



**PARÇACIK SÜRÜ OPTİMİZASYON ALGORİTMASIYLA
LABVIEW ORTAMINDA SONLU İMPALS CEVAPLI
SAYISAL FİLTRE İÇİN GERÇEK ZAMANLI PENCERE
FONKSİYONU TASARIMI**

Fatmanur ŞERBET

**Yüksek Lisans Tezi
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Danışman: Dr. Öğr. Üyesi Turgay KAYA
AĞUSTOS-2018**

**T.C
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**PARÇACIK SÜRÜ OPTİMİZASYON ALGORİTMASIYLA
LABVIEW ORTAMINDA SONLU İMPALS CEVAPLI
SAYISAL FİLTRE İÇİN GERÇEK ZAMANLI PENCERE
FONKSİYONU TASARIMI**

YÜKSEK LİSANS TEZİ

Fatmanur ŞERBET

(161113111)

Tezin Enstitüye Verildiği Tarih: 17. 07. 2018

Tezin Savunulduğu Tarih: 01. 08. 2018

Tez Danışmanı : Dr. Öğr. Üyesi Turgay KAYA (F.Ü)
Diğer Jüri Üyeleri : Dr. Öğr. Üyesi Ayhan AKBAL (F.Ü)
Doç. Dr. Muhammed Fatih TALU (İ.Ü)

AĞUSTOS-2018

ÖNSÖZ

Bu çalışmamda bana her türlü konuda yardımcı olan danışmanım Sayın Dr. Öğr. Üyesi Turgay KAYA' ya ve Arş. Gör. Dr. Duygu KAYA' ya ve her konuda beni destekleyen aileme teşekkürlerimi sunarım.

Fatmanur ŞERBET

ELAĞİĞ – 2018



İÇİNDEKİLER

Sayfa No

ÖNSÖZ	I
İÇİNDEKİLER	II
ŞEKİLLER LİSTESİ	III
TABLolar LİSTESİ	IV
SİMGELER LİSTESİ	V
KISALTMALAR LİSTESİ.....	VI
ÖZET	VII
ABSTRACT.....	VIII
1. GİRİŞ	1
2. SAYISAL FİLTRELER.....	7
3. PENCERE FONKSİYONLARI	13
3.1. Yaygın Olarak Kullanılan Pencere Fonksiyonları	17
4. PROGRAMLAMA DİLLERİ VE LABVIEW	23
5. SEZGİSEL OPTİMİZASYON YÖNTEMLERİ	27
5.1. Parçacık Sürü Optimizasyonu.....	27
6. LABVIEW'DE PENCERE FONKSİYONU GERÇEKLEŞTİRİLMESİ.....	35
7. LABVIEW'DE PSO DESTEKLİ PENCERE FONKSİYONU GERÇEKLEŞTİRİLMESİ	39
8. SONUÇ VE GELECEK ÇALIŞMALAR.....	44
KAYNAKLAR	46
ÖZGEÇMİŞ	51

ŞEKİLLER LİSTESİ

Sayfa No

Şekil 2.1.	Sayısal filtre blok diyagramı	7
Şekil 2.2.	İdeal sayısal filtre frekans karakteristikleri; a) İdeal Alçak geçiren b) İdeal Yüksek geçiren c)İdeal Bant-geçiren d) İdeal Bant-durduran	8
Şekil 2.3.	FIR filtre blok diyagramı.....	11
Şekil 2.4.	IIR filtre blok diyagramı.....	11
Şekil 3.1.	Sonsuz uzunluktaki ideal alçak geçiren sayısal filtrenin impals cevabı.....	15
Şekil 3.2.	Sonlu uzunlukta ($N=23$) tasarlanan FIR filtrenin impals cevabı.....	15
Şekil 3.3.	Fourier Serisi Yöntemi ile tasarlanan FIR filtrelerin genlik cevapları	16
Şekil 3.4.	Pencere Fonksiyonu Spektral Gösterimi	17
Şekil 3.5.	FIR filtre tasarımında kullanılan pencere fonksiyonları genlik cevapları	18
Şekil 3.6.	$N=17$ ve $\alpha=0.2$ için Cosh Pencere Fonksiyonu Spektral Gösterimi	20
Şekil 3.7.	$N=45$ ve $\alpha=0.2$ için Cosh Pencere Fonksiyonu Spektral Gösterimi	20
Şekil 3.8.	$N=25$ ve $\alpha=0.1$ için Cosh Pencere Fonksiyonu Spektral Gösterimi	21
Şekil 3.9.	$N=25$ ve $\alpha=3.5$ için Cosh Pencere Fonksiyonu Spektral Gösterimi	21
Şekil 4.1.	LabVIEW Front Panel.....	25
Şekil 4.2.	LabVIEW Block Diagram	25
Şekil 4.3.	DAQ-6001 Kartı.....	26
Şekil 5.1.	Parçacık Sürü Optimizasyonu Akış Diyagramı.....	29
Şekil 5.2.	Pencere Fonksiyonları Spektral Gösterimi.....	31
Şekil 5.3.	Her İterasyonda İki Pencerenin Spektral Farkı	32
Şekil 5.4.	Ham EEG Sinyali	33
Şekil 5.5.	Filtrelenmiş EEG Sinyalleri	33
Şekil 6.1.	LabVIEW’de Gerçekleştirilen Tasarım.....	35
Şekil 6.2.	$N=51$ ve $\alpha=0.2$ için Cosh Pencere Fonksiyonu Spektrumu.....	36
Şekil 6.3.	$N=51$ ve $\alpha=0.2$ için Üstel Pencere Fonksiyonu Spektrumu	36
Şekil 6.4.	$N=21$ ve $\alpha=2.2$ için Cosh Pencere Fonksiyonu Spektrumu.....	37
Şekil 6.5.	$N=21$ ve $\alpha=2.2$ için Pencere Fonksiyonu Spektrumu.....	37
Şekil 7.1.	Sistemin Blok Diyagramı	39
Şekil 7.2.	LabVIEW’de Gerçekleştirilen Tasarım.....	40
Şekil 7.3.	LabVIEW’de Tasarlanan Pencere Fonksiyonunun Gerçek Zamanlı Analizi	40
Şekil 7.4.	$N=51$ ve $\alpha=0.2$ için Gerçek Zamanlı Cosh Pencere Fonksiyonu Spektral Gösterimi	41
Şekil 7.5.	$N=21$ ve $\alpha=2.2$ için Gerçek Zamanlı Cosh Pencere Fonksiyonu Spektral Gösterimi	42
Şekil 7.6.	LabVIEW’de $N=51$ ve $\alpha=0.2$ için Tasarlanmış Pencere Fonksiyonu ile Tasarlanmış FIR Sayısal Filtre Kullanılarak Filtrelenmiş EEG Sinyali	42
Şekil 7.7.	LabVIEW’de PSO Kullanılarak $N=51$ ve $\alpha=0.2$ için Tasarlanmış Pencere Fonksiyonu ile Tasarlanmış FIR Sayısal Filtre kullanılarak Filtrelenmiş EEG Sinyali	43

TABLÖLAR LİSTESİ

Sayfa No

Tablo 1.1. Literatür Taraması.....	3
Tablo 3.1. Değişken N değerleri için Cosh Penceresine ait Veriler.....	20
Tablo 3.2. Değişken α değerleri için Cosh Penceresine ait Veriler	21
Tablo 6.1. $N=51$ ve $\alpha=0.2$ için Cosh ve Üstel Pencereleere ait Veriler	36
Tablo 6.2. $N=21$ ve $\alpha=2.2$ için Cosh ve Üstel Pencereleere ait Veriler	38
Tablo 7.1. Cosh Pencere Fonksiyonunun $N=51$ ve $\alpha=0.2$ Değerlerindeki Verileri	41
Tablo 7.2. Cosh Pencere Fonksiyonunun $N=21$ ve $\alpha=2.2$ Değerlerindeki Verileri	42



SİMGELER LİSTESİ

dB	: Desibel
<i>N</i>	: Pencere uzunluğu
<i>α</i>	: Ayarlanabilir parametre
<i>t</i>	: Zaman
<i>f</i>	: Frekans
<i>ω</i>	: Analog frekans
<i>Ω</i>	: Sayısal frekans

KISALTMALAR LİSTESİ

PSO	: Parçacık Sürü Optimizasyonu
GA	: Genetic Algorithm (Genetik Algoritma)
CGWO	: Karmaşık Değerli Kodlama Gri Kurt Optimizasyonu
FGSA	: A Fuzzy Gravitational Search Algorithm
GUI	: Graphical User Interface
SEOA	: Structure Evolution Based Optimization Algorithm
FPGA	: Field-Programmable Gate-Array
ANN	: Artificial Neural Networks
DSTFT	: Discrete Short-Time Fourier Transform
MFIR	: Multiplicative Finite Impulse Response
FIR	: Finite Impulse Response (Sonlu İmpuls Cevabı)
IIR	: Infinite Impulse Response (Sonsuz İmpuls Cevabı)
LPF	: Lowpass Filter (Alçak Geçiren Filtre)
HPF	: Highpass Filter (Yüksek Geçiren Filtre)
BPF	: Bandpass Filter (Bant Geçiren Filtre)
BSF	: Bandstop Filter (Bant Durduran Filtre)

ÖZET

Fourier Serisi yöntemi kullanılarak Finite Impulse Response (FIR) sayısal filtre tasarımı sırasında, keskin kesim frekans bölgesinde oluşan ve istenmeyen Gibbs salınımlarını yok etmek için pencere fonksiyonları kullanılır. Pencere fonksiyonları tasarımı literatürde birçok yöntem kullanılarak gerçekleştirilmiştir. Bu tezde, karmaşıklığından uzak ve güçlü bir grafiksel programlama ortamı sağlayan LabVIEW ile iki farklı ayarlanabilir pencere fonksiyonu tasarımı sunulmaktadır.

Aynı zamanda, MATLAB’ da Parçacık Sürü Optimizasyonu (PSO) kullanılarak alternatif cosh pencere fonksiyonu tasarımı gerçekleştirilmiştir. LabVIEW’in paralel program sürdürebilme özelliğinden faydalanarak tasarımı Parçacık Sürü Optimizasyonu (PSO) kullanılarak gerçekleştirilen cosh pencere fonksiyonuna ait veriler LabVIEW ortamına aktarılmıştır ve tasarımları farklı yöntemlerle gerçekleştirilen pencere fonksiyonlarıyla FIR sayısal filtre tasarımı gerçekleştirilmiştir.

Tasarlanan FIR sayısal filtreler ile LabVIEW ortamına aktarılan EEG sinyali filtrelenerek sonuçlar DAQ (Data Acquisition) kart aracılığıyla gerçek zamanlı görüntülenmiştir. Tasarlanan pencere fonksiyonlarının farklı parametre değerlerindeki genlik spektrumları gözlemlenerek yorumlanmıştır.

Anahtar Kelimeler: LabVIEW, Pencere Fonksiyonları, FIR Filtre Tasarımı, Parçacık Sürü Optimizasyonu (PSO)

ABSTRACT

REAL-TIME WINDOW FUNCTION DESIGN FOR FINITE IMPULSE RESPONSE DIGITAL FILTER IN LABVIEW WITH PARTICLE SWARM OPTIMIZATION ALGORITHM

During Finite Impulse Response (FIR) digital filter design using the Fourier Series method, window functions are used to eliminate unwanted Gibbs oscillations in the cut-off frequency domain. The design of window functions has been performed using many methods in the literature. In this thesis, two different adjustable window function designs are presented with LabVIEW, which provides a powerful graphical programming environment that is remote from the complexity.

At the same time, an alternative cosh window function design was implemented in MATLAB using Particle Swarm Optimization (PSO). Utilizing LabVIEW's parallel programmability feature, the cosh window function design data, implemented using Particle Swarm Optimization (PSO), has been exported to the LabVIEW environment and FIR digital filter design has been realized with window functions realized by different methods of designs.

The designed FIR digital filters filtered the EEG signal whose it is transferred to the LabVIEW environment, and the results are displayed in real time by DAQ (Data Acquisition) card. The amplitude spectrum of the designed window functions at different parameter values were observed and interpreted.

Keywords: LabVIEW, Window Functions, Design of FIR Filter, Particle Swarm Optimization (PSO)

1. GİRİŞ

Geçmişten günümüze değişen hayat şartlarıyla birlikte ihtiyaçlar değişmeye ve teknoloji sürekli gelişmeye başlamıştır. Değişen ihtiyaçlar doğrultusunda hızlı ve kullanışlı teknolojiye olan ihtiyaç artmıştır. Bu doğrultuda zamanla analog sistemler yerlerini dijital sistemlere bırakmıştır. Dijital sistemlerin analog sistemlere göre harici faktörlerden daha az etkilenmesi, işlevlerinin kolaylıkla değiştirilebilir olması, kolayca uygulanabilir ve test edilebilir olması daha avantajlı olmasını sağlamaktadır. Buna paralel olarak, istenilen sonuçları daha doğru ve kolay elde edebilmek için dijital sistemlerin performansını artırmaya yönelik çalışmalar yapılmaya devam etmektedir.

Haberleşme sistemleri, biyomedikal sistemler, endüstriyel sistemler gibi içerisinde sinyal işleme yöntemlerine yer verilen sistemlerde sinyal süzme işlemi önemli bir role sahiptir. Sinyal işleme uygulamasının yer aldığı sistemlerde, işaretin işlendiği koşullardan ve çevre şartlarından kaynaklanan sebeplerden dolayı işaret çeşitli gürültülere maruz kalır. Sistemlerden istenilen çıkışı üretebilmek için kullanılan yazılımsal veya donanımsal yapılara filtre denilmektedir. Filtreler, girişlerine uygulanan sinyallerin istenilen değerlerini ileten, istenilmeyen değerlerini bastıran yapılardır. Genel olarak filtreler, doğru bileşenleri süzmek, gürültüyü azaltmak, rezonansı önlemek veya rezonans oluşturmak, işaret biçimlendirmek, işaret zayıflatmak, güç faktörünü düzeltmek gibi işlemler için kullanılmaktadır [1].

Filtreler yapım elemanlarına göre aktif ve pasif filtreler olarak iki sınıfta incelenir. 19.ve 20. yüzyıllarında pasif elemanlar olan direnç, bobin ve kapasitörler kullanılarak ilk filtreler tasarlanmıştır. Tasarlanan filtreler, o zamanki telefon ağlarında kullanılmışlardır. Daha sonra aktif devre elemanı olan işlemsel yükselteçlerin geliştirilmesiyle aktif filtreler tasarlanmıştır. Aktif filtrelerin pasif filtrelerden farkı direnç ve bobin içermemeleri ve tüm devrenin VLSI (Very Large Scale Integration) teknolojisi ile üretilmesidir. Bu teknoloji sayesinde analog sinyallerden belirli zaman aralıklarında örnekler alınıp işlenilmiştir ve böylelikle filtreleme işlemi, sayısal işaret işleme uygulama alanına girmiştir [2,3].

Herhangi bir sayısal sistemde, giriş değerleriyle bağlantılı olarak elde edilmek istenen çıkış değerini sağlayan yapılara sayısal filtre denir. Sayısal filtreler, analog sinyallerin bir dizi işlem ile sayısallaştırılması sonucu elde edilen sinyaller ile istenilen filtreleme işlemini gerçekleştirir. Sayısal filtrelerden beklenen, kararlı olması ve işaretin distorsiyona uğratılmadan filtrelenmesidir.

Literatürde filtre tasarımını konu alan ve filtre tasarımına yenilikler katan birçok çalışma bulunmaktadır. Serbet ve Kaya'nın çalışmalarında Parçacık Sürü Optimizasyonu (PSO) yönteminin kullanılarak; Dhaliwal ve Dhillon'un çalışmalarında karşıt yapay arı kolonisi algoritması kullanarak; Luo ve çalışma arkadaşlarının çalışmalarında Karmaşık Değerli Kodlama Gri Kurt Optimizasyonu (Complex-Valued Encoding Gray Wolf Optimization, CGWO) kullanarak; Chen ve diğerleri, Yapısal Evrim Tabanlı Optimizasyon Algoritması (Structure Evolution Based Optimization Algorithm, SEOA) kullanarak; Vasundhara ve çalışma arkadaşlarının çalışmalarında, Hibrid Rastgele Parçacık Sürü Optimizasyonu'nu kullanarak, filtre tasarımı, sezgisel optimizasyon yöntemleri ile gerçekleştirilmiştir [4,7,8,11,14]. Bazı çalışmalarda, filtre tasarımı ve pencere fonksiyon tasarımı, akıllı hesaplama yöntemleri kullanılarak gerçekleştirilmiştir. Pelusi ve çalışma arkadaşlarının çalışmalarında IIR filtre tasarımı, A Fuzzy Gravitational Search Algorithm (FGSA) kullanılarak tasarlanmıştır; Kaya ve İnce'nin çalışmalarında, pencere parametreleri Genetik Algoritma kullanılarak hesaplanmıştır; Varshney ve Tiwari'nin çalışmalarında, doğrusal faz filtre bankaları Genetik Arama Algoritması (Genetic Search Algorithm, GSA) kullanılarak tasarlandı; Kaya ve İnce eserlerinde, yeni bir pencere fonksiyonunun tasarımı ve ayarlanabilir pencere parametrelerinin hesaplanması Genetik Algoritma (GA) kullanılarak gerçekleştirmiştir [9,13,22,27,28]. Dağıtılmış Aritmetik Yapı ve Değiştirilmiş Dağıtılmış Aritmetik Yapı kullanılarak çeşitli filtre tasarım çalışmaları yapılmıştır [15-18].

Filtre tasarımı farklı platformlarda gerçekleştirilebilir. Yuce ve Tan'ın çalışmaları ve Sharma ve Kaur'un çalışmalarında IIR filtresi LabVIEW kullanılarak tasarlandı; Zhang ve ekip arkadaşlarının çalışmalarında IIR filtresi, Grafik Kullanıcı Arayüzü (Graphical User Interface, GUI) kullanılarak tasarlanmıştır [5,6,10]. FIR filtresi, Vandenbussche ve ekip arkadaşlarının çalışmasında ve Dwivedi ve diğerlerinin çalışmasında FPGA kullanılarak tasarlandı; Varshney ve Tiwari'nin çalışmalarında, FIR

filtresi tasarımı, ATMEGA32 mikrodenetleyici kullanılarak gerçekleştirilmiştir [19-21]. Karmaker ve ekip arkadaşlarının çalışmalarında, FIR filtresini tasarlamak için yeni bir ayarlanabilir pencere fonksiyonu tasarlandı; Avcı ve Nacaroğlu'nun eserlerinde yeni bir pencere işlevi tasarlanmıştır [12,24-26]. He ve Liu'nun çalışmalarında, spektrum kaçağını azaltmak için pencere fonksiyon seçimi, Ayrık Kısa-Zaman Fourier Dönüşümü (DSTFT) algoritması kullanılarak araştırılmıştır [23]. Söz konusu çalışmalar, içeriğe, yöneme ve amacına bağlı olarak Tablo 1.1 'de listelenmiştir.

Tablo 1.1. Literatür Taraması

	İçerik	Metot / Platform	Amaç / Yenilik
[4]	IIR filtre tasarımı	PSO	Alternatif filtre tasarımı
[5,6]	IIR filtre tasarımı	LabVIEW	Tasarım ve analiz
[7]	IIR filtre tasarımı	Karşıt Yapay Arı Kolonisi Algoritması	IIR filtre tasarımı ve optimizasyonu
[8]	IIR filtre tasarımı	Karmaşık Değerli Kodlama Gri Kurt Optimizasyonu (CGWO)	Önerilen yeni yöntem ile IIR filtre tasarımı
[9]	Optimal 8. Derece IIR filtre tasarımı	A Fuzzy Gravitational Search Algorithm (FGSA)	IIR filtre tasarımında DE ve GSA' ya göre FGSA avantajları
[10]	IIR Filtre tasarımı ve Simülasyonu	Graphical User Interface (GUI)	GUI tabanlı filtre tasarımı
[11]	Alçak Geçiren IIR Sayısal Filtre Tasarımı	Yapısal Evrim Tabanlı Optimizasyon Algoritması(SEOA)	Dijital filtreleri optimize etmenin ve otomatik tasarlanmanın yolu
[12]	FIR filtre tasarımı	Pencere Fonksiyonu	Yeni ayarlanabilir pencere fonksiyonu
[13]	Pencere Fonksiyonu Kullanarak FIR filtre tasarımı	Genetik Algoritma	GA ile parametre hesabı
[14]	FIR filtre tasarımı	Diferansiyel Gelişim ile Hibrid Rastgele Parçacık Sürü Optimizasyonu	Hibrid algoritma ile FIR filtre tasarımı
[15]	Adaptif Sayısal Filtre Uygulaması	Dağıtılmış Aritmetik Yapı	FPGA ile sayısal filtre tasarımı ve OrCAD Express ile simülasyonu
[16]	FIR Sayısal Filtre Gerçekleştirilmesi	Dağıtılmış Aritmetik Yapı	FPGA kullanarak standart ve dağıtılmış aritmetik yapının kıyası

Tablo 1. 1. Literatür Taraması (devam).

[17,18]	FIR Sayısal Filtre Gerçekleştirilmesi	Değiştirilmiş Dağıtılmış Aritmetik Yapı	Yeni Değiştirilmiş Dağıtılmış Aritmetik Yapı[17]. Temel ve Kaskat sayısal Filtrelerin tasarımı[18]
[19]	MFIR Filtre Uygulaması	Alan Programlanabilir Kapı Dizisi (FPGA)	MFIR filtrelerde yuvarlama gürültüsünün etkilerinin analizi
[20,21]	FIR filtre tasarımı	FPGA[20] Mikro denetleyici [21]	Farklı platformlarda filtre tasarımı
[22]	Doğrusal Fazlı Filtre Bankaları Tasarımı	Genetik Arama Algoritması (GSA)	GA dayalı yeni tasarım algoritması
[23]	Pencere Fonksiyonu Seçimi	DSTFT algoritması	Spektral sızıntıyı azaltmak için pencere fonksiyonu seçimi
[24]	Pencere Fonksiyonu tasarımı	Üstel fonksiyona dayanarak	Yeni pencere fonksiyonu ve Kaiser penceresi ile kıyaslaması
[25]	Yüksek Kaliteli Düşük Dereceli Yinelemesiz Sayısal Filtre Tasarımı	Pencere Fonksiyonu	Yeni pencere fonksiyonu kullanarak sayısal filtre tasarımı
[26]	Cosh Pencere Ailesi	Cosh pencere fonksiyonuna dayanarak	Yeni pencere ve FIR filtre tasarımında uygulaması
[27]	Kullanışlı Spektral Parametrelere Sahip Pencere Fonksiyonu Tasarımı	Yapay Sinir Ağları (YSA)	Yeni pencere fonksiyonu tasarımı
[28]	Ayarlanabilir Pencere Fonksiyonu	Genetik Algoritma (GA)	Ayarlanabilir pencere fonksiyonu parametreleri hesabı

Tekrarsız olarak tasarlanan filtrenin ideal genlik cevabından sınırlı sayıda eleman alınması durumunda keskin kesim frekansı bölgesinde Gibbs salınımları adı verilen istenmeyen salınımlar oluşur. İstenilmeyen bu salınımları yok etmek için pencere fonksiyonları kullanılır. Bu işlemi gerçekleştirmek için çeşitli pencere fonksiyonu tasarlanmıştır.

1899 yılında J.W.Gibbs tarafından yapılan çalışmayla Fourier Serisi ile tasarlanan filtre yaklaşımında serinin süresiz bölgelerde meydana gelen salınımlarla ilgili saptamalar yapıp matematiksel olarak ifade etmiştir [29]. Bu saptamaların

ardından L.Fejer, salınımları yok etmeye yönelik çalışmalar gerçekleştirmiştir [30]. Lanczos, Fejer'in çalışmalarını geliştirmeye yönelik çalışmıştır [31]. Adams ise yaptığı çalışmalarla uygun bir pencere fonksiyonu önermiştir [32]. Adams önerdiği pencere fonksiyonu ile en yüksek yan lob seviyesi ile toplam yan lob seviyesi arasında denge oluşturmaya çalışmıştır. Dolph ayarlanabilir pencereler ile yaptığı çalışmalarla minimum ana lob genişliği sağlamıştır [33]. Dolph, ayarlanabilir parametre özelliğine sahip pencerenin iki önemli parametresini kullanmıştır. Pek çok alanda tercih edilen iki parametrelili pencere, Kaiser tarafından geliştirilmiştir. Kaiser'in FIR filtre tasarımı üzerine yaptığı çalışmalar, daha önce yapılan çalışmalara göre çok daha başarılı sonuçlara ulaşılmıştır [34]. Saramaki geliştirdiği pencere fonksiyonu ile tasarladığı FIR filtre, durdurma bandı azalması bakımından Kaiser'in tasarladığı filtreden daha kullanışlı olmasına rağmen iki parametrelili pencereler ile yapılan çalışmalardan elde edilen pencere spektral cevapları Kaiser'in tasarladığı pencereden daha iyi olmadığı için fazla tercih edilmemiştir [35]. Deczky tarafından geliştirilen üç parametrelili pencere fonksiyonu (Ultraspherical pencere fonksiyonu) ile yanlob azalması fonksiyona eklenen parametre yardımıyla kontrol edilmektedir [36]. Bergen ve Antoniou, Ultraspherical pencere fonksiyonunun analizini gerçekleştirmişlerdir [37,38]. Bergen ve Antoniou, yaptıkları çalışmalarla Ultraspherical pencere fonksiyonun üstünlüklerini belirlemiş ve bu pencere fonksiyonunun FIR filtre dışında sayısal hüzmeleme ve görüntü işleme gibi diğer uygulama alanları için de kullanışlı olduğunu tespit etmişlerdir [39,40]. Avcı ve Nacaroğlu, Kaiser penceresi denkleminde faydalananarak yapmış oldukları çalışmada üstel pencere fonksiyonu kullanımı önermişlerdir [24]. Bu pencere fonksiyonu ile gerçekleştirilen FIR filtre tasarımıyla başarılı sonuçlar elde etmişlerdir [41].

Pencere parametrelerini hesaplamada akıllı hesaplama yöntemleri de tercih edilmektedir. Kaya ve İnce, pencere parametre değerlerini akıllı hesaplama yöntemlerinden biri olan Genetik Algoritma kullanarak elde etmişlerdir ve elde edilen sonuçlar Kaiser penceresi genlik spektrumu ile karşılaştırıldığında yanlob azalma oranı bakımından daha başarılı olduğu görülmüştür [28,42]. Mladenov ve Mastorakis ise çalışmalarında iki boyutlu filtre tasarımı için sinir ağı kullanımını gerçekleştirerek hesaplama hızını artırarak uygulamayı basitleştirmişlerdir ve ayrıca bu yöntem ile filtrenin daha kararlı çalıştığı gözlemlenmiştir [43].

Yapılan tez çalışması ile FIR sayısal filtre tasarım yöntemlerinden olan pencere fonksiyonu tasarımı gerçekleştirilmiştir ve çeşitli veriler sunulmuştur. Tez çalışması sekiz ana bölümden oluşmaktadır. Bu bölümlerden, Bölüm 2’ de Sayısal Filtreler hakkında genel bilgiler bahsedildikten sonra filtre tasarımını konu alan çalışmaları içeren bir literatür taramasına yer verilmiştir.

Bölüm 3, Pencere Fonksiyonları hakkında genel bilgiler içermektedir. Fourier Serisi kullanılarak gerçekleştirilen filtre tasarımı denklemlerle ifade edilmiştir ve pencere fonksiyonuna neden ihtiyaç duyulduğundan bahsedilmiştir. Bölüm içerisinde yaygın olarak kullanılan pencere fonksiyonlarına yer verilmiştir.

Bölüm 4’te programlama dillerinin tarihi gelişimden bahsedilmiş ve LabVIEW hakkında bilgilere yer verilmiştir.

Bölüm 5’te Cosh ve üstel pencere fonksiyonlarının LabVIEW’de tasarımı gerçekleştirilmiştir. Tasarıma ait program, pencere fonksiyonlarına ait genlik spektrumları sunulmuştur. Pencere fonksiyonlarının farklı parametre değerlerindeki genlik spektrumları incelenmiştir.

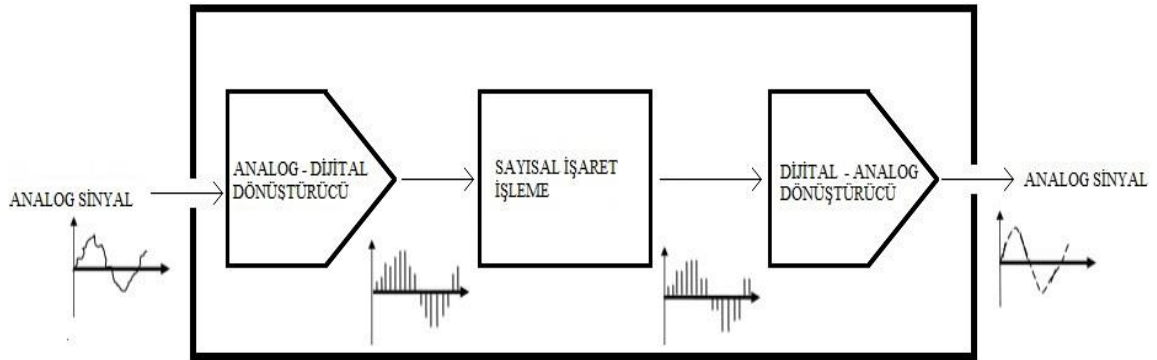
Bölüm 6’da Sezgisel Optimizasyon hakkında bilgilendirme yapıldıktan sonra Parçacık Sürü Optimizasyonu hakkında bilgiler verilmiştir. Çalışmada önerilen PSO algoritmasının ana adımları maddeler halinde belirtilmiştir. MATLAB’ da gerçekleştirilen çalışmaya ait veriler sunulmuştur.

Bölüm 7’de PSO destekli pencere fonksiyonu tasarımına ait program ve verilere yer verilmiştir. Bu bölümde, PSO algoritmasıyla elde edilen cosh pencere fonksiyonuna ait veriler LabVIEW ortamına aktararak LabVIEW ortamında tasarlanan cosh pencere fonksiyonu ile kıyaslanmıştır. Farklı iki yöntem ile tasarlanan cosh pencere fonksiyonlarıyla FIR sayısal filtre tasarımı gerçekleştirilmiştir. Elde edilen FIR filtreler ile EEG sinyali filtrelenerek elde edilen sonuçlar DAQ kart aracılığıyla gerçek zamanlı olarak görüntülenmiştir.

Son bölümde elde edilen sonuçlara doğrultusunda LabVIEW ve PSO ile tasarım hakkında bilgilendirme yapılmış ve gelecekte yapılabilecek çalışmaların neler olabileceğinden bahsedilmiştir.

2. SAYISAL FİLTRELER

Çalışma prensiplerine göre filtreler, analog ve sayısal olarak iki ana başlıkta incelenir. Analog filtreler, yapılarında direnç, kondansatör, işlemsel kuvvetlendirici gibi devre elemanları bulunduran elektronik devrelerdir. Analog filtreler, gürültü filtreleme, ses ve görüntü sinyalleri iyileştirme gibi birçok alanda kullanılmaktadırlar. Sayısal filtreler, girişlerine uygulanan analog sinyalin bir dizi işlem ile sayısallaştırılması ile elde edilen sayısal sinyali filtreleyerek istenilen çıkış sinyalini sunan algoritmalarlardır. Bahsi geçen işlem dizisi Şekil 2.1’de blok diyagram olarak ifade edilmiştir. Sayısal filtrelerin karışmış sinyalleri birbirinden ayırmak, sinyaldeki gürültüyü azaltarak sinyal kalitesini artırmak, bozulmuş sinyali iyileştirmek gibi uygulama alanları bulunmaktadır. Sayısal filtreler, analog filtrelere göre çevre şartları değişiminden etkilenmez, olayca gerçekleşir ve test edilebilir, işlevleri kolaylıkla değiştirilebilir, vb avantajlara sahiptir.

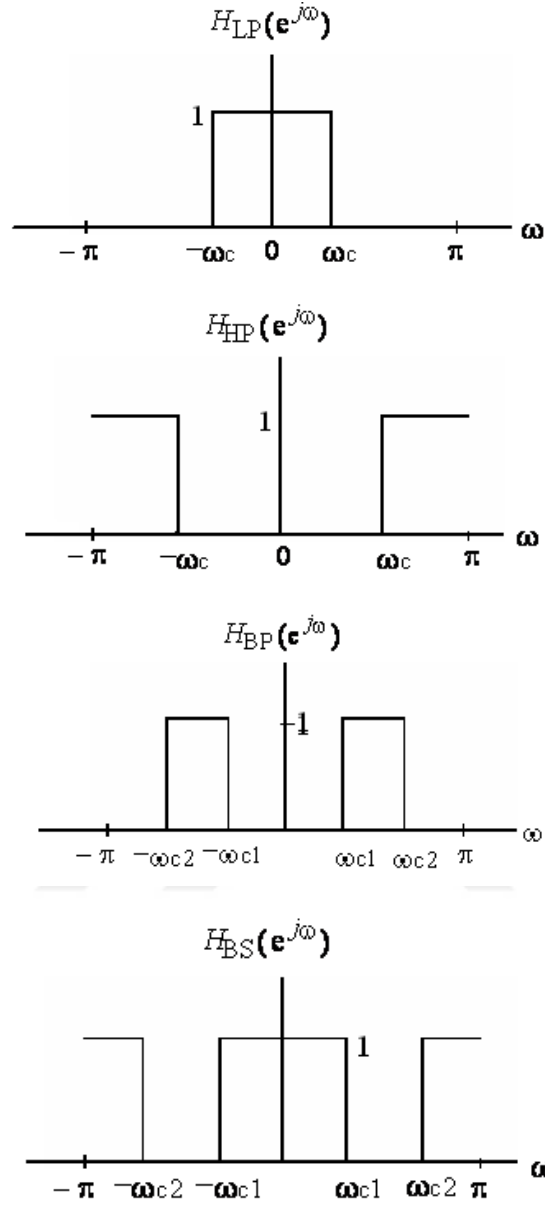


Şekil 2.1. Sayısal filtre blok diyagramı

Sayısal filtreler, geçirdikleri ya da zayıflattıkları frekans alanına bağlı olarak aşağıdaki gibi sınıflandırılabilir:

- Alçak Geçiren Filtreler (LPF)
- Yüksek Geçiren Filtreler (HPF)
- Band Geçiren Filtreler (BPF)
- Band Durduran Filtreler (BSF)

Bahsi geçen sayısal filtrelerin frekans karakteristikleri Şekil 2.2’ de gösterilmiştir:



Şekil 2.2. İdeal sayısal filtre frekans karakteristikleri; a) İdeal Alçak-geçiren b) İdeal Yüksek-geçiren c)İdeal Bant-geçiren d) İdeal Bant-durduran

Sayısal filtreleri sınıflandırmanın bir diğer yöntemi ise impuls cevaplarına göre sınıflandırmaktır. İmpuls cevabına göre filtreler:

- Sonlu İmpuls Cevaplı Filtreler (Finite Impulse Response, FIR)
- Sonsuz İmpuls Cevaplı Filtreler (Infinite Impulse Response, IIR)

olarak sınıflandırılır.

Uygulamalarda filtre seçimi genellikle uygulamanın gereklerine ve sonuçta beklenen frekans cevabına göre yapılmaktadır. Uygulama alanlarına göre sayısal filtre çeşitlerinin birbirine göre avantajları ve dezavantajları bulunmaktadır:

- FIR filtrelerde, IIR filtrelerin aksine herhangi bir andaki çıkış, girişin sadece o andaki değerine bağlıdır. Bu sebeple meydana gelecek hatalar geri besleme olarak sisteme eklenmeyeceğinden dolayı oluşan hatalar önemsizdir.
- FIR filtrelerin IIR filtrelere kıyasla en büyük avantajı doğrusal faz cevabına sahip olmalarıdır.
- IIR filtreler, FIR filtrelere göre daha düşük filtre derecesi ile istenilen frekans cevabını sunabilmektedirler.
- Bahsi geçen durumlar filtre tasarımını doğrudan etkilediğinden maliyeti de etkilemektedir.

Sayısal filtre fark denklemi aşağıdaki gibi ifade edilebilir:

$$b_0 y(n) = a_0 x(n) + a_1 x(n-1) + \dots + a_M x(n-M) - b_1 y(n-1) - \dots - b_N y(n-N) \quad (2.1)$$

Denklem 2.1 'de yer alan ifadeler aşağıdaki gibi açıklanır:

- $x(n)$, giriş sinyalidir.
- $y(n)$, çıkış sinyalidir.
- N , geri besleme filtre derecesidir.
- M , ileri besleme filtre derecesidir.

Denklem 2.1 düzenlendiğinde aşağıdaki ifade elde edilir.

$$y(n) = \frac{1}{b_0} \left(\sum_{k=0}^M a_k x(n-k) - \sum_{k=1}^N b_k y(n-k) \right) \quad (2.2)$$

Filtrenin transfer fonksiyonunu elde etmek için z-Dönüşümü kullanılır.

$$H(z) = \frac{Y(z)}{X(z)} \quad (2.3)$$

$$H(z) = \frac{\sum_{k=0}^M a_k z^{-k}}{b_0 + \sum_{k=1}^N b_k z^{-k}} \quad (2.4)$$

- $b_0 = 1$ ve $N = 0$ olması durumunda transfer fonksiyon düzenlendiğinde FIR filtresine ait transfer fonksiyon elde edilir:

$$H(z) = \sum_{k=0}^M a_k z^{-k} \quad (2.5)$$

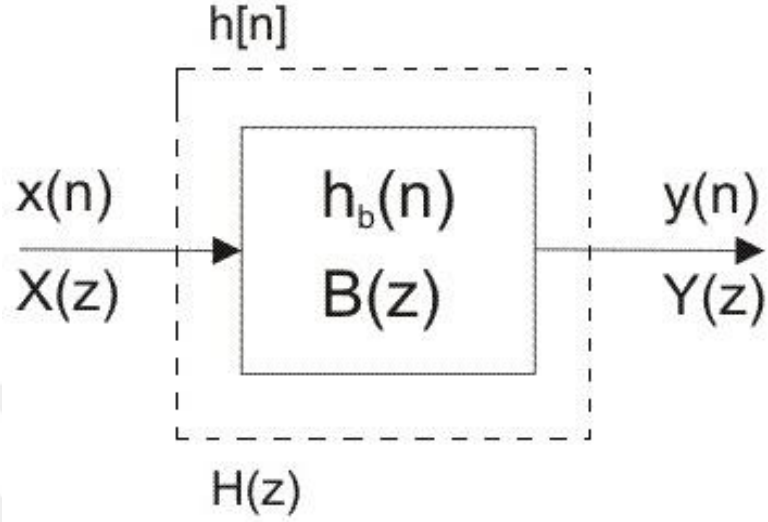
$$H(z) = \frac{Y(z)}{X(z)} = a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M} \quad (2.6)$$

- $b_0 = 1$ olması durumunda transfer fonksiyon düzenlendiğinde IIR filtresine ait transfer fonksiyon elde edilir:

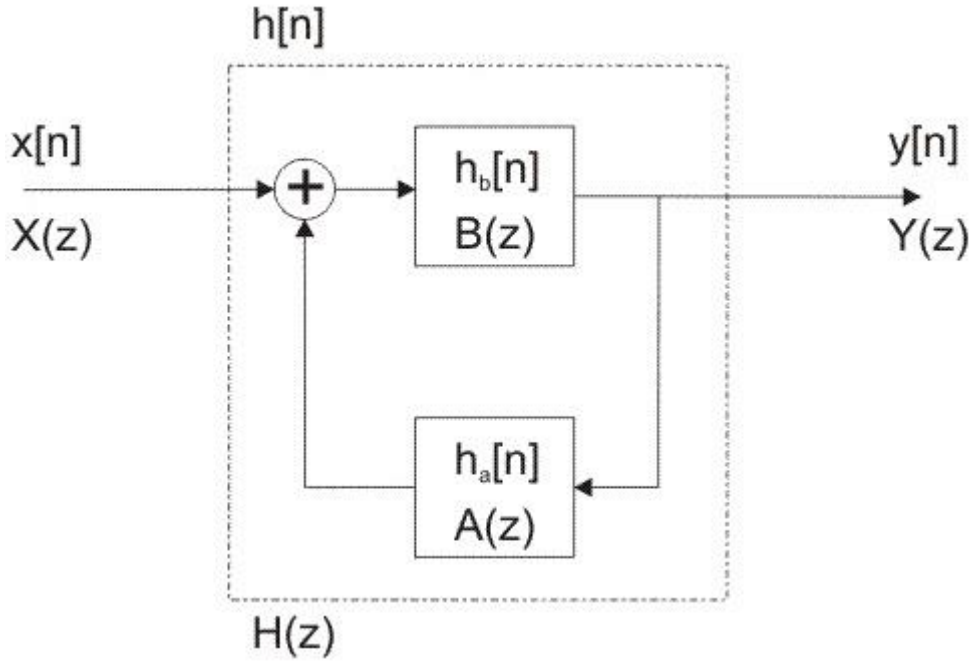
$$H(z) = \frac{\sum_{k=0}^M a_k z^{-k}}{1 + \sum_{k=1}^N b_k z^{-k}} \quad (2.7)$$

$$H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_M z^{-M}}{1 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}} \quad (2.8)$$

Sayısal filtrelerin yapısı, genel olarak Şekil 2.3'teki ve Şekil 2.4'teki gibi gösterilebilir:



Şekil 2.3. FIR filtre blok diyagramı



Şekil 2.4. IIR filtre blok diyagramı

Sayısal filtreleri yazılımsal veya donanımsal olarak tasarlamak mümkündür. Filtreler, yazılımsal olarak gerçekleştirmek için bilgisayar benzetim programlarından faydalanılırken donanımsal olarak gerçekleştirmek için mantıksal devre elemanları

kullanılır. Dolayısıyla, filtre tasarımı, zamandan ve maliyetten tasarruf etme, hata ayıklama sürecini azaltma vb. avantajlarının bulunması nedeniyle yazılımsal tasarım daha çok tercih edilmektedir.

Bir işaretin frekans spektrumunda, belirli frekans değerlerinde istenilen sonucu elde edebilecek yapılara sayısal filtre denilir. FIR filtre tasarımı Fourier Serisi yöntemi ve Frekans Örneklem yöntemi kullanılmaktadır. IIR filtre tasarımı ise değişmez-impuls cevabı yöntemi, değiştirilmiş değişmez-impuls cevabı yöntemi, uygunlaştırılmış z-dönüşümü yöntemi ve Bilineer dönüşüm kullanılmaktadır.

FIR sayısal filtrenin Fourier serisi yöntemi ile tasarlanması durumunda, impuls cevabının sınırlandırılmasından dolayı keskin kesim frekansı bölgesinde istenmeyen salınımlar meydana gelir. Bu istenmeyen salınımları ortadan kaldırmak için pencere fonksiyonları geliştirilmiştir. Pencere fonksiyonu tasarımı temel amaç, istenmeyen salınımları ortadan kaldırmak ve bu sayede istenilen özellikleri tamamen sağlayabilecek filtre tasarlamaktır.

3. PENCERE FONKSİYONLARI

Giriş ifadesi $x(n)$, çıkış ifadesi $y(n)$ olan sayısal filtre fark denklemi aşağıdaki gibi ifade edilir [44]:

$$b_0 y(n) = a_0 x(n) + a_1 x(n-1) + \dots + a_M x(n-M) - b_1 y(n-1) - \dots - b_N y(n-N) \quad (3.1)$$

$$\left(\sum_{k=0}^N b_k y(n-k) = \sum_{k=0}^M a_k x(n-k) \right) \quad (3.2)$$

Filtrenin transfer fonksiyonunu elde etmek için z-Dönüşümü kullanılır [45].

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M a_k z^{-k}}{1 + \sum_{k=1}^N b_k z^{-k}} \quad (3.3)$$

Sayısal bir filtrenin frekans tepkisini bulmak için z-dönüşümü ile Laplace dönüşümü arasındaki $z = e^{sT}$ geçiş bağıntısı kullanarak elde edilebilir. Yakınsama bölgesi $j\omega$ eksenini kapsayan bir sistemin frekans cevabı $s = j\omega$ yazılarak bulunur.

$$H(z) = H(e^{sT}) = H(e^{j\omega T}) = H(e^{j\Omega}) \quad (3.4)$$

Denklem (3. 4)' te yer alan Ω , sayısal frekans olup periyodiktir.

Elde edilen periyodik frekans cevabı, Fourier serisi olarak yazılırsa,

$$H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h(n) e^{-jn\Omega} \quad (3.5)$$

Bu durumdaki frekans cevabı,

$$h(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(e^{j\Omega}) e^{jn\Omega} d\Omega \quad (3.6)$$

şeklinde ifade edilir.

Fourier serisi olarak yazılan filtrenin frekans cevabında $e^{j\Omega} = z$ bağıntısını kullanarak filtrenin transfer fonksiyonu elde edilir.

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-N} \quad (3.7)$$

Fourier serisi kullanarak tasarlanan Sayısal filtrenin transfer fonksiyonundan da anlaşılacağı üzere filtrenin impuls cevabı sonsuzdur. Bu durumda filtrenin pratikte gerçekleştirilebilmesi mümkün değildir. Filtrenin pratikte gerçekleştirilebilmesi için impuls cevabının sınırlandırılması gerekmektedir. Dolayısıyla impuls cevabı sınırlandırılan bu filtre bir FIR filtredir. İmpuls cevabının sınırlı sayıda olabilmesi için belirli bir gecikmeyle çarpılması gerekmektedir. Böylelikle sonsuz uzunluğa sahip olan sayısal filtre sonlu sayıda bir impuls cevabına sahip olacaktır.

$$h(n) = 0 \text{ ve } |n| > N' = \frac{N-1}{2} \quad (3.8)$$

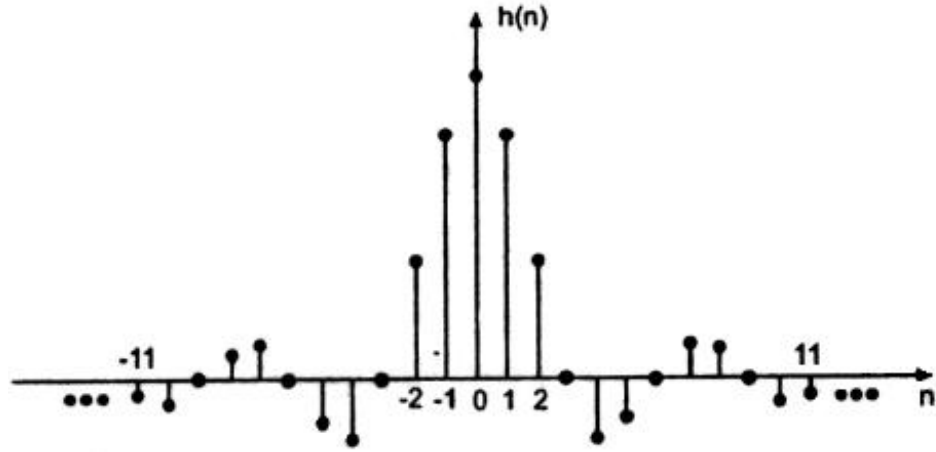
olması durumunda,

$$H(z) = h(0) + \sum_{n=1}^{N'} [h(-n)z^n + h(n)z^{-n}] \quad (3.9)$$

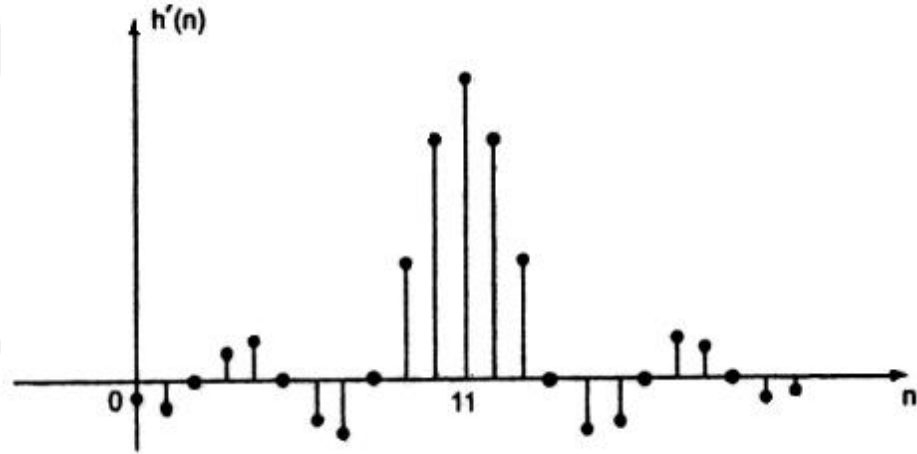
şeklinde bulunur. $H(z)$ nin $z^{-N'}$ ile çarpılmasıyla filtrede nedensellik şartı sağlanmış olur ve sonlu gecikme eklenerek genlik cevabının aynı fakat faz cevabının kaydırıldığı gözlenir.

$$H'(z) = z^{-N'} H(z) \quad (3.10)$$

Şekil 3.1’de sonsuz uzunluktaki ideal alçak geçiren sayısal filtrenin impuls cevabı ve Şekil 3.2’de $N=23$ uzunluğunda tasarlanan FIR sayısal filtrenin impuls cevabı görülmektedir.

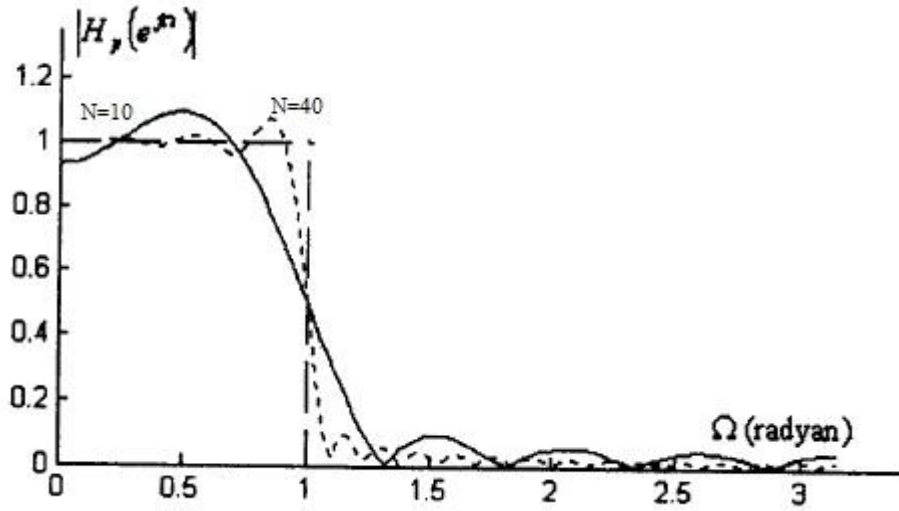


Şekil 3.1. Sonsuz uzunluktaki ideal alçak geçiren sayısal filtrenin impals cevabı



Şekil 3.2. Sonlu uzunlukta ($N=23$) tasarlanan FIR filtrenin impals cevabı

Fourier serisi kullanılarak tasarlanan FIR filtrenin pratikte gerçekleştirilebilmesi için impals cevabının sınırlandırılması gerekir ve bu işlem yüzünden filtrenin keskin kesim frekansı bölgesinde istenmeyen salınımlar meydana gelir. Kesim frekansı $\Omega_c=1$ ve farklı filtre dereceleri (N) için pratikte gerçekleştirilen filtrenin genlik cevabı Şekil 3.3'te gösterilmektedir. Aynı Şekil'de ideal filtrenin genlik cevabı da gösterilmektedir.



Şekil 3.3. Fourier Serisi Yöntemi ile tasarlanan FIR filtrelerin genlik cevapları

Şekil 3.3'ten de görüldüğü üzere filtre derecesinin artmasıyla tasarlanan FIR filtrenin genlik cevabı ideal filtrenin genlik cevabına yaklaşmaktadır.

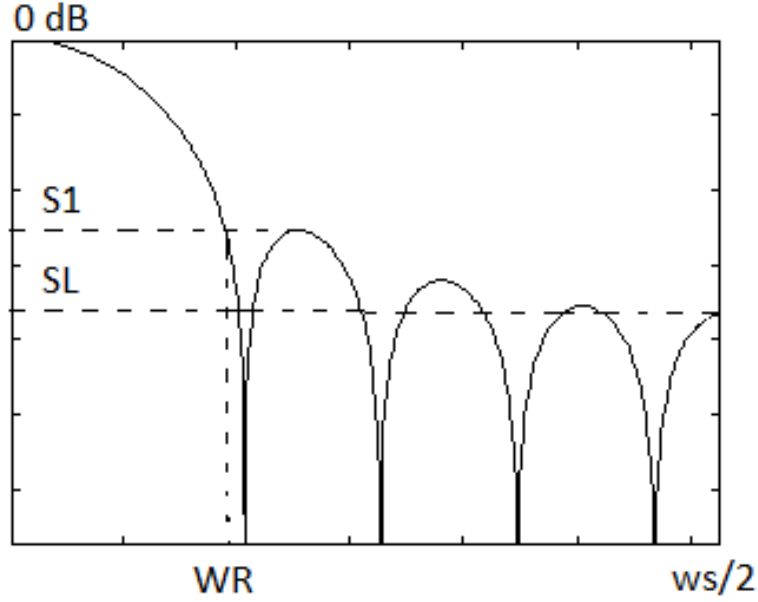
1899 yılında Gibbs tarafından matematiksel olarak ifade edilen bu istenmeyen salınımları ortadan kaldırmak için pencere fonksiyonları kullanılır. Pencere fonksiyonuyla, filtreden istenilen genlik cevabına en uygun parametre değerleriyle ve düşük filtre derecesiyle ulaşmak amaçlanmaktadır. Pencere fonksiyonuyla gerçekleştirilecek filtre tasarımında, filtrenin ideal genlik cevabı ile aynı uzunluktaki pencere fonksiyonu çarpılarak tasarlanacak filtrenin katsayıları elde edilir.

$$h_p = h_{id} * w[n] \quad (3.11)$$

Geçmişten günümüze pencere fonksiyonları gelişimi için birçok çalışma gerçekleştirilmiştir. Son yıllarda yapılan çalışmalarda pencere parametrelerinin belirlenmesinde akıllı hesaplama yöntemleri kullanılmaktadır [13, 27, 28]. Yapılan çalışmalarla, geliştirilen pencere fonksiyonları ile tasarlanacak filtrelerin genlik cevabından beklenen özellikler sağlanmaya çalışılarak filtrenin çalışma verimliliğini artırmak hedeflenmektedir. Pencere fonksiyonu ile tasarlanmış bir filtrenin genlik spektrumundan beklenen özellikler aşağıdaki gibidir:

- Ana lob genişliği dar olmalıdır.
- Dalgalanma oranı küçük olmalıdır.
- Yan lob azalma oranı geniş olmalıdır.

Pencere fonksiyonlarının genel olarak genlik spektrumu aşağıdaki gibidir:



Şekil 3.4. Pencere Fonksiyonu Spektral Gösterimi

Şekil 3.4'te gösterilen pencere fonksiyonu spektral gösteriminde yer alan ifadeler aşağıdaki gibi belirtilir:

R = Maksimum Yan Lob Genliği – Ana Lob Genliği ($R=S1$)

S = Maksimum Yan Lob Genliği – Minimum Yan Lob Genliği ($S=S1-SL$)

3.1. Yaygın Olarak Kullanılan Pencere Fonksiyonları

Literatürde pencere fonksiyonlarının birçok alanla kullanımı bulunmaktadır. Parametrelerine göre pencere fonksiyonları sabit pencere fonksiyonları ve ayarlanabilir pencere fonksiyonları olmak üzere iki ana başlık altında incelenebilir. Sabit pencere fonksiyonları, pencere uzunluğunu parametresi ile yalnızca pencere fonksiyonunun analob genişliğini ayarlanabilmektedir. Literatürde yaygın olarak kullanılan sabit pencere fonksiyonları ve denklemleri aşağıda verilmiştir:

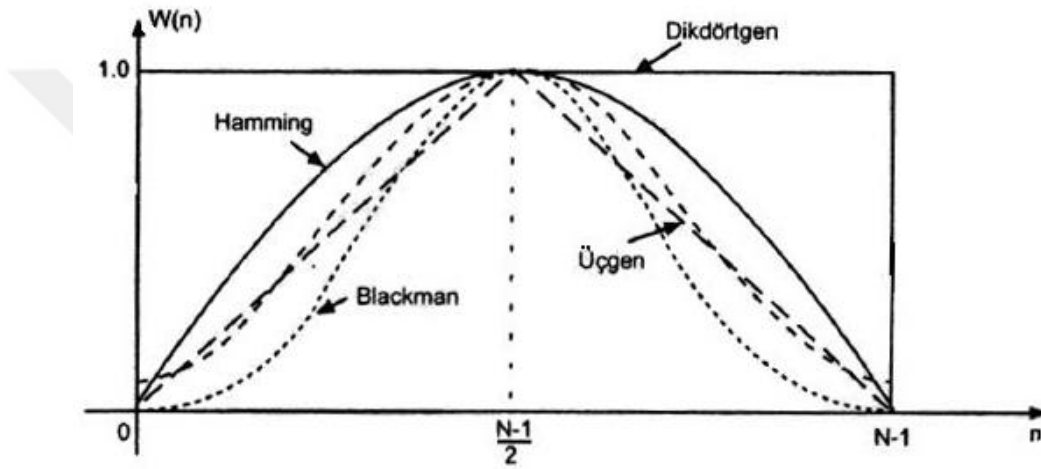
- Üçgen penceresi ; $w[n] = 1 - \left| \frac{2n}{N} \right|$ (3.12)

- Blackman penceresi ; $w[n] = 0.42 + 0.5 \cos\left(\frac{2\pi n}{N}\right) + 0.08 \cos\left(\frac{4\pi n}{N}\right)$ (3.13)

- Hamming penceresi ; $w[n] = 0.54 + 0.46 \cos\left(\frac{2\pi n}{N}\right)$ (3.14)

- Hanning penceresi ; $w[n] = 0.5 + 0.5 \cos\left(\frac{2\pi n}{N}\right)$ (3.15)

Şekil 3.5'te yaygın olarak kullanılan pencere fonksiyonlarının genlik cevapları görülmektedir:



Şekil 3.5. FIR filtre tasarımında kullanılan pencere fonksiyonları genlik cevapları

Ayarlanabilir pencere fonksiyonlarında ise iki veya daha çok parametre kullanılmaktadır. Bu parametrelerden pencere uzunluğu parametresi ile analob genişliği ayarlanabilirken diğer parametrelerle diğer pencere spektral parametrelerini ayarlanabilmektedir. Literatürde kullanılan ve en çok tercih edilen ayarlanabilir pencere fonksiyonları aşağıdaki gibidir:

- Kaiser Pencere Fonksiyonu: Pencere uzunluğu (N) ve ayarlanabilir α_k parametrelerine bağlıdır.

$$w[n] = \begin{cases} \frac{I_0\left(\alpha_k \sqrt{1 - \left(\frac{2n}{N-1}\right)^2}\right)}{I_0(\alpha_k)}, & |n| \leq \frac{N-1}{2} \\ 0, & \text{diğer yerlerde} \end{cases} \quad (3.16)$$

Burada α_k ayarlanabilir parametredir, $I_0(x)$ ise Bessel fonksiyonu olup güç serisi açılımı aşağıdaki gibidir.

$$I_0(x) = 1 + \left[\frac{1}{k} \left(\frac{x}{2} \right)^k \right]^2 \quad (3.17)$$

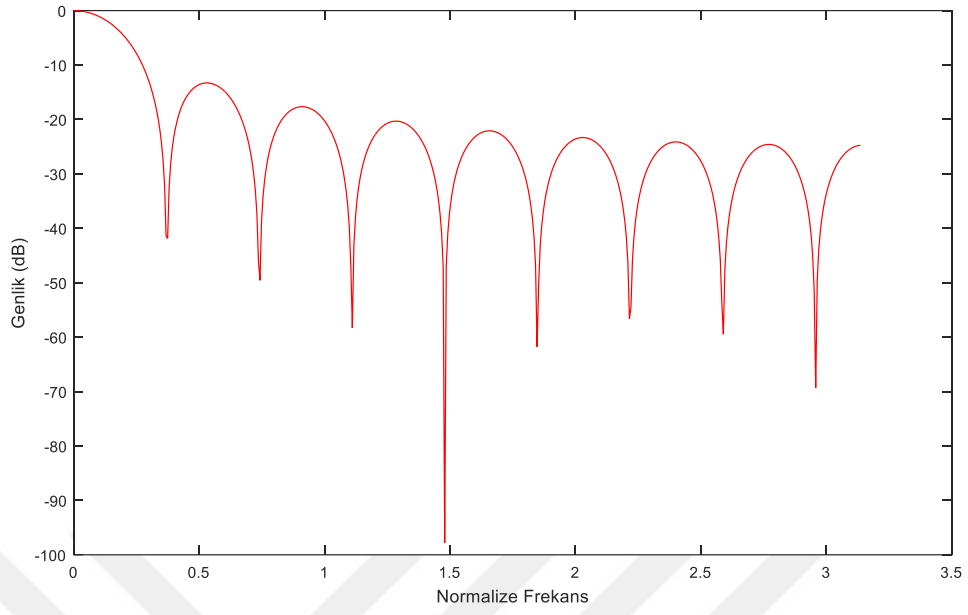
- Üstel Pencere Fonksiyonu: Kaiser pencere fonksiyonunda kullanılan Bessel fonksiyonunun yerine üstel fonksiyon yazılarak oluşturulmuştur.

$$w[n] = \begin{cases} \frac{\exp\left(\alpha_e \sqrt{1 - \left(\frac{2n}{N-1}\right)^2}\right)}{\exp(\alpha_e)}, & |n| \leq \frac{N-1}{2} \\ 0, & \text{diğer yerlerde} \end{cases} \quad (3.18)$$

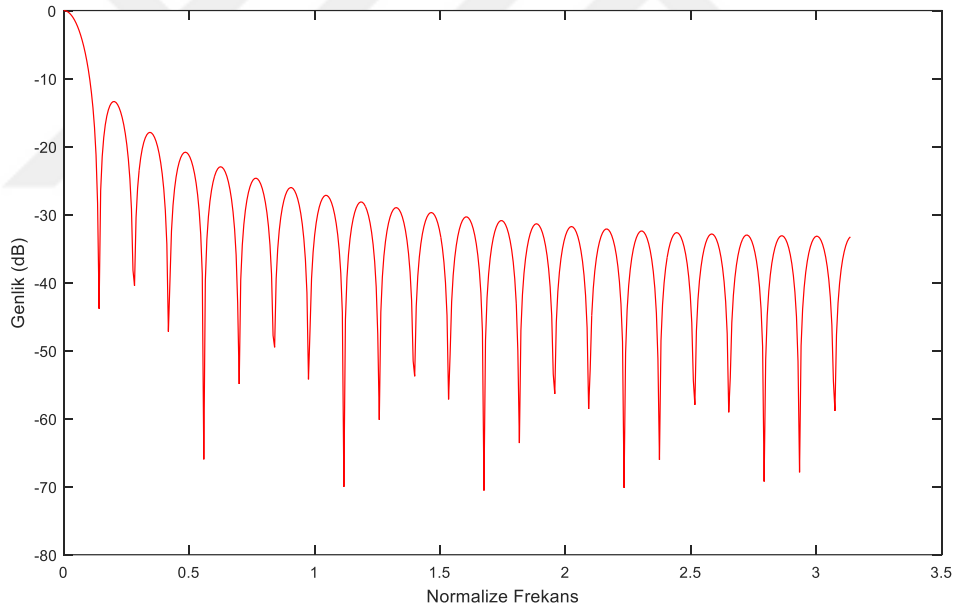
- Cosh Pencere Fonksiyonu: Üstel pencere fonksiyonunda olduğu gibi Kaiser pencere fonksiyonunda yer alan Bessel fonksiyonu kaldırılmıştır ve yerine cosh fonksiyonu eklenerek elde edilmiştir.

$$w[n] = \begin{cases} \frac{\cosh\left(\alpha_c \sqrt{1 - \left(\frac{2n}{N-1}\right)^2}\right)}{\cosh(\alpha_c)}, & |n| \leq \frac{N-1}{2} \\ 0, & \text{diğer yerlerde} \end{cases} \quad (3.19)$$

Bahsi geçen pencere fonksiyonları ayarlanabilir parametrelere sahiptir. N , pencere uzunluğu ve α , ayarlanabilir parametre olarak adlandırılır. Bu parametrelerin pencere fonksiyonu üzerindeki etkisini inceleyebilmek için cosh pencere fonksiyonu ile yapılan uygulamaya ait grafikler aşağıda sunulmuştur. Şekil 3.6'da $N=17$ ve $\alpha=0.2$ için cosh pencere fonksiyonu spektrumu görülürken Şekil 3.7'de $N=45$ ve $\alpha=0.2$ için cosh pencere fonksiyonu spektrumu görülmektedir. Tablo 3.1'de ise Şekil 3.6 ve Şekil 3.7'deki spektrumlardan elde edilen veriler sunulmuştur. Şekil 3.8'de $N=25$ ve $\alpha=0.1$ için cosh pencere fonksiyonu spektrumu ve Şekil 3.9'da $N=25$ ve $\alpha=3.5$ için cosh pencere fonksiyonu spektrumu görülmektedir. Tablo 3.2'de ise Şekil 3.8 ve Şekil 3.9'daki spektrumlardan elde edilen veriler sunulmuştur.



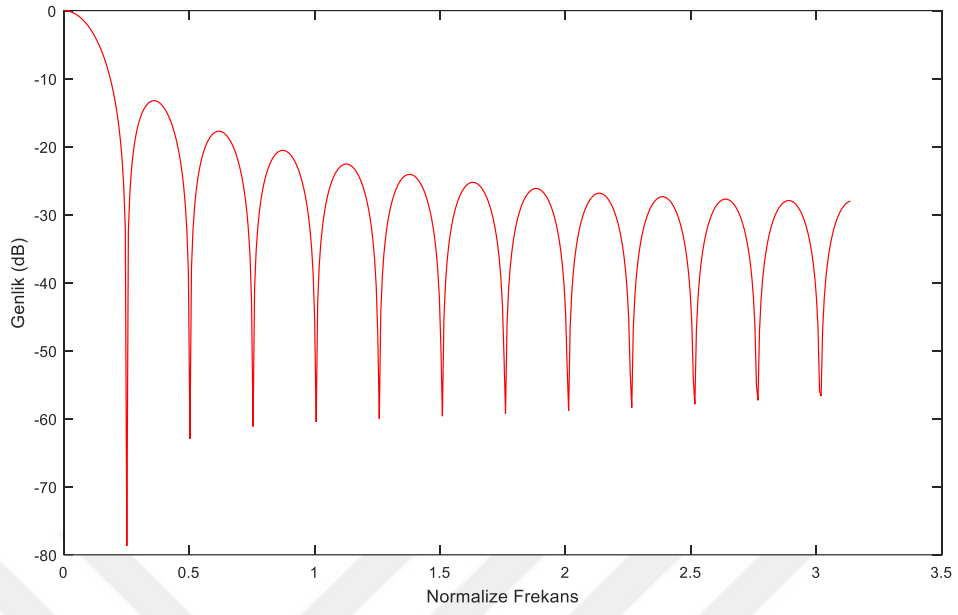
Şekil 3.6. $N=17$ ve $\alpha=0.2$ için Cosh Pencere Fonksiyonu Spektral Gösterimi



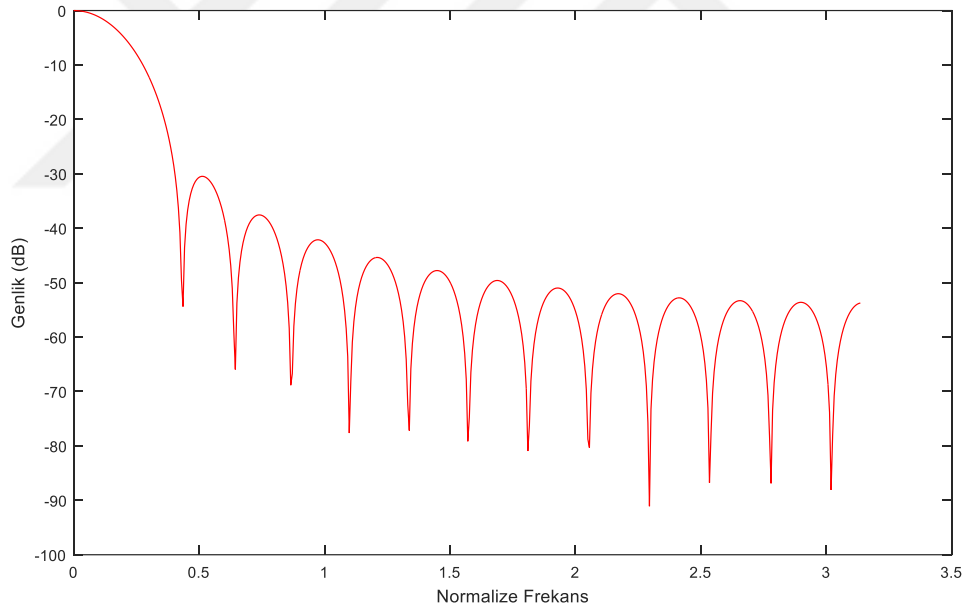
Şekil 3.7. $N=45$ ve $\alpha=0.2$ için Cosh Pencere Fonksiyonu Spektral Gösterimi

Tablo 3.1. Değişken N değerleri için Cosh Penceresine ait Veriler

Pencere Türü	N	α	R	ω_R	S
Cosh	17	0.2	-13.29	0.302	11.46
Cosh	45	0.2	-13.38	0.114	19.89



Şekil 3.8. $N=25$ ve $\alpha=0.1$ için Cosh Pencere Fonksiyonu Spektral Gösterimi



Şekil 3.9. $N=25$ ve $\alpha=3.5$ için Cosh Pencere Fonksiyonu Spektral Gösterimi

Tablo 3.2. Değişken α değerleri için Cosh Penceresine ait Veriler

Pencere Türü	N	α	R	ω_R	S
Cosh	25	0.1	-13.25	0.21	14.77
Cosh	25	3.5	-30.47	0.40	23.27

Bu tezde yer alan ilk uygulamada, LabVIEW ortamında cosh ve üstel pencere fonksiyonu tasarımı gerçekleştirilmiştir ve tasarlanan pencere fonksiyonlarının farklı parametre değerlerinde oluşan genlik spektrumları kıyaslanmıştır. Tezde yer alan ikinci uygulamada, LabVIEW ortamında tasarlanan cosh pencere fonksiyonuyla bir FIR filtre gerçekleştirilmiştir. Aynı zamanda Parçacık Sürü Optimizasyonu (PSO) yöntemiyle tasarlanan alternatif cosh pencere fonksiyonu değerleri LabVIEW ortamına aktarılmıştır ve bu değerlerle LabVIEW’de FIR sayısal filtre tasarımı gerçekleştirilmiştir. Farklı yöntemler kullanılarak tasarlanan FIR filtreler ile EEG sinyali filtrelenerek sonuçlar gerçek zamanlı incelenmiştir.



4. PROGRAMLAMA DİLLERİ VE LABVIEW

Bilgisayarın icadıyla birlikte belirli bir işlemi gerçekleştirebilmek için komutlara; komutları gerçekleştirebilmek için ise program diline ihtiyaç duyulmuştur. İstenilen işlemleri gerçekleştirmek için gerekli olan program dillerinin gelişim tarihi makine dili ile başlamıştır. Makine dili sayesinde elektronik devrelere yüksek yada düşük voltaj uygulanmasıyla cihazların işleyişi kontrol edilmeye başlanmıştır. Programlamada yüksek voltaj 1 ile, düşük voltaj 0 ile ifade edilmiştir. 1 ve 0'ların oluşturduğu komutlardan meydana gelen makine dili ile gerçekleştirilen programlamada komutlar direkt olarak cihaz üzerinde yer alan panel ile herhangi bir işleme maruz kalmadan işlemciye uygulanabilmiştir. Bu özellikleri sayesinde hız ve yetenek bakımından güçlü olmasına rağmen hata durumlarını yönetmekte sorunlar yaşanmıştır. Ayrıca belirli bir işlemci için yazılan kod, farklı bir işlemcide çalışmayıp tamamen yenilenmesi gerekmiştir. Programlamadaki bu problemlerden kurtulmak için İngilizce kelimelerin kısaltılmasıyla elde edilen komutlar geliştirilerek Assembly dili tasarlanmıştır. Fakat Assembly dili ile tasarlanan programların cihazlarda çalışabilmesi için bir çevirici (assembler) ile makine diline dönüştürülmesi gerektiğinden pek kullanışlı olmamıştır.

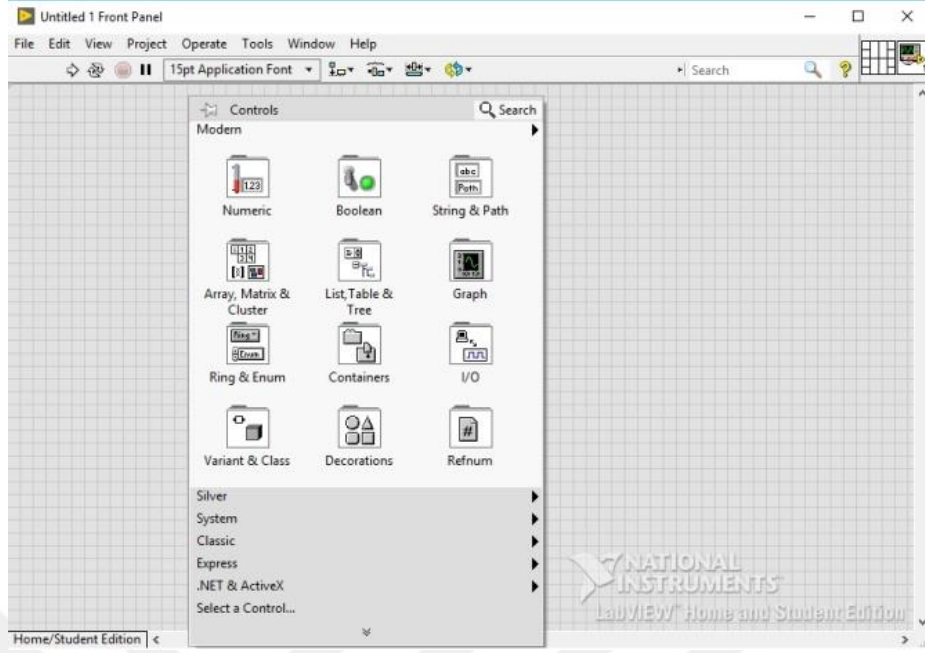
Programlama yapabilmek için tek bir makineye bağlı olma, her program için baştan bilgisayar donanımı kurma, istenilen işlemleri uzun komutlarla gerçekleştirme gibi dezavantajların ortadan kaldırılması için yapılan çalışmalar bazı kavramları ortaya çıkarmıştır: Subroutine (yordam), library, döngü (loop), değişken, lojik vb. Yapılan çalışmalar sonucunda 1968 yılında Pascal programlama dili geliştirilmiştir. Ardından 1972 yılında C programlama dili, Pascal programlama dilindeki birçok hatanın giderilmesiyle oluşturulmuştur. Yapılan iyileştirmelerle sırasıyla C++ , Java, Delphi, C# programlama dilleri geliştirilerek programlama dillerinin performansı arttırılmıştır ve arttırılmaya yönelik çalışmalar günümüzde devam etmektedir.

Buraya kadar bahsi geçen ve yapıda olan bütün programlama dilleri metin tabanlı olup olay güdümlü (event driven programming) programlama olarak tanımlanmaktadır ve görsel bir özelliği yoktur. Son yıllardaki gelişmeler ile tamamen

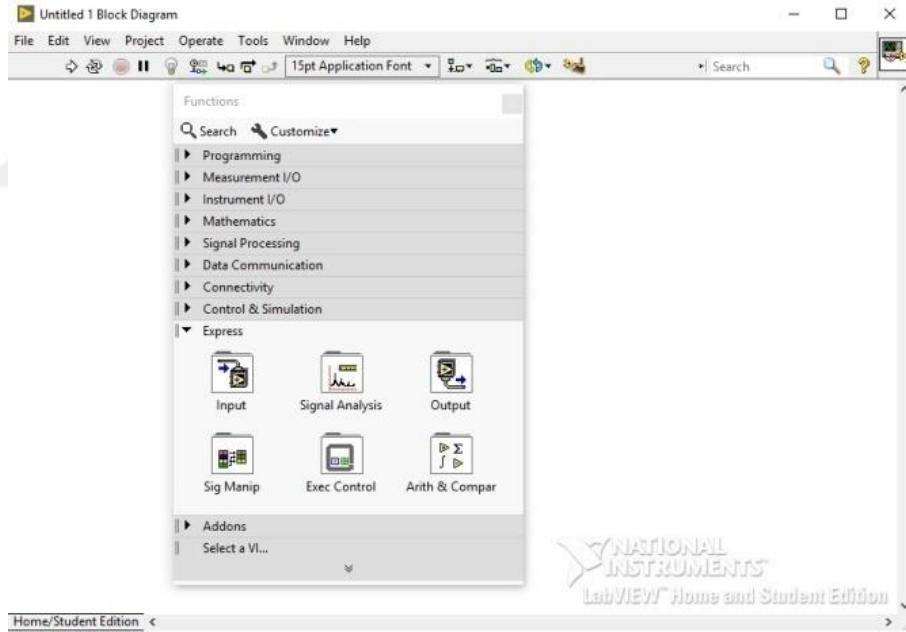
metin tabanlı komutlardan meydana gelen diller yerine görsel tabanlı diller geliştirilmektedir. Bu diller ile komut yazma zorunluluğu kısmen ortadan kaldırılıp program yazma işlemi daha da kolaylaştırmak hedeflenmektedir. Bu tür diller görsel programlama dili olarak adlandırılır. Her ne kadar bu gelişmiş diller görsel tabanlı olsa da kod yazma işleminden tamamen kurtulmak mümkün olmamıştır.

1976 yılında ABD’de kurulan National Instruments firması tarafından geliştirilen LabVIEW (Laboratory Virtual Instruments for Engineering Workbench), metin tabanlı programlama dillerine oranla daha hızlı ve daha basit programlama ortamı sağlayan bir platformdur. Bir başka deyişle LabVIEW, veri toplama, veri kontrolü, veri analizi, veri sunumu için kullanılan Grafiksel Programlama Dili (Graphical Programming Language, GPL) tabanlı interaktif bir program geliştirme ve uygulama sistemidir. Görsel tabanlı olan bu programda hazır fonksiyonlar sayesinde kod yada komut ezberleme gibi zorluklar ortadan kaldırılmıştır. Programlamaya getirdiği kolaylıklarla diğer programlama dillerine oranla daha hızlı çözüm üretebilmektedir. Hızlı programlama, paralel program sürdürebilme, hızlı hata ayıklama gibi özellikleri sayesinde programlamada büyük bir avantaja sahiptir [46-47].

LabVIEW ekranı iki kısımdan oluşur: Front Panel, Block Diagram. Front panel olarak adlandırılan kısım kullanıcı arayüzüdür. Bu kısımda uygulamaya ait giriş ve çıkış değerlerinin kontrolü bir operatör ile kontrol edilir. Block Diagram ise işlemlerin gerçekleştirildiği kısımdır. Bu kısımda her olay bir şemayla veya bir nesneyle ifade edilir ve bu ifadeler geliştirilen algoritmaya uygun olacak şekilde birbirlerine hatlarla bağlanırlar. Front Panel ve Block Diagrama ait görüntüler Şekil 4.1 ve Şekil 4.2’ de görülmektedir.



Şekil 4.1. LabVIEW Front Panel



Şekil 4.2. LabVIEW Block Diagram

Ölçme ve kontrol sistemlerinde yeterince etkin olan LabVIEW, National Instruments bünyesinde geliştirilen veri toplama kartları, modüler enstrümantasyon sistemleri, kompakt kartlar ile dünya piyasasında önemli yer edinmiştir. Gerçek zamanlı veri üretebilmek veya ölçmek için ölçme kartlarından faydalanarak veri toplama, veri analizi, veri görüntüleme işlemleri gerçekleştirilebilir. LabVIEW ile entegre çalışabilen

National Instruments tarafından üretilen donanımlardan biri olan DAQ (Data Acquisition) kartlar ile veriler, birçok kanaldan elde edilebilir veya sunulabilir. Şekil 4.3'te çalışmada kullanılan DAQ-6001 kartı görülmektedir.



Şekil 4.3. DAQ-6001 Kartı

5. SEZGİSEL OPTİMİZASYON YÖNTEMLERİ

Tasarlanan bir sistemin isteklerini, beklenen şekilde karşılayan çok sayıda muhtemel alternatif çözümler olabilir. Muhtemel çözümler arasında en iyisinin en kısa zamanda bulunması işlemine optimizasyon denilmektedir. Bir başka deyişle optimizasyon, bir sistemde, belirli şartlar altında ve belirenmiş bir amaç doğrultusunda en iyi çözüme minimum sürede ulaşılması işlemidir.

Optimizasyon problemlerinin çözümünde matematiksel yöntemler kullanılmaktaydı. Bu yöntemlerin çözüm için esnek olmaması, büyük boyutlu problemlerin çözümü için kullanışlı olmaması gibi dezavantajlarının olması optimizasyon tekniklerini geliştirmeye ve performanslarını arttırmaya yönelik çalışmalar yapılmasına sebep olmuştur. Doğadaki olayların temel alınmasıyla algoritmalar geliştirilmiştir. Bu tür geliştirilen algoritmalara sezgisel optimizasyon algoritmaları denilmektedir. Literatürde yer alan sezgisel optimizasyon çalışmalarında Genetik Algoritmalar (GA), Diferansiyel Gelişim Algoritması (Differential Evolution Algorithm, DE), Parçacık Sürü Optimizasyonu (PSO), Gri Kurt Optimizasyonu (Gray Wolf Optimization, GWO) yaygın olarak kullanılan algoritmalar [4, 8, 13, 14, 28].

Sezgisel optimizasyon yöntemi, büyük boyutlu problemlerin uygun hesaplama süresi içerisinde optimal çözümü elde etmek için geliştirilmiş yöntemlerden biridir. Sezgisel optimizasyon yöntemleri farklı problemler için geliştirilmiş alt problem tekniklerine sahiptir. Bu teknikler fizik, biyoloji, kimya, sosyal, sürü, bilgisayar ve karar üretme bilimlerinden türetilmiştir. Bu tez çalışmasında kullanılan ve sezgisel optimizasyon yöntemlerinden biri olan Parçacık Sürü Optimizasyonu, sürü tabanlı algoritmalarındandır.

5.1. Parçacık Sürü Optimizasyonu

1955 yılında Kenedy ve Eberhart tarafından geliştirilen Parçacık Sürü Optimizasyonu (PSO), sürü halinde hareket edebilen kuş, balık gibi havyaların davranışlarından ilham alınarak tasarlanmıştır [48]. Sürüde yer alan bireylerin bilgi

paylaşma, zengin besin kaynağına yönelme, bulundukları çevreye uyum sağlama gibi davranışları göz önünde bulundurulmuştur. PSO, hızlı sonuç bulması, az parametre gerektirmesi, etkili global arama özelliği ve matematiksel temelli algoritmalarla kıyasla uygulanmasının kolay olması, daha az hafıza gerektirmesi, hesapsal olarak hızlı ve etkili sonuç vermesi gibi özellikleriyle diğer algoritmalarla göre daha iyi performans sunmaktadır [49].

PSO’ da her biri potansiyel çözüm olan ve parçacık olarak adlandırılan bireyler, problem uzayında gezinerek en iyi çözüme ulaşmaya çalışırlar. Her bir parçacık optimize edilmiş problemin muhtemel çözüm adayıdır. Parçacığın en iyi kendi çözüm değeri (Particle_Best) ve tüm parçacıklar arasında bulunan en iyi çözüm değeri (Global_Best) hafızada saklanır. Her parçacığın pozisyonu, sürüdeki en iyi pozisyona göre ayarlanır. Durdurma kriteri sağlanınca algoritma durdurulur ve çözüm bulunan en iyi değer olan Global_Best olarak tanımlanır.

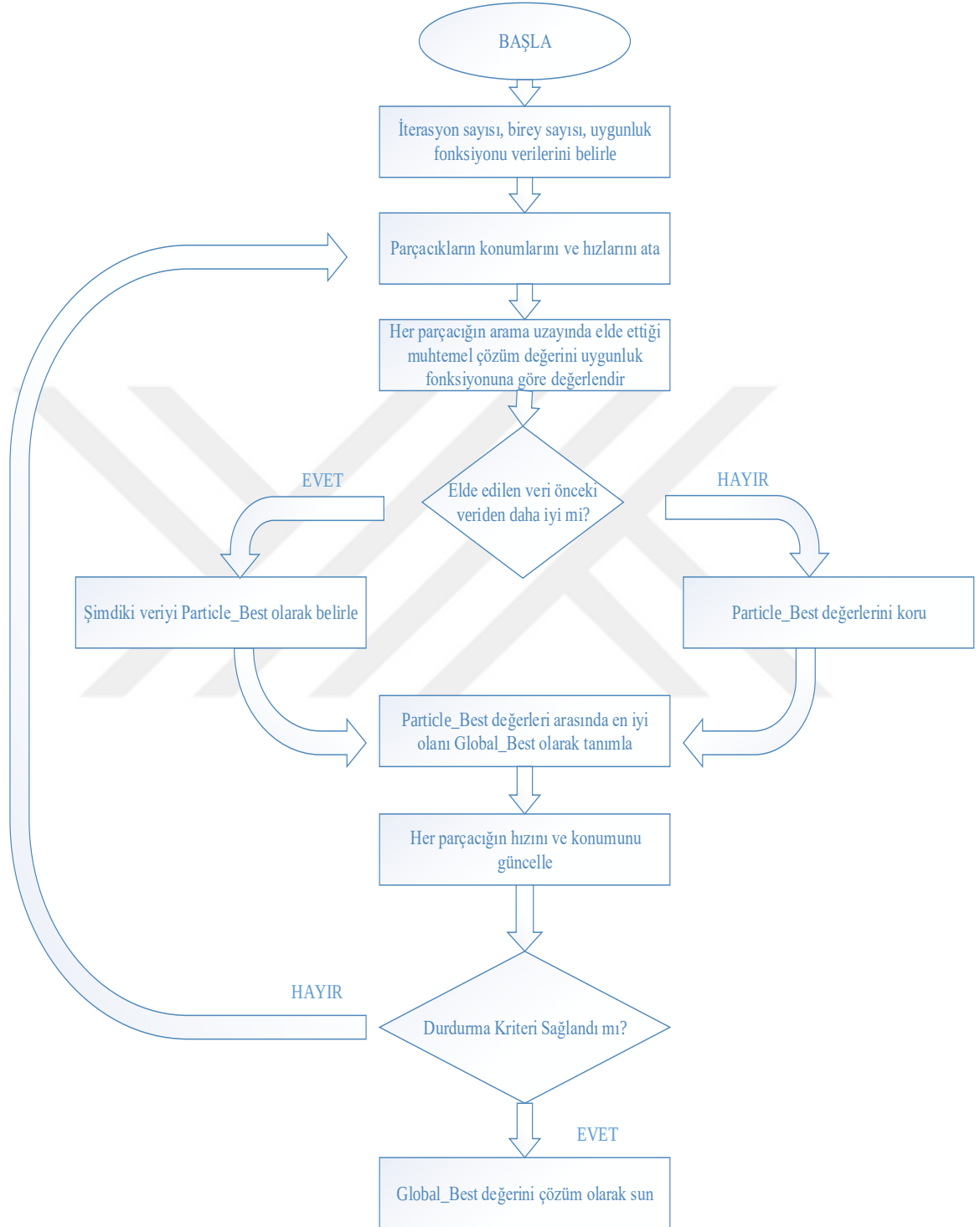
PSO algoritması, konumları ve hızları rastgele belirlenen parçacıkların popülasyonu ile başlar. Parçacıkların yönleri ve hızları, bulunan çözüme göre her iterasyonda güncellenir. Bulunan çözümler, uygunluk fonksiyonuna göre değerlendirilir. Uygunluk fonksiyonu, çözümü değerlendirebilmek için referans olarak belirlenen fonksiyondur. Arama uzayında bulunan sonuçlar uygunluk fonksiyonuna göre değerlendirildikten sonra Particle_Best ve Global_Best değerleri belirlenir. Parçacıkların hızları ve konumları bu değerlere bağlıdır ve sırasıyla aşağıdaki gibi ifade edilir:

$$V_i^{(k+1)} = \omega * V_i^{(k)} + c_1 * rand_1^{(k)} * (Particle_Best_i^{(k)} - x_i^{(k)}) + c_2 * rand_2^{(k)} * (Global_Best^{(k)} - x_i^{(k)}) \quad (5.1)$$

$$x_i^{(k+1)} = x_i^{(k)} + V_i^{(k+1)} \quad (5.2)$$

Denklem 5. 1’de yer alan c_1 ve c_2 katsayıları öğrenme faktörü olarak adlandırılır ve parçacıkları *Global_Best* ve *Particle_Best* konumlarına yönlendirmektedir. Tasarım için kullandığımız algoritmada $c_1 = c_2 = 2.1$ seçilmiştir. $rand_1^{(k)}$ ve $rand_2^{(k)}$ katsayıları 0 ve 1 aralığında rastgele üretilmekte ve her iterasyonda değişmektedir. k katsayısı iterasyon değerini ifade etmektedir. ω katsayısı ise eylemsizlik ağırlığıdır ve 0.1 ve 1 aralığında değişmektedir. Tasarım için kullandığımız algoritmada $\omega = 0.76$ seçilmiştir.

Bu açıklamalar doğrultusunda PSO akış diyagramı aşağıdaki gibi ifade edilebilir:



Şekil 5.1. Parçacık Sürü Optimizasyonu Akış Diyagramı

Şekil 5.1’de akış diyagramı verilen PSO algoritmasının sözde algoritması aşağıda sunulmuştur:

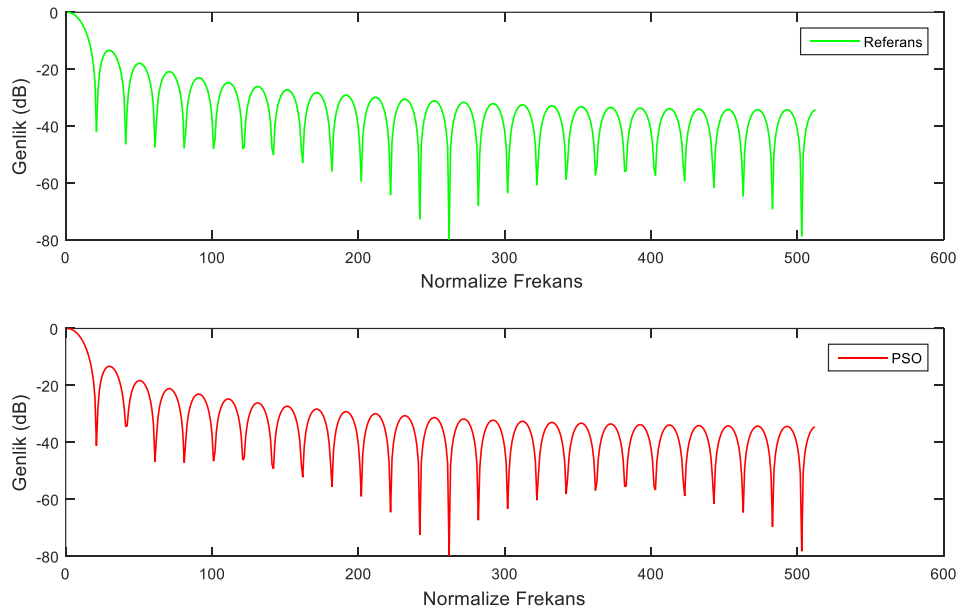
```
FOR Her parçacık için
    Rastgele konum ve hız değeri ata
END
DO
FOR Her parçacık için
    Uygunluk değerini hesapla
    IF Parçacığın bulduğu muhtemel çözüm değeri, önceki Particle_Best
    değerinden daha iyi ise
        Bulunan değeri, yeni Particle_Best değeri olarak ata
    END
    Particle_Best değerlerinden en iyisini Global_Best değerine ata
FOR Her parçacık için (5.1) denklemine göre parçacık hızını hesapla
    (5.2) denklemine göre parçacık konumunu güncelle
END
WHILE Durdurma kriteri sağlanıncaya kadar devam et
```

Yukarıda, akış diyagramı ve sözde algoritması sunulan PSO algoritması genel özelliklerinden bahsedilmiştir. Bu tez çalışmasında, PSO kullanılarak pencere fonksiyonu tasarımı gerçekleştirilmiştir. Çalışmada kullanılan PSO algoritması ile önerilen tasarımın ana adımları aşağıdaki gibi özetlenebilir:

- **Adım1.** Parametre değerleri belirlenen cosh pencere fonksiyonu spektrumu elde edilir. Bu sayede, belirlenen parametre değerleriyle oluşturulan cosh pencere fonksiyonuyla bulunan spektrum ve değerler referans olarak belirlenir.
- **Adım2.** PSO parametreleri belirlenerek parçacıkların başlangıç hızları ve pozisyonları rastgele atanır.
- **Adım3.** Her bir parçacığın bulduğu çözüm değeri hesaplanır. Bulunan en iyi çözüme göre parçacıkların pozisyonları her iterasyonda güncellenir. Bu durum optimizasyon sonlanana kadar tekrar eder.

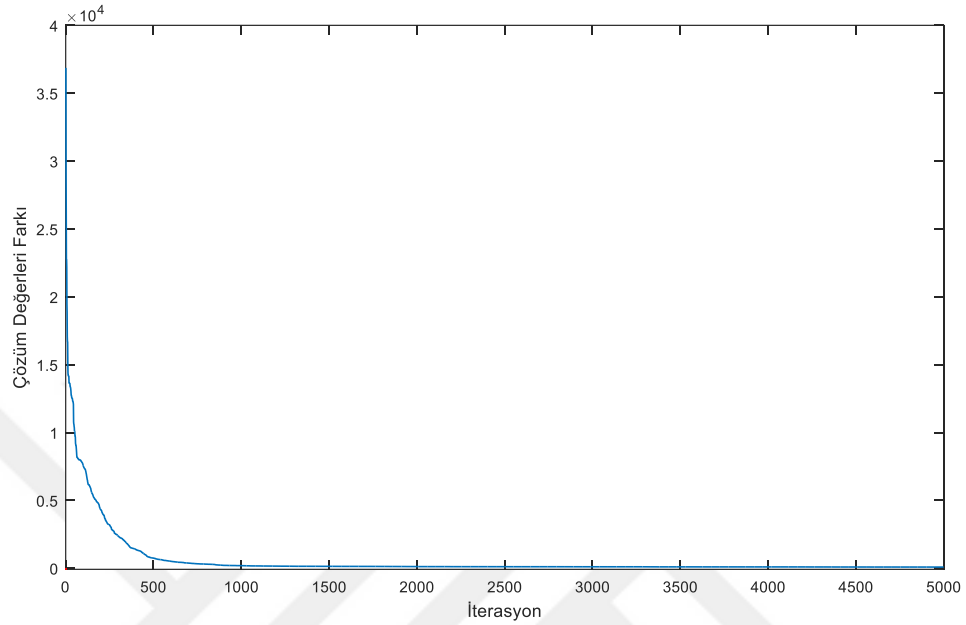
- **Adım4.** Optimizasyon sonlanana kadar bulunan en iyi çözümler ile Adım1’de referans olarak belirlenen değerler arasındaki fark her iterasyon için hesaplanır.
- **Adım5.** Optimizasyon sonlandığında bulunan en iyi çözüm değerleri (Global_Best) bir metin dosyasına kaydedilir ve bu değerler kullanılarak pencere fonksiyonu spektrumu elde edilir.
- **Adım6.** PSO ile tasarlanan pencere fonksiyonu ile kesim frekansı belirlenmiş bir FIR filtre tasarlanır ve verileri alınan bir EEG sinyali filtrelenir. Aynı işlemler Adım1’de bahsi geçen referans olarak belirlenen pencere fonksiyonu için de uygulanır.

PSO yöntemi ile referans olarak belirlenen cosh pencere fonksiyonuna alternatif bir cosh pencere fonksiyonu tasarlanmıştır. Bu sayede, MATLAB’ da aynı parametre değerleri için aynı işlemi gerçekleştirebilecek, farklı değerlere sahip alternatif bir pencere fonksiyonu gerçekleştirilmiştir. Pencere uzunluğu $N=51$, ayarlanabilir α parametresi 0.2 olarak belirlenen referans olarak tasarlanan cosh pencere fonksiyonu ile PSO yöntemi kullanılarak tasarlanan cosh pencere fonksiyonu genlik spektrumları Şekil 5.2’de görülmektedir.



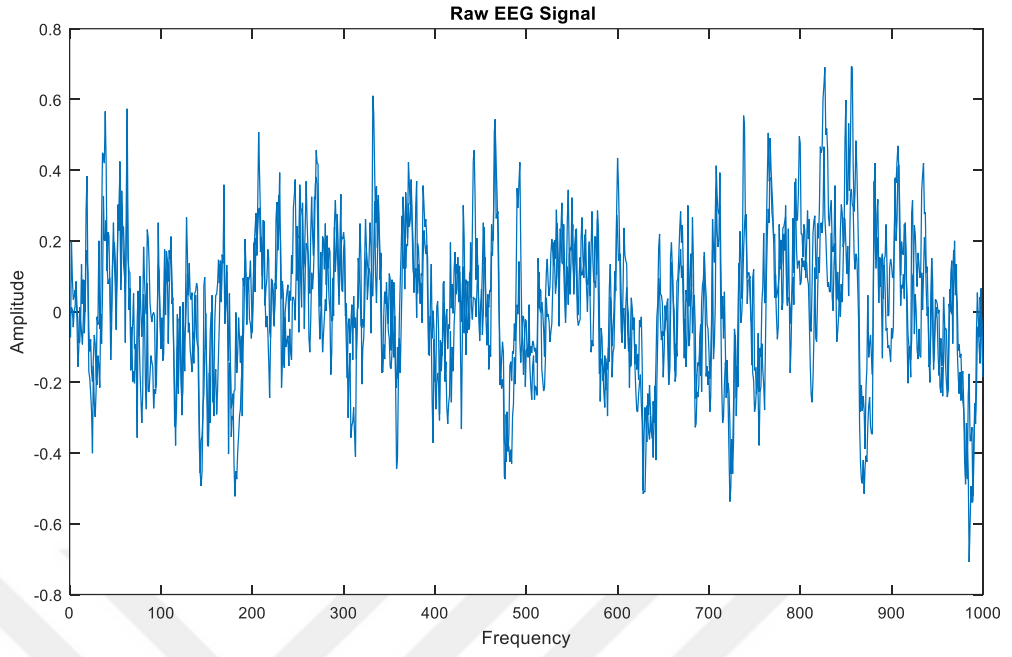
Şekil 5.2. Pencere Fonksiyonları Spektral Gösterimi

Referans olarak belirlenen pencere fonksiyonu ile PSO ile tasarlanan pencere fonksiyonunu arasındaki fark her iterasyon değeri için hesaplanmıştır ve Şekil 5.3'te görülmektedir.



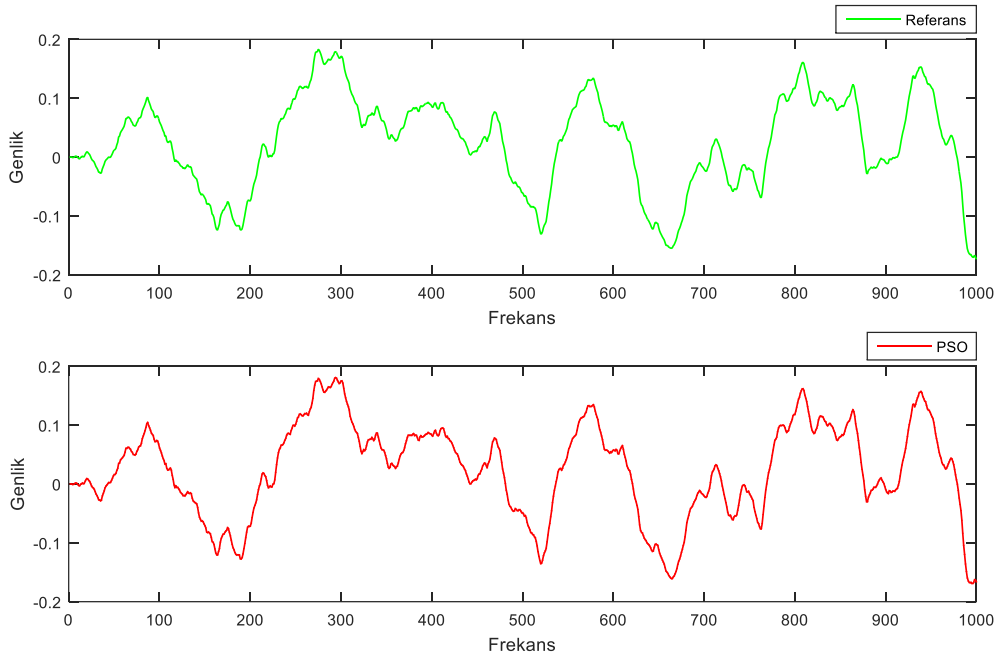
Şekil 5.3. Her İterasyonda İki Pencerenin Spektral Farkı

Tasarlanan cosh pencere fonksiyonlarıyla tasarlanan filtreler ile filtrelenen EEG sinyali Şekil 5.4'te görülmektedir. Kullanılan EEG sinyali Almanya Bonn Üniversitesi Epileptoloji Bölümü veri tabanından alınmıştır [50].



Şekil 5.4. Ham EEG Sinyali

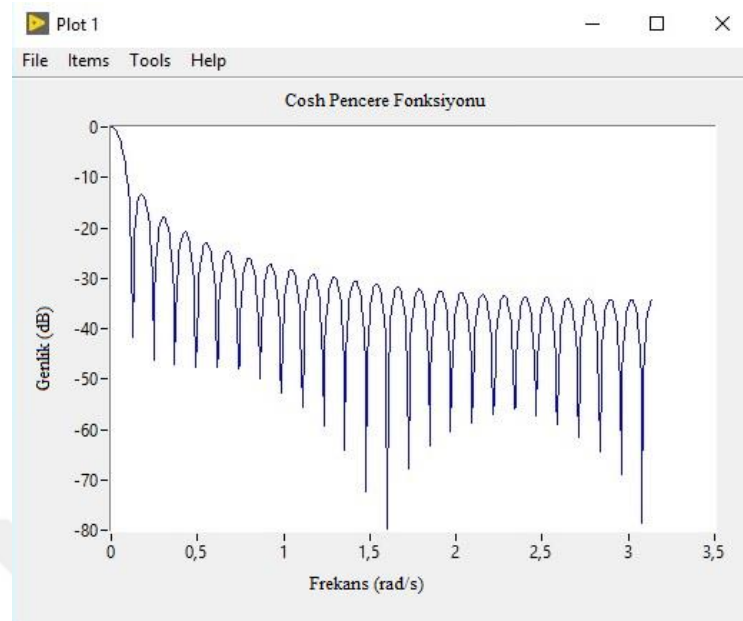
Tasarlanan filtrelerle filtrelenmiş EEG sinyalinin MATLAB simülasyonları Şekil 5.5'te sunulmuştur.



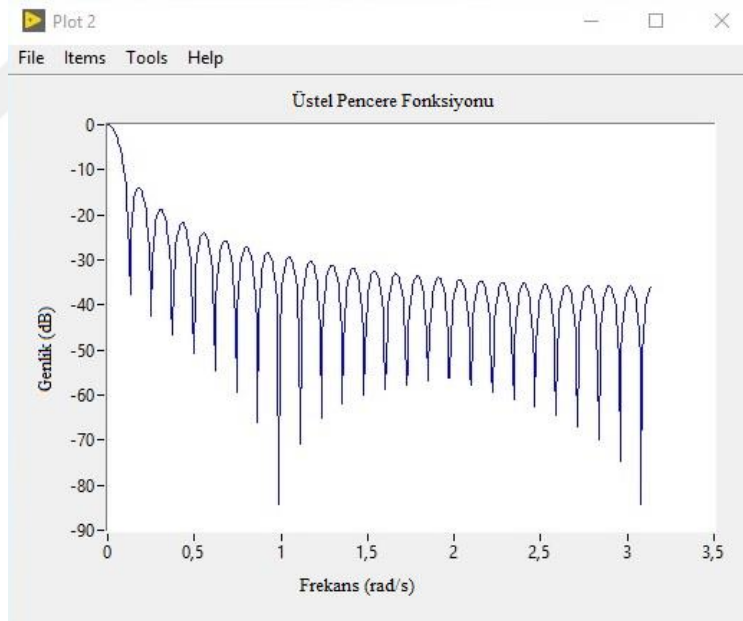
Şekil 5.5. Filtrelenmiş EEG Sinyalleri

Bu çalışmayla, Parçacık Sürü Optimizasyonu ile alternatif cosh pencere fonksiyonu tasarlanmıştır ve tasarlanan pencere fonksiyonu referans ile kıyaslanıp aralarındaki fark her iterasyon için hesaplanmıştır. Şekil 5.3 incelendiğinde, referans pencere fonksiyonu değerleri ile tasarlanacak pencere fonksiyonu değerleri arasındaki fark her iterasyonda azalmıştır. PSO yönteminin karakteristiği ve hızlı çözüme ulaşabilme özelliği sayesinde tasarlanan pencere fonksiyonu referans pencere fonksiyonuna bir alternatif olmuştur ve algoritma her çalıştırıldığında farklı değerler ürettiği unutulmamalıdır. Bu çalışma, tasarım çalışmaları için PSO' nun uygun bir yöntem olduğunun bir örneğidir.





Şekil 6.2. $N=51$ ve $\alpha=0.2$ için Cosh Pencere Fonksiyonu Spektrumu

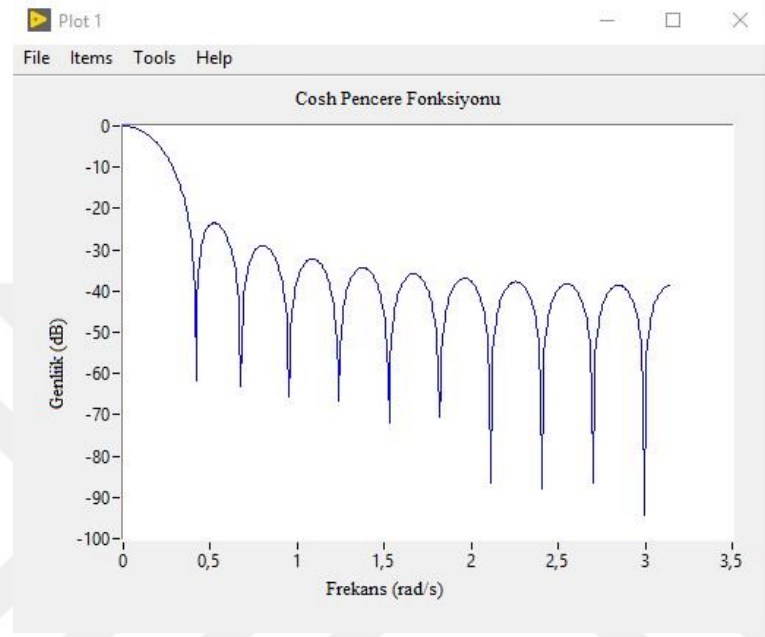


Şekil 6.3. $N=51$ ve $\alpha=0.2$ için Üstel Pencere Fonksiyonu Spektrumu

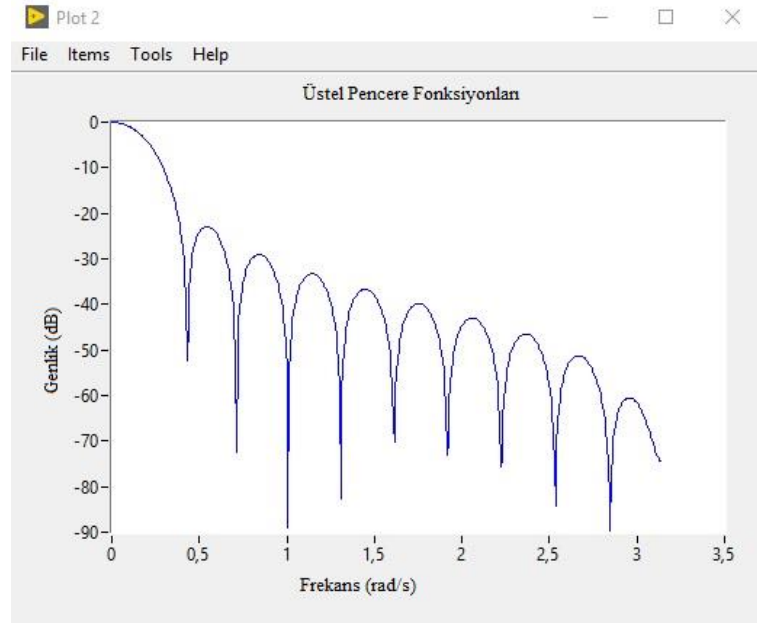
Tablo 6.1. $N=51$ ve $\alpha=0.2$ için Cosh ve Üstel Pencere Fonksiyonlarına ait Veriler

Pencere Türü	N	α	R	ω_R	S
Cosh	51	0.2	-13.38	0.1	20.88
Üstel	51	0.2	-14.03	0.1	21.84

$N=21$ ve $\alpha_c = \alpha_e = 2.2$ durumunda cosh ve üstel pencere fonksiyonlarına ait spektrumlar Şekil 6.4'te ve Şekil 6.5'te görülmektedir ve spektrumlardan elde edilen veriler Tablo 6.2'de sunulmaktadır:



Şekil 6.4. $N=21$ ve $\alpha=2.2$ için Cosh Pencere Fonksiyonu Spektrumu



Şekil 6.5. $N=21$ ve $\alpha=2.2$ için Pencere Fonksiyonu Spektrumu

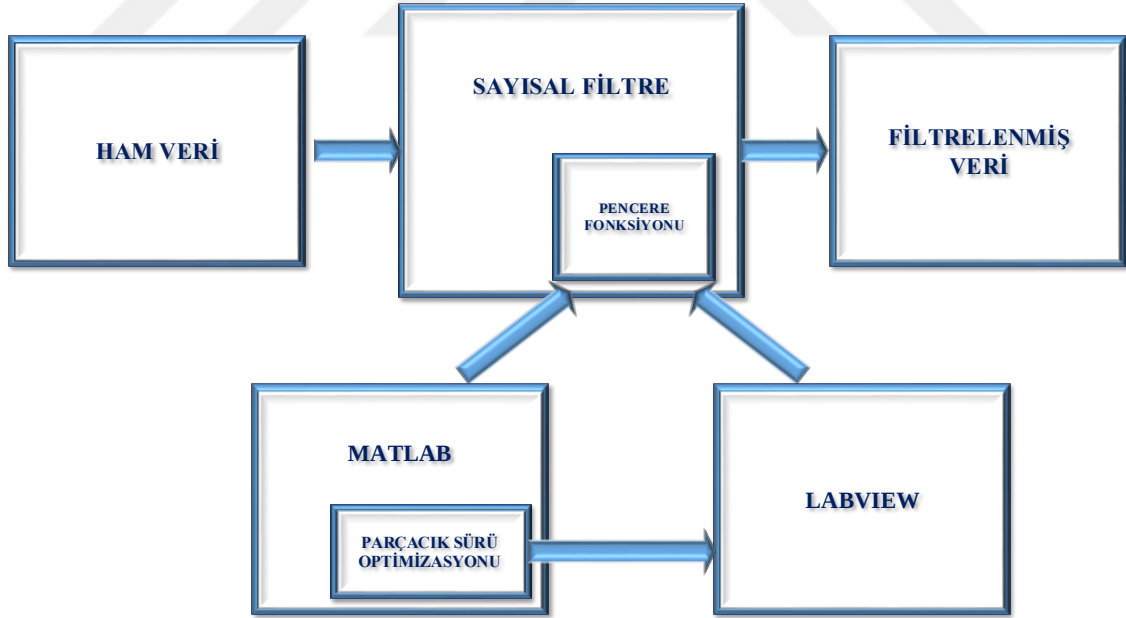
Tablo 6.2. $N=21$ ve $\alpha=2.2$ için Cosh ve Üstel Pencereleere ait Veriler

Pencere Türü	N	α	R	ω_R	S
Cosh	21	2.2	-23.55	0.38	15.16
Üstel	21	2.2	-22.9	0.38	37.58

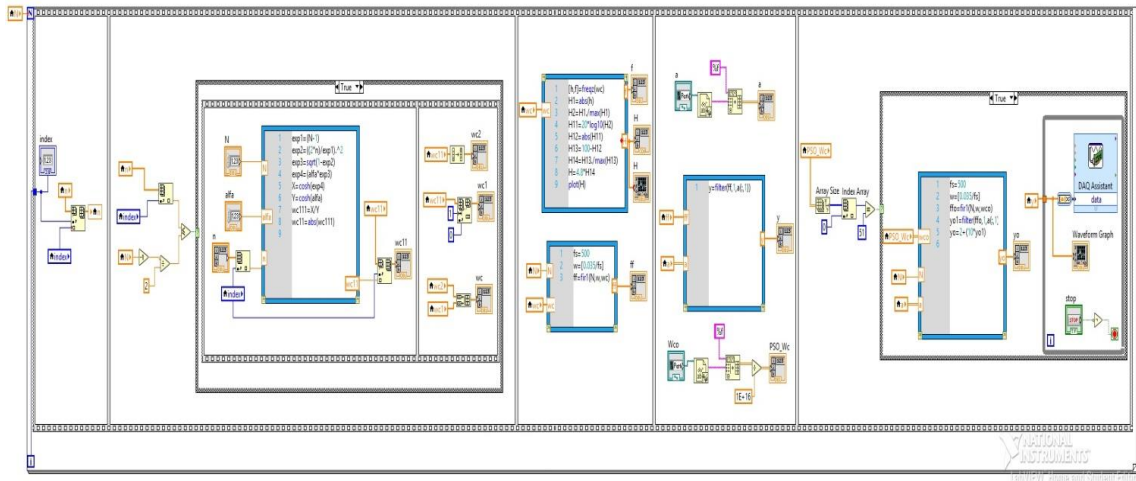
Bu uygulama ile ayarlanabilir parametrelere sahip cosh ve üstel pencere fonksiyonlarının farklı parametre değerlerinde meydana getirdiği genlik spektrumları gözlemlenmiştir. N ve α parametrelerinin ana lob genişliği, dalgalanma oranı ve yan lob azalma oranı üzerindeki etkiler gözlenmiştir. Elde edilen veriler ışığında farklı parametre değerlerinde farklı karakteristikler sergilendiği görülmektedir.

7. LABVIEW'DE PSO DESTEKLİ PENCERE FONKSİYONU GERÇEKLEŞTİRİLMESİ

Literatürde çok sayıda pencere fonksiyonu tasarımı ile ilgili çalışma bulunmaktadır. Bu uygulama ile LabVIEW'de cosh pencere fonksiyonu tasarımı gerçekleştirilmiştir. Aynı zamanda LabVIEW'in paralel programlama özelliğinden faydalanılarak MATLAB ortamında PSO kullanılarak tasarlanan alternatif pencere fonksiyonu değerleri LabVIEW ortamına aktarılmıştır. Bu sayede farklı iki yöntem ile tasarlanan cosh pencere fonksiyonları ile keskin kesim frekansı belirlenen FIR sayısal filtreler tasarlanmıştır. Tasarlanan filtreler ile verileri LabVIEW ortamına aktarılan EEG sinyali filtrelenmiştir ve elde edilen sonuçlar gerçek zamanlı olarak görüntülenmiştir. Sisteme ait blok diyagram Şekil 7.1'de ve LabVIEW'de gerçekleştirilen tasarım programı Şekil 7.2'de gösterilmiştir.

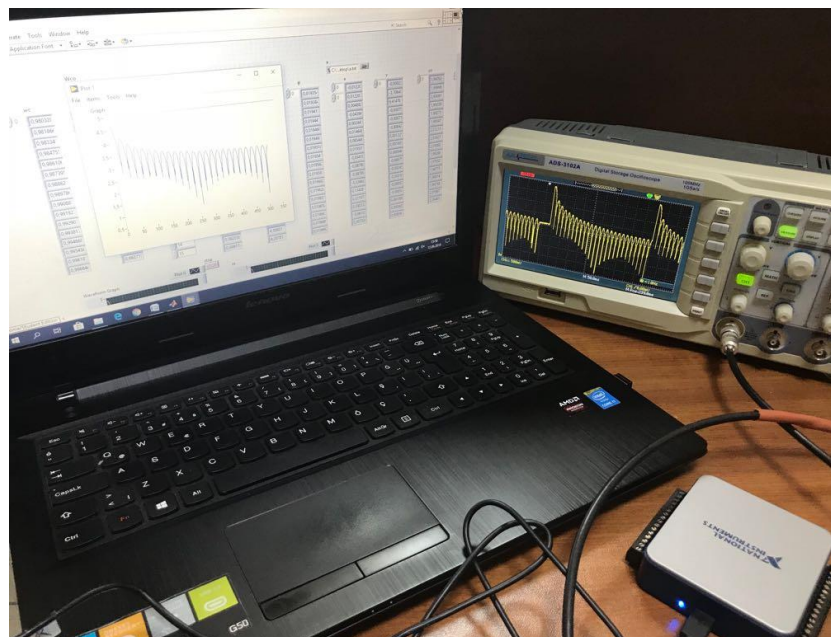


Şekil 7.1. Sistemin Blok Diyagramı



Şekil 7.2. LabVIEW’de Gerçekleştirilen Tasarım

Gerçek zamanlı veri oluşturmak ve toplamak için NI tarafından üretilen ölçüm kartları vardır ve bunlar LabVIEW ile entegre çalışabilmektedir. Bu uygulamada, NI tarafından üretilen ekipmanlardan biri olan DAQ-6001 ölçüm kartı kullanılarak tasarlanan filtreler ile filtrelenen EEG sinyali gerçek zamanlı olarak osiloskop aracılığıyla görüntülenmiştir. LabVIEW ortamında tasarlanan cosh pencere fonksiyonlarının farklı parametre değerleri için genlik spektrumları gerçek zamanlı görüntülenmiştir ve elde edilen simülasyonlar aşağıda sunulmuştur. Şekil 7.3'te gerçek zamanlı sinyal görüntüleme işleminin gerçekleştirildiği ekipmanlar ve çalışma ortamı görülmektedir.



Şekil 7.3. LabVIEW’de Tasarlanan Pencere Fonksiyonunun Gerçek Zamanlı Analizi

LabVIEW’de $N=51$ ve $\alpha = 0.2$ için tasarlanan pencere fonksiyonun genlik spektrumu Şekil 7.4’te gösterilmiştir ve R , S , ω_R değerleri gerekli hesaplamalar yapıldıktan sonra Tablo 7.1’de sunulmuştur.

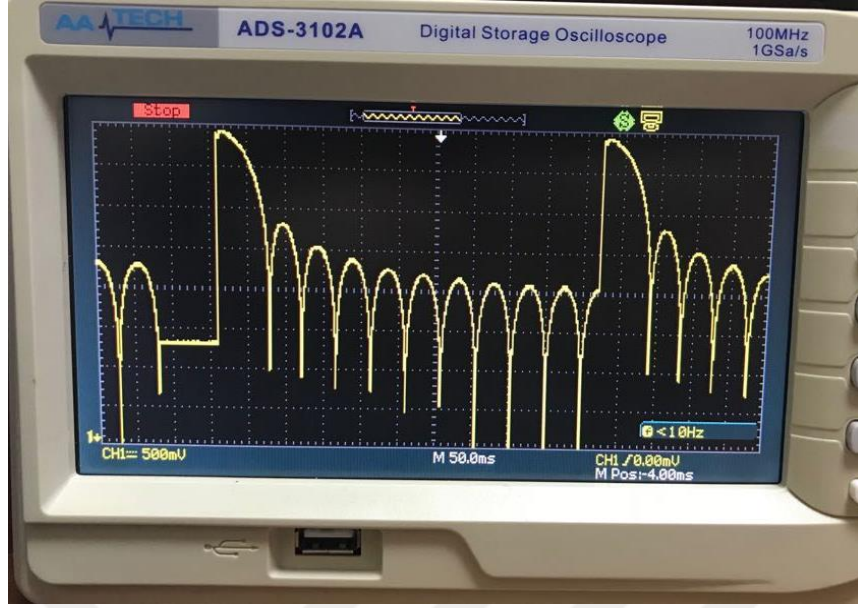


Şekil 7.4. $N=51$ ve $\alpha=0.2$ için Gerçek Zamanlı Cosh Pencere Fonksiyonu Spektral Gösterimi

Tablo 7.1. Cosh Pencere Fonksiyonunun $N=51$ ve $\alpha=0.2$ Değerlerindeki Verileri

N	α	R	ω_R	S
51	0.2	-13.38	0.1005	21

LabVIEW’de $N=21$ ve $\alpha = 2.2$ için tasarlanan pencere fonksiyonun genlik spektrumu Şekil 7.5’te gösterilmiştir ve R , S , ω_R değerleri gerekli hesaplamalar yapıldıktan sonra Tablo 7.2’de sunulmuştur.



Şekil 7.5. $N=21$ ve $\alpha=2.2$ için Gerçek Zamanlı Cosh Pencere Fonksiyonu Spektral Gösterimi

Tablo 7.2. Cosh Pencere Fonksiyonunun $N=21$ ve $\alpha=2.2$ Değerlerindeki Verileri

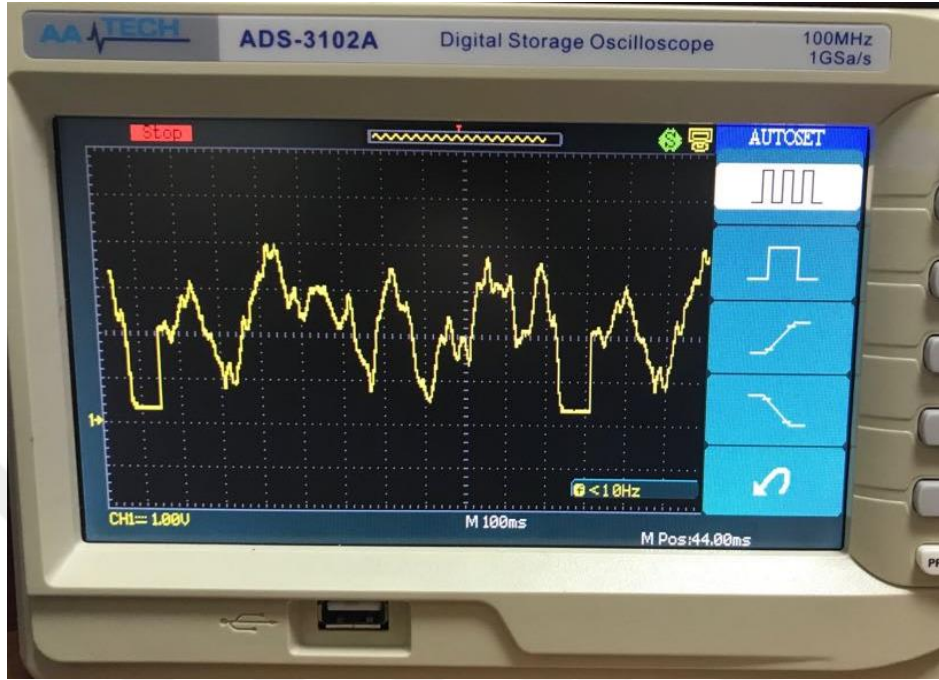
N	α	R	ω_R	S
21	2.2	-23.53	0.3783	15.18

LabVIEW’de tasarlanan pencere fonksiyonu kullanılarak tasarlanan filtre ile filtrelenmiş EEG sinyalinin gerçek zamanlı simülasyonu Şekil 7.6’da sunulmuştur.



Şekil 7.6. LabVIEW’de $N=51$ ve $\alpha=0.2$ için Tasarlanmış Pencere Fonksiyonu ile Tasarlanmış FIR Sayısal Filtre Kullanılarak Filtrelenmiş EEG Sinyali

LabVIEW’de, PSO ile tasarlanan pencere fonksiyonu kullanarak tasarlanan filtre ile filtrelenmiş EEG sinyalinin gerçek zamanlı simülasyonu Şekil 7.7’de sunulmuştur.



Şekil 7.7. LabVIEW’de PSO Kullanılarak $N=51$ ve $\alpha=0.2$ için Tasarlanmış Pencere Fonksiyonu ile Tasarlanmış FIR Sayısal Filtre kullanılarak Filtrelenmiş EEG Sinyali

8. SONUÇ VE GELECEK ÇALIŞMALAR

Ayarlanabilir pencere fonksiyonlarından cosh ve üstel pencere fonksiyonları diğer pencere fonksiyonlarına kıyasla daha güncel ve geliştirilmiştir. Ayrıca, seçilen bu pencereler ayarlanabilir iki parametreye sahip olduklarından ideal genlik cevabına ulaşmak daha mümkündür. Seçilen pencerelerden farklı, ayarlanabilir iki parametreye sahip olan Kaiser penceresi aynı parametre değerlerinde ana lob genişliği ve yan lob azalma oranı bakımında cosh ve üstel pencere fonksiyonlarına göre yetersiz kalmıştır. Bu bilgiler göz önüne alınarak pencere fonksiyonu tasarımı için yeni bir tasarım platformu öneren bu çalışma, belirlenen pencere fonksiyonlarının tasarımında başarılı olmuştur ve diğer pencere fonksiyonlarının LabVIEW’de tasarımı için örnek niteliğindedir.

Gerçekleştirilen ilk tasarım uygulamasında, LabVIEW ortamında tasarlanan cosh ve üstel pencere fonksiyonları tasarlanmıştır. Tasarlanan pencere fonksiyonları için farklı pencere uzunluğundaki (N) ve ayarlanabilir α parametresindeki performansları incelenmiştir ve genlik spektrumları çalışma içerisinde sunulmuştur.

Bu çalışmada, LabVIEW’in programlamada sunduğu avantajlardan faydalanılmıştır ve bu sayede LabVIEW ortamında gerçekleştirilen tasarım, Parçacık Sürü Optimizasyonu kullanılarak özelleştirilmiştir. Yapılan uygulamada, LabVIEW’de ve PSO yöntemi ile tasarlanan pencere fonksiyonları kullanılarak tasarlanan FIR filtreler ile EEG sinyali filtrelemiş ve filtrelenen sinyal gerçek zamanlı görüntülenmiştir. Aynı zamanda, LabVIEW’de tasarımı gerçekleştirilen cosh pencere fonksiyonunun farklı parametre değerlerindeki genlik spektrumları gerçek zamanlı olarak görüntülenmiş ve incelenmiştir. Bu işlemler, NI firması tarafından üretilen ve LabVIEW ile entegre çalışabilen kartlardan olan DAQ-6001 sayesinde gerçekleştirilmiştir. LabVIEW’in gelişmiş bu yazılımı sayesinde verileri anlık olarak görüntülenebilmekte ve kontrol edilebilmektedir. LabVIEW’in bu avantajları sayesinde veri analizi ve sunumu işlemleri kolaylıkla gerçekleştirilmektedir.

Ayrıca, MATLAB’ da Parçacık Sürü Optimizasyonu kullanılarak tasarlanan cosh pencere fonksiyonu, aynı parametre değerlerinde tasarlanan referans cosh pencere fonksiyonu ile kıyaslandığında pencere spektrumlarının yaklaşık olarak aynı olduğunu

gözlemlenmektedir. Bu durumda, PSO yöntemiyle tasarlanacak pencere fonksiyonuyla alternatif bir pencere fonksiyonu tasarımı gerçekleştirilebileceği sonucuna varılmaktadır. Sonuç olarak PSO, alternatif pencere fonksiyonu tasarımında başarılı bir yöntemdir ve aynı zamanda bu çalışma, PSO ile farklı pencere fonksiyonları tasarımı için örnek niteliğindedir.

Bu çalışma gelecek çalışmalar için yol gösterici niteliktedir. Gerçekleştirilen uygulamalar, PSO algoritmasıyla alternatif tasarımlar gerçekleştirilebileceğini göstermiştir. PSO'dan farklı olarak, hız ve çözüm üretme yeteneği daha gelişmiş optimizasyon teknikleri yada Hibrid algoritmalar kullanılarak tasarım yapılabilir. Aynı zamanda, LabVIEW'de gerçekleştirilen tasarımlar, FPGA ile de gerçek zamanlı olarak gerçekleştirilebilir. Tezde yer alan uygulamalar gelecek çalışmalara örnek olabilir ve geliştirilmeye açıktır.

KAYNAKLAR

- [1] **Winder, S.**, 2002, Analog and Digital Filter Design, Newnes.
- [2] **Shenoi, B. A.**, 2006, Introduction to Digital Signal Processing and Filter Design, John Wiley & Sons Inc.
- [3] **Schlichtharle D.**, 2000, Digital Filters Basics and Design, Springer, No:3-540-66841-1, Germany, 361p.
- [4] **Serbet, F., Kaya, T. and Ozdemir, M.T.**, 2017, Design of digital IIR filter using Particle Swarm Optimization, 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO).
- [5] **Yuce, A. and Tan, N.**, 2015, IIR filter design based on LabVIEW, 23rd Signal Processing and Communications Applications Conference (SIU).
- [6] **Sharma, D. and Kaur, R.**, 2015, Design and Analysis of IIR Notch Filter using LabVIEW, 2015 IEEE International Conference on Computational Intelligence & Communication Technology.
- [7] **Dhaliwal, K.K. and Dhillon, J.S.**, 2016, On the design and optimization of digital IIR filter using oppositional artificial bee colony algorithm, 2016 IEEE Students' Conference on Electrical, Electronics and Computer Science (SCEECS).
- [8] **Luo, Q., Zhang, S., Li, Z. and Zhou, Y.**, 2016, A Novel Complex-Valued Encoding Grey Wolf Optimization Algorithm. *Algorithms*, 9(1), 4.
- [9] **Pelusi, D., Mascella, R. and Tallini, L.**, 2018, A Fuzzy Gravitational Search Algorithm to Design Optimal IIR Filters. *Energies*, 11(4), 736.
- [10] **Zhang, X., Liu, S. and Wang, X.**, 2012, Design and Simulation of IIR Filter under Graphical User Interface, Proceedings of the 2012 International Conference on Computer Application and System Modeling.
- [11] **Chen, L., Liu, M., Yang, J., Wu, J. and Dai, Z.**, 2017, Structure Evolution Based Optimization Algorithm for Low Pass IIR Digital Filter Design, International Journal of Computational Intelligence Systems, 10(1), 1036-1055.

- [12] **Karmaker, T., Anower, M.S., Khan, M.A.G. and Habib, M.A.,** 2017, A new adjustable window function to design FIR filter and its application in noise reduction from contaminated ECG signal, Humanitarian Technology Conference (R10-HTC).
- [13] **Kaya, T. and Ince, M.C.,** 2009, The FIR filter design by using window parameters calculated with GA, ICSCCW 2009-Fifth International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control, 1-4.
- [14] **Vasundhara, Mandal D., Ghoshal, S. P. and Kar, R.,** 2013, Digital FIR Filter Design Using Hybrid Random Particle Swarm Optimization with Differential Evolution, International Journal of Computational Intelligence Systems, Vol. 6, 911-927.
- [15] **Osebik, D., Babič, R. and Horvat, B.,** 2001, Adaptive Digital Filter Implementation with Distributed Arithmetic Structure, MIDEEM 31(2001)3.
- [16] **Osebik, D., Kostanjevec, B., Jarc, B., Solar, M. and Babič, R.,** 1997, The FIR Digital Filter Realization with the Field Programmable Gate Array in Distributed Arithmetic Structure, MIDEEM 27(1997)3.
- [17] **Korošec, K., Vesenjsek, A., Jarc, B., Solar, M. and Babič, R.,** 1996, The FIR Digital Filter Realization with Standard Integrated Circuits in the Modified Distributed Arithmetic Structure, MIDEEM 26(1996)2.
- [18] **Babič, R. and Jarc, B.,** 1999, The Modified Distributed Arithmetic Structure for the Basic and the Cascade Digital Filters Realization, MIDEEM 29(1999)3.
- [19] **Vandenbussche, J.-J., Lee P. and Peuteman J.,** 2014, Round-Off Noise of Multiplicative FIR Filters Implemented on an FPGA Platform, *Appl. Sci.*, 4(2), 99-127.
- [20] **Dwivedi, A.K., Ghosh, S. and Londhe, N.D.,** 2016, Modified artificial bee colony optimisation based FIR filter design with experimental validation using field-programmable gate array, IET Signal Processing, Volume: 10, Issue: 8.
- [21] **Varshney, V. and Tiwari, M.,** 2017, Realization of an FIR filter using ATMEGA32 microcontroller, 2017 International Conference on Emerging Trends in Computing and Communication Technologies (ICETCCT).

- [22] **Li, C.**, 2015, A Genetic Search Algorithm for Linear-Phase Filter Banks, Proceedings of the 2015 International Conference on Intelligent Systems Research and Mechatronics Engineering.
- [23] **He, J. and Liu, Y.**, 2017, Research on Window Function Selection in Phase Estimation of Digital Signal, Proceedings of the 2017 5th International Conference on Frontiers of Manufacturing Science and Measuring Technology (FMSMT).
- [24] **Avci K. and Nacaroğlu A.**, 2008, A new window based on exponential function, IEEE Ph.D. Research in Microelectronics and Electronics (PRIME 2008). June. Istanbul, Turkey, 69-72.
- [25] **Avci, K.**, 2008, Design of High-Quality Low order Nonrecursive Digital Filters Using the Window Functions, PH. D. Thesis in University of Gaziantep, Gaziantep.
- [26] **Avci, K. and Nacaroğlu, A.**, 2009, Cosh window family and its application to FIR filter design, Int. J. Electron. Commun. (AEÜ) 63 (2009) 907–916.
- [27] **Kaya, T. and İnce, M.C.**, 2016, “The Using of ANN at Design of Window Function Having Useful Spectral Parameters”, Proc. of The Fifth Intl. Conf. On Advances in Computing, Electronics and Electrical Technology - CEET.
- [28] **Kaya, T. and İnce, M.C.**, 2009, The Calculation of Adjustable Window Parameters With Helping GA, Applied Automatic Systems (AAS’2009), Ohrid, Republic of Macedonia, 135-138 (2009) September 26-29.
- [29] **Gibbs, J.W.**, 1899, Fourier series, s. 200-606.
- [30] **Fejer, L.**, 1900, Sur les fonctions bornees et integrables, Comptes Rendus Hebdomadaires, Seances de l'Academie de Sciences, Paris, 131 984-987.
- [31] **Lanczos, J.C.**, 1956, Applied Analysis. Van Nostrand, Princeton, NJ.
- [32] **Adams, J.W.**, 1991, A new optimal window. IEEE Transactions on Signal Processing. 39 (8), 1753-1769.
- [33] **Dolph, C.L.**, 1946, A current distribution for broadside arrays which optimizes the relationship between beamwidth and side-lobe level, Proc. IRE, 34 335-348.

- [34] **Kaiser, J.F.**, 1974, Nonrecursive digital filter design using J_0 -sinh window function. Proc. IEEE Int. Symp. Circuits and Systems (ISCAS'74), San Francisco, Calif., USA, 20-23.
- [35] **Saramaki, T.**, 1989, A class of window functions with nearly minimum sidelobe energy for designing FIR filters. Proc. IEEE Int. Symp. Circuits and systems (ISCAS'89), Portland, Ore, USA, 359-362.
- [36] **Deczky, A.G.**, 2001, Unispherical Windows, IEEE Int. Symp. on Circuits and Systems. Sydney, Australia, May, 2 85-88.
- [37] **Bergen, S.W.A. and Antoniou A.**, 2002, Generation of Ultraspherical window functions, in XI European Signal Processing Conference, Toulouse, France, September, 2 607-610.
- [38] **Bergen, S.W.A. and Antoniou, A.**, 2004, Design of Ultraspherical Window Functions with Prescribed Spectral Characteristics, EURASIP Journal on Applied Signal Processing, 13 2053-2065.
- [39] **Bergen, S.W.A. and Antoniou, A.**, 2003, Nonrecursive Digital Filter Design Using the Ultraspherical Window, IEEE Pacific Rim Conference On Communications, Computers, And Signal Processing, 260-263.
- [40] **Bergen, S.W.A. and Antoniou, A.**, 2005, Design of Nonrecursive Digital Filters Using the Ultraspherical Window Function, EURASIP Journal on Applied Signal Processing, 12 1910-1922.
- [41] **Avci, K. and Nacaroğlu, A.**, 2008, Kaiser Yaklaşımı Kullanılarak Oluşturulan Üstel Pencereyle Yinelemesiz Sayısal Süzgeç Tasarımı, Çukurova Üniversitesi Mühendislik –Mimarlık Fakültesi 30. Yıl Sempozyumu, Adana, 274-279.
- [42] **Kaya, T. and İnce M.C.**, 2009, Yüksek Performanslı Pencere Fonksiyonlarının Genetik Algoritma Yardımıyla Gerçekleştirilmesi, 3. Haberleşme Teknolojileri ve Uygulamaları Sempozyumu, (HABTEKUS'09) 235-238.
- [43] **Mladenov, V.M. and Mastorakis, N.E.**, 2001, Design of Two-Dimensional Recursive Filters by Using Neural Networks, IEEE Transactions on Neural Networks, 12 (3) 585-560.
- [44] **Van de Vegte, J.** Fundamentals of digital signal processing. New Jersey: Prentice Hall.

- [45] **Jury, I.**, 1964, Theory and Application of the Z-Transform Method, New York: Wiley.
- [46] <http://www.ni.com/company/>
- [47] **Travis, J. and Wells, L.K.**, 2002, LabVIEW for Everyone, Prentice Hall.
- [48] **Kenny, J. and Eberhart, R.**, 1995, Particle swarm optimization. In Proceedings of IEEE international conference neural networks pp. 1942–1948, Vol. IV. Perth, Australia.
- [49] **Ozsaglam, M.Y. and Cunkas, M.**, 2008, Particle Swarm Optimization Algorithm for Solving Optimization Problems. Politelnik dergisi, Vol. 11, Number 4.
- [50] <http://www.meb.uni-bonn.de/epileptologie/science/physik/eegdata.html>.

ÖZGEÇMİŞ

Adı Soyadı: Fatmanur ŞERBET

Doğum Tarihi: 25 Ocak 1993

Doğum Yeri: Elazığ

Öğrenim Durumu:

Derece	Bölüm/ Üniversite	Yıl
Lisans	Elektrik-Elektronik Mühendisliği Bölümü, Fırat Üniversitesi	2012-2016
Yüksek Lisans	Elektrik Elektronik Mühendisliği Bölümü, Telekomünikasyon Anabilim Dalı, Fırat Üniversitesi	2016-2018

Yüksek Lisans Tez Başlığı ve Tez Danışmanı (Bitirme aşamasında):

Pencere Fonksiyonlarının LabVIEW Ortamında Gerçekleştirilmesi
Dr. Öğr. Üyesi Turgay KAYA

Akademik Çalışma Alanları:

- Sayısal Filtreler
- Sezgisel Optimizasyon Yöntemleri
- LabVIEW