



**TOBB EKONOMİ VE TEKNOLOJİ
ÜNİVERSİTESİ**

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ

Ele 515 - Ödev 2

Hande YILDIRIM

141201047

1. ADIM 1

Bu soruda sound01.wav şeklinde verilmiş olan ve telefonda dokuz adet tuşa basılması sesi çıkaran örnek bir ses dosyasının “audioread” komutu ile okunup örnekleme frekansı olan fs’nin elde edilip çizilmesi beklenmiştir. Bunu yapmak için öncelikle command windowa “doc audioread” yazarak bu komut hakkında gerekli olan bilgiler ve kullanım şekli öğrenildi. Daha sonra fs ve örnek sayısı bilindiği için sinyalin kaç saniye boyunca sürdüğü “örnek sayısı / fs” kullanılarak bulundu. Bulunan saniye, örnek sayısı kadar noktaya bölündükten sonra zaman eksenini ve frekans alanında çizdirmek için frekans eksenini oluşturuldu. Sonuç olarak aşağıdaki şekilde bir kod yazıldı:

```
%Adım1
[val,fs] = audioread("sound01.wav");%ses dosyası okunup örnekleme frekansı(fs) elde edildi
t = linspace(0,length(val)/fs,length(val));
plot(t,val);
xlabel('t(s)');
ylabel('Amplitude');
title("(THE AUDIO DATA) Sampling Frequency: " + fs + " Hz");
```

Bu kodun çıktısı olarak şu grafik elde edildi:

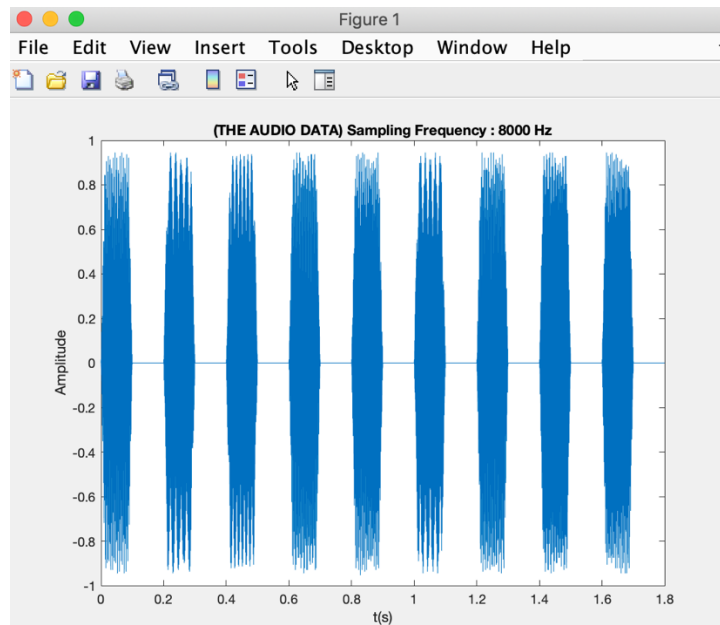


Figure 1

Grafikten sinyallerin her birinin 0.1 saniyede bir gönderildiği ve her bir sinyalin iki frekansta sinüsoidal sinyal içerdiği anlaşılr. Ayrıca grafikte yoğunlaşmış olan 9 tane mavi alandan da anlaşılabacağı gibi bu ses dosyasında 9 adet tuş sesi vurgulanmıştır. Bunların her birinden sonra 1'er saniyelik susma gerçekleştirilmiştir.

Bundan sonrasında ise, sinyali analiz edilerek her bir tuş sesine karşılık gelen parçaların tespit edilmesi ve her parçanın frekans cevaplarının “fft” komutu kullanılarak elde edilmesi istenmiştir. Elde edilen bu sinyal frekans bileşenleri “subplot” komutu kullanılarak hepsi tek bir figure içerisinde çizdirilmiştir. Daha sonra alt şekillerin her birinin üzerinden başlık olarak, ilgili alt sinyali oluşturan iki frekans bileşeni ve bu bileşenlere karşılık gelen tuş yani rakam yazılmıştır. Bunlar yapılırken aşağıdaki şekilde kod parçası yazılmıştır:

ADIM 2

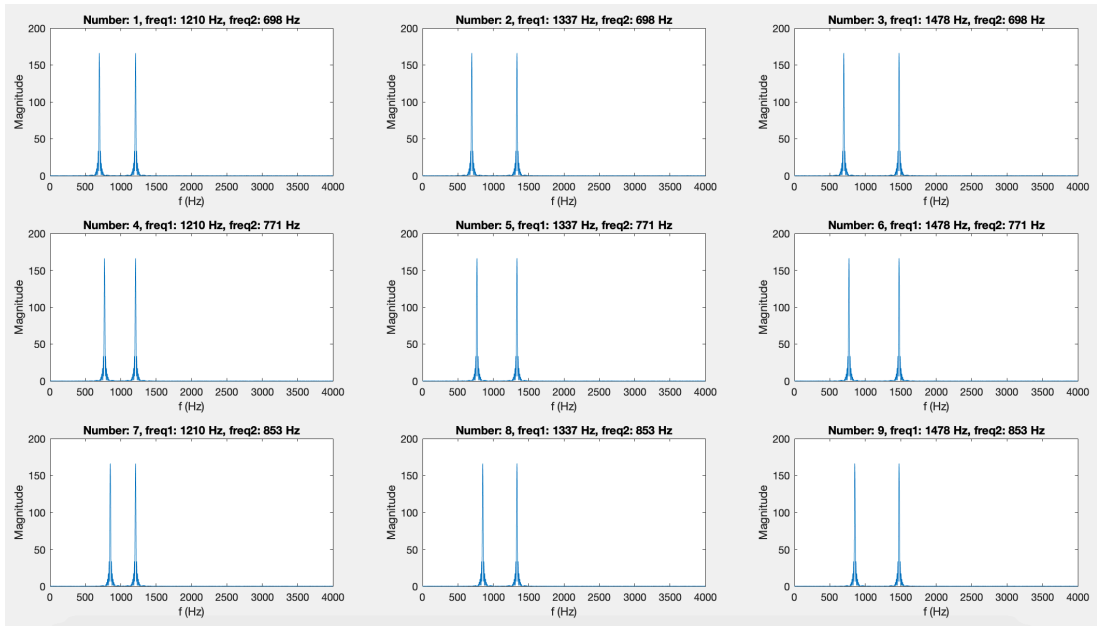


Figure 2

2. Bu soruda “fir1” komutu ile FIR (Finite Impulse Response) filtre tasarımı ve butter komutu ile IIR (Butterworth Infinite Impulse Response) filtre tasarımı ile ilgilenilmiştir.

ADIM 1: Bu kısımda özellikleri aşağıda belirtilen beş adet FIR filtresinin “fir1” komutu ile tasarlanması istenmiştir:

f_s (örnekleme frekansı)	1024 Hz
Filtre tipi	Bant geçiren filtre
$[f_{c1}, f_{c2}]$ (kesim frekansları)	[20,45] Hz
Filtre derecesi	Filtre1 için 60.derece Filtre2 için 90.derece Filtre3 için 120.derece Filtre4 için 150.derece Filtre5 için 180.derece

Figure 3

Bunun için aşağıdaki gibi bir kod parçası yazılmıştır. Burada f1, f2, f3, f4 ve f5 filtre isimlerini göstermektedir.

```
%Adım1
%a.
fs = 1024; %örnekleme frekansı
f = (0:1023);

f1 = fir1(60,[20 45]/(fs/2),'bandpass');
f2 = fir1(90,[20 45]/(fs/2),'bandpass');
f3 = fir1(120,[20 45]/(fs/2),'bandpass');
f4 = fir1(150,[20 45]/(fs/2),'bandpass');
f5 = fir1(180,[20 45]/(fs/2),'bandpass');
```

Figure 4

ADIM 2:

Burada öncelikle belli bir zaman anında verilecek şekilde bir dürtü (impulse) sinyali üretildi ve bu sinyal “subplot” komutu kullanılarak çizildi. Filtre katsayılarını fir1() metodu ile oluşturduk. Parametre olarak kesim frekanslarını girdik ve (fs/2) ye böldük. Girilen kesim parametresi 0 ile 1 arasında fs/2 ye oranlı bir kesim frekansına eşdeğer oluyor. fs/2 ye bölünmüş bir değer girdiğimizde kesim frekansını ayarlamış oluyoruz. u ve t vektörünü pulse gelen nokta t=0 noktasına denk gelecek şekilde oluşturduk. t 1024 elemanlı olacak şekilde 1 saniye boyunca sürüyor. Böylece fs=1024’ü temsil etmiş oldu. u(t) grafiğini çizdirdik ve u nun fourier dönüşümünü alarak frekans alanında çizdirdik. Oluşturduğumuz filtre sabitlerini filter fonksiyonuna u sinyali ile birlikte parametre olarak attık ve filtre çıkış sinyallerini elde ettik. Filtrelenmiş sinyal çıkışlarını çizdirdik ve fourier dönüşümlü hallerini çizdirdik. Fourier dönüşümlerini giriş sinyalinin fourier sinyaline bölerek kazanç grafiğini çizdik. Bu değerleri desibel cinsine çevirip logaritmik olarakta çizdirdik.

```
%Adım2
%b.
subplot(3,2,1)
u = [zeros(1,499) , 1 , zeros(1,500)]; %impulse oluşturuldu
t = 0:0.001:1-0.001;
plot(t,u,'r','linewidth',2)
grid minor
title('impulse input')
xlabel('t(s)')
ylabel('u(t)')
```

Figure 5

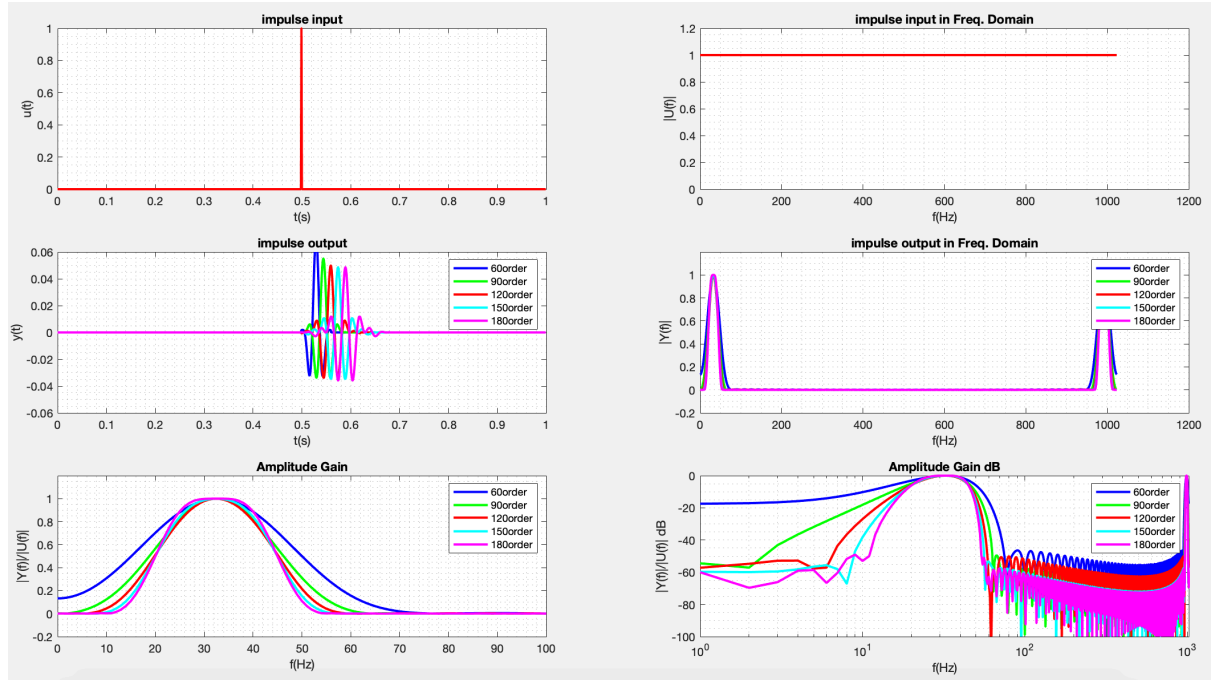


Figure 6

ADIM 3:

Bu bölümde ise özellikleri aşağıda belirtilen beş adet IIR filtreyi butter komutu ile tasarlandı:

f_s (örnekleme frekansı)	1024 Hz
Filtre tipi	Bant geçiren filtre
$[f_{c1}, f_{c2}]$ (kesim frekansları)	[20,45] Hz
Filtre derecesi	Filtre1 için 2.derece Filtre2 için 3. derece Filtre3 için 4. derece

Figure 7

ADIM 4:

Son olarak bu adım da belli bir zaman anında verilecek şekilde bir dürtü (impulse) sinyali üretildi. Oluşturulan subplot komutu ile çizdirildi. Bu dürtü sinyalinin frekans bileşenleri fft komutu ile elde edildi. Buna göre oluşturulan frekans bileşenleri subplot komutu ile çizdirildi. Daha sonra oluşturulan dürtü sinyalini, filter komutu ile 3. adımda oluşturulan filtrelere giriş olarak verildi ve çıkış sinyallerini subplot komutu ile çizdirildi. Ayrıca legend

ile hangi çıkışın, hangi filtreye ait olduğunu belirtildi. Her bir çıkış sinyalinin frekans bileşenleri, fft komutu ile elde edilip ve çizdirilmiştir. Subplot komutu ile ve legend ile frekans bileşenlerinin hangi filtrelere karşılık geldiği gözlemlenmiştir. Çıkış sinyallerinin frekans bileşenleri kullanılarak, filtrelerin frekans-kazanç grafikleri elde edildi. Burada frekans ve kazanç eksenleri lineer ve legend komutu ile hangi kazanç sinyalinin hangi filtreye karşılık geldiği belirtilmiştir.

Son olarak çıkış sinyallerinin frekans bileşenleri kullanılarak, filtrelerin frekans-kazanç grafiklerini elde edildi. Burada ise frekans eksenini logaritmik olarak çizildi ve kazanç eksenleri dB olarak belirtildi. Legend komutu ile hangi kazanç sinyalinin hangi filtreye karşılık geldiği gözlemlenmiştir.

Bu işlemlere uygun olarak elde edilen grafik aşağıdaki gibidir:

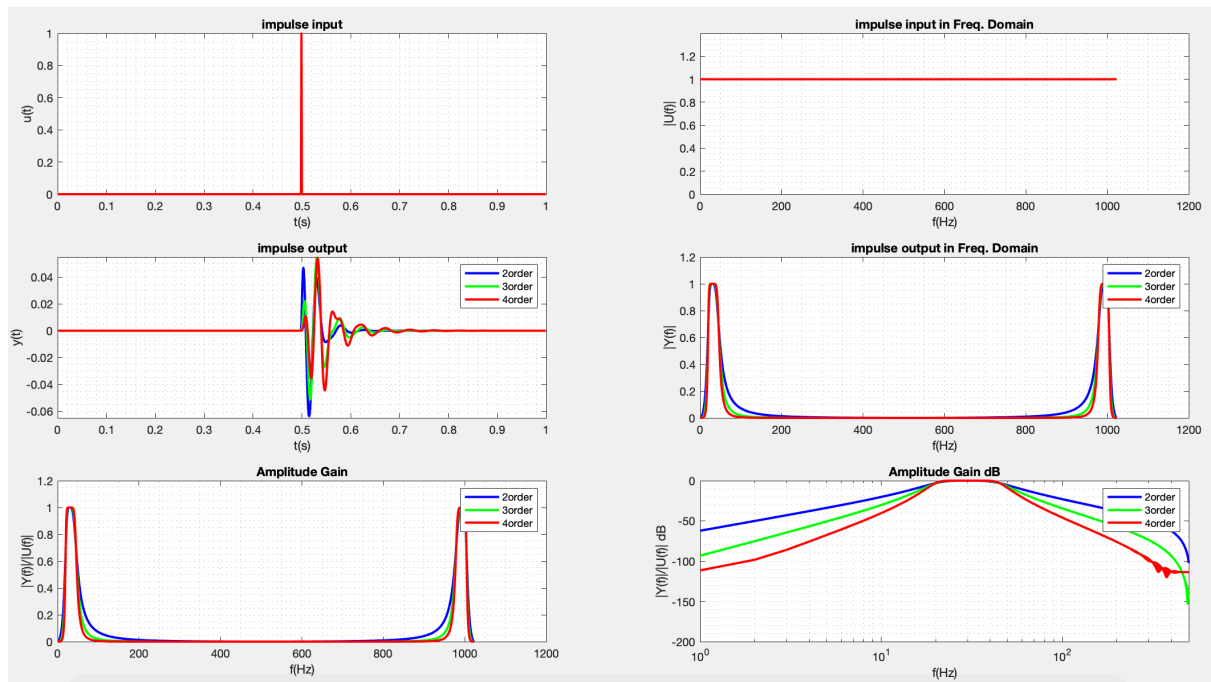


Figure 8

3. ADIM 1

$F_s = 100$ ile örneklenmiş, başlangıç frekansı 0 Hz olan ve $t = 2$ s anındaki frekansı 100 Hz olan bir chirp sinyali oluşturuldu. Sinyal 2000 örnekten oluşmaktadır: $(2s)(1000Hz)=2000$ örnek. Oluşturulan chirp sinyalini ise aşağıdaki gibidir:

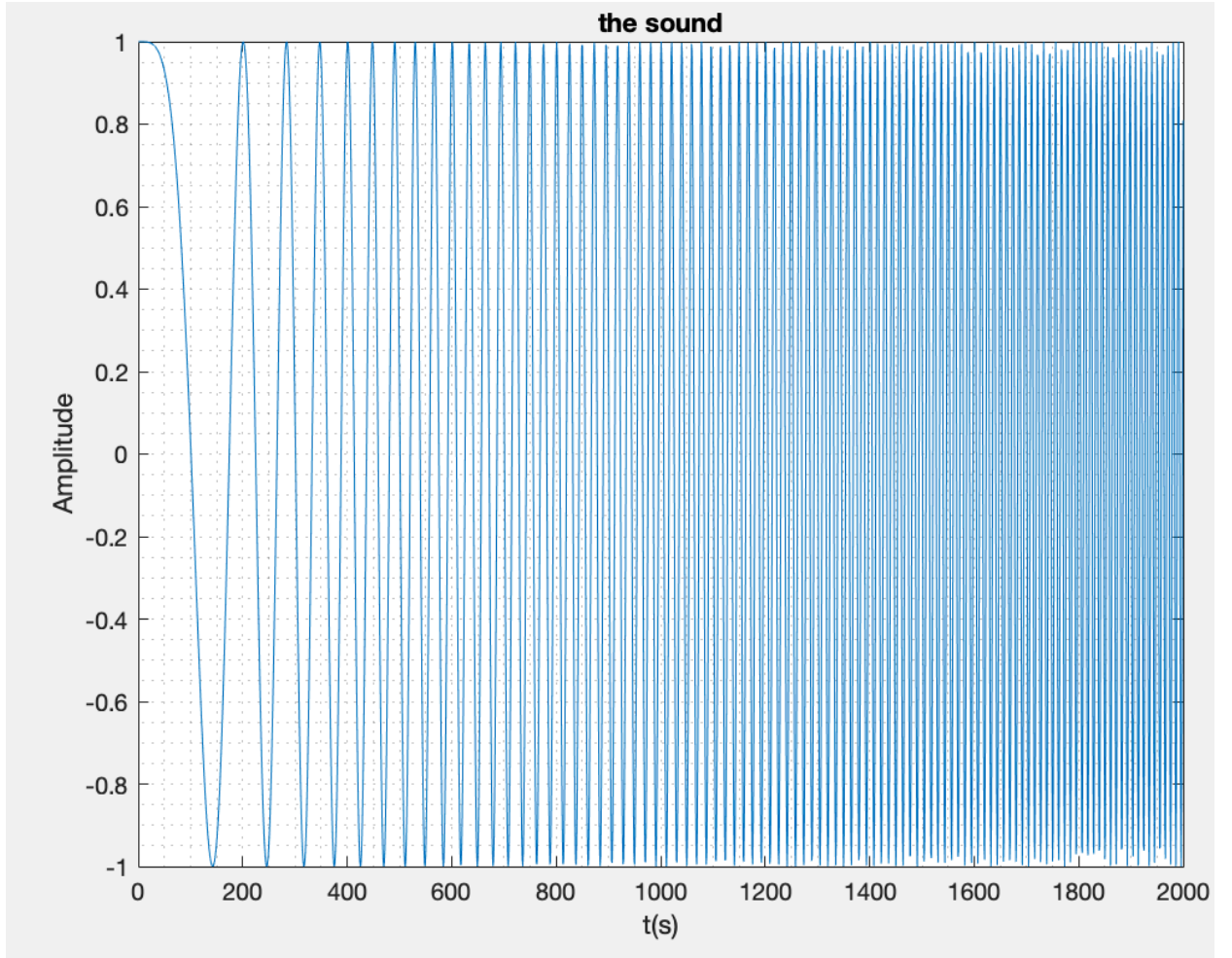


Figure 9

ADIM 2

Oluşturulan bu sinyalin, ilk 500 örneğini içeren bir sinyal oluşturularak frekans bileşenleri fft komutu ile elde edildi. Ardından bu pencere (bu 500 örnekli sinyallere, bu noktadan sonra, pencere (window) denildi) 25 örnek ilerletilerek tekrar bu 500 örnek uzunluğundaki pencerenin frekans bileşenlerini elde etmek için fft komutunu kullanıldı. Bu işlem sinyalin sonuna kadar gerçekleştirildi. Elde edilen frekans bileşenlerine, karşılık gelen pencere (yani zaman aralığı) değerleri ile birlikte kaydedildi. Bu işlem:

$$\frac{(\text{sinyal uzunluğu} - \text{pencere uzunluğu})}{\text{adım uzunluğu}} + 1 = \frac{(2000 - 500)}{25} + 1$$

defa yapıldı. Son

olarak chirp sinyalinin zaman-frekans grafiğini, contourf, surf, ya da imagesc komutlarını

kullanılarak çizdirildi. Elde edilen grafik aşağıdaki gibidir. Bu grafik oluşturulurken `contourf` komutu kullanılmıştır.

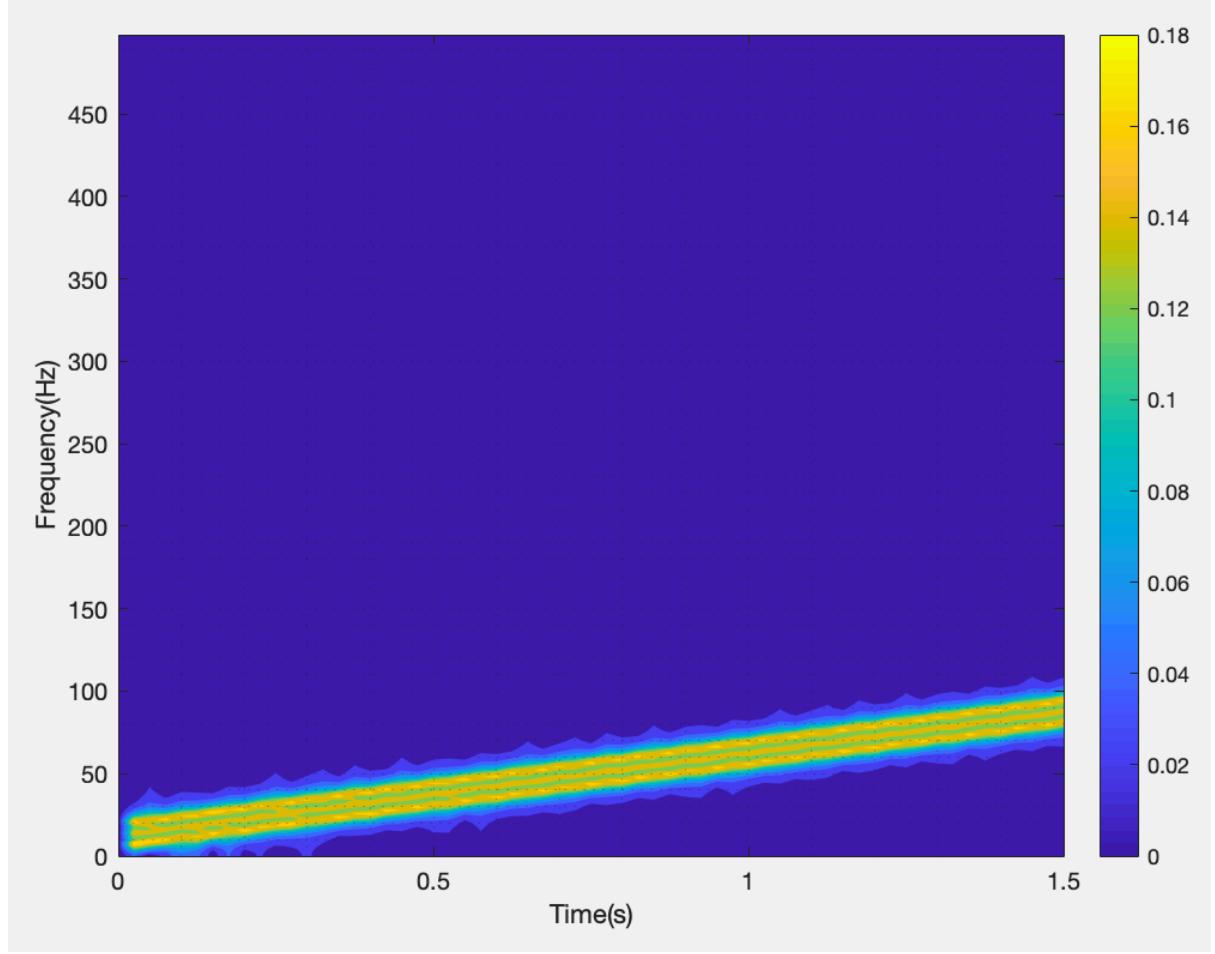


Figure 10

Grafiğe göre 1.5 saniyede baskın frekansın 100Hz'in çok az altında olduğunu gözlemlenmektedir. 2. Saniyenin sonunda ise 100Hz'e ulaşaca beklenmektedir. Bu nedenle elde edilen grafik beklenen şekildedir.

4. ADIM 1

Bu soru, bir doğrusal olmayan (nonlinear) bir sistemin durumlarının (state) zaman ve frekans karakteristiklerinin analizi ile ilgilidir. Soruda yapılacak adımlar aşağıda sıralanmıştır.

İlk olarak dinamik denklemleri verilen 3 boyutlu doğrusal olmayan sistemin MATLAB altında `ode45` komutu ile simülasyonunu gerçekleştirildi.

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= x(\rho - z) - y \\ \dot{z} &= xy - \beta z\end{aligned}$$

Burada başlangıç koşulları sırasıyla 0, 1 ve 20 alındı. Sistem parametreleri ise:

$$\sigma = 10, \rho = 28, \beta = \frac{8}{3}$$

şeklinde dirler. Bunun için yazılan kod parçası aşağıdaki şekildedir:

```
%bilinmeyenler yazildi
simulas = 100;%simülasyon suresi
fs = 1000;%ornekleme freq
sig = 10;
ro = 28;
b = 8/3;
% baslangic koordinatlari
x0 = 0;
y0 = 1;
z0 = 20;

% bu asamada ilk olarak ode45 kullanarak denklem cozdurulur ve hatalar
% ayarlanir. Sonrasinda 1000 Hz icin zaman ayarlanip 1/1000'er 1/1000'er
% artis yapilir.

figure(5)
f = @(t, x) [sig*(x(2)-x(1)); x(1)*(ro-x(3))-x(2); x(1)*x(2)-b*x(3)];
sec = odeset('RelTol', 1e-12, 'AbsTol', 1e-12);
[t, xt] = ode45(f, 0:1/fs:simulas-1/fs, [x0 y0 z0], sec);
```

Figure 11

ADIM 2

Bu kısımda ise simülasyon sonucunda elde edilen sistem durumları aşağıda gösterildiği gibi üç boyutlu olarak aşağıdaki gibi çizildi ve burada $t = 0$ anında sistemin başladığı noktayı * ile gösterilmiştir.

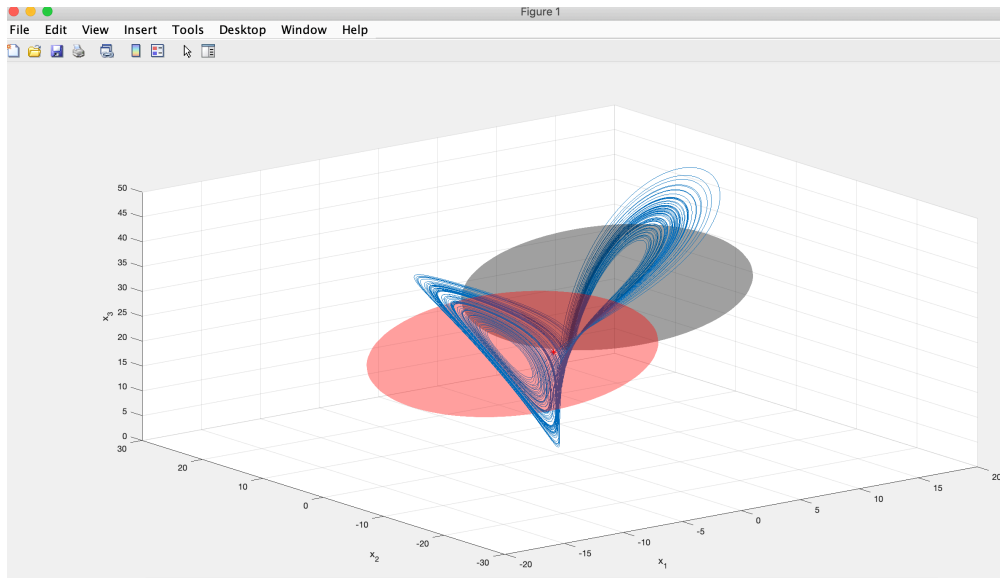


Figure 12

ADIM 3

Simülasyon sonunda, sistem durumlarının her birinin frekans bileşenleri MATABL `fft` komutu ile belirlenmiştir Sistem durumlarının her biri önce zaman ekseninde, sonra da frekans ekseninde subplot komutları kullanılarak ve eksenler hep lineer yapılarak, yani logaritmik eksen ile dB kullanmadan çizildi.

Burada sistem 100 saniye süresince simüle edilmiş ve örnekleme periyodu olarak $TS = 1\text{ ms}$ seçilmiştir. Dolayısı ile sistemde mevcut olabilecek maksimum frekans, örnekleme frekansının yarısı olup, 500 Hz'tir. Ancak verilen grafikte, alçak frekans bileşenlerinin daha yakından incelenmesi amacı ile sadece 0-20 Hz aralığı çizdirilmiştir.

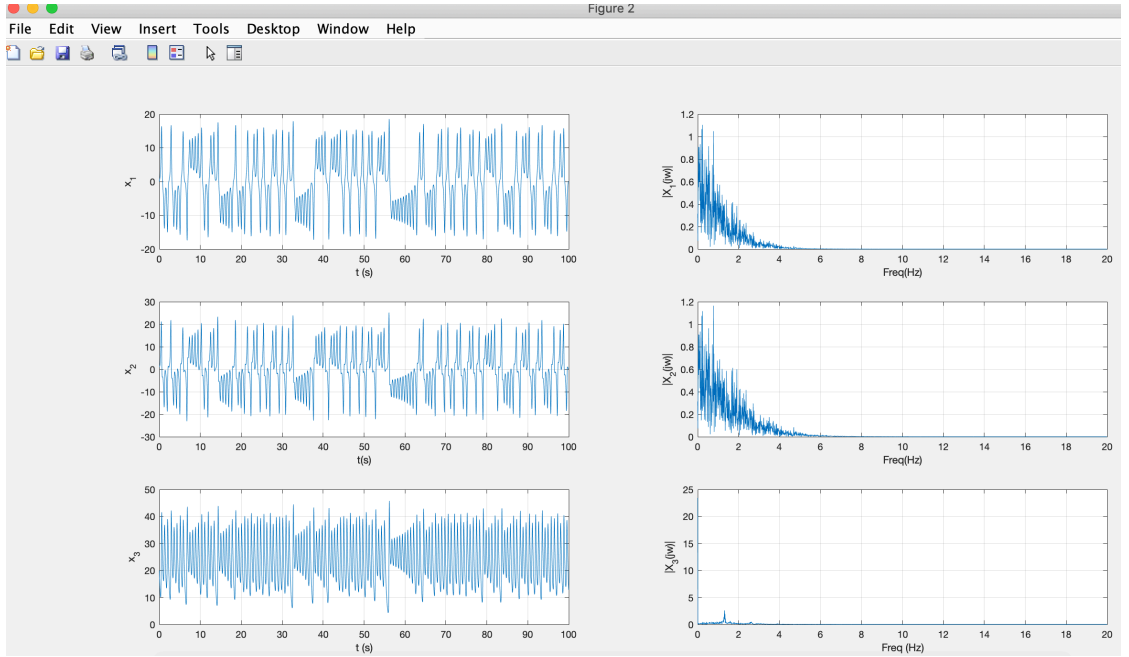


Figure 13

Aşağıdaki şekil incelendiğinde dönen yörüngeler göze çarpmaktadır. Bunlar kabaca iki merkez etrafında dönmekte, zaman zaman bir merkezin yörüngesinden diğerine geçmektedirler.

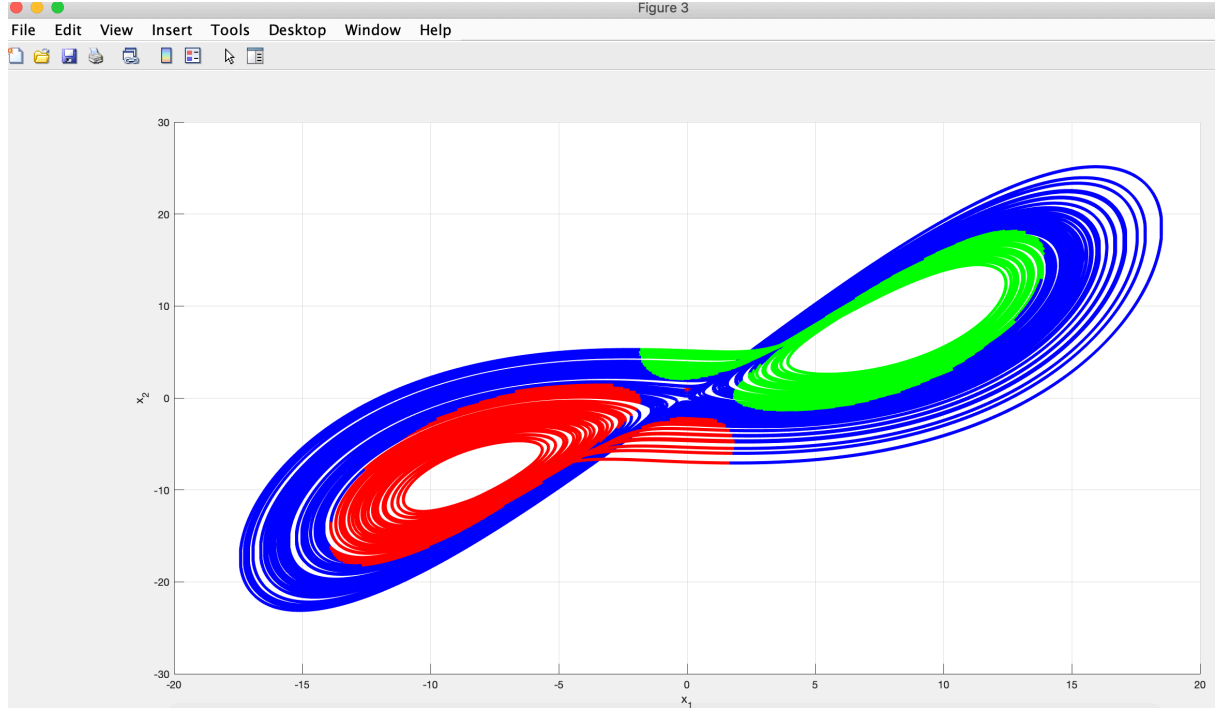


Figure 14

Daha sonrasında bu iki merkez yaklaşık olarak bulunmuştur. Burada farklı yarıçap değerleri denendi fakat ödevde istenilene en benzer olarak 11 alındı. Daha sonra ise x , y ve z durumlarının zamana karşı değişimleri aşağıdaki gibi çizdirildi.

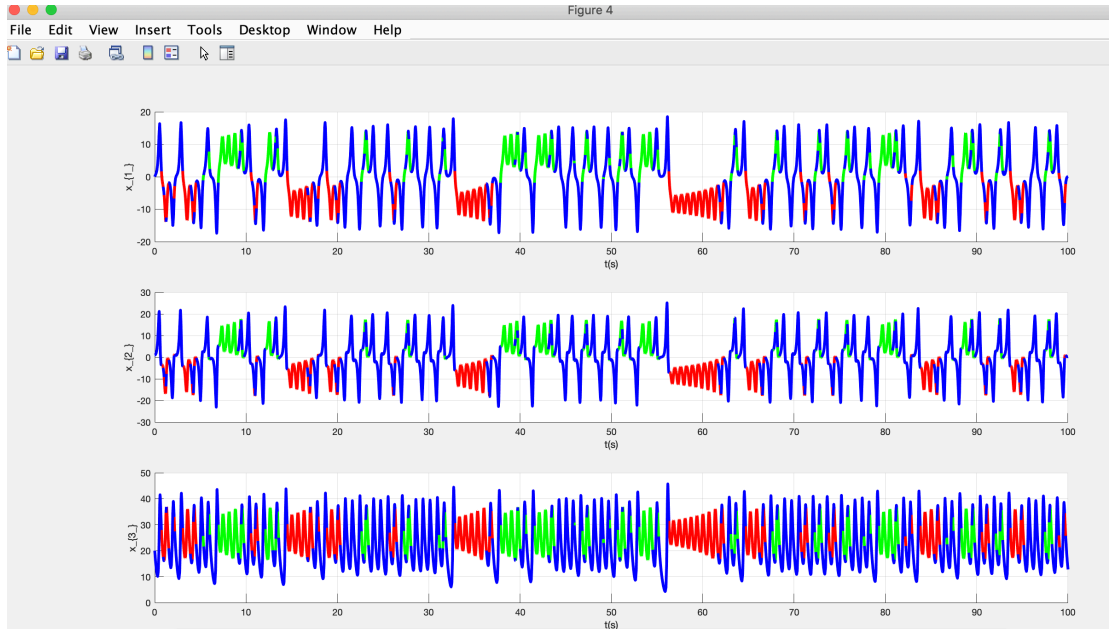


Figure 15

Buradaki grafiklere bakılarak x , y ve z deęişkenlerinin kaotik bir şekilde deęişip dengeye oturmadıkları gözlemlenir. Bu denklemler yandalımdan da bildiğim kadarıyla “Lorenz Denklemleri”dir.

ADIM 4

Lorenz denklemleri: Bu denklemler isimlerini Norton Lorenz'den almışlardır. Bu denklem aslında Lorenz'in 1963 yılında yayınladığı "Deterministic Nonperiodic Flow" makalesinde, atmosferdeki ısı aktarımı yapısının sadeleştirilmiş bir matematiksel modelidir. Bu modelin üç durum x, y, z ve üç parametresi σ, ρ, β vardır. Bu model lineer olmayan, bayağı bir diferansiyel denklem sistemini kastediyor. Ayrıca deterministiklik, yani bir belirsizlik ya da rastlantısallık bulunmamaktadır.