



تمرین سری دوم MATLAB

سیگنال ها و سیستم ها - نیمسال دوم 1399-1400

استاد: دکتر امینی

قبل از انجام تمرینات، به نکات زیر توجه فرمایید:

- یک M-File که شامل دستورات اولیه MATLAB است، در سایت درس قرار داده شده است. در صورت نیاز، آن را مطالعه نمایید.
- پاسخ تمرینات باید به صورت یک پوشه (فولدر) فشرده شده (zip یا rar) در CW قرار داده شود. این فایل باید شامل یک فایل متنی حاوی گزارش به صورت PDF، پاسخ سوالات، نمودارها و نتایج و M-File های نوشته شده در MATLAB باشد. خواسته های متن که در سوالات از شما خواسته شده باید در گزارش باشد.
- مرتب بودن گزارش و کامنت گذاری مناسب در کدها تاثیر مثبت در روند تصحیح دارد. تمام نمودارها باید با عنوان مناسب و محورهای نام گذاری شده باشند. برای تمایز بین نمودارهای سوالات مختلف، پیشنهاد می شود که شماره سوال را نیز در عنوان هر نمودار، بیاورید.
- برای هر مسئله، M-File جداگانه ای قرار دهید (در صورتی که تعداد فایل های سوال از یک فایل بیشتر بود، بهتر است آن ها را در فولدر جداگانه ای از بقیه سوالات بیاورید). هم چنین پیشنهاد می شود برای مرتب بودن کد، از توابع با نام های مناسب استفاده شود.
- پس از اجرای M-File باید تمام قسمت های سوال اجرا شود و همه ی شکل ها به صورت اتوماتیک و با عناوین مشخص و مناسب رسم شود. دقت شود که کسی برنامه ی شما را عیب یابی نخواهد کرد.
- نام پوشه فشرده شده به صورت Signal-HW1-XXXXXXX قرار داده شود که XXXXXXXX نشان دهنده ی شماره دانشجویی است.

برخی دستورهای اولیه مهم که راهنمای آن ها را به کمک help در MATLAB خواهید یافت و ممکن است در این تمرین کاربرد داشته باشد:

abs, plot, isnan, conv, snr, hold on, fft, fftshift

اگر در صورت تمرین، به اشکالی برخوردید، با عنوان ایمیل signals & systems به ایمیل زیر ارسال کنید.
دقت نمایید که پاسخ تمرین را به هیچ وجه به این ایمیل یا ایمیل استاد درس ارسال نکنید که به هیچ عنوان در نظر گرفته نمی شود.

heyezairsina1996@gmail.com

پردازش سیگنال

تبدیل فوریه

در متلب، برای محاسبه تبدیل فوریه، از دستور FFT استفاده می‌شود.

1. یک سیگنال کسینوسی در MATLAB ایجاد کرده و با استفاده از دستور FFT، تبدیل فوریه آن را حساب کنید. سیگنال را در حوزه زمان و اندازه و فاز آن را در حوزه فرکانس رسم کرده و نمودار به دست آمده را توجیه کنید.
2. تابع $X = \text{myFFT}(x, n)$ و $x = \text{myIFFT}(X, n)$ را بدون استفاده از loop بنویسید که `fft` و `ifft` یک سیگنال را محاسبه کند. (می‌توانید برای آشنایی با رابطه این توابع، از `help` متلب استفاده کنید. دستورات معادل در متلب، `fft` و `ifft` هستند)
3. سیگنال زیر را در نظر بگیرید:

$$x = \sin(2\pi f_1 t) + 0.1 \cos(2\pi f_2 t), \quad f_1 = 2, \quad f_2 = 10, \quad t = 0: \frac{1}{F_s}: 1, \quad F_s = 1000$$

- اندازه FFT سیگنال بالا را `plot` کنید.
- علت تقارن شکل حاصل را توضیح دهید.
- همان طور که در نمودار مشخص است، خروجی دستور FFT تقارن حول صفر را نشان نمی‌دهد برای آنکه به تقارن مرکزی (حول صفر) برسیم، از دستور `fftshift(fft(x))` استفاده کنید و با تنظیم محور `x` در بازه $-\frac{N}{2} : \frac{N}{2}$ ، نمودار مناسب را رسم کنید. (`N` تعداد نمونه‌ها (طول سیگنال) است)
- همان طور که در نمودار قبل دیدید، محور `x` به خوبی مقادیر فرکانس را نشان نمی‌دهد. برای رفع این مشکل، بردار `f` مناسبی را تعریف کرده و `plot(f, abs(fftshift(fft(x))))` را طوری رسم کنید که فرکانس‌های مناسب را در محور `x` نشان دهد. (برای تعریف بردار `f` کافی است برداری از $-F_s/2$ تا $F_s/2$ و هم‌اندازه با بردار زمان تعریف کنید)

پدیده گیبس

سیگنال زیر (متناوب با T) را در نظر بگیرید:

$$x(t) = \begin{cases} A; & -\frac{T_0}{2} < t < \frac{T_0}{2} \\ 0; & -\frac{T}{2} < t < \frac{T}{2} \\ 0; & \frac{T_0}{2} < t < \frac{T}{2} \end{cases}$$

برای $T = 8, T_0 = 2, A = 1$

- بیانی کلی برای ضرائب سری فوریه این سیگنال به صورت تحلیلی پیدا کرده و 100 ضریب اول آن را در MATLAB محاسبه کرده و اندازه دامنه و فاز تعدادی از آن‌ها را رسم کرده و مقایسه کنید.
- با استفاده از 1, 5, 10, 20, 50, 100 ضریب ابتدایی سری فوریه، سیگنال اولیه را بازسازی کرده و همه را روی یک نمودار رسم کنید.
- میانگین مربع خطا را بین سیگنال اصلی و سیگنال بازسازی شده، به ازای هر یک از صد مجموع جزئی ممکن (از 1 تا n برای n از 1 تا 100) محاسبه کرده و نمودار آن را رسم کنید.

پدیده گیبس را در این نمودارها توصیف کنید.

- تاثیر پارامترهای A, T, T_0 را در پدیده گیبس چطور ارزیابی می‌کنید؟ بدین منظور می‌توانید هم میانگین مربع خطا را مقایسه کنید و یا برای 100 ضریب اول، پارامترها را تغییر داده و مقایسه کنید. برای تغییر پارامترها، کافی است هر بار دو پارامتر را ثابت در نظر بگیرید و پارامتر سوم را تغییر دهید.
- تابعی با نام GibbsApprox بنویسید که در ورودی سیگنال را در یک دوره تناوب و عدد P را بگیرد و خروجی آن، عدد k باشد. طوری که k حداقل تعداد ضرایب موردنیاز برای رسیدن به تقریب P درصد انرژی را برای سیگنال ورودی نشان دهد. برای سیگنال داده شده، این تابع را با چند مقدار دلخواه P اجرا کنید و نتایج را در گزارش بیاورید.

فیلتر

در این سوال، به طراحی یک فیلتر ساده می‌پردازیم. برای شهود بهتر، می‌توانید فیلتر را به صورت یک سیستم ببینید. (در این سوال، منظور از رسم در حوزه فرکانس، رسم اندازه بردار است)

همان طور که در تمرین دیده‌اید، در MATLAB، سیگنال پیوسته‌ای نداشته و برای تحلیل و کار با سیگنال‌های پیوسته، آن‌ها را با فرکانس بالایی (چندین مرتبه بالاتر از فرکانس‌های عادی خود سیگنال) نمونه‌برداری می‌کنیم. فرکانس نمونه‌برداری همان پارامتر F_s است که در تمرین قبلی دیدید.

1. یک بازه زمانی از 100- تا 100 و با فرکانس نمونه‌برداری 100 تعریف کنید.

(برای این کار، در MATLAB دستور زیر را استفاده می‌کنیم)

$$t = -100:\frac{1}{F_s}:100$$

2. تابع دلتای $x(t)$ را تعریف کنید. (توجه شود که با توجه به نوع تعریف زمان، باید عنصر میانی 1 باشد)

3. بردار f مناسب را که نمایانگر فرکانس‌ها باشد، تعریف کرده و $x(t)$ را در حوزه فرکانس به صورت مناسب در یک نمودار نشان دهید (معمولاً هنگام تبدیل حوزه زمان به فرکانس در MATLAB، دستور `fftshift` استفاده می‌شود تا نمودار متقارن حول صفر باشد).

4. با توجه به دلتا بودن تابع، انتظار می‌رفت در حوزه فرکانس، تابع ثابت 1 را داشته باشیم. با زوم کردن بر روی نمودار، صحت این ادعا را بررسی کرده و در مورد آن توضیح دهید.

5. سه فیلتر زیر را تعریف کنید.

$$h1 = e^{-10t}u(t);$$

$$h2 = e^{-t}u(t);$$

$$h3 = e^{-0.1t}u(t);$$

در تعریف فیلتر اول توجه کنید که با توجه به شدت نزولی بودن این تابع ممکن است از جایی به بعد به جای صفر، `nan` گذاشته شود. در این حالت، با دستور `isnan`، `nan`ها را در بردار پیدا کرده و با صفر جایگزین کنید (سعی کنید بدون استفاده از `loop` این کار را انجام دهید). لازم به ذکر است که معمولاً وقتی حتی یک یا تعدادی از عناصر یک بردار یا ماتریس، `nan` می‌شوند، در عملیات مختلف بر روی آن متغیر، `nan` در سطح کل بردار یا ماتریس خروجی پخش شده و هنگام رسم نمودار یک صفحه کاملاً سفید و بدون هیچ نقطه‌ای می‌بینید.

6. سه فیلتر قسمت قبل را رسم کرده و آن‌ها را مقایسه کنید.

7. خروجی $y3$ را با استفاده از دستور `conv` و با اعمال فیلتر $h3$ بر روی x ، بدست آورید. آن را در حوزه زمان و فرکانس رسم کرده و در مورد نمودار توضیح دهید. (خروجی حوزه فرکانس را $Y3$ بنامید)

8. همان طور که می‌دانید، می‌توان به جای کانولوشن در حوزه زمان، از ضرب در حوزه فرکانس استفاده کرد. x و $h3$ را به حوزه فرکانس برده و $Y3p = X*H3$ را بدست آورید. (برای بردن تابع $h3$ به حوزه فرکانس، از تبدیل زیر می‌توانید استفاده کنید)

$$h_3 = e^{-\alpha t} u(t) \xrightarrow{\text{fourier transform}} H_3 = \frac{1}{j2\pi f + \alpha};$$

9. Y_3 و Y_{3p} (نتیجه قسمت 7) را مقایسه کرده و در صورت تفاوت، علت آن را توضیح دهید.

پردازش سیگنال صوتی

1. سیگنال سینوسی، تک فرکانس

برای بیشتر فایل‌های صوتی که معمولا با آن‌ها کار می‌کنیم F_s عددی در حدود 40000 تا 50000 نمونه بر ثانیه است. (اعدادی مثل 40000 و 48000 بسیار مرسوم هستند.)

یک سیگنال کسینوسی به مدت زمان T ، با فرکانس f ، فرکانس نمونه‌برداری F_s و با دامنه A را در MATLAB به صورت زیر می‌توان تعریف کرد.

$$t = 0:\frac{1}{E_S}:T;$$

$$x = A\cos(2\pi ft);$$

با پارامترهای زیر، چند سیگنال تولید کنید.

$$Fs = 40000, \quad T = 10, \quad A = 1,$$

$$f = 100, 200, 500, 1000, 2000, 4000;$$

با استفاده از دستور **sound**، هر یک از این سیگنال‌ها را پخش کنید. با افزایش فرکانس، چه اتفاقی می‌افتد؟ نتیجه را در گزارش خود بیاورید. (در استفاده از دستور **sound**، فراموش نکنید که پارامتر **Fs** را به عنوان ورودی تابع، مشخص کنید.)

اگر دقت شود، علی رغم آنکه دامنه همه این سیگنال‌ها یکسان است، آنچه که گوش از بلندی صدا احساس می‌کند متفاوت است. در مورد علت این قضیه، در گزارش خود توضیح دهید. می‌توانید از لینک‌های زیر استفاده کنید.

<http://umdborg.pbworks.com/w/page/131541489/Frequency%20response%20of%20the%20human%20ear>

<https://www.widex.com/en/blog/global/human-hearing-range-what-can-you-hear/#:~:text=While%2020%20to%2020%2C000Hz,2000%20%2D%205000%20Hz%20frequency%20range.>

با همان پارامترها و با $f=1000$ ، سیگنالی را درست کرده و یک بار ورودی فرکانس نمونه‌برداری را در تابع *sound* به جای FS ، $\frac{FS}{2}$ و بار دیگر $2FS$ بدهید. این دو حالت را مقایسه کرده و نتیجه را در گزارش خود بیاورید.

2. در کنار صورت تمرین، یک فایل صوتی با نام test.mp3 قرار داده شده است. با استفاده از دستور audioread، این فایل صوتی را در MATLAB به همراه Fs آن، بخوانید.

با توجه به طولانی بودن سیگنال، در این تمرین در صورت تمایل می‌توانید 500,000 نمونه اول را در نظر بگیرید. همان گونه که مشاهده می‌شود، این سیگنال به جای یک بردار، یک ماتریس است. معمولاً فایل‌های صوتی، از چند کانال تشکیل می‌شوند که تعداد کانال‌ها 1 یا 2 و بیشتر هم می‌تواند باشد. (در هر قسمتی که لازم باشد می‌توانید صرفاً کانال اول را در نظر بگیرید. خصوصاً در رسم نمودارها، پیشنهاد می‌شود این کار را انجام دهید)

- یک بار با فرکانس $2F_s$ و یک بار با فرکانس $F_s/2$ ، سیگنال را پخش کرده و با سیگنال اصلی مقایسه کنید و نتیجه را در گزارش خود بیاورید.
- سیگنال y_1 را با ضرب سیگنال اصلی در 5، ایجاد کرده و آن را پخش کنید. در مورد علت ایجاد صدای اضافه‌ای که شبیه به نویز هست، توضیح دهید.¹
- یک سیگنال نویز با میانگین 0 و واریانس 0.04 و هم‌سایز سیگنال اصلی ایجاد کرده و آن را با سیگنال اصلی جمع کنید. نمودار حاصل را رسم کنید. سیگنال حاصل را پخش کنید. در چه مواقعی، اثر نویز بیشتر محسوس است؟
- سیگنال اصلی را با مقدار $DC = 0.5$ جمع کرده و آن را پخش کنید. در مورد تغییر اتفاق افتاده و علت آن در گزارش خود توضیح دهید.
- با مقادیر $std = 0.1:1$ ، نویزهایی گوسی با میانگین صفر ساخته و هر بار یکی از آن‌ها را با سیگنال اصلی جمع کرده و مقدار snr را برای هر یک از حالات حساب کنید و نموداری از مقادیر snr رسم کنید. در همان نمودار بردار snr موردانتظار خود را نیز رسم کرده و در مورد علت تفاوت آن دو، توضیح دهید. (هنگام استفاده از تابع snr ، نویز یا سیگنال اصلی را به صورت جداگانه به آن ندهید و فقط سیگنال حاصل جمع (نویز و سیگنال اصلی) را به تابع بدهید (در این حالت، تابع فقط یک ورودی دارد))

¹ [https://en.wikipedia.org/wiki/Clipping_\(audio\)](https://en.wikipedia.org/wiki/Clipping_(audio))