

به نام خدا



دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر



سیگنال و سیستم

تمرین کامپیوتری ۱

استاد: دکتر صدف صالح کلیبر

آذر ماه ۱۳۹۶

نکات زیر را در مورد گزارش تمرین کامپیوتری و کدهای ارسالی در نظر داشته باشید:

- یک گزارش با فرمت *student_number.pdf* که به جای *student_number* شماره دانشجویی قرار می‌گیرد، آماده کنید.
- گزارش فقط به فرمت *PDF* مورد نیاز است و نیازی به ارسال فایل *word* نیست.
- به همراه فایل گزارش پوشه‌ای به نام *Codes* ایجاد کنید که کد مربوط به هر سؤال به تفکیک در آن آورده شده باشد. لازم است تا هر سؤال در فایل جداگانه‌ای پیاده‌سازی شود. فایل‌های برنامه‌های مربوط به هر سؤال را در یک *m-file* طبق الگوی *student_number_En.m* نام گذاری کنید، که به جای *student_number* شماره دانشجویی و به جای *En* شماره سؤال بصورت *E1*، *E2* و ... قرار می‌گیرد.
- هیچ چیز دیگری، مانند شکل‌ها، نمودارها و ... به صورت جداگانه ارسال نشود و همه شکل‌ها باید در گزارش با عنوان و شماره مناسب آورده شوند و تمرین باید فقط حاوی کدها و گزارشی کامل باشد.
- بخش قابل توجهی از نمره تمرین شما، مربوط به گزارش است. در صورتی که فایل بارگذاری شده پاسخ تمرین، به هر دلیلی فاقد گزارش باشد، نمره تمرین صفر منظور شده و هیچ اعتراضی وارد نیست.
- در انتها، فایل گزارش و پوشه *Codes* را تحت یک فایل فشرده با نام *student_number.rar* یا *student_number.zip* ارسال کنید که به جای *student_number* شماره دانشجویی قرار می‌گیرد.

در تمام سؤال‌ها راهنمایی‌های لازم آورده شده است ولی در صورت ابهام سؤالات خود را با ایمیل

sattari93@yahoo.com در میان بگذارید.

سؤال ۱

سیگنال‌های زیر را در نظر بگیرید:

$$x[n] = \delta[n] + 2\delta[n-1] - \delta[n-3] + 3\delta[n-4] - \delta[n-5]$$

$$h[n] = 2\delta[n] + 2\delta[n-2]$$

الف) با رعایت نکات زیر برنامه ای در MATLAB بنویسید که سیگنال‌های زیر را محاسبه و رسم کند.

$$y_1[n] = x[n] * h[n]$$

$$y_2[n] = x[n-2] * h[n]$$

- خروجی هر یک از کانولوشن‌ها را در بازه $n=0,1,\dots,9$ کنید.
- از دستور *subplot* برای رسم هر دو خروجی $y_1[n]$ و $y_2[n]$ در یک *figure* استفاده کنید.
- برای رسم هر یک از سیگنال‌ها از دستور *stem* استفاده کنید.
- محور x را n (دستور *xlabel*) و محور y را نام سیگنال (دستور *ylabel*) نام گذاری کنید.
- برای محاسبه کانولوشن می‌توانید از دستور *conv* در MATLAB استفاده کنید.

راهنمایی: سیگنال $x[n]$ را می‌توان به صورت یک بردار در محیط MATLAB پیاده کرد:

$$x=[1,2,0,-1,3,-1]$$

همچنین برای اینکه خروجی کانولوشن‌ها به ازای $n=0,1,\dots,9$ رسم کنید، لازم است در آخر سیگنال $x[n]$ صفر اضافه کنید تا بردارهای n و $x[n]$ دارای طول یکسانی باشند. برای مثال برای محاسبه $y_1[n]$ بردار x را به صورت زیر تعریف کنید:

$$x=[1,2,0,-1,3,-1,0,0]$$

ب) با توجه به شکل‌های رسم شده در قسمت الف، درباره متغیر با زمان بودن سیستم بحث کنید.

سؤال ۲

در این سؤال به کاربرد فیلترهای خطی برای حذف نویز از یک سیگنال پرداخته می‌شود. سیگنال گسسته زمان زیر را در نظر بگیرید:

$$x[n] = s[n] + z[n], n = \dots, -2, -1, 0, 1, 2, \dots$$

که $s[n]$ یک سیگنال نامشخص و $z[n]$ سیگنال نویز تصادفی می‌باشد. فرض کنید به ازای هر $n = 0, 1, \dots, N$ به نمونه آخر مشاهده شده از $x[n]$ ، $(x[n], \dots, x[n - N + 1])$ دسترسی داریم. از این نمونه-های مشاهده شده به عنوان ورودی یک فیلتر *moving-average (MA)* استفاده می‌شود که به صورت زیر تعریف می‌شود:

$$y[n] = \sum_{i=0}^{N-1} a_i x[n-i]$$

که a_i ضرایب فیلتر هستند. در این سؤال از *exponentially weighted moving-average (EWMA)* استفاده می‌شود که به صورت زیر تعریف می‌شود:

$$y[n] = \frac{1-b}{1-b^N} \sum_{i=0}^{N-1} b^i x[n-i]$$

که $0 < b < 1$ و در نتیجه :

$$a_i = \frac{1-b}{1-b^N} b^i, i = 0, \dots, N-1$$

حال سیگنال زیر را در نظر بگیرید.

$$s[n] = 5 \sin\left(\frac{2\pi n}{20} + 3\right) u[n]$$

که $u[n]$ سیگنال گسسته زمان پله واحد است. نویز $z[n]$ به صورت یک نویز گوسی با واریانس ۱ در نظر می‌گیریم. برای تولید یک دنباله تصادفی از این نویز به طول m از دستور $randn(1, m)$ در MATLAB استفاده کنید. توجه داشته باشید که مبحث سیگنال‌های تصادفی خارج از مباحث درس است و نیازی به مطالب عمیق تر راجع به سیگنال‌ها و نویز تصادفی نیست و در این سؤال کافیت با همین دستور یک دنباله از اعداد تصادفی را با سیگنال $s[n]$ جمع کنید.

الف) سیگنال نویزی $x[n]$ را به ازای $n = 0, 1, \dots, 60$ رسم کنید. برای محاسبه و رسم $x[n]$ کافیت که به ازای $n = 0, 1, \dots, 60$ ابتدا سیگنال $s[n]$ را طبق رابطه بالا محاسبه کرده و با دنباله‌ای از اعداد تصادفی جمع کنید تا سیگنال $x[n]$ به دست آید و سپس از دستور $stem$ برای رسم این سیگنال در محیط MATLAB استفاده کنید.

ب) فرض کنید $b=0.2$ و $N=3$ و در $n=0$ فیلتر $EWMA$ با مقادیر $x[-1] = \dots = x[-N] = 0$ مقداردهی اولیه شده است (مقدار $x[n]$ به ازای $n < 0$ برابر صفر است). خروجی فیلتر را به ازای $n = 0, 1, \dots, 60$ رسم کنید. برای این کار کافیتست مطابق رابطه بالا و نمونه‌های سیگنال $x[n]$ که در قسمت الف محاسبه کردید، به ازای هر $n = 0, 1, \dots, 60$ نمونه‌های $y[n]$ را به دست آورید.

ج) قسمت ب را به ازای $b=0.2$ و $N=10$ تکرار کنید.

د) خروجی فیلترها در قسمت‌های ب و ج را با سیگنال بدون نویز $s[n]$ مقایسه کنید.

ه) تأثیر افزایش مقدار پارامترهای فیلتر (b و N) را توضیح دهید.

سؤال ۳

در این سؤال با استفاده از تبدیل فوریه با حوزه فرکانس سیگنال‌ها در محیط MATLAB آشنا می‌شویم.

پارامترهای زیر برای تولید سیگنال $s(t)$ در محیط MATLAB در نظر بگیرید:

- $F_s = 1000$ Sampling frequency
- $T = 1/F_s$ Sampling period
- $L = 1000$ Length of signal
- $t = (0:L-1) \times T$ Time vector

$$s(t) = 0.7\sin(2\pi \times 50t) + \sin(2\pi \times 360t)$$

مشابه سؤال ۲ با تولید نویز گوسی تصادفی سیگنال نویزی $x(t)$ را تولید کنید:

$$x(t) = s(t) + 2 * \text{randn}(\text{size}(t))$$

الف) سیگنال نویزی $x(t)$ را در حوزه زمان رسم کنید. برای اینکه محور x در دستور `plot` MATLAB

بیانگر زمان بر حسب میلی ثانیه باشد (نه نمونه‌های سیگنال) از دستور زیر برای `plot` ۵۰ نمونه‌ی سیگنال

استفاده کنید:

```
plot(1000*t(1:50),x(1:50))
```

```
title('Noisy Signal')
```

```
xlabel('t (millisecond)')
```

```
ylabel('x(t)')
```

ب) با استفاده از دستور `fft` تبدیل فوریه سیگنال را به صورت زیر محاسبه کنید:

$$Y = \text{fft}(x)$$

دستور *fft* در MATLAB بیانگر *Fast Fourier Transform* می‌باشد. *FFT* یک الگوریتم سریع برای محاسبه تبدیل فوریه سیگنال‌های گسسته-زمان می‌باشد. دانشجویان علاقه مند می‌توانند توضیحات بیشتر راجع به این الگوریتم را در کتاب‌های مربوط به پردازش سیگنال‌های گسسته (*DSP*) مطالعه کنند.

حال طیف دو طرفه P_2 (*two-sided spectrum*) را با دستور زیر محاسبه کنید:

$$P_2 = \text{abs}(Y/L)$$

و طیف یک طرفه را بر حسب نمونه‌های P_2 با دستورهایی زیر به دست آورید:

$$P_1 = P_2(1:L/2+1)$$

$$P_1(2:\text{end}-1) = 2 * P_1(2:\text{end}-1)$$

توجه داشته باشید که علت محاسبه طیف یک طرفه به علت حقیقی بودن سیگنال $x(t)$ است، چرا که بر طبق تئوری مربوط به خواص تبدیل فوریه، طیف مربوط به یک سیگنال حقیقی دارای خاصیت تقارن می‌باشد.

طیف یک طرفه P_1 را در محیط MATLAB رسم کنید. توجه داشته باشید برای اینکه محور x بیانگر فرکانس باشد از دستور زیر استفاده کنید:

$$f = F_s * (0:(L/2))/L$$

$$\text{plot}(f, P_1)$$

$$\text{title}('Single-Sided Amplitude Spectrum of x(t)')$$

$$\text{xlabel}('f(\text{Hz})')$$

$$\text{ylabel}('|P_1(f)|')$$

چرا اندازه‌های مربوطه در فرکانس‌های سیگنال $x(t)$ دقیقاً 0.7 (در فرکانس 50 Hz) و 1 (در فرکانس 360 Hz) نیستند؟

ج) طیف یک طرفه مربوط به سیگنال بدون نویز ($s(t)$) را مشابه قسمت ب رسم کنید و با طیف رسم شده در قسمت قبل مقایسه کنید.

د) در این قسمت با استفاده از ابزار *fdatool* در محیط MATLAB یکی از فرکانس‌های مربوط به سیگنال $x(t)$ را حذف می‌کنیم. برای استفاده از این ابزار کافیست عبارت *fdatool* در Command Window محیط MATLAB تایپ کنید و دکمه *Enter* را فشار دهید تا پنجره مربوط به طراحی فیلتر باز

شود و یا می‌توان در زبانه *App* در قسمت *Signal Processing and Communication* گزینه *Filter* *design and analysis* را انتخاب کنید. در پنجره باز شده، گزینه‌های مختلفی مانند *Response Type*، *Design Method*، *Frequency Specifications* و ... برای طراحی مناسب فیلتر وجود دارد. با انتخاب مناسب این پارامترها فیلتری طراحی کنید که فرکانس 50 Hz را از سیگنال $x(t)$ حذف کند و پاسخ زمانی و فرکانسی سیگنال فیلتر شده را رسم کنید.

راهنمایی: برای حذف فرکانس 50 Hz می‌توانید از یک فیلتر *bandstop* و مقدار F_s را همان مقدار تعیین شده در سؤال قرار دهید. سایر پارامترها به جز F_{pass} و F_{stop} را می‌توان بدون تغییر مقادیر اولیه استفاده شوند. دقت کنید که مقادیر F_{pass} و F_{stop} باید به درستی انتخاب شوند تا فقط یکی از فرکانس‌ها از سیگنال حذف شود. حال برای استفاده از فیلتر طراحی شده از منوی *File* گزینه *Generate Matlab Code* و سپس *Filter Design Function* را انتخاب کنید و در مسیر فولدر مربوط به کدهای تمرین خود به نام *Filter_Response* ذخیره کنید. و با استفاده از دستور زیر سیگنال فیلتر شده را به دست بیاورید:

$$\text{Filtered_Signal} = \text{filter}(\text{Filter_Response}, x)$$

توجه کنید که *m-file* مربوط به پاسخ فیلتر را در پوشه مربوط به کدها نیز ارسال کنید.