

به نام خدا



دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

تمرین کامپیوتری ۳

مخابرات دیجیتال

دکتر ربیعی

عرفان پناهی ۸۱۰۱۹۸۳۶۹

نیمسال دوم ۱۴۰۰-۰۱

فهرست:

*** فایل مربوط به این تمرین کامپیوتری با نام CA3_panahi.mlx پیوست شده است.

بخش ۱ صفحه ۲ ([لینک](#))

بخش ۲ صفحه ۴ ([لینک](#))

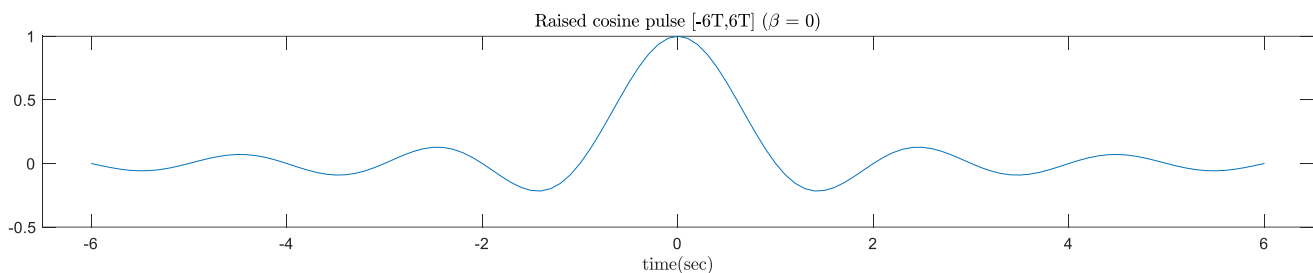
بخش ۱: تولید و ارسال سمبل ها با استفاده از مدولاسیون 4-PAM

* تولید پالس: پالس Raised-cosine را با داشتن β به صورت زیر تعریف می شود:

$$p(t) = \begin{cases} \frac{\pi}{4T} \operatorname{sinc}\left(\frac{1}{2\beta}\right) & , \quad t = \pm \frac{T}{2\beta} \\ \frac{1}{T} \operatorname{sinc}\left(\frac{t}{T}\right) \frac{\cos\left(\frac{\pi\beta t}{T}\right)}{1 - \left(\frac{\pi\beta t}{T}\right)^2} & , \quad o.w. \end{cases}$$

در ابتدا با استفاده از مقادیر اولیه فرکانس نمونه برداری و بازه تعریف پالس، بردار زمانی را در سه حالت خطای نمونه برداری تعریف می کنیم. مطابق با تعریف بالا پالس را برای $\beta = 0$ تعریف می کنیم. تصویر ۱ پالس به ازای $\beta = 0$ را نشان می دهد.

(تصویر از تمرین کامپیوتری ۲)



تصویر ۱: رسم پالس Raised-cosine به ازای $\beta = 0$

* تولید سیگنال ارسالی: در این قسمت ابتدا با استفاده از دستور *rand* یک بردار تصادفی یکنواخت بین 0 و 1 به طول ۱

میلیون تولید می کنیم و برای مادل کردن، با توجه به احتمالات سمبل ها به شکل زیر عمل می کنیم:

$$0 < \text{random} < 0.1 \rightarrow A_m = -3 \text{ (A)}$$

$$0.1 \leq \text{random} < 0.5 \rightarrow A_m = -1 \text{ (B)}$$

$$0.5 \leq \text{random} < 0.9 \rightarrow A_m = 3 \text{ (C)}$$

$$0.9 \leq \text{random} < 1 \rightarrow A_m = 3 \text{ (D)}$$

سپس بردار بدست آمده را *upsample* می کنیم تا بتوان هر دامنه را با یک ضربه مدل کرد و بتوان بک پالس را در محل هر

سمبل قرار داد. در نهایت بردار *upsample* شده را در پالس قسمت قبل کانالو می کنیم تا سیگنال ارسالی ساخته شود.

* **مدلسازی کانال AWGN**: در این قسمت ابتدا یک بردار از 0 تا 10 به عنوان $SNR (dB)$ تعریف می‌کنیم. سپس با استفاده از رابطه $SNR = 10^{\frac{SNR_{dB}}{10}}$ مقدار SNR را بدست می‌آوریم. انرژی مورد نیاز برای ارسال هر سمبل به صورت زیر محاسبه می‌شود.

$$E_s = E\{A_k^2\} = 0.1 \times (-3)^2 + 0.4 \times (-1)^2 + 0.4 \times 1^2 + 0.1 \times 3^2 = 2.6$$

در صورتی که بخواهیم به طور دقیق تر E_s را بدست آوریم باید از سمبل های تولید شده برای محاسبه انرژی مورد نیازشان استفاده کنیم. به این ترتیب به صورت زیر عمل می‌کنیم:

$$E_s = \frac{E_{total}}{N_s} = \frac{\sum(X_T^2)}{N_s}$$

که برای پیاده سازی آن در متلب مطابق با کد زیر عمل می‌کنیم:

```
Es = sum(modulated_symbols.^2) / N
```

```
Es = 2.6036
```

تصویر ۲: محاسبه انرژی مورد نیاز برای ارسال هر سمبل E_s

سپس به صورت زیر η را بدست می‌آوریم.

$$SNR = \frac{E_s}{\eta} \rightarrow \eta = \frac{E_s}{SNR}$$

حال به منظور تولید نویز یک بردار تصادفی نرمال $N(0,1)$ به طول سیگنال ارسالی تولید می‌کنیم. سپس برای اینکه به واریانس

$\frac{\eta}{2}$ شود، هر نویز را در $\sqrt{\frac{\eta}{2}}$ ضرب می‌کنیم. (در کل ۱۱ تا η مختلف داریم.) در نهایت نویز بدست آمده را به سیگنال ارسالی قسمت

قبل اضافه می‌کنیم. (یعنی در کل ۱۱ سیگنال دریافتی داریم و می‌خواهیم این سیگنال ها را بررسی کنیم.)

بخش ۲: آشکارسازی سمبل ها با استفاده از گیرنده های ML و MAP

در این قسمت مطابق با تمرین کامپیوتری سری قبل ، ابتدا اندیس هایی که می خواهیم از آن ها نمونه برداری کنیم را به صورت زیر تعریف می کنیم:

$$T_sampling = 6*L+1:L:(N+6-1)*L+1;$$

در حقیقت به جای *upsample* کردن در فرستنده ، در گیرنده سیگنال دریافتی را *downsample* می کنیم. سپس با استفاده از اندیس های تعریف شده ، سیگنال دریافتی را نمونه برداری می کنیم.

حال برای هر کدام از گیرنده های *MAP* و *ML* ، سطوح آستانه تصمیم گیری را محاسبه می کنیم:

* **گیرنده MAP**: برای محاسبه سطح آستانه ، به صورت زیر عمل می کنیم:

$$\sigma^2 = N_0 = \int_{-\infty}^{+\infty} G_{n_w}(f) df = \int_{-\frac{1}{2T_s}}^{\frac{1}{2T_s}} \frac{\eta}{2} df = \frac{\eta}{2} \times \frac{2}{2T_s} = \frac{\eta}{2T_s} \xrightarrow{T_s=1} \sigma^2 = \frac{\eta}{2}$$

$$f(n) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{n^2}{2\sigma^2}}$$

$$\left\{ \begin{array}{l} \Phi_A(y) = f(y+3)P_A = \frac{0.1}{\sigma\sqrt{2\pi}} e^{-\frac{(y+3)^2}{2\sigma^2}} \\ \Phi_B(y) = f(y+1)P_B = \frac{0.4}{\sigma\sqrt{2\pi}} e^{-\frac{(y+1)^2}{2\sigma^2}} \\ \Phi_C(y) = f(y-1)P_C = \frac{0.4}{\sigma\sqrt{2\pi}} e^{-\frac{(y-1)^2}{2\sigma^2}} \\ \Phi_D(y) = f(y-3)P_D = \frac{0.1}{\sigma\sqrt{2\pi}} e^{-\frac{(y-3)^2}{2\sigma^2}} \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \Phi_A(y) = \Phi_B(y) \rightarrow y = -2 - \sigma^2 \ln(2) \rightarrow \Delta_1 = -2 - \sigma^2 \ln(2) \\ \Phi_B(y) = \Phi_C(y) \rightarrow y = 0 \rightarrow \Delta_2 = 0 \\ \Phi_C(y) = \Phi_D(y) \rightarrow y = -2 - \sigma^2 \ln(2) \rightarrow \Delta_3 = 2 + \sigma^2 \ln(2) \end{array} \right.$$

* **گیرنده ML**: با توجه به اینکه در این گیرنده احتمالات هر سمبل را نمی دانیم، تمامی $\Phi(y)$ ها به یک شکل خواهند بود و در

نتیجه سطوح آستانه وسط دامنه مربوط به سمبل ها قرار می گیرند.

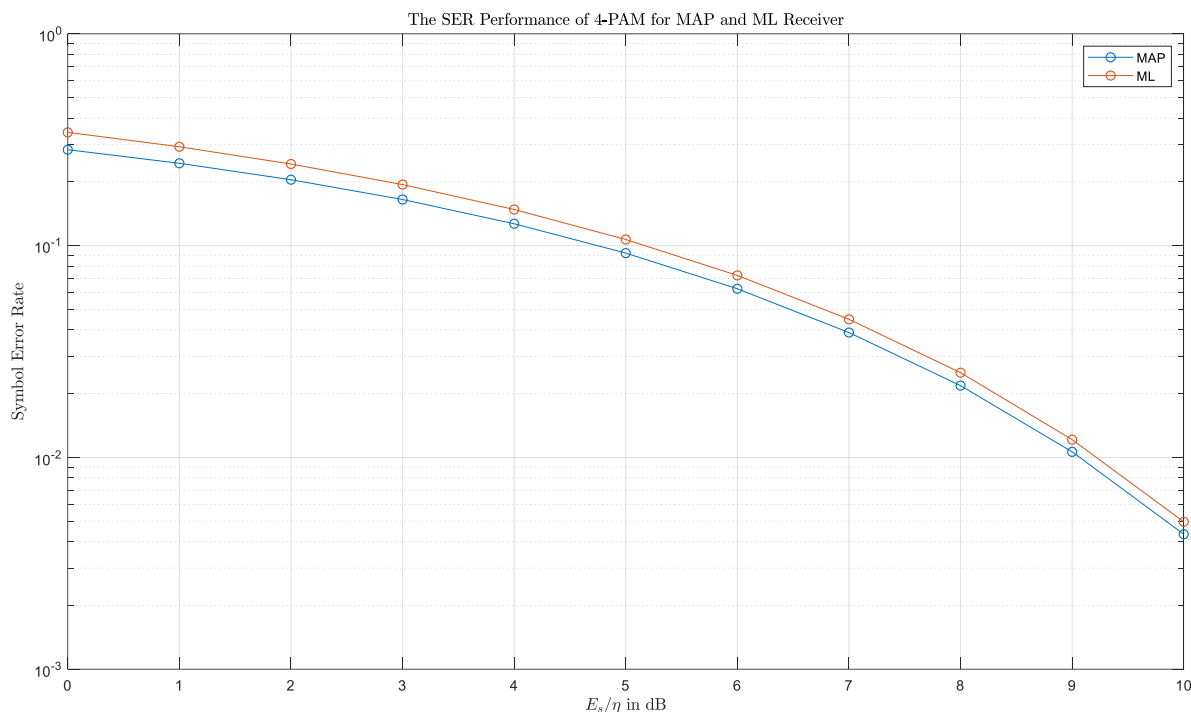
$$\Delta_l = \frac{A^{(l)} + A^{(l+1)}}{2} \Rightarrow \left\{ \begin{array}{l} \Delta_1 = \frac{-3A - A}{2} = -2A \rightarrow \Delta_1 = -2 \\ \Delta_2 = \frac{-A + A}{2} = 0 \rightarrow \Delta_2 = 0 \\ \Delta_3 = \frac{3A + A}{2} = 2A \rightarrow \Delta_3 = 2 \end{array} \right.$$

درنهایت با استفاده از سطوح تصمیم گیری می توانیم تشخیص دهیم که هر کدام از سمبل ها حاوی کدام دامنه بوده است. سپس

مقادیر آشکار شده را در یک بردار به نام `detected_symbols` ذخیره می کنیم.

در قسمت پایانی همه ۱۱ کانال `detected_symbols` را با سیگنال مادل شده در فرستنده مقایسه کرده و تعداد سمبل های خطا را بدست می آوریم. سپس تعداد خطای هر کانال را بر تعداد کل سمبل ها تقسیم می کنیم تا احتمال خطای آشکار سازی را بدست آوریم.

تصویر ۳، نمودار احتمال خطای آشکار سازی هر دو گیرنده MAP و ML را در مواجهه با ۱۱ نویز با انرژی های مختلف نشان می دهد.



تصویر ۳: نمودار احتمال خطای آشکارسازی گیرنده های MAP و ML برحسب SNR

مقایسه دو گیرنده:

- همانطور که مشاهده می شود، حساسیت به SNR در گیرنده MAP به گیرنده ML کمتر است.
- عملکرد گیرنده MAP زمانی که SNR کم است (توان نویز بالاست) بهتر از عملکرد گیرنده ML است.
- در صورتی که نویز کانال توان کمی داشته باشد (SNR زیاد باشد)، با توجه به اینکه نمونه برداری (همزمان سازی) درست بوده، خطای هر دو نوع گیرنده تقریباً مشابه با هم خواهد شد.