به نام خدا

جزوه آموزشی درس فیلترهای دیجیتال

استاد درس: دکتر زمان ملک زاده کبریا

گروه زمین شناسی دانشگاه پیام نور

گروه ژئوفیزیک (زلزله شناسی) دانشگاه پیام نور

مقدمه

پردازش سیگنال رقومی DSP(Digital Signal Processing) از حوزه مهم مطالعاتی محسوب می شود که دستاورد پیشرفت های اتفاق افتاده در رشته های مهندسی ارتباطات و فناوری رایانه ها است. این تکنیک در آسان کردن کارها کمک زیادی به بشر می نماید. بعنوان مثال در عرصه موسیقی و برای بارگذاری یک صدا و با در نظر گرفتن سرعت بارگذاری در حد 56 کیلو بایت در ثانیه، ساعت ها وقت نیاز است. نرم افزار فشرده ساز MP3 به شما این امکان را می دهد تا با کاهش اندازه فایل به 90٪ در عرض چند دقیقه بارگذاری انجام شود. این عمل با دانستن 1- اینکه یک سیگنال چیست؟ 2- چگونه رقمی می شود 3- چگونه می توان بخشی از این سیگنال را حذف نمود تا آسیب جدی به سیگنال اصلی وارد نیاید. این سه مرحله بخشی از پردازش رقمی سیگنال یا DSP است.

ارقام

فلسفه ایجاد رقم حرف A نشان دهنده یک "   " نسبت (مقایسه دو چیز غیر همسان) > اعداد کسری و ممیز DECIMAL. چرا در مبنای 10؟ احتمالا چون 10 انگشت داریم؟!! چرا کامپیوتر از اعداد در مبنای دو استفاده می کند؟ چون از ترانزیستوراستفاده می کند: قطع (0) و وصل جریان (1) چرا گاهی برنامه نویسان از مبنای 16 استفاده میکنند؟ اعداد 0011001000010101 و 3215 برابرند. اولی در مبنای 2 و دومی در مبنای 16. کدام آسانتر است؟ در واقع هر بیت 16 مبنایی 4 بیت در سیستم دوبیتی است. برای تبدیل عدد در مبنای 10 به عدد در مبنای 2 (مثلا 4.375): بطور متوالی 4 را بر 2 تقسیم می کنیم حاصل و باقیمانده را بدست آورده در نهایت عدد در مبنای 2 بطور معکوس نوشته می شود:

2=2/4 و صفر

1=2/2 و صفر

0= 2/1 و 1

بنابر این حاصل 100 است برای عدد کسری مقدار را بترتیب با حذف عدد سمت چپ در 2 ضرب می کنیم و حاصل را بطور مستقیم مینویسیم:

0.375*2=0 و 0.75

0.75*2= 1 و 0.5

0.5*2=1 و 0

بنابر این حاصل تبدیل بخش کسری: 011 است و حاصل نهایی 100.011 است. فلسفه بوجود آمدن این اعداد را بسادگی در جواب معادله درجه 2 وقتی که عدد زیررادیکال منفی شود می توان تجسم نمود:

$$x_1, x_2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

اعداد موهومی

به این منظور جواب چنین معادله را میتوان به دو بخش حقیقی و موهومی تقسیم نمود مشروط به اینکه فاکتوری بنام

$$j = \sqrt{-1}$$

را معرفی نماییم. در واقع میتوان تمامی اعداد را اعداد موهومی تلقی نمود که دارای دو بعد هستند : بخش حقیقی و بخش موهومی و در واقع یک عدد حقیقی عدد موهومی است که بخش موهومی آن صفر است. عدد موهومی را می توان در مختصات قطبی نشان داد:

$$x = a + bj$$

حقیقی در محور x و موهومی در محور y. بنابر این عدد دوبعدی نقطه ای در فضا و بر روی دایره ای است که شعاع (مقدار یا بزرگی) آن اندازه مجموع مربعات x و y و موقعیت آنرا زاویه نسبت به محور x تعیین میکند این زاویه :

$$\theta = \arctan(y / x)$$

اما وقتی موقعیت نقطه بر روی دایره در ربع دوم و سوم قرار گیرد، (یعنی a منفی باشد) شکل 1 نشان می دهد که مقادیر بزرگا و فاز مشابه وقتی است که نقطه در ربع اول و چهارم قرار گیرد. به این منظور با توجه به علامت b به زاویه مقدار منفی و مثبت (بترتیب) π را اضافه می کنیم.

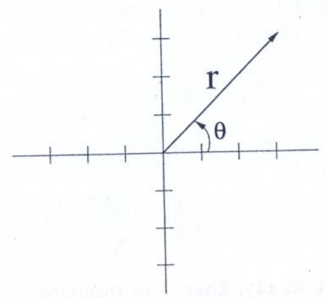


Figure 1.1: An example vector.

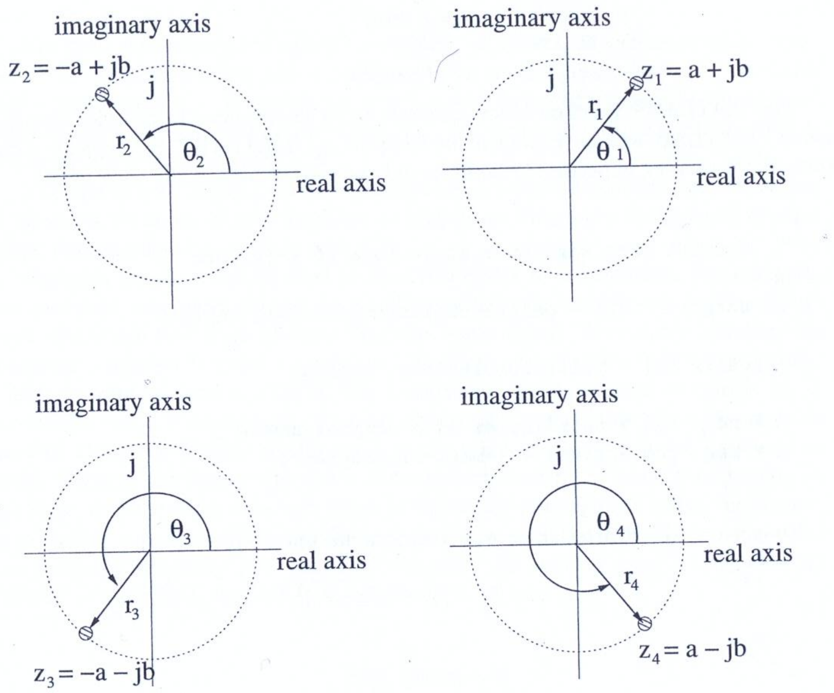


Figure 1.2: Calculating $\theta = \arctan(b/a)$ leads to a problem when a is negative.

سیگنال

هر پدیده متغیر که قابل اندازه گیری باشد سیگنال محسوب می شود. این متغیر می تواند یک پارامتر فیزیکی باشد که مقدار یا بزرگای آن نسبت به زمان و یا مکان تغییر نماید. بعنوان مثال صدا (فشار اکوستیک)، ولتاژ (اختلاف ولتاژ تولیدی توسط یک میکروفون)، موج عبوری از یک فضای مادی، رادار، تصویری که توسط یک دوربین ویدئویی، دما منتقل می شود. اغلب مجبوریم یک سیگنال را در دوره های متفاوتی از زمان تجربه کنیم: درجه حرارت شهر و انتخاب لباس مناسب برای لحظه بخصوصی که در فضای بیرون هستیم. سیگنال ها ممکن است با خود خطاهایی را شامل شوند مانند اندازه گیری دما وقتی که در لحظه خاص باد سرد می وزد. بعضی از سیگنالها بطور دائم قابل اندازه گیری هستند نظیر دما توسط دماسنج ولی در زمان های مختلفی اندازه گیری می شوند لذا در حد فاصل بین دو اندازه گیری ها مقدار دما مشخص نیست. و اگر در همان لحظه بادی بوزد و دما را پایین آورد اطلاعاتی وجود نخواهد داشت. اندازه گیری می تواند پیوسته و یا ناپیوسته discrete باشد. دارای ایندکس و مقدار باشد. مثلا درجه حرارت (مقدار) در زمان مشخص (ایندکس). ممکن است ایندکس پیوسته intiger و مقدار ناپیوسته باشد (تعداد آدم هایی که در یک ساختمان در زمان های مختلف باشند). اما در کامپیوتر با وجود پیوسته بودن سیگنال $x(t)$ آنرا ناپیوسته $x[n]$ اندازه گیری می کنند. در اینجا هدفمان بررسی سیگنال با اندیس پیوسته و مقدار پیوسته continuous/continuous (آنالوگ) و سیگنال با اندیس ناپیوسته و مقدار ناپیوسته discrete/discrete (دیجیتال) است. در حوزه دیجیتال یک سیگنال چیزی بیشتر از یک آرایه (بردار یا ماتریس یک بعدی) نیست. گرچه سیگنالهای دوبعدی نظیر تصویر نیز وجود دارند. در نظر گرفتن آرایه برای سیگنال کار ما را در تحلیل خطی آن آسان می کند.

آنالوگ در برابر رقومی (دیجیتال)

دو نوع سیگنال وجود دارد: آنالوگ و دیجیتال. آنالوگ آنطور که از نامش پیداست یعنی مانسته یا دنیای واقعی بعنوان مثال یک صدا که از طریق یک میکروفون در یک نوار مغناطیسی ضبط می شود. یعنی کپی یک سیگنال x(t) . سیگنال دیجیتال سیگنالی است که از طریق sampling یا نمونه برداری ایجاد می شود و بنابراین نماینده سیگنال اصلی است نه مانسته آن. x[n] که n می تواند کسری یا غیر کسری باشد.

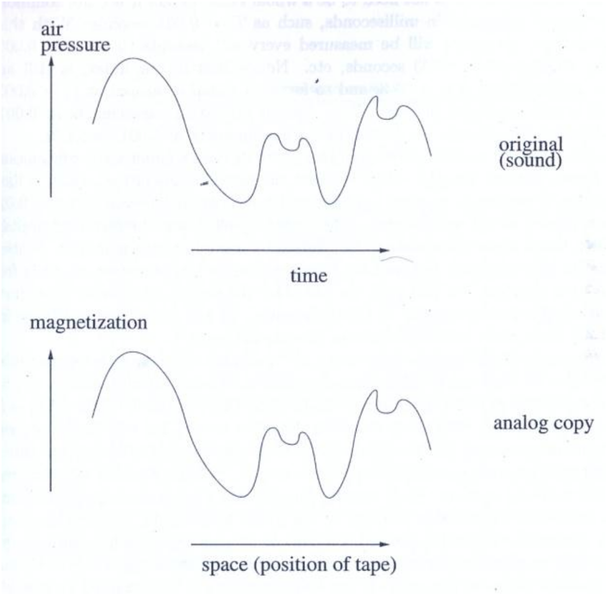


Figure 1.3: A sound signal with a tape analog.

متغیر n به t با رابطه t=nTs وابسته است. که Ts دوره نمونه گیری است. مثلا اندازه گیری درجه حرارت با Ts=1 ساعت، یعنی در هر ساعت یک نمونه یا اندازه گیری. 0–1–2–3–... ساعت. نمونه برداری می تواند در کسری از یک ثانیه باشد. یعنی Ts=0.01sec یعنی ...-0-0.01-0.02-0.03sec .

بعنوان مثال شکل 4 نشاندهنده یک سیگنال دیجیت شده با نمونه برداری 0.01 ثانیه از یک مانسته است در واقع سیگنال بدست آمده مانند ضرب یک سری ضربه impulse با فاصله زمانی 0.02 ثانیه است.

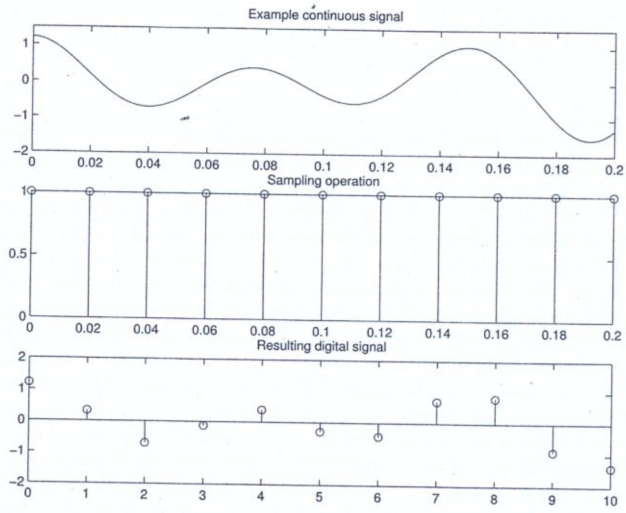


Figure 1.4: Sampling a continuous signal.

بعنوان مثال فرض کنید یک سیگنال دارای مقدار x[1]=2 و x[2]=4 باشد آیا می توان گفت که x[3]=1.5 ؟ جواب منفی است برای اینکه در این بازه اطلاعاتی وجود ندارد گرچه این محتمل ترین حدس است. سیگنال های دیجیتال در اندازه یا بزرگا نیز رقمی یا پله ای quantize است. وقتی یک سیگنال نمونه برداری می شود، مقادیر در حافظه ذخیره می شود. هر حافظه مقدار محدودی دقت دارد اگر عدد مربوطه خیلی بزرگ باشد، یا خیلی کوچک باشد

برای جایگیری در محل اختصاص یافته حافظه بریده می شود. بعنوان مثال یک پمپ بنزین را در نظر بگیرید، ممکن است این پمپ برای نشاندادن قیمت از یک عدد 5 رقمی استفاده می کند، 3 رقم برای دلار و دو رقم برای سنت. اگر یک باک بزرگ داشته باشید و که قیمت 999.99$ را نشان دهد با حتی یک مقدار کمی دیگر از بنزین (چیزی در حدود 2 سنت) عددی که بعنوان قیمت نشان می دهد مثلا 0.01$ است. شکل 6 تبدیل سیگنال آنالوگ به دیجیتال را نشان می دهد. عکس این حالت نیز وجود دارد. پردازش سیگنال آنالوگ قسمت های الکترونیکی دارد نظیر ترانزیستور، خازن و تقویت کننده. این موجب ارزان شدن پردازش می شود ولی در DSP نیاز به اضافه کننده، ضرب کننده و تاخیر دهنده (رجیستور) دارد. این روش منعطف و خطا و حل کردن آن به سرعت انجام می شود.

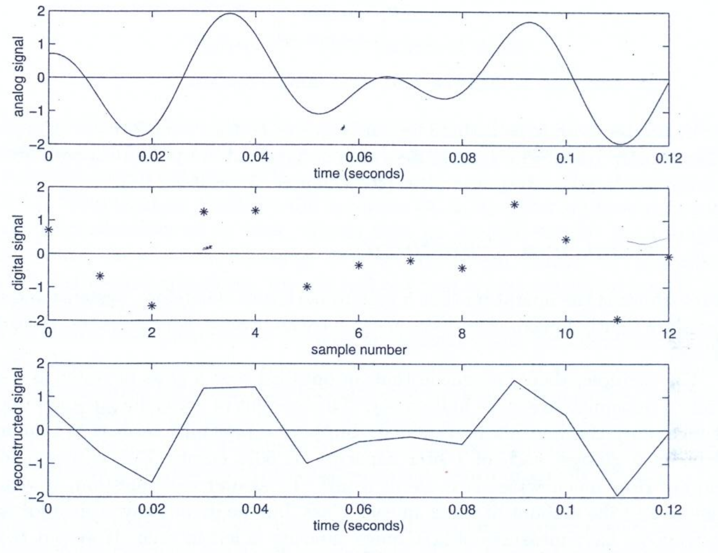


Figure 1.5: Three ways of viewing a signal.

سیستم چیست

سیستم چیزی است که بر روی یک سیگنال اثر (تغییر) بگذارد. سیستم می تواند یک سخت افزار باشد یا نرم افزار. شکل 6 یک سیستم را نشان می دهد که بعنوان یک جعبه سیاه خروجی y[n] را از یک ورودی x[n] می سازد. مثال ساده می تواند یک افزاینده واحد باشد که یک ورودی مانند x[n]={1,2,3,4} را به y[n]=[2,3,4,5] تبدیل می کند. مثالهای یک سیستم شامل تبدیل فوریه و یا معکوس فوریه (FFT ,Ifft) و فیلتر نظیر فیلتر پاسخ ضربه محدود (FIR) و فیلتر پاسخ ضربه نامحدود (IIR) است.

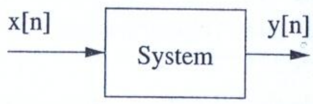


Figure 1.6: An example system.

تبدیل transform

تبدیل عملی است که سیستم انجام می دهد. بنابراین سیستم و تبدیل بهم نزدیک هستند. تبدیل می تواند معکوس باشد و یک سیگنال را به وضعیت قبل از تغییر برگرداند. بعنوان مثال سه ظرف را در نظر بگیرید که بترتیب 50، 75 و 65 درصد خالی باشند. یک بدبین فقط نیمه خالی و یک خوشبین نیمه پر را می بیند. یعنی اگر از مقدار آب موجود در ظرف از آنها سوال شود پاسخ متفاوت خواهید شنید اولی مقدار خالی و دومی مقدار پر. اما مهم اینجا است که هر تفسیر برای یک مورد خاص بدرد می خورد یعنی اگر سوال این باشد چقدر آب نیاز است تا ظرف ها پر شوند تفسیر بدبین و اگر سوال این باشد که مقدار آب در ظرف چقدر است پاسخ خوشبین مناسب تر است. بنابر این واکنش این افراد را می توان یک سیستم تلقی نمود که یکی افزاینده incrementer

(خوشبین) و دیگری کاهنده decrementer است. روش نخست روش forward transform (و روش دومی inverse transform است در روش اول سیستم y[n]=x[n]+1 و در دومی y[n]=x[n]-1

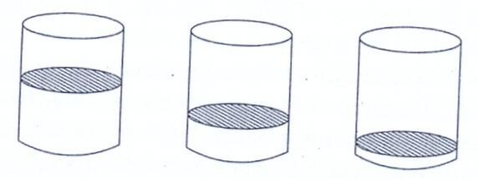


Figure 1.7: Three glasses of water.

بعضی وقت ها عمل تبدیل را به این خاطر انجام می دهیم که اجرای آن آسانتر است. بعنوان مثال اگر قرار است تعداد سالهای بین 1977 و 2002 را در قالب اعداد یونانی (بترتیب LXXVII و MMII) کم کنیم روش متداول تفریق مشکل است چون تفریق عدد 7 به این زبان مشکل دارد لذا بهتر است آنرا نخست به اعداد لاتین تبدیل و تفریق را انجام و سپس حاصل تفریق را به زبان یونانی بنویسیم (یعنی XXV بجای 25) . یا مثلا حل معادله دیفرانسیل به کمک تبدیل لاپلاس. یک معادله را به کمک جدول مخصوص تبدیل و بعد از انجام عملیات جبری بر روی آن نتیجه نهایی باز به کمک یک جدول به حوزه زمان بر می گردد. از تبدیل لاپلاس بعنوان روشی برای بررسی سیستم نیز استفاده می شود که در مباحث بعدی به آن اشاره خواهد شد.

چرا سینوسی ها مورد مطالعه قرار می گیرند؟ توابع سینوسی (سینوس یا کسینوس) جالبند چون اغلب در دنیای مانسته ظاهر می شوند. چون توابع سینوسی تکرار شونده هستند، کافی است تکه ای از اطلاعات مورد نیاز در دسترس بوده تا بقیه نیز حاصل شود. توابع سینوسی در این مباحث بصورت زیر نوشته می شوند:

Magnitude*cos(2π*frequency*t+phase)

بنابر این کافی است فرکانس، بزرگا و فاز را داشته باشیم تا تابع در هر زمان دلخواه بدست آید. این روش بسیار فشرده ای برای نشان دادن یک سیگنال است.

سوال

- سیگنال چیست؟
- سیستم چیست؟
- X[n] به چه چیزی دلالت دارد؟
- X(t) به چه چیزی دلالت دارد؟
- ده دهی 0101.01 چیست؟
- مبنای 2 عدد 36.1 چیست؟
- مختصات قطبی نقطه (2,5) چیست؟
- عدد در مختصات دو قطبی 4, 25 degree را به کارتزین تبدیل کنید.
- کوانتیزه کردن در زمان و کوانتیزه کردن مقدار را تشریح کنید.
- چرا سینوسی در پردازش سیگنال مهم است؟

- چرا استفاده از $\theta=\arctan(b/a)$ برای تبدیل قطبی اشکال دارد؟
- فرق آنالوگ و دیجیتال چیست؟

فیلترها

این تمایل اغلب وجود دارد تا بر روی سیگنال دستکاری شود بعنوان مثال بر روی دستگاه، بیس (bass) را افزایش دهیم تا صدا بمتر (فرکانس کمتر) شود. یا تغییر حدود فرکانسی با اکولایزر. این عملیات الزاما پردازش سیگنال دیجیتال نیست چون بر روی سیگنال آنالوگ نیز قابل اعمال است. در واقع فیلترها سیگنال ها را تغییر می دهند.

- عملکرد یک فیلتر چیست؟
- چگونه می توان یک فیلتر ساخت؟
- در این فصل به این موارد اشاره خواهد شد.

شکل 1 یک سیگنال را نشان میدهد که نماینده تغییرات درجه حرارت است (در اینجا بعنوان یک سیگنال ورودی). در شکل وسط خروجی سیگنال بعد اعمال فیلتر پایین گذر نشان داده شده است. یعنی عملکرد فیلتری که اطلاعات فرکانس کم را نگه می دارد. ضمنا توجه کنید: این سیگنال فیلتر شده چقدر به اولی مانند است؟ داده بدست آمده یکی بیشتر از داده اولیه است. به این خاطر که معمولا دادهای خروجی مساوی تعداد داده ورودی بعلاوه داده تعداد ضرایب فیلتر منهای 1.

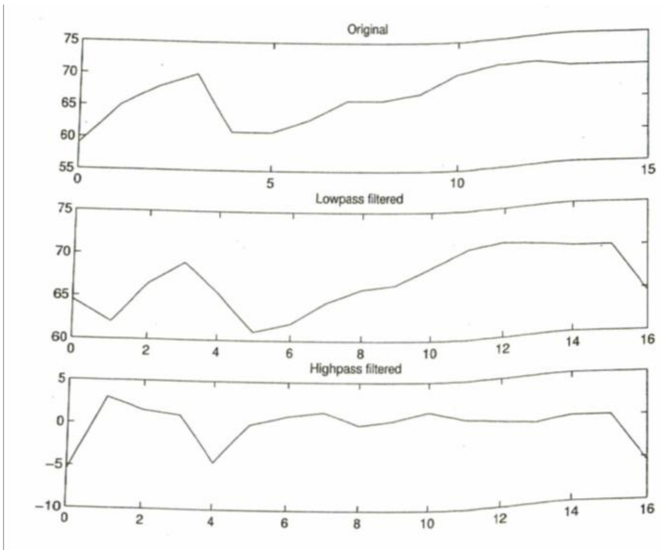
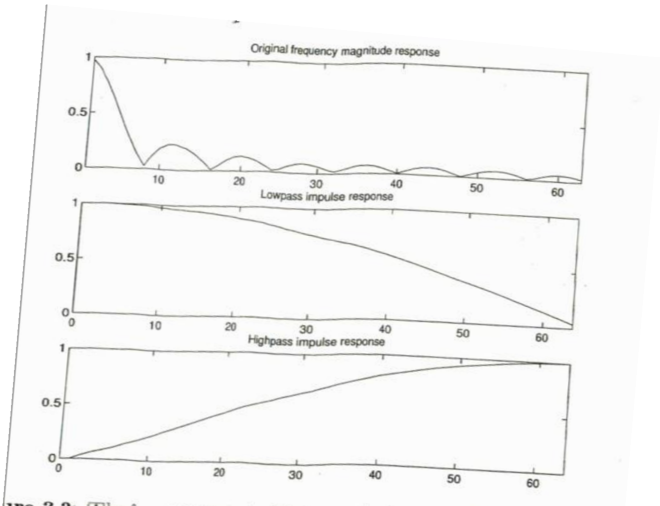


Figure 3.1: An example signal, filtered.

سیگنال پایینی سیگنال فیلتر شده از نوع بالاگذر است. در این فیلترینگ اجازه داده می شود بخش هایی که در خود تغییر سریعی دارند بگذرند. مشاهده می شود که اغلب نقاط حول صفر هستند که نشاندهنده تغییرات کم هستند. همچنین در شکل 2 شاهد اعمال فیلترهای پایین گذر بر روی یک سیگنال اولیه (شکل بالایی) هستیم که حاصل آن عبور داده های فرکانس کم است (شکل وسط). همینطور بعد از اعمال فیلتر بالا گذر داده های فرکانس بالا عبور می کنند. هر دو دو ضریب 0.5 و -0.5 دارند.

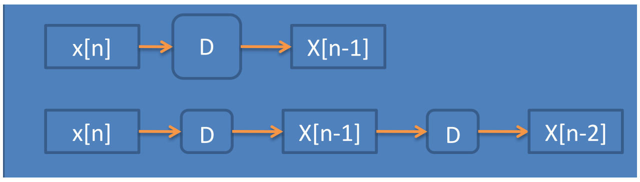


ure 3.2: The frequency content of the example signal, and low/highpass filters.

شکل 3.2 نیز اثر دو فیلتر بالاگذر (شکل پایین) و پایین گذر (شکل بالایی) را بر یک سیگنال ورودی (بالایی) نشان می دهد. آنطور که مشخص است هر دو فیلتر ایده آل نیستند چون یک بازه وسیعی را با شیب کم دربر می گیرند. بنابر این برای یک فیلتر خوب انتظار داریم شاهد برش تیز تر SHARP CUTOFF باشیم. این فیلترها شامل دو ضریب [0.5,0.5] برای فیلتر پایین گذر و [0.5, -0.5] برای فیلتر بالا گذر هستند.

فیلتر پاسخ ضربه محدود FIR

دسته هایی از فیلتر هستند که در پردازش سیگنال رقومی کاربرد زیادی دارند. این فیلترها مرکب از جمع کننده adder، ضرب کننده Multiplier و تاخیر دهنده Delay هستند. در دیاگرام هایی که فیلتر را نشان می دهد یک دایره با علامت مثبت وجود دارد که جمع کننده است و موقعی استفاده می شود که تعداد ورودی از 2 بیشتر باشد. از نظر ساختاری عمل جمع توسط یک سامانه درختی افزاینده ایجاد می شود. ضرب کننده ها از بخش های پیچیده تری برخوردارند: بنابر این در بعضی از اوقات توسط بخش های ساده تر تفکیک می شوند. بعنوان مثال برای ضرب با 8 بهتر است تا عدد را سه تا به سمت چپ جابجا کنیم. یک ایراد این کار در این است که رجیستر جابجا کننده مانند ضرب کننده دارای ویژگی انعطاف پذیری نیست و فقط وقتی کار می کند که اعداد مورد استفاده توان 2 داشته باشند مثلا در مورد بالا $2^3 = 8$. المان های تاخیری می توانند رجیستر باشند. در واقع چون رجیسترها برای ذخیره کردن یک مقدار برای زمان کوتاه مورد استفاده قرار می گیرند، بعنوان المان تاخیر دهنده تلقی می شوند. داشتن دو رجیستر در یک ردیف دو واحد زمانی ، سیگنال را تاخیر می دهند. شکل 3.3 نماینده حالت عمومی است که در آن واحد تاخیر دهنده باعث شیفت زمانی داده ورودی می شود. اصطلاح "پاسخ ضربه محدود" روندی است که طی آن فیلتر بر روی یک سیگنال اثر می گذارد. یک تابع ضربه یک ورودی خاص است که مقدار آن بجز در یک نقطه که واحد است، در بقیه مکانها صفر است. پاسخ یک فیلتر FiR در قبال این تابع محدود به زمان خاص است و لذا به فیلتر پاسخ ضربه محدود نام گذاری می شود.



بخش های یک فیلتر

همانطور که قبلا ذکر شد، بخش های محدود از یک سیستم را فیلترها تشکیل می دهند که این بخش ها بنام اضافه کننده، ضرب کننده و تاخیر دهنده هستند.IIR ها از این نوع فیلتر هستند. در مثال زیر واحد ضرب کننده و جمع کننده دیده می شوند.

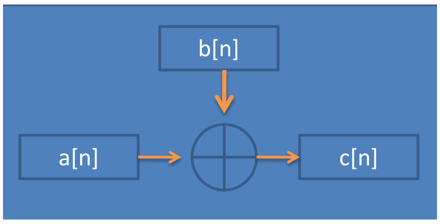
A=[1,2,3,4], b=[2,1,2,1]

c[0]=a[0]+b[0], c[1]=a[1]+b[1], c[2]=a[2]+b[2], c[3]=a[3]+b[3] c[0]=1+2, c[1]=2+1, c[2]=3+2, c[3]=4+1

C=[3,3,5,5]

C=[2,2,6,4]

$\Rightarrow$ c[n]=a[n]+b[n]



واحد تاخیری یک سیگنال را جابجا می کند. فرض می کنیم مقدار هر سیگنال خارج از محدوده اندیس صفر باشد بنابر این اگر سیگنال را یک واحد تاخیر دهیم می توانیم صفر را در شروع آن وارد کنیم. بعنوان مثال اگر سیگنال X را یک واحد تاخیر دهیم تا سیگنال Y ایجاد شود، و مقدار سیگنال X=[1,3,5] باشد سیگنال Y =[0,1,3,5] خواهد شد. بنابر این از نظر ریاضی این دو سیگنال بصورت زیر با هم رابطه خواهند داشت:

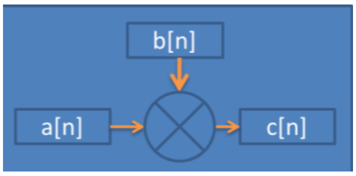
مثال: ضرب کننده ها

a= [1,2,3,4], b=[2,1,21], c?

c[0]=a[0]×b[0], c[1]=a[1] × b[1], c[2]=a[2] × b[2], c[3]=a[3] × b[3]

c[0]=1 ×2, c[1]=2 ×1, c[2]=3 ×2, c[3]=4 × 1

C=[2,2,6,4]$\Rightarrow$ c[n]=a[n]×b[n]

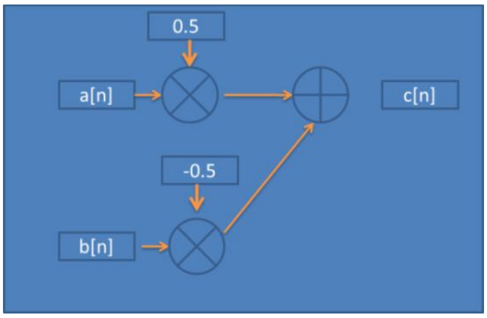


تلفیق ضرب و جمع کننده

a= [1,2,3,4], b=[2,1,2,1], c?

c[0]=[1× o.5-2 ×0.5, 2 ×0.5-1 ×0.5, 3 ×0.5-2 ×0.5, 4 ×0.5-1 ×0.5]

C=[-0.5,0.5,0.5,1.5]$\Rightarrow$ c[n]=0.5×a[n]-0.5×b[n]



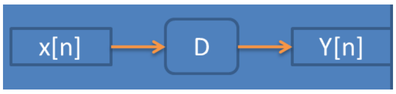
X[-1]=0,   y[0]=0,

X[0]=1,   y[1]=1,

X[1]=3,   y[2]=3,

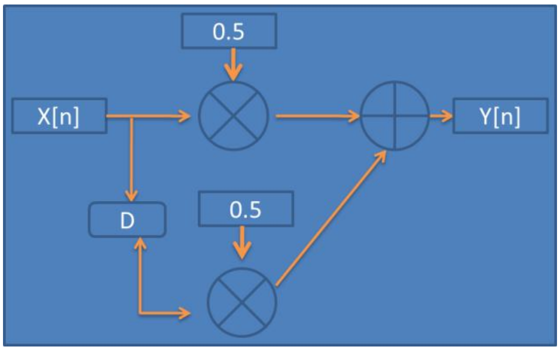
X[2]=5,   y[3]=5,

- مشخصاً :y[n]=x[n-1]
- بجای D (در شکل3.7) المان تاخیر را z-1 نیز نشان می دهند."z-transform"



ساختارهای فیلتر FIR

تا بحال دیده ایم که چگونه فیلتر ساخته می شود. حال چند تا از فیلترهای FIR را نشان می دهیم و به بعضی از ویژگی های آنها می پردازیم: توصیف FIR با معادله، و اینکه چگونه تابع ضربه واحد با آنها چگونه کار می کند. FIR ها منظم هستند و وقتی اعداد مرتبط با ضرب کننده ها مشخص باشند فیلترها کاملاً مشخص می شود. این اعداد به ضرائب فیلتر معروف هستند. شکل 3.8 یک FIR را نشان می دهد (با ضرائب 0.5,0.5).



جهت فلش ها یک سمت (چپ به راست) است که شاخص FIR هستند. به این ویژگی feed-forward گویند. فرض کنید در شکل قبل ورودی x[n]=[1,0] باشد. خروجی باید [0.5,0.5] و بعداً 0 در مرحله زمانی 0 ، x[0] بعنوان ورودی و هر چیزی قبل از این صفر است (پس ورودی D نیز صفر است) در مرحله بعد برای D مقدار x[0] ورودی ولی خروجی کل x[1] است:

y[0]=0.5x[0]+0.5x[-1]

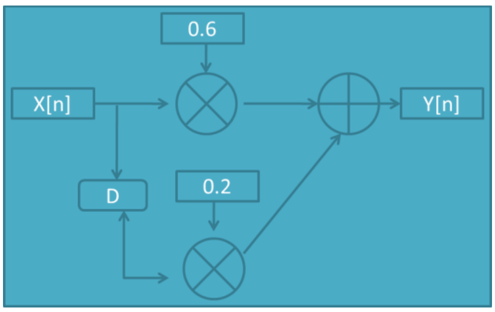
y[1]=0.5x[1]+0.5x[0]

y[n]=0.5x[n]+0.5x[n-1]

مفهومی بنام support برای یک سیگنال محدود و مجزا وجود دارد که در اندیس آن لحاظ می شود. در این مفهوم هر مقدار قبل و بعد از اندیس صفر تلقی می شود. support در درون اندیس نیز می تواند صفر باشد. مثلاً اگر حداقل درجه حرارت در طول هفته ملاک باشد، یک ورودی با اندیس یا support 7 خواهیم داشت مانند: [2,1,0,-1,0,2,0] که حتی مقدار صفر را نیز در حیطه ورودی شامل می شود.

مسئله

در شکل 3.9 اگر x={1} باشد، y چیست؟ y را بصورت یک معادله بیان کنید.



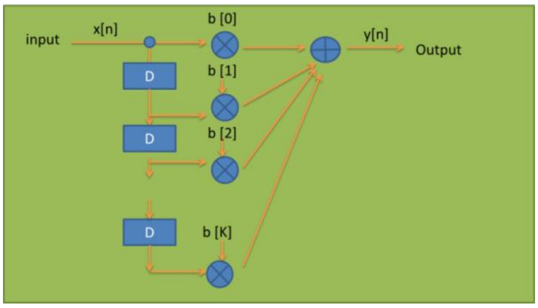
در شکل 3.10 قالب عمومی یک فیلتر FIR را برای فیلتر با ضریب K+1 نشان می دهد. (K+1 به این دلیل که ما از صفر شروع می کنیم تا K). تعداد ضرائب فیلتر تعداد taps نیز می گویند. بصورت قراردادی تعداد ضرائب به تعداد tap مساوی است. اما مرتبه این فیلتر را مرتبه k می نامند. یعنی taps=order+1 بعنوان مثال یک فیلتر با ضرائب {b0 ,b1 , ... ,bk}، K+1 ضریب یا taps دارد و مرتبه یا order این فیلتر K است.

برای شکل 3.10 معادله خروجی بصورت زیر است:

y [n]=b[0]x[n-0]+b[1]x[n-1]+b[2]x[n-2]+.....+b[k]x[n-k].

$$y[n]=\sum_{k=0}^{K} b[k]x[n-k]$$

شکل عمومی فیلتر FIR



مثال

با داشتن b [0.1,0.2,.0.3] و x [1,0,0,2,0,1,4,3] ، y را حساب کنید.

جواب:

y[n] = x[n-2]b[2]+x[n-1]b[1]+x[n]b[0]

معادله فوق را نوشته و چارت را می سازیم:

y[0] = 0 b[2]+ 0 b[1]+x[0]b[0]
y[1] = 0 b[2]+ x[0] b[1]+x[1]b[0]
y[2] = x[0] b[2]+ x[1] b[1]+x[2]b[0]
y[3] = x[1] b[2]+ x[2] b[1]+x[3]b[0]
etc

مقادیر موجود را در معادله قرار می دهیم: برای x بردار ستونی و مقدار اسکالر برای b، خواهد بود:

$$y[n]=\begin{bmatrix}0\\0\\1\\0\\0\\2\\0\\1\\4\\3\end{bmatrix}\times b[2]+\begin{bmatrix}0\\1\\0\\0\\2\\0\\1\\4\\3\\0\end{bmatrix}\times b[1]+\begin{bmatrix}1\\0\\0\\2\\0\\1\\4\\3\\0\\0\end{bmatrix}\times b[0]=\begin{bmatrix}0.1\\0.2\\0.3\\0.2\\0.4\\0.7\\0.6\\1.4\\1.8\\0.9\end{bmatrix}$$
$$\Rightarrow y[n]=[0.1,0.2,0.3,0.2,0.4,0.7,0.6,1.4,1.8,0.9]$$

در واقع حاصل ضرب یک ماتریس و یک بردار است:

$$y[] = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 2 & 0 \\ 2 & 0 & 1 \\ 0 & 1 & 4 \\ 1 & 4 & 3 \\ 4 & 3 & 0 \\ 3 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0.3 \\ 0.2 \\ 0.1 \end{bmatrix}$$

در قالب عمومی می توان نوشت:

$$\begin{matrix} y[0] \\ y[1] \\ y[2] \\ y[3] \\ y[4] \\ y[5] \\ y[6] \\ y[7] \\ y[8] \\ y[9] \end{matrix} = \begin{bmatrix} 0 & 0 & x[0] \\ 0 & x[0] & x[1] \\ x[0] & x[1] & x[2] \\ x[1] & x[2] & x[3] \\ x[2] & x[3] & x[4] \\ x[3] & x[4] & x[5] \\ x[4] & x[5] & x[6] \\ x[5] & x[6] & x[7] \\ x[6] & x[7] & x[8] \\ x[7 & x[8] & x[9] \end{bmatrix} \times \begin{bmatrix} b[1] \\ b[2] \\ b[3] \end{bmatrix}$$

می توان یکی از مقادیر (مثلاً) با اندیس 5 را نوشت:

y[5] = x[3] ×b[2]+ x[4]× b[1]+x[5]×b[0]

که معادل است با:

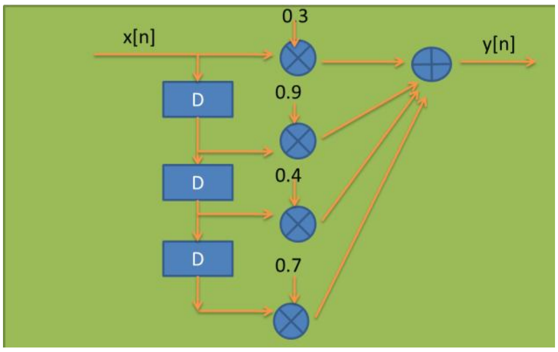
y[n] = x[n-2] ×b[2]+ x[n-1]× b[1]+x[n-0]×b[0]

که در واقع همان معادله قبلی است که اصطلاحاً کانولوشن نام دارد.

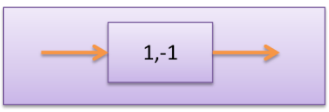
$$y[n] = \sum_{k=0}^{K} b[k]x[n-k] = \sum_{k=0}^{K} b_k x[n-k]$$

مثال

اگر b=[0.3,0.9,0.4,0.7] باشد و بر اساس معادله کانوولوشن، فیلتری که این معادله (کانولوشن) را اجرا می کند چیست؟



برای آسانی کار یک FIR را می توان مانند یک مستطیل نشان داد که در داخل آن اعداد هستند:



X[n]=1 یک مورد خاص است که ضربه واحد نام دارد چون یک مقدار غیر صفر منفرد یک واحدی است. ویژگی جالب آن این است که ضرایب فیلتری از نوع فیلتر FIR به ما می دهد. ما پاسخ ضربه واحد را h[n] نشان می دهیم. در واقع h برابر b به این معنی است پاسخ ضربه مانند ضرایب فیلتر، حداقل برای فیلتر FIR، است.

اجرای دستی کانولوشن

فرض کنید سیگنال ورودی x=[1,2,3,4,5] به یک فیلتر FIR با ضرائب b=[6,7,8] است. خروجی را محاسبه کنید.

```
1  2  3  4  5
6  7  8
......................
6  12  18  24  30
    7  14  21  28  35
       8  16  24  32  40
...........................................
6  19  40  61  82  67  40
```

تذکر: ضرب فیلتر در سیگنال ورودی نیز همین نتیجه را خواهد داشت.

عِلّی بودن، خطی بودن و ناوردایی نسبت به زمان

عِلّی بودن "casualty"، خطی بودن "linearity" و ناوردایی نسبت به زمان "time invariance" سه ویژگی یک سیستم محسوب می شوند. سیستمی عِلّی است که در محاسبه خروجی فقط از مقادیر حاضر و یا قبلی ورودی استفاده نماید. مثلاً سیستم y[n]=0.1x[n] علی است. ولی سیستم زیر علی نیست.

$$y[n]=\sum_{k=0}^{K} b_k x[n+k]$$

مسئله

کدامیک از سیستم های زیر علی نیست؟

$$y[n]=\sum_{k=0}^{K} b_k x[n-k]$$

$$y[n]=\sum_{k=-K}^{0} b[k]x[n-k]$$

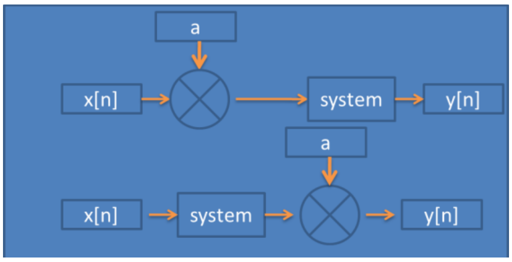
y[n] = 0.1x[n] +[0.4]x[n+1]+0.8x[n+2]

y[n] = 0.1x[n] -[0.4]x[n-1]-0.8x[n-2]

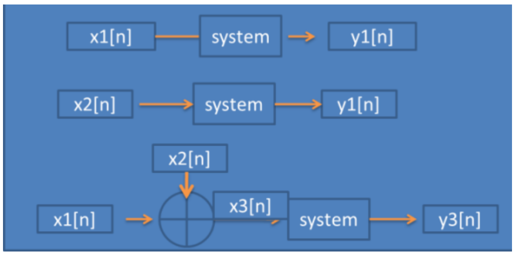
سیستم خطی

اگر یک سیستم خروجی داشته باشد که مساوی ورودی ضرب در یک عدد ثابت باشد به آن سیستم خطی گویند. یک سیستم خطی خصوصیت مقیاس پذیری scaling و جمع پذیری additivity دارد. مقیاس پذیری به این معنی است که نتایج بدست آمده از عملکرد سیستم وقتی که ورودی ضرب در یک مقدار ثابت شود مساوی وضعیتی است که خروجی ضرب در یک عدد ثابت شود. معنی جمع پذیری اینست که می توان دو سیگنال را با هم جمع کرد قبل از اینکه فیلتر خطی بر هر یک اعمال نمودیم یا می توان دو سیگنال را بعد از اعمال فیلتر خطی بر آنها، با هم جمع نمود یعنی y3=system(x1+x2) =y1+y2 جواب مساوی است.

خواص فیلتر scaling



خواص فیلتر additivity



خواص فیلتر "جمع پذیری"

عمل جمع پذیری با ضرب هر ورودی در عدد ثابت (نه ضرب ورودی ها با هم) نیز ممکن است. سیستم زیر خطی است. Y[n] و x[n] بترتیب خروجی و ورودی حاضر و x[n-1] ورودی قبلی و c ثابت است.

$$y[n] = c_0\, x[n] + c_1\, x[n-1] + c_2\, x[n-2] + \ldots + c_k\, x[n-k]$$

Time-invariant

ناوردایی زمان به این معنی است که خروجی یک سیستم انعکاس دهنده همان شیفتی است که سیگنال ورودی دارد. یعنی اگر ورودی x[n]، k واحد شیفت داده شود، ورودی x[n-k] خواهد بود و خروجی y[n] بصورت y[n-k] خواهد بود (مثل هم). فیلترهای FIR تا زمانی که ضرائب شان نسبت به زمان تغییر نکند، زمان ناوردا هستند.

مثال

فرض کنید $system_1$ فیلتر FIR است با h1[n]={1,1,1,1} و system2 فیلتر FIR دیگر با مقدار h2[n]={2,0,2}. همچنین فرض کنید x[n]={2,4,6,8,10} . اگر $output_1$ =h2*(h1*x) و $output_2$ =h1*(h2*x) ، این دو خروجی چه ارتباطی با هم دارند؟ چرا؟

دو سیستم خطی هستند و زمان ناوردا (LTI) بنابراین تاثیر عملگرهای کانوولوشن آنها مساوی است (h1*h2)*x .

مثال: آزمایش خطی بودن یک سیستم را بر حسب ورودی/خروجی بیان کنید.

بعنوان مثال:

System(x)=2x[n]-3x[n-1]

a×system(x)=system(a×x) آزمایش

system(a×x)=2(ax[n])-3(ax[n-1])

=a(2x[n])+a(-3x[n-1])

=a(2x[n]-3x[n-1])

=a×system(x)

آزمایش مساوی بودن **system(x1+x2)** با **system(x1)+system(x2)**

برای مثال:

هر دو آزمایش صحیح بوده و این سیستم خطی است.

$system(x_1+x_2)=2(x_1[n])+x_2[n])-3(x_1[n-1])+x_2[n-1])$
$= 2x_1[n])+2x_2[n])-3x_1[n-1])-3x_2[n-1])$
$= (2x_1[n]-3x_1[n-1])+(2x_2[n]-3x_2[n-1])$
$=system(x_1)+system(x_2)$

آزمایش برای زمان ناوردایی. سیستم را بر حسب ورودی/خروجی بیان کنید. بعنوان مثال:

System(x)=2x[n]-3x[n-1]

سیستم را امتحان کنید تا ببینید که آیا تعداد اختیاری از تاخیرات دارای تاثیر مشابه است فارغ از اینکه آن بر روی ورودی یا خروجی اعمال می شود. بعبارت دیگر delay(system(input)) را برای اینکه ببینیم با system(delay(input)) یکی است. توجه کنید که باید تمام توابع delay() را در نظر بگیریم. اگر تست برای تاخیر یک زمان شکست بخورد ولی برای دومی این مورد تکرار نشود گوییم این سیستم ناوردا است.

بعنوان مثال **x را به مقدار d واحد زمانی تاخیر دهید به این معنی که اندیس n را با n-d جابجا کنید بنابر این**

delay(x)=x[(n-d)] به این معنی که:

System(delayd(x))=2x[(n-d)]-3x[(n-d)-1

از بالا استنباط می شود که:

System(x)=2x[n]-3x[n-1]

بنابر این این تاخیر نتیجه زیر را به همراه دارد:

Delayd(System(x))=2x[(n-d)]-3x[(n-d)-1]

تست زمان ناوردایی سیستم

سیستم LTI با کانولوشن اجرا می شوند و عملگر کانولوشن جابجایی پذیر است یعنی ترتیب دستور مهم نیست:

h1*h2=h2*h1

همچنین خاصیت شرکت پذیری دارد :

h2*(h1*x)=(h2*h1)*x)

کانولوشن حاصیت توزیع پذیری نیز دارد:

x*(h1+h2)=x*h1+x*h2

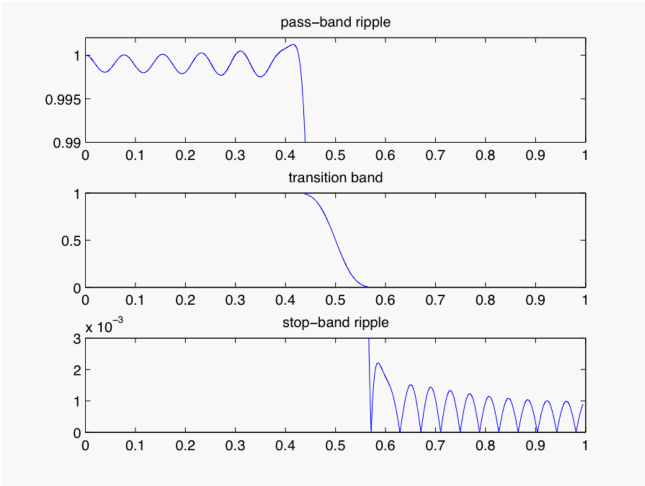
فیلترها (بخش دوم)

پاسخ فرکانسی فیلترها – فیلترهای IIR – correlation – پاسخ فرکانسی فیلترها

فیلترها می توانند چندین وظیفه را فقط توسط تغییر ضرائب فیلتر انجام دهند. وظائف این فیلترها وابسته به این است که این فیلترها با یک سیگنال ورودی در حوزه فرکانس چه رفتاری نشان می دهند. در این مورد یک سیگنال با فرکانس های مختلف وجود دارد و فیلتر مقادیری از این فرکانس ها را نگه داشته و مابقی را حذف می کند. بنابر این یک فیلتر می تواند پایین گذر، بالا گذر و میانگذر یا دندانه ای ”notch“باشد. بعضی از موارد واژه های دیگری نظیر متوقف کننده ”bandstop“ و باند پس زن ”bandreject“ انواع دیگری از فیلترها نظیر فرق گذار ”differentiator“ ، تمام گذر ”allpass“ و فیلترهای سازگار”adaptive filter” نیز وجود دارند. در مقایسه با فیلتر، سیستم را می توان بعنوان یک چیدمان برتر ”superset“ تلقی نمود. به عبارت دیگر یک فیلتر بخشی از یک سیستم است ولی الزاماً سیستم یک فیلتر نیست. اصطلاح های پایین گذر و بالاگذر جامع و فراگیر هستند. مثال خوبی برای یک فیلتر پایین گذر و بالاگذر را می توان در شکل زیر ملاحظه نمود. در طول محور x درصد هایی از فرکانس از صفر تا 1 هستند. اگر چه فرکانس واقعی به نرخ نمونه برداری “ f_s”وابسته است و محدوده ای از 0 تا fs/2 دارد. ”Frequency magnitude response“ بنابر این چه چیزی دو شکل فوق را برجسته می نماید. هر دو نشاندهنده یک انقطاع واضح هستند بین باند گذر ”pass band“ یعنی فرکانس هایی که از فیلتر می گذرند و باند ایست ”stop band“ یعنی فرکانس هایی که فیلتر آنها را از بین می برد. برای روشن شدن موضوع به شکل زیر فوق مراجعه کنید. فیلترها یک باند تدریجی بین گذر و ایست دارند. محدوده ای که پاسخ بزرگای فرکانسی ”Frequency magnitude response“ از در“حدود“ 100 به ” در حدود“ صفر درصد افت می کند. اما در اینجا باید بگوییم ”در حدود“ چون فیلترها باند گذر و ایست شکنج”ripple“ دارند. بجای مابقی یک پاسخ بزرگای ثابت در باندگذر، برای مثال، پاسخ در حقیقت به بالا و پایین شبیه یک موج سینوسی موج دارد که در اینجا شکنج ”ripple“ دارد. اما وقتی از ”باند تدریجی“ یا ”passband“ صحبت می شود، از این شکنج ها صرفنظر می شود. شکل 3.19 نمای نزدیک باندگذر شکنج (بالا)، باند تدریجی (میانی) و باند ایست (پایینی) را نشان می دهد. فارغ از این واقعیت که دارای فرکانس برش ”cutoff“ مختلف است این شکل از شکل 3.17 تفاوت دارد از این جهت که محور y خیلی کمتر در قسمت بالا و پایین شیب دارد. باند گذر و ایست شکنجی در هر دو شکل وجود دارد ولی در 3.17 قابل مشاهده نیست. چرا باند ایست در شکل 3.19 از قله های کوچک تشکیل شده (بجای سینوسی در باند گذر) است؟ چون نموداری که می بینیم ”پاسخ بزرگای فرکانسی“ است که از تبدیل فوریه گسسته بدست آمده ”discrete Fourier transform“ است. که در آن مقدار مختلط را بر می گرداند و در اینجا ما از تابع

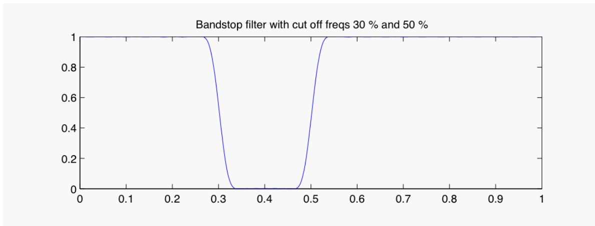
$$\sqrt{real^2 + imaginary^2}$$

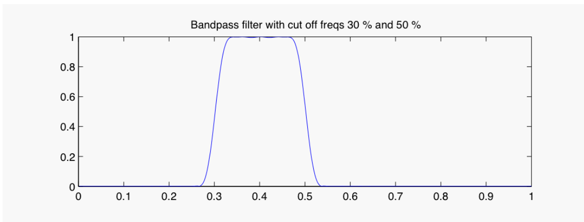
برای پیدا کردن پاسخ بزرگای فرکانسی استفاده می کنیم به همین دلیل بجای دیدن موج سینوسی فقط مقادیر مثبت را مشاهده می کنیم. همچنین ملاحظه می شود که قله ها بتدریج کوچکتر می شوند.



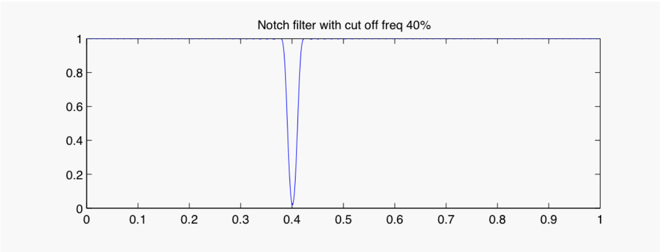
یک فیلتر پایین گذر ممکن است به نام فیلتر میانگین نیز خوانده شود در حالیکه فیلتر بالاگذر را با نام دیگر فیلتر تفاضلی "differencing" نیز می شناسند این نام ها در واقع بدلیل عملکرد این فیلترها به آنها منتسب می شود. بعنوان مثال اگر یک فیلتر FIR دارای دو ضریب 0.5 , 0.5 ملاحظه می شود که خروجی y[n]=0.5x[n]+0.5x[n-1] یا هم ارز آن y[n]=(x[n]+x[n-1])/2 میانگین ورودی حال و قبلی را ارائه می کند اگر ما پاسخ فرکانسی این را مورد بررسی قرار دهیم ملاحظه می کنیم که در این فیلترها فرکانس پایین حفظ و فرکانس بالا حذف می شود عکس این موضوع در مورد فیلتر تفاضلی صادق است. خروجی فیلتر تفاضلی y[n]=0.5x[n]–0.5x[n-1] یا y[n]=(x[n]-x[n-1])/2 است.

باندپس (bandpass) به این معنی است که فرکانس های در محدوده مشخص عبور نموده در حالیکه فرکانس های دیگر و خارج از این محدوده عبور نمی نمایند (میرا شده یا از بزرگای کمی برخوردار خواهند بود). بعنوان مثال یک فیلتر باندپس ممکن است اجازه عبور فرکانس های بین 10 تا 20 کیلوهرتز (kHz) را بدهد ولی برای فرکانس های بالاتر از 20 و کمتر از 10 محدودیت ایجاد کند. مفهوم ایست باند bandstop عکس این مورد است و در شکل های 20 و 21 مشخص است.

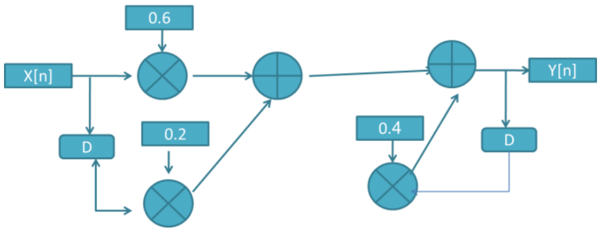




نمونه های دیگری از این فیلترها را می توان در شکل 22 ملاحظه نمود که در آن فقط محدوده باریکی از فرکانس ها قابل عبور هستند چیزی که بعنوان فیلتر شانه ای notch filter قبلا معرفی نمودیم و یا در شکل 21 فیلتر باندپس را ملاحظه می کنید که در دو گستره کار می کند. برای اینکه ببینیم چگونه یک فیلتر عمل می کند، می توانیم نموداری از پاسخ بزرگای فرکانسی ان را ببینیم (مانند شکل های قبل). برای داشتن پاسخ بزرگای فرکانسی اول، تبدیل فوریه مجزا DFT را بر روی ضرائب فیلتر انجام داده می شود. این ضرائب صفر بوده تا باعث نرم شدن منحنی شوند. در اینجا فرض بر آن است که تعداد ضرائب عبوری به این تابع کمتر از 128 باشد که توانی از عدد 2 هستند. مثال زیر به دو فیلتر توجه می کند: یکی پایین گذر و یکی بالاگذر. فقط نیمه نخست طیف را نشان می دهیم نیمه دوم تصویر، تصویر آینه ای آن است چون سیگنال ورودی حقیقی است.



فیلترهای IIR



فیلترهای پاسخ ضربه محدود FIR فقط محاسبات روبه جلو FEED-FORWARD را انجام می دهند. اما اگر اجازه دهیم پاسخ ها برگشت پذیر (بازخوردی) FEED-BACK باشد در این صورت بینهایت پاسخ خواهیم داشت. به همین منظور به این فیلترها فیلترهای پاسخ نامحدود گویند. فیلتر زیر را در نظر بگیرید. معادله خروجی آن به صورت زیر است:

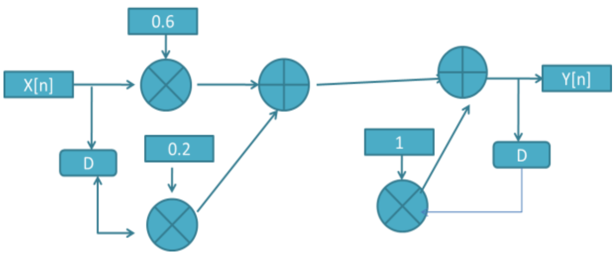
Y[n]=0.4y[n-1]+0.6x[n]+0.2x[n-1]

اغلب عبارت تاخیری اول نوشته می شود. خروجی y[n] در هر دو طرف عبارت آورده می شود به این معنی که بخشی از خروجی فعلی به خروجی قبلی وابسته است.

اگر یک تابع ضربه از این فیلتر عبور کند، خروجی بصورت زیر خواهند بود:

input	output
0	0
1	0 . 6
0	0 . 44
0	0 . 176
0	0 . 0704
0	0 . 02816
0	0 . 011264

خروجی با گذشت زمان کوچکتر می شود و در نهایت به صفر تمایل پیدا می کند. با توجه به محدودیت های توابع، در تحلیل این نوع از فیلترها این مورد مهم است. برای این نوع از فیلتر چون تمام ورودی بعدی (بعد از 1) تابع ضربه صفر است، بخاطر ویژگی بازخوردی بودن این فیلترها، خروجی های بعدی (0.4 * مقدار قبلی) خواهد بود. از نظر تئوریکی خروجی هیچگاه به صفر نمی رسد ولی با توجه به ماهیت دیجیتالی فیلترها که از دقت محدودی برخوردار است، در این مباحث مقدار به صفر می رسد. فرض کنید فیلتر شکل زیر در دسترس است. در یک نگاه سریع، این فیلتر به فیلتر شکل قبل مشابه است. تفاوت کمی وجود دارد اما قابل ملاحظه است چون ضرائب متفاوت هستند. در اینجا ضریب بازخورد 1 است.

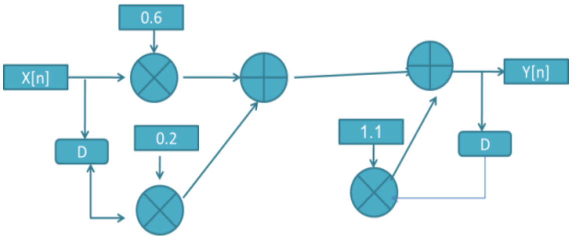


اگر یک تابع ضربه از این فیلتر عبور کند، خروجی بصورت زیر خواهند بود:

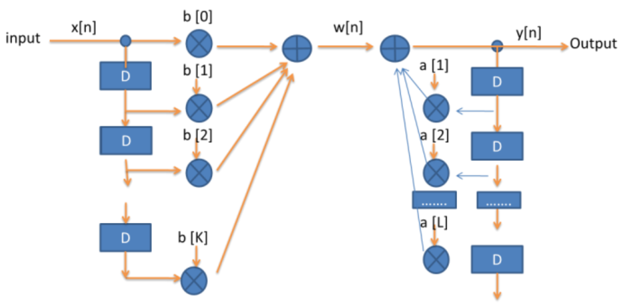
input	output
0	0
1	0.6
0	0.8
0	0.8

خروجی تا هنگامیکه ورودی صفر است 0.8 است: مفهومی که به پاسخ ضربه بینهایت معروف است. فیلترهای IIR که در شکل 3.26 دیده می شود ممکن است رفتار نامناسب داشته باشند. در اینجا ضریب بازخورد 1.1 است. خروجی این فیلتر در شکل 3.26 بصورت زیر است:

input	output
0	0
1	0.6
0	0.86
0	0.946
0	1.0406
0	1.14466
0	1.259126

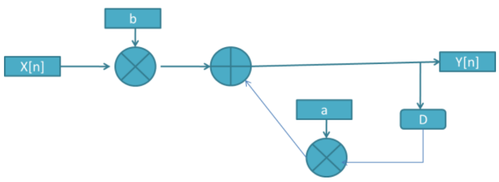


خروجی در این فیلتر ممکن است به بینهایت (مثبت یا منفی) برسد. یک فیلتر پایدار "stable" خوانده می شود در صورتیکه خروجی به صفر برسد در حالیکه ورودی به صفر افت کند. اگر خروجی در حول و حوش یک مقدار ثابت نوسان کند به این حالت "پایدار مشروط" گفته می شود. اگر به مقدار بینهایت میل نماید، به این فیلتر IIR، ناپایدار گفته می شود. شکل عمومی یک فیلتر در شکل 3.27 دیده می شود.



تمایلات یک فیلتر ساده IIR

شکل ساده یک فیلتر IIR که در زیر نشان داده شده است را در نظر بگیرید:



با برانداز کردن شکل فوق می توان معادله زیر را از آن استخراج کرد:

y[n]=bx[n]+ay[n-1]

برای یک تابع ضربه بعنوان ورودی، فقط مقدار اولیه y مستقیماً توسط x متاثر می شوند. با فرض اینکه سیستم در ابتدا در وضعیت سکون است (یعنی y[-1]=0) ، نخستین خروجی بصورت زیر است:

y[0] = bx[0]+0

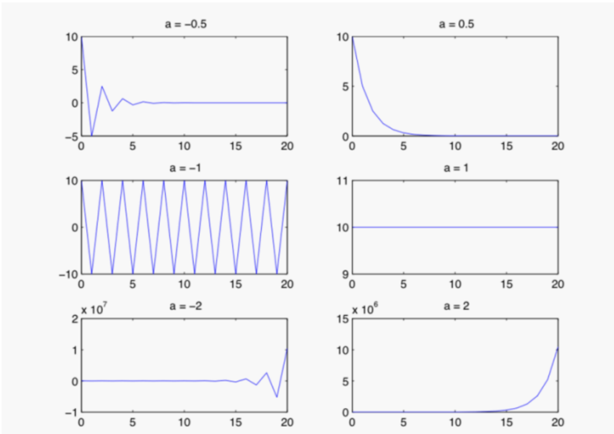
برای تابع ضربه فقط x[0] مقدار غیر صفر دارد و x[0]=1 است. بنابراین خروجی بصورت زیر است:

y[0]=b

و

y[n]=ay[n-1],   n>0

چگونه خروجی در تمام مدت بطور مستقیم به مقدار a وابسته است. در شکل (3.29) زیر آنچه که تعیین کننده است مقدار a است.



همبستگی correlation

همبستگی یعنی تعیین این موضوع که چقدر دو سیگنال با استفاده از کانوولوشن به هم شبیه هستند. با اعمال همبستگی عددی منفرد حاصل می شود که ضریب همبستگی نام دارد. یک عدد مثبت و بزرگ بعنوان ضریب همبستگی نشاندهنده ارتباط زیاد بین دو سیگنال است. در حالیکه عدد کوچک از ضریب همبستگی نشاندهنده عدم ارتباط بین دو سیگنال است. عدد منفی نشاندهنده مقایسه یا همبستگی منفی و به معنی این است که با افزایش یک سیگنال، سیگنال دیگر کاهش می یابد. مشکل وقتی جدی می شود که دو سیگنال ورودی را نمی توان به خط کرد چون محاسبه همبستگی نشاندهنده عدم وابستگی آن دو سیگنال است. برای رفع این کاستی یکی از سیگنال ها باید جابجا شود اما نمی دانیم این جابجایی چقدر باید باشد. گرچه با دیدن دو سیگنال می توان به مقدار این جابجایی پی برد. این از جمله مثال هایی است که حل آن برای انسان آسان ولی برای کامپیوتر مشکل است. بنابر این یک سیگنال N برابر جابجا می شود و حداکثر مقدار از همبستگی متقاطع cross correlation بدست می آید. همبستگی از طریق کانوولوشن و از طریق معکوس نمودن یک سیگنال انجام می شود. بجای در نظر گرفتن سیگنال y[k] با اندیس صفر تا عدد مثبت k آنرا با محوریت صفر و تا k− معکوس می سازیم. بعنوان مثال اگر y[k]={5,3,1,6} باشد، می توان آنرا بصورت y[0]=5,y[1]=3,y[2]=1,y[3]=6 تفسیر می نماییم. اگر این سیگنال را معکوس نماییم، خواهیم داشت:

y[-3]=6,y[-2]=1,y[-1]=3,y[0]=5

اگر y با x کانوولو شود، باید x[n]×y[n-k] را جمع بزنیم. با استفاده از نسخه وارونه (معکوس) y تنها چیزی که تغییر می کند، اندیس است یعنی x[n]×y[n-(-k)] یا x[n]×y[n+k] . خروجی سیگنال حاصل از همبستگی، همبستگی متقاطع نام دارد و اگر یک سیگنال با خودش مورد بررسی قرار گیرد، خود همبستگی یا autocorrelation نام دارد. اما تعداد نمونه داده همبستگی متقاطع را متاثر می سازد. بنابر این همبستگی متقاطع باید بر تعداد نمونه تقسیم شود.

$$s_{x,y}[k] \approx \frac{1}{N}\sum_{n=0}^{N-1} x[n]y[n+k]$$

معادله بالا تخمینی از

$s_{x,y}[k]$

را ارائه می کند چون مقدار متوسط را مورد توجه ندارد و فرض می کند که مقدار آن برای x و y صفر است. اگر مقدار متوسط سیگنال غیر صفر باشد، نیاز است تا بجای آن از معادله زیر (کوواریانس متقاطع) استفاده نماییم:

$$s_{x,y}[k]=\sum_{n=0}^{N-1}x[n]y[n+k]-\frac{(\sum_{n=0}^{N-1}x[n])(\sum_{n=0}^{N-1}y[n])}{N}$$

تخمین خوبی از همبستگی متقاطع به این معنی است که بایست اتوکوواریانس را نیز بدست آوریم:

$$s_{x,x}=\sum_{n=0}^{N-1}x[n]^2-\frac{(\sum_{n=0}^{N-1}x[n])^2}{N}$$

برای y داریم:

$$s_{y,y}=\sum_{n=0}^{N-1}y[n]^2-\frac{(\sum_{n=0}^{N-1}y[n])^{2h}}{N}$$

تخمینی از همبستگی متقاطع را می توان بصورت زیر ارائه نمود:

$$\rho_{x,y}[y]=\frac{s_{x,y}[k]}{\sqrt{s_{x,x}s_{y,y}}}$$

برای خود همبستگی autocorrelation ، بجای x، y قرار می دهیم.

$$\rho_{x,x}[k]=\frac{s_{x,x}[k]}{s_{x,x}}$$

توجه کنید که

$\rho_{x,x}[k]$

یک تخمین است چون ما با داده های نمونه برداری شده سر و کار داریم نه با فرایندی که این داده ها را می سازد. همانطور که احتمالا حدس زده باشید این روابط از آمار و احتمال می آید. همبستگی متقاطع با واریانس

σ^2

مرتبط است.

در ادامه به ذکر چند مثال می پردازیم. در این مثال دو سیگنال x و y داریم که مشابه ولی جابجا شده اند. همچنین بعد از نخستین محاسبه، از محدوده مختلفی از برای y با مشخص نمودن [y(10), y(1:9)] که آخرین مقدار y را بعنوان عضو اول به 9 تا عضو نخست دیگر می چسباند. نیازی به یافتن جمع نمودن ضرب نقطه به نقطه توسط دستور sum() نداریم چون این امر بطور اتوماتیک انجام می شود. (یادآوری می شود که:

[a b c]×transpose[d e f]=ad+be+cf

در نهایت عدد را بر تعداد نمونه ها تقسیم می کنیم.

```
x=[0   0   1   5   1   -2   -3   -2   0   0];
```

```
Y=[1 5 1 -2 -3 -2 0 0 0 0];
X*y.'/length(x)
Ans=-0.800
X*[y(10),y(1:9)].'/length(x)
Ans=2
X*[y(9:10),y(1:8)].'/length(x)
Ans=4.4
.
.
X*[y(2:10),y(1)].'/length(x)
Ans=1.900
```

89

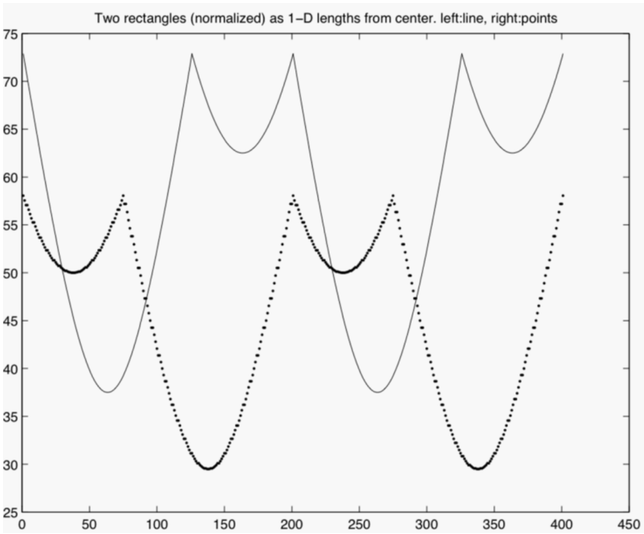
جواب از 1.9- تا 4.4 در تغییر است.

```
[y(9:10),y(1:8)]
ans=
0 0 1 5 1 -2 -3 -2 0 0
>> x
x=
0 0 1 5 1 -2 -3 -2 0 0
```

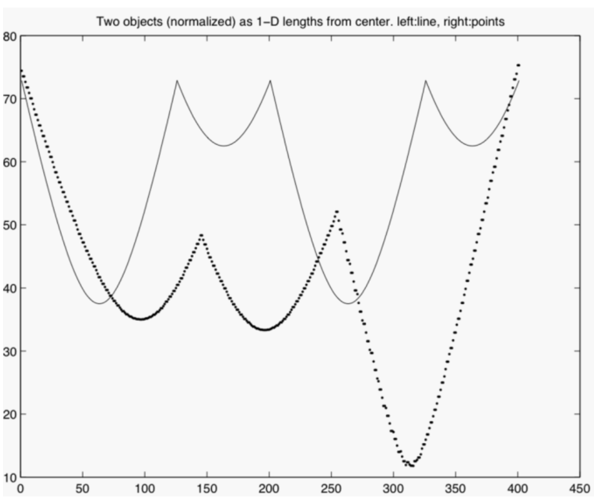
یعنی بزرگترین پاسخ وقتی است که دو سیگنال کاملاً در یک ردیف قرار گیرند.

در یک مثال دیگر، X را مشابه قبل ولی y را در منفی ضرب می کنیم. در واقع عملیات فوق بدست آوردن همبستگی نیست بلکه همبستگی متقاطع است چون حالا سیگنال y متفاوت از سیگنال x است. مشکل دیگر که با این مقدار وجود دارد این است که مشخص نمی کند که آیا این دو سیگنال کاملاً با هم منطبق هستند. به عبارت دیگر در مثال اول کوواریانس متقاطع بین x و y 4.4 بود سوال اینجاست که اگر یک سیگنال دیگر وجود داشت مقدار کوواریانس متقاطع بیشتری بدست می آمد؟ جواب مثبت است. در مثال بعد این موضوع را نشان می دهیم. برای این کار سیگنال y را در 10 ضرب می کنیم. در این شرایط مقدار کوواریانس متقاطع 10 برابر قبل خواهد شد (یعنی 44). برای حل این مشکل همبستگی متقاطع را بدست می آوریم که نیاز به دو داده دیگر داریم: اتوکوواریانس دو سیگنال. در برنامه نویسی ایندو متغیرهای sxx و syy هستند. متغیر sxy بسیار مشابه مثال های بالایی هستند بجز آنکه هیچ فرضی در مورد میانگین سیگنال ندارند. برنامه زیر همبستگی متقاطع را بین دو سیگنال پیدا می کند. وقتی برنامه اجرا شود ملاحظه خواهیم نمود که مقادیر بسیار بهتر از گذشته شدند. زیرا عدد حاصل مقداری است که تا 1 در تغییر است و مقداری که به ما در باره سازگاری دو سیگنال یا تشابه آنها به هم گزارش می دهند. در صورت انتخاب ضریب منفی (یعنی y=-y) مقدار همبستگی -1 می شود که دال بر این است که بین x و yn همبستگی منفی وجود دارد یعنی با افزایش یک پارامتر (x) پارامتر دیگر (y) کاهش می یابد. مثال زیر از یک پروژه پردازش تصویر اقتباس شده است. در شکل 3.30 دو مستطیل می بینیم. هدفمان این است که بطور خودکار این دو تصویر (در اینجا شیئ) را بر هم منطبق یا مطابقت دهیم چیزی که از دید کامپیوتر بسیار سخت ولی انسان براحتی می تواند آنرا تشخیص دهد (یعنی این دو چیز یا دو نفر به هم چقدر شباهت دارند). مستطیل ها در زاویه به هم شباهت دارند ولی اندازه (اضلاع) آنها متفاوت است همچنین ممکن است یکی نسبت به دیگری بچرخد. (شکل)

برای اینکه بکمک کامپیوتر شباهت ایندو را بررسی نماییم، روی مرز (اضلاع) حرکت نموده و مرکز مستطیل را مشخص می نماییم. از این نقطه فاصله هر یک از نقاط بر روی مستطیل را با مرکز بدست می آوریم (حرکت از گوشه سمت چپ بالا در جهت عقربه ساعت). وقتی این اندازه گیری در یک سیکل انجام شد، یک تصویر یک بعدی از این شکل را آنطور که در شکل (3.31) زیر نشان داده شده ملاحظه خواهیم نمود. نمودار خط ممتد متعلق به تصویر سمت چپ و نمودار نقطه چین متعلق به شیء سمت راست است.



توجه نمایید که طول مسیر اندازه گیری برای دو شیء مشابه است. این مورد برای بررسی همبستگی مورد نیاز است. طوری که شکل کوچکتر گسیده شده تا شکل بزرگتر بدست آید. مقادیر بصورت d1 (شکل اول) و d2 (شکل دوم) بوده و برنامه زیر تلاش می کند تا با کشیدن شکل دوم آنرا مانند شکل اول نماید. برنامه زیر به همین مناسبت نوشته دیده شده است. ضریب همبستگی حاصل از خروجی این برنامه 0.99 است. حال اگر عملیات مشابه را برای دو شکل مستطیل و مثلث اجرا نماییم و برنامه مربوطه را در محیط متلب نوشته و اجرا نماییم شکل زیر (3.33) حاصل می شود. با بررسی ضریب همبستگی ملاحظه می شود که در این وضعیت مقدار ام در حدود 0.62 است.



سینوسی ها

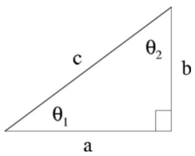
در این بخش می خوانیم: – مقدمه ای بر سینوسی ها – مروری بر هندسه و مثلثات – عدد π – دایره واحد – مقدار اساسی اختلاف فاز – دامنه – سیگنال های هم نوا یا هارمونیک – بیان سیگنال بصورت مجموعی از سینوسی ها – طیف – نتیجه گیری

مقدمه ای بر سینوسی ها

اغلب سیگنال های آنالوگ، یا سینوسی هستند یا ترکیبی از سینوسی ها (و یا تخمینی از سینوسی های ترکیبی). از این نظر سیگنال ها باید جالب باشند چون می توان بآسانی با هم جمع زد، یک تکانش زمین (زمین لرزه) بمانند ضربه ای که بر کلیدهای یک پیانو می زنیم، ترکیبی از سینوسی ها را که باهم جمع می شوند بوجود می آورند. تنها تفاوت آنها این است که این سینوسی ها بر خلاف سینوسی های ایده آل، میرا هستند. در این فصل ما سینوسی ها را بررسی کرده و تحقیق می کنیم که چطور آنها با هم جمع شده تا سیگنال را بوجود آورند. هدفمان نایل شدن به فهم بهتر سینوسی ها و آماده شدن برای تبدیل فوریه است.

مروری بر هندسه و مثلثات

شاید بیمورد نباشد بگوییم بخش مهمی از متغیرهایی که در ریاضیا ت استفاده می شود از کلمات یونانی بهره می گیرد. و اینکه دلیل عمده آن این است که حرف لاتین برای تمام آنها کم باشد ویا اینکه بعضی از اوقات مانند استفاده از l یا o مار به غلط به سمت 1 یا 0 می کشاند. با در نظر گرفتن رابطه فیثاغورس در رابطه با حل هندسی طول اضلاع مثلث قائم الزاویه به کمک رابطه $a2+b2=c2$ ، (شکل 4.1) می توان مقادیر زوایای موجود (θ_1 و θ_2 را نیز بدست آورد: $\cos(\theta1)=a/c$ از طریق کسینوس زاویه θ_1 و $\sin(\theta1)=b/c$ از سینوس زاویه θ_2 . با توابع arcsin و arccos و از روی نسبت اضلاع (روابط فوق) می توان به مقادیر زوایای موجود نیز دست یافت.



عدد π

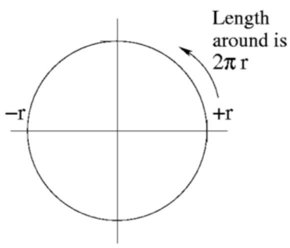
نمی توان چیزی از سینوس و کسینوس گفت بدون اینکه از عدد π (یعنی نسبت بین محیط دایره به قطر"شکل 4.2") صحبت به میان آورد: محیط دایره =2πr که در آن r شعاع است. پی یک عدد گنگ است. تاکنون در محاسبه پی اعداد تکراری دیده نشده است (مثل 3/1). مصری ها مقدار آنرا 81/256 در حالیکه بابلی ها مقدار 3.1415927 برای پی در نظر گرفتند. ما مولفه های سینوسی با فرمت زیر را مشخص می کنیم:

$a*\cos(2\pi ft+\varphi)$

در رابطه بالا a دامنه، f فرکانس، و φ اختلاف فاز است. متغیر t بیان کننده زمان است. سینوس می تواند با کسینوس مورد استفاده قرار گیرد اگر یک π/2 به آن اضافه شود یعنی:

$\cos(\theta)=\sin(\theta+ \pi/2)$

دامنه (a) ، اختلاف فاز (که زاویه فاز نیز نامیده می شود) و فرکانس بطور کامل یک سینوسی را تعریف می کنند.



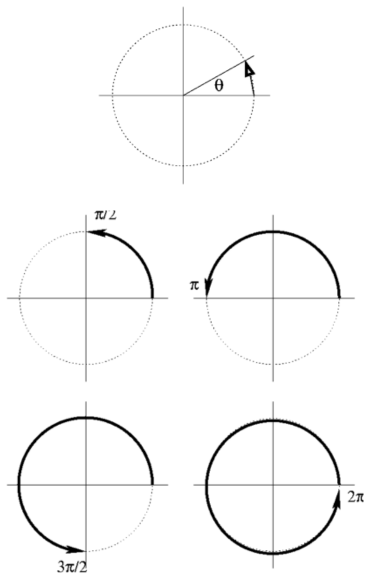
هم بزرگا و هم دامنه دلالت کننده بر یک کمیت بدون بعد هستند ولی بزرگا همیشه مثبت است. |X(t)| به بزرگا اشاره دارد و همانطور که ممکن است انتظار داشته باشید، قدر مطلق تابع در متلب بزرگا را برمی گرداند. در اغلب موارد در DSP بزرگا و دامنه قابل تبدیل هستند. با نگاهی به مولفه سینوسی بعنوان یک مبدل فاز، جاییکه ما آنرا در وضعیت اولیه آن رسم می کنیم بدون نگرانی در مورد چرخش با زمان می توان مبدل فاز را در طول مثبت محور ترسیم کنیم سپس در خلاف عقربه ساعت بطور زاویه ای آنرا حرکت دهیم. طول منفی به این معنی است که مبدل فازی را در طول منفی محور مختصات ترسیم می کنیم. اما این دقیقا برابر ترسیم طول در امتداد مثبت محور مختصات باضافه چرخش آن به اندازه 180 درجه است چون :

A*cos(2πft+φ+π) = -A*cos(2πft+φ)

جریان مستقیم که در آن فرکانس مساوی صفر است، از این قاعده مستثنی است. در این جریان زاویه فاز نیز صفر است. از f بعنوان فرکانس تناوبی یا دوره ای و یا برای سادگی بعنوان فرکانس استفاده می شود که گاه فرکانس رادیان یا همان ω نیز خوانده می شود. واحد این کمیت هرتز (به افتخار Hertz)، است و چون بنام یک فرد است با حرف بزرگ نشان داده می شود.)

دایره واحد

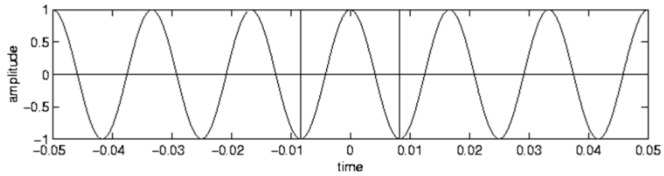
دایره واحد خصوصیت ویژه ای دارد: شعاع آن واحد بوده و وقتی بریده و باز شود، دارای طول 2π است. در نتیجه می توان از زاویه ای گفت که این زاویه بین دو خط با طول واحد و به مقدار کسری از 2π است. شکل 4.3 طول کمان که کسری از محیط دایره است را نشان می دهد. همچنین شکل 4.4 طول کمان را بر روی دایره واحد برای چند زاویه متداول نشان می دهد. واحد دیگر اندازه گیری زاویه درجه است که برای تبدیل این دو واحد از رابطه : a(deg)* 2π/360 برای تبدیل درجه به رادیان و a(rad)*360/2π برای تبدیل رادیان به درجه است. ویژگی دیگر توابع سینوسی در تغییر مقادیر آن از 1 تا −1 است.



مقدار اساسی جابجایی فاز

مقدار یک تابع سینوسی از منفی تا مثبت بینهایت در محور زمان است. اگر بخواهیم آنرا بر حسب محور زمان ببینیم می توانیم بین حداکثرهای آن با محور زمان ارتباط برقرار نماییم. شکل 4.5 سینوسی cos(2π60t) را با فرکانس 60Hz نشان می دهد. بنابراین دارای دوره تناوب 60/1 یا 0.0167 ثانیه است. خطوط عمودی جاییکه که یک دوره شروع و یا تمام می شود یعنی در زمان های t=+0.0167/2 و t=-0.0167/2 را نشان می دهد. این مقدار معادل مقدار تابع در −π و +π است که در آن و در t=(1/60)(-1/2) باعث می شود تابع cos(2π60t) معادل cos(2π60(1/60)(-1/2)) =cos(-π) شود. اما چه حداکثری را انتخاب می کنیم. در شکل 4.5 خیلی آسان است اما معمولا یک سینوسی دارای اختلاف فاز است. چون در یک 2π تکرار می شود، می توانیم هر حداکثری را که می خواهیم سیگنال را طبقه بندی نماییم انتخاب نماییم. یعنی همیشه یک حداکثر مثبت بین −π و +π

قرار دارد. هر موج سینوسی دارای یک قله مثبت بین $-\pi$ و π است. و زاویه ایندو یک زاویه جابجایی فاز است. مقدار قله در صفر بنام جابجایی فاز اساسی "principal phase shift" نام دارد.

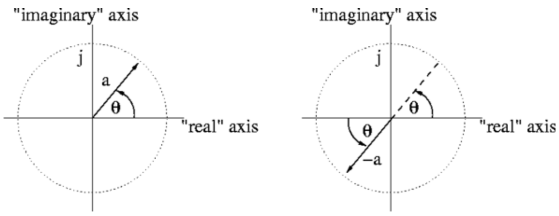


دامنه "amplitude"

معمولا مقدار دامنه مثبت است. اگر دو سینوسی $x1= A*\cos(2\pi ft+\varphi)$ و $x2=-A*\cos(2\pi ft+\varphi)$ را در نظر بگیرید کمترین و بیشترین مقدار دامنه -1 و 1 است. مقدار x1 از a تا -a و x2 از -a تا a با فرض اینکه ایندو دارای فرکانس یکسان هستند، می توان نتیجه گرفت که ایندو به هم شبیه هستند. بنابراین تنها تفاوت آنها در اختلاف فاز است که مقدار آن π است. در واقع

$\varphi 2= \varphi 1+\pi$ است. همچنین می توان استنباط نمود که تبدیل یک عدد موهومی $x+jy$ به عدد قطبی $r<\theta$ است:

معادله $(\sqrt{(x^2 + y^2)}$ منجر به مقدار عدد مثبت برای r می شود. یک بردار با طول مقدار منفی با مقدار با طول مثبت همان بردارکه به مقدار یک نیم دایره چرخیده مساوی است. سیگنال ها بعنوان اجتماع سینوس ها شناخته می شوند. اگر این سینوس ها به هم وابسته باشند این سیگنال را هارمونیک یا هم نوا می گویند.



سیگنال های هم نوا harmonic

سیگنال های واقعی می توانند به یک دسته از سینوس ها که با هم جمع شدند، تجزیه شوند. یک سیگنال هارمونیک مرکب از چندین سینوسی مجتمع که هر یک در یک فرکانس پایه (فرکانس اساسی fundamental frequency) ضرب (عدد صحیح) شدند، تشکیل می شود. بعنوان مثال سیگنال زیر یک هارمونیک است:

X(t)=0.4cos(2pi15t+pi/5)+0.7cos(2pi30t)+cos(2pi45t-pi/4)

سیگنال x(t) به شکل زیر نوشته می شود:

X(t)=a1cos(2pi(1)f0t+φ1)+a2cos(2pi(2)f0t+φ2)+a3cos(2pi(3)f0t+φ3)

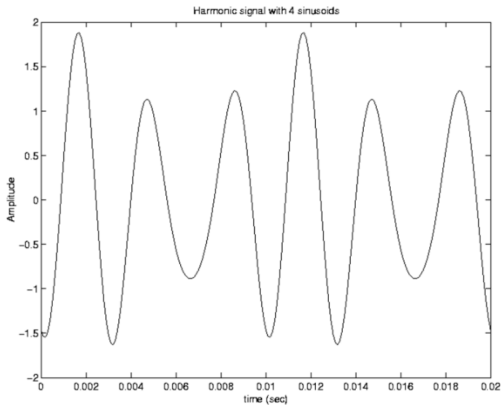
در معادله بالا برای x(t)، a1=0.4، a2=0.7، a3=1، φ1=pi/5 ، φ2=0 و φ3 =-pi/4 و f0=15Hz. هر سیگنال مشابه این بعنوان هارمونیک شناخته می شود. اگر دامنه صفر باشد، بنابراین سینوسی وابسته به آن در تابع مشارکت نخواهد داشت. به عبارت دیگر 0*cos (anything) مساوی صفر است. اگر تابعی مشابه زیر وجود داشته باشد:

X(t) = 0.1 cos (2pi100t-pi/6) + 1.3cos (2pi300t+pi) + 0.5*cos (2pi400t+2pi/3)

بنابر این x2 (t) یک هارمونیک با

a1 = 0.1 ،a2 = 0.0 ،a3 =1.3،a4 =0.5 ،φ1 =-pi/6 ، φ2 =0 و φ3 =pi ، φ4 =2pi/3 و f0 = 100Hz

بخش دیگری از اطلاعات در زمینه سیگنال های همنوا این است که آیا سیگنال جابجایی عمودی نیز دارد یا نه. به عبارت دیگر آیا گراف شکل 4.7 در محور x متمرکز است؟ این کار با اضافه نمودن مقدار ثابت به سیگنال انجام می شود. بعنوان مثال اگر 100 واحد به سیگنال اضافه کنیم شکل سیگنال تغییر نمی کند ولی حالا روی خط y=100 تمرکز پیدا می کند. این بخش از اطلاعات به هارمونیک بالا با اضافه نمودن عبارت $a_0\cos(2\pi(0)f_0t+\varphi_0)$ انجام می شود. در ابتدا ضریب صفر است به این معنی که مولفه نخست آن یک مقدار ثابت است. در این خصوص نیازی به مشخص نمودن φ_0 نیست چون $\cos(\varphi_0)$ مقدار ثابت است و این مورد در مقدار a_0 خود را نشان می دهد. بنابراین فرض می شود که φ_0 صفر است. این عبارت اغلب مولفه DC برای جریان ثابت خوانده می شود.



مثال: از سیگنال های زیر کدامیک هارمونیک و کدام غیر هارمونیک هستند. اگر هستند فرکانس پایه شان چیست؟

$X_1(t)=2\cos(2\pi 7t)+3\cos(2\pi 35t)$

$X_2(t)=2\cos(2\pi 2t)+3\cos(2\pi t)$

$X_3(t)=2\cos(2\pi 7t)+3\cos(2\pi t-\pi/4)$

برای اینکه یک سینوسی هارمونیک باشد، باید هر یک از آنها فرکانسی داشته باشند که ضریب فرکانس پایه باشد: در اینجا k عدد صحیح است.

$$x(t)=\sum_{k=1}^{N} a_k \cos(2\pi k f_0 t+\phi_k)$$

بنابر این در رابطه بالا سیگنال ها را قرار می دهیم و می بینیم که آیا آنها ضرایبی از اعداد صحیح هستند. به عبارات توابع کسینوسی برای x_1 نگاه کنید و در نظر بگیرید k_1 و k_2 مقادیری از k هستند.

$2\pi 7t=2\pi k_1 f_0 t+\varphi_k$, let $\varphi_k=0$

$f_0=7/k_1$

$2\pi 35t=2\pi k_2 f_0 t+\varphi_k$, let $\varphi_k=0$

$f_0= 35/k_2$

$7/k_1 =35/k_2$

$k_1/k_2=7/35=1/5$

بنابر این اگر $k_1=1$ باشد ، $k_2=5$ خواهد بود. در نتیجه:

$f_0=35/k_2=35/5=7\text{Hz}$

بنابر این $x_1(t)$ دارای مقدار صحیح بوده و هارمونیک است. $x_3(t)$ نیز اینچنین است اما $x_2(t)$ مقدار k صحیح ندارد (بخاطر فرکانس π) و بنابراین هارمونیک نیست. سیگنال سومی یک خرده فریبنده است چون مولفه های فرکانسی آنطور که انتظار داریم بخط و مرتب نیستند.

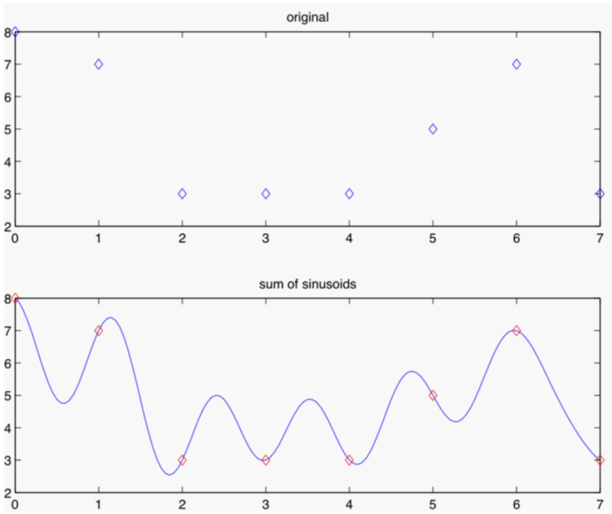
$f_0 = 1Hz$, $a_0=1$, $a_k=\{3,0,0,0,0,0,2\}$, and $\varphi_k=\{-\pi/4,0,0,0,0,0,\pi\}$

همچنین می توان هارمونیک یک سیگنال با مولفه DC را در نظر گرفت. با توجه به قاعده عمومی فوق الذکر، می توان اندیس K را که از صفر شروع می شود و به معنی این است که ما دارای مولفه فرکانسی $0*f_0$ یا صفر هرتز هستیم، را در نظر بگیریم. چون در نظر گرفتن یک مقدار برای φ_0 فقط در مقدار دامنه a_0 خود را نشان می دهد که در یک مقدار ثابتی ضرب می شود، بایستی زاویه فاز را صفر در نظر بگیریم بنابر این a_0 بطور کامل مولفه DC را مشخص می کند. نشان دادن یک سیگنال که ناپیوستگی هایی داشته باشد، نظیر موج مربعی که از اجتماعی از سینوسی هایی که بطور هارمونیکی به هم وابسته اند، مشکل است. عبارات زیادی را نیاز است تا در اینجا مرور کنیم. بر اساس پدیده Gibbs نمیتوان بطور کامل یک سیگنال ناپیوسته را بعنوان مجموعه ای از سینوسی ها در نظر بگیریم ولی می توان آنرا بصورت اوورشوت، آندرشوت یا ریپل فرض کنیم. تحلیل فوریه با سیگنال پریودیک بهترین عملکرد را دارد. انتظار داریم که نمونه ابتدایی و انتهایی با هم مرتبط شوند (در صورتی که یک نمونه بیشتر بر داریم) که این نمونه مشابه اولین نمونه یا آخرین نمونه ای که برداشتیم باشد. ضمنا اگر یک سیگنال واقعاً پریودیک باشد، با نمونه برداری خوب انتظار داریم که آخرین نمونه مشابه اولین نمونه باشد.(یعنی نمونه N+1 با نمونه 0 مساوی باشد). بعضی وقتها با سیگنال پریودیک مواجه نیستیم در این صورت بین نمونه آخری و اولی ناپیوستگی داریم. بعنوان مثال اگرسیگنال زیر را پریودیک در نظر بگیریم، انتظار داریم که این سیگنال تا بینهایت بار تکرار شود. در اینجا ملاحظه می شود که یک ناپیوستگی در میانه وجود دارد که برای ایجاد سری فوریه مشکل ایجاد می کند (پدیده گیبس را بخاطر بیاورید). بنابراین تحلیل فوریه، سیگنال مشابه سیگنال اولی را نتیجه دهد. این تخمین بویژه در اطراف ناپیوستگی ضعیف است. این اشکال به اثر گوشه edge effect معروف است. با این وجود تحلیل فوریه منجر به تخمین مناسبی از سیگنال اولیه می شود. در واقع هر سیگنال غیر پیشا nonrandom توسط تحلیل فوریه تخمین زده می شود.

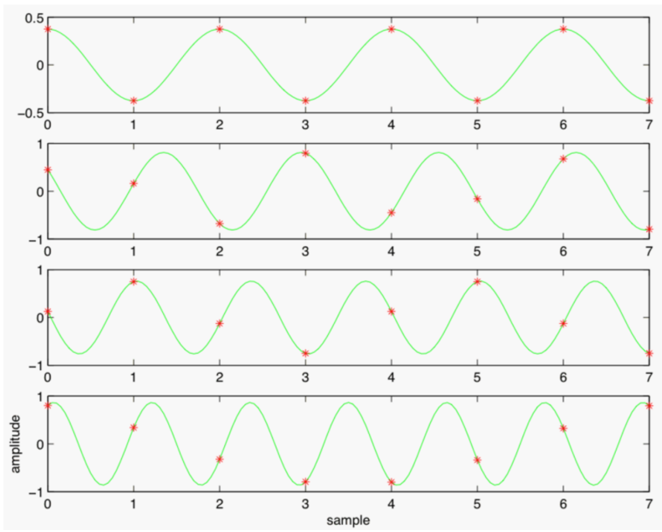
اگر یک سیگنال پیشا را در نظر بگیریم و سری فوریه آنرا بیابیم، در خواهیم یافت که این سری برای مقادیری که مورد نظر ما بوده ، سیگنال را تخمین زده است. با اینحال ما راغب هستیم تا بتوانیم به کمک این سری مقادیر آتی سیگنال را نیز تخمین بزنیم.

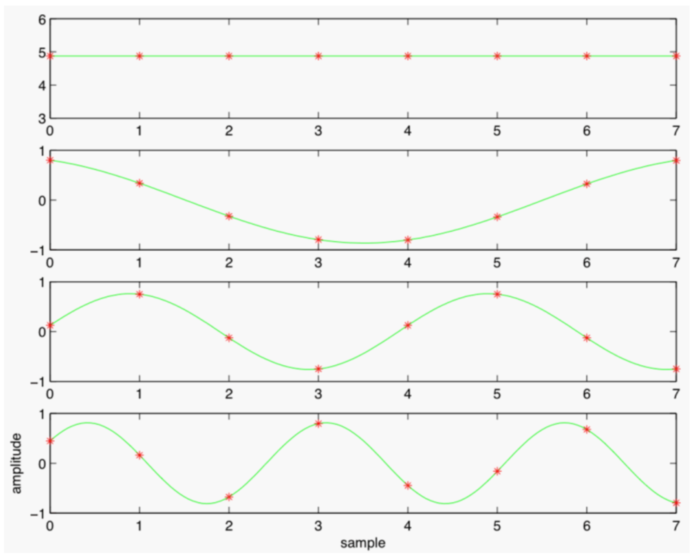
ارائه یک سیگنال دیجیتال بعنوان جمع سینوسی ها

یک سیگنال دیجیتال را می توان از اجتماع سینوسی ها نشان داد. سه شکل پیش رو نحوه این عمل را نشان می دهد. می توان یک سیگنال دیجیتال را انتخاب و تبدیل فوریه آنرا که خود موضوع بحث آینده است، را انجام داد. چیز مهمی که در اینجا باید خاطر نشان نمود این است که در تبدیل فوریه نسخه ای از سیگنال که در حوزه فرکانس است، بدست می آید که شامل لیستی از دامنه و فاز از سینوسی هایی که بطور هارمونیکی به هم وابسته اند. همچنین موضوع انتخاب فرکانس اساسی یا مبنایی موضوعی است که بعداً به آن پرداخته خواهد شد. زمان این نمونه برداری می تواند در تعیین فرکانس اساسی کمک نماید. بنابر این ما فقط به شماره نمونه اکتفا می کنیم بجای زمان بر روی محور x. با توجه به شکل 4.10 سیگنال اولیه را می بینیم که شامل یک سری از نقاط مجزای نمونه برداری است. در زیر این شکل، شکلی از یک سیگنال را می بینید که توسط چند سینوسی تشکیل شده است. توجه کنید چگونه هر یک از این سیگنال ها از این نقاط می گذرند. زمان نمونه برداری هر یک از سیگنال از زمان نمونه های نمونه برداری شده می گذرند. در شکل 4.10 مجموعه ای از سینوسی ها را می بینیم که پیوسته هستند. اما باید دقت کنیم که به این نتیجه گیری که سیگنال آنالوگی ما در تلاش به بازسازی آن توسط چند نمونه برداری مجزا discrete هستیم کاملا مشابه باشند.



دو نمودار 4.11 و 4.12 سینوسی هایی را نشان می دهند که با هم جمع شده تا یک ترکیب را بوجود آورند. نمودار نخست یک سیگنال با فرکانس صفر است. ستاره ها مقادیری از دامنه را نشان می دهند که اگر سینوسی ها با زمان مشابه سیگنال اولیه نمونه برداری می شدند، دیده میشوند. توجه کنید که چگونه تعداد سینوسی ها (8) که از تبدیل فوریه حاصل شدند با تعداد نقاط اولیه منطبق هستند. بعنوان یک تمرین بیایید مقدار دوم ازآخر (یکی مانده به آخری) را تخمین بزنیم. در شکل 4.11 می بینیم که نمونه شماره 6 هر سینوسی دارای مقادیر 4.9، 0.3، 0.1، و 0.7 است. اگر به شکل 4.12 بر گردیم مقادیر 0.4، 0.7، 0.1- و 0.3 را نشان می دهد. اگر همه را با هم جمع کنیم مقداری را نشان می دهد که نمونه مذکور در شکل 4.10 نشان می دهد.





در واقع تنها دلیلی که نمی توانیم به مقدار واقعی برسیم فقط منحصر به چشم ما می شود که نمی تواند مقدار واقعی را در شکل از گراف استخراج نماید. در یک تمرین مشابه می توان به حداکثر مقادیری که برای هر یک سینوسی در شکل های 4.11 و 4.12 وجود دارد، توجه نمود. این حداکثر ها عبارتند از 0.86, 0.76,0.81,0.38,0.81,0.76,o.86,4.88. این مقادیر به دامنه (ak) هر یک از سینوسی ها که برای مشخص کردن و نشان دادن سینوسی لازم است، مرتبط می شود. دیگر اطلاعاتی که (بجز فرکانس اساسی) مورد نیاز است، زاویه فاز است. خواندن این اطلاعات از نمودار قدری مشکل است اما می توان آنرا تخمین زد. با یادآوری این نکته که cos(0)=1 است، توجه می کنیم که مقدار خروجی یک تابع کسینوسی با افزایش زاویه کاهش می یابد تا اینکه به مقدار -1 در زاویه π برسد سپس مقدار آن تا 1 در زاویه 2π افزایش می یابد. این مشاهدات را می توان برای استخراج زاویه فاز مورد استفاده قرار داد. به گراف دوم از شکل 4.11 نگاه کنید. ملاحظه می شود که سینوسی در نمونه 0 مقدار حداکثر را دارد. مقدار حداکثر به مقدار دامنه یا بطور دقیق تر به: a1*cos (2π1f0t+φ1) . بنابر این در این رابطه عبارت کسینوسی مساوی صفر است :2π1f0t+φ1=0:0. در نتیجه φ1=0. حال برای یافتن زاویه فاز دوم $φ_2$ در نمونه 0 (سومین گراف در شکل 4.11) مقدار دامنه 0.1 است. برای یافتن $φ_2$ ، a2*cos (2π1f0t+φ2) در زمان صفر تقریباً مساوی 0.1 است. این به معنی آن است که a2* cos (φ2) ~0.1 . با علم بر اینکه مقدار a2 0.8 است، داریم:

0.8cos(φ2)~0.1

Cos(φ2)~0.1/0.8

Φ2 =~arccos(o.1/0.8)

Φ2~830

φ2~46π/100

اما یک مشکل در اینجا وجود دارد: چگونه می توان دانست که سینوسی در مرحله بعد زمانی افزایش می یابد؟ در این رابطه می دانیم که تابع کسینوسی مقدار مشابهی را در زاویه 2π-θ بر میگرداند بنابر این چه مقدار برای فاز باید مورد استفاده قرار گیرد؟ برای حل این مشکل از حساب جبر کمک می گیریم. ترسیم گرافیکی سینوسی این است که گراف را دنبال می کنیم و می بینیم که آیا مقدار افزایش یا کاهش می یابند. در شکل 4.11 در سومین گراف ملاحظه می شود که مقدار در حال افزایش می یابد. اگر مقدار کمی به زاویه فاز بیافزاییم که نشاندهنده افزایش اندک در زمان است، مقدار بدست آمده از

a2*cos ((a little bit)+φ2) باید مقدار کمی از a2*cos (φ2) بیشتر باشد. برای مقادیر φ2 این رابطه را مورد آزمایش قرار می دهیم:

0.8cos(46π/100)=0.1003

0.8cos(0.01+46π/100)=0.0923

اما این نشان دهنده آنست که سینوسی (برعکس) کاهش می یابد. بنابر این می دانیم که زاویه فاز مقدار منفی زاویه قبلی باشد یعنی: -46π/100 . می توان (و بایستی) کارمان را با روش فوق مجددا چک کنیم:

0.8cos(-46π/100)=0.1003

0.8cos(0.01-46π/100)=0.1082

ملاحظه می کنیم که مقدار بعدی افزایش می یابد یعنی درست آنچیزی که از گراف مشاهده می کنیم. برای بقیه گرافها می توان این مورد را ادامه دهیم. برنامه زیر مفهوم جمع سینوسی ها را نشان می دهد. پارامتری مورد نیاز نیست برای اینکه برنامه خود بطور تصادفی یک سیگنال اولیه می سازد. این برنامه از سیگنال تصادفی x یک نسخه X از حوزه فرکانسی آن می سازد سپس از اطلاعات حوزه فرکانسی موجود برای یک سیگنال اولیه را می سازد (بازیابی). در نهایت این سیگنال را با سیگنال اولیه مقایسه و خطای ناشی از این تبدیل را محاسبه می کند. فرض می کنیم که سیگنال شامل اطلاعات واقعی است. این امر عموماً درست نیست. اما چون ما از سیگنال واقعی شروع کردیم با همین شکل تا پایان ادامه می دهیم. چون ما اطلاعات را تغییر نمی دهیم فقط آنرا از حوزه زمان به حوزه فرکانس تغییر می دهیم و سپس به حالت اولیه بر می گردانیم. نتایج تبدیل فوریه در یک سیگنال مختلط است حتی اگر شکل اولیه آن حقیقی باشد. برای حل این مشکل بگونه ای رفتار می کنیم که سیگنال اولیه حقیقی مثل سیگنال مختلط بوده ولی بخش مجازی آن صفراست. برنامه تبدیل معکوس را تا حدی انجام می دهد اما بخش مجازی آنرا در نظر نمی گیرد. به عبارت دیگر، برنامه با طوری با سینوسی رفتار می کند که آن بشکل

Xmag [k]cos(2πkf0 t+Xangle [k] است در حالیکه در واقع هر سینوسی (با توجه به فرمول اولر)

[k]cos(2πkf0t+Xangle [k])+j Xmag [k]sin(2πkf0t+Xangle [k])

ما قادریم از این ساده سازی خلاص شویم چون می دانیم که در نهایت بخش مجازی jsin(o) همدیگر را حذف می کنند.

طیفspectrum

ترسیم یک سیگنال بر حسب فرکانس را طیف می گویند. طیف نموداری از فرکانس های تشکیل دهنده یک سیگنال است که با دامنه هایشان مرتبط هستند. در یک نمودار جداگانه، برای هر یک از سینوسی فازها نشان داده می شود. قبلاً در شکل 4.10 دیدیم که یک سیگنال دیجیتال می تواند توسط با مجموع چند سینوسی دیگر نشان داده شود. این سینوسی ها آنگونه که در شکل های 4.11 و 4.12 نشان داده شده اند، می توانند بطور منفرد ترسیم شوند. بعضی از اوقات اطلاعات سینوسی بینش ما را در باره سیگنال عمیق تر می کند. بعنوان مثال اگر سیگنال اولیه یک موسیقی باشد، بنابر این سینوسی ها به ما در مورد وسایل مورد استفاده در ایجاد این قطعه موسیقیایی اطلاعات می دهند. اگر این موسیقی صدای یک شخص باشد بصورت بم و غیر اصلی و یا یک صوت قطار، امکان حذف آن وجود داشته باشد. ترسیم اطلاعات ارائه شده به ما توسط سینوسی ها در یک طیف نتیجه می شود. با ادامه مثالهایی که در بخش قبلی داشتیم در می یابیم که بخشی از اطلاعات در فرکانس اصلی گنجانده شده که ما موکول به آینده نمودیم. همانگونه که قبلاً بیان کردیم، نیاز داریم تا بدانیم چه مدت زمان بین نمونه ها باید باشد تا فرکانس ها را شناسایی کنیم. اما ما این را به بهترین وجهی بدون این نگرانی انجام خواهیم داد. ملاحظه می کنیم که سینوسی ها مقدار صحیح را در یک گراف از فرکانس (بسامد) صفر تا هفت تکرار می کنند. یعنی سینوسی ها با فرکانسی که شماره 8 نشان می دهد معادل نمونه 0 است. هر چه تعداد بسامد بیشتر می شود، فرکانس مربوطه نیز بیشتر می شود. فرض کنید سیگنال اصلی یا اولیه با فاصله 125 میلی ثانیه نمونه برداری شود، بنابراین هر نمونه با فاصله زمانی 125 میلی ثانیه نسبت به نمونه قبل خود قرار دارد. در گراف دوم در شکل 4.11، تکرار سینوسی ها را در هر 1000 میلی ثانیه ملاحظه می کنیم(8*125=1000). بنابر این فرکانس آن یک هرتز است. تا آخرین نمودار از شکل 4.12 شامل 7 تا قله و هشت نمونه که شامل 7 هرتز است. برای بقیه این فصل فرض می کنیم که فرکانس اساسی 1 هرتز است. در شکل 4.13 مقادیر دامنه و اختلاف فاز را برای هر یک از سینوسی ها ترسیم نمودیم:

بزرگای فرکانسی (دامنه) و فاز فرکانسی

باید الگوی طیف را ملاحظه نمود: برای بزرگای فرکانسی تصویر تقارنی با مرکزیت 4 هرتز را می بینیم. روی نمودار فاز الگو روی 4 هرتز متمرکز است ولی در اینجا علاوه بر تقارن، وارونگی را نیز می توان دید. تاکنون دیده ایم چگونه می توان از فاز و بزرگا با استفاده از تبدیل فوریه طیف بسازیم. برای پیدا کردن یک سیگنال فشرده نظیر آنچیزی که در مدل ریاضی می بینیم، از فرمول اولر معکوس استفاده می کنیم. در اینجا فقط مقدمه ای بر این روش ذکر می شود و بحث مفصل آن در فصل آینده خواهد آمد. بعنوان مثال برای ترسیم نمودار طیف x(t)=2+2cos(2π(200)t) آنرا در قالب تابع کسینوسی می نویسیم:

x(t)=2+2cos(2π(200)t)=2cos(0)+2cos(2π200t)

بعد با استفاده از فرمول اولر معکوس:

Cos(φ)=(ejφ+e-jφ)/2

توجه کنید که چگونه این منجر به دو مولفه فرکانسی می شود: یکی در فرکانس مشخص و دیگری فرکانس منفی. برای طیف، دال بر این است که محدوده ای از فرکانس که از آنچه که تا بحال از مثال قبلی دیدیم متفاوت است. در واقع اطلاعات مشابه است ولی تنها طیف ترسیمی متفاوت ولی با مقادیر فرکانسی معادل است. این مورد در فصل " تبدیل فوریه " تشریح می شود.

x(t)=2(ej0 +e-j0)/2+2(ej2π(200)t +e-j2π(200)t)/2

=2(1+1)/2+(2/2)(ej2π(200)t +e-j2π(200)t)

$=2+e^{j2\pi(200)t}+e^{-j2\pi(200)t}$

این منجربه بزرگای 2 در فرکانس 0 هرتز و بزرگای 1 در فرکانس 200 هرتز و -200 هرتز می شود.

نمونه برداری sampling

در مبحث اول دراین درس بیان شد که ما در دنیای آنالوگ بسر می بریم. کامپیوترهای ما وسایل دیجیتال هستند به این معنی آنها تنها اطلاعات یا داده های محدود و مجزا هم از نظر دقت و هم از نظر بزرگا را می توانند ذخیره نمایند. چگونه می توان از نبود بین دنیای واقعی و دنیای مصنوعی کامپیوتر ارتباط برقرار نمود؟ این فصل درصدد پاسخگویی به این سوال اساسی است. نمونه برداری، فرایند بدست آوردن سیگنال دیجیتال از یک نمونه آنالوگ آن است. وقتی نمونه برداری از یک سیگنال را انجام می دهیم لحظه به لحظه یک مقدار ضبط می کنیم. نمونه برداری ممکن است منجر به مقادیر زیادی داده شود.

بعنوان مثال یک نمونه 3 دقیقه ای از یک صدا که 16 بیت و در فرکانس 44100 Hz برای دو کانال نمونه برداری می شود می تواند به مقدار زیر اطلاعات ضبط کند:

3 minutes*(16bit/sample)*(44100sample/second)*2

که با تغییر واحد داریم:

3minutes 60sec/1minutes*(16bit/sample*1byte/8bits)*(44100sample/second)*2=180*2(bytes)*44100*2=31,752,000~30MB

اما آیا به ذخیره 30MB داده برای یک صدای 3 دقیقه ای نیاز داریم. در واقع راههای دیگری برای ذخیره این داده وجود دارد. فرمت MP3 یک از این راه حل ها است. در این روش صدای MP3 با صدای واقعی نزدیک است در حالیکه فقط 10٪ فضای فوق را دارد. تبدیل یک سیگنال آنالوگ به دیجیتال برای یک کامپیوتر در تحلیل یک سیگنال یک مرحله اساسی و مورد نیاز است: کامپیوترهای جدید ماشین های دیجیتالی هستند و فقط مقادیر دیجیتال را ذخیره می کند. در تابع پیوسته x(t) ، t (متغیر پیوسته) را با nTs (مقدار مجزا) جایگزین می کنیم. n اندیس آرایه مجزا و Ts دوره نمونه برداری یعنی زمان بین دو نمونه برداری است. زمان نمونه برداری عکس فرکانس نمونه برداری است: یعنی Ts =1/fs .

نمونه برداری sampling (پهنای باند)

قبل از اینکه در باره نمونه برداری صحبت کنیم بیایید تا محدوده فرکانسی سیگنال که علاقمند به آن هستیم را "پهنای باند" یا bandwidth بخوانیم. بعضی اوقات بسادگی بالا ترین فرکانس را بعنوان پهنای باند می خوانیم به این معنی که ما به محدوده فرکانسی صفر تا بالاترین فرکانس علاقمندیم. در دیگر اوقات ممکن است محدود شود یعنی "طیف سفید مرئی"white visible spectrum

نمونه برداری sampling "فرکانس نایکوئست"

اصطلاح نمونه برداری بحرانی به حالتی دلالت می کند که نرخ نمونه برداری دقیقاً دو برابر پهنای باند است یعنی fs=2B. ما به این مقدار توسط معیار فرکانس نایکوئست Nyquist's Frequency که در آن fs≥2B است، می رسیم .

نمونه برداری sampling (بیش نمونه برداری)

بیش نمونه برداری یا oversampling به نمونه برداری با فرکانس بیشتر از مقدار مورد نیاز یا بیشتر از دوبرابر پهنای باند گفته می شود. اگر محدودیت سخت افزاری دارید یا اگر وقت لازم برای پردازش ندارید و یا اگر حافظه به مقدار لازم را ندارید، بیش نمونه برداری مشکل ایجاد می کند. بر عکس اگر به سخت افزار ارزان دسترسی دارید یا مبدل دیجیتال به آنالوگ ارزان قیمت را در اختیار دارید ،نظیر پخش CD ، بیش نمونه برداری مانعی ندارد. یک مبدل

دیجیتال به آنالوگ ارزان ممکن است از یک تابع پله ای ساده برای ایجاد یک سیگنال استفاده نماید. بنابر این با نمونه بیشتر سیگنال را بهتر تخمین می زند. قرائت یا ذخیره درجه حرارت در هر هزارم ثانیه حجم زیادی از اطلاعات را موجب می شود که استفاده ای در پیش بینی درجه حرارت فردا نخواهد داشت.

نمونه برداری sampling (undersampling)

نمونه برداری undersampling وقتی اتفاق می افتد که نمونه برداری کافی نباشد یعنی کمتر از دوبرابر پهنای باند. استفاده از این نوع نمونه برداری مانع بازسازی سیگنال می شود و باعث ایجاد سیگنال غلط می شود چیزی که در برنامه نویسی کامپیوتر معروف به garbage in garbage out (ورودی آشغال خروجی آشغال) است.

نمونه برداری (دوره تناوب و فرکانس سیگنال)

x[n]=x(nTs) فرایند نمونه برداری را توصیف می کند. Ts زمان نمونه برداری در حالیکه n اندیس است. x[n] دقیقاً x(t) نیست چون X(t) تابع پیوسته است. در بهترین حالت x[n] می توان تخمین خوبی از x(t) باشد. دوره تناوب یک سیگنال مدت زمانی است که آن سیگنال تکرار شود. برای یک سینوسی تنها به فرکانس f نیاز داریم چون تناوب که معادل 1/f است را تعیین می کنیم.

مثال: اگر از x(t) با نمونه برداری 20 کیلو هرتز استفاده کرده باشیم، چند نمونه بعد از 60 میلی ثانیه داریم؟

$X(t) = 4\cos(2\pi 404t + \pi/4) + 2\cos(2\pi 6510t) + \cos(2\pi 660t - \pi/5) +$

حل: نخستین چیزی که باید به آن توجه کنید این است که در واقع تعداد نمونه ها به خود سیگنال وابسته نیست. به عبارت دیگر همه نیاز ما برای پاسخ به این سوال نرخ نمونه برداری 20 کیلو هرتز و زمان 60 میلی ثانیه ست. دوره تناوب 1/20kHz یا 1/20000 Hz است. چون هرتز سیکل بر ثانیه است عکس آن ثانیه بر سیکل یا همان ثانیه است. نخستین نمونه را در زمان صفر می گیریم بعد 20000/1 ثانیه منتظر می مانیم سپس نمونه بعدی تا 60 میلی ثانیه (یعنی 0.060 ثانیه) برای جواب باید 20000 را در 0.06 ضرب کنیم باضافه 1 چون در زمان صفر اولین نمونه برداری انجام می شود. یعنی 1201 نمونه.

مثال: اگر $X(t) = 4\cos(2\pi 250t + 2\pi/7)$ باشد، زمان تناوب سینوسی چقدر است؟

تنها چیزی که از معادله فوق نیاز داریم، فرکانس 250 هرتز است. $T=1/f=1/250=0.004sec$ فرض کنید یک سیگنال مختلط داریم که مرکب هستند از چندین سینوسی هایی که بطور هارمونیک به هم وابسته اند. برای پیدا کردن دوره تناوب باید بزرگترین فرکانس f0 طوری که بقیه فرکانس ها ضریب عدد صحیح آن باشند، پیدا کنیم. با تعداد دو یا بیشتر از سینوسی، فرکانس مبنا با تابع بزرگترین مقسوم علیه مشترک بدست می آید. برای سه فرکانس f1 ، f2 و f3 ، f0 را طوری پیدا می کنیم که $f1=k1*f0$ ، $f2=k2*f0$ و $f3=k3*f_0$. متغیرهای k1 ، k2 و k3 باید صحیح باشند.

بازسازی Reconstruction

فرایندی که در پی آن از یک سیگنال دیجیتال مجدداً سیگنال آنالوگ بدست می آید را بازسازی گویند. این تغییر از دیجیتال به آنالوگ شامل درون یابی و یا پر کردن اطلاعات از دست رفته بین نمونه ها است. روشهای مختلفی برای انجام این کار وجود دارد که روشن ترین آن روش خطی ارتباط بین نقاط شامل وصل نمودن دو نقطه مجاور توسط خط راست است. مبدل آنالوگ به دیجیتال، که بطور مختصر ADC نوشته می شود، از یک سیگنال بطور پیوسته نمونه برداری نموده و اطلاعات گسسته discrete از آن تهیه می کند. مخالف ADC ، DAC یا مبدل دیجیتال به آنالوگ است. وقتی یک مبدل دیجیتال به آنالوگ از یک نمونه گسسته، نمونه پیوسته می سازد، آنرا درون یابی نامیم. یک روش آسان برای انجام این کار استفاده از نگهدارنده صفر "Zero Order Hold" برای یک تابع ضربه است. در این روش یک تابع ضربه طوری ایجاد می شود که زمان بین نمونه ها توسط نمونه نخست از دو نمونه مجاور پر می شود. درون

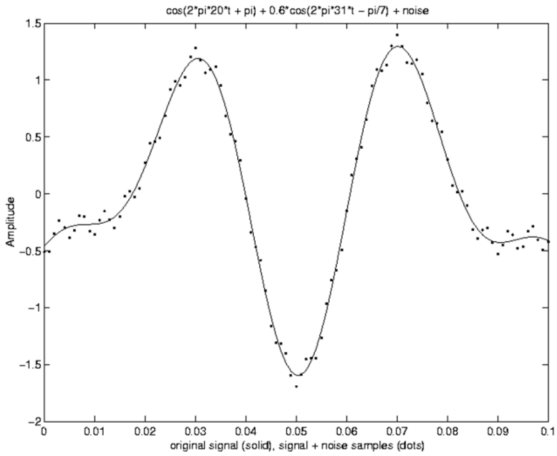
یابی خطی انتخاب (انتخاب) دیگری برای درون یابی است که در آن تابع ضربه وابسته به متغیر زمان است. بعنوان مثال تابع مثلث است جاییکه رمپ ضربه تا نقطه وسط بالا آمده سپس پایین می رود. معادله زیر فرایند مبدل DAC را توصیف می کند:

$$x(t)=\sum_{n=-\infty}^{\infty}x[n]p(t-nT_s)$$

در این رابطه p(t) تابع ضربه است : نسخه ای از تابع ضربه واحد. در واقع معادله مشابه با تابع ضربه را برای نمونه برداری می توانیم استفاده نماییم . اندیس در واقع از منفی تا مثبت بینهایت تمایل پیدا نمی کند.

نمونه برداری و نوفه فرکانس بالا

در هنگام نمونه برداری از یک سیگنال با نوفه فرکانس بالا اثرات نوفه حتی وقتی که فرکانس نمونه برداری از فرکانس نوفه کمتر است را خواهیم دید. دلیل آن این است که نوفه هر نمونه را قدری بزرگتر یا کوچکتر از آنچه که هست می سازد. شکل 5.1 این مورد را نشان می دهد. نمونه های یک سیگنال با بعضی از نوفه فرکانس بالا نشان داده شده است که همراه با سیگنال اولیه بدون نوفه با خط پر است. در فصل آینده "دگرنامی" را که محصول نمونه برداری از یک سیگنال با نوفه است را مرور می کنیم.



دگر نامی Aliasing

مشکل متداولی که ممکن است در حین نمونه برداری از یک سیگنال پیوسته رخ دهد، دگرنامی است جاییکه یک سیگنال نمونه برداری شده شامل نسخه هایی از مولفه های سینوسی خودش است که می تواند با دیگر مولفه ها تداخل داشته باشد. دگرنامی وقتی اتفاق می افتد که از یک سیگنال تناوبی نظیر یک سینوسی یا یک سیگنال که از سینوسی ها تشکیل شده مانند a*cos(2πft+φ) نمونه برداری می شود. وقتی از این سیگنال نمونه برداری می شود، آنرا با نرخ fs انجام می دهیم. هر سینوسی دیگر با فرکانس بالاتر از fs در سیگنال ظاهر می شود چنانکه سیگنال دارای فرکانس کمتر از fs هم ظاهر می شود. بعنوان مثال فرض کنید یک سیگنال سینوسی با معادله x(t)=cos(2π800t) داریم وقتی نمونه برداری می شود، بجای t در سیگنال پیوسته x(t) ، nTs که شامل نقاط زمانی با گامهای منظم است جایگزین می کنیم یعنی: x[n]=x(nTs) فرض کنید که نمونه برداری از این سینوسی با فرکانس 800 هرتز، f_s = 600 نمونه بر ثانیه باشد. وقتی محتوی فرکانسی سیگنال نمونه برداری شده را مورد آزمایش قرار می دهیم، ملاحظه می کنیم سیگنال مذکور یک سینوسی 200 Hz را نشان می دهد. بجای سینوسی 800 هرتز حالا یک سینوسی 200 هرتز داریم : چه اتفاقی افتاده است؟ لذا نخست بیایید (بیایید) ببینیم سیگنال نمونه برداری شده چیست؟

x(t)=cos(2π800t)

X[n]=cos(2π800nTs)

X[n]=cos(2π(200+600)nTs)

X[n]=cos(2π200nTs+2π600nTs)

و آنرا در معادله جایگزین می کنیم: $Ts =1/600$ بنابر $Ts= 1/fs$

$X[n]=\cos(2\pi 200nTs+2\pi 600n(1/600))$

$X[n]=\cos(2\pi 200nTs+2\pi n)$

تابع کسینوسی تابع تناوبی با دوره تناوب 2π است یعنی $\cos(\theta+2\pi)=\cos\theta$. چون می توانیم 2π را به تابع کسینوسی اضافه و مقدار مشابه را بدست آوریم، می توانیم مقادیر 2π برابر هر عدد صحیحی (حتی منفی) را با آن جمع و نتیجه مشابه را رقم بزنیم. n اندیس عدد است بنابر این $2\pi n$ را می توان حذف نموده و چیزی که برای ما می ماند $\cos(2\pi 200nTs)$ است. وقتی از یک سینوسی با فرکانس 800 هرتز در 600 samples/sec نمونه برداری می شود، نمی توانیم آنرا از سینوسی 200 هرتز تمیز قرار بدهیم. اگر یک مولفه 200 هرتزی نیز در سیگنال داشتیم، مولفه 800 هرتزی با آن تداخل می نمود.

دگر نامی (مثالی از دنیای واقعی)

دگر نامی وقتی که چرخهای ماشین را روی تلویزیون می بینیم اتفاق می افتد در این لحظه فکر می کنیم چرخها بطورعکس حرکت می کنند. یک فیلم تلویزیونی ترکیبی از عکسها و تصاویر است که بطور منظم نمونه برداری شده و روی صفحه ظاهر می شود. اگر چرخها کمتر از یک سیکل بین تصاویر بطور کامل بچرخند بنظر می رسند بطور آهسته در خلاف جهت حرکت می کنند. تعمیم عمومی موضوع این است که تصور کنید که فرکانس مولفه سینوسی توسط فرکانس اساسی باضافه ضریب عدد صحیح از فرکانس نمونه برداری جایگزین شود یعنی $f0 + kfs$ بجای f.

$X(t)=a*\cos(2\pi(f0+ kfs)+\varphi)$

$X(t)=a*\cos(2\pi f0t+ 2\pi kfst)+\varphi)$

اکنون از این سیگنال با فاصله داری TS نمونه برداری می کنیم. توجه نمایید که هیچ مشکلی برای سینوسی تا وقتی که نمونه برداری می نماییم ایجاد نمی شود.

$X[n]=x(nTs)=a*\cos(2\pi f0nTs+ 2\pi kfsnTs+\varphi)$

چون $Ts = 1/ fs$ می توان $fsTs$ را حذف نمود.

$X[n]=x(nTs)=a*\cos(2\pi f0nTs+ 2\pi kn+\varphi)$

هم k و هم n عدد صحیح هستند بنابراین ضرب آنها نیز عدد صحیح است. این به معنی آنست که:

$a*\cos(2\pi f0nTs+ 2\pi kn+\varphi)= a*\cos(2\pi f0nTs+\varphi)$

به عبارت دیگر وقتی از یک سیگنال نمونه برداری می کنیم، نمی توانیم یک سینوسی در آن (نظیر 800 هرتز در مثال بالا) از دیگر سینوسی (مثلا 200 هرتز در همان مثال) که شامل یک ضریب عدد صحیح از فرکانس نمونه برداری است (در اینجا 600 هرتز) را تمیز دهیم. معنی آن این است که نمونه برداری با فرکانس fs باید حداقل بزرگتر از بزرگترین مولفه فرکانسی باشد با فرض اینکه ما محتوی طیفی داریم که با فرکانس صفر شروع می شود. در آینده موارد دیگری از محدودیت های موجود در fs را ذکر خواهیم کرد. دگرنامی یعنی اگر ما به طیف یک سینوسی f Hz نگاه کنیم یک مولفه در f موجود خواهد بود اما مولفه های دیگری نیز در فرکانس (f+1 fs) ، سپس در (f+2fs) و بعد در (f+3fs) و ..

مثالی از دگرنامی

در برنامه زیر دگرنامی نشان داده می شود. نمونه برداری را با تعداد 500 نمونه در ثانیه بر روی دو سینوسی: $2\cos(2\pi 100t + \pi/3)$ و $2\cos(2\pi 600t + \pi/3)$ را نشان می دهیم. هر دو دارا ی فاز و بزرگای مشابه هستند چون سیگنال نمونه برداری شده یکی است. از آنجاییکه فرکانس نمونه برداری 500 است، بنابر این در هر 1/500=0.002 ثانیه یک نمونه برداری انجام می شود.

تاخوردگی folding

تاکنون دیدیم که اگر فرکانس نمونه برداری کمتر از فرکانس یک سیگنال باشد، نتیجه آن سیگنالی است با یک فرکانس کم . تاخوردگی نگاهی دیگر به پدیده دگرنامی جاییکه فرکانس مشاهده شده ا زیک سیگنال نمونه برداری شده، از فرکانس واقعی متفاوت باشد حتی اگر از فرکانس نمونه برداری کمتر اما از نصف همین مقدار بیشتر باشد. بسیاری از متون واژه تاخوردگی را معادل دگرنامی می دانند اما از آنجاییکه در این پدیده مقادیر فاز متاثر می شوند، بنابر این تمایل داریم تا در اینجا از این واژه بطور مستقل یاد کنیم. ممکن است fs/2 به فرکانس تاخوردگی نسبت داده شود، در صورتی که یک سینوسی 600 هرتزی در fs= 1000 (نمونه در ثانیه) نمونه برداری شده و فرکانس مشاهده شده 400=600-1000 باشد. این موضوع از اینجا نتیجه می شود که:

$$\cos(2\pi 600nTs+\varphi)=\cos(2\pi(1000-400)nTs+\varphi)=\cos(2\pi(1000nTs-2\pi 400nTs+\varphi)=\cos(2\pi(1000n/1000-\pi 400nTs+\varphi)$$

$$=\cos(2\pi n-2\pi 400nTs+\varphi)=\cos(-2\pi 400nTs+\varphi)=\cos(2\pi 400nTs-\varphi)\ [\cos(-\theta)=\cos(\theta):\text{یادآوری}]$$

فرکانسی که الان ملاحظه می کنیم باید 400 هرتز باشد. اما زاویه فاز آن منفی φ است. بنابر این اگر مترصد ترسیم پاسخ فرکانسی بزرگا برای سیگنال $x(t)=\cos(2\pi 600n/1000+\varphi)$ باشیم یک فرکانس 400 هرتز را ملاحظه خواهیم نمود. اگر زوایای فاز را ترسیم کنیم خود را بصورت φ- نشان خواهد داد. کد زیر نشان می دهد که تابع کسینوسی مقادیر منفی یا مثبت باشند نتایج مشابهی را ارئه خواهند نمود. شکل 5.3 نتیجه اجرای این کد را نشان می دهد. در این شکل $x(t)=\cos(-2\pi 10t-\pi/3)$ در سمت چپ و $x(t)=\cos(-2\pi 10t+\pi/3)$ در سمت راست نشان داده شده است. تاخوردگی زمانی اتفاق می افتد که سیگنال نمونه برداری شده یک مولفه سینوسی بزرگتر از fs/2 اما کمتر از f_s داشته باشد. در کل می توان بعضی از فرکانس f را با (fs-f0) جایگزین نماییم.

$$\cos(2\pi fnTs+\varphi)=\cos(2\pi(fs-f0)nTs+\varphi)=\cos(2\pi fsnTs-2\pi f0nTs+\varphi)=\cos(2\pi n-2\pi f0nTs+\varphi)=\cos(-2\pi f0nTs+\varphi)=\cos(2\pi f0nTs-\varphi)$$

در این رابطه مقدار f0 باید کمتر از مقدار اولیه f باشد در غیر اینصورت فرکانس تاخورده خارج از طیف خواهد بود. بعنوان مثال اگر با $\cos(2\pi 400nTs-\varphi)$ با نمونه برداری Ts=1000Hz شروع کنیم، و 400 را با 600-1000 جایگزین نماییم، نتیجه $\cos(2\pi 600nTs-\varphi)$ خواهد بود. این مولفه طیف (-fs/2 to fs/2) را نشان می دهد بجز $\cos(2\pi 400nTs-\varphi)$. مولفه ای که از آن شروع کردیم.

مکانهای نسخه برداری بعد از نمونه برداری

زمانی که سیگنال نمونه برداری می شود، تعداد بینهایت از نسخه "replication" مولفه های فرکانسی در فواصل منظم ظاهر می شوند. این موضوع بر مبنای تناوب سینوسی ها است که در واقع از $\cos(2\pi ft)= \cos(2\pi ft+2\pi n)$ ناشی می شود. این به معنی آن است که نمونه برداری یک سیگنال همهٔ محتوی فرکانسی از صفر تا fs را شامل می شود.

چون n و k هر دو عدد صحیح هستند، بنابر این ضرب آنها نیز عدد صحیح است.

$$\cos(2\pi f1\ nTs)= \cos(2\pi f1nTs\ t+2\pi nk)$$

$$= \cos(2\pi n(f1\ /fs+(fs\ /fs)k)$$

بنابر این $\cos(2\pi f1nTs)$ از $\cos(2\pi(f1+kfs)nTs)$ برای همه اعداد صحیح k قابل تشخیص نیست. بنابر این وقتی نمونه برداری با fs نمونه برداری در ثانیه انجام می شود، $\cos(2\pi f1t\ )$ از $\cos(2\pi(f1\ +kfs)t)$ قابل تشخیص نیست به این معنی که نمونه برداری نسخه ای از فرکانس در هر ضریب عدد صحیح باضافه همان فرکانس را بوجود می آورد. بعداً در باره مقادیر منفی k صحبت بعمل می آید. تابع کسینوس یک تابع زوج است به این معنی که

$\cos(-\theta)=\cos(\theta)$ یا $\cos(-2\pi f1t)= \cos(2\pi f1t)$است

این به معنی آنست که تمام محتوی فرکانسی روی طیف بین 0 تا φp+ بین 0 تا φp- آینه می شود. بنابر این یک فرکانس f1Hz بصورت f1Hz- نیز ظاهر می شود. با توجه به مبحث دگرنامی، می بینیم که این نسخه برداری نیز در بین fs/2 و fs ظاهر می شود. فرض کنید k=-1 باشد :

$\cos(2\pi f1nTs)= \cos(2\pi n(f1 /fs-(fs /fs)k)$

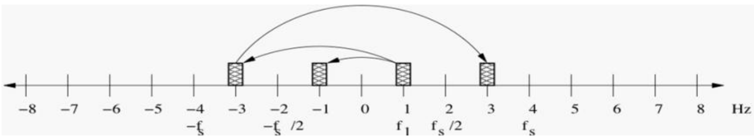
بنابر این $\cos(2\pi f1nTs)$ بصورت $\cos(2\pi (f1-fs) nTs)$ هم ظاهر می شود.

مثال 1 از نسخه برداری

فرض کنید f1=1 Hz، و fs=4 نمونه بر ثانیه باشد،

$\cos(2\pi (1) nTs) = \cos(2\pi n (1-4) Ts) = \cos(2\pi n (-3) Ts) = \cos(2\pi n (3) Ts)$

شکل 5.4 را ملاحظه نمایید : یک نسخه در 1Hz- نیز داریم. در اینجا از بینهایت نسخه ممکن چند تا را نشان دادیم گرچه نسخه هایی در ±5 و ±7 نیز وجود دارند

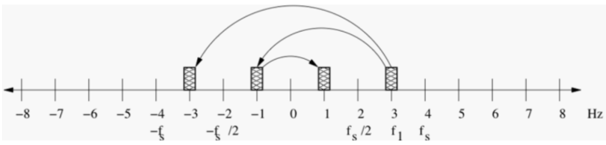


مثال 2 از نسخه برداری

اگر f1 > fs/2 باشد چه اتفاقی می افتد: با k=-1 فرض می کنیم f1 = 3 Hz و fs = 4 sample/sec :

$\cos(2\pi (3-4) nTs) = \cos(2\pi (-1) nTs) = \cos(2\pi nTs)$

شکل 5.6 را ملاحظه نمایید که در 3Hz- نیز یک نسخه دیگر دارد.



نرخ نایکوئست (مفهوم پهنای باند)

در بخش های قبل از همین فصل تشریح نمودیم که برای گریز از دگرنامی باید نرخ نمونه برداری fs از بزرگترین مولفه فرکانسی موجود در سیگنال بیشتر باشد. همچنین نشان دادیم که چگونه این بزرگترین f بایستی از نصف نرخ نمونه برداری fs کمتر باشد تا از تاخوردگی جلوگیری شود. در بخش بعدی از این مبحث نشان خواهیم داد که پهنای باند عبارت مهمی است: تا این لحظه ما فرض کرده ایم که به تمامی فرکانس از 0 تا بزرگترین مولفه فرکانسی علاقمند هستیم. یعنی پهنای باند همان بزرگترین مولفه است. در واقع دقیق تر است بگوییم حداقل فرکانس نمونه برداری بر مبنای پهنای باند است یعنی: fs ≥ 2 (پهنای باند) ما این حداقل نرخ نمونه برداری را به نرخ نایکوئست نامگذاری می کنیم. نام دیگر این عبارت در بعضی از کتب به نام نمونه برداری shannon نیز عنوان می شود. برای پیدا کردن این نرخ، پهنای باند را تعیین کنید (که اغلب به عنوان بزرگترین فرکانس موجود در سیگنال تلقی می شود) و مقدار آنرا در 2 ضرب کنید. این روند از این جهت مهم است که از دگرنامی که بعنوان یکی از پیامدهای نمونه برداری است جلوگیری می نماید.

مثال از نرخ نایکوئست

مطلوبست حداقل نرخ نمونه برداری را برای سیگنال :

$X(t)= \cos(2\pi 700t-5\pi/2)+ 3\cos(2\pi 450t-5\pi/2)+ \cos(2\pi 630t+2\pi/5)$

فرض کنید تمایل ما تمامی فرکانس است که از 0 شروع می شود. جواب: سینوسی ها را در قالب $a_k \times \cos(2\pi f_k t+\varphi_k)$ قرار دهید. فرکانس (f_k) در x(t) ، 700، 450، و 630 هرتز هستند. بزرگترین آن 700 است از فرض مطرح شده در سوال این استفاده می شود که بازه یا پهنای باند فرکانس از 0 تا 700 است یعنی 700-0=700Hz.

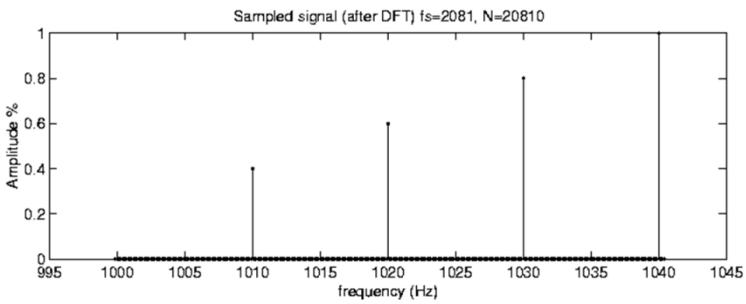
Nyquistrate=2(bandwidth)=2(max(700Hz, 450Hz, 630 Hz)=2(700Hz)=1400Hz

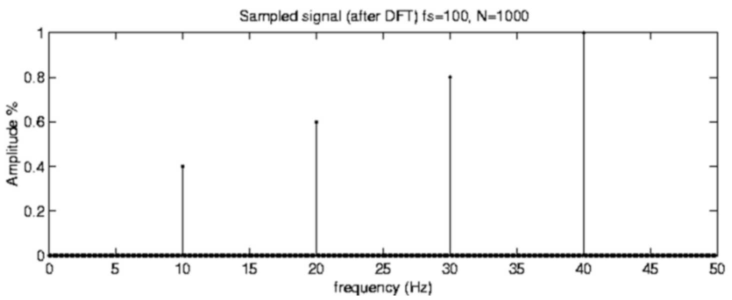
Bandpass نمونه برداری باندگذر

این امکان وجود دارد تا از یک سیگنال در یک نرخ کمتر از دو برابر بیشترین فرکانس نمونه برداری شود: اگر سیگنال، پهنای باند محدودی داشته باشد. این مورد به نمونه برداری باندگذر موسوم است. بعنوان مثال فرض کنید سیگنالی با ترکیبی از چهار سینوسی با فرکانس های 1010، 1020، 1030، و 1040 هرتز وجود داشته باشد:

$$X(t)= 4\cos(2\pi 1010t)+ 6\cos(2\pi 1020t)+ 8\cos(2\pi 1030t) + 10\cos(2\pi 1040t)$$

شکل 5.9 طیف سیگنال نمونه برداری شده را نشان می دهد. تعداد نمونه های شبیه سازی شده طوری انتخاب می شود که ضریبی از fs باشد که در این وضعیت به این معنی است که تحلیل فرکانسی DFT دقیقا با فرکانس موجود در سیگنال منطبق است. به عبارت دیگر نشتی DFT "DFT leakage" وجود ندارد. بنابر این در این حالت شکل مطلوب تر بنظر می رسد. همچنین در این شکل فقط بخشی از طیف نشان داده شده است. دامنه کمتر از 1000 هرتز در شکل 5.9 همه صفر هستند. نکته آخر در مورد این شکل این است که یک فرکانس نمونه برداری شبیه سازی شده از 2081 sample/sec. انتخاب میشود چون این فرکانس یک سیگنال مطلوبی را ارائه می کند. اگر 2080 انتخاب می شد (بعنوان فرکانس نمونه برداری) مولفه فرکانسی که در 1040 هرتز ظاهر می شد دامنه دوبرابر می داشت. این امکان وجود داشت که از x(t) در 2080 Hz یا بیشتر نمونه برداری می شد (آنطور که در شکل 5.9 دیده می شود) ولی اگر نمونه برداری با فرکانس 100 هرتز انجام می شد چه اتفاقی می افتاد؟ در این وضعیت یک طیفی مشابه طیف نشان داده شده در شکل 5.10 می داشتیم. آنطور که از شکل استنباط می شود، اطلاعات مشابه شکل قبل در این وضعیت نیز دیده می شود. مزیتی که در نمونه برداری با نرخ کمتر وجود دارد چیزی نظیر استفاده از وسایل ارزانتر در انجام نمونه برداری است. اما چرا این کار انجام می شود؟





جواب به این سوال مشابه چیزی است که که در دگرنامی انجام شده است. وقتی یک سیگنال نمونه برداری می شود (رقمی می شود) مقادیر بطور تناوبی از سیگنال آنالوگ خوانده می شوند. مشخصاً ما بجای t از n/fs در بیان ریاضی سیگنال آنالوگ استفاده می کنیم. بنظر نمی رسد از سیگنال آنالوگ نمونه ریاضی داشته باشیم. بالاخره، اگر می دانستیم سیگنال آنالوگ چه بوده چرا برای نمونه برداری از آن به دردسر می افتادیم؟ در هر حال هر سیگنال آنالوگی بصورت جمع سینوسی ها نشان داده می شود لذا این فرض که هر سیگنال آنالوگ دارای عبارت ریاضی است معقول بنظر می رسد. این عبارت (برای هر

سینوسی) را در قالب $a\times\cos(2\pi ft+\varphi)$ می نویسیم. وقتی از سیگنال نمونه برداری می شود، این مولفه به $a\times\cos(2\pi fn/fs+\varphi)$ تبدیل می شود. می دانیم که $\cos(2\pi k+\varphi)=\cos(\varphi)$ تا وقتی که k عدد صحیح است و البته چیزی عجیب نیست چون تابع مذکور تناوبی بوده و به بارها مثل هم تکرار می شوند. در واقع این تابع بینهایت بار در جهت مثبت و منفی تکرار می شود (به عبارت دیگر k عدد صحیح مثبت یا منفی است). بنابر این یک مولفه از یک سیگنال آنالوگ بارها در زمان مشخص و به تعداد کافی تکرار می شود. وقتی fs=100 Hz باشد، اولین مولفه برابر است با:

$$4\cos(2\pi fn/fs)=4\cos(2\pi 1010n/100)=4\cos(2\pi(1000+10)n/100)=4\cos(2\pi 1000n/100+2\pi 10n/100)=4\cos(2\pi 10n+2\pi n/10)$$

چون n عدد صحیح است (یعنی فرضاً k=10n) مقدار فوق به مقدار زیر کاهش می یابد:

$$= 4\cos(2\pi n/10)$$

به عبارت دیگر، مولفه 1010Hz خود را بصورت مولفه 10Hz نشان می دهد. دانشجویان می توانند مولفه های دیگر که بصورت 20، 30، 40 هرتز هستند هنگامی که نمونه برداری با نرخ فوق انجام می شود، را بدست آورند. همچنین دانشجویان می توانند عملیات فوق را برای شرایطی که اختلاف فاز نیز داشته باشیم یعنی مثلاً $4\cos(2\pi 1010n/fs+\pi/4)$ ، بدست آورند. وقتی فرکانس ها با هم مغایرتی نداشته باشند، این مثال کاربرد دارد. اما اگر یک فرکانس 10 هرتز هم وجود می داشت چه اتفاقی می افتاد؟ در این حالت اطلاعات گم می شده است چون هم مولفه 10 هرتز و هم 1010 هرتز بصورت یک مولفه در فرکانس 10 هرتز ظاهر می شدند. برای احتراز از همپوشانی از نمونه برداری پهنای باند استفاده می شود. همانطور که در بالا دیدیم نمونه برداری پهنای باند به فرکانس نمونه برداری وابسته است. اگر fs مثلاً 103 نمونه در ثانیه بود، بنابر این مولفه اول می شد:

$$4\cos(2\pi fn/fs)=4\cos(2\pi 1010n/103)=4\cos(2\pi(927+83)n/103)=4\cos(2\pi 927n/103+2\pi 83n/103)=4\cos(2\pi 9n+2\pi 83n/103)$$

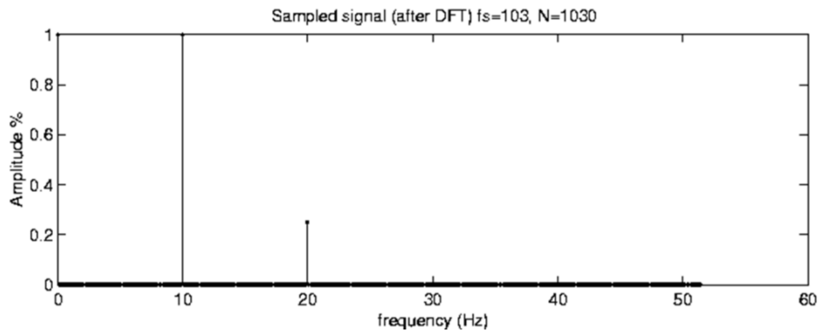
چون n عدد صحیح است، این عبارت به صورت زیر خلاصه می شود:

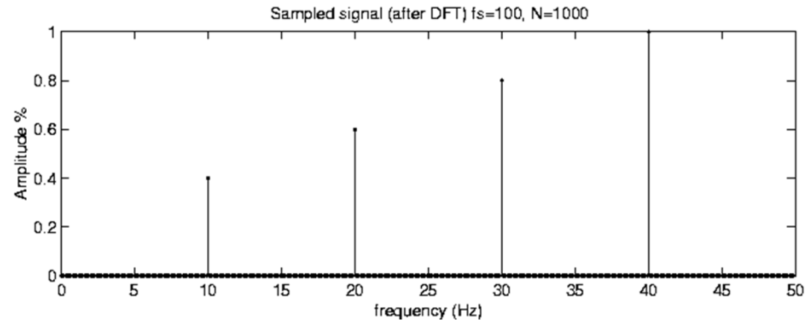
$$= 4\cos(2\pi 83n/103)$$

عملیات اینجا متوقف نمی شود. چون $83n/103$ را می توان بصورت $(103-20)n/103$ نوشت، می توان عبارت دیگری از $2\pi n$ را از آن حذف نمود که در آن صورت ما تنها 20Hz- را خواهیم داشت. در هر حال 20Hz- وجود خواهد داشت. بنا بر این طیف اطلاعات از fs- تا fs را شامل خواهد بود که اطلاعات مشابه را در هر دو سوی محور با مرکزیت صفر خواهد داشت. مولفه فرکانسی 83/103 بزرگتر از مولفه 103/2 (یعنی fs /2) است بنابراین در طیف بصورت 20/103- ظاهر خواهد شد. و چون قسمت چپ طیف بصورت آینه قسمت راست است بنابر این این مولفه بصورت 20+ هرتز نیز ظاهر خواهد شد. به عبارت دیگر برای هر مقدار حقیقی φ، $\cos(-\varphi)=\cos(\varphi)$ است. بطور مشابه وقتی ضرایب 103 را از مولفه های دیگر حذف می کنیم داریم:

$$X[n]= 4\cos(2\pi\text{-}20n/103)+ 6\cos(2\pi\text{-}10n/103)+ 8\cos(0) + 10\cos(2\pi 10n/103).$$

اطلاعات مولفه های دوم و چهارم در 10 هرتز ظاهر می شود به این معنی که ما اطلاعات را در اینجا از دست دادیم. شکل 5.11 نشان می دهد که طیف باید مشابه چه چیزی باشد اگر نمونه برداری در 103 هرتز انجام می شد. نکته جالب این است که مولفه در 0 هرتز بنظر می رسد دو برابر هر مولفه 0 هرتزی است. دامنه ها به هم وابسته اند و از چون انتظار داریم، مثلا مولفه 10 هرتزی دارای دامنه مساوی با مولفه 10- هرتزی باشد، انتظار داریم این اتفاق برای 0 Hz (یعنی 0Hz- و 0Hz) نیز بیافتد. با استفاده از نمونه برداری 2081 در ثانیه آنطور که برای شکل 5.9 انجام شد، از دوبرابر شدن دامنه در 1040 ثانیه جلوگیری به عمل آمده است.





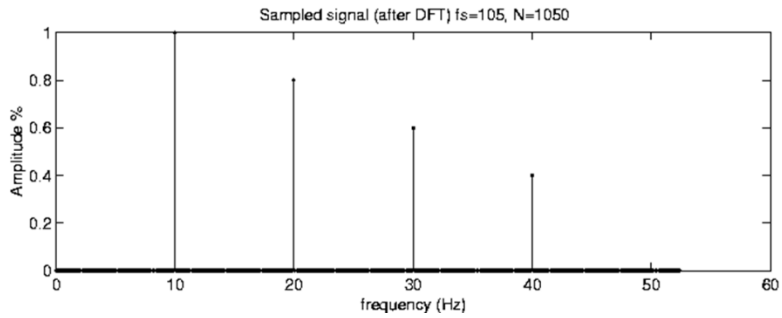
تمام اطلاعات سیگنال حقیقی دربین 0 و fs /2 طیف نشان داده خواهد شد حتی اگر مولفه بزرگتر از fs /2 نیز وجود داشته باشد. ایده ای که در پشت این منطق قرار دارد از نرخ نایکوئست سرچشمه می گیرد یعنی fs /2 باید حداقل مساوی پهنای باند باشد. چون ما از پهنای باند آگاهیم و نیازمند به مشخص شدن fs هستیم، می توان نوشت: fs≥ 2B. دو روش از اطلاعات سیگنال به طیف (از 0 تا fs /2) وجود دارد: در روش نخست اطلاعات طیفی بطور بینهایت برای هر یک از مضرب fs تکرار خواهد شد. راه دیگر این است که تصور کنیم هر اطلاعات طیفی که در سیگنال آنالوگ وجود دارد، در فرکانس های مابین صفر تا fs /2 درست ماتند آینه محدب که هر شی را با هر اندازه ای نشان می دهد، در بر خواهد گرفت. اگر دقت کنیم تا از همپوشانی (دگرنامی) احتراز نماییم این خود یک برتری محسوب می شود. چگونه مقدار fs را انتخاب می کنیم؟ همانگونه که قبلاً تشریح نمودیم، محدودیت اختیار نمودن fs ≥2B کماکان وجود دارد. علاوه بر این ، هر مولفه فرکانسی باید به فرکانس یگانه ای که بین صفر تا fs /2 است نسبت داده شود. این موضوع را می توان با معادله زیر نشان داد:

$$(2fc - B)/m \geq fs \geq (2fc + B)/(m+1)$$

متغیر fc فرکانس مرکز است که نماینده نقطه میانی حداقل و حداکثر فرکانس مورد علاقه است. این متغیر را بصورت بزرگترین فرکانس مورد علاقه منهای نصف پهنای باند نیز می توان تعریف نمود. متغیر m یک عدد صحیح است و در واقع تعداد نسخه برداری و یا اینکه چند بار اطلاعات بین 0 و حداقل فرکانس مورد علاقه تکرار می شود، را نشان می دهد. آن به تعداد مضرب ها fs مشابه است که می توانیم از فرکانس های f مان حذف کنیم. به مثال قبل مراجعه می کنیم جاییکه علاقمند بودیم تا اطلاعات بین 1010 و 1040 هرتز را حفظ کنیم:

در اینجا B=1040-1010=30 و fc=1010+B/2=1025Hz

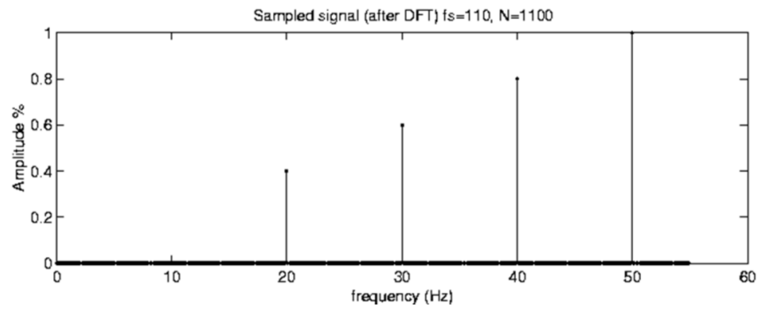
می توان محدوده fs را با معادله بالا برای مقدار مشخص از m پیدا کنیم. اگر m=20 باشد، ملاحظه می شود که می توان محدوده 2020/20Hz ≥ fs ≥ 2080/21 Hz یا 101 Hz ≥ fs≥ 99.0476 Hz را بدست آورد. مشاهده می شود که fs=100samples/sec. بین این محدوده قرار می گیرد. همچنین باید اطمینان حاصل نمود که معیار نایکوئست نیز رعایت شده و چون 100Hz ≥ 2B یا 100Hz≥60Hz است، پس مورد مذکور صحت دارد. اگر مجددا محاسبات را با انتخاب m=19 پی گیری کنیم، محدوده 106.3158 Hz ≥ fs≥ 104 Hz بدست می آید. در اینجا لازم است به دو نکته توجه شود: نخست اینکه نرخ نمونه برداری 103 نمونه در ثانیه خارج از این محدوده است که خود موید این است که نمی توانستیم در مثال قبل از آن استفاده کنیم. ثانیا اینکه اگر نرخی نظیر 105 samples/sec. را انتخاب می کردیم (که در محدوده قبلی بوده) به گراف 5.12 می رسیم. چیز جالب در اینجا این است که به اطلاعات مشابهی نایل آمده فقط ، برعکس آنچیزی است که ما انتظارش را داشتیم.



اگر سیگنال اولی را با نمونه برداری .105 samples/sec امتحان کنیم خواهیم داشت:

4cos(2πfn/fs)=4cos(2π1010n/105)=4cos(2π(1050-40)n/105)=4cos(2π1050n/105+2π40n/105)=4cos(2π10n-2π40n/105)

این عبارت به 4 cos(-2π40n/105) خلاصه می شود و چون cos(-θ)=cos(θ) این مولفه 1010Hz در 40Hz ظاهر می شود. در یک وضعیت مشابه مولفه دیگر فرکانسی برعکس فرم اولیه خود ظاهر می شود. اگر m فرد باشد اطلاعات وجود دارد ولی برعکس است. اینکه این اطلاعات مورد قبول باشد، به کاربرد آن وابسته است. اتفاقاً اگر بنا بود که fs = 110 samples/sec را که در محدوده fs وقتی m=18 باشد را انتخاب می کردیم، اطلاعات آنطور که در شکل 5.13 نشان داد شده صحیح است. جدول 5 این تحلیل ها را بطور خلاصه نشان داده است. در این جدول با نمونه های 103 در یک ثانیه نتایج در دو فرکانس که مقدار مشاهده شده مشابهی را منتج می شوند، بدست می آید. دیگر فرکانس های نمونه برداری مطلوب بوده چون هر فرکانس واقعی با تنها یک فرکانس مشاهده شده آدرس داده می شود.



جدول 5.1: فرکانسهایی که با نمونه برداری باندگذر آشکار شدند.

SAMPLING FREQUENCY	1010	1020	1030	1040
100	10	20	30	40
103	20	10	0	10
105	40	30	20	10
110	20	30	40	50

چرا فرکانس 83/103 بزرگتر از 103/2 است؟

- Zaman Malekzade
- Dear Michael after studying some of chapters of your book (DSP 2007 VERSION) ...
- May 1 (5 days ago)
- Michael Weeks
- May 2 (4 days ago)
- to me, mweeks
- Hello Zaman,

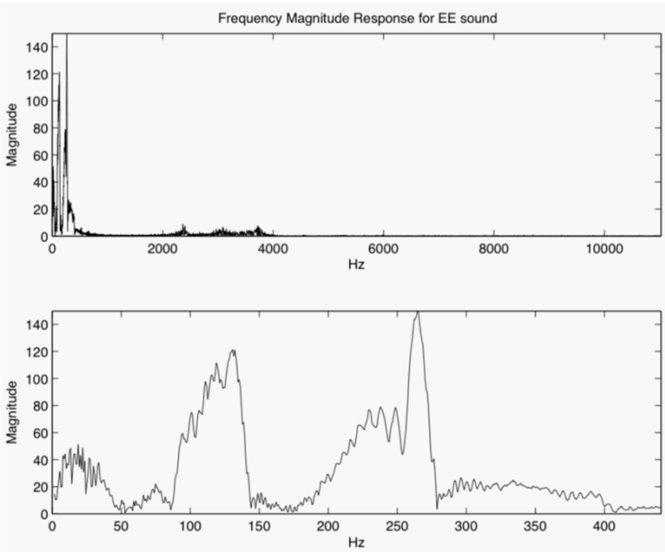
  I see why this is confusing. Perhaps "103/2" should be "103/2/103" to make the two comparable. Or "This frequency component 83/103 is greater" would be clearer as "This apparent frequency at 83 Hz is greater".

By the way, there is a second edition of the book available.

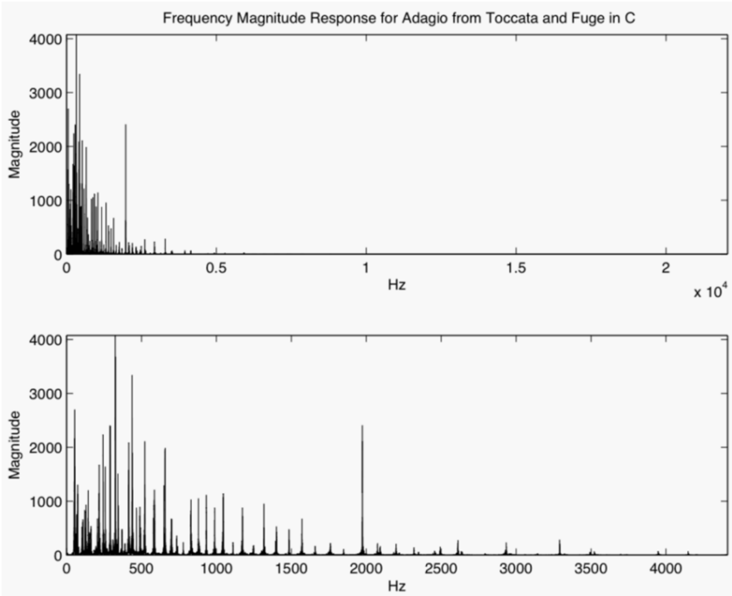
-MCW

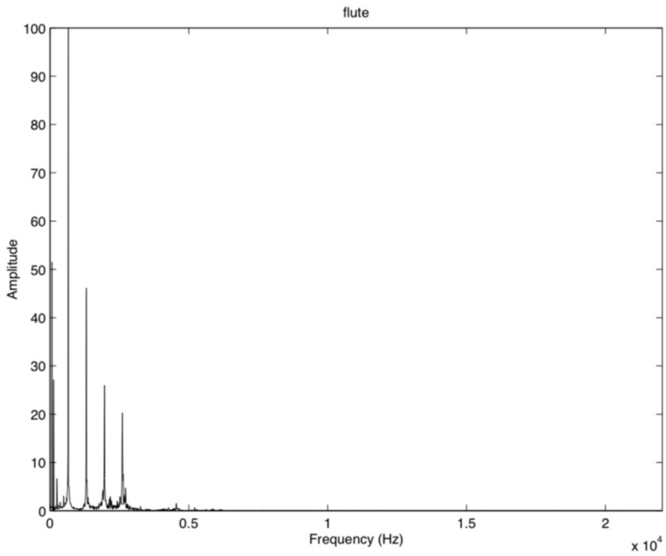
تبدیل فوریه

در زمینه های مختلف شامل پزشکی، نور، فیزیک، ژئوفیزیک و مهندسی الکترونیک تبدیل فوریه بعنوان یک وسیله و ابزار تجزیه و تحلیل متداول محسوب می شود. در عمل JPEG و MPEG از نسخه تغییر شکل یافته تبدیل فوریه استفاده می کنند. اساسا، این تبدیل به ما اجازه می دهد تا اطلاعات فرکانسی را بجای اطلاعات زمانی نگاه کنیم چیزی که معمولا اغلب مردم با آن مواجهند. بعنوان مثال بسیاری از سامانه استریویی ردیف هایی از چراغهای کوچک دارند که بر حسب قدرت باند فرکانسی روشن می شوند. بعنوان مثال، هر چه صدا ریزتر باشد چرغهای بیشتری در یک ردیف روشن شده و ستونهایی منور را ایجاد می کنند که در حین پخش صدا برخاسته و یا نشسته می شوند. این پدیده نوعی از اطلاعات محسوب می شود که توسط تبدیل فوریه تولید می شود. تبدیل گسسته فوریه نسخه ای از تبدیل فوریه است که ما قصد داریم به آن بپردازیم چون این نسخه با اطلاعات یا داده گسسته کار می کند. داده های ما در زمان گسسته هستند پس می توانیم فرض کنیم که تناوبی هستند. به عبارت دیگر اگر N تعداد دیگر نمونه بگیریم این داده ها بر حسب فرکانس هایی که ارائه می کنند، در واقع تکرار داده هایی هستند که قبلا داشتیم. در این وضعیت ما از DFT استفاده می کنیم که اطلاعات فرکانسی گسسته که فرض می کنیم تناوبی هستند تولید می کنند. در شکل 6.1 نمودار پاسخ بزرگای فرکانسی را برای صدای ee مشاهده می کنیم. شکل مذکور در قسمت بالا کل پاسخ و در بخش پایینی مقدار منتخب در فرکانس تا تقریباً 500 هرتز را نشان می دهد.



نمودار شکل بعدی (6.2) پاسخ بزرگای فرکانسی را برای فایل صوتی دیگری را که مربوط است به یک آهنگ ملایم از باخ نشان می دهد.قسمت بالای شکل محدوده کل فرکانسی از 0 تا 22050 هرتز را نشان می دهد در حالیکه در قسمت پایینی بخش کمتری از این شکل را که محتوی اطلاعات تا 4500 هرتز است را نشان می دهد. چیز جالبی که باید توجه شود این است که قله ها در بزرگا بطور منظم نسبت به هم قرار دارند. این مورد وقتی که با سیگنال واقعی سر و کار داریم خصوصاً موسیقی برخورد می شود. بعنوان مثال درست قبل از 1000Hz سه قله را که بزرگای شان افزایش پیدا می کند و مربوط به سه فرکانس مختلف ولی مرتبط به هم هستند را مشاهده می کنیم ما به این هارمونیک می گوییم. در شکل 6.3 که مربوط است به یک نوای نی بطور واضح تری این پدیده را مشاهده می کنیم. چهار فرکانس بطور ملموسی مشخص و بقیه در حد صفر هستند.

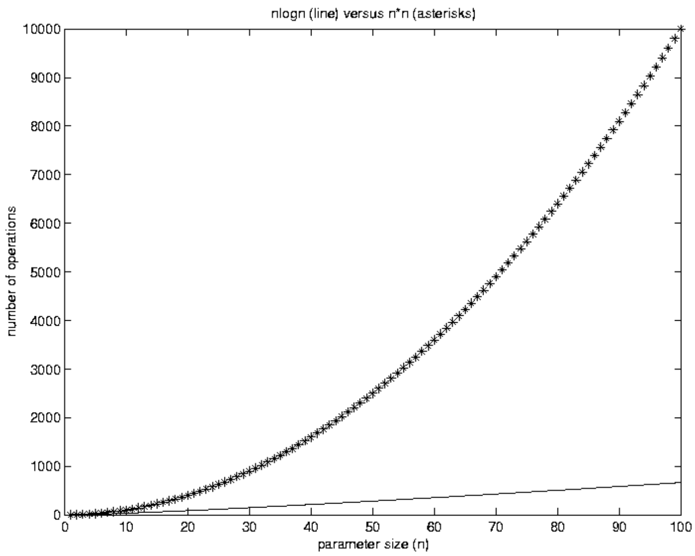




محدوده فرکانسی که در شکل بالا ظاهر می شود، (از 0 تا 22050 هرتز) بطور اختیاری انتخاب نشدند. دیسک های فشرده (CDs) موسیقی هایی ضبط شده با 44100 نمونه برثانیه که اجازه می دهند تا صداهایی در محدوده فرکانسی 0 تا 22050 هرتز را که قدری بیشتر از فرکانس شنوایی ما هستند را ذخیره کنند. جای تعجب ندارد تقریباً همه محتوی فرکانسی در موسیقی مندرج در شکل 6.2 دارای فرکانس کمتر از 4000 هرتز هستند. ارگ که این موسیقی را می نوازد صداها با محدوده وسیعی فرکانسی را تولید می کند. بعض از ارگ ها نت های ماورا صوت را تولید می کنند (یعنی زیر 20 هرتز) که اغلب آدم ها قادر به شنیدن آن نیستند (مشابه امواج زمین لرزه). همچنین صداهایی که بیش از 10 کیلوهرتز هستند. اما این اصوات برای یک موسیقی خوب مورد نیاز نیستند. برای تجسم بیشتر اغلب وسایل صوتی را در نظر بگیرید (شامل گیتار، ویولون، چنگ، طبل و شیپور) نمی توانند نت هایی با فرکانس اساسی بیش از 4000 هرتز را تولید نمایند. گرچه وسایلی هستند که می توانند هارمونیک های بیش از این فرکانس را تولید نمایند. یک پیانو فرکانس با محدوده 27.5 تا بیش از 4186 فرکانس را تولید می کند. تبدیل فوریه شیوه ای است برای به تصویر کشاندن زمان پیوسته، داده غیر متناوب به فرکانس پیوسته و داده غیر متناوب در حوزه فرکانس. یک متغیر موسوم به سری های فوریه با داده های تناوبی که در زمان پیوسته کار می کند و آنرا به داده غیر تناوبی و به فرکانس گسسته غیر تناوبی بر میگرداند. وقتی داده ها در زمان گسسته وغیر تناوبی هستند، می توانیم از تبدیل فوریه زمان گسسته (DTFT) استفاده کنیم تا نمایشی از یک داده با فرکانس پیوسته و تناوبی را ارائه نماییم. خلاصه اینکه وقتی داده ها در زمان گسسته (و ما فرض می کنیم که تناوبی هستند) می توانیم از DFT برای بدست آوردن یک فرکانس پیوسته با فرض تناوبی بودن استفاده کنیم.

تبدیل سریع در مقابل تبدیل گسسته فوریه

Fast Fourier Transform (FFT) یا تبدیل سریع فوریه یک الگوریتم هوشمند برای انجام تبدیل فوریه گسسته یا DFT است. همانطور که انتظار می رود، با انجام این تبدیل نتایج مشابهی نسبت به DFT رقم می خورد اما بسیار سریعتر چون از الگوریتم خاصی تبعیت می کند. این تبدیل یک میانبر بزرگ محسوب شده چون به محقق اجازه می داد تا طی زمان معقولی تبدیل فوریه را انجام دهد. شکل 6.4 این اختلاف را نشان می دهد. منحنی زیری یک نمودار $Nlog_2(N)$ در حالیکه دیگری N^2 است.



بعنوان مثال پردازشگر مدل جدید پنتیوم 4 در 2GHz ، 2 بیلیون محاسبه را در یک ثانیه انجام می دهد. یک الگوریتم (نظیر DFT) که N^2 عملیات را انجام می دهد 5 ثانیه طول می کشد تا 100000 نمونه داده را تحلیل کند. اگر داده به 1000 برابر فوق برسد این مدت به حدود 3 ماه می کشد ولی برای الگوریتم FFT که تعداد عملیات $Nlog_2N$ را انجام می دهد این عملیات فقط 1.33 ثانیه طول می کشد. متلب هم تابع FFT و هم IFFT را ممکن می سازد. برای سرعت بهینه FFT نیاز دارد اندازه داده به توان 2 باشد ولی این موضوع در متلب نیاز به این محدودیت ندارد. در اغلب کاربردها صفرها به داده ها اضافه می شوند بدون آنکه تاثیر منفی بر نتایج بگذارند. در واقع صفرها برای آنکه نتایج بهتر بنظر برسند اضافه می شوند. صفرها اطلاعاتی را اضافه نمی کنند اما فرکانس های تحلیل مورد استفاده FFT و یا DFT را تغییر می دهند. این تکنیک موسوم به zero - padding است. در بخش سه دیدیم فیلترها کانولوشن را انجام می دهند. می توان بکمک fft بطور کارامدی این عمل را انجام داد. استفاده دیگر شامل تحلیل اثر فیلتر بر روی سیگنال و طراحی فیلتر FIR (بکمک FFT) را می توان نام برد. هر وقت مسئله به DFT نیاز دارد FFT را بجای آن می توان انجام داد.

DFT

رابطه زیر DFT را نشان می دهد:

$$X[m] = \sum_{n=0}^{N-1} x[n](\cos(2\pi nm/N) - j\sin(2\pi nm/N))$$

در رابطه فوق m=0.....N-1 است. ما به این معادله "شکل مستطیلی" از DFT می نامیم. این نوع تبدیل انواع مختلفی دارد. بعنوان مثال استفاده از $e^{-j2\pi nm/N}$ بجای سینوسی است که حل آنرا برای ما آسانتر می سازد.

$$X[m] = \sum_{n=1}^{N-1} x[n] e^{-j2\pi nm/N}$$

در اینجا یک تابع "function" متلب وجود دارد که DFT را حساب می کند. هدف آن این است که نشان دهد چگونه DFT می تواند محاسبه شود اما کارایی FFT یی که بر روی متلب نصب است را ندارد. این تابع آرایه مختلط را بر می گرداند که دو قسمت اطلاعات را بر حسب عدد مختلط در ارتباط با

مختصات x و y شامل می شود. که معمولا بزرگا و زاویه فاز هستند. درست مانند وقتی که می خواهیم داده را از مختصات کارتزین به مختصات قطبی ببریم، می توانیم اطلاعاتی که بصورت اعداد مختلط هستند را به بزرگا و فاز تبدیل نماییم. توجه کنید که چقدر عبارت جبری کسینوس و سینوس به هم مشابه و تابعی از n و m هستند. این تبدیل بسط داده یک بعدی به ماتریس دو بعدی است. با یک مثال موضوع را بازتر می کنیم: کد زیر DFT را از سیگنال بدست می آورد. (این کد از fft استفاده می کند اما dft نیز با فرض اینکه برنامه فوق وجود داشته باشد کار می کند.) جدول 6.1 یک ماتریس را نشان می دهد که حاصل حل سیگنال مثال فوق است. ردیف مربوط به n است و ملاحظه می کنیم که مقادیر را بر حسب مقادیر m که در ستون دارد برای سیگنال محاسبه می کند. مشاهده کنید که چطور هر ردیف دارای نمونه سیگنال مربوطه (داده های حوزه زمان) ولی حاصل جمع هر یک از ستون حاصل اعمال DFT بر سیگنال است (داده حوزه فرکانس).

Table 6.1: Example DFT calculations.

	0	1	2	3	4	5	6	7
0	$6e^{-j2\pi 0}$	$6e^{-j2\pi 0}$	$6e^{-j2\pi 0}$	$6e^{-j2\pi 0}$	$6e^{-j2\pi 0}$	$6e^{-j2\pi 0}$	$6e^{-j2\pi 0}$	$6e^{-j2\pi 0}$
1	$4e^{-j2\pi 0}$	$4e^{-j2\pi 0.125}$	$4e^{-j2\pi 0.25}$	$4e^{-j2\pi 0.375}$	$4e^{-j2\pi 0.5}$	$4e^{-j2\pi 0.625}$	$4e^{-j2\pi 0.75}$	$4e^{-j2\pi 0.875}$
2	$9e^{-j2\pi 0}$	$9e^{-j2\pi 0.25}$	$9e^{-j2\pi 0.5}$	$9e^{-j2\pi 0.75}$	$9e^{-j2\pi}$	$9e^{-j2\pi 1.25}$	$9e^{-j2\pi 1.5}$	$9e^{-j2\pi 1.75}$
3	$0e^{-j2\pi 0}$	$0e^{-j2\pi 0.375}$	$0e^{-j2\pi 0.75}$	$0e^{-j2\pi 1.125}$	$0e^{-j2\pi 1.5}$	$0e^{-j2\pi 1.875}$	$0e^{-j2\pi 2.25}$	$0e^{-j2\pi 2.625}$
4	$1e^{-j2\pi 0}$	$1e^{-j2\pi 0.5}$	$1e^{-j2\pi 1}$	$1e^{-j2\pi 1.5}$	$1e^{-j2\pi 2}$	$1e^{-j2\pi 2.5}$	$1e^{-j2\pi 3}$	$1e^{-j2\pi 3.5}$
5	$5e^{-j2\pi 0}$	$5e^{-j2\pi 0.625}$	$5e^{-j2\pi 1.25}$	$5e^{-j2\pi 1.875}$	$5e^{-j2\pi 2.5}$	$5e^{-j2\pi 3.125}$	$5e^{-j2\pi 3.75}$	$5e^{-j2\pi 4.375}$
6	$2e^{-j2\pi 0}$	$2e^{-j2\pi 0.75}$	$2e^{-j2\pi 1.5}$	$2e^{-j2\pi 2.25}$	$2e^{-j2\pi 3}$	$2e^{-j2\pi 3.75}$	$2e^{-j2\pi 4.5}$	$2e^{-j2\pi 5.25}$
7	$7e^{-j2\pi 0}$	$7e^{-j2\pi 0.875}$	$7e^{-j2\pi 1.75}$	$7e^{-j2\pi 2.625}$	$7e^{-j2\pi 3.5}$	$7e^{-j2\pi 4.375}$	$7e^{-j2\pi 5.25}$	$7e^{-j2\pi 6.125}$
Σ	$34e^{j2\pi 0}$	$9.3e^{-j2\pi 0.023}$	$4.5e^{-j2\pi 0.426}$	$12.7e^{j2\pi 0.24}$	$2e^{-j2\pi 0}$	$12.7e^{-j2\pi 0.24}$	$4.5e^{j2\pi 0.4}$	$9.3e^{j2\pi 0.023}$

اگر بنا بود جمع بزرگاها را در طول هر ردیف بدست آوریم (یعنی ((sum(abs(Matrix(r,:))) بایستی حاصل ضرب نمونه در حوزه زمان را در تعداد نقاط N بدست می آوردیم. هر $e^{-j\theta}$ بزرگای واحد دارد این مقادیر بردار مختلط با بزرگای واحد هستند. چگونگی حاصل جمع مقادیر نمایی مختلط ممکن است واضح نباشد. می توانیم همیشه این مقادیر را نخست به مختصات کارتزین تبدیل کنیم (به شکل a+jb). جدول 6.2 راه دیگری را برای نمایش این داده در قالب مستطیلی معادله DFT ارائه می کند. براحتی می توانیم چک کنیم که جمع ستون مساوی نتیجه DFT ارائه شده از سوی متلب است چون بخش های حقیقی و مجازی را بطور جداگانه جمع می کنیم.

Table 6.2: Example DFT calculations (rectangular form).

	0	1	2	3	4	5	6	7
0	6.0 + 0.0j	6.0 + 0.0j	6.0 + 0.0j	6.0 + 0.0j	6.0 + 0.0j	6.0 + 0.0j	6.0 + 0.0j	6.0 + 0.0j
1	4.0 + 0.0j	2.8 - 2.8j	0.0 - 4.0j	-2.8 - 2.8j	-4.0 - 0.0j	-2.8 + 2.8j	-0.0 + 4.0j	2.8 + 2.8j
2	9.0 + 0.0j	0.0 - 9.0j	-9.0 - 0.0j	-0.0 + 9.0j	9.0 + 0.0j	0.0 - 9.0j	-9.0 - 0.0j	-0.0 + 9.0j
3	0.0 + 0.0j	0.0 + 0.0j	0.0 + 0.0j	0.0 + 0.0j	0.0 + 0.0j	0.0 + 0.0j	0.0 + 0.0j	0.0 + 0.0j
4	1.0 + 0.0j	-1.0 - 0.0j	1.0 + 0.0j	-1.0 - 0.0j	1.0 + 0.0j	-1.0 - 0.0j	1.0 + 0.0j	-1.0 - 0.0j
5	5.0 + 0.0j	-3.5 + 3.5j	0.0 - 5.0j	3.5 + 3.5j	-5.0 - 0.0j	3.5 - 3.5j	-0.0 + 5.0j	-3.5 - 3.5j
6	2.0 + 0.0j	-0.0 + 2.0j	-2.0 - 0.0j	0.0 - 2.0j	2.0 + 0.0j	-0.0 + 2.0j	-2.0 - 0.0j	-0.0 - 2.0j
7	7.0 + 0.0j	4.9 + 4.9j	-0.0 + 7.0j	-4.9 + 4.9j	-7.0 - 0.0j	-4.9 - 4.9j	-0.0 - 7.0j	4.9 - 4.9j
Σ	34.0 + 0.0j	9.2 - 1.3j	-4.0 - 2.0j	0.8 + 12.7j	2.0 - 0.0j	0.8 - 12.7j	-4.0 + 2.0j	9.2 + 1.3j

هر ستون یک نقطه حوزه فرکانس منفرد را منتج می شود. به عبارت دیگر برای هر خروجی X[m] فرکانس های مورد استفاده برای X[m] ، $2\pi m(0)/N$, $2\pi m(1)/N, \ldots, 2\pi m(N-1)/N$ است. تعداد فرکانس های مورد استفاده کاملا به تعداد نقاطی که داریم وابسته است که توسط فرکانس نمونه برداری ما تعیین می شود. بنابر این فرکانس های تحلیل با رابطه زیر ارائه می شوند:

$$f_{analysis}[m] = m f_{sampling} / N$$

ترسیم طیف

برای اینکه به چگونگی بدست آوردن یک طیف آشنا شویم به یک مثال بسنده می کنیم: کد زیر یک سیگنال X را معرفی می کند:

```
% Set up an example signal
n = 0:99; % number of points
fs = 200;  % sampling frequency
Ts = 1/fs; % sampling period
```

```
% x is our example signal
x = cos(2*pi*20*n*Ts + pi/4) +  3*cos(2*pi*40*n*Ts - 2*pi/5) +   2*cos(2*pi*60*n*Ts + pi/8);
```

اکنون به اطلاعات فرکانسی از X نیاز است که از تبدیل فوریه قابل استخراج است.

```
% Find the spectrum
X = fft(x);
m = 0:length(X)-1;
```

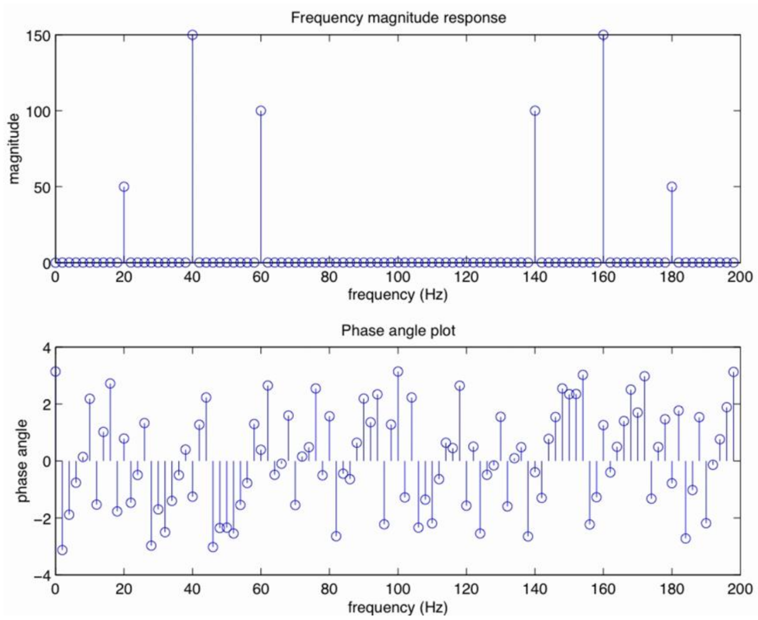
اگر به وضوح فرکانسی نیاز باشد کد زیر آنرا نشان می دهد:

```
disp(sprintf('Freq resolution is every %5.2f Hz',...
        fs/length(X)));
```

برای دیدن طیف هم فرکانس و هم زاویه فاز را نشان می دهیم:

```
% Plot magnitudes
subplot(2,1,1);
stem(m*fs/length(X),abs(X), 'b');
ylabel('magnitude');
xlabel('frequency (Hz)');
title('Frequency magnitude response');
% Plot phase angles
subplot(2,1,2);
stem(m*fs/length(X),angle(X).*X3, 'b');
ylabel('phase angle');
xlabel('frequency (Hz)');
title('Phase angle plot');
```

اگر برنامه فوق را اجرا نماییم شکل گراف چیزی مشابه شکل 6.5 خواهد شد که بوضوح حضور محتوی فرکانسی در 20 ، 40 و 60 هرتز را مشاهده می کنیم با بزرگای بترتیب 50، 150 و 100 که متناسب است با 1، 3 و 2 در سیگنال اولیه ای که ارائه نمودیم. با این حال مواردی در شکل ملاحظه می شود که ممکن است ناخوشایند باشد بعنوان مثال اولا چون سیگنال اولیه حقیقی است، پاسخ بزرگای فرکانسی یک تصویر آینه ای دارد. در حالیکه ما به نیمه نخست آن نیاز داریم. برای فازها نیز همین اتفاق در حول محور x می افتد که در این مورد نیز به نصف این تصویر نیاز داریم. نمودار فاز حاوی اطلاعات زیادی است بعنوان مثال در فرکانس 80 هرتز ملاحظه می شود که دامنه صفر است در حالیکه زاویه فاز نسبتا بزرگ است. لازم بذکر است که ما انتظار داریم زوایای فاز بین π− تا π+ باشد.



برای حل مشکل نخست، براحتی می توان نیمه نخست را هم برای بزرگا و هم برای فاز تصویر نمود. می توانیم یک متغیر بنام half_m را معرفی که به ما نصف محدوده ای که از برنامه استخراج می شود را ارائه نموده و آنرا جای m قرار دهد. اما وقتی که از آن استفاده می کنیم، باید دقت نمود که 1 را به آن اضافه نمایم چون ما از صفر آغاز کردیم.

```
half_m = 0:ceil(length(X)/2);
stem(half_m*fs/length(X),abs(X(half_m+1)), 'b')
```

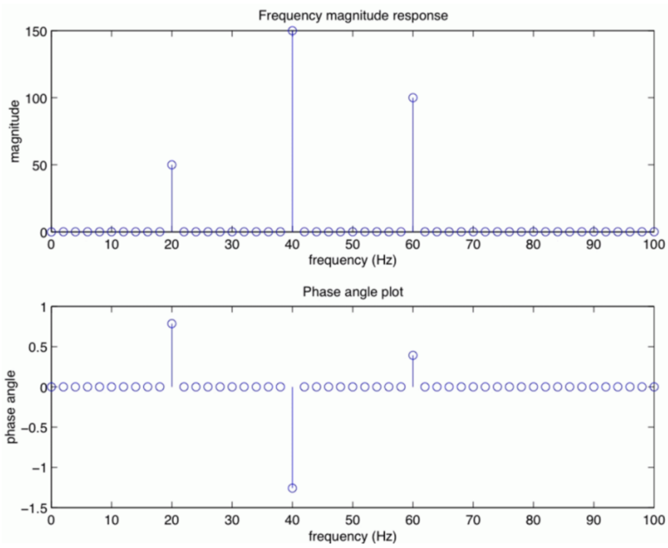
برای مشکل دوم (فاز) به اطلاعات بیشتری نیازمند است. مقداری را به نرم افزار می دهیم: مثلاً X=2 جواب 16871e-13-2.2465e-15i- است همچنین مقدار angle(X(2)) را جستجو128. می کنیم مقدار برگشتی یا جواب 3.1283- است. همچنین abs(x(2)) را جستجو می کنیم، مقدار یا جواب متلب 1.6873e-13 است. دستورات فوق به تردیدات ما مبنی بر اینکه زوایای فاز برای هر مقدار از X حتی وقتی که نزدیک به صفر هستند، صحه میگذارند. برای حل این مشکل از مقدار مجاز استفاده می کنیم. اگر بزرگا کوچکتر از مقدار مجاز باشد، بنابر این فرض می کنیم مقدار فاز صفر است. بالاخره بزرگا ها به مقدار مشارکت سینوسی ها دلالت دارند. اگر مشارکت سینوسی تقریباً صفر باشد در نظر گرفتن فاز برای آن منطقی نیست.

```
% The next 3 lines allow us to ignore phases
% that go with very small magnitudes.
tolerance = 0.00001;
X2 = ceil(abs(X) - tolerance);
X3 = round(X2 ./ (X2+1));
% X3 above is a vector of 0s and 1s
```

دستورات بالا ممکن است قدری مبهم باشند اما با اجرای این دستورات برداری باینری ایجاد می شود. تفریق، بزرگا ها را برای مان تفکیک می کنند چون همه بزرگاها مثبت هستند، مقادیر منفی بعد از تفریق به فازهایی که ما باید آنها را در نظر نگیریم منسوب می شوند و تابع ceil آنها را صفر می کند. اکنون مشکل ما این است که باید تکلیف مقادیر X2 که بین 0 و 1 هستند را مشخص نماییم. با تقسیم هر X2 به خودش (باضافه 1) مقادیری بدست می ایند که صفر یا تقریباً 1 هستند ضمن اینکه تقسیم بر 0 نیز موضوعیت ندارد. گرد کردن عدد نیز اعداد نزدیک به یک را بصورت 1 بر می گرداند. نتیجه عملیات در بردار X3 ذخیره می شود که شامل صفرها برای مقادیری است که مورد نظر ما نیست و 1 ها که برای فازهایی است که می خواهیم آنها را ببینیم. نمودار زیر نتایج را نشان می دهد که مقادیر فاز را که بزرگای نزدیک به صفر دارند را صفر نگه داشته است. این نمودار مقادیر را نصف ننموده تا از تکرار آن جلوگیری نماید ولی می توان این موضوع را در کد گنجاند.

```
subplot(2,1,2);
stem(m*fs/length(X),angle(X).*X3, 'b');
```

حالا می توانیم همه موارد فوق را در یک برنامه لحاظ نموده و طیف را ترسیم نماییم. وقتی این برنامه را اجرا نماییم، شکلی مشابه شکل 6.6 خواهیم داشت که در آن بزرگا و فاز بر حسب فرکانس ها ترسیم شده اند. برنامه الحاق شده به این اسلاید یک جمع بندی از برنامه ای که تاکنون بصورت قطعه قطعه نشان دادیم را ارائه نموده که از اهم مراحل آن عبارتست از ارائه یک سیگنال با چند سینوسی – پیدا کردن تبدیل فوریه – چاپ وضوح فرکانسی – ترسیم بزرگا – ترسیم فاز که البته قبل از آن فازهایی که بزرگا های مربوطه شان مقادیر کوچک دارند صفر می شوند.



صفر گذاری Zero Padding

وقتی سطح یک سیگنال را در حوزه زمان صفر می کنیم با یک وضوح فرکانسی با شکل صاف تری از آن سیگنال مواجه می شویم. اتفاقی که در این فرایند حادث می شود این است که در واقع طول سیگنال نمونه برداری شده وضوح سیگنال را مشخص می کند و صفرگذاری اطلاعات جدیدی را اضافه نمی کند. اگر چه سیگنال صفرگذاری شده باعث می شود شکل سیگنال در حوزه فرکانس مطلوبتر شود، این روش تاثیری بر وضوح فرکانسی ندارد. چگونه می توانیم سیگنال خود در حوزه زمان را صفرگذاری نماییم؟ بخشی از جواب باید با تبدیل فوریه پیوسته انجام شود. برای راحتی کار از فرکانس رادیان ω=2πf استفاده می کنیم. استفاده از این قالب فرکانس به ما اجازه می دهد از f(t) برای تابع خود استفاده نماییم بدون اینکه با f فرکانس اشتباه گرفته شود.

تابع پیوسته فوریه:

$$F(\omega) = \int_{-\infty}^{+\infty} f(t)e^{-j\omega t}dt$$

فرض کنید f(t) سیگنالی است که در خارج از بازه [a,b] مقدار صفر و فرض نمایید a <b< c است. بنابر این می توان رابطه قبل را با رابطه زیر جایگزین نماییم:

$$F(\omega) = \int_{a}^{c} f(t)e^{-j\omega t}dt$$

$$= \int_{a}^{b} f(t)e^{-j\omega t}dt + \int_{b}^{c} f(t)e^{-j\omega t}dt$$

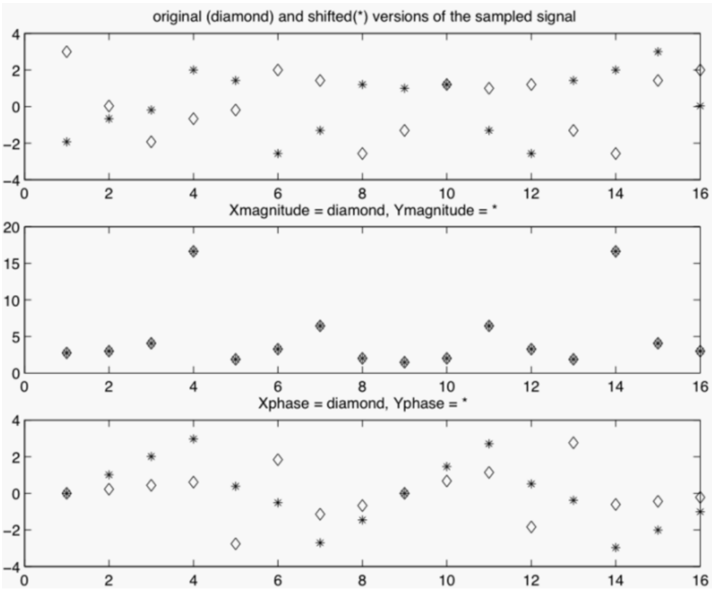
چون مقدار تابع خارج از بازه [a,b] صفر است پس عبارت دوم صفر خواهد بود. پس

$$F(\omega)=\int_a^b f(t)e^{-j\omega t}dt+0$$

بنابراین برای سیگنال های پیوسته مقادیر F(ω) مشابه هستند چه مقدار f(t) را در خارج از محدوده در نظر بگیریم و چه نگیریم. برای یک سیگنال ناپیوسته "discrete" تعداد نقاط برای سیگنال در حوزه فرکانس مساوی تعداد نقاط سیگنال در حوزه زمان بوده بنابر این اضافه کردن صفر به این معنی است که مقادیر بیشتری از نقاط در حوزه فرکانسی وجود خواهد داشت. در نتیجه تحلیل فرکانس ها ریزتر می شود چون به هم نزدیک تر هستند.

تئوری جابجایی DFT (shifting theory)

تئوری موسوم به جابجایی DFT وجود دارد که ابراز می دارد نمونه برداری یک سیگنال حتی وقتی که نمونه ها جابجا شوند نظیر حذف k نمونه نخست و اضافه کردن آن به انتهای توالی نمونه ها، نتایج مشابه (بر حسب پاسخ بزرگای فرکانسی) را به همراه خواهد داشت. این تئوری حائز اهمیت است چون مبین این است که یک زمان بحرانی برای آغاز ثبت نمونه ها وجود ندارد. برنامه زیر این ایده را بازتر می کند. توجه نمایید که چگونه دستور پلات، دو سیگنال را در یک زمان روی نمودار می گذارد یکی با رنگ آبی و دیگری سبز. رنگ های دیگر نظیر قرمز و سیاه را نیز امتحان کنید. شکل 6.7 نشان می دهد که چقدر خروجی برای این برنامه به هم شبیه هستند. اگرچه زوایای فاز مختلف هستند، بزرگاهای فرکانسی برای سیگنال اولیه و جابجا شده مشابه است. باید انتظار داشته باشیم زوایای فاز مختلف باشد چون این مقادیر چگونگی به خط شدگی سیگنال مرکب با نمونه ها را تعیین می کنند. در غیر اینصورت قادر نخواهیم بود تا به سیگنال اولیه از طریق تبدیل معکوس برگردیم.



تبدیل گسسته معکوس فوریه

تبدیل فوریه گسسته معکوس (IDFT) بصورت زیر است:

$$x[n]=\frac{1}{N}\sum_{m=0}^{N-1}X[m](\cos(2\pi nm/N)+j\sin(2\pi nm/N))$$

در این رابطه n=0….N-1 است. به تشابه این تابع با DFT دقت کنید. تفاوت اصلی این دو در وجود 1/N و علامت مثبت است که در رابطه تبدیل معکوس وجود دارد. علامت مثبت انطباق ندارد، تبدیل معکوس با استفاده از آنچه که ما بعنوان مزدوج مختلط می شناسیم کار می کند. مزدوج مختلط وقتی بوجود می آید که بخش مختلط علامت منفی دارد بعنوان مثال 4-j2 و 4+j2 مختلط مزدوج هستند. مزدوج مختلط را با علامت ستاره در بالای عبارت نشان می دهند. اگر a=3+j7 و b=3-j7 باشد می توان گفت که a=b* یا b=a* .

شکل دیگر IDFT بصورت زیر است که در آن فرم توانی جای سینوسی را با استفاده از فرمول اولر می گیرد.

$$x[n]=\frac{1}{N}\sum_{m=0}^{N-1}X[m]e^{j2\pi nm/N}$$

در زیر این پاراگراف تابع متلب اضافه شده که IDFT را محاسبه می کند. مشابه برنامه DFT که قبلاً ارائه شد، این برنامه نیز درصدد این است تا فقط نشان دهد که چگونه IDFT محاسبه می شود. در خود متلب تابعی بنام **ifft** وجود دارد که عملکرد مشابه ولی سریعتری دارد. در پاراگراف زیر دو تابع DFT و IDFT بوسیله متلب مورد مقایسه قرار گرفته است. نخست سیگنال mysignal چند نمونه را ارائه نموده بعد تابع DFT فراخوانده می شود. سپس IDFT اجرا و در mysignal_2 ذخیره می شود. این فایل باید مشابه mysignal باشد. مقادیر یک رقم گرد شده و مقادیر حقیقی منعکس و مجازی حذف می شوند.

تبدیل معکوس و مستقیم فوریه

در این بخش نشان خواهیم داد که وقتی یک سیگنال را اختیار می کنیم، بر روی آن DFT اجرا می کنیم، و سپس IDFT انجام می دهیم با مقدار مشابهی کار را که آغاز نمودیم، پایان خواهیم برد. نخست کار را با مرور فرمول های گذشته شروع می کنیم:

DFT:

$$X[m]=\sum_{n=1}^{N-1}x[n]e^{-j2\pi nm/N}$$

IDFT:

$$x[n]=\frac{1}{N}\sum_{m=0}^{N-1}X[m]e^{j2\pi nm/N}$$

کار را با یک مثال از سیگنال x={x0, x1, x2, x3 } شروع می کنیم. DFT را با استفاده از رابطه فوق پیدا کرده و چون می دانیم که N=4 است، می توان آنرا در فرمول جایگذاری نماییم. سپس تبدیل معکوس را انجام می دهیم:

$$X[m]=\sum_{n=1}^{3}x[n]e^{-j2\pi nm/N}$$

$$X[m]=x_0e^{-j2\pi 0m/4}+x_1e^{-j2\pi 1m/4}+x_2e^{-j2\pi 2m/4}+x_3e^{-j2\pi 3m/4}$$

$$x[n]=\frac{1}{4}\sum_{m=0}^{3}X[m]e^{j2\pi nm/4}$$

$$x[n]=\frac{1}{4}(X(0)e^{j2\pi n0/4}+X(1)e^{j2\pi n1/4}+X(2)e^{j2\pi n2/4}+X(3)e^{j2\pi n3/4}$$

سپس مقادیر X[m] را که در صفحه قبل بدست آوردیم، در رابطه فوق جایگزین می کنیم:

$$\begin{aligned}x[n]=\frac{1}{4}((&x_0e^{-j2\pi 0\times 0/4}+x_1e^{-j2\pi 1\times 0/4}+x_2e^{-j2\pi 2\times 0/4}+x_3e^{-j2\pi 3\times 0/4})e^{j2\pi n0/4}+\\
&+(x_0e^{-j2\pi 0\times 1/4}+x_1e^{-j2\pi 1\times 1/4}+x_2e^{-j2\pi 2\times 1/4}+x_3e^{-j2\pi 3\times 1/4})e^{j2\pi n1/4}+\\
&(x_0e^{-j2\pi 0\times 2/4}+x_1e^{-j2\pi 1\times 2/4}+x_2e^{-j2\pi 2\times 2/4}+x_3e^{-j2\pi 3\times 2/4})e^{j2\pi n2/4}+\\
&(x_0e^{-j2\pi 0\times 3/4}+x_1e^{-j2\pi 1\times 3/4}+x_2e^{-j2\pi 2\times 3/4}+x_3e^{-j2\pi 3\times 3/4})e^{j2\pi n3/4})\end{aligned}$$

با ترکیب نمایی ها:

$$\begin{aligned}x[n]=\frac{1}{4}((&x_0e^{-j2\pi 0\times 0/4+j2\pi n0/4}+x_1e^{-j2\pi 1\times 0/4+j2\pi n\times 0/4}+x_2e^{-j2\pi 2\times 0/4+j2\pi n\times 0/4}\\
&+x_3e^{-j2\pi 3\times 0/4+j2\pi n0/4})+(x_0e^{-j2\pi 0\times 1/4+j2\pi n1/4}+x_1e^{-j2\pi 1\times 1/4+j2\pi n1/4}\\
&+x_2e^{-j2\pi 2\times 1/4+j2\pi n1/4}+x_3e^{-j2\pi 3\times 1/4+j2\pi n1/4})+\\
&(x_0e^{-j2\pi 0\times 2/4+j2\pi n2/4}+x_1e^{-j2\pi 1\times 2/4+j2\pi n2/4}+x_2e^{-j2\pi 2\times 2/4+j2\pi n2/4}+x_3e^{-j2\pi 3\times 2/4+j2\pi n2/4})+\\
&(x_0e^{-j2\pi 0\times 3/4+j2\pi n3/4}+x_1e^{-j2\pi 1\times 3/4+j2\pi n3/4}+x_2e^{-j2\pi 2\times 3/4+j2\pi n3/4}+x_3e^{-j2\pi 3\times 3/4+j2\pi n3/4}))\end{aligned}$$

فاکتور از عامل مشترک:

$$x[n]=\frac{1}{4}((x_0e^{j2\pi(-0\times0+n0)/4}+x_1e^{j2\pi(-1\times0+n\times0)/4}+$$
$$x_2e^{j2\pi(-2\times0+n\times0)/4}+x_3e^{j2\pi(-3\times0+n0)/4})+$$
$$(x_0e^{j2\pi(-0\times1+n1)/4}+x_1e^{j2\pi(-1\times1+n1)/4}+$$
$$x_2e^{j2\pi(-2\times1+n1)/4}+x_3e^{j2\pi(-3\times1+n1)/4})+$$
$$(x_0e^{j2\pi(-0\times2+n2)/4}+x_1e^{j2\pi(-1\times2+n2)/4}+$$
$$x_2e^{j2\pi(-2\times2+n2)/4}+x_3e^{j2\pi(-3\times2+n2)/4})+$$
$$(x_0e^{j2\pi(-0\times3+n3)/4}+x_1e^{j2\pi(-1\times3+n3)/4}$$
$$+x_2e^{j2\pi(-2\times3+n3)/4}+x_3e^{j2\pi(-3\times3+n3/4})))$$

با ساده نمودن

$$x[n]=\frac{1}{4}((x_0e^{j2\pi0/4}+x_1e^{j2\pi0/4}+x_2e^{j2\pi0/4}+x_3e^{j2\pi0/4})+$$
$$(x_0e^{j2\pi n/4}+x_1e^{j2\pi(n-1)/4}+x_2e^{j2\pi(n-2)/4}+x_3e^{j2\pi(n-3)/4})+$$
$$(x_0e^{j2\pi(n2)/4}+x_1e^{j2\pi(2n-2)/4}+x_2e^{j2\pi(2n-4)/4}+x_3e^{j2\pi(2n-6)/4})+$$
$$(x_0e^{j2\pi(n3)/4}+x_1e^{j2\pi(3n-3)/4}+x_2e^{j2\pi(3n-6)/4}+x_3e^{j2\pi(3n-9)/4}))$$

با ساده تر کردن بیشتر داریم:

$$x[n]=\frac{1}{4}((x_0+x_1+x_2+x_3)+$$
$$(x_0e^{j2\pi n/4}+x_1e^{j2\pi(n-1)/4}+x_2e^{j2\pi(n-2)/4}+x_3e^{j2\pi(n-3)/4})+$$
$$(x_0e^{j2\pi(n2)/4}+x_1e^{j2\pi(2n-2)/4}+x_2e^{j2\pi(2n-4)/4}+x_3e^{j2\pi(2n-6)/4})+$$
$$(x_0e^{j2\pi(n3)/4}+x_1e^{j2\pi(3n-3)/4}+x_2e^{j2\pi(3n-6)/4}+x_3e^{j2\pi(3n-9)/4}))$$

بیایید ببینیم که اگر n=1 باشد چه چیزی بدست می آید:

بیاد آورید که این مقادیر نمایی نظیر 2π(-6)/4 مربوط به زوایا و هر زاویه ای که بزرگتر از 2π ضریب عدد صحیحی از آن حذف می شوند. بعنوان مثال:

ej2π(-6)/4 = e-j12π/4 = e-j3π= e-j(2π+π) = e-jπ

$$x[n]=\frac{1}{4}((x_0+x_1+x_2+x_3)+$$
$$(x_0e^{j2\pi1/4}+x_1e^{j2\pi0/4}+x_2e^{j2\pi(-1)/4}+x_3e^{j2\pi(-2)/4})+$$
$$(x_0e^{j2\pi(2)/4}+x_1e^{j2\pi(0)/4}+x_2e^{j2\pi(-2)/4}+x_3e^{j2\pi(-4)/4})+$$
$$(x_0e^{j2\pi(3)/4}+x_1e^{j2\pi(0)/4}+x_2e^{j2\pi(-3)/4}+x_3e^{j2\pi(-6)/4}))$$

بنابر این:

$$x[n]=\frac{1}{4}((x_0+x_1+x_2+x_3)+$$
$$(x_0e^{j\pi/2}+x_1+x_2e^{-j\pi)/2}+x_3e^{-j\pi})+$$
$$(x_0e^{j\pi}+x_1+x_2e^{-j\pi}+x_3e^{-j2\pi})+$$
$$(x_0e^{j\pi3/2}+x_1+x_2e^{-j\pi3/2}+x_3e^{-j\pi}))$$

سپس و بکمک فرمول اولر بجای نمایی ها، سینوسی ها را جایگزین می نماییم. در جدول زیر سینوسی ها محاسبه و مقادیر آن در عبارت فوق قرار داده می شود.

$$x[n] = \frac{1}{4}((x_0 + x_1 + x_2 + x_3) +$$
$$(x_0(\cos(\pi/2) + j\sin(\pi/2) + x_1 + x_2(\cos(\pi/2) + j\sin(\pi/2) + x_3(\cos(\pi) - j\sin(\pi))) +$$
$$(x_0(\cos(\pi) + j\sin(\pi) + x_1 + x_2(\cos(\pi) - j\sin(\pi/2) + x_3(\cos(2\pi) - j\sin(2\pi))) +$$
$$(x_0(\cos(3\pi/2) + j\sin(3\pi/2) + x_1 + x_2(\cos(3\pi/2) - j\sin(3\pi/2) + x_3(\cos(\pi) - j\sin(\pi))) +$$

cos(π/2)=0	sin(π/2) = 1
Cos(π)=-1	sin(π) = 0
Cos(2π)=1	sin(2π) = 0
Cos(3π/2)=0	sin(3π/2) = -1

$$x[n] = \frac{1}{4}((x_0 + x_1 + x_2 + x_3) +$$
$$(x_0(j) + x_1 + x_2(-j) + x_3(-1 - j0)) +$$
$$(x_0(-1 + j0) + x_1 + x_2(-1 - j0) + x_3(1 - j0)) +$$
$$(x_0(j(-1) + x_1 + x_2(-j(-1) + x_3(-1 - j0)))$$

با ساده نمودن عبارت قبل داریم:

$$x[n] = \frac{1}{4}((x_0 + x_1 + x_2 + x_3) +$$
$$(x_0(j) + x_1 + x_2(-j) + x_3(-1)) +$$
$$(x_0(-1) + x_1 + x_2(-1) + x_3(1)) +$$
$$(x_0(-j) + x_1 + x_2(j) + x_3(-1)))$$

با دسته بندی عبارات فوق عبارت زیر بدست می آید:

$$x[1] = \frac{1}{4}(x_0(1 + j - 1 - j) + x_1(1 + 1 + 1) + x_2(1 - j - 1 + j) + x_3(1 - 1 + 1 - 1))$$
$$x[1] = \frac{1}{4}(x_0(0) + x_1(4) + x_2(0) + x_3(0))$$
$$x[1] = x_1$$

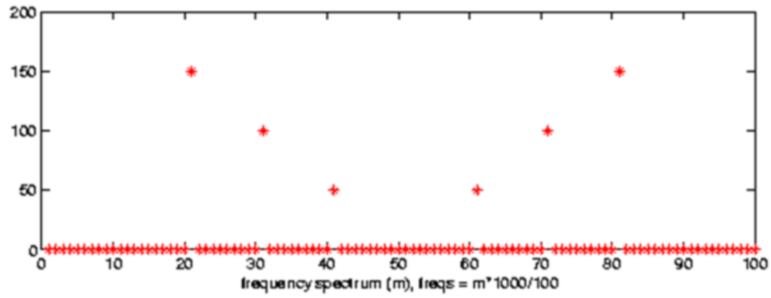
بنابراین تبدیل معکوس به ما داده هایی که در تبدیل مستقیم از آن استفاده کردیم را بر می گرداند. عبارت 1/N در تبدیل معکوس ظاهر می شود تا خروجی ها را به مقیاس در آورد و به مقادیر ورودی با بزرگای اولیه برگرداند. در اینجا عملیات برگرداندن داده هایی که تبدیل معکوس بر روی آنها انجام شده فقط بر روی اولین داده اجرا شد. دانشجویان می توانند مشابه این روند را برای دیگر داده ها مورد بررسی قرار دهند.

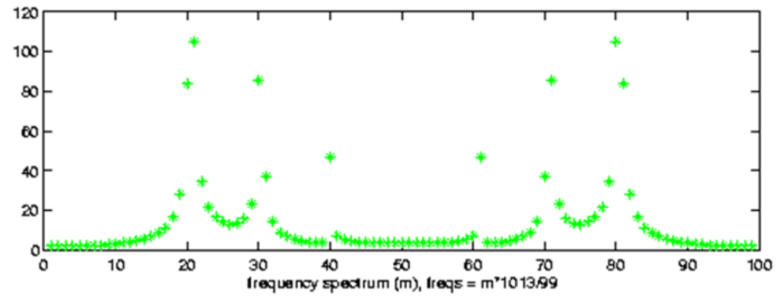
نشت(Leakage)

DFT وضوح بینهایت ندارد. پیامد این موضوع این است که بعضی وقت ها فرکانس هایی که در سیگنال حاضر هستند، در DFT بطور واضح مشخص نمی شوند و پاسخ بزرگای فرکانسی شان بنظر می رسد که در سطح چندین فرکانس تحلیلی پخش می شود. این پدیده به نشت معروف است. کد زیر این موضوع را مشخص می کند. در کد زیر ما دو سیگنال نمونه برداری شده را نمایش می دهیم: x1 و x2 . اگر به دقت نگاه کنیم مشاهده خواهید نمود که معادلاتی که این دو سیگنال را معرفی می کنند مشابه هم هستند. آنچه که ما تغییر می دهیم پارامترها هستند شامل فرکانس نمونه برداری (fs1 یا fs2) که دوره تناوب نمونه برداری را تغییر می دهد (Ts1 یا Ts2) و تعداد نمونه ها یی که می خوانیم (که به تعداد نمونه های n1 و n2 هستند). نتایج در نمودارهای زیر منعکس شده است. وقتی 100 نمونه می گیریم از سیگنال x با فرکانس 1000 نمونه در ثانیه، یک نمودار طیفی مشابه آنچیزی که در شکل 6.8 منعکس شده بدست می آید. این سیگنال نتایج را وقتی که فرکانس های تحلیلی با فرکانس حاضر در سیگنال جور باشند، نشان می دهد. شکل دوم همان سیگنال را نشان می دهد لیکن در اینجا تعداد نمونه برداری N و فرکانس نمونه برداری fs به 99 نمونه در 1013 نمونه در ثانیه تغییر می کند (شکل 6.9). چون

$$x(t) = 3\cos(2\pi 200t - 7\pi/8) + \cos(2\pi 300t) + 2\cos(2\pi 400t + \pi/4)$$

$$f_{analysis}[m] = mf_s / N$$

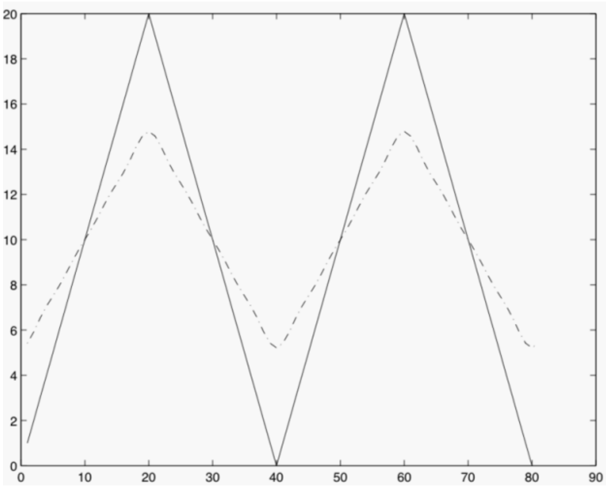




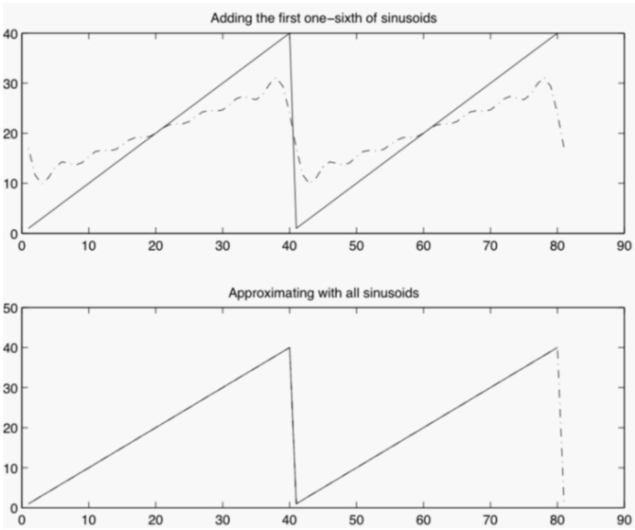
است، تغییر در هر یک (یا هر دو) پارامترها (یعنی پارامتر N و fs)، فرکانس تحلیلی را که در DFT استفاده می شود را متاثر می سازد. برای شکل دوم (6.9) فرکانس تحلیلی با فرکانس های سیگنال جور نیست بنابر این محتوی طیفی در طول تمام فرکانس ها پخش شده است. سیگنال تغییر نکرده ولی نگاهمان به اطلاعات سیگنال چرا. یک پنجره روشی است برای تغییر سیگنال ورودی طوری که پرش ناگهانی (ناپیوستگی) در آن وجود نداشته باشد. حتی اگر شما یک تابع پنجره استفاده نمی کنید، وقتی از یک سیگنال نمونه برداری می کنید، یک تابع پنجره به آن تحمیل می نمایید. به عبارت دیگر ورودی برای همه مقادیر قبل از آنکه نمونه برداری نمایید و آنهایی که بعد از نمونه برداری وجود دارند، 0 در نظر گرفته می شود. پرش های ناگهانی (ناپیوستگی ها) در پاسخ فرکانسی بصورت تابع سینک خود را نشان می دهند. نشت وقتی بوجود می آید که فرکانس های تحلیلی بر روی فرکانس واقعی موجود قرار نمی گیرند. این به معنی آن است که اطلاعات واقعی به دیگر دسته های خروجی DFT نشت پیدا می کند یعنی خود را بصورت دیگر فرکانس جا می زند. عمل پنجره لبه تابع سینک را (از تابع پیوسته فوریه (CFT)) کاهش می دهد که بنوبه خود باعث کاهش نشت می شود چون DFT یک نسخه نمونه برداری شده از یک تابع پیوسته است. به عبارت دیگر به CFT یک سیگنال نگاه کنید (یعنی یک سینوسی منفرد) تابع سینک را ملاحظه می کنیم. به یک CFT نسخه پنجره از همان سیگنال نگاه کنید یک سیگنال سینک با سطح لبه کمتر را ملاحظه می کنیم (گرچه لبه اصلی پهن تر است). اگر از CFT نمونه برداری کنیم ممکن است خوشحال باشیم و از آن درست در نقطه بین لبه های سینک نمونه برداری کنیم جاییکه دو یا سه قله و چندین مقادیر غیر صفر بخاطر لبه ها داریم. هر چه لبه های این سینک کمتر باشد، DFT ما فرکانس های بهتری که با مقادیر واقعی موجود جور باشند را نشان خواهند داد.

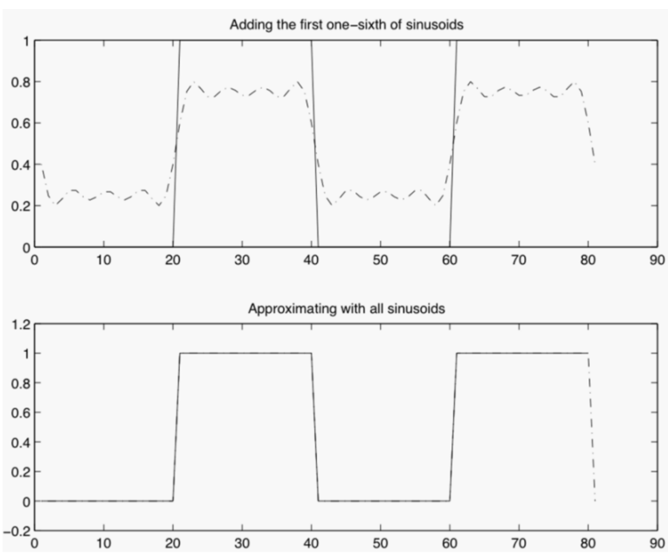
هارمونیک ها و تبدیل فوریه

هارمونیک ها و تبدیل فوریه بطور نزدیکی به هم وابسته هستند. هارمونیک ها سینوسی هایی هستند که به فرکانس اساسی وابسته و ضرائبی از آنها هستند نظیر $3f_0$, $2f_0$, f_0 و.... برنامه زیر هارمونیک ها را نشان می دهد و هدف آن تخمین یک موج مثلثی مرکب از سینوسی ها است. وقتی این برنامه اجرا می شود، ملاحظه می کنیم که تخمین ها با اضافه نمودن سینوسی ها بهتر و بهتر می شوند. از یک فرکانس اساسی یک هرتز استفاده می کنیم. بعد از ایجاد یک سیگنال x به صورت یک موج مثلثی، برنامه DFT آنرا بکمک دستور fft پیدا می کند. سپس بزرگاها و فازها را برای همه سینوسی ها مرتبط با نتایج بدست آمده از DFT محاسبه می شود. در نهایت برنامه مجموع سینوسی ها را نشان خواهد داد که با توقف بین اجرای هر لوپ، که خود معرف پیشرفت عملیات محاسبه است، همراه است. شکل 6.10 نتیجه را در دو وضعیت یکی در وضعیتی که 6/1 عملیات اجرا شده (با نقطه چین) و دیگری در پایان (که با خط بر مشخص است).



اولین چیزی که برنامه مبادرت به انجام آن می نماید ایجاد یک سیگنال است که با آن کار می کند: یک موج مثلثی. در زیر نشان می دهیم که چگونه برنامه با یک موج مربعی کار می کند. به انتهای لوپ نگاه کنید که در آن دستور پلات و توقف تعبیه شده است. می توانستیم بجای آن از دستور "plotharmonic" استفاده کنیم اما برنامه حاضر بازسازی یک سیگنال اولیه را با فرایند پله به پله انجام می دهد. بجای موج مثلثی در یک تمرین دیگر سیگنال زیر را بگنجانید. بهتر است بین ایندو دستور clear all تایپ شود. در شکل 6.11 روند تخمین یک موج اره ای را مشاهده می کنید. این شکل همچنین نتیجه را وقتی که تمامی عبارات مورد استفاده قرار گرفتند نشان می دهد. توجه کنید که بزرگترین تفاوت بین اولیه و مقدار تخمین زده شده نقطه پایانی است. سینوسی تخمینی فرض می کند که الگوی مورد نظر تکرار می شود. شکل 6.12 تخمین موج مربعی قبلی را در یک ششم سینوسی ها و نیز همه سینوسی ها نشان می دهد.

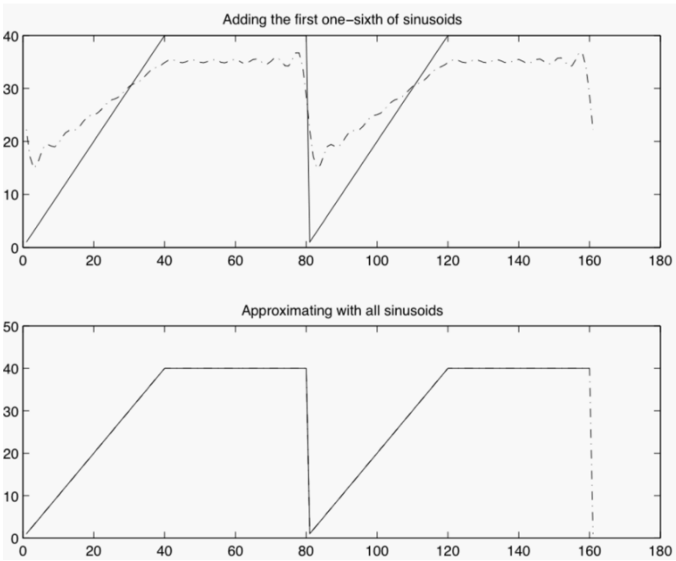




شکل 6.13 ترکیبی از موج اره ای و مربعی را در مرحله 6/1 و نهایی نشان می دهد. نسخه DFT سیگنال ها مطلوب هستند اما آنها کامل نیستند. در حوزه فرکانس ناپیوستگی ها در سیگنال نظیر فراز و نشیب تند و سریع به دشواری نشان داده می شود در حالیکه در حوزه زمان براحتی این کار عملی می شود. بعنوان مثال تابع ضربه:

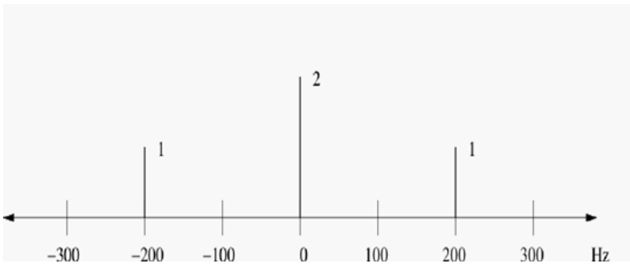
X(t) =1 , t=0

X(t) =0 , هر جای دیگر



اما برای نشان دادن این (تابع ضربه) در حوزه فرکانس به تعداد بینهایت سینوسی نیاز است. به این مشکل از طریق دیگر نگاه کنید برای این کار یک سینوسی ساده را در نظر بگیرید. در حوزه فرکانس بسادگی توسط یک قله "spike" نیم دامنه در دو فرکانس مثبت و منفی نشان داده می شود و برای همه فرکانس های دیگر مقادیر صفر است:

یادآوری فرمول معکوس اولر: Cos(φ)=(ejφ+e-jφ)/2



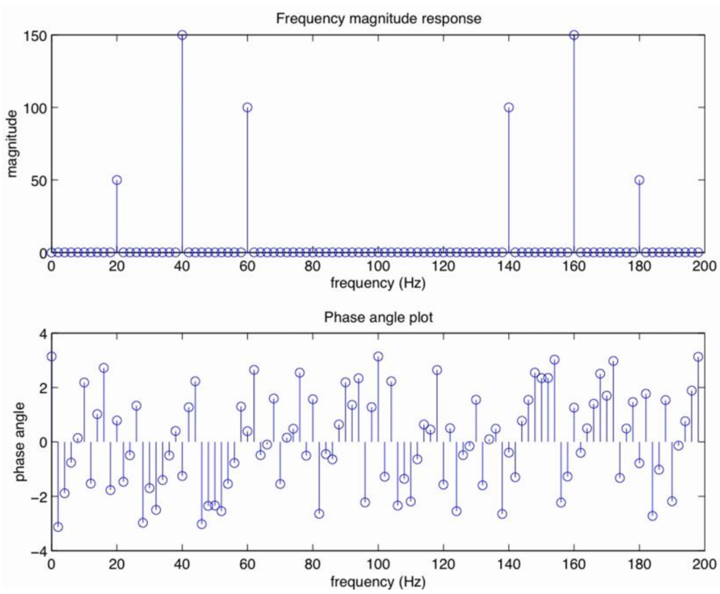
$x(t)=2+2\cos(2\pi(200)t)$
$=2+ e^{j2\pi(200)t} + e^{-j2\pi(200)t}$

اگر بخواهیم همین سیگنال را در زمان نشان بدهیم داریم: (دامنه *cos(2πft+φ)) که به ما مقدار تابع را در هر زمان ارائه می دهد. ثبت مقادیر این سیگنال برای هر زمان نیازمند نوشتن بینهایت عبارت است. این امر ما را به این سمت سوق می دهد که نتیجه بگیریم آنچه که در حوزه زمان بخوبی تعریف شده است، در حوزه فرکانس بگونه ی ضعیف نمودار می شود. همچنین آنچه که در حوزه فرکانس بخوبی قابل ارائه است در حوزه زمان به دشواری بیان می شود. در واقع این اصل عدم قطعیت ورنر هیسنبرگ است:

Werner Heisenberg's uncertainty principle

فرکانس نمونه برداری و طیف

همانطور که با DFT دیدیم وقتی از یک سیگنال واقعی با نرخ fs نمونه در ثانیه نمونه برداری می شود، نمودار بزرگا فرکانسی از fs/2 تا fs تصویر آینه ای 0 تا fs/2 است. یک سیگنال واقعی از مولفه هایی نظیر $\cos(\theta)$ ساخته شده و یا می توان آنها را به این صورت نوشت. اگر یک سیگنال مختلط بود یک مولفه $j\sin(\theta)$ یا چیزی مشابه آن نظیر $j\cos(\theta)$ باید وجود داشته باشد. چون: $\cos(\varphi)=(e^{j\varphi}+e^{-j\varphi})/2$ ، یک سیگنال واقعی همیشه مولفه فرکانسی مثبت و منفی بر روی نمودار طیف خواهد داشت.



با داشتن $\cos(\theta)=(e^{j\theta}+e^{-j\theta})/2$ و داشتن قانون اولر :

$$e^{j\theta} = \cos(\theta) + j\sin(\theta)$$

$$e^{j\theta} = (e^{j\theta} + e^{-j\theta})/2 + j\sin(\theta)$$

$$e^{j\theta}/2 = e^{-j\theta}/2 + j\sin(\theta)$$

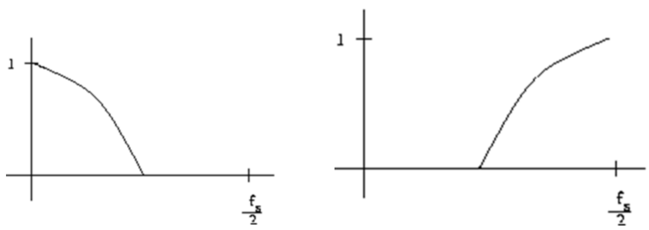
$$j\sin(\theta) = (e^{j\theta} - e^{-j\theta})/2$$

این به معنی آنست که سیگنال مختلط نیزدارای دو مولفه فرکانسی است. البته اگر ما یک سیگنال نظیر $\cos(\theta)+j\sin(\varphi)$ داشتیم :

$$\cos(\theta) + j\sin(\varphi)= e^{j\theta}/2+e^{-j\theta}/2 + e^{j\varphi}/2 - e^{-j\varphi}/2$$

اگر $\theta=\varphi$ باشد بنابر این: $\cos(\theta) + j\sin(\theta) = e^{j\theta}$

به این دلیل توجهمان را فقط به نیمه نخست پاسخ فرکانسی معطوف می نماییم. در اینصورت اگر فرکانس های مورد نظر فرکانس های کم باشند به فیلتر گذر پایین (شکل 6.14) و برعکس فیلتر گذر بالا (شکل 6.15) موسوم هستند. این به معنی آن است که هر مولفه فرکانس کم (تغییر آرام) بعد از آنکه سیستم بر سیگنال عمل می کند باقی می ماند. مولفه های بالا گذر کاهنده و یا فیلتر می شوند. وقتی که به صورت شکل 6.15 خود را نشان دهد به آن فیلتر بالاگذر گوییم.



پاسخ فرکانسی را می توان بعد از اجرای DFT بر روی خروجی سیستم وقتی تابع ضربه بصورت ورودی داده شود، یافت. این مورد به پاسخ ضربه ی سیستم موسوم است. در صورت تمایل به دریافت پاسخ ضربه با ظاهر صافتر تابع ضربه باید صفرگذاری شود zero padded.

خلاصه

این فصل به تبدیل فوریه و تبدیل معکوس فوریه و مباحث مرتبط پرداخته است. تبدیل فوریه محتوی (طیف) فرکانسی یک سیگنال را ارائه می کند. با داشتن هر سری از نمونه ها می توانیم مجموعه ای از سینوسی هایی که سیگنال را تخمین می زنند را داشته باشیم گرچه دقیقا خود سیگنال را نتوان بدست آورد.

موفق و پیروز باشید

پایان

تیرماه 1393