به نام خدا

جداسازی کور منابع (دکتر اخوان)

تمرین سری **هفتم**

نیم‌سال دوم ۱۴۰۳-۱۴۰۲

آرمان مجیدی – ۸۱۰۱۰۰۲۰۵

بخش اول

مقدمه

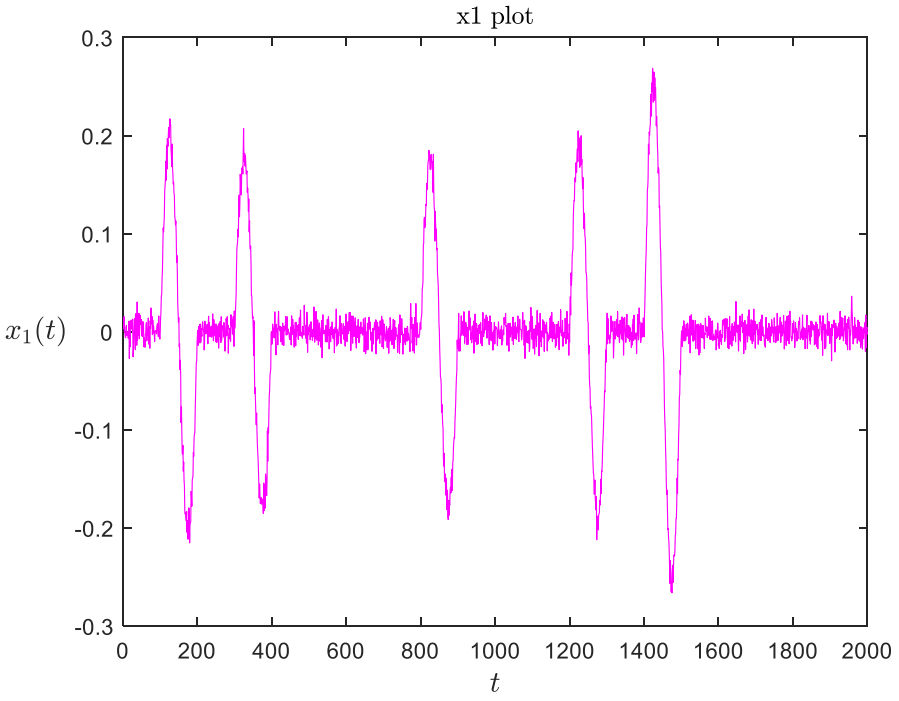
در این بخش به دنبال این هستیم که ابتدا روش Single-Channel Sparse Blind Deconvolution را روی داده نویزی x_1 و سپس روی داده x_2 پیاده سازی کنیم و شکل موج و سری زمانی را طبق این الگوریتم استخراج نماییم.

سپس روی داده دو کاناله بدون نویز X، الگوریتم Multi-Channel Sparse Blind Deconvolution را پیاده سازی می کنیم و دو شکل موج و سری زمانی که این داده ها را به وجود آورده اند را به دست می آوریم.

در انتها، داده x_1 را به حوزه فرکانس می بریم و با پیاده سازی الگوریتمی مناسب، مجددا مسئله قسمت ۱ را حل می کنیم.

قسمت ۱

در این قسمت ابتدا نمودار x_1 را رسم می کنیم تا به صورت چشمی آن را بررسی نماییم. تصویر ۱.۱ نمودار x_1 را نشان می دهد.



تصویر ۱.۱: نمودار x_1 در حوزه زمان

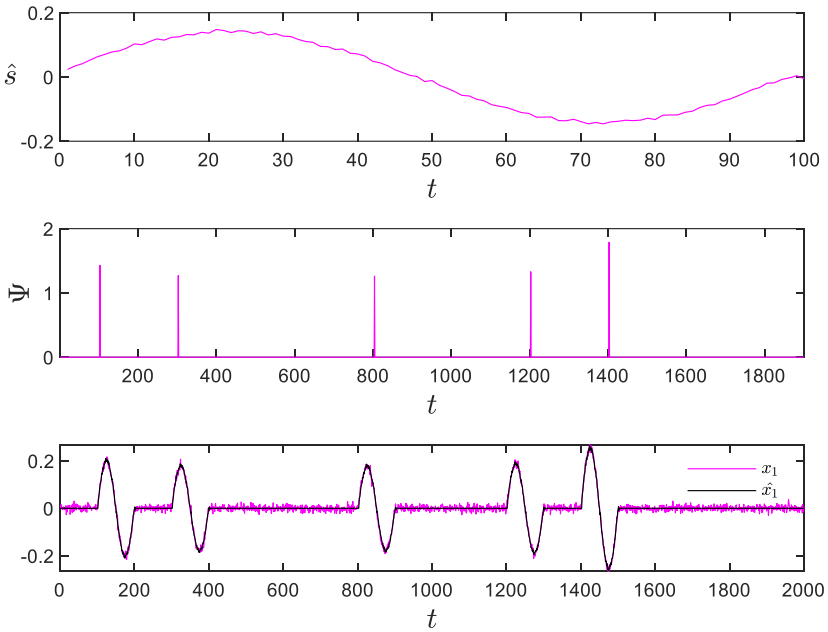
همانطور که مشاهده می شود، ما 5 اسپایک در نمودار x_1 داریم. برای شناسایی شکل موج این اسپایک ها و همچنین قطار ضربه ای در این شکل موج کانوالو می شود، از `Single-Channel Sparse Blind Deconvolution` استفاده می کنیم. این روش به طور خلاصه در پایین توضیح داده شده است:

$$x(t) = s(t) * \Psi(t) \quad ; \quad \{\Psi(t): comb\ function\} \quad , \quad t = 1,2,\dots,T$$

$$\{\hat{s}, \hat{\Psi}\} = \begin{cases} argmin \sum_{t=0}^{T} \left(x(t) - s(t) * \Psi(t)\right)^2 \\ s.t. \begin{cases} \Psi(t) = \sum_k \alpha_k\, \delta(t - t_k) \\ \|s\|_2 = 1 \\ |\tau_i - \tau_j| > L \\ \alpha_k > 0 \end{cases} \end{cases}$$

$$Alternation\ Minimization: \begin{cases} 1)\ \Psi(t)\ is\ fixed \rightarrow \hat{s}^T = \dfrac{\alpha^T Y}{\alpha^T \alpha} \\ 2)\ s(t)\ is\ fixed \rightarrow z = s\, b^T \end{cases}$$

حال این الگوریتم را پیاده سازی می کنیم. نمودار بدست آمده از این الگوریتم در تصویر ۱.۲ نشان داده شده است.

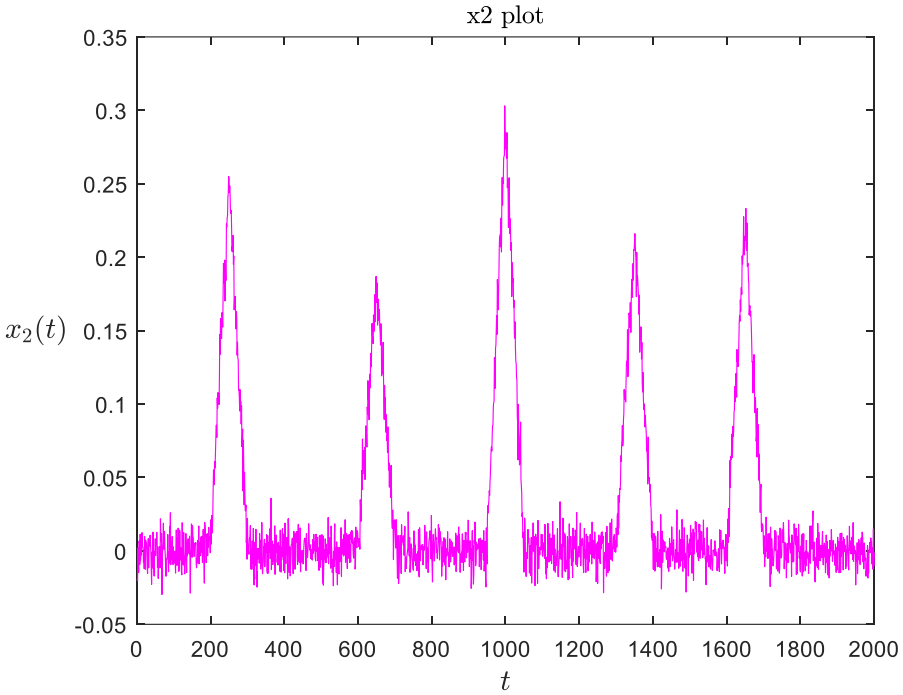


تصویر ۱.۲: نمودار های x_1 بدست آمده از الگوریتم `Single-Channel Sparse Blind Deconvolution`

با بررسی نمودار سوم در تصویر ۱.۲ می بینیم که بین دو نمودار رسم شده، تفاوت هایی وجود دارد. این تفاوت ها به علت حضور نویز در داده است.

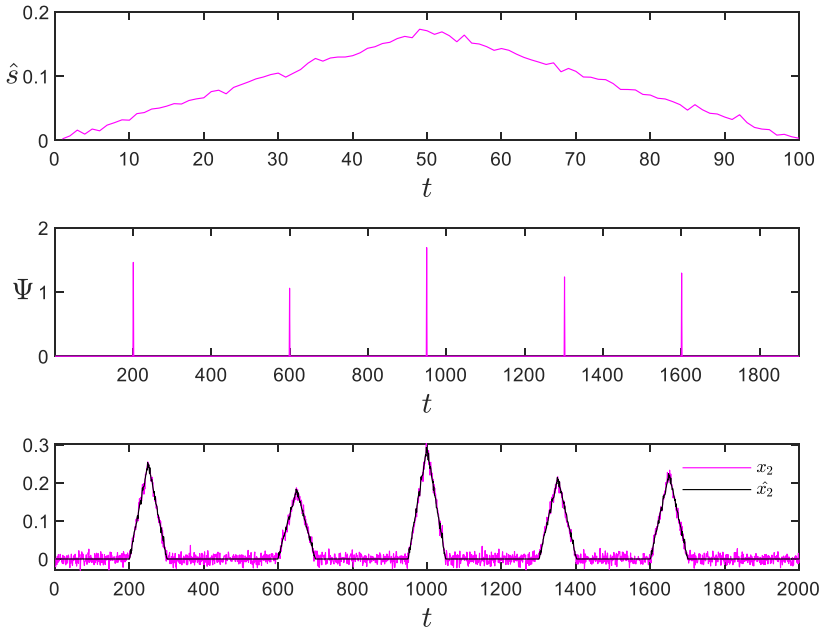
قسمت ۲

ابتدا داده x_2 را رسم می کنیم تا به صورت چشمی آن را مشاهده نماییم. این نمودار در تصویر ۱.۳ نشان داده شده است.



تصویر ۱.۳: نمودار x_2 در حوزه زمان

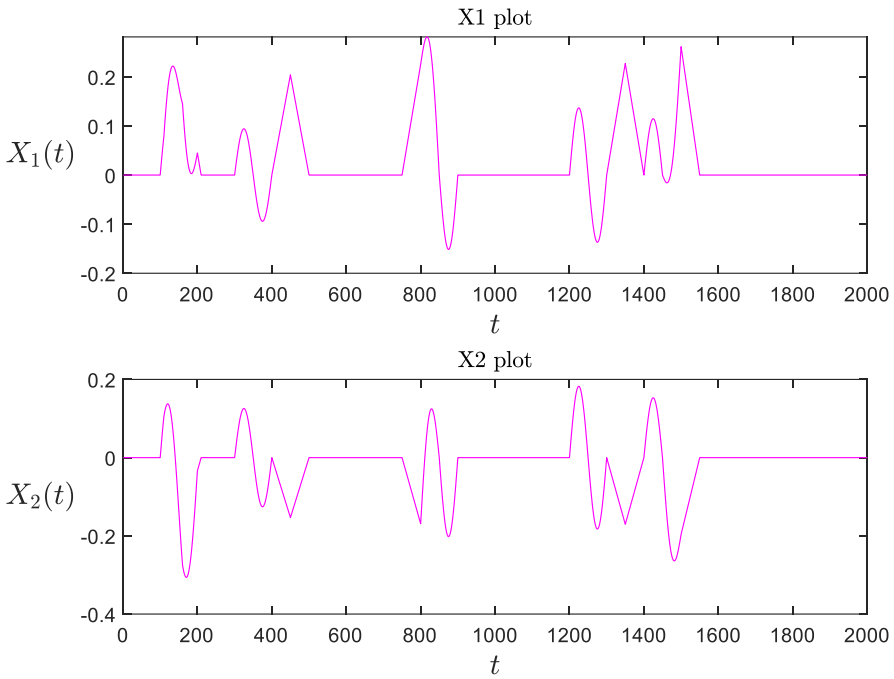
مجددا الگوریتم `Single-Channel Sparse Blind Deconvolution` را روی داده x_2 اعمال می کنیم. نتیجه اجرا در تصویر ۱.۴ آمده است.



تصویر ۱.۴: نمودار های x_2 بدست آمده از الگوریتم `Single-Channel Sparse Blind Deconvolution`

قسمت ۳

در این قسمت، یک داده دو کاناله X داریم. نمودار این کانال در تصویر ۱.۵ آمده است:



تصویر ۱.۵: نمودار X در حوزه زمان

برای حل این مسئله، باید از الگوریتم `Multi-Channel Sparse Blind Deconvolution` استفاده کنیم. این الگوریتم به صورت خلاصه در زیر آمده است:

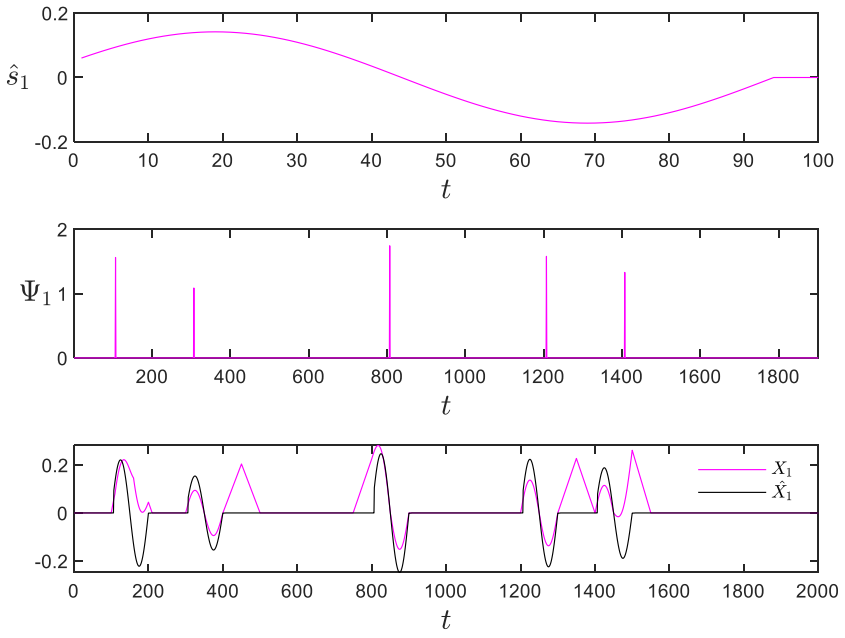
$$x(t) = s(t) * \Psi(t) \rightarrow \begin{cases} X = \sum_{n=1}^{N} a_n\, b_n^T = AB \\ {b_n}^T = s_n * \Psi_n \end{cases} \rightarrow Target\ function: \|X - Model\|_F{}^2$$

$$Alternation\ Minimization: \begin{cases} 1)\ A\ is\ fixed \rightarrow \hat{B} = A^{\dagger}X \rightarrow Single\ Ch.B.D\ on\ each\ row\ of\ B \\ 2)\ B\ is\ fixed \rightarrow \hat{A} = XB^{\dagger} \rightarrow Normalize\ Columns\ of\ A \end{cases}$$

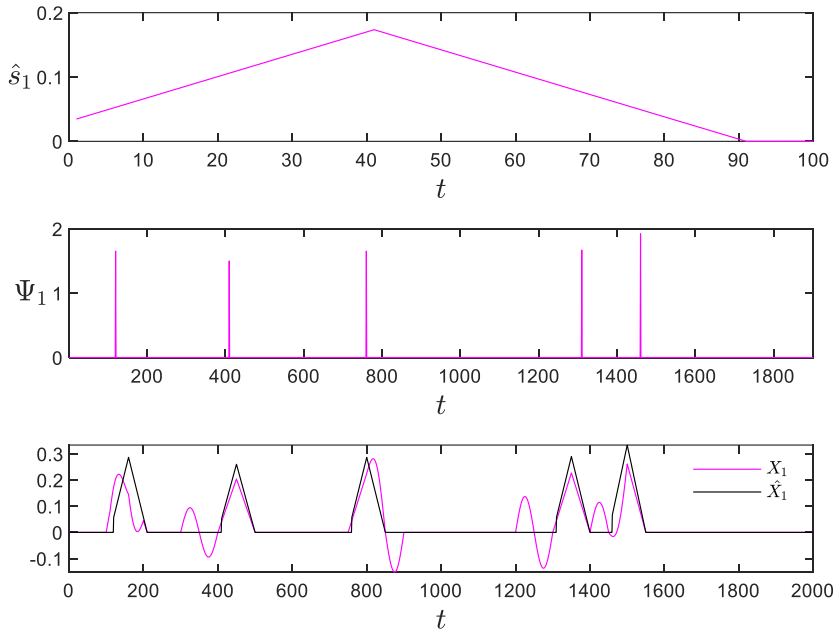
حال اگر این الگوریتم را پیاده سازی کنیم، خواهیم داشت:

$$A = \begin{bmatrix} 0.6127 & -0.7903 \\ 0.8125 & 0.5830 \end{bmatrix}$$

همچنین، تصویر ۱.۶ و ۱.۷ دو شکل موج اسپایکی و سری زمانی را نشان می دهند.

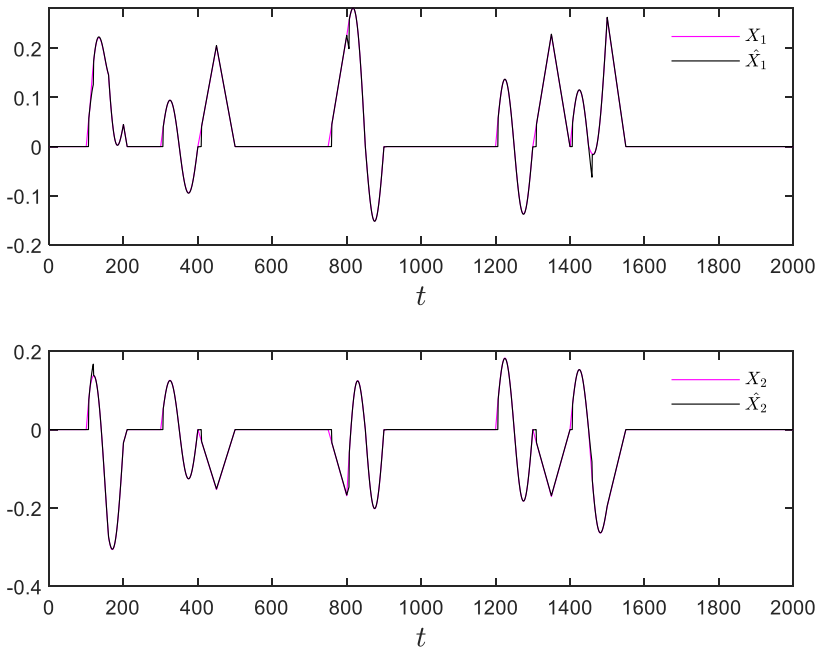


تصویر ۱.۶: شکل موج اسپایکی و سری زمانی موج اول



تصویر ۱.۷: شکل موج اسپایکی و سری زمانی موج دوم

حال، می دانیم که $X = AB$. حال، این دو ماتریس را در هم دیگر ضرب می کنیم تا ماتریس X_hat بدست آید؛ سپس این ماتریس را با ماتریس X مقایسه می کنیم. نمودار این مقایسه در تصویر ۱.۸ آمده است.



تصویر ۱.۸: مقایسه مشاهدات تخمین زده شده با مشاهدات داده شده

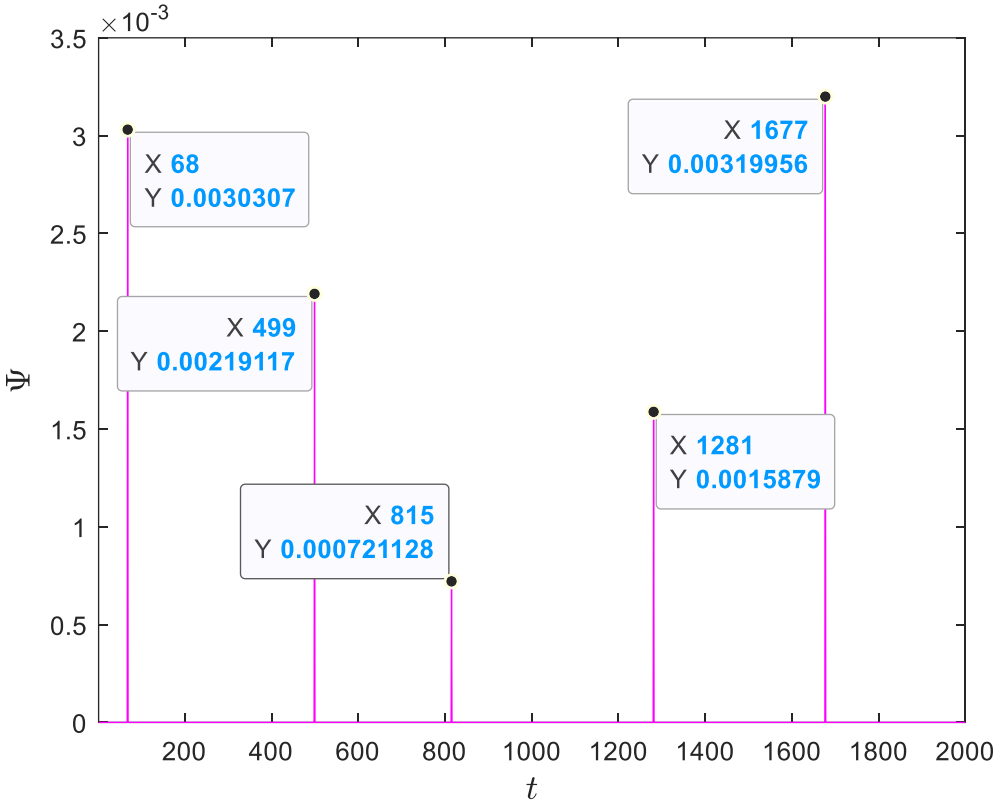
می بینیم که با اینکه داده ما بدون نویز بوده، اما مجددا سیگنال ما دچار اعوجاج شده است. این، اعوجاج، هم به دلیل گسسته بودن محیط MATLAB و هم به دلیل این است که کران کانوولوشن نامحدود بوده، در حالی که ما در اینجا این کران را محدود فرض کرده ایم.

قسمت ۴

برای این قسمت الگوریتمی را پیاده سازی می کنیم. این الگوریتم در پایین بیان شده است:

- ابتدا بردار S_f را به صورت تصادفی تولید می کنیم و آن را نرمالیزه می کنیم. سپس در یک حلقه for که برای تکرار اجرا کد می باشد، الگوریتم Alternation Minimization را اعمال می کنیم.
- در **گام اول**، فرض می کنیم که ما بردار S_f را داریم. (ابعاد این ماتریس برابر با 1×2000 می باشد). همچنین می دانیم هنگامی که در حوزه زمان کانوولوشن داریم، متناظر با آن در حوزه فرکانس ضرب داریم. پس ماتریس مشاهدات خود را تقسیم بر ماتریس S_f می کنیم تا بردار comb_f بدست آید.
- حال با فوریه معکوس گرفتن از comb_f به comb_t می رسیم. حال با زدن یک for روی تعداد رخداد، بیشترین مقادیر در حوزه زمان را پیدا می کنیم؛ سپس با اعمال شروط روی تابع قطار ضربه، آن را به دست می آوریم.
- در **گام دوم**، ما فرض می کنیم که بردار comb_f(comb_t) را داریم. با تقسیم مشاهدات بر این بردار و اعمال شرط 1 بودن نُرم آن، ماتریس S_f جدید را بدست می آوریم.

این روش، دقت مناسبی به ما نمی دهد. تصویر ۱.۹، نمودار بردار comb بدست آمده در یک بار اجرا را نشان می دهد.



تصویر ۱.۹: نمودار بردار comb بدست آمده از الگوریتم حوزه فرکانس

با بررسی این قسمت با قسمت ۱، می بینیم که دقت چندان خوبی در این الگوریتم نداریم. همچنین، برای ضرایب نیز می توان دید که به علت fft گرفتن متوالی، ضرایب ما بسیار کوچک می شوند.