algoritmasının beşinci ve altıncı iterasyon çözümü ile MATLAB (dde23) sayısal çözümünün karşılaştırılması ( $\delta = 18.0$ , $\varepsilon = 5$ , $\zeta = 0.5$ , ve $\tau = 2\pi$ ).....	80



Şekil 5.5. Gecikmeli Mathieu denkleminin $\varepsilon = 1, 5$ ve $10$ değerleri için PIA(1,1) çözümleri ( $\delta = 18, \zeta = 0.5, \tau = 2\pi$ ) .....	81
Şekil 5.6. Gecikmeli Mathieu denkleminin $\delta = 5, 10$ ve $18$ değerleri için PIA(1,1) çözümleri ( $\varepsilon = 1, \zeta = 0.5, \tau = 2\pi$ ) .....	82
Şekil 5.7. Gecikmeli sönümlü Mathieu denkleminin PIA(1,1) algoritmasının beşinci iterasyon çözümü ve birinci mertebe EMHPM çözümü ile sayısal çözümünün karşılaştırılması ( $\kappa = 0.2, \delta = 3.0, \varepsilon = 1, \zeta = -1, \tau = 2\pi$ )....	84
Şekil 5.8. Gecikmeli sönümlü Mathieu denkleminin sönüm teriminin katsayısının $\kappa = 0, 0.1$ ve $0.2$ değerleri için PIA(1,1) beşinci iterasyon çözümleri ( $\delta = 18, \varepsilon = 1, \tau = 2\pi, \zeta = 0.5$ ) .....	85
Şekil 5.9. Gecikmeli sönümlü Mathieu denkleminin sönüm altında gecikme teriminin katsayısının $\zeta = 0, 0.5$ ve $-0.87$ değerleri için PIA(1,1) dördüncü iterasyon çözümleri ( $\delta = 18, \varepsilon = 1, \tau = 2\pi, \kappa = 0.2$ ) .....	86
Şekil 5.10. Frezeleme işleminde gecikme etkisinin mekanik modeli [1] .....	87
Şekil 5.11. Frezeleme işleminin tek serbestlik dereceli mekanik modeli [106]..	88
Şekil 5.12. Frezeleme işleminde kesme kuvvetinin ( $F_j$ ) bileşenleri [106] .....	89
Şekil 5.13. Denklem (5.25)'in $a_p = 2\text{mm}$ ve $n = 10000$ rpm için PIA(1,1) algoritması ile dördüncü iterasyon çözümü .....	93
Şekil 5.14. Denklem (5.25)'in $a_p = 3\text{mm}$ ve $n = 10000$ rpm için PIA(1,1) algoritması ile dördüncü iterasyon çözümü .....	93
Şekil 5.15. Denklem (5.25)'in $a_p = 2\text{mm}$ ve $n = 20000$ rpm için PIA(1,1) algoritması ile dördüncü iterasyon çözümü .....	94

## TABLO DİZİNİ

	<b>Sayfa</b>
Tablo 4.1. Problem (4.3)'ün farklı iterasyon değerleri için elde edilen maksimum mutlak hata değerleri .....	33
Tablo 4.2. Problem (4.10)'un $n = 2, 3, 4, 5$ ve $6$ için PIA(1,1) algoritması ile elde edilen iterasyon çözümlerinin maksimum mutlak hata değerleri.....	36
Tablo 4.3. Problem (4.24)'ün PIA(1,1) ve diğer yöntemlerle elde edilen sayısal çözümlerinin mutlak hata değerlerinin karşılaştırılması .....	42
Tablo 4.4. Problem (4.31)'in PIA(1,1) çözümlerinin belirli noktalardaki değerlerinin karşılaştırılması .....	44
Tablo 4.5. Problem (4.31)'in PIA(1,1) çözümünün diğer sayısal yöntemler ile karşılaştırılması .....	45
Tablo 4.6. Problem (4.39)'un ilk altı iterasyon PIA(1,1) çözümlerinin maksimum mutlak hata değerleri .....	47
Tablo 4.7. Problem (4.57)'nin $n = 2, 3, \dots, 7$ için PIA(1,1) çözümlerinin maksimum mutlak hata değerleri .....	55
Tablo 4.8. Problem (4.70)'in PIA(1,1) çözümlerinin Cubic spline ve Direkt Adams-Moulton yöntemlerinin çözümleri ile karşılaştırılması .....	62
Tablo 4.9. Problem (4.77)'nin PIA(1,1) çözümünün diğer sayısal yöntemler ile karşılaştırılması .....	67
Tablo 4.10. Problem (4.77)'nin farklı iterasyon değerleri için PIA(1,1) çözümlerinin maksimum mutlak hata değerlerinin ve çözümlerin elde edilme sürelerinin karşılaştırılması .....	68

## TEŞEKKÜR

Doktora öğrenimimin her aşamasında bana destek olan, bilgi ve deneyimleri ile yol gösteren danışman hocam Sayın Prof. Dr. Mehmet ÇEVİK' e, bilgi ve tecrübesinden istifade etmekten büyük mutluluk duyduğum sevgili hocam Sayın Prof. Dr. Mehmet SEZER' e, tez çalışmam boyunca olumlu yaklaşımı ve katkılarından dolayı Sayın hocam Doç. Dr. B. Burak ÖZHAN' a, üzerimde emeği olan tüm hocalarıma ve tezimin yazım esnasında yayınlarından sıkça istifade ettiğim ve görüşüne başvurduğum Tecnolo'gico de Monterrey Makine Mühendisliği Bölümü öğretim üyesi Dr. Daniel Olvera Trejo'ya teşekkür ederim.

Desteğini bir an bile eksik hissetmediğim, çalışma arkadaşım, yol arkadaşım, canım eşim Dr. Ayşe KURT BAHŞI'ya doktora çalışmalarım da destek olmasının yanında huzurlu evlilik ve Murat Alp gibi güzel bir evlat verdiği için teşekkür ederim. Desteklerini bir an bile esirgemeyen annem, babam ve kardeşime teşekkür ederim.

Öğrenimimin her adımında iyi bir çalışma arkadaşının yanında dostluğuyla da çok şey kazandığım Dr. Sümeyye SINIR'a teşekkür ederim.

Çalışmalarımıza destek olan İzmir Katip Çelebi Üniversitesi Bilimsel Araştırma Projeleri Koordinatörlüğü'ne teşekkür ederim.

Ayrıca bu tez çalışmasına 2014-077 numaralı BAP projesi ile destek olan Manisa Celal Bayar Üniversitesi'ne teşekkür ederim.

Muhammet Mustafa BAHŞI  
Manisa, 2019

## ÖZET

### Doktora Tezi

#### Gecikmeli Diferansiyel Denklemlerin Perturbasyon-İterasyon Metodu ile Analizi

**Muhammet Mustafa BAHŞI**

**Manisa Celal Bayar Üniversitesi  
Fen Bilimleri Enstitüsü  
Makine Mühendisliği Anabilim Dalı**

**Danışman: Prof. Dr. Mehmet ÇEVİK**

Bu tezde gecikmeli diferansiyel denklemler için yeni bir yöntem olan perturbasyon-iterasyon metodu ortaya konulmuştur. Yöntem, Taylor seri açılımındaki türevin mertebesi ve yaya açılımındaki düzeltme teriminin sayısına bağlı olarak iki tip algoritma ile sunulmuştur. Gecikmeli diferansiyel denklemlerin üç farklı sınıfı için ayrı ayrı algoritmalar elde edilmiştir. İlk olarak zaman gecikmeli diferansiyel denklemler için perturbasyon-iterasyon algoritmaları geliştirilmiştir. Bu denklem sınıfı için üç farklı sayısal örnek ele alınmıştır. Sonrasında pantograf tipi gecikmeli diferansiyel denklemler için perturbasyon-iterasyon algoritmaları geliştirilmiş, bu denklem sınıfı için ise altı farklı problem ele alınmıştır. Son olarak birçok mühendislik probleminin matematiksel modeli olarak ortaya çıkan geçmiş fonksiyonuna dayalı zaman gecikmeli diferansiyel denklemler için perturbasyon-iterasyon algoritması geliştirilmiş ve üç farklı sayısal örneğe uygulanmıştır. Tüm bu sayısal örneklerin perturbasyon-iterasyon çözümleri grafik ve tablolar ile sunulmuştur. Yöntem, varyasyonel iterasyon yöntemi, sıralama yöntemi ve homotopi perturbasyon yöntemi gibi bilinen birçok yöntem ile karşılaştırılmıştır. Ayrıca geliştirilen algoritma ile fiziksel sistemlerin matematiksel modeli olan gecikmeli Mathieu ve gecikmeli sönümlü Mathieu denklemleri çözülerek denklemlerdeki gecikme teriminin etkisi ortaya konmuştur. Frezeleme kesme işleminin matematiksel denklemine de uygulanan yöntem ile sistemin kararlı ve kararsız çözümleri elde edilmiştir.

**Anahtar Kelimeler:** Perturbasyon-iterasyon metodu, perturbasyon-iterasyon algoritması, gecikmeli diferansiyel denklem, pantograf tipi gecikmeli diferansiyel denklem, geçmiş fonksiyonu, gecikmeli Mathieu denklemi, frezeleme kesme işlemi.

**2019, 117 sayfa**

## **ABSTRACT**

### **PhD Thesis**

#### **The Analysis of Delay Differential Equations with the Perturbation-Iteration Method**

**Muhammet Mustafa BAHŞI**

**Manisa Celal Bayar University  
Graduate School of Applied and Natural Sciences  
Department of Mechanical Engineering**

**Supervisor: Prof. Dr. Mehmet ÇEVİK**

In this thesis, perturbation-iteration method, which is a new method for delay differential equations, has been introduced. The method is presented with two types of algorithms depending on the order of the derivative in the Taylor series expansion and on the number of correction terms in the straightforward expansion. Perturbation-iteration algorithms are obtained separately for three different classes of delay differential equations. Firstly, perturbation-iteration algorithms are developed for time delayed differential equations. Three different numerical examples for this class of equations are discussed. Then, perturbation-iteration algorithms are developed for pantograph type delay differential equations and six different problems are considered for this class. Finally, perturbation-iteration algorithm is developed and applied to three different numerical examples for time delayed differential equations based on the history function which emerged as the mathematical model of many engineering problems. Perturbation-iteration solutions of all these numerical examples are presented with graphs and tables. The method is compared with many well-known methods such as variational iteration method, collocation method and homotopy perturbation method. In addition, the mathematical model of the physical systems of delayed Mathieu and delayed damped Mathieu equations are solved and the effect of the delay term in the equations is demonstrated. Stable and unstable solutions of the system are obtained by the method applied to the mathematical equation of the milling cutting process.

**Keywords:** Perturbation-iteration method, perturbation-iteration algorithm, delay differential equation, pantograph type delay differential equation, history function, delayed Mathieu equation, milling cutting process.

**2019, 117 pages**

## 1. GİRİŞ

Özellikle II. Dünya Savaşı sonrası otomatik kontrol sistemlerinin gelişmesi ve kullanımının yaygınlaşması sonucunda fonksiyonel diferansiyel denklemlerin özel bir sınıfı olan gecikmeli diferansiyel denklemler, otomatik kontrol sistemlerinin modeli olarak ortaya çıkmıştır. Geri bildirim kontrolü içeren herhangi bir sistem zaman gecikmeleri içermektedir. Gecikme, sınırlı zamanda hassas bilgi edinme ve ona geri tepki gösterme gerekliliğinden ortaya çıkmaktadır. Ortaya çıkan bu durum kontrol içeren sistemlerin matematiksel modellerinde gecikmeli diferansiyel denklemlerin ortaya çıkmasına neden olmuştur. Daha sonra biyolojide popülasyon dinamiği problemlerinin kimyada termokimyasal reaksiyon problemlerinin makina mühendisliğinde mekanik titreşim problemlerinin, fizikte bazı lazer problemlerinin ve çeşitli bilim dallarının birçok probleminin modeli olarak gecikmeli diferansiyel denklemler kullanılmıştır. Tüm bu problemlerin modellerinin analizinde hassas bilgi edinme gereksiniminden dolayı modellerin matematiksel denklemleri olan gecikmeli diferansiyel denklemler için hassas çözümlerin elde edilmesi de önem kazanmıştır [1].

Diğer taraftan, Avrupa'daki sanayi devrimiyle birlikte, insan gücünün yerini makineler aldı, bunlar buhar kazanları, modern değirmenler ve çeşitli makinalarla kendini gösterdi. Bu cihazlar elle yeteri kadar düzenlenemedi ve böylece otomatik kontrol sistemleri için yeni gereksinimler ortaya çıktı. Şamandıra regülatörü, sıcaklık regülatörü, basınç regülatörü ve hız kontrol cihazı gibi çeşitli kontrol cihazları icat edildi. Ondokuzuncu yüzyılın ortalarında matematikçiler otomatik kontrol sistemlerinin kararlılıklarını ilk kez inceledi ve G. B. Airy kontrol sistemlerinin kararsızlıklarını diferansiyel denklemleri kullanarak ilk kez tartıştı [1]. Sonrasında, J. C. Maxwell doğrusallaştırdığı diferansiyel denklemlerin karakteristik denkleminin köklerini bularak kararlılık analizi yapan bir yöntem geliştirdi [1]. Sistemin parametrelerinin sistemin kararlılığı üzerindeki etkisini karakteristik denklemin negatif reel kısmına bakarak belirledi [1].

Kontrol mühendisliği, uzaktan kontrol, biyoloji, nüfus dinamiği, yapılarda aktif kontrol sistemleri ve mühendisliğin diğer bazı alanlarında meydana gelen gelişmeler gecikmeli diferansiyel denklemlere olan ilgiyi her geçen gün arttırmaktadır. Bu tür denklemler ile modellenen sistemler, bu sistemlerin davranışları ve bunların farklı

yöntemlerle çözümleri konusunda birçok çalışma yapılmaktadır. Bu alandaki güncel ve kaydedeğer çalışmaların önemli bir kısmı Erneux [1] ile Shampine ve Thomson'a [2] ait kitaplarda bulunabilir. Herhangi bir geribildirim kontrolü içeren bir sistem kesinlikle zaman gecikmesi içermektedir. Sonlu zamandan kaynaklanan bu durum hassas çözüm gerektirmektedir ve bu nedenle zaman gecikmeli diferansiyel denklemlerin sayısal çözümlerinden elde edilecek sonuçların hassasiyeti büyük önem taşımaktadır.

Insperger ve Stepan [3] lineer gecikmeli sistemlerin kararlılık analizi için etkin bir sayısal yöntem sunmuşlardır. Yöntem, sürekli zaman gecikmesi olan Mathieu denklemine de uygulanmıştır. Asl ve Ulsoy [4] gecikmeli diferansiyel denklemlerin tam çözümlerini elde etmek için Lambert fonksiyonları kavramına dayalı yeni bir analitik yaklaşım sunmuşlardır. Çözüm Lambert fonksiyonları cinsinden yazılmış modların sonsuz serisi formunda bulunmuştur. Farklı örneklerdeki gecikme denklemleri için tekil modlar, serbest ve zorlamalı tepkiler incelenmiş ve sonuçlar sunulmuştur. Bocharov ve Rihan [5] biyobilimlerde kullanılan ve dinamiklerinin anlaşılmasında sayısal yaklaşımların temel bir araç olduğu gecikmeli diferansiyel denklemlerin matematiksel modellerini incelemiş ve bu modelleri araştırmak için kullanılan sayısal teknikleri yorumlamışlardır. Buna gerekçe olarak da doğadaki belli biyolojik işlemlerde, biyosistemlerin analizinde, adi diferansiyel denklemlere göre gecikmeli diferansiyel denklemlerin daha uygun ve gerçekçi olmasını göstermişlerdir. Liu vd. [6] ikinci mertebe gecikmeli diferansiyel denklemlerin analitik çözümlerini elde etmek için varyasyonel iterasyon yöntemini (VIM) kullanmışlardır. Yakınsama sonuçları elde edilmiş ve çözüm prosesinde makul bir Lagrange çarpanı seçebilmek için etkin bir teknik tasarlanmıştır. Anakira vd. [7] gecikmeli fark denklemlerinin yaklaşık analitik çözüm algoritmasını bulmak için ilk kez optimal homotopi asimtotik yöntemini kullanmışlardır. Bu yaklaşım, diğer perturbasyon yöntemlerinin aksine küçük veya büyük parametrelere bağlı değildir ve yaklaşım serilerinin yakınsamasının kontrol edilebilmesine imkân vermektedir. Dehghan ve Salehi [8] biyolojideki doğrusal olmayan zaman gecikmeli bir modelin varyasyonel iterasyon ve Adomian ayrıştırma yöntemleri gibi yarı-analitik yaklaşımlarla çözümlerini elde etmişlerdir. Bu yöntemler küçük perturbasyonlara, doğrusallaştırmaya ve değişkenlerin ayrıklaştırılmasına gerek duymadıkları için işlem hacmini azaltmaktadırlar. Xu vd. [9] negatif sönümleme ve gecikmeli geri besleme kontrolü olan gecikmeli bir osilatörün

dinamik analizini yapmışlardır. Çalışma sonucunda, düzgün ayarlanmış gecikme sayesinde kontrollü gecikmeli sistemin kararlı dengeye veya periyodik çözümlere veya yarı-periyodik çözümlere veya birlikte var olan kararlı çözümlere sahip olabileceğini göstermiştir. Gecikmeli diferansiyel denklemlerin farklı yöntemlerle analizi konusunda birçok bilim insanı çalışmalar gerçekleştirmiştir. Saeed ve Rahman [10] gecikmeli diferansiyel denklem sistemlerini ve Evans ve Raslan [11] da gecikmeli diferansiyel denklemleri Adomian decomposition yöntemi ile çözmüşlerdir. Vanani ve Aminataei gecikmeli diferansiyel denklemlerin bir alt sınıfı için çoklu karesel yaklaşım ile sayısal çözümler elde etmiştir [12]. Yu [13] varyasyonel iterasyon yöntemi ile çoklu-pantograf tipi gecikmeli denklemleri ve Wang ve Wang [14] ise Legendre-Gauss sıralama yöntemi ile lineer olmayan gecikmeli diferansiyel denklemleri çözmüşlerdir. Bu araştırmacıların dışında birçok araştırmacı tarafından yapılmış başkaca çalışmalar da mevcuttur.

Gülsu ve Sezer [15, 16] geliştirdikleri matris işlemlerine dayalı Taylor matris yöntemini kullanarak lineer fark denklemleri ve lineer diferansiyel-fark denklemleri için sayısal çözümler bulmuşlardır. Gülsu vd. [17] benzer bir çalışmayı Legendre ve Taylor polinomlarını hibrit ederek matris yöntemiyle lineer fark denklemlerinin sayısal çözümünü bulmuşlardır. Matris işlemlerine dayalı bu seri çözüm yöntemi sıralama yöntemi ile birleştirilerek farklı seriler ile çözüm yöntemlerinin önü açılmıştır. Bu konuda, Gülsu vd. [18, 19] Hermit ve Chebyshev polinomları ile lineer diferansiyel-fark denklemlerinin, Yüzbaşı [20, 21] Bessel polinomlarını kullanarak lineer tekil diferansiyel-fark denklemlerinin ve retartded ve advanced tipi diferansiyel denklemlerin sayısal çözümlerini elde etmiştir. Gökmen ve Sezer [22], Taylor polinomlarını kullanarak fark denklemleri için, Kurt vd. [23] Fibonacci polinomlarını kullanarak diferansiyel-fark denklemleri için matrise dayalı sıralama yöntemini kullanmışlardır. Erdem ve Yalçınbaş [24] diferansiyel-fark denklemleri için sayısal çözüm algoritmasını Bernoulli serileri ile matris işlemlerine dayalı sıralama metodu kullanarak geliştirmişlerdir.

Yüzbaşı ve Sezer [25] lineer gecikmeli diferansiyel denklemler için Legendre serilerini kullanarak sayısal çözümler elde etmiş ve residüel hata fonksiyonuna dayalı hata tahmin algoritmaları geliştirmiştir. Yüzbaşı ve Sezer [26] başka bir çalışmalarında ise gecikmeli diferansiyel denklemlerin sınır değer problemleri için hata tahmin



algoritmaları ve sayısal çözüm yöntemleri geliştirmişlerdir. Yüzbaşı vd. [27, 28] lineer gecikmeli diferansiyel denklemler için hata tahmin algoritmaları ve sayısal çözüm yöntemleri geliştirmişlerdir. Bu hata tahmin algoritmalarını kullanarak tam çözümü bilinmeyen gecikmeli tek serbestlik dereceli kütle yay sisteminin matematiksel modeli için Çevik vd. [29] üstel formdaki serileri kullanarak sıralama yöntemi ile bazı modellerin sayısal çözümlerini elde etmişlerdir. Bu çalışma ile yöntem ve hata tahmin algoritmaları bir mühendislik problemine uygulanmıştır. Bahşı vd. [30] ortogonal üstel fonksiyonlara dayalı sıralama yöntemi ile gecikmeli ve pantograf tipi denklemler için sayısal çözümler elde etmiş ve hata tahmin algoritmaları sunmuşlardır.

Gecikmeli diferansiyel denklemlerin çözümü için bu yöntemlerin dışında perturbasyon yöntemleri de geliştirilmiştir [1]. Wang ve Hu [31] ikinci mertebe gecikmeli diferansiyel denklemlerin çözümü için çok ölçekli perturbasyon yöntemini ve Lindstedt–Poincaré yöntemini kullanmıştır. Shakeri ve Dehghan [32] gecikmeli diferansiyel denklemlere ve Biazar ve Ghanbari [33] ise pantograf tipi gecikmeli denklemlere homotopi perturbasyon yöntemini uygulamışlardır. Ayrıca perturbasyon yöntemleri fark denklemleri, diferansiyel denklemler, integro diferansiyel denklemler ve cebirsel denklemler için de sıkça kullanılmıştır [34].

Perturbasyon yöntemlerinin temel kısıtlaması küçük bir parametrenin olma zorunluluğudur. Bu parametre, verilen denklemden kaynaklanan fiziksel parametre ya da alternatif olarak suni olarak eklenen bir parametredir. Birçok yöntem bu kısıtlamanın üzerine kurulmuştur. Bu kısıtlamanın kaldırılması için perturbasyon yöntemlerinin hibrit olarak kullanıldığı birçok yöntem geliştirilmiştir. Bunlardan biri çok ölçekli metot (multiple scales) ile Lindstedt–Poincaré yönteminin beraber kullanıldığı çok ölçekli Lindstedt–Poincaré (MSLP) yöntemidir [35]. Diğer bir hibrit perturbasyon yöntemi ise, direk perturbasyon yöntemi ve iterasyon yöntemlerinin birleşimi olan iterasyon perturbasyon yöntemleridir [36-41]. İterasyon perturbasyon yöntemleri kuvvetli lineer olmayan salınım problemi [26], van der Pol denklemi [39] ve tek serbestlik dereceli lineer olmayan salınımlı mekanik sistemin denklemi [40] gibi temel mühendislik problemlerine uygulanmıştır. İterasyon perturbasyon yöntemlerine alternatif ve daha etkili bir hibrit yöntem, Pakdemirli [38] tarafından perturbasyon-iterasyon yöntemi olarak sunulmuştur. Perturbasyon-iterasyon yöntemi, perturbasyon açılımındaki düzeltme teriminin sayısına ve Taylor serisine açılımının türevinin

mertebesine göre perturbasyon iterasyon algoritmaları (PIA) ile sunulmuştur. Bu yeni perturbasyon-iterasyon yöntemi ilk olarak cebirsel denklemler için kök bulma algoritması olarak sunulmuştur [42]. Sonrasında yöntem, yüksek dereceli cebirsel denklemler için geliştirilmiştir [43, 44]. Daha sonrasında, bir fizik problemi olan Bratu tipi [45] diferansiyel denklemlere ve Fredholm ve Volterra tipi integral denklemlere [46] uygulanmıştır. Bunun dışında kabul gören bu yeni yöntem birçok araştırmacı tarafından farklı problemler için geliştirilmektedir. Yıldız vd. [47] değişken viskoziteye sahip bir Newton akışkanı ve üçüncü mertebe akışkanın paralel plaka akışı problemi için geliştirdikleri perturbasyon-iterasyon yöntemi ile direk perturbasyon yöntemine göre daha iyi sonuçlar elde etmişlerdir. Ayrıca yöntem fin denklemine [48], ikinci mertebe sinüs hiperbolik lineer olmayan Troesch denklemine [49] ve Lotka-Volterra denklemine [50] uygulanmıştır.

Bildik ve Deniz perturbasyon-iterasyon yöntemini modifiye ederek yöntemin etkinliğini arttıran çalışmalar yayınlamışlardır. Bildik ve Deniz modifiye edilmiş perturbasyon-iterasyon yöntemi ile lineer olmayan diferansiyel denklemlerin bazı sınıfları [51], Klein-Gordon denklemi [52], Burger ve dalga denklemleri [53] için perturbasyon-iterasyon algoritmaları geliştirmişlerdir. Bahşı ve Çevik [54] pantograf tipi gecikmeli diferansiyel denklemler sayısal çözümleri için perturbasyon-iterasyon yöntemini sunmuşlardır. Kovacic vd. [55] Mathieu denklemlerinin farklı sınıfları ile ilgili genel bir çalışma sunmuşlardır.

Bu tezde, birçok bilim insanı tarafından ele alınan fonksiyonel diferansiyel denklemlerin bir alt sınıfı olan gecikmeli diferansiyel denklemler ve diğer bir alt sınıfı olan pantograf tipi gecikmeli diferansiyel denklemlerin sayısal çözümlerini bulmak için yeni bir perturbasyon yöntemi geliştirilmesi amaçlanmaktadır. Birçok araştırmacı için güncel bir konu olan gecikmeli diferansiyel denklemler için klasik perturbasyon yöntemlerinin aksine küçük perturbasyon parametresi ihtiyacı olmayan yeni bir yöntem olan perturbasyon-iterasyon metodu önerilmiştir. Metot, perturbasyon açılımındaki düzeltme teriminin sayısına ve Taylor seri açılımındaki türevin mertebesine göre iki tip algoritma ile sunulmuştur. Sunulan algoritmalar ile elde edilen perturbasyon-iterasyon çözümleri, farklı yöntemler ile elde edilen çözümler ile karşılaştırılmıştır. Literatürde sıkça karşılaşılan zaman gecikmeli diferansiyel denklemler için geliştirilen perturbasyon-iterasyon algoritmaları, değişken katsayılı

yüksek mertebeli üç farklı sayısal örneğe uygulanmış ve bilinen bir iterasyon yöntemi ile karşılaştırılmıştır. Pantograf tipi gecikmeli diferansiyel denklemler için geliştirilen algoritmalar ise altı farklı örneğe uygulanmıştır. İki lineer ve dört lineer olmayan altı farklı problem ele alınmış, literatürde bilinen birçok yöntemle göre daha iyi sonuçlar elde edilmiştir. Birçok mühendislik probleminin matematiksel modeli olarak ortaya çıkan geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklemler için perturbasyon-iterasyon algoritması geliştirilmiş ve üç farklı sayısal örneğe uygulanmıştır. Tüm bu sayısal örneklerin perturbasyon-iterasyon çözümleri grafik ve tablolar ile sunulmuş ve birçok bilinen yöntemle karşılaştırılmıştır. Geliştirilen algoritmalar ile fiziksel sistemlerin matematiksel modeli olan gecikmeli Mathieu ve gecikmeli sönümlü Mathieu denklemleri çözülerek denklemdaki gecikme teriminin etkisi ortaya konmuştur. Ayrıca Frezeleme kesme işleminin matematiksel denklemine de uygulanan yöntemle sistemin kararlı ve kararsız çözümleri elde edilmiştir.

## 2. GECİKMELİ DİFERANSİYEL DENKLEMLER

Gecikmeli diferansiyel denklemler fonksiyonel diferansiyel denklemlerin bir sınıfıdır [56]. Myshkis [57] fonksiyonel diferansiyel denklemleri,  $t$  zaman bağımsız değişkeni ve  $u(t)$  bağımlı değişkeninin çeşitli formlarını içeren denklemler olarak tanımlamıştır. Bu denklemler birçok mühendislik probleminin modeli olarak ortaya çıkmalarının yanında fizik, kimya, biyoloji ve ekonomi gibi temel bilimlerde de birçok olayın matematiksel olarak modellenmesinde sıkça kullanılmışlardır.

Fonksiyonel diferansiyel denklemler literatürde birkaç farklı sınıfa ayrılmışlardır. Fonksiyonel diferansiyel denklemlerin bazı özel sınıfları; retarded fonksiyonel diferansiyel denklemler, neutral fonksiyonel diferansiyel denklemler, stochastic fonksiyonel diferansiyel denklemler ve stochastic retarded fonksiyonel diferansiyel denklemler olarak sıralanabilir [58].

Neutral tipi fonksiyonel diferansiyel denklemler literatürde gecikmeli diferansiyel denklemler veya fark denklemleri olarak da adlandırılmaktadır [58]. Bu tezde fonksiyonel diferansiyel denklemlerin alt sınıfı olan gecikmeli diferansiyel denklemler ele alınacaktır. Gecikmeli diferansiyel denklemler, tek serbestlik dereceli sistemlerde [29], vakum borulu salınım probleminde [59], takım tezgâhının titreşim probleminde [60], tren pantograflarının matematiksel denkleminde [61] ve uçak motorlarının ayrılabilir kaplamasının sökülmesinin matematiksel olarak modellenmesinde [62, 63] olduğu gibi birçok mühendislik probleminin matematiksel modeli olarak karşımıza çıkmaktadır. Bu problemlerin ve diğer alanlarda karşılaşılan problemlerin analizi için birçok araştırmacı gecikmeli diferansiyel denklemleri bağımsız terimleri ve ele alınan koşulları ile birbirinden farklı sınıflara ayırarak incelemiştir. Bu denklemlerin tamamını temel teşkil eden ve literatürde en sık karşılaşılan problemler üç ana başlık altında incelenmiştir.

### 2.1. Zaman Gecikmeli Diferansiyel Denklemler

Bu gruptaki denklemler  $\tau > 0$  ve  $t$  bağımsız değişken olmak üzere  $t - \tau$  noktasının reel sayılar üzerinde tanımlı  $u$  fonksiyonu altındaki görüntüsü  $u(t - \tau)$  terimini içeren klasik diferansiyel denklem teorisi altında ele alınan gecikmeli diferansiyel denklemlerdir. Araştırmacılar tarafından birçok sayısal yöntem

kullanılarak ele alınan problemlerin sayısal çözümleri elde edilmiştir [64-66]. Bu problemler başlangıç veya koşulları altında ele alınmıştır [15-29].

Yüksek mertebeden değişken katsayılı lineer gecikmeli diferansiyel denklem

$$\sum_{k=0}^m P_k(t)u^{(k)}(t) + u(t - \tau) = g(t) \quad (2.1)$$

başlangıç koşulları olan

$$u^{(k)}(0) = \alpha_k, \quad k = 0, 1, 2, \dots, m - 1 \quad (2.2)$$

ile verilir [29]. Burada  $P_k(t)$  ve  $g(t)$ ;  $0 \leq t \leq \infty$  zaman aralığında sürekli fonksiyonlar ve  $\alpha_k$  bilinen sabitlerdir.

## 2.2. Pantograf Tipi Gecikmeli Diferansiyel Denklemler

Orantılı (Proportional) gecikmeli diferansiyel denklemler olarak da bilinen, pantograf tipi gecikmeli diferansiyel denklemler, fonksiyonel diferansiyel denklemlerin özel bir tipi olan diferansiyel denklemlerdir. Bu denklemler uygulamalı matematiğin elektrodinamik, kontrol sistemleri, sayı teorisi, olasılık ve kuantum mekaniği gibi birçok farklı alanında problemlerin modellenmesinde kullanılmaktadır. Birçok araştırmacı pantograf tipi gecikmeli diferansiyel denklemler için analitik ve sayısal yöntemler geliştirmişlerdir [67-74]. Liu ve Li pantograf tipi diferansiyel denklemlerin analitik çözümlerini ve  $\theta$ -yöntemi ile sayısal çözümlerini bulmuşlardır [67]. Bu denklemlerin sıralama yöntemleri ile sayısal çözümlerini bulmak için araştırmacılar Hermite [69], Taylor [70], Chebyshev [71] polinomları ile matris işlemlerine dayalı yöntemler sunmuşlardır. Brahim vd. lineer olmayan pantograf tipi gecikmeli diferansiyel denklemlerin tam çözümü için prosedürler geliştirmişlerdir [72].

$a, b, \lambda$  birer reel sayı,  $|a| + b < 0$  ve  $0 < \lambda < 1$  koşullarını sağlayan birinci mertebe pantograf tipi başlangıç değer problemi

$$\frac{du}{dt} = au(t) + bu(\lambda t), \quad u(0) = 1 \quad (2.3)$$

için Liu [75]  $\theta$ -yöntemini sunmuştur.

$T > 0$  ve  $0 < \lambda < 1$  olmak üzere ikinci mertebe pantograf tipi gecikmeli diferansiyel denkleminin

$$\frac{d^2u}{dt^2} = f(t, u(t), u(\lambda t)), \quad t \in [0, T] \quad (2.4)$$

sınır değer koşulları,

$$u(0) = b, \quad u(T) = c \quad (2.5)$$

altındaki çözümü için Bica vd. [73] sabit nokta tekniği ile sayısal çözümler elde etmiştir. Ayrıca yapay sinir ağları, örnek arama yöntemi, genetik algoritmaları, aktif küme algoritması gibi yöntemler ile (2.4)'ün çözümleri elde etmiştir [73].

### 2.3. Geçmiş Fonksiyonuna Dayalı Gecikmeli Diferansiyel Denklemler

Bazı gecikmeli diferansiyel denklemlerin zamana bağlı çözümü için sadece belirli bir andaki başlangıç durumunun bilinmesi tek başına fiziksel sistemlerin tanımlanmasında yeterli olmamaktadır. Bun tip denklemlerde zaman gecikmesi  $\tau$ 'ya eşit uzunluktaki bir aralıktaki çözüm profili verilmesi gerekmektedir. Yani,  $t = -\tau$  ile  $t = 0$  aralığında sonsuz boyutlu bir başlangıç koşulları seti tanımlanması gerekmektedir. Dolayısıyla bir zaman gecikmeli diferansiyel denklem sonsuz boyutlu bir problemdir. Bu da geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklemlerin yapıcı adi diferansiyel denklemlerden oldukça farklı olduğunu göstermektedir [1].

Yüksek mertebeden geçmiş fonksiyonuna dayalı koşulları olan gecikmeli diferansiyel denklem

$$u^{(k)}(t) = f(t, u(t), u(t - \tau)), \quad a \leq t \leq b \quad (2.6)$$

ve koşulları

$$\begin{aligned} u^{(i)}(t) &= \alpha_i, & 0 \leq i \leq k-1 \\ u(t) &= h(t), & a-\tau \leq t \leq a \end{aligned} \quad (2.7)$$

olarak tanımlanır [76]. Burada  $\tau > 0$  ve  $f(t)$  sürekli fonksiyon ve  $h(t)$  fonksiyonu  $[a-\tau, a]$  aralığında sürekli geçmiş fonksiyonudur (history function). (2.7) de tanımlı geçmiş fonksiyonuna dayalı koşulları olan (2.6) denklemi Bölüm 2.1'de tanımlanan klasik gecikmeli diferansiyel denklemlerden farklı olarak geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklem olarak adlandırılmaktadır.

Geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklemin adi diferansiyel denklemden nasıl farklılaştığını görmek için temel birinci mertebe lineer adi bir diferansiyel denklem ele alalım.

$$\frac{du}{dt} = ku(t) \quad (2.8)$$

diferansiyel denkleminin

$$u(0) = 1 \quad (2.9)$$

başlangıç fonksiyonu altındaki çözümü

$$u(t) = e^{kt} \quad (2.10)$$

üstel fonksiyon şeklindedir.

Fiziksel olarak, bu çözümde  $u(0) = 1$  başlangıç koşulu herhangi bir  $t$  anındaki bağımlı değişkenin değerini bulmamızı sağlamasına rağmen problemin başladığı andan öncesi yani geçmişi hakkında herhangi bir bilgi vermemektedir.

Bir gecikmeli diferansiyel denklem için gecikme teriminin probleme eklenmesi, problemin başlangıcını ve herhangi bir  $t$  anındaki durumunu doğrudan etkilemektedir.

(2.8) adi diferansiyel denklemini ile karşılaştırmak için aşağıdaki gecikmeli diferansiyel denklemi

$$\frac{du}{dt} = ku(t - \tau) \quad (2.11)$$

ve (2.11) denkleminin geçmiş fonksiyonunu

$$u(t) = 1, \quad -\tau \leq t \leq 0 \quad (2.12)$$

ele alalım. (2.11) denkleminin  $u(t - \tau)$  terimi  $t$  anında problemin  $t - \tau$  anını verir. Burada,  $\tau$  gecikme olarak adlandırılır. Ayrıca (2.8) adi diferansiyel denkleminin başlangıç değeri (2.9)'dan farklı olarak (2.11) gecikmeli diferansiyel denkleminin  $[-\tau, 0)$  aralığında tanımlı geçmiş fonksiyonu ile verilmiştir. Bu farklılığın problemin çözümünü doğrudan etkileyeceği açıktır.

(2.11) gecikmeli diferansiyel denklemi  $t = 0$  başlangıç anını değil başlangıç anından önceki geçmiş fonksiyonunu da sağlamalıdır. Bunun için problemin başlangıç anından başlayarak  $\tau$  büyüklüğündeki yarı açık aralıklarda adım yöntemini [77] kullanarak analitik olarak çözelim.

$[0, \tau)$  aralığı için (2.11) denklemini

$$\frac{du}{dt} = k, \quad u(0) = 1, \quad 0 \leq t \leq \tau \quad (2.13)$$

olur. Buradan (2.13) ün çözümü;

$$u(t) = kt + 1, \quad 0 \leq t \leq \tau \quad (2.14)$$



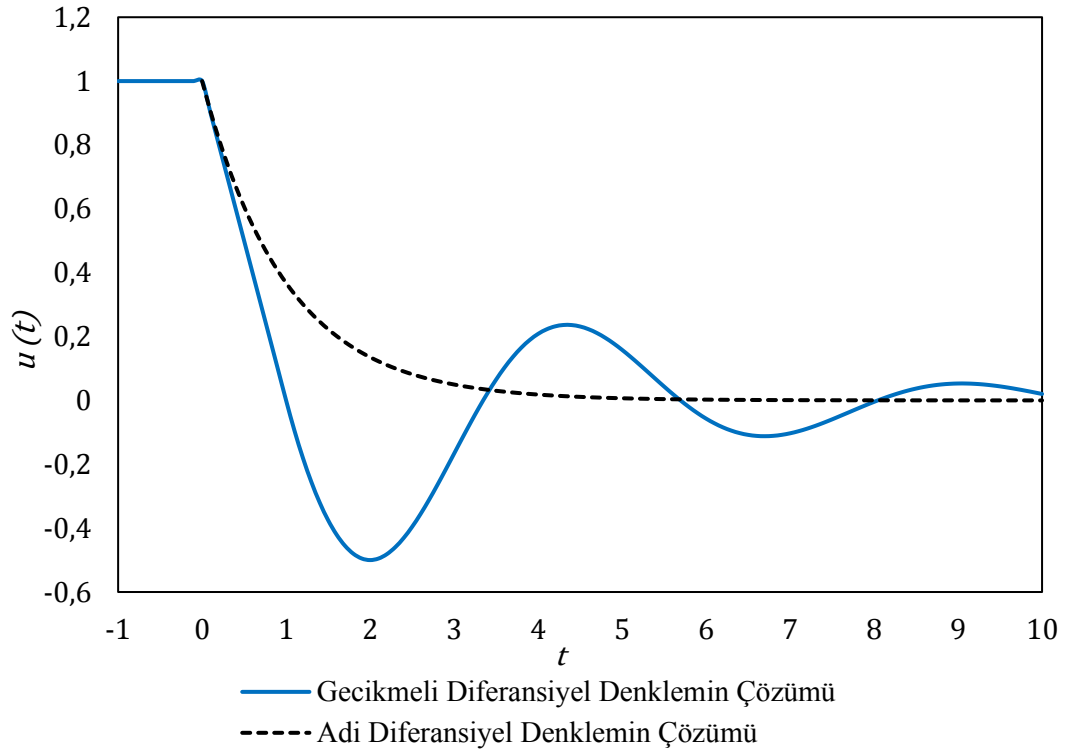
olacaktır.  $[\tau, 2\tau]$  aralığı için (2.11) denklemi (2.14)'den

$$\frac{du}{dt} = k^2t - k^2\tau + k, \quad u(\tau) = k\tau + 1, \quad \tau \leq t \leq 2\tau \quad (2.15)$$

olur. Buradan (2.15)'in çözümü;

$$u(t) = \frac{k^2}{2}t^2 - (k^2\tau)t + kt + \frac{(k\tau)^2}{2} + 1, \quad \tau \leq t \leq 2\tau \quad (2.16)$$

olacaktır. Problemin bu şekilde çözülmüşse  $k = -1$  ve  $\tau = 1$  için çözümü Şekil 2.1'de görülmektedir. Ayrıca problemin çözümünün MATLAB kodu ve çıktıları EK A.1'de verilmiştir. Şekil 2.1'den görüldüğü üzere (2.11) gecikmeli diferansiyel denklemi salınım yapmaktadır. Yani (2.8) adi diferansiyel denklemin çözümünden farklıdır.



**Şekil 2.1.** Geçmiş fonksiyonuna dayalı (2.11) gecikmeli diferansiyel denkleminin  $k = -1$  ve  $\tau = 1$  için analitik çözümü ile  $k = -1$  için (2.8) adi diferansiyel denkleminin analitik çözümünün karşılaştırılması

### 3. PERTURBASYON-İTERASYON METODU İLE ÇÖZÜM

Perturbasyon yöntemleri [34] fizik ve mühendislikte ortaya çıkan lineer olmayan matematiksel modellerin analizinde kullanılan en yaygın yaklaşık çözüm yöntemlerinden birisidir. Direk perturbasyon yönteminde  $\varepsilon \ll 1$  olacak şekilde  $u(t)$  bağımlı değişkenin direk perturbasyon açılımı aşağıdaki gibi tanımlanır.

$$u(t) = u_0(t) + \varepsilon u_{c_1}(t) + \varepsilon^2 u_{c_2}(t) + \dots \quad (3.1)$$

Burada  $u_{c_1}(t), u_{c_2}(t), \dots$  ler düzeltme terimleridir. (3.1) direk açılımı ile problemin  $\varepsilon^n$  ( $n > 1$ ) mertebesindeki çözümü; açılımın  $u_0(t)$  ve  $u_{c_1}(t), u_{c_2}(t), \dots, u_{c_n}(t)$  düzeltme terimlerini içeren  $n + 1$  lineer denklem sisteminin çözümü ile elde edilir. Direk perturbasyon yönteminin en önemli kısıtı küçük  $\varepsilon$  parametresi bulunma zorunluluğudur. Bu kısıtın ortadan kaldırılması için bazı alternatif yöntemler geliştirilmiştir. Bu yöntemlerden bazıları modifiyeli Lindstedt–Poincaré yöntemi [78], lineerleştirilmiş perturbasyon yöntemi [79], Homotopy perturbasyon yöntemi [80] ve çok ölçekli Lindstedt–Poincaré yöntemidir [35]. Bu yöntemlerde genellikle lineer olmayan terimler belirli dönüşümlerle lineerleştirilerek problemler çözülmektedir [79].

İterasyon-perturbasyon yöntemi olarak adlandırılan perturbasyon yöntemi de küçük parametre kısıtını ortadan kaldırmak için geliştirilmiştir [81]. Bu yöntem birçok probleme uygulanmıştır. [36, 37, 39-41, 81-87]. He tarafından geliştirilen iterasyon perturbasyon yöntemlerinde [81] bir önceki iterasyon sonuçları kullanılarak iterasyon fonksiyonunda lineer olmayan terimler lineerleştirilir [45]. Genellikle iterasyon-perturbasyon yöntemlerinde iterasyon uygulanmadan önce denklemler farklı forma dönüştürülürler. Bu yüzden geliştirilen algoritmalar belirli problemler için geçerlidirler [45].

Bu tezde geliştirilecek olan perturbasyon-iterasyon yöntemi iterasyon-perturbasyon yöntemi gibi iterasyona dayalı bir perturbasyon yöntemi olmasına rağmen temel farklılıklar içermektedir. Perturbasyon-iterasyon yöntemi yukarıda bahsedilen yöntemlerin aksine yöntemin uygulanabilmesi için ele alınan denklemin

farklı formda yazılmasına ve ön koşullar bulunmasına gerek olmayan bir yöntemdir. Ayrıca yöntem sadece belirli bir problem için değil genel denklem sınıfları için çözüm algoritmaları sunmaktadır [45]. Perturbasyon-iterasyon yöntemi, cebirsel denklemlere [42-44], Bratu denklemine [45], Fredholm ve Volterra tipi integral denklemlere [46], değişken viskoziteye sahip bir Newtonyen akışkan ve üçüncü dereceden akışkanın paralel plaka akışı problemine [47], fin denklemine [48], ikinci mertebe sinüs hiperbolik lineer olmayan Troesch denklemine [49] ve Lotka-Volterra denklemine [50] uygulanmış ve etkili çözümler elde edilmiştir.

Perturbasyon-iterasyon algoritmaları, kısaca PIA(n,m) olarak tanımlanır. Burada n değeri (3.1) perturbasyon açılımındaki düzeltme terimi sayısını ve m ise Taylor seri açılımındaki türevin mertebesini tanımlar. Genellikle  $n \leq m$  dir, aksi halde düzeltme terimi hesaplanamaz. n ve m değerlerinin artırılması ile yöntemin daha doğru ve etkin sonuçlar vermeyeceği yukarıda bahsedilen çalışmalarda gözlemlenmiştir. Ayrıca bu çalışmalarda araştırmacılar sıklıkla PIA(1,1) ve PIA(1,2) algoritmalarını tercih ederken ve bazı çalışmalarda ise PIA(2,2) algoritması da sunulmuştur.

Bu tezde zaman gecikmeli diferansiyel denklemler, pantograf tipi gecikmeli diferansiyel denklemler için PIA(1,1) ve PIA(1,2) algoritmaları ve geçmiş fonksiyonuna dayalı (history function) zaman gecikmeli diferansiyel denklemler için PIA(1,1) algoritması geliştirilmiştir.

### 3.1. Zaman Gecikmeli Diferansiyel Denklemler için Perturbasyon İterasyon Metodu

$\tau \in \mathbb{R}^+$  olmak üzere  $u(t - \tau)$  gecikme terimini içeren k. mertebe gecikmeli diferansiyel denklemin kapalı formdaki en genel hali aşağıda verilmiştir.

$$F(u^{(k)}, \dots, u', u, u_\tau, \varepsilon, t) = 0, \quad 0 \leq t \leq b \quad (3.2)$$

Burada  $u = u(t)$ ,  $u_\tau = u(t - \tau)$  ve  $\varepsilon$  perturbasyon parametresidir. (3.2) denklemi, (2.1) denklemi gibi birçok araştırmacı tarafından ele alınan problemlerin tamamını kapsadığından dolayı (3.2) denklemi için algoritmalar geliştirilecektir.

(3.2) denkleminin başlangıç koşulları

$$u^{(i)}(0) = \gamma_i, \quad i = 0, 1, 2, \dots, k - 1 \quad (3.3)$$

şeklindedir.

### 3.1.1. PIA(1,1) Algoritması

PIA(1,1) algoritmasında, bir düzeltme terimli pertürbasyon açılımı ve birinci mertebe türev için Taylor açılımı yapılmıştır. (3.2) denkleminde verilen  $u(t)$  bağımlı değişkenine her bir iterasyon için bir düzeltme terimli direk açılımı (yaya açılımı) uygularsak (3.1)'den aşağıda verilen eşitlik elde edilir.

$$u_{n+1} = u_n + \varepsilon(u_c)_n \quad (3.4)$$

burada  $(u_c)_n$  pertürbasyon-iterasyon algoritmasının  $n$ . düzeltme terimidir. Daha sonra (3.4) denklemi (3.2) denkleminde yerine konur ve birinci mertebe türeve göre Taylor serisine açılırsa

$$\begin{aligned} & F(u_n^{(k)}, \dots, u_n', u_n, (u_\tau)_n, 0, t) + \\ & F_{u^{(k)}}(u_n^{(k)}, \dots, u_n', u_n, (u_\tau)_n, 0, t) \varepsilon(u_c)_n^{(k)} + \dots + \\ & F_{u'}(u_n^{(k)}, \dots, u_n', u_n, (u_\tau)_n, 0, t) \varepsilon(u_c)_n' + \\ & F_u(u_n^{(k)}, \dots, u_n', u_n, (u_\tau)_n, 0, t) \varepsilon(u_c)_n + \\ & F_{u_\tau}(u_n^{(k)}, \dots, u_n', u_n, (u_\tau)_n, 0, t) \varepsilon((u_\tau)_c)_n + \\ & F_\varepsilon(u_n^{(k)}, \dots, u_n', u_n, (u_\tau)_n, 0, t) \varepsilon = 0 \end{aligned} \quad (3.5)$$

denklemi elde edilir. Burada  $( )'$  zamana göre türevi tanımlar ve

$$F_{u^{(k)}} = \frac{\partial F}{\partial u^{(k)}}, \dots, F_{u'} = \frac{\partial F}{\partial u'}, F_u = \frac{\partial F}{\partial u}, F_{u_\tau} = \frac{\partial F}{\partial u_\tau}, F_\varepsilon = \frac{\partial F}{\partial \varepsilon} \quad (3.6)$$

olarak elde edilir. Tüm türevler  $\varepsilon = 0$  da bulunmuştur ve (3.5) denklemini aşağıdaki gibi daha sade formda yazılabilir.

$$F_{u^{(k)}}(u_c)_n^{(k)} + \dots + F_{u'}(u_c)_n' + F_u(u_c)_n + F_{u_\tau}((u_\tau)_c)_n = -F_\varepsilon - \frac{F}{\varepsilon} \quad (3.7)$$

(3.2) denkleminin PIA(1,1) çözümü için (3.3)'de verilen koşullarını sağlayan en düşük mertebeden polinom başlangıç fonksiyonu  $u_0$  olarak tanımlanırsa, (3.7) algoritması kullanılarak  $(u_c)_0$  düzeltme terimi bulunur. Bulunan düzeltme terimi ve  $u_0$  çözümü (3.4) yaya açılımında yazılarak  $u_1$  birinci iterasyon çözümü bulunur. Benzer şekilde, iterasyona bağlı (3.4) direk açılımı ve (3.7) algoritması ile bir önceki iterasyonda elde edilen PIA(1,1) çözümü kullanılarak problemin  $n$ . iterasyon çözümü  $u_n$  bulunur.

### 3.1.2. PIA(1,2) Algoritması

PIA(1,2) algoritmasında, perturbasyon açılımında bir düzeltme terimi ve Taylor açılımında ikinci mertebe türev için açılım yapılmıştır. (3.4) denklemini (3.2) denkleminde yerine konur ve ikinci mertebe türeve göre Taylor serisine açılırsa

$$\begin{aligned}
& F(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \\
& + F_{u^{(k)}}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon (u_c)_n^{(k)} + \dots \\
& + F_{u'}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon (u_c)'_n \\
& + F_u(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon (u_c)_n \\
& + F_{u_\tau}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon ((u_\tau)_c)_n \\
& + F_\varepsilon(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon \\
& + \frac{1}{2} F_{u^{(k)}u^{(k)}}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 ((u_c)_n^{(k)})^2 + \dots \\
& + \frac{1}{2} F_{u'u'}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 ((u_c)'_n)^2 \\
& + \frac{1}{2} F_{uu}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)_n^2 \\
& + \frac{1}{2} F_{u_\tau u_\tau}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 ((u_\tau)_c)_n^2 \\
& + \frac{1}{2} \varepsilon^2 F_{\varepsilon\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \\
& + F_{u^{(k)}u^{(k-1)}}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} (u_c)_n^{(k-1)} + \dots \\
& + F_{u^{(k)}u}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} (u_c)_n \\
& + F_{u^{(k)}u_\tau}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} ((u_\tau)_c)_n \\
& + F_{u^{(k)}\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} + \dots \\
& + F_{u'u}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)'_n (u_c)_n \\
& + F_{u'u_\tau}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)'_n ((u_\tau)_c)_n \\
& + F_{u'\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)'_n \\
& + F_{uu_\tau}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)_n ((u_\tau)_c)_n \\
& + F_{u\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 (u_c)_n \\
& + F_{u_\tau\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\tau)_n, 0, t) \varepsilon^2 ((u_\tau)_c)_n = 0
\end{aligned} \tag{3.8}$$

olarak elde edilir. Tüm türevler  $\varepsilon = 0$  da bulunmuştur ve (3.8) denklemini aşağıdaki gibi daha sade şekilde yazılabilir.

$$\begin{aligned}
& (\varepsilon F_{u^{(k)}} + \varepsilon^2 F_{u^{(k)}\varepsilon})(u_c)_n^{(k)} + \dots + (\varepsilon F_{u'} + \varepsilon^2 F_{u'\varepsilon})(u_c)'_n \\
& + (\varepsilon F_u + \varepsilon^2 F_{u\varepsilon})(u_c)_n + (\varepsilon F_{u_\tau} + \varepsilon^2 F_{u_\tau\varepsilon})((u_\tau)_c)_n \\
& + \frac{1}{2} \varepsilon^2 F_{u^{(k)}u^{(k)}} \left( (u_c)_n^{(k)} \right)^2 + \dots + \frac{1}{2} \varepsilon^2 F_{u'u'} \left( (u_c)'_n \right)^2 \\
& + \frac{1}{2} \varepsilon^2 F_{uu} (u_c)_n^2 + \frac{1}{2} \varepsilon^2 F_{u_\tau u_\tau} \left( (u_\tau)_c \right)_n^2 \\
& + \varepsilon^2 F_{u^{(k)}u^{(k-1)}} (u_c)_n^{(k)} (u_c)_n^{(k-1)} + \dots \\
& + \varepsilon^2 F_{u^{(k)}u} (u_c)_n^{(k)} (u_c)_n \\
& + \varepsilon^2 F_{u^{(k)}u_\tau} (u_c)_n^{(k)} \left( (u_\tau)_c \right)_n + \dots \\
& + \varepsilon^2 F_{u'u} (u_c)'_n (u_c)_n + \varepsilon^2 F_{u'u_\tau} (u_c)'_n \left( (u_\tau)_c \right)_n \\
& + \varepsilon^2 F_{uu_\tau} (u_c)_n \left( (u_\tau)_c \right)_n = -F - \varepsilon F_\varepsilon - \frac{1}{2} \varepsilon^2 F_{\varepsilon\varepsilon}
\end{aligned} \tag{3.9}$$

(3.2) denkleminin PIA(1,2) çözümü için (3.3)'de verilen koşulları sağlayan en düşük mertebeden polinom başlangıç fonksiyonu  $u_0$  olarak tanımlanırsa, (3.9) algoritması kullanılarak  $(u_c)_0$  düzeltme terimi bulunur. Bulunan düzeltme terimi ve  $u_0$  çözümü (3.4) yaya açılımında yazılarak  $u_1$  birinci iterasyon çözümü bulunur. Benzer şekilde, iterasyona bağlı (3.4) direk açılımı ve (3.9) algoritması ile bir önceki iterasyonda elde edilen PIA(1,2) çözümü kullanılarak problemin  $n$ . iterasyon çözümü  $u_n$  bulunur.

### 3.2. Pantograf Tipi Gecikmeli Diferansiyel Denklemler için Perturbasyon-İterasyon Metodu

$0 < \lambda < 1$  olmak üzere  $u(\lambda t)$  terimini içeren  $k$ . mertebe pantograf tipi gecikmeli diferansiyel denklemin kapalı formdaki en genel hali aşağıda verilmiştir.

$$F(u^{(k)}, \dots, u', u, u_\lambda, \varepsilon, t) = 0, \quad a \leq t \leq b \tag{3.10}$$

Burada  $u = u(t)$ ,  $u_\lambda = u(\lambda t)$  ve  $\varepsilon$  perturbasyon parametresidir. (3.10) denklemi, (2.3) ve (2.4) denklemleri gibi birçok araştırmacı tarafından ele alınan problemlerin tamamını kapsadığından dolayı (3.10) denklemi için algoritmalar geliştirilecektir.

(3.10)'da verilen denklemin karışık koşulları şöyledir.

$$\begin{aligned} (\alpha_1)_i u^{(i)}(a) + (\alpha_2)_i u^{(i)}(\beta) + (\alpha_3)_i u^{(i)}(b) &= \gamma_i, \\ i &= 0, 1, 2, \dots, k-1 \end{aligned} \quad (3.11)$$

Burada  $\forall i$  için  $(\alpha_1)_i, (\alpha_2)_i, (\alpha_3)_i \in \mathbb{R}$  ve  $\beta \in \mathbb{R}$  öyleki  $a < \beta < b$  dir.

### 3.2.1. PIA(1,1) Algoritması

(3.4) denklemi (3.10) denkleminde yerine konur ve birinci mertbe türeve göre Taylor serisine açılırsa

$$\begin{aligned} &F(u_n^{(k)}, \dots, u_n', u_n, (u_\lambda)_n, 0, t) \\ &+ F_{u^{(k)}}(u_n^{(k)}, \dots, u_n', u_n, (u_\lambda)_n, 0, t) \varepsilon (u_c)_n^{(k)} + \dots \\ &+ F_{u'}(u_n^{(k)}, \dots, u_n', u_n, (u_\lambda)_n, 0, t) \varepsilon (u_c)'_n \\ &+ F_u(u_n^{(k)}, \dots, u_n', u_n, (u_\lambda)_n, 0, t) \varepsilon (u_c)_n \\ &+ F_{u_\lambda}(u_n^{(k)}, \dots, u_n', u_n, (u_\lambda)_n, 0, t) \varepsilon ((u_\lambda)_c)_n \\ &+ F_\varepsilon(u_n^{(k)}, \dots, u_n', u_n, (u_\lambda)_n, 0, t) \varepsilon = 0 \end{aligned} \quad (3.12)$$

denklemini elde edilir ve

$$F_{u^{(k)}} = \frac{\partial F}{\partial u^{(k)}}, \dots, F_{u'} = \frac{\partial F}{\partial u'}, F_u = \frac{\partial F}{\partial u}, F_{u_\lambda} = \frac{\partial F}{\partial u_\lambda}, F_\varepsilon = \frac{\partial F}{\partial \varepsilon} \quad (3.13)$$

olarak elde edilir. Tüm türevler  $\varepsilon = 0$  da bulunmuştur ve (3.12) denklemi aşağıdaki gibi daha sade formda yazılabilir.

$$\begin{aligned} &F_{u^{(k)}}(u_c)_n^{(k)} + \dots + F_{u'}(u_c)'_n + F_u(u_c)_n + F_{u_\lambda}((u_\lambda)_c)_n \\ &= -F_\varepsilon - \frac{F}{\varepsilon} \end{aligned} \quad (3.14)$$



(3.10) denkleminin PIA(1,1) çözümü için (3.11)'de verilen koşulları sağlayan en düşük mertebeden polinom başlangıç fonksiyonu  $u_0$  olarak tanımlanırsa (3.14) algoritması kullanılarak  $(u_c)_0$  düzeltme terimi bulunur. Bulunan düzeltme terimi ve  $u_0$  çözümü (3.4) yaya açılımında yazılarak  $u_1$  birinci iterasyon çözümü bulunur. Benzer şekilde, iterasyona bağlı (3.4) direk açılımı ve (3.14) algoritması ile bir önceki iterasyonda elde edilen PIA(1,1) çözümü kullanılarak problemin  $n$ . iterasyon çözümü  $u_n$  bulunur.

### 3.2.2. PIA(1,2) Algoritması

(3.4) açılımı (3.10) denkleminde yerine konur ve ikinci mertebe türeve göre Taylor serisine açılırsa



$$\begin{aligned}
& F(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \\
& + F_{u^{(k)}}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon (u_c)_n^{(k)} + \dots \\
& + F_{u'}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon (u_c)'_n \\
& + F_u(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon (u_c)_n \\
& + F_{u_\lambda}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon ((u_\lambda)_c)_n \\
& + F_\varepsilon(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon \\
& + \frac{1}{2} F_{u^{(k)}u^{(k)}}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 ((u_c)_n^{(k)})^2 + \dots \\
& + \frac{1}{2} F_{u'u'}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 ((u_c)'_n)^2 \\
& + \frac{1}{2} F_{uu}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)_n^2 \\
& + \frac{1}{2} F_{u_\lambda u_\lambda}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 ((u_\lambda)_c)_n^2 \\
& + \frac{1}{2} \varepsilon^2 F_{\varepsilon\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \\
& + F_{u^{(k)}u^{(k-1)}}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} (u_c)_n^{(k-1)} + \dots \\
& + F_{u^{(k)}u}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} (u_c)_n \\
& + F_{u^{(k)}u_\lambda}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} ((u_\lambda)_c)_n \\
& + F_{u^{(k)}\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)_n^{(k)} + \dots \\
& + F_{u'u}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)'_n (u_c)_n \\
& + F_{u'u_\lambda}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)'_n ((u_\lambda)_c)_n \\
& + F_{u'\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)'_n \\
& + F_{uu_\lambda}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)_n ((u_\lambda)_c)_n \\
& + F_{u\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 (u_c)_n \\
& + F_{u_\lambda\varepsilon}(u_n^{(k)}, \dots, u'_n, u_n, (u_\lambda)_n, 0, t) \varepsilon^2 ((u_\lambda)_c)_n = 0
\end{aligned} \tag{3.15}$$

olarak elde edilir. Tüm türevler  $\varepsilon = 0$  da bulunmuştur ve (3.15) denklemi aşağıdaki formda yazılabilir.

$$\begin{aligned}
& (\varepsilon F_{u^{(k)}} + \varepsilon^2 F_{u^{(k)}\varepsilon})(u_c)_n^{(k)} + \dots + (\varepsilon F_{u'} + \varepsilon^2 F_{u'\varepsilon})(u_c)_n' \\
& + (\varepsilon F_u + \varepsilon^2 F_{u\varepsilon})(u_c)_n + (\varepsilon F_{u_\lambda} + \varepsilon^2 F_{u_\lambda\varepsilon})((u_\lambda)_c)_n \\
& + \frac{1}{2} \varepsilon^2 F_{u^{(k)}u^{(k)}} \left( (u_c)_n^{(k)} \right)^2 + \dots + \frac{1}{2} \varepsilon^2 F_{u'u'} \left( (u_c)_n' \right)^2 \\
& + \frac{1}{2} \varepsilon^2 F_{uu} (u_c)_n^2 + \frac{1}{2} \varepsilon^2 F_{u_\lambda u_\lambda} \left( (u_\lambda)_c \right)_n^2 \\
& + \varepsilon^2 F_{u^{(k)}u^{(k-1)}} (u_c)_n^{(k)} (u_c)_n^{(k-1)} + \dots \\
& + \varepsilon^2 F_{u^{(k)}u} (u_c)_n^{(k)} (u_c)_n \\
& + \varepsilon^2 F_{u^{(k)}u_\lambda} (u_c)_n^{(k)} \left( (u_\lambda)_c \right)_n + \dots \\
& + \varepsilon^2 F_{u'u} (u_c)_n' (u_c)_n + \varepsilon^2 F_{u'u_\lambda} (u_c)_n' \left( (u_\lambda)_c \right)_n \\
& + \varepsilon^2 F_{uu_\lambda} (u_c)_n \left( (u_\lambda)_c \right)_n = -F - \varepsilon F_\varepsilon - \frac{1}{2} \varepsilon^2 F_{\varepsilon\varepsilon}
\end{aligned} \tag{3.16}$$

(3.10) denkleminin PIA(1,2) çözümü için (3.11)'de verilen koşulları sağlayan en düşük mertebeden polinom başlangıç fonksiyonu  $u_0$  olarak tanımlanırsa, (3.16) algoritması kullanılarak  $(u_c)_0$  düzeltme terimi bulunur. Bulunan düzeltme terimi ve  $u_0$  çözümü (3.4) yaya açılımında yazılarak  $u_1$  birinci iterasyon çözümü bulunur. Benzer şekilde, iterasyona bağlı (3.4) direk açılımı ve (3.16) algoritması ile bir önceki iterasyon da elde edilen PIA(1,2) çözümü kullanılarak problemin  $n$ . iterasyon çözümü  $u_n$  bulunur.

### 3.3. Geçmiş Fonksiyonuna Dayalı Gecikmeli Diferansiyel Denklemler için Perturbasyon-İterasyon Metodu

$\tau \in \mathbb{R}^+$  olmak üzere  $u(t - \tau)$  gecikme terimi veya türevlerinden en az birini içeren ve  $0 < \lambda < 1$  olmak üzere  $u(\lambda t)$  terimi veya türevlerini de içeren  $k$ . mertebe gecikmeli diferansiyel denklemin kapalı formdaki en genel hali aşağıda verilmiştir.

$$F(u^{(k)}, \dots, u', u, u_\tau^{(k)}, \dots, u_\tau', u_\tau, u_\lambda^{(k)}, \dots, u_\lambda', u_\lambda, \varepsilon, t) = 0 \tag{3.17}$$

burada  $0 \leq t \leq b$ ,  $u = u(t)$ ,  $u_\tau = u(t - \tau)$ ,  $u_\lambda = u(\lambda t)$  ve  $\varepsilon$  perturbasyon parametresidir.

$\forall i$  için  $(\alpha_1)_i, (\alpha_2)_i, (\alpha_3)_i \in \mathbb{R}$  ve  $\beta \in \mathbb{R}$  öyleki  $a < \beta < b$  ve  $h(t)$  fonksiyonu  $[-\tau, 0]$  aralığında tanımlı olmak üzere (3.17)'de verilen denkleminin koşulları

$$\begin{aligned} (\alpha_1)_i u^{(i)}(0) + (\alpha_2)_i u^{(i)}(\beta) + (\alpha_3)_i u^{(i)}(b) &= \gamma_i, \\ i &= 1, \dots, k-1 \end{aligned} \quad (3.18)$$

ve

$$u(t) = h(t), \quad -\tau \leq t \leq 0 \quad (3.19)$$

olarak tanımlanmıştır.

(3.17)'de ele alınan  $k$ . mertebe  $u(t - \tau)$  gecikme terimini veya türevlerinden en az birini içeren gecikmeli diferansiyel denklemin  $[0, b]$  tanım aralığındaki çözümü için (3.19) da verilen problemin başlangıç zamanından  $\tau$  gecikme zamanı kadar önceki çözümü problemin geçmiş fonksiyonu (history function) olarak tanımlanmaktadır.  $h(t)$  geçmiş fonksiyonu  $-\tau \leq t \leq 0$  aralığında tanımlıdır.

(3.17) denkleminin  $[0, \tau]$  alt aralığının çözümünde  $u(t - \tau)$  gecikme terimi problemin geçmiş fonksiyonu  $h(t)$  ile hesaplanır.  $[0, \tau]$  için  $u(t - \tau) = h(t - \tau)$  dir. Aynı alt aralıkta problemin gecikme teriminin herhangi bir mertebeden türevi ise geçmiş fonksiyonun aynı mertebeden türevine eşit olacaktır. Problemde geçmiş fonksiyon veya herhangi türevi problemdeki ilgili gecikme teriminde yerine konur ve geliştirilecek PIA algoritması ile problemin  $[0, \tau]$  aralığındaki çözümü bulunur. Problemin tanım aralığı  $[0, b]$ 'nin  $t \geq \tau$  olmak üzere her bir alt aralığındaki PIA çözümü için geçmiş fonksiyonu, bir önceki alt aralıktaki PIA çözümüdür. Elde edilecek tüm alt aralıklardaki çözümler birleştirilirse problemin tanım aralığındaki çözüm bulunur.

$b \in \mathbb{R}^+$  olmak üzere, problemin  $[0, b]$  tanım aralığı

$$J = \left\lceil \frac{b}{\tau} \right\rceil \quad (3.20)$$

alt aralığa bölünecektir. Burada,  $\lceil \cdot \rceil$  tavan (ceiling) fonksiyonudur. Tavan fonksiyonu  $x \in \mathbb{R}$  olmak üzere  $\lceil x \rceil = \begin{cases} x & , x \in \mathbb{Z} \\ \exists \tilde{x} \in \mathbb{Z} \ni x < \tilde{x} < x + 1 & , x \notin \mathbb{Z} \end{cases}$  olarak tanımlanmaktadır.

Problemin tanım aralığı  $D = [0, b]$  ve  $j = 1, 2, 3, \dots, J$  olmak üzere problemin  $j$ . alt aralığını  $D_j$  olarak tanımlayalım. Bu durumda problemin tanım aralığı  $D$ ,  $D_j$  alt aralıklarının birleşimi olarak aşağıdaki gibi ifade edilir.

$$D = \bigcup_{j=1}^J D_j = \begin{cases} \bigcup_{j=1}^J [(j-1)\tau, j\tau] & , b = J\tau \\ \bigcup_{j=1}^{J-1} [(j-1)\tau, i\tau] \cup [(J-1)\tau, b] & , b \neq J\tau \end{cases} \quad (3.21)$$

(3.17) geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denkleminin yukarıda tanımlanan her bir  $D_j = [(j-1)\tau, j\tau]$  alt aralığında ayrı ayrı çözümleri bulunacaktır.  $j = 1, 2, \dots, J$  olmak üzere (3.17) denkleminin bilinmeyen bağımlı değişkeni  $u(t)$ ,  $D_j$  alt aralığında  $u^{D_j}$  olarak tanımlanacaktır.

(3.17) denklemi (3.21)' de tanımlanan her bir alt aralıktaki çözümünü için

$$\begin{aligned} F^{D_1} \left( (u^{D_1})^{(k)}, \dots, (u^{D_1})', u^{D_1}, (u_\lambda^{D_1})^{(k)}, \dots, (u_\lambda^{D_1})', u_\lambda^{D_1} \varepsilon, t \right) &= 0, \quad t \in D_1 \\ F^{D_2} \left( (u^{D_2})^{(k)}, \dots, (u^{D_2})', u^{D_2}, (u_\lambda^{D_2})^{(k)}, \dots, (u_\lambda^{D_2})', u_\lambda^{D_2} \varepsilon, t \right) &= 0, \quad t \in D_2 \\ \vdots \\ F^{D_J} \left( (u^{D_J})^{(k)}, \dots, (u^{D_J})', u^{D_J}, (u_\lambda^{D_J})^{(k)}, \dots, (u_\lambda^{D_J})', u_\lambda^{D_J} \varepsilon, t \right) &= 0, \quad t \in D_J \end{aligned} \quad (3.22)$$

denkleme sistemine dönüştürülür. (3.17) denkleminde gecikme terimleri;  $D_1$  alt aralığında problemin bilinen geçmiş fonksiyonuna ( $h(t)$ ) ve  $j > 1$  için  $D_j$  alt aralığında ise  $D_{j-1}$  alt aralığının  $n$ . iterasyon çözümüne ( $u_n^{D_{j-1}}(t)$ ) eşit olacaktır. Bu yüzden (3.17) denkleminin  $u_\tau^{(k)}, \dots, u_\tau'$  ve  $u_\tau$  bağımsız değişkenleri her bir aralıkta

$t$ 'ye bağılı bilinen fonksiyonlardır. Dolayısıyla (3.22) denklem sisteminin her birinde gecikme terimleri  $t$  bağımsız değişkeninin bir fonksiyonu olarak alınmıştır.

(3.17) geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denkleminin (3.22)'de verilen  $D_j$  alt aralığındaki  $F^{D_j}$  denkleminin  $n$ . iterasyon çözümü  $u_n^{D_j}$  olmak üzere, problemin  $n$ . iterasyon çözümü

$$u_n(t) = \begin{cases} u_n^{D_1}(t) & , & t \in D_1 \\ u_n^{D_2}(t) & , & t \in D_2 \\ \vdots & \vdots & \vdots \\ u_n^{D_J}(t) & , & t \in D_J \end{cases} \quad (3.23)$$

olacaktır.

### 3.3.1. PIA(1,1) Algoritması

(3.17) geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklemin (3.22)'de verilen  $D_j$  alt aralığındaki  $F^{D_j}$  denkleminin  $n$ . iterasyon çözümü  $u_n^{D_j}$  için bir düzeltme terimli direk açılımı her bir iterasyon için uygulanırsa aşağıda verilen eşitlik elde edilir.

$$(u^{D_j})_{n+1} = (u^{D_j})_n + \varepsilon (u_c^{D_j})_n, \quad j = 1, 2, \dots, J \quad (3.24)$$

Burada  $(u_c^{D_j})_n$  perturbasyon-iterasyon algoritmasının  $n$ . düzeltme terimidir. Daha sonra (3.24) denklemi  $F^{D_j}$  denkleminde yerine konur ve birinci merteye türeve göre Taylor serisine açılırsa

$$\begin{aligned}
& F^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \\
& + F_{(u^{Dj})^{(k)}}^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \varepsilon \left( (u_c^{Dj})_n \right)^{(k)} \\
& + \dots \\
& + F_{(u^{Dj})'}^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \varepsilon \left( (u_c^{Dj})_n \right)' \\
& + F_{u^{Dj}}^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \varepsilon (u_c)_n^{Dj} + \\
& F_{(u_\lambda^{Dj})^{(k)}}^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \varepsilon \left( (u_\lambda^{Dj})_c \right)_n^{(k)} \quad (3.25) \\
& + \dots \\
& + F_{(u_\lambda^{Dj})'}^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \varepsilon \left( (u_\lambda^{Dj})_c \right)_n' \\
& + F_{u_\lambda^{Dj}}^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \varepsilon \left( (u_\lambda^{Dj})_c \right)_n \\
& + F_\varepsilon^{Dj} \left( (u^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, (u_\lambda^{Dj})_n^{(k)}, \dots, (u^{Dj})_n, 0, t \right) \varepsilon = 0
\end{aligned}$$

denklemini elde edilir.

(3.25)'deki terimler

$$\begin{aligned}
F_{(u^{Dj})^{(k)}}^{Dj} &= \frac{\partial F^{Dj}}{\partial (u^{Dj})^{(k)}}, \dots, F_{(u^{Dj})'}^{Dj} = \frac{\partial F^{Dj}}{\partial (u^{Dj})'}, F_{u^{Dj}}^{Dj} = \frac{\partial F^{Dj}}{\partial u^{Dj}}, \\
F_{(u_\lambda^{Dj})^{(k)}}^{Dj} &= \frac{\partial F^{Dj}}{\partial (u_\lambda^{Dj})^{(k)}}, \dots, F_{(u_\lambda^{Dj})'}^{Dj} = \frac{\partial F^{Dj}}{\partial (u_\lambda^{Dj})'}, F_{u_\lambda^{Dj}}^{Dj} = \frac{\partial F^{Dj}}{\partial u_\lambda^{Dj}}, \quad (3.26) \\
F_\varepsilon^{Dj} &= \frac{\partial F^{Dj}}{\partial \varepsilon}
\end{aligned}$$

şeklinde tanımlıdır. (3.25) kısaca aşağıdaki gibi yazılabilir:

$$\begin{aligned}
& F_{(u^{D_j})}^{D_j (k)} (u_c^{D_j})_n^{(k)} + \dots + F_{(u^{D_j})}'^{D_j} (u_c^{D_j})_n' + F_{u^{D_j}}^{D_j} (u_c^{D_j})_n \\
& + F_{(u_\lambda^{D_j})}^{D_j (k)} ((u_\lambda^{D_j})_c)_n^{(k)} + \dots + F_{(u_\lambda^{D_j})}'^{D_j} ((u_\lambda^{D_j})_c)_n' + F_{u_\lambda^{D_j}}^{D_j} ((u_\lambda^{D_j})_c)_n \quad (3.27) \\
& = -F_\varepsilon^{D_j} - \frac{F^{D_j}}{\varepsilon}
\end{aligned}$$

(3.27) denklemini  $F^{D_j}$  denkleminin PIA(1,1) algoritmasıdır.

$D_1$  alt aralığında tanımlı  $F^{D_1}$  denkleminin PIA(1,1) çözümü için önereceğimiz  $u_0^{D_j}(t)$  başlangıç fonksiyonu için (3.17) probleminin (3.19)'da sunulan geçmiş fonksiyonu  $h(t)$  ve (3.18)'de verilen diğer koşullar kullanılır.  $j > 1$  için  $D_j$  alt aralığında tanımlı  $F^{D_j}$  denkleminin PIA(1,1) çözümü için önereceğimiz  $u_0^{D_j}(t)$  başlangıç fonksiyonu,  $D_{j-1}$  alt aralığının üst sınırı  $(j-1)\tau$  noktası  $D_j$  alt aralığının alt sınırı olduğundan dolayı bir önceki  $D_{j-1}$  alt aralığında elde edilmiş PIA(1,1) çözümü kullanılarak bulunacaktır.

$$\begin{aligned}
u_0^{D_j}((j-1)\tau) &= u_n^{D_{j-1}}((j-1)\tau) \\
(u_0^{D_j})'((j-1)\tau) &= (u_n^{D_{j-1}})'((j-1)\tau) \\
(u_0^{D_j})''((j-1)\tau) &= (u_n^{D_{j-1}})''((j-1)\tau) \\
&\vdots \\
(u_0^{D_j})^{(k-1)}((j-1)\tau) &= (u_n^{D_{j-1}})^{(k-1)}((j-1)\tau)
\end{aligned} \quad (3.28)$$

$D_j$  alt aralığının başlangıç fonksiyonu (3.28)'de verilen  $k$  adet eşitliği sağlayan  $(k-1)$  . mertebeden  $P_{k-1}^{D_j}(t) = a_0 + a_1 t + a_2 t^2 + \dots + a_{k-1} t^{k-1}$  polinomu ile belirlenir.  $P_{k-1}^{D_j}(t)$  polinomunun katsayılarının matrisi  $\mathbf{A}^{D_j}$

$$\mathbf{A}^{D_j} = [a_0 \quad a_1 \quad a_2 \quad \dots \quad a_{k-1}]^T = (\mathbf{M}^{D_j})^{-1} \mathbf{B}^{D_j} \quad (3.29)$$

matris çarpımıyla bulunur. Burada



$$\mathbf{M}^{D_j} = \begin{bmatrix} 1 & (j-1)\tau & ((j-1)\tau)^2 & \cdots & ((j-1)\tau)^{k-1} \\ 0 & 1 & 2(j-1)\tau & \cdots & (k-1)((j-1)\tau)^{k-2} \\ 0 & 0 & 2 & \cdots & (k-1)(k-2)((j-1)\tau)^{k-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & (k-1)! \end{bmatrix} \quad (3.30)$$

ve

$$\mathbf{B}^{D_j} = \begin{bmatrix} u_n^{D_{j-1}}(j-1)\tau \\ (u_n^{D_{j-1}})'(j-1)\tau \\ (u_n^{D_{j-1}})''(j-1)\tau \\ \vdots \\ (u_n^{D_{j-1}})^{(k-1)}(j-1)\tau \end{bmatrix} \quad (3.31)$$

şeklindedir.

$u_0^{D_j}(t) = P_{k-1}^{D_j}(t)$  olarak belirlenen başlangıç fonksiyonunun (3.27) algoritması kullanılarak  $(u_c)_0^{D_j}$  düzeltme terimi bulunur. Bulunan  $(u_c)_0^{D_j}$  düzeltme terimi ve  $u_0^{D_j}$  başlangıç fonksiyonu (3.24) yaya açılımında yazılarak  $u_1^{D_j}$  birinci iterasyon çözümü bulunur. Benzer şekilde, iterasyona bağlı (3.24) yaya açılımı ve (3.27) algoritması ile bir önceki iterasyonda elde edilen PIA(1,1) çözümü kullanılarak problemin  $n$ . iterasyon çözümü  $u_n^{D_j}$  bulunur.

#### 4. PERTURBASYON-İTERASYON METODUNUN ETKİNLİĞİNİN SAYISAL ÖRNEKLERLE DOĞRULANMASI

Bu bölümde, 3. Bölüm’de geliştirilen PIA algoritmaları zaman gecikmeli diferansiyel denklemler için üç farklı sayısal örneğe, pantograf tipi gecikmeli diferansiyel denklemler için altı farklı sayısal örneğe ve geçmiş fonksiyonuna dayalı gecikmeli denklemler için ise üç farklı sayısal örneğe uygulanmıştır. Tüm bu sayısal örnekler, literatürdeki bilinen diğer yöntemlerle karşılaştırılarak yöntemin doğruluğu ve etkinliği test edilmiştir.

PIA algoritmaları ile çözülen tüm problemler MAPLE ve MATLAB sembolik programları yardımıyla çözülmüştür. Elde edilen çözümler grafikler ile gösterilmiş ve çözümlerin analizinde ve diğer yöntemler ile karşılaştırılmasında grafikler ve tablolardan yararlanılmıştır. Ayrıca bulunan çözümlerin doğruluğunu göstermek için hata analizi yapılmıştır.

Ele alınan problemin, tam çözümü  $u(t)$ ,  $n$ . iterasyon PIA çözümü  $u_n(t)$  ve tanımlı olduğu aralık  $[a, b]$  olmak üzere, hata fonksiyonu ve maksimum mutlak hata değeri sırasıyla aşağıdaki gibi tanımlanmıştır.

$$e_n(t) = u(t) - u_n(t) \quad (4.1)$$

$$E_n = \| u(t) - u_n(t) \|_\infty = \max\{| u_n(t) - u(t) |, a \leq t \leq b\} \quad (4.2)$$

(4.1)’de tanımlanan hata fonksiyonun mutlak değeri mutlak hata fonksiyonu olarak adlandırılacaktır.

##### 4.1. Zaman Gecikmeli Diferansiyel Denklemler için Sayısal Örnekler

Bu bölümde (3.2) denklemi için geliştirilen PIA algoritması ile üç farklı sayısal örnek çözülmüştür.

#### 4.1.1. Sayısal Örnek 1

Aşağıdaki dördüncü mertebeden değişken katsayılı gecikmeli diferansiyel denklemin [64]

$$(e^x u'''(t))' + \frac{e^{x-\frac{1}{2}}}{16} u(t-1) = 0, \quad 0 \leq t \leq 1 \quad (4.3)$$

aşağıdaki başlangıç koşullarına göre

$$u(0) = 0, u'(0) = -1/2, u''(0) = 1/4, u'''(0) = -1/8 \quad (4.4)$$

tam çözümü  $u(t) = e^{-\frac{t}{2}}$  dir. (4.3) denklemini (3.2) kapalı formunda aşağıdaki gibi yazalım.

$$F(u^{(iv)}, \dots, u, u_\tau, \varepsilon, t) = u^{(iv)}(t) + u'''(t) + \frac{\varepsilon}{16e^{\frac{1}{2}}} u(t-1) = 0 \quad (4.5)$$

(4.5) denkleminin  $u(t-1)$  gecikmeli teriminin önüne  $\varepsilon$  suni perturbasyon parametresi koyulmuştur. Burada  $\varepsilon = 1$  dir. (4.5) denkleminin PIA(1,1) çözümü için (3.6) daki terimler hesaplanır. (4.5)'in sıfır olmayan terimleri

$$\begin{aligned} F_{u^{(iv)}} &= 1 \\ F_{u'''} &= 1 \\ F_\varepsilon &= \frac{1}{16e^{\frac{1}{2}}} u_n(t-1) \\ F &= u_n^{(iv)}(t) + u_n'''(t) \end{aligned} \quad (4.6)$$

olarak bulunur. Buradan (3.7) denkleminde verilen perturbasyon açılımında  $n$ . düzeltme terimi için aşağıdaki iterasyon formülü elde edilir.

$$(u_c)_n^{(iv)}(t) + (u_c)_n'''(t) = -\frac{1}{16e^{\frac{1}{2}}} u_n(t-1) - u_n^{(iv)}(t) - u_n'''(t) \quad (4.7)$$

(4.4)' de verilen başlangıç koşullarını sağlayan başlangıç fonksiyonu

$$u_0(t) = 1 - \frac{1}{2}t + \frac{1}{8}t^2 - \frac{1}{48}t^3 \quad (4.8)$$

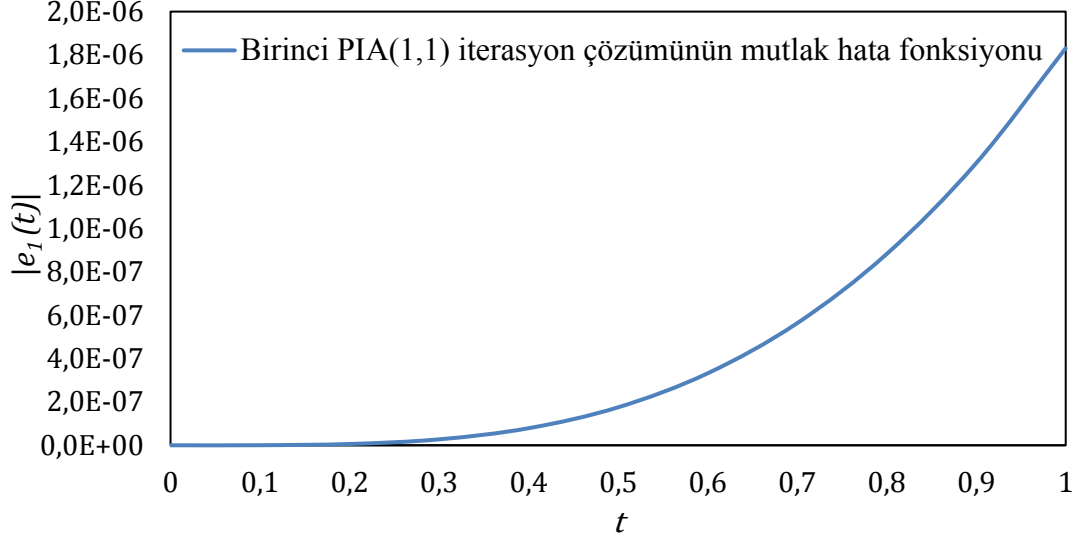
olarak alınır ve (4.7) denkleminde (4.8) başlangıç koşulu yazıldığında  $(u_c)_0(t)$  düzeltme terimi bulunur. Bu terim (3.4) yaya açılımında yazıldığında (4.3) probleminin birinci iterasyon çözümü PIA(1,1) algoritması ile elde edilen birinci iterasyon çözümü

$$\begin{aligned} u_1(t) = & \frac{1}{384} \frac{e^{-t}(-71 + 48e^{\frac{1}{2}})}{e^{\frac{1}{2}}} + \frac{7}{2048} e^{-\frac{1}{2}t^4} - \frac{1}{3840} e^{-\frac{1}{2}t^5} \\ & + \frac{1}{92160} e^{-\frac{1}{2}t^6} - \frac{71}{2304} e^{-\frac{1}{2}t^3} + \frac{t^3}{48} \\ & - \frac{1}{768} \frac{t^2(-71 + 48e^{\frac{1}{2}})}{e^{\frac{1}{2}}} + \frac{1}{384} \frac{t(-71 + 48e^{\frac{1}{2}})}{e^{\frac{1}{2}}} \\ & - \frac{1}{384} \frac{-71 + 48e^{\frac{1}{2}}}{e^{\frac{1}{2}}} \end{aligned} \quad (4.9)$$

olarak bulunur.

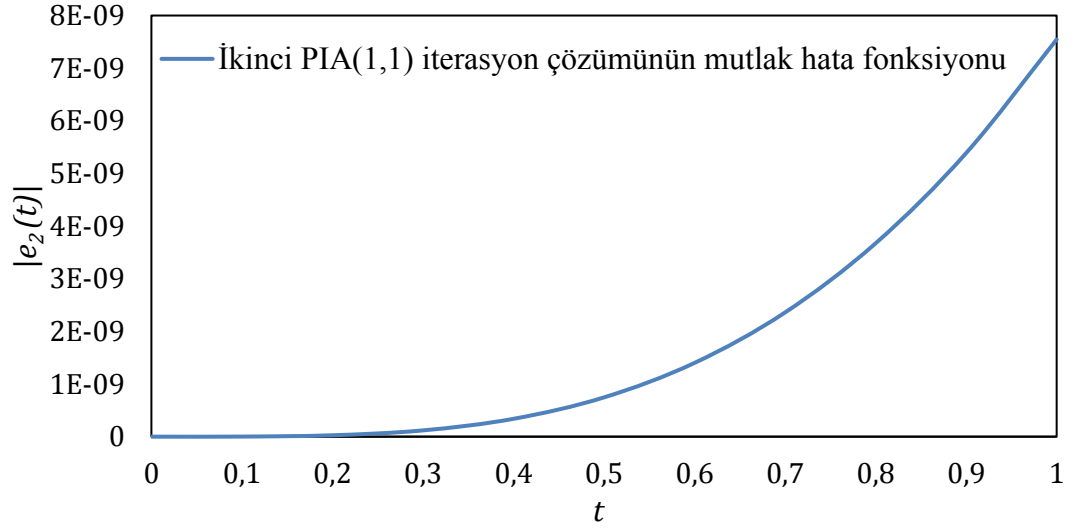
Problemin PIA(1,1) algoritması ile elde edilen birinci iterasyon çözümü  $u_1(t)$  nin (4.2) kullanılarak  $E_1 = 1,82 \times 10^{-6}$  maksimum mutlak hata değeri hesaplanmıştır. (4.9) çözümünün mutlak hata fonksiyonunun grafiği Şekil 4.1.'de verilmiştir.

Problem (4.3)'ün (4.9) birinci iterasyon çözümü ve (4.7) PIA(1,1) algoritması ile ikinci iterasyon çözümü elde edilmiştir. İkinci iterasyon çözümünün elde edilmesinde kullanılan çözüm algoritmasının MATLAB kodları ve bu kodların çıkışı EK A.2'de sunulmuştur.



**Şekil 4.1.** Problem (4.3)'ün PIA(1,1) algoritması ile elde edilmiş birinci iterasyon çözümünün mutlak hata fonksiyonu

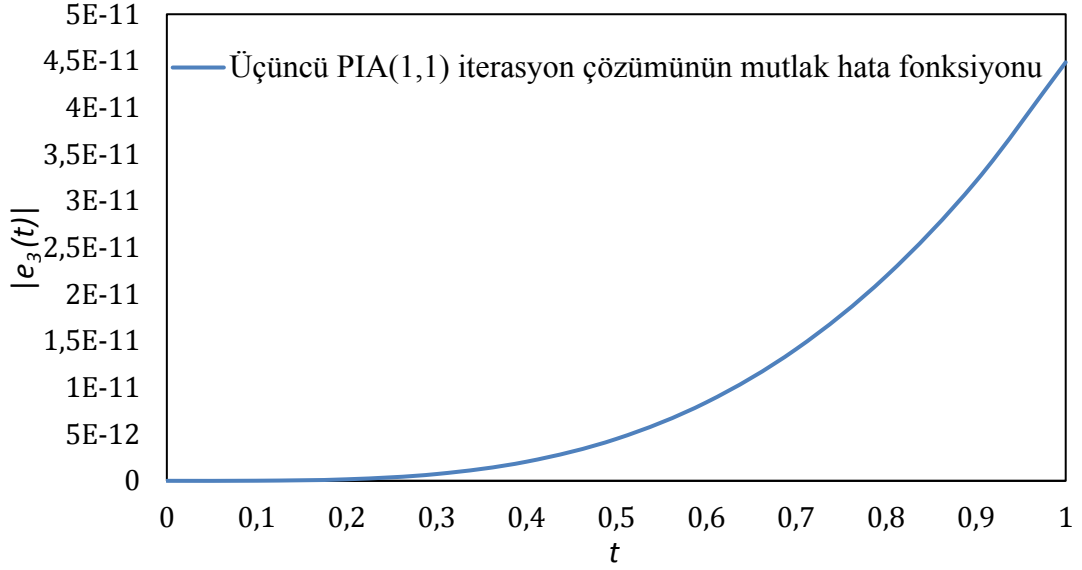
Şekil 4.2.'de verilen  $u_2(t)$  çözümün  $[0,1]$  aralığındaki mutlak hata fonksiyonunun maksimum mutlak hata değeri  $E_2 = 7,54 \times 10^{-9}$  olarak bulunmuştur.



**Şekil 4.2.** Problem (4.3)'ün PIA(1,1) algoritması ile elde edilmiş ikinci iterasyon çözümünün mutlak hata fonksiyonu

Diğer taraftan problemin üçüncü iterasyon çözümü  $u_3(t)$ 'ün mutlak hata fonksiyonu Şekil 4.3.' de verilmiştir.  $u_3(t)$  çözümünün maksimum mutlak hata değeri

ise  $E_3 = 4,48 \times 10^{-11}$  olarak bulunmuştur. İlk üç iterasyon çözümünde maksimum mutlak hata değerinin giderek azaldığı görülmektedir.



**Şekil 4.3.** Problem (4.3)'ün PIA(1,1) algoritması ile elde edilmiş üçüncü iterasyon çözümünün mutlak hata fonksiyonu

İterasyon değeri arttıkça maksimum mutlak hata değerlerinin azaldığı Tablo 4.1.'de açıkça görülmektedir.

**Tablo 4.1.** Problem (4.3)'ün farklı iterasyon değerleri için elde edilen maksimum mutlak hata değerleri

$n$	2	3	4	5	6
$E_n$	$7.51e - 9$	$4.50e - 11$	$3.22e - 13$	$2.43e - 15$	$4.98e - 17$

#### 4.1.2. Sayısal Örnek 2

Aşağıda verilen dördüncü mertebeden değişken katsayılı gecikmeli diferansiyel denklemin

$$(e^t u''''(t))' + 2\sqrt{10}e^{\arcsin \frac{\sqrt{10}}{10}} e^t u \left( t - \arcsin \frac{\sqrt{10}}{10} \right) = 0, \quad (4.10)$$

$$t \geq 0$$

aşağıda verilen başlangıç koşulları altındaki

$$u(0) = 0, \quad u'(0) = 1, \quad u''(0) = 2, \quad u'''(0) = 2 \quad (4.11)$$

tam çözümü  $u(t) = e^t \sin t$  dir [65].

(4.10) denklemi (3.2) kapalı formuna göre düzenlenirse;

$$\begin{aligned} F(u^{(iv)}, \dots, u, u_\tau, \varepsilon, t) \\ &= u^{(iv)}(t) + u'''(t) \\ &+ \varepsilon 2\sqrt{10} e^{\arcsin \frac{\sqrt{10}}{10}} u \left( t - \arcsin \frac{\sqrt{10}}{10} \right) = 0 \end{aligned} \quad (4.12)$$

şeklinde yazılabilir. Burada (4.10) denkleminin  $u \left( t - \arcsin \frac{\sqrt{10}}{10} \right)$  gecikmeli teriminin önüne  $\varepsilon$  suni perturbasyon parametresi konulmuştur. (4.12) denkleminin PIA(1,1) çözümü için (3.6) daki sıfırdan farklı terimleri hesaplanır. Bu terimler

$$\begin{aligned} F_{u^{(iv)}} &= 1 \\ F_{u'''} &= 1 \\ F_\varepsilon &= 2\sqrt{10} e^{\arcsin \frac{\sqrt{10}}{10}} u_n \left( t - \arcsin \frac{\sqrt{10}}{10} \right) \\ F &= u_n^{(iv)}(t) + u_n'''(t) \end{aligned} \quad (4.13)$$

olarak bulunur. Buradan (3.7) denkleminde verilen perturbasyon açılımında  $n$ . düzeltme terimi için aşağıdaki iterasyon formülü elde edilir.

$$\begin{aligned} (u_c)_n^{(iv)}(t) + (u_c)_n'''(t) \\ &= -2\sqrt{10} e^{\arcsin \frac{\sqrt{10}}{10}} u_n \left( t - \arcsin \frac{\sqrt{10}}{10} \right) \\ &- u_n^{(iv)}(t) - u_n'''(t) \end{aligned} \quad (4.14)$$

(4.11)'de verilen başlangıç koşullarını sağlayan en düşük mertebeden polinom başlangıç fonksiyonu

$$u_0(t) = t + t^2 + \frac{1}{3}t^3 \quad (4.15)$$

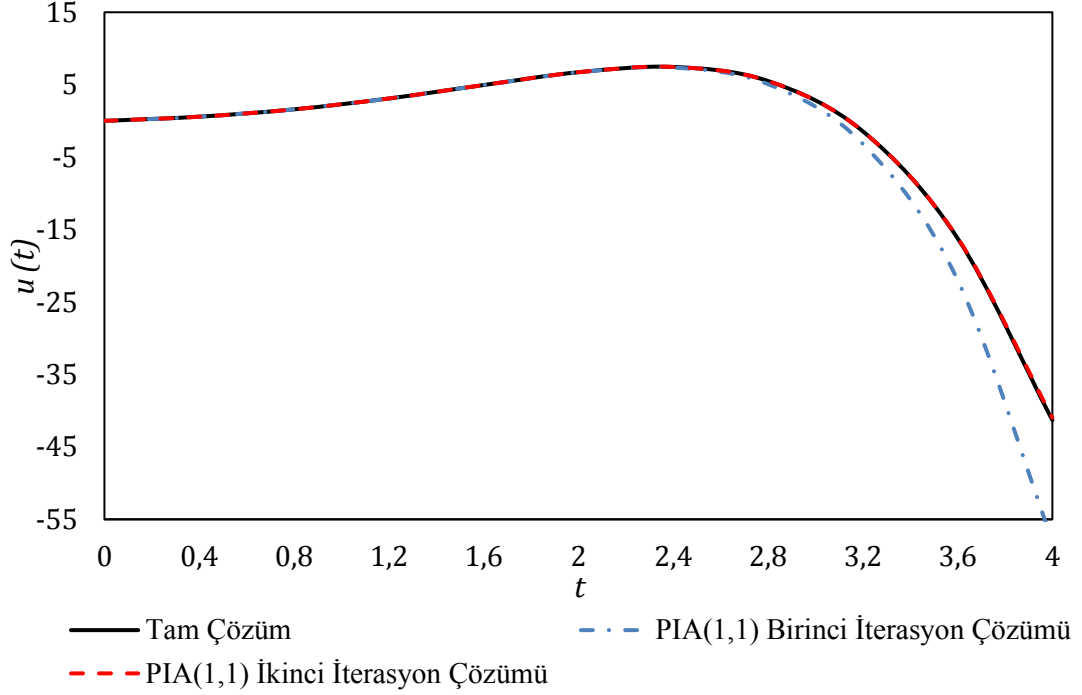
olarak alınır ve (4.10) probleminin PIA(1,1) algoritması (4.14) ile düzeltme terimi elde edilir. Bulunan düzeltme terimi ile (3.4) direk açılımından birinci ve ikinci iterasyon çözümleri

$$\begin{aligned} u_1(t) &= 10.6291210119482t + 9.62912101194816e^{-t} \\ &\quad - 3.81456050597408t^2 + 1.93818683532469t^3 \\ &\quad - 0.401175865815396t^4 + 0.0467877908105845t^5 \\ &\quad - 0.0242360581156431t^6 - 9.62912101194816 \\ u_2(t) &= 1062.24636221263t + 1177.14694532216e^{-t} \\ &\quad + 115.90058310953te^{-t} - 471.67288955155t^2 \\ &\quad - 471.67288955155t^2 + 138.574199332262t^3 \\ &\quad - 29.7310259885565t^4 + 4.94703512031942t^5 \\ &\quad - 0.680202771716038t^6 + 0.0710064083504364t^7 \\ &\quad - 0.00620595247744794t^8 + 0.000419561794989165t^9 \\ &\quad - 1177.14694532216 \end{aligned} \quad (4.16)$$

olarak elde edilir.

Problemin PIA(1,1) algoritması ile elde edilen birinci iterasyon çözümü  $u_1(t)$ , ikinci iterasyon çözümü  $u_2(t)$  ve problemin tam çözümü Şekil 4.4.'de verilmiştir. Şekilde görüldüğü gibi,  $[0,4]$  aralığında  $u_1(t)$  tam çözümden küçük bir sapma gösterirken,  $u_2(t)$  ise tam çözüm ile neredeyse tamamen çakışmıştır.





**Şekil 4.4.** Problem (4.10)'un PIA(1,1) algoritması ile elde edilmiş birinci ve ikinci iterasyon çözümleri ile problemin tam çözümünün karşılaştırılması

Tablo 4.2'de  $n = 2, 3, 4, 5$  ve 6 için PIA(1,1) algoritması ile elde edilen iterasyon çözümlerinin  $0 \leq t \leq 1$  aralığındaki maksimum mutlak hata değerleri verilmiştir. İterasyon değeri  $n$  arttıkça maksimum mutlak hata değeri azalmaktadır.

**Tablo 4.2.** Problem (4.10)'un  $n = 2, 3, 4, 5$  ve 6 için PIA(1,1) algoritması ile elde edilen iterasyon çözümlerinin maksimum mutlak hata değerleri

$n$	2	3	4	5	6
$E_n$	$3.47e - 01$	$9.88e - 04$	$1.53e - 07$	$1.01e - 08$	$4.62e - 09$

### 4.1.3 Sayısal Örnek 3

Aşağıdaki üçüncü mertebeden gecikmeli diferansiyel denklemin

$$u'''(t) = -u(t) - u(t - 0.3) + e^{-t+0.3}, \quad 0 \leq t \leq 1 \quad (4.17)$$

başlangıç koşulları

$$u(0) = 1, \quad u'(0) = -1, \quad u''(0) = 1 \quad (4.18)$$

altındaki tam çözümü  $u(t) = e^{-t}$  dir [66].

(4.17) denklemini (3.2) kapalı formuna göre aşağıdaki gibi düzenlenmiştir.

$$\begin{aligned} F(u^{(iv)}, \dots, u, u_\tau, \varepsilon, t) &= u'''(t) + u(t) - e^{-t+0.3} + \varepsilon u(t - 0.3) \\ &= 0 \end{aligned} \quad (4.19)$$

Ayrıca  $u(t - 0.3)$  gecikmeli teriminin önüne  $\varepsilon$  suni perturbasyon parametresi konulmuştur. Burada  $\varepsilon = 1$  dir. (4.17) denkleminin PIA(1,1) çözümü için (4.19)'un sıfır olmayan terimleri

$$\begin{aligned} F_{u''''} &= 1 \\ F_u &= 1 \\ F_\varepsilon &= u_n(t - 0.3) \\ F &= u_n'''(t) + u_n(t) - e^{-t+0.3} \end{aligned} \quad (4.20)$$

olarak bulunur. Buradan (3.7) perturbasyon açılımında  $n$ . düzeltme terimi için aşağıdaki iterasyon formülü elde edilir.

$$(u_c)_n'''(t) + (u_c)_n(t) = -u_n(t - 0.3) - u_n'''(t) - u_n(t) + e^{-t+0.3} \quad (4.21)$$

(4.18)'de verilen başlangıç koşullarını sağlayan en düşük mertebeden polinom başlangıç fonksiyonu olarak

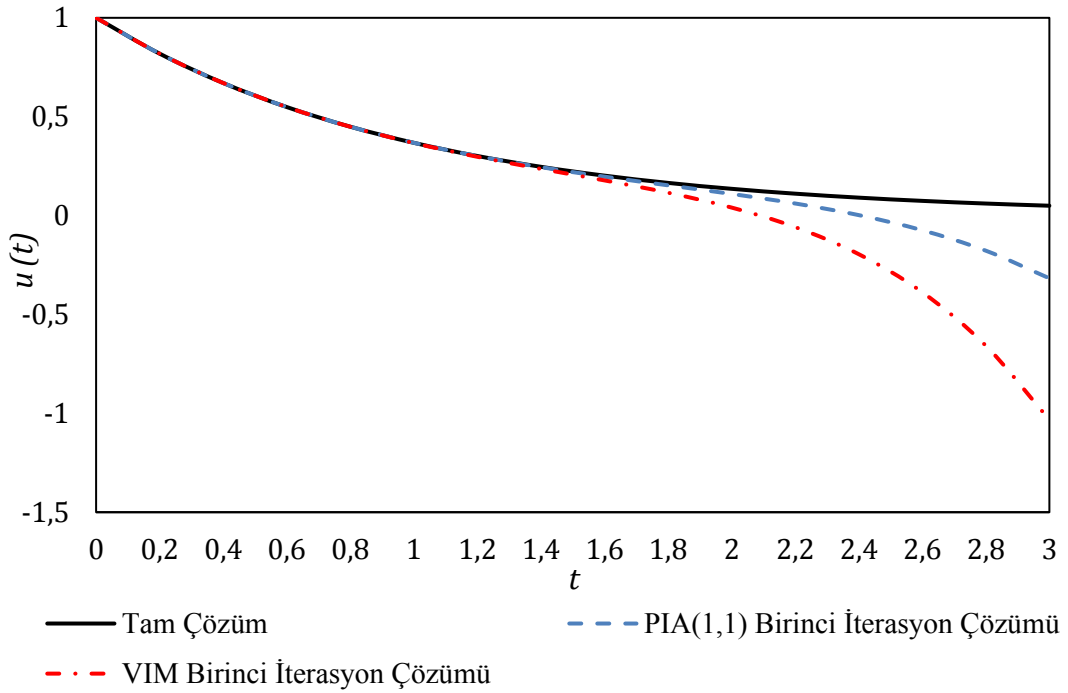
$$u_0(t) = 1 - t + \frac{1}{2}t^2 \quad (4.22)$$

alınırsa (4.21) PIA(1,1) algoritması ve (3.4) yaya açılımı ile elde edilen birinci iterasyon çözümü

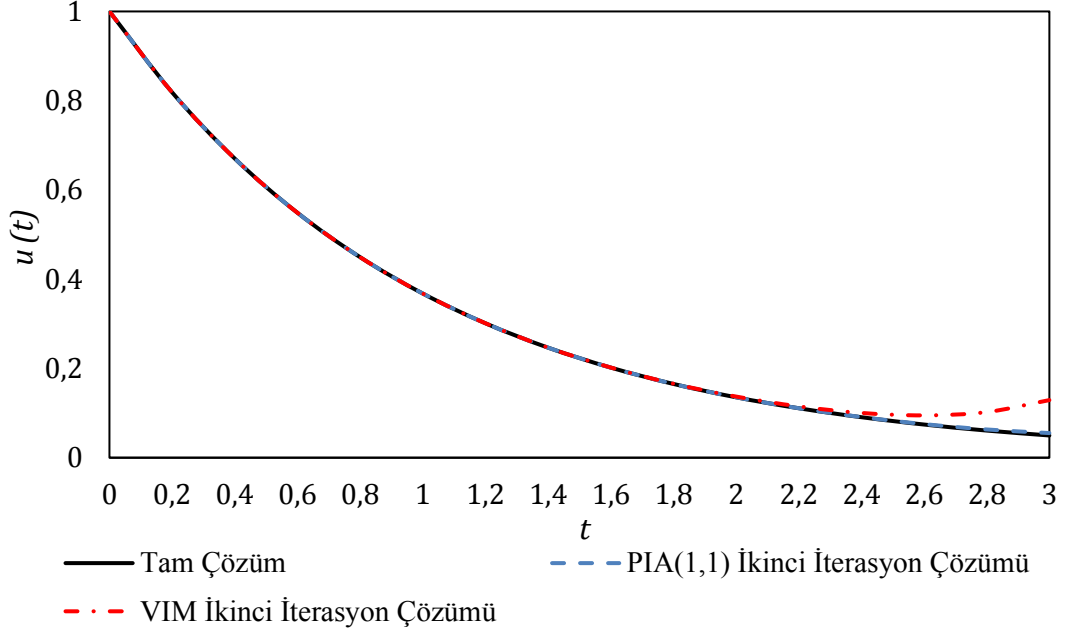
$$\begin{aligned}
u_1(t) = & -\frac{269}{200} + \frac{13}{10}t - \frac{t^2}{2} + \frac{1}{3}e^{-t+\frac{3}{10}} + \frac{1}{3}xe^{-t+\frac{3}{10}} + \frac{443}{200}e^{-t} \\
& + \left(\frac{3}{100} - \frac{1}{3}e^{\frac{3}{10}}\right)e^{\frac{t}{2}}\cos\left(\frac{\sqrt{3}t}{2}\right) \\
& + \frac{1}{90}\left(-9 + 10e^{\frac{3}{10}}\right)\sqrt{3}e^{\frac{t}{2}}\sin\left(\frac{\sqrt{3}}{2}t\right)
\end{aligned} \tag{4.23}$$

olarak elde edilir.

PIA algoritması ile elde edilen sonuçların karşılaştırılmasını yapmak amacıyla bu örnekte düzeltme fonksiyonlarının iterasyon özelliği kullanılarak sayısal yaklaşımlar elde edilmek üzere geliştirilmiş bir iterasyon yöntemi olan Varyasyonel İterasyon Metodu (VIM) çözümleri kullanılmıştır. Şekil 4.5’de PIA(1,1) ve VIM birinci iterasyon çözümleri verilmiştir. Şekil 4.6’da her iki metodun ikinci iterasyon çözümleri verilmiştir. PIA(1,1) algoritmasıyla bulunan ilk iki iterasyon çözümleri ile ilk iki iterasyon VIM [66] çözümleri karşılaştırıldığında, PIA çözümleri VIM çözümlerine göre tam çözüme daha çok yakınsadığı görülmektedir.

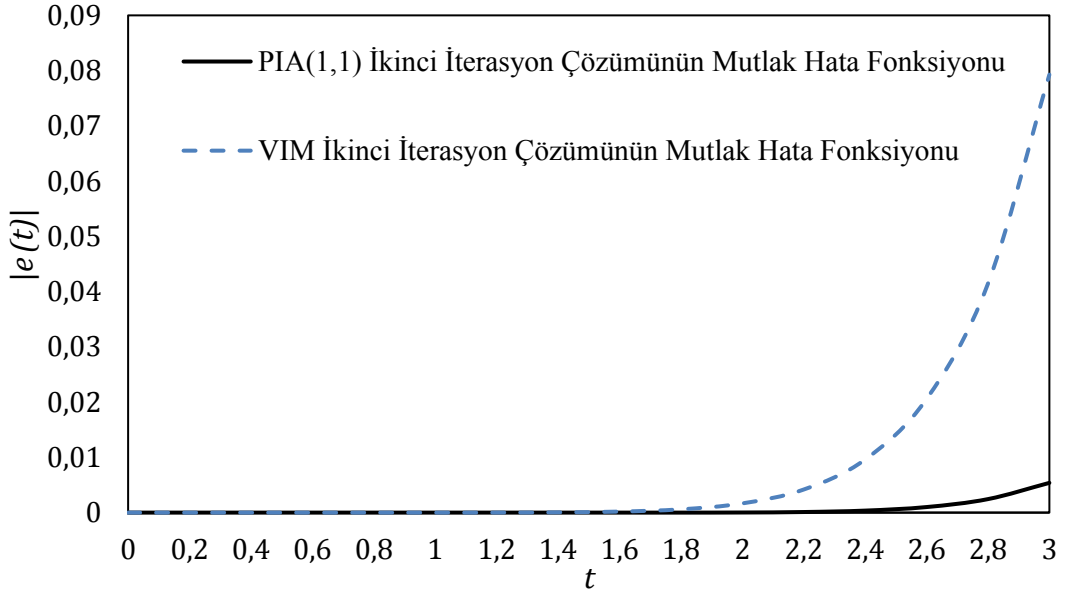


**Şekil 4.5.** Problem (4.17)’nin PIA(1,1) algoritması ve VIM ile elde edilmiş birinci iterasyon çözümlerinin problemin tam çözümünü ile karşılaştırılması



**Şekil 4.6.** Problem (4.17)'nin PIA(1,1) algoritması ve VIM ile elde edilmiş ikinci iterasyon çözümlerinin problemin tam çözümü ile karşılaştırılması

Problem (4.17)'nin PIA(1,1) algoritması ile ve VIM ile elde edilmiş ikinci iterasyon çözümlerinin mutlak hata fonksiyonları Şekil 4.7.'de verilmiştir.



**Şekil 4.7.** Problem (4.17)'nin PIA(1,1) algoritması ile ve VIM ile elde edilmiş ikinci iterasyon çözümlerinin mutlak hata fonksiyonlarının karşılaştırılması

## 4.2. Pantograf Tipi Gecikmeli Diferansiyel Denklemler için Sayısal Örnekler

Bu bölümde, (3.10) denklemi için geliştirilen perturbasyon iterasyon yöntemi ile sayısal örnekler çözülmüştür.

### 4.2.1. Sayısal Örnek 1

Aşağıda verilen birinci mertebeden lineer denklemin

$$u'(t) + u(t) - \frac{1}{10}u\left(\frac{t}{5}\right) = -\frac{1}{10}e^{-\frac{1}{5}t}, \quad 0 \leq t \leq 1 \quad (4.24)$$

başlangıç koşulu

$$u(0) = 1 \quad (4.25)$$

altındaki tam çözümü  $u(t) = e^{-t}$  dir [88].

(4.24) denklemi (3.10) kapalı formunda  $\varepsilon$  suni parametresi ile aşağıdaki gibi yazılabilir.

$$F(u', u, u_\lambda, \varepsilon, t) = u'(t) + u(t) - \varepsilon \frac{1}{10}u\left(\frac{t}{5}\right) + \frac{1}{10}e^{-\frac{1}{5}t} = 0 \quad (4.26)$$

Burada  $\varepsilon = 1$  olarak alınmıştır. (4.26) denkleminin (3.14) de verilen PIA(1,1) algoritmasına göre sıfır olmayan tüm terimleri

$$\begin{aligned} F_{u'} &= 1 \\ F_u &= 1 \\ F_\varepsilon &= -\frac{1}{10}u_n\left(\frac{t}{5}\right) \\ F &= u'_n(t) + u_n(t) + \frac{1}{10}e^{-\frac{1}{5}t} \end{aligned} \quad (4.27)$$

olarak hesaplanır. (4.27) de bulunan tüm terimler (3.14) PIA(1,1) algoritmasında yerlerine yazılırsa

$$(u_c)'_n(t) + (u_c)_n(t) = \frac{1}{10}u_n\left(\frac{t}{5}\right) - u'_n(t) - u_n(t) - \frac{1}{10}e^{-\frac{1}{5}t} \quad (4.28)$$

elde edilir. (4.28) denklemi (4.26)'nın PIA(1,1) algoritmasıdır. (4.25) başlangıç koşulunu sağlayan en düşük mertebeli polinom olan

$$u_0 = 1 \quad (4.29)$$

başlangıç fonksiyonu olarak seçilirse (3.4) açılımı ile (4.28) in çözülmesi ile elde edilen  $n$ . iterasyonun düzeltme terimleri (4.24) ün PIA(1,1) çözümleri aşağıdaki gibi elde edilir.

$$\begin{aligned} u_1 &= \frac{1}{10} - \frac{1}{8}e^{-\frac{1}{5}t} + \frac{41}{40}e^{-t} \\ u_2 &= \frac{1}{10^2} + \frac{1}{320}e^{-\frac{1}{5}t} - \frac{5}{384}e^{-\frac{1}{25}t} + \frac{9599}{9600}e^{-t} \\ u_3 &= \frac{1}{10^3} - \frac{1}{76800}e^{-\frac{1}{5}t} + \frac{1}{3072}e^{-\frac{1}{25}t} - \frac{125}{95232}e^{-\frac{1}{125}t} \\ &\quad + \frac{11904001}{11904000}e^{-t} \\ u_4 &= \frac{1}{10^4} + \frac{1}{95232000}e^{-\frac{1}{5}t} - \frac{1}{737280}e^{-\frac{1}{25}t} + \frac{25}{761825}e^{-\frac{1}{125}t} \\ &\quad - \frac{15625}{118849536}e^{-\frac{1}{625}t} + \frac{74280959999}{74280960000}e^{-t} \\ &\quad \vdots \end{aligned} \quad (4.30)$$

(4.30)'de verilen çözümlerin elde edilmesi için yazılan MATLAB kodu EK A.3'de verilmiştir.

PIA(1,1) yöntemi ile elde edilen çözümlerin mutlak hata değerleri ile diğer sayısal çözümlerin literatürde verilen mutlak hata değerlerinin karşılaştırılması Tablo 4.3' de verilmiştir. Açıkça görülmektedir ki, 4. iterasyon için PIA(1,1) ile elde edilen çözüme ait mutlak hata değerleri diğer yöntemlere göre daha küçüktür. Elde edilen sayısal çözümler problemin tam çözümüne yakınsamıştır. Ayrıca  $n = 4$  için elde edilen çözümler VIM [66] ile elde edilen çözümlerle neredeyse aynıdır.

**Tablo 4.3.** Problem (4.24)'ün PIA(1,1) ve diğer yöntemlerle elde edilen sayısal çözümlerinin mutlak hata değerlerinin karşılaştırılması

$t$	VIM [66] $n = 4$	Taylor	Sürekli	JRC	PIA(1,1)
		Matris metodu [89]	metod [90]	$\alpha = \beta = 0.5$ [88]	
				$N = 64$	$n = 4$
$2^{-1}$	$2.40e - 15$	$1.24e - 10$	$3.33e - 11$	$3.93e - 14$	$1.43e - 15$
$2^{-2}$	$7.91e - 17$	$9.74e - 11$	$4.13e - 11$	$2.16e - 14$	$7.91e - 17$
$2^{-3}$	$2.53e - 18$	$7.00e - 11$	$4.62e - 11$	$1.66e - 15$	$2.53e - 18$
$2^{-4}$	$8.03e - 20$	$9.14e - 11$	$4.89e - 11$	$6.21e - 15$	$8.03e - 20$
$2^{-5}$	$2.52e - 21$	$5.28e - 11$	$5.05e - 11$	$3.68e - 14$	$2.52e - 21$
$2^{-6}$	$7.00e - 23$	$1.95e - 11$	$4.74e - 11$	$4.54e - 14$	$7.92e - 23$

Araştırmacılar tarafından pantograf tipi denklemler için geliştirilen ve literatürde yer almış dört farklı yöntemden elde edilmiş çözümlerin mutlak hata fonksiyonlarının belirli noktalardaki değerleri ile bu noktalardaki PIA(1,1) algoritması çözümlerinin hata değerleri karşılaştırılmıştır. Elde edilen çözümler ve hata değerlerini bulan MATLAB kodu EK A.3'de sunulmuştur.

Tablo 4.3'de bahsedilen yöntemlerin genel olarak yaklaşımlarını özetleyelim. VIM düzeltme fonksiyonlarının iterasyon özelliği kullanılarak çeşitli problemler için sayısal yaklaşımlar elde edilmek üzere geliştirilmiş bir iterasyon yöntemidir. Jacobi-rasyonel Gauss yöntemi (JRC) Jacobi polinomları ve sıralama noktaları kullanılarak geliştirilmiş bir seri çözüm yöntemidir [88]. Taylor Matris Metod Taylor polinomları ve temel matris işlemleri ile problemlerin matris denklemlerine dönüştürülerek problemlerin sayısal çözümlerinin elde edilmesine meşine seri çözüm yöntemidir [89]. Sürekli Metod çekirdek teorisinin fonksiyonel diferansiyel denklemlerde yeterli yaklaşımlar göstermemesiyle yeniden üretilerek geliştirilmiş, çekirdek teorisine dayalı yöntemdir [90].

#### 4.2.2. Sayısal Örnek 2

Aşağıda birinci mertebeli pantograf tipi gecikmeli diferansiyel denklemini ele alalım [69].

$$u'(t) + u(t) + u(0.8t) = 0, \quad 0 \leq t \leq 1 \quad (4.31)$$

Başlangıç koşulu

$$u(0) = 1 \quad (4.32)$$

olan bu denklemin tam çözümü yoktur. (4.31) denklemini (3.10) kapalı formunda  $\varepsilon$  suni parametresi ile aşağıdaki gibi yazılabilir.

$$F(u', u, u_\lambda, \varepsilon, t) = u'(t) + u(t) + \varepsilon u(0.8t) = 0 \quad (4.33)$$

burada  $\varepsilon = 1$  dir. (3.14) denkleminde verilen PIA(1,1) algoritmasının sıfırdan farklı terimleri

$$\begin{aligned} F_{u'} &= 1 \\ F_u &= 1 \\ F_\varepsilon &= u_n(0.8t) \\ F &= u'_n(t) + u_n(t) \end{aligned} \quad (4.34)$$

olarak bulunur ve bu terimler (3.14) denkleminde yerlerine yazılırsa

$$(u_c)'_n(t) + (u_c)_n(t) = -u_n(0.8t) - u'_n(t) - u_n(t) \quad (4.35)$$

denklemini elde edilir. (4.32) başlangıç koşulunu sağlayan

$$u_0 = 1 \quad (4.36)$$

en düşük mertebeli polinom başlangıç fonksiyonu olarak alınır ve (4.35) denkleminde uygulanarak aşağıdaki iterasyon çözümleri elde edilir.



$$\begin{aligned}
u_1 &= -1 + 2e^{-t} \\
u_2 &= 1 + 10e^{-t} - 10e^{-\frac{4}{5}t} \\
u_3 &= -1 + \frac{218}{9}e^{-t} - 50e^{-\frac{4}{5}t} + \frac{250}{9}e^{-\frac{16}{25}t} \\
u_4 &= 1 + \frac{21490}{549}e^{-t} - \frac{1090}{9}e^{-\frac{4}{5}t} + \frac{1250}{9}e^{-\frac{16}{25}t} - \frac{31250}{549}e^{-\frac{64}{125}t} \\
&\vdots
\end{aligned} \tag{4.37}$$

Tablo 4.4.'de (4.31) denkleminin  $n = 5, 6, 7, 8, 9$  iterasyon değerleri için PIA(1,1) çözümleri verilmiştir. İterasyon sayısı arttıkça çözümlerin yakınsadığı görülmektedir.

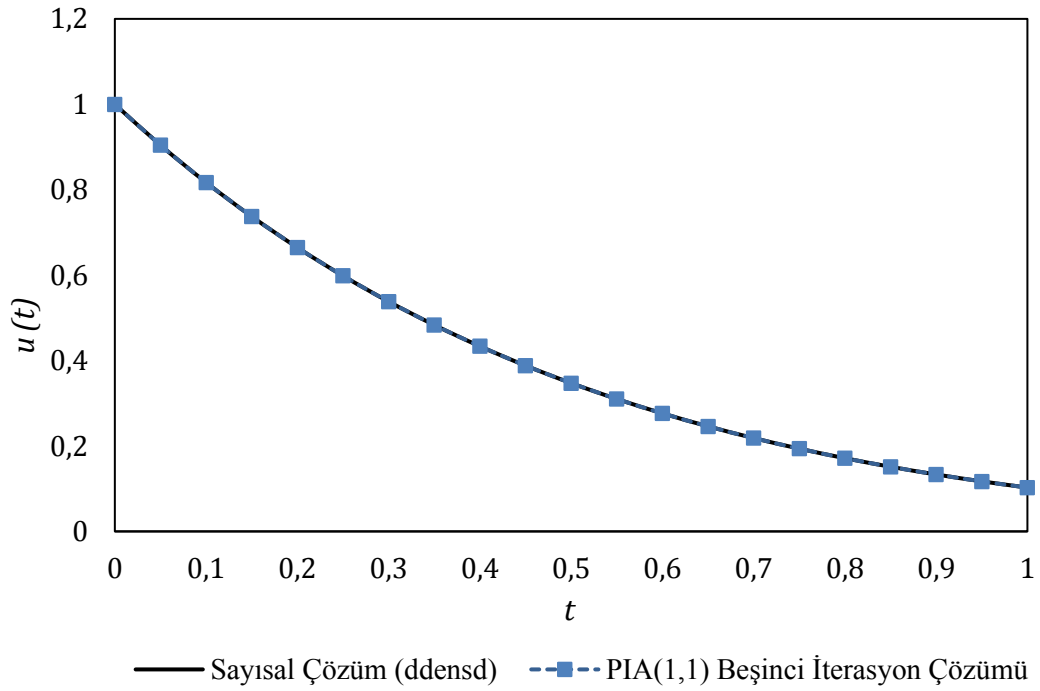
**Tablo 4.4.** Problem (4.31)'in PIA(1,1) çözümlerinin belirli noktalardaki değerlerinin karşılaştırılması

$t_i$	$u_5(t)$	$u_6(t)$	$u_7(t)$	$u_8(t)$	$u_9(t)$
0	1	1	1	1	1
0.2	0.6646909	0.6646910	0.6646910	0.6646910	0.6646910
0.4	0.4335604	0.4335607	0.4335607	0.4335607	0.4335607
0.6	0.2764790	0.2764824	0.2764823	0.2764823	0.2764823
0.8	0.1714677	0.1714846	0.1714841	0.1714841	0.1714841
1	0.1026142	0.1026723	0.1026700	0.1026701	0.1026701

Diğer sayısal yöntemlerin sonuçları ile PIA(1,1) sonuçlarının karşılaştırılması Tablo 4.5.' de verilmiştir. Tablo 4.5.'de sunulan, Hermite [69], Taylor [70] ve Laguerre [91] metotları matris işlemlerine dayalı seri çözüm yöntemleridirler. Bu yöntemlerde sırası ile Taylor, Hermite ve Laguerre polinomları kullanılarak sayısal çözümler elde edilmiştir. Walsh Metodu [92] operasyonel matrisler ve Walsh fonksiyonları kullanılarak geliştirilmiş bir yöntemdir. Direk Adams-Moulton metodu [93] ise birim adım fonksiyonları kullanılarak Walsh Metoduna benzer bir yaklaşımla problemin sayısal çözümünü bulmak için geliştirilmiş bir yöntemdir. Hem farklı polinomların ve fonksiyonların kullanıldığı tüm bu yöntemler ile hem de PIA(1,1) algoritması ile elde edilen sonuçlar eşit sayıda işlem tekrarı ve çözüm mertebeleri esas alınarak karşılaştırılmıştır.

**Tablo 4.5.** Problem (4.31)'in PIA(1,1) çözümünün diğer sayısal yöntemler ile karşılaştırılması

	Walsh metot [92]	DAM [93]	Hermite metot [69]	Taylor metot [70]	Laguerre metot [91]	PIA(1,1)
$t_i$		$m = 100,$ $h = 0.01$	$N = 8$	$N = 8$	$N = 8$	$n = 8$
0	1	1	1	1	1	1
0.2	0.665621	0.664677	0.664691	0.664691	0.6646910	0.6646910
0.4	0.432426	0.433540	0.433561	0.433561	0.4335607	0.4335607
0.6	0.275140	0.276460	0.276482	0.276482	0.2764831	0.2764823
0.8	0.170320	0.171464	0.171484	0.171484	0.1714942	0.1714841
1	0.100856	0.102652	0.102670	0.102744	0.1027437	0.1026701



**Şekil 4.8.** Problem (4.31)'in PIA(1,1) beşinci iterasyon çözümü ile MATLAB ddensd fonksiyonu ile elde edilen sayısal çözümün karşılaştırılması

Shampine [94] neutral tipi gecikmeli diferansiyel denklemler geliştirdiği sayısal yöntemin MATLAB fonksiyonu paket programın kütüphanesine ddensd ismi

ile eklenmiştir. Problem (4.31)'in PIA(1,1) beşinci iterasyon çözümü ile MATLAB'ın ddsnd fonksiyonu ile elde edilen sayısal çözümünün karşılaştırılması Şekil 4.8.'de verilmiştir.

### 4.2.3. Sayısal Örnek 3

Lineer olmayan pantograf tipi gecikmeli diferansiyel denklemin

$$u'(t) = -2u^2\left(\frac{t}{2}\right) + 1, \quad 0 \leq t \quad (4.38)$$

başlangıç koşulu

$$u(0) = 0 \quad (4.39)$$

ile tam çözümü  $u(t) = \sin(t)$  dir [95].

(4.38) denkleminin (3.10) kapalı formunda  $u^2\left(\frac{t}{2}\right)$  pantograf teriminin önündeki katsayı suni perturbasyon parametresi  $\varepsilon = 2$  alınarak aşağıdaki formda yazılmıştır.

$$F(u', u, u_\lambda, \varepsilon, t) = u'(t) + \varepsilon u^2\left(\frac{t}{2}\right) - 1 = 0 \quad (4.40)$$

PIA(1,1) algoritmasında verilen (3.14) denkleminde sıfır olmayan terimler

$$\begin{aligned} F_{u'} &= 1 \\ F_\varepsilon &= u_n^2\left(\frac{t}{2}\right) \\ F &= u'_n(t) - 1 \end{aligned} \quad (4.41)$$

olarak bulunur. (3.14) denkleminde verilen perturbasyon açılımında  $n$ . düzeltme terimi için aşağıdaki iterasyon formülü elde edilir.

$$(u_c')_n(t) = -u_n^2\left(\frac{t}{2}\right) - \frac{u_n'(t) - 1}{2} \quad (4.42)$$

Başlangıç fonksiyonu

$$u_0 = t \quad (4.43)$$

olarak seçilirse, PIA(1,1) çözümleri

$$\begin{aligned} u_1 &= t - \frac{1}{6}t^3 \\ u_2 &= t - \frac{1}{6}t^3 + \frac{1}{120}t^5 - \frac{1}{8064}t^7 \\ u_3 &= t - \frac{1}{6}t^3 + \frac{1}{120}t^5 - \frac{1}{5040}t^7 + \frac{61}{23224320}t^9 - \\ &\quad \frac{67}{3406233600}t^{11} + \frac{1}{12881756160}t^{13} - \frac{1}{7990652436480}t^{15} \\ &\quad \vdots \end{aligned} \quad (4.44)$$

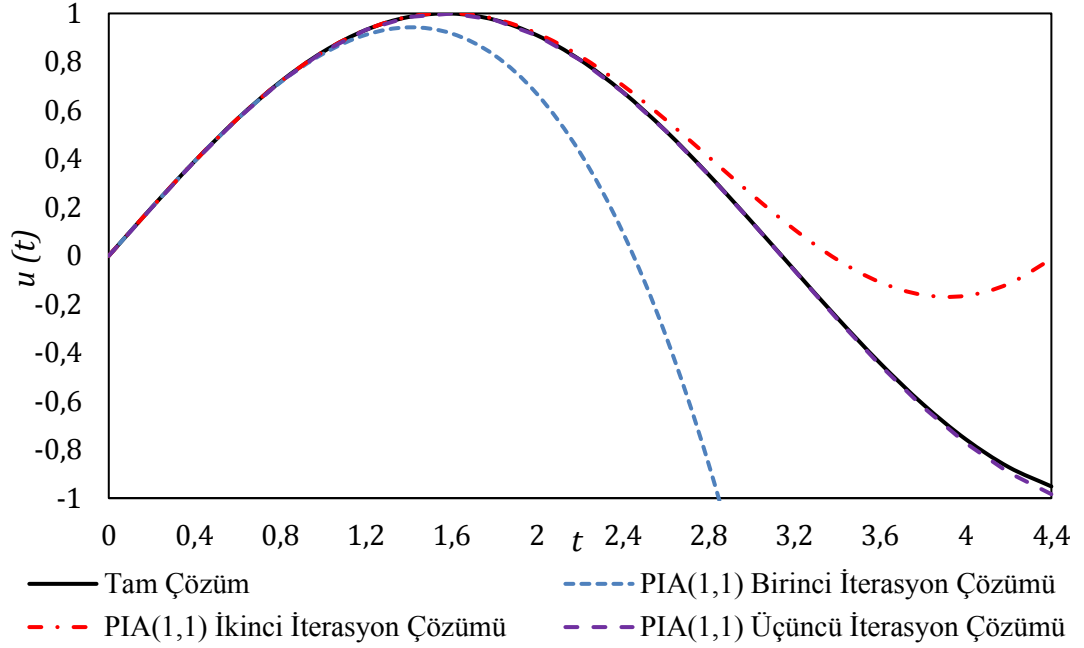
elde edilir. Farklı  $n$  iterasyon değerleri için maksimum mutlak hata değerleri Tablo 4.6.'da verilmiştir. Burada  $n$  iterasyon değeri arttıkça elde edilen PIA çözümlerinin maksimum mutlak hata değerlerinin azaldığı görülmüştür.

**Tablo 4.6.** Problem (4.39)'un ilk altı iterasyon PIA(1,1) çözümlerinin maksimum mutlak hata değerleri

$n$	1	2	3	4	5	6
$E_n$	$8.13e - 3$	$7.16e - 5$	$1.23e - 7$	$4.38e - 11$	$2.91e - 15$	$5.35e - 20$

Problem (4.39) literatürde [95, 96], Homotopi Asimtotik Metot (HAM) ve Optimal Homotopi Asimtotik Metot (OHAM) ile çözülmüştür. HAM yöntemi problemin yakınsama bölgesi ve ayarlanması esasına dayalı iken, OHAM yönteminde benzer bir yaklaşım esas alınsa da HAM yönteminden esneklik derecesi açısından daha iyi bir yöntem olduğu ortaya konmuştur [60]. Problemin HAM ile elde edilen maksimum mutlak hata değeri  $1.2 \times 10^{-6}$  ve OHAM ile elde edilen maksimum mutlak hata değeri  $4 \times 10^{-8}$  iken PIA(1,1) ile elde edilen maksimum mutlak hata değeri  $5.35 \times 10^{-20}$  dir. Ayrıca, Şekil 4.9'da, (4.39)'un PIA(1,1) algoritması ile elde

edilen ilk üç iterasyon çözümleri ( $n = 1, 2, 3$ ) ile problemin tam çözümü verilmiştir. Görüldüğü gibi  $n$  değeri arttıkça PIA çözümü tam çözüme yakınsamaktadır.



**Şekil 4.9.** Problem (4.39)'un PIA(1,1) ile elde edilen ilk üç iterasyon çözümü ile problemin tam çözümünün karşılaştırılması

#### 4.2.4. Sayısal Örnek 4

Aşağıdaki ikinci mertebeden lineer olmayan pantograf tipi gecikmeli diferansiyel denklemin

$$u''(t) = -u\left(\frac{t}{2}\right) - u^2(t) + \sin^4(t) + \sin^2\left(\frac{t}{2}\right) + 8, \quad t \geq 0 \quad (4.45)$$

başlangıç koşulları

$$u(0) = 2, \quad u'(0) = 0 \quad (4.46)$$

altındaki tam çözümü  $u(t) = (5 - \cos 2t)/2$  dir [97].

PIA(1,1) çözümü;

(4.45) denklemini  $\varepsilon = 1$  suni perturbasyon parametresi ile (3.10) kapalı formunda aşağıdaki gibi yazılır.

$$\begin{aligned}
F(u'', u', u, u_\lambda, \varepsilon, t) \\
&= u''(t) + \varepsilon \left( u\left(\frac{t}{2}\right) + u^2(x) \right) - \sin^4(t) \\
&\quad - \sin^2\left(\frac{t}{2}\right) - 8
\end{aligned} \tag{4.47}$$

(4.47) denklemini kullanılarak (3.14) denkleminde verilen PIA(1,1) algoritmasındaki sıfır olmayan terimler aşağıdaki gibi bulunur.

$$\begin{aligned}
F_{u''} &= 1 \\
F_\varepsilon &= u_n\left(\frac{t}{2}\right) + u_n^2(t) \\
F &= u_n''(t) - \sin^4(t) - \sin^2\left(\frac{t}{2}\right) - 8
\end{aligned} \tag{4.48}$$

(4.48) denklemindeki terimler (3.14) denkleminde yerlerine yazılır ise ele alınan problemin düzeltme terimini bulacağımız denklem elde edilir.

$$(u_n)''_n(t) = -u_n\left(\frac{t}{2}\right) - u_n''(t) - u_n^2(t) + \sin^4(t) + \sin^2\left(\frac{t}{2}\right) + 8 \tag{4.49}$$

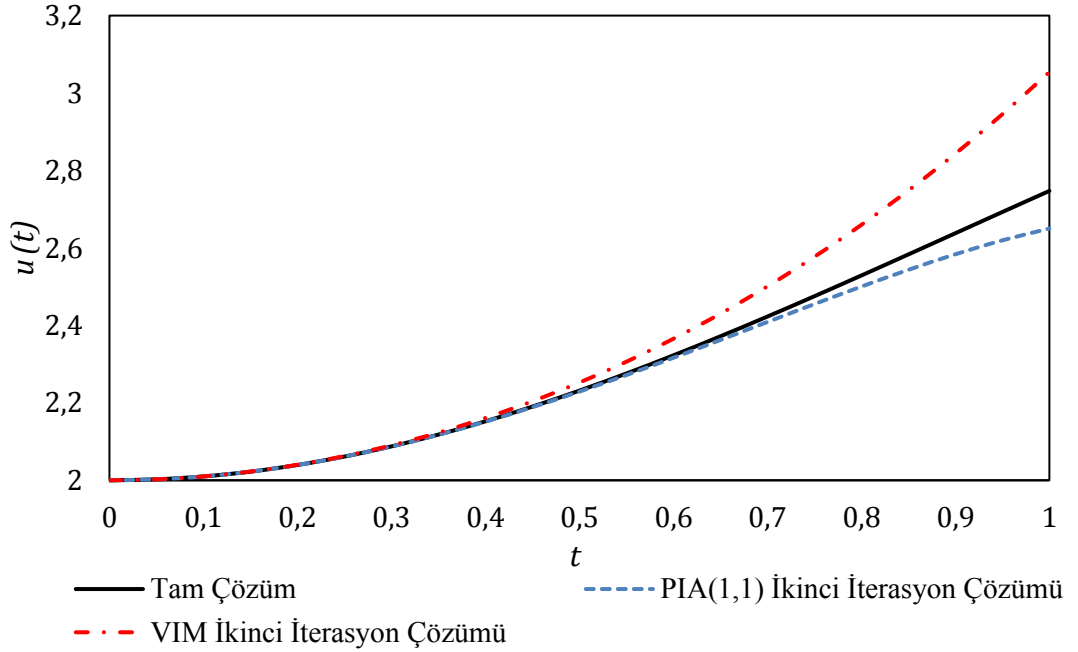
Denklemin (4.46) koşullarını sağlayan başlangıç fonksiyonu

$$u_0 = 2 \tag{4.50}$$

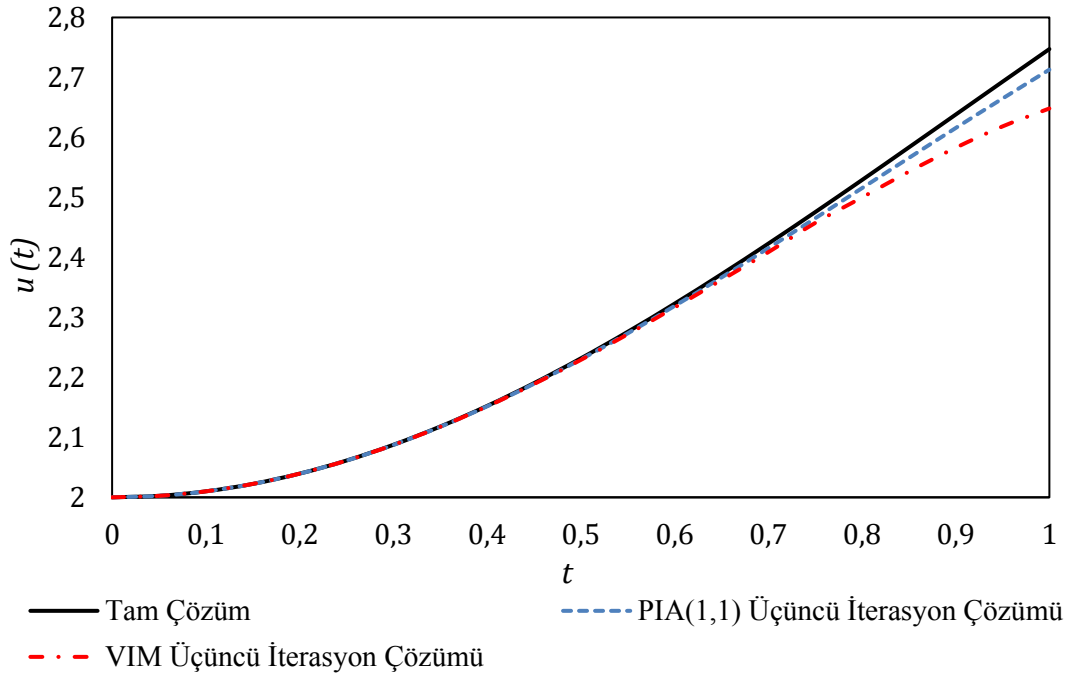
olarak seçilir ve (4.49) denkleminde uygulanırsa aşağıdaki çözümler elde edilir.

$$\begin{aligned}
u_1 &= \frac{1}{10} - \frac{1}{8}e^{-\frac{1}{5}t} + \frac{41}{40}e^{-t} \\
u_2 &= \frac{433}{4096}\cos(2t) + 2\cos\left(\frac{t}{2}\right) + \frac{5}{768}\cos(3t) - \frac{267}{32768}\cos(4t) \\
&\quad - \frac{1}{6400}\cos(5t) - \frac{1}{36864}\cos(6t) + \frac{1}{2097152}\cos(8t) \\
&\quad - \frac{839}{128}\cos t + \frac{23}{16}t^2\cos t - \frac{23}{128}t\sin(2t) + \frac{23}{16384}t\sin(4t) \\
&\quad + \frac{23}{256}t^2\cos(2t) - \frac{23}{16384}t^2\cos(4t) - \frac{23}{4}t\sin t \\
&\quad + \frac{178493}{65536}t^2 - \frac{4439}{12288}t^4 - \frac{529}{7680}t^6 + \frac{3043867103}{471859200} \\
&\quad \vdots
\end{aligned} \tag{4.51}$$

PIA(1,1) algoritmasıyla bulunan birinci iterasyon çözümü, VIM ile elde edilen birinci iterasyon çözümü ile aynı sonuçları vermiştir [97]. Öte yandan, Şekil 4.10 ve Şekil 4.11’ de görüldüğü üzere ikinci ve üçüncü iterasyon PIA(1,1) çözümleri VIM ile elde edilen çözümlere göre tam çözüme daha yakındır.



**Şekil 4.10.** Problem (4.45)’in ikinci iterasyon PIA(1,1) ve VIM çözümleri ile tam çözümün karşılaştırılması



**Şekil 4.11.** Problem (4.45)'in üçüncü iterasyon PIA(1,1) ve VIM çözümleri ile tam çözümün karşılaştırılması

PIA(1,2) çözümü;

(4.45) denkleminde (3.16) denkleminde verilen PIA(1,2) algoritmasını uygulamak için denklemin lineer olmayan  $u^2(t)$  teriminin ve pantograf  $u\left(\frac{t}{2}\right)$  teriminin önüne uygun suni  $\varepsilon$  parametresi konularak problem, (3.10) kapalı formunda aşağıda yazılmıştır.

$$F(u'', u', u, u_\lambda, \varepsilon, t) = u''(t) + \varepsilon u^2(t) + \varepsilon^2 u\left(\frac{t}{2}\right) - \left(\sin^4(t) + \sin^2\left(\frac{t}{2}\right) + 8\right) = 0 \quad (4.52)$$

burada  $\varepsilon = 1$  dir. (3.16) PIA(1,2) algoritmasının sıfırdan farklı terimleri



$$\begin{aligned}
F_{u''} &= 1 \\
F_{u_\varepsilon} &= 2u_n(t) \\
F_\varepsilon &= u_n^2(t) \\
F_{\varepsilon\varepsilon} &= 2u_n\left(\frac{t}{2}\right) \\
F &= u_n''(t) - \sin^4(t) - \sin^2\left(\frac{t}{2}\right) - 8
\end{aligned} \tag{4.53}$$

olarak elde edilmiştir. (4.52) denkleminin (4.53)'de verilen terimleri (3.16) denkleminde yerine konursa düzeltme terimini veren denklem aşağıdaki gibi bulunur.

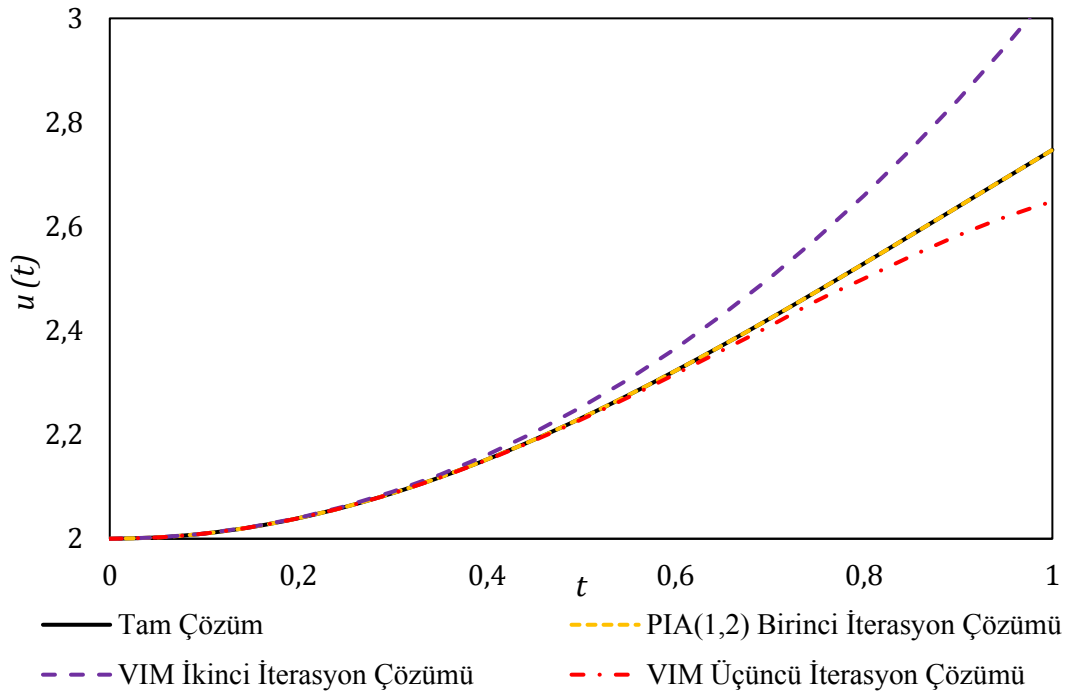
$$\begin{aligned}
&(u_c)''_n + (2u_n(t))(u_c)_n(t) \\
&= -u_n\left(\frac{t}{2}\right) - u_n''(t) - u_n^2(t) + \sin^4(t) + \sin^2\left(\frac{t}{2}\right) \\
&+ 8
\end{aligned} \tag{4.54}$$

Problemin başlangıç koşullarını sağlayan başlangıç iterasyon fonksiyonu (4.50), (4.54) algoritmasında uygulanarak problemin PIA(1,2) birinci iterasyon çözümü

$$\begin{aligned}
u_1 &= \frac{263}{96} - \frac{49}{96} \cos(2t) - \frac{1}{12} \cos^4(2t) + \frac{1}{48} \cos^2(t) \\
&- \frac{1}{4} t \sin(t) \cos(t) - \frac{1}{6} \cos(t)
\end{aligned} \tag{4.55}$$

olarak bulunur.

Şekil 4.12'de, (4.51)'de verilen problemin, PIA(1,2) algoritması ile elde edilen birinci iterasyon çözümü, VIM ile elde edilen çözümler ve problemin tam çözümü karşılaştırılmıştır. Burada açıkça görüldüğü üzere PIA(1,2) algoritmasıyla elde edilen birinci iterasyon çözümü VIM ile elde edilen ilk üç iterasyon çözümünden daha iyi sonuç vermiştir.



**Şekil 4.12.** Problem (4.45)'in PIA(1,2) algoritması ile elde edilen birinci iterasyon çözümünün VIM ile elde edilen ikinci ve üçüncü iterasyon çözümleri ve problemin tam çözümünün karşılaştırılması

#### 4.2.5. Sayısal Örnek 5

Değişken katsayılı lineer olmayan pantograf tipi gecikmeli diferansiyel denklem

$$u''(t) = u(t) - \frac{8}{t^2} u^2\left(\frac{t}{2}\right), \quad t \geq 0 \quad (4.56)$$

ve başlangıç koşulları

$$u(0) = 0, \quad u'(0) = 1 \quad (4.57)$$

ile verilen problemin tam çözümü  $u(t) = te^{-t}$  dir [72].

(4.56) denklemini  $\varepsilon = 1$  olarak seçilen suni perturbasyon parametresi kullanılarak (3.10) kapalı formunda

$$F(u'', u', u, u_\alpha, \varepsilon, t) = u''(t) + \varepsilon \left( \frac{8}{t^2} u^2 \left( \frac{t}{2} \right) - u(t) \right) = 0 \quad (4.58)$$

Şeklinde yazılabilir. PIA(1,1) algoritmasında verilen tüm sıfırdan farklı terimler (4.58)'den aşağıdaki gibi elde edilir.

$$\begin{aligned} F_{u''} &= 1 \\ F_\varepsilon &= \frac{8}{t^2} u_n^2 \left( \frac{t}{2} \right) - u_n(t) \\ F &= u_n''(t) \end{aligned} \quad (4.59)$$

(4.59) denkleminde elde edilen tüm terimler (3.14) algoritmasında yazılırsa her bir iterasyon için

$$(u_c)''_n(t) = -\frac{8}{t^2} u_n^2 \left( \frac{t}{2} \right) + u_n(t) - u_n''(t) \quad (4.60)$$

düzeltilme terimi elde edilir. (4.57) başlangıç koşullarını sağlayan başlangıç iterasyon fonksiyonu

$$u_0(t) = t \quad (4.61)$$

olarak seçilirse (4.60) denklemi ile düzeltilme terimleri bulunur. (3.4) açılımı kullanılarak problemin PIA(1,1) çözümleri aşağıdaki gibi elde edilir.

$$\begin{aligned} u_1(t) &= t - t^2 + \frac{1}{6} t^3 \\ u_2(t) &= t - t^2 + \frac{1}{2} t^3 - \frac{5}{36} t^4 + \frac{1}{80} t^5 - \frac{1}{8640} t^6 \\ &\vdots \end{aligned} \quad (4.62)$$

Farklı  $n$  iterasyon sayıları için maksimum mutlak hata değerleri  $E_n$  Tablo 4.7.'de verilmiştir. Burada iterasyon sayısı arttıkça maksimum mutlak hata değerlerinin düzgün bir şekilde azaldığı görülmektedir.

**Tablo 4.7.** Problem (4.57)'nin  $n = 2, 3, \dots, 7$  için PIA(1,1) çözümlerinin maksimum mutlak hata değerleri

$n$	2	3	4	5	6	7
$E_n$	$5.61e - 3$	$1.59e - 5$	$2.75e - 7$	$3.68e - 9$	$6.76e - 11$	$4.94e - 13$

#### 4.2.6. Sayısal Örnek 6

İkinci mertebe lineer olmayan pantograf tipi gecikmeli diferansiyel denklem

$$u''(x) = ([u(x)]^2 + |u(x)|^3)u\left(\frac{x}{2}\right), \quad x \geq 0 \quad (4.63)$$

ve sınır koşulları

$$u(0) = 1, \quad u(1) = \frac{1}{2} \quad (4.64)$$

ile tanımlı problemin tam çözümü  $u(x) = 1/(1+x)$  dir [73, 74].

(4.63) denklemi aşağıdaki formda  $\varepsilon$  perturbasyon parametresi ile aşağıdaki formda yazılır.

$$\begin{aligned} F(u'', u', u, u_\alpha, \varepsilon, x) &= u''(x) - \varepsilon \left( ([u_n(x)]^2 + |u(x)|^3)u\left(\frac{x}{2}\right) \right) \\ &= 0 \end{aligned} \quad (4.65)$$

Burada  $\varepsilon = 1$  alınmıştır. (4.65)'in PIA(1,1) algoritması için sıfırdan farklı terimleri

$$\begin{aligned} F_{u''} &= 1 \\ F_\varepsilon &= - \left( ([u_n(x)]^2 + |u_n(t)|^3)u_n\left(\frac{x}{2}\right) \right) \\ F &= u_n''(x) \end{aligned} \quad (4.66)$$

olarak bulunur. (4.66)'da verilen terimler (3.14) PIA(1,1) algoritmasında yerine konursa

$$(u_c)_n''(x) = -u_n''(x) + \left( ([u(x)]^2 + |u_n(x)|^3)u_n\left(\frac{x}{2}\right) \right) \quad (4.67)$$

diferansiyel denklemi elde edilir. Problemin başlangıç koşullarını sağlayan başlangıç fonksiyonu;

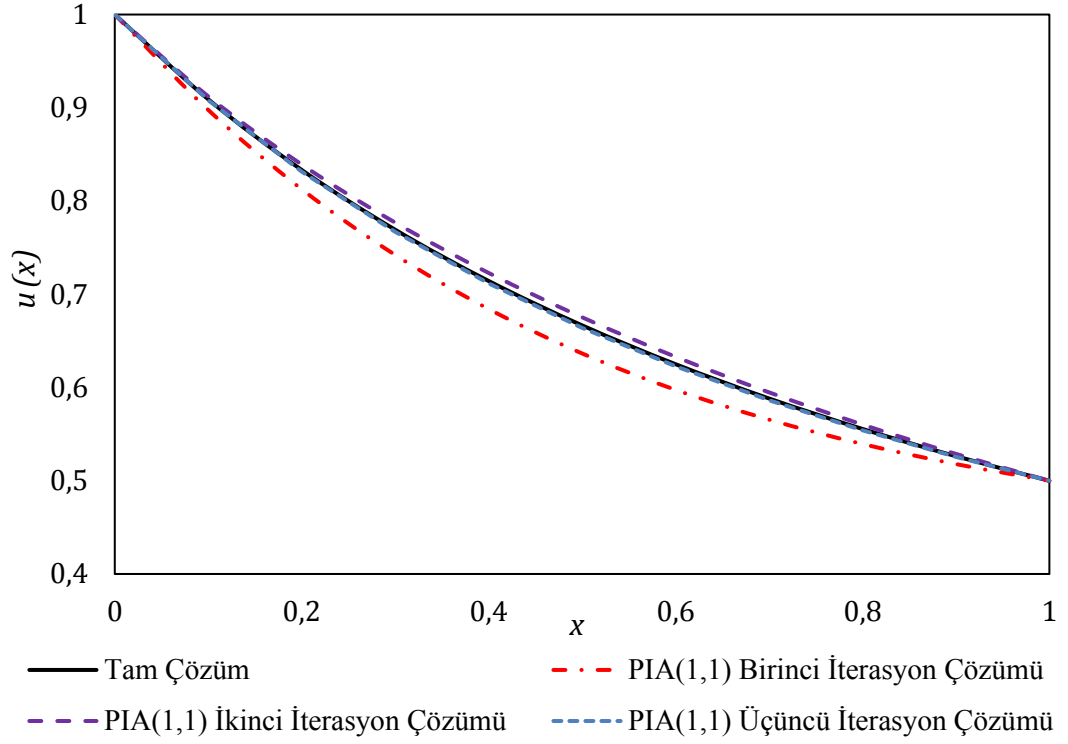
$$u_0(x) = 1 - \frac{x}{2} \quad (4.68)$$

olarak seçilebilir. (4.68) başlangıç iterasyon fonksiyonu (4.67) denkleminde yazılır ve elde edilen düzeltme terimi ile (3.4) açılımı kullanılarak PIA(1,1) çözümleri aşağıdaki gibi bulunur.

$$u_1(x) = 1 - \frac{1073}{960}x + x^2 - \frac{1}{2}x^3 + \frac{13}{96}x^4 - \frac{3}{160}x^5 + \frac{1}{960}x^6 \quad (4.69)$$

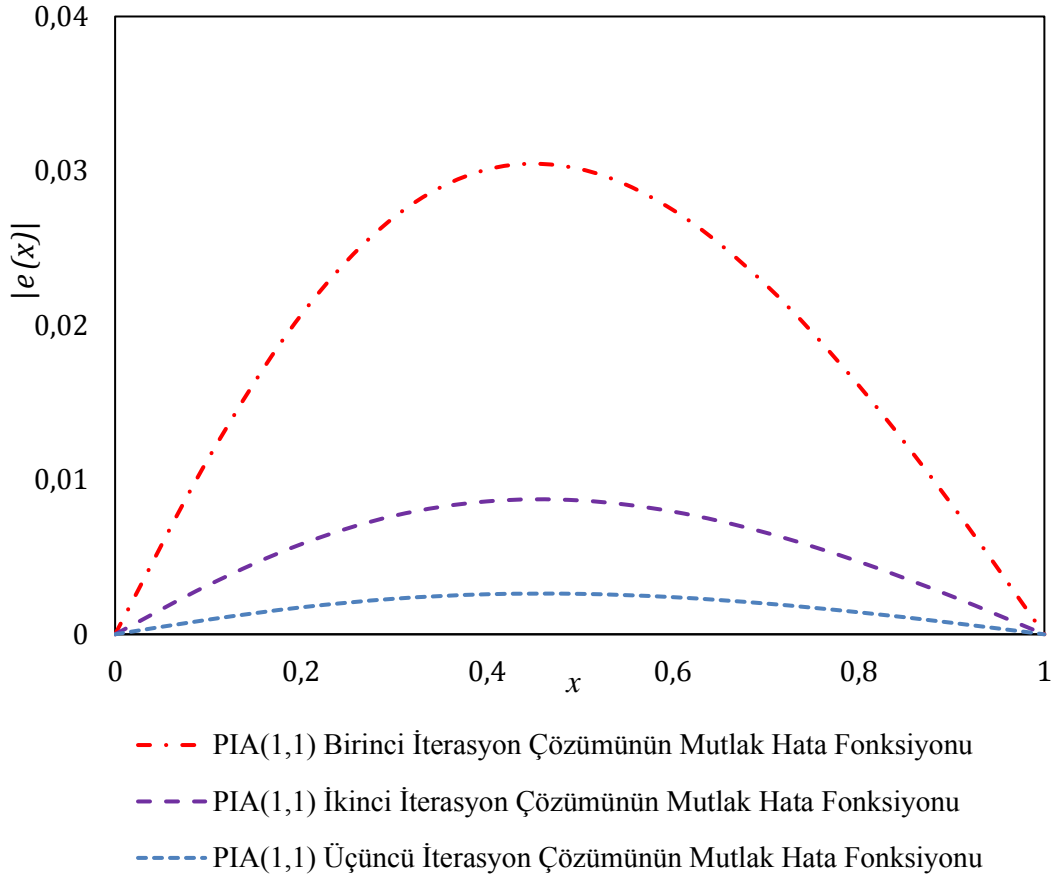
⋮

Şekil 4.13'de PIA(1,1) algoritmasının ilk üç iterasyon çözümü ile problemin tam çözümü verilmiştir. İterasyon çözümleri tam çözüme yakınsamaktadır.



**Şekil 4.13.** Problem (4.63)'ün PIA(1,1) algoritmasının ilk üç iterasyon çözümü ile problemin tam çözümünün karşılaştırılması

Şekil 4.14'de PIA(1,1) algoritmasının ilk üç iterasyon çözümünün mutlak hata fonksiyonları verilmiştir. Maksimum mutlak hata değeri ilk iterasyon için 0.03013 iken, üçüncü iterasyon değeri için 0.00262 değerine düşmüştür.



**Şekil 4.14.** Problem (4.63)'ün PIA(1,1) algoritmasının ilk üç iterasyon çözümünün mutlak hata fonksiyonlarının grafiklerinin karşılaştırılması

### 4.3. Geçmiş Fonksiyonuna Dayalı Gecikmeli Diferansiyel Denklemler için Sayısal Örnekler

Bu bölümde (3.17) formundaki geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklemler için Bölüm 3.3.'de geliştirilen PIA(1,1) algoritması kullanılarak çeşitli sayısal örnekler çözülmüştür.

#### 4.3.1. Sayısal Örnek 1

İkinci mertebe lineer geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklem

$$u''(t) = -\frac{1}{2}u(t) + \frac{1}{2}u(t - \pi), \quad t \in [0, \pi], \quad (4.70)$$

ve geçmiş fonksiyonu

$$h(t) = 1 - \sin(t), \quad -\pi \leq t \leq 0. \quad (4.71)$$

olan probleminin tam çözümü  $u(t) = 1 - \sin(t)$  dir [98, 99]. Burada (2.6)'da verilen  $a = 0$  ,  $b = \pi$  ve  $\tau = \pi$  olduğundan (3.20)'den  $J = \left[ \frac{\pi}{\pi} \right] = 1$  alt aralıkta çözüm bulunacaktır. Bu yüzden (4.70)'in PIA çözümü  $u_n(t) = u_n^{D_1}(t)$  olarak alınır.

(4.70), (3.22) formunda  $\varepsilon$  suni perturbasyon parametresi ile

$$\begin{aligned} F^{D_1}((u^{D_1})'', (u^{D_1})', u^{D_1}, \varepsilon, t) \\ &= (u^{D_1})''(t) + \frac{1}{2}\varepsilon u^{D_1}(t) - \frac{1}{2}u^{D_0}(t - \pi) \\ &= (u^{D_1})''(t) + \frac{1}{2}\varepsilon u^{D_1}(t) - \frac{1}{2}h(t - \pi) = 0 \end{aligned} \quad (4.72)$$

olarak belirlenir. Burada  $D_0 = [-\pi, 0]$  dir. (3.27) algoritmasının (3.26)'da verilen sıfırdan farklı terimleri

$$\begin{aligned} F_{(u^{D_1})}^{D_1} &= 1 \\ F_{\varepsilon}^{D_1} &= \frac{1}{2}u_n^{D_1}(t) \\ F^{D_1} &= (u^{D_1})_n''(t) - \frac{1}{2}h(t - \pi) \end{aligned} \quad (4.73)$$

olarak bulunur. (4.73)'teki terimler (3.27)'de yerine konursa

$$\begin{aligned} (u_c^{D_1})_n''(t) &= -\frac{1}{2}u_n^{D_1}(t) - \left( (u^{D_1})_n''(t) - \frac{1}{2}h(t - \pi) \right) \\ &= -\frac{1}{2}u_n^{D_1}(t) - (u^{D_1})_n''(t) + \frac{1}{2}(1 - \sin(t - \pi)) \end{aligned} \quad (4.74)$$

diferansiyel denklemini elde edilir.



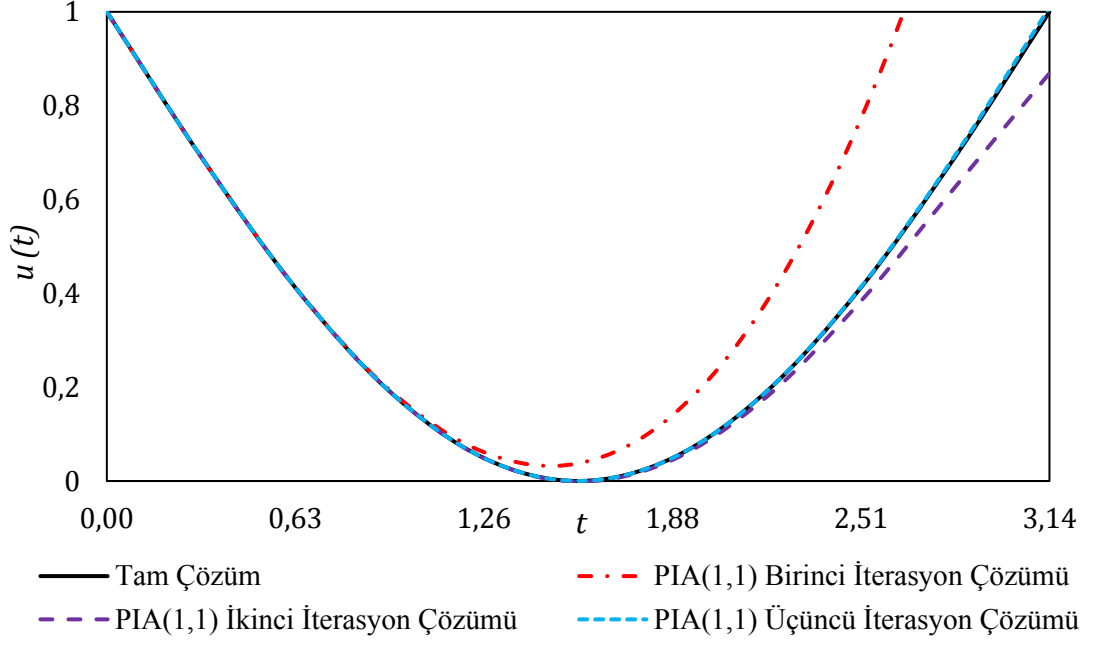
Problem ikinci mertebe ( $k = 2$ ) olduğundan  $P_1^{D_1}(t)$  polinomunun katsayılar matrisi (3.30)'da tanımlanan  $\mathbf{M}^{D_1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$  ve (3.31)'de tanımlanan  $\mathbf{B}^{D_1} = \begin{bmatrix} h(0) \\ h'(0) \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$  matrislerinin (3.29)'da tanımlı  $\mathbf{A}^{D_1} = (\mathbf{M}^{D_1})^{-1}\mathbf{B}^{D_1} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$  çarpımına eşit olacağından,  $u_0^{D_1}(t)$  başlangıç fonksiyonu

$$u_0^{D_1}(t) = P_1^{D_1}(t) = 1 - t \quad (4.75)$$

olarak seçilir ve (4.70) denklemini ve (3.24) açılımını ile PIA(1,1) çözümleri aşağıdaki gibi bulunur.

$$\begin{aligned} u_1^{D_1}(t) &= u_1(t) = \frac{1}{12}t^3 - \frac{1}{2}t + 1 - \frac{\sin(t)}{2} \\ u_2^{D_1}(t) &= u_2(t) = -\frac{1}{480}t^5 + \frac{1}{24}t^3 - \frac{1}{4}t + 1 - \frac{3\sin(t)}{4} \\ u_3^{D_1}(t) &= u_3(t) = \frac{1}{40320}t^7 - \frac{1}{960}t^5 + \frac{1}{48}t^3 - \frac{1}{8}t + 1 - \frac{7\sin(t)}{8} \\ u_4^{D_1}(t) &= u_4(t) = -\frac{1}{5806080}t^9 + \frac{1}{80640}t^7 - \frac{1}{1920}t^5 \\ &+ \frac{1}{96}t^3 - \frac{1}{16}t + 1 - \frac{15\sin(t)}{16} \\ &\vdots \end{aligned} \quad (4.76)$$

Şekil 4.15.'de PIA(1,1) algoritmasının ilk dört iterasyon çözümü ile problemin tam çözümü verilmiştir. Görüldüğü üzere iterasyon sayısı arttıkça çözümler tam çözüme yakınsamaktadır. Problem (4.70)'in PIA(1,1) çözümü için geliştirilen MATLAB kodu ve programın çıktıları EK A.4' de sunulmuştur.



**Şekil 4.15.** Problem (4.70)'in PIA(1,1) algoritmasının ilk üç iterasyon çözümü ile problemin tam çözümünün karşılaştırılması

(4.70)'in PIA(1,1) çözümlerinin maksimum mutlak hata değerleri ile Cubic spline [99] ve Direk Adams-Moulton metot [99] yöntemlerinin çözümlerinin karşılaştırılması Tablo 4.8'de sunulmuştur. Seong vd. [99] geliştirdikleri yöntemlerin hesaplama sürecinde kullandıkları kodların çağırdığı toplam fonksiyon sayısını "Totally Called Function" (FCN) olarak adlandırmışlardır. Ayrıca PIA(1,1) algoritmasının çözümlerinin elde edilme süreleri saniye cinsinden verilmiştir. Görüldüğü gibi kesme sınır  $n$  arttıkça maksimum mutlak hata değeri azalmaktadır. Ayrıca PIA(1,1) çözümü ile elde edilen çözümler bu iki yöntemle elde edilen sonuçlara göre daha iyi sonuçlar vermektedir.

İterasyon sayısının artması işlemci süresi (CPU) arttırmasına karşın Maksimum mutlak hata değerinin çok küçük olduğu durumlarda dahi işlem süresi birkaç saniye olarak ölçülmüştür.

**Tablo 4.8.** Problem (4.70)'in PIA(1,1) çözümlerinin Cubic spline ve Direk Adams-Moulton yöntemlerinin çözümleri ile karşılaştırılması

	Cubic spline [99]	DAM [99]			PIA(1,1)	PIA(1,1)
$i$	$h = \pi/10 \cdot 2^i$ (Adım Sayısı)	FCN	$n$	$E_n$	CPU zamanı (s)	
0	$1.84e - 02$	$6.12e - 06$	122	4	$4.32e - 05$	2.712
1	$4.62e - 03$	$3.10e - 07$	288	5	$1.39e - 05$	3.385
2	$1.16e - 03$	$2.92e - 08$	494	6	$3.30e - 07$	4.058
3	$2.89e - 04$	$2.87e - 09$	819	7	$6.03e - 09$	4.733
4	$7.23e - 05$	$2.31e - 10$	1231	8	$8.75e - 11$	5.406
5	$1.81e - 05$	$2.10e - 11$	1960	9	$1.03e - 12$	6.124
6	$4.52e - 06$	$6.37e - 12$	3331	10	$1.01e - 14$	6.797

### 4.3.2. Sayısal Örnek 2

İkinci mertebe lineer gecikme fonksiyonlu zaman gecikmeli diferansiyel denklem

$$u''(t) + u(t) = u(t - 1), \quad t \in [0,2] \quad (4.77)$$

geçmiş fonksiyonu ve başlangıç değeri

$$u(t) = t^2 + 3t + 2, \quad -1 \leq t \leq 0, \quad u'(0) = 0 \quad (4.78)$$

olan problemin tam çözümü

$$u(t) = \begin{cases} 4 \cos(t) - \sin(t) + t^2 + t - 2 & , \quad 0 \leq t \leq 1 \\ -2t \cos(t) \sin(1) + \frac{t \cos(t) \cos(1)}{2} + t^2 \\ -4 + 2 \cos(1) (\cos(t) + t \sin(t)) - t \\ + \frac{\cos(t) \cos(1)}{2} + \frac{t}{2} \sin(t) \sin(1) & , \quad 1 \leq t \leq 2 \\ + \left( \frac{3 \cos(1) + 8}{2} + 4 \right) \cos(t) \\ + \left( \frac{-\cos(1)}{2} + \frac{7 \sin(1)}{2} - 1 \right) \sin(t) \end{cases}$$

dir [76, 99].

(2.6)'da verilen  $a = 0$ ,  $b = 2$  ve  $\tau = 1$  olduğundan (3.20)'den  $J = \begin{bmatrix} 2 \\ 1 \end{bmatrix} = 2$  alt aralıkta çözüm bulunacaktır. (4.77) probleminin tanım aralığı  $D = [0,2]$  ve (3.21)'e göre problemin tanım aralığı  $D = \cup_{i=1}^2 D_i = D_1 \cup D_2 = [0,1] \cup [1,2]$  alt aralıklarının birleşimi olarak yazılır. Böylece (4.77)'nin PIA ile  $n$ . iterasyon çözümü (3.23)'den

$$u_n(t) = \begin{cases} u_n^{D_1}(t) & , \quad t \in D_1 = [0,1] \\ u_n^{D_2}(t) & , \quad t \in D_2 = [1,2] \end{cases} \text{ olarak alınır.}$$

(4.77), (3.22) formunda  $\varepsilon$  suni perturbasyon parametresi ile

$$\begin{aligned} F^{D_1} &= (u^{D_1})''(t) + \varepsilon u^{D_1}(t) - u^{D_0}(t-1) = 0 \\ F^{D_2} &= (u^{D_2})''(t) + \varepsilon u^{D_2}(t) - u^{D_1}(t-1) = 0 \end{aligned} \quad (4.79)$$

denklem sistemine dönüştürülür. Burada  $\varepsilon = 1$  alınmıştır. (4.79) denklem sisteminin  $u_n^{D_j}(t)$ , ( $j = 1, 2$ ) çözümlerinin bulunması için (3.27) algoritmasının (3.26)'da tanımlanan sıfırdan farklı terimleri

$$\begin{aligned} F^{D_j} &= (u^{D_j})'' = 1 \\ F_\varepsilon^{D_j} &= u_n^{D_j}(t) \\ F^{D_j} &= (u_n^{D_j})''(t) - u_n^{D_{j-1}}(t-1) \end{aligned} \quad (4.80)$$

olarak bulunur. Bu terimler (3.27) PIA(1,1) algoritmasında yerine konursa;

$$\left(u_c^{Dj}\right)_n''(t) = u_n^{Dj}(t) - \left(u_n^{Dj}\right)''(t) + u_n^{Dj-1}(t-1), \quad j = 1, 2 \quad (4.81)$$

diferansiyel denklemi elde edilir. (4.81) denklemi (4.77)'nin tanım aralığındaki alt aralıkların çözümlerinin bulunması için kullanılacaktır.

$j = 1$  için;

$\forall n \in N^+$  için  $u_n^{D_0}(t-1) = u(t-1) = (t-1)^2 + 3(t-1) + 2$  olduğundan (4.81) denklemi  $D_1$  alt aralığında;

$$\left(u_c^{D_1}\right)_n''(t) = -u_n^{D_1}(t) - \left(u_n^{D_1}\right)''(t) - (t-1)^2 - 3(t-1) - 2 \quad (4.82)$$

olacaktır. (4.77) problemi ikinci mertebe ( $k = 2$ ) olduğundan  $P_1^{D_1}(t)$  polinomunun katsayılar matrisi (3.30)'da tanımlanan  $\mathbf{M}^{D_1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$  ve (3.31)'de tanımlanan  $\mathbf{B}^{D_1} = \begin{bmatrix} h(0) \\ u'(0) \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$  matrislerinin (3.29)'da tanımlı  $\mathbf{A}^{D_1} = (\mathbf{M}^{D_1})^{-1}\mathbf{B}^{D_1} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$  çarpımına eşit olacağından,  $u_0^{D_1}(t)$  başlangıç fonksiyonu

$$u_0^{D_1}(t) = P_1^{D_1}(t) = 2 \quad (4.83)$$

olarak seçilir. Böylece (4.77) denklemi ve (3.24) açılımı ile PIA(1,1) çözümleri aşağıdaki gibi bulunur.

$$\begin{aligned} u_1^{D_1}(t) &= \frac{t^4}{12} + \frac{t^3}{6} - t^2 + 2 \\ u_2^{D_1}(t) &= -\frac{t^6}{360} - \frac{t^5}{120} + \frac{t^4}{6} + \frac{t^3}{6} - t^2 + 2 \end{aligned} \quad (4.84)$$

$j = 2$  için;

$$u_2^{D_1}(t-1) = -\frac{(t-1)^6}{360} - \frac{(t-1)^5}{120} + \frac{(t-1)^4}{6} + \frac{(t-1)^3}{6} - (t-1)^2 + 2 \text{ olduğundan}$$

(4.81) denklemi

$$\begin{aligned} (u_c^{D_2})''_n(t) &= -u_n^{D_2}(t) - (u_n^{D_2})''(t) - \frac{(t-1)^6}{360} - \frac{(t-1)^5}{120} \\ &+ \frac{(t-1)^4}{6} + \frac{(t-1)^3}{6} - (t-1)^2 + 2 \end{aligned} \quad (4.85)$$

olacaktır. (4.77) problemi ikinci merteye ( $k=2$ ) olduğundan  $P_1^{D_2}(t)$  polinomunun katsayılar matrisi (3.30)'da tanımlanan  $\mathbf{M}^{D_2} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$  ve (3.31)'de tanımlanan

$$\mathbf{B}^{D_2} = \begin{bmatrix} u_2^{D_1}(1) \\ (u_2^{D_1})'(1) \end{bmatrix} = \begin{bmatrix} \frac{119}{90} \\ -\frac{107}{120} \end{bmatrix} \text{ matrislerinin (3.29)' da tanımlı } \mathbf{A}^{D_2} = \begin{bmatrix} \frac{797}{360} \\ \frac{107}{120} \end{bmatrix}$$

çarpımına eşit olacağından,  $u_0^{D_2}(t)$  başlangıç fonksiyonu

$$u_0^{D_2}(t) = P_1^{D_2}(t) = \frac{797}{360} - \frac{107}{120}t \quad (4.86)$$

olarak seçilir. Böylece (4.77) denklemi ve (3.24) açılımı ile PIA(1,1) çözümleri aşağıdaki gibi bulunur.

$$\begin{aligned} u_1^{D_2}(t) &= \frac{t^8}{20160} + \frac{t^7}{5040} - \frac{t^6}{180} - \frac{t^5}{120} + \frac{t^4}{6} + \frac{t^3}{6} - t^2 + 2 \\ u_2^{D_2}(t) &= -\frac{t^{10}}{1814400} - \frac{t^9}{362880} + \frac{t^8}{10080} + \frac{t^7}{5040} - \frac{t^6}{360} - \frac{t^5}{120} \\ &+ \frac{t^4}{6} + \frac{t^3}{6} - t^2 + 2 \end{aligned} \quad (4.87)$$

Böylece ele alınan (4.77) probleminin PIA(1,1) algoritmasıyla 2. iterasyon çözümü (4.84) ve (4.87)'den

$$u_2(t) = \begin{cases} -\frac{t^6}{360} - \frac{t^5}{120} + \frac{t^4}{6} + \frac{t^3}{6} - t^2 + 2 & , t \in D_1 \\ -\frac{1814400}{t^{10}} - \frac{362880}{t^9} + \frac{10080}{t^8} + \frac{t^7}{5040} & , t \in D_2 \\ -\frac{t^6}{360} - \frac{t^5}{120} + \frac{t^4}{6} + \frac{t^3}{6} - t^2 + 2 & \end{cases} \quad (4.88)$$

olarak bulunmuş olur.

Seong ve ark. [99] bu problemi çözmek için Direk Adams-Moulton Metodunu (DAM) ve Martin ve ark. [76] ise Variable multistep metodunu (VMM) kullanmışlardır. DAM yöntemi direk metot ve sabit adım boyutları kullanılarak geliştirilmiş bir yöntem iken VMM yöntemi çeşitli adım boyutlarına dayalı bir yöntemdir. Tablo 4.9'da problem (4.77)'nin PIA(1,1) çözümlerinin belirli noktalardaki mutlak hata değerleri ile DAM [99] ve VMM [76] yöntemleri ile elde edilmiş çözümlerin mutlak hata değerleri verilmiştir. Tabloda açıkça görüldüğü gibi PIA(1,1) ile elde edilen çözümlerin hata değerleri diğer iki yöntemle göre daha iyi sonuçlar vermiştir. Seong ve ark. tarafından tanımlanan toplam fonksiyon çağırılma sayıları da (FCN) tabloda sunulmuştur. İterasyon sayısı  $n = 5$  olduğunda PIA(1,1) algoritması için çağırılan fonksiyon sayısı 52 iken,  $n = 6$  için 60 olarak belirlenmiştir. Seong ve ark. bir kodun sayısal hesaplamaları yapabilmek için toplam kullandığı fonksiyon sayısını FCN olarak tanımlamıştır. Ayrıca, PIA(1,1) algoritmasının karmaşıklığı  $O(n^2)$  mertebesindedir.

**Tablo 4.9.** Problem (4.77)'nin PIA(1,1) çözümünün diğer sayısal yöntemler ile karşılaştırılması

$t$	VMM	DAM [99]		PIA(1,1)	
	[76, 99]	FCN	$n = 5$	$n = 6$	
0.1280	$4.98e - 08$	$1.18e - 09$	61	$8.03e - 20$	$7.23e - 24$
0.2360	$1.72e - 07$	$1.37e - 08$	67	$1.23e - 16$	$3.78e - 20$
0.3418	$3.58e - 07$	$6.10e - 08$	67	$1.04e - 14$	$6.73e - 18$
0.4444	$5.91e - 07$	$1.76e - 07$	67	$2.42e - 13$	$2.64e - 16$
0.5440	$8.58e - 07$	$7.82e - 07$	67	$2.73e - 12$	$4.46e - 15$
0.6412	$1.14e - 06$	$2.57e - 07$	67	$1.95e - 11$	$4.44e - 14$
0.7346	$1.44e - 06$	$1.36e - 06$	67	$9.95e - 11$	$2.96e - 13$
0.8264	$1.75e - 06$	$2.21e - 06$	69	$4.07e - 10$	$1.53e - 12$
0.9136	$2.05e - 06$	$3.35e - 06$	73	$1.34e - 09$	$6.23e - 12$
1	$2.34e - 06$	$4.86e - 06$	73	$3.96e - 09$	$2.19e - 11$
1.1264	$4.97e - 08$			$9.91e - 09$	$6.04e - 11$
1.2368	$1.91e - 07$			$1.49e - 08$	$9.33e - 11$
1.3405	$4.05e - 07$			$1.95e - 08$	$1.23e - 10$
1.4440	$6.88e - 07$			$2.39e - 08$	$1.51e - 10$
1.5420	$1.01e - 06$			$2.78e - 08$	$1.77e - 10$
1.6386	$1.37e - 06$			$3.14e - 08$	$2.01e - 10$
1.7309	$1.75e - 06$			$3.45e - 08$	$2.21e - 10$
1.8206	$2.14e - 06$			$3.72e - 08$	$2.38e - 10$
1.9103	$254e - 06$			$3.93e - 08$	$2.53e - 10$
2	$2.94e - 06$			$4.05e - 08$	$2.61e - 10$

Problem (4.77)'nin farklı iterasyon değerleri için PIA(1,1) çözümlerinin maksimum mutlak hata değerlerinin karşılaştırılması ve çözümlerin elde edilme süreleri Tablo 4.10' da sunulmuştur. Buna göre, kesme sınırının artması mutlak maksimum mutlak hata değerlerini düşürürken, çözüm süresini arttırmaktadır. Buna karşın yüksek hassasiyetteki çözümler için PIA(1,1) algoritması ile çözüm birkaç saniye gibi kısa sürelerde elde edilmektedir.



**Tablo 4.10.** Problem (4.77)'nin farklı iterasyon değerleri için PIA(1,1) çözümlerinin maksimum mutlak hata değerlerinin ve çözümlerin elde edilme sürelerinin karşılaştırılması

$n$	2	3	4	5	6
$E_n$	$1.25e - 02$	$3.10e - 04$	$4.44e - 06$	$4.05e - 08$	$2.61e - 10$
CPU süresi	2.69s	3.94s	5.26s	6.48s	7.66s

### 4.3.3. Sayısal Örnek 3

Birinci mertebe lineer olmayan gecikme fonksiyonlu zaman gecikmeli diferansiyel denklemin

$$u'(t) = 1 + u(t) - 2u\left(\frac{t}{2}\right)^2 - u'(t - \pi), \quad 0 \leq t \leq \pi \quad (4.89)$$

geçmiş fonksiyonu

$$h(t) = \cos(t), \quad t \leq 0 \quad (4.90)$$

olarak verilmiştir [94]. Burada  $a = 0$ ,  $b = \pi$  ve  $\tau = \pi$  olduğundan (3.20)'den  $J = \left[\frac{\pi}{\pi}\right] = 1$  alt aralıkta çözüm bulunacaktır. Bu yüzden (4.89)'un PIA çözümü  $u_n(t) = u_n^{D_1}(t)$  olarak alınır.

Denklem (4.89), (3.22) formunda  $\varepsilon$  suni perturbasyon parametresi ile

$$\begin{aligned} F^{D_1}((u^{D_1})', u^{D_1}, u^{D_0}(t), \varepsilon, t) &= (u^{D_1})''(t) - 1 - u^{D_1}(t) + 2\varepsilon u^{D_1}\left(\frac{t}{2}\right)^2 \\ &+ (u^{D_0})'(t - \pi) \\ &= (u^{D_1})''(t) - 1 - u^{D_1}(t) + 2\varepsilon u^{D_1}\left(\frac{t}{2}\right)^2 \\ &- \sin(t - \pi) = 0 \end{aligned} \quad (4.91)$$

olarak belirlenir. Burada  $\varepsilon = 1$  olarak alınmıştır. (3.27) algoritmasının (3.26)'da verilen sıfırdan farklı terimleri

$$\begin{aligned}
 F_{(u^{D_1})'}^{D_1} &= 1 \\
 F_{\varepsilon}^{D_1} &= 2 \left( u_n^{D_1} \left( \frac{t}{2} \right) \right)^2 \\
 F^{D_1} &= (u^{D_1})'_n(t) - 1 - u_n^{D_1}(t) - \sin(t - \pi)
 \end{aligned} \tag{4.92}$$

olarak bulunur. (4.92)'deki terimler (3.27)'de yerine konursa

$$\begin{aligned}
 (u_c^{D_1})'_n(t) - (u_c^{D_1})_n(t) & \\
 &= -(u^{D_1})'_n(t) + 1 + u_n^{D_1}(t) + \sin(t - \pi) \\
 &\quad - 2 \left( u_n^{D_1} \left( \frac{t}{2} \right) \right)^2
 \end{aligned} \tag{4.93}$$

diferansiyel denklemi elde edilir.  $u_0^{D_1}(t)$  başlangıç fonksiyonu

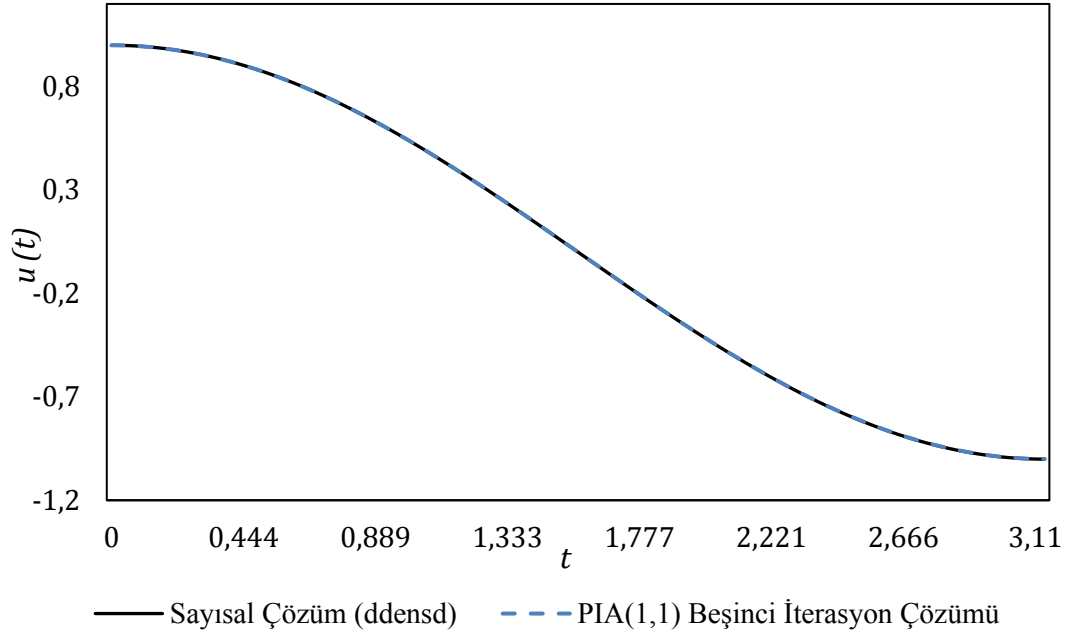
$$u_0^{D_1}(t) = 1 \tag{4.94}$$

olarak seçilirse (4.93) denklemi ve (3.24) açılımı ile PIA(1,1) çözümleri aşağıdaki gibi bulunur.

$$\begin{aligned}
 u_1^{D_1}(t) = u_1(t) &= \frac{\cos(t)}{2} - \frac{e^t}{2} + \frac{\sin(t)}{2} + 1 \\
 u_2^{D_1}(t) = u_2(t) &= \frac{12 \cos\left(\frac{t}{2}\right)}{5} - 4e^{\frac{t}{2}} + \frac{4 \sin\left(\frac{t}{2}\right)}{5} + \frac{47e^t}{20} \\
 &+ \frac{3 \left( \cos\left(\frac{t}{2}\right) \right)^2}{2} - \frac{te^t}{2} - 2 \cos\left(\frac{t}{2}\right) e^{\frac{t}{2}} + \frac{3 \cos\left(\frac{t}{2}\right)}{2} + \frac{3}{4} \\
 &\vdots
 \end{aligned} \tag{4.95}$$

Şekil 4.16.'da (4.89)'un PIA(1,1) algoritmasının beşinci iterasyon çözümü ile problemin MATLAB ddsd fonksiyonu ile çözümü verilmiştir. PIA ile elde edilen

çözüm problemin ddensd fonksiyonu ile elde edilen sayısal çözümü ile aynıdır. Şekilde görüldüğü gibi PIA(1,1) çözümü, ddensd fonksiyonu ile elde edilen sayısal çözüm ile uyumludur.



**Şekil 4.16.** Problem (4.89)'un PIA(1,1) algoritmasının beşinci iterasyon çözümü ile MATLAB ddensd fonksiyonu ile çözümünün karşılaştırılması

## 5. PERTURBASYON-İTERASYON METODUNUN MÜHENDİSLİK PROBLEMLERİNE UYGULANMASI

Bu bölümde, 3. Bölüm'de geliştirilen PIA algoritmalarının etkinliğini 4. Bölüm'de çözülen sayısal örnekler ile doğruladığımız perturbasyon-iterasyon yöntemi çeşitli mühendislik problemlerine uygulayacağız. Yöntem, gecikmeli diferansiyel denklem ile modellenmiş mekanik problemleri olan, gecikmeli Mathieu denklemine, gecikmeli sönümlü Mathieu denklemine ve takım tezgahı titreşim denklemine uygulanmıştır.

### 5.1. Gecikmeli Mathieu Denkleminin Perturbasyon-İterasyon Metodu ile Çözümü

Mathieu denklemleri lineer olmayan titreşim teorisinin en önemli denklemlerinden birisidir [100]. İlk olarak eliptik membranın titreşim probleminde tartışılan denklem, Mathieu'nun kendi ismi ile anılmıştır [101]. Bunun yanında Mathieu denkleminin farklı formları birçok mühendislik probleminde ortaya çıkmıştır. Bunlardan biri periyodik zorlamalı sistemlerdir. Şekil 5.1'de sunulan düşey doğrultuda periyodik hareketli mesnetli matematiksel sarkaç problemi bunlardan biridir. Sistemin hareket denklemi

$$u'' + \left( \frac{g}{L} - \frac{A}{L} \cos t \right) \sin u = 0 \quad (5.1)$$

dir [101]. Burada  $u$  yer değiştirme açısı,  $g$  yerçekimi ivmesi,  $L$  sarkacın uzunluğu,  $A$  genlik ve mesnetin düşey hareketi  $A \cos t$  dir. Sistemin zorlama frekansı  $\Omega$  ve doğal frekansı  $\omega_0 = \sqrt{\frac{g}{L}}$  olarak tanımlanır. Mesnetin hareketi  $A \cos \Omega t$  ve  $\sin u \sim u$  kabul edilirse (5.1) denklemi

$$u'' + \left( \omega_0^2 - \frac{A\Omega^2}{L} \cos \Omega t \right) u = 0 \quad (5.2)$$

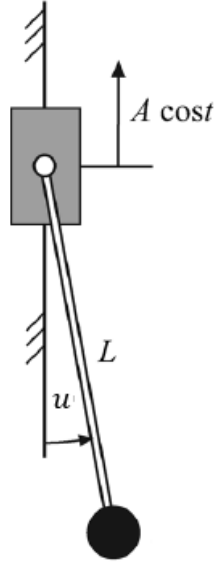
formuna dönüşür. Denklemi boyutsuz hale getirmek için  $t \rightarrow \Omega t$  dönüşümü (5.2) denkleminde uygulanır. Diğer taraftan boyutsuzlaştırma parametreleri

$$\varepsilon = -\frac{A}{L}, \quad \delta = \frac{g}{L\Omega^2} = \frac{\omega_0^2}{\Omega^2} \quad (5.3)$$

olarak tanımlanırsa (5.2) denklemi aşağıdaki formda yazılır.

$$u'' + (\delta + \varepsilon \cos t)u = 0 \quad (5.4)$$

İkinci mertebe lineer adi diferansiyel denklem olan (5.4) denklemi klasik Mathieu denklemdir. (4.97) denkleminin farklı formları da Mathieu denklemleri olarak adlandırılmaktadır. Kovacic ve ark. [55] bu denklemlerin bir arada sunulduğu bir çalışma yayınlamışlardır.



**Şekil 5.1.** Düşey doğrultuda periyodik hareketli mesnetli matematiksel sarkaç problemi [55]

Parametrik hareketli mesnetli olan bir sisteme bağlı salınım yapan bir sistemi denge konumuna getirebilmek için lineer bir kontrol kuvveti uygulanırsa, geri bildirimden kaynaklanan bir gecikme olacaktır. Gecikmenin sisteme etkisi, ikinci mertebe lineer gecikmeli diferansiyel denklem olan gecikmeli Mathieu denklemleri

$$u''(t) + (\delta + \varepsilon \cos t)u(t) = \zeta u(t - \tau) \quad (5.5)$$

ile tanımlanmıştır [102]. Burada  $\zeta$  gecikme teriminin katsayısı ve gecikme temel period değeri  $2\pi$  ye eşittir [102]. Probleminin başlangıç koşulları  $u(t) = 0.001$ ,  $-\tau \leq t \leq 0$  ve  $u'(0) = 0$  olarak alınmıştır.

Bölüm 3.3'de sunulan PIA algoritması ile Gecikmeli Mathieu denkleminin çözümünü  $[0, 4\pi]$  aralığında bulalım.

(2.6)'da tanımlanan  $a = 0$ ,  $b = 4\pi$  ve  $\tau = 2\pi$  olduğundan (3.20)'den  $J = \left\lfloor \frac{4\pi-0}{2\pi} \right\rfloor = 2$  alt aralıkta çözüm bulunacaktır. (5.5) probleminin tanım aralığı  $D = [0, 4\pi]$  ve (3.21)'e göre problemin tanım aralığı  $D = \cup_{i=1}^2 D_i = D_1 \cup D_2 = [0, 2\pi] \cup [2\pi, 4\pi]$  alt aralıkların birleşimi olarak yazılır. Böylece (4.77)'nin PIA ile  $n$ . iterasyon

çözümü (3.23)'den  $u_n(t) = \begin{cases} u_n^{D_1}(t) & , \quad t \in D_1 = [0, 2\pi] \\ u_n^{D_2}(t) & , \quad t \in D_2 = [2\pi, 4\pi] \end{cases}$  olarak alınır.

(4.99), (3.22) formunda  $\varepsilon$  suni perturbasyon parametresi ile

$$\begin{aligned} F^{D_1} &= (u^{D_1})''(t) + (\delta + \varepsilon \cos t)u^{D_1}(t) - \zeta u^{D_0}(t - \tau) = 0 \\ F^{D_2} &= (u^{D_2})''(t) + (\delta + \varepsilon \cos t)u^{D_2}(t) - \zeta u^{D_1}(t - \tau) = 0 \end{aligned} \quad (5.6)$$

denkleme sistemine dönüştürülür. (5.6) denkleminin  $u_n^{D_j}(t)$ , ( $j = 1, 2$ ) çözümlerinin bulunması için (3.27) algoritmasının (3.26)'da tanımlanan sıfırdan farklı terimleri

$$\begin{aligned} F^{D_j} &= (u^{D_j})'' = 1 \\ F_{u^{D_j}}^{D_j} &= \delta \\ F_{\varepsilon}^{D_j} &= \cos t u_n^{D_j}(t) \\ F^{D_j} &= (u_n^{D_j})''(t) + \delta u_n^{D_j}(t) - \zeta u_n^{D_{j-1}}(t - \tau) \end{aligned} \quad (5.7)$$

olarak bulunur. Bu terimler (3.27) PIA(1,1) algoritmasında yerine konursa;

$$\begin{aligned}
& (u_c^{Dj})''(t) + \delta (u_c^{Dj})_n \\
& = -\cos t u_n^{Dj}(t) \\
& - \frac{(u_n^{Dj})''(t) + \delta u_n^{Dj}(t) - \zeta u_n^{D_{j-1}}(t - \tau)}{\varepsilon}, \\
& j = 1, 2
\end{aligned} \tag{5.8}$$

diferansiyel denklemi elde edilir. (5.8) denklemi, (5.5)'in tanım aralığındaki alt aralıkların çözümlerinin bulunması için kullanılacaktır.

$\delta = 18$ ,  $\varepsilon = 1$  ve  $\zeta = 0.5$  için problemin çözümünü bulalım.

$j = 1$  için;

$\forall n \in N^+$  için  $u_n^{D_0}(t - 2\pi) = u(t - 2\pi) = 0.001$  olduğundan (5.8) denklemi  $D_1$  alt aralığında;

$$\begin{aligned}
& (u_c^{D_1})''(t) + 18(u_c^{D_1})_n \\
& = -\cos t u_n^{D_1}(t) - (u_n^{D_1})''(t) + 18u_n^{D_1}(t) \\
& - 0.0005
\end{aligned} \tag{5.9}$$

olacaktır ve (5.5) problemi ikinci mertebe ( $k = 2$ ) olduğundan  $P_1^{D_1}(t)$  polinomunun katsayılar matrisi (3.30)'da tanımlanan  $\mathbf{M}^{D_1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$  ve (3.31)'de tanımlanan

$\mathbf{B}^{D_j} = \begin{bmatrix} h(0) \\ u'(0) \end{bmatrix} = \begin{bmatrix} 0.001 \\ 0 \end{bmatrix}$  matrisleri (3.29)'da tanımlı  $\mathbf{A}^{D_1} = (\mathbf{M}^{D_1})^{-1} \mathbf{B}^{D_1} = \begin{bmatrix} 0.001 \\ 0 \end{bmatrix}$  çarpımına eşit olacağından,  $u_0^{D_1}(t)$  başlangıç fonksiyonu

$$u_0^{D_1}(t) = P_1^{D_1}(t) = 0.001 \tag{5.10}$$

olarak seçilir ve (5.10) denklemi ve (3.24) açılımı ile PIA(1,1) çözümleri aşağıdaki gibi bulunur.

$$\begin{aligned}
u_1^{D_1}(t) &= \frac{631}{612000} \cos(3\sqrt{2}t) - \frac{1}{17000} \cos(t) + \frac{1}{36000} \\
u_2^{D_1}(t) &= \frac{33277}{33796000} \cos(3\sqrt{2}t) + \frac{1}{612000} \cos(t) \\
&\quad + \frac{1}{238000} \cos^2(t) \\
&\quad - \frac{631}{43452000} \cos(3\sqrt{2}t) \cos(t) \\
&\quad - \frac{631\sqrt{2}}{7242000} \sin(3\sqrt{2}t) \sin(t) + \frac{13}{476000} \\
&\quad \dots
\end{aligned} \tag{5.11}$$

(5.11)'de sunulan 1. ve 2. iterasyon çözümlerinin yanında 3. ve 4. iterasyon PIA çözümleri EK A.5'de sunulmuştur.

$j = 2$  için;

$$\begin{aligned}
u_2^{D_1}(t - 2\pi) &= \frac{33277}{33796000} \cos\left(3\sqrt{2}(t - 2\pi)\right) + \frac{1}{612000} \cos(t - 2\pi) + \\
&\quad \frac{1}{238000} \cos^2(t - 2\pi) - \frac{631}{43452000} \cos\left(3\sqrt{2}(t - 2\pi)\right) \cos(t - 2\pi) - \\
&\quad \frac{631\sqrt{2}}{7242000} \sin\left(3\sqrt{2}(t - 2\pi)\right) \sin(t - 2\pi) + \frac{13}{476000} = \frac{33277}{33796000} \cos\left(3\sqrt{2}(t - 2\pi)\right) + \\
&\quad \frac{1}{612000} \cos(t) + \frac{1}{238000} \cos^2(t) - \frac{631}{43452000} \cos\left(3\sqrt{2}(t - 2\pi)\right) \cos(t) + \\
&\quad \frac{631\sqrt{2}}{7242000} \sin\left(3\sqrt{2}(t - 2\pi)\right) \sin(t) + \frac{13}{476000}
\end{aligned}$$

olduğundan (5.8) denklemi



$$\begin{aligned}
& (u_c^{D_2})''(t) + \delta(u_c^{D_2})_n \\
&= -\cos t u_n^{D_2}(t) - (u_n^{D_2})''(t) - 18u_n^{D_2}(t) \\
&+ 0.5 \left( \frac{33277}{33796000} \cos(3\sqrt{2}(t - 2\pi)) \right) \\
&+ \frac{1}{612000} \cos(t) + \frac{1}{238000} \cos^2(t) \\
&- \frac{631}{43452000} \cos(3\sqrt{2}(t - 2\pi)) \cos(t) \\
&+ \frac{631\sqrt{2}}{7242000} \sin(3\sqrt{2}(t - 2\pi)) \sin(t) \\
&+ \frac{13}{476000} \Big)
\end{aligned} \tag{5.12}$$

olacaktır. (5.5) problemi ikinci merteye olduğundan  $P_1^{D_2}(t)$  polinomunun katsayılar matrisi  $u_2^{D_1}(t)$  çözümü ve (3.30) ve (3.31) çözümleri hesaplanarak, (3.29)' da tanımlı katsayı matrisi elde edilir. Buradan  $u_0^{D_2}(t)$  başlangıç fonksiyonu

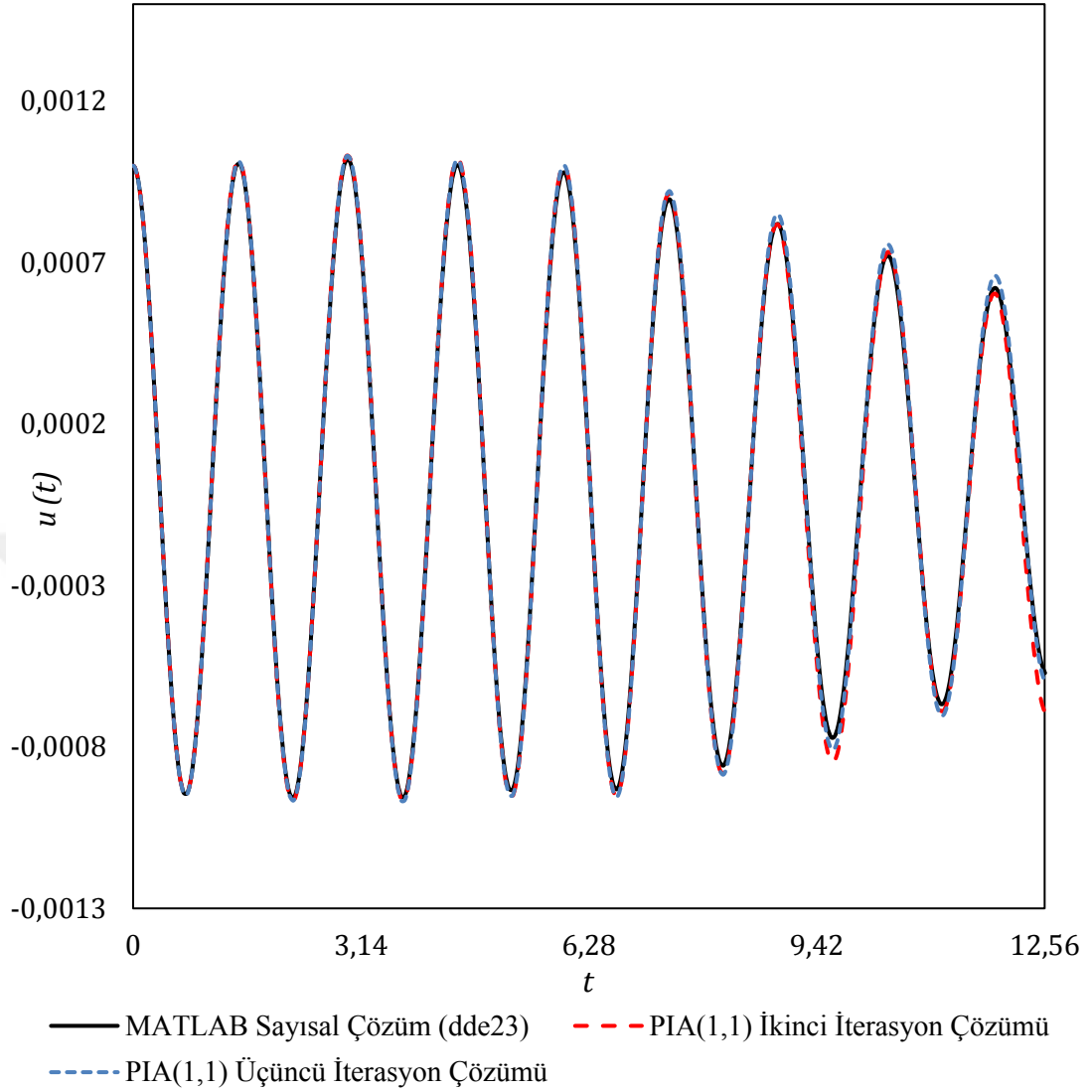
$$\begin{aligned}
u_0^{D_2}(t) &= P_1^{D_2}(t) \\
&= \frac{1039}{1071000} \cos(6\sqrt{2}\pi) \\
&+ \frac{30391}{5069400} \sqrt{2}\pi \sin(6\sqrt{2}\pi) + \frac{4}{133875} \\
&- \frac{30391}{10138800} \sqrt{2}\pi \sin(6\sqrt{2}\pi) t
\end{aligned} \tag{5.13}$$

olarak seçilir ve (5.5) denklemini ve (3.24) açılımını ile PIA(1,1) çözümünü aşağıdaki gibi bulunur.

$$\begin{aligned}
& u_1^{D_2}(t) \\
&= \frac{892813112904416637092824217957}{31603965892678671397168349184000} \cos(3\sqrt{2}(t - 2\pi)) \\
&+ \frac{\cos(2t)}{13328000} + \frac{20772908464171673 \cos(t - 6\sqrt{2}\pi) + 3\sqrt{2}t}{54273512428139446272000} \\
&- \frac{26323243161184921 \cos(t + 6\sqrt{2}\pi) - 3\sqrt{2}t}{54273512428139446272000} \\
&- \frac{150612577646717851053529}{95959962671437087506432000} \cos(t) \\
&+ \frac{2214541899769534380210542855229}{75568842706637735635449011607961600} \sin(t) \\
&- \frac{(2\sqrt{2} \times 20772908464171673) \cos(t - 6\sqrt{2}\pi) + 3\sqrt{2}t}{9045585404689907712000} \\
&- \frac{(2\sqrt{2} \times 26323243161184921) \cos(t + 6\sqrt{2}\pi) - 3\sqrt{2}t}{9045585404689907712000} \\
&+ \frac{2214541899769534380210542855229}{8890452083133851251229295483289600} t \cos(t) \\
&- \frac{19930877097925809421894885697061\pi \cos(3\sqrt{2}t - 6\sqrt{2}\pi)}{2222613020783462812807323870822400} \\
&+ \frac{7849358604226099}{1507597567448317952000} 2\sqrt{2} \cos(3\sqrt{2}t - 6\sqrt{2}\pi) \\
&- \frac{8119986965821626060771990469173}{10795548958091105090778430229708800} 2\sqrt{2} \sin(3\sqrt{2}t \\
&- 6\sqrt{2}\pi) - \frac{33277}{405552000} 2\sqrt{2}\pi \sin(3\sqrt{2}t - 6\sqrt{2}\pi) \\
&+ \frac{33277}{811104000} 2\sqrt{2}t \sin(3\sqrt{2}t - 6\sqrt{2}\pi) \\
&+ \frac{43042402838655619061}{52683901074514479415296000} \\
& u_2^{D_2}(t) = \dots
\end{aligned} \tag{5.14}$$

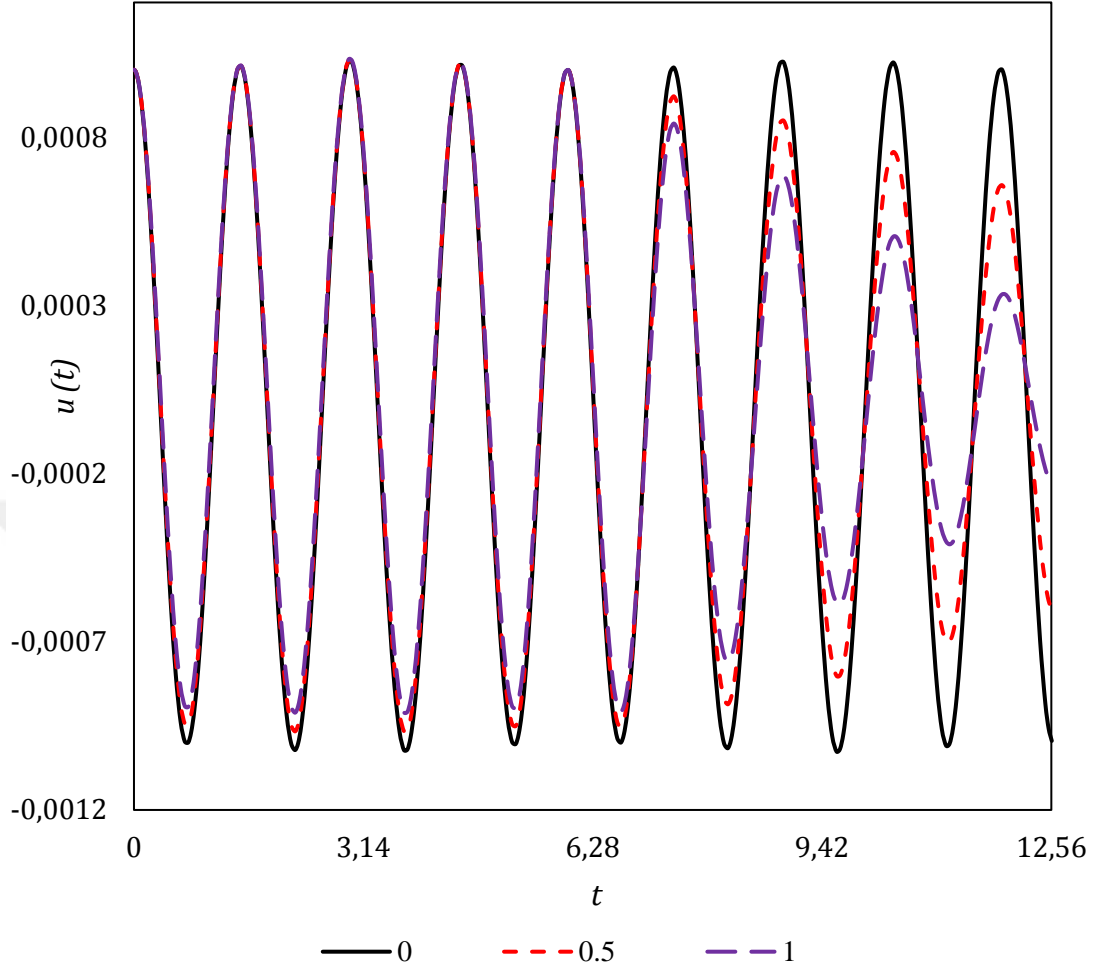
Böylece ele alınan (5.5) probleminin PIA(1,1) algoritmasıyla 2. iterasyon çözümü elde edilmiş olur. İkinci iterasyon çözümü  $u_2^{D_2}(t)$  EK B.1’de sunulmuştur.

Şekil 5.2’de gecikmeli Mathieu denkleminin ikinci iterasyon ve üçüncü iterasyon PIA(1,1) çözümlerinin MATLAB dde23 kodu ile bulunan sayısal çözümlerle uyum gösterdiği görülmektedir.



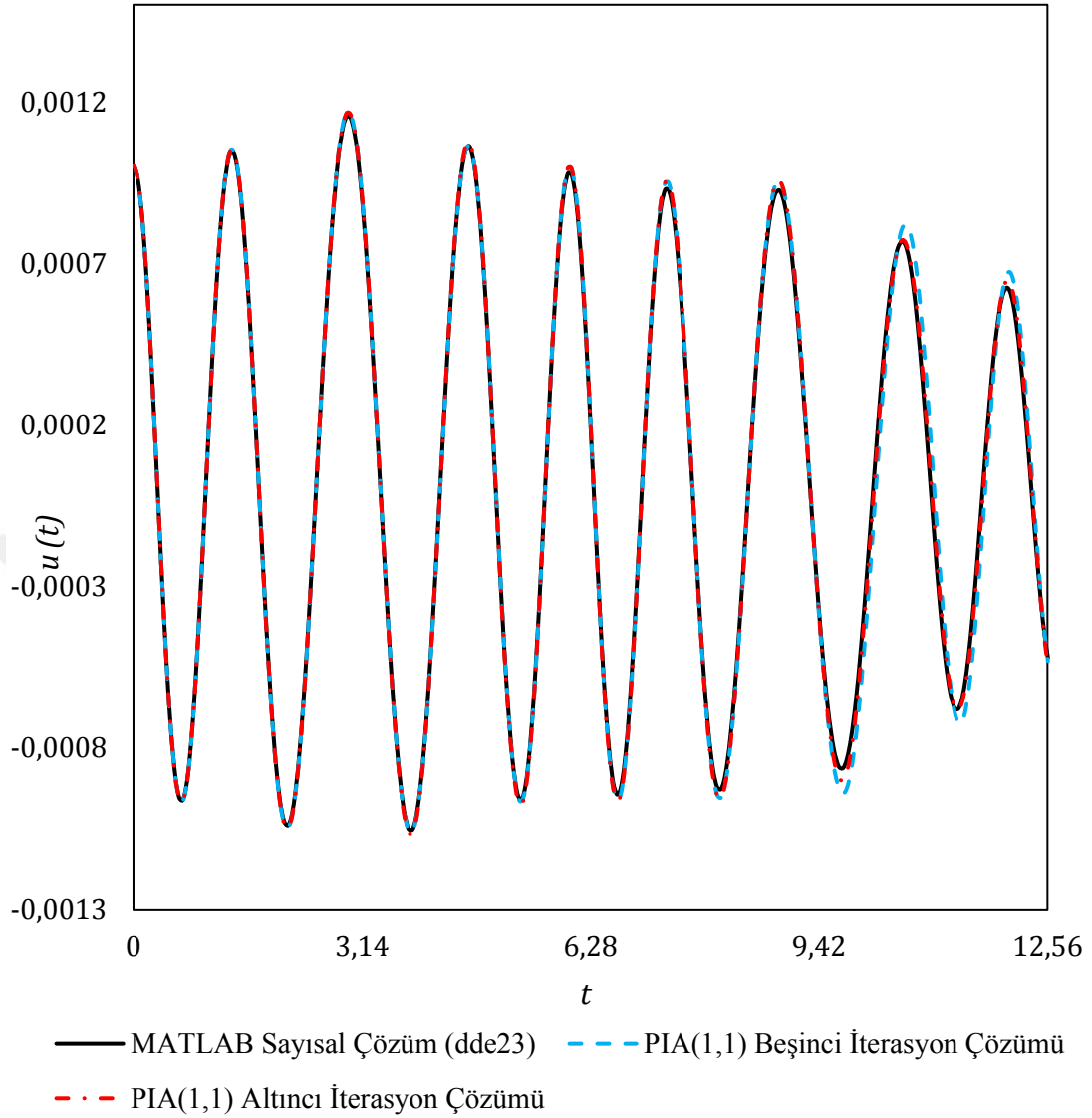
**Şekil 5.2.** Gecikmeli Mathieu denkleminin PIA(1,1) algoritmasının ikinci ve üçüncü iterasyon çözümü ile MATLAB (dde23) sayısal çözümünün karşılaştırılması ( $\delta = 18.0$ ,  $\varepsilon = 1$ ,  $\zeta = 0.5$ , ve  $\tau = 2\pi$ )

$\zeta = 0$  durumu için klasik Mathieu denkleminin PIA(1,1) algoritması ile elde edilen dördüncü iterasyon çözümü ile gecikmeli Mathieu denkleminin dördüncü iterasyon çözümleri Şekil 5.3’de sunulmuştur. Gecikme teriminin katsayısı  $\zeta$ ’nın farklı değerleri ile gecikme teriminin sisteme etkisi açıkça görülmektedir. Gecikmeli terimin sistemde sönüm etkisi yaptığı görülmektedir. Burada klasik Mathieu denklemi Bölüm 3.1’de sunulan PIA(1,1) algoritması ile gecikme teriminin katsayıları sıfır alınarak çözülmüştür.



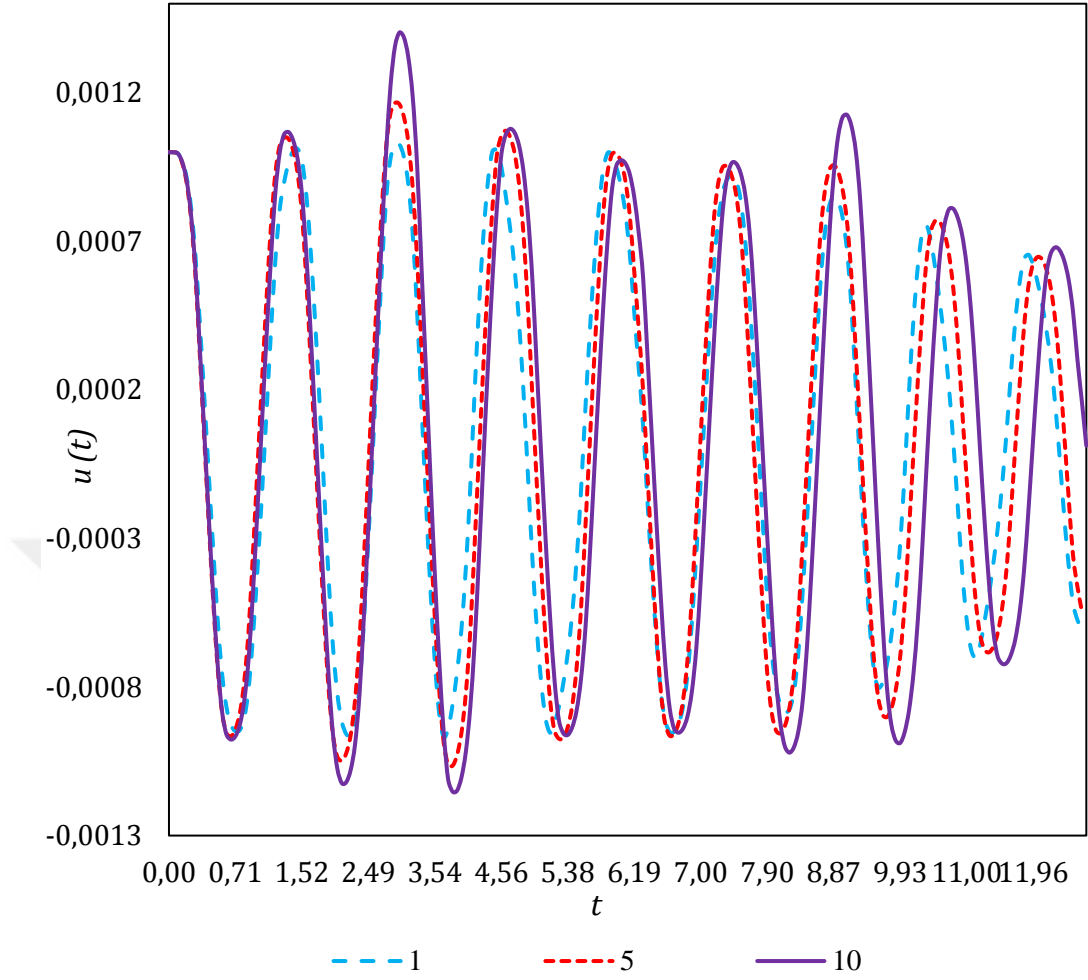
**Şekil 5.3.** Gecikmeli Mathieu denkleminin gecikme teriminin katsayısının farklı değerleri  $\zeta = 0, 0.5$  ve  $1$  için PIA(1,1) dördüncü iterasyon çözümleri ( $\delta = 18, \varepsilon = 1, \tau = 2\pi$ )

Şekil 5.4'de gecikmeli Mathieu denkleminin  $\varepsilon = 5$  için PIA(1,1) algoritmasının beşinci ve altıncı iterasyon çözümü ile MATLAB (dde23) sayısal çözümünün karşılaştırılması verilmiştir. PIA(1,1) algoritması ile elde edilen sonuçlar sayısal çözümle uyumludur.  $\delta = 18$  ve  $d = 0.5$  olmak üzere  $\varepsilon = 1$  için sayısal çözüm ile problemin üçüncü iterasyon PIA(1,1) algoritması çözümü uyumlu iken,  $\varepsilon = 5$  için sayısal çözüm ile problemin altıncı iterasyon PIA(1,1) algoritması çözümü uyumludur. Sistemin perturbasyon parametresi  $\varepsilon$  arttıkça sistemin sayısal çözümüne yaklaşma hızının azaldığı gözlemlenmiştir.



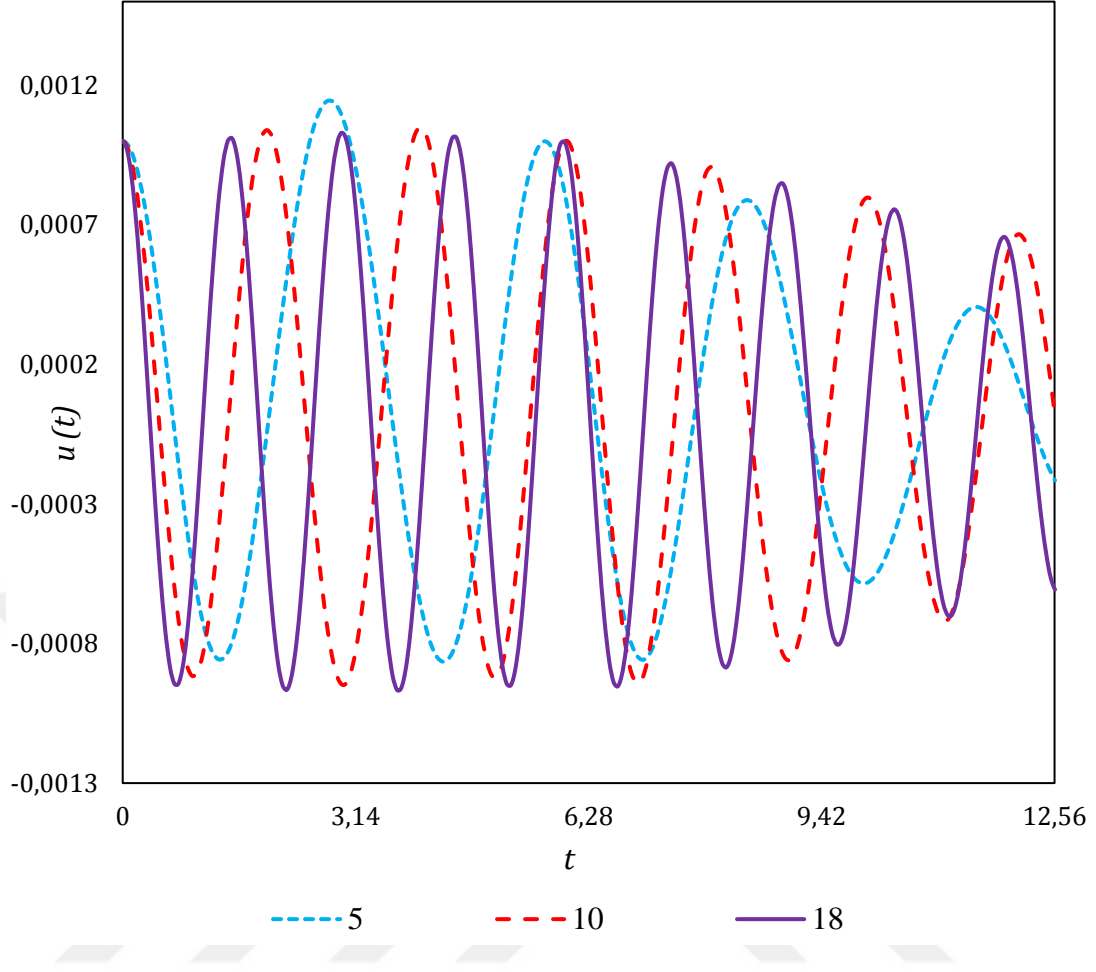
**Şekil 5.4.** Gecikmeli Mathieu denkleminin PIA(1,1) algoritmasının beşinci ve altıncı iterasyon çözümü ile MATLAB (dde23) sayısal çözümünün karşılaştırılması ( $\delta = 18.0$ ,  $\varepsilon = 5$ ,  $\zeta = 0.5$ , ve  $\tau = 2\pi$ )

Mathieu denkleminin periyodik mesnet hareketinin problemdeki teriminin katsayısının sisteme etkisi üç farklı  $\varepsilon$  değeri için incelenmiş ve gecikmeli Mathieu probleminin PIA(1,1) çözümleri Şekil 5.5’de verilmiştir. Periyodik hareketin katsayısının büyümesi ile sistemin belirli periyotlarla genliğinin arttığı gözlemlenmiştir. Burada,  $\delta = 18$ ,  $\zeta = 0.5$  ve  $\tau = 2\pi$  olmak üzere,  $\varepsilon = 1$  ve  $\varepsilon = 5$  için altıncı ve  $\varepsilon = 10$  için sekizinci iterasyon PIA(1,1) algoritması çözümleri sunulmuştur.



**Şekil 5.5.** Gecikmeli Mathieu denkleminin  $\varepsilon = 1, 5$  ve  $10$  değerleri için PIA(1,1) çözümleri ( $\delta = 18, \zeta = 0.5, \tau = 2\pi$ )

Şekil 5.6'da  $\varepsilon = 1, \zeta = 0.5$  ve  $\tau = 2\pi$  olmak üzere, Gecikmeli Mathieu denkleminin farklı  $\delta$  değerleri için PIA(1,1) çözümleri sunulmuştur. Bu değerlerin değişiminin titreşimin periyodunu etkilediği açıkça görülmektedir.



**Şekil 5.6.** Gecikmeli Mathieu denkleminin  $\delta = 5, 10$  ve  $18$  değerleri için PIA(1,1) çözümleri ( $\varepsilon = 1, \zeta = 0.5, \tau = 2\pi$ )

Şekil 5.3’de, gecikmeli Mathieu denkleminin gecikme teriminin katsayısının farklı değerleri için gecikme teriminin etkisi araştırılmış ve gecikme teriminin sisteme sönümleyici etki yaptığı gözlemlenmiştir. Diğer taraftan denklemin gecikme terimi

$$\zeta u(t - \tau) \sim \zeta u(t) - \zeta \tau u'(t)$$

olarak kabul edilirse [1], gecikmeli Mathieu denkleminin gecikme terimi sönüm terimi olarak denkleme yazılır. Gecikme teriminin pozitif değerleri için ( $\zeta > 0$ ), kabul gereği sönümleme artar ve sistemin salınımlarına yavaşlatıcı etki yapar. Bu da elde ettiğimiz sonuçlar ile uyumludur.

Gecikme terimi için sunulan bu kabul genel bir yaklaşım sunsa da problemde gecikme etkisinin tam olarak gözlemlenmesine engel olabilecektir. Bu yüzden sönüm ve gecikme etkisini birlikte incelemek için gecikmeli sönümlü Mathieu denklemini özel olarak inceleyeceğiz. Hem gecikme hem de sönüm etkisinin birlikte inceleneceği bu bölümde gecikme teriminin katsayısının negatif değer alması durumunda sisteme etkisi de incelenecektir.

## 5.2. Gecikmeli Sönümlü Mathieu Denkleminin Perturbasyon-İterasyon Metodu ile Çözümü

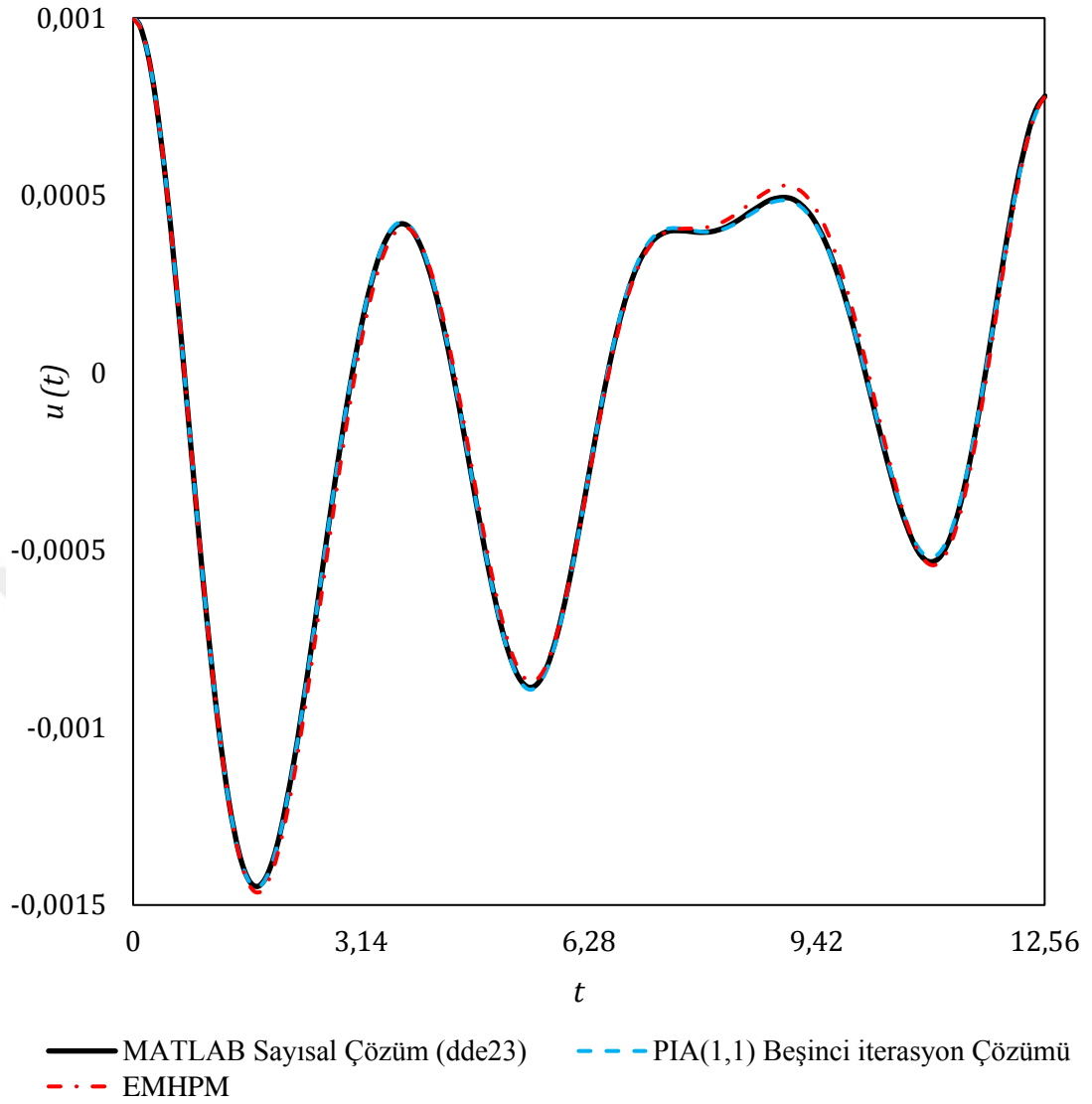
İkinci mertebe lineer gecikmeli diferansiyel denklem olan gecikmeli sönümlü Mathieu denklemi [102, 103]

$$u''(t) + \kappa u'(t) + (\delta + \varepsilon \cos(t))u(t) = \zeta u(t - \tau) \quad (5.15)$$

şeklinde tanımlanmıştır.

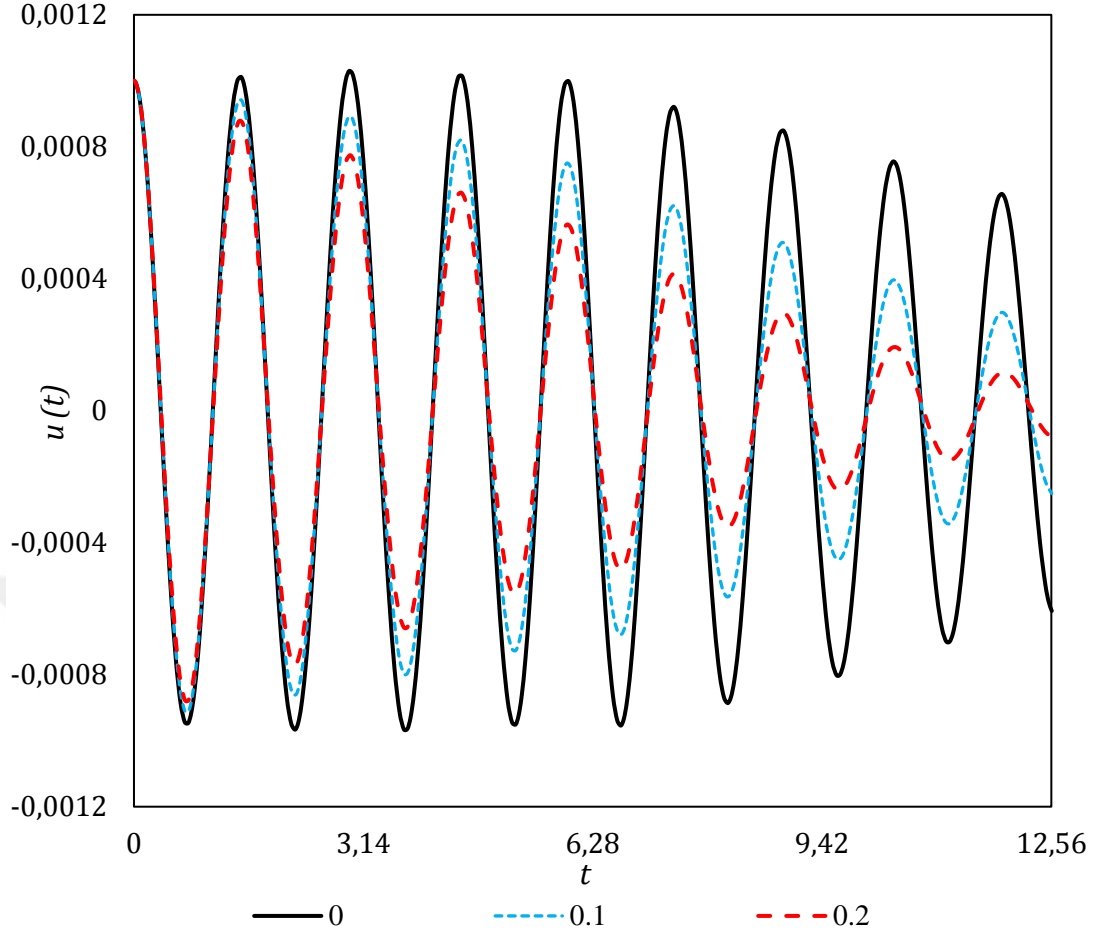
Gecikmeli sönümlü Mathieu denkleminin ikinci iterasyon PIA çözümü EK B.2'de sunulmuştur. PIA(1,1) beşinci iterasyon çözümü ve birinci mertebe gelişmiş çok kademeli homotopi perturbasyon yöntemi (Enhanced Multistage Homotopy Perturbation Method-EMHPM) [104] çözümü ile MATLAB dde23 kodu ile bulunan sayısal sonuçlar Şekil 5.7'de karşılaştırılmıştır. PIA çözümünün EMHPM ile elde edilen çözüme göre sayısal çözümle daha uyumlu olduğu görülmüştür.





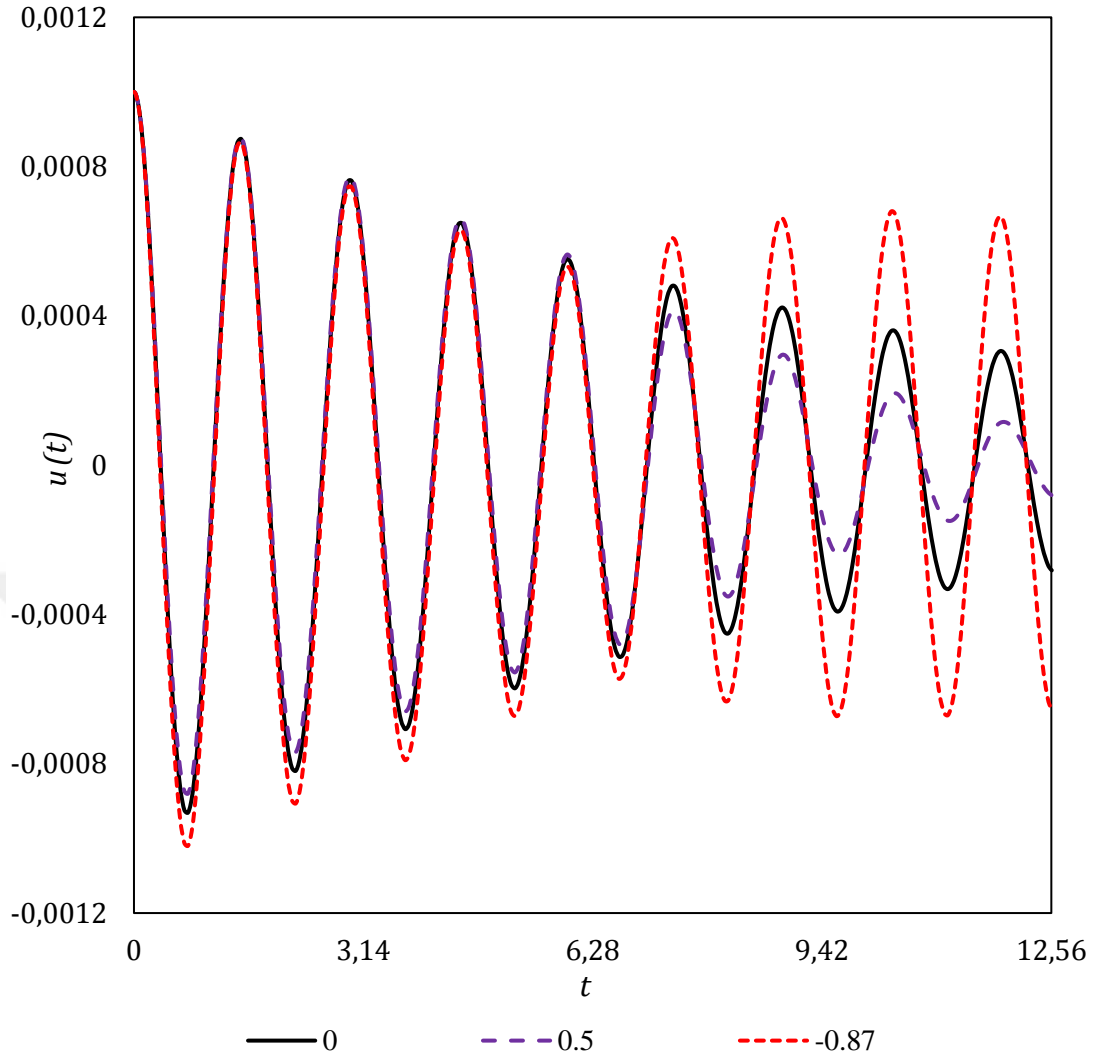
**Şekil 5.7.** Gecikmeli sönümlü Mathieu denkleminin PIA(1,1) algoritmasının beşinci iterasyon çözümü ve birinci mertebe EMHPM çözümü ile sayısal çözümünün karşılaştırılması ( $\kappa = 0.2$ ,  $\delta = 3.0$ ,  $\varepsilon = 1$ ,  $\zeta = -1$ , ve  $\tau = 2\pi$ )

Sönümlü ve gecikmeli Mathieu denkleminde sönüm teriminin sisteme etkisi Şekil 5.8’de farklı sönüm katsayıları ile analiz edilmiştir. Buradan açıkça görülmektedir ki aktif kontrol altında sönüm terimi sistemi sönümlemektedir.



**Şekil 5.8.** Gecikmeli sönümlü Mathieu denkleminin sönüm teriminin katsayısının  $\kappa = 0, 0.1$  ve  $0.2$  değerleri için PIA(1,1) beşinci iterasyon çözümleri ( $\delta = 18, \varepsilon = 1, \tau = 2\pi, \zeta = 0.5$ )

Gecikmeli sönümsüz Mathieu probleminde gecikme teriminin katsayısının pozitif değerleri için sönüm etkisi oluşturduğu gözlemlenmişti. Şekil 5.9’de sönümlü gecikmeli Mathieu denkleminin gecikme teriminin katsayısının pozitif değerleri için aynı etki gözlemlenmiştir. Ayrıca gecikme teriminin negatif değerlerinin sistemin hareketinin belirli bir genlikte kalmasına neden olduğu görülmektedir.



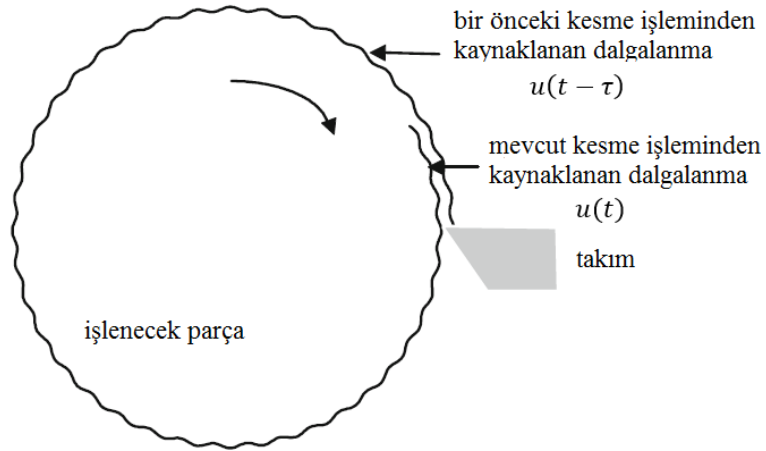
**Şekil 5.9.** Gecikmeli sönümlü Mathieu denkleminin sönüm altında gecikme teriminin katsayısının  $\zeta = 0$ ,  $0.5$  ve  $-0.87$  değerleri için PIA(1,1) dördüncü iterasyon çözümleri ( $\delta = 18$ ,  $\varepsilon = 1$ ,  $\tau = 2\pi$ ,  $\kappa = 0.2$ )

### 5.3. Frezeleme Makinesinde Kesme İşlemi Denkleminin Perturbasyon-İterasyon Metodu ile Çözümü

Artan endüstriyel rekabetin üretici maliyetlerinin düşmesini teşvik etmesiyle üretim sürecinde talaşlı şekillendirme işlemlerinin kullanımı yaygınlaştı [105]. Talaşlı şekillendirme işleminin etkinliği, talaş debisinin oranına bağlı olarak devir süresi, makinanın çalışma süresi ve takım yıpranması gibi temel etkenlere bağlıdır. Bu parametrelerin optimizasyonu önemlidir. Titreşim ise bu optimizasyonu sınırlayan en önemli etkenlerden biri olarak görülmektedir [106]. Yineleme etkisi, takım

tezgâhlarının titreşimini açıklamada kabul görmüş en yaygın nedendir. Bu etki önceki geçişlerin dişli geçişinden dolayı ürettiği dalgali parça ile ilgilidir [60, 106-108]. Bu yineleme etkisi başka bir deyişle gecikme ihmal edilirse, sistemin modeli sönümlü bir salınım problemine dönüşür ve bu sistem kararludur. Bu da problemin doğru analiz edilmesini imkânsız kılar. Bu yüzden ele alınan sistemlerin matematiksel olarak modellenmesi ve elde edilen modelin sayısal çözümü önem kazanmıştır. Bunun sonucunda, birçok araştırmacı bu alanda çalışmakta ve birçok sayısal yöntemi bu problemin matematiksel modeline uygulamaktadır [104, 109].

Makine takımlarının titreşimi kompleks dinamik işlemlerin en önemlilerinden biridir. Makine takımlarının hassasiyeti ve kesme işlemi esnasında oluşan titreşimden dolayı bu titreşimleri açıklamak ve tahmin etmek için birçok model ortaya konmuştur. İşlenecek bir malzemedan talaş şeklinde parça çıkarılma işlemi esnasında meydana gelebilecek küçük bir bozulma, işlenecek parça üzerinde bir sönümlü titreşime neden olur. İşlenecek parçanın yüzeyinde dalgalanmalar meydana gelerek, bir sonraki adımda talaş kalınlığında farklılıklar meydana gelir. Bu işlemin mekanik modeli Şekil 5.10'da verilmiştir. Takımın kesme kuvveti, işlenecek parçadaki ve takımdaki asıl ve gecikmeli yer değiştirme değerlerine bağlı olarak değişeceğinden sistemde gecikme olacaktır [1]. Bu gecikme etkisinden dolayı oluşan zaman gecikmesi olmakta ve takım işlemi (frezeleme işlemi) gecikmeli diferansiyel denklemler ile matematiksel olarak modellenmektedir [60].

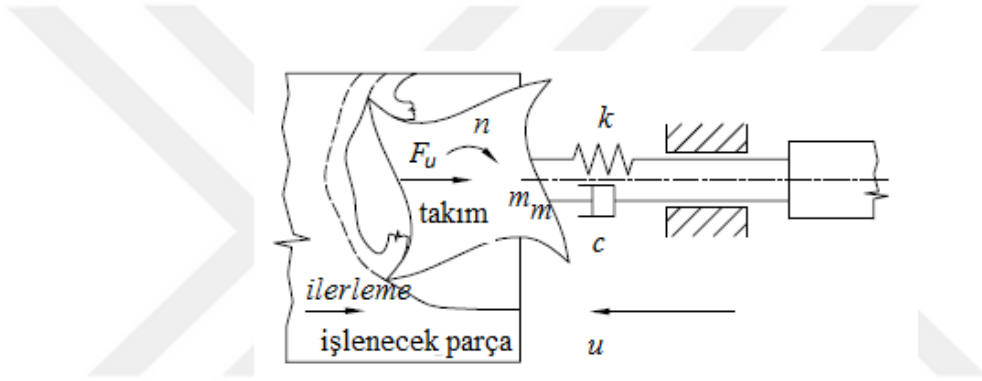


**Şekil 5.10.** Frezeleme işleminde gecikme etkisinin mekanik modeli [1]

Frezeleme işleminin mekanik modeli için Şekil 5.11’de bir şematik diyagram gösterilmiştir. Modelin sadece  $u$  doğrultusunda esnek olduğu kabul edilirse, sistem tek serbestlik dereceli olarak ele alınmış olacaktır. Takıma etki eden kesme kuvvetlerinin toplamından sistemin hareket denklemi

$$u''(t) + 2\zeta\omega_n u'(t) + \omega_n^2 u(t) = \frac{F_u}{m_m} \quad (5.16)$$

olarak elde edilir [106]. Burada  $\omega_n$  sistemin açışal doğal frekansı,  $\zeta$  sönüm katsayısı,  $m_m$  aletin modal kütlesi ve  $F_u$  sıfır helezonlu kesici için  $u$  yönünde kesme kuvvetidir.



Şekil 5.11. Frezeleme işleminin tek serbestlik dereceli mekanik modeli [106]

Frezeleme işleminde kesme kuvveti  $F_u$  ve bileşenleri Şekil 5.12’de sunulmuştur.  $j$ . diş üzerindeki kesme kuvveti  $F_j$ ’nin teğetsel kesme kuvveti bileşeni  $F_{tj}$  ve normal kesme kuvveti bileşeni  $F_{nj}$  olmak üzere,  $u$  doğrultusunda,  $j$ . diş üzerindeki kesme kuvveti

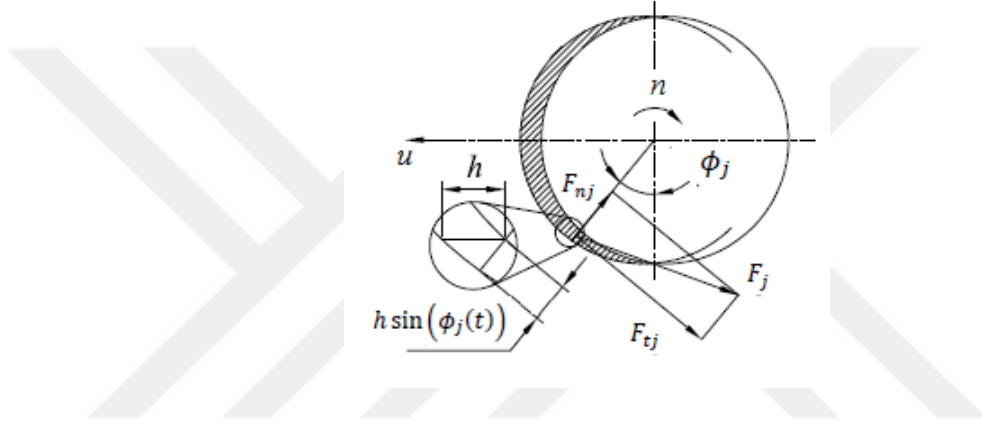
$$F_{uj} = g(\phi_j(t))(-F_{tj} \cos(\phi_j(t)) - F_{nj} \sin(\phi_j(t))) \quad (5.17)$$

olarak bulunur. (5.17)’de tanımlı  $g(\phi_j(t))$  fonksiyonu, anahtarlama fonksiyonudur.  $j$ . diş kestiğinde değeri 1, diğer durumlarda 0 dır [110, 111].

$$g(\phi_j(t)) = \begin{cases} 1 & , \quad \phi_{st} < \phi_j(t) \bmod 2\pi < \phi_{ex} \\ 0 & , \quad \text{diğer} \end{cases} \quad (5.18)$$

Burada  $\phi_{st}$  ve  $\phi_{ex}$  parçanın dişliye giriş ve çıkış açısıdır.

Takımın dönme yönü ve ilerlemesi arasındaki ilişki temel alınarak aşağı frezeleme ve yukarı frezeleme şeklinde iki frezeleme yöntemi tanımlanmaktadır. Her iki frezeleme temelde aynı sonuçlar vermesine karşın dinamikleri birbirinden farklıdır.  $a_d$  kesme oranının açısal derinliği olarak tanımlarsak, sistem  $\phi_{st} = 0$  ve  $\phi_{ex} = \arccos(1 - 2a_d)$  için yukarı frezeleme,  $\phi_{st} = \arccos(2a_d - 1)$  ve  $\phi_{ex} = \pi$  içinse aşağı frezeleme yapmaktadır [104].



Şekil 5.12. Frezeleme işleminde kesme kuvvetinin ( $F_j$ ) bileşenleri [106]

Teğetsel ve normal kesme kuvveti bileşenleri doğrusallaştırılmış kesme katsayıları  $K_t$  ve  $K_n$ 'nin çarpımı olarak kabul edilir.  $a_p$  kesmenin nominal derinliği ve  $w$  ani talaş genişliği olmak üzere

$$F_{tj} = K_t b w_j \quad (5.19)$$

ve

$$F_{nj} = K_n b w_j \quad (5.20)$$

dir.  $w_j(t)$  fonksiyonu, her bir diş için ilerleme büyüklüğü olan  $h$  değerine,  $j$ . dişin açısal konumu olan  $\phi_j(t)$  fonksiyonuna ve yapının esnek yöndeki yinelenmesine bağlı olacak şekilde aşağıdaki gibi yazılır.

$$w_j(t) = h \sin(\phi_j(t)) + [u(t) - u(t - \tau)] \sin(\phi_j(t)) \quad (5.21)$$

burada zaman gecikmesi iki dişli arasındaki geçiş periyoduna eşittir. Yani  $\tau = 60/nz_n$  dir.  $n$  devir/dak cinsinden hız ve  $z_n$  takım diş sayısıdır.

(5.18), (5.19), (5.20) ve (5.21) denklemleri (5.17) denklemine yazılır ve  $z_n$  takım diş sayısı kadar toplanırsa, diş üzerindeki kesme kuvvetinin  $u$  bileşeni

$$F_u(t) = \sum_{j=1}^{z_n} g(\phi_j(t)) [-K_t b \cos(\phi_j(t)) - K_n b \sin(\phi_j(t))] \left\{ h \sin(\phi_j(t)) + [u(t) - u(t - \tau)] \sin(\phi_j(t)) \right\} \quad (5.22)$$

olarak elde edilir [104].

$j$ . dişin açısal konumu

$$\phi_j(t) = \left( \frac{2\pi n}{60} \right) t + \frac{2\pi j}{z_n} \quad (5.23)$$

ve  $\tau$ -periyodik özel kesme kuvvetleri

$$K_s(t) = \sum_{j=1}^{z_n} g(\phi_j(t)) \sin(\phi_j(t)) \left( K_t \cos(\phi_j(t)) + K_n \sin(\phi_j(t)) \right) \quad (5.24)$$

$$f_0(t) = \sum_{j=1}^{z_n} g(\phi_j(t)) h \sin(\phi_j(t)) \left( K_t \cos(\phi_j(t)) + K_n \sin(\phi_j(t)) \right)$$

olarak tanımlanır ve (5.24) kuvvetleri (5.16) denklemde yerine konur ve  $\frac{a_p f_0(t)}{m_m}$  terimi elimine edilirse [106], sistemin hareket denklemi

$$u''(t) + 2\zeta\omega_n u'(t) + \omega_n^2 u(t) = -\frac{a_p K_s(t)}{m_m} [u(t) - u(t - \tau)] \quad (5.25)$$

olarak bulunur [40].

Bölüm 3.3'de sunulan PIA algoritması ile (5.25) denkleminin çözümünü  $[0, 4\tau]$  aralığında bulmak probleminin tanım aralığı  $D = \cup_{j=1}^4 D_j$  olacak şekilde  $D_j$  alt aralıkların birleşimi olarak yazılır. (5.25) denklemini  $\varepsilon$  suni parametresi ile  $D_j$  alt aralığında

$$\begin{aligned} F^{D_j} &= (u^{D_j})''(t) + 2\zeta\omega_n (u^{D_j})'(t) + \omega_n^2 u^{D_j}(t) \\ &\quad + \frac{a_p K_s(t)}{m_m} [\varepsilon u^{D_j}(t) - u^{D_{j-1}}(t - \tau)] = 0, \quad (5.26) \\ & j = 1, \dots, 4 \end{aligned}$$

denkleme dönüştürülür. (5.26) denkleminin  $u_n^{D_j}(t)$ , ( $j = 1, \dots, 4$ ) çözümlerinin bulunması için (3.27) algoritmasının (3.26)'da tanımlanan sıfırdan farklı terimleri

$$\begin{aligned} F^{D_j} &= 1 \\ (u^{D_j})'' &= 2\zeta\omega_n \\ (u^{D_j})' &= \omega_n^2 \\ F_{u^{D_j}}^{D_j} &= \frac{a_p K_s(t)}{m_m} u_n^{D_j}(t) \\ F_{\varepsilon}^{D_j} &= \frac{a_p K_s(t)}{m_m} u_n^{D_j}(t) \\ F^{D_j} &= (u_n^{D_j})''(t) + 2\zeta\omega_n (u_n^{D_j})'(t) + \omega_n^2 u_n^{D_j}(t) \\ &\quad - \frac{a_p K_s(t)}{m_m} u_n^{D_{j-1}}(t - \tau) \end{aligned} \quad (5.27)$$

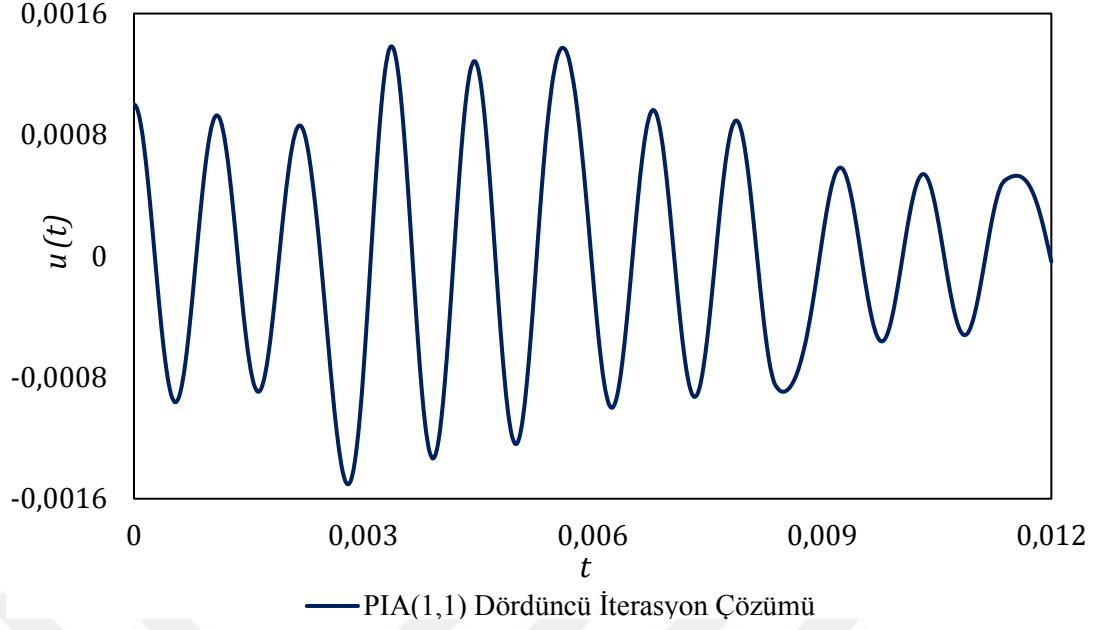
olarak bulunur. (5.27)'deki terimler (3.27) PIA(1,1) algoritmasında yerine konursa;



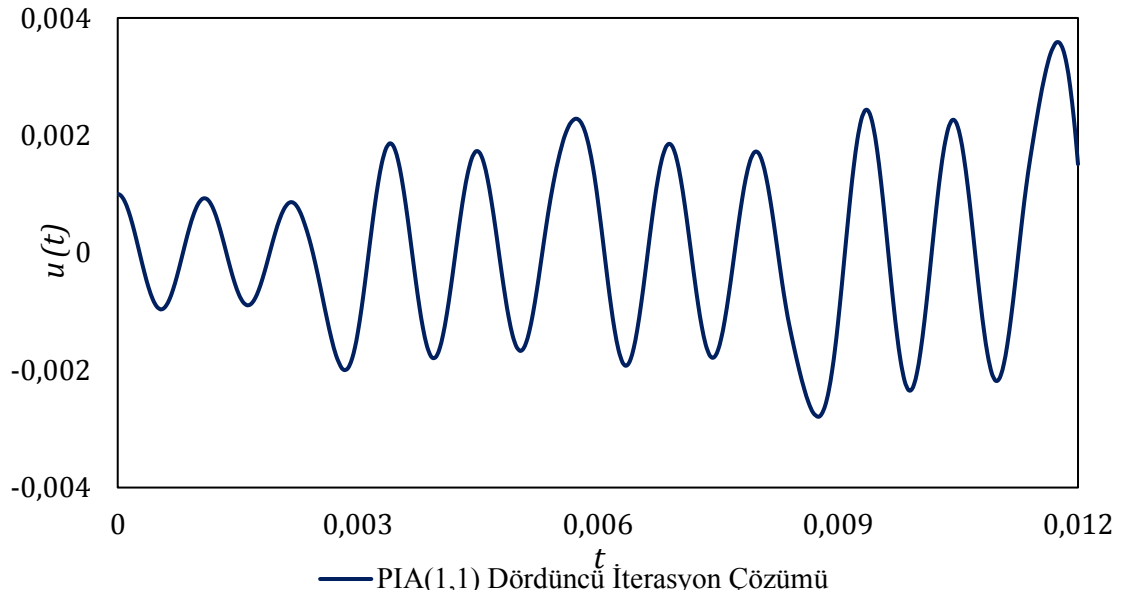
$$\begin{aligned}
& \left( u_c^{Dj} \right)_n''(t) + 2\zeta\omega_n \left( u_c^{Dj} \right)_n'(t) + \omega_n^2 \left( u_c^{Dj} \right)_n \\
& = - \frac{a_p K_s(t)}{m_m} u_n^{Dj}(t) \\
& - \left( \left( u_n^{Dj} \right)''(t) + 2\zeta\omega_n \left( u_n^{Dj} \right)'(t) + \omega_n^2 u_n^{Dj}(t) \right. \\
& \left. - \frac{a_p K_s(t)}{m_m} u_n^{Dj-1}(t - \tau) \right) / \varepsilon
\end{aligned} \tag{5.28}$$

diferansiyel denklemi elde edilir. (5.28) denklemi, (5.25)'un  $j$ . alt aralığındaki çözümlerini verir.

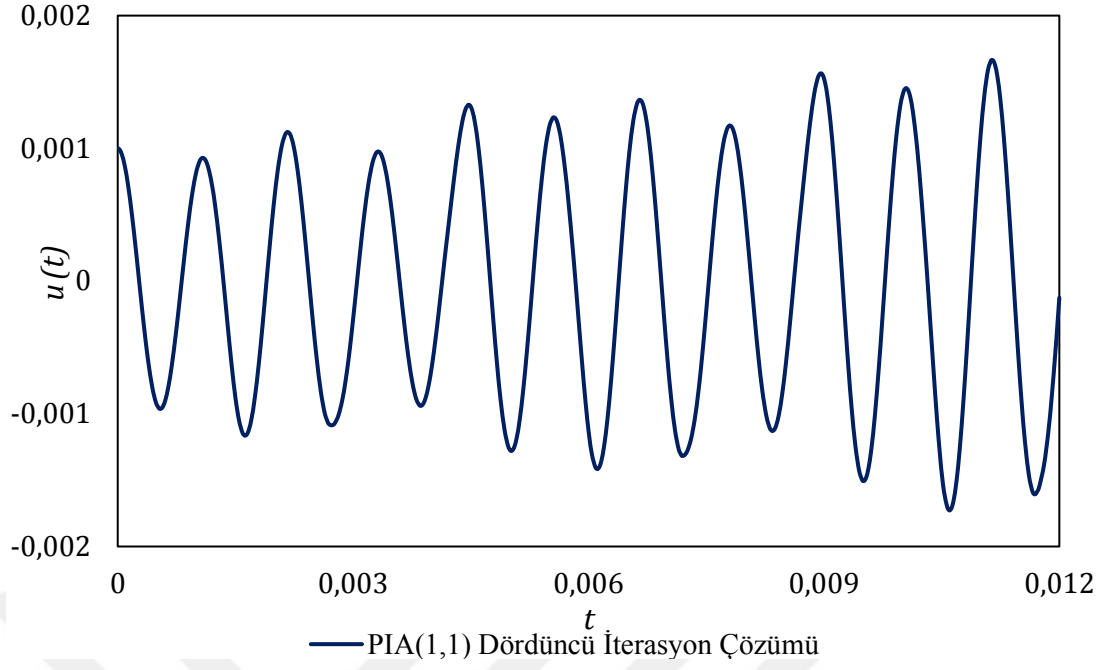
Aşağı frezeleme durumu için parametrelerin değerleri  $z_n = 2$ ,  $a_d = 0.1$ ,  $\omega_n = 5793$  rad/s,  $\zeta = 0.011$ ,  $m_m = 0.03993$ kg,  $K_t = 6 \times 10^8$ N/m<sup>2</sup> ve  $K_n = 2 \times 10^8$ N/m<sup>2</sup> olarak alınmıştır. Ayrıca, sistemin geçmiş fonksiyonu  $h(t) = 0.001$ 'dir. (5.25) denkleminin (5.28) PIA algoritması ile  $a_p = 2$ mm ve  $n = 10000$  rpm için dördüncü iterasyon çözümü Şekil 5.13'de verilmiştir. Burada sistem kontrol altında sönümlenmektedir. Hızın aynı, kesme derinliğinin  $a_p = 3$ mm olması durumunda sistemin sönümlenmediği ve parça deformasyonunun arttığı Şekil 5.14'de PIA(1,1) dördüncü iterasyon çözümünde görülmektedir. Hızın aynı  $n = 20000$  rpm, kesme derinliğinin  $a_p = 2$ mm olması durumunda da sistemin sönümlenmediği ve parça deformasyonunun arttığı Şekil 5.15'de PIA(1,1) beşinci iterasyon çözümünde görülmektedir.



**Şekil 5.13.** Denklem (5.25)'in  $a_p = 2\text{mm}$  ve  $n = 10000$  rpm için PIA(1,1) algoritması ile dördüncü iterasyon çözümü



**Şekil 5.14.** Denklem (5.25)'in  $a_p = 3\text{mm}$  ve  $n = 10000$  rpm için PIA(1,1) algoritması ile dördüncü iterasyon çözümü



**Şekil 5.15.** Denklem (5.25)'in  $a_p = 2\text{mm}$  ve  $n = 20000$  rpm için PIA(1,1) algoritması ile dördüncü iterasyon çözümü

## 6. SONUÇ VE ÖNERİLER

Fonksiyonel diferansiyel denklemlerin bir sınıfı olan gecikmeli diferansiyel denklemler için perturbasyon-iterasyon metodu geliştirilmiştir. Perturbasyon açılımındaki düzeltme teriminin sayısı bir ve Taylor seri açılımındaki birinci mertebe ve ikinci mertebe türevlerine göre PIA(1,1) ve PIA(1,2) perturbasyon iterasyon algoritmaları elde edilmiştir. Yöntemin küçük perturbasyon kısıtlaması olmadığından, gecikmeli diferansiyel denklemlerin alt sınıfı olan zaman gecikmeli diferansiyel denklemler, pantograf tipi gecikmeli diferansiyel denklemler ve geçmiş fonksiyonuna dayalı zaman gecikmeli diferansiyel denklemler için perturbasyon-iterasyon algoritmaları geliştirilebilmiştir. Başka bir deyişle, keyfi perturbasyon parametresi kullanılabilmesi yöntemin önemli bir avantajı olmaktadır. Ayrıca elde edilen çözümlerin fonksiyon olması yöntemin başka bir avantajıdır. Küçük iterasyon değerleri için uygun çözümler bulunsa da iterasyon arttıkça çözüm fonksiyonlarının terim sayısının hızla büyümesi bazı problemleri çözümlerinin karmaşık sayı yada fonksiyonlar içermesine sebep olmaktadır. Bunun için seküler terimlerin ayıklanması gerekliliği ortaya çıkmaktadır.

Yöntem gecikmeli diferansiyel denklemlerin ele alınan tüm alt sınıflarına başarıyla uygulanmıştır. İncelenen tüm sayısal örneklerde iterasyon sayısının artması ile maksimum mutlak hata değerlerinin azaldığı gözlemlenmiştir. Perturbasyon iterasyon çözümleri literatürde bilinen birçok yöntem ile karşılaştırılmış ve elde edilen sonuçların genellikle daha iyi olduğu gözlemlenmiştir. Yüksek mertebe, değişken katsayılı/sabit katsayılı veya lineer/lineer olmayan birçok problem için de etkin sonuçlar elde edilmiştir.

Literatürde sıkça karşılaşılan zaman gecikmeli diferansiyel denklemler için geliştirilen perturbasyon-iterasyon algoritmaları değişken katsayılı yüksek mertebeli üç farklı sayısal örneğe uygulanmış ve bilinen bir iterasyon yöntemi ile karşılaştırılmıştır.

Pantograf tipi gecikmeli diferansiyel denklemler için geliştirilen algoritmalar ise altı farklı sayısal örneğe uygulanmıştır [54]. İki lineer ve dört lineer olmayan

altı farklı problem ele alınmış, literatürde bilinen birçok yönteme göre daha iyi sonuçlar elde edilmiştir.

Mühendislik problemlerinin matematiksel modeli olarak ortaya çıkan geçmiş fonksiyonuna dayalı gecikmeli diferansiyel denklemler için perturbasyon-iterasyon algoritması geliştirilmiş ve üç farklı sayısal örneğe uygulanmıştır. Tüm bu sayısal örneklerin perturbasyon-iterasyon çözümleri grafik ve tablolar ile sunulmuş ve birçok bilinen yöntem ile karşılaştırılmıştır.

Geliştirilen algoritma ile fiziksel sistemlerin matematiksel modeli olan gecikmeli Mathieu ve gecikmeli sönümlü Mathieu denklemleri çözülerek denklemdaki gecikme teriminin etkisi ortaya konmuştur. Ayrıca Frezeleme kesme işleminin matematiksel denklemine de uygulanan yöntem ile sistemin kararlı ve kararsız çözümleri elde edilmiştir.

Geçmiş fonksiyonuna dayalı lineer olmayan gecikmeli diferansiyel denklemler için perturbasyon-iterasyon yöntemi geliştirilebilir. Ele alınan mekanik sistemlerin lineer olmayan matematiksel modelleri geliştirilecek perturbasyon-iterasyon algoritmaları ile analiz edilebilir. Sistemlerin kararlılık analizleri için gelecek çalışmalarda yöntem uygulanabilir.

## KAYNAKLAR

- [1] Erneux, T. Applied Delay Differential Equations. Springer, Brüksel, Belçika, 2009.
- [2] Shampine, L. F., Thompson S. Numerical Solution of Delay Differential Equations, Delay Differential Equations. Ed.: Gilsinn, D. E., Nagy, T K., Balachandran B, Springer, US, 2009, 1-27.
- [3] Insperger, T., Stépán G. Semi-discretization method for delayed systems. International Journal for Numerical Methods in Engineering. 2002, 55(5), 503-518.
- [4] Asl, F. M., Ulsoy A. G. Analysis of a system of linear delay differential equations. Transactions-American Society of Mechanical Engineering Journal of Dynamic Systems Measurement and Control. 2003, 125(2), 215-223.
- [5] Bocharov, G. A., Rihan, F. A. Numerical modelling in biosciences using delay differential equations. Journal of Computational and Applied Mathematics. 2000, 125(1), 183-199.
- [6] Liu, H., Xiao A., Su L. Convergence of variational iteration method for second-order delay differential equations. Journal of Applied Mathematics. 2013, DOI: 10.1155/2013/634670.
- [7] Anakira, N. R., Alomari A. K., Hashim, I. Optimal homotopy asymptotic method for solving delay differential equations. Mathematical Problems in Engineering. 2013, DOI: 10.1155/2013/498902.
- [8] Dehghan, M., Salehi, R. Solution for a nonlinear time-delay model in biology via semi-analytical approaches. Computer Physics Computations. 2010, 181, 1255-1265.
- [9] Xu, X., Hu, H. Y., Wang H. L. Stability, bifurcation and chaos of delayed oscillator with negative damping and delayed feedback control. Nonlinear Dynamics. 2007, 49, 117-129.
- [10] Saeed, R. K., Rahman, B. M. Adomian decomposition method for solving system of delay differential equations. Australian Journal of Basic and Applied Sciences. 2010, 4 (8), 3613-3621.
- [11] Evans, D. J., Raslan K. R. The Adomian decomposition method for solving delay differential equations. International Journal of Computer Mathematics. 2005, 82 (1), 49-54.
- [12] Vanani, S. K., Aminataei, A. On the numerical solution of neutral delay differential equations using multiquadric approximation scheme. Bulletin of the Korean Mathematical Society. 2008, 45 (4), 663-670.
- [13] Yu, Z. H. Variational iteration method for solving the multi-pantograph delay equation. Physics Letters A. 2008, 372 (43), 6475-6479.

- [14] Wang, Z. Q., Wang, L. L. A Legendre-Gauss collocation method for nonlinear delay differential equations. *AIMS Discrete and Continuous Dynamical Systems-B*. 2010, 13(3), 685-708.
- [15] Gülsu, M., Sezer, M. A method for the approximate solution of the high-order linear difference equations in terms of Taylor polynomials. *International Journal of Computer Mathematics*. 2005, 82(5), 629-642.
- [16] Gülsu, M., Sezer, M. A Taylor polynomial approach for solving differential-difference equations. *Journal of Computational and Applied Mathematics*. 2006, 186(2), 349-364.
- [17] Gülsu, M., Sezer, M., Tanay, B. A matrix method for solving high-order linear difference equations with mixed argument using hybrid Legendre and Taylor polynomials. *Journal of the Franklin Institute*. 2006, 343(6), 647-659.
- [18] Gülsu, M., Yalman, H., Öztürk, Y., Sezer, M. A new Hermite Collocation method for solving differential difference equations. *Applications and Applied Mathematics*. 2011, 6(1), 116-129.
- [19] Gülsu, M., Öztürk, Y. ve Sezer, M. A new Chebyshev polynomial approximation for solving delay differential equations. *Journal of Difference Equations and Applications*. 2012, 18(6), 1043-1065.
- [20] Yüzbaşı, Ş. A numerical approach for solving the high-order linear singular differential–difference equations. *Computers & Mathematics with Applications*. 2011, 62(5), 2289-2303.
- [21] Yüzbaşı, Ş. On the solutions of a system of linear retarded and advanced differential equations by the Bessel collocation approximation. *Computers & Mathematics with Applications*. 2012, 63(10), 1442-1455.
- [22] Gökmen, E., Sezer, M. Taylor collocation method for systems of high-order linear differential–difference equations with variable coefficients. *Ain Shams Engineering Journal*. 2013, 4(1), 117-125.
- [23] Kurt, A., Yalçınbaş, S., Sezer, M. Fibonacci collocation method or solving linear differential-difference equations. *Mathematical and Computational Applications*. 2013, 18(3), 448-458.
- [24] Erdem, K., Yalçınbaş S. Bernoulli polynomial approach to high-order linear differential-difference equations. *AIP conference proceedings*. 2012 1479(1).
- [25] Yüzbaşı, Ş., Sezer, M. Shifted Legendre approximation with the residual correction to solve pantograph-delay type differential equations, *Applied Mathematical Modelling*. 2015, 39(21), 6529-6542.
- [26] Yüzbaşı, Ş., Sezer, M. Exponential collocation method for solutions of singularly perturbed delay differential equations. *Abstract and Applied Analysis*. 2013, DOI: 10.1155/2013/493204.

- [27] Yüzbaşı, Ş., Gök, E., Sezer, M. Laguerre matrix method with the residual error estimation for solutions of a class of delay differential equations. *Mathematical Methods in the Applied Sciences*. 2014, 37(4), 453-463.
- [28] Yüzbaşı, Ş., Sezer, M. An exponential approximation for solutions of generalized pantograph-delay differential equations. *Applied Mathematical Modelling*. 2013, 37(22), 9160-9173.
- [29] Çevik, M., Bahşı, M. M., Sezer, M. Solution of the delayed single degree of freedom system equation by exponential matrix method. *Applied Mathematics and Computation*. 2014, 242, 444-453.
- [30] Bahşı, M. M., Çevik, M., Sezer, M. Orthoexponential polynomial solutions of delay pantograph differential equations with residual error estimation. *Applied Mathematics and Computation*. 2015, 271, 11-21.
- [31] Wang, H., Hu H. Remarks on the perturbation methods in solving the second-order delay differential equations. *Nonlinear Dynamics*. 2003, 33(4), 379-398.
- [32] Shakeri, F., Dehghan, M. Solution of delay differential equations via a homotopy perturbation method. *Mathematical and Computer Modelling*. 2008, 48(3-4), 486-498.
- [33] Biazar, J., Ghanbari, B. The homotopy perturbation method for solving neutral functional–differential equations with proportional delays. *Journal of King Saud University-Science*. 2012, 24(1), 33-37.
- [34] Nayfeh, A. H. *Introduction to Perturbation Techniques*. John Wiley and Sons, New York, USA, 1981, 533.
- [35] Pakdemirli, M., Karahan, M. M. F., Boyacı, H. A new perturbation algorithm with better convergence properties: Multiple Scales Lindstedt Poincare method. *Mathematical and Computational Applications*. 2009, 14(1), 31-44.
- [36] He, H. J. Iteration perturbation method for strongly nonlinear oscillations. *Journal of Vibration Control*. 2001, 7, 631-642.
- [37] Mickens, R. E., Iteration method solutions for conservative and limit-cycle  $x^{1/3}$  force oscillators. *Journal of Sound and Vibrations*. 2006, 292, 964-968.
- [38] Pakdemirli M., Review of the new perturbation-iteration method. *Mathematical and Computational Applications*. 2013, 18(3), 139-151.
- [39] Öziş T., Yıldırım A. Generating the periodic solutions for forcing van der Pol oscillators by the iteration perturbation method. *Nonlinear Analysis: Real World Application*. 2009, 10, 1984-1989.
- [40] Ganji, D. D., Karimpour, S., Ganji S. S. He's iteration perturbation method to nonlinear oscillations of mechanical systems with single-degree-of freedom. *International Journal of the Modern Physics B*. 2009, 23, 2469-2477.



- [41] Marinca, V., Herisanu N. A modified iteration perturbation method for some nonlinear oscillation problems. *Acta Mechanica*. 2006, 184, 231-242.
- [42] Pakdemirli, M., Boyacı H. Generation of root finding algorithms via perturbation theory and some formulas. *Applied Mathematics and Computation*, 2007, 184, 783-788.
- [43] Pakdemirli, M., Boyacı, H., Yurtsever H. A. Perturbative derivation and comparisons of root-finding algorithms with fourth order derivatives. *Mathematical and Computational Applications*. 2007, 12(2), 117-124.
- [44] Pakdemirli, M., Boyacı H., Yurtsever H. A. A root finding algorithm with fifth order derivatives. *Mathematical and Computational Applications*. 2008, 13, 123-128.
- [45] Aksoy, Y., Pakdemirli M. New perturbation-iteration solutions for Bratu-type equations *Computers & Mathematics with Applications*. 2010, 59(8), 2802-2808.
- [46] Dolapçı İ. T., Şenol M., Pakdemirli M. New perturbation iteration solutions for Fredholm and Volterra integral equations. *Journal Applied Mathematics*. 2013, DOI: 10.1155/2013/682537.
- [47] Yıldız, V., Pakdemirli, M., Aksoy, Y. Parallel Plate Flow of a Third-Grade Fluid and a Newtonian Fluid with Variable Viscosity. *Zeitschrift für Naturforschung A*. 2016, 71(7), 595-606.
- [48] Aksoy, Y., Pakdemirli. M. Perturbation iteration method solutions of a nonlinear fin equation. *AIP Conference Proceedings*. AIP Publishing .2016 1738(1).
- [49] Bahşı, M. M., Çevik, M. Troesch Probleminin Perturbasyon İterasyon Yöntemi ile Analizi. XX. Ulusal Mekanik Kongresi. 5-9 Eylül, 2017, Bursa, Türkiye. (Bildiri Kitabı 722-726).
- [50] Aksoy, Y., Göktaş, Ü., Pakdemirli, M., Dolapçı, İ. T. Application of perturbation–iteration method to Lotka–Volterra equations. *Alexandria Engineering Journal*. 2016, 55(2), 1661-1666.
- [51] Bildik, N., Deniz, S. Comparative study between optimal homotopy asymptotic method and perturbation-iteration technique for different types of nonlinear equations. *Iranian Journal of Science and Technology, Transactions A: Science*. 2018, 42(2), 647-654.
- [52] Bildik, N., Deniz, S. New approximate solutions to the nonlinear Klein-Gordon equations using perturbation iteration techniques. *Discrete & Continuous Dynamical Systems-S*. 2019, 83-87.
- [53] Bildik, N., Deniz, S. Solving the burgers' and regularized long wave equations using the new perturbation iteration technique. *Numerical Methods for Partial Differential Equations*. 2018, 34(5), 1489-1501.

- [54] Bahşı M. M., Çevik M. Numerical solution of pantograph-type delay differential equations using perturbation-iteration algorithms. *Journal of Applied Mathematics*. 2015, DOI: 10.1155/2015/139821.
- [55] Kovacic, I., Richard, R., Sah, S. M. Mathieu's equation and its generalizations: overview of stability charts and their features. *Applied Mechanics Reviews*. 2018, 70(2), 1-22.
- [56] Kolmanovskii, V., Myshkis, A. *Introduction to the Theory and Applications of Functional Differential Equations*. Springer Science & Business Media, Moscow, Russia, 2013, 643 s.
- [57] Myshkis, A. D. *Linear Differential Equations with Delaying Argument*. Nauka, Moscow, Russia, 1972, s.
- [58] Kolmanovskii, V. B., Nosov, V. R. *Stability of Functional Differential Equations*. Elsevier, Moskow, Russia, 1986, 217s.
- [59] Gorelik, G., To the theory of feedback with delay, *Journal of Technical Physics*. 1939, 9(5), 450-454.
- [60] Stépán, G. Delay-differential equation models for machine tool chatter. *Dynamics and Chaos in Manufacturing Processes*. 1998, 471152935, 165-192.
- [61] Sedaghat, S., Ordokhani, Y., Dehghan, M. Numerical solution of the delay differential equations of pantograph type via Chebyshev polynomials. *Communications in Nonlinear Science and Numerical Simulation*. 2012, 17(12), 4815-4830.
- [62] Salvat, N., Batailly A., Legrand M. Modeling of abradable coating removal in aircraft engines through delay differential equations. *Journal of engineering for gas turbines and power*. 2013, 135(10), 102102.
- [63] Olgac, N., Zalluhoglu U., Kammer A. S. On blade/casing rub problems in turbomachinery: An efficient delayed differential equation approach. *Journal of Sound and Vibration*. 2014, 333(24), 6662-6675.
- [64] Zhang, C., Li, T., Sun, B., Thandapani, E. On the oscillation of higher-order half-linear delay differential equations. *Applied Mathematics Letters*. 2011, 24(9), 1618-1621.
- [65] Zhang, C., Agarwal, R. P., Bohner, M., Li, T. New results for oscillatory behavior of even-order half-linear delay differential equations. *Applied Mathematics Letters*. 2013, 26(2), 179-183.
- [66] Saadatmandi, A., Dehghan, A. Variational iteration method for solving a generalized pantograph equation. *Computational Mathematics with Applications*. 2009, 58(11-12), 2190-2196.

- [67] Liu M. Z., Li D. Properties of analytic solution and numerical solution of multi-pantograph equation. *Applied Mathematics and Computation*. 2004, 155(3), 853-871.
- [68] Muroya, Y., Ishiwata, E., Brunner, H. On the attainable order of collocation methods for pantograph integro-differential equations. *Journal of Computational and Applied Mathematics*. 2003, 152(1-2), 347-366.
- [69] Yalçınbaş, S., Aynigül, M., Sezer, M. A collocation method using Hermite polynomials for approximate solution of pantograph equations. *Journal of the Franklin Institute*. 2011, 348(6), 1128-1139.
- [70] Sezer M., Akyüz Daşcıoğlu A. A Taylor method for numerical solution of generalized pantograph equations with linear functional argument. *Journal of Computational and Applied Mathematics*. 2007, 200(1), 217-225.
- [71] Trif, D. Direct operational tau method for pantograph-type equations. *Applied Mathematics and Computation*. 2012, 219(4), 2194-2203.
- [72] Brahim, B., Vazquez-Leal, H., Hernandez-Martinez, L. Procedure for exact solutions of nonlinear pantograph delay differential equations. *British Journal of Mathematics & Computer Science*. 2014, 4(199), 2738-2751.
- [73] Bica, A. M., Curila, M., Curila, S. About a numerical method of successive interpolations for two-point boundary value problems with deviating argument. *Applied Mathematics and Computation*. 2011, 217(19), 7772-7789.
- [74] Zahoor, R. M. A. Numerical treatment for boundary value problems of pantograph functional differential equation using computational intelligence algorithms. *Applied Soft Computing*. 2014, 24, 806-821.
- [75] Liu, Yunkang. Numerical investigation of the pantograph equation. *Applied Numerical Mathematics* 24.2-3 (1997): 309-317.
- [76] Martin, J. A., García, O. Variable multistep methods for higher-order delay differential equations. *Mathematical and Computer Modelling*. 2002, 36(7-8), 805-820.
- [77] Zwillinger D., *Handbook of Differential Equations*, Academic Press, Boston, USA, 1989 172s.
- [78] Hu, H. A classical perturbation technique which is valid for large parameters. *Journal of Sound Vibration*. 2004, 269, 409-412.
- [79] He, J. H. Linearized perturbation technique and its applications to strongly nonlinear oscillators. *Computers and Mathematics with Applications*. 200, 345, 1-8.
- [80] He, J. H. Homotopy perturbation method: A new nonlinear analytical technique. *Applied Mathematics and Computation*. 2003, 135, 73-79.

- [81] He, J. H. Some asymptotic methods for strongly nonlinear equations. *International Journal of Modern Physics B*. 2006, 20, 1141–1199.
- [82] Lim, C. W., Wu B. S. A modified Mickens procedure for certain non-linear oscillators. *Journal of Sound and Vibration*. 200, 257, 202–206.
- [83] He, J. H. A simple perturbation approach to Blasius equation. *Applied Mathematics and Computation*. 2003, 140, 217–222.
- [84] Li, P. S., Wu, B. S. An iteration approach to nonlinear oscillations of conservative single-degree-of-freedom systems. *Acta Mechanica*. 2004, 170, 69–75.
- [85] He, J. H. Limit cycle and bifurcation of nonlinear problems. *Chaos, Solitons and Fractals*. 2005, 26, 827–833.
- [86] Mickens, R. E. A generalized iteration procedure for calculating approximations to periodic solutions of “truly nonlinear oscillators”. *Journal of Sound and Vibration*. 2005, 287, 1045–1051.
- [87] Hu, H. Solutions of the Duffing-harmonic oscillator by an iteration procedure. *Journal of Sound and Vibration*. 2006, 298, 446–452.
- [88] Doha, E. H., Bhrawy, A. H., Baleanu, D., Hafez, R. M. A new Jacobi rational–Gauss collocation method for numerical solution of generalized pantograph equations. *Applied Numerical Mathematics*. 2014, 77, 43-54.
- [89] Sezer, M., Yalçınbaş, S., Gülsu, M. A Taylor polynomial approach for solving generalized pantograph equations with nonhomogenous term. *International Journal Computational Mathematics*. 2008, 85, 1055-1063.
- [90] Li, X. Y., Wu, B. Y. A continuous method for nonlocal functional differential equations with delayed or advanced arguments. *Journal of Mathematical Analysis Application*. 2014, 409, 485-493.
- [91] Yüzbaşı, Ş., Gök, E., Sezer, M. Laguerre matrix method with the residual error estimation for solutions of a class of delay differential equations. *Mathematical Method in the Applied Sciences*. 2014, 37(4), 453-463.
- [92] Rao G. P., Palanisamy, K. R. Walsh stretch matrices and functional differential equations. *IEEE Transactions on Automatic Control*. 1982, 27(1), 272-276.
- [93] Hwang, C. Solution of a functional differential equation via delayed unit step functions. *International Journal Systems Science*. 1983, 14(9), 1065–1073.
- [94] Shampine, L. F. Dissipative Approximations to Neutral DDEs. *Applied Mathematics & Computation*. 2008, 203(2), 641–648.
- [95] Anakira, R. Alomari, A. K., Hashim, I. Optimal Homotopy Asymptotic Method for solving delay differential equations. *Mathematical Problem in Engineering*. 2013, DOI: 10.1155/2013/498902.

- [96] Alomari, A. K., Noorani, M. S. M., Nazar, R. Solution of delay differential equation by means of homotopy analysis method. *Acta Applicandae Mathematicae*. 2009, 108(2), 395–412.
- [97] Liu, H., Xiao, A., Lihong, S. Convergence of variational iteration method for second-order delay differential equations. *Journal Applied Mathematics*. 2013, DOI:10.1155/2013/634670.
- [98] Falbo, C. E. Analytic and numerical solutions to the delay differential equations. Joint Meeting of the Northern and Southern California Sections of the MAA. 1995.
- [99] Seong, H. Y., Majid, Z. A., Ismail, F. Solving second-order delay differential equations by direct Adams-Moulton method. *Mathematical Problems in Engineering*. 2013, DOI: 10.1155/2013/261240.
- [100] Rand, R. H. *Lecture Notes on Nonlinear Vibrations (Version 53)*. Cornell University. Ithaca, NY, 2017, <http://dspace.library.cornell.edu/handle/1813/28989>.
- [101] Mathieu, E. Memoire sur Le Mouvement Vibratoire d'une Membrane de forme Elliptique. *Journal of Mathematics Pures Applied*. 1868, 13, 137–203.
- [102] Insperger, T., Stépán, G. Stability chart for the delayed Mathieu equation. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*. 2002, 458(2024), 1989-1998.
- [103] Insperger, T., Stepan, G. Stability of the damped Mathieu equation with time delay. *Journal of Dynamic Systems, Measurement, and Control*. 2003, 125(2), 166-171.
- [104] Olvera, D., Elías-Zúñiga, A., López de Lacalle, L. N., Rodríguez, C. A. Approximate solutions of delay differential equations with constant and variable coefficients by the enhanced multistage homotopy perturbation method. *Abstract and Applied Analysis*. 2015, DOI: 10.1155/2015/382475.
- [105] Tlusty, J. *Manufacturing Processes and Equipment*. Prentice-Hall. 2000, 928s.
- [106] Insperger, T., Mann, B. P., Stépán, G., Bayly, P. V. Stability of up-milling and down-milling, part 1: alternative analytical methods. *International journal of Machine Tools and Manufacture*. 2003, 43(1), 25-34.
- [107] Segalman D. J., Butcher E. A. Suppression of regenerative chatter via impedance modulation. *Journal of Vibration and Control*. 2000. 6(2) 243-256.
- [108] Stépán, G. Modelling nonlinear regenerative effects in metal cutting. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. 2001, 359(1781), 739-757.
- [109] Insperger, T., Lehotzky, D., Stepan, G. Regenerative delay, parametric forcing and machine tool chatter: A review. *IFAC-PapersOnLine*. 2015, 48(12), 322-327.

[110] Minis, I., Yanushevsky. R. A new theoretical approach for the prediction of machine tool chatter in milling. *Journal of engineering for industry*. 1993, 115(1), 1-8.

[111] Budak, E. Analytical prediction of chatter stability in milling—part I: general formulation. *Journal of dynamic systems, measurement, and control*. 1998, 120(1), 22-30.

[110] Butcher, E., Brian M. Stability analysis and control of linear periodic delayed systems using Chebyshev and temporal finite element methods. *Delay Differential Equations*. Springer, Boston, MA, 2009. 93-129s.



## EKLER

### EK A. (Bazı Problemlerin MATLAB Kodları ve Çıktıları)

#### EK A.1. (Problem (2.11)'in MATLAB Kodları ve Çıktıları)

Geçmiş fonksiyonuna dayalı (2.11) gecikmeli diferansiyel denkleminin (2.12) koşulu altındaki analitik çözümünü adım yöntemi [1] ile bulan MATLAB kodu ve çıktıları.

```
Editor
%Problem DDE
syms t y(t)
%%%%% Problem Parametreleri %%%%%
k=-1;
tau=1;      % Gecikme
J=10;      % Aralık Sayısı
h(t)=1+0*t; % Geçmiş fonksiyonu

%%%%% Adım Yöntemi %%%%%
for j=1:J
    f(t)=dsolve(diff(y,1)==k*h(t-tau), y((j-1)*tau)==h((j-1)*tau));
    h(t)=f(t);
    fprintf(['[', num2str((j-1)*tau), ', ', num2str(j*tau), ']', ' aralığındaki çözüm;
    '\n'])
    disp(f(t));
end
```

```
Command Window
ProblemDelay
[0,1) aralığındaki çözüm; 1 - t

[1,2) aralığındaki çözüm; (t*(t - 4))/2 + 3/2

[2,3) aralığındaki çözüm; 17/6 - (t*(t^2 - 9*t + 24))/6

[3,4) aralığındaki çözüm; t^4/24 - (2*t^3)/3 + (15*t^2)/4 - (17*t)/2 + 149/24

[4,5) aralığındaki çözüm; - t^5/120 + (5*t^4)/24 - 2*t^3 + (109*t^2)/12 - (115*t)/6 + 1769/120

[5,6) aralığındaki çözüm; t^6/720 - t^5/20 + (35*t^4)/48 - (197*t^3)/36 + (1061*t^2)/48 - (1085*t)/24 + 26239/720

[6,7) aralığındaki çözüm; - t^7/5040 + (7*t^6)/720 - t^5/5 + (107*t^4)/48 - (521*t^3)/36 + (13081*t^2)/240 - (13201*t)/120 + 463609/5040
```

[7,8) aralığındaki çözüm;  $t^8/40320 - t^7/630 + (7*t^6)/160 - (487*t^5)/720 + (3685*t^4)/576 - (27227*t^3)/720 + (39227*t^2)/288 - (39371*t)/144 + 3157891/13440$

[8,9) aralığındaki çözüm;  $-t^9/362880 + t^8/4480 - t^7/126 + (701*t^6)/4320 - (1511*t^5)/720 + (51193*t^4)/2880 - (212753*t^3)/2160 + (1156699*t^2)/3360 - (1158379*t)/1680 + 43896157/72576$

[9,10) aralığındaki çözüm;  $t^{10}/3628800 - t^9/36288 + (11*t^8)/8960 - (323*t^7)/10080 + (1873*t^6)/3456 - (89269*t^5)/14400 + (279533*t^4)/5760 - (7761511*t^3)/30240 + (23602499*t^2)/26880 - (23615939*t)/13440 + 5681592251/3628800$

### EK A.2. (Bölüm 4.1 Sayısal Örnek 1'in MATLAB Kodları ve Çıktıları)

Bölüm 4.1 Sayısal Örnek 1'in PIA(1,1) algoritması ile çözümünün MATLAB

Kodu

Editor

```
%4.1. Sayısal Örnek 1
n=2; % İterasyon Sayısı
syms t y(t)

f(t)=1-(1/2)*t+(1/8)*t^2-(1/48)*t^3; % Başlangıç
Fonksiyonu

% Algoritma %
for i=1:n
    Dy=diff(y);
    D2y=diff(y,2);
    D3y=diff(y,3);
    uc(t)=dsolve(diff(y,4)+diff(y,3)==(-
1/(16*(exp(1/2))))*f(t-1)-diff(f,4)-diff(f,3), y(0)==0,
Dy(0)==0, D2y(0)==0, D3y(0)==0); % Düzeltme terimi
    f(t)=f(t)+uc(t); %İterasyon Çözümleri
    f(t)
end

ye(t)=exp(-t/2); % Tam Çözüm
t=linspace(0,1); % Problemin tanım aralığı
figure
plot(t,abs(f(t)-ye(t))) % Mutlak Hata Fonksiyonun
Grafiği
```

Command Window

```
>> SayisalOrnek411
```



1. iterasyon çözümü

$$\begin{aligned} & (22231223145771421 \cdot \exp(-t)) / 1729382256910270464 - \\ & (842459905309363811 \cdot t) / 1729382256910270464 + \\ & (410114341081796195 \cdot t^2) / 3458764513820540928 - \\ & (193941558968012387 \cdot t^3) / 10376293541461622784 + \\ & (19120998771494179 \cdot t^4) / 9223372036854775808 - \\ & (2731571253070597 \cdot t^5) / 17293822569102704640 + \\ & (2731571253070597 \cdot t^6) / 415051741658464911360 + \\ & 1707151033764499043 / 1729382256910270464 \end{aligned}$$

2. iterasyon çözümü

$$\begin{aligned} & (22231223145771421 \cdot \exp(-t)) / 1729382256910270464 - \\ & (842459905309363811 \cdot t) / 1729382256910270464 + \exp(- \\ & t) \cdot ((60726170065586899772982538008337 \cdot t \cdot \exp(1)) / 1246151 \\ & 24604835863084731911901282304 - \\ & \exp(t) \cdot ((1065925930085953068643009419487 \cdot t^9) / 215334935 \\ & 3171563714104167437654158213120 - \\ & (1065925930085953068643009419487 \cdot t^8) / 39876839873547476 \\ & 187114211808410337280 + \\ & (7461481510601671480501065936409 \cdot t^7) / 99692099683868690 \\ & 46778552952102584320 - \\ & (173220865626295479939406280392243 \cdot t^6) / 224307224288704 \\ & 55355251744142230814720 + \\ & (626493426112011553711659199258319 \cdot t^5) / 132922799578491 \\ & 58729038070602803445760 - \\ & (11921594698715408025861078198550567 \cdot t^4) / 5981525981032 \\ & 1214280671317712615505920 + \\ & (4447137135675577031624769853152091 \cdot t^3) / 74769074762901 \\ & 51785083914714076938240 + \\ & ((60726170065586899772982538008337 \cdot \exp(1)) / 249230249209 \\ & 671726169463823802564608 - \\ & 4447137135675577031624769853152091 / 24923024920967172616 \\ & 94638238025646080) \cdot t^2 + \\ & (4447137135675577031624769853152091 / 1246151246048358630 \\ & 847319119012823040 - \\ & (60726170065586899772982538008337 \cdot \exp(1)) / 6230756230241 \\ & 7931542365955950641152) \cdot t + \\ & (60726170065586899772982538008337 \cdot \exp(1)) / 4153837486827 \\ & 8621028243970633760768 - \\ & 4447137135675577031624769853152091 / 12461512460483586308 \\ & 47319119012823040) + \exp(- \\ & t) \cdot ((60726170065586899772982538008337 \cdot \exp(1)) / 415383748 \\ & 68278621028243970633760768 - \\ & 4447137135675577031624769853152091 / 12461512460483586308 \\ & 47319119012823040) + \\ & (410114341081796195 \cdot t^2) / 3458764513820540928 - \\ & (193941558968012387 \cdot t^3) / 10376293541461622784 + \\ & (19120998771494179 \cdot t^4) / 9223372036854775808 - \\ & (2731571253070597 \cdot t^5) / 17293822569102704640 + \end{aligned}$$

$$(2731571253070597*t^6)/415051741658464911360 + 1707151033764499043/1729382256910270464$$

**EK A.3.** (Bölüm 4.2 Sayısal Örnek 1'in MATLAB Kodları ve Çıktıları)  
Bölüm 4.2 Sayısal Örnek 1'in PIA(1,1) algoritması ile çözümünün MATLAB Kodu

Editor

```
% 4.2 Sayısal Örnek 1
iterasyonSayisi=4;
syms t y(t)
f(t)=1+0*t; % Başlangıç Fonksiyonu
ye(t)=exp(-t); % Tam Çözüm
for i=1:iterasyonSayisi
    Dy=diff(y);
    uc(t)=dsolve(diff(y,1)+y==(1/10)*f(t/5)-diff(f,1)-
f(t)-(1/10)*exp(-(1/5)*t), y(0)==0); % Düzeltme terimi
    f(t)=f(t)+uc(t); %İterasyon Çözümleri
end
f(t)
for i=1:6
vpa(abs(f(2^(-i))-ye(2^(-i))))
end
```

Command Window

```
>> SayisalOrnek421
ans =
(74280959999*exp(-t))/74280960000 - exp(-
t)*((5*exp((24*t)/25))/384 - (41*exp((4*t)/5))/320 +
(9*exp(t))/100) - exp(-t)*((241*exp((24*t)/25))/737280
- (1241*exp((4*t)/5))/95232000 -
(1025*exp((124*t)/125))/761856 +
(15625*exp((624*t)/625))/118849536 + (9*exp(t))/10000)
- exp(-t)*(exp((4*t)/5)/8 + (9*exp(t))/10) - exp(-
t)*((241*exp((4*t)/5))/76800 - (41*exp((24*t)/25))/3072
+ (125*exp((124*t)/125))/95232 + (9*exp(t))/1000) + 1

ans =
0.000000000000000024084051461292704192294061073729

ans =
0.0000000000000000079150904545856673204124037705924

ans =
0.0000000000000000025376129286392516904219161778595

ans =
0.00000000000000000080330683721215782558270019744134

ans =
0.0000000000000000002526656625519315187964549923279
```



```

Command Window
>> SayisalOrnek431
t^3/12 - sin(t)/2 - t/2 + 1

t^3/24 - (3*sin(t))/4 - t/4 - t^5/480 + 1

t^3/48 - (7*sin(t))/8 - t/8 - t^5/960 + t^7/40320 + 1

t^3/96 - (15*sin(t))/16 - t/16 - t^5/1920 + t^7/80640 -
t^9/5806080 + 1

t^3/192 - (31*sin(t))/32 - t/32 - t^5/3840 + t^7/161280
- t^9/11612160 + t^11/1277337600 + 1

3.2686

```

**EK B. (Bazı Problemlerin Çözüm Fonksiyonlarının MATLAB Çıktıları)**

**EK B.1. (Gecikmeli Mathieu Denklemine  $u_2^{D_2}(t)$  Fonksiyonunun MATLAB Çıktısı)**

(5.5) gecikmeli Mathieu denklemin için PIA(1,1) algoritması ile elde edilen  $D_2$  bölgesindeki ikinci iterasyon çözümünün  $u_2^{D_2}(t)$ 'nin MATLAB çıktısı

```

u_2^(D_2) (t) =(1634097482518347941811470984269699*cos(3*2^(1/2)*t -
6*pi*2^(1/2)))/60584802616265013068371725385728000 -
(246060211085503820023393650581*t)/35561808332535405004917181933158400
+ (1055699219448663738073327*cos(2*t))/18808152683601669151260672000 -
cos(3*t)/239904000 -
(6643625699308603140631628565687*sin(2*t))/1851436646312624523068500784
395059200 - (2621999565540885850935319718959*cos(t - 6*pi*2^(1/2) +
3*2^(1/2)*t))/13463289470281114015193716752384000 -
(2645529805955040244828521491503*cos(t + 6*pi*2^(1/2) -
3*2^(1/2)*t))/13463289470281114015193716752384000 -
(24359960897464878182315971407519*sin(t - 6*pi*2^(1/2) +
3*2^(1/2)*t))/383241988012234230722634273154662400 -
(24359960897464878182315971407519*sin(t + 6*pi*2^(1/2) -
3*2^(1/2)*t))/383241988012234230722634273154662400 -
(1849286093389382520601*cos(t))/18808152683601669151260672000 -
(768597613174351901*cos(2*t - 6*pi*2^(1/2) +
3*2^(1/2)*t))/7381197690226964692992000 + (973959996963842077*cos(2*t +
6*pi*2^(1/2) - 3*2^(1/2)*t))/7381197690226964692992000 -
(2214541899769534380210542855229*t*cos(2*t))/24893265832774783503442027
3532108800 + (19930877097925809421894885697061*pi*cos(t - 6*pi*2^(1/2) +
3*2^(1/2)*t))/315611048951251719418639989656780800 +
(19930877097925809421894885697061*pi*cos(t + 6*pi*2^(1/2) -
3*2^(1/2)*t))/315611048951251719418639989656780800 +
(891074107423983069716862859621*2^(1/2)*cos(t - 6*pi*2^(1/2) +
3*2^(1/2)*t))/747960526126728556399650930688000 -
(895023238682302688272365255013*2^(1/2)*cos(t + 6*pi*2^(1/2) -
3*2^(1/2)*t))/747960526126728556399650930688000 - (33277*pi*sin(t -

```

$$\begin{aligned}
& 6\pi^{2^{1/2}} + 3^{2^{1/2}}t)/4799032000 - (33277\pi\sin(t + 6\pi^{2^{1/2}} - \\
& 3^{2^{1/2}}t))/4799032000 + (8119986965821626060771990469173^{2^{1/2}}\sin(t - \\
& 6\pi^{2^{1/2}} + 3^{2^{1/2}}t))/1532967952048936922890537092618649600 - \\
& (8119986965821626060771990469173^{2^{1/2}}\sin(t + 6\pi^{2^{1/2}} - \\
& 3^{2^{1/2}}t))/1532967952048936922890537092618649600 + (33277t\sin(t - \\
& 6\pi^{2^{1/2}} + 3^{2^{1/2}}t))/9598064000 + (33277t\sin(t + 6\pi^{2^{1/2}} - \\
& 3^{2^{1/2}}t))/9598064000 + (20772908464171673^{2^{1/2}}\cos(2t - 6\pi^{2^{1/2}} + \\
& 3^{2^{1/2}}t))/820133076691884965888000 + \\
& (26323243161184921^{2^{1/2}}\cos(2t + 6\pi^{2^{1/2}} - \\
& 3^{2^{1/2}}t))/820133076691884965888000 - \\
& (1112930334739733777965809481577863\pi\cos(3^{2^{1/2}}t - \\
& 6\pi^{2^{1/2}}))/129957490744633060937087054564556800 + \\
& (152049925522463763729^{2^{1/2}}\cos(3^{2^{1/2}}t - \\
& 6\pi^{2^{1/2}}))/29114724222561916289024000 - \\
& (7849358604226099\pi\sin(3^{2^{1/2}}t - \\
& 6\pi^{2^{1/2}}))/9045585404689907712000 - \\
& (7546912734203487663937506656969851^{2^{1/2}}\sin(3^{2^{1/2}}t - \\
& 6\pi^{2^{1/2}}))/10916429222549177118715312583422771200 + \\
& (7849358604226099t\sin(3^{2^{1/2}}t - \\
& 6\pi^{2^{1/2}}))/18091170809379815424000 - (42532423^{2^{1/2}}\pi\sin(3^{2^{1/2}}t \\
& - 6\pi^{2^{1/2}}))/518295456000 + (42532423^{2^{1/2}}t\sin(3^{2^{1/2}}t - \\
& 6\pi^{2^{1/2}}))/1036590912000 - \\
& (59792631293777428265684657091183^{2^{1/2}}\pi\cos(t - 6\pi^{2^{1/2}} + \\
& 3^{2^{1/2}}t))/157805524475625859709319994828390400 + \\
& (59792631293777428265684657091183^{2^{1/2}}\pi\cos(t + 6\pi^{2^{1/2}} - \\
& 3^{2^{1/2}}t))/157805524475625859709319994828390400 + \\
& (33277^{2^{1/2}}\pi\sin(t - 6\pi^{2^{1/2}} + 3^{2^{1/2}}t))/57588384000 - \\
& (33277^{2^{1/2}}\pi\sin(t + 6\pi^{2^{1/2}} - 3^{2^{1/2}}t))/57588384000 - \\
& (33277^{2^{1/2}}t\sin(t - 6\pi^{2^{1/2}} + 3^{2^{1/2}}t))/115176768000 + \\
& (33277^{2^{1/2}}t\sin(t + 6\pi^{2^{1/2}} - 3^{2^{1/2}}t))/115176768000 + \\
& 537022253214983943261851/12090955296601073025810432000
\end{aligned}$$

**EK B.2.** (Gecikmeli Sönümlü Mathieu Denkleminin  $u_2^{D_j}(t)$  Fonksiyonunun MATLAB Çıktısı)

Gecikmeli sönümlü Mathieu denkleminin  $\kappa = 0.2$ ,  $\delta = 3.0$ ,  $\varepsilon = 1$ ,  $d = -1$ , ve  $\tau = 2\pi$  için ikinci iterasyon PIA çözümünün MATLAB çıktıları

$$\begin{aligned}
u_2^{(D_1)}(t) = & (3\sin(2t))/46864 + \cos(t)/6060 + \sin(t)/60600 + (443773\exp(- \\
& t/10)\cos((299^{1/2}t)/10))/300954750 - (13\cos(t)^2)/29290 - (277\exp(- \\
& t/10)\cos((299^{1/2}t)/10)\cos(t))/1660440 + (44\exp(- \\
& t/10)\cos((299^{1/2}t)/10)\sin(t))/1037775 - (420833^{299^{1/2}}\exp(- \\
& t/10)\sin((299^{1/2}t)/10))/359941881000 - (44^{299^{1/2}}\exp(- \\
& t/10)\sin((299^{1/2}t)/10)\cos(t))/62058945 - (277^{299^{1/2}}\exp(- \\
& t/10)\sin((299^{1/2}t)/10)\sin(t))/8302200 - 127/4393500
\end{aligned}$$

$$\begin{aligned}
u_2^{(D_2)}(t) = & (\exp(\pi/5) - \\
& t/10)^{(268934376584768635696982710933547900036737572268607072255293846 \\
& 5\cos(t - (\pi^{299^{1/2}})/5 + (299^{1/2}t)/10) - \\
& 9257627971025065492305583034069994845921837056520941952040960\sin(299
\end{aligned}$$

$^{(1/2)}*(\pi/5 - t/10)) -$   
3554490917298765761102614914024155077702259418533789636666709970\*exp  
 $(t/10 - \pi/5) -$   
59938257239362174676933033760483174909972550563655020364892494646\*cos  
 $(299^{(1/2)}*(\pi/5 - t/10)) +$   
3154387874506730000055786409724432924953585187628831152379805665\*cos(  
 $t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) +$   
320738904717629207084586770795941505605164581168495716369195545\*sin(  
 $t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) +$   
311377058415354567786027632370239139714716338527591659153154585\*sin(  
 $t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) +$   
34653400489317376797946562743337791816372415272969248650035200\*cos(2\*  
 $t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) -$   
62846112640532806522095936092451337057668088937911527093043200\*cos(2\*  
 $t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) +$   
2546672848059645102343712751798349466370776094354933861580800\*sin(2\*t  
 $- (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) +$   
4618550745606659058595333760034118349467546743233284197580800\*sin(2\*t  
 $+ (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) +$   
576539334824958964612439087262310073679446938029976080423608100\*t\*exp  
 $(t/10 - \pi/5) -$   
945391071140405216390543582599549880861229405919032883793684490\*pi\*co  
 $s(t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) -$   
960976218463291214043287851924730873713876162895413571913574410\*pi\*co  
 $s(t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) -$   
607438561688098926629709253197625219318505502320784174493889325\*299^  
 $(1/2)*\cos(t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) +$   
655971960811678001422973417531411335706385652414286920922592045\*299^  
 $(1/2)*\cos(t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) +$   
822126817710261942726713680638872479224206373765806078609122110\*pi\*si  
 $n(t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) +$   
822126817710261942726713680638872479224206373765806078609122110\*pi\*si  
 $n(t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) -$   
4074885177332168045789729515534023078840256551444424897945335\*299^(1/  
 $2)*\sin(t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) +$   
3500409456475438652811454693928689439315505176146152365055735\*299^(1/  
 $2)*\sin(t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) -$   
3896286830721499413186067331295248213161689244095172029972480\*t\*cos(  
 $t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) +$   
3896286830721499413186067331295248213161689244095172029972480\*t\*cos(  
 $t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) -$   
284182672030514089735144880604409606608925630812802463941263360\*t\*sin  
 $(t - (\pi*299^{(1/2)})/5 + (299^{(1/2)}*t)/10) -$   
284182672030514089735144880604409606608925630812802463941263360\*t\*sin  
 $(t + (\pi*299^{(1/2)})/5 - (299^{(1/2)}*t)/10) -$   
325773400425918747076470215365321678582335830961699537215815680\*exp(  
 $t/10 - \pi/5)*\cos(t) -$   
206535454711343634122159480335895899380163190689459226856652800\*exp(  
 $t/10 - \pi/5)*\sin(t) -$   
1489401167162637971258448255444317699844086616316729884672000\*299^(1/

$2) \cdot \cos(2 \cdot t - (\pi \cdot 299^{1/2})/5 + (299^{1/2} \cdot t)/10) -$   
2701122319793673632754839660134011621389745942890180247552000 $\cdot 299^{1/2}$   
 $\cdot \cos(2 \cdot t + (\pi \cdot 299^{1/2})/5 - (299^{1/2} \cdot t)/10) -$   
109455853068466121877236937756376051563213872250212057088000 $\cdot 299^{1/2}$   
 $\cdot \sin(2 \cdot t - (\pi \cdot 299^{1/2})/5 + (299^{1/2} \cdot t)/10) +$   
198505046372779042633037267623242909002903155153293017088000 $\cdot 299^{1/2}$   
 $\cdot \sin(2 \cdot t + (\pi \cdot 299^{1/2})/5 - (299^{1/2} \cdot t)/10) +$   
17813643645737685582300803848885924490503011423542635213199485044 $\cdot \pi$   
 $\cdot \cos(299^{1/2} \cdot (\pi/5 - t/10)) -$   
44342875636622763189250876418207787066646317534295836296478720 $\cdot 299^{1/2}$   
 $\cdot \cos(299^{1/2} \cdot (\pi/5 - t/10)) -$   
222377113041148269119638478720023342646808714475246697683353600 $\cdot \pi \cdot \sin$   
 $\sin(299^{1/2} \cdot (\pi/5 - t/10)) -$   
7581533286845346680798199952907063034492777949943356896743966 $\cdot 299^{1/2}$   
 $\cdot \sin(299^{1/2} \cdot (\pi/5 - t/10)) +$   
8171229198507705031672608916072556420354305127106334634803200 $\cdot t \cdot \cos(2$   
 $99^{1/2} \cdot (\pi/5 - t/10)) +$   
12540413280214150392355081890222216825021271172403479007639745350 $\cdot \cos$   
 $\cos(2 \cdot t) \cdot \exp(t/10 - \pi/5) -$   
54397528263204880393483257412392613645251679067530447552512000 $\cdot \cos(3 \cdot$   
 $t) \cdot \exp(t/10 - \pi/5) +$   
111188556520574134559819239360011671323404357237623348841676800 $\cdot t \cdot \sin$   
 $(299^{1/2} \cdot (\pi/5 - t/10)) +$   
977527688593073370720605246913045796163929096929928135963223250 $\cdot \sin(2$   
 $\cdot t) \cdot \exp(t/10 - \pi/5) +$   
51807169774480838469984054678469155852620646730981378621440000 $\cdot \sin(3 \cdot$   
 $t) \cdot \exp(t/10 - \pi/5) +$   
2777385799344732362530145835547078268063694735480638442307584 $\cdot 299^{1/2}$   
 $\cdot \pi \cdot \cos(299^{1/2} \cdot (\pi/5 - t/10)) -$   
194340320245262966305168681764989272511538428015132746794856044 $\cdot 299^{1/2}$   
 $\cdot \pi \cdot \sin(299^{1/2} \cdot (\pi/5 - t/10)) -$   
1388692899672366181265072917773539134031847367740319221153792 $\cdot 299^{1/2}$   
 $\cdot t \cdot \cos(299^{1/2} \cdot (\pi/5 - t/10)) +$   
104706273781917708916601115408914578410022207160786715037138944 $\cdot 299^{1/2}$   
 $\cdot t \cdot \sin(299^{1/2} \cdot (\pi/5 - t/10)) -$   
1550692004011958594474836165740006405068857281597866699070394200 $\cdot t \cdot \cos$   
 $\cos(2 \cdot t) \cdot \exp(t/10 - \pi/5) +$   
447315001157295748406202740117309539923708831230153855501075250 $\cdot t \cdot \sin$   
 $(2 \cdot t) \cdot \exp(t/10 - \pi/5) +$   
190506418364024442394112706417936595584160667609066880619890690 $\cdot 299^{1/2}$   
 $\cdot \pi \cdot \cos(t - (\pi \cdot 299^{1/2})/5 + (299^{1/2} \cdot t)/10) -$   
190767039556714843692653580486919555330860446153822410521561090 $\cdot 299^{1/2}$   
 $\cdot \pi \cdot \cos(t + (\pi \cdot 299^{1/2})/5 - (299^{1/2} \cdot t)/10) -$   
13747940095489330145931666900315593298063651735214148471724450 $\cdot 299^{1/2}$   
 $\cdot \pi \cdot \sin(t - (\pi \cdot 299^{1/2})/5 + (299^{1/2} \cdot t)/10) +$   
13747940095489330145931666900315593298063651735214148471724450 $\cdot 299^{1/2}$   
 $\cdot \pi \cdot \sin(t + (\pi \cdot 299^{1/2})/5 - (299^{1/2} \cdot t)/10) +$   
65155298172600324635218517245739936674944636188882475417600 $\cdot 299^{1/2}$   
 $\cdot t \cdot \cos(t - (\pi \cdot 299^{1/2})/5 + (299^{1/2} \cdot t)/10) +$   
65155298172600324635218517245739936674944636188882475417600 $\cdot 299^{1/2}$

$t \cos(t + (\pi \cdot 299^{1/2})/5 - (299^{1/2} \cdot t)/10) +$   
 $4752218595827994811624496331177418170717819913257566286643200 \cdot 299^{1/2}$   
 $t \sin(t - (\pi \cdot 299^{1/2})/5 + (299^{1/2} \cdot t)/10) -$   
 $4752218595827994811624496331177418170717819913257566286643200 \cdot 299^{1/2}$   
 $t \sin(t + (\pi \cdot 299^{1/2})/5 -$   
 $(299^{1/2} \cdot t)/10)) / 4224530695501609824459758964906094624586352339817237$   
 $710224385638400$





## ÖZGEÇMİŞ

Adı Soyadı : Muhammet Mustafa BAHŞI

Doğum Yeri ve Yılı : İzmir, 1988

Medeni Hali : Evli

Yabancı Dili : İngilizce

E-posta : mustafa.bahsi@cbu.edu.tr

### Eğitim Durumu

Lise : Buca Hoca Ahmet Yesevi Lisesi, 2005

Lisans : Çanakkale Onsekiz Mart Üniversitesi, Matematik Bölümü, 2009

Yüksek Lisans : Çanakkale Onsekiz Mart Üniversitesi, Matematik Bölümü, 2011

### Mesleki Deneyim

Çanakkale Uğur Dershane, Matematik Öğretmeni, 2009-2011

Hüseyin Avni Karabacak Lisesi, Matematik Öğretmeni, 2011-2011

Manisa Celal Bayar Üniversitesi, Kırkağaç Meslek Yüksekokulu, Öğretim Görevlisi  
2011-...

### Yayınları

Bahşı, M. M., Çevik, M., Sezer, M. Jacobi polynomial solutions of Volterra integro differential equations with weakly singular kernel. *New Trends in Mathematical Sciences*. 2018, 6(3), 24-38.

Bahşı, M. M., Kurt Bahşı, A., Çevik, M., Sezer, M. Improved Jacobi matrix method for the numerical solution of Fredholm integro-differential-difference equations. *Mathematical Sciences*. 2016, 3, 83-93.

Bahşı, M. M., Çevik, M., Sezer, M. Orthoexponential polynomial solutions of delay pantograph differential equations with residual error estimation. *Applied Mathematics and Computation*. 2015, 271, 11-21.

Bahşı, M. M., Çevik, M. Numerical Solution of Pantograph-Type Delay Differential Equations Using Perturbation-Iteration Algorithms. *Journal of Applied Mathematics*. 2015, DOI: 10.1155/2015/139821.

Çevik, M., Bahşı, M. M., Sezer, M. Solution of the delayed single degree of freedom system equation by exponential matrix method. Applied Mathematics and Computation. 2014, 242, 444-453.

Bahşı, M. M., Çevik, M. Taylor Matrix Solution of the Mathematical Model of the RLC Circuits. Mathematical and Computational Applications, 2013, 18(3), 467-475.

Bahşı, M. M., İstatistik, Ed: Editörler: Aysen Şimşek Kandemir, Tahir Benli, Lisans Yayıncılık, 2016, Ankara, Türkiye, 2016, 16 (4. Bölüm).

Bahşı, M. M., Çevik, M. A New Perturbation-Iteration Method for Solving Delay Differential Equations. 2. International Students Science Conference. 4-5 Mayıs, 2018, İzmir, Türkiye. (Bildiri Özetleri Kitabı 123s)

Kurt Bahşı, A., Sezer, M., Bahşı, M. M. Fibonacci Sıralama Yöntemi ile Bagley-Torvik Denklemine Sayısal Çözümü. XX. Ulusal Mekanik Kongresi. 5-9 Eylül 2017, Bursa, Türkiye. (Bildiri Kitabı 694-700).

Bahşı, M. M., Çevik, M. Troesch Probleminin Perturbasyon İterasyon Yöntemi ile Analizi. XX. Ulusal Mekanik Kongresi. 5-9 Eylül, 2017, Bursa, Türkiye. (Bildiri Kitabı 722-726).

Bahşı, M. M., Çevik, M. Jacobi Collocation Method for Solving Mathieu Equation. 1. International Students Science Conference. 5-6 Mayıs, 2017, İzmir, Türkiye. (Bildiri Özetleri Kitabı 73s)

M. M. Bahşı, M. Çevik, Kurt Bahşı, A. Jacobi Collocation Method for Numerical Solutions of the Fractional Convection-Diffusion Equations with Variable Coefficients. International Conference on Advances in Applied and Computational Mechanics (ACM 2015), 5- 7 August 2015, İzmir.

Bahşı, M. M., Çevik, M. Taylor Matrix Solution of the Mathematical Model of the RLC Circuits. 4th International Conference on Mathematical and Computational Applications. (ICMCA 2013), 11- 13 Temmuz, 2013, Manisa.

Bahşı, M. M., Çevik, M. Eksenel Yüklemeye Mağruz Değişken Kesitli Euler-Bernoulli Kirişinin Üstel matris Yöntemi ile Enine Titreşim Analizi. XVIII. Ulusal Mekanik Kongresi. 26-30 Ağustos, 2013, Manisa. (Bildiriler Kitabı).