Window Beamformer for Sparse Concentric Circular Array

Rajib Sharma (Postdoc: 2018-2020)
Israel Cohen (Postdoc supervisor)
Baruch Berdugo (Postdoc co-supervisor)

Faculty of Electrical Engineering
Technion, Israel Institute of Technology, Haifa

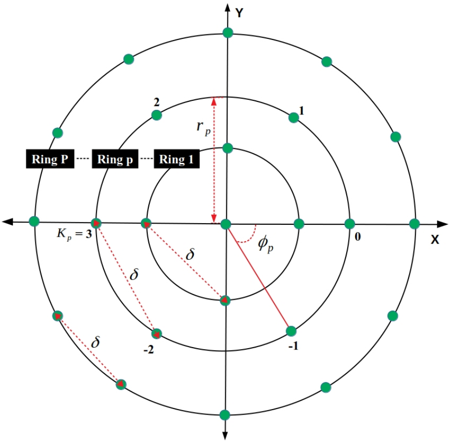
Topics of this presentation

1. **Beamforming and Concentric Circular Array (CCA).**
2. **Design of sparse CCA.**
   - **Maximizing broadband Directivity Factor.**
3. **Frequency-invariant or constant-beamwidth beamforming.**
   - **Variation of beampattern w.r.t. frequency. Misestimation of DOA ?**
   - **Limitations of analytical solutions.**
4. **Controlling both elevation and azimuth beamwidths.**
   - **Sensor weighting.**
   - **Modified Gradient Descent Algorithm.**
   - **Elevation beamwidth, Azimuth beamwidth, Directivity Factor, and White Noise Gain.**
   - **Gaussian Window (GW) vs. Kaiser Window (KW) vs. other beamformers.**

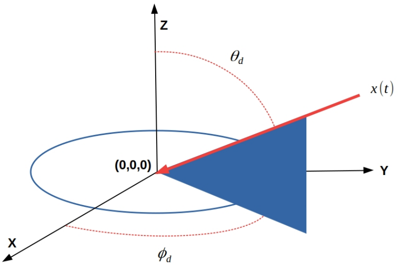
References :

- **GW:** R. Sharma, I. Cohen, and B. Berdugo, "Window beamformer for sparse concentric circular array," accepted in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Canada, 2021.
- **KW:** R. Sharma, I. Cohen, and B. Berdugo, "Controlling Elevation and Azimuth Beamwidths with Concentric Circular Microphone Arrays," accepted in IEEE Transactions on Audio, Speech and Language Processing, 2021.
- **Method-1:** Yixin Yang, Chao Sun, and Chunru Wan, "Theoretical and experimental studies on broadband constant beamwidth beamforming for circular arrays," in Oceans 2003. Celebrating the Past... Teaming Toward the Future (IEEE Cat. No. 03CH37492). IEEE, 2003, vol. 3, pp. 1647-1653.
- **Method-2:** Gongping Huang, Jingdong Chen, and Jacob Benesty, "Insights into frequency-invariant beamforming with concentric circular microphone arrays," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 26, no. 12, pp. 2305-2318, 2018.

Concentric Circular Array (CCA)



(a) Top view of a CCA.



(b) Circular co-ordinate system.

Beamforming on a CCA

$$y_{p,m_p}(t) = x_{p,m_p}(t) + v_{p,m_p}(t),\ p \in [1,\ P],\ m_p \in [-K_p + 1,\ K_p],$$
$$x_{p,m_p}(t) = x(t - \tau_{p,m_p}),\ \boxed{\tau_{p,m_p} = -F_s \frac{r_p}{c} \sin\theta_{\mathrm{d}} \cos(\phi_{\mathrm{d}} - \phi_{p,m_p})},$$
$$M = \sum_{p=1}^{P} M_p = \sum_{p=1}^{P} = 2K_p + 1. \quad (1)$$

$$Y_{p,m_p}(f) = X_{p,m_p}(f) + V_{p,m_p}(f),$$
$$X_{p,m_p}(f) = e^{j2\pi f \tau_{p,m_p}} X(f) = d_{p,m_p}(f, \theta_{\mathrm{d}}, \phi_{\mathrm{d}}) X(f). \quad (2)$$

$$\boxed{\mathbf{y}(f) = \mathbf{d}(f, \theta_{\mathrm{d}}, \phi_{\mathrm{d}}) X(f) + \mathbf{v}(f)},$$
$$Z(f) = \mathbf{h}^H(f)\mathbf{y}(f) \approx X(f). \quad (3)$$

$\mathbf{y}(f)$: data; $\mathbf{v}(f)$: interference and noise; $\mathbf{d}(f, \theta_{\mathrm{d}}, \phi_{\mathrm{d}})$: steering-vector.

$\mathbf{h}(f)$: M-dimensional filter; $Z(f)$: Output.

Beampattern, Directivity Factor, and White Noise Gain

Beampattern:

$$\mathcal{B}(f,\theta,\phi) = \mathbf{h}^H(f)\mathbf{d}(f,\theta,\phi). \tag{4}$$

Directivity Factor (DF):

$$\begin{aligned}\mathcal{D}(f) &= \frac{|\mathcal{B}(f,\theta_\mathrm{d},\phi_\mathrm{d})|^2}{\frac{1}{4\pi}\int_0^\pi\int_{-\pi}^\pi |\mathcal{B}(f,\theta,\phi)|^2 \sin\theta \; d\phi \; d\theta}, \\ &= \frac{|\mathbf{h}^H(f)\mathbf{d}(f,\theta_\mathrm{d},\phi_\mathrm{d})|^2}{\mathbf{h}^H(f)\Gamma(f)\mathbf{h}(f)}, \\ \Gamma(f)|_{i,j} &= \mathrm{sinc}\left[2\pi f l_{i,j}/c\right],\ 1 \leq \ i,j \ \leq M.\end{aligned} \tag{5}$$

In the above equation, $\Gamma(f)|_{i,j}$ represents the $(i,j)^{\mathrm{th}}$ position of the matrix, $\Gamma(f)$, and $l_{i,j}$ represents the Euclidean distance between the microphones corresponding to that position.

White Noise Gain (WNG):

$$\mathcal{W}(f) = \frac{|\mathbf{h}^H(f)\mathbf{d}(f,\theta_\mathrm{d},\phi_\mathrm{d})|^2}{\mathbf{h}^H(f)\mathbf{h}(f)}. \tag{6}$$

Design of sparse CCA

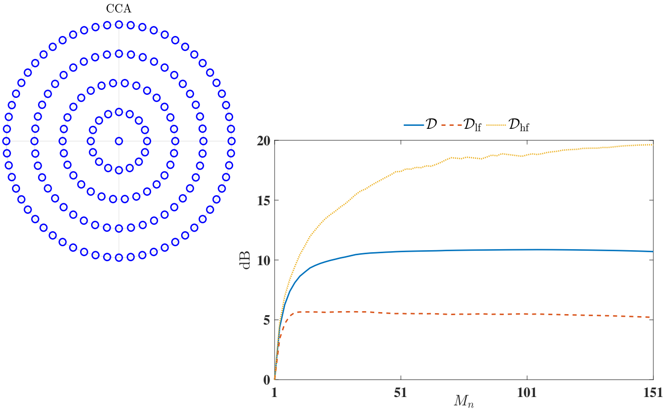


Figure: Sparse CCA design based on maximization of the broadband Directivity Factor.

$$\begin{aligned} \mathbf{h}(f) &= \frac{1}{M}\mathbf{d}(f, \theta_{\mathrm{d}}, \phi_{\mathrm{d}});\ \text{delay and sum (DS)} \\ &= \frac{1}{M}\mathbf{i}_M;\ \theta_{\mathrm{d}} = 0^o\ \text{(broadside)}. \end{aligned} \tag{7}$$

Frequency dependent characteristics

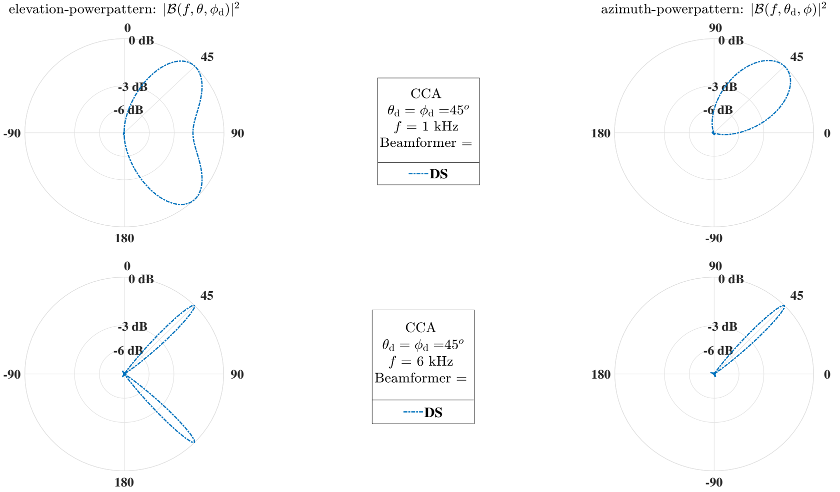


Figure: Powerpatterns (normalized) at 1 KHz (top row) and 6 kHz (bottom row) for the DS beamformer.

- What will happen if there is misestimation of the DOA ?

Limitations of current analytical solutions

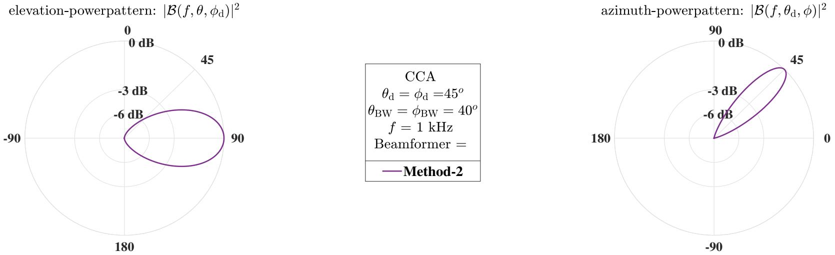
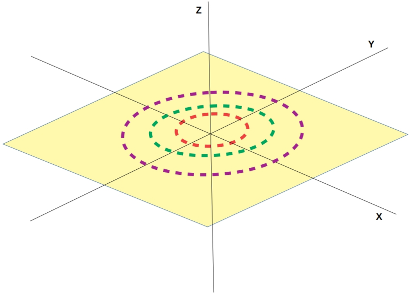


Figure: Powerpatterns (normalized) at 1 KHz frequency for the Method-2 beamformer.

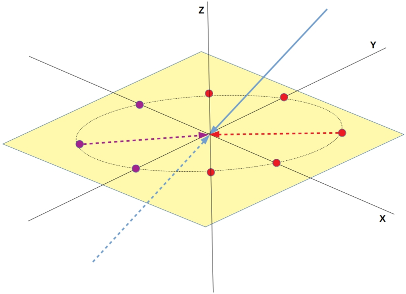
Known analytical solutions:

- Neglect the elevation beampattern, assume $\theta_{\mathrm{d}} = 90^o$.
- Require specific designs: constraints regarding the number of microphones in each ring.
- Do not consider the overall performance: DF and WNG.

Sensor weighting



(a) Weights assigned to each ring.



(b) Weights assigned to each microphone of a ring based on its alignment with the DOA.

- Characteristics vary w.r.t. ring radius: Each ring has its own weight.
- Characteristics vary w.r.t. the number of sensors in a ring: Symmetric window function provides weight based on alignment of the microphone with the DOA.
- **Modified gradient-descent algorithm finds the optimum weights: Optimization function switches between DF and beamwidth at each iteration as necessary.**

Performance - Beampatterns

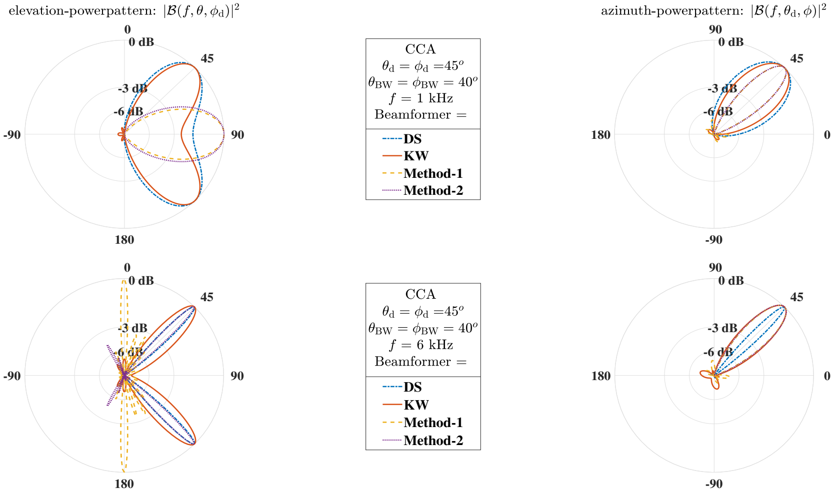


Figure: Powerpatterns (normalized) at 1 KHz (top row) and 6 kHz (bottom row) for the four beamformers.

- KW: Kaiser Window beamformer.

Performance: Beamwidths

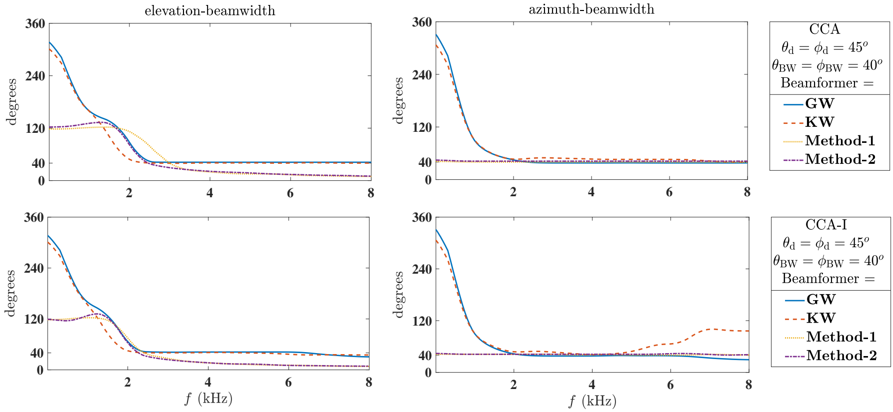


Figure: Elevation and azimuths beamwidths vs. frequency plots of the four beamformers applied on the CCA (top row) and the CCA-I (bottom row).

• GW: Gaussian Window beamformer. Refined version of KW. Computes distances differently. Optimization function also takes care that the DF never falls below a minimum value.

Performance: DF and WNG

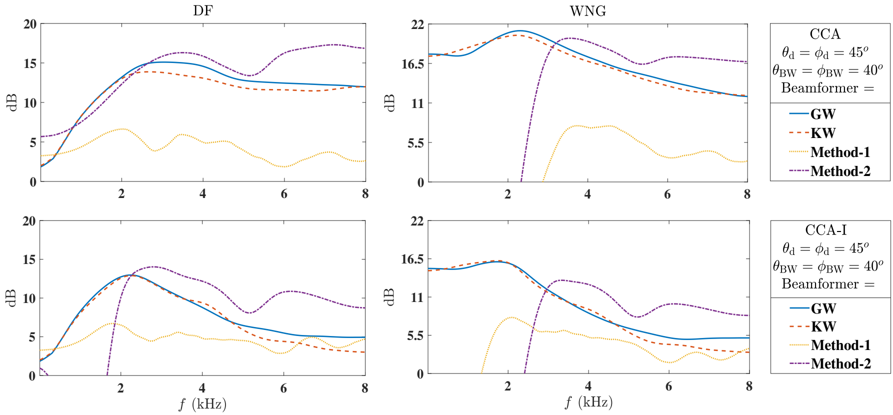


Figure: DF and WNG vs. frequency plots of the four beamformers applied on the CCA (top row) and the CCA-I (bottom row).

- GW: Gaussian Window beamformer. Refined version of KW. Computes distances differently. Optimization function also takes care of the DF (hence WNG).

End of the Presentation

Thank You !