ASP Final Project: Adaptive Beamforming

R11942085 comm1 許宸睿

1. Details of the beamforming

At first, we make some assumptions:

(a) There is only one source
(b) The DOA of the source is θ_1,
(c) Uniform linear array with N isotropic antennas and inter-element spacing $\lambda/2$,
(d) The source waveform is $s_1(t) = Ae^{j2\pi}$, which the complex amplitude A satisfies $E[A] = 0$ and $E[|A|^2] = \sigma_1^2$,
(e) The noise term satisfies $E[\mathbf{n}(t)] = \mathbf{0}$ and $E[\mathbf{n}(t)\mathbf{n}^H(t)] = \sigma_n^2\boldsymbol{I}$.

(1) The beamformer with uniform weights

At this model, we assume that the DOA θ_1=0, so every antenna will receive source signals with the same phase. The equation of the data model is denoted as:

$$\begin{aligned} x(t) &= \mathbf{a}(\theta_1)s_1(t) + \boldsymbol{n}(t) \\ &= [1\ \ 1\ \ 1 \dots 1]^T Ae^{j2\pi ft} + \boldsymbol{n}(t) = \mathbf{1}\left(Ae^{j2\pi ft}\right) + \boldsymbol{n}(t) \end{aligned}$$

Where $\mathbf{a}(\theta)$ is known as steering vector, written as:

$$\mathbf{a}(\theta) = \left[e^{j2pi\left(\frac{d_0}{\lambda}\sin(\theta)\right)}\ \ e^{j2pi\left(\frac{d_1}{\lambda}\sin(\theta)\right)} \dots e^{j2pi\left(\frac{d_{N-1}}{\lambda}\sin(\theta)\right)}\right]^T$$

We could apply uniform weightings $\mathbf{w} = \mathbf{1}/N$, so the beamformer output is:

$$\begin{aligned} y(t) &= \mathbf{w}^H x(t) \\ &= \left(\frac{1}{N}\mathbf{1}^H\mathbf{1}\right)\left(Ae^{j2\pi ft}\right) + \left(\frac{1}{N}\mathbf{1}^H\boldsymbol{n}(t)\right) \\ &= Ae^{j2\pi ft} + \frac{1}{N}\mathbf{1}^H\boldsymbol{n}(t) \end{aligned}$$

By doing this, we could recover the source signal while the noise is decreased by a factor 1/N. The SNR then becomes to:

$$\mathrm{SNR} = \frac{E\left[\left|Ae^{j2\pi ft}\right|^2\right]}{\frac{\mathbf{1}^H E[\boldsymbol{n}(t)\boldsymbol{n}^H(t)]\mathbf{1}}{N^2}} = \frac{N\sigma_1^2}{\sigma_n^2}$$

This model will get the best performance when the source $\mathrm{DOA} = 0$. If $\mathrm{DOA} \neq 0$, the SNR will decrease. For example, let's set $\mathrm{DOA} = \theta$, then the input is rewritten as:

$$x(t) = \mathbf{a}(\theta)s_1(t) + \boldsymbol{n}(t)$$

The output becomes to:

$$\begin{aligned} y(t) &= \mathbf{w}^H\big(\mathbf{a}(\theta)s_1(t) + \boldsymbol{n}(t)\big) \\ &= \mathbf{w}^H\mathbf{a}(\theta)s_1(t) + \mathbf{w}^H\boldsymbol{n}(t) \end{aligned}$$

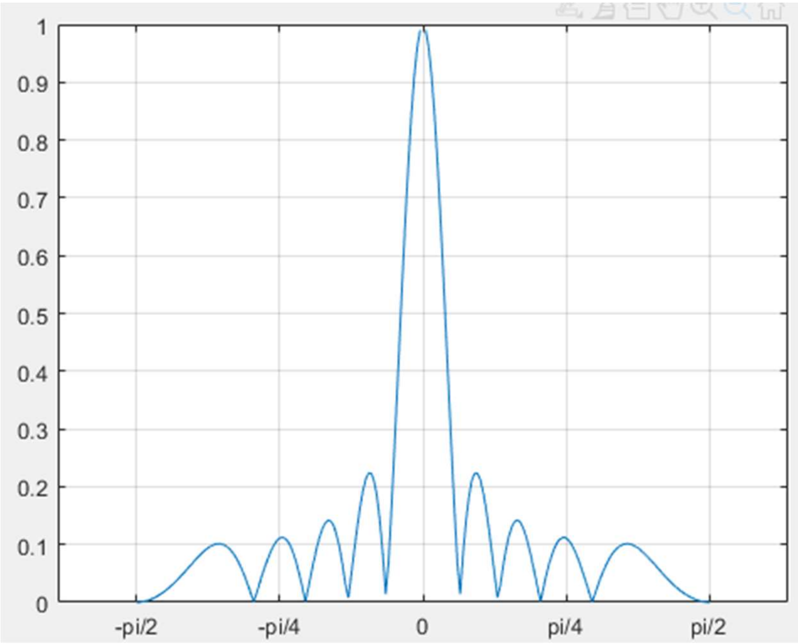
$$\text{Let } B(\theta) = \mathbf{w}^H\mathbf{a}(\theta),$$

$$\text{SNR} = \frac{E[|B(\theta)s_1(t)|^2]}{E[|\mathbf{w}^H\boldsymbol{n}(t)|^2]} = |B(\theta)|^2 * \frac{N\sigma_1^2}{\sigma_n^2}$$

$B(\theta)$ is an important factor to SNR. Below show the expression of $B(\theta)$.

$$\begin{aligned} B(\theta) &= \frac{1}{N}\mathbf{1}^H\mathbf{a}(\theta) \\ &= \frac{1}{N}\Sigma_{n=0}^{N-1} e^{j\pi \sin(\theta) n} \\ &= e^{j\frac{N-1}{2}\pi\sin(\theta)} \; \frac{1}{N} \; \frac{\sin(\frac{N\pi\sin\theta}{2})}{\sin(\frac{\pi\sin\theta}{2})} \end{aligned}$$

From the figure below, we could observe that $|B(\theta)|$ get its maximum at rad =0, and at some specified angle, $B(\theta) = 0$ though.



Figure(a): plot of $|B(\theta)|$ with radius changing. Setting N = 10

(2) The beamformer with array steering

Since uniform weightings only works well at degree=0, we want to derive a method for specified angle of DOA. Therefore, at this model, we assume that the DOA $\theta_1 = \theta$. And we adjust the weight vector so that the signal waveform is kept when it near $\theta = \theta_s$, and is rejected when it is far from $\theta = \theta_s$. The new weight vector is denoted as:

$$\mathbf{w} = \frac{1}{\mathrm{N}}\boldsymbol{a}(\theta_s)$$

So we could rewrite $\mathrm{B}(\theta)$ as:

$$\mathrm{B}(\theta) = \mathbf{w}^H\mathbf{a}(\theta)$$

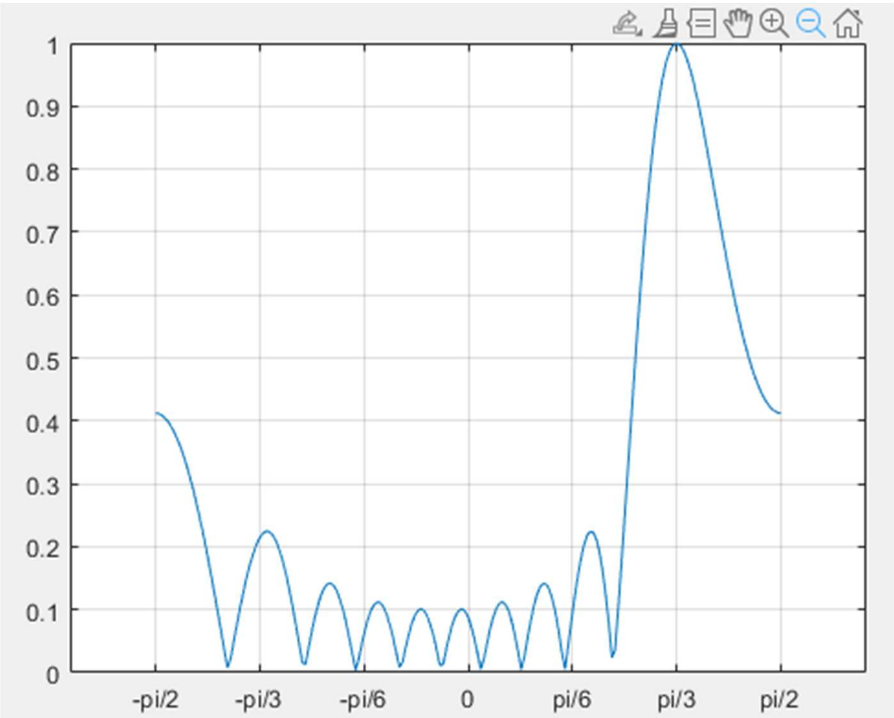
$$= \frac{1}{\mathrm{N}}\boldsymbol{a}(\theta_s)^H\mathbf{a}(\theta)$$

$$= \frac{1}{N}\Sigma_{n=0}^{N-1} e^{j\pi\,(-\sin(\theta_s)+\sin(\theta))n}$$

$$= e^{j\frac{N-1}{2}\pi\,(\sin(\theta)-\sin(\theta_s))}\ \frac{1}{N}\ \frac{\sin(\frac{N\pi\,(\sin(\theta)-\sin(\theta_s))}{2})}{\sin(\frac{\pi\,(\sin(\theta)-\sin(\theta_s))}{2})}$$

$$\mathrm{SNR} = \frac{E[|B(\theta)s_1(t)|^2]}{E[|\mathbf{w}^H\boldsymbol{n}(t)|^2]} = |\mathrm{B}(\theta)|^2 * \frac{N\sigma_1^2}{\sigma_n^2}$$

$|\mathrm{B}(\theta)|$ is still an important factor of SNR, and from the figure below, we can observe that $|\mathrm{B}(\theta)|$ will get its maximum at $\theta = \theta_s$. Therefore, the beamformer of with array steering performs better when the source signal comes from the same direction $\theta \neq 0$. The difficulty is how to know the DOA.



Figure(b): Plot of $|\mathrm{B}(\theta)|$ with θ changing. Set N =10 and $\theta_s = \frac{\pi}{3}$

(3) The MVDR beamformer

Set DOA of source $\theta_1 = \theta_s$

Since the output is denoted as:

$$y(t) = \mathbf{w}^H x(t) = \mathbf{w}^H \mathbf{a}(\theta_s) s_1(t) + \mathbf{w}^H \boldsymbol{n}(t)$$

we could have minimum mean square error with the distortionless constraint:

$$\min_{\boldsymbol{w}} E[|\mathbf{w}^H \boldsymbol{n}(t)|^2] \quad subject \ \ to \ \ \mathbf{w}^H \mathbf{a}(\theta_s) = 1$$

$$E[|\mathbf{w}^H \boldsymbol{n}(t)|^2] = \ \ \mathbf{w}^H E[\boldsymbol{n}(t)\boldsymbol{n}^H(t)]\boldsymbol{w}$$

But the correlation of noise $E[\boldsymbol{n}(t)\boldsymbol{n}^H(t)]$ is unavailable. We can derive it from the power of output with the constraint $\mathbf{w}^H \mathbf{a}(\theta_s) = 1$:

$$E[|y(t)|^2] = E[|\mathbf{w}^H x(t)|^2]$$

$$= E[|\mathbf{w}^H \mathbf{a}(\theta_s) s_1(t) + \mathbf{w}^H \boldsymbol{n}(t)|^2]$$

$$= E\left[\left(s_1(t) + \mathbf{w}^H \boldsymbol{n}(t)\right)\left(s_1(t) + \mathbf{w}^H \boldsymbol{n}(t)\right)^H\right]$$

$$= \mathbf{w}^H E[\boldsymbol{n}(t)\boldsymbol{n}^H(t)]\boldsymbol{w} + \mathbf{w}^H E[\boldsymbol{n}(t) s_1(t)^*] + E[s_1(t)\boldsymbol{n}^H(t)]\boldsymbol{w} + E[|s_1(t)|^2]$$

If the correlation of noise and desired signal is **0**, then $E[|\mathbf{w}^H \boldsymbol{n}(t)|^2]$ shares the same weight vector with $E[|\mathbf{w}^H \boldsymbol{x}(t)|^2]$. Thus, the optimization problem becomes to:

$$\boldsymbol{w}_{MVDR} = \arg \min_{\boldsymbol{w}} E[|y(t)|^2] \quad subject \ \ to \ \ \mathbf{w}^H \mathbf{a}(\theta_s) = 1$$

$$where \ \ y(t) = \ \ \mathbf{w}^H \boldsymbol{x}(t)$$

Assuming that $x(t)$ is WSS, so the correlation matrix $\mathbf{R} = \mathrm{E}[\mathbf{x}(t)\mathbf{x}^H(t)]$.

Solving the optimization function, we could get:

$$\boldsymbol{w}_{MVDR} = \frac{\mathbf{R}^{-1}\boldsymbol{a}(\theta_s)}{\boldsymbol{a}^H(\theta_s)\boldsymbol{R}^{-1}\boldsymbol{a}(\theta_s)}$$

(4) The LCMV beamformer

In this model, there is another signal called interference, which is we don't want. Set the DOA of interference as θ_i.

Similar to MVDR beamformer, LCMV beamformer is an optimization but with linear constraints. The optimization problem is:

$$\boldsymbol{w}_{MVDR} = \arg \min_{\boldsymbol{w}} E[|y(t)|^2] \quad subject \ \ to \ \ \boldsymbol{C}^H \boldsymbol{w} = \boldsymbol{g}$$

If $\boldsymbol{C}^H = \boldsymbol{a}(\theta_s)$ & $\boldsymbol{g} = 1$, then LCMV is equivalent to MVDR. Since we want the source signal is same as coming, and get less from interference, $\boldsymbol{C}$ and $\boldsymbol{g}$ is set as the following:

$$\begin{cases}\mathbf{w}^H\mathbf{a}(\theta_s) = 1 \\ \mathbf{w}^H\mathbf{a}(\theta_i) = g_i\end{cases} \rightarrow [\mathbf{a}(\theta_s)\ \ \mathbf{a}(\theta_i)]^H\boldsymbol{w} = [1\ \ g_i^*]^T$$

$$\text{where } \mathrm{g_i}\ \ is\ \ an\ \ extreme\ \ small\ \ number.$$

Solving the optimization problem, we get:

$$\mathbf{w}_{\mathrm{LCMV}} = \boldsymbol{R}^{-1}\boldsymbol{C}(\boldsymbol{C}^H\boldsymbol{R}^{-1}\boldsymbol{C})^{-1}\boldsymbol{g}$$

$$\text{where } \mathbf{R} = \mathrm{E}[\mathbf{x}(\mathrm{t})\mathbf{x}^{\mathrm{H}}(\mathrm{t})]$$

When the collected data contains desired signal and interference, and when their directions are different, we could apply LCMV.

2. Design an algorithm to denoise $\widetilde{\boldsymbol{\theta}_s}(\boldsymbol{t})$ and $\widetilde{\boldsymbol{\theta}_\iota}(\boldsymbol{t})$ over the time index t. The denoised results are denoted by $\widehat{\boldsymbol{\theta}_s}(\boldsymbol{t})$ and $\widehat{\boldsymbol{\theta}_\iota}(\boldsymbol{t})$.

Details of Algorithm:

Hilbert-Huang Transform, a way to decompose a signal into intrinsic mode functions along the trends, is designed to work well for data that is non-stationary and non-linear. The methods of HHT is (1) to draw the max envelope and min envelope of signals by going through all local maxima and minima, (2) to get the mean by (max_envelope + min_envelope) /2, (3) to extract the signal from origin one, (4) then to redo until no local maxima and minima existing.

Since we don't know the actual signal, assume that it is non-stationary. Also, assume that the magnitude of noise is smaller than the source. Using HHT to decompose the signal into few components such as noise and trends. The noise is Gaussian white, so the mean of the noise is near to zero. Therefore, set a threshold small enough, and go through all components from the first to the end. If the mean of one component is larger than threshold, stop the loop and sum the rest of components.

Pseudo code:

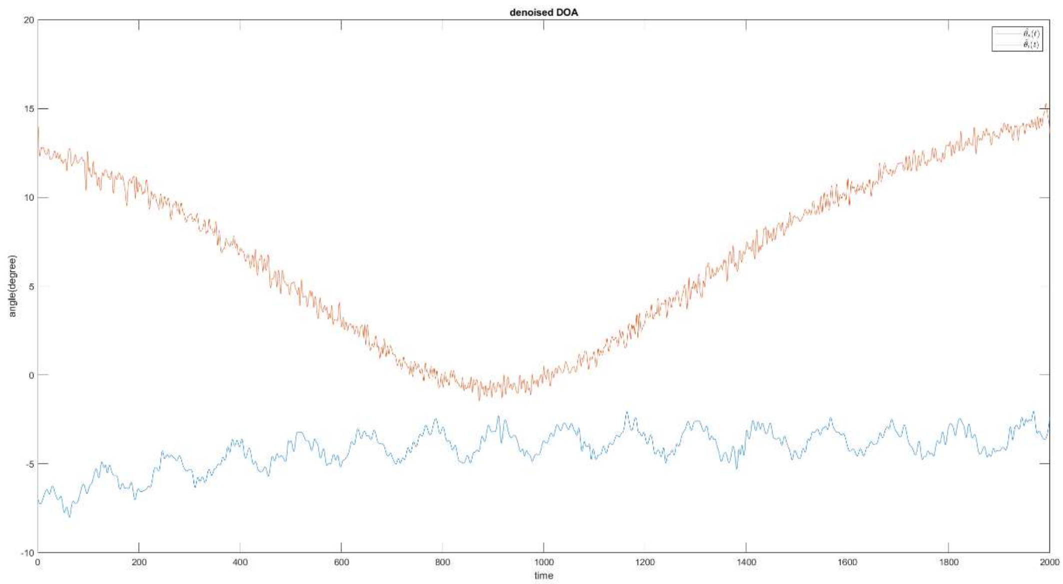
```
1)  IMF_sets = hht(signal, threshold_hht)
2)  for i =1:length(i) do
3)          if mean(IMF_set(i)) > threshold
4)                  index_start = i
```

5) break;
6) end for
7) denoised_signal = sum(IMF_sets(index_start:end));

Advantage:

Since not all the signals are sinusoid, and the magnitude of signals will decrease or increase with time, using Fourier Transform and selecting the magnitude over threshold will destroy some details. HHT will reserve the details, also could find out the trends. What's more, it doesn't need to convert to frequency domain.

3. Estimated DOA



Figure(c): denoised DOA

Describe:

1) $\widehat{\theta_\iota}$: The trend looks like a parabola, and there are some sinusoid signals added.
2) $\widehat{\theta_s}$: The trend looks like a log scale and a low frequency sinusoid signal dominated. Also, some high frequency low-magnitude sinusoid signals exist.

4. Design a beamformer that extracts the source signal in (1).

Details of the beamformer:

I applied LCMV model for extracting beamformer. The constraints **g** is set to $[1 \;\; 0.002]^{\mathrm{T}}$. And since LCMV model needs to calculate $\mathbf{R}^{-1}$ every iteration, I take the method in RLS as reference. Thus, the speed is much quicker than before.
When calculating $\mathbf{R}^{-1}$, the mathematical equation takes expect value of inputs, like $\mathrm{E}[\mathbf{x}(\mathrm{t})\mathbf{x}^{\mathrm{H}}(\mathrm{t})]$. However, a deterministic value is given. Therefore, we calculating $\widehat{\boldsymbol{R}}^{-1}$ by $\mathbf{x}(\mathrm{t})\mathbf{x}^{\mathrm{H}}(\mathrm{t})$.
The hyperparameters δ , $\lambda, \mathrm{g_i}$ are set to 0.01, 0.995 and 0.002.

Pseudo code:

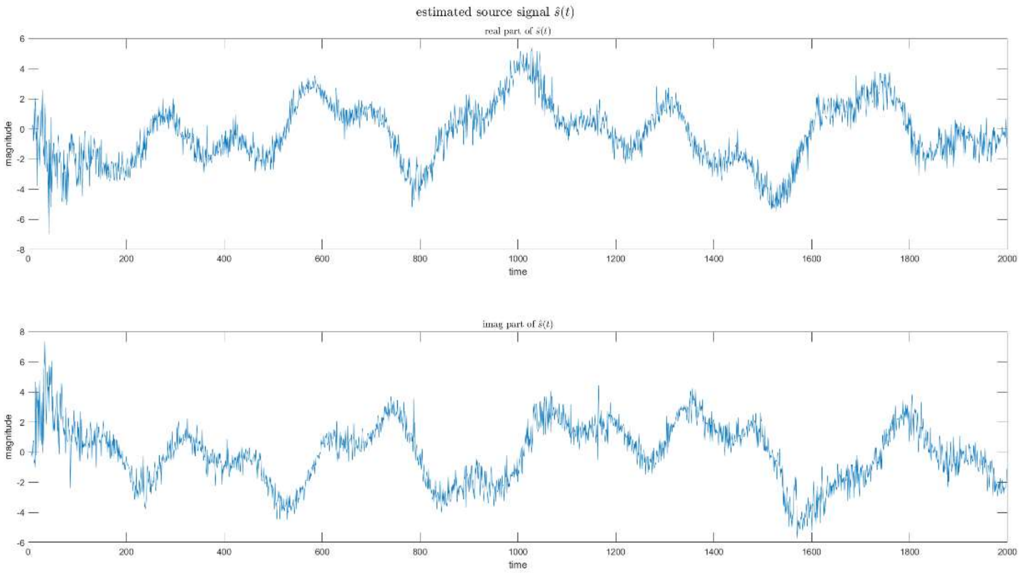
Required: $\delta > 0 \;\; \& \;\; 0 < \lambda \leq 1$

1) **P**(0) = $\delta^{-1}\boldsymbol{I}$
2) $\boldsymbol{g} = [1 \;\; 0.002]^{\mathrm{T}}$
3) for all n = 1,2,3… do
4) $\quad \mathbf{k}(\mathrm{n}) = \frac{\lambda^{-1}\boldsymbol{P}(n-1)\boldsymbol{x}(n)}{1+\lambda^{-1}\boldsymbol{x}^H\boldsymbol{P}(n-1)\boldsymbol{x}(n)}$,
5) $\quad \mathbf{P}(\mathrm{n}) = \lambda^{-1}\boldsymbol{P}(n-1) - \lambda^{-1}\boldsymbol{k}(n)\boldsymbol{x}^H(n)\boldsymbol{P}(n-1)$,
6) $\quad \mathbf{C} = [\mathbf{a}(\theta_s(t)),\;\; \mathbf{a}(\theta_i(t))]$,
7) $\quad \mathbf{w}_{\mathrm{LCMV}} = \mathbf{PC}(\mathbf{C}^{\mathrm{H}}\mathbf{PC})^{-\mathbf{1}}\boldsymbol{g}$,
8) end for

Advantages:

LCMV will decrease the interference signal by the constraints
$\begin{cases} \mathbf{w}^H\mathbf{a}(\theta_s) = 1 \\ \mathbf{w}^H\mathbf{a}(\theta_i) = g_i \end{cases} \rightarrow [\mathbf{a}(\theta_s) \;\; \mathbf{a}(\theta_i)]^H\boldsymbol{w} = [1 \;\; g_i^*]^T$, which is much better than MVDR and Uniform. And it could update the weight adaptively.

5. Plot the source signal $\hat{s}(t)$



Figure(d): estimated source signal